

I. Capítulo 1: El Proyecto

1.1 Presentación del Proyecto

1.1.1 Título del Proyecto

El título del presente proyecto es: Mejoramiento en los Procesos de Registro en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija.

1.1.2 Carrera / Unidad

Ingeniería Informática.

1.1.3 Facultad

Ciencias y Tecnologías.

1.1.4 Duración del Proyecto

Ocho meses.

1.1.5 Área/línea de Investigación Priorizado

Tecnologías de la Información y Comunicación/Desarrollo de Sistemas y Software.

1.1.6 Responsable del Proyecto

Carrera de Ingeniería Informática – Taller III.

1.1.7 Entidad Asociada (s)

Caja Nacional de Salud – Administración Regional de Tarija.

1.2 Personal Vinculado al Proyecto

1.2.1 Director del Proyecto

Apellido Paterno: Yana	Apellido Materno: Cerezo	Nombre: Omar Elías	C.I.: 5796977 Tja.
Carrera: Ingeniería Informática		Facultad: Ciencias y Tecnología	
Telf.: Domicilio: Calle Suipacha entre Cochabamba y Felipe Echazu	Celular: 76185429	Correo electrónico: omar182_ya@hotmail.com	Firma:

Tabla 1. Director del Proyecto

1.2.2 Participantes del Equipo de Trabajo

Categoría	Nombres y Apellidos	Carrera/Profesión	C.I.	Firma
Director	Omar E. Yana Cerezo	Ingeniería Informática	5796977	
Asesor	Lic. Efraín Torrejón	Lic. en Ingeniería Informática	1337531	
Asesor	Ing. Silvana Paz	Ingeniera en Informática	1860481	
Asesor	Lic. Deysi Arancibia	Lic. en Ingeniería Informática	-----	

Tabla 2. Participantes del Equipo de Trabajo

1.2.3 Equipo de Trabajo de: Empresas/Instituciones/Organizaciones Participantes/Cooperantes

Nombre: Caja Nacional de Salud, Administración Regional Tarija			
Dirección: Calle Daniel Campos 0831		Teléf. Oficina: 66-42940	
Nombre y Apellidos	Cargo	C.I.	Firma
Ing. Carlos Michel Cachambi	Encargado de Sistemas Caja Nacional de Salud - Regional Tarija	5745158	

Tabla 3. Equipo de Trabajo

1.2.4 Actividades Previstas para los Integrantes del Equipo de Investigación

Responsable *	Actividades
Director	<p>Como Jefe de Proyecto:</p> <p>Organizar el equipo de trabajo.</p> <p>Planificar las actividades y controlar del cronograma del proyecto.</p> <p>Asignar y gestionar recursos y prioridades a los distintos componentes y actividades del proyecto.</p> <p>Mantener al equipo del proyecto enfocado en los objetivos.</p> <p>Realizar el seguimiento a cada etapa del proyecto.</p> <p>Supervisar el desarrollo del proyecto.</p> <p>Presentación final del sistema.</p>

	<p>Como Analista de Sistemas:</p> <p>Capturar la especificación y validación de requisitos interactuando con los usuarios mediante entrevistas.</p> <p>Elaborar el Análisis y Diseño del Sistema.</p> <p>Elaborar el Modelo de Datos (Base de Datos del Sistema).</p> <p>Como Programador:</p> <p>Realizar la Programación del Sistema Informático.</p> <p>Construcción de prototipos.</p> <p>Como Ingeniero de Software:</p> <p>Elaborar las pruebas funcionales del Sistema Informático.</p> <p>Como Formador o Capacitador:</p> <p>Formar al personal en el uso de las TIC para el manejo del producto final.</p>
Asesor	<p>Asesoramiento en los aspectos tecnológicos para el desarrollo del proyecto.</p> <p>Asesoramiento en el uso de la Metodología RUP (Utilización UML).</p> <p>Evaluación del documento del proyecto.</p>

Tabla 4. Actividades Previstas para los Integrantes del Equipo de Investigación.

1.2.4.1 Unidades de Gestión: Organigrama del Equipo del Proyecto

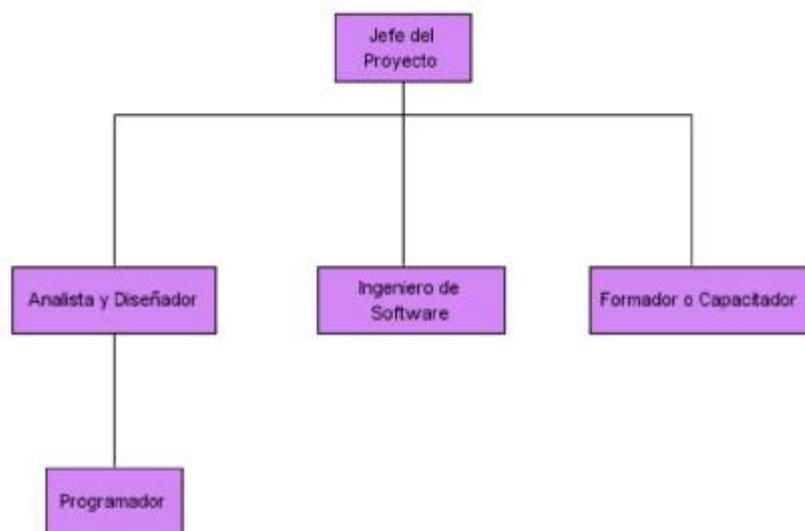


Figura 1. Organigrama del Equipo del Proyecto

1.3 Descripción del Proyecto

1.3.1 Resumen Ejecutivo del Proyecto

La Caja Nacional de Salud en sus más de cincuenta años de vida no ha podido sumarse de manera *consistente* al avance tecnológico referido al uso de los Sistemas Informáticos Basados en Computadoras.

La C.N.S. fue creada el 14 de diciembre de 1956 con el nombre de “Caja Nacional de Seguridad Social”, en sus inicios el Seguro Social tenía por objeto proteger a los trabajadores y a sus familias en los casos siguientes: enfermedad, maternidad, riesgos profesionales, invalidez, vejez era un régimen de seguridad social a corto y largo plazo [1].

La C.N.S. –Regional Tarija cuenta con dos grandes áreas operativas, que son el Área Médica, que incluye al sector salud tanto médico como paramédico y el Área Administrativa, que comprende a todos los funcionarios administrativos.

- Área Médica.- Comprende al Jefe Regional que abarca al Director del Hospital Obrero, del cual dependen el Jefe de Enfermeras y el Administrador del Hospital, el Policlínico y el coordinador del puesto médico de Bermejo.
- Área Administrativa.- Comprende el Jefe de Servicios Generales que tiene bajo su dependencia las Secciones de Contabilidad Cotizaciones, Tesorería, Almacenes, Afiliación y Vigencia de Derechos.

El fin de este trabajo es proponer la utilización de un software de computación para realizar el registro en la sección de afiliación, que coadyuve a proporcionar mayor seguridad y fiabilidad en el manejo de información de los afiliados en el menor tiempo.

La información generada por el sistema ayudará a proporcionar reportes fiables para las demás secciones dependientes de la información de la Sección de Afiliaciones.

Para lograr el propósito del proyecto debe desarrollarse en una plataforma Web. Ya que estos funcionan a través de navegadores, y se puede integrar varios programas. Esto proporciona una ventaja competitiva, en cual la idea de utilizar un sistema de plataforma web, es desarrollar un sistema en línea, integrando los negocios de la compañía en una única base de datos. Y esto permite a la vez: Flexibilidad y movilidad, Mantenimiento, Seguridad, Gestión centralizada

Al mismo tiempo se fortalezca la relación entre las secciones de Contabilidad Cotizaciones, Afiliación y Vigencia de Derechos.

La aplicación informática proporcionará estadísticas que apoyará en la gestión de registro de empresas empleadoras y sus empleados a asegurados, permitiendo a sus altos ejecutivos mejorar en la toma de

decisiones.

En el inicio del proyecto se lleva a cabo la identificación de las necesidades de los empleados de la Sección de Afiliación, así como la del administrador que será participe del proyecto, realizando una serie de entrevistas a los empleados.

Por lo que el proyecto concluirá con la capacitación al personal involucrado que forma parte del proyecto a través de exposiciones y demostraciones del sistema terminado.

Y por último el sistema establecerá una red de cooperación de información fiable entre las demás Áreas de la C.N.S.- Regional Tarija.

Para el desarrollo de estos dos componentes se realizaran las actividades previstas en el cronograma

1.3.2 Descripción, Fundamentación y Justificación del Proyecto

El Decreto Supremo 25875 del 15 de agosto del 2000 aprueba el sistema de gestión y modernización de la administración, el mismo que es de aplicación obligatoria en todas las entidades públicas. Al interior de la C.N.S. – Regional Tarija la inminente aplicación de este Decreto Supremo requirió el establecimiento de una estructura de personal informático capacitado en las nuevas tecnologías y manejo de equipo de servidores y comunicaciones que aseguren el uso de sistemas informáticos para el buen funcionamiento de estos.

El gran crecimiento vegetativo de la población asegurada en la ciudad de Tarija ha generado gran cantidad de información que tiene que ser correctamente administrada por la Sección de Afiliaciones.

La C.N.S. – Regional Tarija actualmente realiza sus registros en la Sección de Afiliación de manera manual, con mucho papeleo y tiempo para el registro de afiliados, propenso a fallos de registro, pues las circunstancias llevan a la demora del buen funcionamiento.

La emisión de reportes de las empresas empleadoras y sus empleados afiliados está propensa a proporcionar datos erróneos, por la gran cantidad de documentación que es archivada y que la baja actualización de datos de sus archivos que se encuentran en hojas impresas en grandes depósitos, que ha demostrado que lo difícil de depurar, ha provocado que se emitan reportes de baja fiabilidad para las demás secciones que dependen de esta sección, observándose un alto índice de incoherencia, de registros mal almacenados, ilegibles y repetidos.

El manejo de formularios en el registro de afiliados para las diferentes circunstancias que se puedan presentar ha generado un gran incremento de documentos, que deben ser archivados y que la búsqueda de los mismos genera mala atención a los asegurados.

En este sentido el presente proyecto de investigación pretende la creación de una aplicación informática que mejore en el registro de afiliados reduciendo el tiempo de registro y atención a los

asegurados, y permita el almacenamiento digital capaz de optimizar el espacio físico.

Esto permitirá realizar reportes con mayor fiabilidad y en poco tiempo presentando datos actualizados, a las demás secciones que dependen de la información de la Sección de Afiliación.

La búsqueda de información referente a las empresas empleadoras y empleados asegurados, mejorará el rendimiento de atención a los asegurados.

La participación activa de los involucrados será necesaria para la conclusión en éxito del presente proyecto.

1.3.2.1 Análisis de Causas del Problema

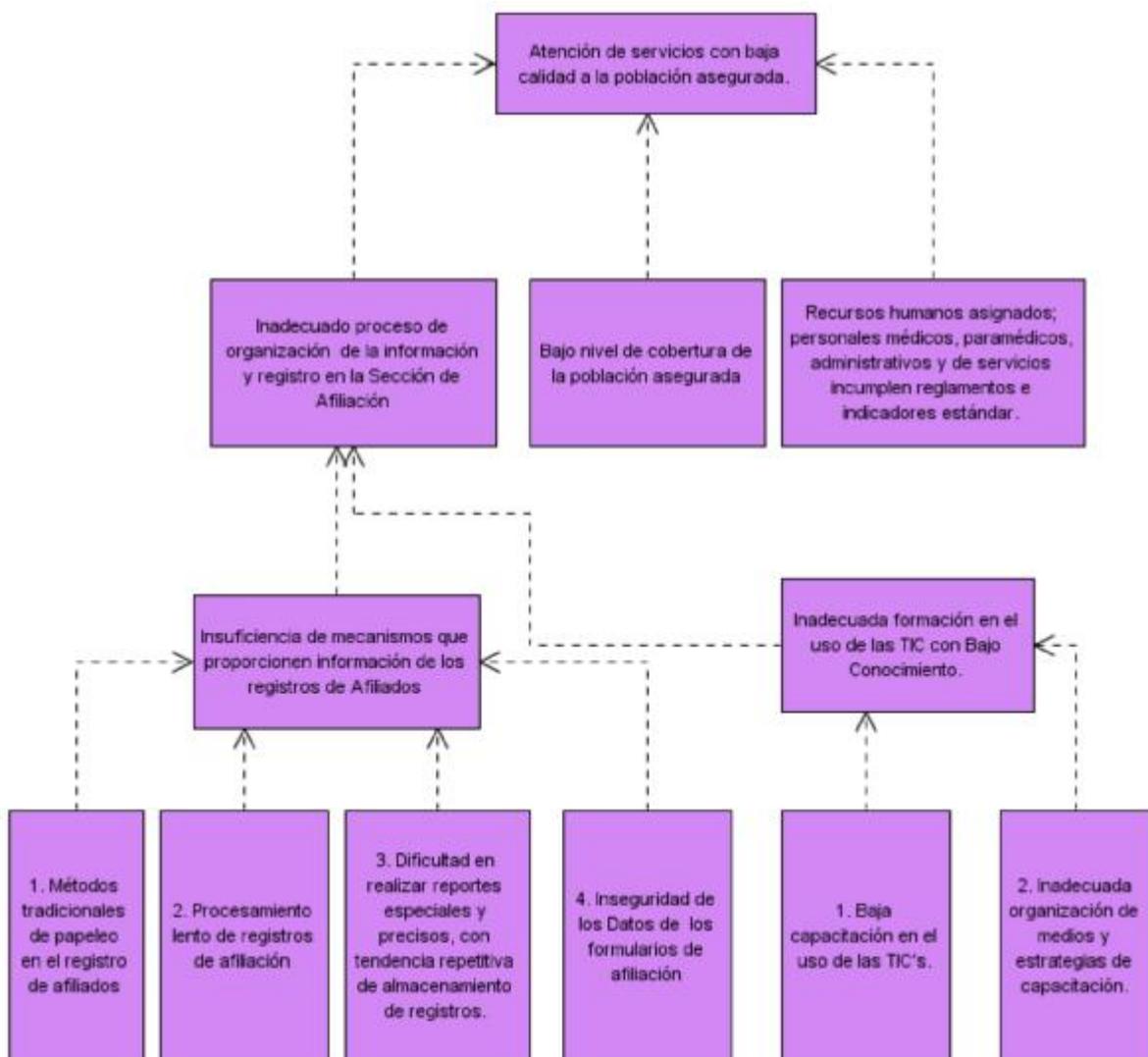


Figura 2. Árbol Problema

1.3.2 Análisis de Objetivos

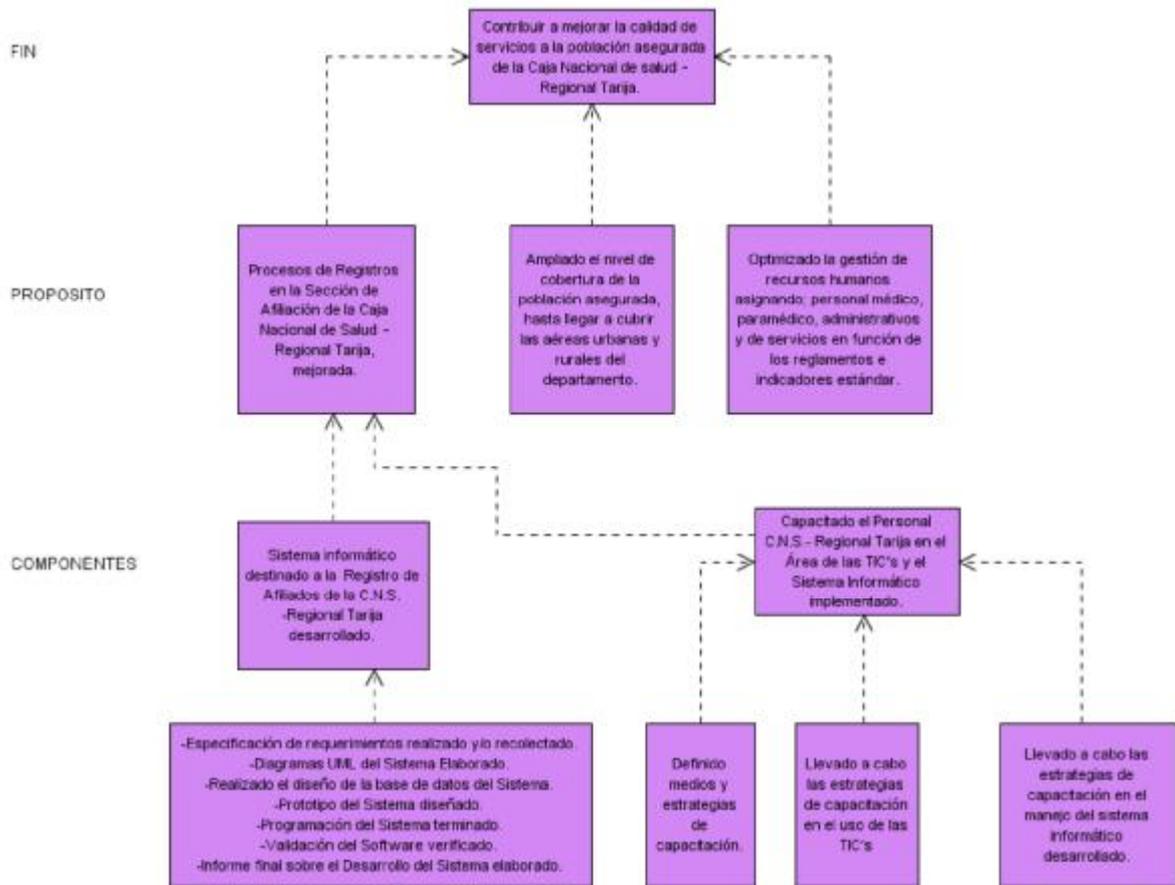


Figura 3. Árbol de Objetivos

1.3.3 Objetivos

1.3.3.1 Objetivo General

Procesos de Registros en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija, mejorado.

1.3.3.2 Objetivos Específicos

- Sistema Informático destinado al Registro de Afiliados de la Caja Nacional de Salud - Regional Tarija, desarrollado.
- Capacitado el Personal Caja Nacional de Salud - Regional Tarija, en el Área de las TIC's y el Sistema Informático Implementado.

1.3.4 Metodología

La forma de trabajo de este proyecto estará basada en:

1.3.4.1 Metodología para el Desarrollo de las Aplicaciones

Se utilizará la metodología **RUP (Racional Unified Process)**, que mejora considerablemente la calidad de desarrollo del sistema, ya que la misma utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software.

RUP es un proceso ágil de desarrollo que se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo concluye con una versión del producto para los clientes.

El flujo de trabajo fundamental tiene los siguientes pasos:

- **Requerimientos:** necesidades del negocio trasladadas a un sistema automatizado.
- **Análisis y Diseño:** requerimientos dentro de la arquitectura de software.
- **Programación e Implementación:** software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- **Pruebas:** el comportamiento requerido es el correcto y que todo lo solicitado está presente.

Requerimientos: en base a las entrevistas se obtendrá la información que refleje las necesidades de los involucrados para la determinación de requerimientos.

Análisis y diseño: en base a la determinación de requerimientos, se estructurará las diferentes vistas (Diagramas, Base de Datos, Pantallas) de la aplicación, tomando en cuenta metodologías de desarrollo de software.

Programación e Implementación: la programación será modular y orientada a objetos, se utilizarán tecnologías de punta, creando la aplicación informática que tenga el comportamiento deseado.

Pruebas y Validación: antes de desarrollar las pruebas se procederá a la introducción de datos. Introducida esta información al sistema se dará inicio a la fase de pruebas de desarrollo que serán mediante casos de prueba tomados de cada módulo y se realizarán los ajustes necesarios para una correcta validación.

Este proceso se torna repetitivo si se detectan inconsistencias en el sistema implicando el retorno de cualquiera de las fases anteriores para su corrección. [4]

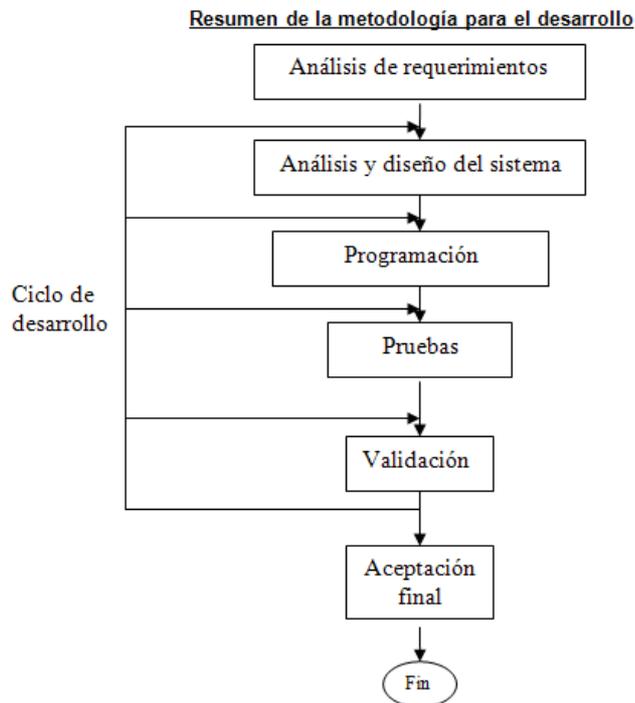


Figura 4. Resumen de la Metodología para el Desarrollo

1.3.4.2 Metodología para las Capacitaciones i/o Talleres

La metodología utilizada para esta capacitación i/o taller se basa en el modelo de experiencia “Aprender Haciendo”, logrando de esta manera la motivación para ser constructores de su propio conocimiento. Involucra trabajos de completa actividad en Talleres de Construcción de Contenidos, Laboratorios, Guías, donde se incentiva el trabajo creativo y práctico, experimentándose con procesos o fenómenos a partir de ideas o propuestas teóricas previamente formuladas.

Es una metodología que busca desarrollar la sensibilidad frente a problemas reales, estudiar alternativas de solución y evaluar sus implicancias en conjunto con la utilización de tecnologías.

Estrategias Didácticas

Descubriendo la Tecnología

Tipo: Sesiones Individuales y Grupales.

Descripción: Se desarrolla Actividades Guiadas donde los capacitadores hacen uso del material guía y van descubriendo por sí solos las potencialidades de los programas y herramientas y que luego ellos mismos documentan haciendo énfasis en qué experiencias descubiertas son de principal apoyo para su práctica pedagógica.

Aspectos Prácticos

Actividad teórico- práctica: propicia la modalidad del curso teórico con una actividad de la práctica en relación a la temática de estudio. Lo teórico y lo práctico se dan simultáneamente en forma conjunta e interrelacionada y es dirigida directamente por el capacitador participante.

Entrenamiento Virtual: formación en la que los elementos teóricos y los prácticos no se dan simultáneamente, sino separadamente y en entornos virtuales*. Para este sistema se utiliza el seguimiento de Tutores Facilitadores** para cada área temática.

Talleres de Formación continua: inclusión de la formación virtual* constante comprometiendo a los capacitadores*** a la continuidad de su capacitación a través de talleres y cursos de especialización.

Mecanismos de Trabajo durante la capacitación:

Fase Inicial:

Análisis de la Realidad Social de la Región: donde los Facilitadores** intercambian diferentes criterios con los docentes logrando tener una visión clara de las necesidades pedagógicas de los participantes enfocando la capacitación a la Presentación de Temas Individuales.

Diagnóstico de Iniciación: donde se recolecta información mediante evaluaciones de conocimientos que nos permitirá medir el nivel que poseen los participantes en la utilización de las aplicaciones.

Motivación y Sensibilidad: donde los Facilitadores** realizan dinámicas para motivar y sensibilizar sobre la importancia de desarrollar capacidades en el manejo de las TIC's, y de esta manera los capacitadores puedan aplicar la herramienta y aportar en su conjunto al desarrollo de su comunidad.

Fase de Capacitación:

Investigación y documentación: se promueve en todas las estrategias didácticas aplicadas a la investigación continua y descubrimiento de la tecnología reconociendo su valor e identificando su potencial para ser utilizados en su preparación de material, todo esto desarrollado por los mismos capacitadores***.

Defensa de Trabajos: donde los capacitadores trabajan en grupos o en forma individual presentando sus productos y siendo ellos mismos quienes se evalúan y corrigen para el enriquecimiento en conjunto del conocimiento y experiencia.

Talleres y Ponencias de Proyectos: donde los capacitadores*** presentan trabajos y exponen proyectos que van surgiendo a lo largo de la capacitación y los cuales son motivo de medición de conocimiento y desarrollo de destrezas.

Fase de Finalización:

*Medición de aprovechamiento: donde los Facilitadores** realizan las evaluaciones de trabajos en forma individualizada como a su vez de término de grupo capacitado. [3]*

** La formación virtual será desarrollada a través del sistema informático desarrollado o terminado, con el fin de que puedan conocer los sistemas informáticos o programas que utilizarán a futuro.*

*** Los facilitadores o facilitadores encargado de motivar en la importancia en el manejo de las TIC's, así como tiene la visión de las necesidades pedagógicas para la capacitación. En este presente proyecto vendría a ser el Encargado o Jefe de Sistemas.*

**** Lo Capacitadores los capacitadores o capacitador es el encargado de la formación continua, ya sea mediante él trabaja en grupo o individual, donde elaboran el contenido temático.*

1.3.5 Resultados Esperados

Sistema Informático desarrollado, probado y validado; que integre de manera eficaz la lógica de procesamiento de información de la sección involucrada y presentado al finalizar la ejecución del proyecto.

Al finalizar el proyecto al menos un 80% de los participantes en el proyecto han sido capacitados.

Mejorar la atención en la Sección de Afiliación, para las empresas empleadoras y sus empleados.

Crear una nueva conciencia tecnológica en el personal de la C.N.S.- Regional Tarija, respecto al gran beneficio que proporciona.

1.3.6 Transferencia de Resultados

1.3.6.1 Medios y Estrategias para la Transferencia de Resultados

Presentación final del sistema informático a la administración y la sección vinculada de la C.N.S.- Regional Tarija.

Entrega de instaladores del sistema informático y la documentación desarrollada en el proyecto.

Socialización del producto final con los demandantes e instituciones involucradas.

Elaboración de gigantografías.

En la que forman parte de la socialización del proyecto.

1.3.6.2 Grupo de Beneficiarios de los Resultados

Los beneficiarios del proyecto son:

Jefes y Funcionarios de la Sección de Afiliación de la C.N.S. – Regional Tarija.

Jefes y Funcionarios de la Sección de Cotización de la C.N.S. – Regional Tarija.

Jefes y Funcionarios de la Sección de Vigencia de Derechos de la C.N.S. – Regional Tarija.

La Administración de la C.N.S. –Regional Tarija, que tiene estrecha relación con el proyecto a desarrollar.

Empresas empleadoras del Departamento, y por consiguiente:

Empleados.

Beneficiarios.

Asegurados.

1.3.7 Cronograma de Actividades



Figura 5. Cronograma de Actividades

1.3.8 Marco Lógico del Proyecto

RESUMEN NARRATIVO	INDICADORES VERIFICABLES OBJETIVAMENTE	MEDIOS DE VERIFICACIÓN	SUPUESTOS
<p><u>Fin</u></p> <p>Contribuir a mejorar la calidad de servicios a la población asegurada de la Caja Nacional de Salud –Regional Tarija.</p>	<p>Después de un año de ejecución del proyecto al menos el 60% de los beneficiados con el proyecto expresan su satisfacción por la atención brindada en la C.N.S. – Regional Tarija.</p>	<p>Encuestas realizadas a los asegurados y beneficiarios con respecto a la calidad de atención avalada por el administrador de la C.N.S – Regional Tarija.</p>	<p>Se mantenga la Estructura Organizativa de la C.N.S. - Regional Tarija.</p> <p>La C. N. S. – Regional Tarija comprometida en la ejecución del proyecto.</p> <p>El proyecto es considerado de interés estratégico para la C.N.S. - Regional</p>

			<p>Tarija y su aplicación involucra la actualización de los manuales de funciones de los empleados, de tal forma que se de sostenibilidad al proyecto.</p> <p>Monto presupuestado por la C.N.S. – Regional Tarija para invertir en iniciativas y opciones de incorporar nuevos componentes al proyecto.</p>
<p><u>Propósito</u></p> <p>“Procesos de Registro en la Sección de Afiliación de la Caja Nacional de Salud</p>	<p>El tiempo promedio en los Procesos de Registro se ha reducido en un</p>	<p>Cuadro Comparativo referente al tiempo de ejecución de Procesos</p>	<p>RR.HH. del Área Informática deben</p>

<p>– Regional Tarija”, mejorados.</p>	<p>50% en la atención a las empresas empleadoras y sus empleados en el área administrativa de la Sección de Afiliación respecto a los resultados del año anterior.</p>	<p>de Registro antes y después, avalado por el Jefe de Sección de Afiliación de la C.N.S.- Regional Tarija.</p>	<p>cumplir con el perfil adecuado.</p> <p>Infraestructura tecnológica referente a las TIC’s, es implementado bajo estándares de calidad.</p>
<p><u>Componentes</u></p> <p>Sistema Informático destinado al Registro de Afiliados de la Caja Nacional Salud – Regional Tarija desarrollado.</p>	<p>Al finalizar el proyecto se ha concluido el desarrollo el sistema informático de acuerdo a las especificaciones de requerimiento cumpliendo la norma 830 de IEEE., recopiladas de la Sección de Afiliación de la C.N.S.- Regional Tarija.</p> <p>Validación del software en un 80% de verificación para brindar mayor calidad del Sistema mediante pruebas</p>	<p>Documentación técnica del sistema informático basado en Web, aprobado por el Encargado de Sistemas de la C.N.S. – Regional Tarija.</p> <p>Informe de Conformidad del Encargado de Sistemas de la C.N.S.- Regional Tarija, con respecto al Sistema Informático desarrollado.</p>	<p>Información de requerimientos recopilada, confiable.</p>

<p>Capacitación del Personal Caja Nacional Salud -Regional Tarija, en el Área de las TIC's y el Sistema Informático implementado.</p>	<p>realizadas.</p> <p>Al finalizar el proyecto al menos el 80% de los trabajadores del área administrativa y médica de la C.N.S. – Regional Tarija, han sido capacitados sobre el uso de las TIC's, con una duración de 3 horas académicas, a nivel Inicial.</p> <p>Al finalizar el proyecto al menos el 95% de los trabajadores de la Sección de Afiliación de la C.N.S. – Regional Tarija, han sido capacitados en el uso del Nuevo Sistema</p>	<p>Informe de Conformidad del Administrador de la C.N.S. – Regional Tarija con respecto al Sistema Informático desarrollado.</p> <p>Copias de los Certificados de asistencia de capacitación del personal de la C.N.S.- Regional Tarija, avalados por la F.C. y T. (Vicedecanatura), (D.I.S.) Director(a) Departamento de Informática y Sistemas de la U.A.J.M.S. y Encargado de Sistemas de la C.N.S. – Regional Tarija.</p> <p>Lista firmada por los participantes de la capacitación,</p>	<p>Los miembros involucrados en el proyecto reconozcan la importancia de la capacitación.</p> <p>Los participantes en la capacitación llevada a cabo sobre el manejo del sistema tienen permanencia en la Sección de Afiliación de la C.N.S.- Regional</p>
---	---	--	--

	<p>Automatizado Implementado.</p>	<p>avalados por el Encargado de Sistemas de la C.N.S – Regional Tarija.</p> <p>Fotografías tomadas durante las capacitaciones, avalados por el Encargado de Sistemas de la C.N.S – Regional Tarija.</p>	<p>Tarija y aplican los conocimientos adquiridos.</p> <p>Usuarios (Personal Administrativo de la Sección de Afiliación de la C.N.S. - Regional Tarija) con disponibilidad e interés en capacitarse en el manejo del sistema.</p> <p>Presupuesto disponible para realizar la capacitación.</p>
--	-----------------------------------	---	---

<u>Actividades</u>			
C1. Sistema Informático			
1			
Especificación de requerimientos realizado y/o recolectado.	Fecha: 16 Abr. / 19 May. 201		Información del presupuesto de las partidas, de la Sección de Tesorería.
2			
Diagramas UML del Sistema Elaborado.	Fecha: 20 May. / 23 Jun. 201		Documentación del análisis y diseño del sistema, avalado por el Encargado de Sistemas de la C.N.S. – Regional Tarija.
3			
Realizado el diseño de la base de datos del Sistema.	Fecha: 20 May. / 02 Jul. 2010		
4			
Prototipo del Sistema diseñado.	Fecha: 15 Jun. / 19 Jul. 2010		Manual de Usuario, aprobado por el Encargado de Sistemas de la C.N.S. – Regional Tarija.
5			
Programación del Sistema terminado.	Fecha: 12 Jul. / 22 Oct. 2010		
6			
Validación del Software verificado.	Fecha: 16 Sep. / 04 Nov. 2010		Manual de Instalación, aprobado por el Encargado de Sistemas de la C.N.S. – Regional Tarija.
7			

<p>Informe final sobre el Desarrollo del Sistema elaborado.</p> <p>C2.Capacitacion Personal</p> <p>1 Definido medios y estrategias de capacitación.</p> <p>2 Llevado a cabo las estrategias de capacitación en el uso de las TIC's.</p> <p>3 Llevado a cabo las estrategias de capacitación en el manejo del sistema informático desarrollado.</p>	<p>Fecha: 20 Oct. / 08 Nov. 2010</p> <p>Fecha: 05 Jul. / 29 Nov. 2010</p> <table border="1" data-bbox="801 647 1229 1267"> <thead> <tr> <th>ITEM</th> <th>RUBROS</th> <th>TOTAL (Bs)</th> </tr> </thead> <tbody> <tr> <td>20000</td> <td>SERVICIOS NO PERSONALES</td> <td></td> </tr> <tr> <td></td> <td>21000. Servicios Basicos</td> <td>2760</td> </tr> <tr> <td></td> <td>22000. Servicios de Transporte</td> <td></td> </tr> <tr> <td></td> <td>23000. Alquileres</td> <td></td> </tr> <tr> <td></td> <td>24000. Mantenimiento y Reparacion</td> <td></td> </tr> <tr> <td></td> <td>25000. Servicios Profesionales y Comerciales</td> <td>1560</td> </tr> <tr> <td></td> <td>Sub total rubro</td> <td>4320</td> </tr> <tr> <td>30000</td> <td>MATERIALES Y SUMINISTROS</td> <td></td> </tr> <tr> <td></td> <td>31000. Alimentos y Productos Forestales</td> <td>180</td> </tr> <tr> <td></td> <td>32000. Productos de Papel, Carton e Impresos</td> <td>590</td> </tr> <tr> <td></td> <td>33000. Textiles y Vestuario.</td> <td></td> </tr> <tr> <td></td> <td>34000. Productos Quimicos, Combustibles y Lubricantes</td> <td></td> </tr> <tr> <td></td> <td>39000. Productos Varios.</td> <td>933</td> </tr> <tr> <td></td> <td>Sub total rubro</td> <td>1703</td> </tr> <tr> <td>40000</td> <td>ACTIVOS REALES</td> <td></td> </tr> <tr> <td></td> <td>43000. Maquinaria y Equipo.</td> <td>800</td> </tr> <tr> <td></td> <td>46000. Descripcion de Estudios y Proyectos para Inversion</td> <td></td> </tr> <tr> <td></td> <td>49000. Otros Activos</td> <td></td> </tr> <tr> <td></td> <td>Sub total rubro</td> <td>800</td> </tr> <tr> <td></td> <td>TOTAL</td> <td>6823</td> </tr> </tbody> </table>	ITEM	RUBROS	TOTAL (Bs)	20000	SERVICIOS NO PERSONALES			21000. Servicios Basicos	2760		22000. Servicios de Transporte			23000. Alquileres			24000. Mantenimiento y Reparacion			25000. Servicios Profesionales y Comerciales	1560		Sub total rubro	4320	30000	MATERIALES Y SUMINISTROS			31000. Alimentos y Productos Forestales	180		32000. Productos de Papel, Carton e Impresos	590		33000. Textiles y Vestuario.			34000. Productos Quimicos, Combustibles y Lubricantes			39000. Productos Varios.	933		Sub total rubro	1703	40000	ACTIVOS REALES			43000. Maquinaria y Equipo.	800		46000. Descripcion de Estudios y Proyectos para Inversion			49000. Otros Activos			Sub total rubro	800		TOTAL	6823	<p>Información presupuesto de gastos.</p>	
ITEM	RUBROS	TOTAL (Bs)																																																																
20000	SERVICIOS NO PERSONALES																																																																	
	21000. Servicios Basicos	2760																																																																
	22000. Servicios de Transporte																																																																	
	23000. Alquileres																																																																	
	24000. Mantenimiento y Reparacion																																																																	
	25000. Servicios Profesionales y Comerciales	1560																																																																
	Sub total rubro	4320																																																																
30000	MATERIALES Y SUMINISTROS																																																																	
	31000. Alimentos y Productos Forestales	180																																																																
	32000. Productos de Papel, Carton e Impresos	590																																																																
	33000. Textiles y Vestuario.																																																																	
	34000. Productos Quimicos, Combustibles y Lubricantes																																																																	
	39000. Productos Varios.	933																																																																
	Sub total rubro	1703																																																																
40000	ACTIVOS REALES																																																																	
	43000. Maquinaria y Equipo.	800																																																																
	46000. Descripcion de Estudios y Proyectos para Inversion																																																																	
	49000. Otros Activos																																																																	
	Sub total rubro	800																																																																
	TOTAL	6823																																																																

Tabla 5. Marco Lógico del Proyecto

1.4 Presupuesto / Justificación

ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs)
20000	SERVICIOS NO PERSONALES			
	21000. Servicios Básicos			2760
	22000. Servicios de Transporte			
	23000. Alquileres			
	24000. Mantenimiento y Reparación			
	25000. Servicios Profesionales y Comerciales			1560
	Sub total rubro			4320
30000	MATERIALES Y SUMINISTROS			
	31000. Alimentos y Productos Forestales			180
	32000. Productos de Papel, Cartón e Impresos			590
	33000. Textiles y Vestuario.			
	34000. Productos Químicos, Combustibles y Lubricantes			
	39000. Productos Varios.			933
	Sub total rubro			1703
40000	ACTIVOS REALES			
	43000. Maquinaria y Equipo.			800
	46000. Descripción de Estudios y Proyectos para Inversión			
	49000. Otros Activos			
	Sub total rubro			800
	TOTAL			6823

Tabla 6. Presupuesto / Justificación

- GRUPO 20000. SERVICIOS NO PERSONALES

b) SUB GRUPO 21000. Descripción de los Gastos de Servicios Básicos

Partida	Tipo de servicio básico *	Costo	Tiempo mes	Costo Total
21100	Comunicación	250	8	2000
21200	Energía Eléctrica	60	8	480
21300	Agua			
21400	Servicios Telefónicos	35	8	280
Total				2760

* Se refiere principalmente a los gastos por servicios; como: servicio de correo, radiogramas, servicio telefónico, fax, Internet.

c) SUB GRUPO 22000. Descripción de los Gastos de Viajes y Transporte de Personal

Partida	Personal	Lugar	Nº de viajes	Costo unitario*	Costo total
22100	Pasajes				
Total					

* En el caso de pasajes debe indicarse el costo de ida y vuelta (costo unitario), indicando el número de viajes.

Partida	Personal	Lugar	Duración (días)	Costo unitario*	Costo total
22200	Viáticos				
22300	Fletes y Almacenamientos				
22600	Transporte de Personal				
Total					
Total sub grupo 22000					

* En el caso de los viáticos, debe considerarse la escala establecida por la UAJMS.

d) SUB GRUPO 23000. Descripción de los Gastos por Concepto de Alquileres de Equipos y Maquinarias

Partida	Alquiler de equipo y maquinaria	Costo unitario	Tiempo mes	Costo total
23100	Alquiler de Edificios			

23200	Alquiler de Equipos y Maquinaria			
23300	Alquiler de Tierras y Terrenos			
Total				

* Se refiere principalmente a los Gastos por el uso de Edificios y Equipos y Maquinaria en General

e) SUB GRUPO 24000. Descripción Mantenimiento y Reparación

Partida	Mantenimiento y reparación de equipo y maquinaria	Costo unitario	Tiempo mes	Costo total
24100	Mantenimiento y Reparación de Edificios y Equipos			
24300	Otros Gastos por Mantenimiento y Reparación			
Total				

* Se refiere principalmente a los gastos por el mantenimiento y reparación de edificios y equipos y maquinaria en general.

f) SUB GRUPO 25000. Descripción de los Gastos en Servicios Profesionales y Comerciales

Partida	Tipo de servicio profesional y comercial *	Cantidad	Costo unitario	Tiempo mes	Costo total
25500	Publicidad				
25600	Imprenta				
	Certificados de Participación	30	5		150
	Empastados	12	50		600
	Gigantografías	3	120		360
25700	Capacitación de Personal	30 personas	15	1	450
25800	Estudios e Investigaciones Para Proyectos de Inversión				
25810	Consultores por Producto				
25820	Consultores en Línea				
Total					1560

* Se refiere a gastos por servicios profesionales de asesoramiento especializado, se incluyen, estudios, investigaciones, publicidad, imprenta, fotocopias, capacitación de personal y otros ejecutados por terceros.

- GRUPO 30000. MATERIALES Y SUMINISTROS

g) SUB GRUPO 31000. Descripción de los Gastos Alimentos y Productos Agroforestales

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
31110	Refrigerios y Gastos Administrativos Capacitación: Bebida o Refrigerio Aperitivos	10 30	9 3	90 90
31200	Alimento para Animales			
31300	Productos Agroforestales y Pecuarios			
Total				180

* Se refiere a la adquisición de materiales y bienes como: alimentos y productos agroforestales, alimentos y bebidas para personas (indicar el total de refrigerios), alimentos para animales, productos pecuarios.

h) SUB GRUPO 32000. Descripción del Gasto de Productos de Papel, Cartón e Impresos

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
32100	Papel de Escritorio Resmas Hojas Bond	8	40	320
32200	Productos de Artes Gráficas, Papel y Cartón			
32300	Libros y Revistas Libro Base de Datos SQLServer Libro Referencia a Diseño grafico	1 1	150 120	150 120
32400	Textos de Enseñanza			
Total				590

* Se refiere a la adquisición de papel y cartón en sus diversas formas y clases, impresos y publicaciones, periódicos, revistas, libros, fotocopias, etc.

i) SUB GRUPO 33000. Descripción del Gasto en Textiles y Vestuario

Partida	Productos textiles y vestuarios	Cantidad	Costo/Unitario	Total

33100	Hilados y Telas			
33200	Confecciones Textiles			
33300	Prendas de Vestir			
33400	Calzados			
Total				

* Se refiere principalmente a los gastos por vestuario uniformes, ropa de trabajo

j) SUB GRUPO 34000. Combustibles, Productos Químicos, Farmacéuticos y Otros

Partida	Combustibles, Productos Químicos, Farmacéuticos y Otros	Cantidad	Costo/Unitario	Total
34110	Combustibles y Lubricantes para Consumo			
34200	Productos Químicos y Farmacéuticos			
34400	Productos de Cuero y Caucho			
34500	Productos de Minerales no Metálicos y Plásticos			
34600	Productos Metálicos			
34700	Minerales			
34800	Herramientas Menores			
Total				

* Se refiere a gastos de combustibles, químicos, productos farmacéuticos, llantas etc.

k) SUB GRUPO 39000. Descripción del Gasto en Productos Varios

Partida	Productos de cuero y caucho	Cantidad	Costo/Unitario	Total
39100	Material de Limpieza			
39400	Instrumental Menor Médico - Quirúrgico			
39500	Útiles de Escritorio y de Oficina			
	Agenda Personal	1	25	25
	Memorias Flash	2	150	300
	Bolígrafos, Lápices	6	3	18

	CD, DVD	20	3	60
	Tinta Negra	10	35	350
	Tinta a Colores	6	30	180
39700	Útiles y Materiales Eléctricos			
39800	Otros Repuestos y Accesorios			
Total				933

*Se refiere principalmente a los gastos por productos de limpieza, todo lo referente a la funcionamiento de la oficina en material de escritorio.

- GRUPO 40000. **ACTIVOS REALES**

l) SUB GRUPO 43000. Descripción del Gasto de Maquinaria y Equipo

Partida	Tipos de productos	Cantidad	Costo/Unitario	Total
43100	Equipo de Oficina y Muebles			
	Impresoras	1	800	800
43200	Maquinaria y Equipo de Producción			
43300	Equipos de Transporte, Tracción y Elevación			
43400	Equipo Médico y de Laboratorio			
43700	Otra Maquinaria y Equipo			
Total				800

* Se refiere principalmente a los gastos por muebles y enseres, equipo de oficina, comunicación, equipamiento.

m) SUB GRUPO 46000. Descripción de Estudios y Proyectos para Inversión

Partida	Productos textiles y vestuarios	Cantidad	Costo/Unitario	Total
46100	Para Construcción de Bienes de Dominio Privado			
Total				

* Se refiere principalmente a los gastos por servicios de terceros para la realización de investigaciones y otras actividades técnico – profesionales necesarias para la construcción y mejoramiento de bienes.

n) SUB GRUPO 49000. Descripción del Gasto de Otros Activos

Partida	Tipos de productos *	Cantidad	Costo/Unitario	Total
---------	----------------------	----------	----------------	-------

49100	Activos Intangibles			
Total				

* Se refiere a los gastos en la compra de software, licencias.

1.4.1 Curriculum Vitae

1.4.1.1 Antecedentes Personales

Yana	Cerezo	Omar Elías	5796977 Tja.
Apellido Paterno	Apellido Materno	Nombre	C.I.
29/10/1987	M...X... F.....	Calle Suipacha entre Cochabamba y Felipe Echazu	
Fecha de Nacimiento	Sexo	Dirección	
Tarija	---	76185429	omar182_ya@hotmail.com
Ciudad	Teléfono Domicilio	Celular	Correo electrónico

Tabla 7. Antecedentes Personales

1.4.1.2 Antecedentes Académicos

Título obtenido	Universidad	País	Año
Diplomado en Pedagogía y Didáctica en la Educación Superior.	“Universidad Privada de Oruro – Institución Boliviana de Capacitación”.	Bolivia	2006
Ing. Informática - 1° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2006
Ing. Informática - 2° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2006
Ing. Informática - 3° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2007
Ing. Informática - 4° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2007
Ing. Informática - 5° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2008
Ing. Informática - 6° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2008
Ing. Informática - 7° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 8° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 9° Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2010

Tabla 8. Antecedentes Académicos

1.4.1.3 Participación en Proyectos de Investigación

Título proyecto	Institución	Cargo	Año
Sistema de Seguimiento a Graduados de la UAJMS como Herramienta de Apoyo a la Gestión Universitaria.	“Universidad Autónoma Juan Misael Saracho”.	Investigador Junior	2008

Tabla 9. Participación en Proyectos de Investigación

1.4.1.4 Publicaciones Realizadas (libros, revistas, compendios y otros)

Autor	Tipo de publicación, año, título, volumen, páginas, editorial

Tabla 10. Publicaciones Realizadas

1.4.1.5 Antecedentes en Ayudantías

Carrera	Semestre	Año
Ing. Informática	7° Semestre	2009
Ing. Informática	8° Semestre	2009
Ing. Informática	9° Semestre	2010

Tabla 11. Antecedentes en Ayudantías

1.4.1.6 Cursos Realizados

Curso	Institución	Duración en Hrs.	Año
Formación a auxiliares de cátedra en el Área de la Matemática.	UAJMS- Departamento de Matemática.	8	2006
Profesorado en administración y gestión de la educación.	Institución Boliviana de Capacitación Bolivia.	180	2006
Interculturalidad y análisis del proyecto de ley Avelino Siñani y Elisardo Pérez.	Institución Boliviana de Capacitación – Consultora Académica & Empresarial S.R.L.	10	2006
Seminario Taller: Módulo III: Lectura Veloz en Vertical. Módulo IV: Agilidad Mental.	Institución Boliviana de Capacitación – Consultora Académica & Empresarial S.R.L.	20	2006
Retos del tercer milenio Autoestima-Extra Valor-Liderazgo.	Escuela de Liderazgo y Desarrollo Humano.	30	2006
Relaciones Humanas para el Desarrollo Personal.	Escuela de Liderazgo y Desarrollo Humano.	30	2006
1° Seminario Internacional a Nivel Diplomado en Formación Pedagógica y Didáctica en la Educación Superior. Nuevas tecnologías aplicadas a la	Institución Boliviana de Capacitación Académica & Empresarial.	200	2006

<p>educación en el proceso de enseñanza y modalidades de aprendizaje.</p> <p>Calidad educativa, problemas del aprendizaje y diagnóstico docente.</p> <p>Visión de la nueva forma de aprender y enseñar.</p> <p>El proceso de integración e inclusión de personas de personas con necesidades especiales a través del sistema educativo regular.</p> <p>Procesos psicolingüísticos de la educación superior.</p> <p>Procesos de aprendizaje y enseñanza didáctica.</p> <p>Aprendizaje significativo en contraste con el proceso natural.</p> <p>El pensamiento crítico, la creatividad y la práctica en la educación.</p>			
XIV Congreso Nacional de Ciencias de la Computación – Conferencias.	Universidad Mayor de San Andrés.	20	2007
XIV Congreso Nacional de Ciencias de la Computación – Tutoriales.	Universidad Mayor de San Andrés.	30	2007
El Poder de la Inteligencia Emocional y de las Inteligencias Múltiples.	Escuela de Liderazgo y Desarrollo Humano.	30	2007
Oratoria y Liderazgo.	Taller de Expresión Artística Tarija.	24	2008

Tabla 12. Cursos Realizados

II. Capítulo 2: Contexto

2.1 Componente 1: Sistema Informático para el Mejoramiento de Afiliación

2.1.1 Marco Teórico

2.1.1.1 Antecedentes

Desde el surgimiento de la teoría de la organización, la función esencial de la información en las organizaciones se ha acentuado. Una organización es un sistema compuesto por tres elementos: personas, materiales e información. Los sistemas de información, por su parte, surgen como sistemas complejos y abiertos que interactúan con otros sistemas y subsistemas como parte de su actuación. Por los años 90, una de las concepciones más defendidas por la gestión de la información fue que las organizaciones son sistemas de información.

El uso de ciertos conceptos tomados de la teoría de sistemas y del campo de la informática llevó a un alto grado de desarrollo entre los sistemas de información. Aunque existen diversas definiciones, hechas desde diferentes enfoques, sobre los sistemas de información, en su gran mayoría tienen puntos

en común. El análisis realizado sobre las definiciones más frecuentes efectuadas en la década de los años 90 revela que constituyen un conjunto integrado de procesos, elementos o componentes que – según las estrategias y necesidades de una organización– recopilan, elaboran y distribuyen la información necesaria.

Un sistema moderno de gestión de información exige la aplicación de nuevas tecnologías de información; sin embargo, la tecnología por sí sola no es suficiente para lograr una buena gestión de información. Son diversos los procesos que conforman los sistemas de gestión de información; ellos generan las entradas y salidas del sistema o de otros procesos relacionados; también pueden identificarse, controlarse, corregirse o actualizarse en la medida en que se producen las transformaciones del entorno y evoluciona la organización, como vía incuestionable para garantizar su calidad, eficiencia y mejora continua.

Como resumen de este antecedente, puede decirse que los sistemas de gestión de información, en su definición más general, se refieren al conjunto de todos los componentes necesarios que se interrelacionan, con el objetivo de tramitar y facilitar la información sobre el tema de interés para su consumo en cualquier medio, momento y lugar.

2.1.1.2 Estudio de la Variable Independiente: Gestión de la Información

2.1.1.2.1 Caja Nacional de Salud

Adentrándonos a la gestión de la información en la Caja Nacional de Salud, diríamos que es una institución descentralizada de derecho público sin fines de lucro, con personería jurídica, autonomía de gestión y patrimonio independiente, encargada de la gestión, aplicación y ejecución del Régimen de Seguridad Social a corto plazo (Enfermedad, Maternidad y Riesgos Profesionales).

La Caja Nacional de Salud brinda protección integral en el campo de la salud a toda su población protegida a través de sus Administraciones Regionales y Agencias Distritales, uno de sus objetivos es remodelar, refundar y construir hospitales además de policlínicos, acorde a los niveles de la demanda.

La Caja Nacional de Salud hace prestación de sus servicios en varios Centros Asistenciales que están ubicados en los Departamentos de La Paz, Cochabamba, Santa Cruz, Chuquisaca, Oruro, Potosí, Beni, Pando y Tarija.

2.1.1.2.2 Antecedentes

La Caja Nacional de Salud (C.N.S.), inicia sus actividades como Caja Nacional de Seguridad Social (C.N.S.S.), etapa que abarca de diciembre de 1956 hasta marzo de 1987 y comprende la promulgación del Código de Seguridad Social en fecha 14 de diciembre de 1956 y la de su Decreto Reglamentario o Reglamento del Código de Seguridad Social el 30 de septiembre de 1959.

En esta etapa también están comprendidos el Decreto Ley de Racionalización de Aportes de 28 de marzo de 1972, el Decreto Ley de Reformas al Código de Seguridad Social y el Decreto Ley de Complementación de Reformas de 3 de junio de 1977.

La promulgación del Código de Seguridad Social significó un avance de la Seguridad Social Boliviana con relación a los demás países latinoamericanos. Sin embargo, desde su inicio la administración de los seguros establecidos en el citado Código no cumplieron con el principio de unidad de gestión, por cuanto se encargó la gestión del Seguro Social Obligatorio a varias instituciones, siendo la más importante, la Caja Nacional de Seguridad Social, entidad matriz gestora del Seguro Social Obligatorio Integral, con más del 80% de asegurados activos y pasivos, pertenecientes a la mayoría de las ramas de la actividad económica.

Las prestaciones señaladas en el Código de Seguridad Social comprendían los regímenes de enfermedad, maternidad, riesgos profesionales, invalidez, vejez, muerte y el régimen especial de asignaciones familiares.

Después de 30 años de administración integral del Seguro Social, el 15 de abril de 1987 se promulga la Ley Financial 0924, que en su artículo tercero afecta los esquemas administrativo y financiero del Sistema de Seguridad Social, procediéndose a la separación de los seguros, administrados integralmente hasta ese entonces. Dejándose a las Cajas la administración de los seguros a corto plazo: enfermedad, maternidad, riesgos profesionales y a los fondos complementarios la administración de las prestaciones a largo plazo: invalidez, vejez y muerte, aspectos que son ratificados por su Decreto Reglamentario No. 21637 del 25 de junio de 1987.

En consecuencia la Caja Nacional de Seguridad Social que hasta marzo de 1987 administraba el seguro integral, se convierte en la Caja Nacional de Salud, institución descentralizada de derecho público sin fines de lucro, con personería jurídica, autonomía de gestión y patrimonio independiente, encargada de la gestión aplicación y ejecución del régimen de Seguridad Social a Corto Plazo: enfermedad, maternidad y riesgos profesionales, instituidos por el Código de Seguridad Social, su Reglamento, la Ley Financial 924, el Decreto Supremo 21637 y demás disposiciones legales conexas.

2.1.1.2.3 Misión y Visión

La **misión** de la Caja Nacional de Salud a través de sus Administraciones Regionales y Agencias Distritales es brindar protección integral en el campo de la salud a toda su población protegida, como parte activa y componente de la población boliviana. Se rige por los principios de Universalidad, Solidaridad, Unidad de Gestión, Economía, Oportunidad y Eficacia en el otorgamiento de las prestaciones de salud, optimizando el uso de recursos y buscando ampliar el nivel de cobertura.

2.1.1.2.4 Objetivos Institucionales

- Optimizar la gestión de recursos humanos asignando y utilizando personal médico, paramédico, administrativos y de servicios en función de parámetros e indicadores estándar.
- Remodelar y construir hospitales además de policlínicos, acorde a los niveles de la demanda.
- Brindar atenciones en salud con calidad a la población asegurada con la implementación de planes, programas y control de calidad.
- Lograr el equilibrio financiero, incrementando los ingresos y optimizando el gasto.
- Incrementar la población cubierta y disminuir el nivel de desafiliaciones.
- Elevar el modelo de atención en salud (Medicina Familiar y Comunitaria) hasta alcanzar niveles óptimos de eficacia, eficiencia y economía.
- Implementar por fases, un modelo de administración con desconcentración administrativa, financiera y técnica.
- Proveer a los centros médicos de manera oportuna, suficientes medicamentos, insumos, materiales y equipo médico.
- Mejorar los índices de productividad y rendimiento (salud y administración) hasta cubrir la demanda insatisfecha.

2.1.1.2.5 Organigrama – Administración Regional Tarija

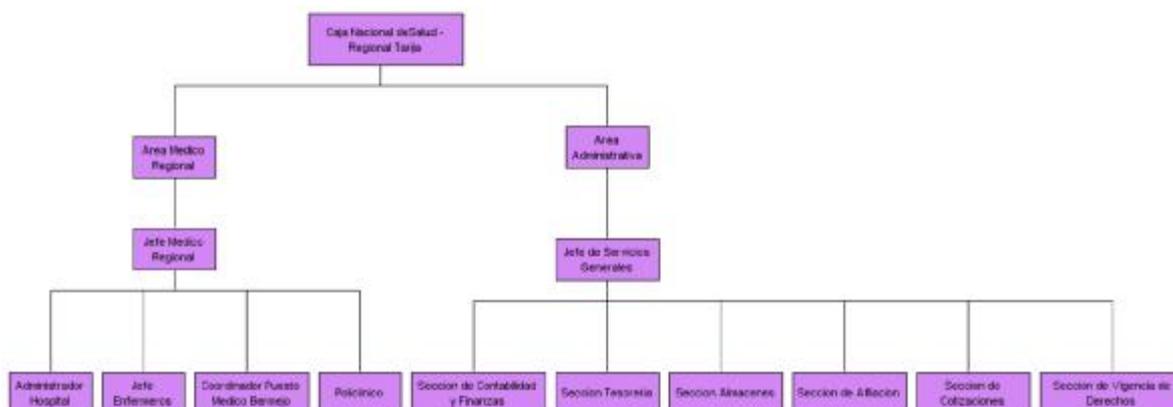


Figura 6. Organigrama – Administración Regional Tarija

La Caja Nacional de Salud tiene la siguiente información: afiliación para el trabajador, afiliación para la esposa, afiliación para hijo/hija, afiliación para conviviente, afiliación para el esposo, afiliación para los hermanos, afiliación para madre y madre, afiliación de estudiantes, afiliación para el seguro voluntario, afiliación de viudas y beneméritos, etc.

La Caja Nacional de Salud, Administración Regional Tarija está organizada de la siguiente forma:

El Dr. Juan Seborga Miranda es el actual Administrador Regional C.N.S.

El Lic. Carlos Michel Cachambi es el Encargado de Sistemas de la C.N.S.

Actualmente la gestión de la información académica está cubierta en parte mediante un sistema vía web que cubre sólo el área administrativa de la Sección de Afiliación y no así el Área Médica.

2.1.1.3 Estudio de la Variable Dependiente: Sistema de Gestión de la Información

El manejo constante de información valiosa y cuantiosa de diversos establecimientos académicos que procesan tal información aun basándose en un sistema manual que muchas veces es lento, inseguro e inadecuado ha puesto en claro la necesidad de un sistema mucho más rápido y confiable es por eso que se propone como solución el desarrollo de un sistema de gestión de información que facilite a la institución mejorar la eficiencia en sus áreas de operación interna en tareas como gestión empresas empleadoras, gestión de asegurado y gestión beneficiarios, Para esto debemos tomar en cuenta qué metodologías y tecnologías se aplicarán.

2.1.1.4 Relación entre la Variable Independiente y Dependiente

X (i) Desarrollo de Sistema Web para un mejor control del Área Administrativa	Y (i) Control del Área Administrativa
Analizar la situación actual de la Caja Nacional de Salud Regional Tarija.	Factibilidad de los procesos.
Determinar los requerimientos necesarios.	Mejorarle proceso de registro de la Sección de Afiliación de la C.N.S. Regional Tarija.
Elaborar el diseño lógico y físico del Sistema Automatizado.	Solucionar necesidades de las áreas administrativas.

Tabla 13. Relación entre la Variable Independiente y Dependiente

2.1.1.5 Metodología de Desarrollo

2.1.1.5.1 Metodología RUP(Racional Unified Process)

a) **Definición.- RUP** es un marco del proyecto que describe una clase de los procesos que son iterativos e incrementales. Es un proceso de Ingeniería de Software que captura las mejores prácticas del conocimiento de líderes en Ingeniería de Software y que provee un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo.

Los **procesos de RUP** estiman tareas y horario del plan midiendo la velocidad de iteraciones concerniente a sus estimaciones originales. Las iteraciones tempranas de proyectos conducidos por RUP se enfocan fuertemente sobre arquitectura del software; la puesta en práctica rápida de características se retrasa hasta que se ha identificado y se ha probado una arquitectura firme.

Nos permite realizar un levantamiento exhaustivo de requerimientos.

Las actividades de RUP se centran en crear y mantener modelos, utilizando UML, en forma efectiva.

- Busca detectar defectos en las fases iniciales.
- Intenta reducir al número de cambios tanto como sea posible.
- Realiza el Análisis y Diseño, tan completo como sea posible.
- Diseño genérico, intenta anticiparse a futuras necesidades.
- Las necesidades de clientes no son fáciles de discernir.
- Existe un contrato prefijado con los clientes.
- El cliente interactúa con el equipo de desarrollo mediante reuniones.

b) Características Esenciales.- Los autores de RUP destacan que el proceso de software propuesto por RUP tiene tres características esenciales: está dirigido por los Casos de Uso, está centrado en la arquitectura, y es iterativo e incremental.

- **Está dirigido por los Casos de Uso:** los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. Se define un Caso de Uso como un fragmento de funcionalidad del sistema que proporciona al usuario un valor añadido. Los Casos de Uso representan los requisitos funcionales del sistema.

Los Casos de Uso no sólo inician el proceso de desarrollo sino que proporcionan un hilo conductor, permitiendo establecer trazabilidad entre los artefactos que son generados en las diferentes actividades del proceso de desarrollo.

En RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba.

- **Está Centrado en su Arquitectura:** la arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo.

La arquitectura involucra los aspectos estáticos y dinámicos más significativos del sistema, está relacionada con la toma de decisiones que indican cómo tiene que ser construido el sistema y ayuda a determinar en qué orden. Además la definición de la arquitectura debe tomar en consideración elementos de calidad del sistema, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible durante todo el proceso de desarrollo. La arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados. Muchas de estas restricciones constituyen requisitos no funcionales del sistema.

En el caso de RUP además de utilizar los Casos de Uso para guiar el proceso se presta especial atención al establecimiento temprano de una buena arquitectura que no se vea fuertemente impactada ante cambios posteriores durante la construcción y el mantenimiento.

Cada producto tiene tanto una función como una forma. La función corresponde a la funcionalidad reflejada en los Casos de Uso y la forma la proporciona la arquitectura. Existe una interacción entre los Casos de Uso y la arquitectura, los Casos de Uso deben encajar en la arquitectura cuando se llevan a cabo y la arquitectura debe permitir el desarrollo de todos los Casos de Uso requeridos, actualmente y en el futuro. Esto provoca que tanto arquitectura como Casos de Uso deban evolucionar en paralelo durante todo el proceso de desarrollo de software.

- **Es Iterativo e Incremental:** según el equilibrio correcto entre los Casos de Uso y la Arquitectura es algo muy parecido al equilibrio de la forma y la función en el desarrollo del producto, lo cual se consigue con el tiempo. Para esto, la estrategia que se propone en RUP es tener un proceso iterativo e incremental en donde el trabajo se divide en partes más pequeñas o mini proyectos. Permitiendo que el equilibrio entre Casos de Uso y Arquitectura se vaya logrando durante cada mini proyecto, así durante todo el proceso de desarrollo. Cada mini proyecto se puede ver como una iteración (un recorrido más o menos completo a lo largo de todos los flujos de trabajo fundamentales) del cual se obtiene un incremento que produce un crecimiento en el producto.

Una iteración puede realizarse por medio de una cascada. Se pasa por los flujos fundamentales (Requisitos, Análisis, Diseño, Implementación y Pruebas), también existe una planificación de la iteración, un análisis de la iteración y algunas actividades específicas de la iteración. Al finalizar se realiza una integración de los resultados con lo obtenido de las iteraciones anteriores.

c) Fases en el Ciclo de Desarrollo.- Este proceso de desarrollo considera que cualquier desarrollo de un sistema software debe pasar por cuatro fases que se describirán a continuación, la figura muestra las fases de desarrollo y los diversos flujos de trabajo involucrados dentro de cada fase con una representación gráfica, donde se hace mayor énfasis según la fase, cabe destacar el flujo de trabajo concerniente al negocio.

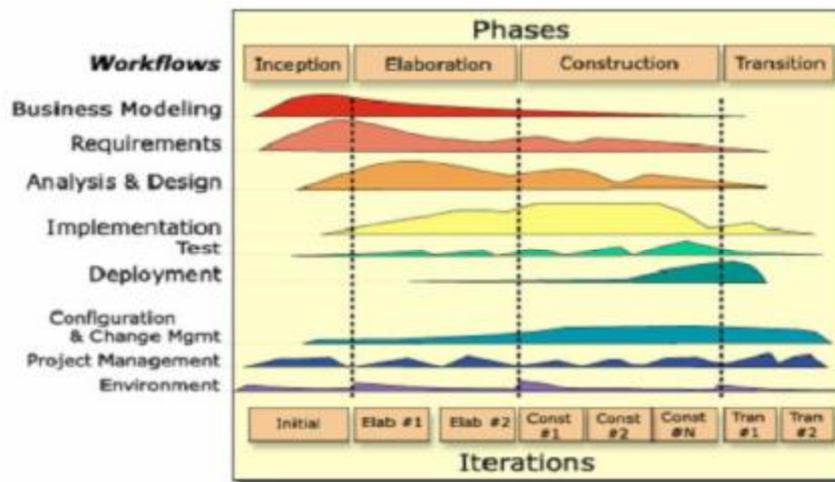


Figura 7. Fases en el Ciclo de Desarrollo de R.U.P.

- **Fase 1: Preparación Inicial (“Incepción”)**

Su objetivo principal es establecer los objetivos para el ciclo de vida del producto. En esta fase se establece el caso del negocio con el fin de delimitar el alcance del sistema, saber qué se cubrirá y delimitar el alcance del proyecto.

El caso de negocio incluye criterios de éxito, la evaluación de riesgos, y la estimación de los recursos necesarios, y un plan de la fase que muestre las fechas previstas e hitos importantes.

- **Fase 2: Preparación Detallada (“Elaboración”)**

Su objetivo principal es plantear la arquitectura para el ciclo de vida del producto. En esta fase se realiza la captura de la mayor parte de los requerimientos funcionales, manejando los riesgos que interfieran con los objetivos del sistema, acumulando la información necesaria para el plan de construcción y obteniendo suficiente información para hacer realizable el caso del negocio.

El resultado de la fase de elaboración es:

Un modelo de Caso de Uso (por lo menos 80% completo) - todos los Casos de Uso y actores deben haber sido identificados-, y se han desarrollado la mayoría de las descripciones de Casos de Uso.

Requerimientos suplementarios que capturan los requerimientos no funcionales o cualquier requerimiento que no se asocie a un caso de uso específico.

- **Fase 3: Construcción (“Construcción”)**

Su objetivo principal es alcanzar la capacidad operacional del producto. En esta fase a través de sucesivas iteraciones e incrementos se desarrolla un producto software, listo para operar,

éste es frecuentemente llamado versión beta.

- **Fase 4: Transición (“Transición”)**

Su objetivo principal es realizar la entrega del producto operando, una vez realizadas las pruebas de aceptación por un grupo especial de usuarios y habiendo efectuado los ajustes y correcciones que sean requeridos.

Éste incluye:

- Operación en paralelo con un sistema anterior que el nuevo sistema esté sustituyendo.
- La conversión de las bases de datos operacionales.
- Entrenamientos y capacitación de los usuarios y la gente de mantenimiento.

2.1.1.5.2 UML (Lenguaje Unificado de Modelado).

UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y unas reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema.

Se ha convertido en el estándar de facto de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh.

Este lenguaje nos indica cómo crear y leer los modelos, pero no dice cómo crearlos. Esto último es el objetivo de las metodologías de desarrollo.

a) Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: a partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Aunque UML está pensado para modelar sistemas complejos con gran cantidad de software, el lenguaje es lo suficientemente expresivo como para modelar sistemas que no son informáticos, como flujos de trabajo (workflow) en una empresa, diseño de la estructura de una organización y por supuesto, en el diseño de hardware.

b) Un modelo UML está compuesto por tres clases de bloques de construcción:

- Elementos: los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.).
- Relaciones: relacionan los elementos entre sí.
- Diagramas: son colecciones de elementos con sus relaciones.

c) UML es además un método formal de modelado. Esto aporta las siguientes ventajas:

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos). Esto permite que el modelo y el código estén actualizados, con lo que siempre se puede mantener la visión en el diseño, de más alto nivel, de la estructura de un proyecto.

d) UML ofrece notación y semántica estándar:

UML prescribe una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos. Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de las metodologías populares, causando a los revisores tener que aprender las semánticas y notaciones de la metodología empleada antes que intentar entender el diseño en sí.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.

En UML 2.0 hay 13 tipos diferentes de diagramas. Para comprenderlos de manera concreta, es útil categorizarlos jerárquicamente.

Los *Diagramas de Estructura* enfatizan en los elementos que deben existir en el sistema modelado:

- Diagrama de Clases, representan la estructura estática en términos de clases y relaciones.
- Diagrama de Componentes, representan los componentes físicos de una aplicación.
- Diagrama de Objetos, representan los objetos y sus relaciones, corresponden a diagramas de colaboración simplificados sin la representación del envío de mensajes.
- Diagrama de Estructura compuesta (UML 2.0)
- Diagrama de Despliegue, representan el despliegue de los componentes sobre los dispositivos físicos.
- Diagrama de Paquetes, muestra como un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones.

Los *Diagramas de Comportamiento* enfatizan en lo que debe suceder en el sistema modelado:

- Diagrama de Actividades, representan el comportamiento del sistema en términos de acciones.
- Diagrama de Casos de Uso, representan funcionalidad del sistema desde el punto de vista del usuario.
- Diagrama de Estados, representan el comportamiento de una clase en término de estados.

Los **Diagramas de Interacción** son un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado:

- Diagrama de Secuencia, son una representación temporal de los objetos y sus interacciones.
- Diagrama de Comunicación, que es una versión simplificada del Diagrama de Colaboración (UML 1.x), son una representación espacial de los objetos, enlaces e interacciones entre ellos.
- Diagrama de Tiempos (UML 2.0).
- Diagrama Global de Interacciones o Diagrama de Vista de Interacción (UML 2.0).

2.1.1.5.2.1 Tipos de Diagramas Utilizados

2.1.1.5.2.1.1 Diagramas de Clases

Es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro.

2.1.1.5.2.1.2 Diagramas de Casos de Uso

Que captura las interacciones de los Casos de Uso y los actores. Describe los requisitos funcionales del sistema, la forma en la que las cosas externas (actores) interactúan a través del límite del sistema y la respuesta del sistema.

2.1.1.5.2.1.3 Diagramas de Actividades

Que se usa para modelar el comportamiento de un sistema, y la manera en que este comportamiento está relacionado con un flujo global del sistema. Se usan los caminos lógicos que sigue un proceso basado en varias condiciones, concurrencia en el proceso, los datos de acceso, interrupciones y otras alternativas del camino lógico para construir un proceso, sistema o procedimiento.

2.1.1.5.2.1.4 Diagramas de Secuencias

Que es una representación estructurada del comportamiento como una serie de pasos secuenciales a lo largo del tiempo. Se usa para representar el flujo de trabajo, el paso de mensajes y cómo los elementos en general cooperan a lo largo del tiempo para lograr un resultado.

2.1.1.5.2.1.5 Diagrama de Paquetes

Muestra como un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.

Los paquetes están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los paquetes. Con estas líneas maestras sobre la mesa, los paquetes son buenos elementos de gestión. Cada paquete puede asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido.

2.1.1.5.2.1.6 Diagrama de Componentes

Ilustra los fragmentos de software, controladores embebidos, etc. que conformarán un sistema. Un diagrama de componentes tiene un nivel de abstracción más elevado que un diagrama de clase - usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, como así eventualmente un componente puede comprender una gran porción de un sistema.

2.1.1.5.2.1.7 Diagrama de Despliegue

Muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos, y la construcción interna puede ser representada por nodos o artefactos embebidos. Como los artefactos se ubican en los nodos para modelar el despliegue del sistema, la ubicación es guiada por el uso de las especificaciones de despliegue.

Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria.

2.1.1.6 Herramientas de Construcción de Software

2.1.1.6.1 Eclipse Ganímedes

Es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.¹

Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado *Java Development Toolkit* (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent Azureus.

Eclipse es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación

¹ http://es.wikipedia.org/wiki/Eclipse_%28software%29

cubiertas.

Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.



Figura 8. Eclipse Ganimedes

2.1.1.6.2 Macromedia Dreamweaver

Adobe Dreamweaver es una aplicación en forma de estudio (basada en la forma de estudio de Adobe Flash) enfocada a la construcción y edición de sitios y aplicaciones Web basadas en estándares.²

Creado inicialmente por Macromedia (actualmente producido por Adobe Systems). Es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium. Dreamweaver ha tenido un gran éxito desde finales de los 90 y actualmente mantiene el 90% del mercado de editores HTML.

Permite la conexión a un servidor, a base de datos, soporte para programación en ASP, PHP, Javascript, cliente FTP integrado entre otros.



Figura 9. Macromedia Dreamweaver

2.1.1.6.3 Visual Paradigm for UML

Es una herramienta “modelado visual” de UML. La herramienta está diseñada para una gama amplia de usuarios como ser ingenieros de software, analistas de sistemas, analistas comerciales y arquitectos de sistemas o para cualquiera que está interesado en construir sistemas, software de gran potencia que usan un acercamiento orientado a objeto fiablemente.

² http://es.wikipedia.org/wiki/Adobe_Dreamweaver

Además, VP-UML apoya las últimas normas de anotación de UML.



Figura 10. Visual Paradigm for UML

2.1.1.6.4 Tomcat

Tomcat (también llamado Jakarta Tomcat o Apache Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.³

Podemos dividir los contenedores de Servlets en:

Contenedores de Servlets Stand-alone (Independientes)

Estos son una parte integral del servidor web. Este es el caso cuando usando un servidor web basado en Java, por ejemplo, el contenedor de servlets es parte de JavaWebServer (actualmente sustituido por iPlanet). Éste el modo por defecto usado por Tomcat.

Sin embargo, la mayoría de los servidores, no están basados en Java, lo que nos lleva a los dos siguientes tipos de contenedores:

Contenedores de Servales dentro-de-Proceso

El contenedor Servales es una combinación de un plumín para el servidor web y una implementación de contenedor Java. El plumín del servidor web abre una JVM (Máquina Virtual Java) dentro del espacio de direcciones del servidor web y permite que el contenedor Java se ejecute en él. Si una cierta petición debería ejecutar un serles, el plumín toma el control sobre la petición y lo pasa al contenedor Java (usando JNI). Un contenedor de este tipo es adecuado para servidores multi-thread de un sólo proceso y proporciona un buen rendimiento pero está limitado en escalabilidad.

Contenedores de Servlets fuera-de-proceso

El contenedor Servlets es una combinación de un plugin para el servidor web y una implementación de contenedor Java que se ejecuta en una JVM fuera del servidor web. El plugin del servidor web y el JVM del contenedor Java se comunican usando algún mecanismo IPC (normalmente sockets TCP/IP). Si una cierta petición debería ejecutar un servlets, el plugin toma el control sobre la petición y lo pasa al contenedor Java. El tiempo de respuesta en este tipo de contenedores no es tan bueno como el anterior, pero obtiene mejores rendimientos en otras cosas (escalabilidad, estabilidad, etc.).

³ <http://es.wikipedia.org/wiki/Tomcat>

Tomcat puede utilizarse como un contenedor solitario (principalmente para desarrollo y depuración) o como plugin para un servidor web existente (actualmente se soportan los servidores Apache, IIS y Netscape). Esto significa que siempre que despluguemos Tomcat tendremos que decidir cómo usarlo, y, si seleccionamos las opciones 2 ó 3, también necesitaremos instalar un adaptador de servidor web.

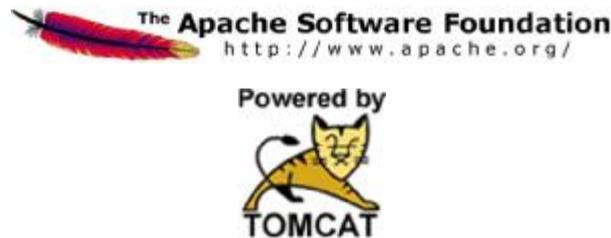


Figura 11. Tomcat

2.1.1.6.5 pgAdmin III PostgreSQL Tools

PgAdmin es un proyecto de software libre publicado bajo la licencia de PostgreSQL. El software está disponible en fuente y el formato binario de la red de servidores espejos de PostgreSQL.

Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo.

pgAdmin 3 es una herramienta de código abierto para la administración de bases de datos PostgreSQL y derivados (EnterpriseDB Postgres Plus Advanced Server y Greenplum Database).⁴ Incluye:

- Interfaz administrativa gráfica
- Herramienta de consulta SQL (con un EXPLAIN gráfico)
- Editor de código procedural
- Agente de planificación SQL/shell/batch
- Administración de Slony-I

pgAdmin se diseña para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas.

La interface gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, FreeBSD, Mac OSX y Solaris.

pgAdmin III soporta versiones de servidores 7.3 y superiores. Versiones anteriores a 7.3 deben usar pgAdmin II.

⁴ <http://www.arpug.com.ar/trac/wiki/PgAdmin>



Figura 12. PgAdmin III PostgreSQL Tools

2.1.1.6.6 iReport

iReport es una herramienta visual que sirve para generar ficheros XML (plantillas de informe) que se puedan utilizar con la herramienta de generación de informes JasperReports.

Jasper Reports es una herramienta open source para la generación de reportes basado en la tecnología JAVA, con capacidad de EXPORTACIÓN en formato PDF, XML, HTML, CSV y XLS.⁵



Figura 13. iReport

2.1.1.7 Técnica

2.1.1.7.1 HTML (*HyperText Markup Language*)

HTML (Lenguaje de Marcas de Hipertexto) es un lenguaje estático para el desarrollo de sitios web (Lenguaje de Marcas Hipertextuales). Es un lenguaje de fácil aprendizaje siendo el despliegue del mismo rápido. Así, el HTML es una aplicación del estándar ISO 8879:1986(SGML), que se formalizó en 1990 con la aparición de la Word Wide Web.

2.1.1.7.2 XML (*Markup Language*)

Es un metalenguaje para la descripción y estructuración de datos utilizando marcas (Markup Language). Metalenguaje significa un lenguaje para definir otros lenguajes: XHTML, WML, etc.

El Objetivo de XML es separar de un documento o información.

Facilidad de administración (separar contenido, lógica y presentación).

Muchas aplicaciones Web que extraen información de BD, la convierten en tablas, perdiendo información de los campos.

⁵ <http://www.buenastareas.com/ensayos/Configuracion/1272199.html>

2.1.1.7.3 CCS (Cascading Style Sheets)

Una hoja de estilo es un conjunto de instrucciones que definen la apariencia de diversos elementos de un documento HTML. En otras palabras una hoja de estilo nos permite indicar por ejemplo el tamaño de la fuente, color y estilo de cierto párrafo que nosotros indiquemos, mediante la definición de estilos y aplicación de los mismos.

Las hojas de estilo se usan porque tienen muchas ventajas sobre los tags tradicionales, ya que por ejemplo es posible crear una sola hoja de estilo que compartan muchos documentos, y al hacer un cambio en la hoja de estilo todos los documentos que la usan tendrán la apariencia deseada.

También se puede tener control sobre ciertos aspectos que antes no se tenía, por ejemplo se pueden definir los márgenes de un documento o párrafo, o definir el espacio entre caracteres.

Las hojas de estilo son la innovación más importante al HTML (también se usan en otros lenguajes como XML y SGML), ya que le dan capacidades que nunca tuvo.

El uso de diversas unidades de medición pixeles, puntos, picas, mm, en los principales elementos del HTML, como son tablas, fuentes, bordes y en general los elementos que tenían atributos como “size” “height” “width”.

El posicionamiento de bloques de texto en cualquier parte del documento HTML, ya que es posible definir en diversas unidades la posición de un bloque de texto.

Mejor control sobre las fuentes que es necesario para otras tecnologías relacionadas como las fuentes dinámicas.

El poder cambiar las características de una hoja de estilo mediante el uso de lenguajes de programación Web como “Javascript” o “VB script”.

Las hojas de estilo son la base de la implementación estándar del HTML dinámico o DHTML.

2.1.1.7.4 ExtJs

ExtJS es una biblioteca de JavaScript para el desarrollo de Aplicaciones Ricas en Internet (RIA) usando tecnologías como AJAX, DHTML y DOM.

Originalmente construida como una extensión de la biblioteca YUI, en la actualidad puede usarse como extensión para las bibliotecas jQuery y Prototype. Desde la versión 1.1 puede ejecutarse como una aplicación independiente.

ExtJS incluye widgets personalizables, buenos diseños, intuitivo, API extensa y fácil de utilizar,

licencia Libre y Comercial disponibles.⁶



Figura 14. ExtJs

2.1.1.7.5 jQuery

Query es una biblioteca o framework de Javascript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas web.⁷

jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en Javascript que de otra manera requerirían de mucho más código. Es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

jQuery usa las licencias MIT y GPL permitiendo su uso en proyectos libres y privativos.



Figura 15. jQuery

2.1.1.7.6 JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.⁸

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems.

Características:

- Javascript es un lenguaje interpretado.
- Desarrollo en web.

⁶ http://www.ecured.cu/index.php/Sencha_Ext_JS

⁷ <http://es.wikipedia.org/wiki/JQuery>

⁸ <http://www.librosweb.es/javascript/capitulo1.html>

- Ejecución de los programas en el navegador cliente: es el navegador el que interpreta las instrucciones.
- No hay intervención por parte del servidor.

2.1.1.7.7 Zk

ZK es un framework de código abierto para desarrollo de clientes ricos AJAX.⁹

ZK se creó gracias a una comunidad de desarrolladores que se han propuesto que la implementación de interfaces de usuario (GUI) en Ajax sea mucho más fácil y cómoda de implementar y de desarrollar. Su implementación está basada en lenguaje Java, pero se puede conectar con cualquier backend escrito en cualquier otro lenguaje.

Lo más relevante y apasionante es que el desarrollador se "olvida" del JavaScript en su capa de presentación, y sólo se preocupa de desarrollar. Además, se tiende a desarrollar como se hacía tradicionalmente sin manipular una gran cantidad de archivos de configuración y capas. ZK nos permite desarrollar aplicaciones web AJAX similar a como se desarrollaba en las aplicaciones de escritorio (como ser, con Swing en Java). Resulta un framework muy atractivo para todos aquellos que estén interesados en conocer las últimas tendencias en lo que a desarrollo de software se refiere.



Figura 16. Zk

Limitaciones:

ZK no es para aplicaciones que se ejecutan la mayor parte de las tareas a los clientes, tales como juegos 3D de acción. A menos que escribir un componente especial.

2.1.1.7.8 Java

Java es un lenguaje originalmente desarrollado por un grupo de ingenieros de Sun, utilizado por Netscape posteriormente como base para Java script. Si bien su uso se destaca en el Web, sirve para crear todo tipo de aplicaciones (locales, intranet o internet).¹⁰

Java es un lenguaje de objetos, independiente de la plataforma.

⁹<http://www.dosideas.com/noticias/java/718-framework-zkoss-ya-en-espanol.html/>

¹⁰<http://informatica-full2.blogspot.com/2009/06/que-es-java.html>

Algunas características notables:

- Robusto.
- Gestiona la memoria automáticamente.
- No permite el uso de técnicas de programación inadecuadas.
- Multithreading.
- Cliente-Servidor.
- Mecanismos de seguridad incorporados.
- Herramientas de documentación incorporadas.

Java posee ciertas características que hoy día se consideran estándares en los lenguajes OO:

- Objetos.
- Clases.
- Métodos.
- Subclases.
- Herencia simple.
- Enlace dinámico.
- Encapsulamiento.

Java es un lenguaje que ha sido diseñado para producir software:

- Confiable: minimiza los errores que se escapan a la fase de prueba.
- Multiplataforma: los mismos binarios funcionan correctamente en Windows/95/NT/XP/VISTA, Linux, Unix/Motif y Power/Mac.
- Seguro: Applets recuperados por medio de la red no pueden causar daño a los usuarios.
- Orientado a objetos: beneficioso tanto para el proveedor de bibliotecas de clases como para el programador de aplicaciones.
- Robusto: los errores se detectan en el momento de producirse, lo que facilita la depuración.



Figura 17. Java

2.1.1.7.9 Arquitectura y Diseño: Modelo Vista Controlador (MVC)

Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

El patrón MVC se ve frecuentemente en *aplicaciones web*, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista.¹¹

Para el diseño de aplicaciones con sofisticados interfaces se utiliza el patrón de diseño Modelo-Vista-Controlador. La lógica de un interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Si realizamos un diseño ofuscado, es decir, que mezcle los componentes de interfaz y de negocio, entonces la consecuencia será que, cuando necesitemos cambiar el interfaz, tendremos que modificar trabajosamente los componentes de negocio. Mayor trabajo y más riesgo de error.

A todo esto, se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

Los elementos del patrón son tres:

- Modelo: datos y reglas de negocio.
- Vista: muestra la información del modelo al usuario.
- Controlador: gestiona las entradas del usuario.

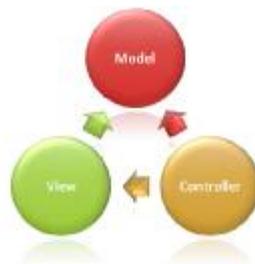


Figura 18. Modelo Vista Controlador (MVC)

2.1.1.7.9.1 Concepto : Patrón MVC

A alto nivel, el patrón MVC permite separar los distintos aspectos de una aplicación (Inputs Lógicos, Lógica de Negocio, Lógica de UI), otorgando un bajo acoplamiento entre estos elementos. Esto permite que al momento de construir una aplicación, nos focalicemos sólo en un aspecto, como por ejemplo en una vista, sin tener dependencia de la lógica del negocio.

Además el desacoplamiento de sus componentes, permite que los distintos aspectos se pueda desarrollar en paralelo. Así, un desarrollador se puede abocar a la vista, un segundo desarrollador a

¹¹ http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

la lógica del controlador y otro focalizarse en la lógica de negocio en el modelo.

2.1.1.7.9.2 Ventajas de Usar MVC

- Permite tener completo control sobre el comportamiento de una aplicación
- Testear una aplicación se vuelve más fácil, al no tener que instanciar la clase de la página individual, sus controles anidados y clases adicionales que dependan de su uso para el testeo.
- Soporta muy bien aplicaciones que requieran de un gran equipo de desarrollo, donde se necesita un gran nivel de profundidad y modularización, otorgando un completo control sobre el módulo asignado, independiente del resto.

2.1.1.7.9.3 Componentes Patrón MVC

- **Modelo:** corresponde a la parte de la aplicación que implementa la lógica del Dominio de Datos de la aplicación, además de incorporar la persistencia de datos. Frecuentemente, los objetos del modelo recuperan y almacenan los estados del modelo en la base de datos. Aquí se hace el levantamiento de los objetos que el sistema deba utilizar y es el proveedor de los recursos al Controlador
- **Vista:** es el componente que despliega la interfaz de usuario de aplicación. Normalmente (no siempre), esta UI es creada a partir de los datos del modelo (Un ejemplo puede ser una vista de edición de una tabla de Productos que muestra cuadros de texto, listas desplegables y casillas de verificación basado en el estado actual de los objetos Producto). Esta puede ser una web HTML, un XML, un archivo binario, etc.
- **Controlador:** el controlador maneja la interacción del usuario, trabaja con el modelo, y por último selecciona la vista a renderizar en la interfaz de usuario. En una aplicación MVC, la vista sólo despliega información; el controlador maneja y responde los inputs e interacción del usuario. Éste escucha los cambios a la vista y se los envía al modelo.

Un ciclo a alto nivel se desarrolla de la siguiente forma: el usuario envía peticiones (a la vista) las cuales son recibidas por el lado servidor con el controlador, éste sabe cómo y quién se debe comunicar con el Modelo para responder ante el requerimiento. La respuesta del controlador finalmente es una vista, o una vista parcial.

Muchos sistemas informáticos que utilizan un Sistema de Gestión de Base de Datos para gestionar los datos: en líneas generales del **MVC** que corresponde al modelo. La unión entre *capa de presentación* y *capa de negocio* conocido en el paradigma de la Programación por capas representaría la integración entre **Vista** y su correspondiente **Controlador** de eventos y acceso a datos, MVC no pretende discriminar entre capa de negocio y capa de presentación pero si pretende separar la capa *visual gráfica* de su correspondiente *programación y acceso a datos*, algo que mejora el desarrollo y mantenimiento

de la *Vista* y el *Controlador* en paralelo, ya que ambos cumplen ciclos de vida muy distintos entre sí.

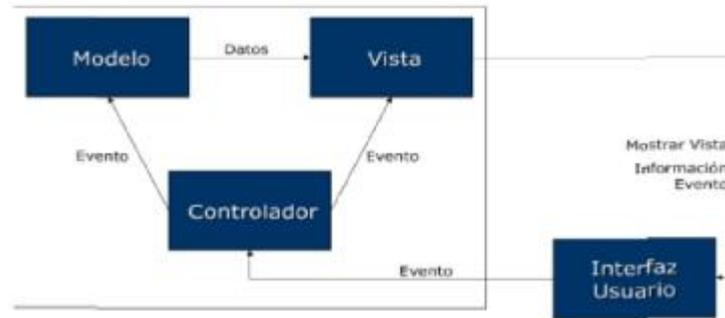


Figura 19. Patrón MVC

2.1.1.7.10 Servlet

Los servlets son objetos que corren dentro del contexto de un contenedor de servlets (ej.: Tomcat) y extienden su funcionalidad. También podrían correr dentro de un servidor de aplicaciones (ej.: OC4J Oracle), que, además de contenedor para servlet, tendrá contenedor para objetos más avanzados, como son los EJB (Tomcat sólo es un contenedor de servlets).¹²

La palabra servlet deriva de otra anterior, applet, que se refería a pequeños programas escritos en Java que se ejecutan en el contexto de un navegador web. Por contraposición, un servlet es un programa que se ejecuta en un servidor.

El uso más común de los servlets es generar páginas web de forma dinámica a partir de los parámetros de la petición que envía el navegador web.

Un servlet implementa la interfaz `javax.servlet.Servlet` o hereda alguna de las clases más convenientes para un protocolo específico (ej: `javax.servlet.HttpServlet`). Al implementar esta interfaz el servlet es capaz de interpretar los objetos de tipo `HttpServletRequest` y `HttpServletResponse` quienes contienen la información de la página que invocó al servlet.

2.1.1.7.11 Spring

Spring es un framework de aplicaciones Java/J2EE desarrollado usando licencia de OpenSource.

Lo primero que hay que entender de Spring, es que es un conjunto de módulos, de los cuales podemos utilizar los que queramos.¹³

¹²<http://www.vagos.es/showthread.php?t=62737>

¹³<http://www.consultoriajava.com/tools/spring.shtml>

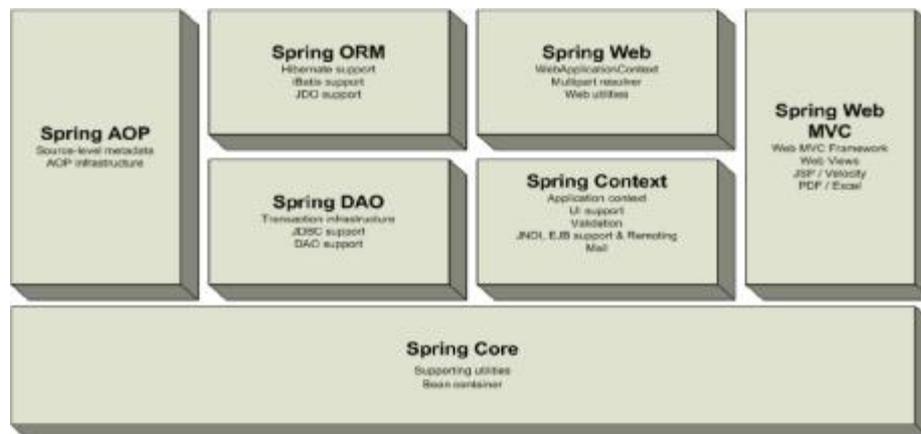


Figura 20. Arquitectura de Spring

En el proyecto utilizaremos el módulo Spring Web MVC para el desarrollo de aplicaciones Java basadas en Web construido sobre el núcleo de Spring. Dos de los objetivos más importantes de Spring MVC es permitir que el desarrollo se concentre en la lógica del negocio y que se haga empleando buenos principios de diseño orientado a objetos.

Para lograrlo se utiliza un concepto muy interesante llamado Inversión del Control (IoC), también conocido como el principio Hollywood: “No nos llames, nosotros te llamaremos.” Esto permite que el código escrito por los desarrolladores para la lógica principal del sistema no tenga dependencias sobre las clases del *framework*; lo cual redundará en un código mucho más limpio y con la posibilidad de utilizar todas las ventajas de la programación orientada a objetos (específicamente la herencia).

2.1.1.7.12 Hibernate

Hibernate ofrece la *Persistencia Relacional para Java*, que para los no iniciados, proporciona unas muy buenas maneras para la persistencia de sus objetos de Java a y desde una base de datos subyacente. Más que ensuciar con SQL tus objetos y convertir consultas a y desde los objetos de primera magnitud, Hibernate puede preocuparse de todo ese maremágnum por ti. Tú utilizas solamente a los objetos, Hibernate se preocupa del SQL y de que las cosas terminan en la tabla correcta.¹⁴

Permite trabajar con software orientado a objetos y bases de datos relacionales puede hacernos invertir mucho tiempo en los entornos actuales.

Hibernate es una herramienta que realiza el *mapping* entre el mundo orientado a objetos de las aplicaciones y el mundo entidad-relación de las bases de datos en entornos Java. El término utilizado es ORM (object / relational mapping) y consiste en la técnica de realizar la transición de una representación de los datos de un modelo relacional a un modelo orientado a objetos y viceversa.

¹⁴ http://www.javahispano.org/contenidos/archivo/75/hibernate_1.pdf

Hibernate no sólo realiza esta transformación sino que nos proporciona capacidades para la obtención y almacenamiento de datos de la base de datos que nos reducen el tiempo de desarrollo.

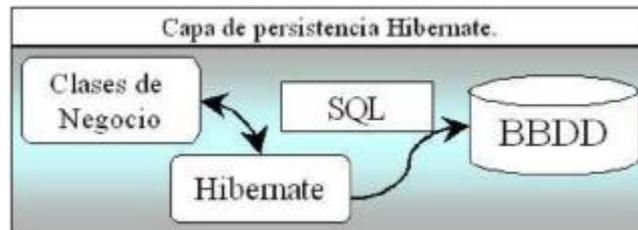


Figura 21. Arquitectura de Hibernate

Hibernate funciona asociando a cada tabla de la base de datos un Plain Old Java Object (POJO, a veces llamado Plain Ordinary Java Object). Un POJO es similar a una Java Bean, con propiedades accesibles mediante métodos setter y getter.

Algunas de las características se mencionan a continuación:

- Opensource (LGPL).
- Una tecnología muy madura.
- Un API personalizable.
- Persistencia utilizando JavaBeans.
- Consultar flexibles y poderosas.
- Trabaja los objetos persistentes en transacciones separadas.
- Provee el HQL que es un lenguaje de Consultas orientadas a Objetos muy flexible.
- Provee un mecanismo de peticiones a través de criterios.
- Soporta queries de SQL nativas.
- Soporta las operaciones relacionales.
- Inner/outer/full joins.
- Agregaciones (max, avg) y agrupamientos (Group).
- Ordenamientos (Order By).
- Consultas anidadas.

De una manera muy rápida y optimizada podremos generar BBDD en cualquiera de los entornos soportados: Oracle, DB2, MySql, PostgreSQL, etc.

Motores de persistencia:

Un motor de persistencia es un componente software encargado de traducir entre objetos (de un programa orientado a objetos) y registros (de la base de datos relacional).

Es decir, es el encargado de que el programa y la base de datos se “entiendan”.

Esto nos da dos ventajas importantes:

- Podemos programar con orientación a objetos, aprovechando las ventajas de flexibilidad, mantenimiento y reusabilidad.
- Y además podemos usar una base de datos relacional, aprovechándonos de su madurez y su estandarización así como de las herramientas relacionales que hay para ella.

Tipos de relaciones (Componentes y Colecciones):

En todo diseño relacional los objetos se referencian unos a otros a través de relaciones, las típicas son:

- Uno a Uno (One-To-One) 1 – 1
- Uno a Muchos (One-To-Many) 1 – N
- Muchos a Muchos (Many-To-Many) N – M

El Proceso del Desarrollo:

Hay varias maneras de acercarse al desarrollo con Hibérnate. Aquí está el que estamos utilizando nosotros hoy, porque es probablemente el más simple de entender:

- Crea tu tabla del SQL para guardar tus objetos persistentes.
- Crea un JavaBean que represente ese objeto en código.
- Crea un archivo de mapeo de manera que Hibérnate sepa qué características del bean se mapean a que campos del SQL.
- Crea un archivo de propiedades de manera que Hibérnate conozca la configuración JDBC para acceder a la base de datos.
- Comenzar a usar el Hibérnate API.



Figura 22. Hibérnate

2.1.1.8 Base de Datos

2.1.1.8.1 Definición de Base de Datos

Una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

Entre las principales características de los sistemas de Base de Datos podemos mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.

- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

Los Sistemas de Gestión de Base de Datos (en inglés DataBase Management System) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.¹⁵

2.1.1.8.2 Herramienta PostgreSQL 8.3 para el Manejo de la BD

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la universidad de Berkeley.

PostgreSQL es una derivación libre (OpenSource), y utiliza el lenguaje SQL92/SQL99, así como otras características.

PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, PostgreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos.

Algunas de las características para hacer uso de este gestor se mencionan a continuación:

- *Alta concurrencia.* Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés), PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.
- *Soporte para una amplia variedad de tipos nativos.* PostgreSQL provee nativamente soporte para: números de precisión arbitraria, texto de largo ilimitado, figuras geométricas (con una variedad de funciones asociadas), direcciones IP (IPv4 e IPv6), bloques de direcciones estilo CIDR, direcciones MAC, Arrays.

Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL. Algunos ejemplos son los tipos de datos

¹⁵ <http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/>

GIS creados por el proyecto PostGIS.

Consta con un lenguaje propio llamado [PL/PgSQL (similar al PL/SQL de oracle), pero también soporta los lenguajes C, C++, Java PL/Java web, PL/Perl, plPHP, PL/Python, PL/Ruby, PL/sh, PL/Tcl, PL/Scheme.

PostgreSQL es un magnífico gestor de bases de datos. Tiene prácticamente todo lo que tienen los gestores comerciales, haciendo de él una muy buena alternativa GPL para el presente proyecto.

Postgres cuenta con Herramientas de Administración como es PgAdmin3 y PgAccess: Entorno de escritorio visual, PhpPgAdmin y psql que es un Cliente de consola.

2.1.1.9 Sistema de Información Automatizado

2.1.1.9.1 El Internet

Algunos definen Internet como "La Red de Redes", y otros como "La Autopista de la Información".

Efectivamente, Internet es una Red de Redes porque está hecha a base de unir muchas redes locales de ordenadores, o sea de unos pocos ordenadores en un mismo edificio o empresa.

Por la Red Internet circulan constantemente cantidades increíbles de información. Por este motivo se le llama también La Autopista de la Información. Hay 50 millones de "Internautas", es decir, de personas que "navegan" por Internet en todo el mundo. Se dice "navegar" porque es normal el ver información que proviene de muchas partes distintas del mundo en una sola sesión.

Una de las ventajas de Internet es que posibilita la conexión con todo tipo de ordenadores, desde los personales, hasta los más grandes que ocupan habitaciones enteras. Incluso podemos ver conectados a la red cámaras de vídeo, robots, y máquinas de refrescos, etc.

2.1.1.9.2 La World Wide Web o la Web

La World Wide Web consiste en ofrecer una interface simple y consistente para acceder a la inmensidad de los recursos de Internet. Es la forma más moderna de ofrecer información. El medio más potente. La información se ofrece en forma de páginas electrónicas.

El World Wide Web o WWW o W3 o simplemente Web, permite saltar de un lugar a otro en pos de lo que no interesa. Lo más interesante es que con unas pocas órdenes se puede mover por toda la Internet.

La World Wide Web permite una manera más organizada de acceder a la información disponible en Internet, presentando una interfaz amigable con el usuario mediante navegadores como Netscape, Mosaic y Microsoft Internet Explorer, Mozilla FireFox.

El surgimiento de la World Wide Web ha ayudado a un crecimiento considerable de Internet en la actualidad. Compañías pequeñas, empresas grandes, ayuntamientos, estados, gobiernos de distintos

países, universidades, bibliotecas, están presentes en Internet.

Es básicamente un medio de comunicación de texto, gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte o desde otro punto de vista, una forma gráfica de explorar Internet.

2.1.1.10 Sistemas de Información vía Web

La evolución de Internet como red de comunicación global y el surgimiento y desarrollo del Web como servicio imprescindible para compartir información, creó un excelente espacio para la interacción del hombre con la información hipertextual, a la vez que sentó las bases para el desarrollo de una herramienta integradora de los servicios existentes en Internet. Los sitios Web, como expresión de sistemas de información, deben poseer los siguientes componentes:

- Usuarios.
- Mecanismos de entrada y salida de la información.
- Almacenes de datos, información y conocimiento.
- Mecanismos de recuperación de información.

Pudiésemos definir entonces como sistema de información al conjunto de elementos relacionados y ordenados, según ciertas reglas que aporta al sistema objeto, es decir, a la organización a la que sirve y que marca sus directrices de funcionamiento, la información necesaria para el cumplimiento de sus fines; para ello, debe recoger, procesar y almacenar datos, procedentes tanto de la organización como de fuentes externas, con el propósito de facilitar su recuperación, elaboración y presentación. Actualmente, los sistemas de información se encuentran al alcance de las grandes masas de usuarios por medio de Internet; así se crean las bases de un nuevo modelo, en el que los usuarios interactúan directamente con los sistemas de información para satisfacer sus necesidades de información.

2.1.2 Plan de Desarrollo de Software

2.1.2.1 Introducción

El presente documento es un Plan de Desarrollo del Software que sentará las bases para el desarrollo del proyecto, es una versión preparada para ser incluida en la propuesta elaborada en respuesta al proyecto de la asignatura de Taller III de la Carrera de Ingeniería Informática de la Facultad de Ciencias y Tecnología de la Universidad Autónoma “Juan Misael Saracho”. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto ha sido propuesto por el universitario Omar Elías Yana Cerezo, basado en la Metodología RUP (Rational Unified Process), en la que se procederá a cumplir con las cuatro fases que marca la metodología. Es importante destacar esto puesto que utilizaremos la terminología RUP en este

documento. Se incluirá el detalle para las fases de Inicio y Elaboración y adicionalmente se esbozarán las fases posteriores de Construcción y Transición para dar una visión global de todo proceso.

El enfoque desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

2.1.2.1.1 Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

- El Director del proyecto, que lo utiliza para organizar la agenda y necesidades de recursos y para realizar su seguimiento.
- El Docente para evaluar el cumplimiento del proyecto.

2.1.2.1.2 Alcance

Aplicando el Plan de Desarrollo Software obtenemos una herramienta importante para realizar nuestro plan de trabajo el cual coadyuvará al cumplimiento de nuestros objetivos en el tiempo propuesto gracias al cronograma de actividades establecido.

2.1.2.1.3 Resumen

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

Planes y Guías de Aplicación — proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

2.1.2.2 Vista General del Proyecto

2.1.2.2.1 Propósito, Alcance y Objetivos

La información que a continuación se incluye ha sido extraída de las diferentes reuniones, entrevistas y charlas que se han realizado con los usuarios destino de la Caja Nacional de Salud – Administración Regional Tarija desde el inicio del proyecto.

2.1.2.2.1.1 Propósito

Con el presente proyecto se pretende mejorar “Procesos de Registros en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija”.

2.1.2.2.1.2 Alcance

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo de SIMA “Sistema Informático para el Mejoramiento de Afiliación”, un Sistema Informático que está comprendido por los siguientes módulos:

- **Módulo Administrador.**
- **Módulo Afiliación.**
- **Módulo Reportes.**
- **Módulo Backup.**

Analizando factores predominantes dentro de la institución, podemos llegar a la conclusión de que la implementación del sistema automatizado dará beneficios claramente perceptibles, dando soluciones a problemas que arrastra; este sistema proyectará una solución a mediano plazo que beneficiará a dicha institución, pero lo más importante mejorará el sistema actual.

Esta propuesta de sistema (Software) contiene una serie de alternativas de mejoramiento para las expectativas futuras de la institución, las cuales se detallan a continuación:

- Brindar seguridad al sistema mediante una clave de ingreso, permitiendo el acceso al mismo sólo al personal autorizado.
- Opciones que permitan el registro computarizado consistente de empresas empleadoras, asegurados, beneficiarios, además de realizar copia de seguridad de datos.
- Desarrollar un manual de usuario y de instalación del sistema.

2.1.2.2.1.3 Objetivos

2.1.2.2.1.3.1 Objetivo General

Mejorar los “Procesos de Registros en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija”.

2.1.2.2.1.3.2 Objetivos Específicos

- Llegar a brindar un sistema rápido y eficiente para la manipulación de registros.
- Conseguir mayor organización en los registros.
- Obtener un sistema que permita generar reportes de forma rápida y sencilla.
- Otorgar a los usuarios total seguridad en el resguardo de su información, permitiendo el acceso

sólo a usuarios autorizados.

- Crear una base de datos de acuerdo a las necesidades de la institución.
- Diseñar una interfaz de usuario de fácil manejo.

2.1.2.2.2 Suposiciones y Restricciones

Acontecimientos que deben ocurrir para que el proyecto sea ejecutado con éxito pero que están totalmente fuera del ámbito del control del equipo de proyecto.

2.1.2.2.2.1 Suposiciones

Suponemos que la Caja Nacional de Salud, Regional Tarija cuenta con:

- Equipo de computación. En el que incluye un computador, impresora.
- Un ambiente adecuado para la implementación del Sistema.
- La disponibilidad de otorgar toda la información necesaria para la creación del software.
- Personal con conocimientos básicos de computación.
- Cuenta con equipos en los cuales se podría implementar el Sistema.
- Presupuesto suficiente para la implementación del Sistema.
- Disponibilidad de un departamento (Sistemas) encargado de la administración del Sistema.
- Interés de la institución para utilizar el Sistema.
- La formación del personal encargado se lleve en la fecha determinada.
- Se presentarán en la fecha indicada o prescrita por el docente los diferentes documentos como el Plan de Desarrollo de Software y los distintos entregables descritos en el presente documento.

2.1.2.2.2.2 Restricciones

Limitaciones generalmente fuera del ámbito de control del equipo de proyecto que pueden afectar negativamente a su alcance.

- El sistema será restringido, sólo usuarios privilegiados podrán acceder al sistema.
- Para la manipulación de la base de datos, sólo podrá acceder el personal autorizado.
- Deberá contar como mínimo con un sistema operativo Windows XP/Vista/Seven o Linux.
- El administrador deberá contar con el conocimiento necesario para el manejo del sistema.
- No se incluirá ningún reporte contable.
- El sistema no abarca el control de la asistencia de los empleados de la institución.
- No contar con un Servidor Web que presente las características necesarias para la ejecución del Sistema.
- No contar con los fondos suficientes para llevar a cabo la capacitación a los administradores

del Sistema.

2.1.2.2.3 Entregables del Proyecto

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es necesario destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

2.1.2.2.3.1 Plan de Desarrollo del Software.

Es el presente documento.

2.1.2.2.3.2 Modelo de Casos de Uso del Negocio

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.) permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

La definición del conjunto de procesos del negocio es una tarea crucial, ya que define los límites del proceso de modelado posterior, consideramos los objetivos estratégicos de la organización, teniendo en cuenta que esos objetivos serán descompuestos en un conjunto de subobjetivos más concretos, para la identificación de procesos de negocio. Se presentan los modelos definidos en RUP como modelo del negocio (modelo de casos de Uso del Negocio y de Objetos del Negocio).

2.1.2.2.3.2.1 Introducción

El Modelo de Caso de Uso del Negocio es un artefacto de la disciplina requisitos en la metodología RUP la cual estamos implementando.

2.1.2.2.3.2.2 Propósito

Comprender la estructura y la dinámica de la organización.

Comprender problemas actuales e identificar posibles mejoras.

2.1.2.2.3.2.3 Alcance

Describe los Procesos de Negocio y los Clientes.

Identifica y describe los Procesos de Negocio según los Objetivos de la Organización.

Definir un Caso de Uso del Negocio para cada proceso de negocio.

2.1.2.2.3.2.4 Proceso del Negocio (BPMN)

2.1.2.2.3.2.4.1 Diagrama de Procesos de Negocio Administrador

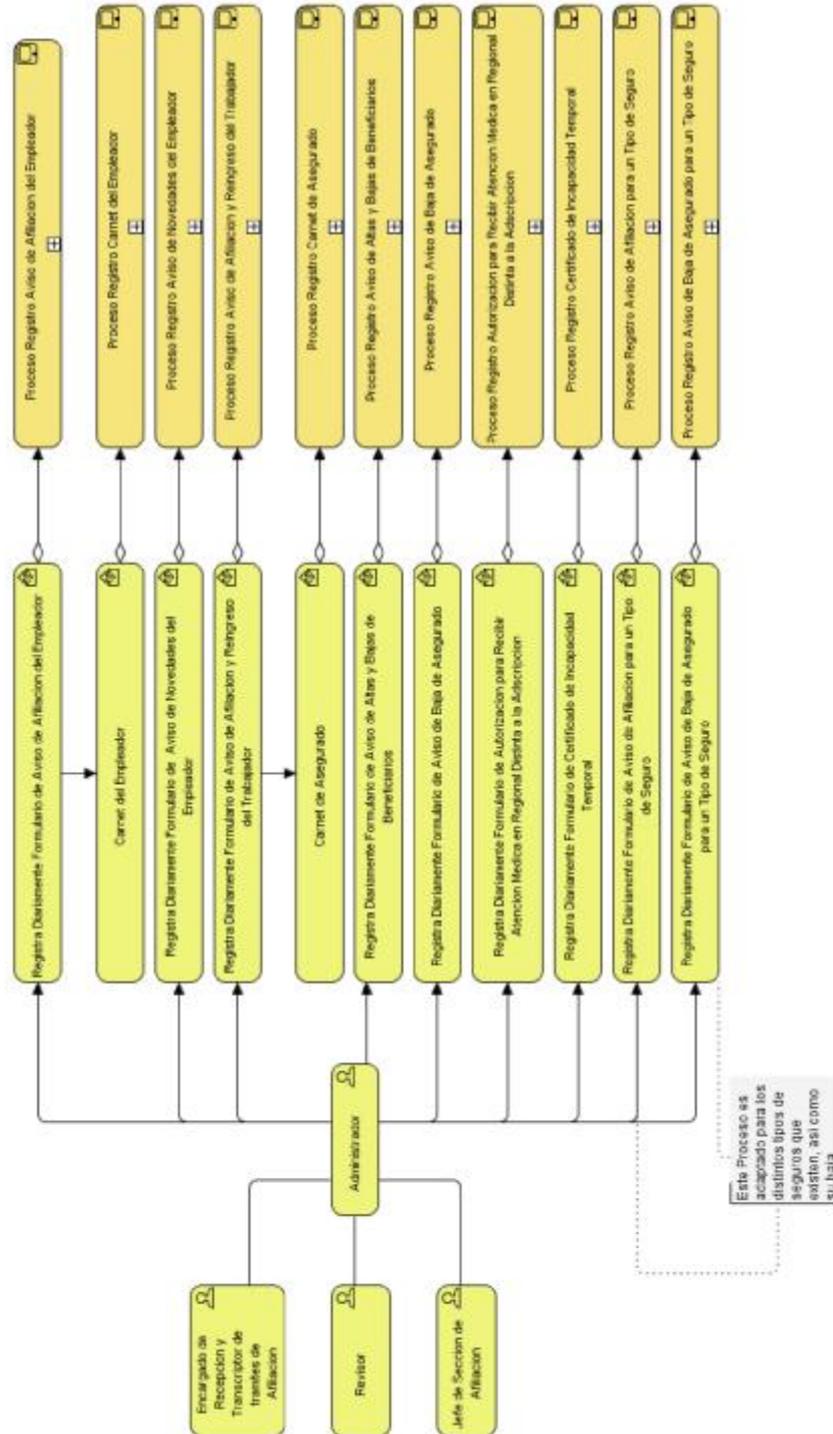


Figura 23. Diagrama de Procesos de Negocio Administrador

2.1.2.2.3.2.4.2 Diagrama de Procesos de Negocio Cliente

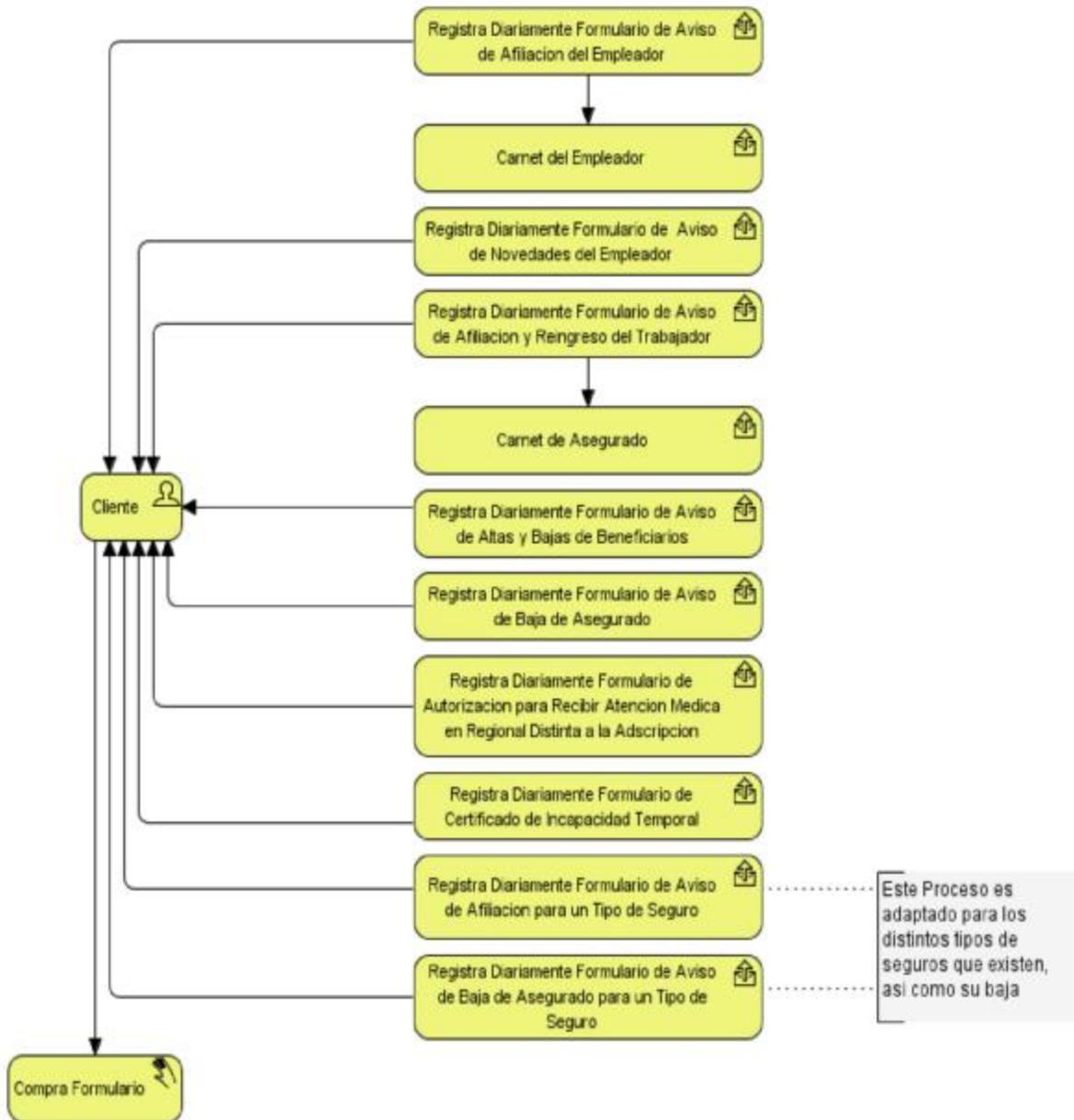


Figura 24. Diagrama de Procesos de Negocio Cliente

2.1.2.2.3.2.4.3 Proceso Registro Aviso de Afiliación del Empleador

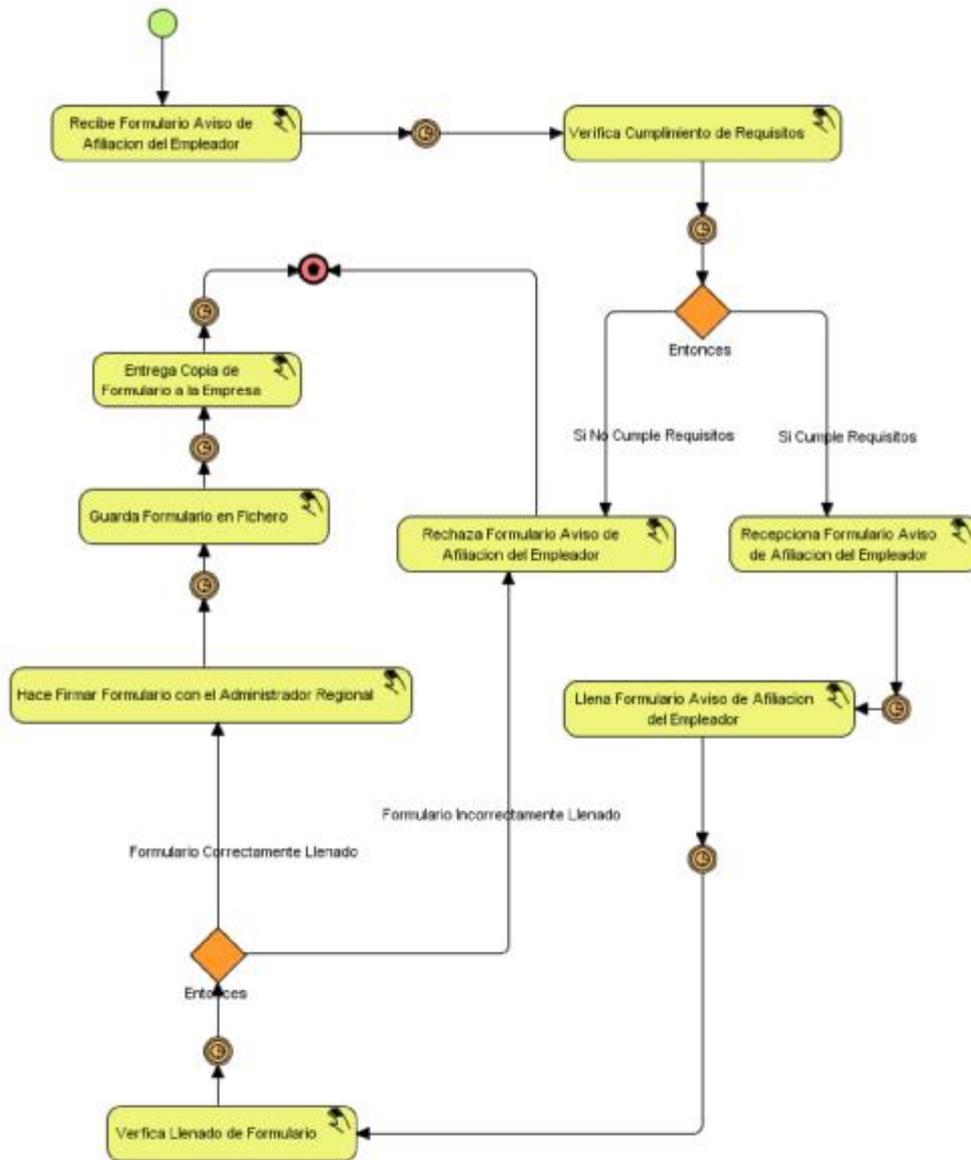


Figura 25. Proceso Registro Aviso de Afiliación del Empleador

2.1.2.2.3.2.4.4 Proceso Registro Carnet del Empleador

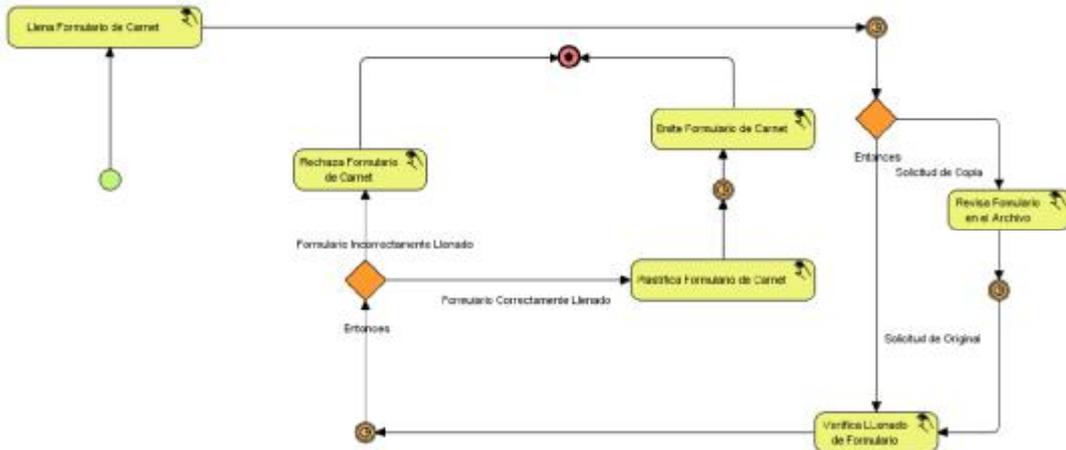


Figura 26. Proceso Registro Carnet del Empleador

2.1.2.2.3.2.4.5 Proceso Registro Aviso de Novedades del Empleador

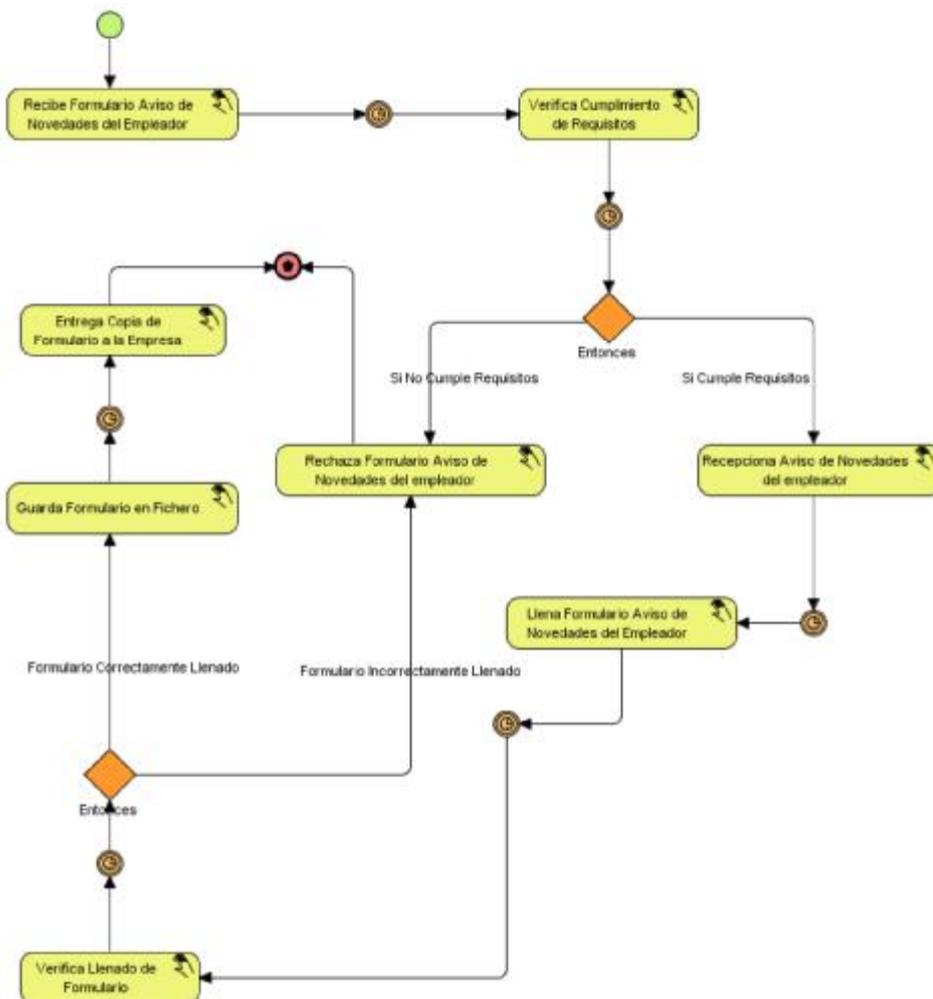


Figura 27. Proceso Registro Aviso de Novedades del Empleador
2.1.2.2.3.2.4.6 Proceso Registro Carnet de Asegurado

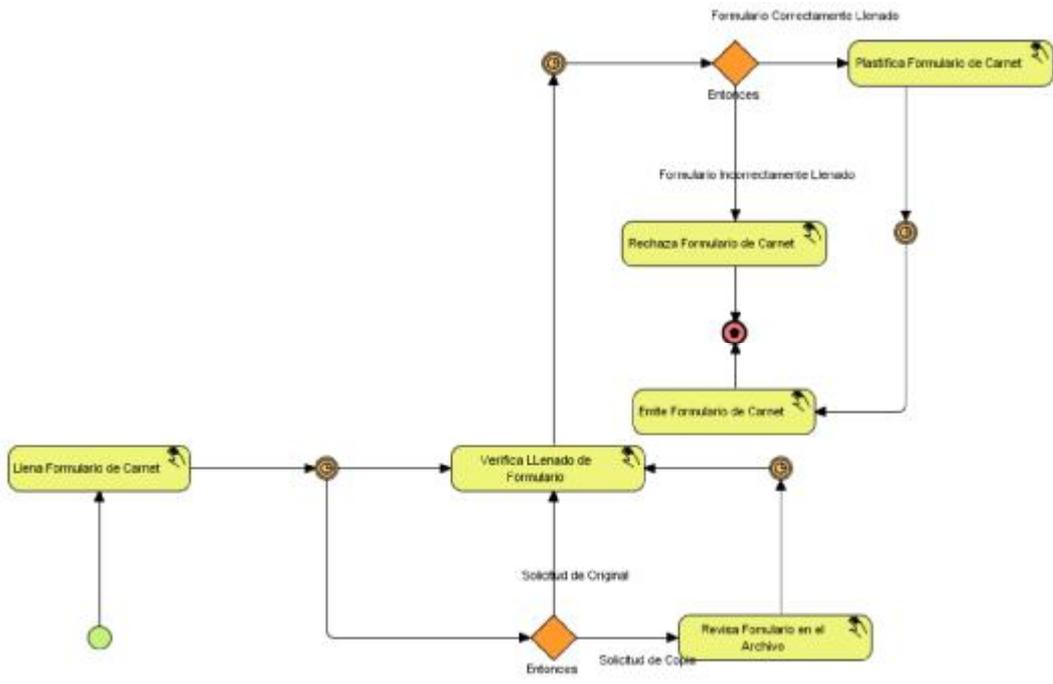


Figura 28. Proceso Registro Carnet de Asegurado

2.1.2.2.3.2.4.7 Proceso Registro Aviso de Afiliación y Reingreso del Trabajador

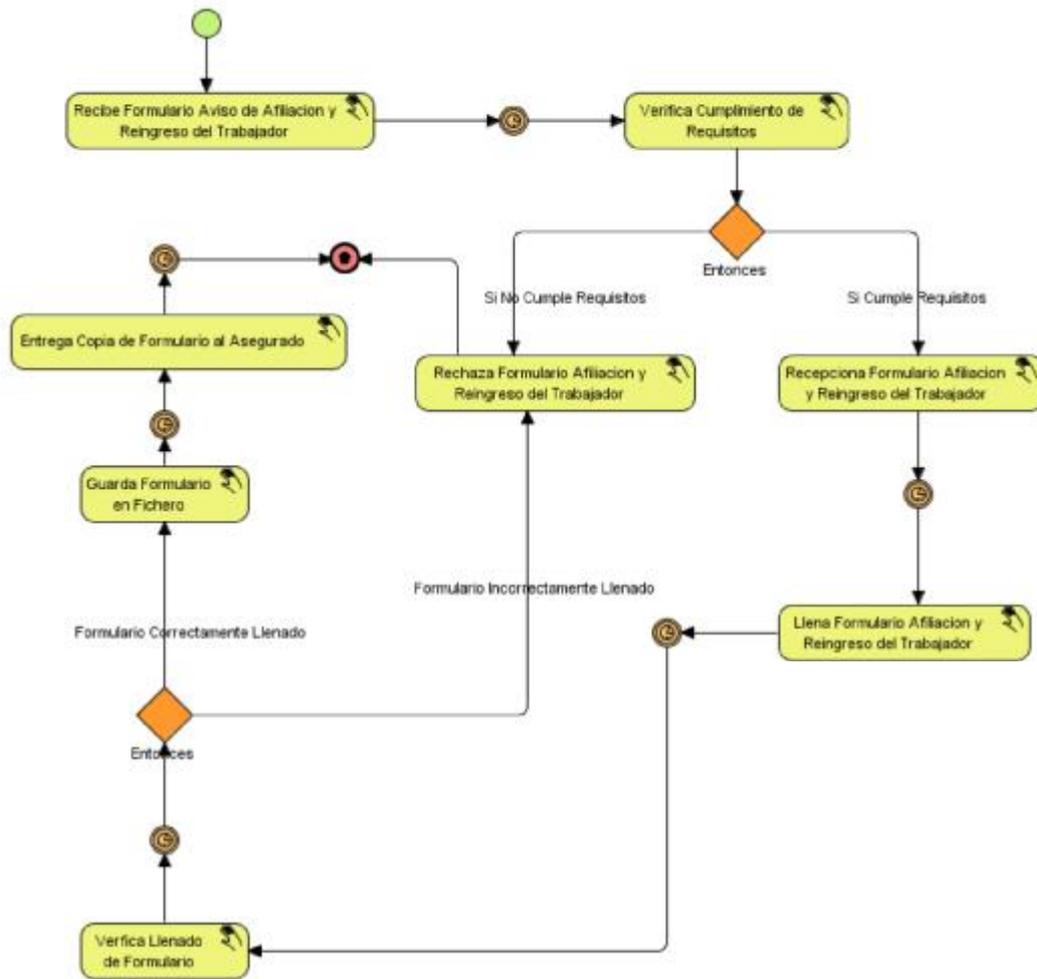


Figura 29. Proceso Registro Aviso de Afiliación y Reingreso del Trabajador

2.1.2.2.3.2.4.8 Proceso Registro Aviso de Altas y Bajas de Beneficiarios

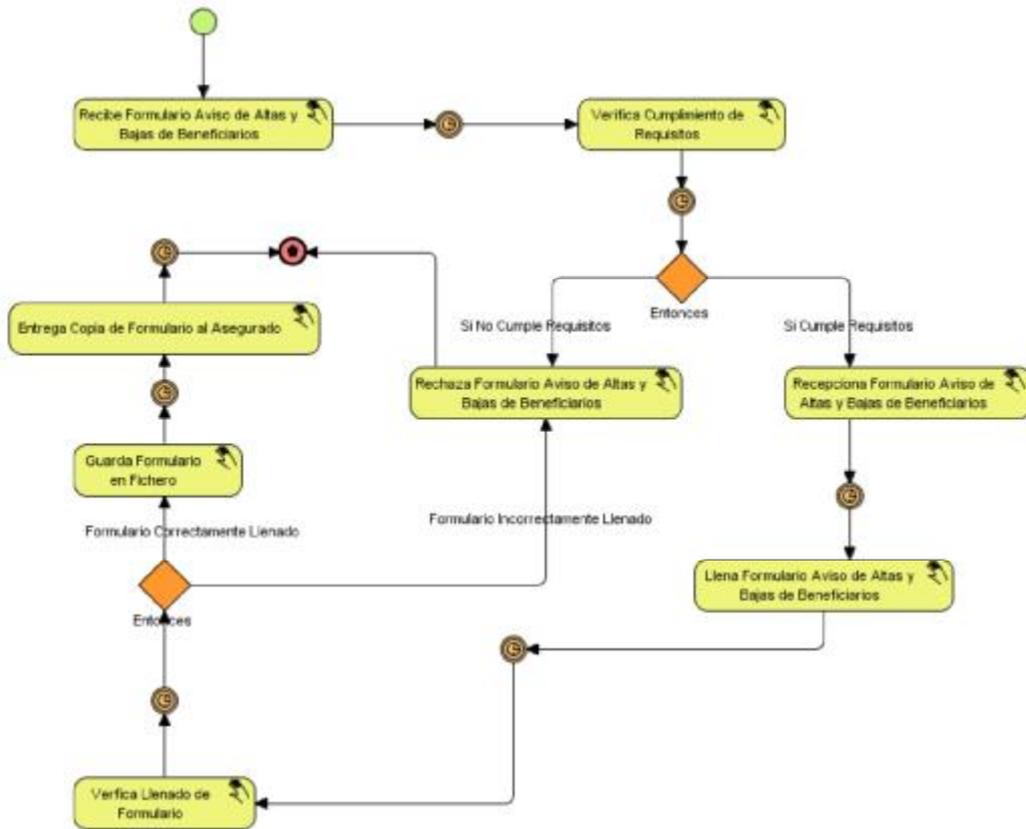


Figura 30. Proceso Registro Aviso de Altas y Bajas de Beneficiarios

2.1.2.2.3.2.4.9 Proceso Registro Aviso de Baja de Asegurado

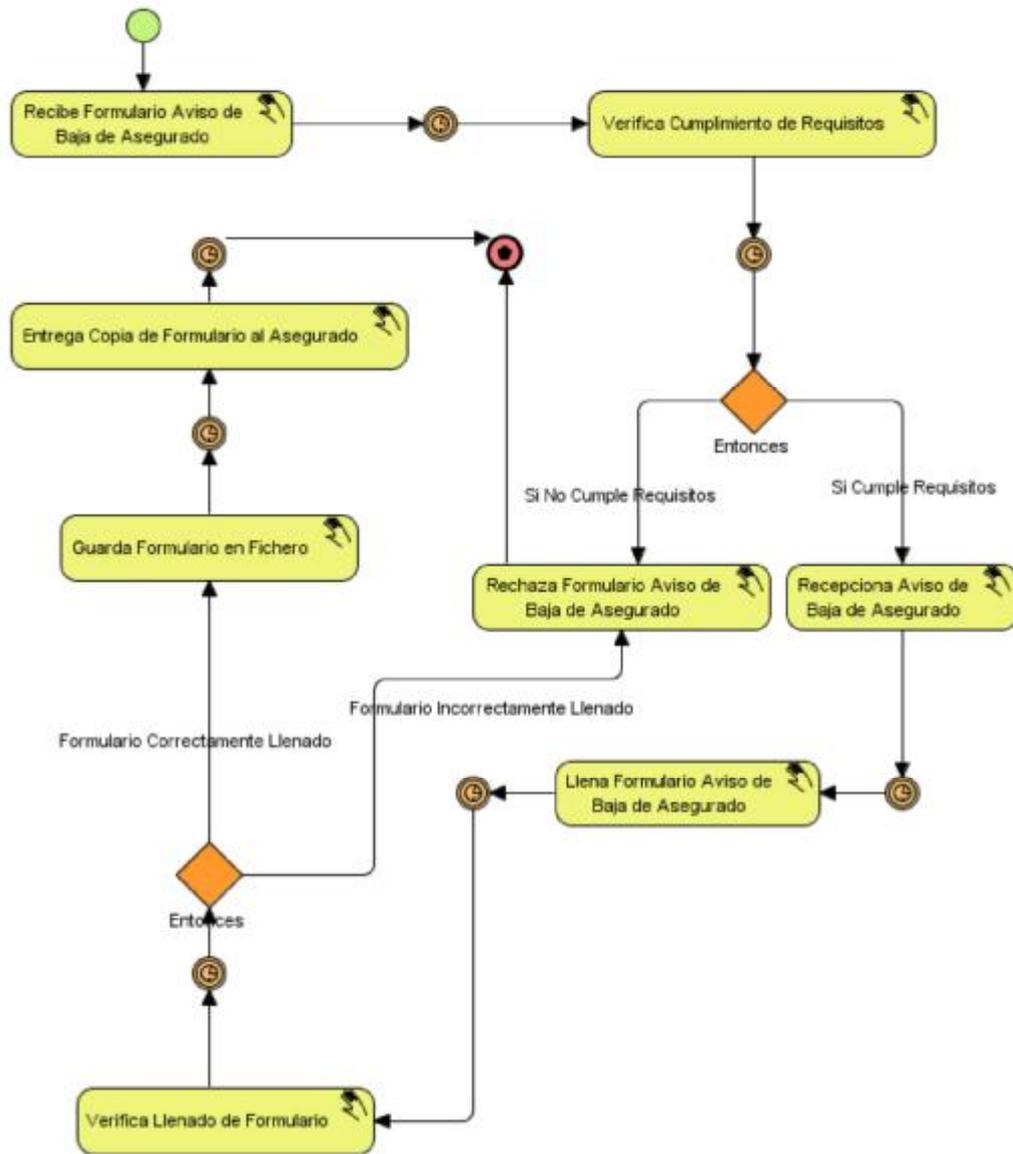


Figura 31. Proceso Registro Aviso de Baja de Asegurado

2.1.2.2.3.2.4.10 Proceso Registro Autorización para Recibir Atención Médica en Regional Distinta a la Adscripción

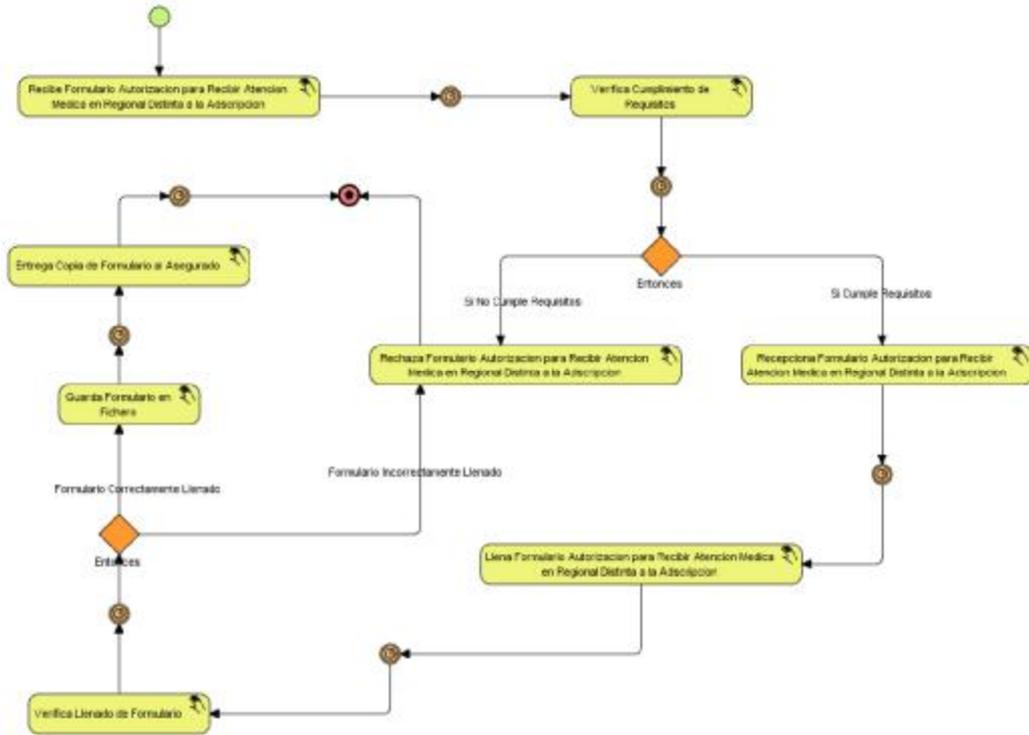


Figura 32. Proceso Registro Autorización para Recibir Atención Médica en Regional Distinta a la Adscripción

2.1.2.2.3.2.4.11 Proceso Registro Certificado de Incapacidad Temporal

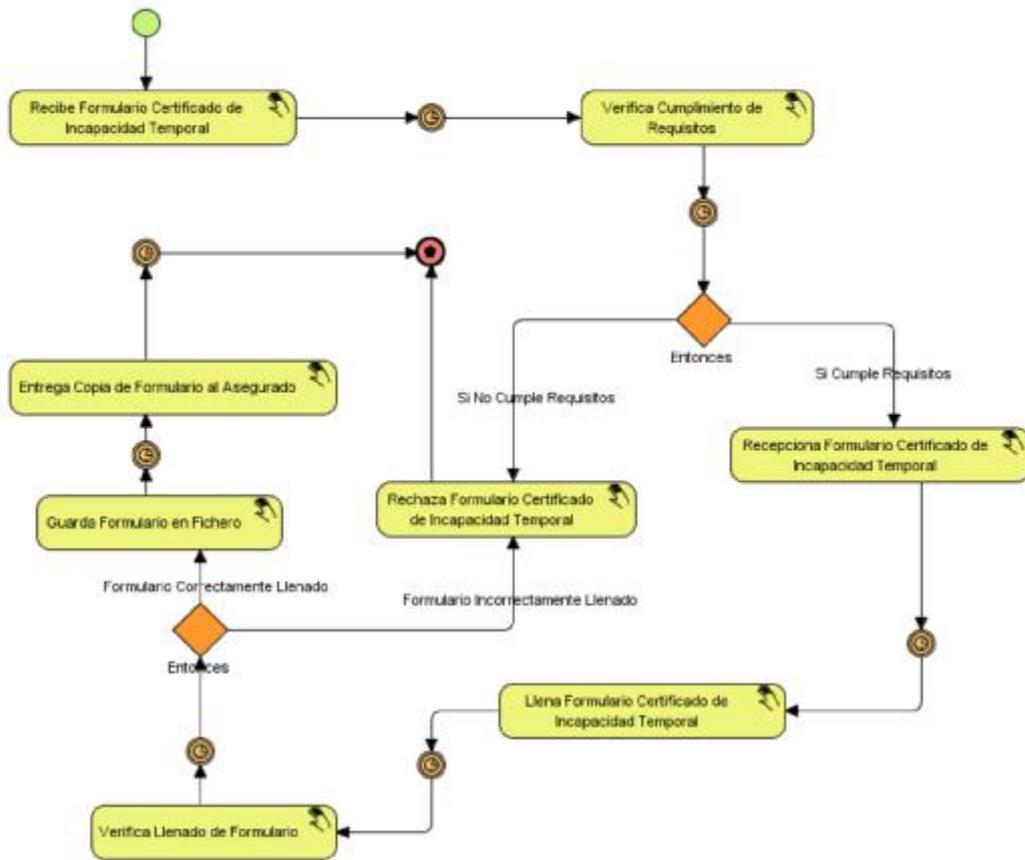


Figura 33. Proceso Registro Certificado de Incapacidad Temporal

2.1.2.2.3.2.4.12 Proceso Registro Aviso de Afiliación para un Tipo de Seguro

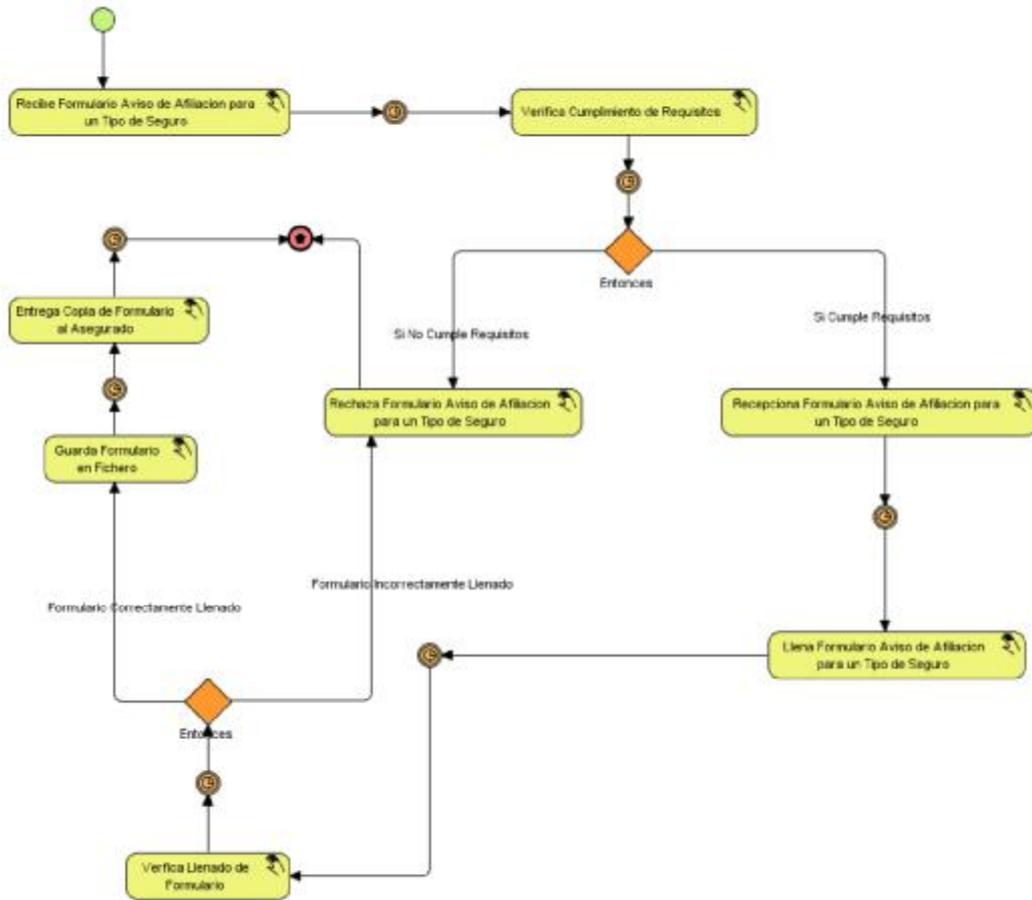


Figura 34. Proceso Registro Aviso de Afiliación para un Tipo de Seguro

2.1.2.2.3.2.4.13 Proceso Registro Aviso de Baja de Asegurado para un Tipo de Seguro

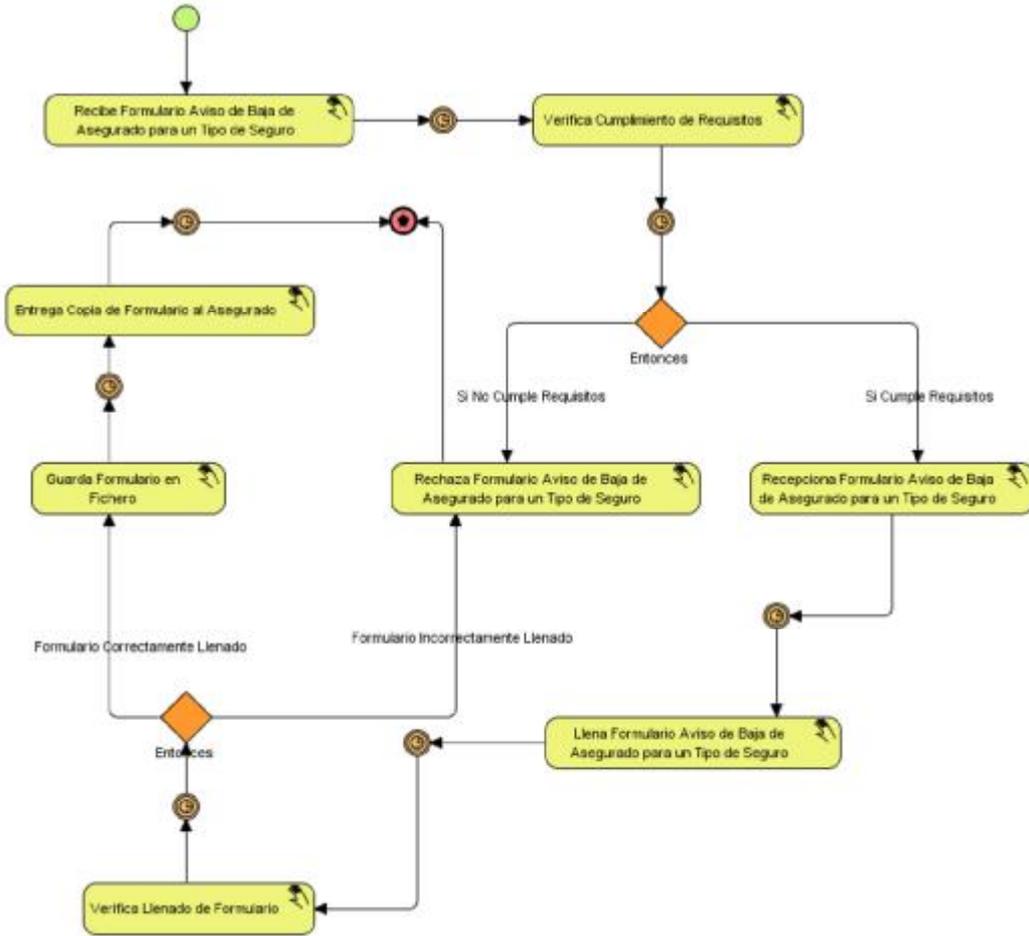


Figura 35. Proceso Registro Aviso de Baja de Asegurado para un Tipo de Seguro

2.1.2.2.3.2.5 Diagramas de Casos de Uso del Negocio

2.1.2.2.3.2.5.1 Modelo Casos de Uso del Negocio Empresa

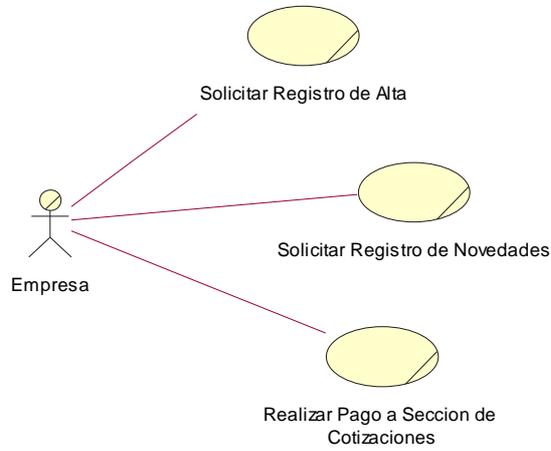


Figura 36. Modelo Casos de Uso del Negocio Empresa

2.1.2.2.3.2.5.2 Modelo Casos de Uso del Negocio Asegurado

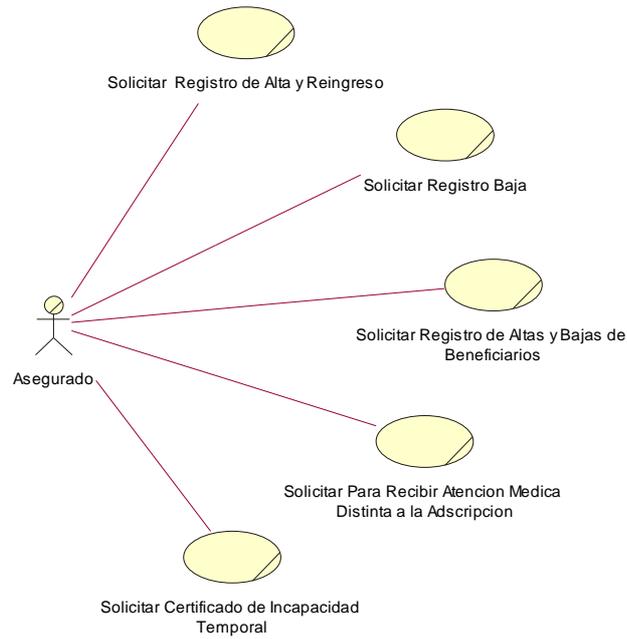


Figura 37. Modelo Casos de Uso del Negocio Asegurado

2.1.2.2.3.2.5.3 Modelo Casos de Uso del Negocio Jefe de Sección Afiliación

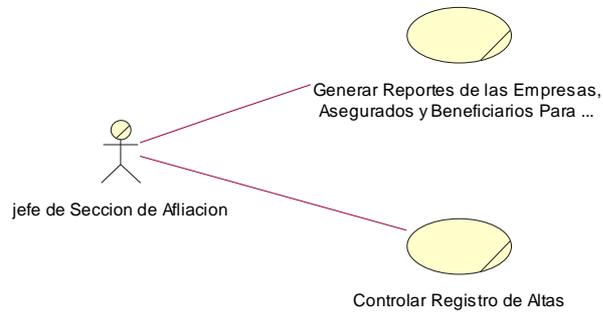


Figura 38. Modelo Casos de Uso del Negocio Jefe de Sección Afiliación

2.1.2.2.3.2.5.4 Modelo Casos de Uso del Negocio Revisor

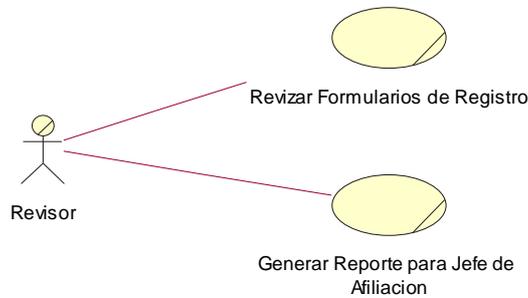


Figura 39. Modelo Casos de Uso del Negocio Revisor

2.1.2.2.3.2.5.5 Modelo Casos de Uso del Negocio Encargado de Recepción y Transcripción de Trámites de Afiliación

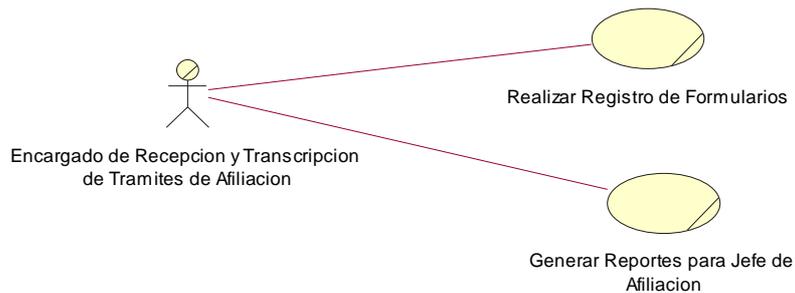


Figura 40. Modelo Casos de Uso del Negocio Encargado de Recepción y Transcripción de Trámites de Afiliación

2.1.2.2.3.3 Modelo de Objetos del Negocio

Es un modelo que describe la realización de cada Caso de Uso del Negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración (para mostrar actores externos, internos y las entidades (información) que manipulan, un

Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

2.1.2.2.3.3.1 Introducción

El Modelo de Modelo de Objetos del Negocio es un artefacto de la disciplina requisitos en la metodología RUP la cual estamos implementando.

2.1.2.2.3.3.2 Propósito

Comprender la Estructura y la Dinámica de la Organización.

Comprender los Procesos del Negocio de la Organización.

2.1.2.2.3.3.3 Alcance

Describe el comportamiento de los procesos de negocio.

Identificar y definir los objetos del negocio.

2.1.2.2.3.3.4 Diagramas de Objetos del Negocio

2.1.2.2.3.3.4.1 Modelo de Objeto del Negocio: Empresa | Solicitar Registro de Alta

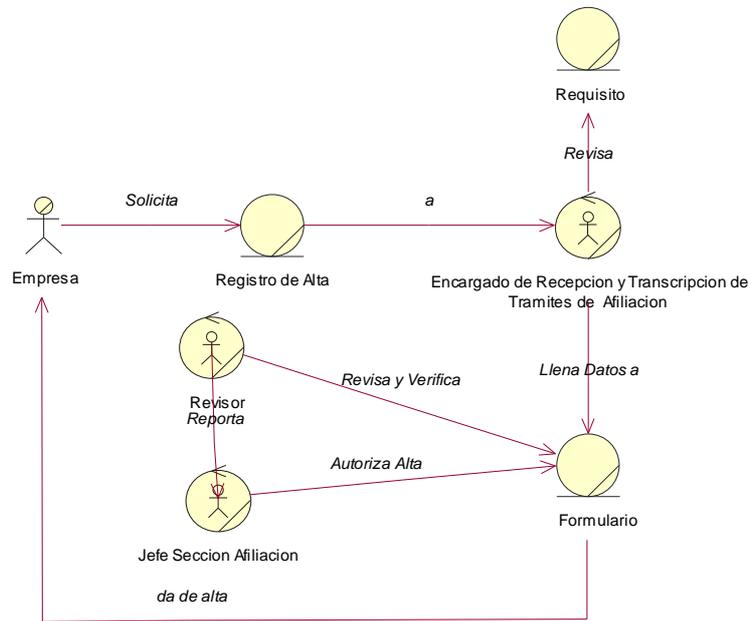


Figura 41. Modelo de Objeto del Negocio: Empresa | Solicitar Registro de Alta

2.1.2.2.3.3.4.2 Modelo de Objeto del Negocio: Empresa | Solicitar Registro de Novedades

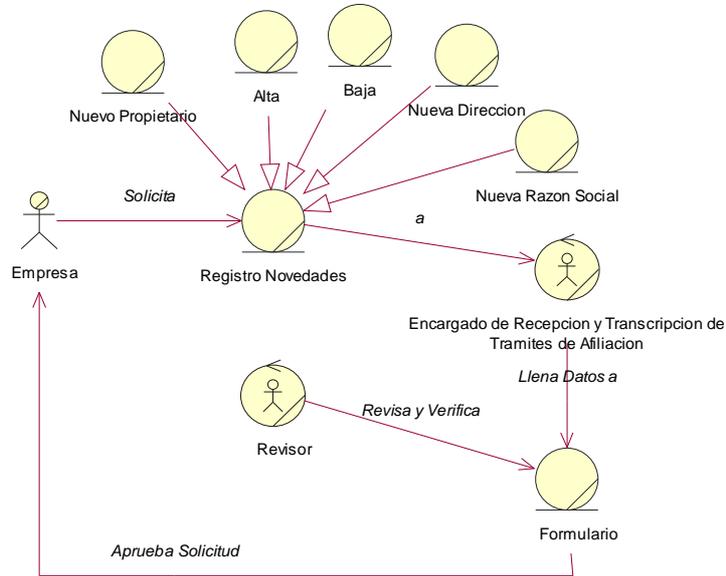


Figura 42. Modelo de Objeto del Negocio: Empresa | Solicitar Registro de Novedades

2.1.2.2.3.3.4.3 Modelo de Objeto del Negocio: Empresa | Realizar Pago a Sección de Cotizaciones

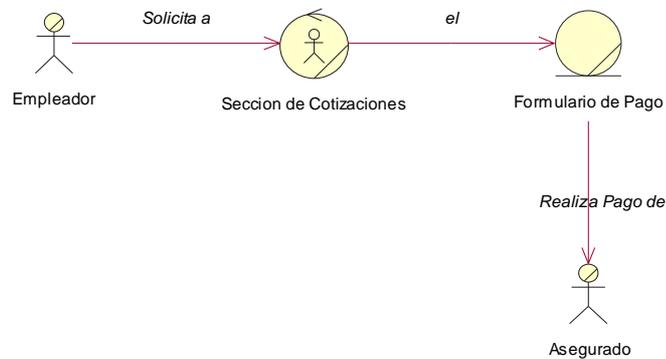


Figura 43. Modelo de Objeto del Negocio: Empresa | Realizar Pago a Sección de Cotizaciones

2.1.2.2.3.3.4.4 Modelo de Objeto del Negocio: Asegurado | Solicitar Registro de Alta y Reingreso

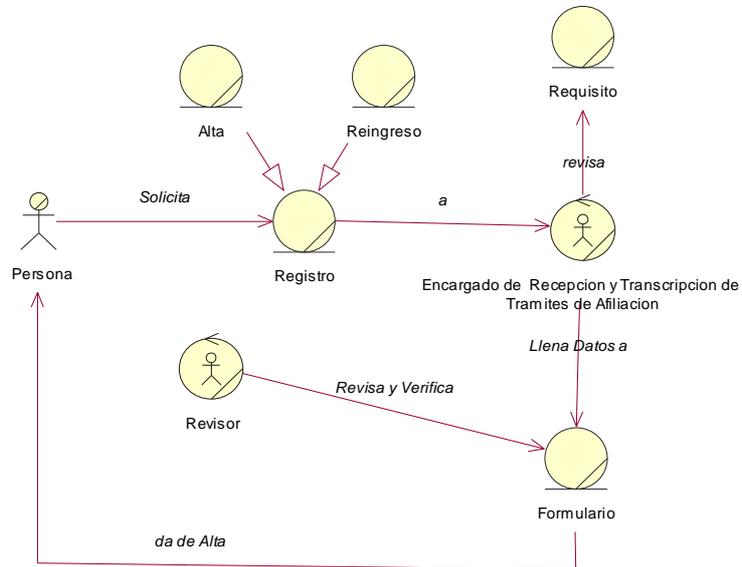


Figura 44. Modelo de Objeto del Negocio: Asegurado | Solicitar Registro de Alta y Reingreso

2.1.2.2.3.3.4.5 Modelo de Objeto del Negocio: Asegurado | Solicitar Registro Baja

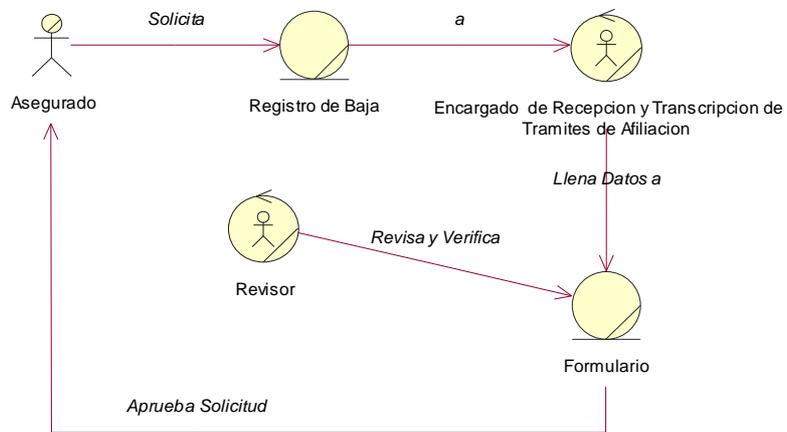


Figura 45. Modelo de Objeto del Negocio: Asegurado | Solicitar Registro Baja

2.1.2.2.3.3.4.6 Modelo de Objeto del Negocio: Asegurado | Solicitar Registro de Altas y Bajas de Beneficiarios

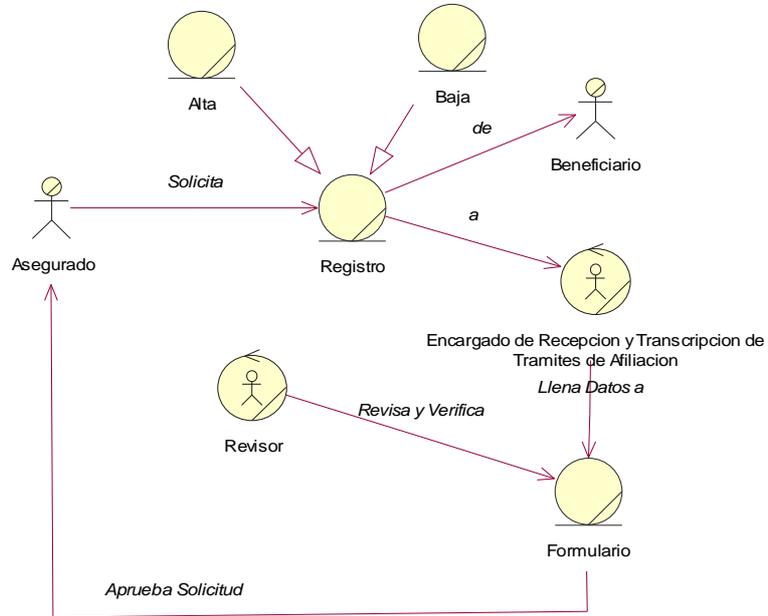


Figura 46. Modelo de Objeto del Negocio: Asegurado | Solicitar Registro de Altas y Bajas de Beneficiarios

2.1.2.2.3.3.4.7 Modelo de Objeto del Negocio: Asegurado | Solicitar Atención Médica Distinta a la Adscripción

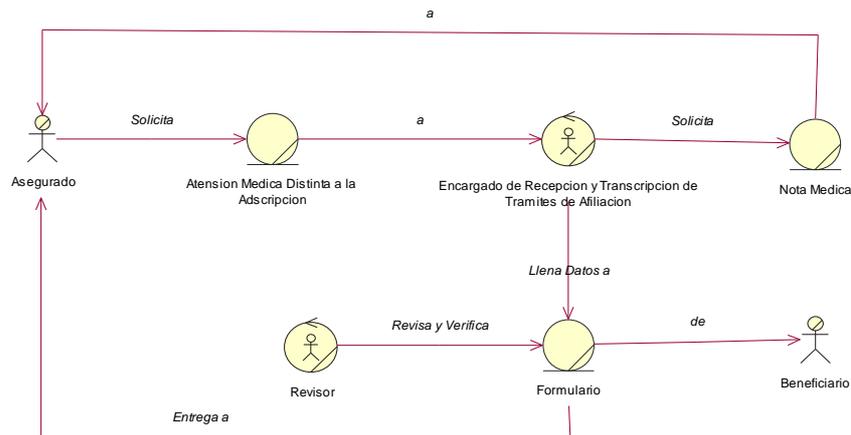


Figura 47. Modelo de Objeto del Negocio: Asegurado | Solicitar Atención Médica Distinta a la Adscripción

2.1.2.2.3.3.4.8 Modelo de Objeto del Negocio: Asegurado | Solicitar Certificado de Incapacidad Temporal

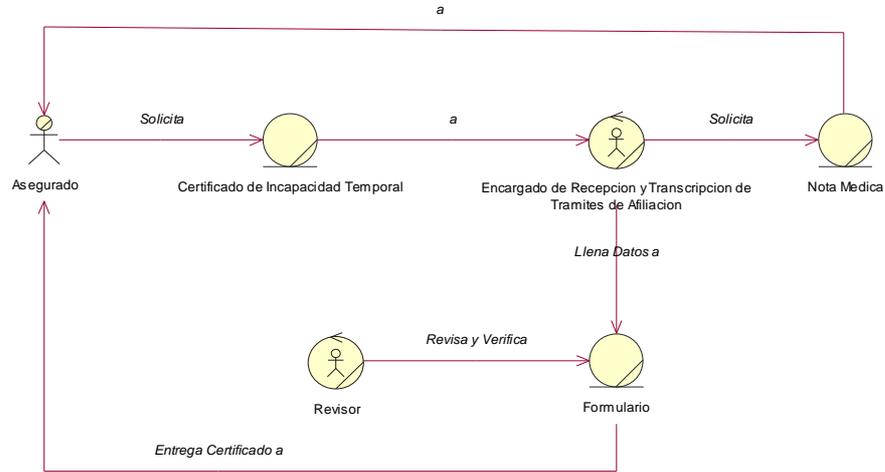


Figura 48. Modelo de Objeto del Negocio: Asegurado | Solicitar Certificado de Incapacidad Temporal

2.1.2.2.3.3.4.9 Modelo de Objeto del Negocio: Jefe de Sección Afiliación | Generar Reportes de las Empresas, Asegurados y Beneficiarios para el Administrador Regional

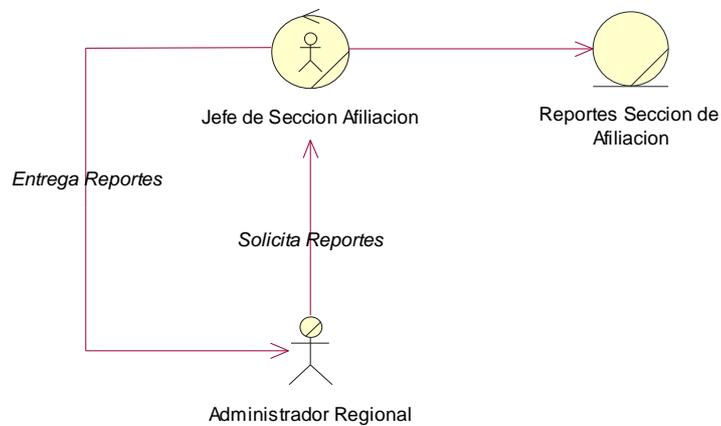


Figura 49. Modelo de Objeto del Negocio: Jefe de Sección Afiliación | Generar Reportes de las Empresas, Asegurados y Beneficiarios Para el Administrador Regional

2.1.2.2.3.3.4.10 Modelo de Objeto del Negocio: Jefe de Sección Afiliación | Controlar Registro de Altas

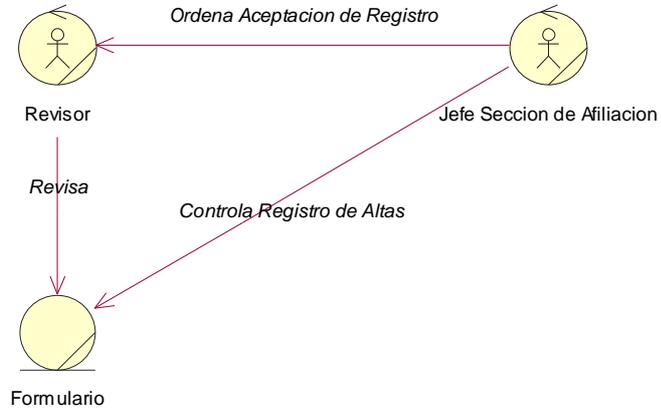


Figura 50. Modelo de Objeto del Negocio: Empresa | Controlar Registro de Altas

2.1.2.2.3.3.4.11 Modelo de Objeto del Negocio: Revisor | Revisar Formularios de Registro

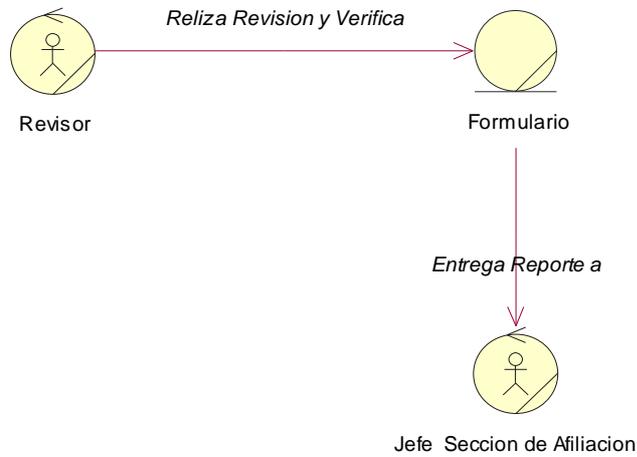


Figura 51. Modelo de Objeto del Negocio: Revisor | Revisar Formularios de Registro

2.1.2.2.3.3.4.12 Modelo de Objeto del Negocio: Revisor| Generar Reporte para Jefe de Afiliación

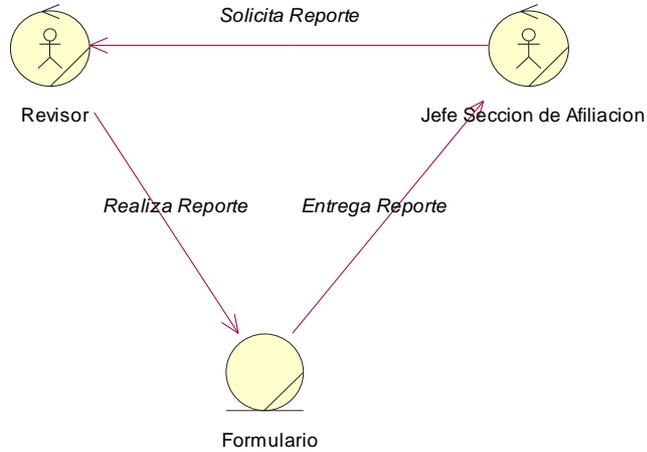


Figura 52. Modelo de Objeto del Negocio: Revisor| Generar Reporte para Jefe de Afiliación

2.1.2.2.3.3.4.13 Modelo de Objeto del Negocio: Encargado de Recepción y Transcripción de Trámites de Afiliación | Generar Reporte para Jefe de Afiliación

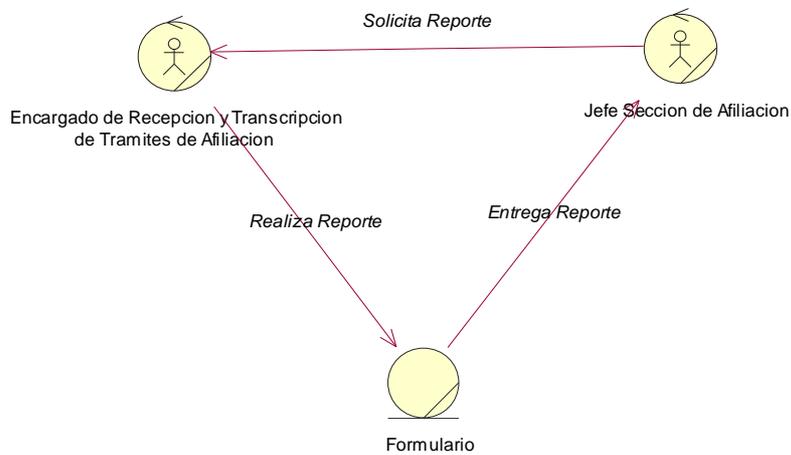


Figura 53. Modelo de Objeto del Negocio: Encargado de Recepción y Transcripción de Trámites de Afiliación | Generar Reporte para Jefe de Afiliación

2.1.2.2.3.3.4.14 Modelo de Objeto del Negocio: Encargado de Recepción y Transcripción de Trámites de Afiliación | Realizar Recepción y Registro de Formularios

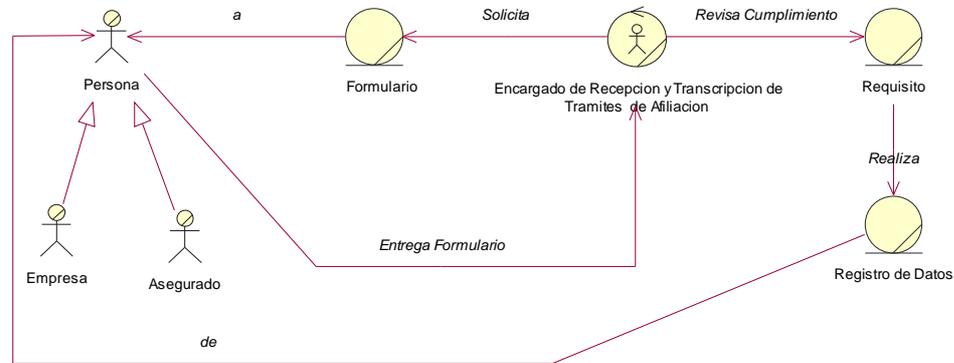


Figura 54. Modelo de Objeto del Negocio: Encargado de Recepción y Transcripción de Trámites de Afiliación | Realizar Recepción y Registro de Formularios

2.1.2.2.3.4 Modelo de Casos de Uso

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

2.1.2.2.3.4.1 Introducción

El presente documento es un artefacto de la disciplina Requisitos en la metodología RUP la cual estamos implementando.

2.1.2.2.3.4.2 Propósito

Comprender la estructura y la dinámica del sistema desarrollado.

Identificar el nivel de complejidad del sistema.

Identificar posibles mejoras.

2.1.2.2.3.4.3 Alcance

Identificar y definir procesos del sistema según los objetivos de la organización.

Definir un Caso de Uso para cada proceso del sistema (el diagrama de Caso de Uso nos detalla el contexto y los límites de la organización).

2.1.2.2.3.4.4 Diagramas de Casos de Uso

2.1.2.2.3.4.4.1 Casos de Usos del Sistema General

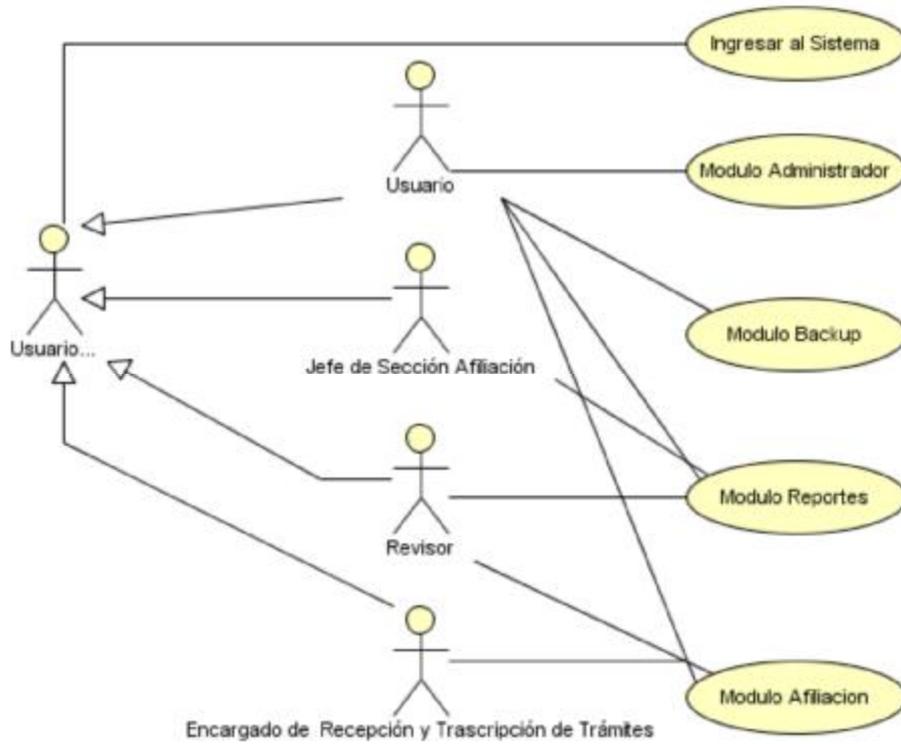


Figura 55. Casos de Usos del Sistema General

2.1.2.2.3.4.4.1.1 Modelo Casos de Uso del Módulo Administrador General

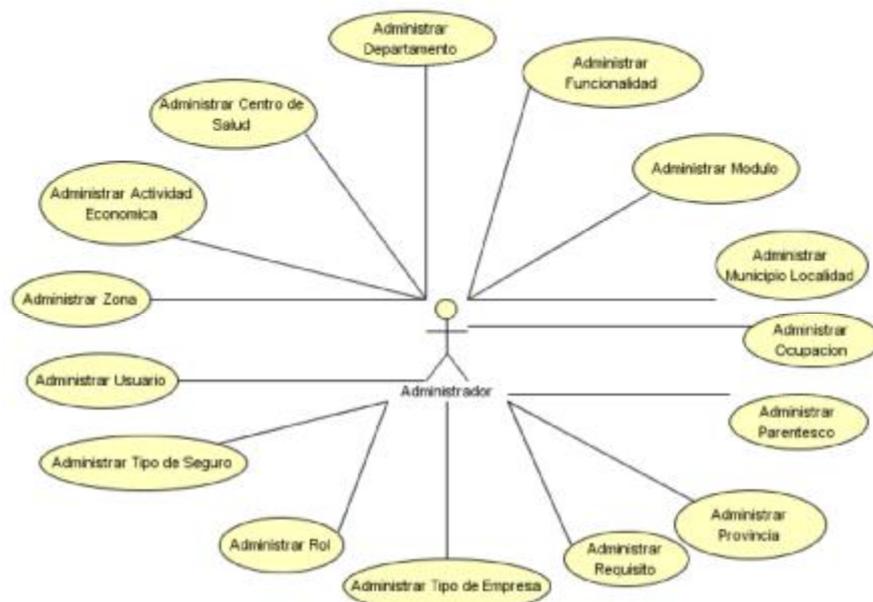


Figura 56. Modelo Casos de Uso del Módulo Administrador General

2.1.2.2.3.4.4.1.2 Modelo Casos de Uso del Módulo Afiliación General

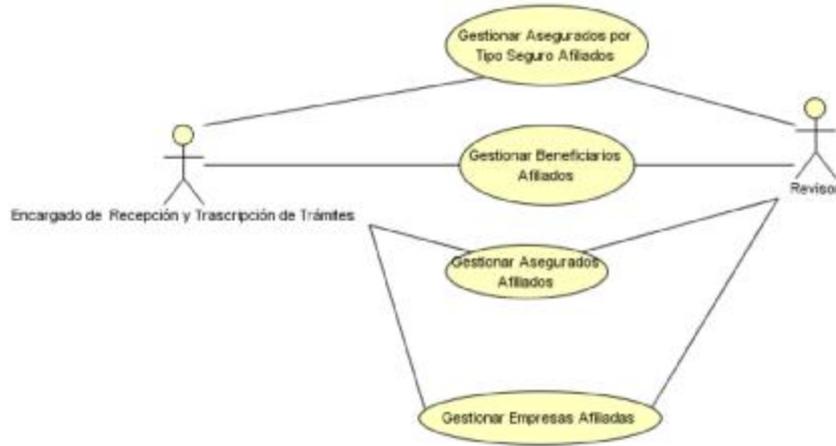


Figura 57. Modelo Casos de Usos del Módulo Afiliación General

2.1.2.2.3.4.4.1.3 Modelo Casos de Uso del Módulo Reporte General

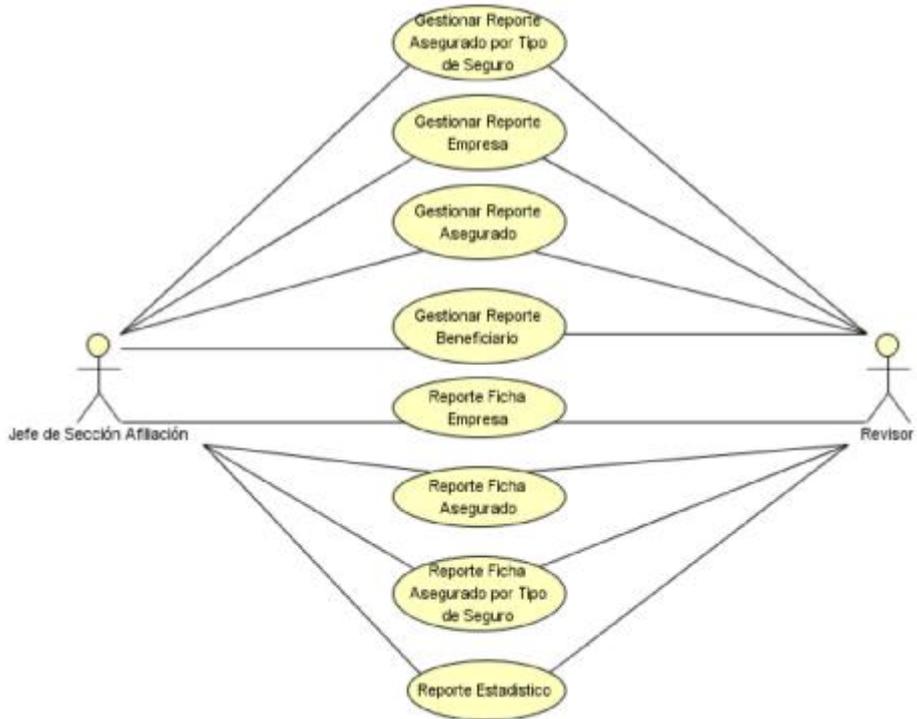


Figura 58. Modelo Casos de Usos del Módulo Reporte General

2.1.2.2.3.4.4.1.4 Modelo Casos de Uso del Módulo Backup General

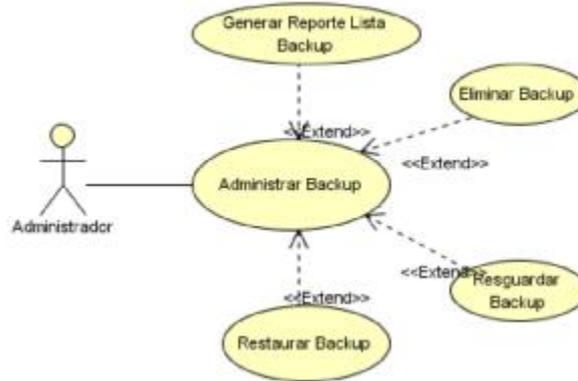


Figura 59. Modelo Casos de Uso del Módulo Backup General

2.1.2.2.3.4.4.2 Casos de Usos del Sistema Específicos

2.1.2.2.3.4.4.2.1 Modelo Casos de Uso Ingresar al Sistema



Figura 60. Modelo Casos de Uso Ingresar al Sistema

2.1.2.2.3.4.4.2.2 Modelo Casos de Uso Administrar Actividad Académica



Figura 61. Modelo Casos de Uso Administrar Actividad Académica

2.1.2.2.3.4.4.2.3 Modelo Casos de Uso Administrar Centro de Salud



Figura 62. Modelo Casos de Uso Administrar Centro de Salud

2.1.2.2.3.4.4.2.4 Modelo Casos de Uso Administrar Departamento



Figura 63. Modelo Casos de Uso Administrar Departamento

2.1.2.2.3.4.4.2.5 Modelo Casos de Uso Administrar Funcionalidad

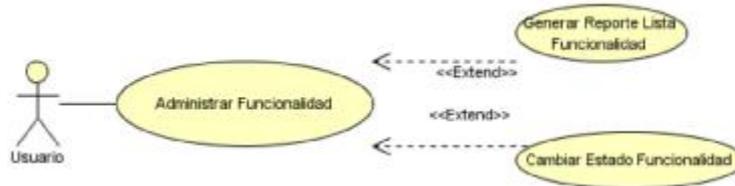


Figura 64. Modelo Casos de Uso Administrar Funcionalidad

2.1.2.2.3.4.4.2.6 Modelo Casos de Uso Administrar Módulo

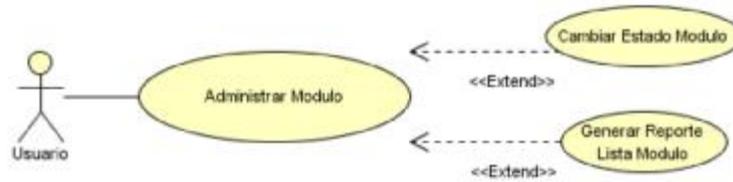


Figura 65. Modelo Casos de Uso Administrar Módulo

2.1.2.2.3.4.4.2.7 Modelo Casos de Uso Administrar Municipio Localidad



Figura 66. Modelo Casos de Uso Administrar Municipio Localidad

2.1.2.2.3.4.4.2.8 Modelo Casos de Uso Administrar Ocupación

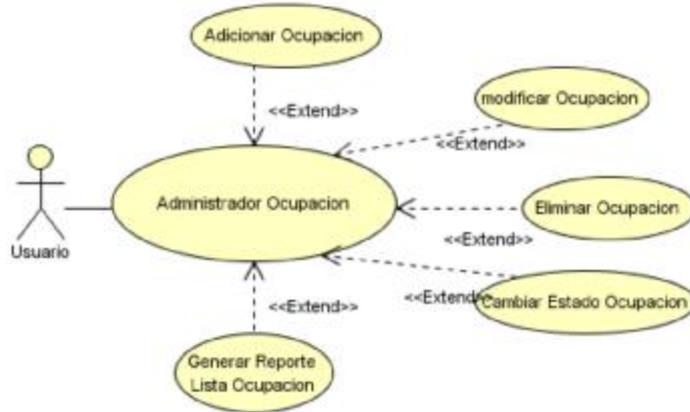


Figura 67. Modelo Casos de Uso Administrar Ocupación

2.1.2.2.3.4.4.2.9 Modelo Casos de Uso Administrar Parentesco



Figura 68. Modelo Casos de Uso Administrar Parentesco

2.1.2.2.3.4.4.2.10 Modelo Casos de Uso Administrar Provincia



Figura 69. Modelo Casos de Uso Administrar Provincia

2.1.2.2.3.4.4.2.11 Modelo Casos de Uso Administrar Requisito

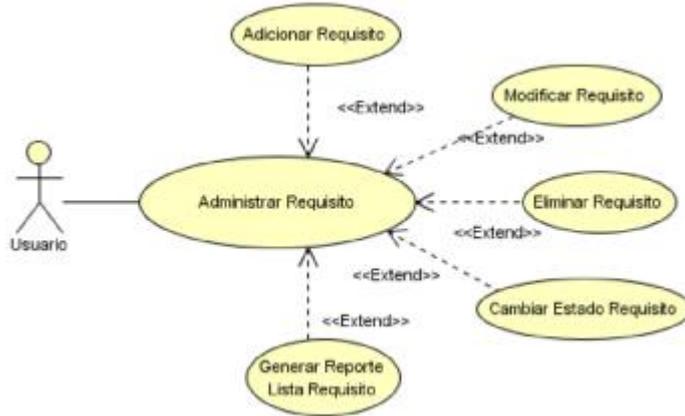


Figura 70. Modelo Casos de Uso Administrar Requisito

2.1.2.2.3.4.4.2.12 Modelo Casos de Uso Administrar Rol



Figura 71. Modelo Casos de Uso Administrar Rol

2.1.2.2.3.4.4.2.13 Modelo Casos de Uso Administrar Tipo de Empresa



Figura 72. Modelo Casos de Uso Administrar Tipo de Empresa

2.1.2.2.3.4.4.2.14 Modelo Casos de Uso Administrar Tipo de Seguro



Figura 73. Modelo Casos de Uso Administrar Tipo de Seguro

2.1.2.2.3.4.4.2.15 Modelo Casos de Uso Administrar Usuario

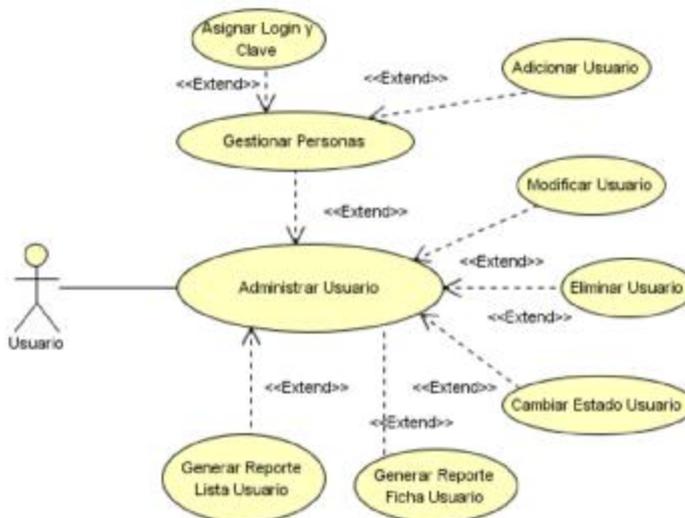


Figura 74. Modelo Casos de Uso Administrar Usuario

2.1.2.2.3.4.4.2.16 Modelo Casos de Uso Administrar Zona

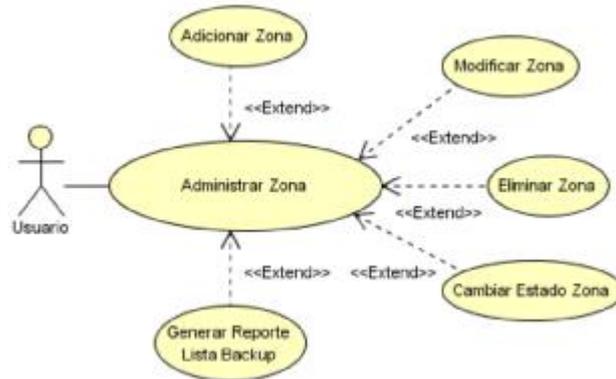


Figura 75. Modelo Casos de Uso Administrar Zona

2.1.2.2.3.4.4.2.17 Modelo de Casos de Uso Gestionar Asegurados por Tipo Seguro Afiliados

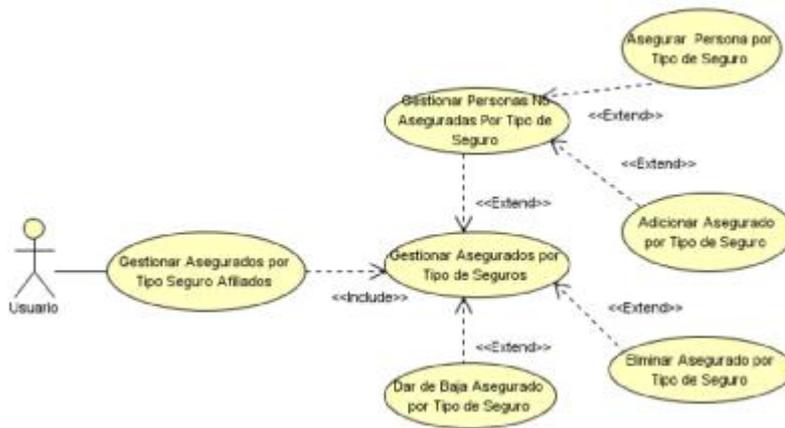


Figura 76. Modelo Casos de Uso Gestionar Asegurados por Tipo Seguro Afiliados

2.1.2.2.3.4.4.2.18 Modelo de Casos de Uso Gestionar Empresas Afiliadas



Figura 77. Modelo Casos de Uso Gestionar Empresas Afiliadas

2.1.2.2.3.4.4.2.19 Modelo de Casos de Uso Gestionar Asegurados Afiliados

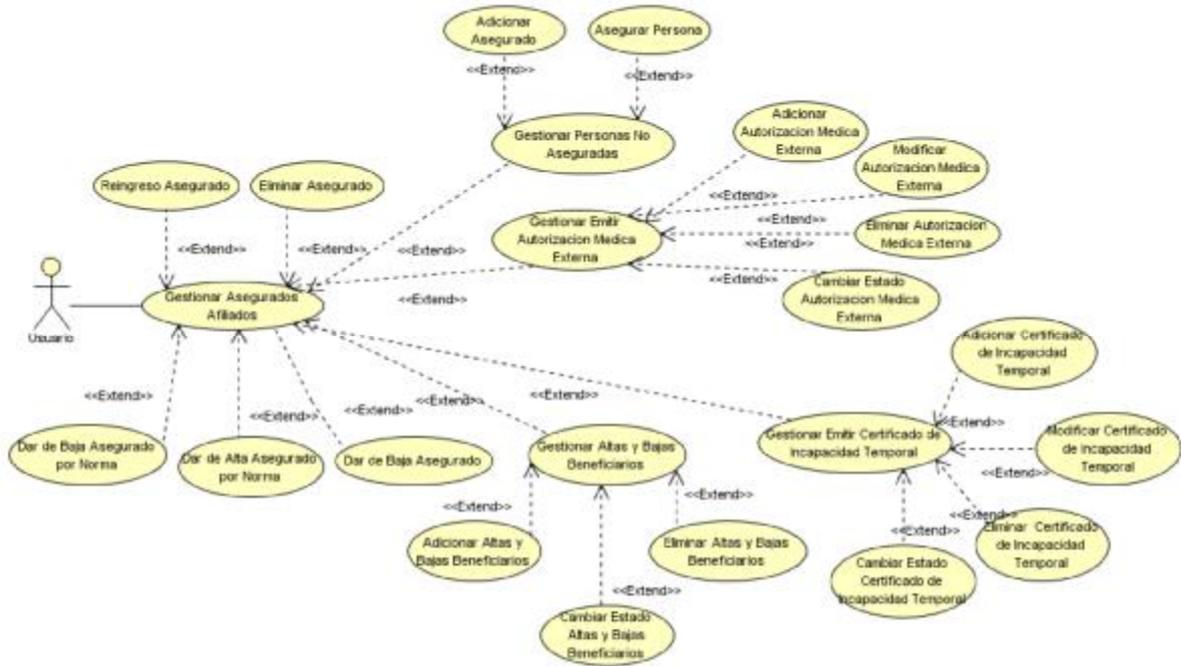


Figura 78. Modelo Casos de Uso Gestionar Asegurados Afiliados

2.1.2.2.3.4.4.2.20 Modelo de Casos de Uso Gestionar Beneficiarios Afiliados

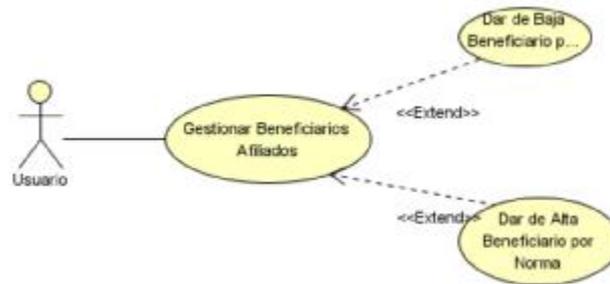


Figura 79. Modelo Casos de Uso Gestionar Beneficiarios Afiliados

2.1.2.2.3.5 Visión

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

2.1.2.2.3.5.1 Introducción

2.1.2.2.3.5.1.1 Propósito

El propósito de éste documento es recoger, definir y analizar las necesidades más importantes y las características del desarrollo del Sistema Informático para mejorar la afiliación, con el objetivo de llevar a cabo un producto que tome en cuenta las estrategias necesarias para poder brindar información

valedera para el Usuario final.

2.1.2.2.3.5.1.2 Alcance

El presente documento se ocupa de reunir todas las necesidades del Usuario, para así poder diseñar un Sistema que satisfaga a las mismas como ser:

Contar con un medio de publicación y promoción de nuevos artistas y compositores de cueca tarijeña.

Contar con la información centralizada en una sola organización.

Brindar un medio de consultas de información acerca de las Empresas Empleadoras, Asegurados, Beneficiarios.

2.1.2.2.3.5.2 Posicionamiento

2.1.2.2.3.5.2.1 Oportunidad de Negocio

Actualmente el mantenimiento de la información de los afiliados de la Caja Nacional de Salud Regional Tarija, en la cual la información (datos) debe ser almacenada de manera digital.

Los beneficios de la utilización de nuevas tecnologías de comunicación como ser el Internet, nos aseguran la integridad, veracidad y la organización de nuestra información.

Tanto el inadecuado proceso de organización de la información y registro en la Sección de Afiliación, fueron motivos suficientes para elaborar el proyecto que permita el Mejoramiento en los Procesos de Registro en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija.

2.1.2.2.3.5.2.2 Sentencia que Define el Problema

El problema de	Inadecuado proceso de organización de la información y registro en la Sección de Afiliación.
afecta a	Caja Nacional de Salud Regional Tarija y Sociedad Boliviana.
El impacto asociado es	Insuficiencia de mecanismos que proporcionen información de los registros de Afiliados.
Una solución adecuada sería	Diseñar un Sistema Informático destinado al Registro de Afiliados de la C.N.S. Regional Tarija desarrollado.

Tabla 14. Sentencia que Define el Problema

2.1.2.2.3.5.2.3 Sentencia que Define la Posición del Producto

para	La Caja Nacional de Salud – Regional Tarija.
quienes	Población en General, Jefe de Sección Afiliación, Revisor, Encargado de Recepción, Transcripción de Trámites de Afiliación y Encargado de Sistemas.
El nombre del producto	“SIMA” Sistema Informático para Mejoramiento de Afiliación.
que	Contribuir a mejorar la calidad de servicios a la población

	asegurada de la Caja Nacional de salud - Tarija.
Nuestro producto	Permitir gestionar las distintas actividades de la empresa mediante una interfaz gráfica sencilla y amigable, proporcionar un acceso rápido y actualizado a la base de datos.

Tabla 15. Sentencia que Define la Posición del Producto

2.1.2.2.3.5.3 Descripción de los Stakeholders (Participantes del proyecto) y Usuarios

Para proveer de una forma efectiva productos y servicios que se ajusten a las necesidades de los usuarios, es necesario identificar e involucrar a todos los participantes en el proyecto como parte del proceso de modelado de requerimientos. También es necesario identificar a los usuarios del sistema y asegurarse de que el conjunto de participantes en el proyecto los representa adecuadamente. Esta sección muestra un perfil de los participantes y de los usuarios involucrados en el proyecto, así como los problemas más importantes que estos perciben para enfocar la solución propuesta hacia ellos. No describe sus requisitos específicos ya que estos se capturan mediante otro artefacto. En lugar de esto proporciona la justificación de porqué estos requisitos son necesarios.

2.1.2.2.3.5.3.1 Perfil de los Stakeholders

2.1.2.2.3.5.3.1.1 Universidad Autónoma Juan Misael Saracho

Representante	Universidad Autónoma Juan Misael Saracho.
Descripción	Entidad que asesora el proyecto.
Tipo	Proveedor.
Responsabilidades	Seguimiento del desarrollo del proyecto. Aprueba requisitos y funcionalidades.
Grado de participación	Velar que el proyecto sea ejecutado.

Tabla 16. Universidad Autónoma Juan Misael Saracho

2.1.2.2.3.5.3.1.2 Basilka

Representante	Basilka
Descripción	Grupo desarrollador del software.
Tipo	Desarrolladores.
Responsabilidades	Provee los equipos para el desarrollo tanto del análisis, diseño y programación del software. Diseña las posibles interfaces finales a ser implementadas en el sistema. Programa el software de acuerdo a los requisitos del sistema. Ejecuta y realiza reiteradas pruebas al sistema en su posible versión final.
Grado de participación	Encargados del análisis, diseño y desarrollo del sistema.

Tabla 17. Basilka

2.1.2.2.3.5.3.1.3Caja Nacional de Salud Regional Tarija

Representante	Lic. Carlos Michel Cachambi
Descripción	Encargado de Sistemas Caja Nacional de Salud Regional Tarija.
Tipo	Administrador.
Responsabilidades	Administración del Sistema Informático para el Mejoramiento de la Afiliación, y realiza la revisión y validación de la información aportada por los Usuarios.
Grado de participación	Velar por que se cumplan requisitos y funcionalidades, y que el proyecto sea ejecutado.

Tabla 18. Encargado de Sistemas Caja Nacional de Salud Regional Tarija.

2.1.2.2.3.5.3.1.4Docente

Representante	Docente
Descripción	Ente regulador universitario.
Tipo	Guía de desarrollo.
Responsabilidades	Realiza un control paso a paso del desarrollo del proyecto. Aprueba los distintos puntos tratados en el proyecto. Analiza los distintos documentos presentados acerca del producto.
Grado de participación	Guiar y evaluar el análisis, diseño y desarrollo del sistema.

Tabla 19. Docente

2.1.2.2.3.5.3.2 Perfil de Usuario

2.1.2.2.3.5.3.2.1Administrador

Representante	Administrador
Descripción	Administrador de los recursos del sistema y del módulo Administrar.
Tipo	Usuario.
Responsabilidades	Responsable del control de roles y acceso, usuarios del sistema; además es el responsable de resguardar y restaurar la base de datos del sistema.
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 20. Administrador

2.1.2.2.3.5.3.2.2Jefe de Sección Afiliación

Representante	Operador.
Descripción	Usuario independiente y directo, con ciertos atributos- permisos de ejecución de la aplicación.
Tipo	Usuario causal.
Responsabilidades	Hacer uso adecuado del sistema velar por la integridad del mismo, respetando las funcionalidades del mismo.
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 21. Jefe de Sección Afiliación

2.1.2.2.3.5.3.2.3 Revisor

Representante	Operador.
Descripción	Usuario independiente y directo, con ciertos atributos- permisos de ejecución de la aplicación.
Tipo	Usuario causal.
Responsabilidades	Hacer uso adecuado del sistema velar por la integridad del mismo, respetando las funcionalidades del mismo.
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 22. Revisor

2.1.2.2.3.5.3.2.4 Encargado de Recepción y Transcripción de Trámites de Afiliación

Representante	Operador.
Descripción	Usuario independiente y directo, con ciertos atributos- permisos de ejecución de la aplicación.
Tipo	Usuario causal.
Responsabilidades	Hacer uso adecuado del sistema velar por la integridad del mismo, respetando las funcionalidades del mismo.
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 23. Encargado de Recepción y Transcripción de Trámites de Afiliación

2.1.2.2.3.5.3.2.5 Usuario Registrado

Representante	Operador.
Descripción	Usuario independiente y directo, con ciertos atributos- permisos de ejecución de la aplicación.
Tipo	Usuario causal.
Responsabilidades	Hacer uso adecuado del sistema velar por la integridad del mismo, respetando las funcionalidades del mismo.
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 24. Usuario Registrado

2.1.2.2.3.5.3.3 Entorno del Usuario

La Sección de Afiliación de la C. N. S. –Regional Tarija debe acceder al Sitio Web del Sistema donde podrá hacer uso de información del mismo, en el cual el Usuario debe estar registrado, para poder obtener algunas características adicionales-atributos-permisos. Si es un administrador debe acceder al programa instalado donde debe estar registrado como administrador para poder realizar la gestión de todo el Sistema.

2.1.2.2.3.5.4 Descripción Global del Producto

Debido a la creciente demanda de las nuevas tecnologías de información y comunicación y el impacto sociocultural que causan en nuestro entorno, elaboramos el proyecto “SIMA” Sistema Informático para

el Mejoramiento de Afiliación.

Los beneficios de la utilización de nuevas tecnologías de comunicación como ser el Internet nos aseguran la integridad, veracidad y la organización de nuestra información.

2.1.2.2.3.6 Especificaciones de Casos de Uso

Para los Casos de Uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, postcondiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

2.1.2.2.3.6.1 Introducción

Las Especificaciones de los Casos de Uso es una descripción detallada de los Casos de Uso del Sistema.

2.1.2.2.3.6.2 Propósito

Interpretar y describir los Casos de Uso.

2.1.2.2.3.6.3 Alcance

Describe los procesos internos de los Casos de Uso.

Detalla los flujos de los Casos de Uso según lo establecido por la organización.

2.1.2.2.3.6.4 Especificación de los Casos de Uso

2.1.2.2.3.6.4.1 Especificación de Caso de Uso Ingresar al Sistema

Caso de Uso:	Ingresar al Sistema.
Descripción: Permite ingresar al Sistema, este caso tiene como función controlar el acceso y al mismo tiempo recuperar los permisos correspondientes al momento que el Usuario introduzca su login y clave en el Sistema.	
Actores: Administrador, Jefe Sección de Afiliación, Revisor, Encargado de Recepción y Transcripción de Trámites de Afiliación.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El actor introduce datos de login y clave en el Sistema. 2. El actor presiona botón Ingresar. 3. El Sistema valida sus datos. 4. El actor espera su validación. 5. Si los datos son correctos muestra la Pantalla Principal de Administración.	Flujo Alternativo: 3. Si el actor no llena el formulario de acceso, se muestra un mensaje donde se informa que los campos son requeridos. 3. Con más de tres intentos fallidos el Sistema deshabilita la opción de Ingresar. 5. Si el usuario no existe se muestra mensaje de error y vuelve a mostrar la pantalla índice.

Pos condiciones: Ninguno

Tabla 25. Especificación de Caso de Uso Ingresar al Sistema
2.1.2.2.3.6.4.2 Especificación de Casos de Uso Módulo Administrador
2.1.2.2.3.6.4.2.1 Especificación Caso de Uso Administrar Actividad Económica

Caso de Uso:	Administrar Actividad Económica.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Actividad Económica.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Actividades Económicas registrados en el Sistema. 2. El actor realiza la Búsqueda de Actividad Económica en el listado. 3. El Sistema muestra el(los) Actividad Económica(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Actividad Económica”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 26. Especificación Caso de Uso Administrar Actividad Económica
2.1.2.2.3.6.4.2.2 Especificación Caso de Uso Adicionar Actividad Económica

Caso de Uso:	Adicionar Actividad Económica.
Descripción: Permite registrar nueva Actividad Económica en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de la Actividad Económica en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Actividad Económica.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Actividad Económica. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 27. Especificación Caso de Uso Adicionar Actividad Económica

2.1.2.2.3.6.4.2.3 Especificación Caso de Uso Modificar Actividad Económica

Caso de Uso:	Modificar Actividad Económica.	
Descripción: Permite modificar datos de una Actividad Económica en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Actividad Económica.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos de la Actividad Económica. 3. El actor realiza los cambios de datos de la Actividad Económica en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Actividad Económica. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Actividad Económica. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno		

Tabla 28. Especificación Caso de Uso Modificar Actividad Económica

2.1.2.2.3.6.4.2.4 Especificación Caso de Uso Eliminar Actividad Económica

Caso de Uso:	Eliminar Actividad Económica.	
Descripción: Permite eliminar del Sistema una Actividad Económica.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Actividad Económica.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si, El Sistema realiza la eliminación de Actividad Económica. 4. El Sistema regresa a la pantalla de Administrar Actividad Económica. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.		

Tabla 29. Especificación Caso de Uso Eliminar Actividad Económica

2.1.2.2.3.6.4.2.5 Especificación Caso de Uso Cambiar Estado Actividad Económica

Caso de Uso:	Cambiar Estado Actividad Económica.	
Descripción: Permite cambiar estado de una Actividad Económica en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Actividad Económica		
Flujo Normal:	Flujo Alternativo:	

<ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Actividad Económica. 4. El Sistema regresa a la pantalla de Administrar Actividad Económica. 	<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 30. Especificación Caso de Uso Cambiar Estado Actividad Económica

2.1.2.2.3.6.4.2.6 Especificación Caso de Uso Generar Reporte Lista Actividad Económica

Caso de Uso:	Generar Reporte Lista Actividad Económica.		
Descripción: Permite visualizar reporte.			
Actores: Administrador.			
Precondiciones: Actor haber iniciado sesión.			
Flujo Normal:		Flujo Alternativo:	
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte. 			
Pos condiciones:			

Tabla 31. Especificación Caso de Uso Generar Reporte Lista Actividad Económica

2.1.2.2.3.6.4.2.7 Especificación Caso de Uso Administrar Centro de Salud

Caso de Uso:	Administrar Centro de Salud.		
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Centro de Salud.			
Actores: Administrador.			
Precondiciones: Actor haber iniciado sesión.			
Flujo Normal:		Flujo Alternativo:	
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Centros de Salud registrados en el Sistema. 2. El actor realiza la Búsqueda de Centros de Salud en el Listado. 3. El Sistema muestra el(los) Centros de Salud que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 		<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia "Debe Seleccionar un Centro de Salud". 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación. 	
Pos condiciones: Ninguno			

Tabla 32. Especificación Caso de Uso Administrar Centro de Salud
2.1.2.2.3.6.4.2.8 Especificación Caso de Uso Adicionar Centro de Salud

Caso de Uso:	Adicionar Centro de Salud.	
Descripción: Permite registrar nuevo Centro de Salud en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Centro de Salud en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Centro de Salud. 		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Centro de Salud. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno		

Tabla 33. Especificación Caso de Uso Adicionar Centro de Salud
2.1.2.2.3.6.4.2.9 Especificación Caso de Uso Modificar Centro de Salud

Caso de Uso:	Modificar Centro de Salud.	
Descripción: Permite modificar datos de un Centro de Salud en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Centro de Salud.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos del Centro de Salud. 3. El actor realiza los cambios de datos del Centro de Salud en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Centro de Salud. 		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Centro de Salud. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno		

Tabla 34. Especificación Caso de Uso Modificar Centro de Salud
2.1.2.2.3.6.4.2.10 Especificación Caso de Uso Eliminar Centro de Salud

Caso de Uso:	Eliminar Centro de Salud.	
Descripción: Permite eliminar del Sistema un Centro de Salud.		
Actores: Administrador.		

Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Centro de Salud.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Centro de Salud. 4. El Sistema regresa a la pantalla de Administrar Centro de Salud.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 35. Especificación Caso de Uso Eliminar Centro de Salud
2.1.2.2.3.6.4.2.11 Especificación Caso de Uso Cambiar Estado Centro de Salud

Caso de Uso:	Cambiar Estado Centro de Salud.
Descripción: Permite cambiar estado de un Centro de Salud en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Centro de Salud.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Centro de Salud. 4. El Sistema regresa a la pantalla de Administrar Centro de Salud.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 36. Especificación Caso de Uso Cambiar Estado Centro de Salud
2.1.2.2.3.6.4.2.12 Especificación Caso de Uso Generar Reporte Lista Centro de Salud

Caso de Uso:	Generar Reporte Lista Centro de Salud.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 37. Especificación Caso de Uso Generar Reporte Lista Centro de Salud
2.1.2.2.3.6.4.2.13 Especificación Caso de Uso Administrar Departamento

Caso de Uso:	Administrar Departamento.
---------------------	---------------------------

Descripción: Permite Listar y Generar Reporte PDF de Departamento.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Departamentos registrados en el Sistema. 2. El actor realiza la Búsqueda de Departamento en el Listado. 3. El Sistema muestra el(los) Departamento(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Departamento”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 38. Especificación Caso de Uso Administrar Departamento
2.1.2.2.3.6.4.2.14 Especificación Caso de Uso Generar Reporte Lista Departamento

Caso de Uso:	Generar Reporte Lista Departamento.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 39. Especificación Caso de Uso Generar Reporte Lista Departamento
2.1.2.2.3.6.4.2.15 Especificación Caso de Uso Administrar Funcionalidad

Caso de Uso:	Administrar Funcionalidad.
Descripción: Permite Listar, Cambiar Estado y Generar Reporte PDF de Funcionalidad.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Funcionalidades registrados en el Sistema. 2. El actor realiza la Búsqueda de Funcionalidad en el Listado. 3. El Sistema muestra el(los) Funcionalidades(s) que cumple(n) con los Criterios de	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Funcionalidad”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.

Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	
Pos condiciones: Ninguno	

Tabla 40. Especificación Caso de Uso Administrar Funcionalidad

2.1.2.2.3.6.4.2.16 Especificación Caso de Uso Cambiar Estado Funcionalidad

Caso de Uso:	Cambiar Estado Funcionalidad.
Descripción: Permite cambiar estado de una Funcionalidad en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Funcionalidad.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Funcionalidad. 4. El Sistema regresa a la pantalla de Administrar Funcionalidad.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 41. Especificación Caso de Uso Cambiar Estado Funcionalidad

2.1.2.2.3.6.4.2.17 Especificación Caso de Uso Generar Reporte Lista Funcionalidad

Caso de Uso:	Generar Reporte Lista Funcionalidad.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 42. Especificación Caso de Uso Generar Reporte Lista Funcionalidad

2.1.2.2.3.6.4.2.18 Especificación Caso de Uso Administrar Módulo

Caso de Uso:	Administrar Módulo.
Descripción: Permite Listar, Cambiar Estado y Generar Reporte PDF de Módulo.	
Actores: Administrador.	

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El Sistema muestra Listado de Módulos registrados en el Sistema. 2. El actor realiza la Búsqueda de Módulos en el Listado. 3. El Sistema muestra el(los) Módulos (s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	Flujo Alternativo: <ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Módulo”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 43. Especificación Caso de Uso Administrar Módulo
2.1.2.2.3.6.4.2.19 Especificación Caso de Uso Cambiar Estado Módulo

Caso de Uso:	Cambiar Estado Módulo.
Descripción: Permite cambiar estado de una Módulo en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Módulo.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Módulo. 4. El Sistema regresa a la pantalla de Administrar Módulo. 	Flujo Alternativo: <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 44. Especificación Caso de Uso Cambiar Estado Módulo
2.1.2.2.3.6.4.2.20 Especificación Caso de Uso Generar Reporte Lista Módulo

Caso de Uso:	Generar Reporte Lista Módulo.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte. 	Flujo Alternativo:
Pos condiciones:	

Tabla 45. Especificación Caso de Uso Generar Reporte Lista Módulo
2.1.2.2.3.6.4.2.21 Especificación Caso de Uso Administrar Municipio Localidad

Caso de Uso:	Administrar Municipio Localidad.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Municipio Localidad.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Municipios Localidades registrados en el Sistema. 2. El actor realiza la Búsqueda de Municipio Localidad en el Listado. 3. El Sistema muestra el(los) Municipios Localidades que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Clic derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Municipio Localidad”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 46. Especificación Caso de Uso Administrar Municipio Localidad
2.1.2.2.3.6.4.2.22 Especificación Caso de Uso Adicionar Municipio Localidad

Caso de Uso:	Adicionar Municipio Localidad.
Descripción: Permite registrar nuevo Municipio Localidad en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Municipio Localidad en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Municipio Localidad.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Municipio Localidad. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 47. Especificación Caso de Uso Adicionar Municipio Localidad
2.1.2.2.3.6.4.2.23 Especificación Caso de Uso Modificar Municipio Localidad

Caso de Uso:	Modificar Municipio Localidad.
---------------------	--------------------------------

Descripción: Permite modificar datos de un Municipio Localidad en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Municipio Localidad.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos del Municipio Localidad. 3. El actor realiza los cambios de datos del Municipio Localidad en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Municipio Localidad.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Municipio Localidad. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 48. Especificación Caso de Uso Modificar Municipio Localidad
2.1.2.2.3.6.4.2.24 Especificación Caso de Uso Eliminar Municipio Localidad

Caso de Uso:	Eliminar Municipio Localidad.
Descripción: Permite eliminar del Sistema un Municipio Localidad.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Municipio Localidad.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Municipio Localidad. 4. El Sistema regresa a la pantalla de Administrar Municipio Localidad.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 49. Especificación Caso de Uso Eliminar Municipio Localidad
2.1.2.2.3.6.4.2.25 Especificación Caso de Uso Cambiar Estado Municipio Localidad

Caso de Uso:	Cambiar Estado Municipio Localidad.
Descripción: Permite cambiar estado de un Municipio Localidad en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Municipio Localidad.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.

confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Municipio Localidad. 4. El Sistema regresa a la pantalla de Administrar Municipio Localidad.	
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 50. Especificación Caso de Uso Cambiar Estado Municipio Localidad
2.1.2.2.3.6.4.2.26 Especificación Caso de Uso Generar Reporte Lista Municipio Localidad

Caso de Uso:	Generar Reporte Lista Municipio Localidad.	
Descripción: Permite visualizar reporte.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.		
Pos condiciones:		

Tabla 51. Especificación Caso de Uso Generar Reporte Lista Municipio Localidad
2.1.2.2.3.6.4.2.27 Especificación Caso de Uso Administrar Ocupación

Caso de Uso:	Administrar Ocupación.	
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Ocupación.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
1. El Sistema muestra Listado de Ocupaciones registrados en el Sistema. 2. El actor realiza la Búsqueda de Ocupación en el Listado. 3. El Sistema muestra (las) Ocupaciones que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.		6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Ocupación”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno		

Tabla 52. Especificación Caso de Uso Administrar Ocupación
2.1.2.2.3.6.4.2.28 Especificación Caso de Uso Adicionar Ocupación

Caso de Uso:	Adicionar Ocupación.
---------------------	----------------------

Descripción: Permite registrar nueva Ocupación en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de la Ocupación en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Ocupación.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Ocupación. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 53. Especificación Caso de Uso Adicionar Ocupación
2.1.2.2.3.6.4.2.29 Especificación Caso de Uso Modificar Ocupación

Caso de Uso:	Modificar Ocupación.
Descripción: Permite modificar datos de una Ocupación en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Ocupación.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos de la Ocupación. 3. El actor realiza los cambios de datos de la Ocupación en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Ocupación.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Ocupación. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 54. Especificación Caso de Uso Modificar Ocupación
2.1.2.2.3.6.4.2.30 Especificación Caso de Uso Eliminar Ocupación

Caso de Uso:	Eliminar Ocupación.
Descripción: Permite eliminar del Sistema una Ocupación.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Ocupación.	
Flujo Normal: 1. El actor presiona en el menú la opción	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.

Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación de Ocupación. 4. El Sistema regresa a la pantalla de Administrar Ocupación.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 55. Especificación Caso de Uso Eliminar Ocupación
2.1.2.2.3.6.4.2.31 Especificación Caso de Uso Cambiar Estado Ocupación

Caso de Uso:	Cambiar Estado Ocupación.
Descripción: Permite cambiar estado de una Ocupación en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Ocupación.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Ocupación. 4. El Sistema regresa a la pantalla de Administrar Ocupación.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 56. Especificación Caso de Uso Cambiar Estado Ocupación
2.1.2.2.3.6.4.2.32 Especificación Caso de Uso Generar Reporte Lista Ocupación

Caso de Uso:	Generar Reporte Lista Ocupación.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 57. Especificación Caso de Uso Generar Reporte Lista Ocupación
2.1.2.2.3.6.4.2.33 Especificación Caso de Uso Administrar Parentesco

Caso de Uso:	Administrar Parentesco.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Parentesco.	
Actores: Administrador.	

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El Sistema muestra Listado de Parentescos registrados en el Sistema. 2. El actor realiza la Búsqueda de Parentesco en el Listado. 3. El Sistema muestra el(los) Parentesco (s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	Flujo Alternativo: <ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Parentesco”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 58. Especificación Caso de Uso Administrar Parentesco
2.1.2.2.3.6.4.2.34 Especificación Caso de Uso Adicionar Parentesco

Caso de Uso:	Adicionar Parentesco.
Descripción: Permite registrar nuevo Parentesco en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Parentesco en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Parentesco. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Parentesco. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 59. Especificación Caso de Uso Adicionar Parentesco
2.1.2.2.3.6.4.2.35 Especificación Caso de Uso Modificar Parentesco

Caso de Uso:	Modificar Parentesco.
Descripción: Permite modificar datos de un Parentesco en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Parentesco.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Parentesco.

<ol style="list-style-type: none"> 2. El Sistema le muestra formulario con datos del Parentesco. 3. El actor realiza los cambios de datos del Parentesco en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Parentesco. 	<ol style="list-style-type: none"> 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 60. Especificación Caso de Uso Modificar Parentesco
2.1.2.2.3.6.4.2.36 Especificación Caso de Uso Eliminar Parentesco

Caso de Uso:	Eliminar Parentesco.
Descripción: Permite eliminar del Sistema un Parentesco.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Parentesco.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Parentesco. 4. El Sistema regresa a la pantalla de Administrar Parentesco. 	Flujo Alternativo: <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 61. Especificación Caso de Uso Eliminar Parentesco
2.1.2.2.3.6.4.2.37 Especificación Caso de Uso Cambiar Estado Parentesco

Caso de Uso:	Cambiar Estado Parentesco.
Descripción: Permite cambiar estado de un Parentesco en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Parentesco.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Parentesco. 4. El Sistema regresa a la pantalla de Administrar Parentesco. 	Flujo Alternativo: <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 62. Especificación Caso de Uso Cambiar Estado Parentesco

2.1.2.2.3.6.4.2.38 Especificación Caso de Uso Generar Reporte Lista Parentesco

Caso de Uso:	Generar Reporte Lista Parentesco.	
Descripción: Permite visualizar reporte.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:	Flujo Alternativo:	
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte. 		
Pos condiciones:		

Tabla 63. Especificación Caso de Uso Generar Reporte Lista Parentesco

2.1.2.2.3.6.4.2.39 Especificación Caso de Uso Administrar Provincia

Caso de Uso:	Administrar Provincia.	
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Provincia.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:	Flujo Alternativo:	
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Provincias registrados en el Sistema. 2. El actor realiza la Búsqueda de Provincia en el Listado. 3. El Sistema muestra (las) Provincia (s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Provincia”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación. 	
Pos condiciones: Ninguno		

Tabla 64. Especificación Caso de Uso Administrar Provincia

2.1.2.2.3.6.4.2.40 Especificación Caso de Uso Adicionar Provincia

Caso de Uso:	Adicionar Provincia.	
Descripción: Permite registrar nueva Provincia en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:	Flujo Alternativo:	
<ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de la Provincia en 	<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Provincia. 5. Se muestra un mensaje donde se informa que los 	

el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Provincia.	campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 65. Especificación Caso de Uso Adicionar Provincia
2.1.2.2.3.6.4.2.41 Especificación Caso de Uso Modificar Provincia

Caso de Uso:	Modificar Provincia.
Descripción: Permite modificar datos de una Provincia en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Provincia.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos de la Provincia. 3. El actor realiza los cambios de datos de la Provincia en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Provincia.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Provincia. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 66. Especificación Caso de Uso Modificar Provincia
2.1.2.2.3.6.4.2.42 Especificación Caso de Uso Eliminar Provincia

Caso de Uso:	Eliminar Provincia.
Descripción: Permite eliminar del Sistema una Provincia.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Provincia.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación de Provincia. 4. El Sistema regresa a la pantalla de Administrar Provincia.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 67. Especificación Caso de Uso Eliminar Provincia
2.1.2.2.3.6.4.2.43 Especificación Caso de Uso Cambiar Estado Provincia

Caso de Uso:	Cambiar Estado Provincia.	
Descripción: Permite cambiar estado de una Provincia en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Provincia.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Provincia. 4. El Sistema regresa a la pantalla de Administrar Provincia. 		<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.		

Tabla 68. Especificación Caso de Uso Cambiar Estado Provincia
2.1.2.2.3.6.4.2.44 Especificación Caso de Uso Generar Reporte Lista Provincia

Caso de Uso:	Generar Reporte Lista Provincia.	
Descripción: Permite visualizar reporte.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte. 		
Pos condiciones:		

Tabla 69. Especificación Caso de Uso Generar Reporte Lista Provincia
2.1.2.2.3.6.4.2.45 Especificación Caso de Uso Administrar Requisito

Caso de Uso:	Administrar Requisito.	
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Requisito.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Requisitos registrados en el Sistema. 2. El actor realiza la Búsqueda de Requisito en el Listado. 3. El Sistema muestra el(los) Requisito (s) que cumple(n) con los Criterios de Búsqueda. 		<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia "Debe Seleccionar un Requisito". 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.

<ol style="list-style-type: none"> 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	
Pos condiciones: Ninguno	

Tabla 70. Especificación Caso de Uso Administrar Requisito
2.1.2.2.3.6.4.2.46 Especificación Caso de Uso Adicionar Requisito

Caso de Uso:	Adicionar Requisito.
Descripción: Permite registrar nuevo Requisito en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Requisito en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Requisito. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Requisito. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 71. Especificación Caso de Uso Adicionar Requisito
2.1.2.2.3.6.4.2.47 Especificación Caso de Uso Modificar Requisito

Caso de Uso:	Modificar Requisito.
Descripción: Permite modificar datos de un Requisito en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Requisito.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos del Requisito. 3. El actor realiza los cambios de datos del Requisito en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Requisito. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Requisito. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.

Pos condiciones: Ninguno

Tabla 72. Especificación Caso de Uso Modificar Requisito
2.1.2.2.3.6.4.2.48 Especificación Caso de Uso Eliminar Requisito

Caso de Uso:	Eliminar Requisito.
Descripción: Permite eliminar del Sistema un Requisito.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Requisito.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Requisito. 4. El Sistema regresa a la pantalla de Administrar Requisito.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 73. Especificación Caso de Uso Eliminar Requisito
2.1.2.2.3.6.4.2.49 Especificación Caso de Uso Cambiar Estado Requisito

Caso de Uso:	Cambiar Estado Requisito.
Descripción: Permite cambiar estado de un Requisito en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Requisito.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Requisito. 4. El Sistema regresa a la pantalla de Administrar Requisito.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 74. Especificación Caso de Uso Cambiar Estado Requisito
2.1.2.2.3.6.4.2.50 Especificación Caso de Uso Generar Reporte Lista Requisito

Caso de Uso:	Generar Reporte Lista Requisito.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción	Flujo Alternativo:

Reporte. 2. El Sistema muestra el Reporte.	
Pos condiciones:	

Tabla 75. Especificación Caso de Uso Generar Reporte Lista Requisito
2.1.2.2.3.6.4.2.51 Especificación Caso de Uso Administrar Rol

Caso de Uso:	Administrar Rol.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado, Asignar Funcionalidad y Generar Reporte PDF de Rol.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Roles registrados en el Sistema. 2. El actor realiza la Búsqueda de Rol en el Listado. 3. El Sistema muestra el(los) Roles que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Rol”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 76. Especificación Caso de Uso Administrar Rol
2.1.2.2.3.6.4.2.52 Especificación Caso de Uso Adicionar Rol

Caso de Uso:	Adicionar Rol.
Descripción: Permite registrar nuevo Rol en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Rol en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Rol.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Rol. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 77. Especificación Caso de Uso Adicionar Rol

2.1.2.2.3.6.4.2.53 Especificación Caso de Uso Modificar Rol

Caso de Uso:	Modificar Rol.
Descripción: Permite modificar datos de un Rol en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Rol.	
Flujo Normal: 8. El actor presiona en el menú la opción Modificar. 9. El Sistema le muestra formulario con datos del Rol. 10. El actor realiza los cambios de datos del Rol en el formulario. 11. Selecciona botón Modificar. 12. Se valida datos del formulario que son obligatorios estén introducidos. 13. El Sistema actualiza los datos del formulario. 14. El Sistema regresa a la pantalla de Administrar Rol.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Rol. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 78. Especificación Caso de Uso Modificar Rol

2.1.2.2.3.6.4.2.54 Especificación Caso de Uso Eliminar Rol

Caso de Uso:	Eliminar Rol.
Descripción: Permite eliminar del Sistema un Rol.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Rol.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Rol. 4. El Sistema regresa a la pantalla de Administrar Rol.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 79. Especificación Caso de Uso Eliminar Rol

2.1.2.2.3.6.4.2.55 Especificación Caso de Uso Cambiar Estado Rol

Caso de Uso:	Cambiar Estado Rol.
Descripción: Permite cambiar estado de un Rol en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Rol.	
Flujo Normal:	Flujo Alternativo:

<ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Rol. 4. El Sistema regresa a la pantalla de Administrar Rol. 	<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 80. Especificación Caso de Uso Cambiar Estado Rol

2.1.2.2.3.6.4.2.56 Especificación Caso de Uso Generar Reporte Lista Rol

Caso de Uso:	Generar Reporte Lista Rol.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal:	Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte. 	
Pos condiciones:	

Tabla 81. Especificación Caso de Uso Generar Reporte Lista Rol

2.1.2.2.3.6.4.2.57 Especificación Caso de Uso Asignar Funcionalidad

Caso de Uso:	Asignar Funcionalidad.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Rol.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal:	Flujo Alternativo:
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Funcionalidades Asignadas y no Asignadas al Rol registrados en el Sistema. 2. El actor selecciona una o varias filas del Listado y hace Click en adicionar y eliminar funcionalidad. 3. Selecciona botón Asignar. 4. El Sistema guarda datos del formulario. 5. El Sistema regresa a la pantalla de Administrar Rol. 	<ol style="list-style-type: none"> 2. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia "Por Favor Seleccione una Funcionalidad para Adicionar" o Por Favor Seleccione una Funcionalidad para Eliminar. 3. Selecciona botón Cancelar, se regresa a la pantalla Administrar Rol.
Pos condiciones: Ninguno	

Tabla 82. Especificación Caso de Uso Asignar Funcionalidad

2.1.2.2.3.6.4.2.58 Especificación Caso de Uso Administrar Tipo de Empresa

Caso de Uso:	Administrar Tipo de Empresa.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de	

Tipo de Empresa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Tipos de Empresas registrados en el Sistema. 2. El actor realiza la Búsqueda de Tipo de Empresa en el Listado. 3. El Sistema muestra el(los) Tipos de Empresa(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Tipo de Empresa”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 83. Especificación Caso de Uso Administrar Tipo de Empresa
2.1.2.2.3.6.4.2.59 Especificación Caso de Uso Adicionar Tipo de Empresa

Caso de Uso:	Adicionar Tipo de Empresa.
Descripción: Permite registrar nuevo Tipo de Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Tipo de Empresa en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Tipo de Empresa.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Tipo de Empresa. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 84. Especificación Caso de Uso Adicionar Tipo de Empresa
2.1.2.2.3.6.4.2.60 Especificación Caso de Uso Modificar Tipo de Empresa

Caso de Uso:	Modificar Tipo de Empresa.
Descripción: Permite modificar datos de un Tipo de Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Tipo de Empresa.	

<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos del Tipo de Empresa. 3. El actor realiza los cambios de datos del Tipo de Empresa en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Tipo de Empresa. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Tipo de Empresa. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
<p>Pos condiciones: Ninguno</p>	

Tabla 85. Especificación Caso de Uso Modificar Tipo de Empresa
2.1.2.2.3.6.4.2.61 Especificación Caso de Uso Eliminar Tipo de Empresa

Caso de Uso:	Eliminar Tipo de Empresa.
Descripción: Permite eliminar del Sistema un Tipo de Empresa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Tipo de Empresa.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Tipo de Empresa. 4. El Sistema regresa a la pantalla de Administrar Tipo de Empresa. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 86. Especificación Caso de Uso Eliminar Tipo de Empresa
2.1.2.2.3.6.4.2.62 Especificación Caso de Uso Cambiar Estado Tipo de Empresa

Caso de Uso:	Cambiar Estado Tipo de Empresa.
Descripción: Permite cambiar estado de un Tipo de Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Tipo de Empresa.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Tipo de Empresa. 4. El Sistema regresa a la pantalla de Administrar Tipo de Empresa. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.

Pos condiciones: Cambio de Estado realizado con éxito.

Tabla 87. Especificación Caso de Uso Cambiar Estado Tipo de Empresa
2.1.2.2.3.6.4.2.63 Especificación Caso de Uso Generar Reporte Lista Tipo de Empresa

Caso de Uso:	Generar Reporte Lista Tipo de Empresa.	
Descripción: Permite visualizar reporte.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
3. El actor presiona en el menú la opción Reporte.		
4. El Sistema muestra el Reporte.		
Pos condiciones:		

Tabla 88. Especificación Caso de Uso Generar Reporte Lista Tipo de Empresa
2.1.2.2.3.6.4.2.64 Especificación Caso de Uso Administrar Tipo de Seguro

Caso de Uso:	Administrar Tipo de Seguro.	
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Tipo de Seguro.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
1. El Sistema muestra Listado de Tipos de Seguros registrados en el Sistema.		6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Tipo de Seguro”.
2. El actor realiza la Búsqueda de Tipo de Seguro en el Listado.		7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
3. El Sistema muestra el(los) Tipos de Seguro(s) que cumple(n) con los Criterios de Búsqueda.		
4. El actor selecciona una fila del Listado y hace Click derecho.		
5. El actor selecciona la opción que requiere.		
6. El Sistema verifica que se haya seleccionado una fila del listado.		
7. El Sistema solicita la confirmación de la selección de la fila.		
Pos condiciones: Ninguno		

Tabla 89. Especificación Caso de Uso Administrar Tipo de Seguro
2.1.2.2.3.6.4.2.65 Especificación Caso de Uso Adicionar Tipo de Seguro

Caso de Uso:	Adicionar Tipo de Seguro.	
Descripción: Permite registrar nuevo Tipo de Seguro en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		

<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Tipo de Seguro en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Tipo de Seguro. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Tipo de Seguro. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
<p>Pos condiciones: Ninguno</p>	

Tabla 90. Especificación Caso de Uso Adicionar Tipo de Seguro
2.1.2.2.3.6.4.2.66 Especificación Caso de Uso Modificar Tipo de Seguro

Caso de Uso:	Modificar Tipo de Seguro.
Descripción: Permite modificar datos de un Tipo de Seguro en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Tipo de Seguro.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Tipo de Seguro. 3. El actor realiza los cambios de datos del Tipo de Seguro en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El sistema actualiza los datos del formulario. 7. El sistema regresa a la pantalla de Administrar Tipo de Seguro. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Tipo de Seguro. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
<p>Pos condiciones: Ninguno</p>	

Tabla 91. Especificación Caso de Uso Modificar Tipo de Seguro
2.1.2.2.3.6.4.2.67 Especificación Caso de Uso Eliminar Tipo de Seguro

Caso de Uso:	Eliminar Tipo de Seguro.
Descripción: Permite eliminar del Sistema un Tipo de Seguro.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Tipo de Seguro.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Tipo de Seguro. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.

4. El Sistema regresa a la pantalla de Administrar Tipo de Seguro.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 92. Especificación Caso de Uso Eliminar Tipo de Seguro
2.1.2.2.3.6.4.2.68 Especificación Caso de Uso Cambiar Estado Tipo de Seguro

Caso de Uso:	Cambiar Estado Tipo de Seguro.
Descripción: Permite cambiar estado de un Tipo de Seguro en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Tipo de Seguro.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El sistema muestra el mensaje de confirmación. 3. Por Si. El sistema realiza el cambio de estado del Tipo de Seguro. 4. El sistema regresa a la pantalla de Administrar Tipo de Seguro.	Flujo Alternativo: 3. Por No. El sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 93. Especificación Caso de Uso Cambiar Estado Tipo de Seguro
2.1.2.2.3.6.4.2.69 Especificación Caso de Uso Generar Reporte Lista Tipo de Seguro

Caso de Uso:	Generar Reporte Lista Tipo de Seguro.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 94. Especificación Caso de Uso Generar Reporte Lista Tipo de Seguro
2.1.2.2.3.6.4.2.70 Especificación Caso de Uso Administrar Usuario

Caso de Uso:	Administrar Usuario.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF, Asignar Login y Clave de Usuario y Lista Personas Que no Son Usuarios..	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Usuarios registrados en el Sistema. 2. El actor realiza la Búsqueda de Rol en el	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Usuario”.

<p>Listado.</p> <ol style="list-style-type: none"> El Sistema muestra el(los) Usuario(s) que cumple(n) con los Criterios de Búsqueda. El actor selecciona una fila del Listado y hace Click derecho. El actor selecciona la opción que requiere. El Sistema verifica que se haya seleccionado una fila del listado. El Sistema solicita la confirmación de la selección de la fila. 	<ol style="list-style-type: none"> El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
<p>Pos condiciones: Ninguno</p>	

Tabla 95. Especificación Caso de Uso Administrar Usuario
2.1.2.2.3.6.4.2.71 Especificación Caso de Uso Adicionar Usuario

Caso de Uso:	Adicionar Usuario.
Descripción: Permite registrar nuevo Usuario en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona botón Adicionar. El Sistema le muestra formulario. El actor introduce datos del Usuario (que es una Persona) en el formulario. Selecciona botón Guardar. Se valida datos del formulario que son obligatorios estén introducidos. El Sistema guarda datos del formulario. El Sistema regresa a la pantalla de Administrar Usuario. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Selecciona botón Cancelar, se regresa a la pantalla Administrar Usuario. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 96. Especificación Caso de Uso Adicionar Usuario
2.1.2.2.3.6.4.2.72 Especificación Caso de Uso Modificar Usuario

Caso de Uso:	Modificar Usuario.
Descripción: Permite modificar datos de un Usuario en el sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Usuario.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona en el menú la opción Modificar. El Sistema le muestra formulario con datos del Usuario. El actor realiza los cambios de datos del Usuario en el formulario. Selecciona botón Modificar. Se valida datos del formulario que son obligatorios estén introducidos. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Selecciona botón Cancelar, se regresa a la pantalla Administrar Usuario. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.

6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Usuario.	
Pos condiciones: Ninguno	

Tabla 97. Especificación Caso de Uso Modificar Usuario
2.1.2.2.3.6.4.2.73 Especificación Caso de Uso Eliminar Usuario

Caso de Uso:	Eliminar Usuario.
Descripción: Permite eliminar del Sistema un Usuario.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Usuario.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Usuario. 4. El Sistema regresa a la pantalla de Administrar Usuario.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 98. Especificación Caso de Uso Eliminar Usuario
2.1.2.2.3.6.4.2.74 Especificación Caso de Uso Cambiar Estado Usuario

Caso de Uso:	Cambiar Estado Usuario.
Descripción: Permite cambiar estado de un Usuario en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Usuario.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Usuario. 4. El Sistema regresa a la pantalla de Administrar Usuario.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 99. Especificación Caso de Uso Cambiar Estado Usuario
2.1.2.2.3.6.4.2.75 Especificación Caso de Uso Generar Reporte Lista Usuario

Caso de Uso:	Generar Reporte Lista Usuario.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 100. Especificación Caso de Uso Generar Reporte Lista Usuario
2.1.2.2.3.6.4.2.1 Especificación Caso de Uso Generar Reporte Ficha Usuario

Caso de Uso:	Generar Reporte Ficha Usuario.
Descripción: Permite visualizar reporte de un Usuario.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Usuario.	
Flujo Normal: 1. El actor presión en el menú la opción Generar Reporte Ficha de Usuario. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 101. Especificación Caso de Uso Generar Reporte Ficha Usuario
2.1.2.2.3.6.4.2.2 Especificación Caso de Uso Gestionar Personas

Caso de Uso:	Gestionar Personas
Descripción: Permite listar, Asignar Login y Clave de Persona.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Personas que no son Usuarios registrados en el Sistema. 2. El actor realiza la Búsqueda de Persona en el Listado. 3. El Sistema muestra (las) Persona(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia "Debe Seleccionar una Persona".
Pos condiciones: Ninguno.	

Tabla 102. Especificación Caso de Uso Gestionar Personas
2.1.2.2.3.6.4.2.3 Especificación Caso de Uso Asignar Login y Clave

Caso de Uso:	Asignar Login y Clave.
Descripción: Permite registrar un Usuario.	
Actores: Administrador.	

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Asignar Login y Clave. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Usuario en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Usuario.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Usuario. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno.	

Tabla 103. Especificación Caso de Uso Asignar Login y Clave
2.1.2.2.3.6.4.2.4 Especificación Caso de Uso Administrar Zona

Caso de Uso:	Administrar Zona.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado y Generar Reporte PDF de Zona.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Zonas registrados en el Sistema. 2. El actor realiza la Búsqueda de Zona en el Listado. 3. El Sistema muestra (las) Zona(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Zona”. 7. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 104. Especificación Caso de Uso Administrar Zona
2.1.2.2.3.6.4.2.5 Especificación Caso de Uso Adicionar Zona

Caso de Uso:	Adicionar Zona.
Descripción: Permite registrar nueva Zona en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla

<ol style="list-style-type: none"> 2. El Sistema le muestra formulario. 3. El actor introduce datos de la Zona en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Zona. 	<ol style="list-style-type: none"> 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 105. Especificación Caso de Uso Adicionar Zona

2.1.2.2.3.6.4.2.6 Especificación Caso de Uso Modificar Zona

Caso de Uso:	Modificar Zona.
Descripción: Permite modificar datos de una Zona en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Zona.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos de la Zona. 3. El actor realiza los cambios de datos de la Zona en el formulario. 4. Selecciona botón Modificar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Zona. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Zona. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 106. Especificación Caso de Uso Modificar Zona

2.1.2.2.3.6.4.2.7 Especificación Caso de Uso Eliminar Zona

Caso de Uso:	Eliminar Zona.
Descripción: Permite eliminar del Sistema una Zona.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el sistema la Zona.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El sistema muestra el mensaje de confirmación. 3. Por Si. El sistema realiza la eliminación de Zona. 4. El sistema regresa a la pantalla de Administrar Zona. 	Flujo Alternativo: <ol style="list-style-type: none"> 3. Por No. El sistema no realiza ninguna operación.

Pos condiciones: Eliminación realizada con éxito.

Tabla 107. Especificación Caso de Uso Eliminar Zona

2.1.2.2.3.6.4.2.8 Especificación Caso de Uso Cambiar Estado Zona

Caso de Uso:	Cambiar Estado Zona.
Descripción: Permite cambiar estado de una Zona en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Zona.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Zona. 4. El Sistema regresa a la pantalla de Administrar Zona.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 108. Especificación Caso de Uso Cambiar Estado Zona

2.1.2.2.3.6.4.2.9 Especificación Caso de Uso Generar Reporte Lista Zona

Caso de Uso:	Generar Reporte Lista Zona.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 109. Especificación Caso de Uso Generar Reporte Lista Zona

2.1.2.2.3.6.4.3 Especificación de Casos de Uso Módulo Afiliación

2.1.2.2.3.6.4.3.1 Especificación Caso de Uso Gestionar Asegurados por Tipo Seguro Afiliados

Caso de Uso:	Gestionar Asegurados por Tipo Seguro Afiliados.
Descripción: Permite Listar Tipo de Seguros.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Tipos de Seguro registrados en el Sistema. 2. El actor selecciona un Tipo de Seguro del Listado y hace Click en Aceptar.	Flujo Alternativo:

3. El Sistema lo envía a la pantalla de Gestionar Asegurados por Tipo Seguro.	
Pos condiciones: Ninguno	

Tabla 110. Especificación Caso de Uso Gestionar Asegurados por Tipo Seguro Afiliados

2.1.2.2.3.6.4.3.2 Especificación Caso de Uso Gestionar Asegurados por Tipo Seguro

Caso de Uso:	Gestionar Asegurados por Tipo Seguro.
Descripción: Permite Listar, Gestionar Personas No Asegurados por Tipo de Seguro, Modificar, Eliminar, Dar de Baja Asegurados por Tipo Seguro.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión y haber seleccionado un Tipo de Seguro.	
Flujo Normal:	Flujo Alternativo:
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Asegurado registrados en el Sistema. 2. El actor realiza la Búsqueda de Asegurado en el Listado. 3. El Sistema muestra (los) Asegurado(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Asegurado”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 111. Especificación Caso de Uso Gestionar Asegurados por Tipo Seguro

2.1.2.2.3.6.4.3.3 Especificación Caso de Uso Gestionar Personas No Aseguradas por Tipo de Seguro

Caso de Uso:	Gestionar Personas No Asegurados por Tipo de Seguro.
Descripción: Permite Listar, Adicionar Asegurado, Asegurar Persona por Tipo de Persona.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal:	Flujo Alternativo:
<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Persona registrados en el Sistema. 2. El actor realiza la Búsqueda de Persona en el Listado. 3. El Sistema muestra (las) Persona(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la 	<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Persona”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.

selección de la fila.	
Pos condiciones: Ninguno	

Tabla 112. Especificación Caso de Uso Gestionar Personas No Asegurados por Tipo de Seguro
2.1.2.2.3.6.4.3.4 Especificación Caso de Uso Asegurar Persona por Tipo de Seguro

Caso de Uso:	Asegurar Persona por Tipo de Seguro.
Descripción: Permite registrar nuevo Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión y haber seleccionado una Persona.	
Flujo Normal: 1. El actor presión en el menú la opción Asegurar Persona por Tipo de Seguro. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Asegurado en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Personas Sin Tipo de Seguro.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Zona. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 113. Especificación Caso de Uso Asegurar Persona por Tipo de Seguro
2.1.2.2.3.6.4.3.5 Especificación Caso de Uso Adicionar Asegurado por Tipo Seguro

Caso de Uso:	Adicionar Asegurado por Tipo Seguro.
Descripción: Permite registrar nuevo Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Asegurado en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Asegurados con Tipo de Seguro.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Asegurados con Tipo de Seguro. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 114. Especificación Caso de Uso Adicionar Asegurado por Tipo Seguro
2.1.2.2.3.6.4.3.6 Especificación Caso de Uso Eliminar Asegurado por Tipo Seguro

Caso de Uso:	Eliminar Asegurado por Tipo Seguro.
Descripción: Permite eliminar del Sistema un Asegurado por Tipo Seguro.	

Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Asegurado.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Asegurado. 4. El Sistema regresa a la pantalla de Gestionar Asegurados con Tipo de Seguro.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 115. Especificación Caso de Uso Eliminar Asegurado por Tipo Seguro
2.1.2.2.3.6.4.3.7 Especificación Caso de Uso Dar de Baja Asegurado por Tipo Seguro

Caso de Uso:	Dar de Baja Asegurado por Tipo Seguro.
Descripción: Permite registrar nueva Baja de Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Dar de Baja. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema le muestra formulario. 4. El actor introduce datos de la Baja del Asegurado en el formulario. 5. Selecciona botón Guardar. 6. Se valida datos del formulario que son obligatorios estén introducidos. 7. El Sistema guarda datos del formulario. 8. El Sistema regresa a la pantalla de Gestionar Asegurados con Tipo de Seguro.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación. 5. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Asegurados con Tipo de Seguro. 6. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 116. Especificación Caso de Uso Dar de Baja Asegurado por Tipo Seguro
2.1.2.2.3.6.4.3.8 Especificación Caso de Uso Gestionar Empresas Afiliadas

Caso de Uso:	Gestionar Empresas Afiliadas.
Descripción: Permite Listar, Eliminar, Aviso Novedades, Dar de Alta por Cumplimiento a Norma, Dar de Baja por Incumplimiento a Norma Empresa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Empresa	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y

<p>registrados en el Sistema.</p> <ol style="list-style-type: none"> El actor realiza la Búsqueda de Empresa en el Listado. El Sistema muestra (las) Empresa(s) que cumple(n) con los Criterios de Búsqueda. El actor selecciona una fila del Listado y hace Click derecho. El actor selecciona la opción que requiere. El Sistema verifica que se haya seleccionado una fila del listado. El Sistema solicita la confirmación de la selección de la fila. 	<p>el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Empresa”.</p> <ol style="list-style-type: none"> El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 117. Especificación Caso de Uso Gestionar Empresas Afiliadas
2.1.2.2.3.6.4.3.9 Especificación Caso de Uso Adicionar Empresa

Caso de Uso:	Adicionar Empresa.
Descripción: Permite registrar nueva Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona botón Adicionar. El Sistema le muestra formulario. El actor introduce datos de la Empresa en el formulario. Selecciona botón Guardar. Se valida datos del formulario que son obligatorios estén introducidos. El Sistema guarda datos del formulario. El Sistema regresa a la pantalla de Gestionar Empresa. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Selecciona botón Cancelar, se regresa a la pantalla Gestionar Empresa. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 118. Especificación Caso de Uso Adicionar Empresa
2.1.2.2.3.6.4.3.10 Especificación Caso de Uso Eliminar Empresa

Caso de Uso:	Eliminar Empresa.
Descripción: Permite eliminar del Sistema una Empresa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Empresa.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona en el menú la opción Eliminar. El Sistema muestra el mensaje de confirmación. Por Si. El Sistema realiza la eliminación de la Empresa. El Sistema regresa a la pantalla de Gestionar 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Por No. El Sistema no realiza ninguna operación.

Empresa.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 119. Especificación Caso de Uso Eliminar Empresa
2.1.2.2.3.6.4.3.11 Especificación Caso de Uso Aviso Novedades Empresa

Caso de Uso:	Aviso Novedades Empresa.
Descripción: Permite registrar nueva Aviso Novedades Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Aviso de Novedades. 2. El Sistema le muestra formulario con datos de la Empresa. 3. El actor introduce datos de las Novedades de la Empresa en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Empresa.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Empresa. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 120. Especificación Caso de Uso Aviso Novedades Empresa
2.1.2.2.3.6.4.3.12 Especificación Caso de Uso Dar de Alta Empresa por Norma

Caso de Uso:	Dar de Alta Empresa por Norma.
Descripción: Permite Dar de Alta una Empresa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Empresa.	
Flujo Normal: 1. El actor presión en el menú la opción Dar de Alta por Cumplimiento a Norma. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la Alta de la Empresa. 4. El Sistema regresa a la pantalla de Gestionar Empresa.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Alta realizado con éxito.	

Tabla 121. Especificación Caso de Uso Dar de Alta Empresa por Norma
2.1.2.2.3.6.4.3.13 Especificación Caso de Uso Dar de Baja Empresa por Norma

Caso de Uso:	Dar de Baja Empresa por Norma.
Descripción: Permite Dar de Baja una Empresa en el Sistema.	

Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Empresa.	
Flujo Normal: 1. El actor presión en el menú la opción Dar de Baja por Incumplimiento a Norma. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la Baja de la Empresa. 4. El Sistema regresa a la pantalla de Gestionar Empresa.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Baja realizado con éxito.	

Tabla 122. Especificación Caso de Uso Dar de Baja Empresa por Norma
2.1.2.2.3.6.4.3.14 Especificación Caso de Uso Gestionar Asegurados Afiliados

Caso de Uso:	Gestionar Empresas Afiliadas.
Descripción: Permite Listar, Reingreso Asegurado, Gestionar Personas No Aseguradas, Dar de Baja, Emitir Certificado de Incapacidad Temporal, Emitir Autorización Medica Externa, Altas y Bajas Beneficiarios, Dar de Alta por Cumplimiento a Norma, Dar de Baja por Incumplimiento a Norma Asegurado.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Asegurados registrados en el Sistema. 2. El actor realiza la Búsqueda del Asegurado en el Listado. 3. El Sistema muestra (los) Asegurado(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Asegurado”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 123. Especificación Caso de Uso Gestionar Asegurados Afiliados
2.1.2.2.3.6.4.3.15 Especificación Caso de Uso Reingreso Asegurado

Caso de Uso:	Reingreso Asegurado.
Descripción: Permite registrar nueva Reingreso Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla

<p>Reingreso.</p> <ol style="list-style-type: none"> El Sistema le muestra formulario con datos del Asegurado. El actor introduce datos del Reingreso del Asegurado en el formulario. Selecciona botón Guardar. Se valida datos del formulario que son obligatorios estén introducidos. El Sistema guarda datos del formulario. El Sistema regresa a la pantalla de Gestionar Asegurado. 	<p>Gestionar Asegurado.</p> <ol style="list-style-type: none"> Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 124. Especificación Caso de Uso Reingreso Asegurado
2.1.2.2.3.6.4.3.16 Especificación Caso de Uso Eliminar Asegurado

Caso de Uso:	Eliminar Asegurado.
Descripción: Permite eliminar del Sistema un Asegurado.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Asegurado.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona en el menú la opción Eliminar. El sistema muestra el mensaje de confirmación. Por Si. El sistema realiza la eliminación del Asegurado. El sistema regresa a la pantalla de Gestionar Asegurado. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Por No. El sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.	

Tabla 125. Especificación Caso de Uso Eliminar Asegurado
2.1.2.2.3.6.4.3.17 Especificación Caso de Uso Dar de Baja Asegurado

Caso de Uso:	Dar de Baja Asegurado.
Descripción: Permite registrar nueva Baja de Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> El actor presiona en el menú la opción Dar de Baja. El Sistema muestra el mensaje de confirmación. Por Si. El Sistema le muestra formulario. El actor introduce datos de la Baja del Asegurado en el formulario. Selecciona botón Guardar. Se valida datos del formulario que son obligatorios estén introducidos. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> Por No. El Sistema no realiza ninguna operación. El Sistema muestra la pantalla Altas beneficiarios que deben ser dados de Baja para Dar de Baja al Asegurado. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Asegurado. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.

7. El Sistema guarda datos del formulario. 8. El Sistema regresa a la pantalla de Gestionar Asegurado.	
Pos condiciones: Ninguno	

Tabla 126. Especificación Caso de Uso Dar de Baja Asegurado
2.1.2.2.3.6.4.3.18 Especificación Caso de Uso Dar de Alta Asegurado por Norma

Caso de Uso:	Dar de Alta Asegurado por Norma.	
Descripción: Permite Dar de Alta un Asegurado en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Asegurado.		
Flujo Normal:		Flujo Alternativo:
1. El actor presión en el menú la opción Dar de Alta por Cumplimiento a Norma. 2. El sistema muestra el mensaje de confirmación. 3. Por Si. El sistema realiza la Alta del Asegurado. 4. El sistema regresa a la pantalla de Gestionar Asegurado.		3. Por No. El sistema no realiza ninguna operación.
Pos condiciones: Alta realizado con éxito.		

Tabla 127. Especificación Caso de Uso Dar de Alta Asegurado por Norma
2.1.2.2.3.6.4.3.19 Especificación Caso de Uso Dar de Baja Asegurado por Norma

Caso de Uso:	Dar de Baja Asegurado por Norma.	
Descripción: Permite Dar de Baja un Asegurado en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el sistema el Asegurado.		
Flujo Normal:		Flujo Alternativo:
1. El actor presión en el menú la opción Dar de Baja por Incumplimiento a Norma. 2. El sistema muestra el mensaje de confirmación. 3. Por Si. El sistema realiza la Baja del Asegurado. 4. El sistema regresa a la pantalla de Gestionar Asegurado.		3. Por No. El sistema no realiza ninguna operación.
Pos condiciones: Baja realizado con éxito.		

Tabla 128. Especificación Caso de Uso Dar de Baja Asegurado por Norma
2.1.2.2.3.6.4.3.20 Especificación Caso de Uso Gestionar Personas No Aseguradas

Caso de Uso:	Gestionar Personas No Aseguradas.	
Descripción: Permite Listar, Adicionar Asegurado, Asegurar Persona.		
Actores: Administrador.		

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El Sistema muestra Listado de Persona registrados en el Sistema. 2. El actor realiza la Búsqueda de Persona en el Listado. 3. El Sistema muestra (las) Persona (s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	Flujo Alternativo: <ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar una Persona”. 7. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 129. Especificación Caso de Uso Gestionar Personas No Aseguradas
2.1.2.2.3.6.4.3.21 Especificación Caso de Uso Adicionar Asegurado

Caso de Uso:	Adicionar Asegurado.
Descripción: Permite registrar nuevo Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos del Asegurado en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Asegurado. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Personas No Aseguradas. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 130. Especificación Caso de Uso Adicionar Asegurado
2.1.2.2.3.6.4.3.22 Especificación Caso de Uso Asegurar Persona

Caso de Uso:	Asegurar Persona.
Descripción: Permite registrar nuevo Asegurado en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión y haber seleccionado una Persona.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presión en el menú la opción Asegurar Persona. 2. El Sistema le muestra formulario. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Personas No Aseguradas. 5. Se muestra un mensaje donde se informa que los

<ol style="list-style-type: none"> 3. El actor introduce datos del Asegurado en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Personas No Aseguradas. 	campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 131. Especificación Caso de Uso Asegurar Persona
2.1.2.2.3.6.4.3.23 Especificación Caso de Uso Gestionar Emitir Autorización Medica Externa

Caso de Uso:	Gestionar Emitir Autorización Medica Externa.	
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión y haber seleccionado un Asegurado.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El Sistema muestra Listado de Autorización Medica Externa registrados en el Sistema. 2. El actor realiza la Búsqueda de Autorización Medica Externa en el Listado. 3. El Sistema muestra (las) Autorización Medica Externa(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Formulario”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno		

Tabla 132. Especificación Caso de Uso Gestionar Emitir Autorización Medica Externa
2.1.2.2.3.6.4.3.24 Especificación Caso de Uso Adicionar Autorización Médica Externa

Caso de Uso:	Adicionar Autorización Médica Externa.	
Descripción: Permite registrar nueva Autorización Médica Externa en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de Autorización Médica Externa en el formulario. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Autorización Médica Externa. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que

<ol style="list-style-type: none"> 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Autorización Médica Externa. 	el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 133. Especificación Caso de Uso Adicionar Autorización Médica Externa
2.1.2.2.3.6.4.3.25 Especificación Caso de Uso Modificar Autorización Médica Externa

Caso de Uso:	Modificar Autorización Médica Externa.
Descripción: Permite modificar datos de una Autorización Médica Externa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Autorización Médica Externa.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos de la Autorización Médica Externa. 3. El actor realiza los cambios de datos de la Autorización Médica Externa en el formulario. 4. Selecciona botón Actualizar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Autorización Médica Externa. 	Flujo Alternativo: <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Autorización Médica Externa. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 134. Especificación Caso de Uso Modificar Autorización Médica Externa
2.1.2.2.3.6.4.3.26 Especificación Caso de Uso Eliminar Autorización Médica Externa

Caso de Uso:	Eliminar Autorización Médica Externa.
Descripción: Permite eliminar del Sistema una Autorización Médica Externa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Autorización Médica Externa.	
Flujo Normal: <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación de la Autorización Médica Externa. 	Flujo Alternativo: <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.

4. El Sistema regresa a la pantalla de Gestionar Autorización Médica Externa.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 135. Especificación Caso de Uso Eliminar Autorización Médica Externa
2.1.2.2.3.6.4.3.27 Especificación Caso de Uso Cambiar Estado Autorización Médica Externa

Caso de Uso:	Cambiar Estado Autorización Médica Externa.
Descripción: Permite cambiar estado de Autorización Médica Externa en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema la Autorización Médica Externa.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Autorización Médica Externa. 4. El Sistema regresa a la pantalla de Gestionar Autorización Médica Externa.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 136. Especificación Caso de Uso Cambiar Estado Autorización Médica Externa
2.1.2.2.3.6.4.3.28 Especificación Caso de Uso Gestionar Emitir Certificado de Incapacidad Temporal

Caso de Uso:	Gestionar Emitir Certificado de Incapacidad Temporal.
Descripción: Permite Listar, Adicionar, Modificar, Eliminar, Cambiar Estado.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión y haber seleccionado un Asegurado.	
Flujo Normal: 1. El Sistema muestra Listado de Altas y Bajas Beneficiarios registrados en el Sistema. 2. El actor realiza la Búsqueda de Certificado de Incapacidad Temporal en el Listado. 3. El Sistema muestra (los) Certificado de Incapacidad Temporal (es) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Formulario”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.

Pos condiciones: Ninguno

Tabla 137. Especificación Caso de Uso Gestionar Emitir Certificado de Incapacidad Temporal
2.1.2.2.3.6.4.3.29 Especificación Caso de Uso Adicionar Certificado de Incapacidad Temporal

Caso de Uso:	Adicionar Certificado de Incapacidad Temporal.	
Descripción: Permite registrar nuevo Certificado de Incapacidad Temporal en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de Certificado de Incapacidad Temporal en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Certificado de Incapacidad Temporal. 		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Certificado de Incapacidad Temporal. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pues condiciones: Ninguno		

Tabla 138. Especificación Caso de Uso Adicionar Certificado de Incapacidad Temporal
2.1.2.2.3.6.4.3.30 Especificación Caso de Uso Modificar Certificado de Incapacidad Temporal

Caso de Uso:	Modificar Certificado de Incapacidad Temporal.	
Descripción: Permite modificar datos de un Certificado de Incapacidad Temporal en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Certificado de Incapacidad Temporal.		
Flujo Normal:		Flujo Alternativo:
<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Modificar. 2. El Sistema le muestra formulario con datos del Certificado de Incapacidad Temporal. 3. El actor realiza los cambios de datos del Certificado de Incapacidad Temporal en el formulario. 4. Selecciona botón Actualizar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema actualiza los datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Certificado de Incapacidad Temporal. 		<ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Certificado de Incapacidad Temporal. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno		

Tabla 139. Especificación Caso de Uso Modificar Certificado de Incapacidad Temporal

2.1.2.2.3.6.4.3.31 Especificación Caso de Uso Eliminar Certificado de Incapacidad Temporal

Caso de Uso:	Eliminar Certificado de Incapacidad Temporal.	
Descripción: Permite eliminar del Sistema un Certificado de Incapacidad Temporal.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Certificado de Incapacidad Temporal.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la eliminación del Certificado de Incapacidad Temporal. 4. El Sistema regresa a la pantalla de Gestionar Certificado de Incapacidad Temporal. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Eliminación realizada con éxito.		

Tabla 140. Especificación Caso de Uso Eliminar Certificado de Incapacidad Temporal

2.1.2.2.3.6.4.3.32 Especificación Caso de Uso Cambiar Estado Certificado de Incapacidad Temporal

Caso de Uso:	Cambiar Estado Certificado de Incapacidad Temporal.	
Descripción: Permite cambiar estado de Certificado de Incapacidad Temporal en el Sistema.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Certificado de Incapacidad Temporal.		
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado del Certificado de Incapacidad Temporal. 4. El Sistema regresa a la pantalla de Gestionar Certificado de Incapacidad Temporal. 	Flujo Alternativo:
		<ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.		

Tabla 141. Especificación Caso de Uso Cambiar Estado Certificado de Incapacidad Temporal

2.1.2.2.3.6.4.3.33 Especificación Caso de Uso Gestionar Altas y Bajas Beneficiarios

Caso de Uso:	Gestionar Altas y Bajas Beneficiarios.	
Descripción: Permite Listar, Adicionar, Eliminar, Cambiar Estado.		
Actores: Administrador.		
Precondiciones: Actor haber iniciado sesión y haber seleccionado un Asegurado.		

<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El Sistema muestra Listado de Altas y Bajas Beneficiarios registrados en el Sistema. 2. El actor realiza la Búsqueda de Altas y Bajas Beneficiarios en el Listado. 3. El Sistema muestra (las) Altas y Bajas Beneficiario(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Formulario”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
<p>Pos condiciones: Ninguno</p>	

Tabla 142. Especificación Caso de Uso Gestionar Altas y Bajas Beneficiarios

2.1.2.2.3.6.4.3.34 Especificación Caso de Uso Adicionar Altas y Bajas Beneficiarios

Caso de Uso:	Adicionar Altas y Bajas Beneficiarios.
Descripción: Permite registrar nueva Altas y Bajas Beneficiarios en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona botón Adicionar. 2. El Sistema le muestra formulario. 3. El actor introduce datos de Altas y Bajas Beneficiarios en el formulario. 4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Gestionar Altas y Bajas Beneficiarios. 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4. Selecciona botón Cancelar, se regresa a la pantalla Gestionar Altas y Bajas Beneficiarios. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
<p>Pos condiciones: Ninguno</p>	

Tabla 143. Especificación Caso de Uso Adicionar Altas y Bajas Beneficiarios

2.1.2.2.3.6.4.3.35 Especificación Caso de Uso Eliminar Altas y Bajas Beneficiarios

Caso de Uso:	Eliminar Altas y Bajas Beneficiarios.
Descripción: Permite eliminar del Sistema una Altas y Bajas Beneficiarios.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Altas y Bajas Beneficiarios.	
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de 	<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 3. Por No. El Sistema no realiza ninguna operación.

confirmación. 3. Por Si. El Sistema realiza la eliminación de Altas y Bajas Beneficiarios. 4. El Sistema regresa a la pantalla de Gestionar Altas y Bajas Beneficiarios.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 144. Especificación Caso de Uso Eliminar Altas y Bajas Beneficiarios

2.1.2.2.3.6.4.3.36 Especificación Caso de Uso Cambiar Estado Altas y Bajas Beneficiarios

Caso de Uso:	Cambiar Estado Altas y Bajas Beneficiarios.
Descripción: Permite cambiar estado de Altas y Bajas Beneficiarios en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Altas y Bajas Beneficiarios.	
Flujo Normal: 1. El actor presión en el menú la opción Cambiar Estado. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza el cambio de estado de la Altas y Bajas Beneficiarios. 4. El Sistema regresa a la pantalla de Gestionar Altas y Bajas Beneficiarios.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Cambio de Estado realizado con éxito.	

Tabla 145. Especificación Caso de Uso Cambiar Estado Altas y Bajas Beneficiarios

2.1.2.2.3.6.4.3.37 Especificación Caso de Uso Gestionar Beneficiarios Afiliados

Caso de Uso:	Gestionar Beneficiarios Afiliados.
Descripción: Permite Dar de Alta y Dar de Baja de Beneficiario.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El Sistema muestra Listado de Beneficiarios registrados en el Sistema. 2. El actor realiza la Búsqueda de Beneficiario en el Listado. 3. El Sistema muestra (los) Beneficiario(s) que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado y hace Click derecho. 5. El actor selecciona la opción que requiere. 6. El Sistema verifica que se haya seleccionado una fila del listado. 7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: 6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia “Debe Seleccionar un Beneficiario”. 7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.

Pos condiciones: Ninguno

Tabla 146. Especificación Caso de Uso Gestionar Beneficiarios Afiliados

2.1.2.2.3.6.4.3.38 Especificación Caso de Uso Dar de Alta Beneficiario por Norma

Caso de Uso:	Dar de Alta Beneficiario por Norma.
Descripción: Permite Dar de Alta un Beneficiario en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Beneficiario.	
Flujo Normal: 5. El actor presión en el menú la opción Dar de Alta por Cumplimiento a Norma. 6. El Sistema muestra el mensaje de confirmación. 7. Por Si. El Sistema realiza la Alta del Beneficiario. 8. El Sistema regresa a la pantalla de Gestionar Beneficiarios.	Flujo Alternativo: 4. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Alta realizado con éxito.	

Tabla 147. Especificación Caso de Uso Dar de Alta Beneficiario por Norma

2.1.2.2.3.6.4.3.39 Especificación Caso de Uso Dar de Baja Beneficiario por Norma

Caso de Uso:	Dar de Baja Beneficiario por Norma.
Descripción: Permite Dar de Baja un Beneficiario en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Beneficiario.	
Flujo Normal: 1. El actor presión en el menú la opción Dar de Baja por Incumplimiento a Norma. 2. El Sistema muestra el mensaje de confirmación. 3. Por Si. El Sistema realiza la Baja del Beneficiario. 4. El Sistema regresa a la pantalla de Gestionar Beneficiarios.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.
Pos condiciones: Baja realizado con éxito.	

Tabla 148. Especificación Caso de Uso Dar de Baja Beneficiario por Norma

2.1.2.2.3.6.4.4 Especificación de Casos de Uso Módulo Reporte

2.1.2.2.3.6.4.4.1 Especificación Caso de Uso Gestionar Reporte Asegurado por Tipo de Seguro

Caso de Uso:	Gestionar Reporte Asegurado por Tipo de Seguro.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	

Flujo Normal: 1. El actor introduce la fecha de reporte. 2. Selecciona el Tipo de Seguro. 3. Selecciona el Formulario para el reporte. 4. El actor presiona el Botón Generar Reporte. 5. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 149. Especificación Caso de Uso Gestionar Reporte Asegurado por Tipo de Seguro
2.1.2.2.3.6.4.4.2Especificación Caso de Uso Gestionar Reporte Empresa

Caso de Uso:	Gestionar Reporte Empresa.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor introduce la fecha de reporte. 2. Selecciona el Formulario para el reporte. 3. El actor presiona el Botón Generar Reporte. 4. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 150. Especificación Caso de Uso Gestionar Reporte Empresa
2.1.2.2.3.6.4.4.3Especificación Caso de Uso Gestionar Reporte Asegurado

Caso de Uso:	Gestionar Reporte Asegurado.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor introduce la fecha de reporte. 2. Selecciona el Formulario para el reporte. 3. El actor presiona el Botón Generar Reporte. 4. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 151. Especificación Caso de Uso Gestionar Reporte Asegurado
2.1.2.2.3.6.4.4.4Especificación Caso de Uso Gestionar Reporte Beneficiario

Caso de Uso:	Gestionar Reporte Beneficiario.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor introduce la fecha de reporte. 2. Selecciona el Formulario para el reporte. 3. El actor presiona el Botón Generar Reporte.	Flujo Alternativo:

4. El Sistema muestra el Reporte.	
Pos condiciones:	

Tabla 152. Especificación Caso de Uso Gestionar Reporte Beneficiario
2.1.2.2.3.6.4.4.5 Especificación Caso de Uso Generar Reporte Ficha Empresa

Caso de Uso:	Generar Reporte Ficha Empresa.
Descripción: Permite visualizar reporte de una Empresa.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrada en el Sistema la Empresa.	
Flujo Normal: 1. El Sistema le muestra formulario. 2. El actor introduce Nro. Empresa. 3. Presiona botón Cargar. 4. El Sistema muestra el Reporte. 5. Presiona la opción Imprimir.	Flujo Alternativo:
Pos condiciones:	

Tabla 153. Especificación Caso de Uso Generar Reporte Ficha Empresa
2.1.2.2.3.6.4.4.6 Especificación Caso de Uso Generar Reporte Ficha Asegurado

Caso de Uso:	Generar Reporte Ficha Asegurado.
Descripción: Permite visualizar reporte de un Asegurado.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Asegurado.	
Flujo Normal: 1. El Sistema le muestra formulario. 2. El actor introduce Nro. Asegurado. 3. Presiona botón Cargar. 4. El Sistema muestra el Reporte. 5. Presiona la opción Imprimir.	Flujo Alternativo:
Pos condiciones:	

Tabla 154. Especificación Caso de Uso Generar Reporte Ficha Asegurado
2.1.2.2.3.6.4.4.7 Especificación Caso de Uso Generar Reporte Ficha Asegurado por Tipo de Seguro

Caso de Uso:	Generar Reporte Ficha Asegurado.
Descripción: Permite visualizar reporte de un Asegurado.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Asegurado.	
Flujo Normal: 1. El Sistema le muestra formulario. 2. El actor introduce Nro. Asegurado. 3. Presiona botón Cargar. 4. El Sistema muestra el Reporte. 5. Presiona la opción Imprimir.	Flujo Alternativo:

Pos condiciones:

Tabla 155. Especificación Caso de Uso Generar Reporte Ficha Asegurado por Tipo de Seguro

2.1.2.2.3.6.4.4.8 Especificación Caso de Uso Generar Reporte Estadístico

Caso de Uso:	Generar Reporte Estadístico.
Descripción: Permite visualizar reporte estadístico.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none">1. El Sistema le muestra formulario.2. El actor introduce Fecha de Inicio y Fecha Final.3. Presiona botón Generar Reporte.4. El Sistema muestra el Reporte.5. Presiona la opción Imprimir.	Flujo Alternativo:
Pos condiciones:	

Tabla 156. Especificación Caso de Uso Generar Reporte Estadístico

2.1.2.2.3.6.4.5 Especificación de Casos de Uso Módulo Backup

2.1.2.2.3.6.4.5.1 Especificación Caso de Uso Administrar Backup

Caso de Uso:	Administrar Backup.
Descripción: Permite Listar, Resguardar, Restaurar, Eliminar, Cambiar Estado y Generar Reporte PDF de Backup.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: <ol style="list-style-type: none">1. El Sistema muestra Listado de Backup's registrados en el Sistema.2. El actor realiza la Búsqueda de Backup en el Listado.3. El Sistema muestra (los) Backup(s) que cumple(n) con los Criterios de Búsqueda.4. El actor selecciona una fila del Listado y hace Click derecho.5. El actor selecciona la opción que requiere.6. El Sistema verifica que se haya seleccionado una fila del listado.7. El Sistema solicita la confirmación de la selección de la fila.	Flujo Alternativo: <ol style="list-style-type: none">6. El actor no ha seleccionado ninguna fila del Listado y el Sistema muestra un mensaje de advertencia "Debe Seleccionar un Backup".7. El actor decide no confirmar selección y el Sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 157. Especificación Caso de Uso Administrar Backup

2.1.2.2.3.6.4.5.2 Especificación Caso de Uso Resguardar Backup

Caso de Uso:	Resguardar Backup.
Descripción: Permite resguardar un nueva Backup en el Sistema.	
Actores: Administrador.	

Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 6. El actor presiona botón Resguardar. 7. El Sistema le muestra formulario. 8. El actor introduce datos del Backup en el formulario. 9. Selecciona botón Resguardar. 10. Se valida datos del formulario que son obligatorios estén introducidos. 11. El Sistema guarda datos del formulario. 12. El Sistema regresa a la pantalla de Administrar Backup.	Flujo Alternativo: 6. Selecciona botón Cancelar, se regresa a la pantalla Administrar Backup. 7. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 158. Especificación Caso de Uso Resguardar Backup

2.1.2.2.3.6.4.5.3 Especificación Caso de Uso Restaurar Backup

Caso de Uso:	Restaurar Backup.
Descripción: Permite restaurar un Backup en el Sistema.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Backup.	
Flujo Normal: 1. El actor presiona en el menú la opción Restaurar. 2. El Sistema le muestra formulario con datos del Backup. 3. El actor realiza la introducción de datos del Backup en el formulario. 4. Selecciona botón Restaurar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El Sistema guarda datos del formulario. 7. El Sistema regresa a la pantalla de Administrar Backup.	Flujo Alternativo: 4. Selecciona botón Cancelar, se regresa a la pantalla Administrar Backup. 5. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 159. Especificación Caso de Uso Restaurar Backup

2.1.2.2.3.6.4.5.4 Especificación Caso de Uso Eliminar Backup

Caso de Uso:	Eliminar Backup.
Descripción: Permite eliminar del Sistema una Backup.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión; Estar registrado en el Sistema el Backup.	
Flujo Normal: 1. El actor presiona en el menú la opción Eliminar. 2. El Sistema muestra el mensaje de confirmación.	Flujo Alternativo: 3. Por No. El Sistema no realiza ninguna operación.

3. Por Si. El Sistema realiza la eliminación del Backup.	
4. El Sistema regresa a la pantalla de Administrar Backup.	
Pos condiciones: Eliminación realizada con éxito.	

Tabla 160. Especificación Caso de Uso Eliminar Backup
2.1.2.2.3.6.4.5 Especificación Caso de Uso Generar Reporte Lista Backup

Caso de Uso:	Generar Reporte Lista Backup.
Descripción: Permite visualizar reporte.	
Actores: Administrador.	
Precondiciones: Actor haber iniciado sesión.	
Flujo Normal: 1. El actor presiona en el menú la opción Reporte. 2. El Sistema muestra el Reporte.	Flujo Alternativo:
Pos condiciones:	

Tabla 161. Especificación Caso de Uso Generar Reporte Lista Backup
2.1.2.2.3.7 Especificaciones Adicionales

Este documento capturará todos los requisitos que no han sido incluidos como parte de los Casos de Uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc. **Ver Anexo: Especificación de Requisitos de Software Norma IEEE835**

2.1.2.2.3.8 Prototipos de Interfaces de Usuario

2.1.2.2.3.8.1 Introducción

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del Sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la Fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la Fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

2.1.2.2.3.8.2 Propósito

Presentar los prototipos de pantallas para que el usuario tenga una idea de la interfaz que se presentaran en el Sistema.

2.1.2.2.3.8.3 Alcance

Mostrar los Prototipos de Pantallas, sujeto a modificaciones a lo largo del desarrollo del Sistema.

2.1.2.2.3.8.4 Prototipo de Pantallas

2.1.2.2.3.8.4.1 Pantalla Ingresar al Sistema



Figura 80. Pantalla Ingresar al Sistema

2.1.2.2.3.8.4.2 Pantalla Principal

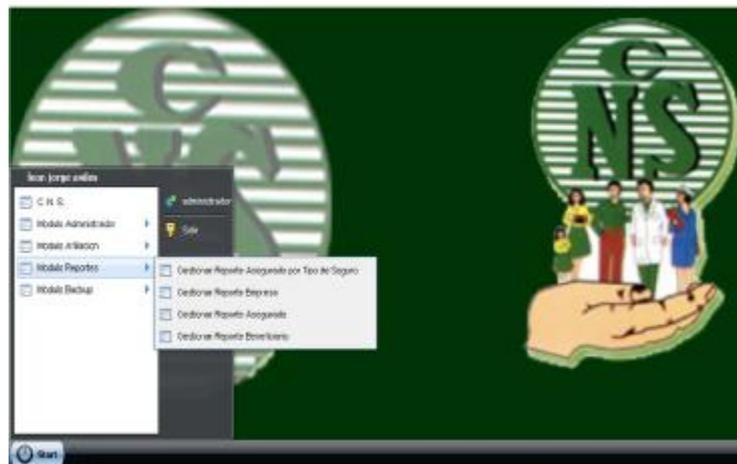


Figura 81. Pantalla Principal

2.1.2.2.3.8.4.3 Pantalla Administrar Actividad Económica



Figura 82. Pantalla Administrar Actividad Económica

2.1.2.2.3.8.4.4 Pantalla Formulario (ABM) Actividad Económica

Formulario Actividad Económica

Nombre *

Descripcion

Guardar Cancelar

Figura 83. Pantalla Formulario (ABM) Actividad Económica

2.1.2.2.3.8.4.5 Pantalla Reporte Actividad Económica

Reporte Actividad Económica

LISTA DE ACTIVIDADES ECONOMICAS

ID	NOMBRE	DESCRIPCION	ESTADO
1	Hospital	Hospital	Activo

Figura 84. Pantalla Reporte Actividad Económica

2.1.2.2.3.8.4.6 Pantalla Administrar Centro Salud

Lista Centro de Salud

Filtro Nombre Buscar

ID	NUMERO	NOMBRE	DESCRIPCION	DIRECCION	TELEFONO	FECHA REGISTRO	ESTADO
1		HOSPITAL	HOSPITAL	CALLE POTOSI	66-95632	2009-10-21	ACTIVO

Modificar
Eliminar
Cambiar Estado
Generar Reporte

Agregar

Figura 85. Pantalla Administrar Centro Salud

2.1.2.2.3.8.4.7 Pantalla Formulario (ABM) Centro Salud

Formulario Centro de Salud

Numero *	<input type="text" value="0"/>
Nombre *	<input type="text"/>
Descripcion	<input type="text"/>
Direccion *	<input type="text"/>
Telefono	<input type="text"/>

Figura 86. Pantalla Formulario (ABM) Centro Salud

2.1.2.2.3.8.4.8 Pantalla Reporte Centro Salud

Reporte Centro de Salud

Volver

LISTA DE CENTROS DE SALUD

UBICACION	NOMBRE	DIRECCION	TELEFONO	ESTADO
1	SE	HOSPITAL HERPES	0442 87008	ACTIVO

Figura 87. Pantalla Reporte Centro Salud

2.1.2.2.3.8.4.9 Pantalla Administrar Departamento

Lista Departamentos

Filtro Nombre

ID	NOMBRE	DESCRIPCION
2	COCHABAMBA	NINGUNA
1	TARAJA	NINGUNA

Figura 88. Pantalla Administrar Departamento

2.1.2.2.3.8.4.10 Pantalla Reporte Departamento

Reporte Departamento

Volver

LISTA DE DEPARTAMENTOS

ID	NOMBRE	DESCRIPCION
1	UNIV	NINGUNA
2	COCHABAMBA	NINGUNA

Figura 89. Pantalla Reporte Departamento

2.1.2.2.3.8.4.11 Pantalla Administrar Funcionalidad

ID	NOMBRE	LINK	MODULO	DESCRIPCION	ESTADO
1.3	ADMINISTRAR FUNCIONALIDAD	ListaFuncionalidad	ADMINISTRAD	NINGUNA	ACTIVO
1.2	ADMINISTRAR MODULO	ListaModuloView	ADMINISTRAD	NINGUNA	ACTIVO
1	ADMINISTRAR REQUISITO	ListaRequisitoV	ADMINISTRAD	NINGUNA	ACTIVO
1.1	ADMINISTRAR USUARIO	ListaUsuariosV	ADMINISTRAD	NINGUNA	ACTIVO
1.0	ADMINISTRAR ZONA	ListaZonaView	ADMINISTRAD	NINGUNA	ACTIVO

Figura 90. Pantalla Administrar Funcionalidad

2.1.2.2.3.8.4.12 Pantalla Reporte Funcionalidad

ID	NOMBRE	LINK	DESCRIPCION	MODULO	ESTADO
1.3	ADMINISTRAR FUNCIONALIDAD	ListaFuncionalidad	NINGUNA	ADMINISTRAD	ACTIVO
1.2	ADMINISTRAR MODULO	ListaModuloView	NINGUNA	ADMINISTRAD	ACTIVO
1	ADMINISTRAR REQUISITO	ListaRequisitoV	NINGUNA	ADMINISTRAD	ACTIVO
1.1	ADMINISTRAR USUARIO	ListaUsuariosV	NINGUNA	ADMINISTRAD	ACTIVO
1.0	ADMINISTRAR ZONA	ListaZonaView	NINGUNA	ADMINISTRAD	ACTIVO

Figura 91. Pantalla Reporte Funcionalidad

2.1.2.2.3.8.4.13 Pantalla Administrar Módulo

ID	NOMBRE	DESCRIPCION	ESTADO
1	ADMINISTRADOR	ADMINISTRACION DEL SISTEMA	ACTIVO

Figura 92. Pantalla Administrar Módulo

2.1.2.2.3.8.4.14 Pantalla Reporte Módulo

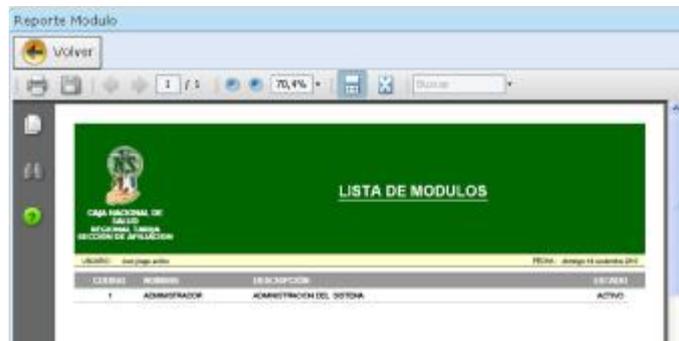


Figura 93. Pantalla Reporte Módulo

2.1.2.2.3.8.4.15 Pantalla Administrar Municipio Localidad



Figura 94. Pantalla Administrar Municipio Localidad

2.1.2.2.3.8.4.16 Pantalla Formulario (ABM) Municipio Localidad



Figura 95. Pantalla Formulario (ABM) Municipio Localidad

2.1.2.2.3.8.4.17 Pantalla Reporte Municipio Localidad

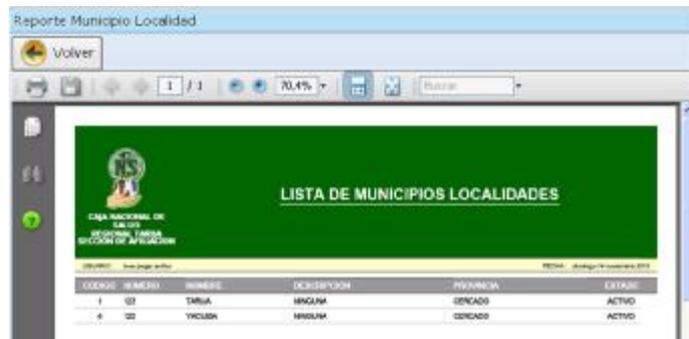


Figura 96. Pantalla Reporte Municipio Localidad

2.1.2.2.3.8.4.18 Pantalla Administrar Ocupación

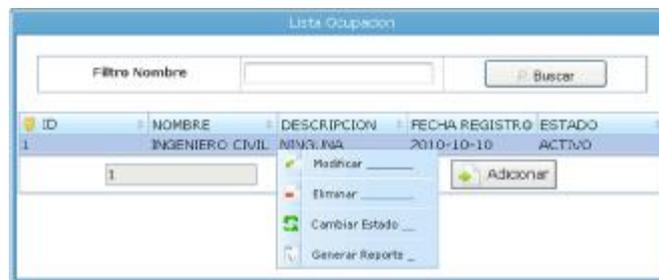


Figura 97. Pantalla Administrar Ocupación

2.1.2.2.3.8.4.19 Pantalla Formulario (ABM) Ocupación



Figura 98. Pantalla Formulario (ABM) Ocupación

2.1.2.2.3.8.4.20 Pantalla Reporte Ocupación

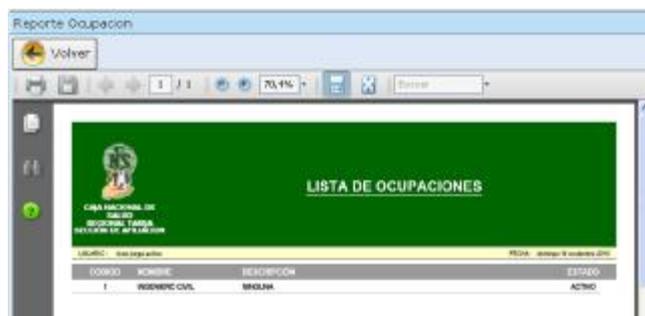


Figura 99. Pantalla Reporte Ocupación

2.1.2.2.3.8.4.21 Pantalla Administrar Parentesco

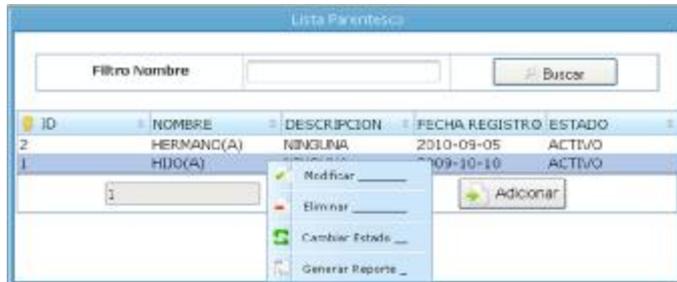


Figura 100. Pantalla Administrar Parentesco

2.1.2.2.3.8.4.22 Pantalla Formulario (ABM) Parentesco



Figura 101. Pantalla Formulario (ABM) Parentesco

2.1.2.2.3.8.4.23 Pantalla Reporte Parentesco

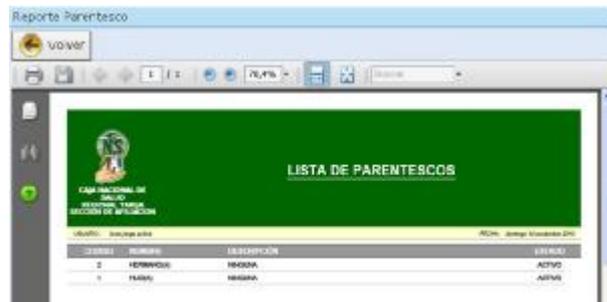


Figura 102. Pantalla Reporte Parentesco

2.1.2.2.3.8.4.24 Pantalla Administrar Provincia

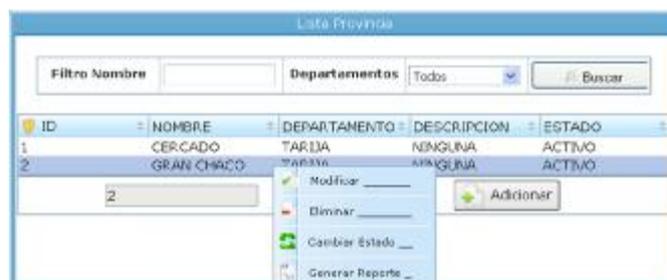


Figura 103. Pantalla Administrar Provincia

2.1.2.2.3.8.4.25 Pantalla Formulario (ABM) Provincia

Formulario Provincia

Nombre *

Descripcion

Departamentos * TAREJA

Guardar Cancelar

Figura 104. Pantalla Formulario (ABM) Provincia

2.1.2.2.3.8.4.26 Pantalla Reporte Provincia

Reporte Provincia

Volver

LISTA DE PROVINCIAS

CODIGO	NOMBRE	DESCRIPCION	PROVINCIA	ESTADO
1	GUAYAMA	GUAYAMA	TURIS	ACTIVO
2	GUAYAMA	GUAYAMA	TURIS	ACTIVO

Figura 105. Pantalla Reporte Provincia

2.1.2.2.3.8.4.27 Pantalla Administrar Requisito

Lista Requisito

Filtro Nombre Tipo Seguro Todos Buscar

ID	REQUISITO	TIPOSEGURO	DESCRIPCION	FECHA REGIST	ESTADO
1	CEDULA IDENTIDAD	ADULTO MAYOR	NINGUNO	2010-07-31	ACTIVO
6	CERTIFICADO TRABAJO	ADULTO MAYOR	NINGUNO	2010-10-10	ACTIVO
3		ADULTO MAYOR	NINGUNO	2010-10-10	ACTIVO
5		ADULTO MAYOR	NINGUNO	2010-10-10	ACTIVO
2		ADULTO MAYOR	NINGUNO	2010-10-10	ACTIVO

Modificar Eliminar Cambiar Estado Generar Reporte

Adicionar

Figura 106. Pantalla Administrar Requisito

2.1.2.2.3.8.4.28 Pantalla Formulario (ABM) Requisito

Formulario Requisito

Nombre Requisito *

Descripcion

Tipos de Seguros * ADULTO MAYOR

Guardar Cancelar

Figura 107. Pantalla Formulario (ABM) Requisito

2.1.2.2.3.8.4.29 Pantalla Reporte Requisito

Reporte Requisito

LISTA DE PROVINCIAS

CONVUL. SECRETO	SITUACION	TIPO DE SEGURO	ESTADO
1. SIDA	SIDA	ADULTO MAYOR	ACTIVO
2. OSELA ENTIDAD	INGLAD	ADULTO MAYOR	ACTIVO
3. DISFORDO TRANKO	INGLAD	ADULTO MAYOR	ACTIVO

Figura 108. Pantalla Reporte Requisito

2.1.2.2.3.8.4.30 Pantalla Administrar Rol

Lista Rol

Filtro Nombre Buscar

ID	NOMBRE	DESCRIPCION	FECHA REGISTRO	ESTADO
1	ADMINISTRACION	ADMINISTRACION	10-10-10	ACTIVO

Modificar
Eliminar
Cambiar Estado
Asignar Personalidad
Generar Reporte

Adicionar

Figura 109. Pantalla Administrar Rol

2.1.2.2.3.8.4.31 Pantalla Formulario (ABM) Rol

Formulario Rol

Nombre *

Descripcion

Guardar Cancelar

Figura 110. Pantalla Formulario (ABM) Rol

2.1.2.2.3.8.4.32 Pantalla Asignar Funcionalidad



Figura 111. Pantalla Asignar Funcionalidad

2.1.2.2.3.8.4.33 Pantalla Reporte Rol

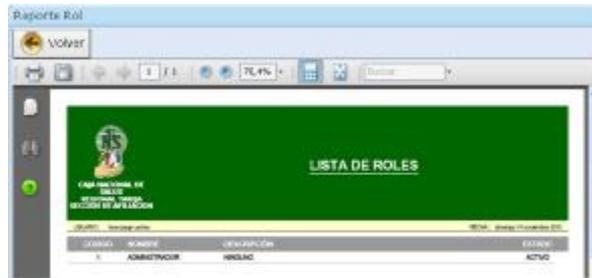


Figura 112. Pantalla Reporte Rol

2.1.2.2.3.8.4.34 Pantalla Administrar Tipo Empresa



Figura 113. Pantalla Administrar Tipo Empresa

2.1.2.2.3.8.4.35 Pantalla Formulario (ABM) Tipo Empresa

Formulario Tipo de Empresa

Nombre *

Descripcion

Guardar Cancelar

Figura 114. Pantalla Formulario (ABM) Tipo Empresa

2.1.2.2.3.8.4.36 Pantalla Administrar Tipo Seguro

Lista Tipo de Seguro

Filtro Nombre Buscar

ID	NOMBRE	DESCRIPCION	FECHA REGISTRO	ESTADO
1	ADULTO MAYOR	MINORADO	2010-10-10	ACTIVO
2	MATERNO INFANTIL		2010-10-10	INACTIVO

Adicionar

Figura 115. Pantalla Administrar Tipo Seguro

2.1.2.2.3.8.4.37 Pantalla Formulario (ABM) Tipo Seguro

Formulario Tipo de Seguro

Nombre *

Descripcion

Guardar Cancelar

Figura 116. Pantalla Formulario (ABM) Tipo Seguro

2.1.2.2.3.8.4.38 Pantalla Reporte Tipo Seguro

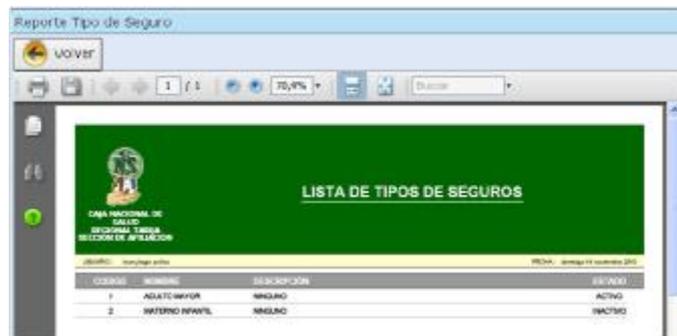


Figura 117. Pantalla Reporte Tipo Seguro

2.1.2.2.3.8.4.39 Pantalla Administrar Usuario

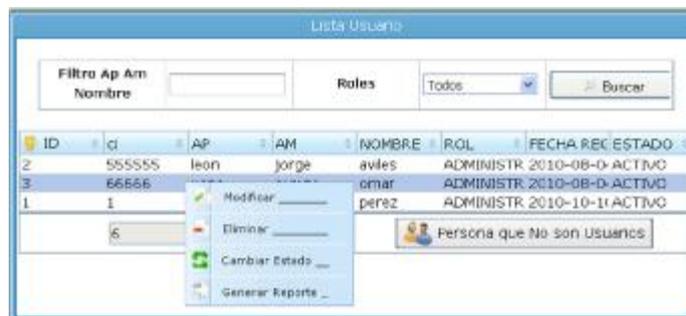


Figura 118. Pantalla Administrar Usuario

2.1.2.2.3.8.4.40 Pantalla Gestión Personas que no son Usuarios



Figura 119. Pantalla Gestión Personas que no son Usuarios

2.1.2.2.3.8.4.41 Pantalla Formulario (ABM) Usuario

The screenshot shows two overlapping forms. The top form, titled 'Datos Personales', contains the following fields: CI *, Nombre *, Ape. Pat., Ape. Mat. *, Fecha Nac. *, Edad *, Sexo * (dropdown menu with 'Masculino' selected), and Estado Civil * (dropdown menu with 'Soltero(e)' selected). The bottom form, titled 'Formulario Usuario', contains: Username *, Password *, Password Verificar *, Roles * (dropdown menu with 'ADMINISTRADO' selected), and buttons for 'Guardar' and 'Cancelar'.

Figura 120. Pantalla Formulario (ABM) Usuario

2.1.2.2.3.8.4.42 Pantalla Asignar Login y Clave

The screenshot shows the 'Formulario Usuario' form with the following fields: Usuario : (text field with 'JIMBA MARCO ANASTAYO'), Username *, Password *, Password Verificar *, and Roles * (dropdown menu with 'ADMINISTRADOR' selected). Buttons for 'Guardar' and 'Cancelar' are at the bottom.

Figura 121. Pantalla Asignar Login y Clave

2.1.2.2.3.8.4.43 Pantalla Reporte Usuario

The screenshot shows the 'Reporte Usuario' screen with a table of system users. The table has columns for ID, CI, NOMBRE, ROL, FECHA, and ESTADO. The data is as follows:

ID	CI	NOMBRE	ROL	FECHA	ESTADO
1	1	MERCADO JUAN BRITO	ADMINISTRADOR	2010-01-01	ACTIVO
2	9999	FERRER GREGO DAMIEN	ADMINISTRADOR	2010-01-01	ACTIVO
3	3333	MENDOZA RONALD AFRICA	ADMINISTRADOR	2010-11-01	ACTIVO
4	0000	LEON JORGE PAUL	ADMINISTRADOR	2010-01-01	ACTIVO
5	999999	MERCADO DEBATE CARMEN	ADMINISTRADOR	2010-12-01	ACTIVO

Figura 122. Pantalla Reporte Usuario

2.1.2.2.3.8.4.44 Pantalla Reporte Ficha Usuario



Figura 123. Pantalla Reporte Ficha Usuario

2.1.2.2.3.8.4.45 Pantalla Administrar Zona



Figura 124. Pantalla Administrar Zona

2.1.2.2.3.8.4.46 Pantalla Formulario (ABM) Zona



Figura 125. Pantalla Formulario (ABM) Zona

2.1.2.2.3.8.4.47 Pantalla Reporte Zona

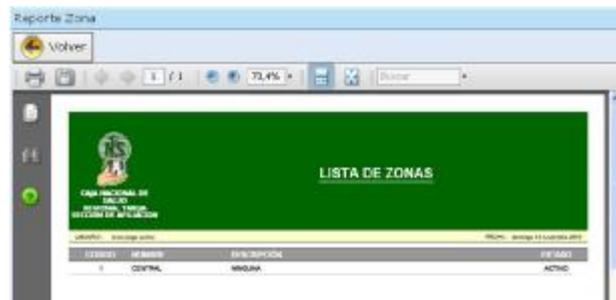


Figura 126. Pantalla Reporte Zona

2.1.2.2.3.8.4.48 Pantalla Asegurados por Tipo de Seguro

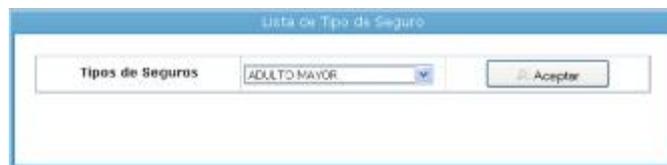


Figura 127. Pantalla Asegurados por Tipo de Seguro

2.1.2.2.3.8.4.49 Pantalla Gestionar Asegurados con Tipo de Seguro



Figura 128. Pantalla Gestionar Asegurados con Tipo de Seguro

2.1.2.2.3.8.4.50 Pantalla Gestionar Personas sin Tipo de Seguro

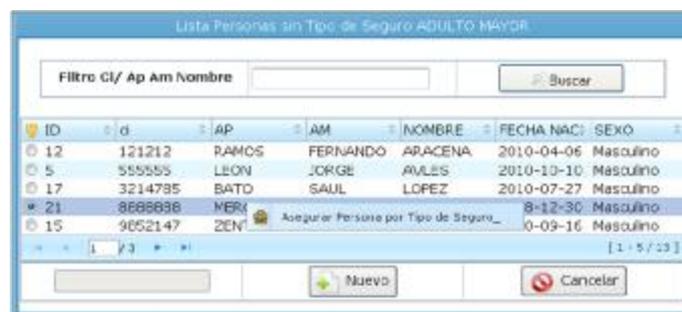


Figura 129. Pantalla Gestionar Personas sin Tipo de Seguro

2.1.2.2.3.8.4.51 Pantalla Formulario (ABM) Asegurado con Tipo de Seguro (FormAVC04A)

Formulario Alta Asegurado

Tipo de Seguro : ADULTO MAYOR

Nro. Formulario

Ap. Paterno * Nombre *

Ap. Materno * Numero Asegurado *

Fecha Nacimiento * Sexo *

Edad * Estado Civil *

Documento Identidad * Numero *

Zona * Numero *

Calle * Localidad *

Departamento * Municipio *

Provincia * Numero Municipio *

Lugar * Fecha * Fecha Recapcion *

Observaciones

ID	MARCAR	REQUISITO	DESCRIPCION
2	<input type="checkbox"/>	DDDD	NINGUNO

Guardar Cancelar

Figura 130. Pantalla Formulario (ABM) Asegurado con Tipo de Seguro (FormAVC04A)

2.1.2.2.3.8.4.52 Pantalla Formulario Baja Asegurado (FormAVC07A)

Formulario Baja Asegurado

Tipo de Seguro : ADULTO MAYOR

Nro. Formulario

Ap. Paterno * Nombre *

Ap. Materno * Numero Asegurado *

Fecha de Baja * Motivo de Baja *

Municipio * Departamento *

Provincia * Numero Municipio *

Lugar *

Guardar Cancelar

Observaciones

Figura 131. Pantalla Formulario Baja Asegurado (FormAVC07A)

2.1.2.2.3.8.4.53 Pantalla Gestionar Empresa

Lista Empresa

Filtro Nro / Nombre: Actividad Económica:

ID	Nro. Empl	Nombre o Actividad	Actividad Económica	Tipo Empl	Fecha Inic	FECHA RE	ESTADO	NORMA
4	10-10-10	IMPORTAC CCC	MINERA	INDIVIDUAL	2010-09-	2010-09-	ACTIVO	INACTIVO
3		Eliminar			0-07-	2010-08-	ACTIVO	ACTIVO
2		Aviso Novedades			0-07-	2010-08-	ACTIVO	ACTIVO

Figura 132. Pantalla Gestionar Empresa

2.1.2.2.3.8.4.54 Pantalla Formulario (ABM) Empresa (FormAVC01)

Formulario Alta Empresa

Nro. Formulario:

Nombre o Razón Social Empleador *: Numero Empleador *:

Ubicación Centro de Trabajo

Departamento *: Provincia *:

Localidad *: Tipo Empresa *:

Propietario o Representante Legal

Nueva Persona

Nombre *:

Ap. Paterno *:

Ap. Materno *:

Fecha Nacimiento *:

Edad *:

Sexo *:

Estado Civil *:

Propietario o Representante Legal

Buscar Persona

Buscar CI / Ap. Pat. Nombre:

Domicilio Legal

Zona *: Numero *:

Calle *: Telefono *:

Actividad Económica *: Fecha Inicio Actividad *:

Numero Trabajadores *: Nro. Padron Renta *:

Lugar Presentación *:

Fecha Presentación *:

Fecha Recipien *:

Observaciones:

Figura 133. Pantalla Formulario (ABM) Empresa
2.1.2.2.3.8.4.55 Pantalla Formulario Aviso de Novedades (FormAVC03)

Figura 134. Pantalla Formulario Aviso de Novedades (FormAVC03)

2.1.2.2.3.8.4.56 Pantalla Gestionar Asegurados

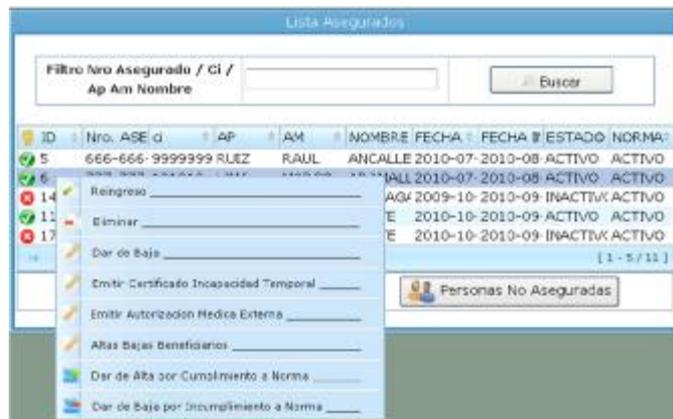


Figura 135. Pantalla Gestionar Asegurados

2.1.2.2.3.8.4.57 Pantalla Gestionar Personas No Aseguradas



Figura 136. Pantalla Gestionar Personas No Aseguradas

2.1.2.2.3.8.4.58 Pantalla Formulario (ABM-Reingreso) Asegurados (FormAVC04)

Calle * **Numero ***

Departamento * **Municipio ***

Provincia * **Numero Municipio***

Ocupacion * **INGENIERO CIVIL ***

Salario Mensual * **Fecha Ingreso Trabajo ***

Nombre o Razon Social del Empleador * **Numero del Empleador ***

Lugar Presentacion *

Fecha Presentacion * **Fecha Recepcion ***

Observaciones

Marcas	A.P.	A.M.	NOBRE F. NAC.	PARENTE F. EXT.	CODIGO	CAUSAS
<input type="checkbox"/> S/N	<input type="text"/>					
<input type="checkbox"/> S/N	<input type="text"/>					
<input type="checkbox"/> S/N	<input type="text"/>					
<input type="checkbox"/> S/N	<input type="text"/>					
<input type="checkbox"/> S/N	<input type="text"/>					

Figura 137. Pantalla Formulario (ABM-Reingreso) Asegurados (FormAVC04)
2.1.2.2.3.8.4.59 Pantalla Formulario Dar de Baja Asegurado (FormAVC07)

Nro. Formulario

Ap. Paterno * **Nombre ***

Ap. Materno * **Numero Asegurado ***

Salario Retiro * **Fecha de Baja ***

Motivo de Baja *

Lugar Presentacion * **Fecha Presentacion ***

Fecha Recepcion *

Observaciones

Figura 138. Pantalla Formulario Dar de Baja Asegurado (FormAVC07)

2.1.2.2.3.8.4.60 Pantalla Gestión Certificado Incapacidad Temporal



Figura 139. Pantalla Gestión Certificado Incapacidad Temporal

2.1.2.2.3.8.4.61 Pantalla Formulario (ABM) Certificado Incapacidad Temporal (FormAVC09)

Formulario Certificado de Incapacidad Temporal

Nro. Formulario: []

Ap. Paterno *: [] Nombre *: []

Ap. Materno *: [] Numero Assegurado *: []

Nombre o Razon Social del Empleador *: [] Numero Empleador *: []

Riesgo *: [] PROFESIONAL

INCAPACIDAD DEBE *: []

INCAPACIDAD HASTA *: []

DIAS INCAPACIDAD *: []

CLAVE *: []

SALARIO *: []

IMPORTE SUBSIDIO *: []

Lugar *: [] Fecha *: []

Observaciones: []

Guardar Cancelar

Figura 140. Pantalla Formulario (ABM) Certificado Incapacidad Temporal (FormAVC09)

2.1.2.2.3.8.4.62 Pantalla Gestionar Autorización Médica Externa



Figura 141. Pantalla Gestionar Autorización Médica Externa

2.1.2.2.3.8.4.63 Pantalla Formulario (ABM) Autorización Médica Externa (FormAVC08)

Figura 142. Pantalla Formulario (ABM) Autorización Médica Externa (FormAVC08)

2.1.2.2.3.8.4.64 Pantalla Gestionar Altas y Bajas Beneficiarios

Figura 143. Pantalla Gestionar Altas y Bajas Beneficiarios

2.1.2.2.3.8.4.65 Pantalla Formulario (ABM) Altas y Bajas Beneficiarios (FormAVC06)

Figura 144. Pantalla Formulario (ABM) Altas y Bajas Beneficiarios (FormAVC06)

2.1.2.2.3.8.4.66 Pantalla Gestionar Beneficiarios

Figura 145. Pantalla Gestionar Beneficiarios

2.1.2.2.3.8.4.67 Pantalla Reporte Asegurado por Tipo de Seguro



Figura 146. Pantalla Reporte Asegurado por Tipo de Seguro

2.1.2.2.3.8.4.68 Pantalla Reporte Empresa

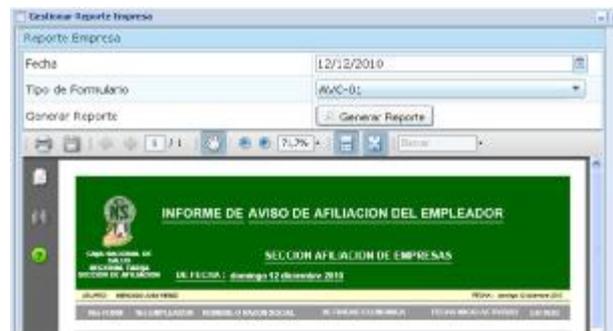


Figura 147. Pantalla Reporte Empresa

2.1.2.2.3.8.4.69 Pantalla Reporte Asegurado

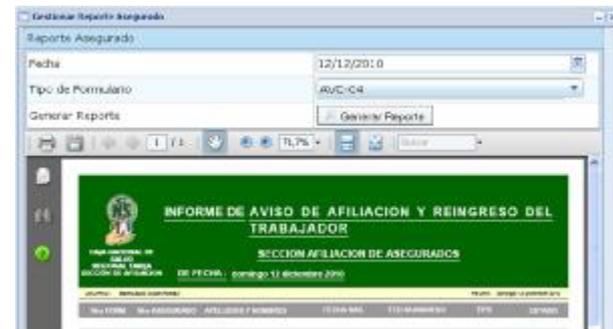


Figura 148. Pantalla Reporte Asegurado

2.1.2.2.3.8.4.70 Pantalla Reporte Beneficiario

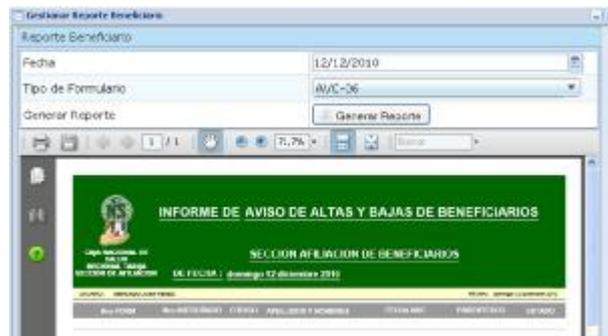


Figura 149. Pantalla Reporte Beneficiario

2.1.2.2.3.8.4.71 Pantalla Reporte Ficha Empresa



Figura 150. Pantalla Reporte Ficha Empresa

2.1.2.2.3.8.4.72 Pantalla Reporte Ficha Asegurado



Figura 151. Pantalla Reporte Ficha Asegurado

2.1.2.2.3.8.4.73 Pantalla Reporte Ficha Asegurado por Tipo de Seguro

Reporte Ficha Asegurado por Tipo Seguro

Reporte Ficha Asegurado por Tipo Seguro

Nro. Asegurado: 988-988-888

FICHA DE ASEGURADO [TIPO DE SEGURO]

EMPLEADOR			
Código Asegurado	3	Nro. Asegurado	666-666-666
C.I.	9999999	Apellidos y Nombres	RUIZ PAUL ANCALLE
Fecha Nac.	2010-07-29	Edad	21
Sexo	Edadulano	Estado Civil	Soltero(a)
Estado	ACTIVO	Fecha Registro	2010-08-12

HISTORIAL DE AVCO4A : SEGURO DE SALUD PARA [ADULTO MAYOR]

Nro. Funcionario	Estado	Tipo de Seguro	Fecha Recepcion	Observacion
5	ACTIVO	ADULTO MAYOR	2010-08-05	NINGUNA

Figura 152. Pantalla Reporte Ficha Asegurado por Tipo de Seguro

2.1.2.2.3.8.4.74 Pantalla Reporte Estadístico

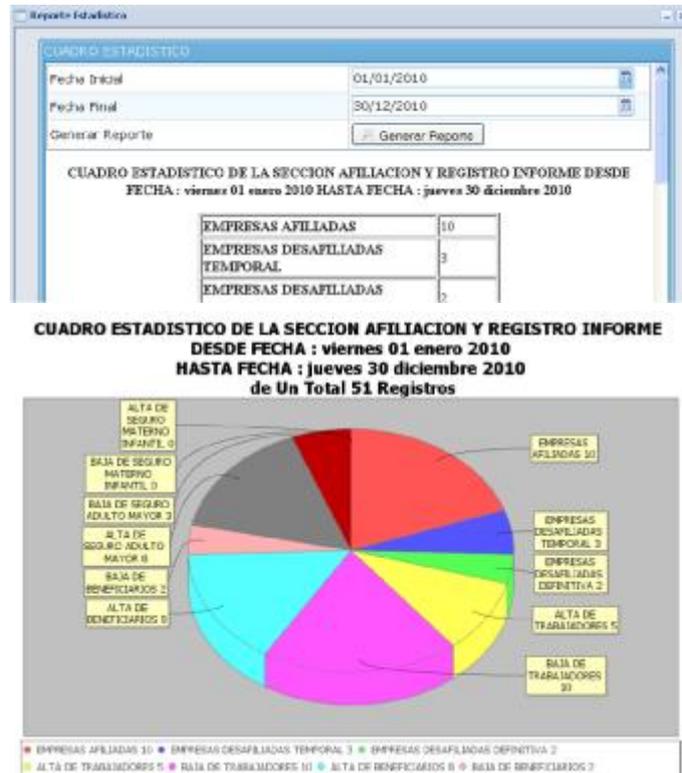


Figura 153. Pantalla Reporte Ficha Asegurado por Tipo de Seguro

2.1.2.2.3.8.4.75 Pantalla Administered Backup



Figura 154. Pantalla Administrar Backup

2.1.2.2.3.8.4.76 Pantalla Formulario (ABM) Backup



Figura 155. Pantalla Formulario (ABM) Backup

2.1.2.2.3.8.4.77 Pantalla Reporte Backup

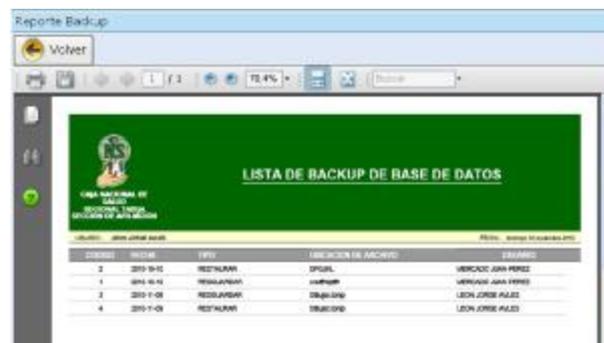


Figura 156. Pantalla Reporte Backup

2.1.2.2.3.9 Modelo de Análisis y Diseño

Este modelo establece la realización de los Casos de Uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

2.1.2.2.3.9.1 Modelado de Diagrama de Actividades

2.1.2.2.3.9.1.1 Introducción

El Diagrama de Actividades es un artefacto de la disciplina Análisis de Sistemas en la metodología RUP la cual estamos implementando.

Los Diagramas de Actividades se utilizan para modelar aspectos dinámicos de un Sistema, esto implica modelar los pasos secuenciales de un proceso.

2.1.2.2.3.9.1.2 Propósito

Comprender la estructura y la dinámica del Sistema deseado.

Identificar posibles mejoras en el Sistema.

2.1.2.2.3.9.1.3 Alcance

Describir los procesos del Sistema y los clientes.

Identificar y definir los procesos de los casos de uso según los objetivos de la organización.

Definir un Diagrama de actividad para cada Caso de Uso del sistema.

2.1.2.2.3.9.1.4 Diagrama de Actividades

2.1.2.2.3.9.1.4.1 Diagrama de Actividad: Caso de Uso Ingresar al Sistema

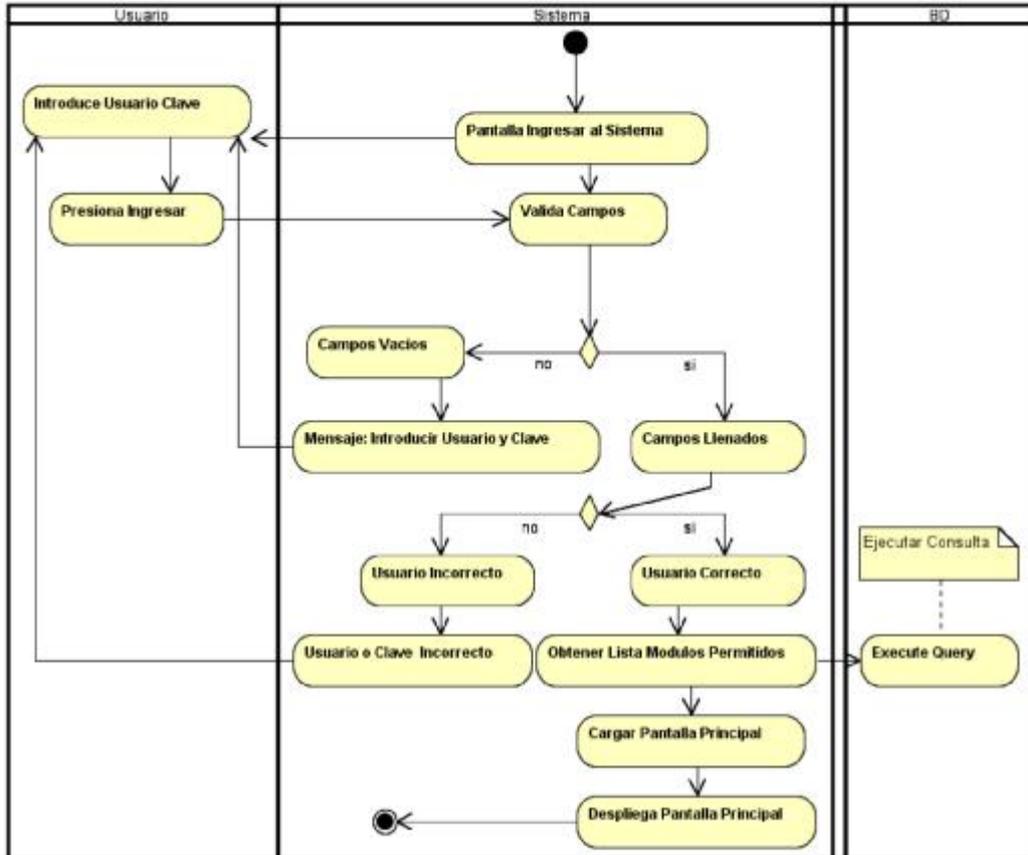


Figura 157. Diagrama de Actividad: CU Ingresar al Sistema

2.1.2.2.3.9.1.4.2 Diagrama de Actividad: Caso de Uso Administrar Actividad Económica

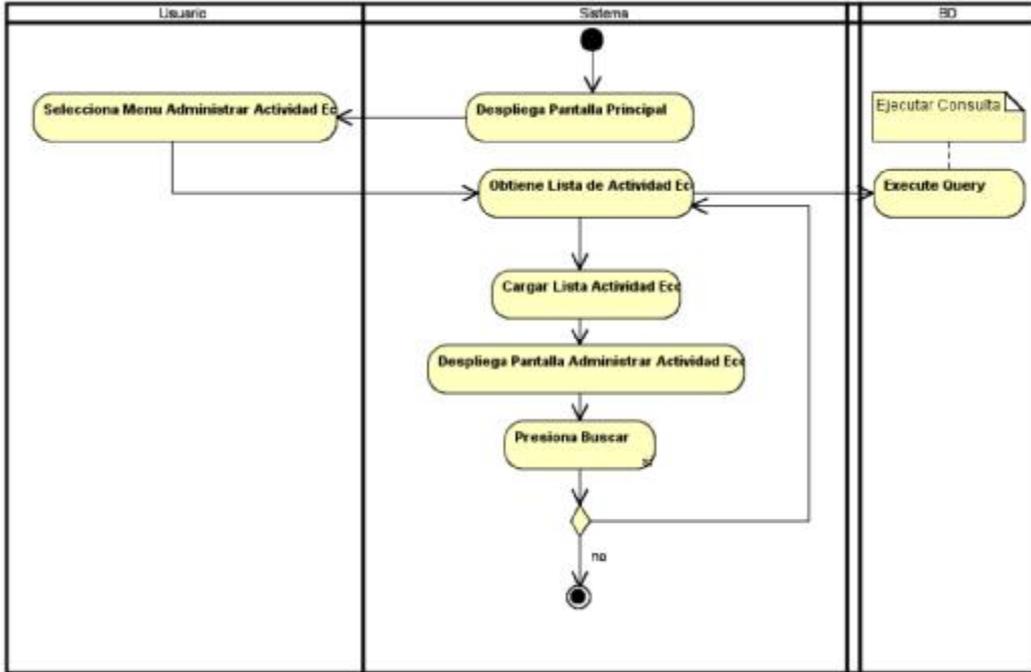


Figura 158. Diagrama de Actividad: CU Administrar Actividad Económica

2.1.2.2.3.9.1.4.3 Diagrama de Actividad: Caso de Uso Adicionar Actividad Económica

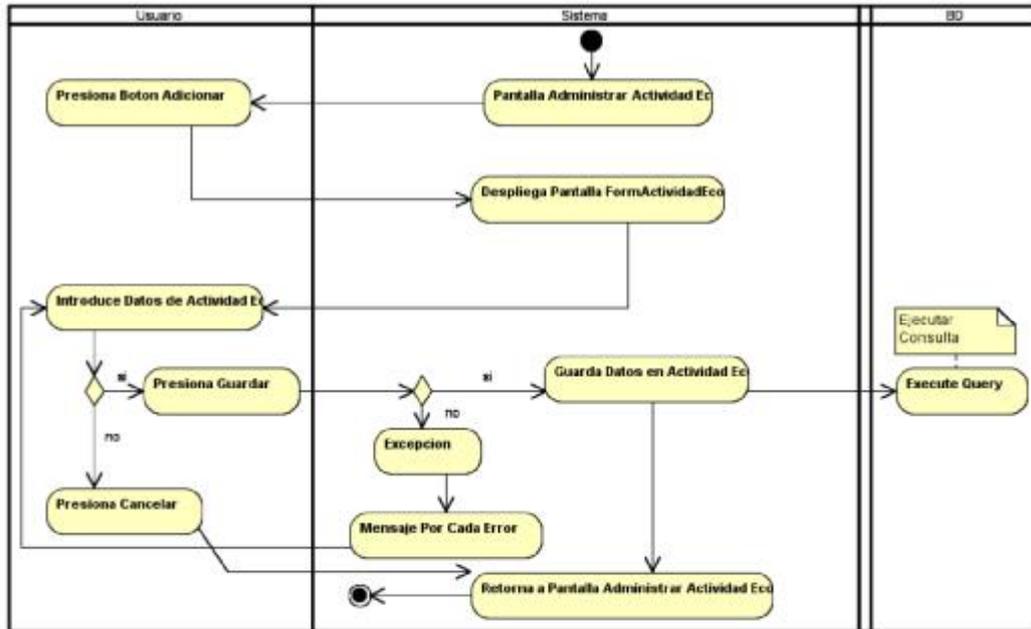


Figura 159. Diagrama de Actividad: CU Adicionar Actividad Económica

2.1.2.2.3.9.1.4.4 Diagrama de Actividad: Caso de Uso Modificar Actividad Económica

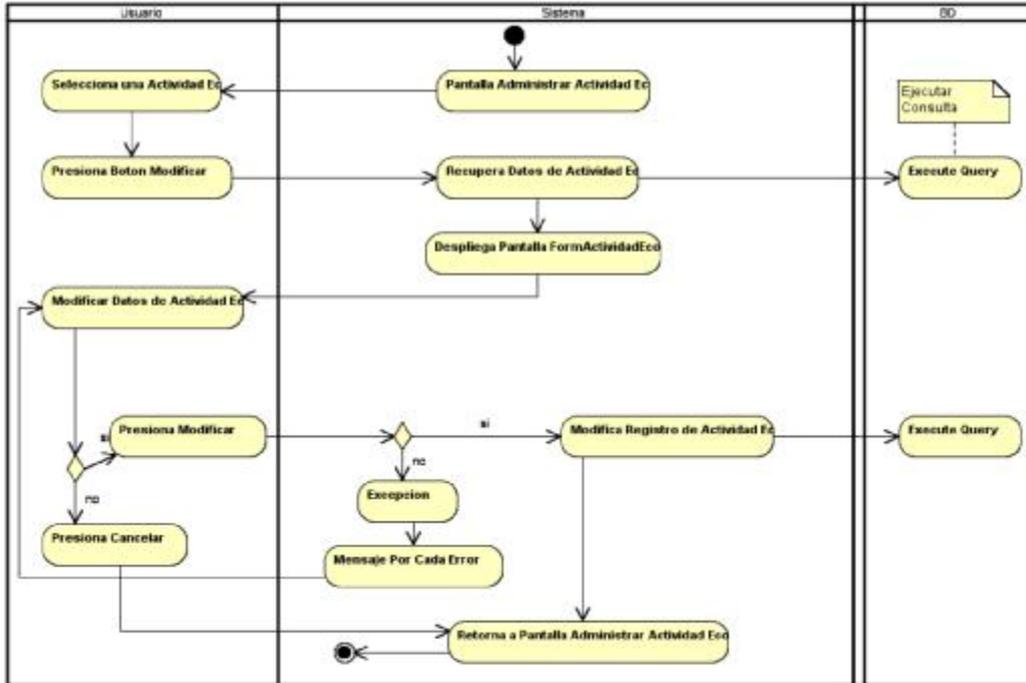


Figura 160. Diagrama de Actividad: CU Modificar Actividad Económica

2.1.2.2.3.9.1.4.5 Diagrama de Actividad: Caso de Uso Eliminar Actividad Económica

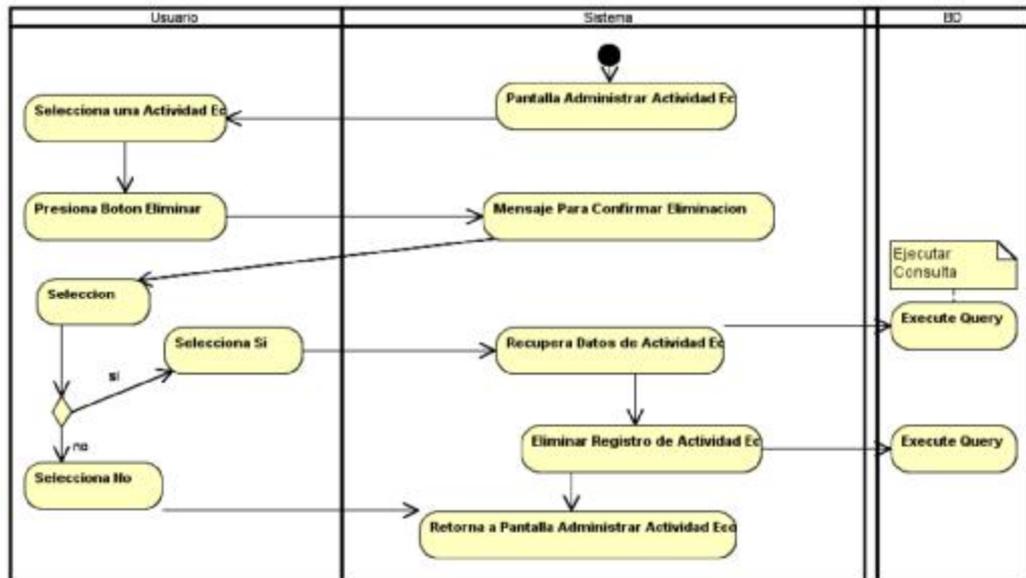


Figura 161. Diagrama de Actividad: CU Eliminar Actividad Económica

2.1.2.2.3.9.1.4.6 Diagrama de Actividad: Caso de Uso Cambiar Estado Actividad

Económica

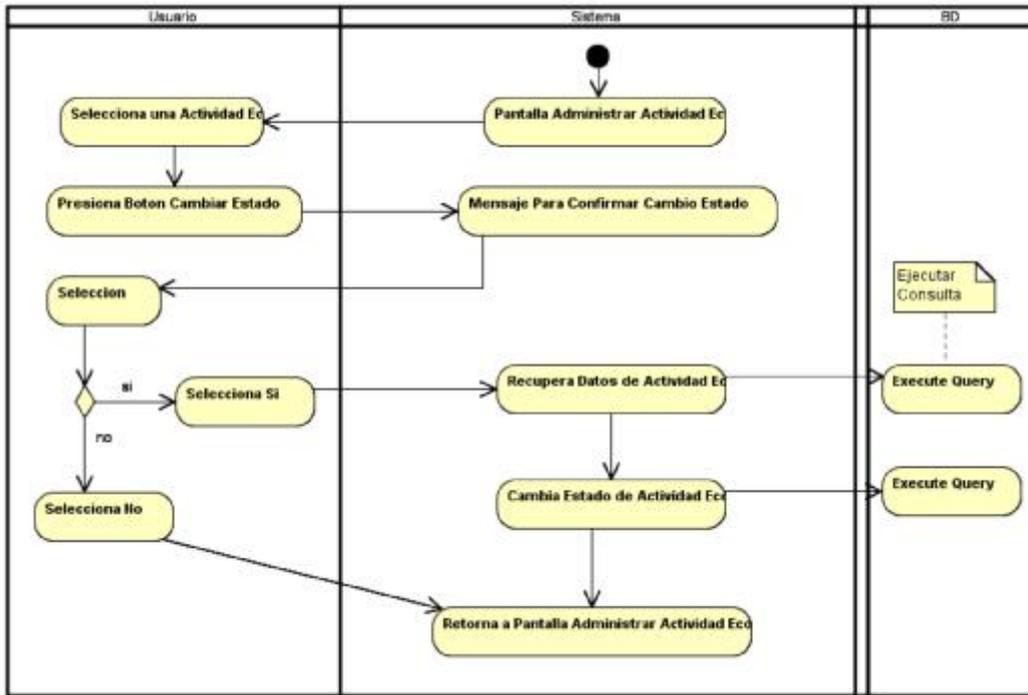


Figura 162. Diagrama de Actividad: CU Cambiar Estado Actividad Económica

2.1.2.2.3.9.1.4.7 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Actividad Económica

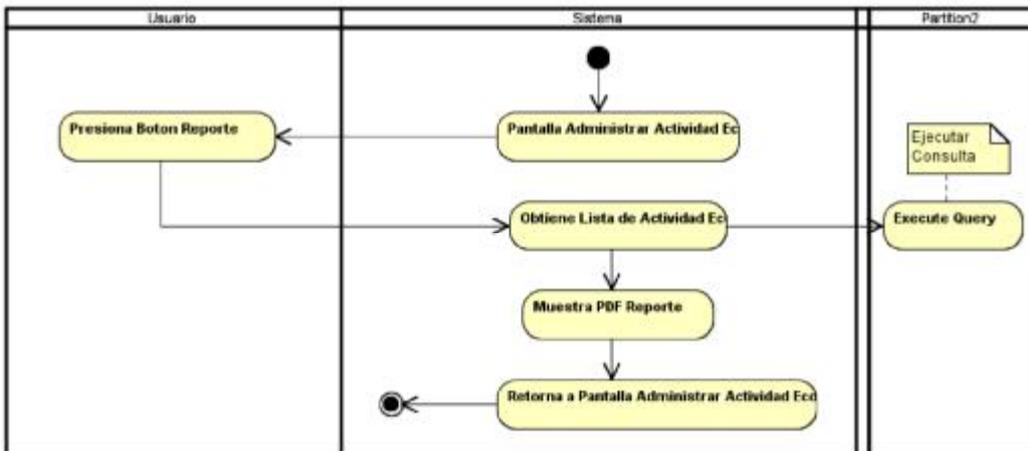


Figura 163. Diagrama de Actividad: CU Generar Reporte Lista Actividad Económica

2.1.2.2.3.9.1.4.8 Diagrama de Actividad: Caso de Uso Administrar Centro Salud

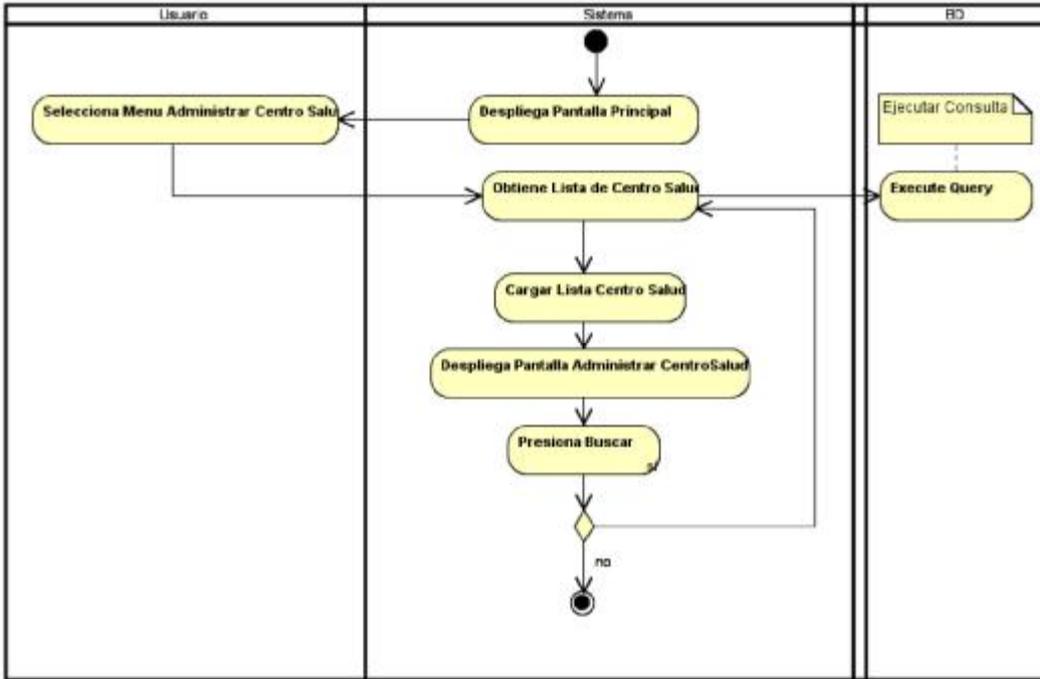


Figura 164. Diagrama de Actividad: CU Administrar Centro Salud

2.1.2.2.3.9.1.4.9 Diagrama de Actividad: Caso de Uso Adicionar Centro Salud

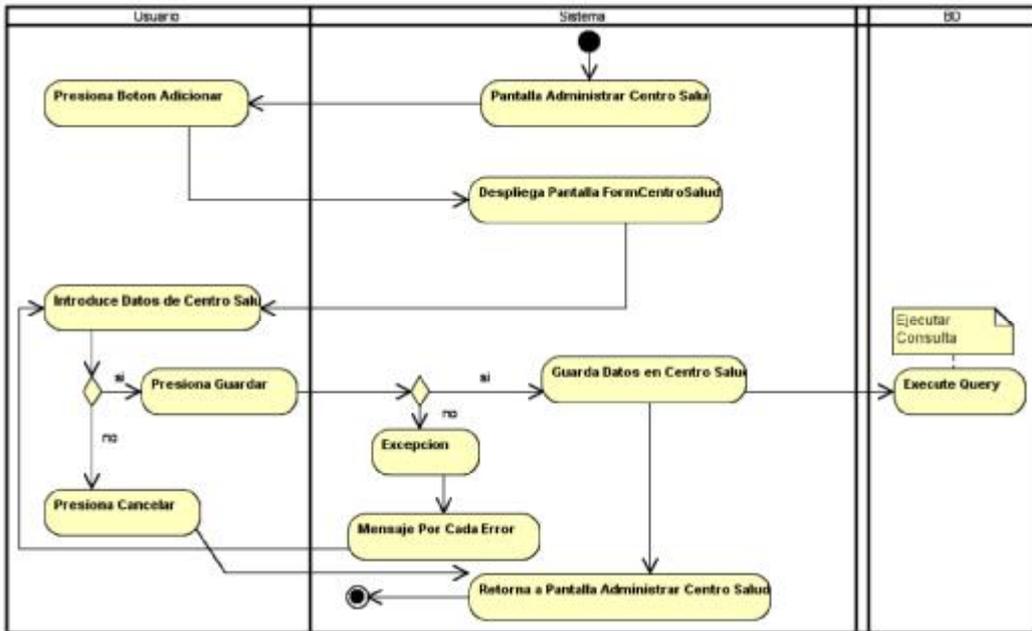


Figura 165. Diagrama de Actividad: CU Adicionar Centro Salud

2.1.2.2.3.9.1.4.10 Diagrama de Actividad: Caso de Uso Modificar Centro Salud

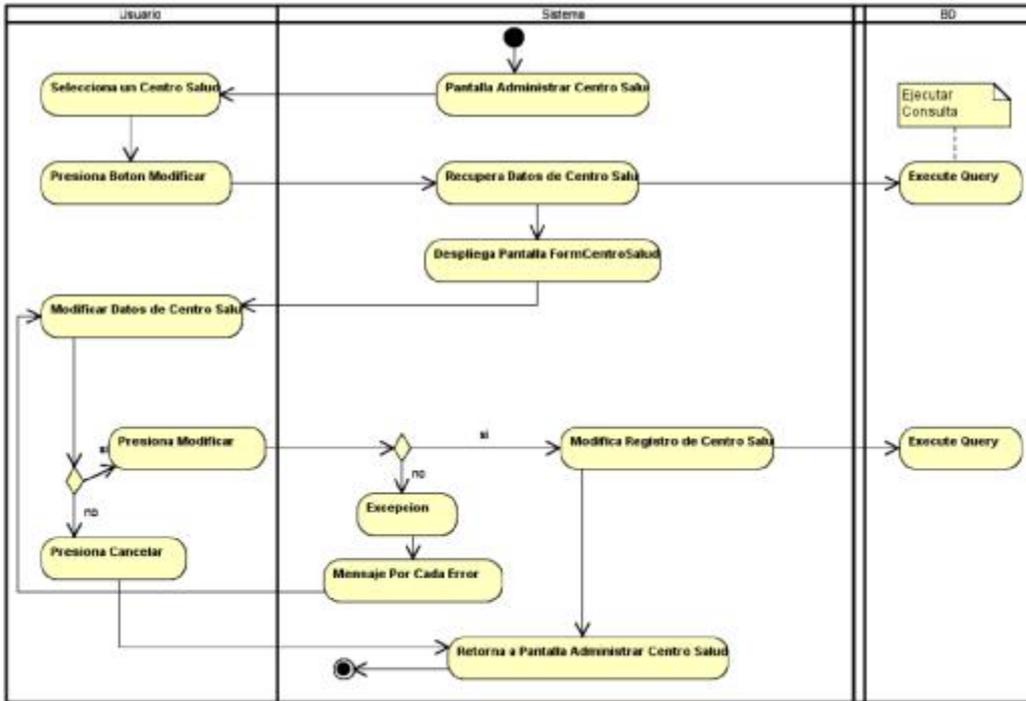


Figura 166. Diagrama de Actividad: CU Modificar Centro Salud

2.1.2.2.3.9.1.4.11 Diagrama de Actividad: Caso de Uso Eliminar Centro Salud

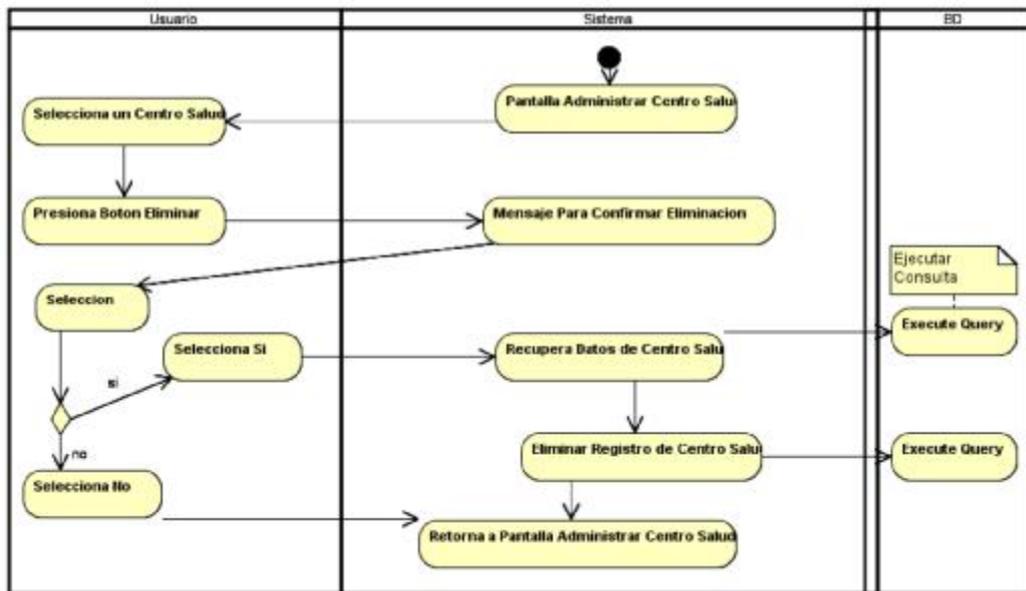


Figura 167. Diagrama de Actividad: CU Eliminar Centro Salud

2.1.2.2.3.9.1.4.12 Diagrama de Actividad: Caso de Uso Cambiar Estado Centro Salud

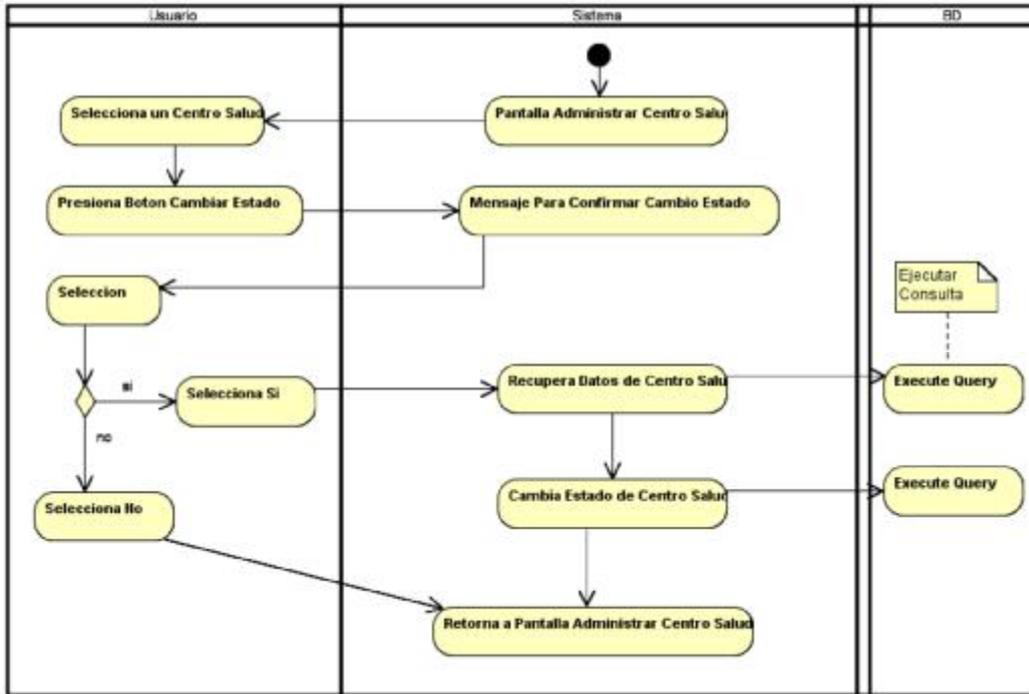


Figura 168. Diagrama de Actividad: CU Cambiar Estado Centro Salud

2.1.2.2.3.9.1.4.13 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Centro Salud

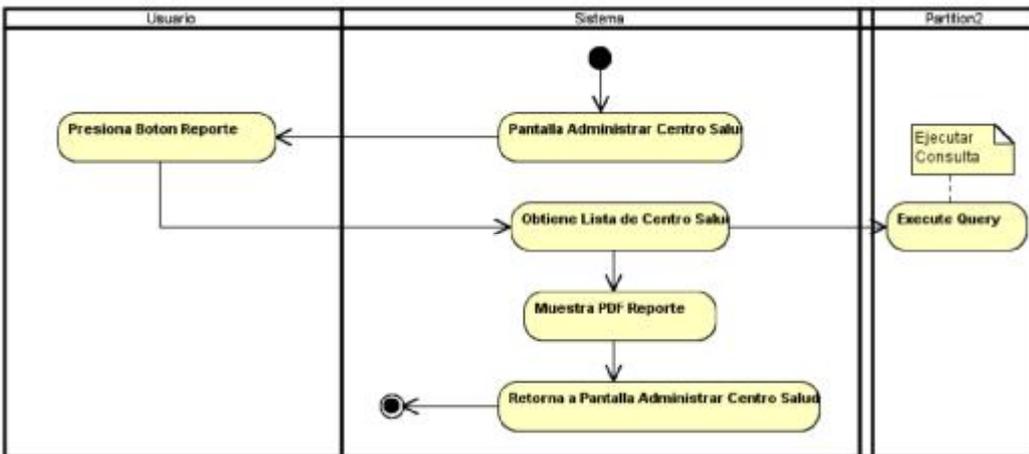


Figura 169. Diagrama de Actividad: CU Generar Reporte Lista Centro Salud

2.1.2.2.3.9.1.4.14 Diagrama de Actividad: Caso de Uso Administrar Departamento

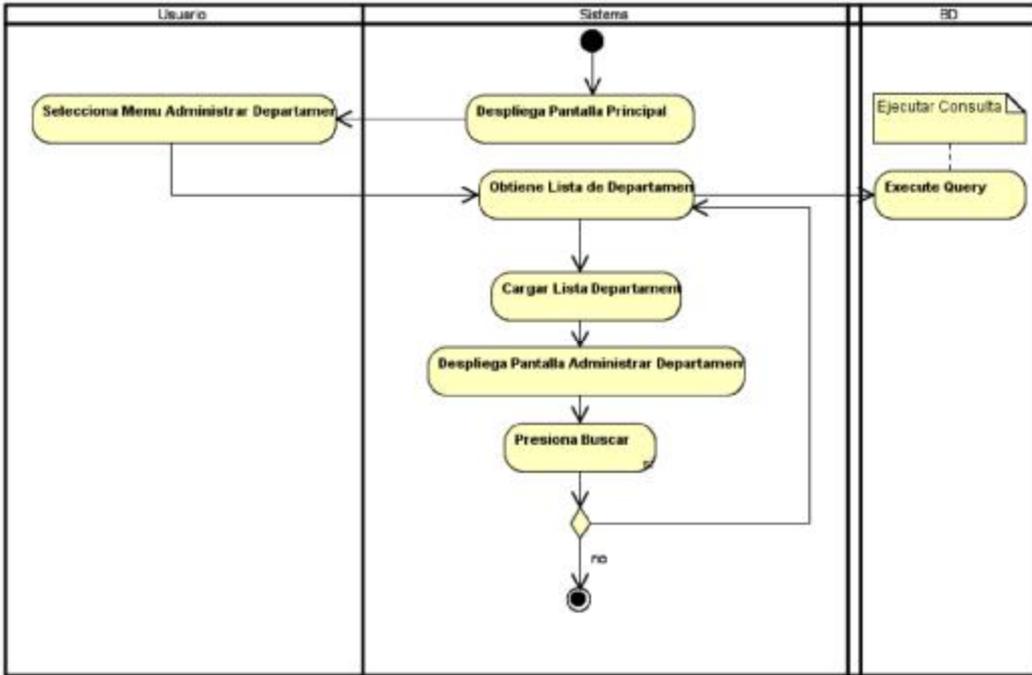


Figura 170. Diagrama de Actividad: CU Administrar Departamento

2.1.2.2.3.9.1.4.15 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Departamento

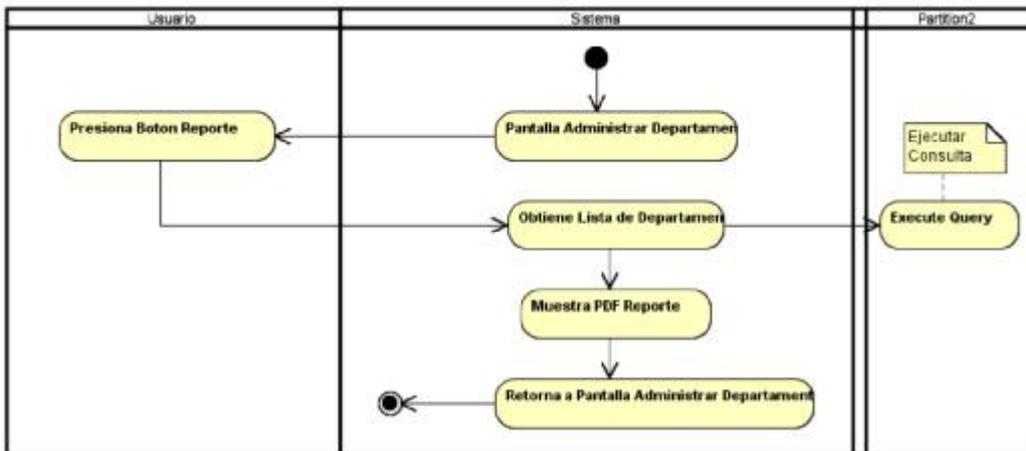


Figura 171. Diagrama de Actividad: CU Generar Reporte Lista Departamento

2.1.2.2.3.9.1.4.16 Diagrama de Actividad: Caso de Uso Administrar Funcionalidad

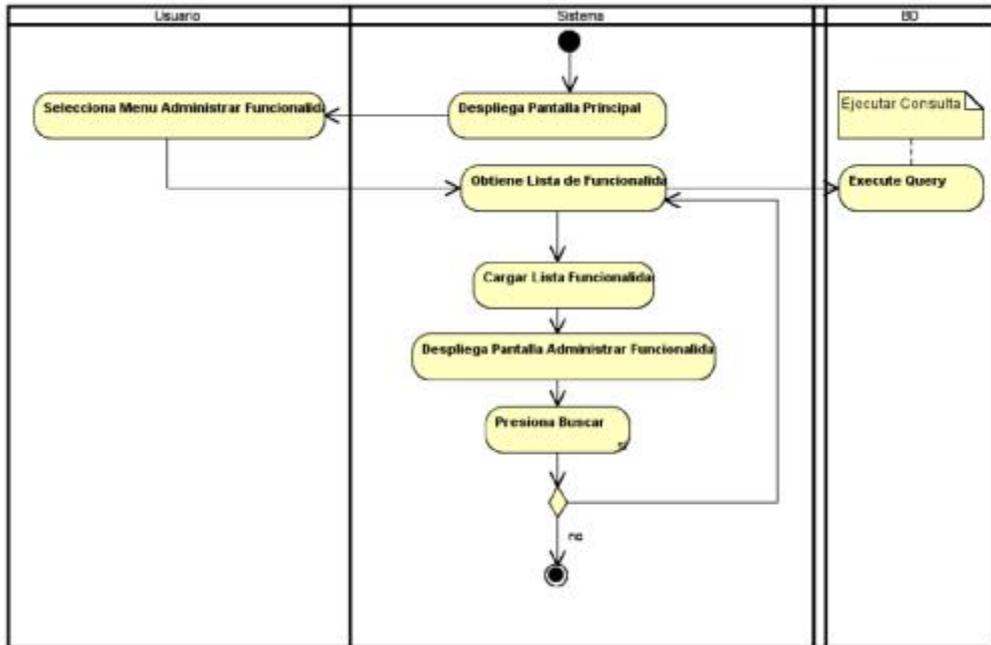


Figura 172. Diagrama de Actividad: CU Administrar Funcionalidad

2.1.2.2.3.9.1.4.17 Diagrama de Actividad: Caso de Uso Cambiar Estado Funcionalidad

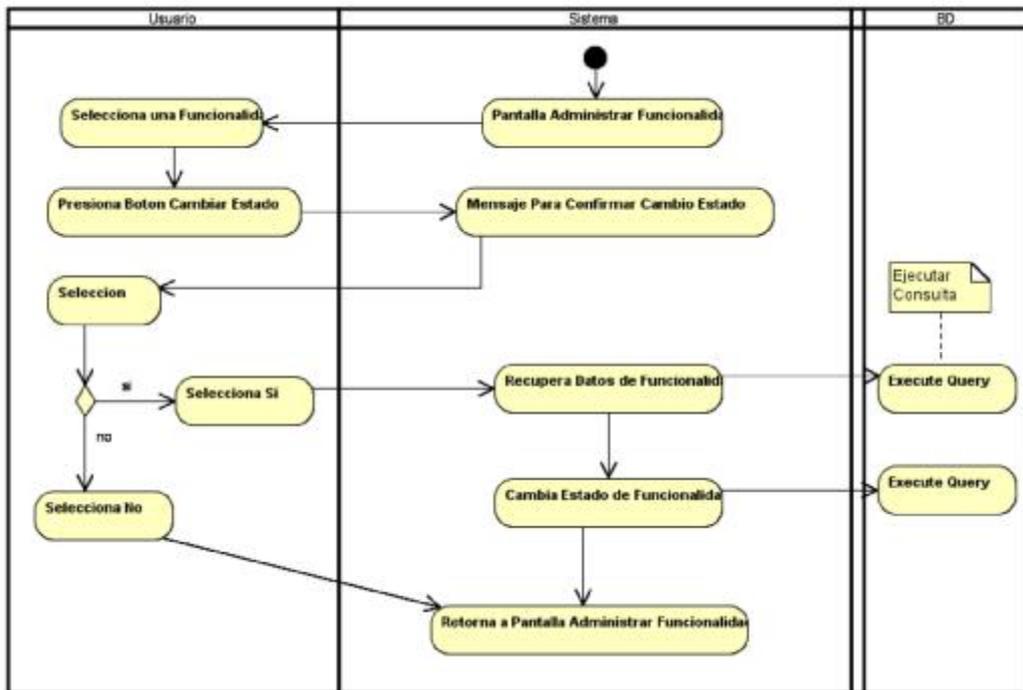


Figura 173. Diagrama de Actividad: CU Cambiar Estado Funcionalidad

2.1.2.2.3.9.1.4.18 Diagrama de Actividad: Caso de Uso Generar Reporte Lista

Funcionalidad

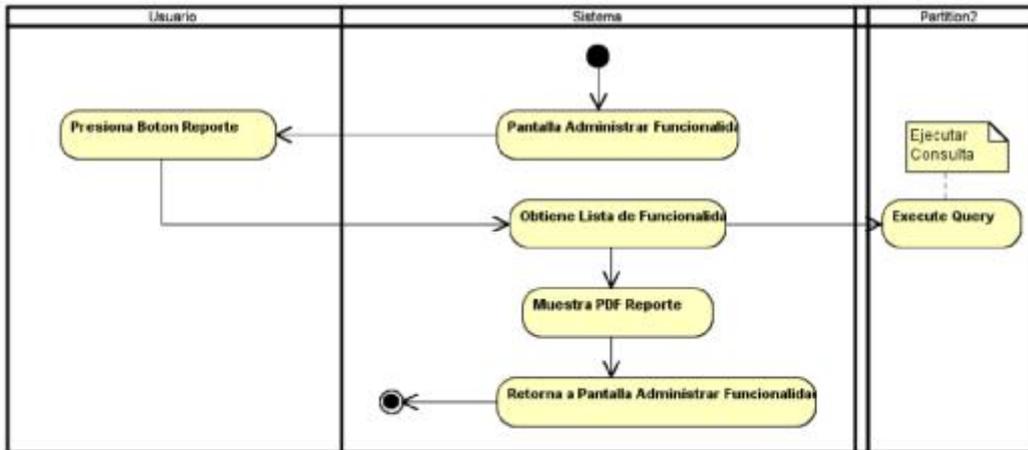


Figura 174. Diagrama de Actividad: CU Generar Reporte Lista Funcionalidad
 2.1.2.2.3.9.1.4.19 Diagrama de Actividad: Caso de Uso Administrar Módulo

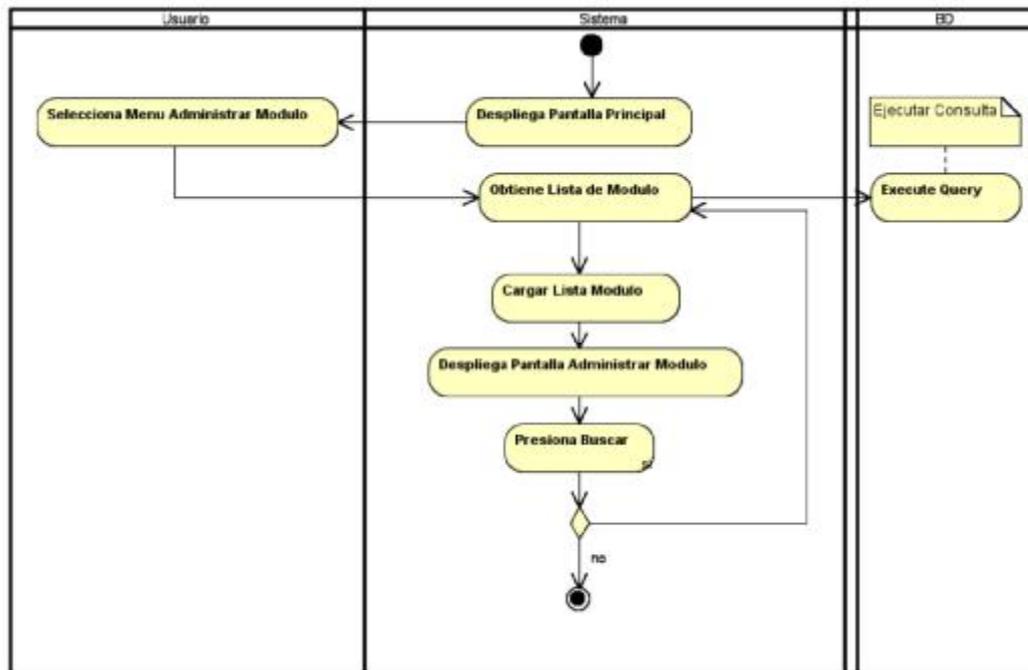


Figura 175. Diagrama de Actividad: CU Administrar Módulo

2.1.2.2.3.9.1.4.20 Diagrama de Actividad: Caso de Uso Cambiar Estado Módulo

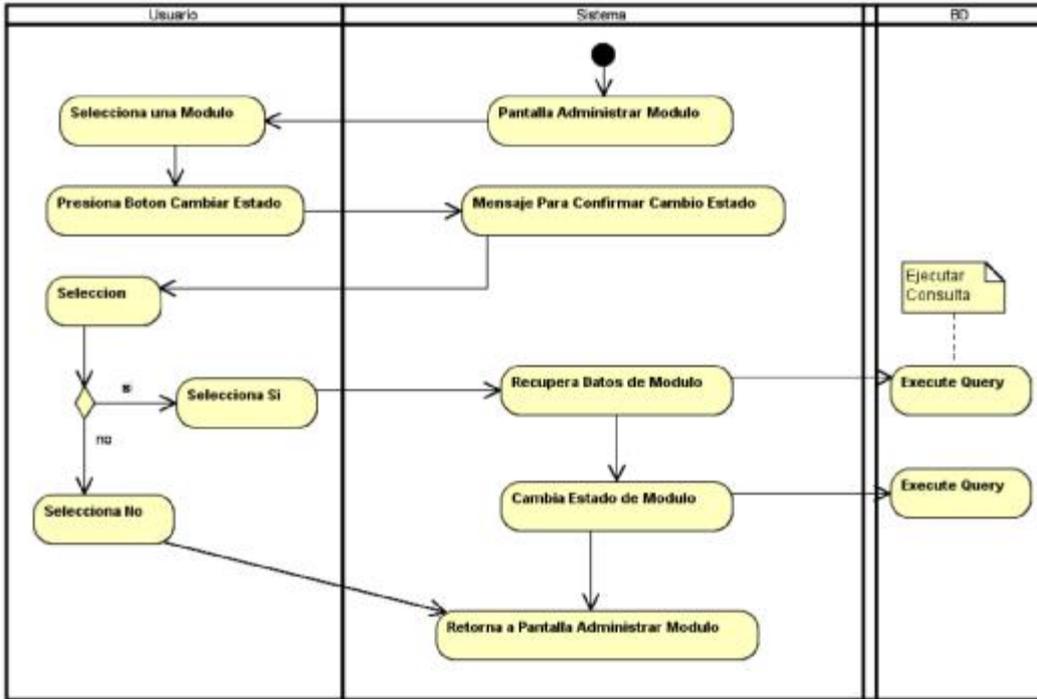


Figura 176. Diagrama de Actividad: CU Cambiar Estado Módulo

2.1.2.2.3.9.1.4.21 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Módulo

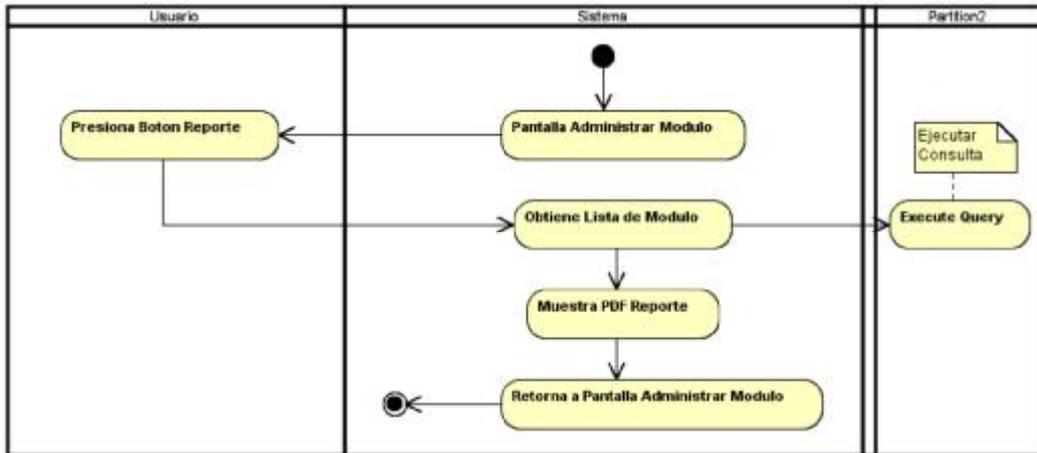


Figura 177. Diagrama de Actividad: CU Generar Reporte Lista Módulo

2.1.2.2.3.9.1.4.22 Diagrama de Actividad: Caso de Uso Administrar Municipio Localidad

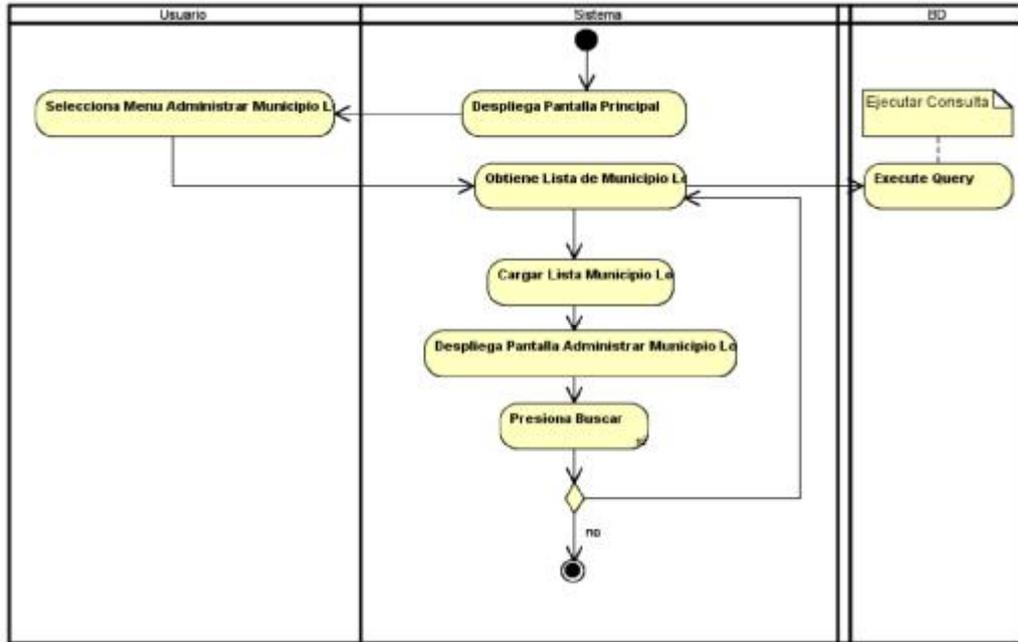


Figura 178. Diagrama de Actividad: CU Administrar Municipio Localidad

2.1.2.2.3.9.1.4.23 Diagrama de Actividad: Caso de Uso Adicionar Municipio Localidad

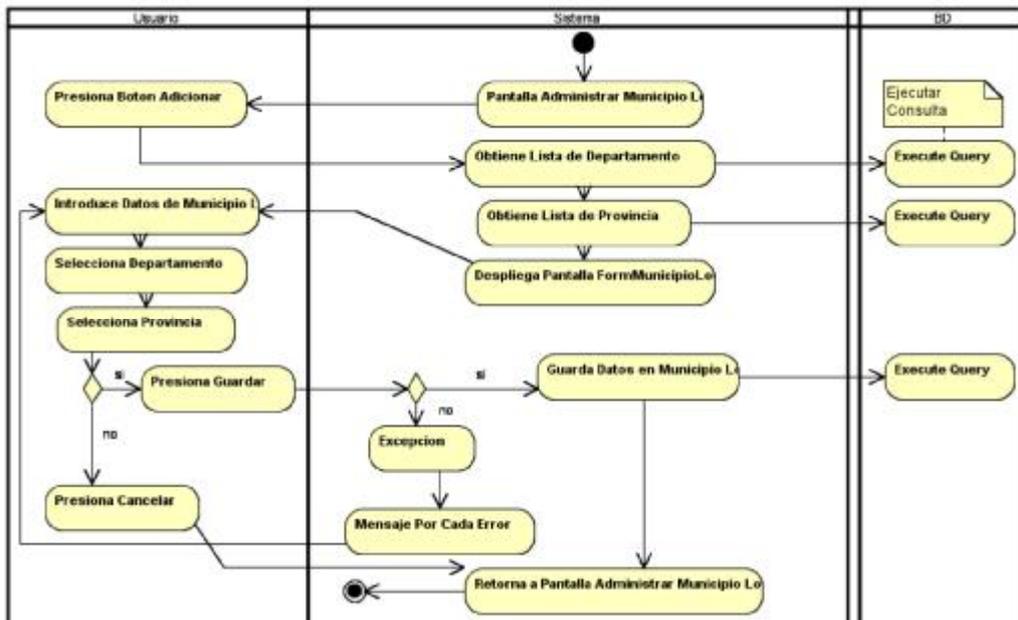


Figura 179. Diagrama de Actividad: CU Adicionar Municipio Localidad

2.1.2.2.3.9.1.4.24 Diagrama de Actividad: Caso de Uso Modificar Municipio Localidad

Localidad

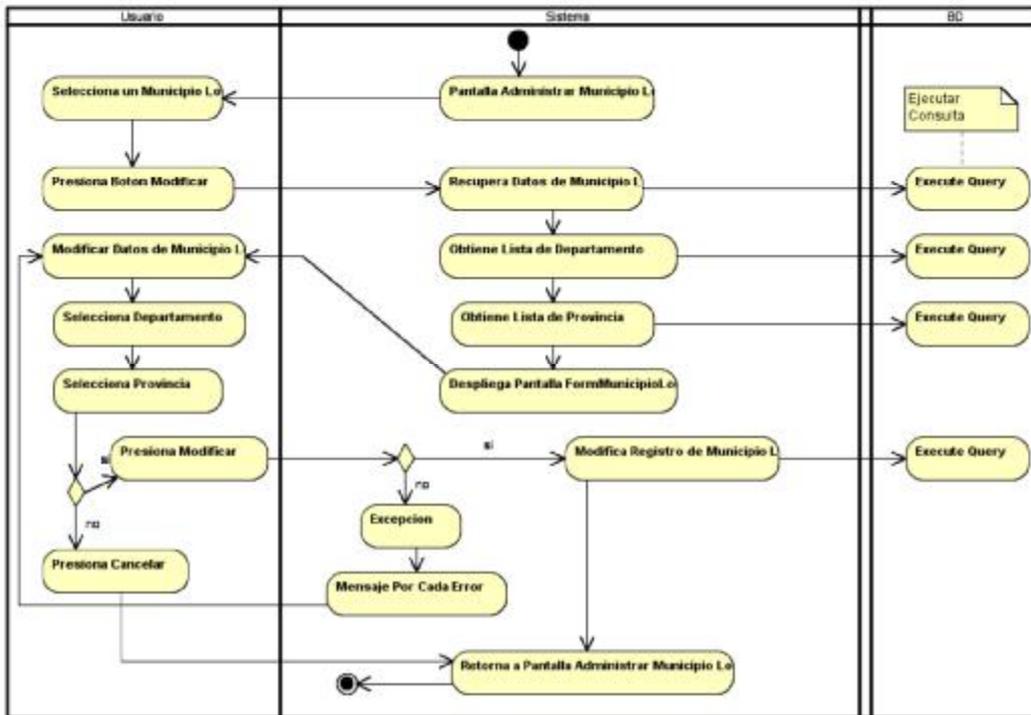


Figura 180. Diagrama de Actividad: CU Modificar Municipio Localidad

2.1.2.2.3.9.1.4.25 Diagrama de Actividad: Caso de Uso Eliminar Municipio

Localidad

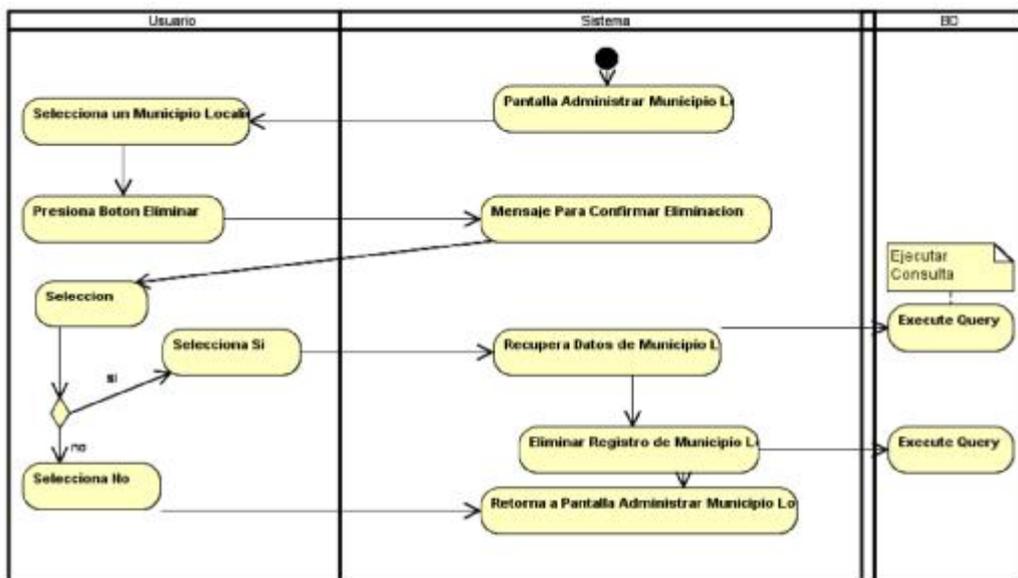


Figura 181. Diagrama de Actividad: CU Eliminar Municipio Localidad

2.1.2.2.3.9.1.4.26 Diagrama de Actividad: Caso de Uso Cambiar Estado Municipio Localidad

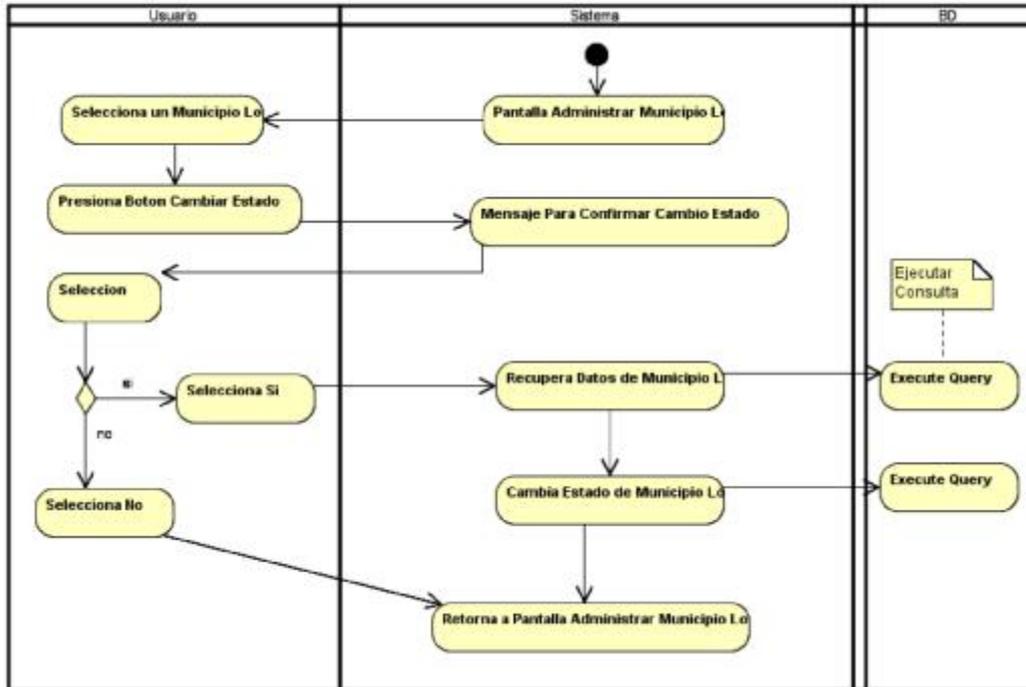


Figura 182. Diagrama de Actividad: CU Cambiar Estado Municipio Localidad

2.1.2.2.3.9.1.4.27 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Municipio Localidad

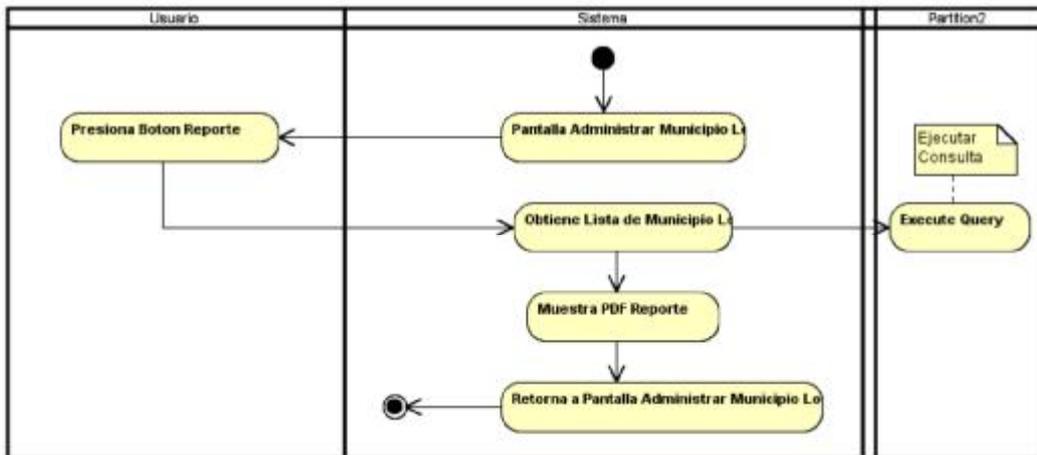


Figura 183. Diagrama de Actividad: CU Generar Reporte Lista Municipio Localidad

2.1.2.2.3.9.1.4.28 Diagrama de Actividad: Caso de Uso Administrar Ocupación

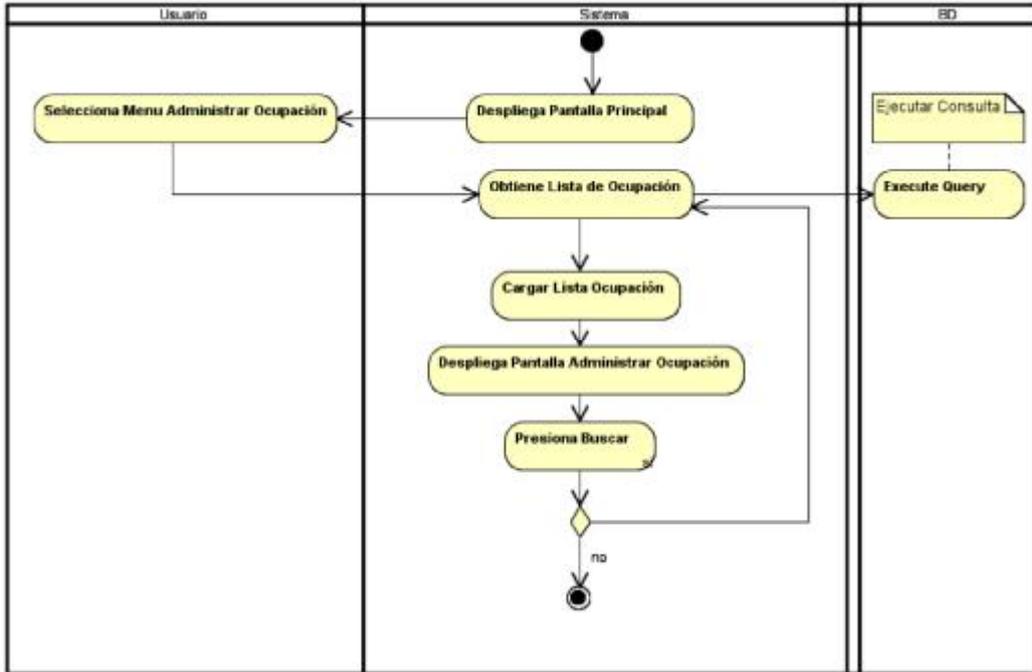


Figura 184. Diagrama de Actividad: CU Administrar Ocupación

2.1.2.2.3.9.1.4.29 Diagrama de Actividad: Caso de Uso Adicionar Ocupación

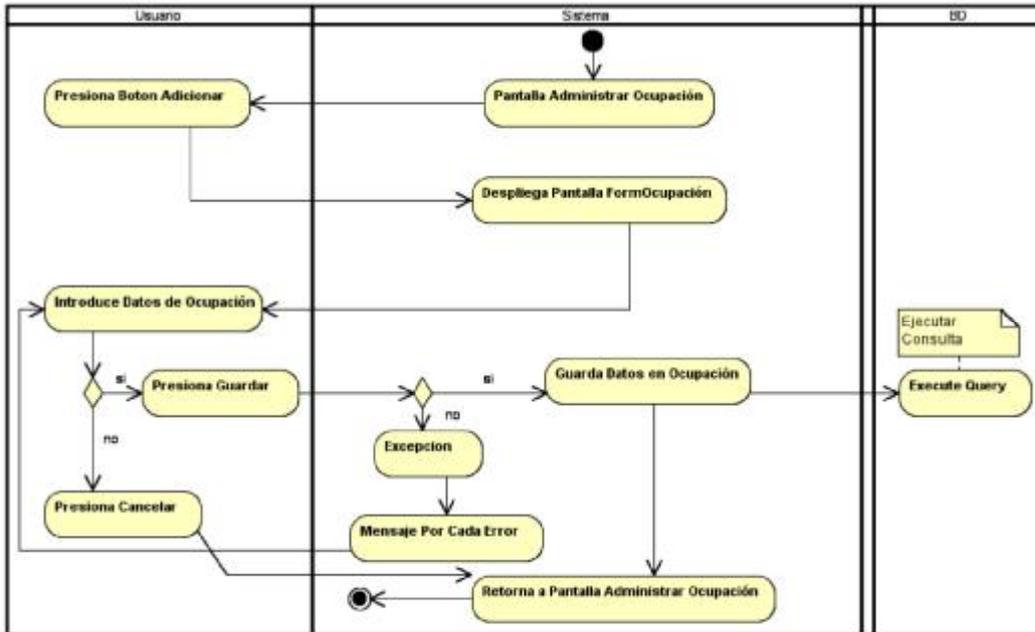


Figura 185. Diagrama de Actividad: CU Adicionar Ocupación

2.1.2.2.3.9.1.4.30 Diagrama de Actividad: Caso de Uso Modificar Ocupación

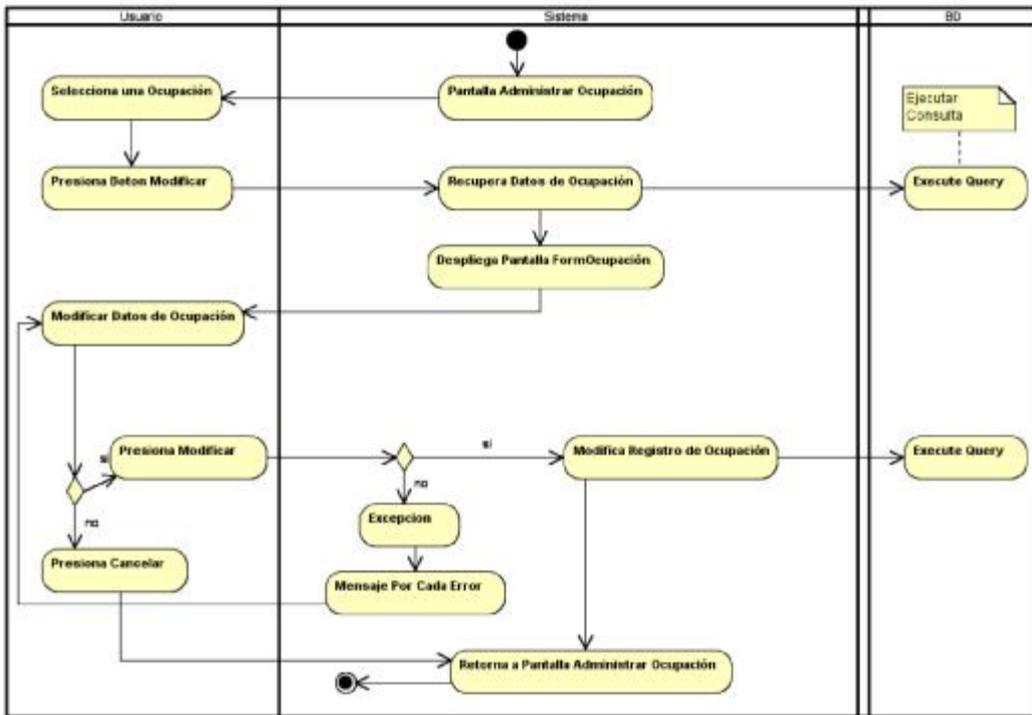


Figura 186. Diagrama de Actividad: CU Modificar Ocupación

2.1.2.2.3.9.1.4.31 Diagrama de Actividad: Caso de Uso Eliminar Ocupación

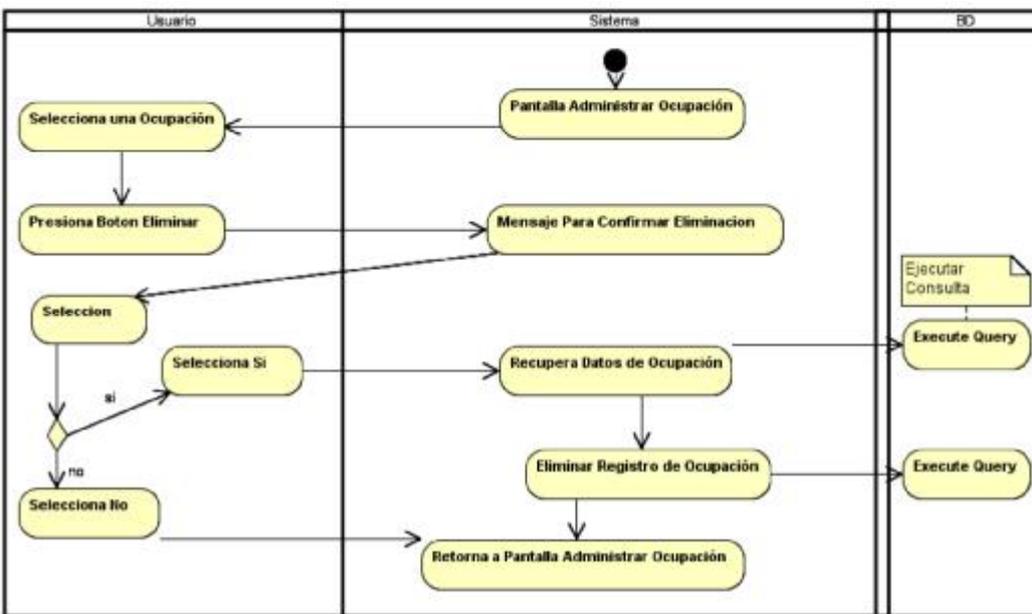


Figura 187. Diagrama de Actividad: CU Eliminar Ocupación

2.1.2.2.3.9.1.4.32 Diagrama de Actividad: Caso de Uso Cambiar Estado Ocupación

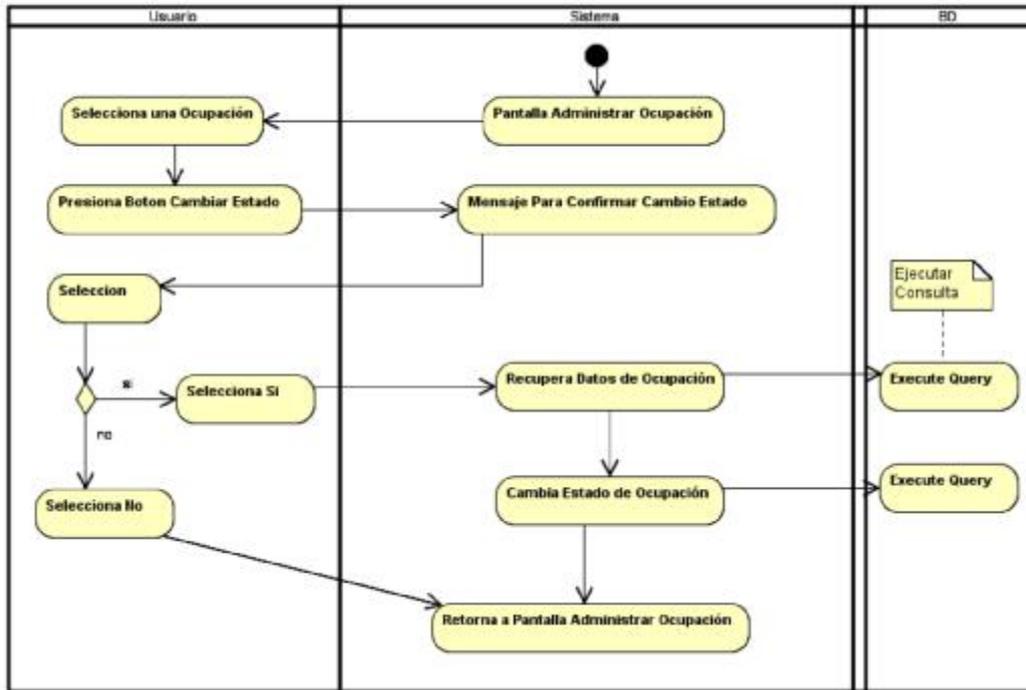


Figura 188. Diagrama de Actividad: CU Cambiar Estado Ocupación

2.1.2.2.3.9.1.4.33 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Ocupación

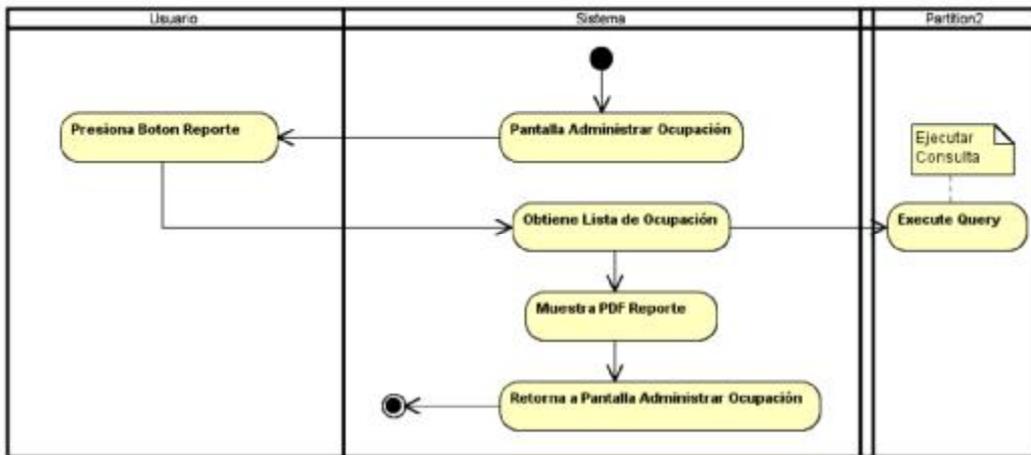


Figura 189. Diagrama de Actividad: CU Generar Reporte Lista Ocupación

2.1.2.2.3.9.1.4.34 Diagrama de Actividad: Caso de Uso Administrar Parentesco

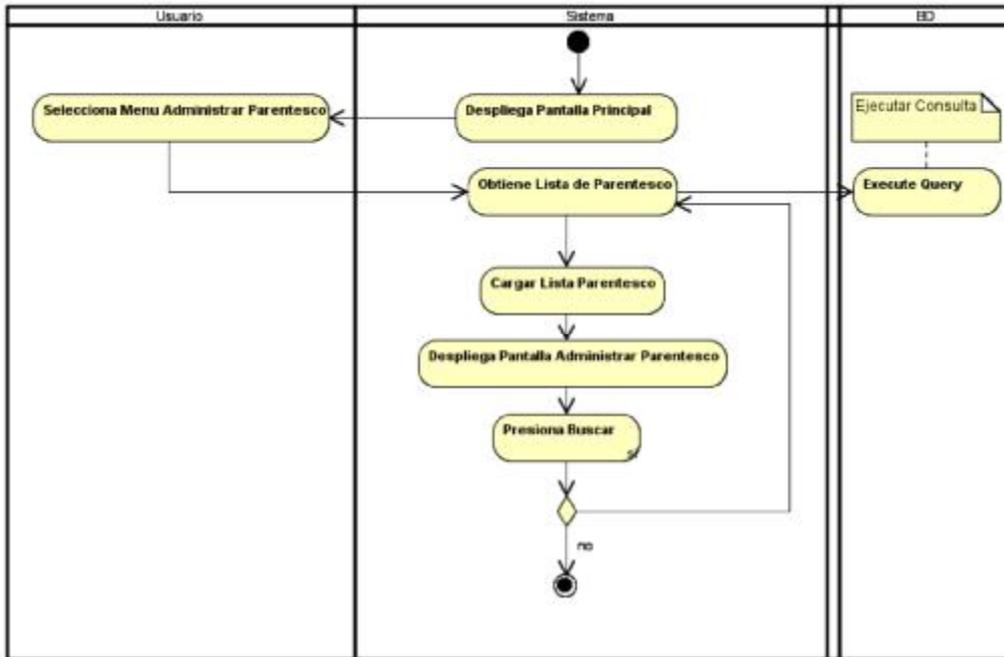


Figura 190. Diagrama de Actividad: CU Administrar Parentesco

2.1.2.2.3.9.1.4.35 Diagrama de Actividad: Caso de Uso Adicionar Parentesco

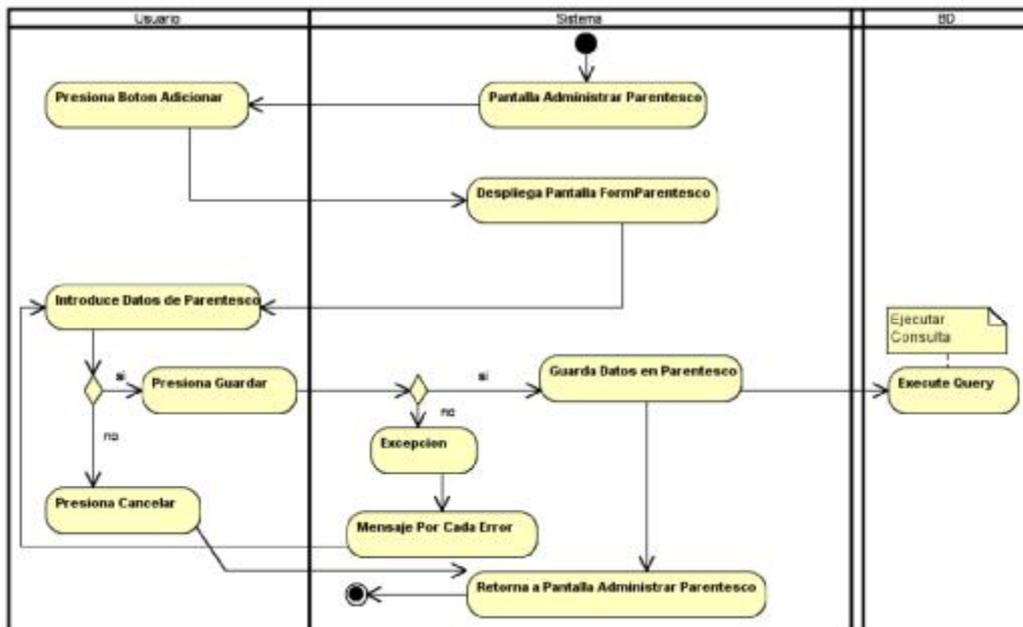


Figura 191. Diagrama de Actividad: CU Adicionar Parentesco

2.1.2.2.3.9.1.4.36 Diagrama de Actividad: Caso de Uso Modificar Parentesco

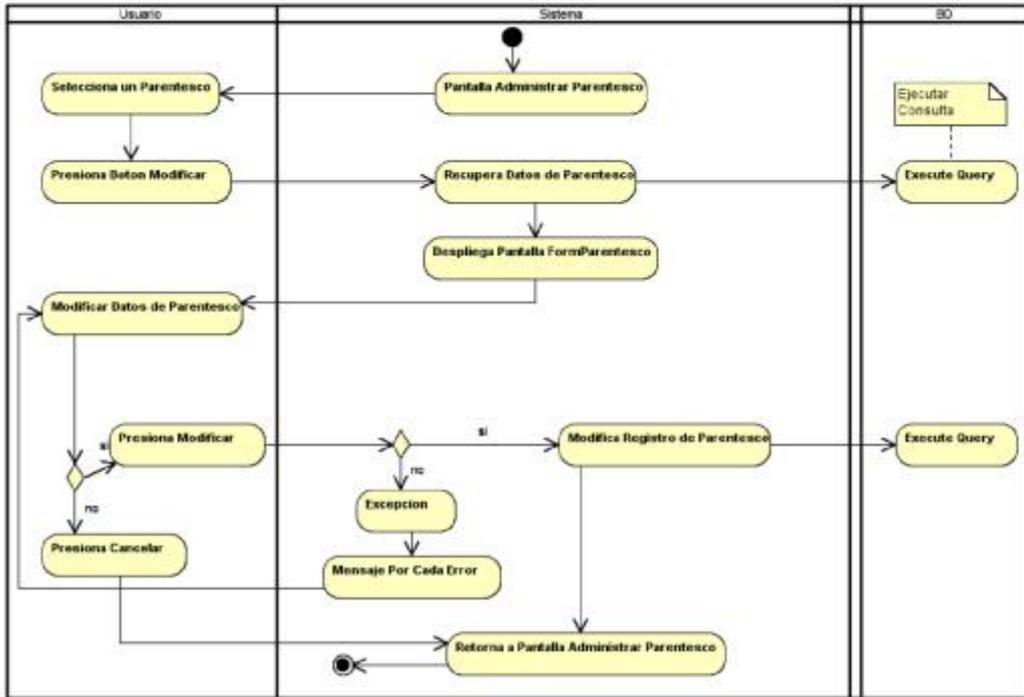


Figura 192. Diagrama de Actividad: CU Modificar Parentesco

2.1.2.2.3.9.1.4.37 Diagrama de Actividad: Caso de Uso Eliminar Parentesco

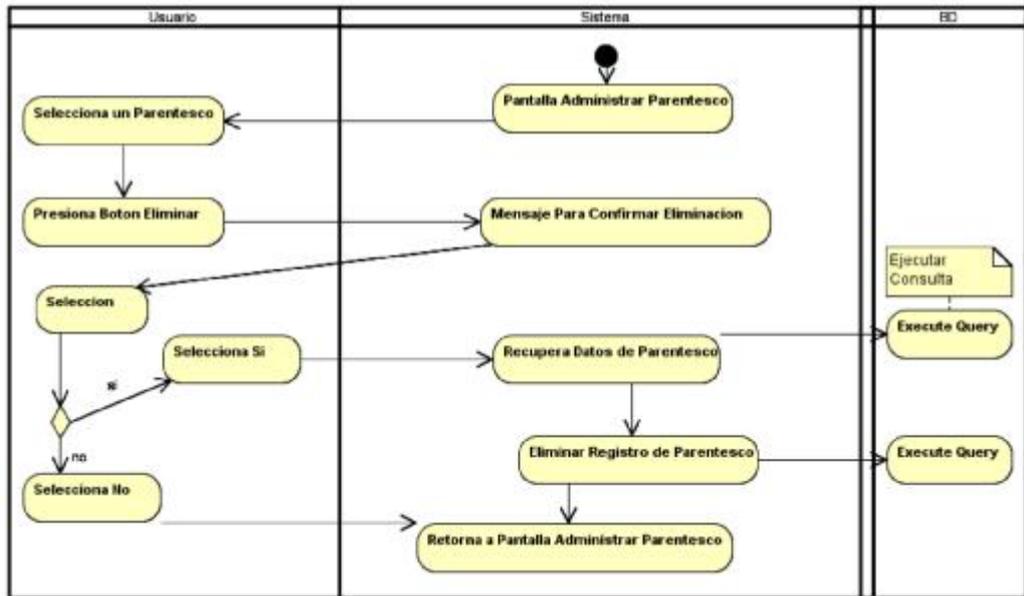


Figura 193. Diagrama de Actividad: CU Eliminar Parentesco

2.1.2.2.3.9.1.4.38 Diagrama de Actividad: Caso de Uso Cambiar Estado Parentesco

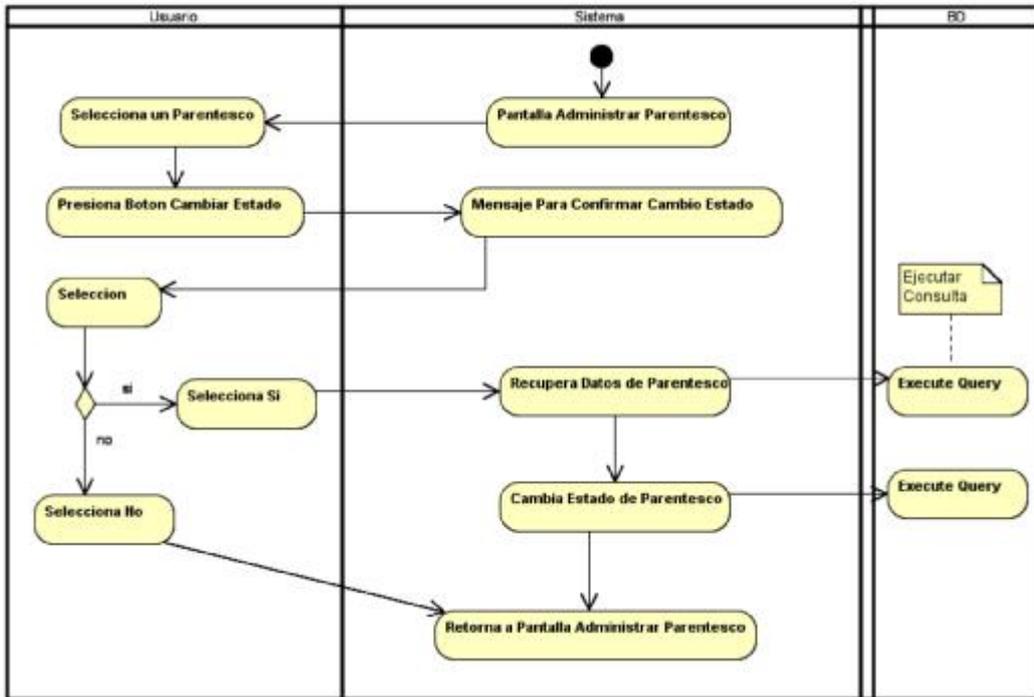


Figura 194. Diagrama de Actividad: CU Cambiar Estado Parentesco

2.1.2.2.3.9.1.4.39 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Parentesco

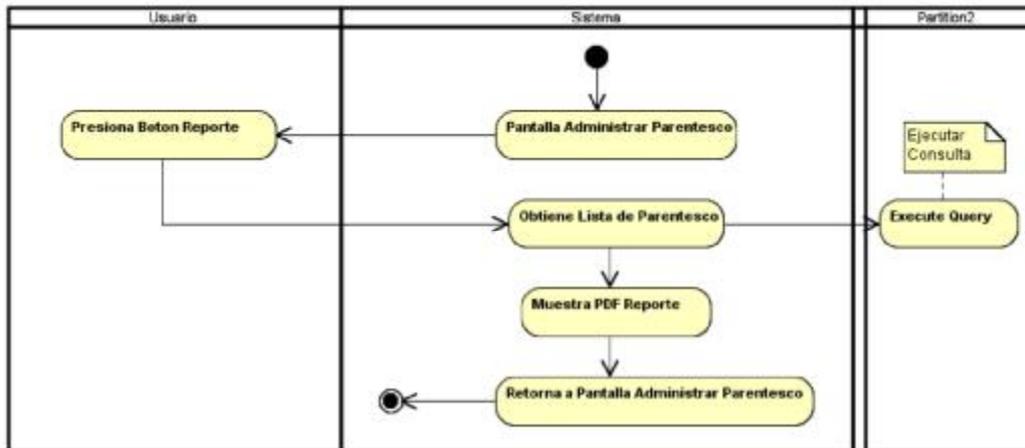


Figura 195. Diagrama de Actividad: CU Generar Reporte Lista Parentesco

2.1.2.2.3.9.1.4.40 Diagrama de Actividad: Caso de Uso Administrar Provincia

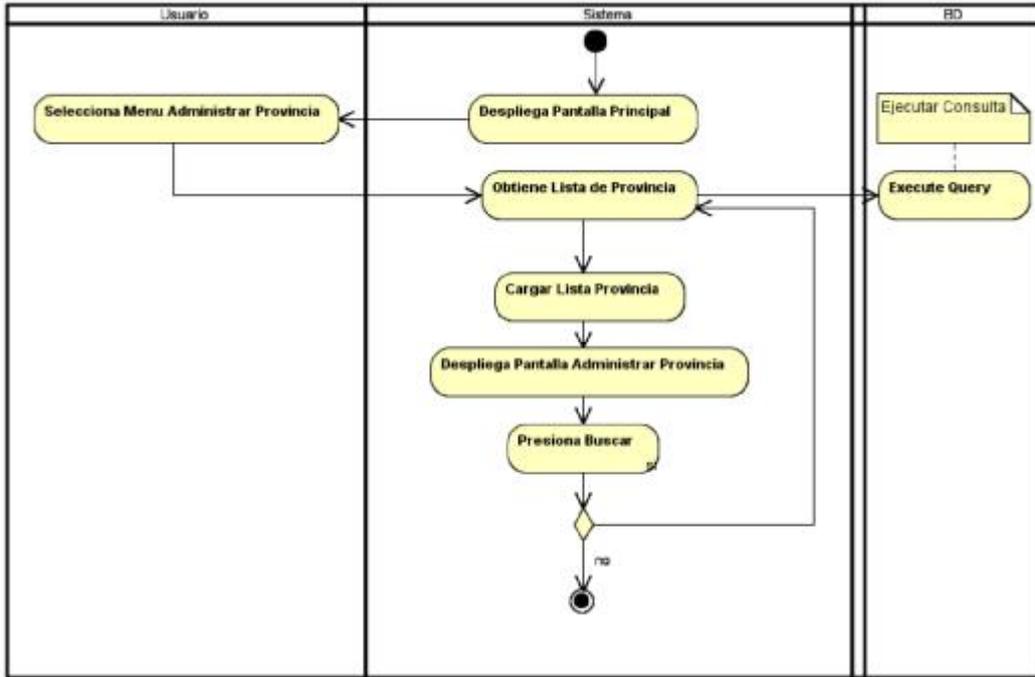


Figura 196. Diagrama de Actividad: CU Administrar Provincia

2.1.2.2.3.9.1.4.41 Diagrama de Actividad: Caso de Uso Adicionar Provincia

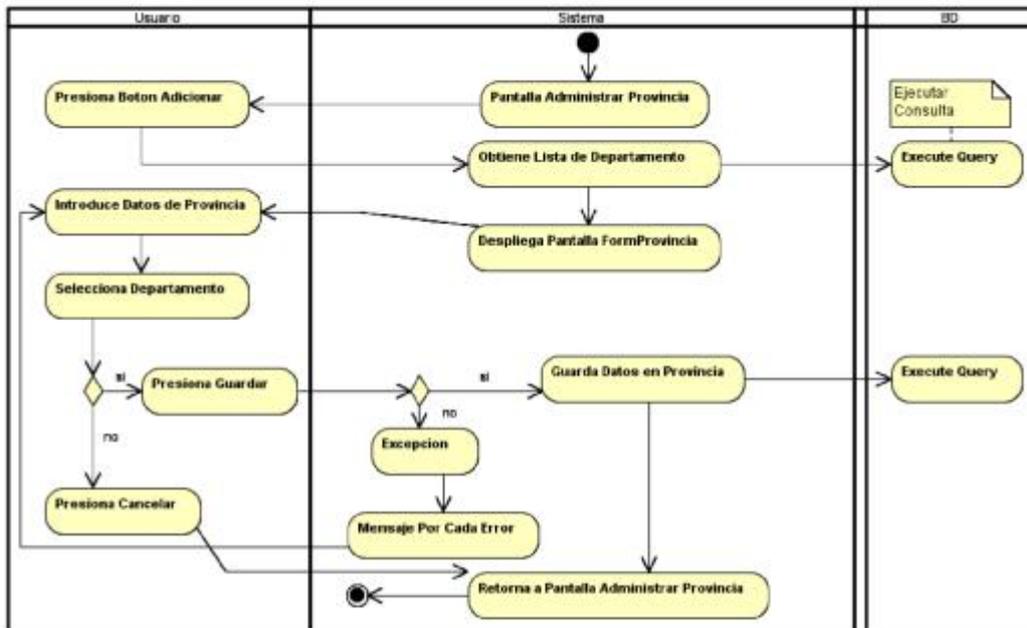


Figura 197. Diagrama de Actividad: CU Adicionar Provincia

2.1.2.2.3.9.1.4.42 Diagrama de Actividad: Caso de Uso Modificar Provincia

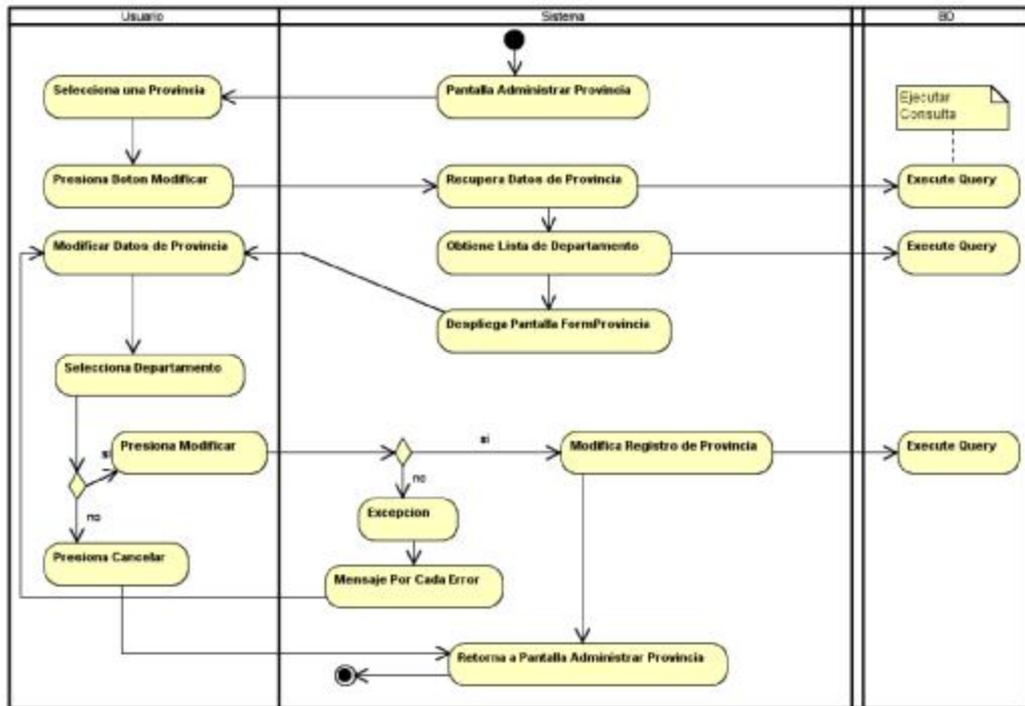


Figura 198. Diagrama de Actividad: CU Modificar Provincia

2.1.2.2.3.9.1.4.43 Diagrama de Actividad: Caso de Uso Eliminar Provincia

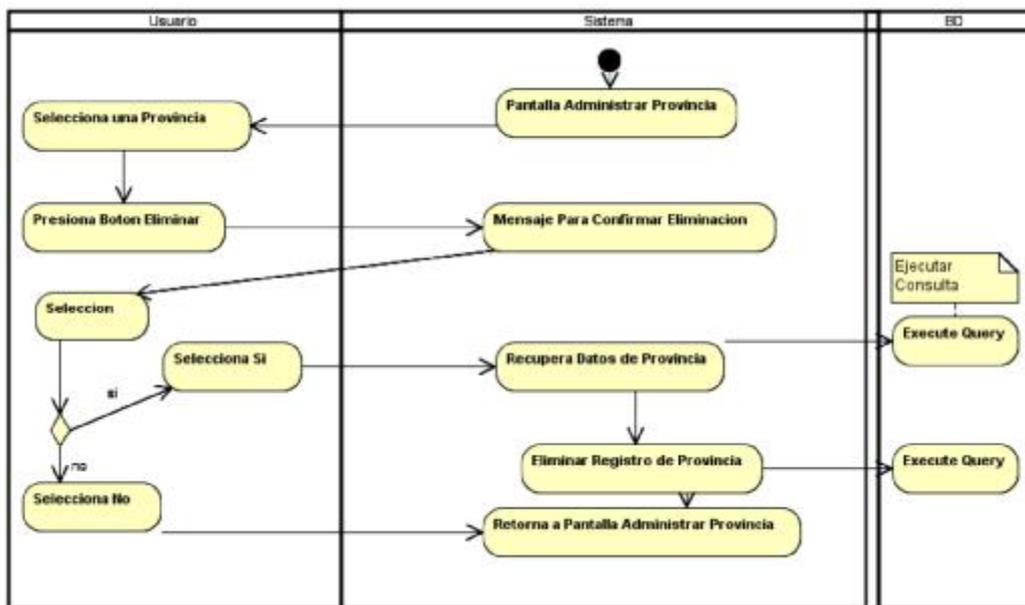


Figura 199. Diagrama de Actividad: CU Eliminar Provincia

2.1.2.2.3.9.1.4.44 Diagrama de Actividad: Caso de Uso Cambiar Estado Provincia

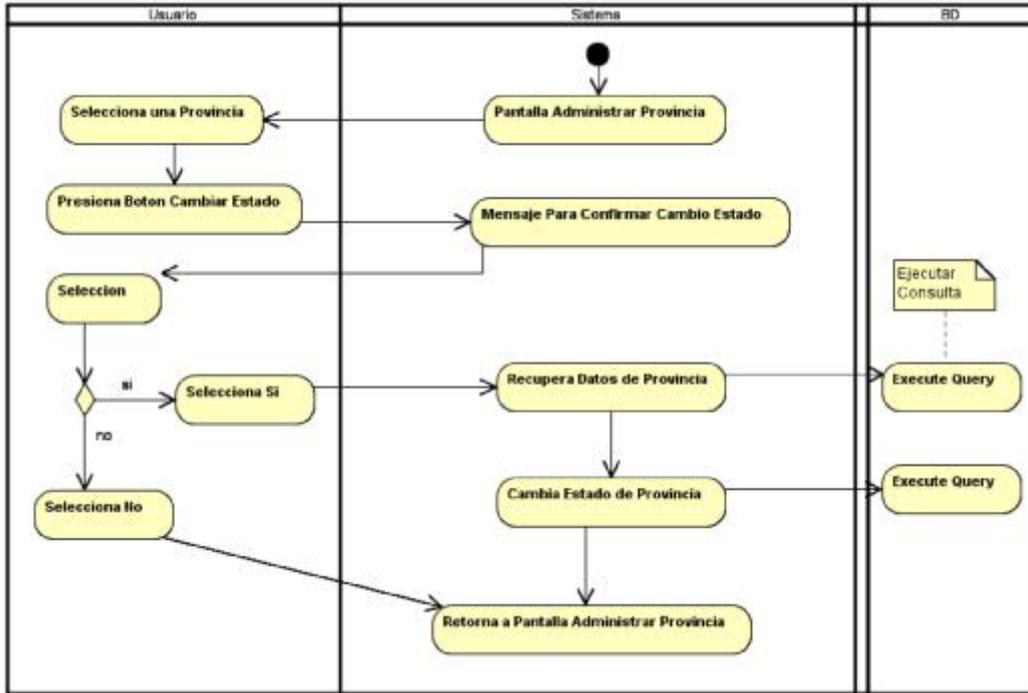


Figura 200. Diagrama de Actividad: CU Cambiar Estado Provincia

2.1.2.2.3.9.1.4.45 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Provincia

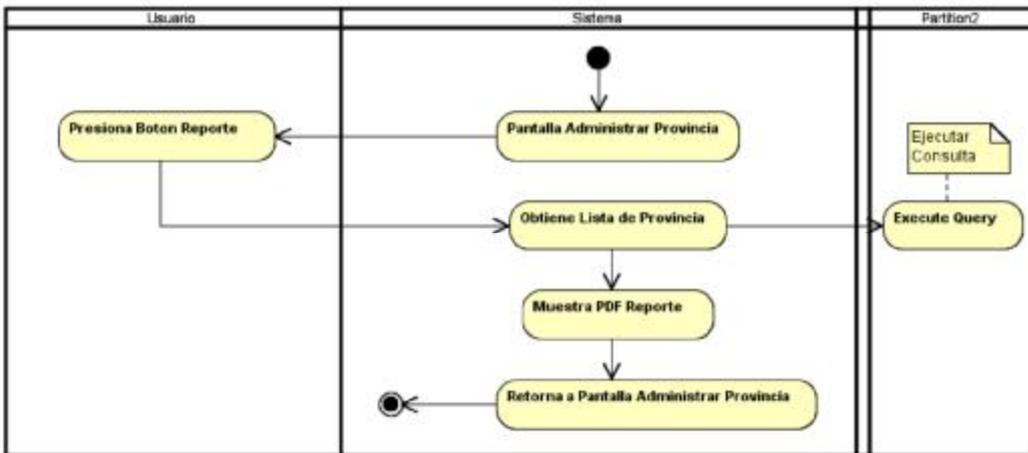


Figura 201. Diagrama de Actividad: CU Generar Reporte Lista Provincia

2.1.2.2.3.9.1.4.46 Diagrama de Actividad: Caso de Uso Administrar Requisito

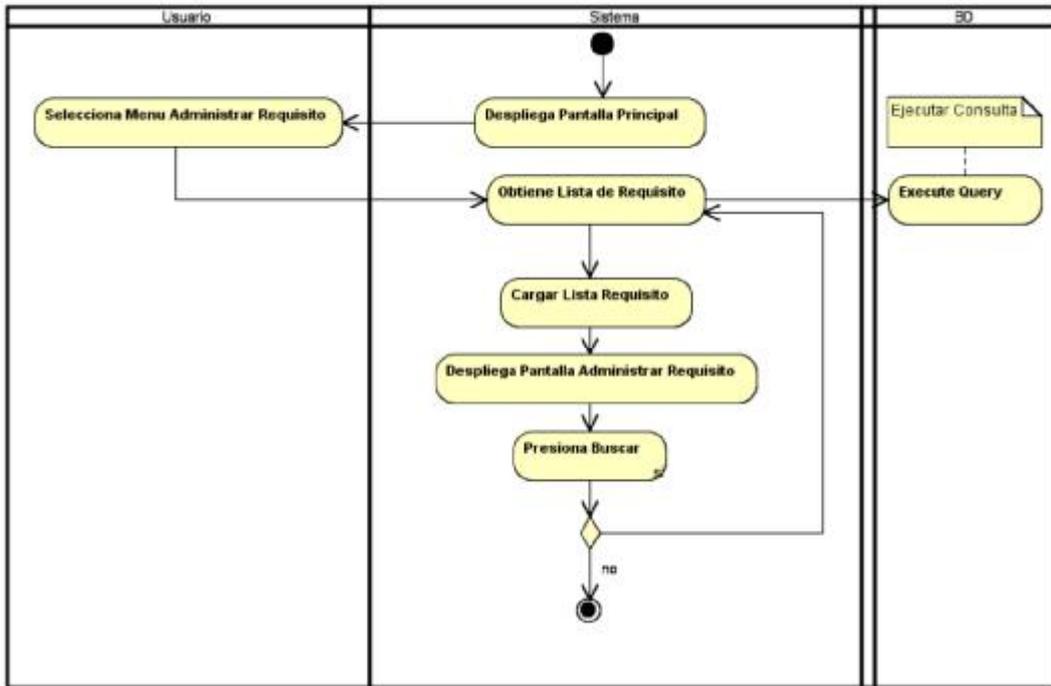


Figura 202. Diagrama de Actividad: CU Administrar Requisito

2.1.2.2.3.9.1.4.47 Diagrama de Actividad: Caso de Uso Adicionar Requisito

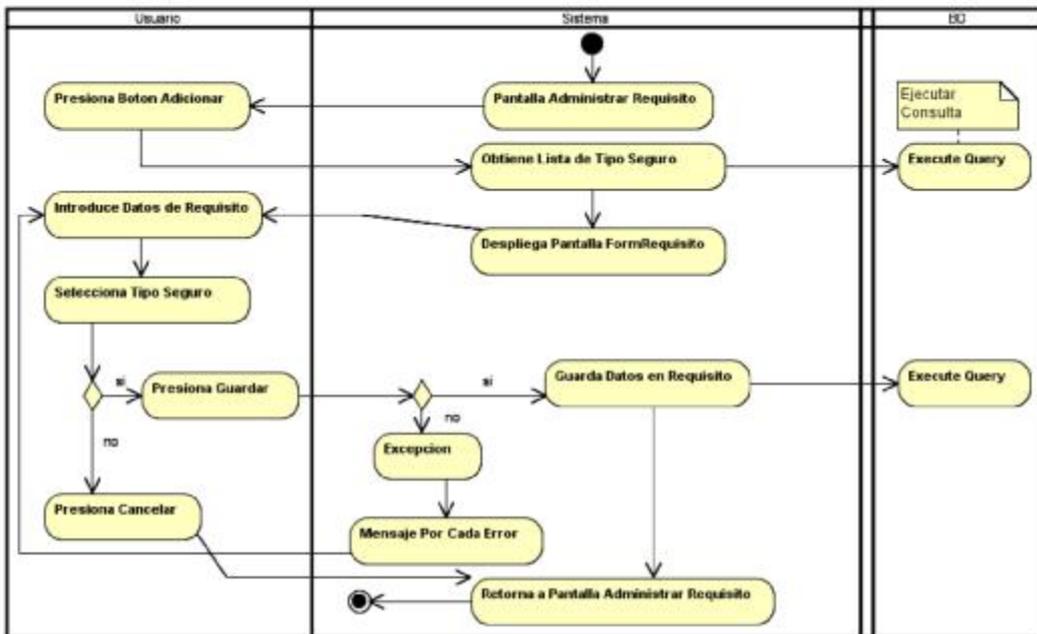


Figura 203. Diagrama de Actividad: CU Adicionar Requisito

2.1.2.2.3.9.1.4.48 Diagrama de Actividad: Caso de Uso Modificar Requisito

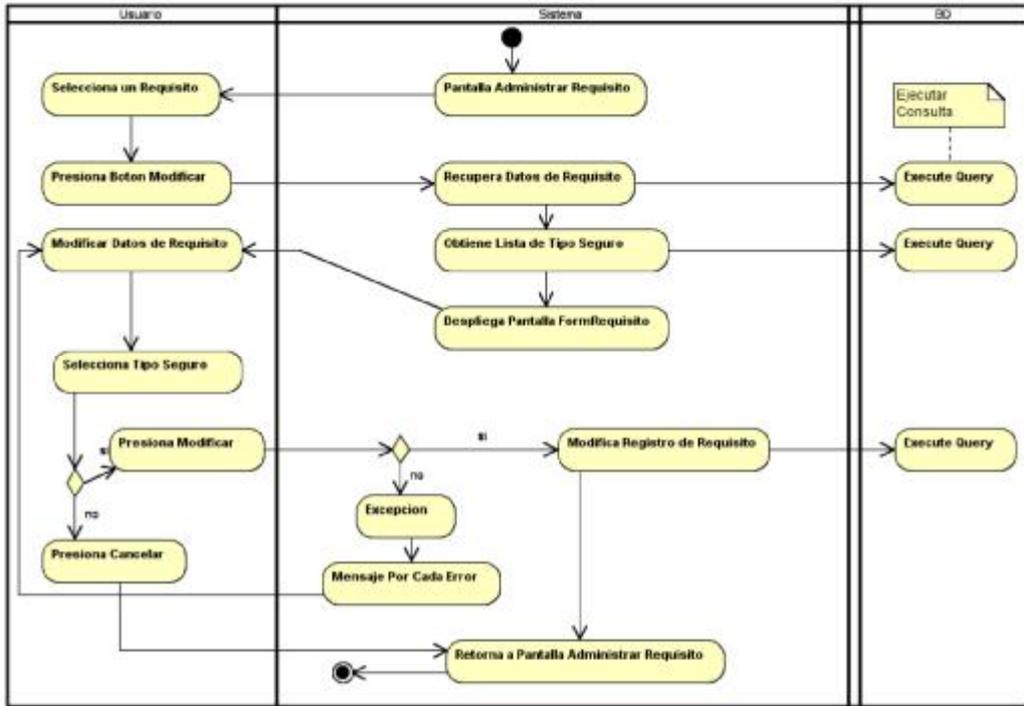


Figura 204. Diagrama de Actividad: CU Modificar Requisito

2.1.2.2.3.9.1.4.49 Diagrama de Actividad: Caso de Uso Eliminar Requisito

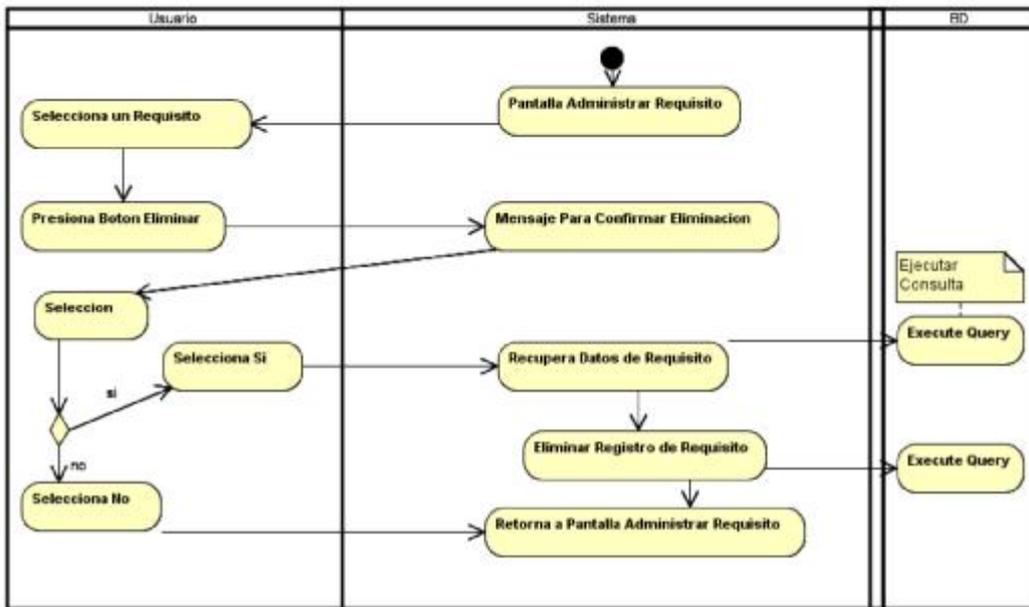


Figura 205. Diagrama de Actividad: CU Eliminar Requisito

2.1.2.2.3.9.1.4.50 Diagrama de Actividad: Caso de Uso Cambiar Estado Requisito

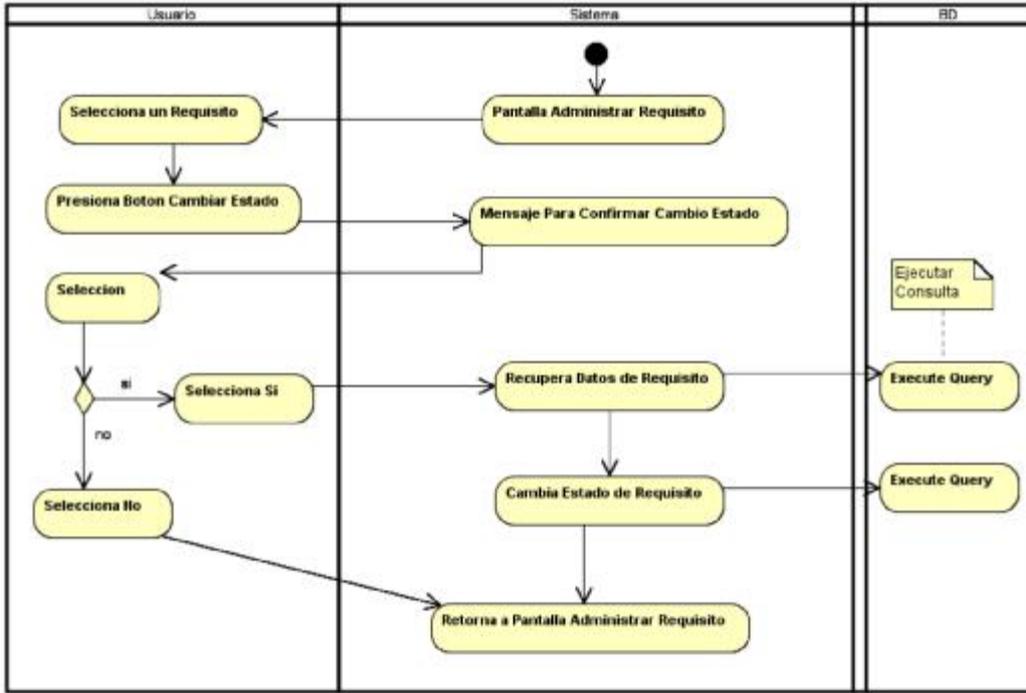


Figura 206. Diagrama de Actividad: CU Cambiar Estado Requisito

2.1.2.2.3.9.1.4.51 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Requisito

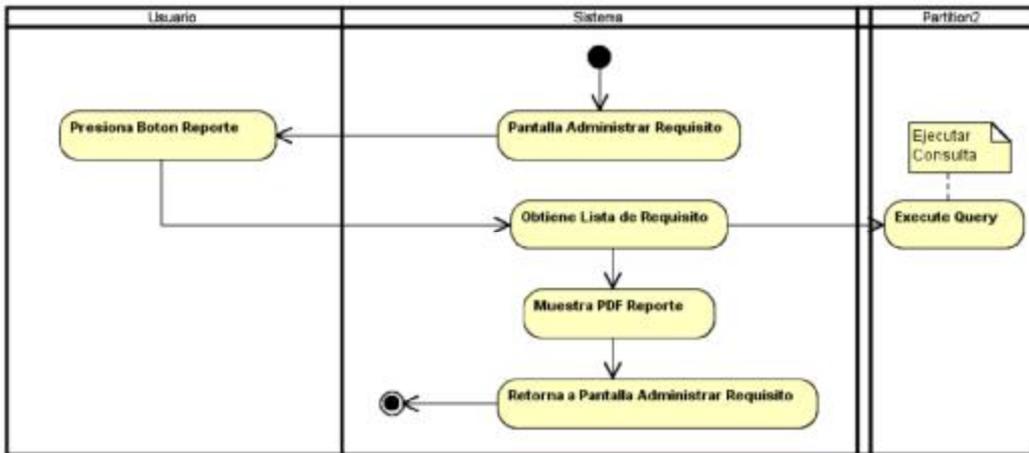


Figura 207. Diagrama de Actividad: CU Generar Reporte Lista Requisito

2.1.2.2.3.9.1.4.52 Diagrama de Actividad: Caso de Uso Administrar Rol

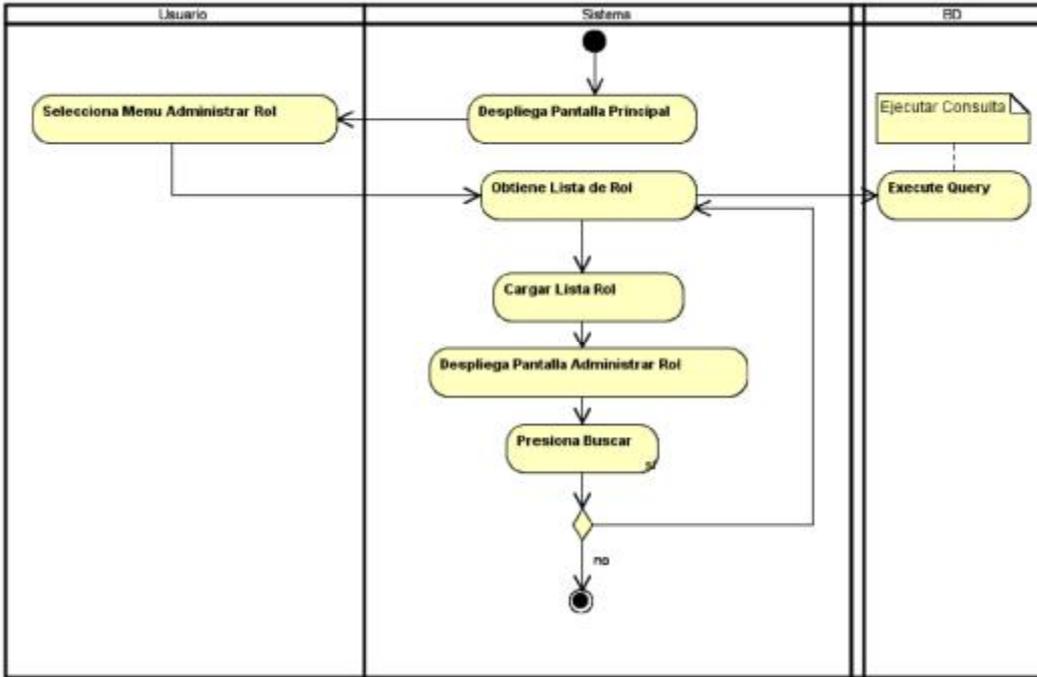


Figura 208. Diagrama de Actividad: CU Administrar Rol

2.1.2.2.3.9.1.4.53 Diagrama de Actividad: Caso de Uso Adicionar Rol

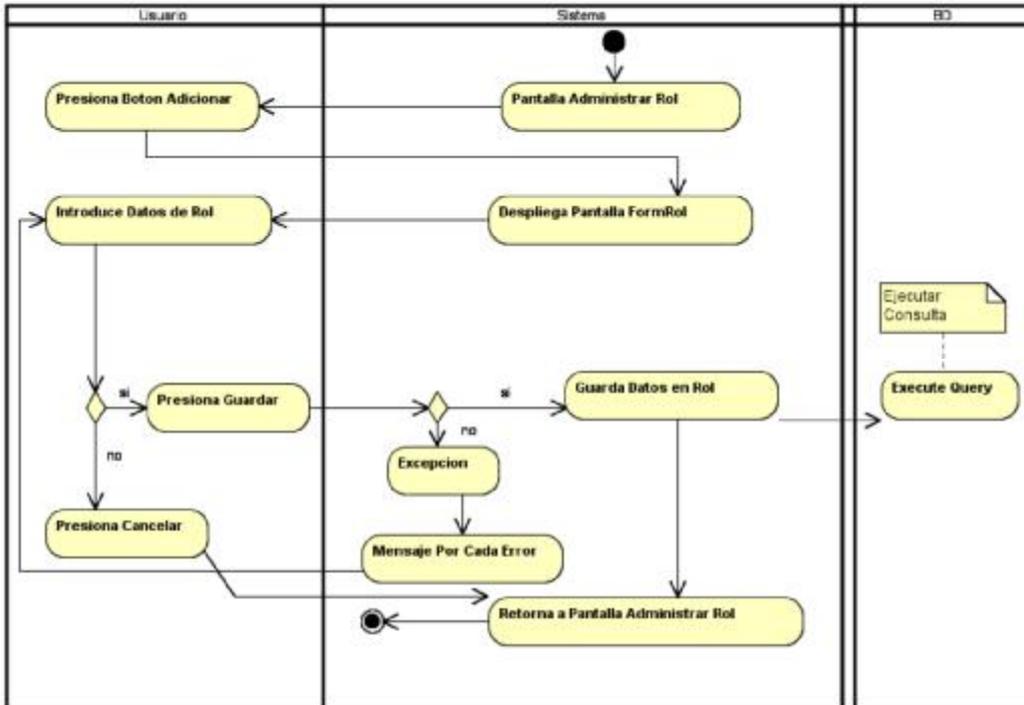


Figura 209. Diagrama de Actividad: CU Adicionar Rol

2.1.2.2.3.9.1.4.54 Diagrama de Actividad: Caso de Uso Modificar Rol

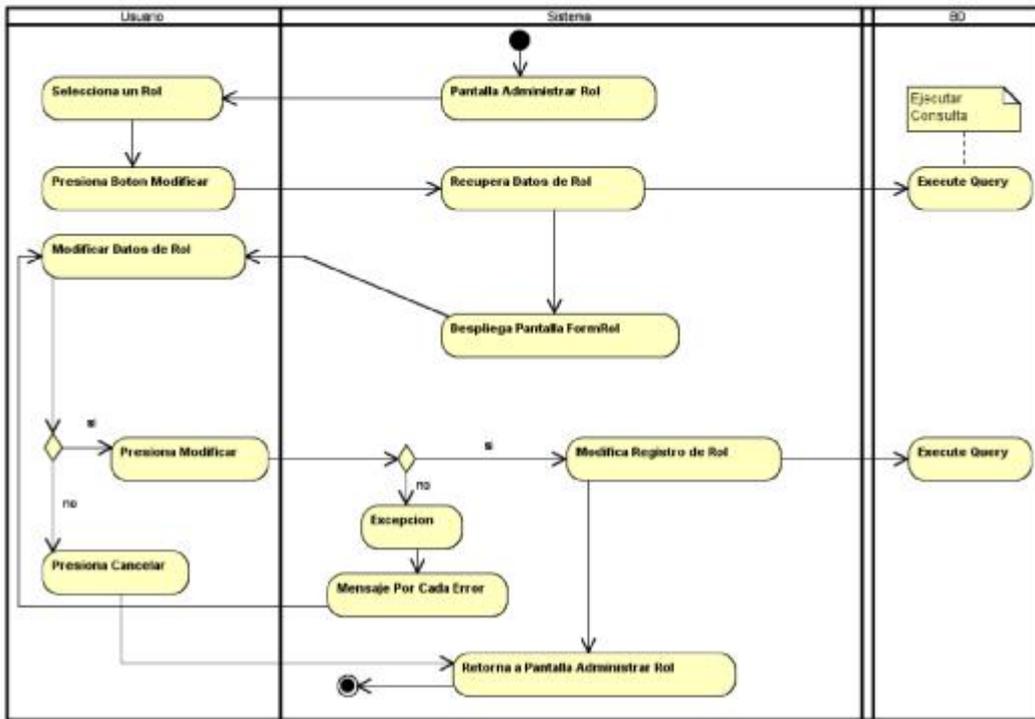


Figura 210. Diagrama de Actividad: CU Modificar Rol

2.1.2.2.3.9.1.4.55 Diagrama de Actividad: Caso de Uso Eliminar Rol

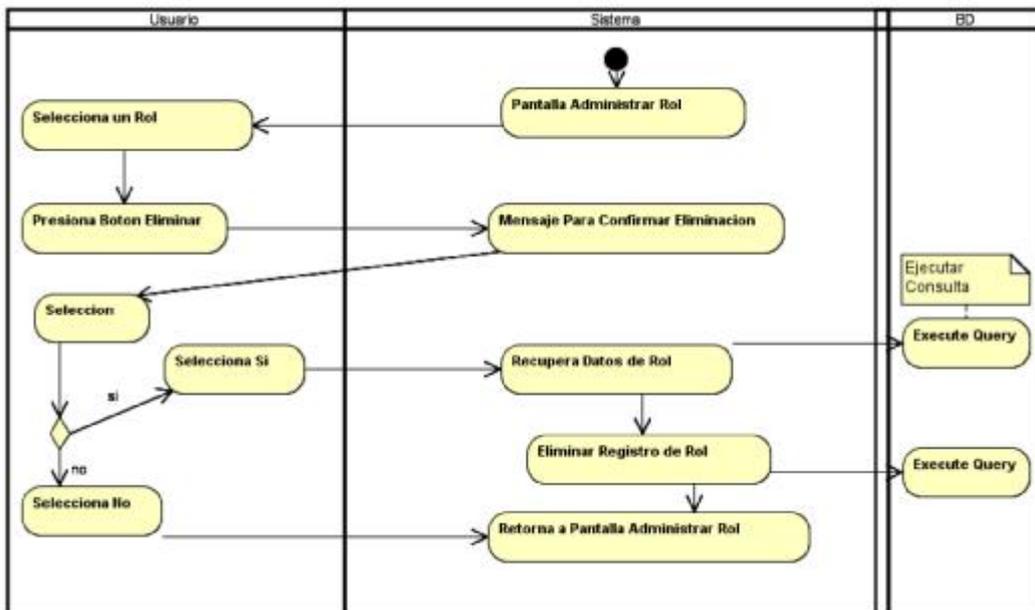


Figura 211. Diagrama de Actividad: CU Eliminar Rol

2.1.2.2.3.9.1.4.56 Diagrama de Actividad: Caso de Uso Cambiar Estado Rol

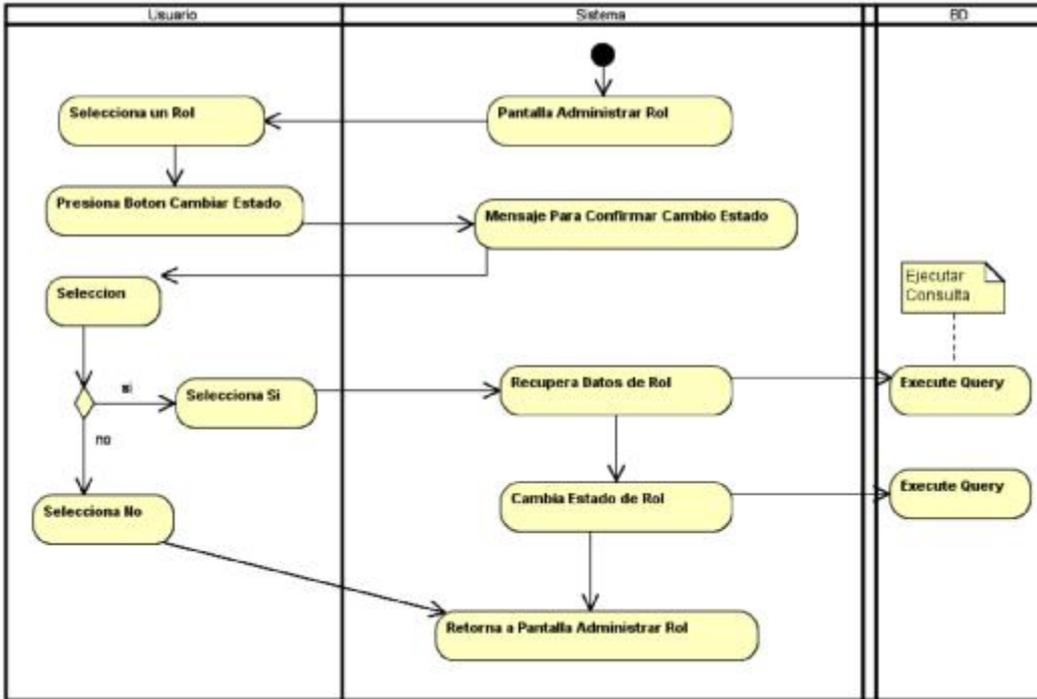


Figura 212. Diagrama de Actividad: CU Cambiar Estado Rol

2.1.2.2.3.9.1.4.57 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Rol

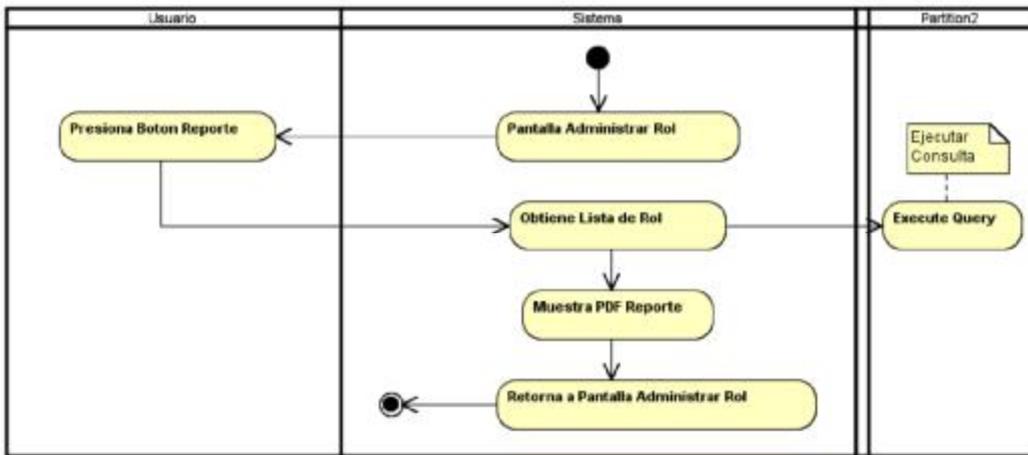


Figura 213. Diagrama de Actividad: CU Generar Reporte Lista Rol

2.1.2.2.3.9.1.4.58 Diagrama de Actividad: Caso de Uso Asignar Funcionalidad

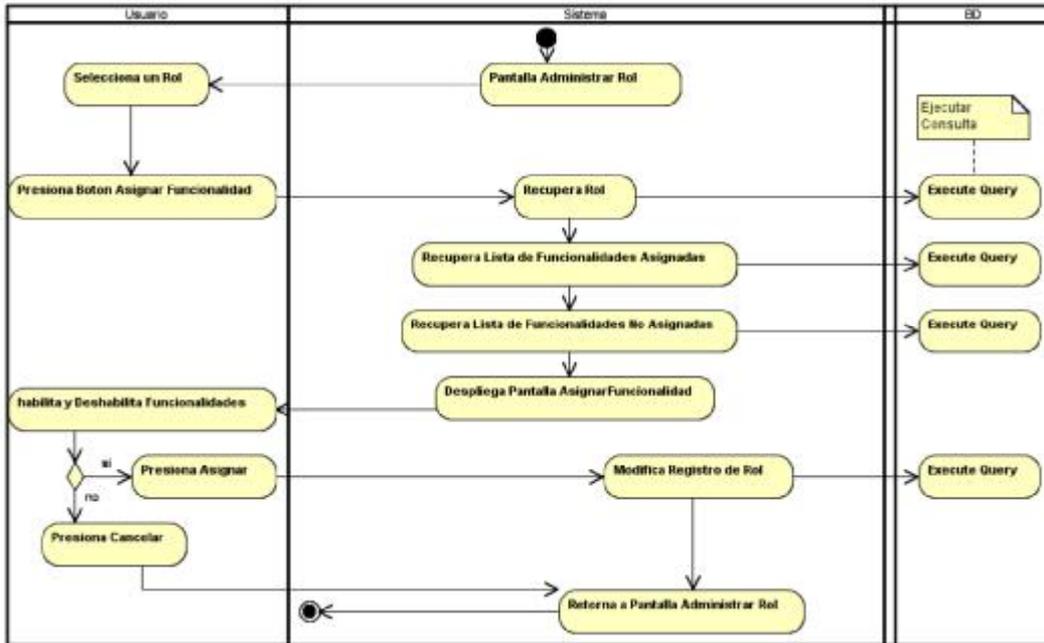


Figura 214. Diagrama de Actividad: CU Asignar Funcionalidad

2.1.2.2.3.9.1.4.59 Diagrama de Actividad: Caso de Uso Administrar Tipo Empresa

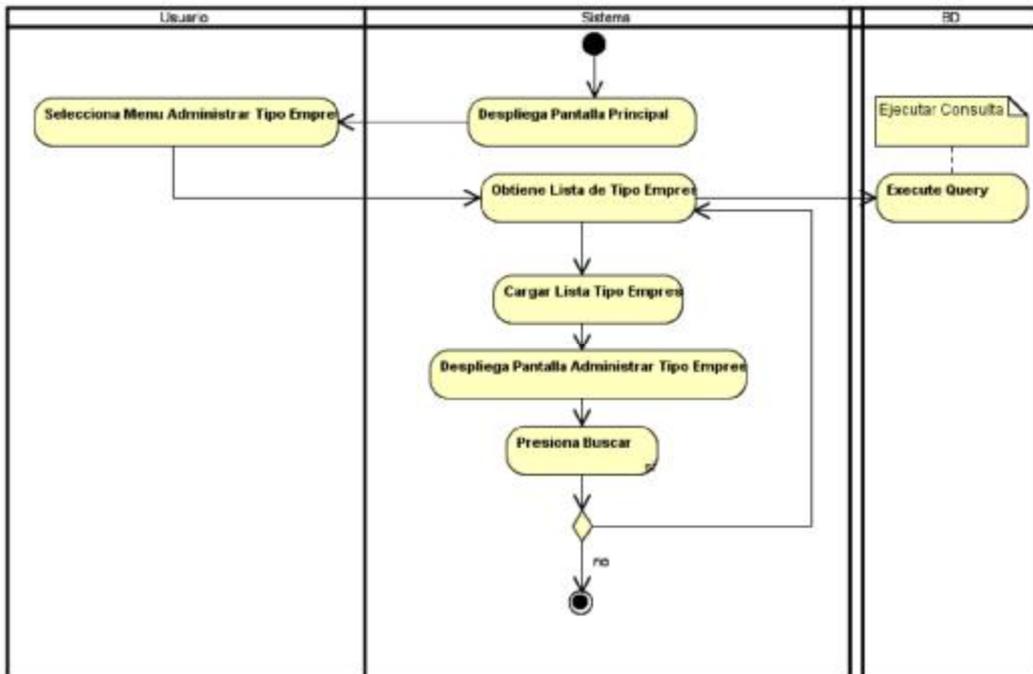


Figura 215. Diagrama de Actividad: CU Administrar Tipo Empresa

2.1.2.2.3.9.1.4.60 Diagrama de Actividad: Caso de Uso Adicionar Tipo Empresa

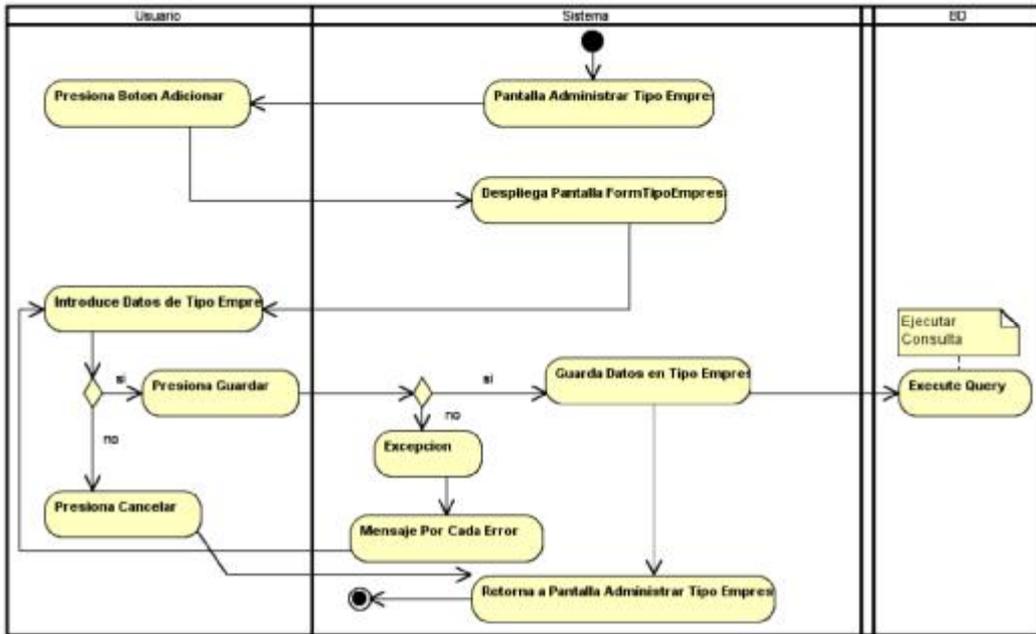


Figura 216. Diagrama de Actividad: CU Adicionar Tipo Empresa

2.1.2.2.3.9.1.4.61 Diagrama de Actividad: Caso de Uso Modificar Tipo Empresa

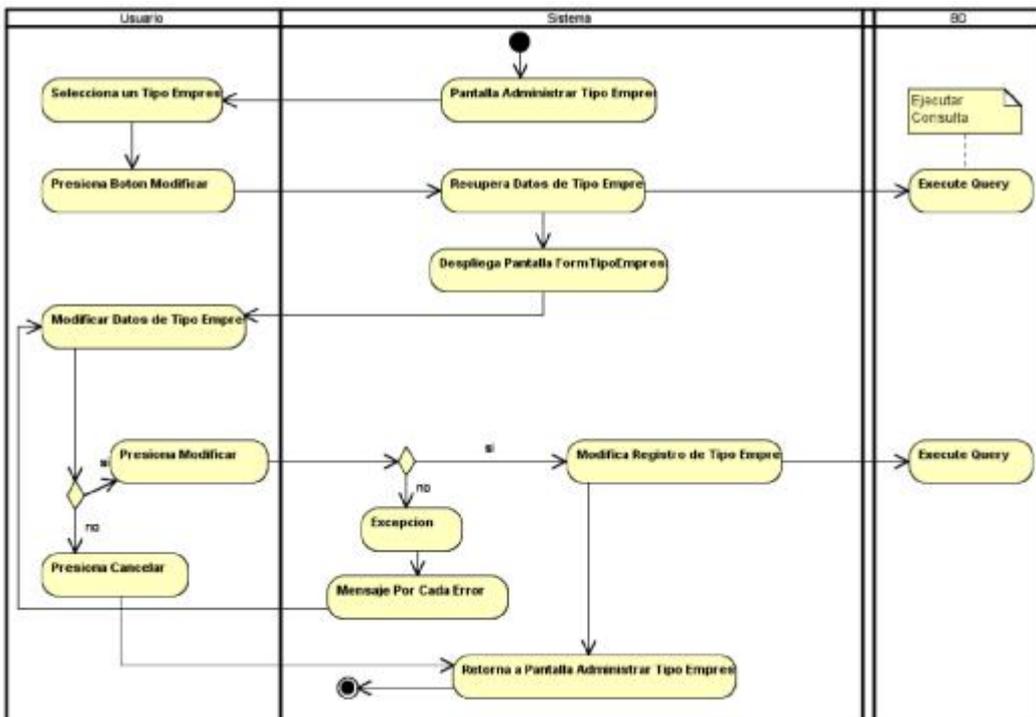


Figura 217. Diagrama de Actividad: CU Modificar Tipo Empresa

2.1.2.2.3.9.1.4.62 Diagrama de Actividad: Caso de Uso Eliminar Tipo Empresa

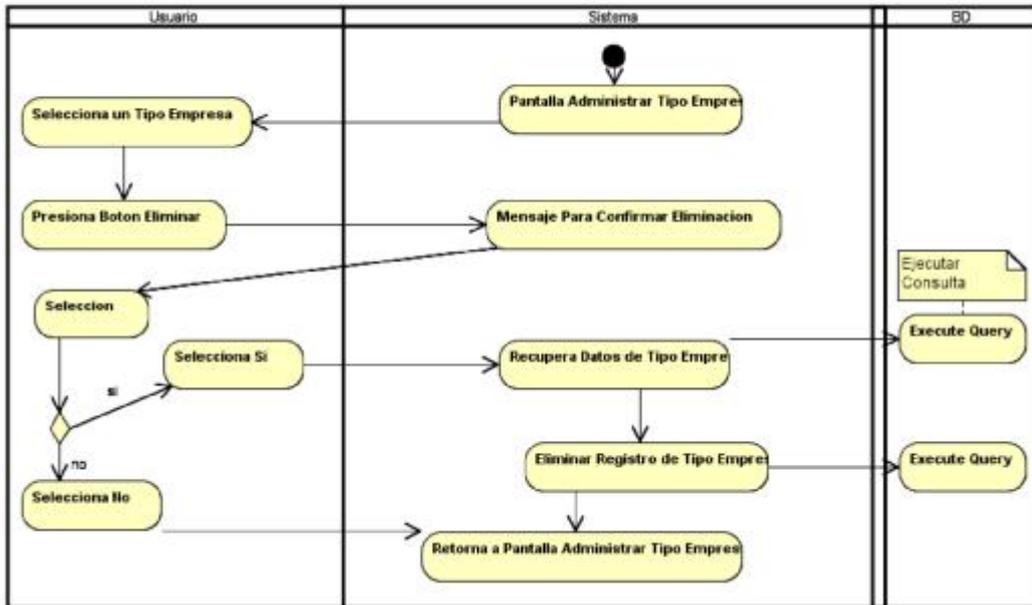


Figura 218. Diagrama de Actividad: CU Eliminar Tipo Empresa

2.1.2.2.3.9.1.4.63 Diagrama de Actividad: Caso de Uso Cambiar Estado Tipo Empresa

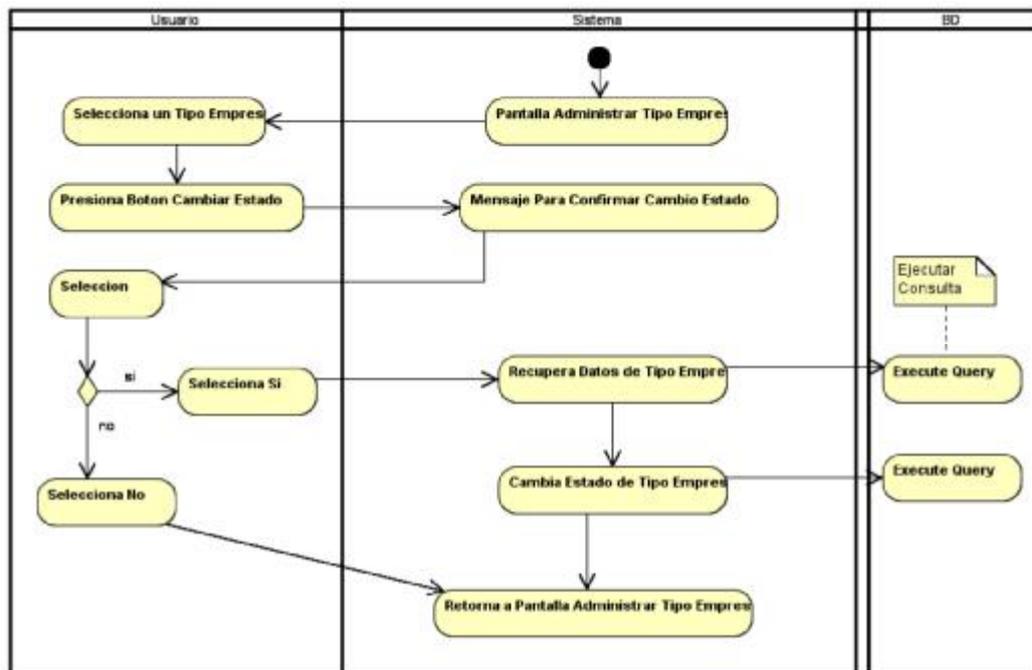


Figura 219. Diagrama de Actividad: CU Cambiar Estado Tipo Empresa

2.1.2.2.3.9.1.4.64 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Tipo Empresa

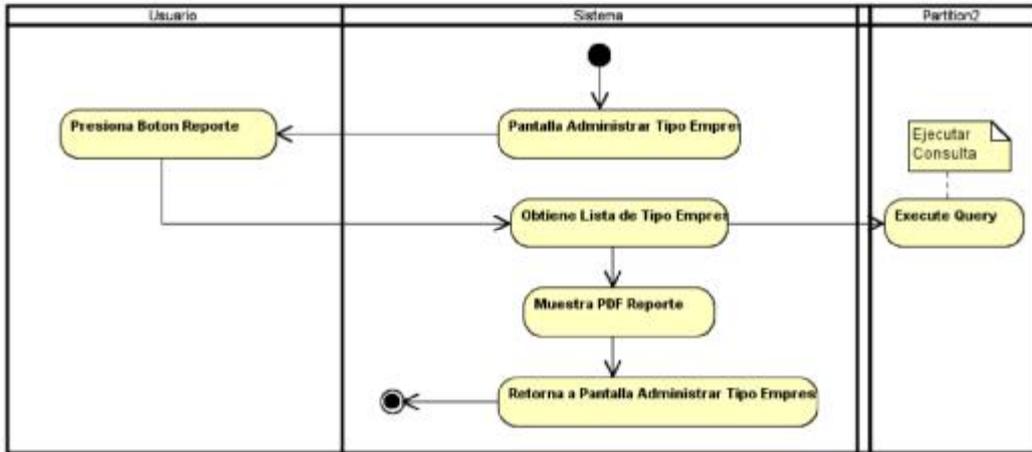


Figura 220. Diagrama de Actividad: CU Generar Reporte Lista Tipo Empresa

2.1.2.2.3.9.1.4.65 Diagrama de Actividad: Caso de Uso Administrar Tipo Seguro

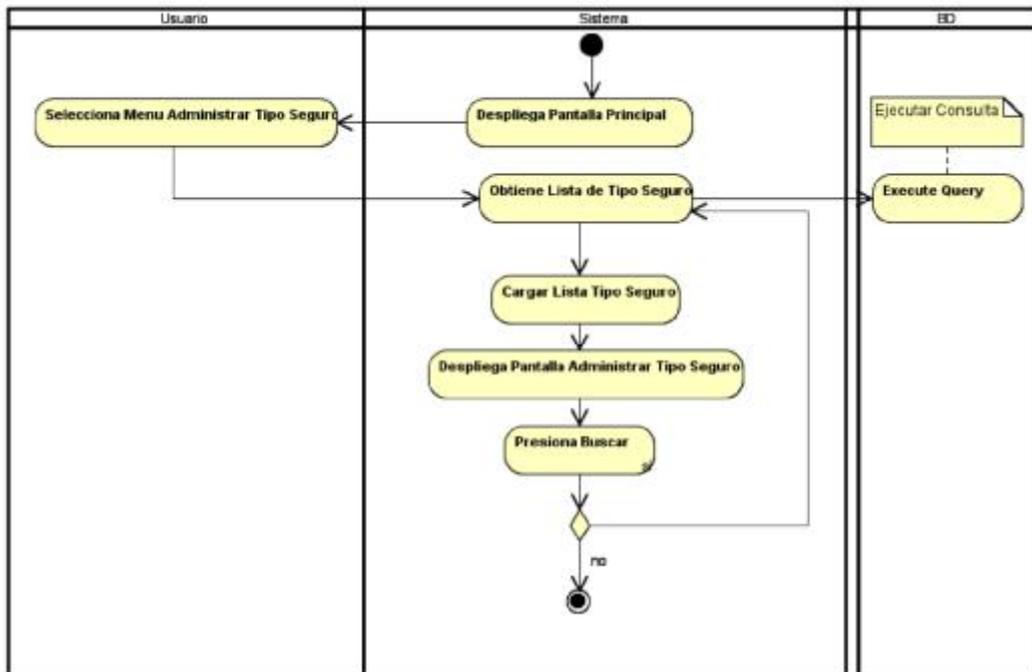


Figura 221. Diagrama de Actividad: CU Administrar Tipo Seguro

2.1.2.2.3.9.1.4.66 Diagrama de Actividad: Caso de Uso Adicionar Tipo Seguro

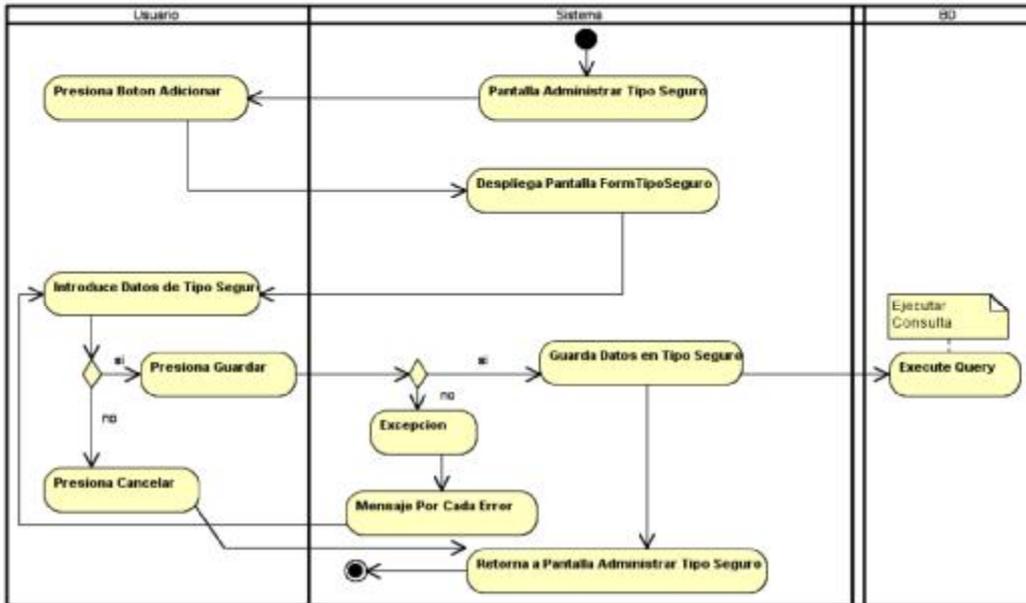


Figura 222. Diagrama de Actividad: CU Adicionar Tipo Seguro

2.1.2.2.3.9.1.4.67 Diagrama de Actividad: Caso de Uso Modificar Tipo Seguro

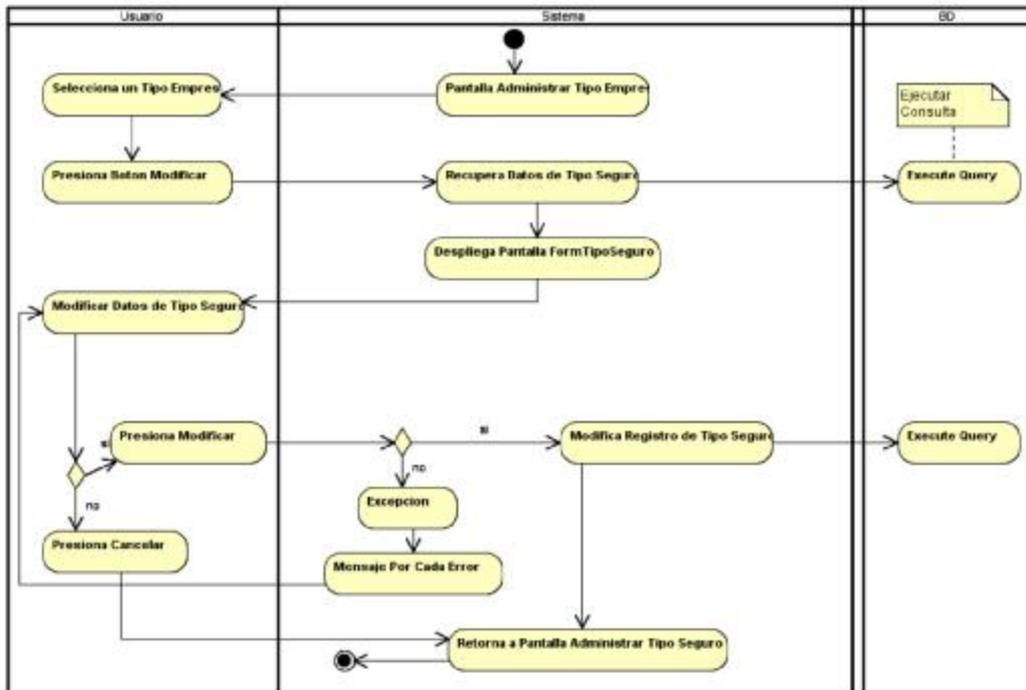


Figura 223. Diagrama de Actividad: CU Modificar Tipo Seguro

2.1.2.2.3.9.1.4.68 Diagrama de Actividad: Caso de Uso Eliminar Tipo Seguro

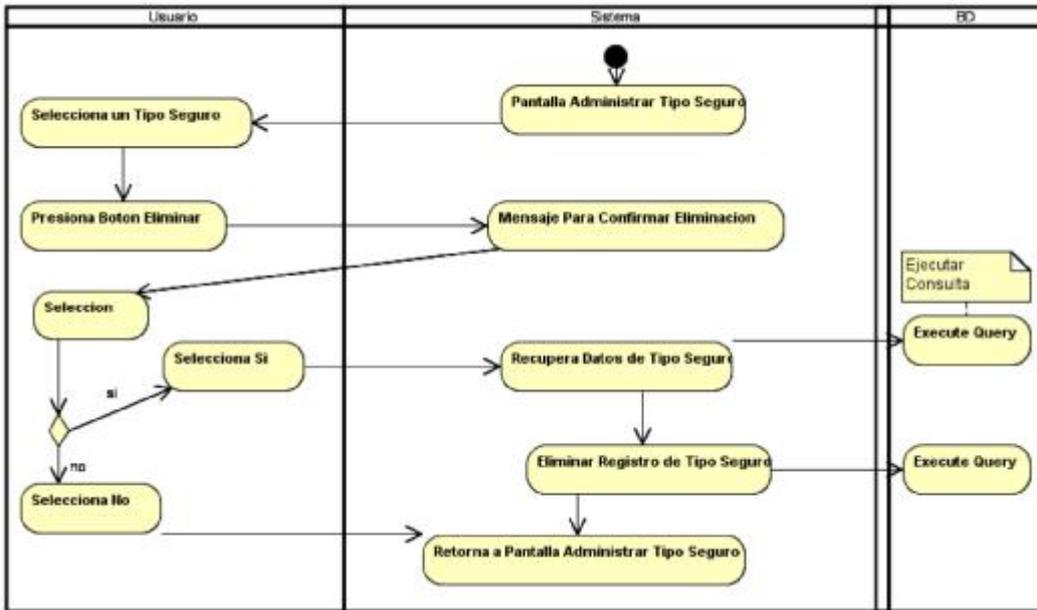


Figura 224. Diagrama de Actividad: CU Eliminar Tipo Seguro

2.1.2.2.3.9.1.4.69 Diagrama de Actividad: Caso de Uso Cambiar Estado Tipo Seguro

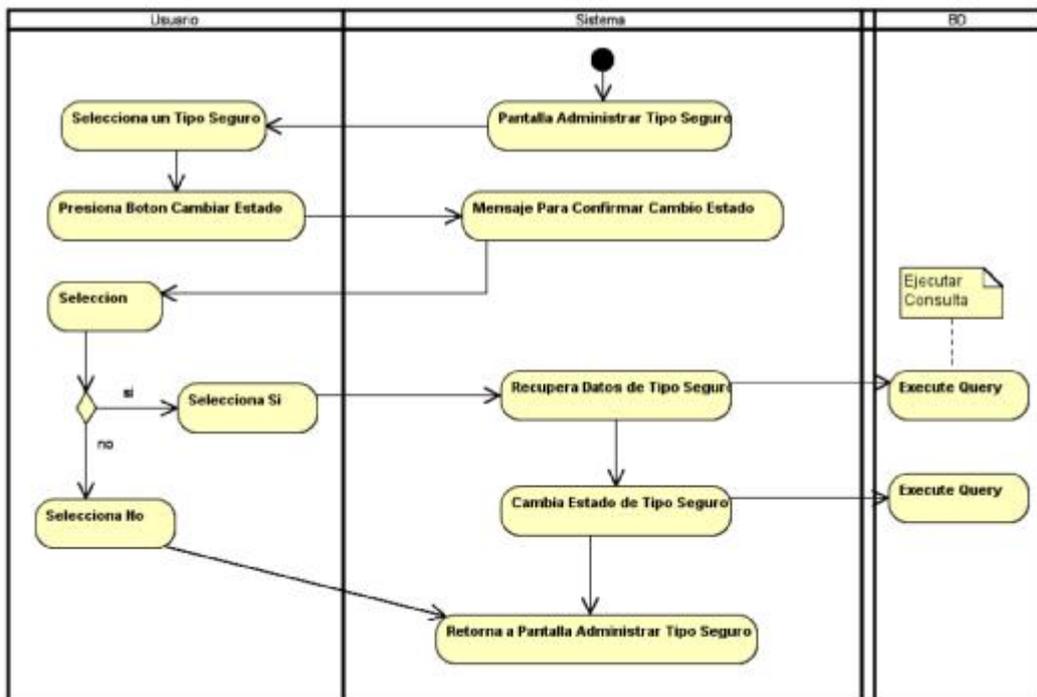


Figura 225. Diagrama de Actividad: CU Cambiar Estado Tipo Seguro

2.1.2.2.3.9.1.4.70 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Tipo

Seguro

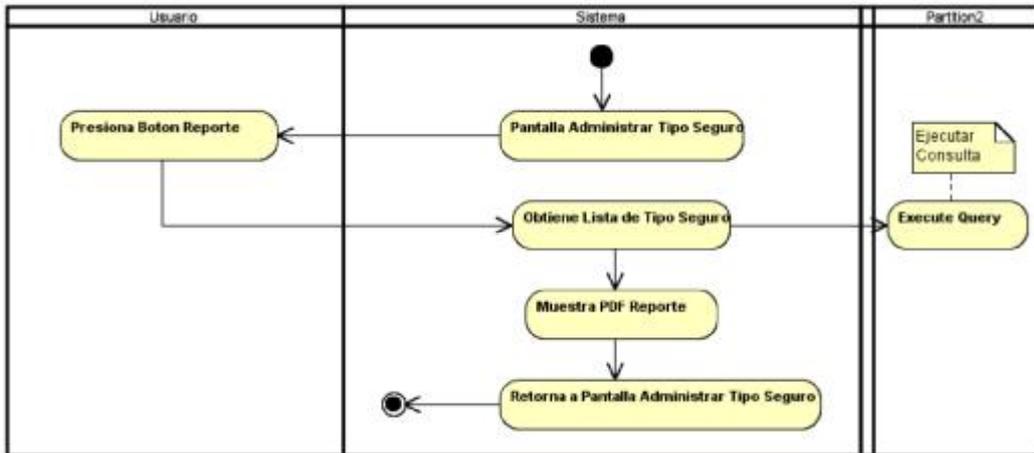


Figura 226. Diagrama de Actividad: CU Generar Reporte Lista Tipo Seguro
 2.1.2.2.3.9.1.4.71 Diagrama de Actividad: Caso de Uso Administrar Usuario

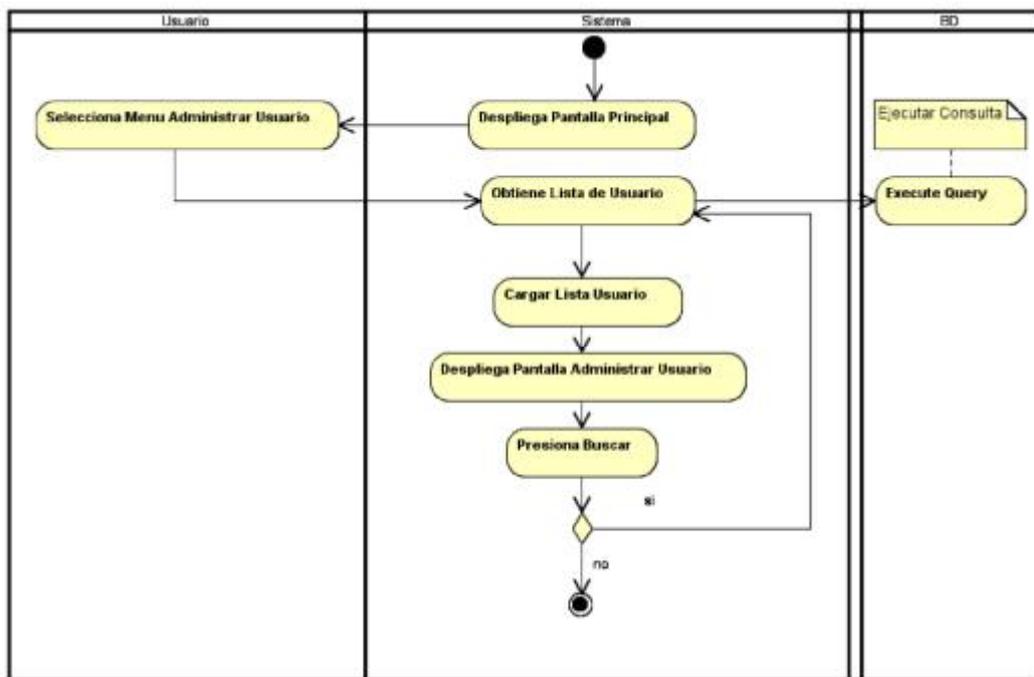


Figura 227. Diagrama de Actividad: CU Administrar Usuario

2.1.2.2.3.9.1.4.72 Diagrama de Actividad: Caso de Uso Adicionar Usuario

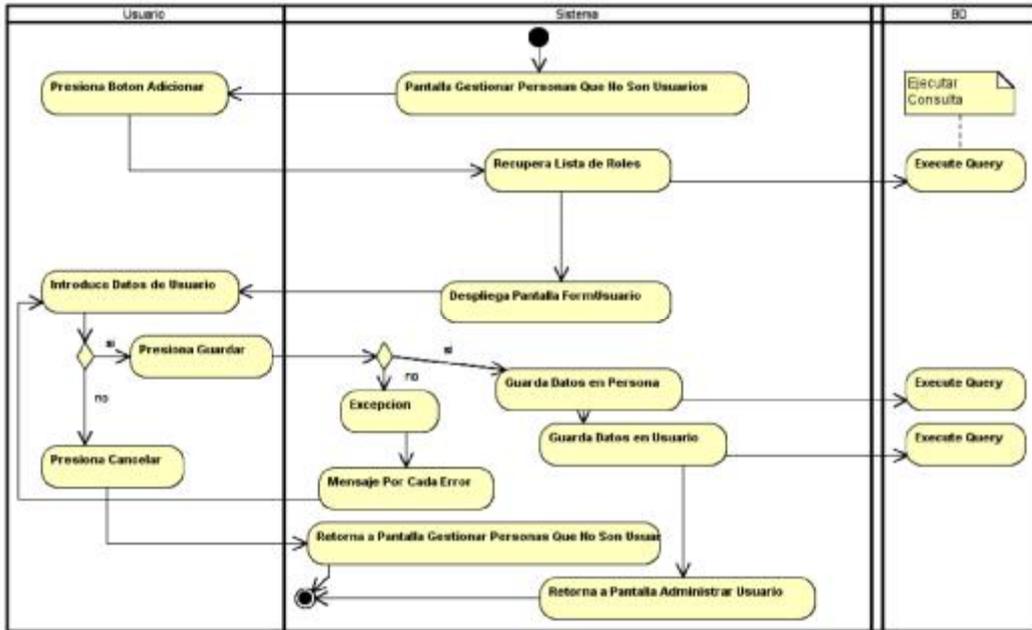


Figura 228. Diagrama de Actividad: CU Adicionar Usuario

2.1.2.2.3.9.1.4.73 Diagrama de Actividad: Caso de Uso Modificar Usuario

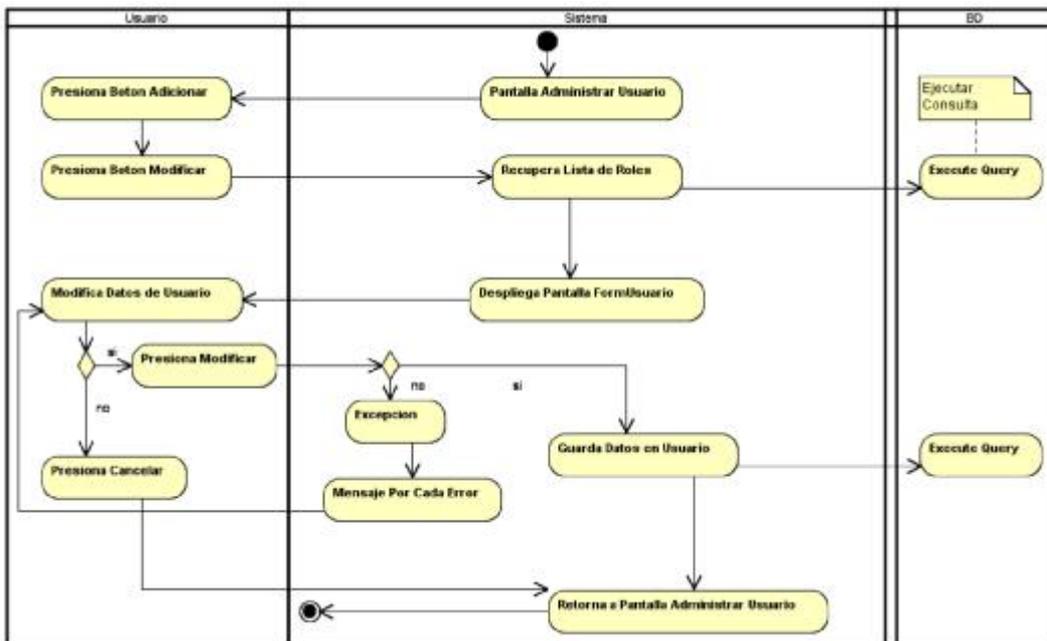


Figura 229. Diagrama de Actividad: CU Modificar Usuario

2.1.2.2.3.9.1.4.74 Diagrama de Actividad: Caso de Uso Eliminar Usuario

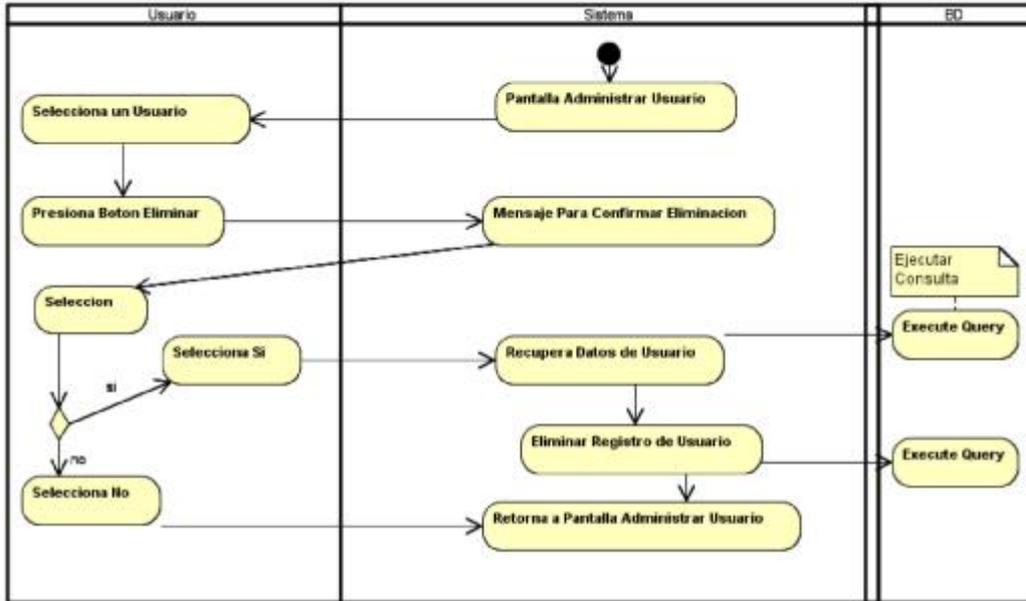


Figura 230. Diagrama de Actividad: CU Eliminar Usuario

2.1.2.2.3.9.1.4.75 Diagrama de Actividad: Caso de Uso Cambiar Estado Usuario

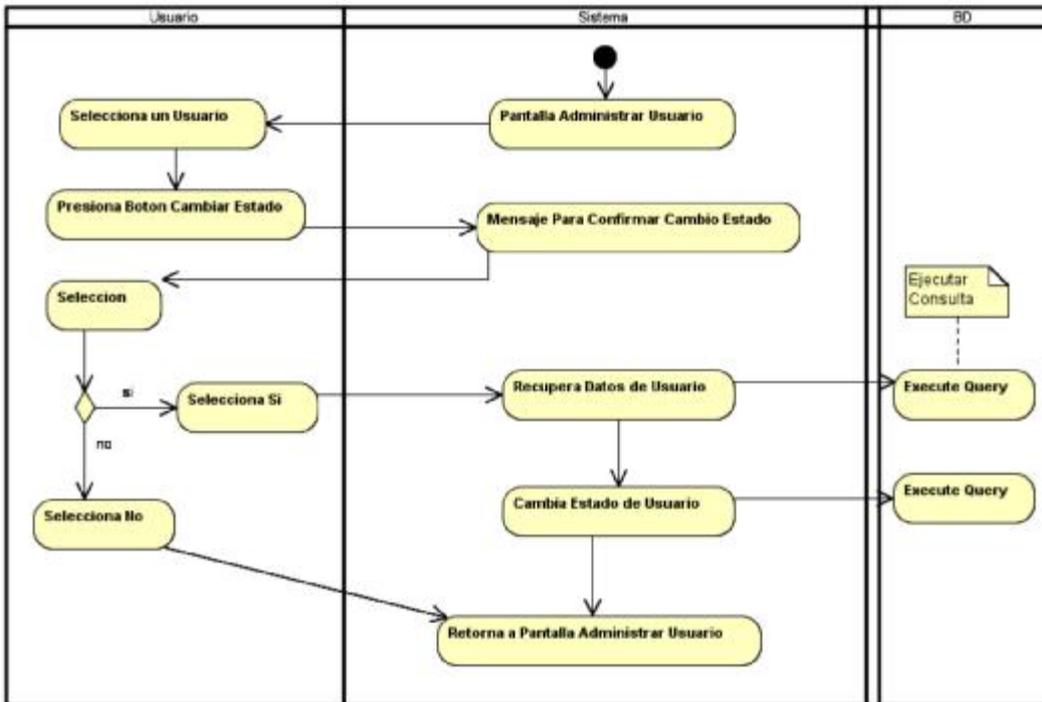


Figura 231. Diagrama de Actividad: CU Cambiar Estado Usuario

2.1.2.2.3.9.1.4.76 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Usuario

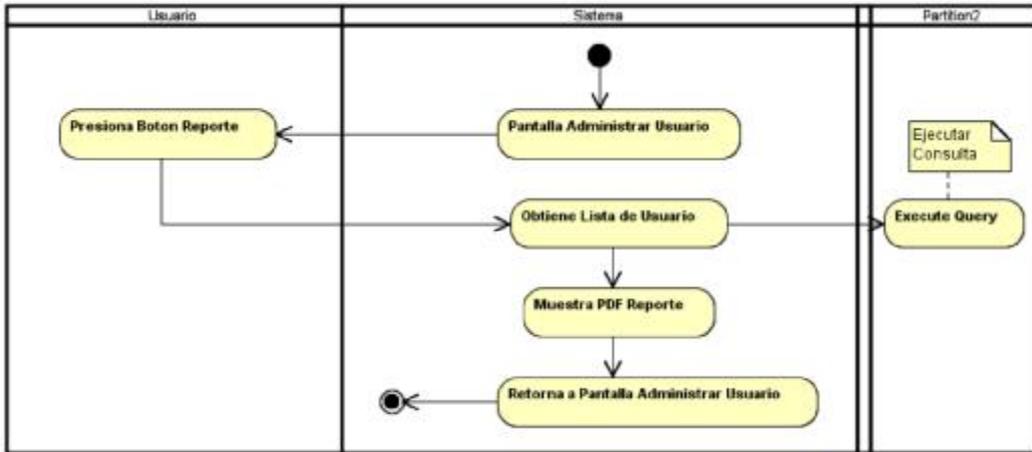


Figura 232. Diagrama de Actividad: CU Generar Reporte Lista Usuario

2.1.2.2.3.9.1.4.77 Diagrama de Actividad: Caso de Uso Generar Reporte Ficha Usuario

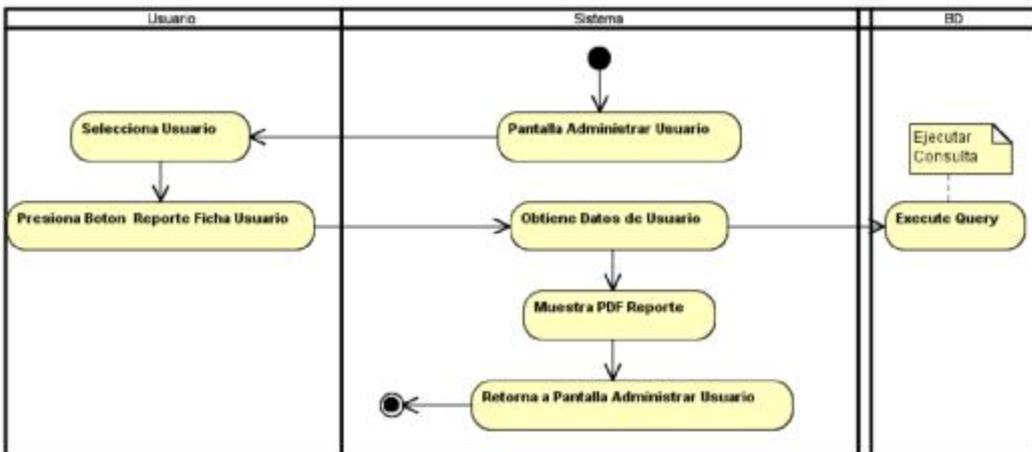


Figura 233. Diagrama de Actividad: CU Generar Reporte Ficha Usuario

2.1.2.2.3.9.1.4.78 Diagrama de Actividad: Caso de Uso Gestionar Personas

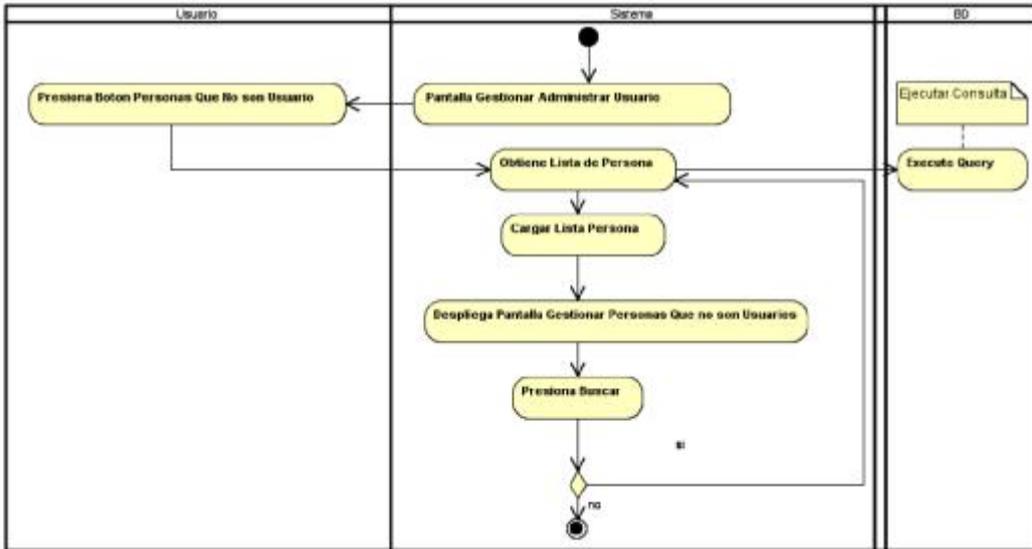


Figura 234. Diagrama de Actividad: CU Gestionar Personas

2.1.2.2.3.9.1.4.79 Diagrama de Actividad: Caso de Uso Asignar Login y Clave

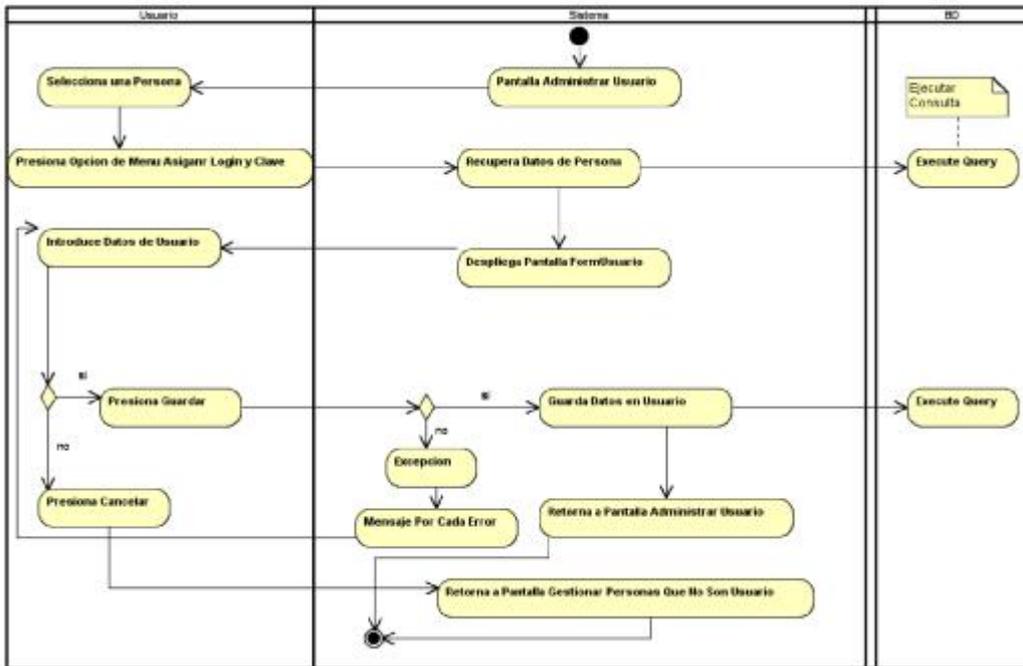


Figura 235. Diagrama de Actividad: CU Asignar Login y Clave

2.1.2.2.3.9.1.4.80 Diagrama de Actividad: Caso de Uso Administrar Zona

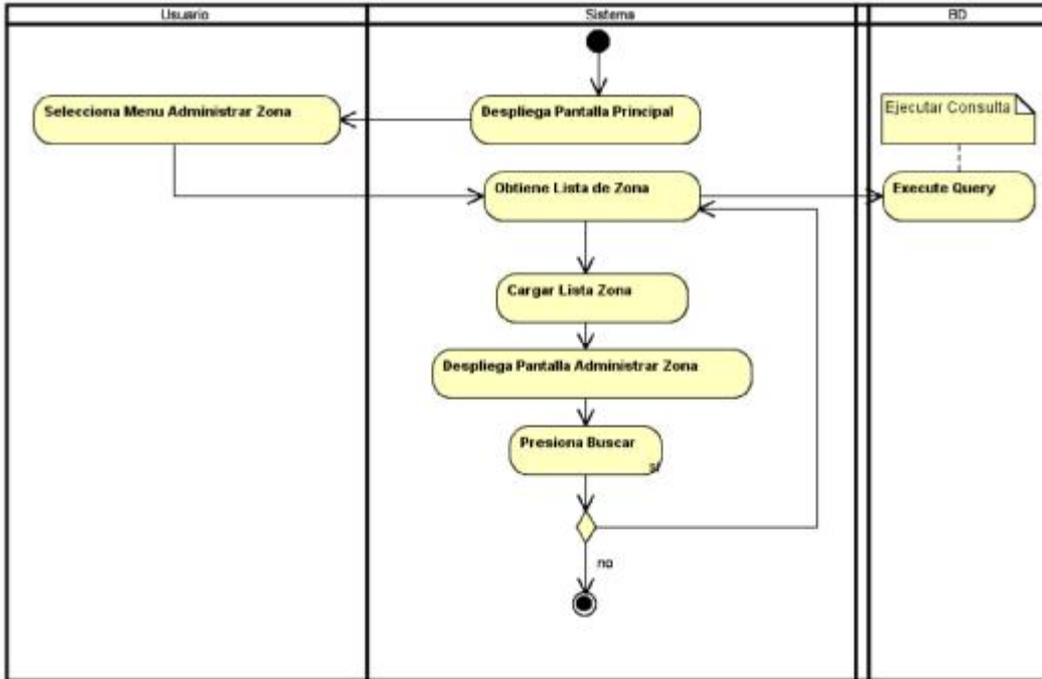


Figura 236. Diagrama de Actividad: CU Administrar Zona

2.1.2.2.3.9.1.4.81 Diagrama de Actividad: Caso de Uso Adicionar Zona

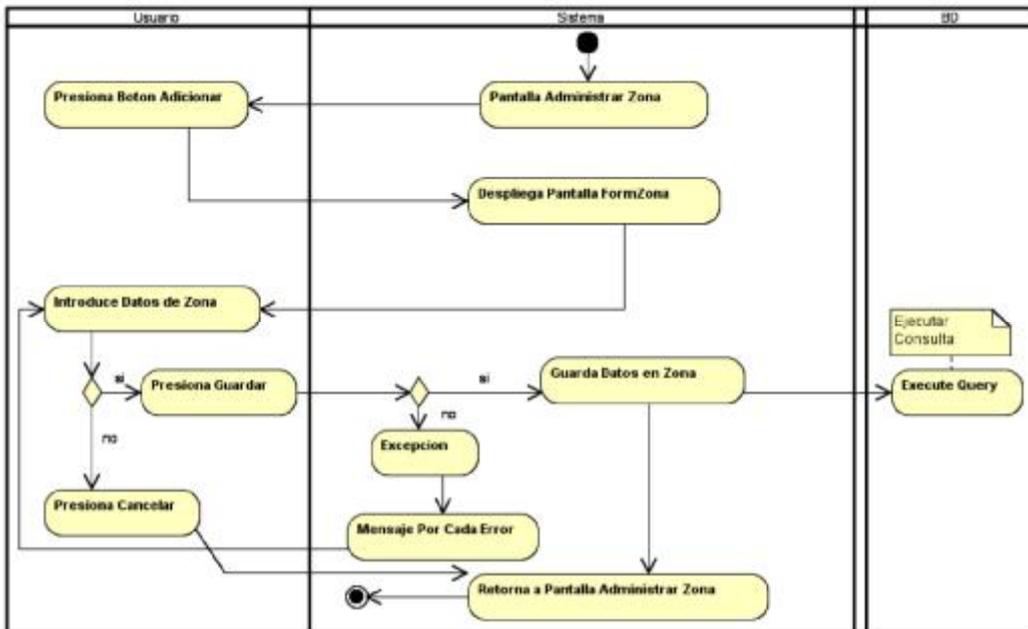


Figura 237. Diagrama de Actividad: CU Adicionar Zona

2.1.2.2.3.9.1.4.82 Diagrama de Actividad: Caso de Uso Modificar Zona

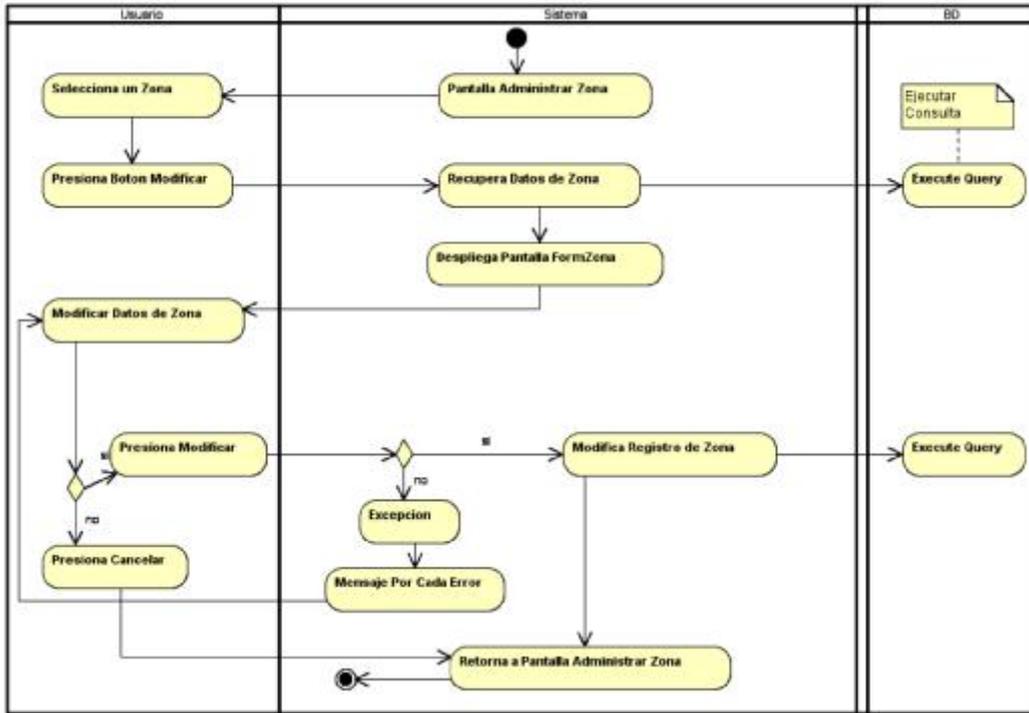


Figura 238. Diagrama de Actividad: CU Modificar Zona

2.1.2.2.3.9.1.4.83 Diagrama de Actividad: Caso de Uso Eliminar Zona

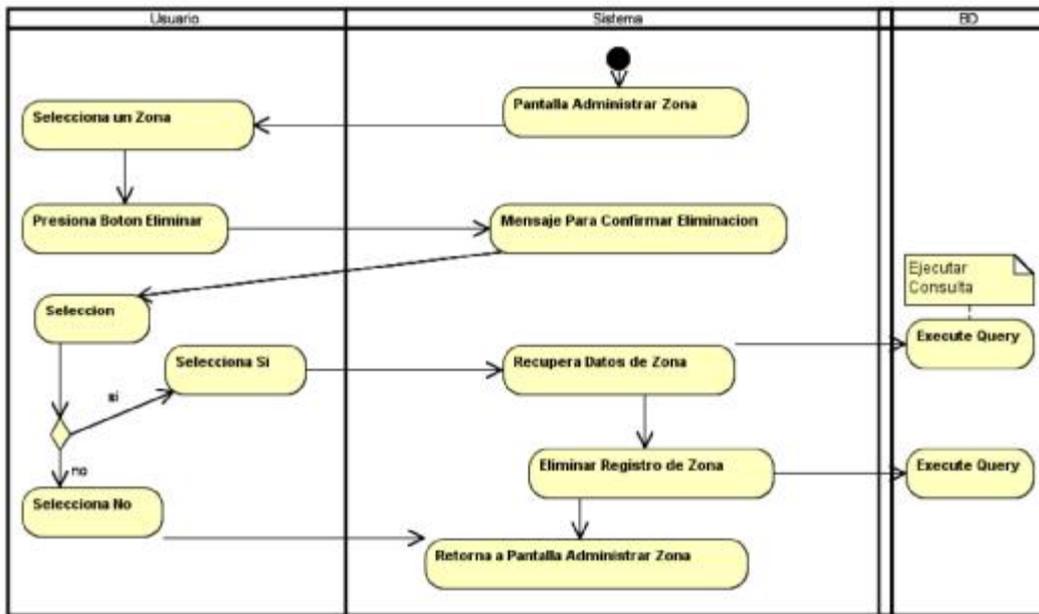


Figura 239. Diagrama de Actividad: CU Eliminar Zona

2.1.2.2.3.9.1.4.84 Diagrama de Actividad: Caso de Uso Cambiar Estado Zona

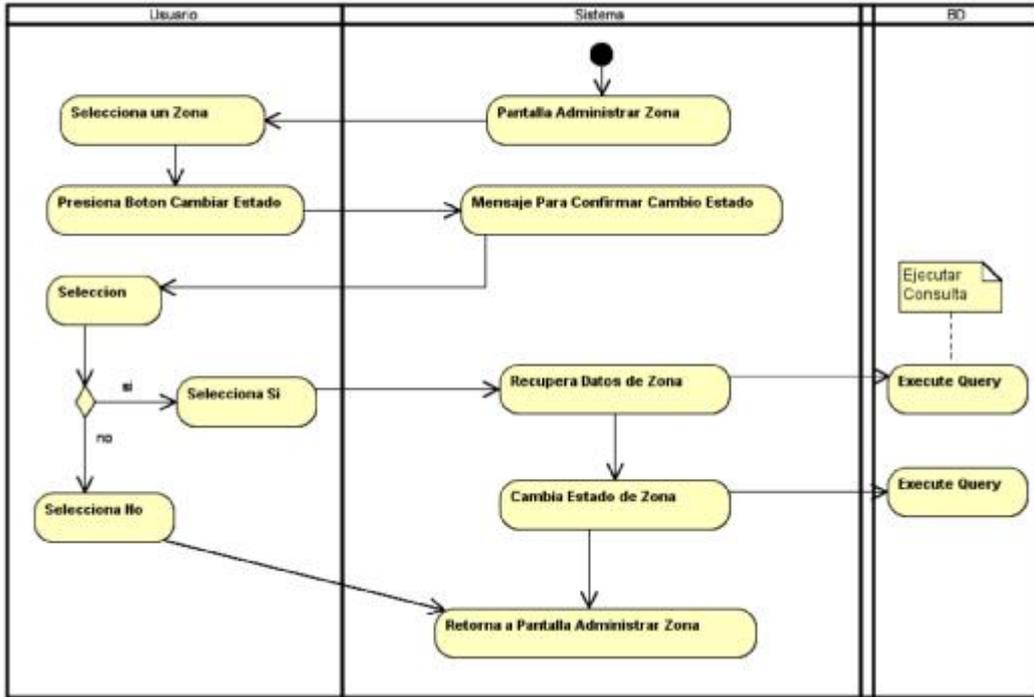


Figura 240. Diagrama de Actividad: CU Cambiar Estado Zona

2.1.2.2.3.9.1.4.85 Diagrama de Actividad: Caso de Uso Generar Reporte Lista Zona

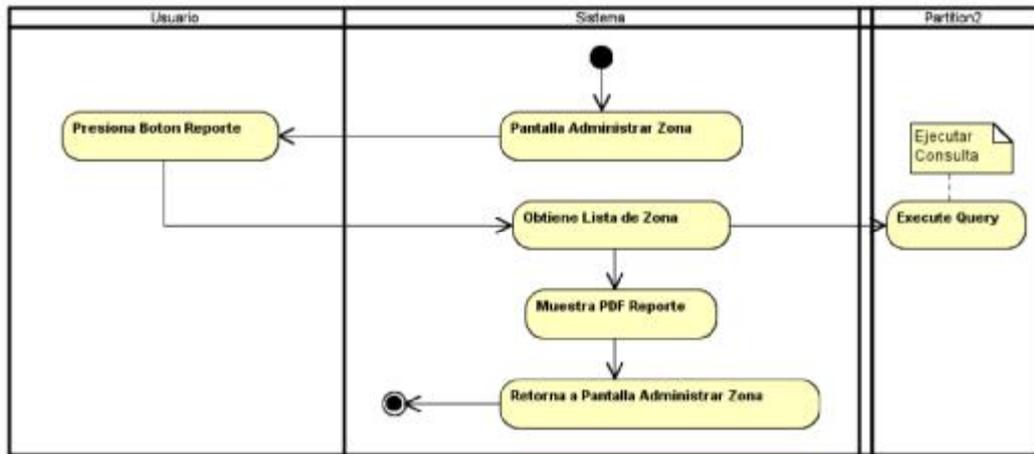


Figura 241. Diagrama de Actividad: CU Generar Reporte Lista Zona

2.1.2.2.3.9.1.4.86 Diagrama de Actividad: Caso de Uso Gestionar Asegurados por Tipo Seguro Afiliados

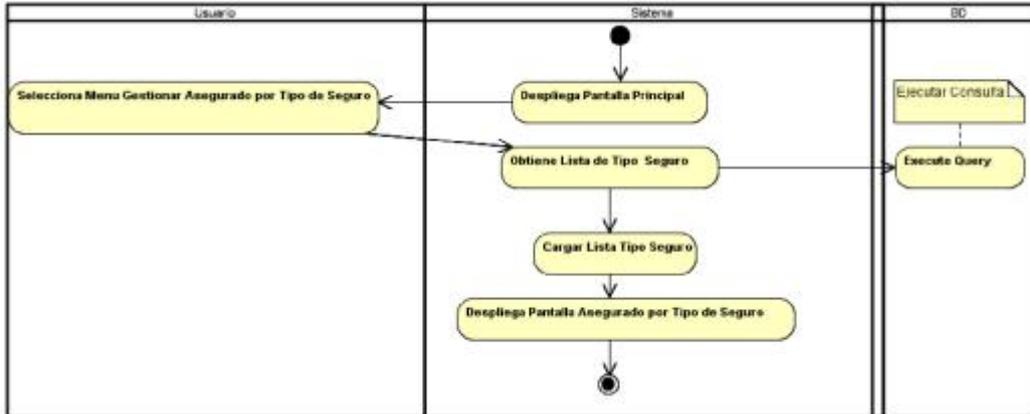


Figura 242. Diagrama de Actividad: CU Gestionar Asegurados por Tipo Seguro Afiliados

2.1.2.2.3.9.1.4.87 Diagrama de Actividad: Caso de Uso Gestionar Asegurados por Tipo Seguro

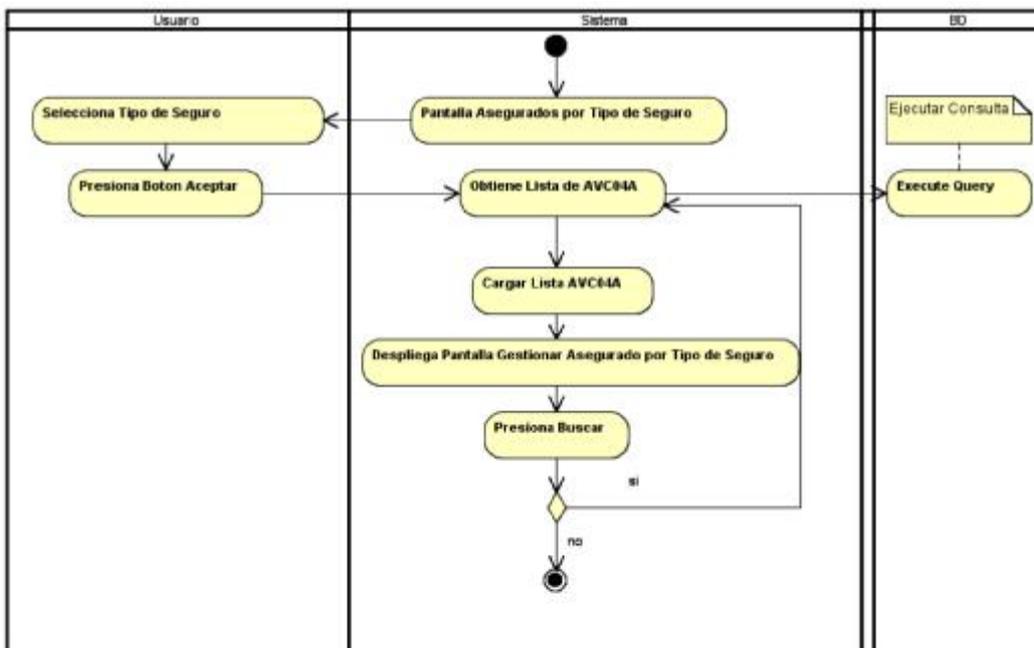


Figura 243. Diagrama de Actividad: CU Gestionar Asegurados por Tipo Seguro

2.1.2.2.3.9.1.4.88 Diagrama de Actividad: Caso de Uso Gestionar Personas No Aseguradas por Tipo de Seguro

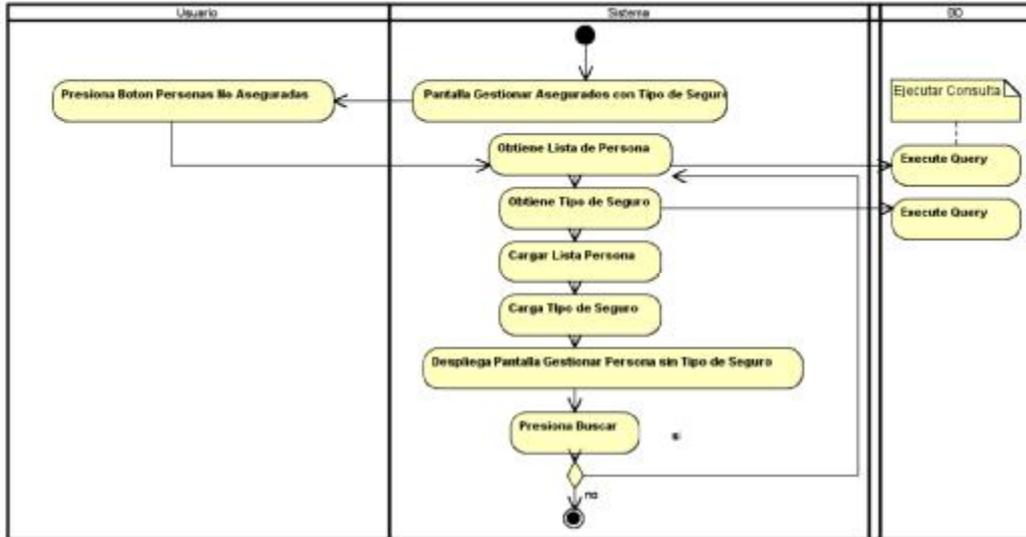


Figura 244. Diagrama de Actividad: CU Gestionar Personas No Aseguradas por Tipo de Seguro

2.1.2.2.3.9.1.4.89 Diagrama de Actividad: Caso de Uso Asegurar Persona por Tipo de Seguro

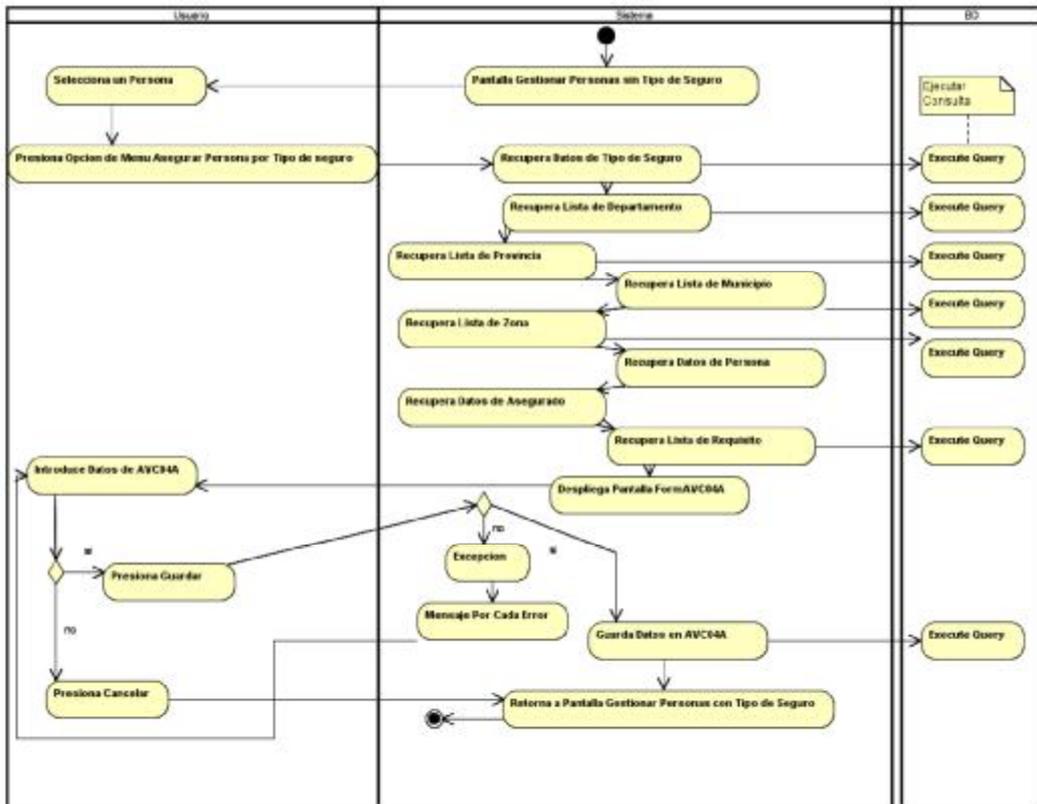


Figura 245. Diagrama de Actividad: CU Asegurar Persona por Tipo de Seguro

2.1.2.2.3.9.1.4.90 Diagrama de Actividad: Caso de Uso Adicionar Asegurado por Tipo Seguro

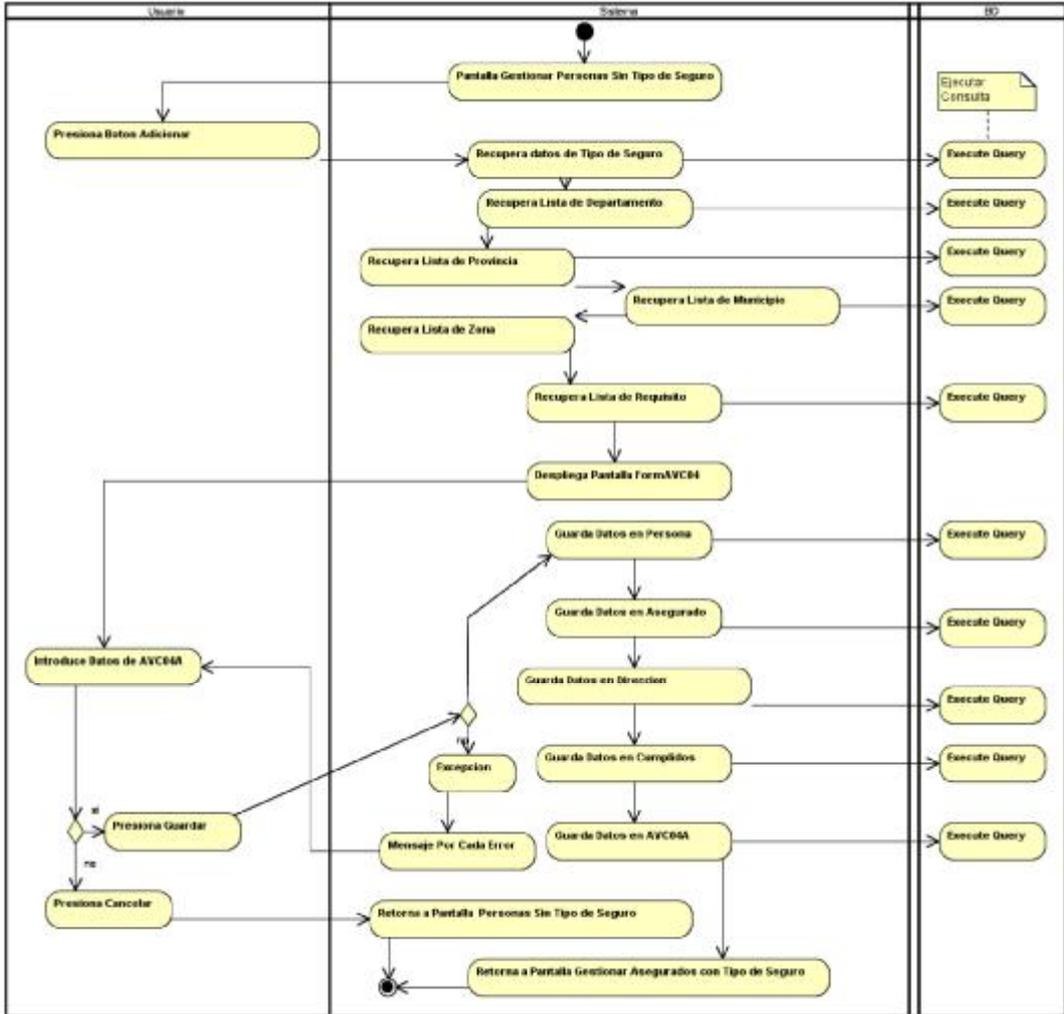


Figura 246. Diagrama de Actividad: CU Adicionar Asegurado por Tipo Seguro

2.1.2.2.3.9.1.4.91 Diagrama de Actividad: Caso de Uso Eliminar Asegurado por Tipo Seguro

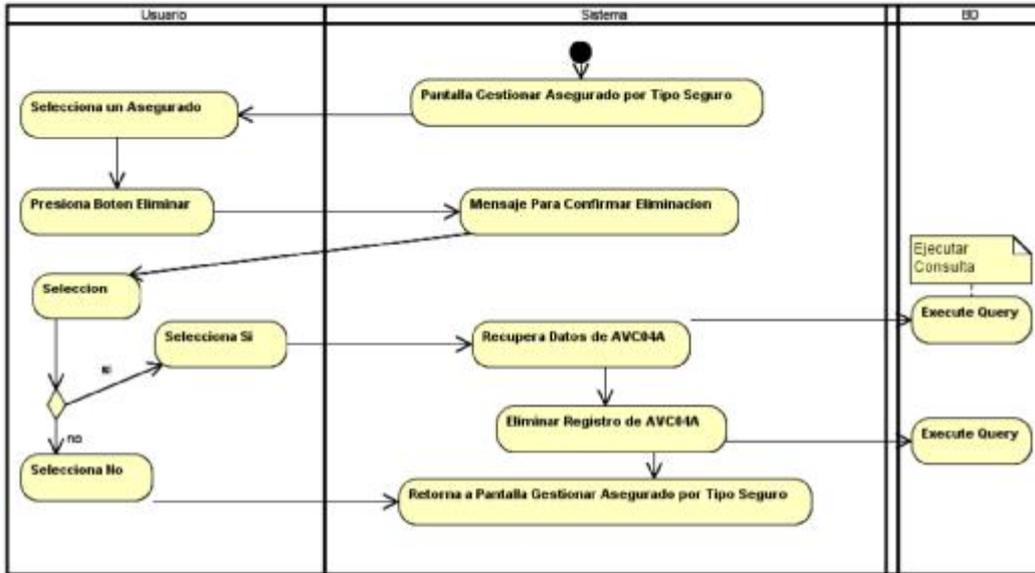


Figura 247. Diagrama de Actividad: CU Eliminar Asegurado por Tipo Seguro

2.1.2.2.3.9.1.4.92 Diagrama de Actividad: Caso de Uso Dar de Baja Asegurado por Tipo Seguro

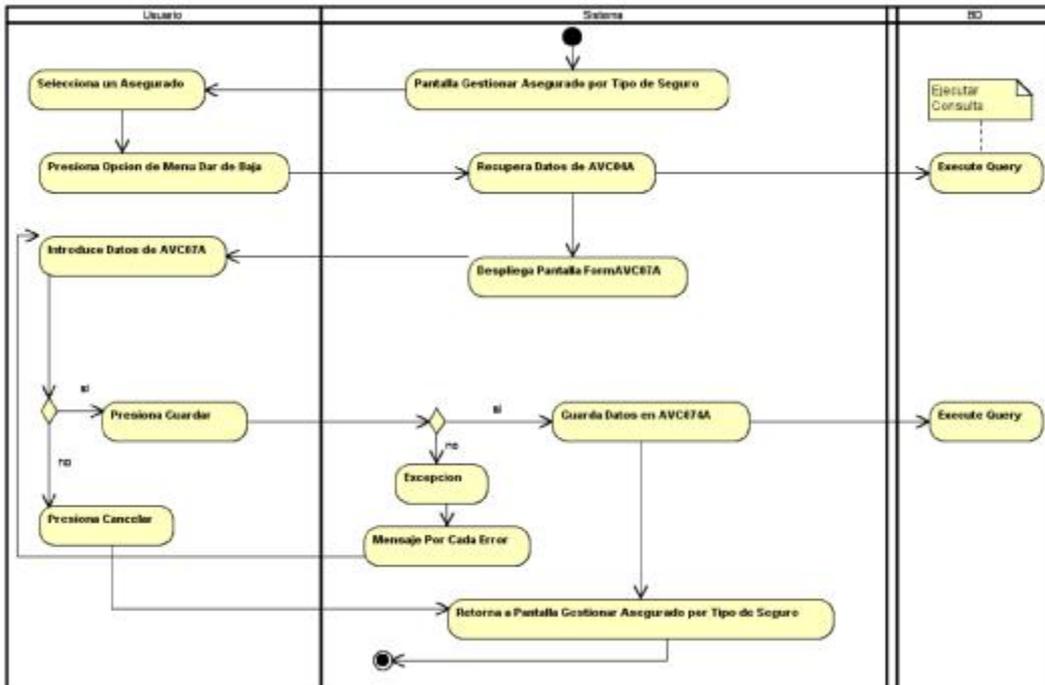


Figura 248. Diagrama de Actividad: CU Dar de Baja Asegurado por Tipo Seguro

2.1.2.2.3.9.1.4.93 Diagrama de Actividad: Caso de Uso Gestionar Empresas

Afiliadas

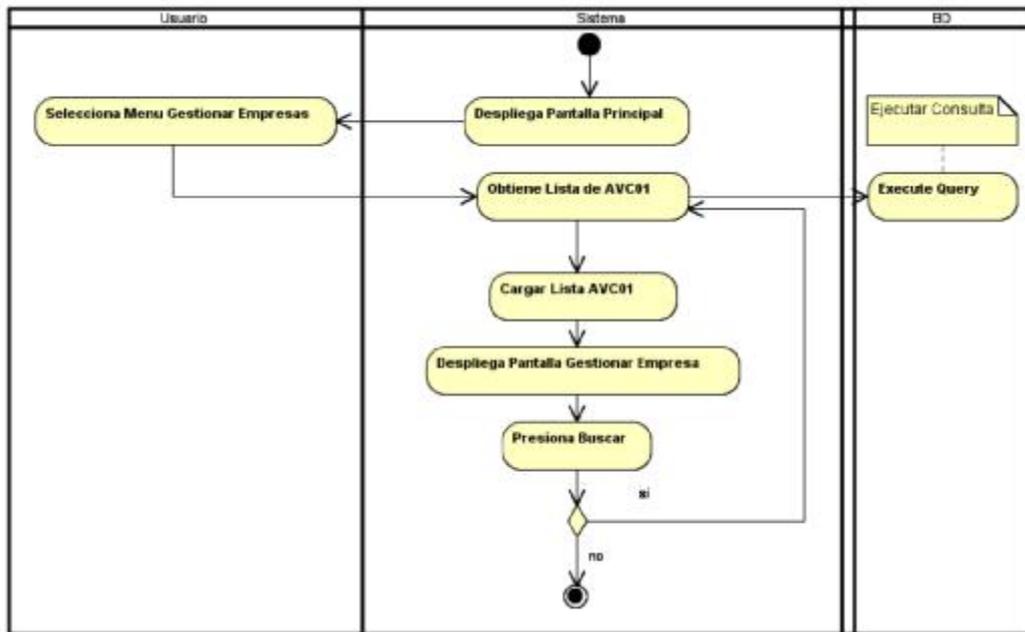


Figura 249. Diagrama de Actividad: CU Gestionar Empresas Afiliadas

2.1.2.2.3.9.1.4.94 Diagrama de Actividad: Caso de Uso Adicionar Empresa

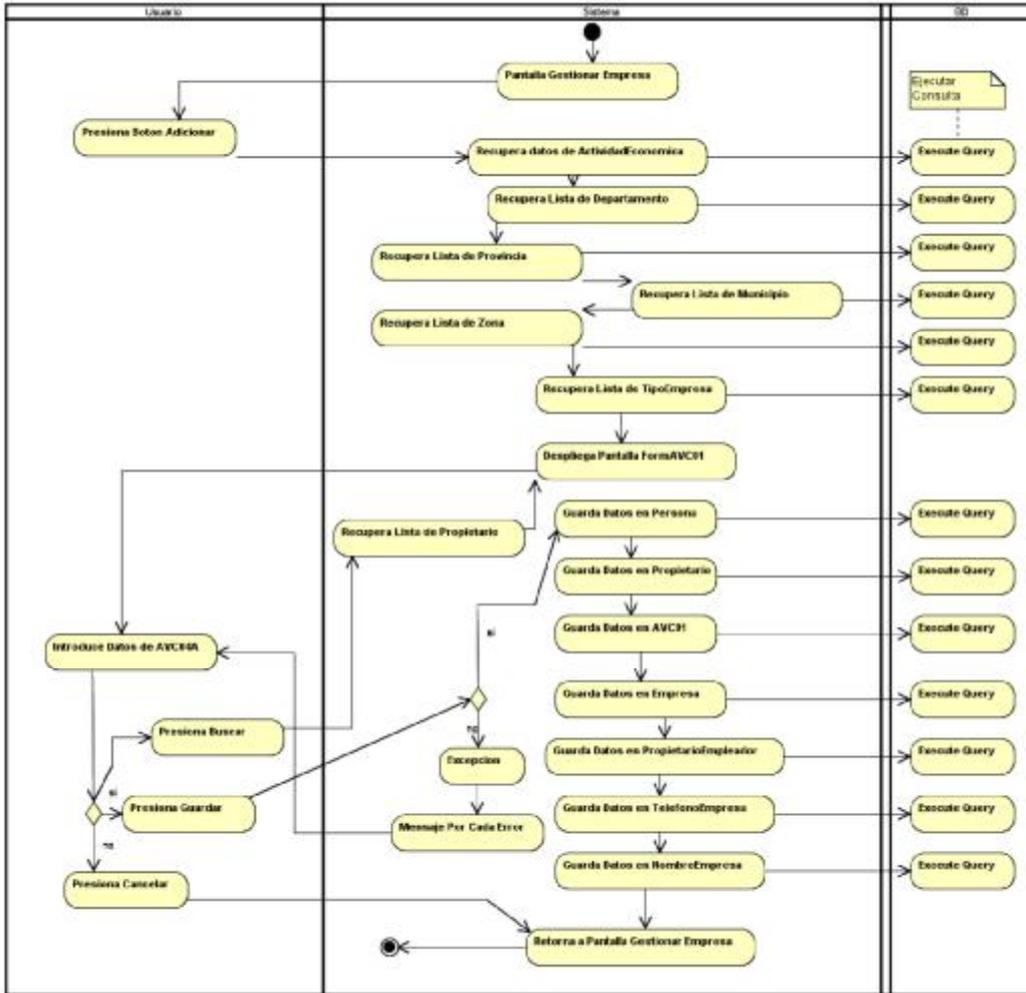


Figura 250. Diagrama de Actividad: CU Adicionar Empresa

2.1.2.2.3.9.1.4.95 Diagrama de Actividad: Caso de Uso Eliminar Empresa

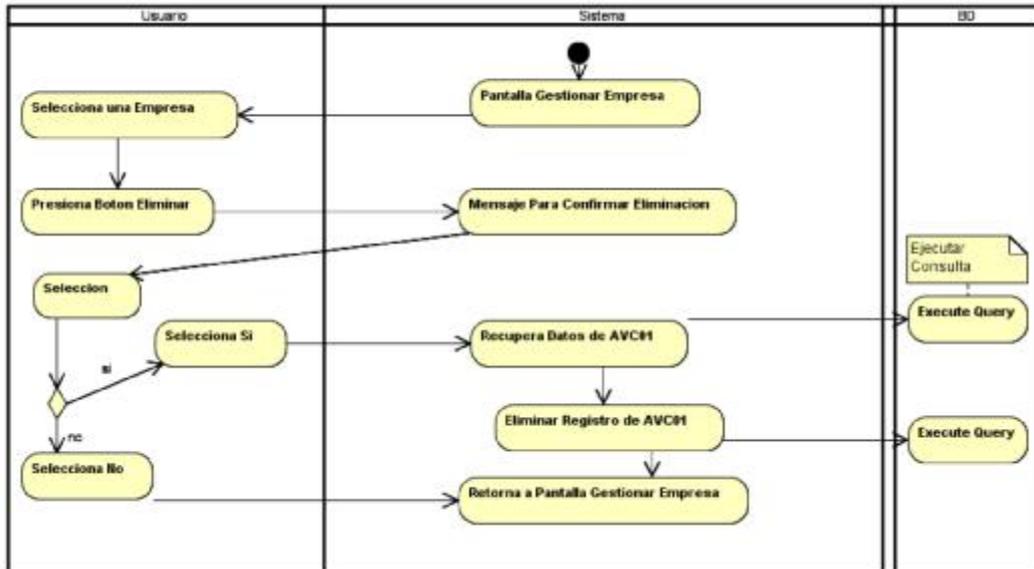


Figura 251. Diagrama de Actividad: CU Eliminar Empresa

2.1.2.2.3.9.1.4.96 Diagrama de Actividad: Caso de Uso Aviso Novedades Empresa

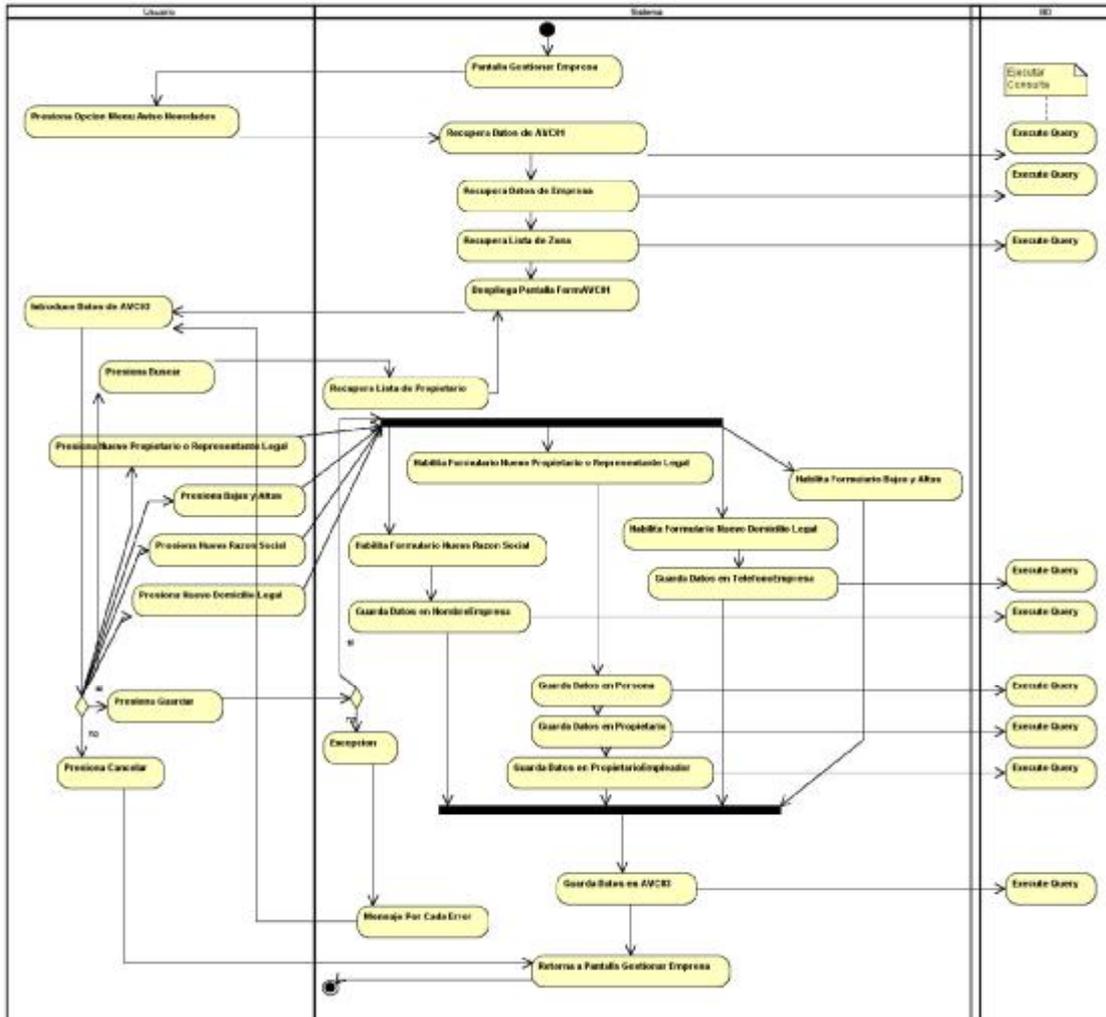


Figura 252. Diagrama de Actividad: CU Aviso Novedades Empresa

2.1.2.2.3.9.1.4.97 Diagrama de Actividad: Caso de Uso Dar de Alta Empresa por Norma

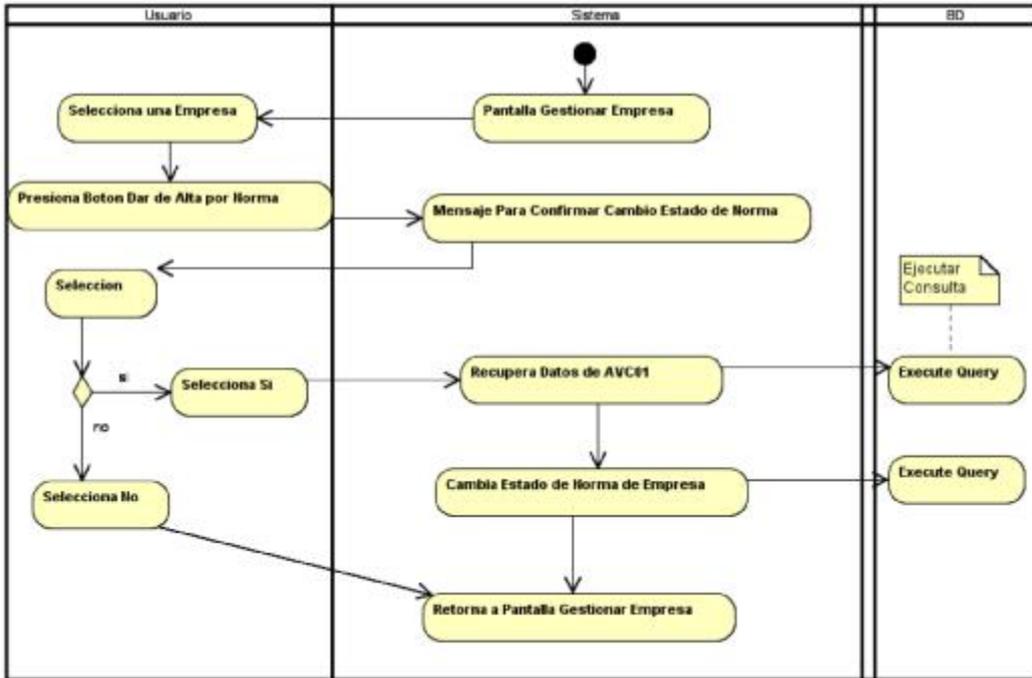


Figura 253. Diagrama de Actividad: CU Dar de Alta Empresa por Norma

2.1.2.2.3.9.1.4.98 Diagrama de Actividad: Caso de Uso Dar de Baja Empresa por Norma

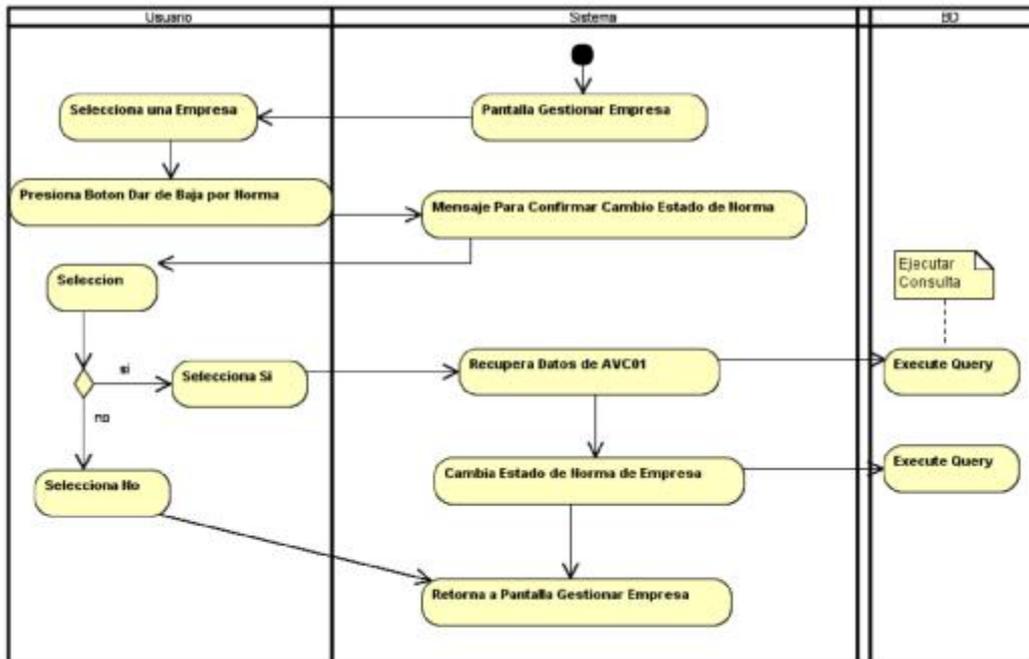


Figura 254. Diagrama de Actividad: CU Dar de Baja Empresa por Norma

2.1.2.2.3.9.1.4.99 Diagrama de Actividad: Caso de Uso Gestionar Asegurados Afiliados

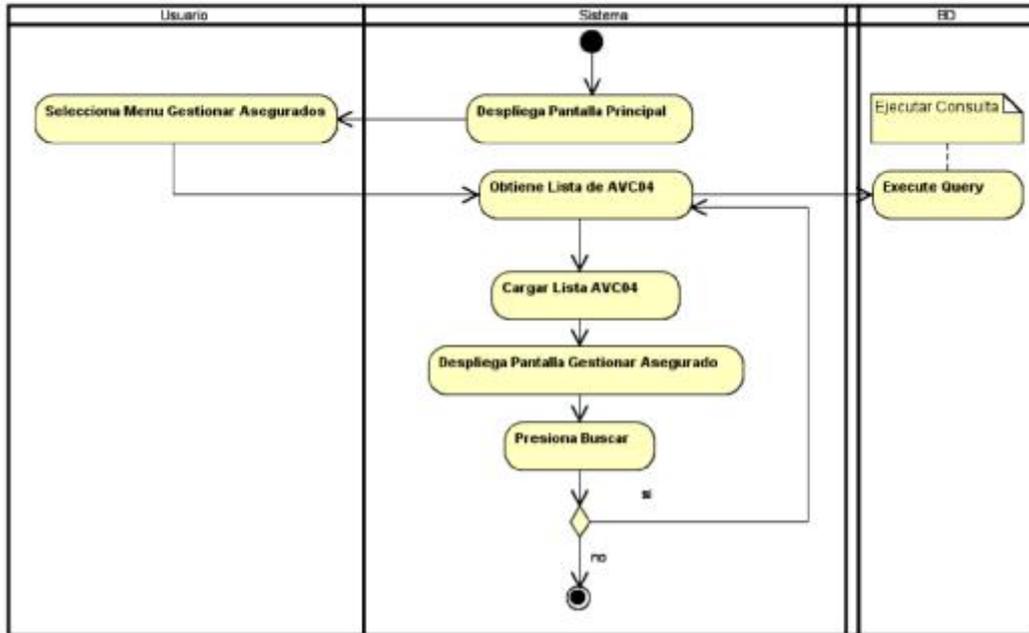


Figura 255. Diagrama de Actividad: CU Gestionar Asegurados Afiliados

2.1.2.2.3.9.1.4.100 Diagrama de Actividad: Caso de Uso Reingreso Asegurado

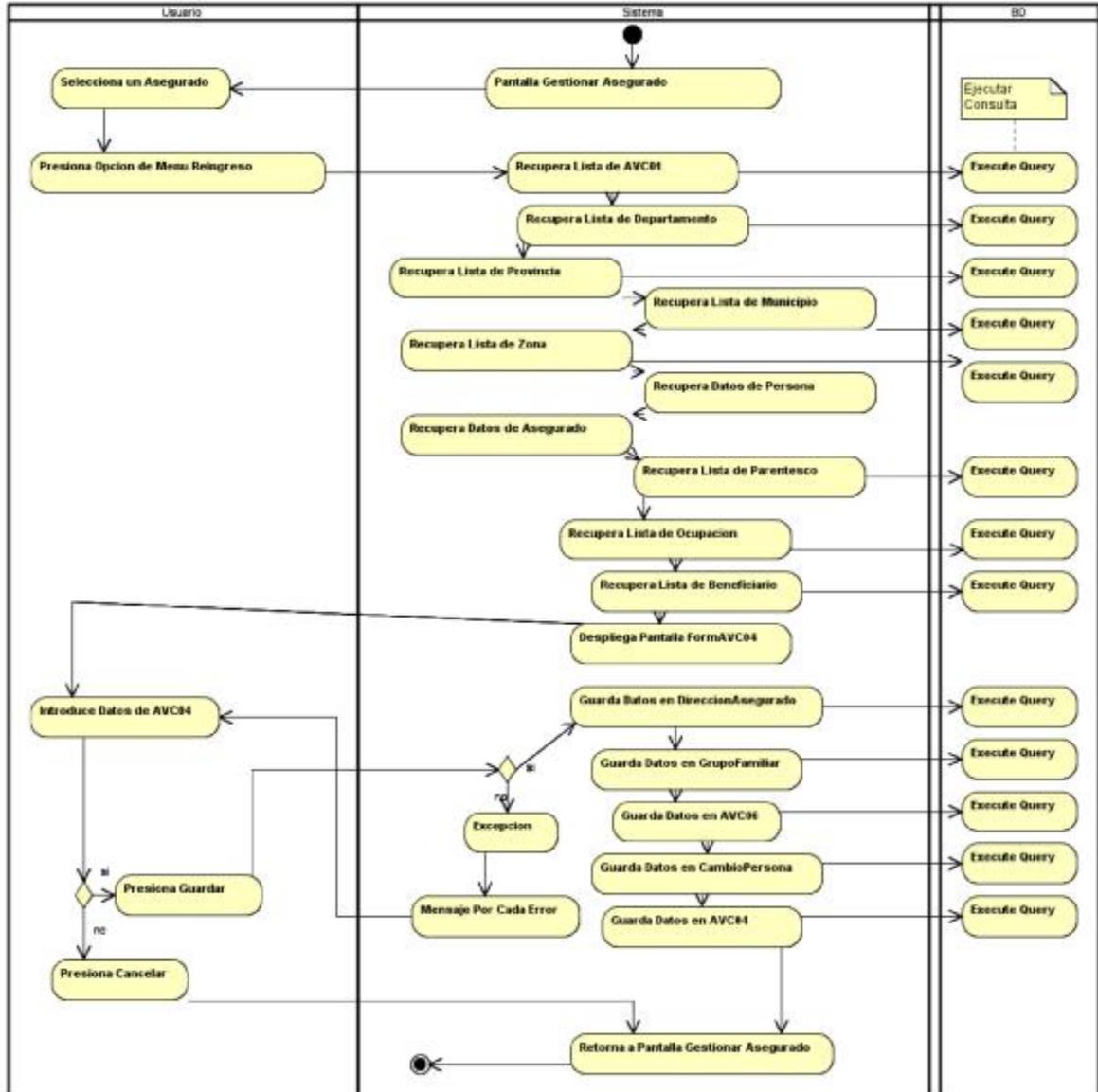


Figura 256. Diagrama de Actividad: CU Reingreso Asegurado

2.1.2.2.3.9.1.4.101

Diagrama de Actividad: Caso de Uso Eliminar Asegurado

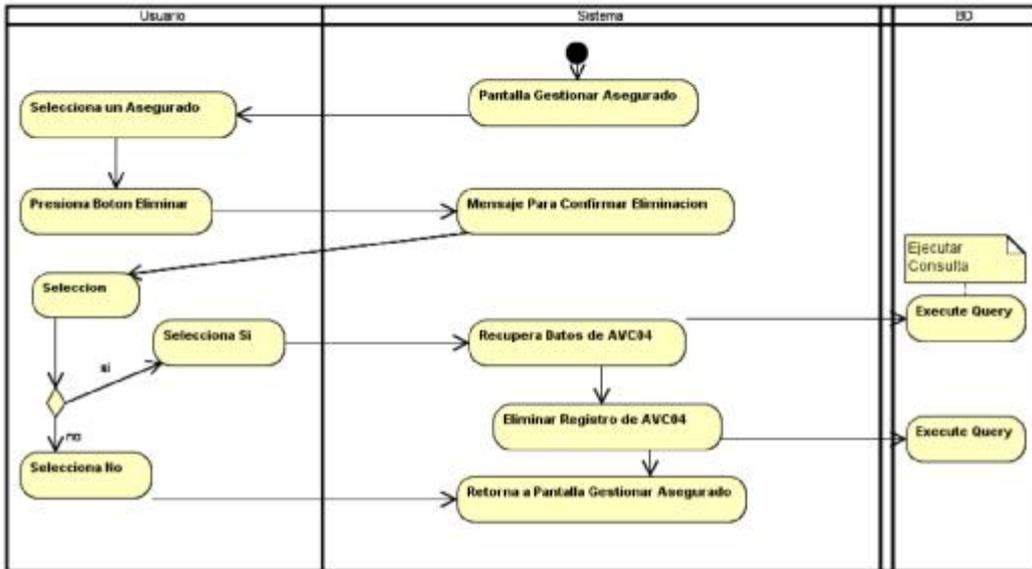


Figura 257. Diagrama de Actividad: CU Eliminar Asegurado

2.1.2.2.3.9.1.4.102

Diagrama de Actividad: Caso de Uso Dar de Baja Asegurado

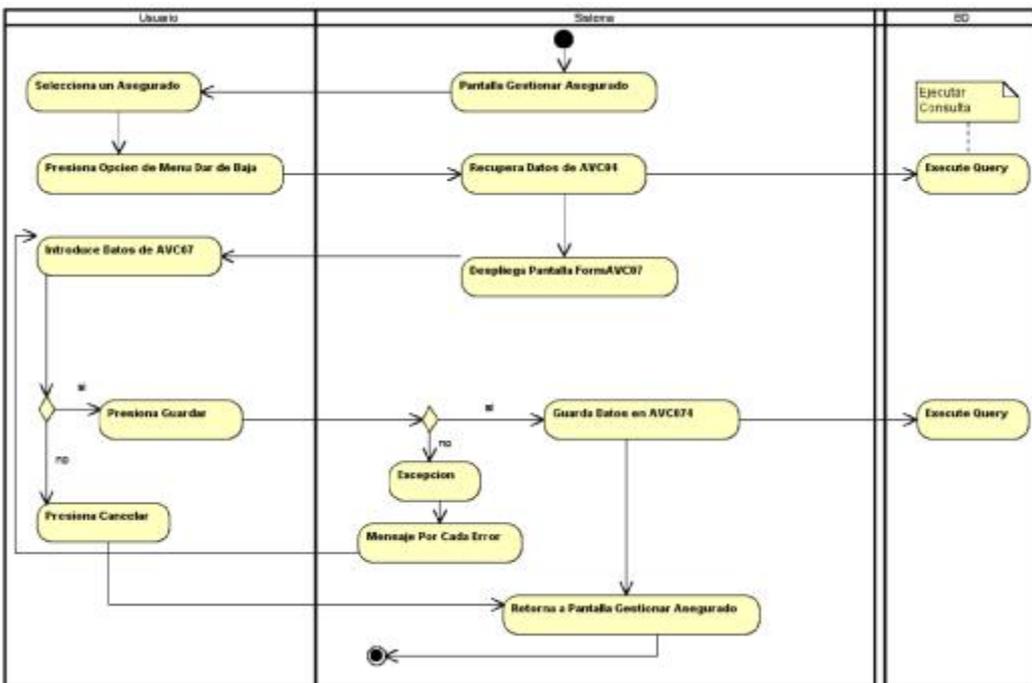


Figura 258. Diagrama de Actividad: CU Dar de Baja Asegurado

2.1.2.2.3.9.1.4.103 Diagrama de Actividad: Caso de Uso Dar de Alta Asegurado por Norma

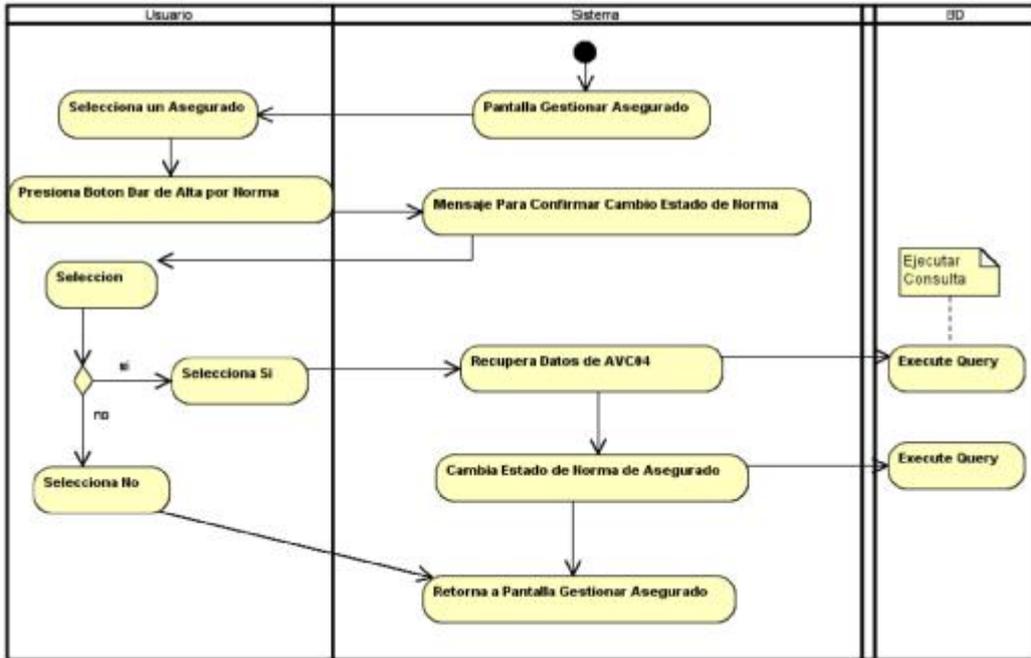


Figura 259. Diagrama de Actividad: CU Dar de Alta Asegurado por Norma

2.1.2.2.3.9.1.4.104 Diagrama de Actividad: Caso de Uso Dar de Baja Asegurado por Norma

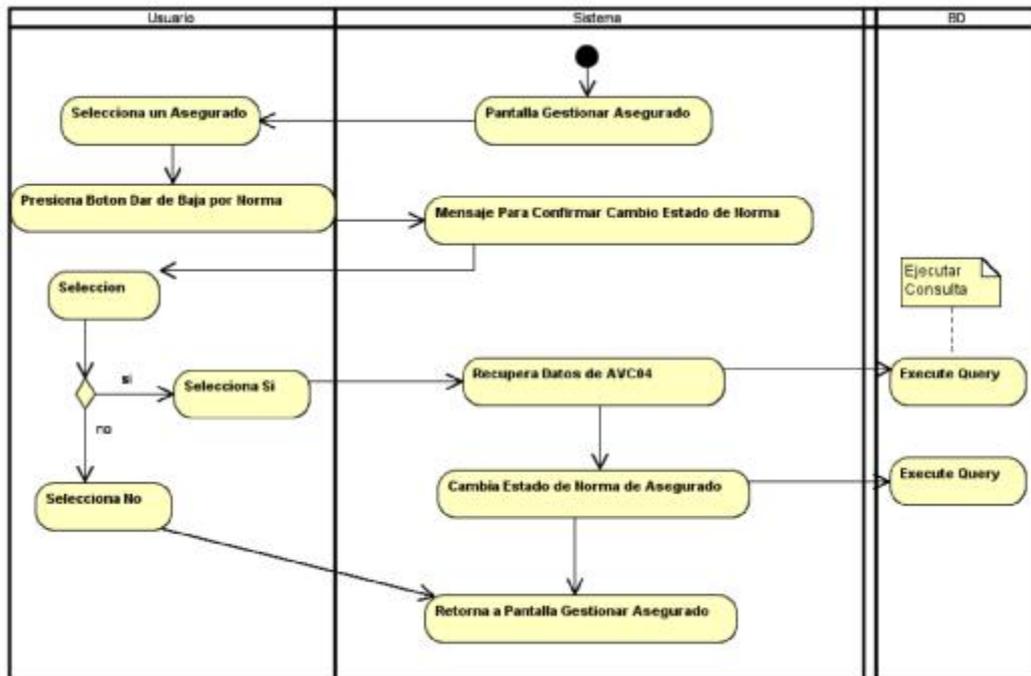


Figura 260. Diagrama de Actividad: CU Dar de Baja Asegurado por Norma

2.1.2.2.3.9.1.4.105 Diagrama de Actividad: Caso de Uso Gestionar Personas No Aseguradas

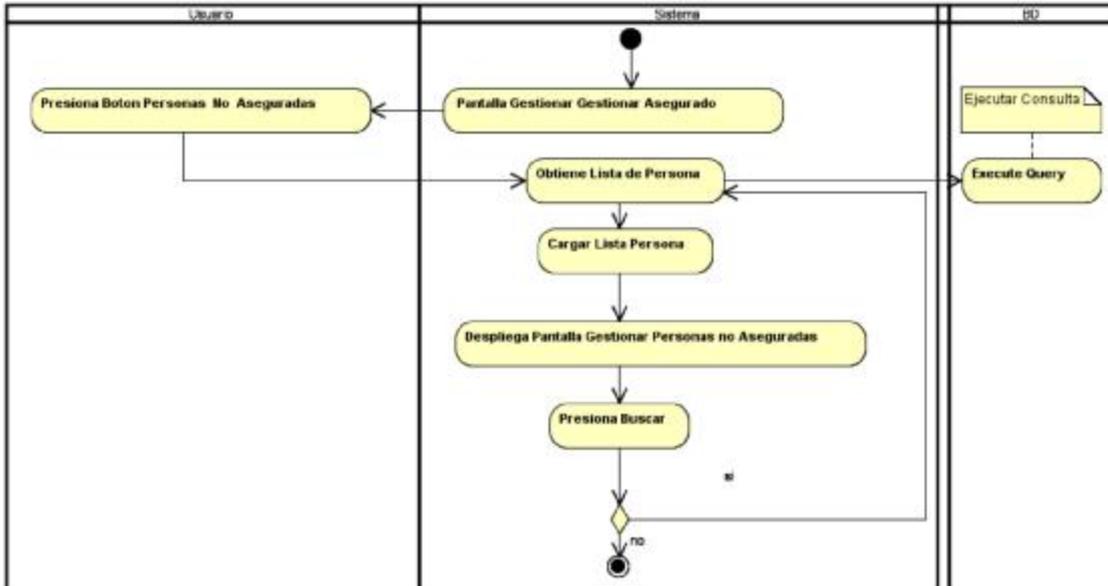


Figura 261. Diagrama de Actividad: CU Gestionar Personas No Aseguradas

2.1.2.2.3.9.1.4.106 Diagrama de Actividad: Caso de Uso Adicionar Asegurado

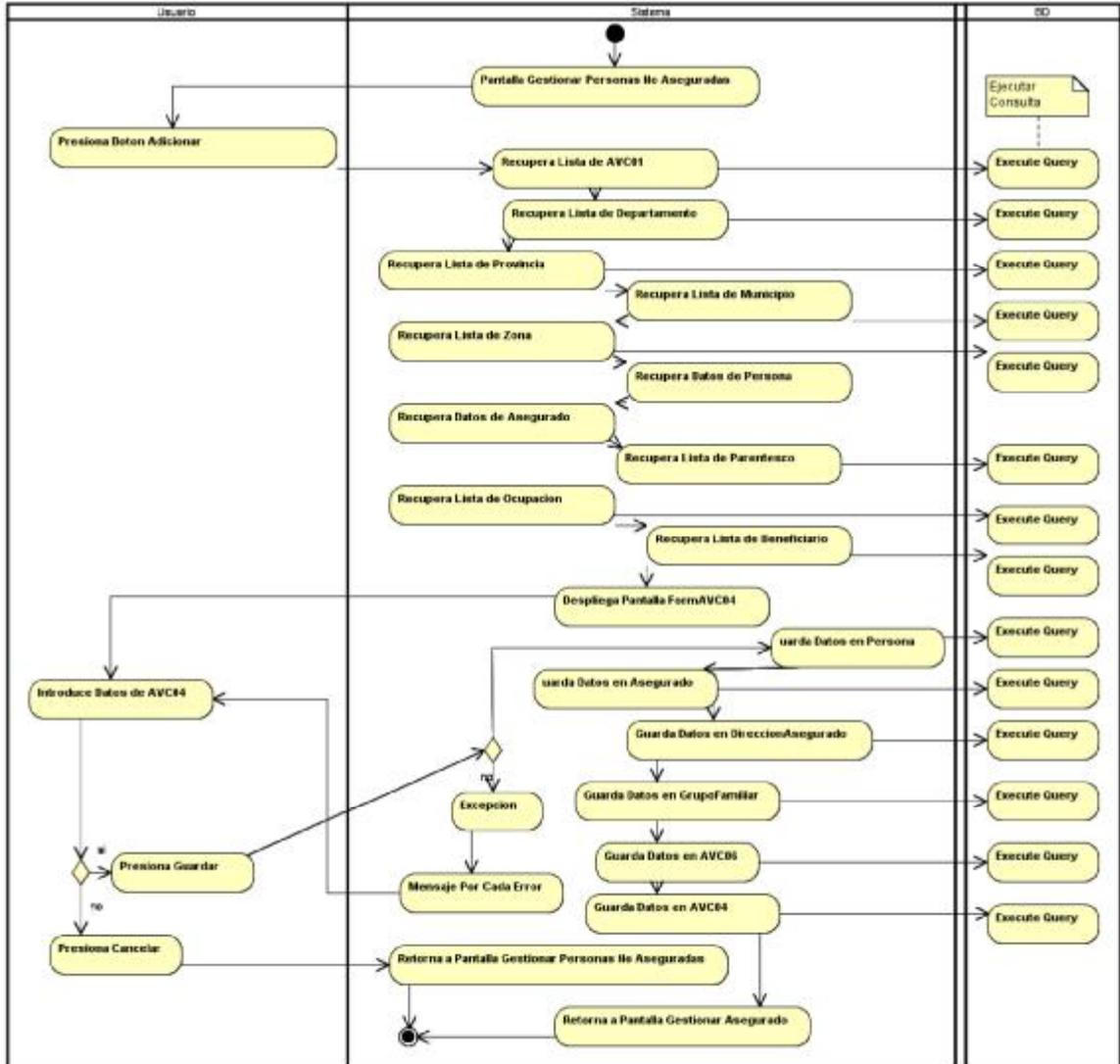


Figura 262. Diagrama de Actividad: CU Adicionar Asegurado

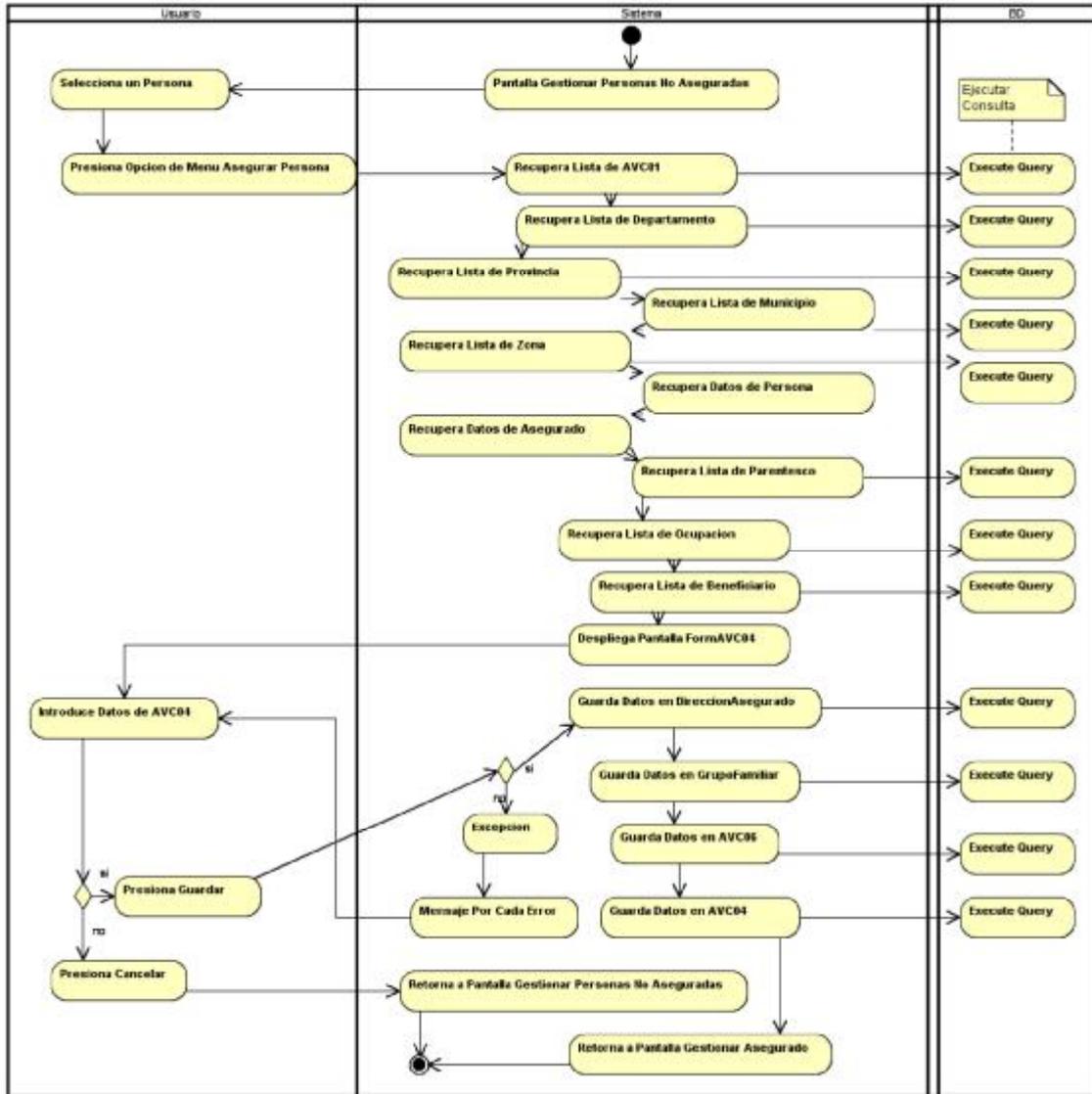


Figura 263. Diagrama de Actividad: CU Asegurar Persona

2.1.2.2.3.9.1.4.108 Diagrama de Actividad: Caso de Uso Gestionar Emitir Autorización Médica Externa

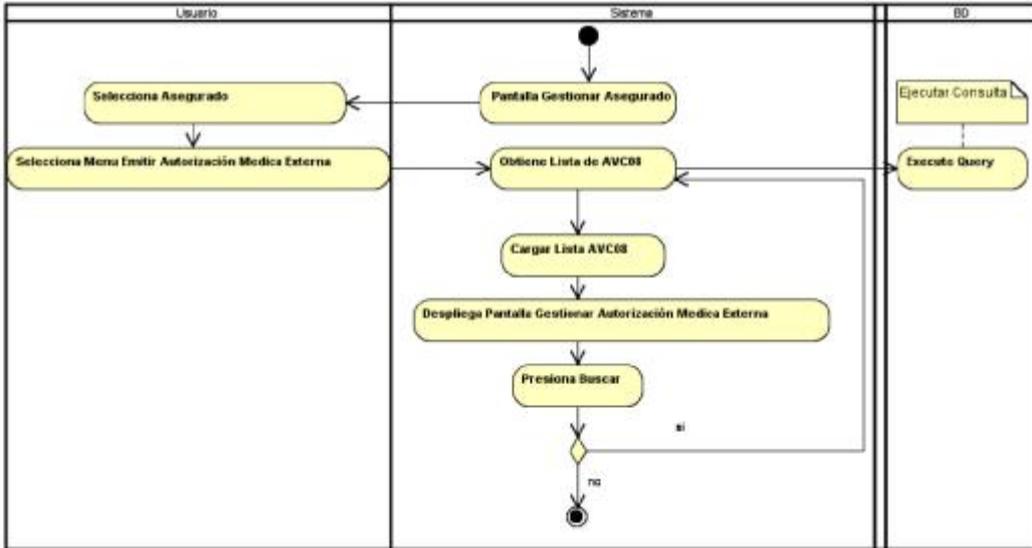


Figura 264. Diagrama de Actividad: CU Gestionar Emitir Autorización Médica Externa

2.1.2.2.3.9.1.4.109 Diagrama de Actividad: Caso de Uso Adicionar Autorización Médica Externa

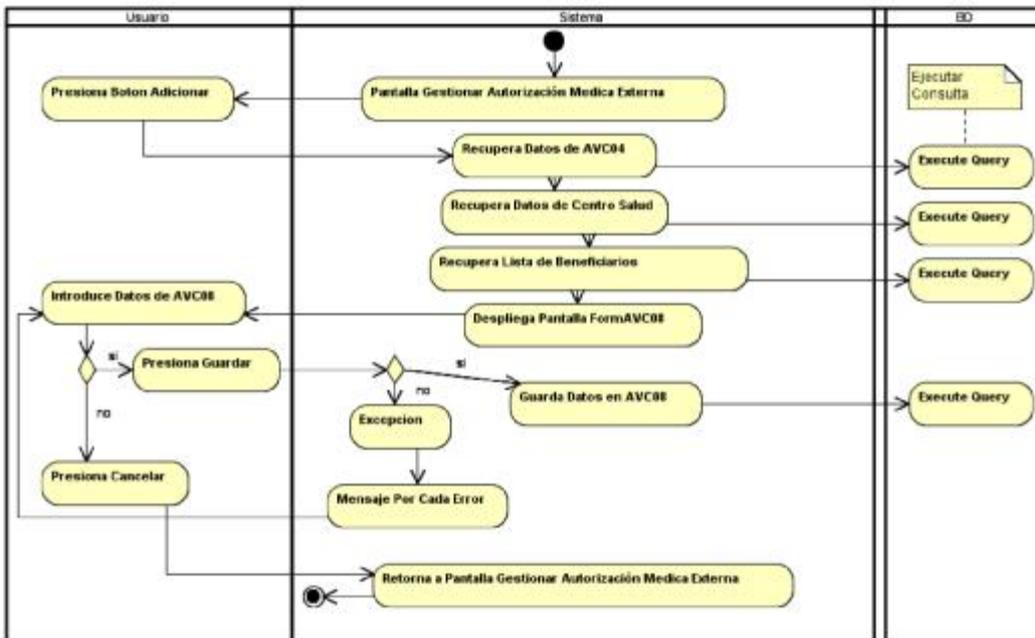


Figura 265. Diagrama de Actividad: CU Adicionar Autorización Médica Externa

2.1.2.2.3.9.1.4.110 Diagrama de Actividad: Caso de Uso Modificar Autorización Médica Externa

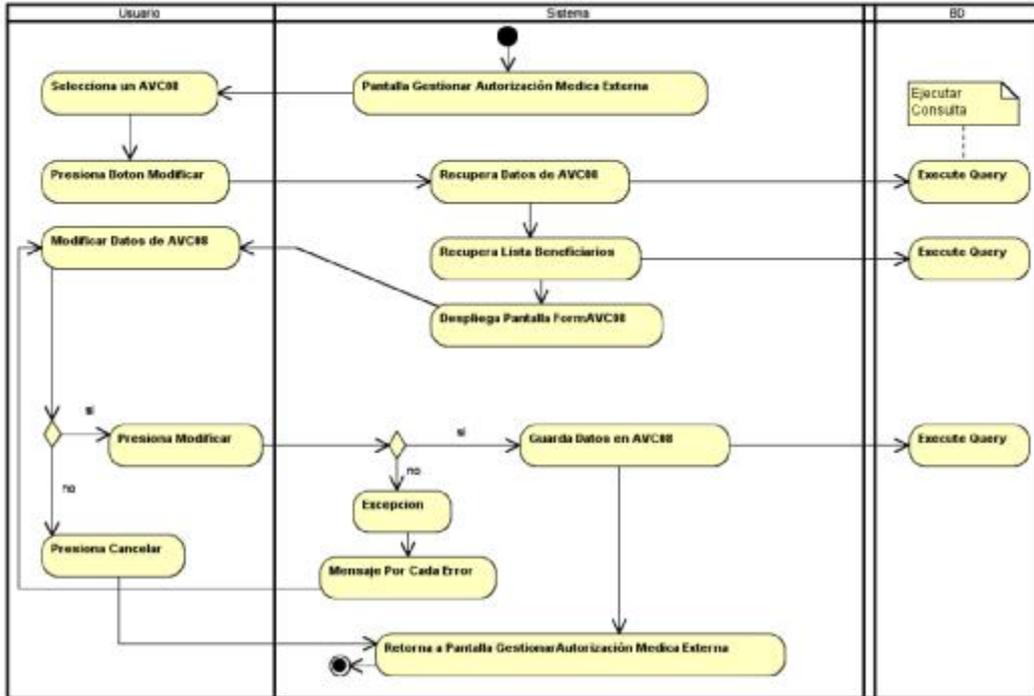


Figura 266. Diagrama de Actividad: CU Modificar Autorización Médica Externa

2.1.2.2.3.9.1.4.111 Diagrama de Actividad: Caso de Uso Eliminar Autorización Médica Externa

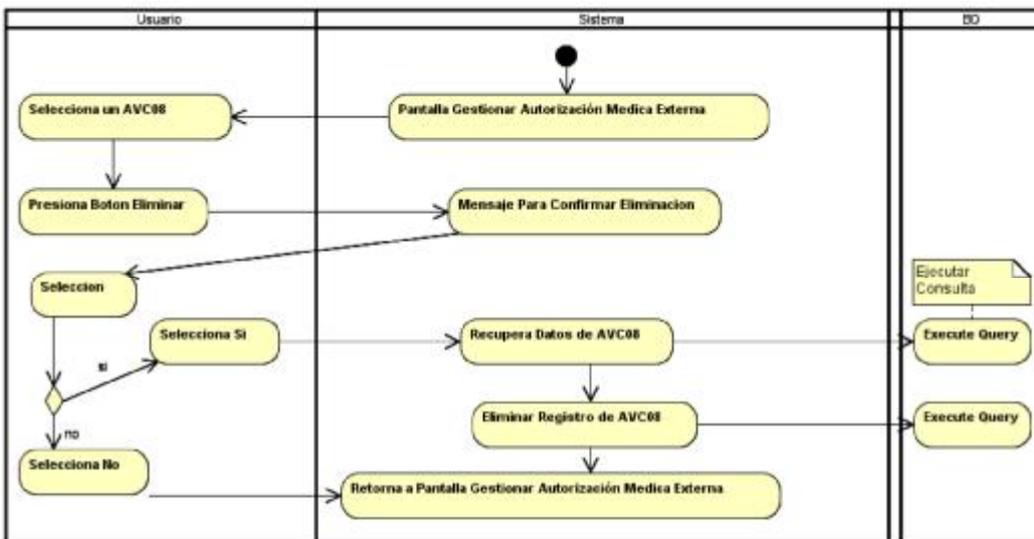


Figura 267. Diagrama de Actividad: CU Eliminar Autorización Médica Externa

2.1.2.2.3.9.1.4.112 Diagrama de Actividad: Caso de Uso Cambiar Estado Autorización Médica Externa

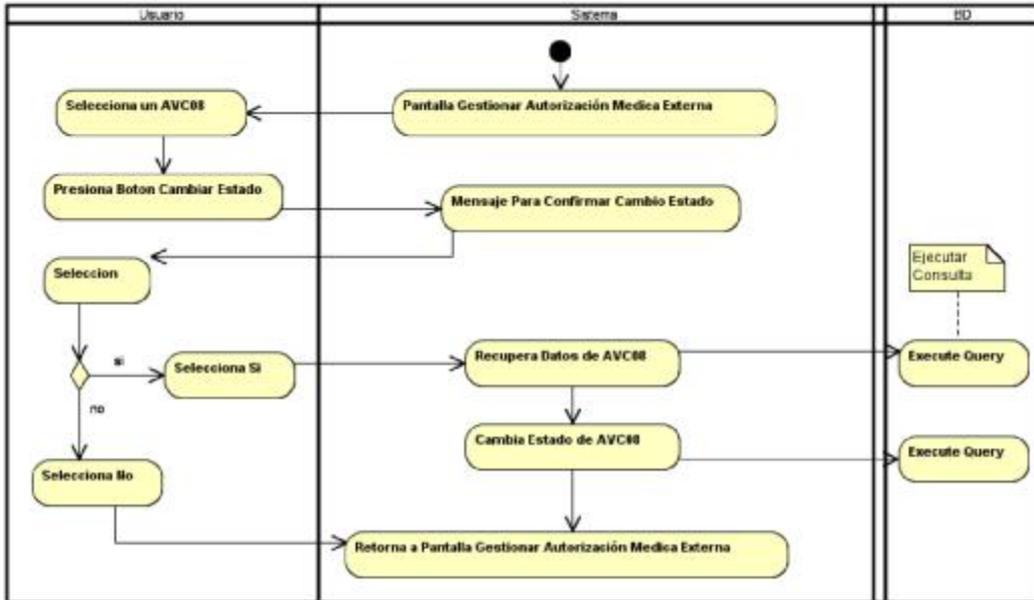


Figura 268. Diagrama de Actividad: CU Cambiar Estado Autorización Médica Externa

2.1.2.2.3.9.1.4.113 Diagrama de Actividad: Caso de Uso Gestionar Emitir Certificado de Incapacidad Temporal

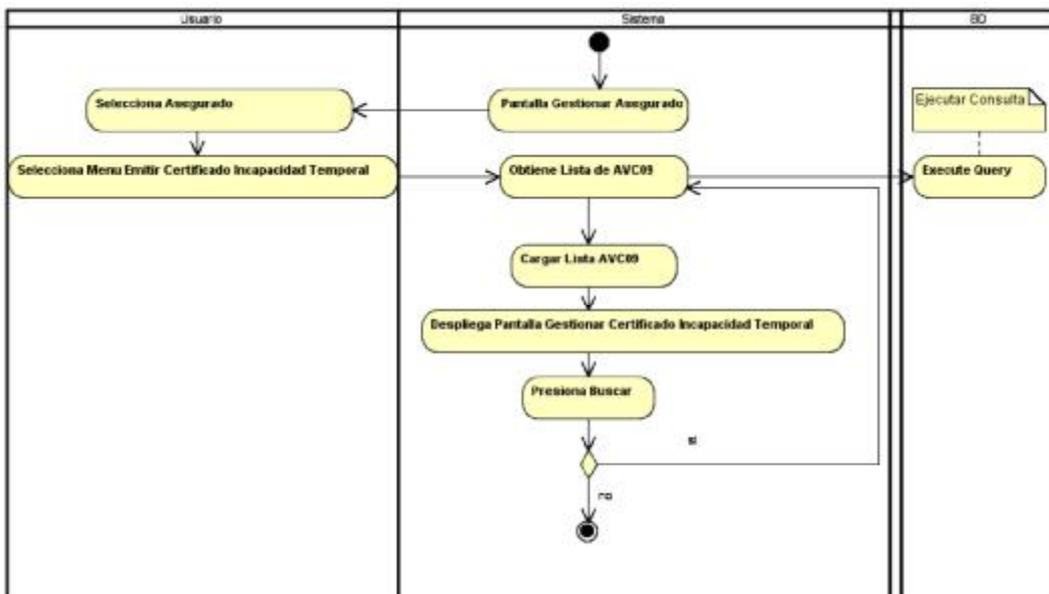


Figura 269. Diagrama de Actividad: CU Gestionar Emitir Certificado de Incapacidad Temporal

2.1.2.2.3.9.1.4.114 Diagrama de Actividad: Caso de Uso Adicionar Certificado de Incapacidad Temporal

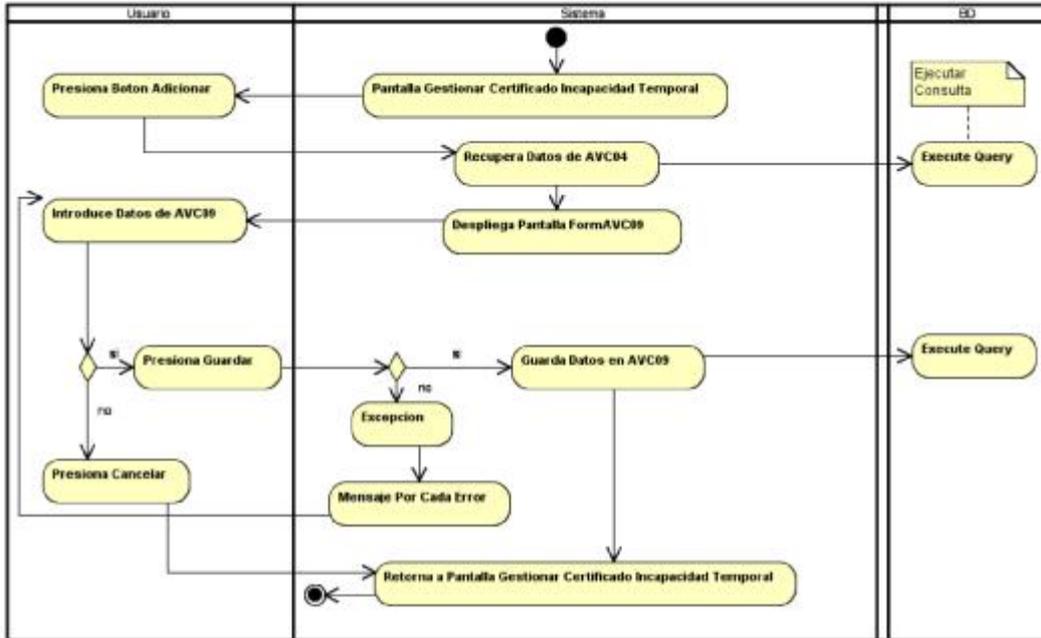


Figura 270. Diagrama de Actividad: CU Adicionar Certificado de Incapacidad Temporal

2.1.2.2.3.9.1.4.115 Diagrama de Actividad: Caso de Uso Modificar Certificado de Incapacidad Temporal

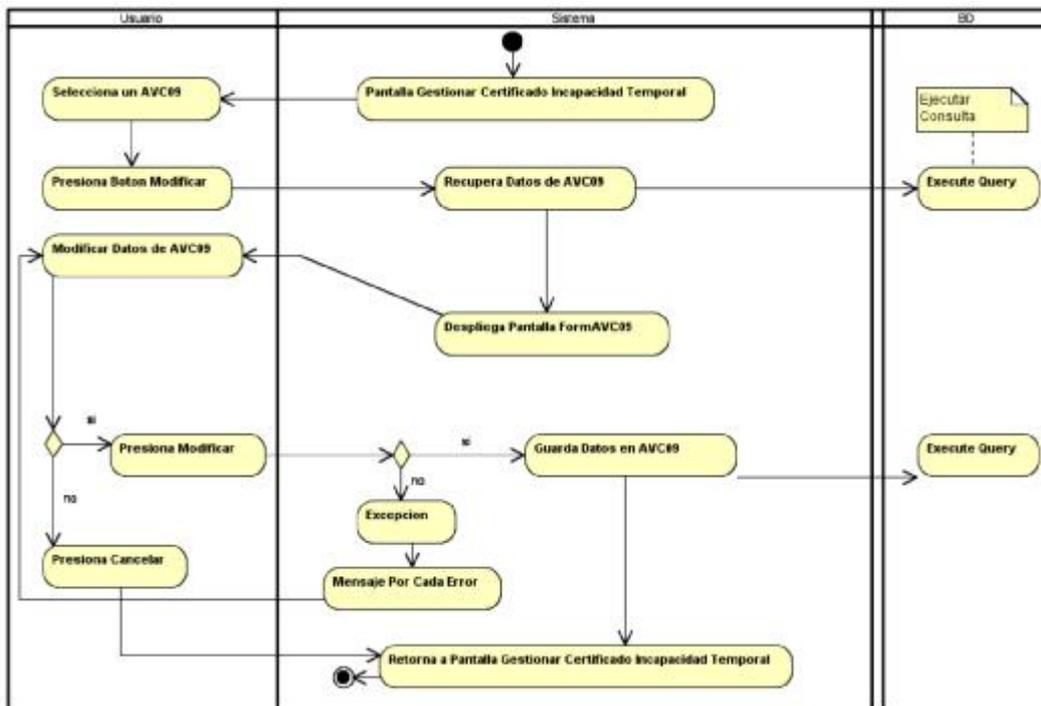


Figura 271. Diagrama de Actividad: CU Modificar Certificado de Incapacidad Temporal

2.1.2.2.3.9.1.4.116 Diagrama de Actividad: Caso de Uso Eliminar Certificado de Incapacidad Temporal

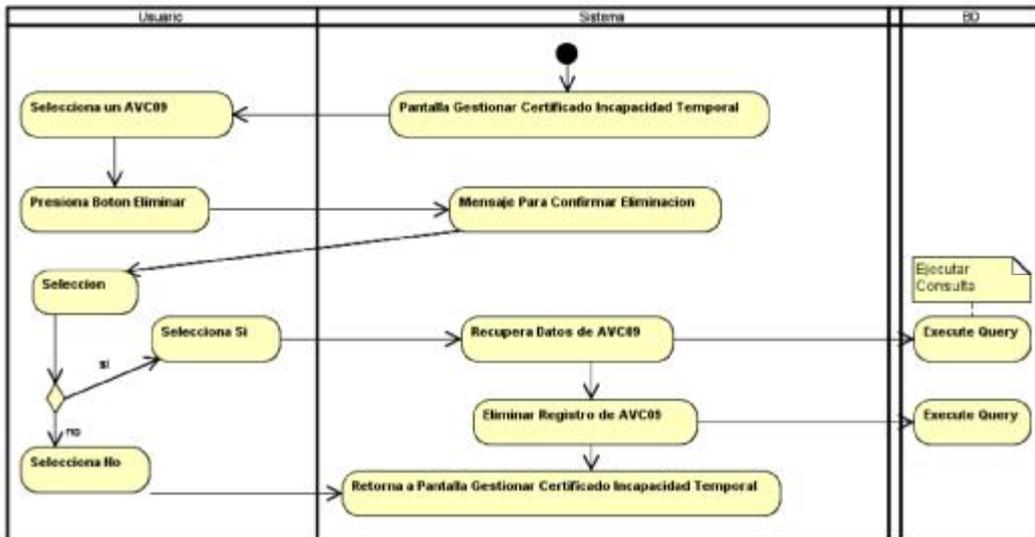


Figura 272. Diagrama de Actividad: CU Eliminar Certificado de Incapacidad Temporal

2.1.2.2.3.9.1.4.117 Diagrama de Actividad: Caso de Uso Cambiar Estado Certificado de Incapacidad Temporal

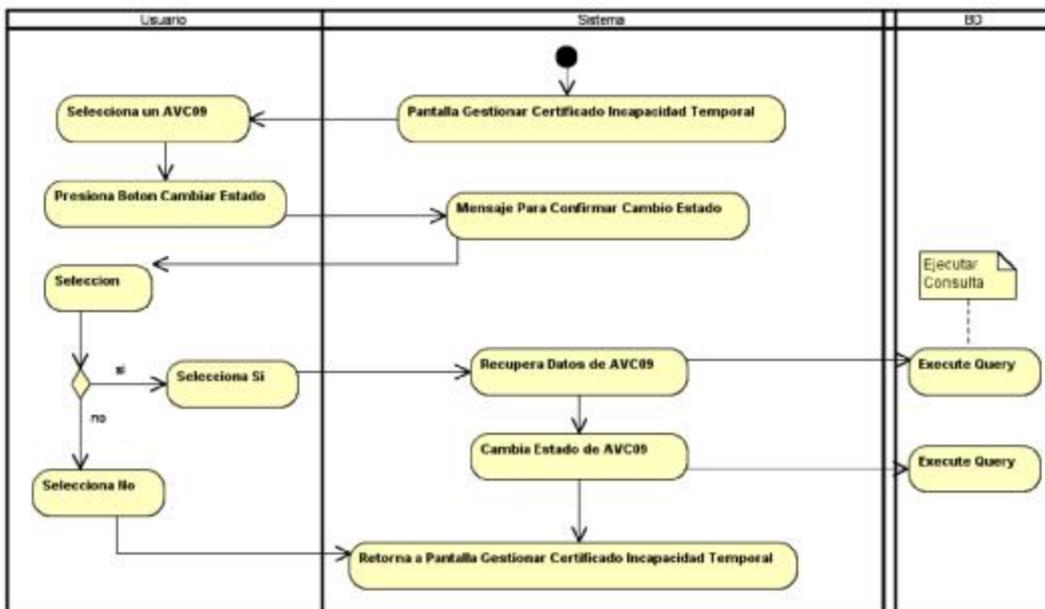


Figura 273. Diagrama de Actividad: CU Cambiar Estado Certificado de Incapacidad Temporal

2.1.2.2.3.9.1.4.118 Diagrama de Actividad: Caso de Uso Gestionar Altas y Bajas Beneficiarios

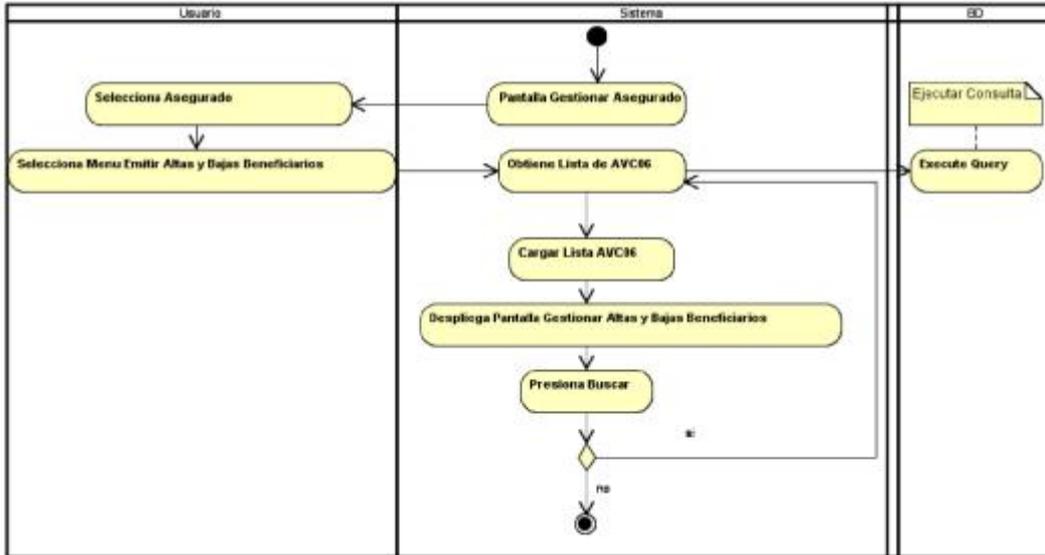


Figura 274. Diagrama de Actividad: CU Gestionar Altas y Bajas Beneficiarios

2.1.2.2.3.9.1.4.119 Diagrama de Actividad: Caso de Uso Adicionar Altas y Bajas Beneficiarios

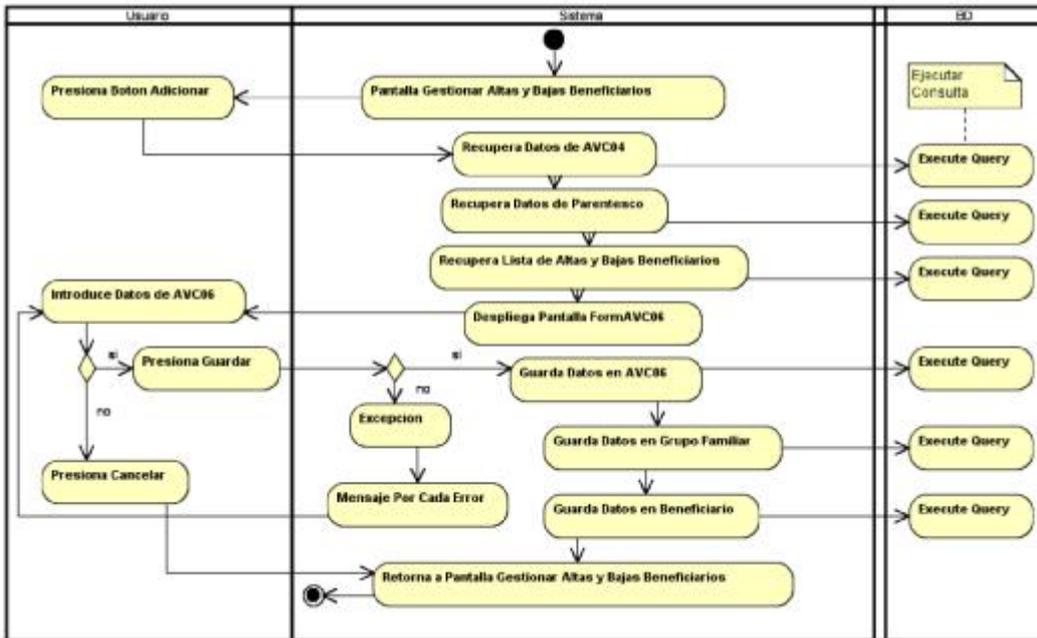


Figura 275. Diagrama de Actividad: CU Adicionar Altas y Bajas Beneficiarios

2.1.2.2.3.9.1.4.120 Diagrama de Actividad: Caso de Uso Eliminar Altas y Bajas Beneficiarios

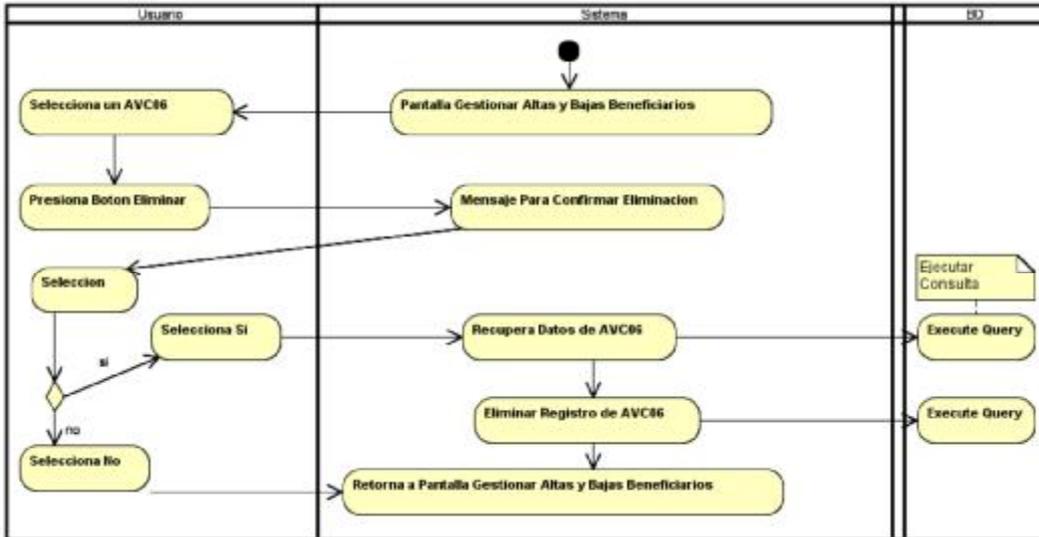


Figura 276. Diagrama de Actividad: CU Eliminar Altas y Bajas Beneficiarios

2.1.2.2.3.9.1.4.121 Diagrama de Actividad: Caso de Uso Cambiar Estado Altas y Bajas Beneficiarios

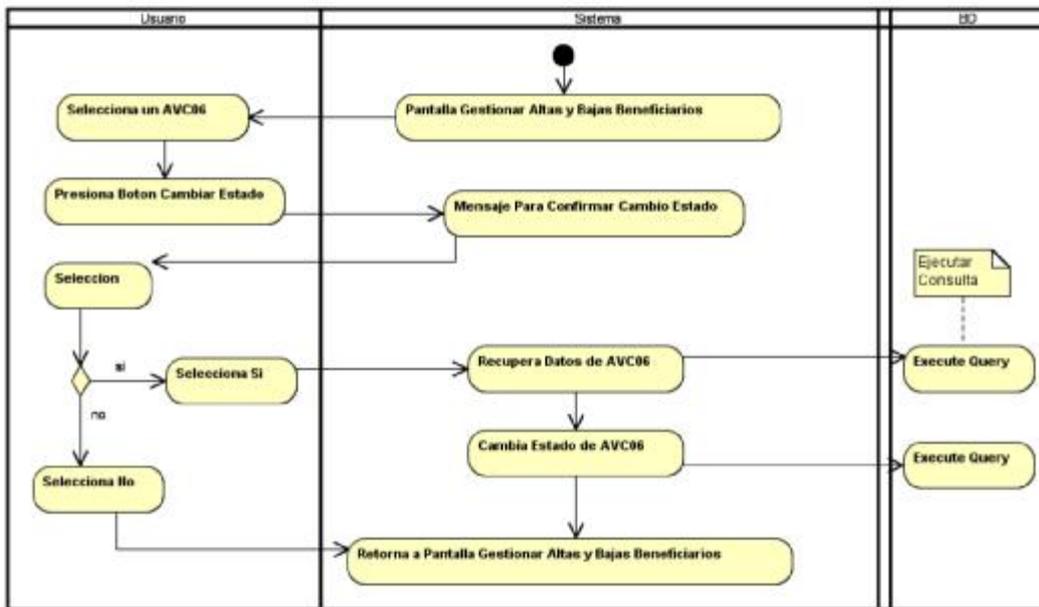


Figura 277. Diagrama de Actividad: CU Cambiar Estado Altas y Bajas Beneficiarios

2.1.2.2.3.9.1.4.122 Diagrama de Actividad: Caso de Uso Gestionar Beneficiarios Afiliados

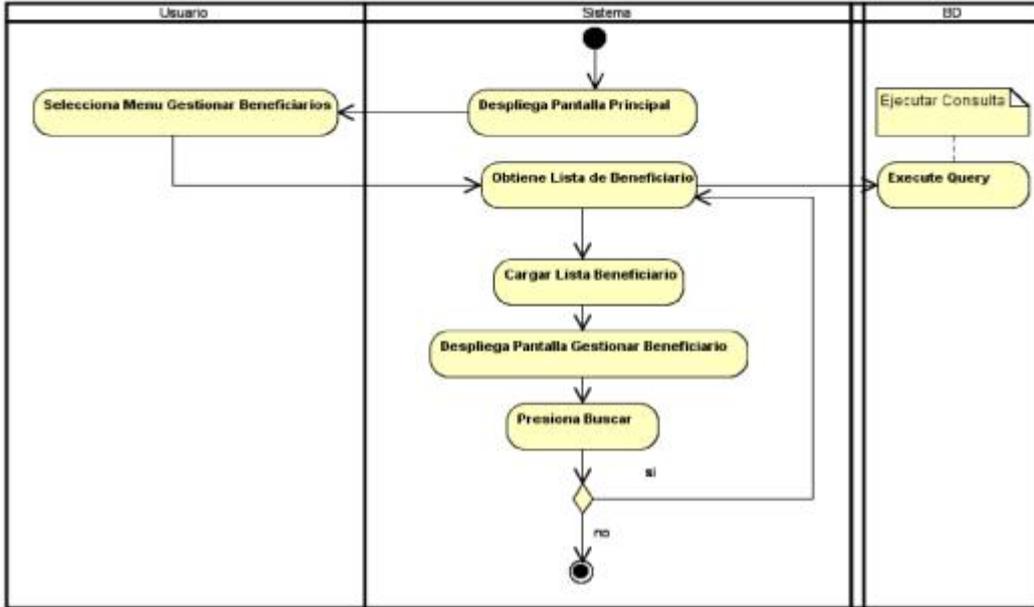


Figura 278. Diagrama de Actividad: CU Gestionar Beneficiarios Afiliados

2.1.2.2.3.9.1.4.123 Diagrama de Actividad: Caso de Uso Dar de Alta Beneficiario por Norma

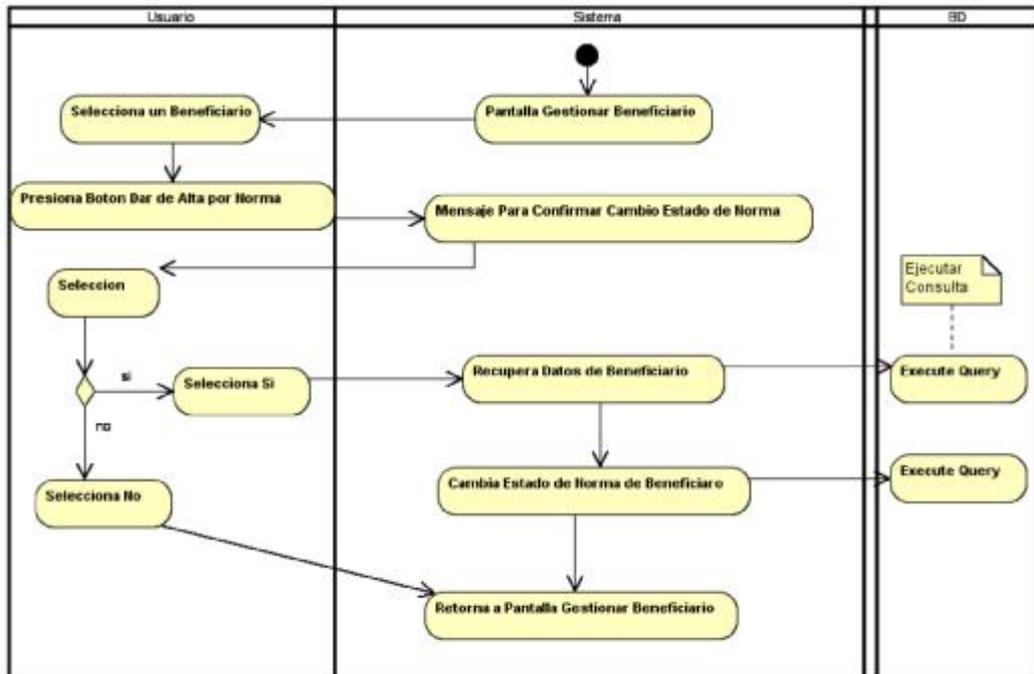


Figura 279. Diagrama de Actividad: CU Dar de Alta Beneficiario por Norma

2.1.2.2.3.9.1.4.124 Diagrama de Actividad: Caso de Uso Dar de Baja Beneficiario por Norma

Beneficiario por Norma

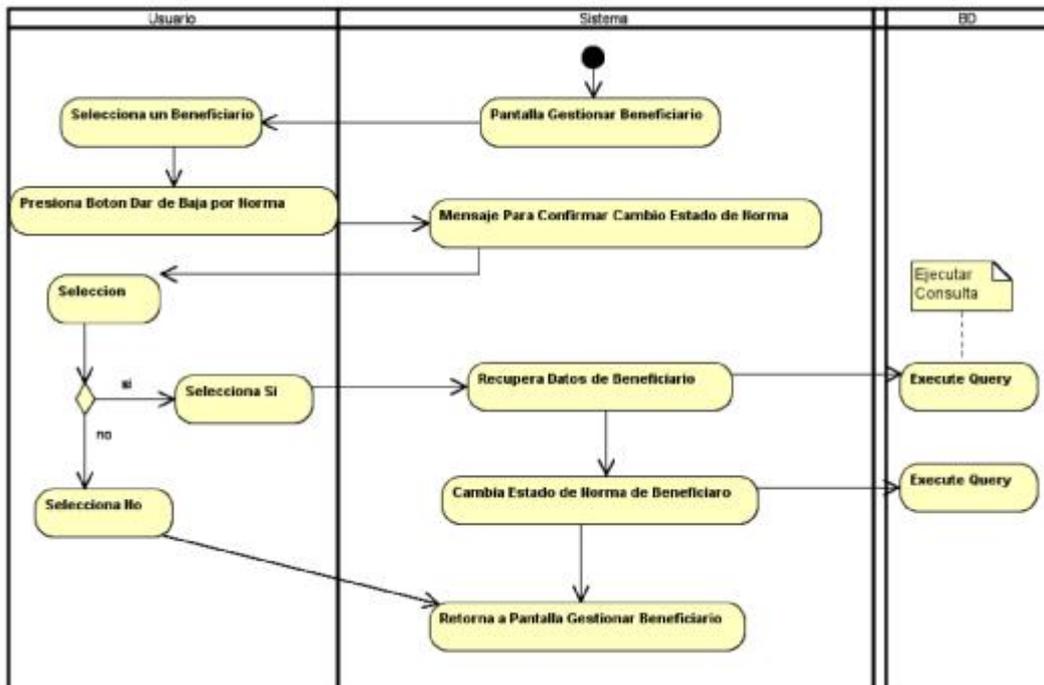


Figura 280. Diagrama de Actividad: CU Dar de Baja Beneficiario por Norma

2.1.2.2.3.9.1.4.125 Diagrama de Actividad: Caso de Uso Gestionar Reporte Asegurado por Tipo de Seguro

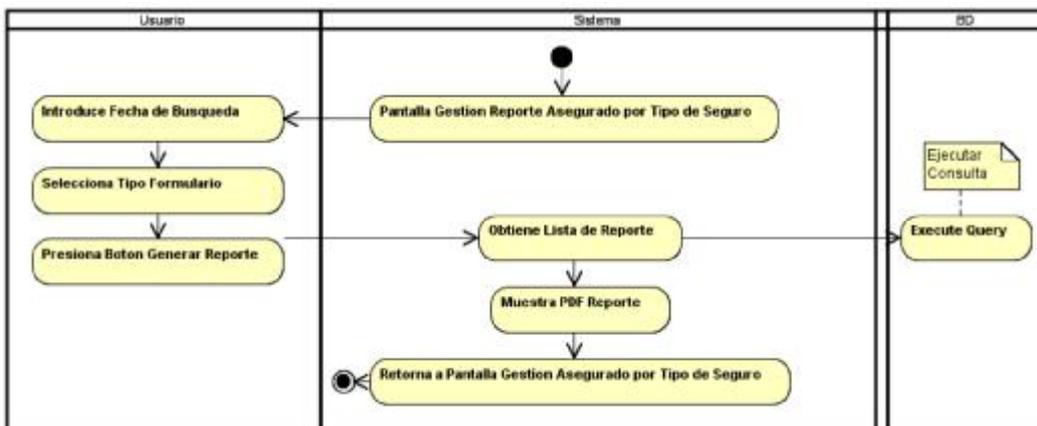


Figura 281. Diagrama de Actividad: CU Gestionar Reporte Asegurado por Tipo de Seguro

2.1.2.2.3.9.1.4.126 Diagrama de Actividad: Caso de Uso Gestionar Reporte Empresa

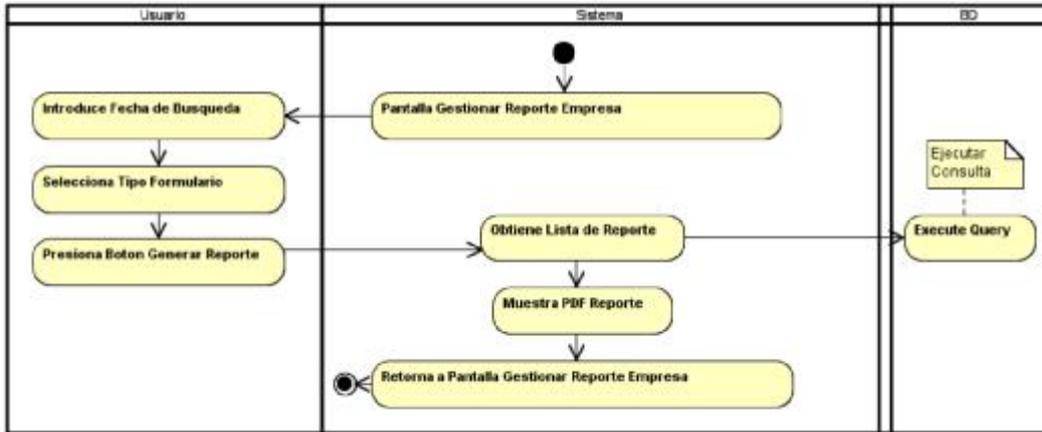


Figura 282. Diagrama de Actividad: CU Gestionar Reporte Empresa

2.1.2.2.3.9.1.4.127 Diagrama de Actividad: Caso de Uso Gestionar Reporte Asegurado

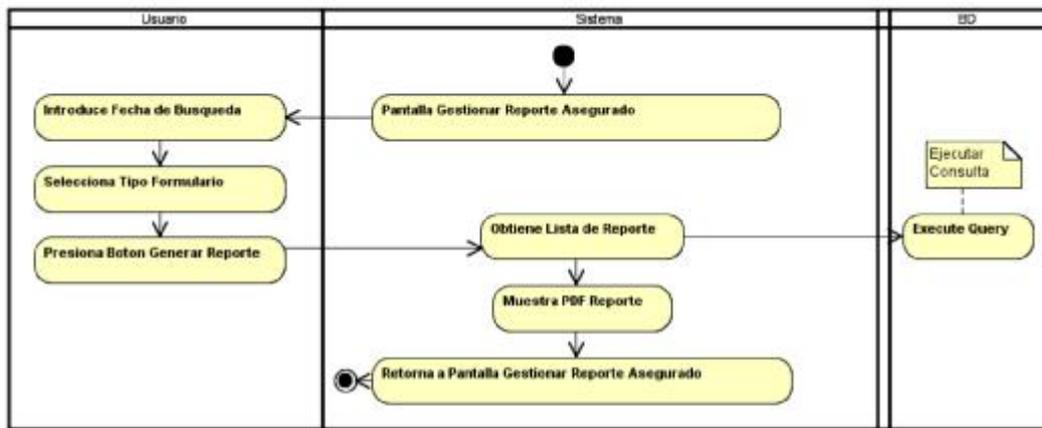


Figura 283. Diagrama de Actividad: CU Gestionar Reporte Asegurado

2.1.2.2.3.9.1.4.128 Diagrama de Actividad: Caso de Uso Gestionar Reporte Beneficiario

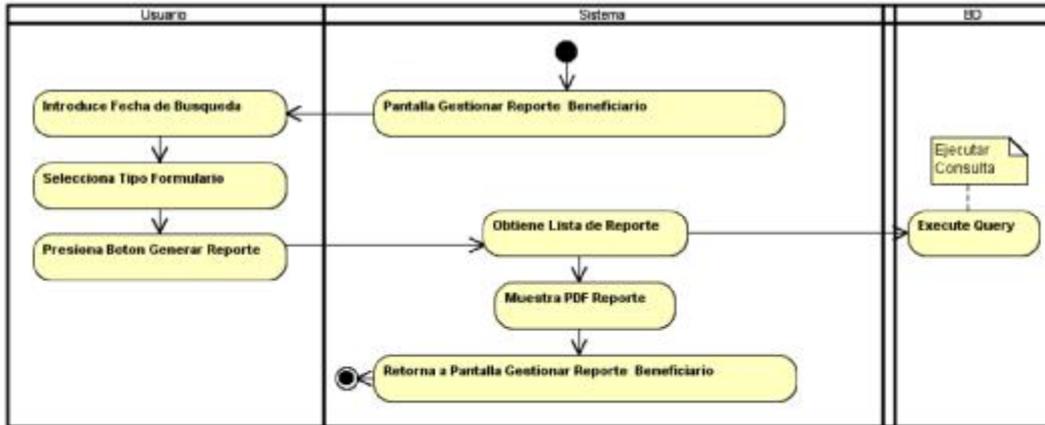


Figura 284. Diagrama de Actividad: CU Gestionar Reporte Beneficiario

2.1.2.2.3.9.1.4.129 Diagrama de Actividad: Caso de Uso Reporte Ficha Empresa

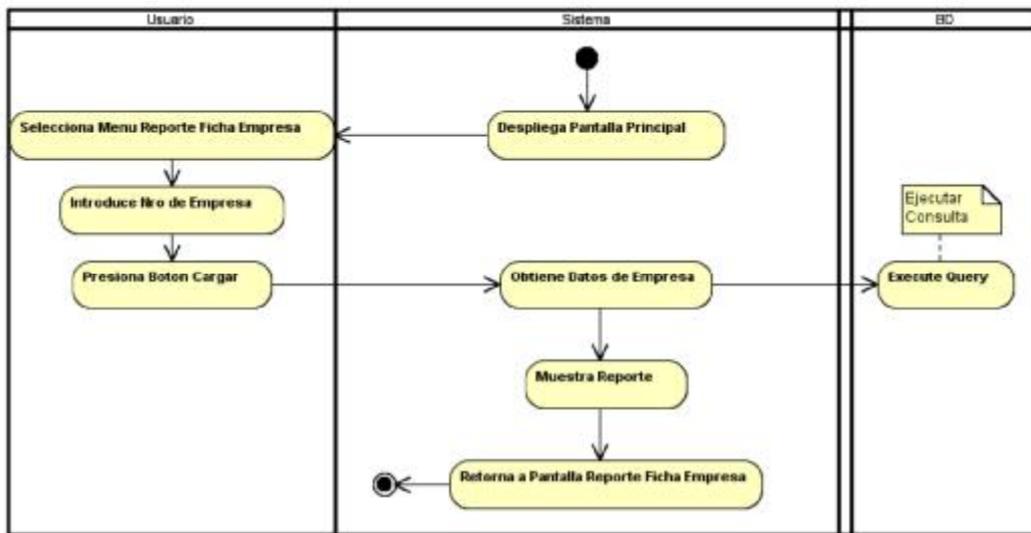


Figura 285. Diagrama de Actividad: CU Reporte Ficha Empresa

2.1.2.2.3.9.1.4.130 Diagrama de Actividad: Caso de Uso Reporte Ficha Asegurado

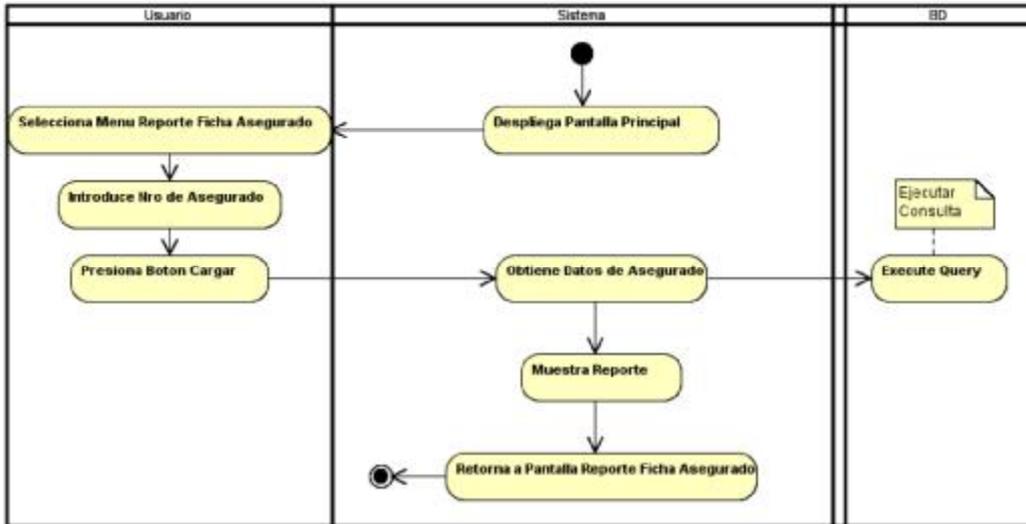


Figura 286. Diagrama de Actividad: CU Reporte Ficha Asegurado

2.1.2.2.3.9.1.4.131 Diagrama de Actividad: Caso de Uso Reporte Ficha Asegurado por Tipo de Seguro

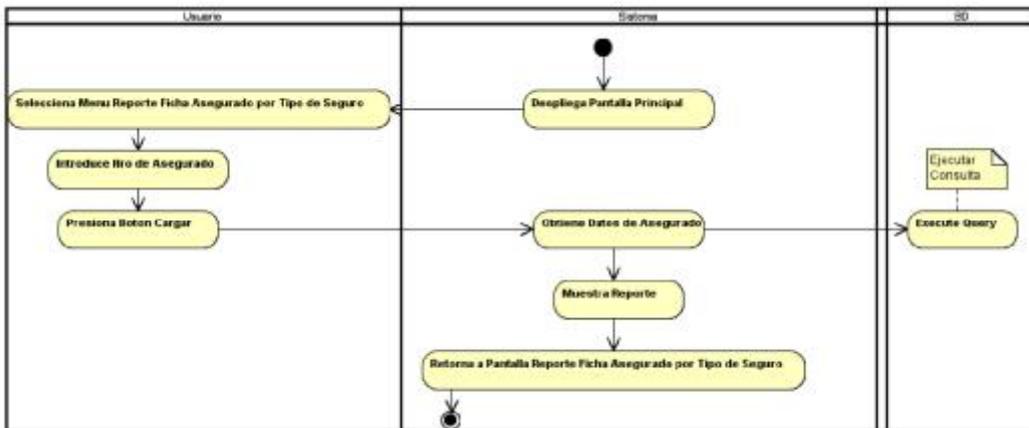


Figura 287. Diagrama de Actividad: CU Reporte Ficha Asegurado por Tipo de Seguro

2.1.2.2.3.9.1.4.132

Diagrama de Actividad: Caso de Uso Reporte Estadístico

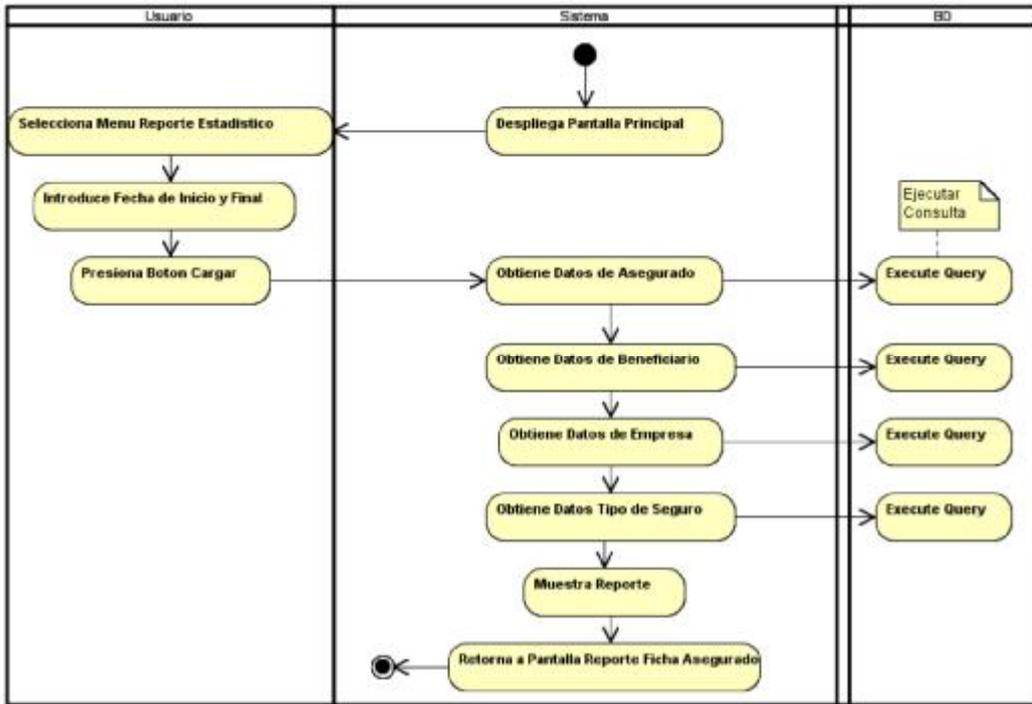


Figura 288. Diagrama de Actividad: CU Reporte Estadístico

2.1.2.2.3.9.1.4.133

Diagrama de Actividad: Caso de Uso Administrar Backup

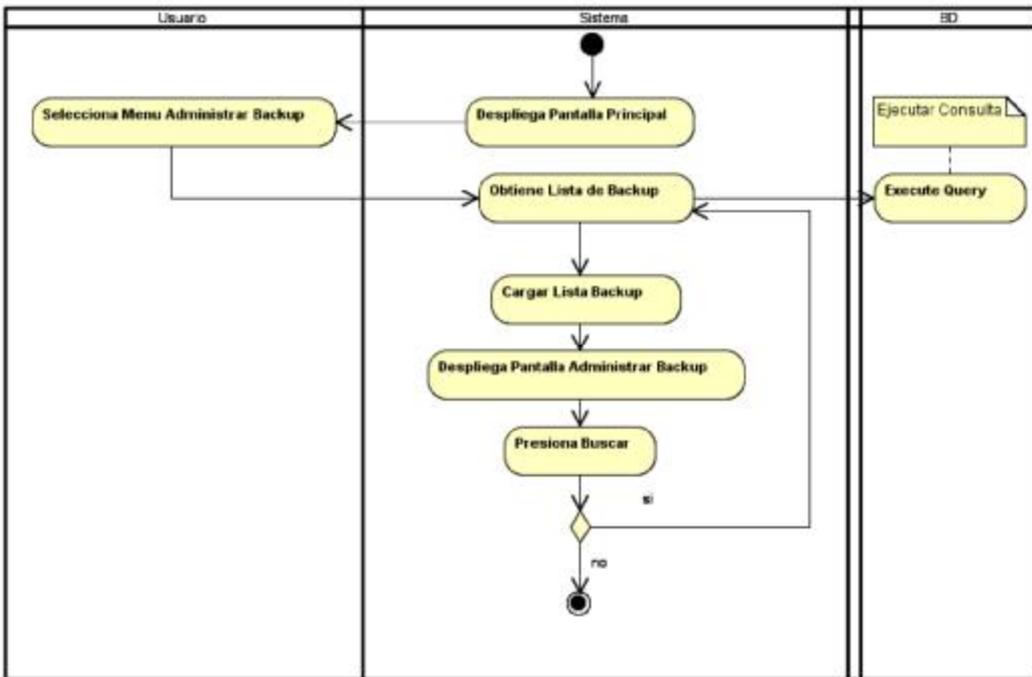


Figura 289. Diagrama de Actividad: CU Administrar Backup

2.1.2.2.3.9.1.4.134

Diagrama de Actividad: Caso de Uso Resguardar Backup

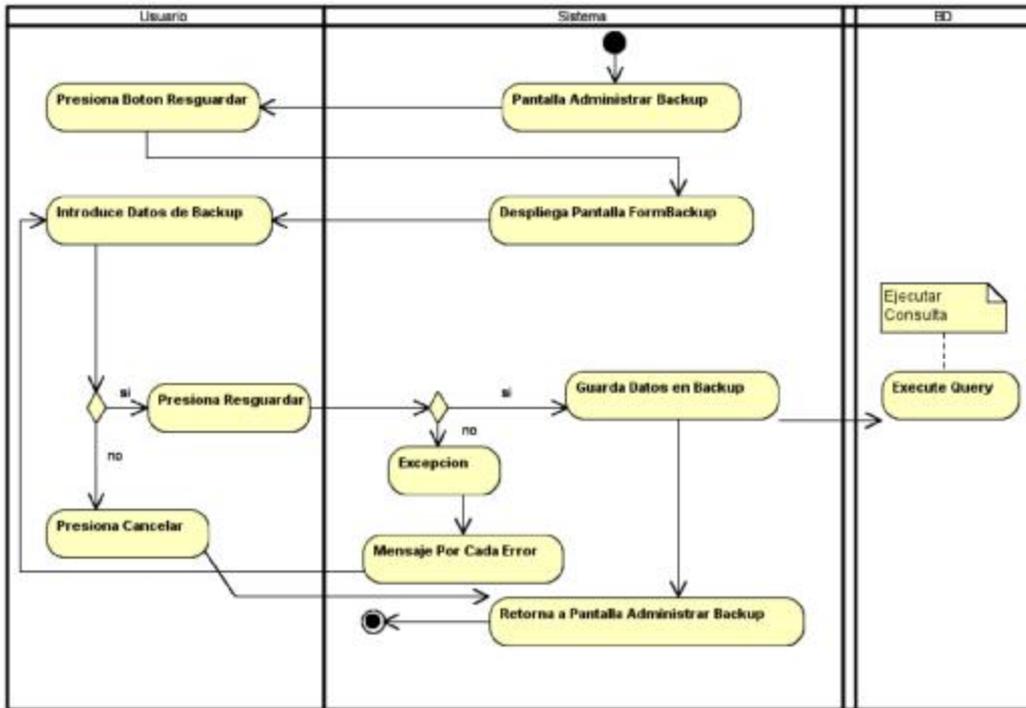


Figura 290. Diagrama de Actividad: CU Resguardar Backup

2.1.2.2.3.9.1.4.135

Diagrama de Actividad: Caso de Uso Restaurar Backup

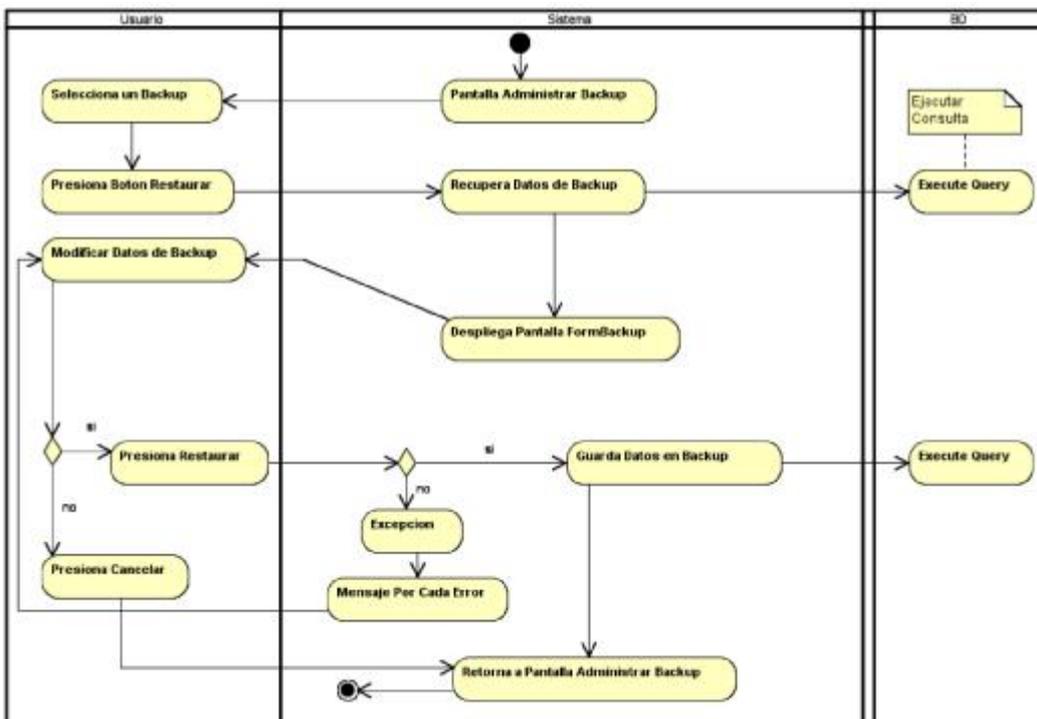


Figura 291. Diagrama de Actividad: CU Restaurar Backup

2.1.2.2.3.9.1.4.136

Diagrama de Actividad: Caso de Uso Eliminar Backup

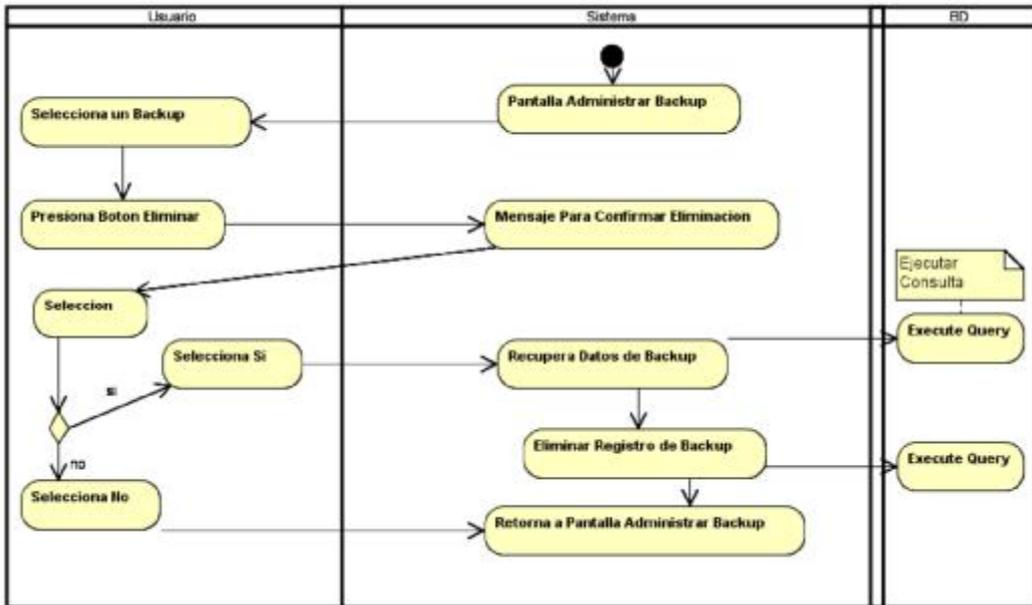


Figura 292. Diagrama de Actividad: CU Eliminar Backup

2.1.2.2.3.9.1.4.137

Diagrama de Actividad: Caso de Uso Generar Reporte Lista Backup

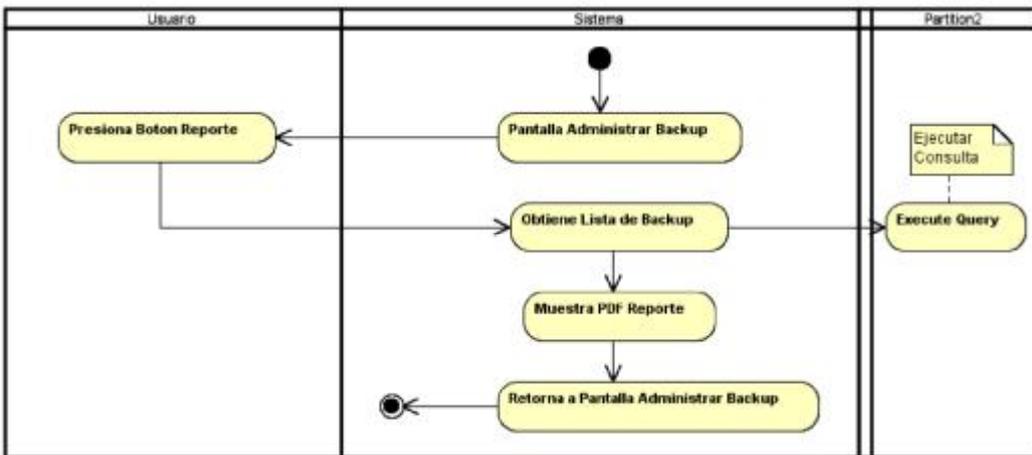


Figura 293. Diagrama de Actividad: CU Generar Reporte Lista Backup

2.1.2.2.3.9.2 Modelado de Diagrama de Secuencias

Un Diagrama de Secuencia muestra las interacciones entre objetos ordenadas en secuencia temporal. Muestra los objetos que se encuentran en el escenario y la secuencia de mensajes intercambiados entre los objetos para llevar a cabo la funcionalidad descrita por el escenario. En aplicaciones grandes además de los objetos se muestran también los componentes y Casos de Uso. El mostrar los componentes tiene sentido ya que se trata de objetos reutilizables, en cuanto a los Casos de Uso hay

que recordar que se implementan como objetos cuyo rol es encapsular lo definido en el Caso de Uso.

Para mostrar la interacción con el usuario o con otro Sistema se introducen en los Diagramas de Secuencia las *boundary classes*. En las primeras fases de diseño el propósito de introducir estas clases es capturar y documentar los requisitos de interfaz, pero no el mostrar cómo se va a implementar dicha interfaz.

Los Diagramas de Secuencia, formalmente diagramas de traza de eventos o de interacción de objetos, se utilizan con frecuencia para validar los Casos de Uso. Documentan el diseño desde el punto de vista de los Casos de Uso. Observando qué mensajes se envían a los objetos, componentes o Casos de Uso y viendo a grosso modo cuánto tiempo consume el método invocado, los Diagramas de Secuencia nos ayudan a comprender los cuellos de botella potenciales, para así poder eliminarlos. A la hora de documentar un diagrama de secuencia resulta importante mantener los enlaces de los mensajes a los métodos apropiados del Diagrama de Clases.

2.1.2.2.3.9.2.1 Diagrama de Secuencias

2.1.2.2.3.9.2.1.1 Diagrama de Secuencia: Caso de Uso Ingresar al Sistema

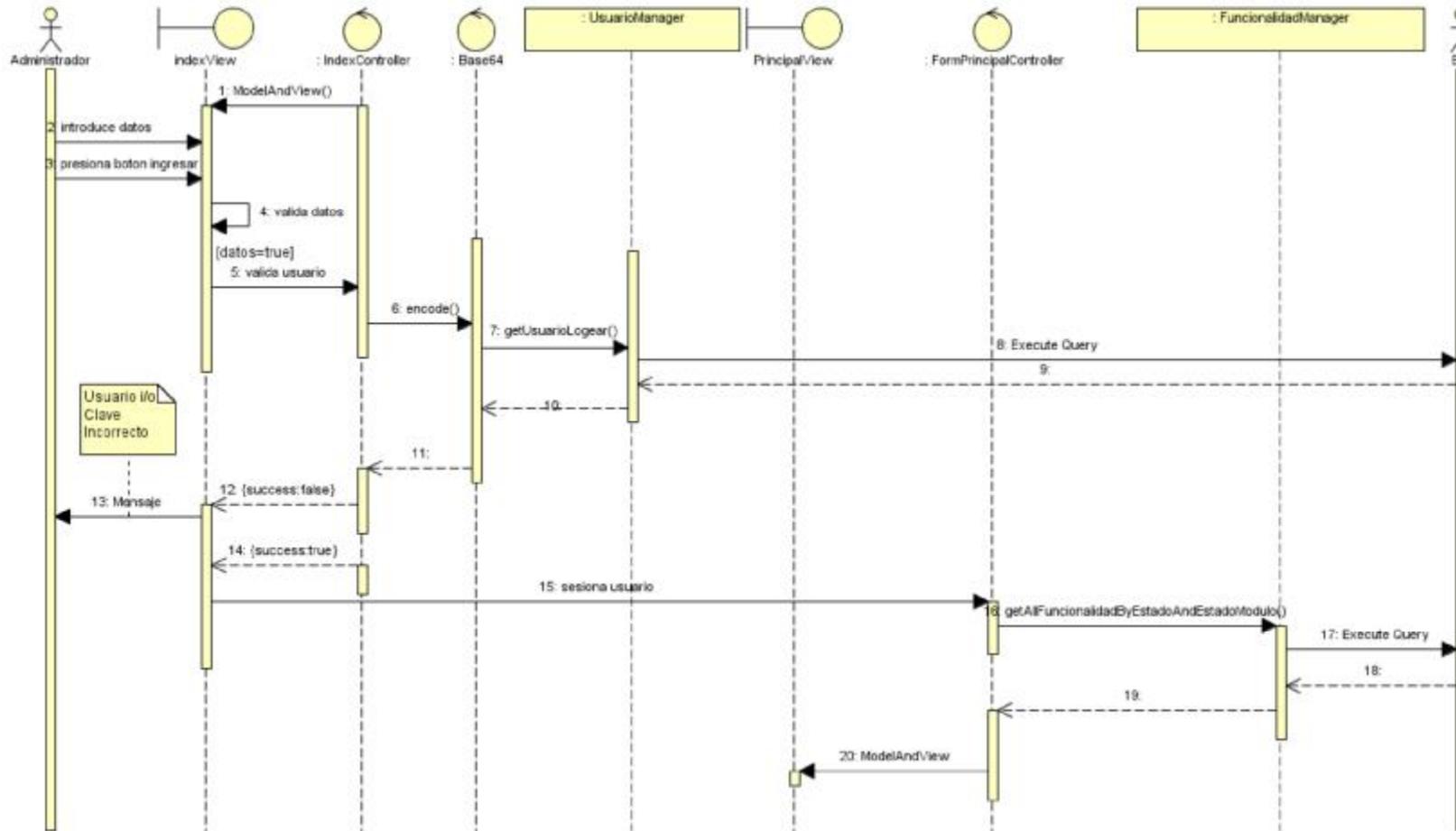


Figura 294. Diagrama de Secuencia: CU Ingresar al Sistema

2.1.2.2.3.9.2.1.2 Diagrama de Secuencia: Caso de Uso Administrar Actividad Económica

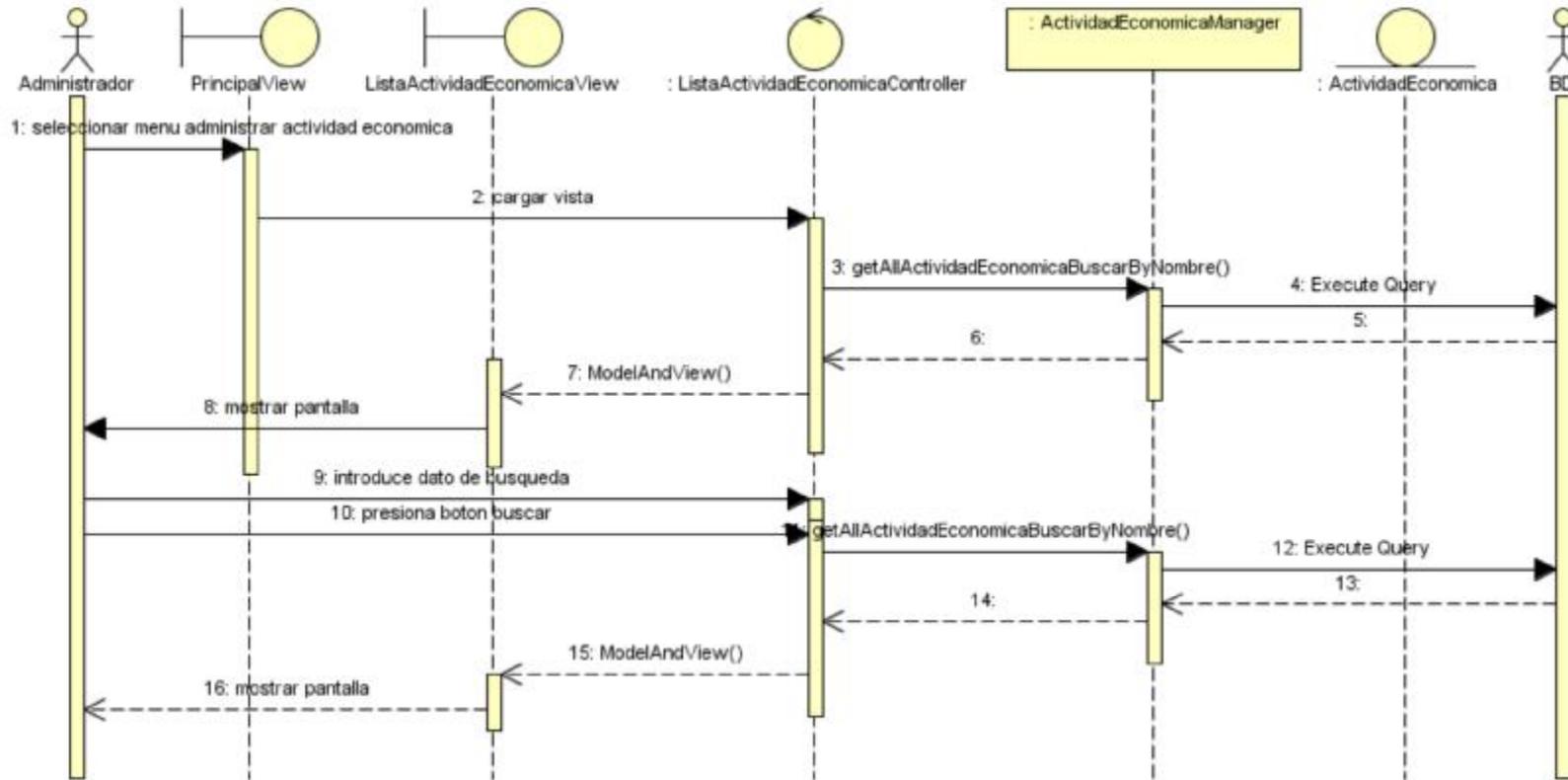


Figura 295. Diagrama de Secuencia: CU Administrar Actividad Económica

2.1.2.2.3.9.2.1.3 Diagrama de Secuencia: Caso de Uso Adicionar Actividad Económica

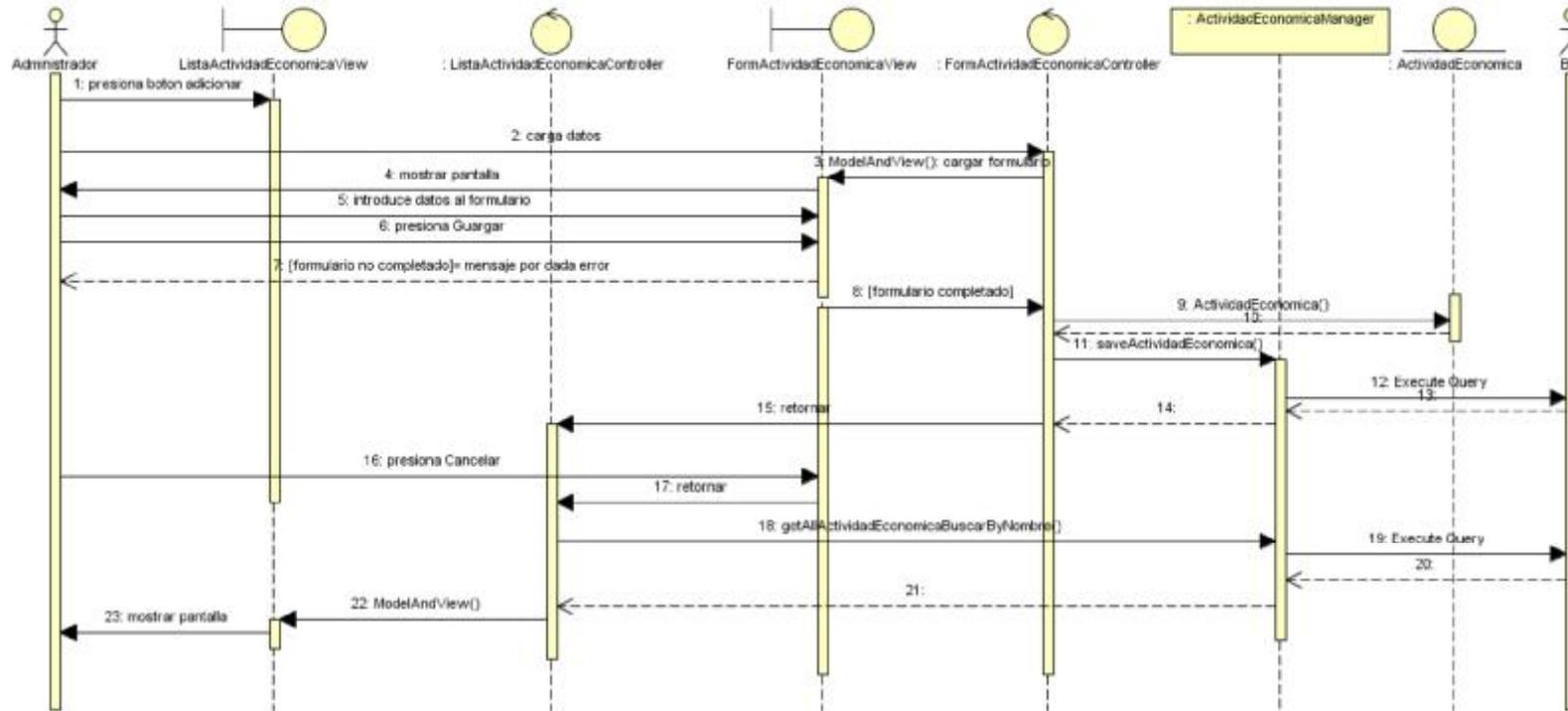


Figura 296. Diagrama de Secuencia: CU Adicionar Actividad Económica

2.1.2.2.3.9.2.1.4 Diagrama de Secuencia: Caso de Uso Modificar Actividad Económica

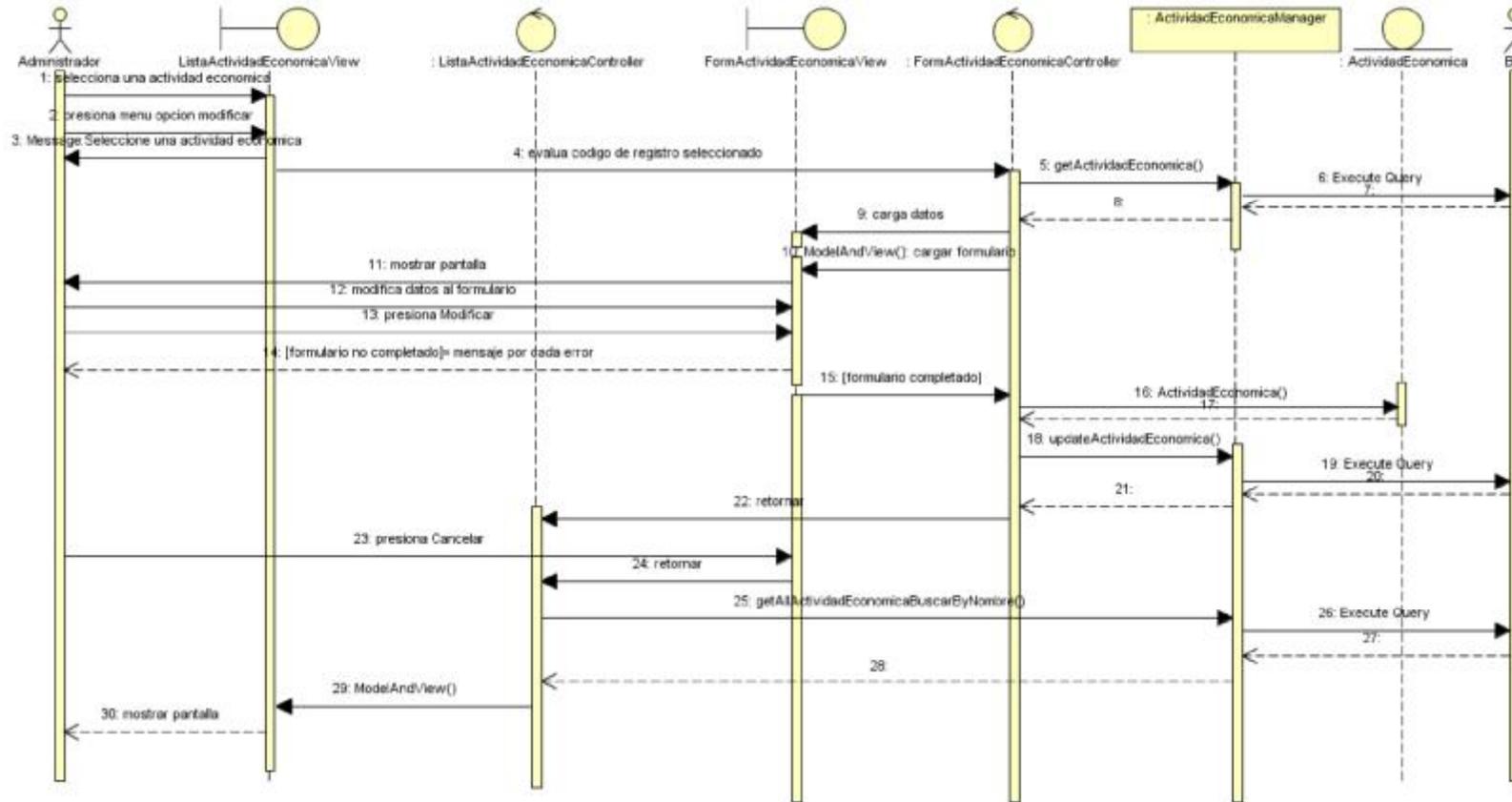


Figura 297. Diagrama de Secuencia: CU Modificar Actividad Económica

2.1.2.2.3.9.2.1.5 Diagrama de Secuencia: Caso de Uso Eliminar Actividad Económica

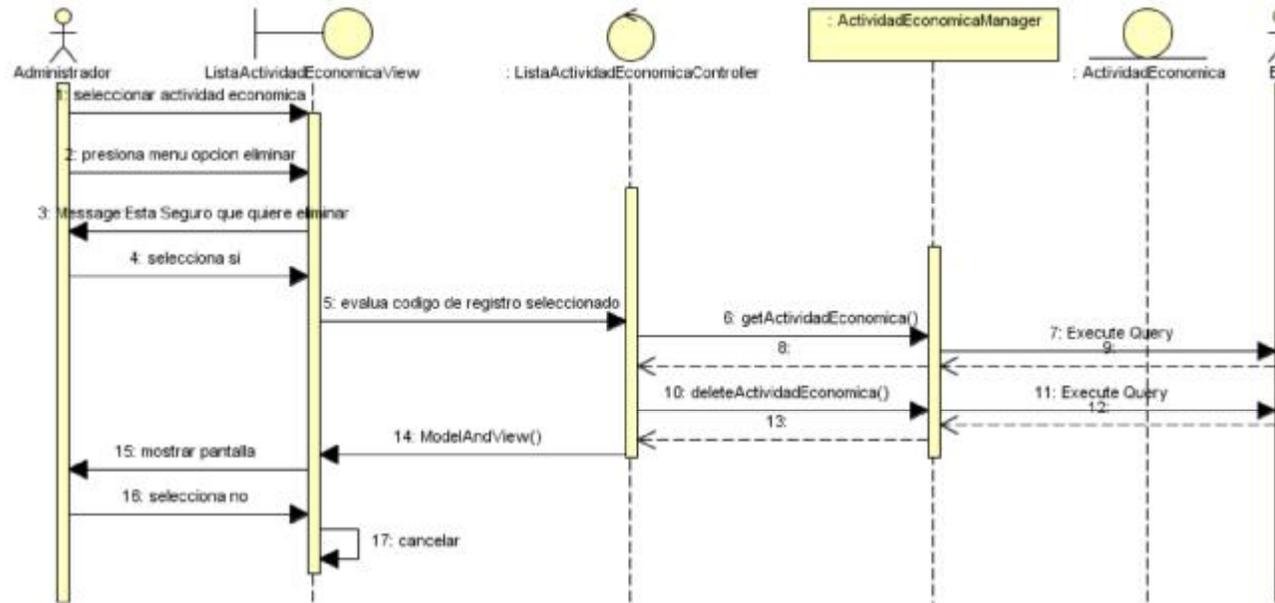


Figura 298. Diagrama de Secuencia: CU Eliminar Actividad Económica

2.1.2.2.3.9.2.1.6 Diagrama de Secuencia: Caso de Uso Cambiar Estado Actividad Económica

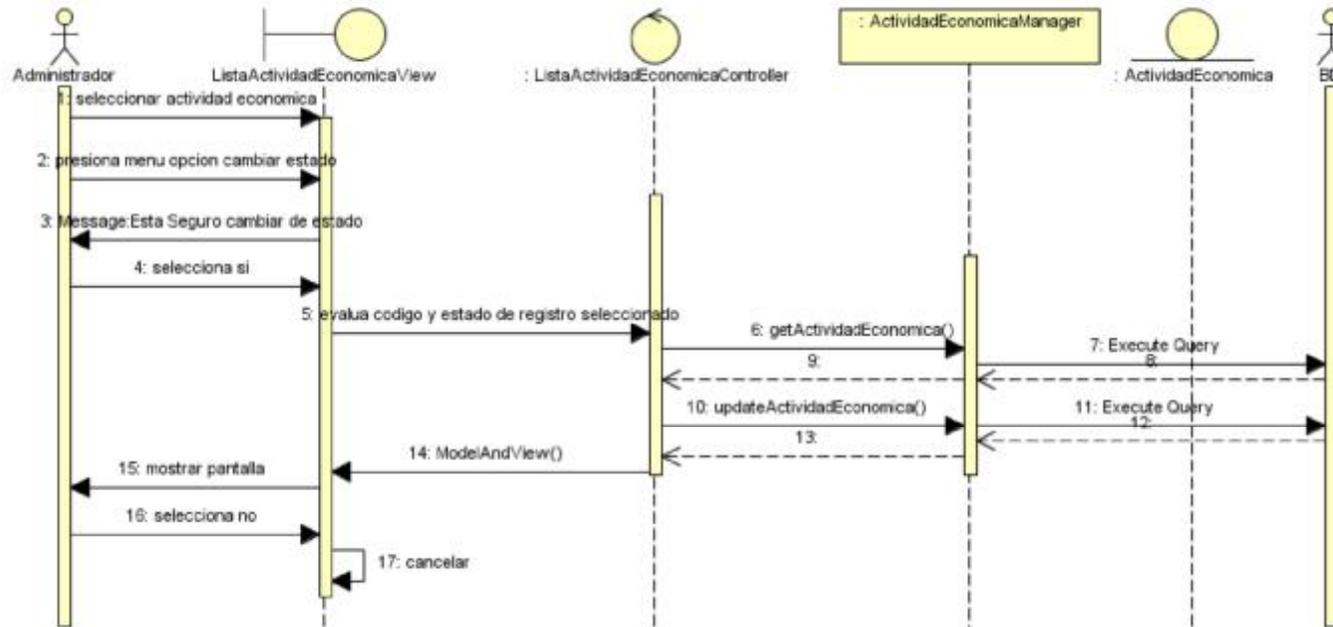


Figura 299. Diagrama de Secuencia: CU Cambiar Estado Actividad Económica

2.1.2.2.3.9.2.1.7 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Actividad Económica

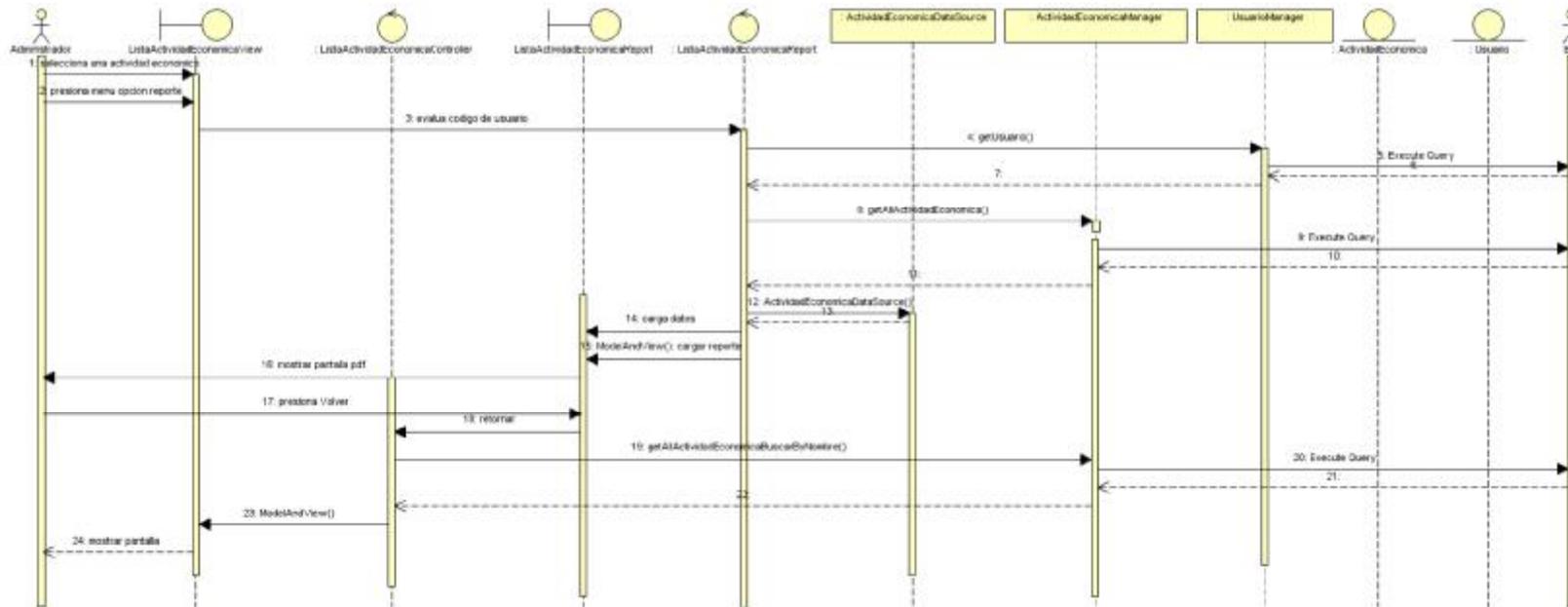


Figura 300. Diagrama de Secuencia: CU Generar Reporte Lista Actividad Económica

2.1.2.2.3.9.2.1.8 Diagrama de Secuencia: Caso de Uso Administrar Centro Salud

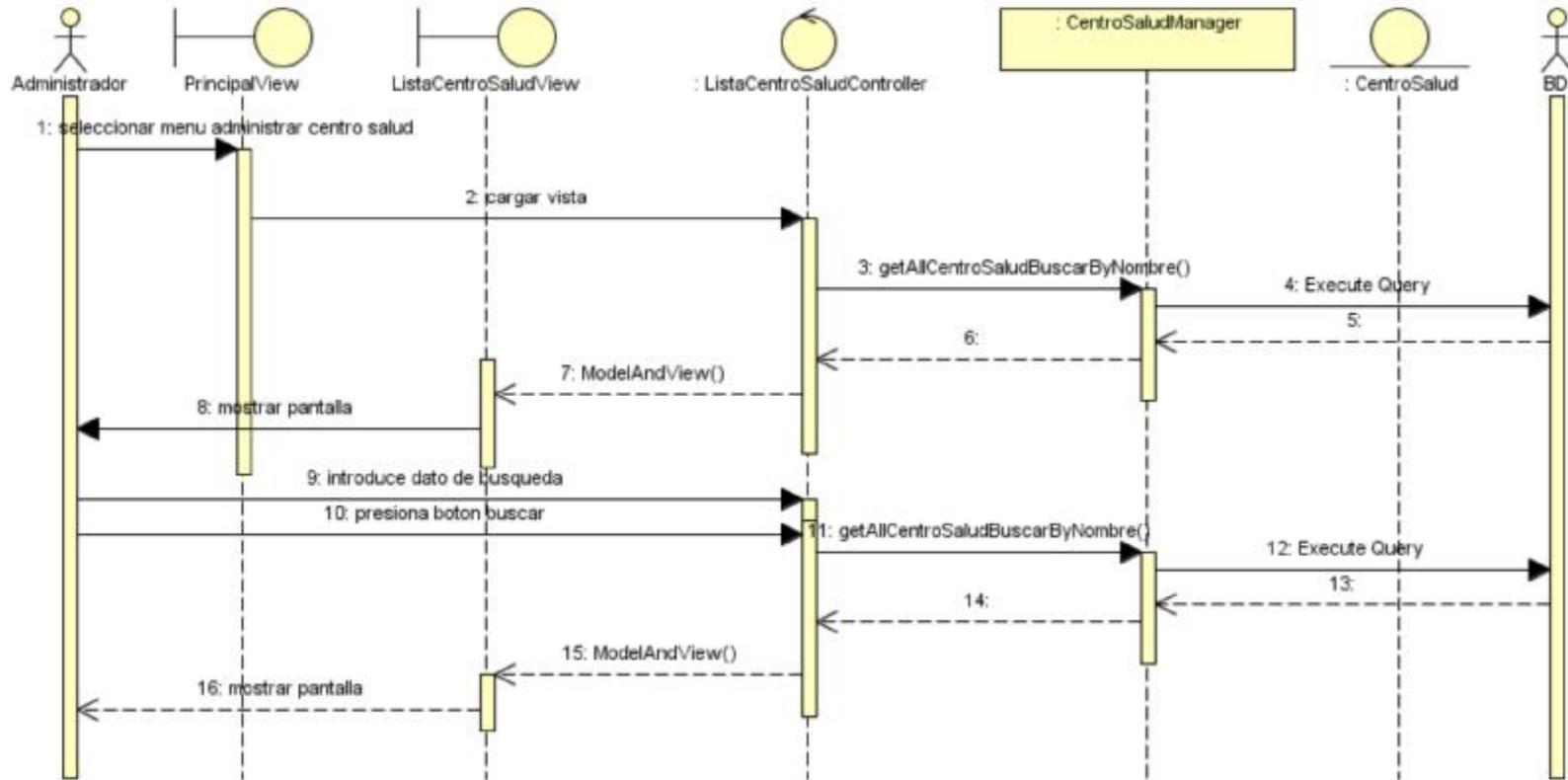


Figura 301. Diagrama de Secuencia: CU Administrar Centro Salud

2.1.2.2.3.9.2.1.9 Diagrama de Secuencia: Caso de Uso Adicionar Centro Salud

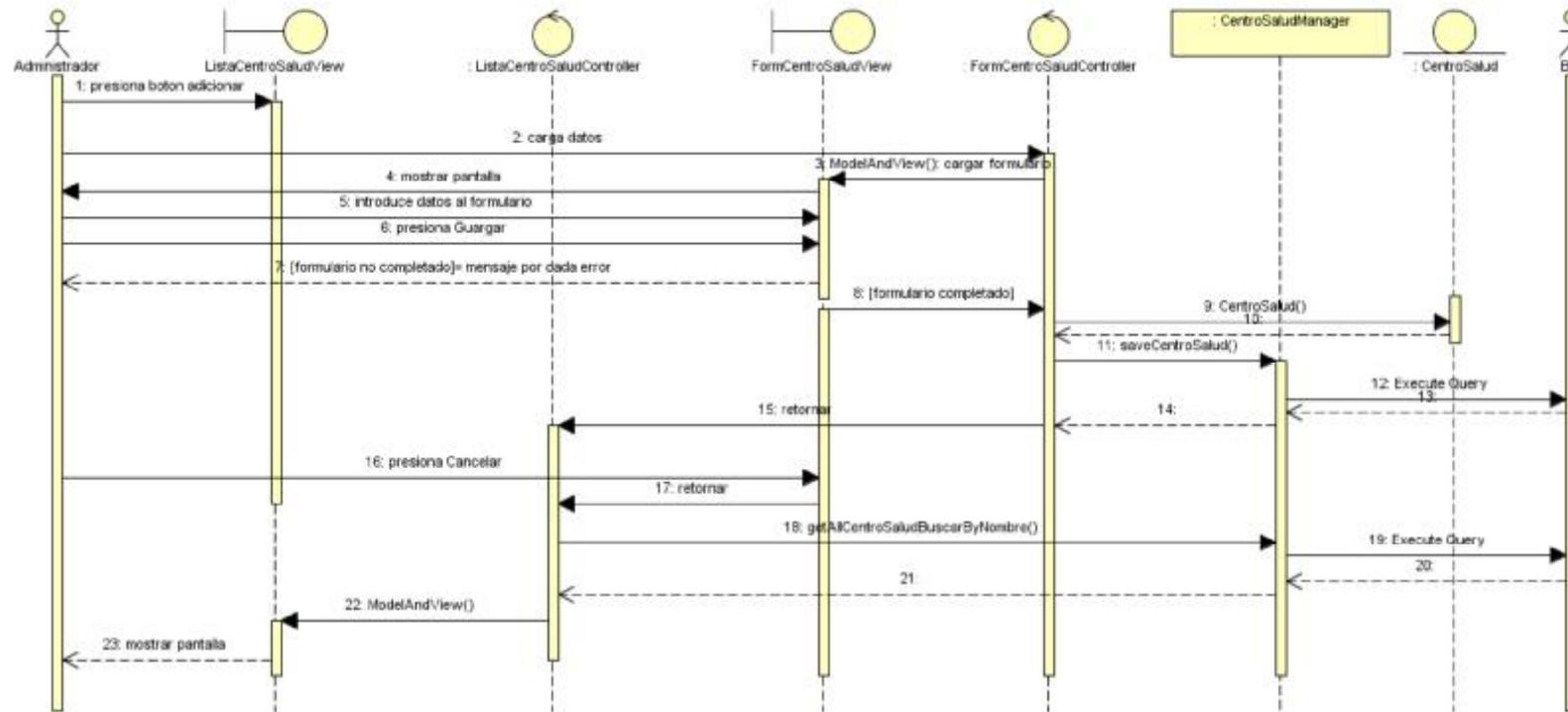


Figura 302. Diagrama de Secuencia: CU Adicionar Centro Salud

2.1.2.2.3.9.2.1.10 Diagrama de Secuencia: Caso de Uso Modificar Centro Salud

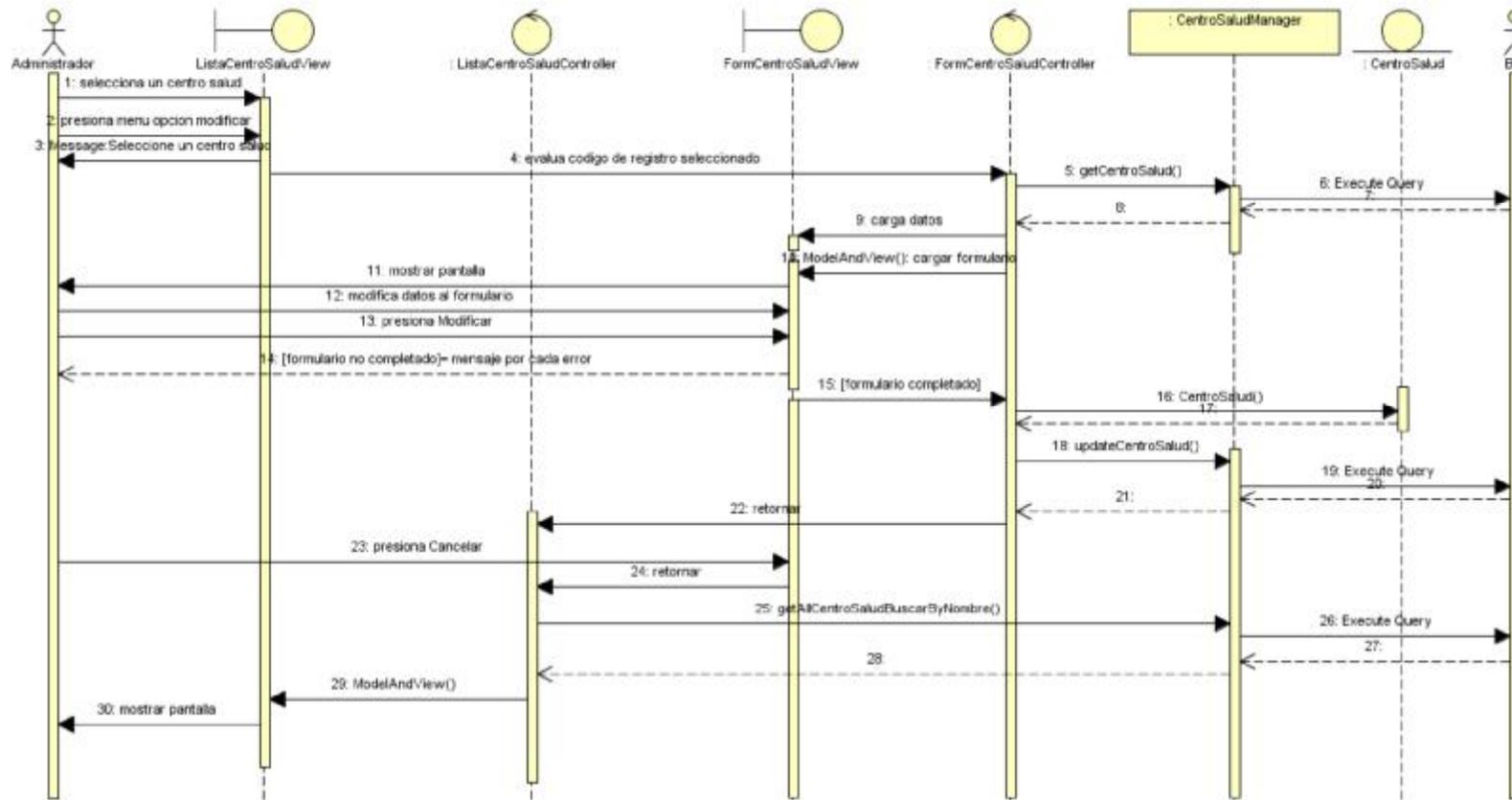


Figura 303. Diagrama de Secuencia: CU Modificar Centro Salud

2.1.2.2.3.9.2.1.11 Diagrama de Secuencia: Caso de Uso Eliminar Centro Salud

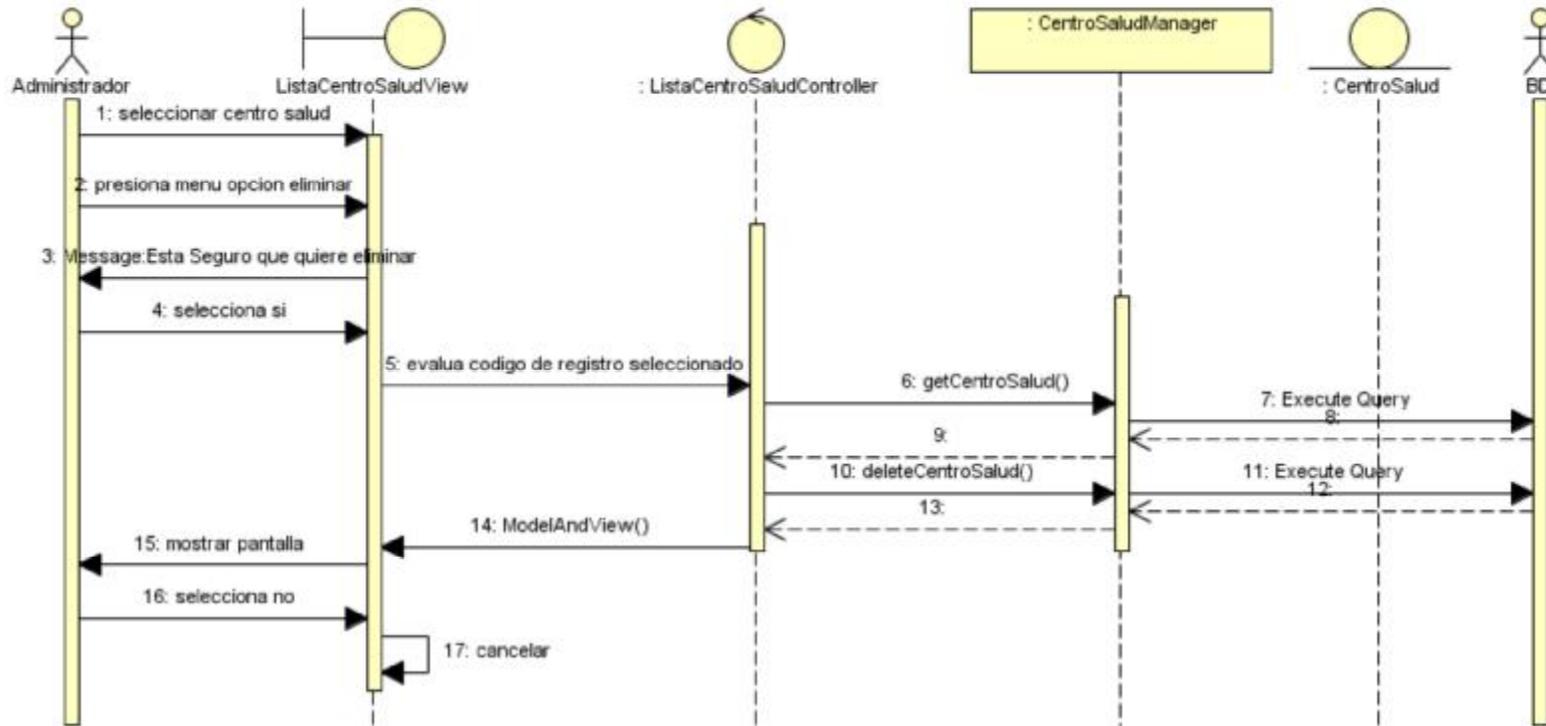


Figura 304. Diagrama de Secuencia: CU Eliminar Centro Salud

2.1.2.2.3.9.2.1.12 Diagrama de Secuencia: Caso de Uso Cambiar Estado Centro Salud

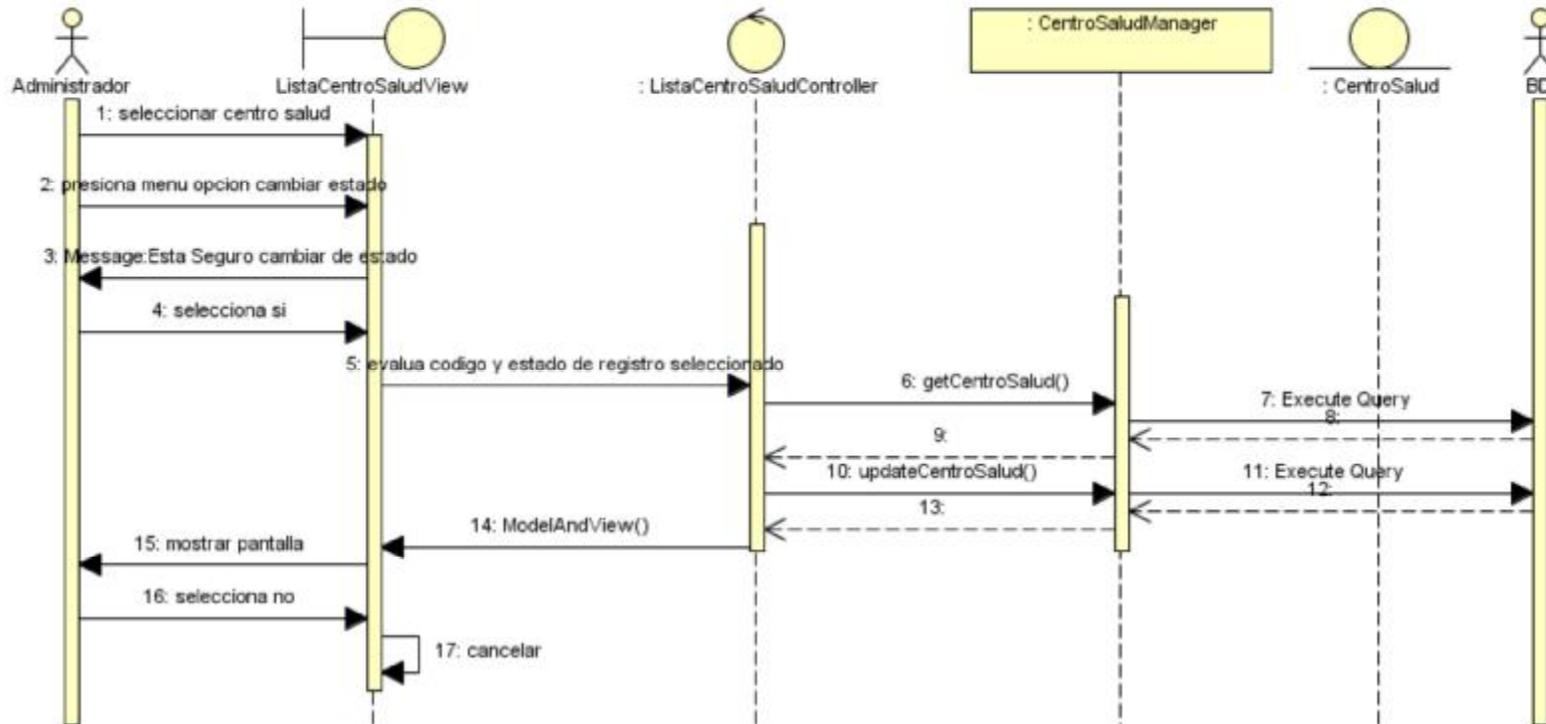


Figura 305. Diagrama de Secuencia: CU Cambiar Estado Centro Salud

2.1.2.2.3.9.2.1.13 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Centro Salud

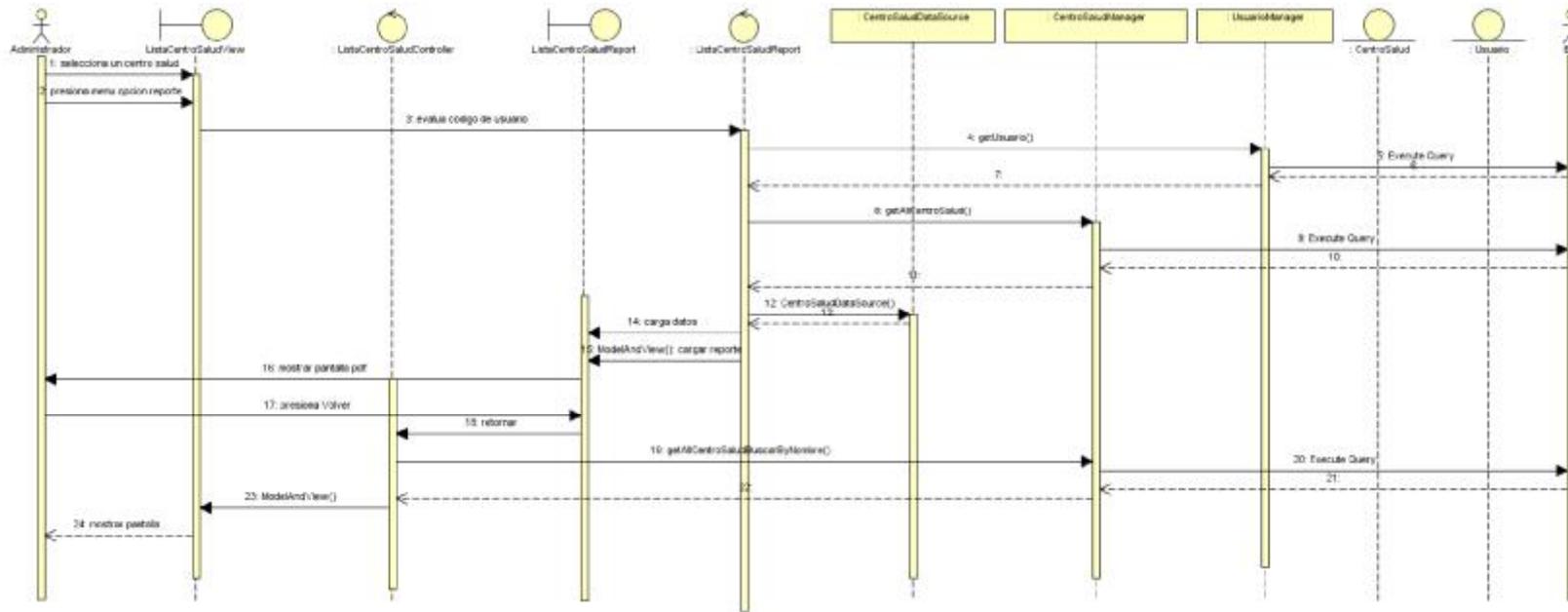


Figura 306. Diagrama de Secuencia: CU Generar Reporte Lista Centro Salud

2.1.2.2.3.9.2.1.14 Diagrama de Secuencia: Caso de Uso Administrar Departamento

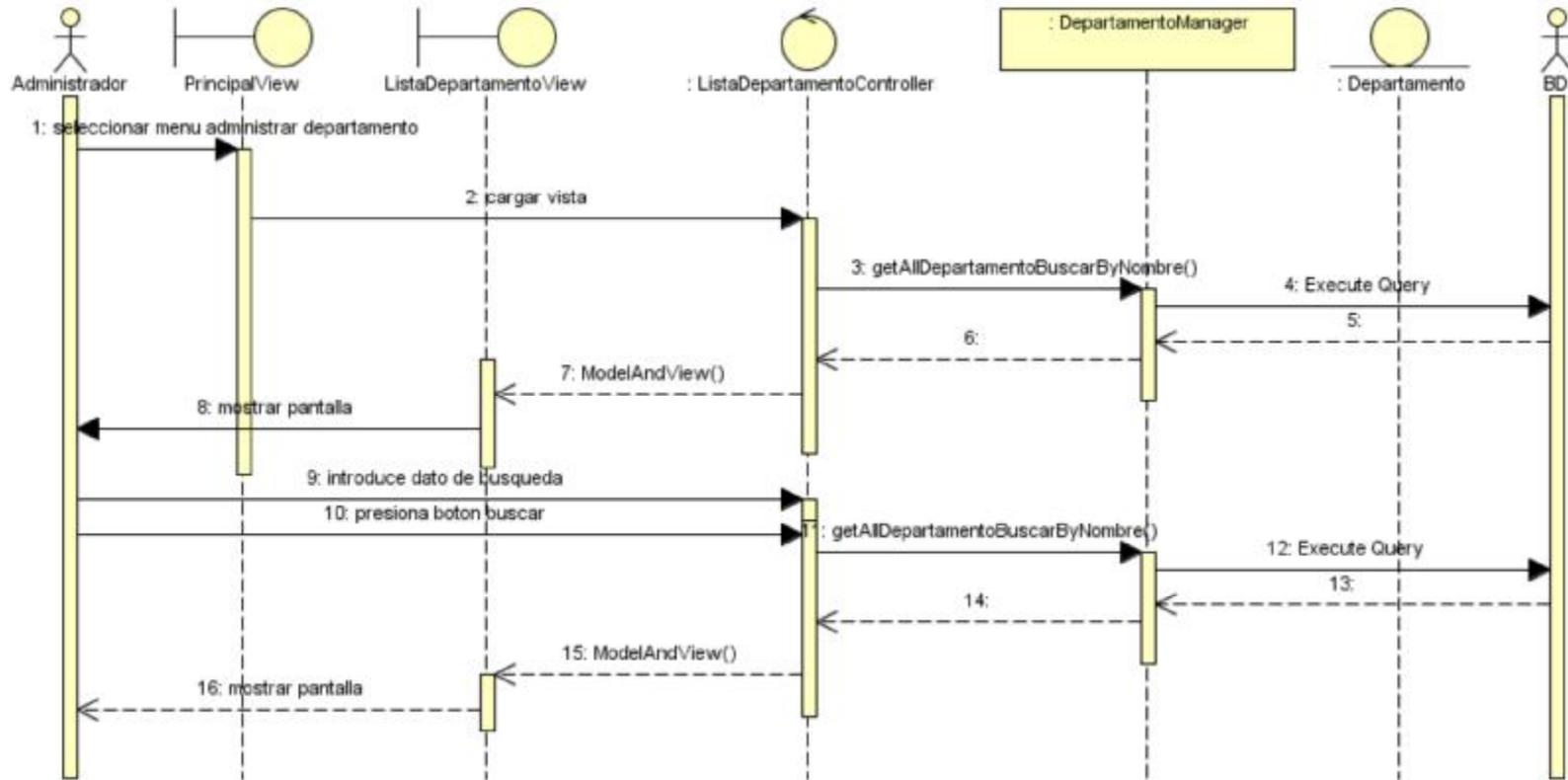


Figura 307. Diagrama de Secuencia: CU Administrar Departamento

2.1.2.2.3.9.2.1.15 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Departamento

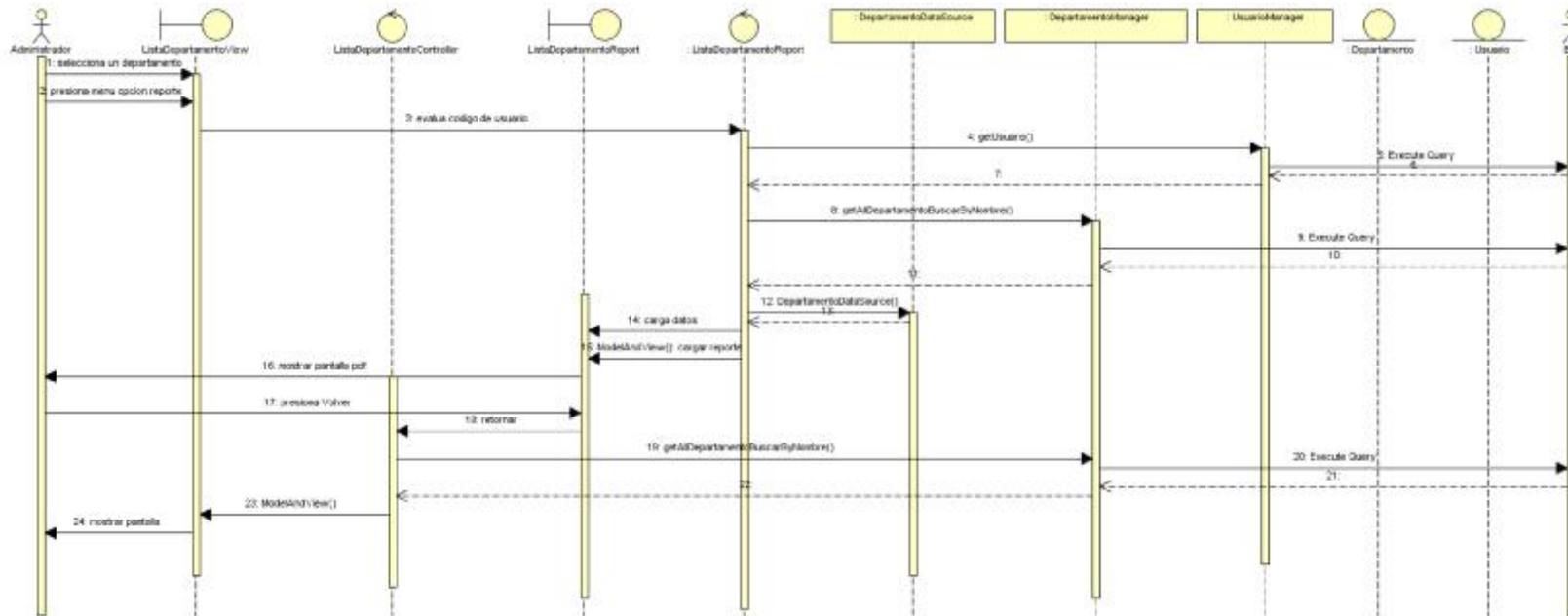


Figura 308. Diagrama de Secuencia: CU Generar Reporte Lista Departamento

2.1.2.2.3.9.2.1.16 Diagrama de Secuencia: Caso de Uso Administrar Funcionalidad

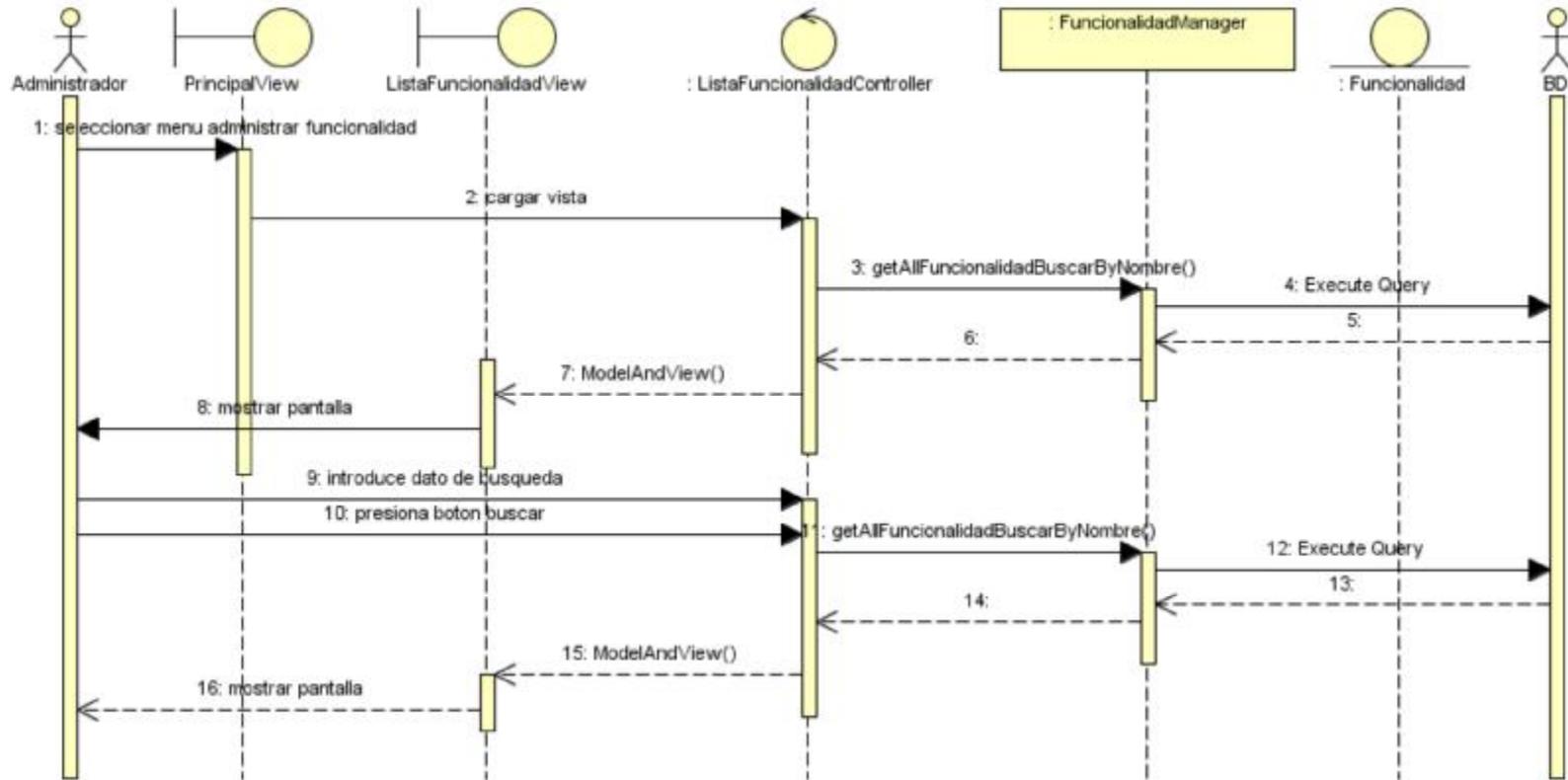


Figura 309. Diagrama de Secuencia: CU Administrar Funcionalidad

2.1.2.2.3.9.2.1.17 Diagrama de Secuencia: Caso de Uso Cambiar Estado Funcionalidad

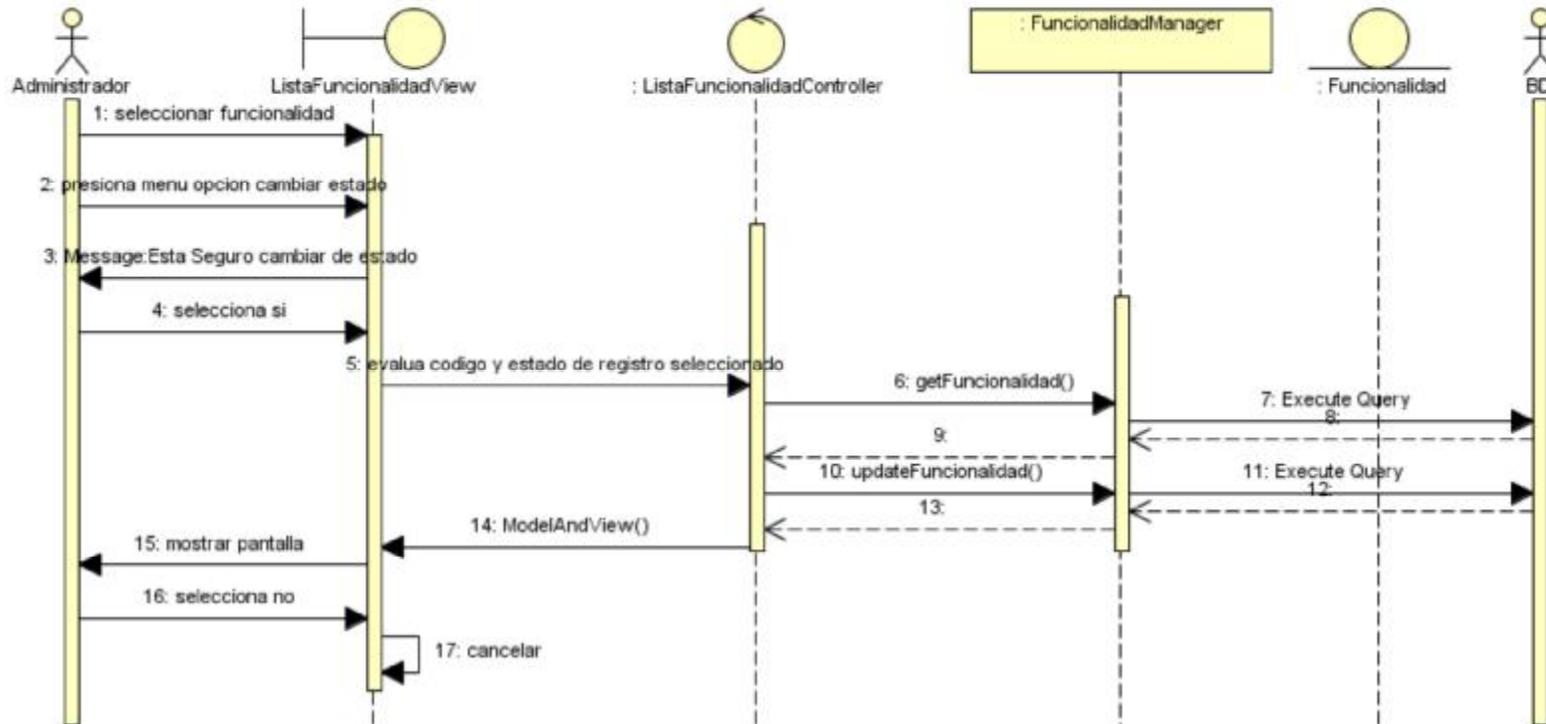


Figura 310. Diagrama de Secuencia: CU Cambiar Estado Funcionalidad

2.1.2.2.3.9.2.1.18 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Funcionalidad

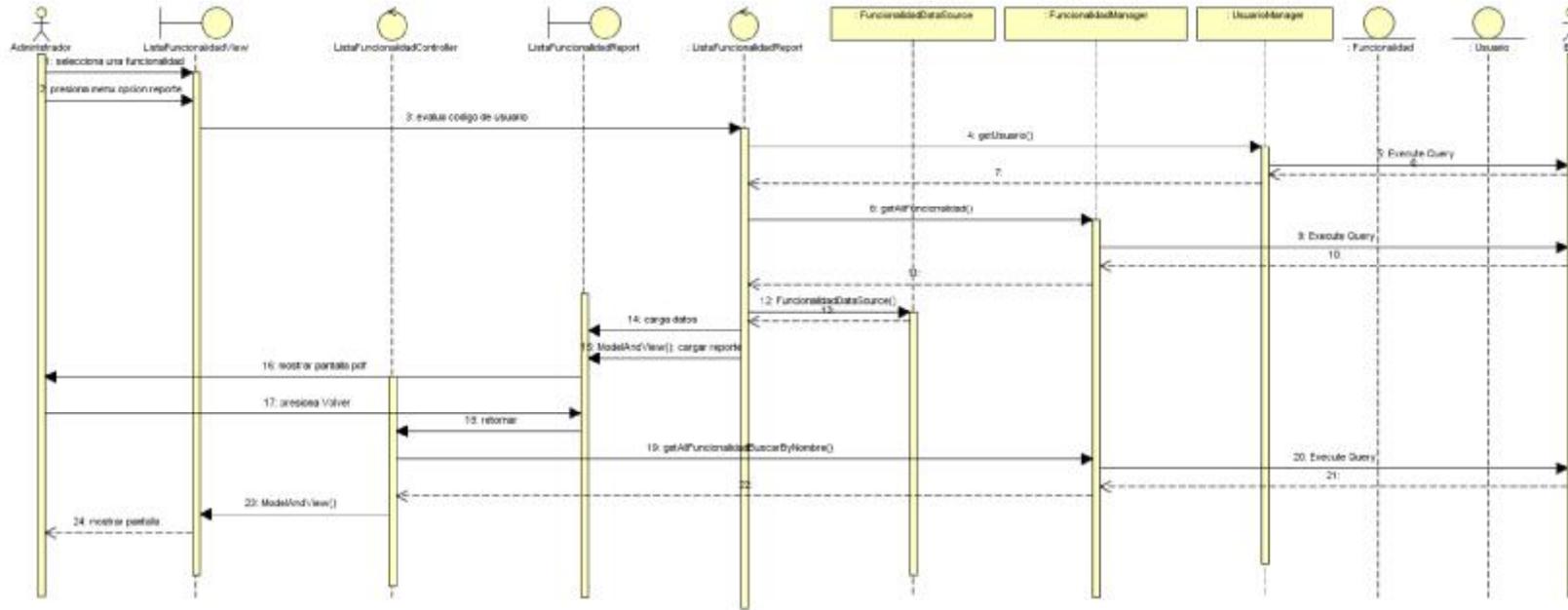


Figura 311. Diagrama de Secuencia: CU Generar Reporte Lista Funcionalidad

2.1.2.2.3.9.2.1.19 Diagrama de Secuencia: Caso de Uso Administrar Módulo

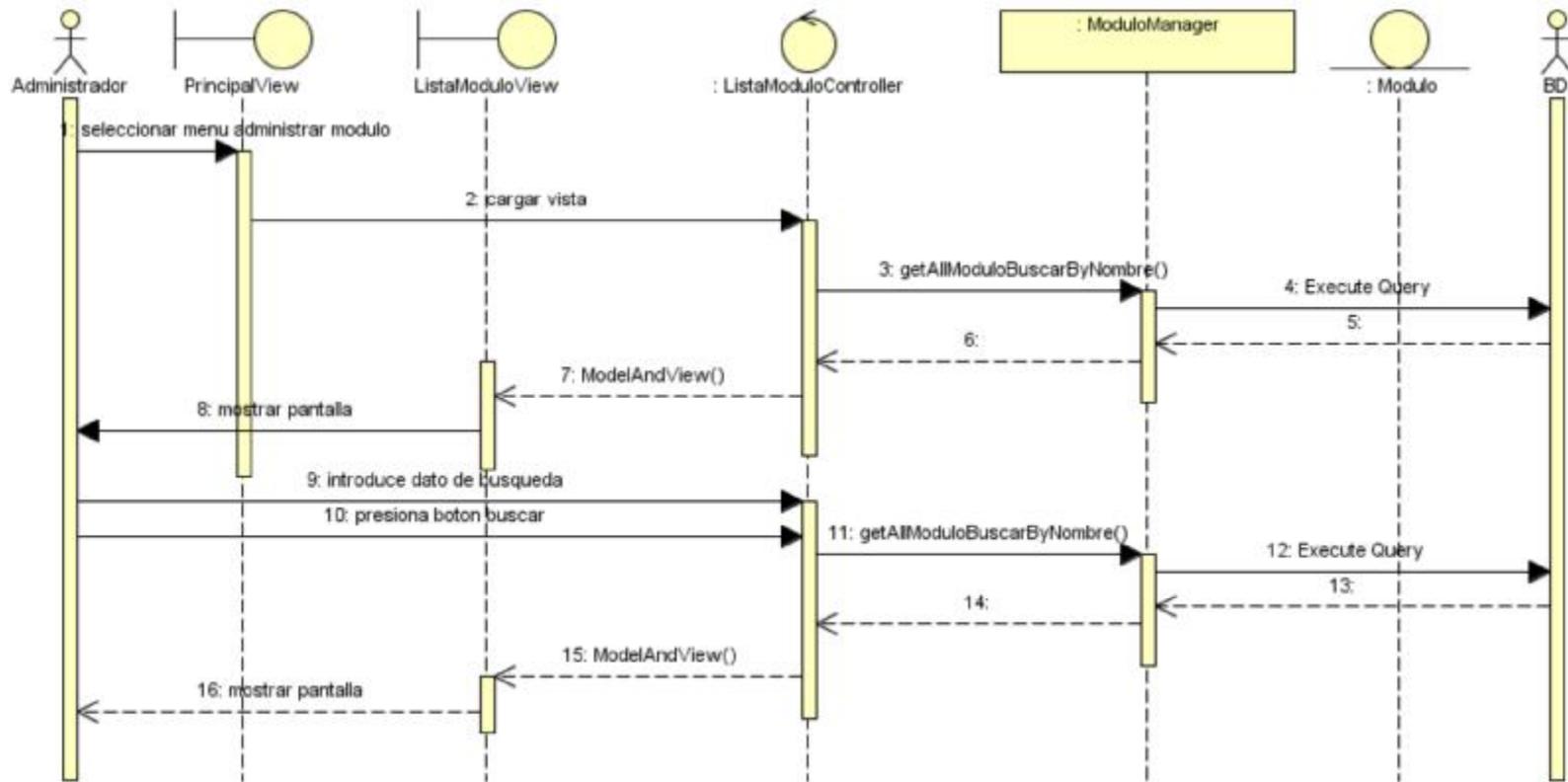


Figura 312. Diagrama de Secuencia: CU Administrar Módulo

2.1.2.2.3.9.2.1.20 Diagrama de Secuencia: Caso de Uso Cambiar Estado Módulo

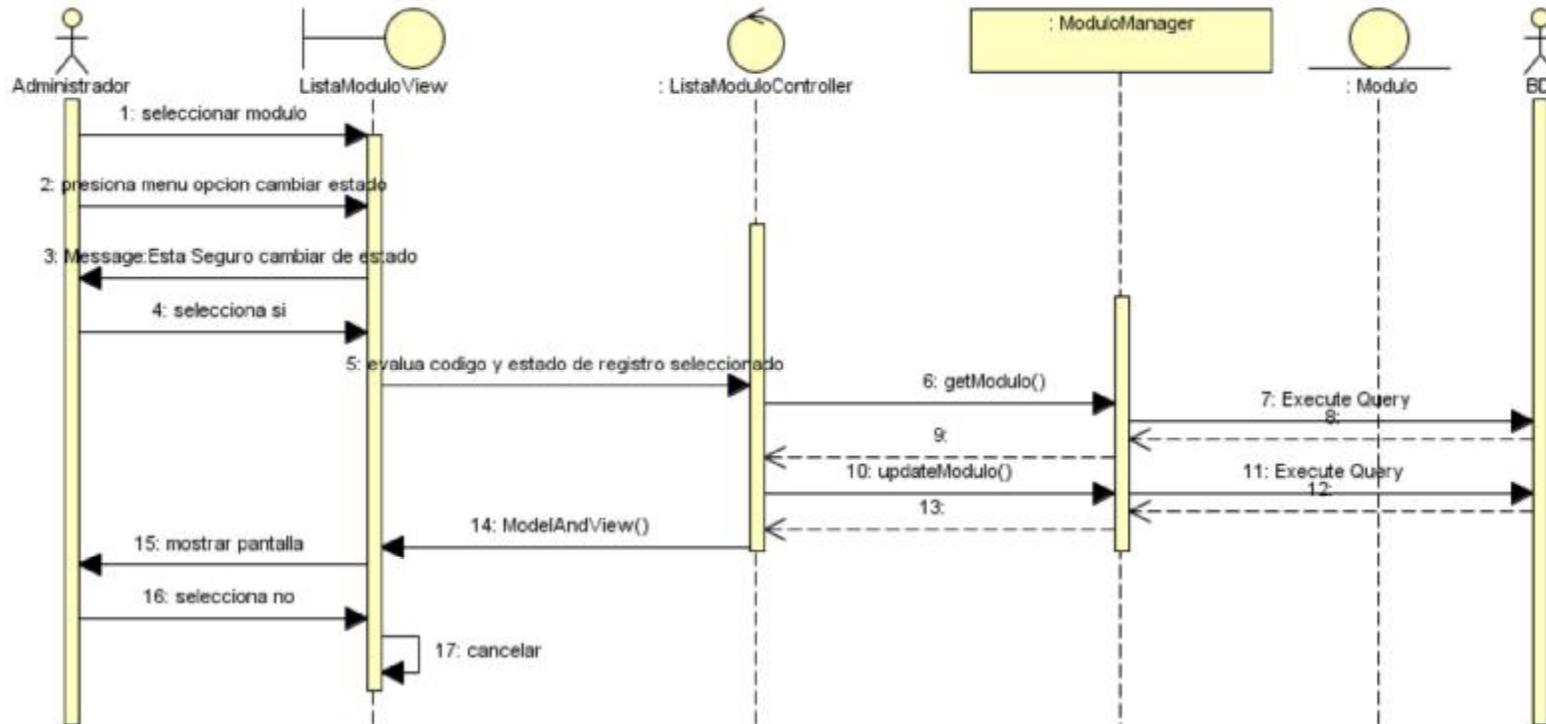


Figura 313. Diagrama de Secuencia: CU Cambiar Estado Módulo

2.1.2.2.3.9.2.1.21 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Módulo

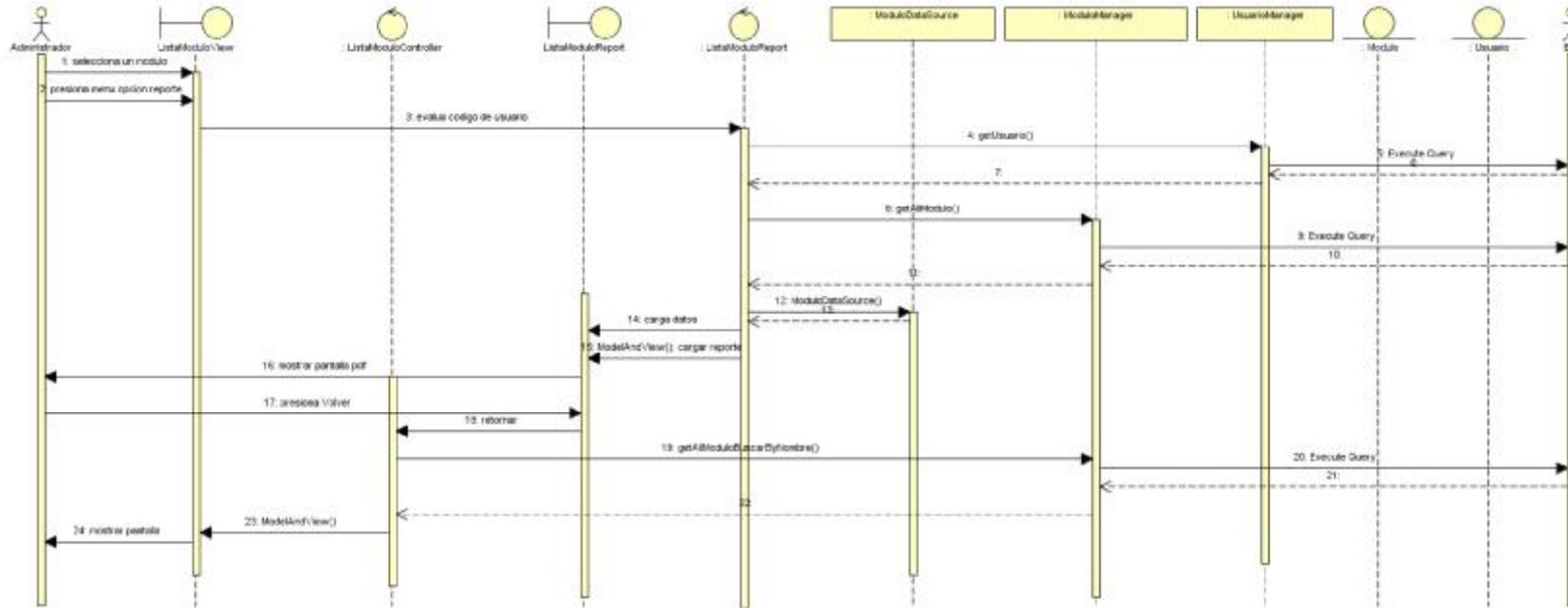


Figura 314. Diagrama de Secuencia: CU Generar Reporte Lista Módulo

2.1.2.2.3.9.2.1.22 Diagrama de Secuencia: Caso de Uso Administrar Municipio Localidad

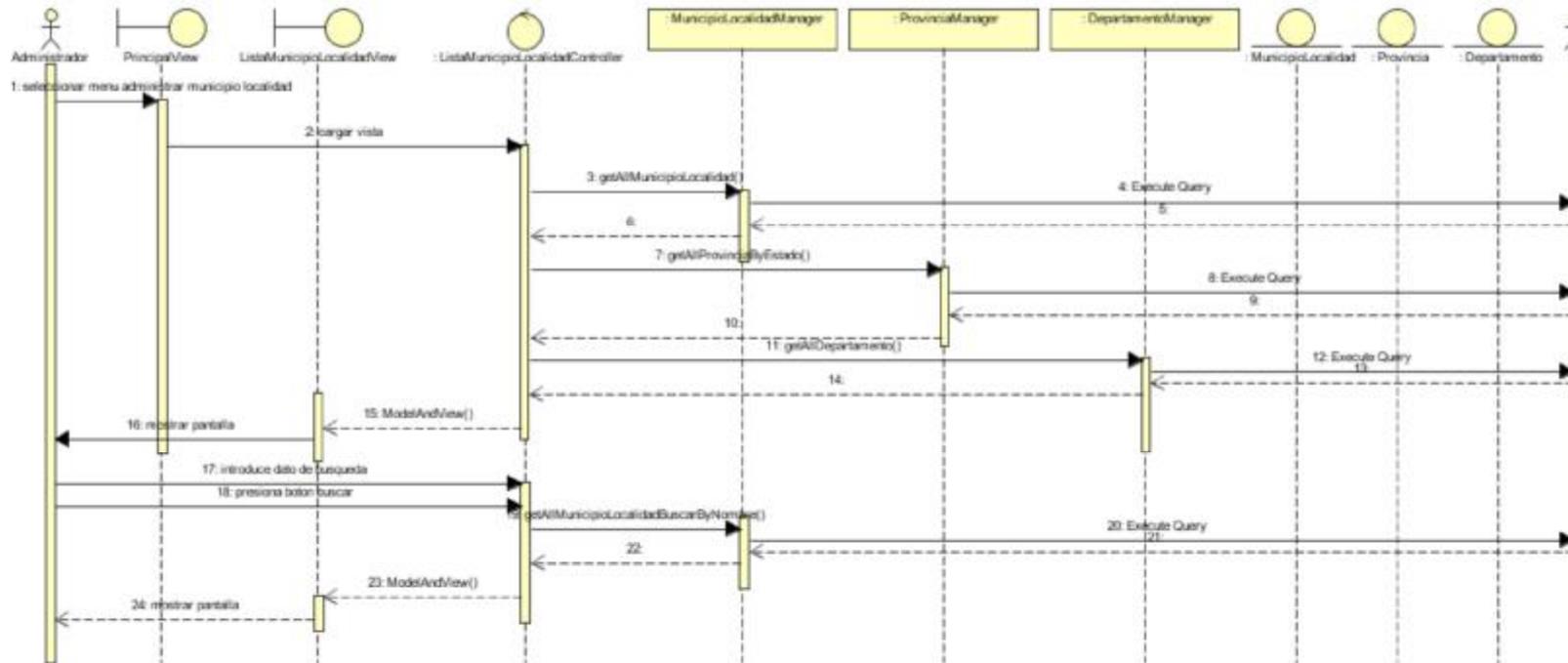


Figura 315. Diagrama de Secuencia: CU Administrar Municipio Localidad

2.1.2.2.3.9.2.1.23 Diagrama de Secuencia: Caso de Uso Adicionar Municipio Localidad

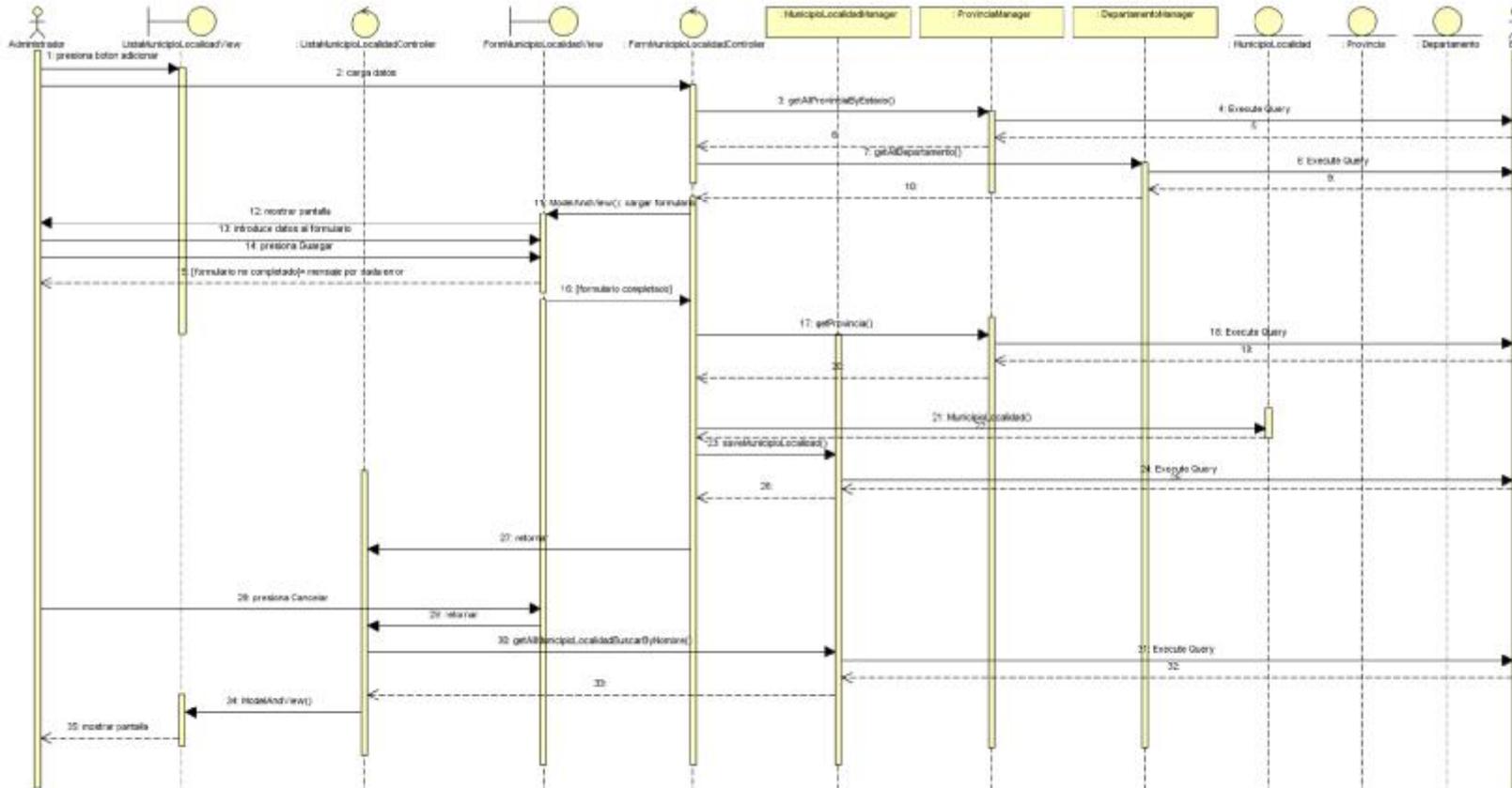


Figura 316. Diagrama de Secuencia: CU Adicionar Municipio Localidad

2.1.2.2.3.9.2.1.24 Diagrama de Secuencia: Caso de Uso Modificar Municipio Localidad

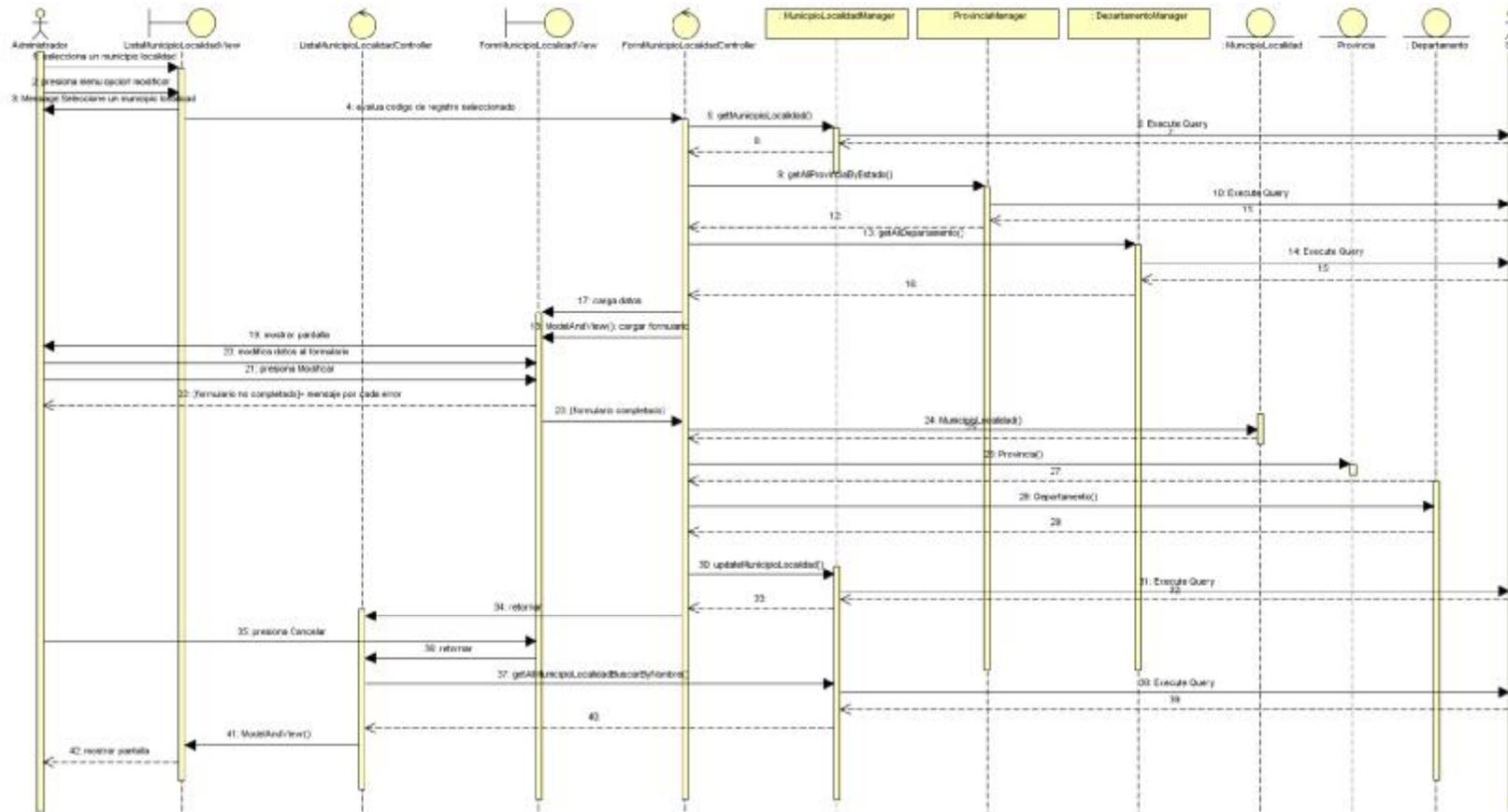


Figura 317. Diagrama de Secuencia: CU Modificar Municipio Localidad

2.1.2.2.3.9.2.1.25 Diagrama de Secuencia: Caso de Uso Eliminar Municipio Localidad

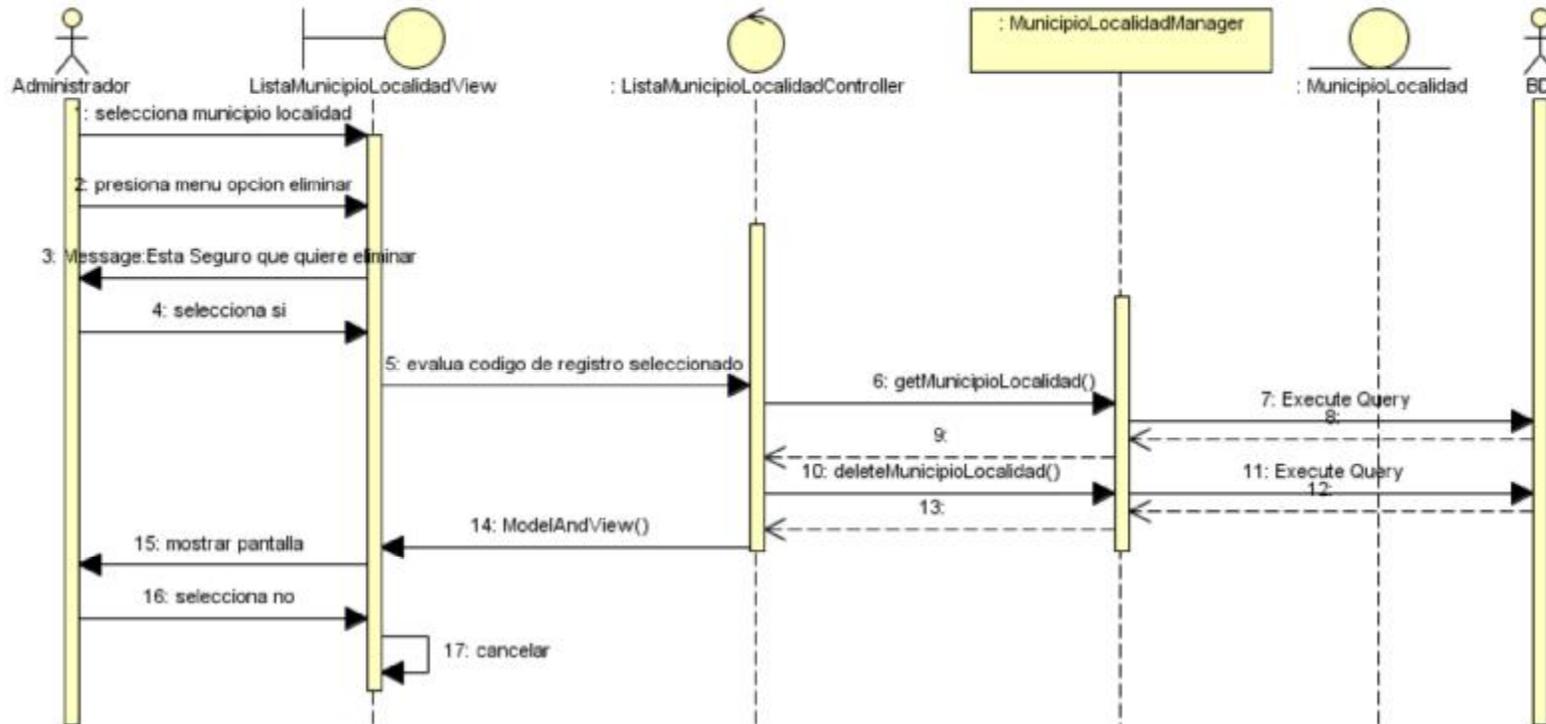


Figura 318. Diagrama de Secuencia: CU Eliminar Municipio Localidad

2.1.2.2.3.9.2.1.26 Diagrama de Secuencia: Caso de Uso Cambiar Estado Municipio Localidad

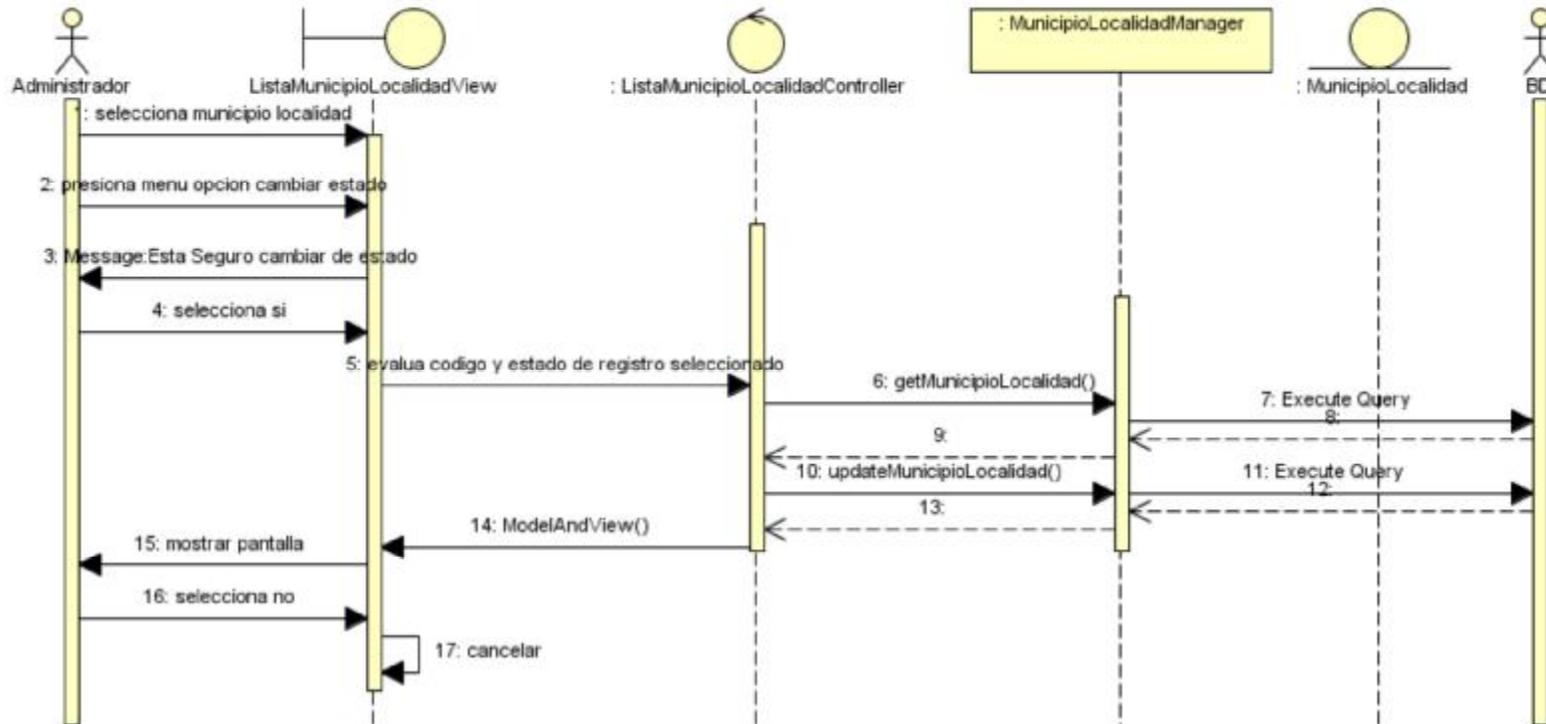


Figura 319. Diagrama de Secuencia: CU Cambiar Estado Municipio Localidad

2.1.2.2.3.9.2.1.27 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Municipio Localidad

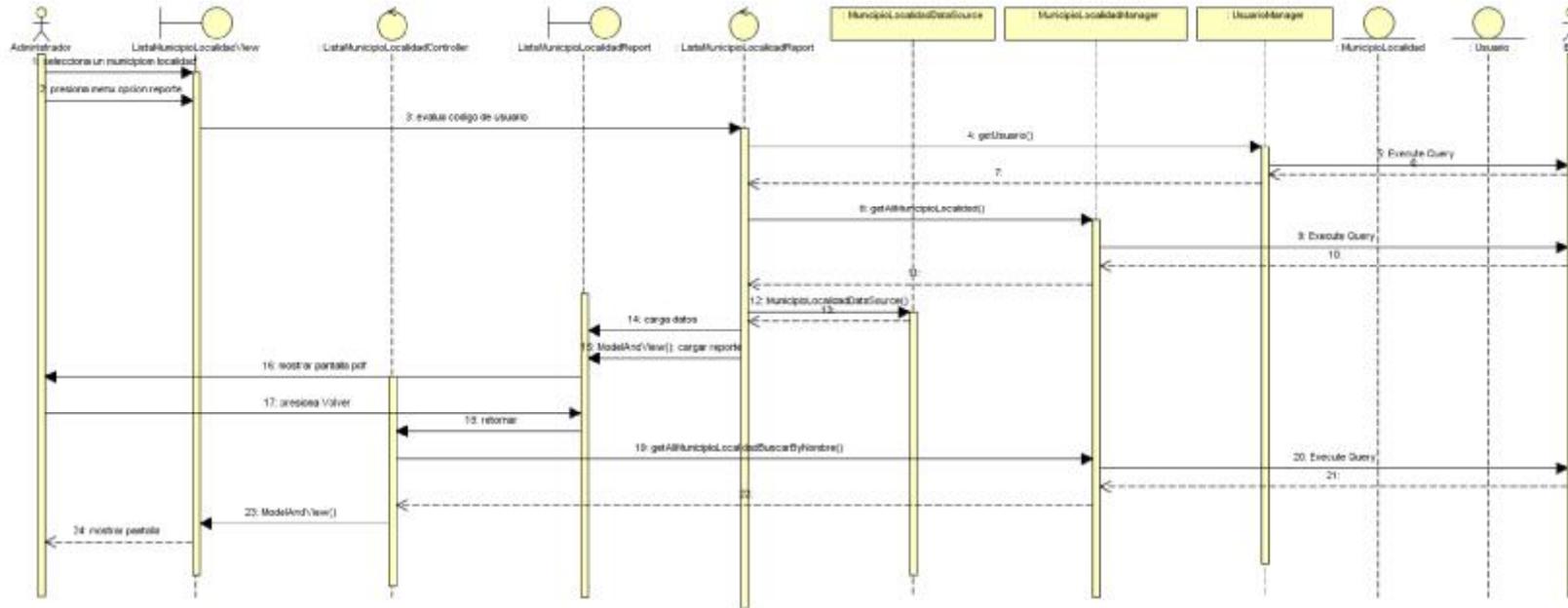


Figura 320. Diagrama de Secuencia: CU Generar Reporte Lista Municipio Localidad

2.1.2.2.3.9.2.1.28 Diagrama de Secuencia: Caso de Uso Administrar Ocupación

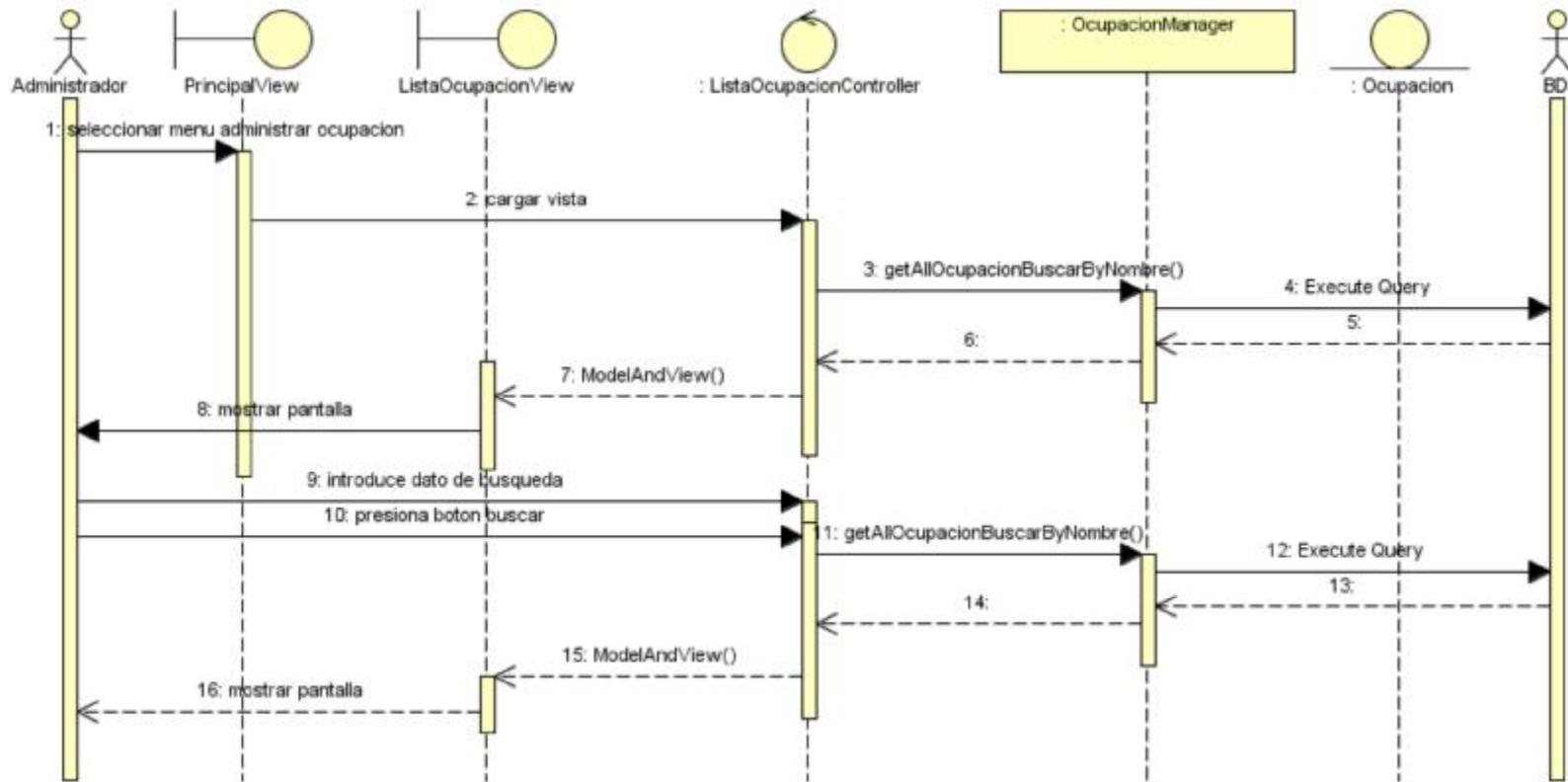


Figura 321. Diagrama de Secuencia: CU Administrar Ocupación

2.1.2.2.3.9.2.1.29 Diagrama de Secuencia: Caso de Uso Adicionar Ocupación

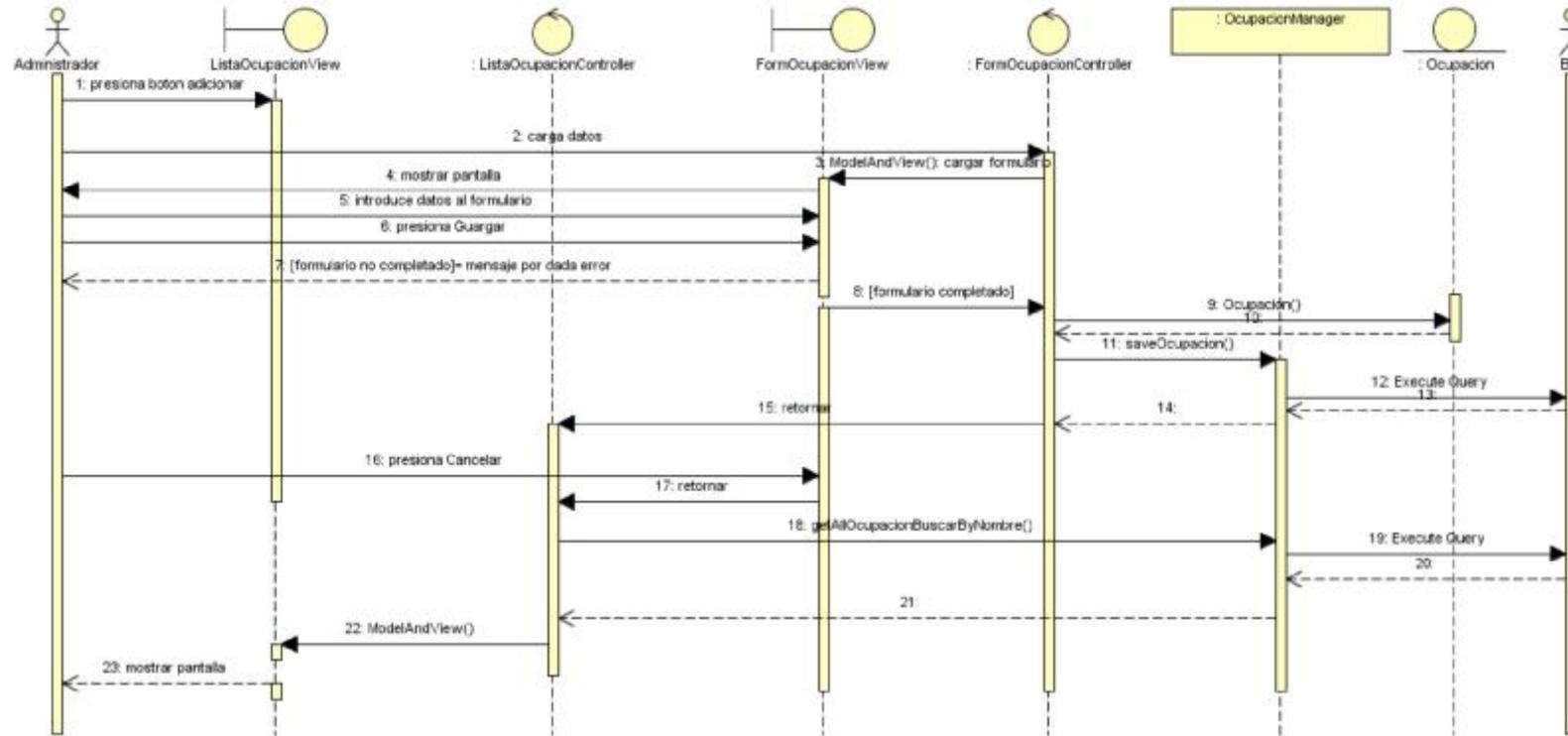


Figura 322. Diagrama de Secuencia: CU Adicionar Ocupación

2.1.2.2.3.9.2.1.30 Diagrama de Secuencia: Caso de Uso Modificar Ocupación

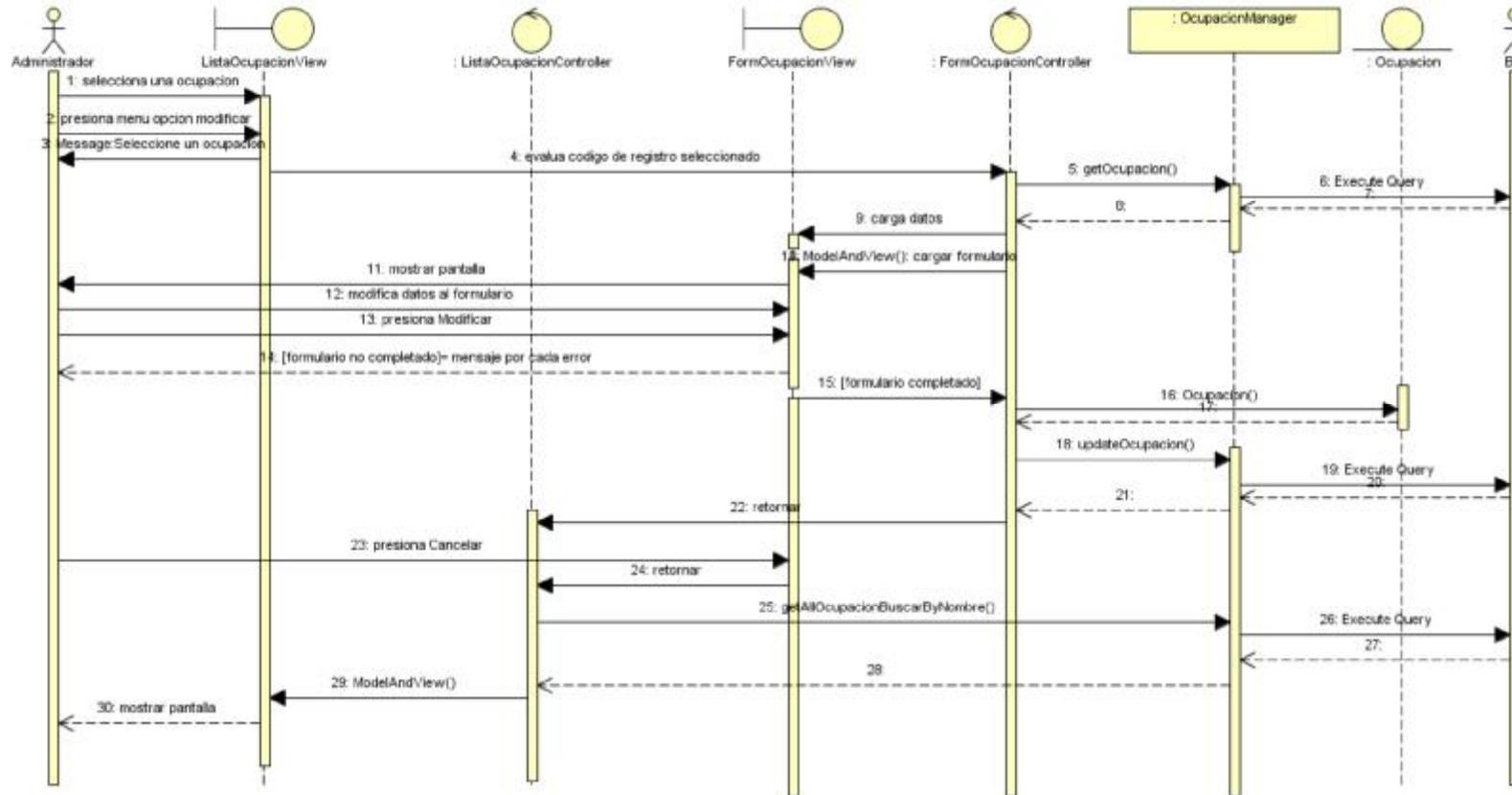


Figura 323. Diagrama de Secuencia: CU Modificar Ocupación

2.1.2.2.3.9.2.1.31 Diagrama de Secuencia: Caso de Uso Eliminar Ocupación

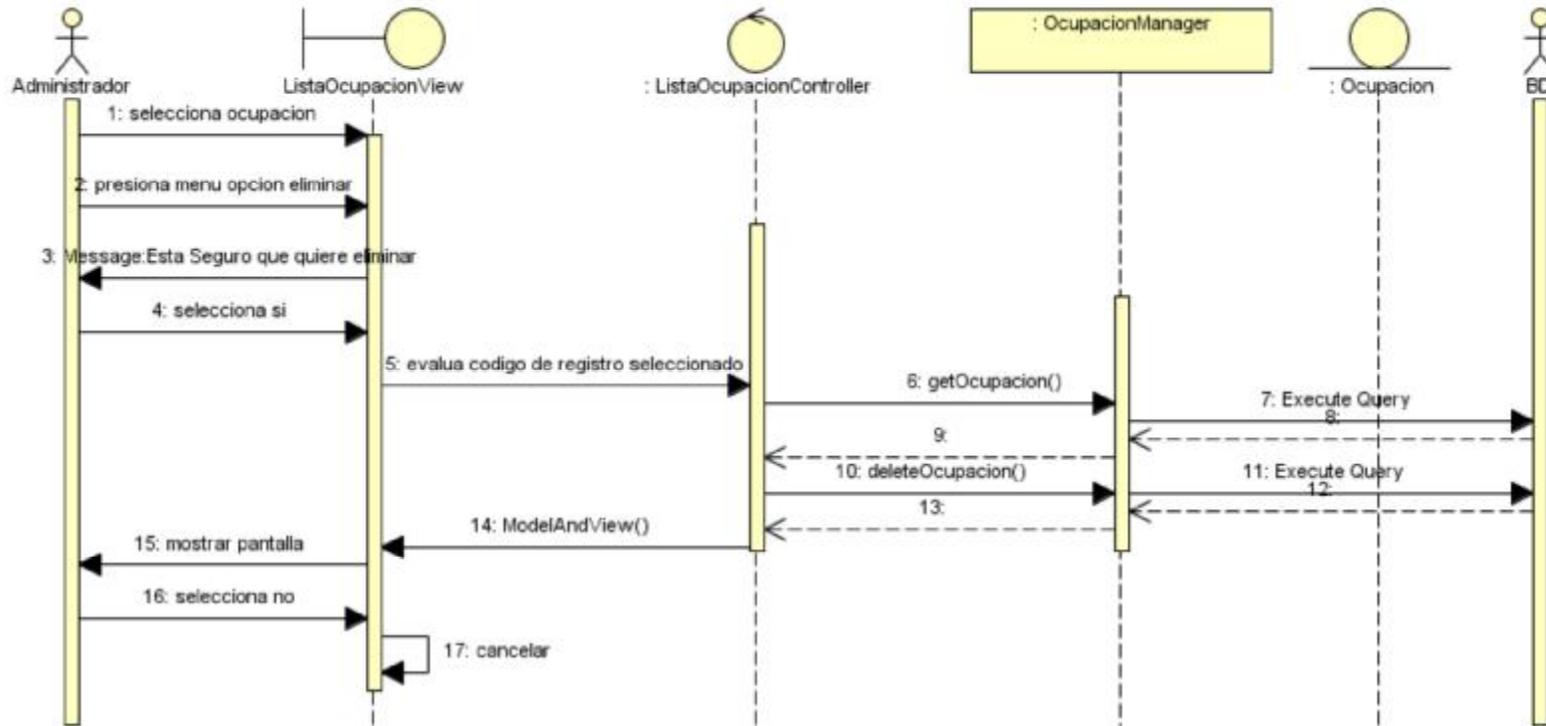


Figura 324. Diagrama de Secuencia: CU Eliminar Ocupación

2.1.2.2.3.9.2.1.32 Diagrama de Secuencia: Caso de Uso Cambiar Estado Ocupación

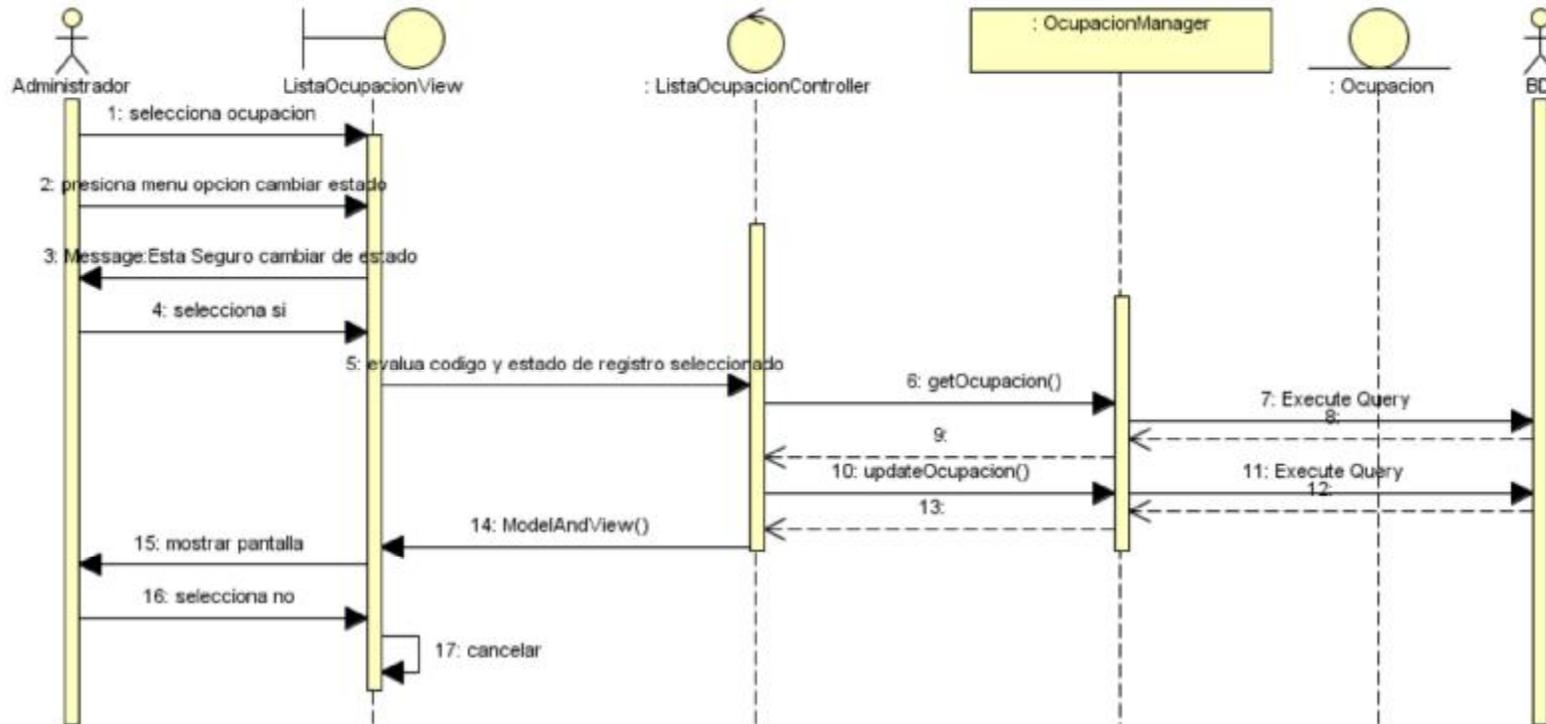


Figura 325. Diagrama de Secuencia: CU Cambiar Estado Ocupación

2.1.2.2.3.9.2.1.33 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Ocupación

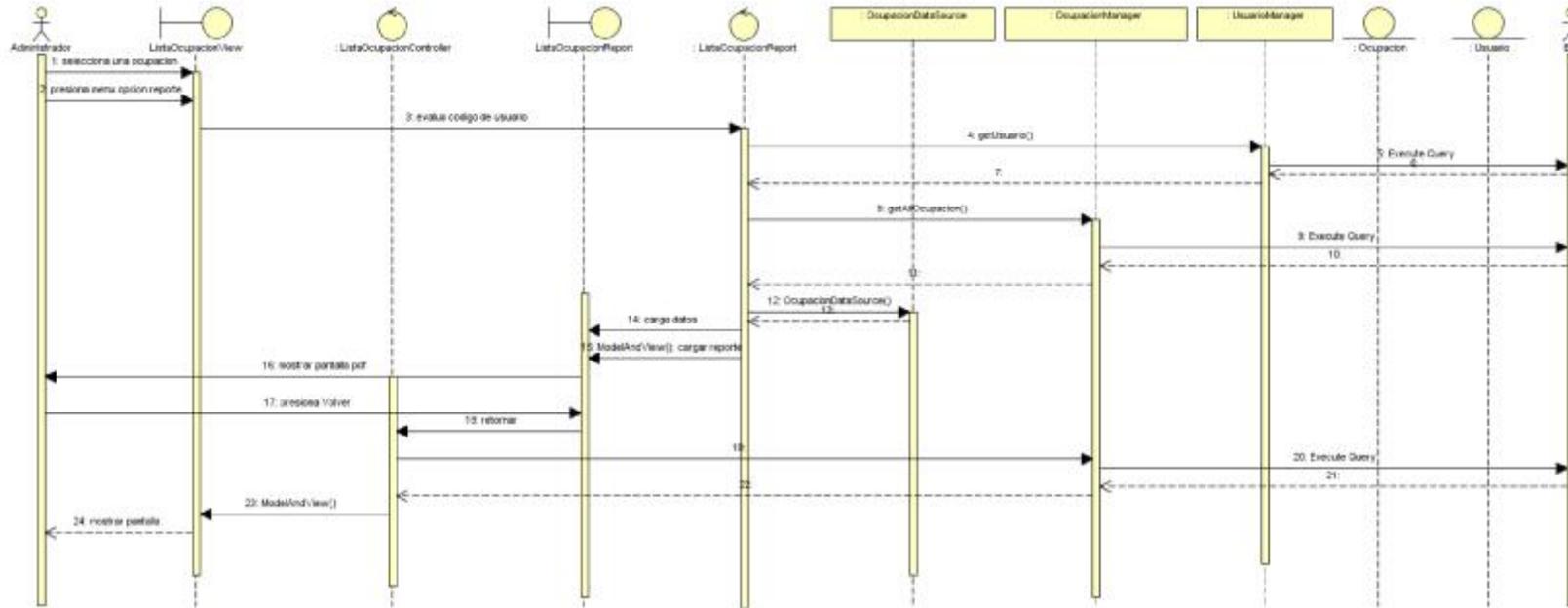


Figura 326. Diagrama de Secuencia: CU Generar Reporte Lista Ocupación

2.1.2.2.3.9.2.1.34 Diagrama de Secuencia: Caso de Uso Administrar Parentesco

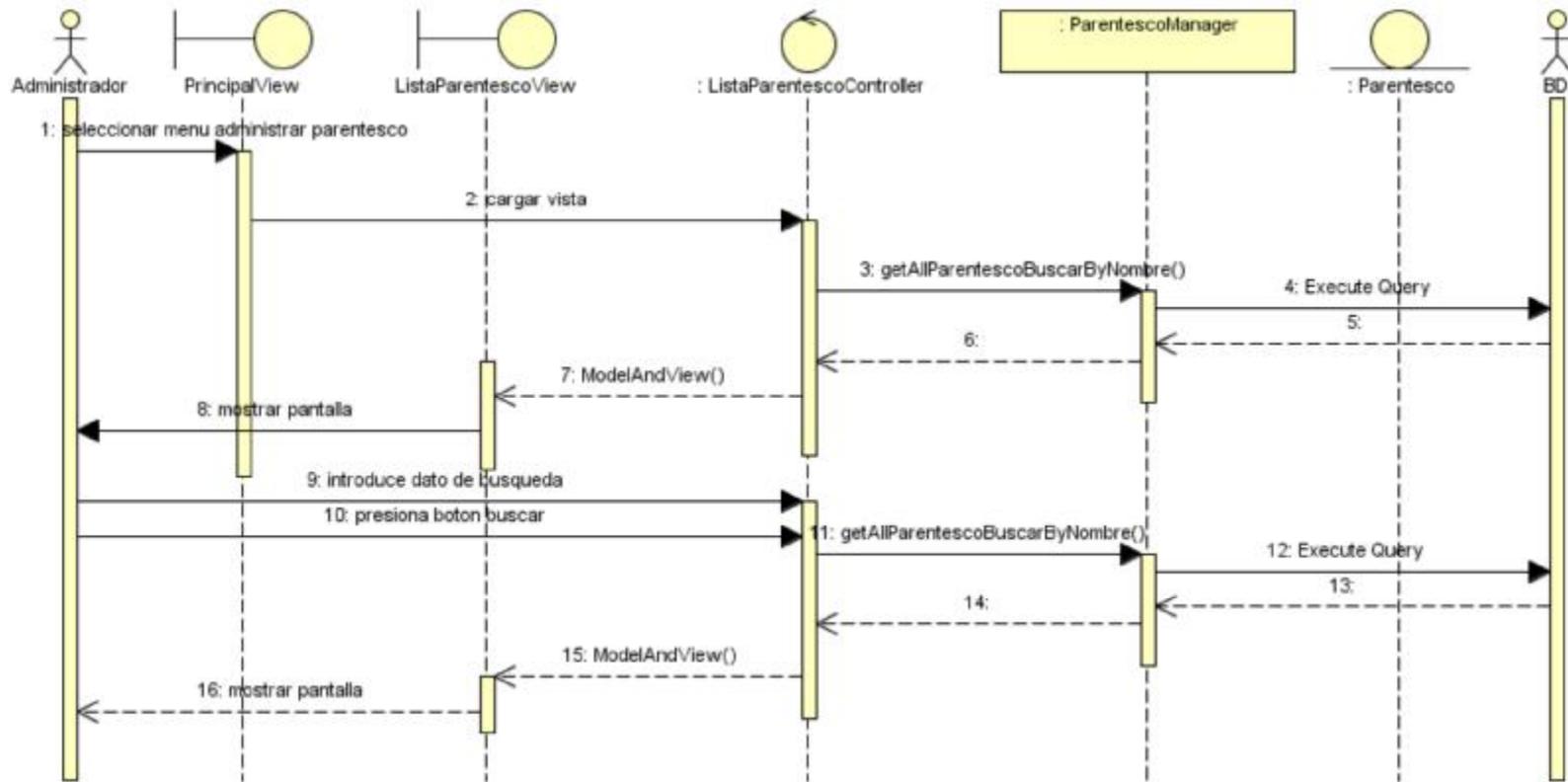


Figura 327. Diagrama de Secuencia: CU Administrar Parentesco

2.1.2.2.3.9.2.1.35 Diagrama de Secuencia: Caso de Uso Adicionar Parentesco

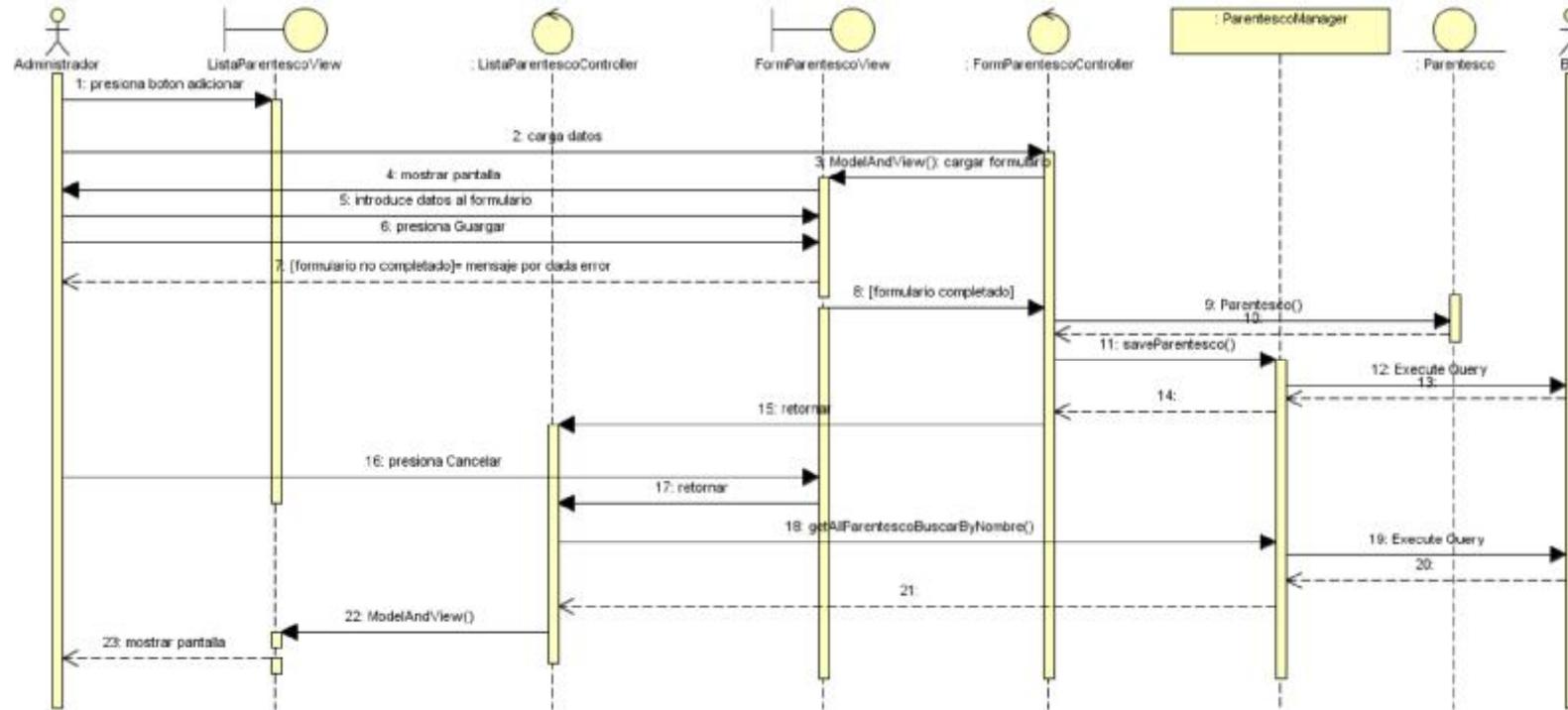


Figura 328. Diagrama de Secuencia: CU Adicionar Parentesco

2.1.2.2.3.9.2.1.36 Diagrama de Secuencia: Caso de Uso Modificar Parentesco

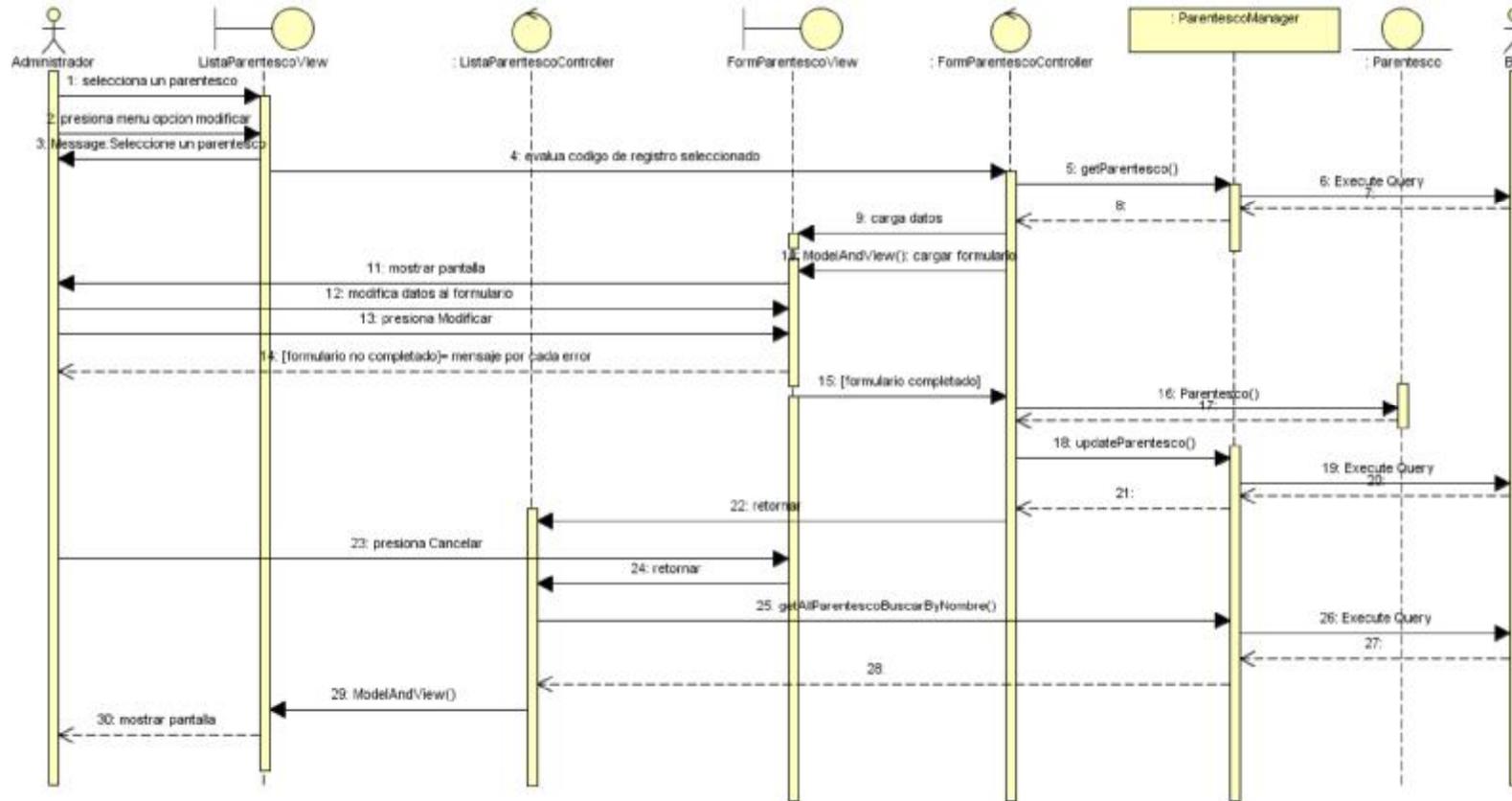


Figura 329. Diagrama de Secuencia: CU Modificar Parentesco

2.1.2.2.3.9.2.1.37 Diagrama de Secuencia: Caso de Uso Eliminar Parentesco

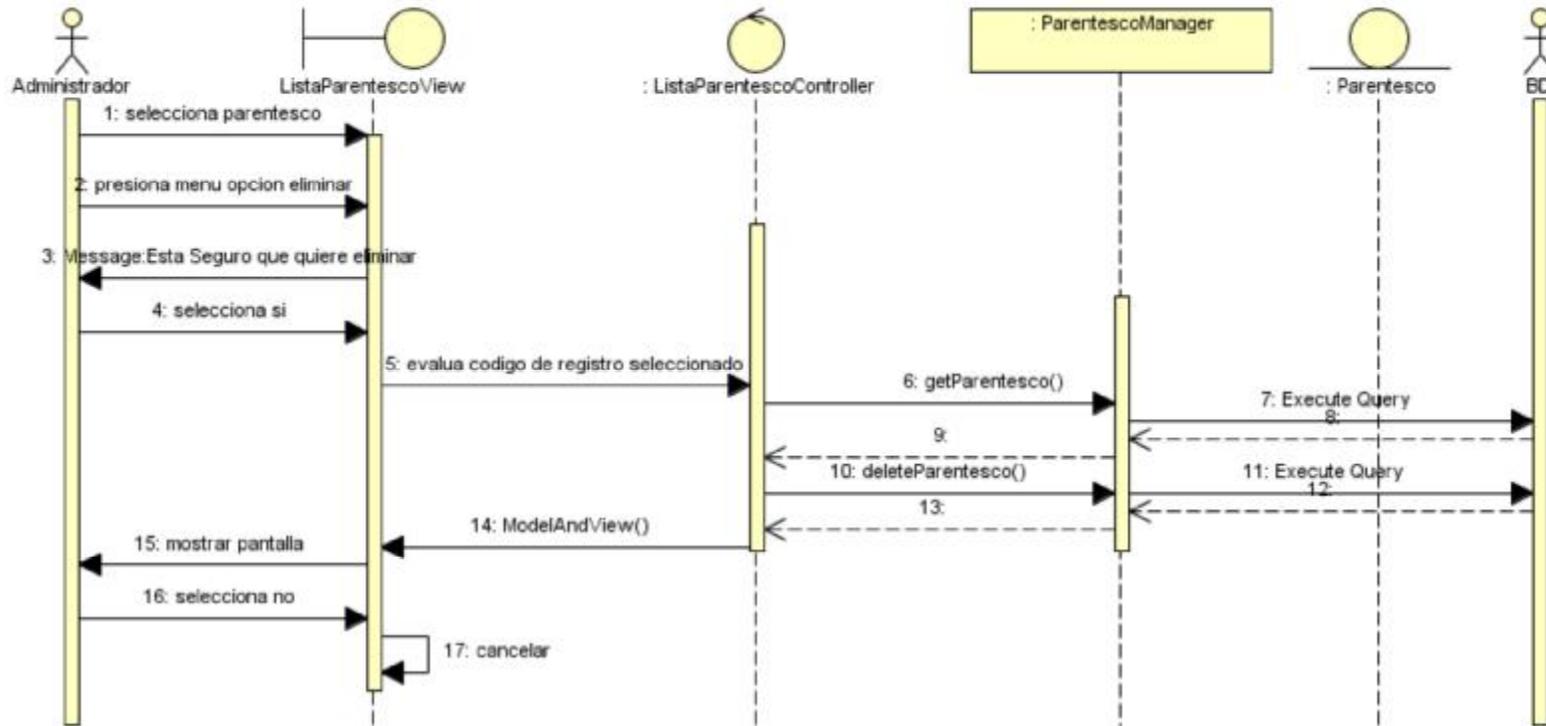


Figura 330. Diagrama de Secuencia: CU Eliminar Parentesco

2.1.2.2.3.9.2.1.38 Diagrama de Secuencia: Caso de Uso Cambiar Estado Parentesco

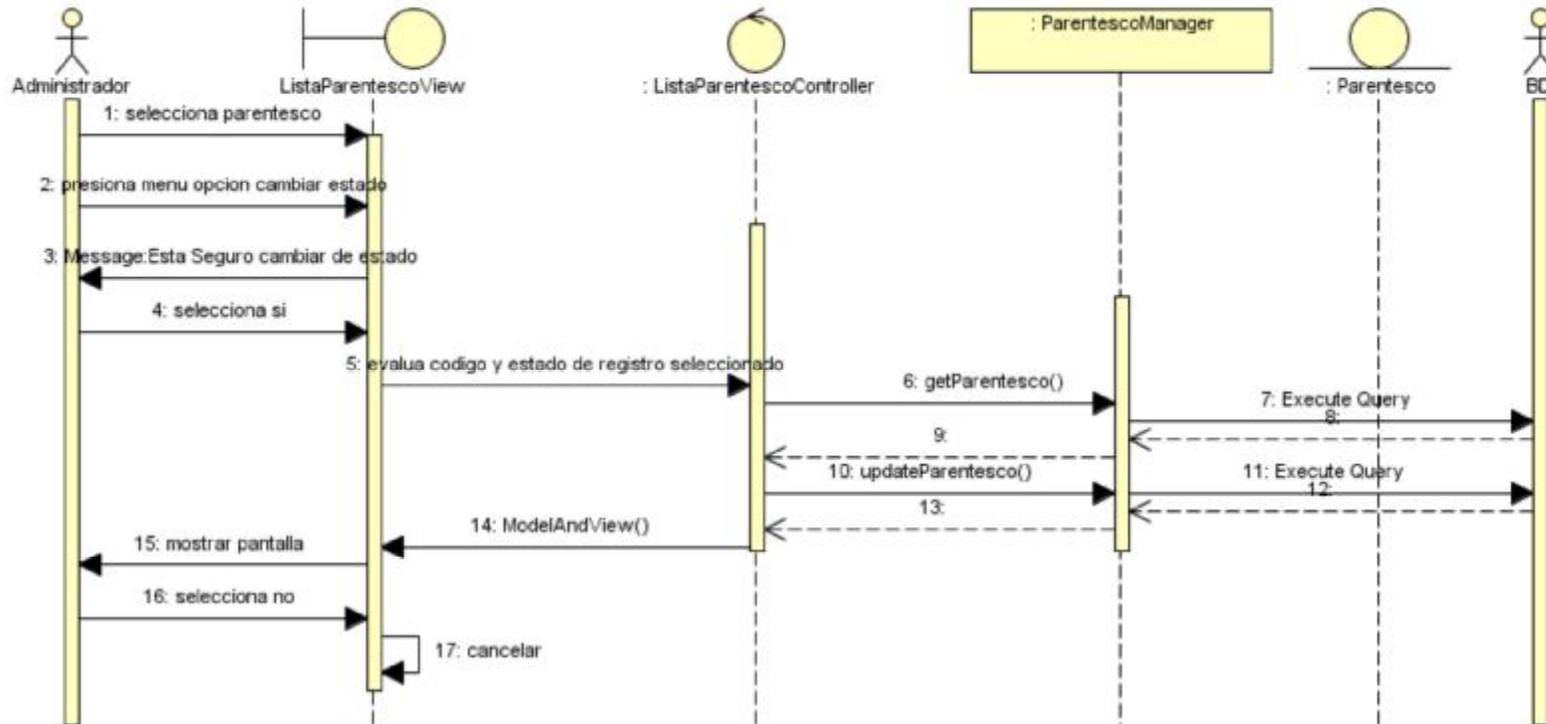


Figura 331. Diagrama de Secuencia: CU Cambiar Estado Parentesco

2.1.2.2.3.9.2.1.39 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Parentesco

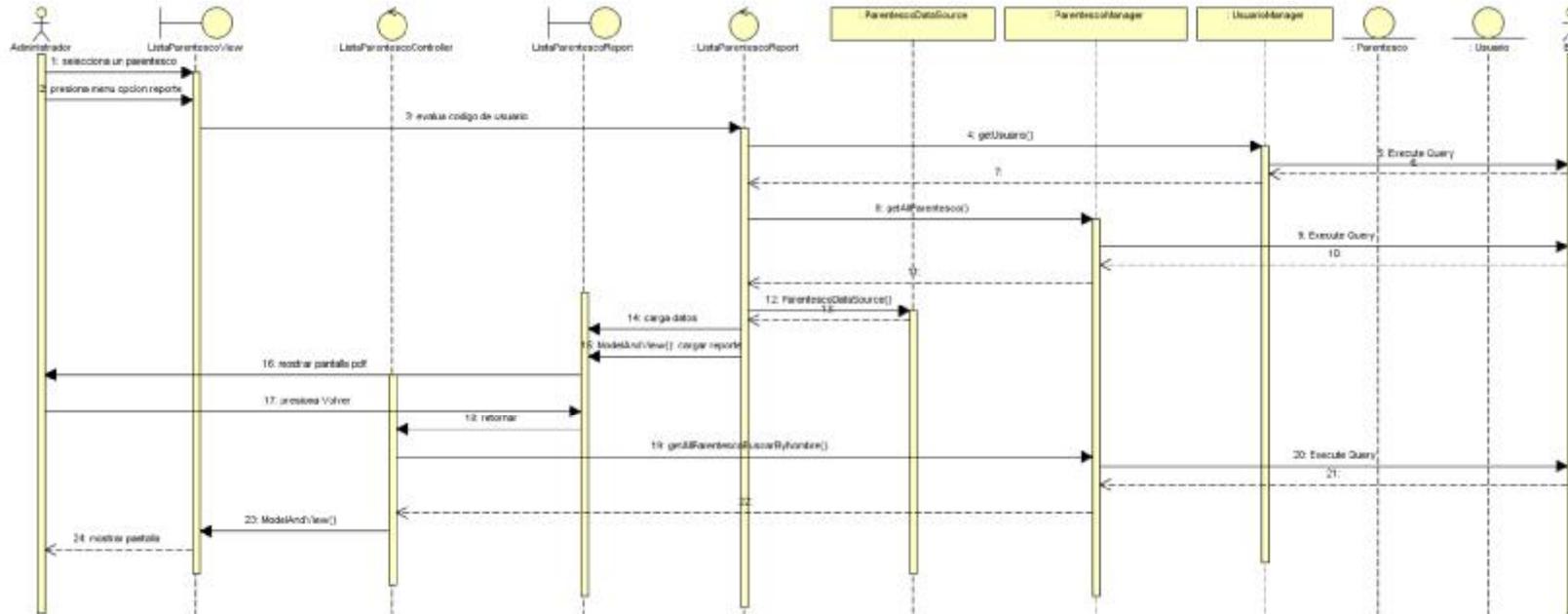


Figura 332. Diagrama de Secuencia: CU Generar Reporte Lista Parentesco

2.1.2.2.3.9.2.1.40 Diagrama de Secuencia: Caso de Uso Administrar Provincia

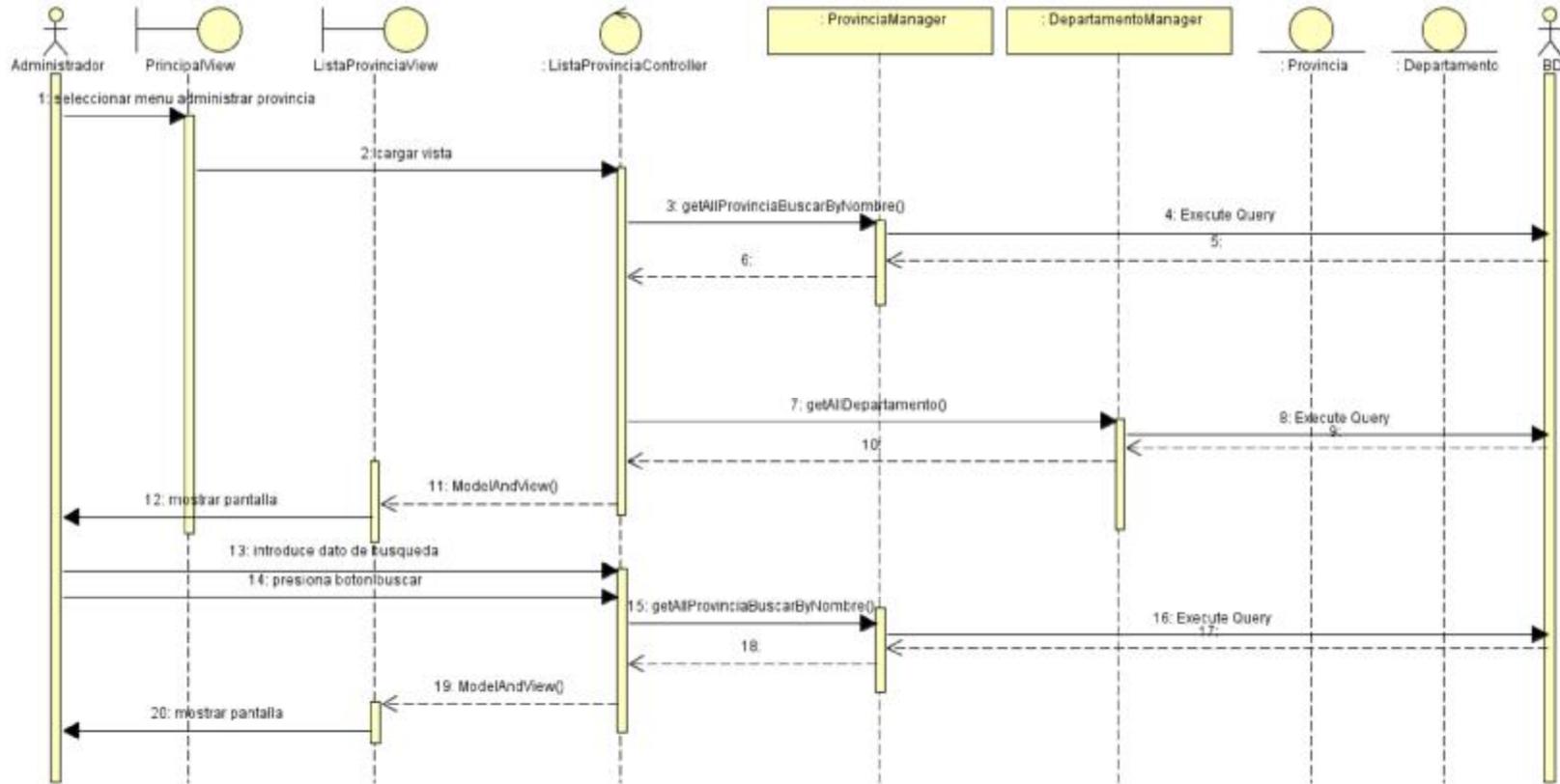


Figura 333. Diagrama de Secuencia: CU Administrar Provincia

2.1.2.2.3.9.2.1.41 Diagrama de Secuencia: Caso de Uso Adicionar Provincia

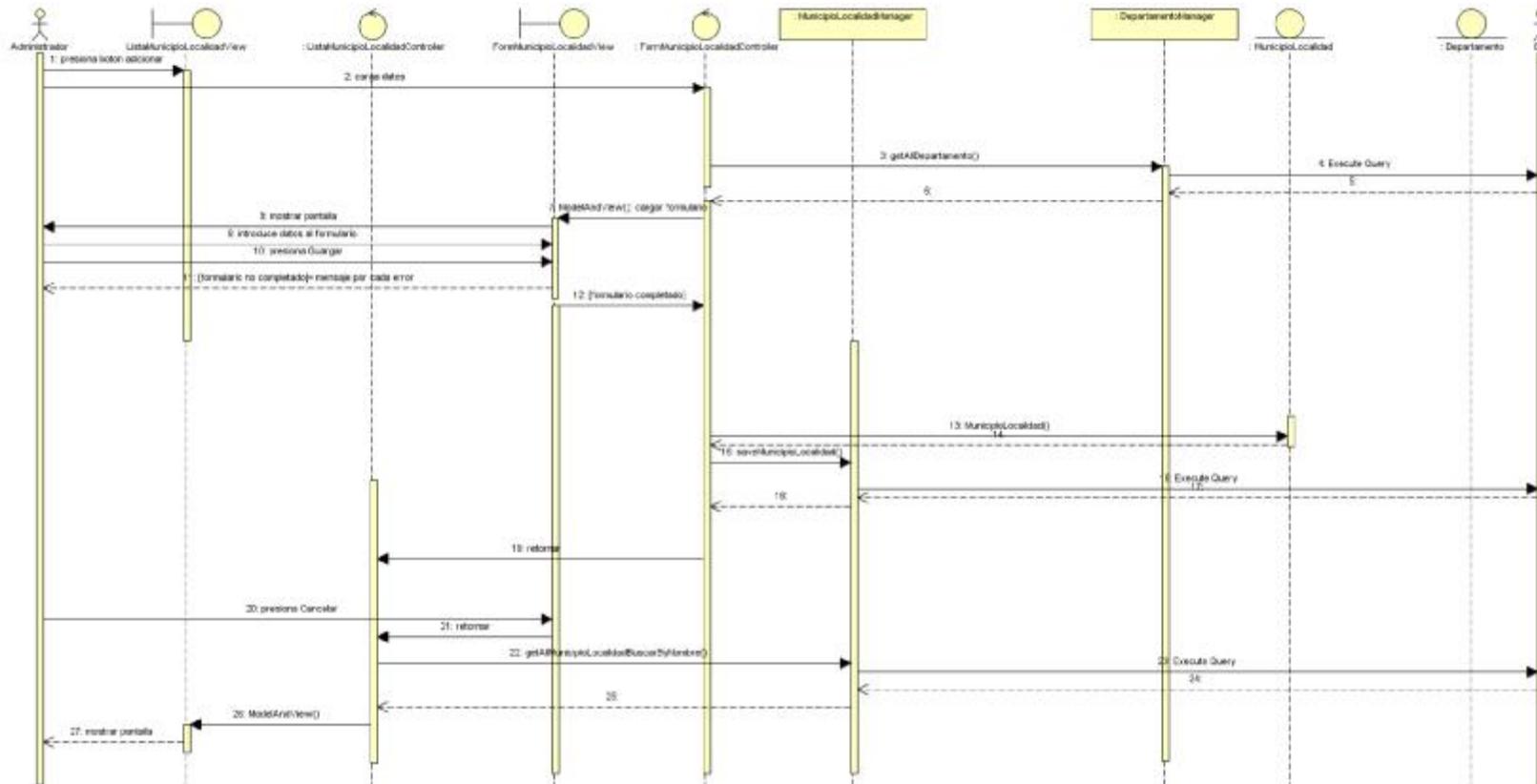


Figura 334. Diagrama de Secuencia: CU Adicionar Provincia

2.1.2.2.3.9.2.1.42 Diagrama de Secuencia: Caso de Uso Modificar Provincia

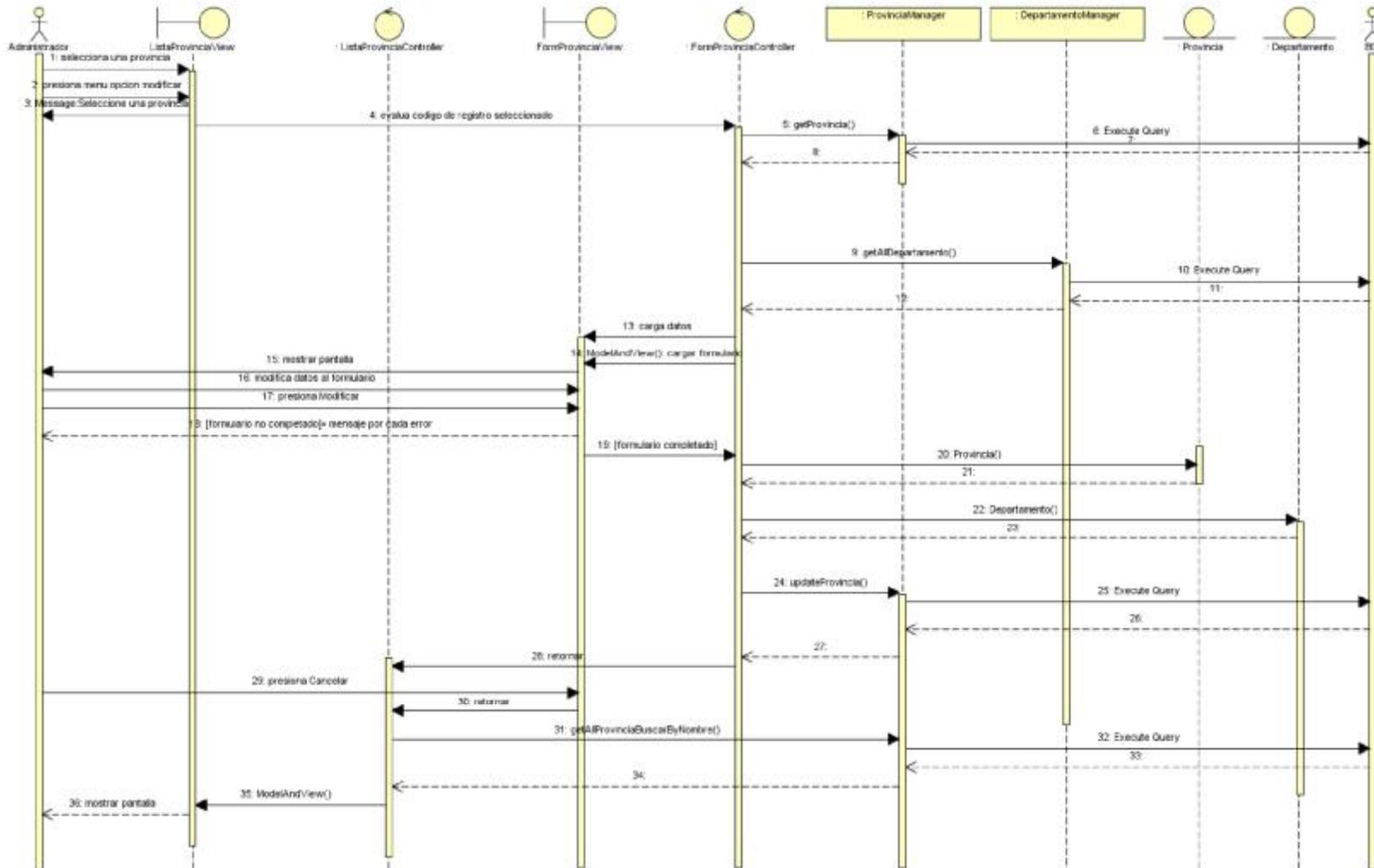


Figura 335. Diagrama de Secuencia: CU Modificar Provincia

2.1.2.2.3.9.2.1.43 Diagrama de Secuencia: Caso de Uso Eliminar Provincia

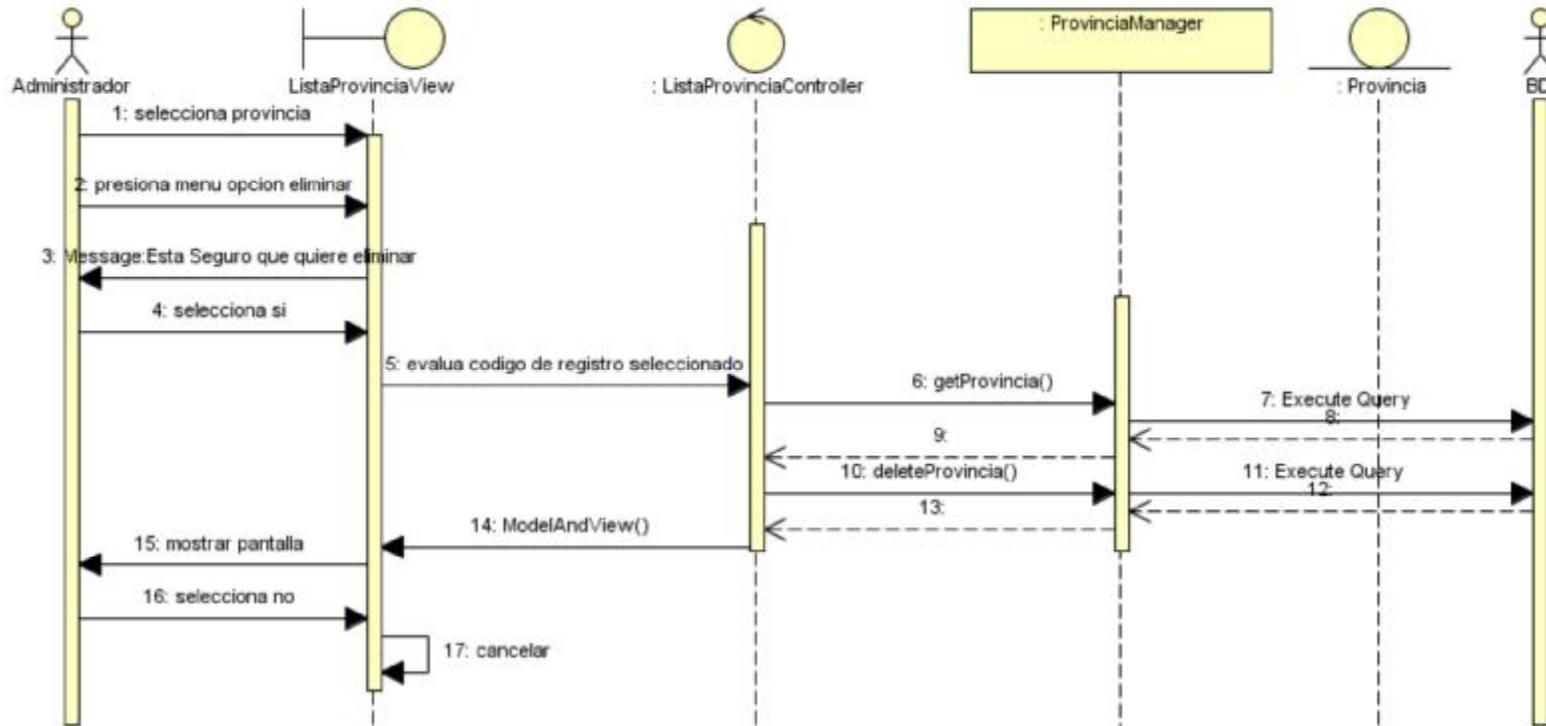


Figura 336. Diagrama de Secuencia: CU Eliminar Provincia

2.1.2.2.3.9.2.1.44 Diagrama de Secuencia: Caso de Uso Cambiar Estado Provincia

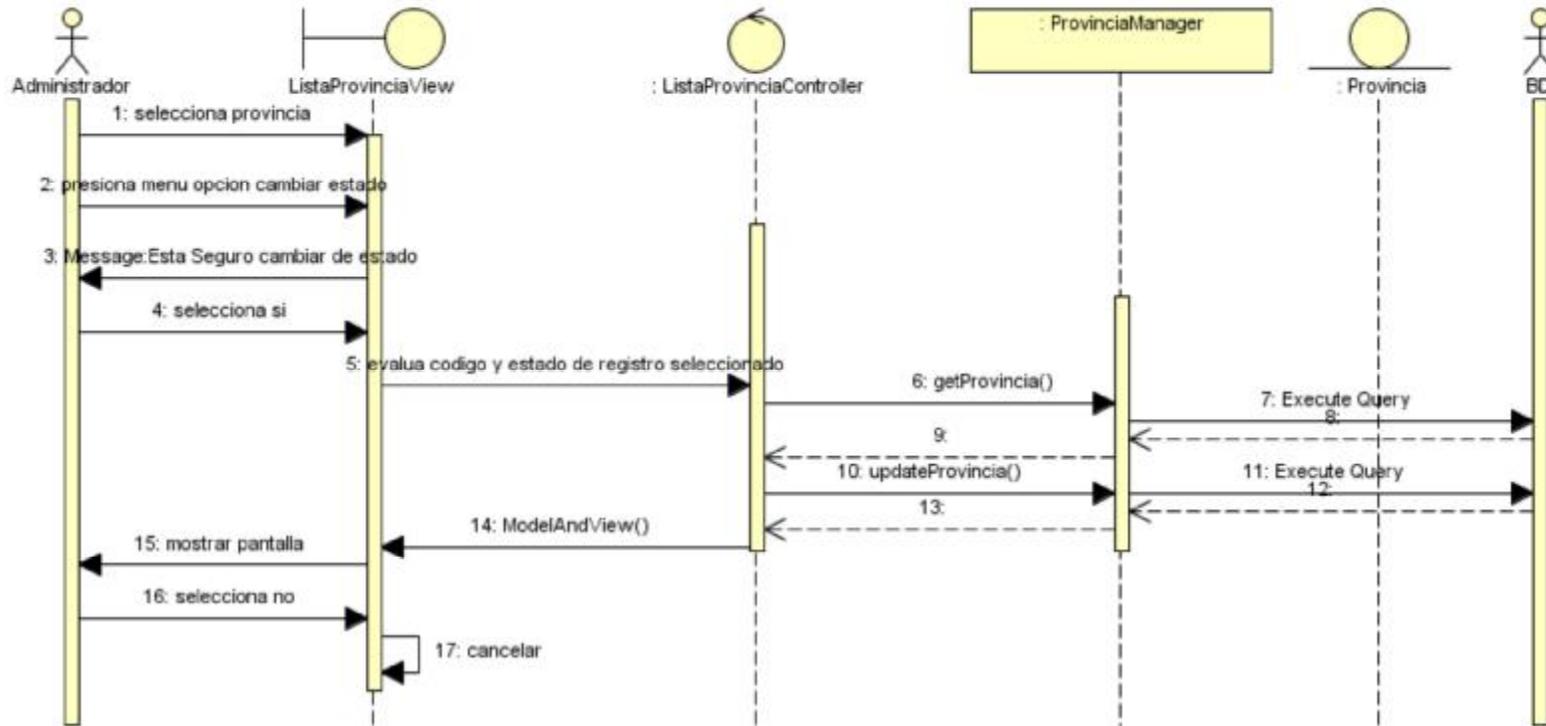


Figura 337. Diagrama de Secuencia: CU Cambiar Estado Provincia

2.1.2.2.3.9.2.1.45 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Provincia

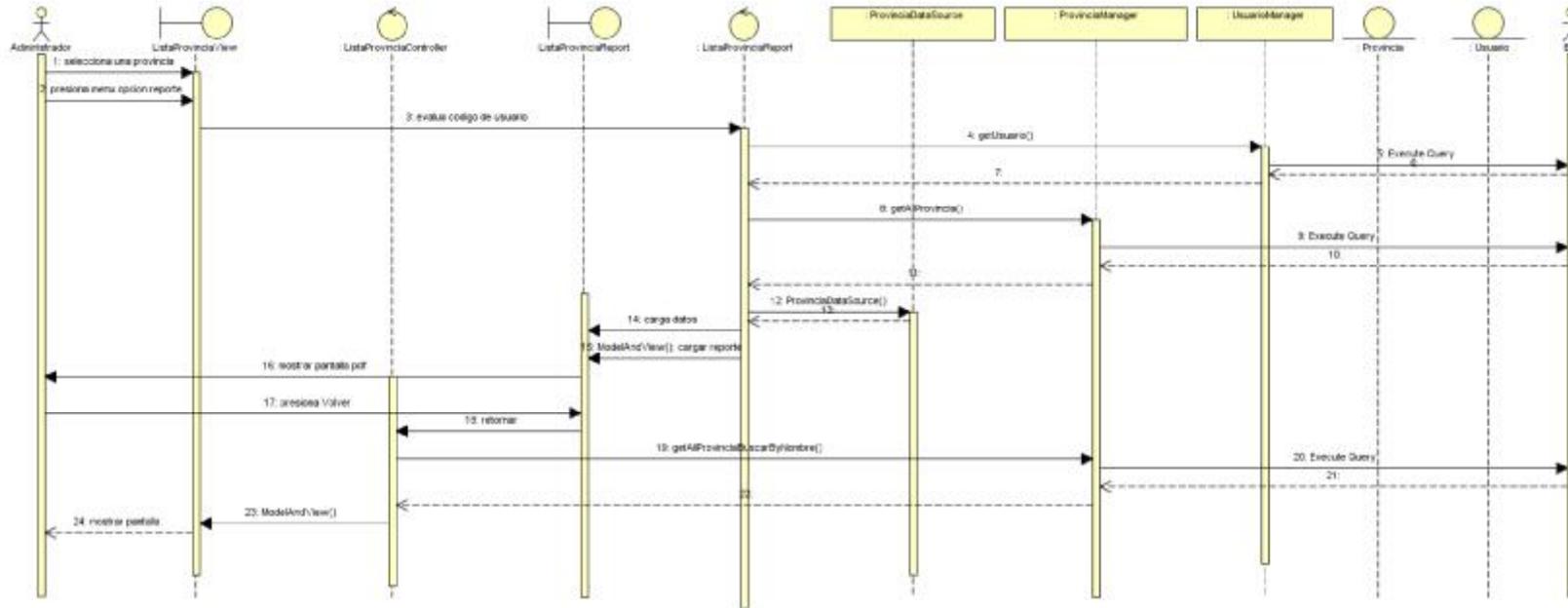


Figura 338. Diagrama de Secuencia: CU Generar Reporte Lista Provincia

2.1.2.2.3.9.2.1.46 Diagrama de Secuencia: Caso de Uso Administrar Requisito

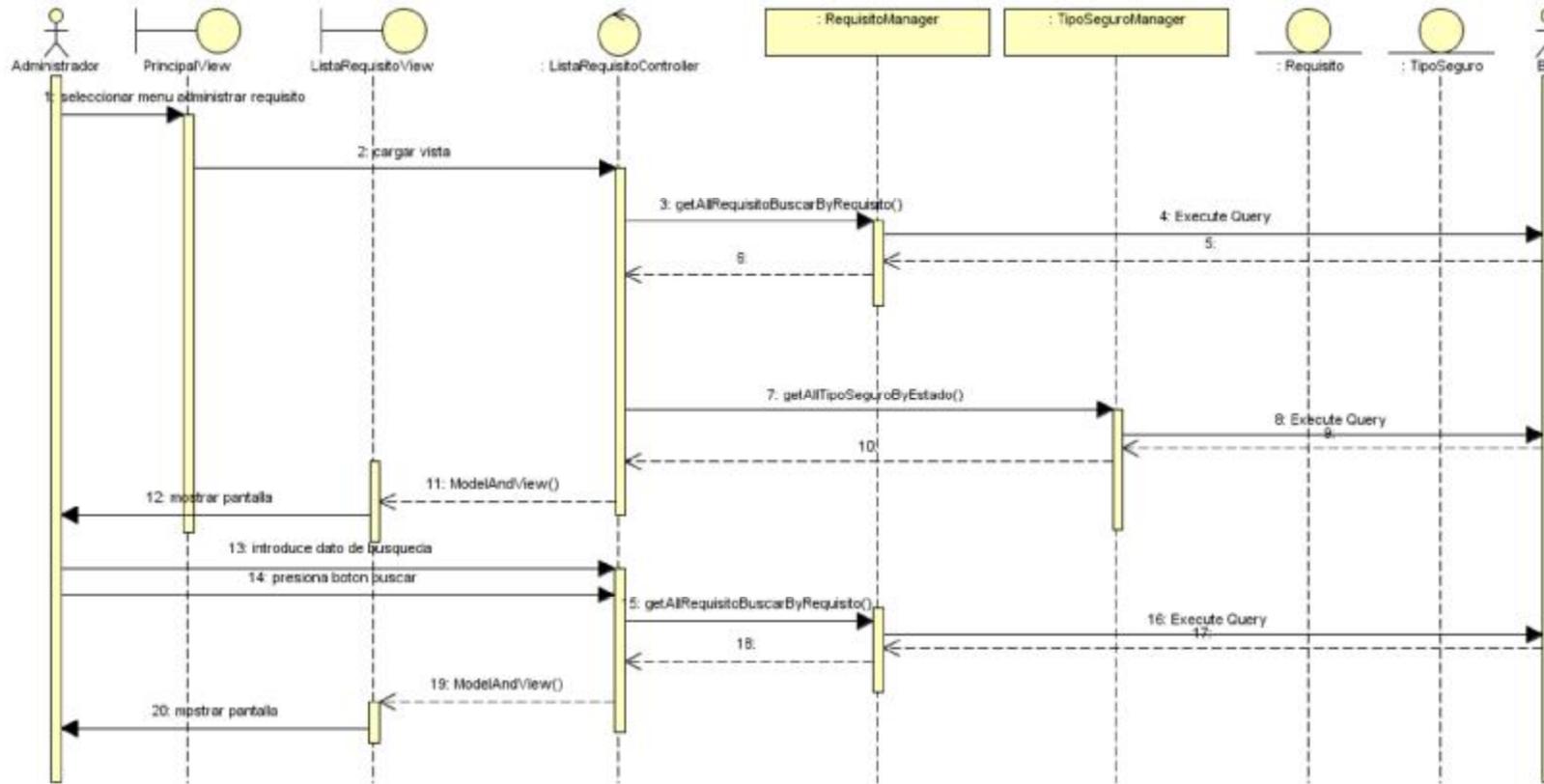


Figura 339. Diagrama de Secuencia: CU Administrar Requisito

2.1.2.2.3.9.2.1.47 Diagrama de Secuencia: Caso de Uso Adicionar Requisito

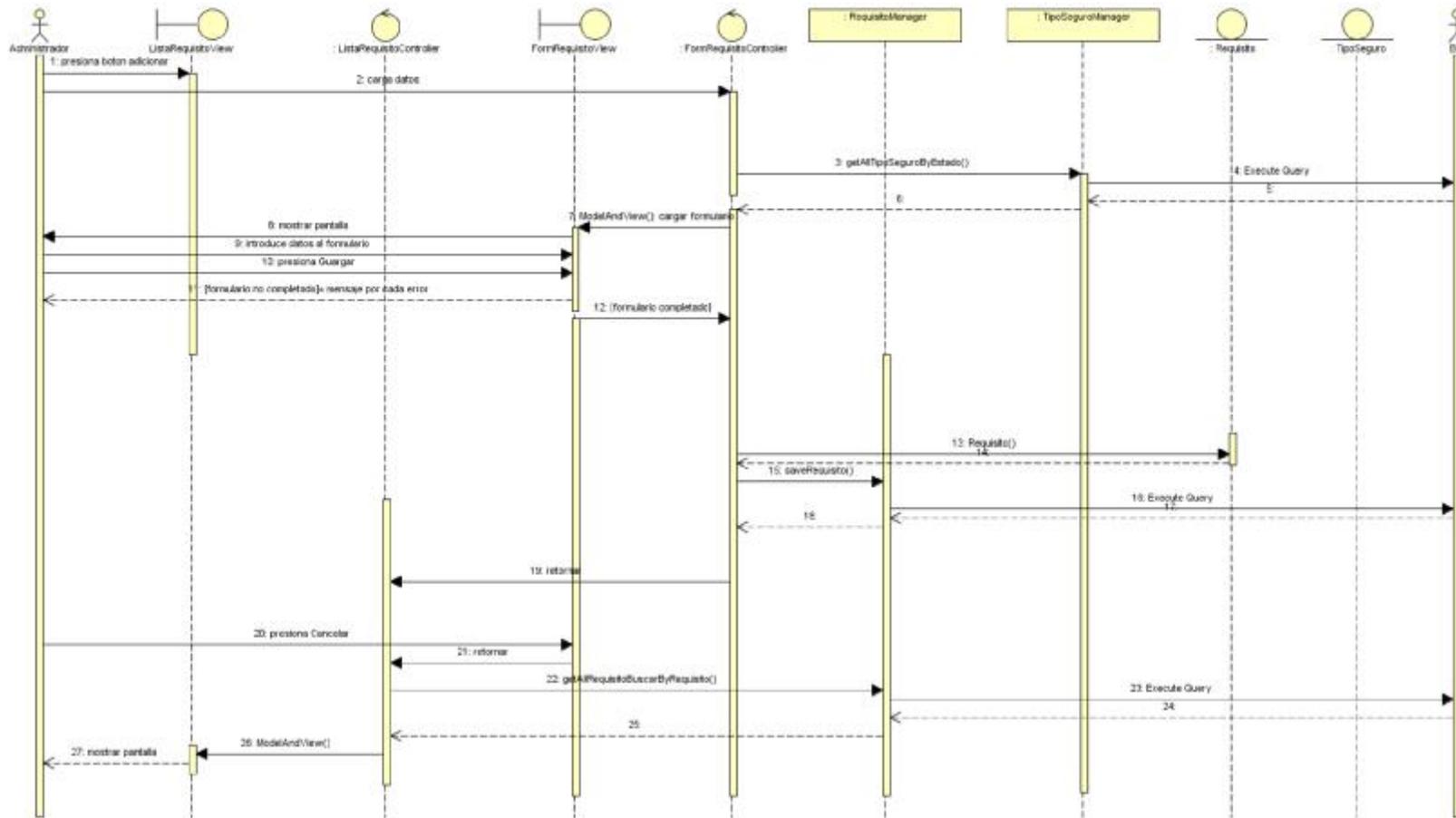


Figura 340. Diagrama de Secuencia: CU Adicionar Requisito

2.1.2.2.3.9.2.1.48 Diagrama de Secuencia: Caso de Uso Modificar Requisito

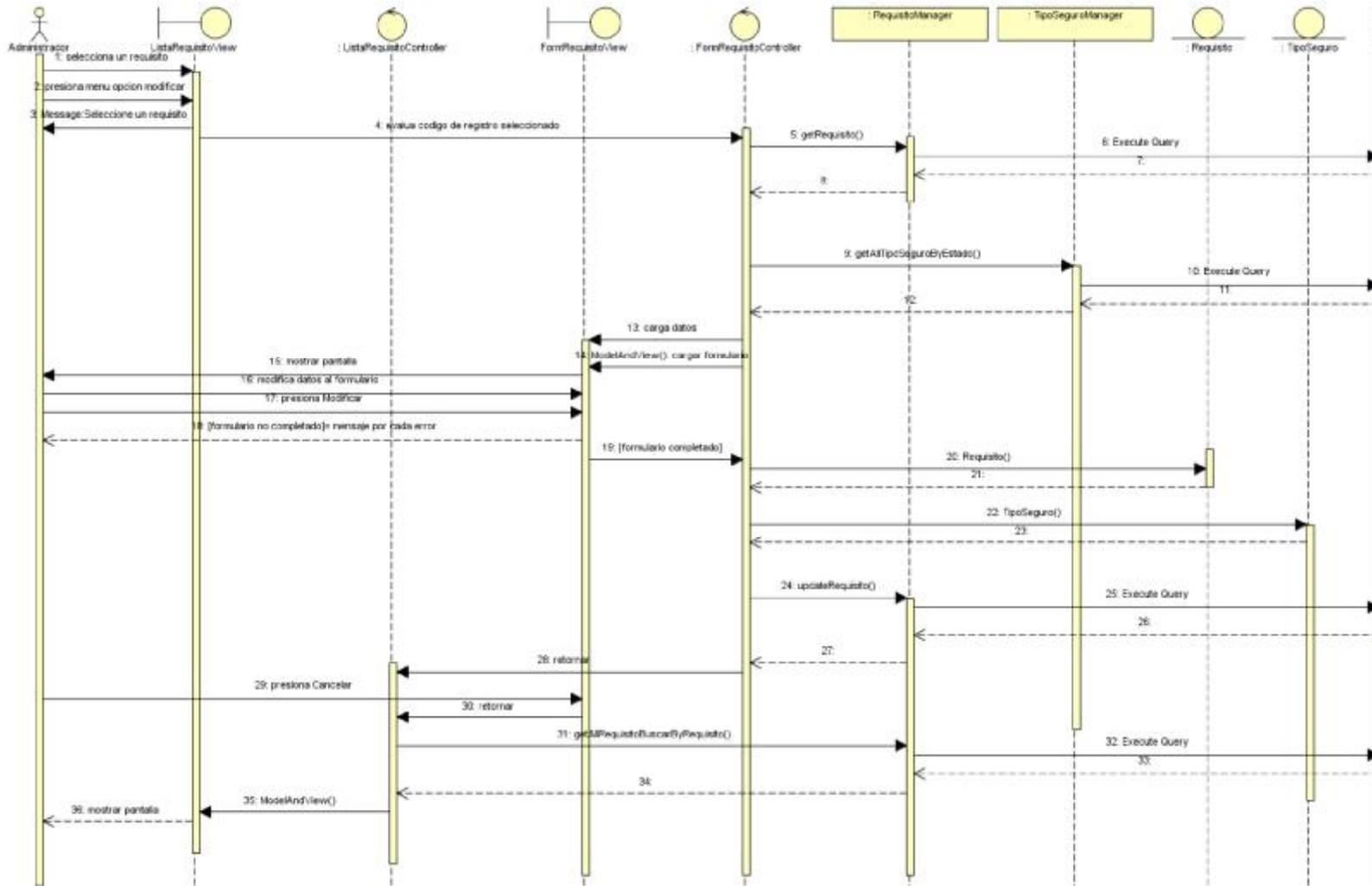


Figura 341. Diagrama de Secuencia: CU Modificar Requisito

2.1.2.2.3.9.2.1.49 Diagrama de Secuencia: Caso de Uso Eliminar Requisito

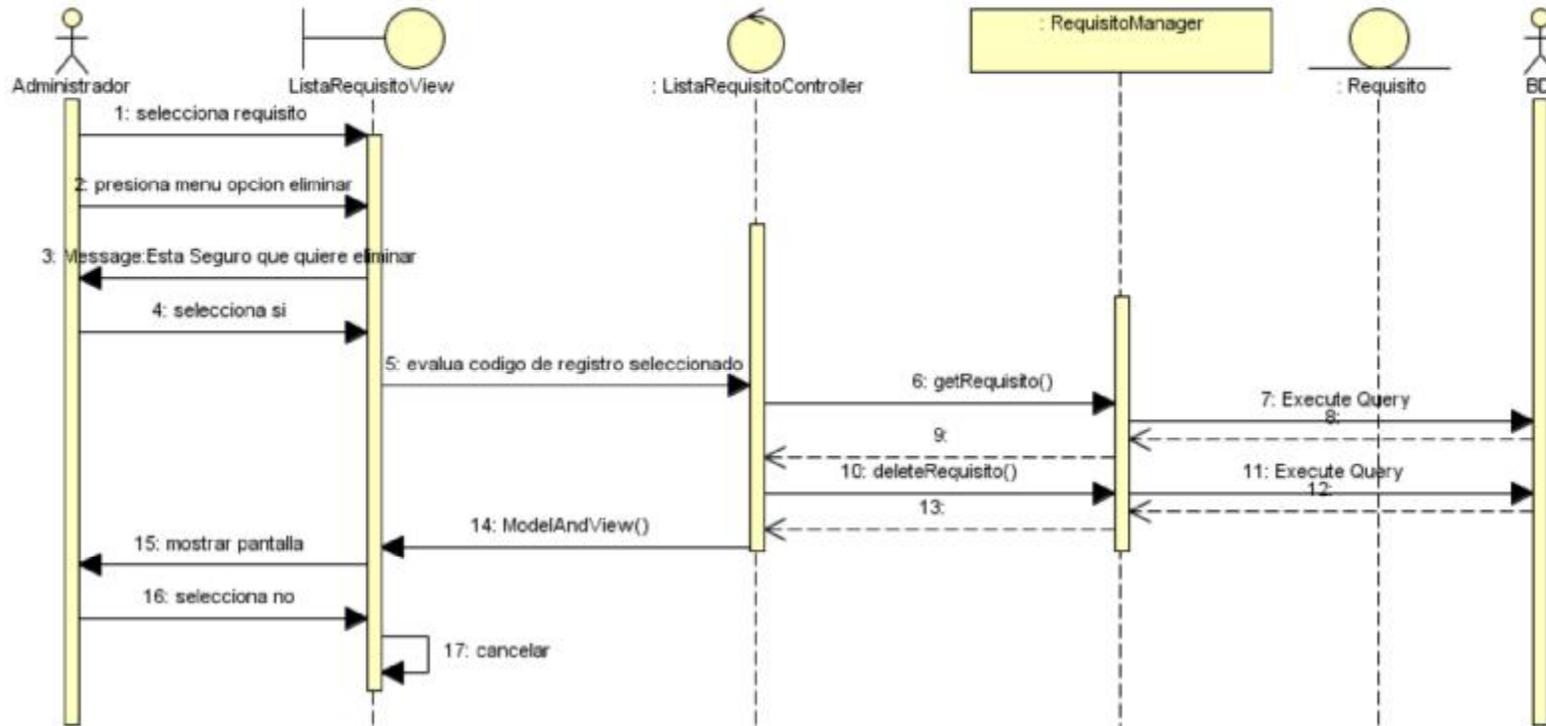


Figura 342. Diagrama de Secuencia: CU Eliminar Requisito

2.1.2.2.3.9.2.1.50 Diagrama de Secuencia: Caso de Uso Cambiar Estado Requisito

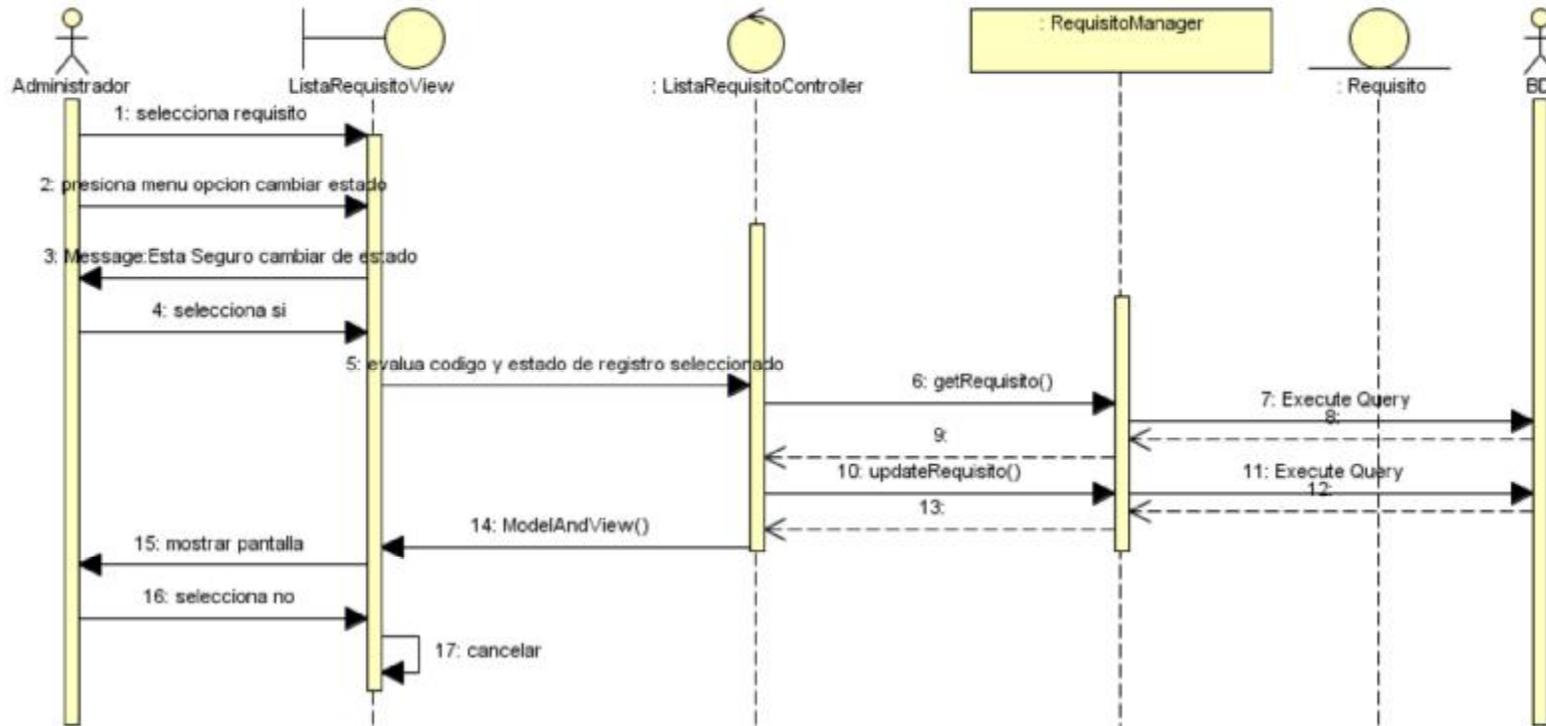


Figura 343. Diagrama de Secuencia: CU Cambiar Estado Requisito

2.1.2.2.3.9.2.1.51 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Requisito

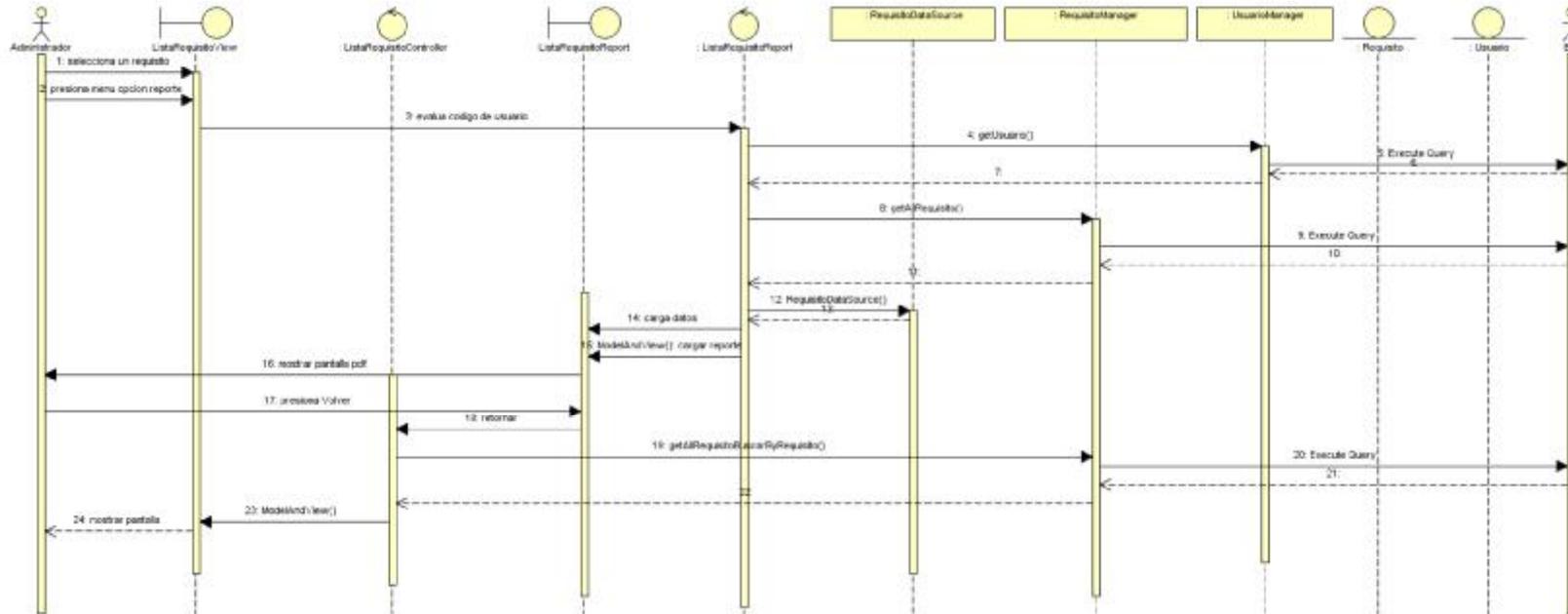


Figura 344. Diagrama de Secuencia: CU Generar Reporte Lista Requisito

2.1.2.2.3.9.2.1.52 Diagrama de Secuencia: Caso de Uso Administrar Rol

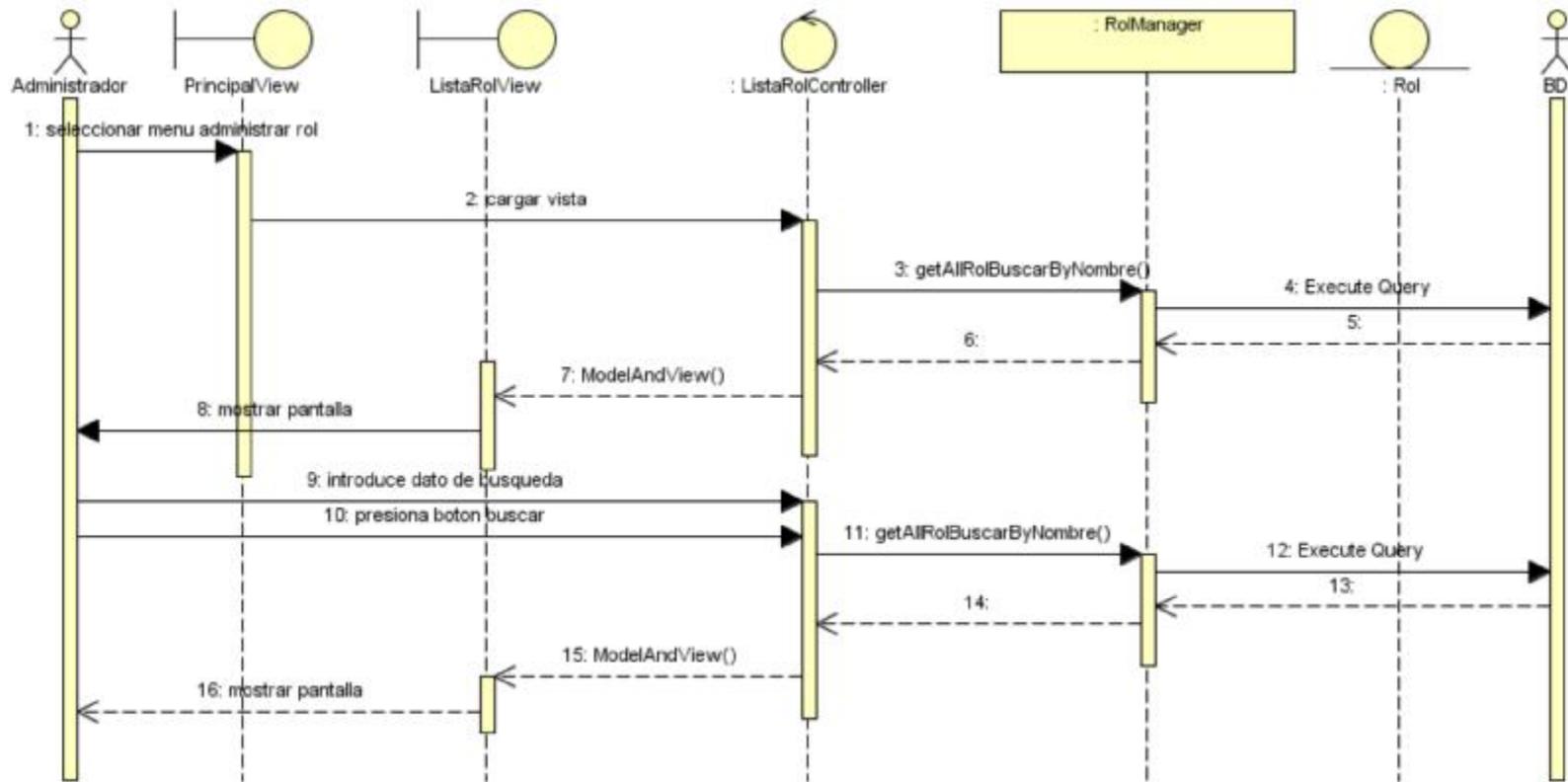


Figura 345. Diagrama de Secuencia: CU Administrar Rol

2.1.2.2.3.9.2.1.53 Diagrama de Secuencia: Caso de Uso Adicionar Rol

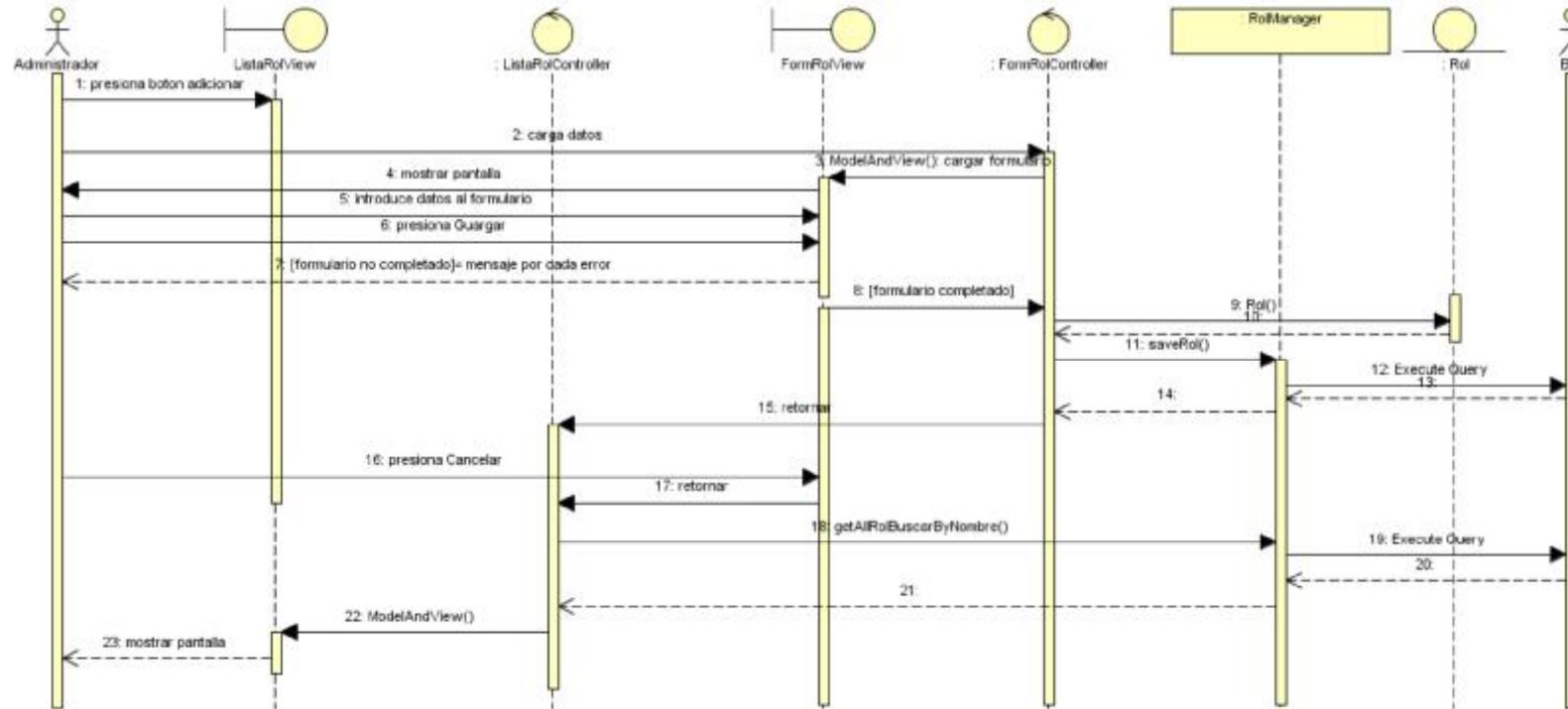


Figura 346. Diagrama de Secuencia: CU Adicionar Rol

2.1.2.2.3.9.2.1.54 Diagrama de Secuencia: Caso de Uso Modificar Rol

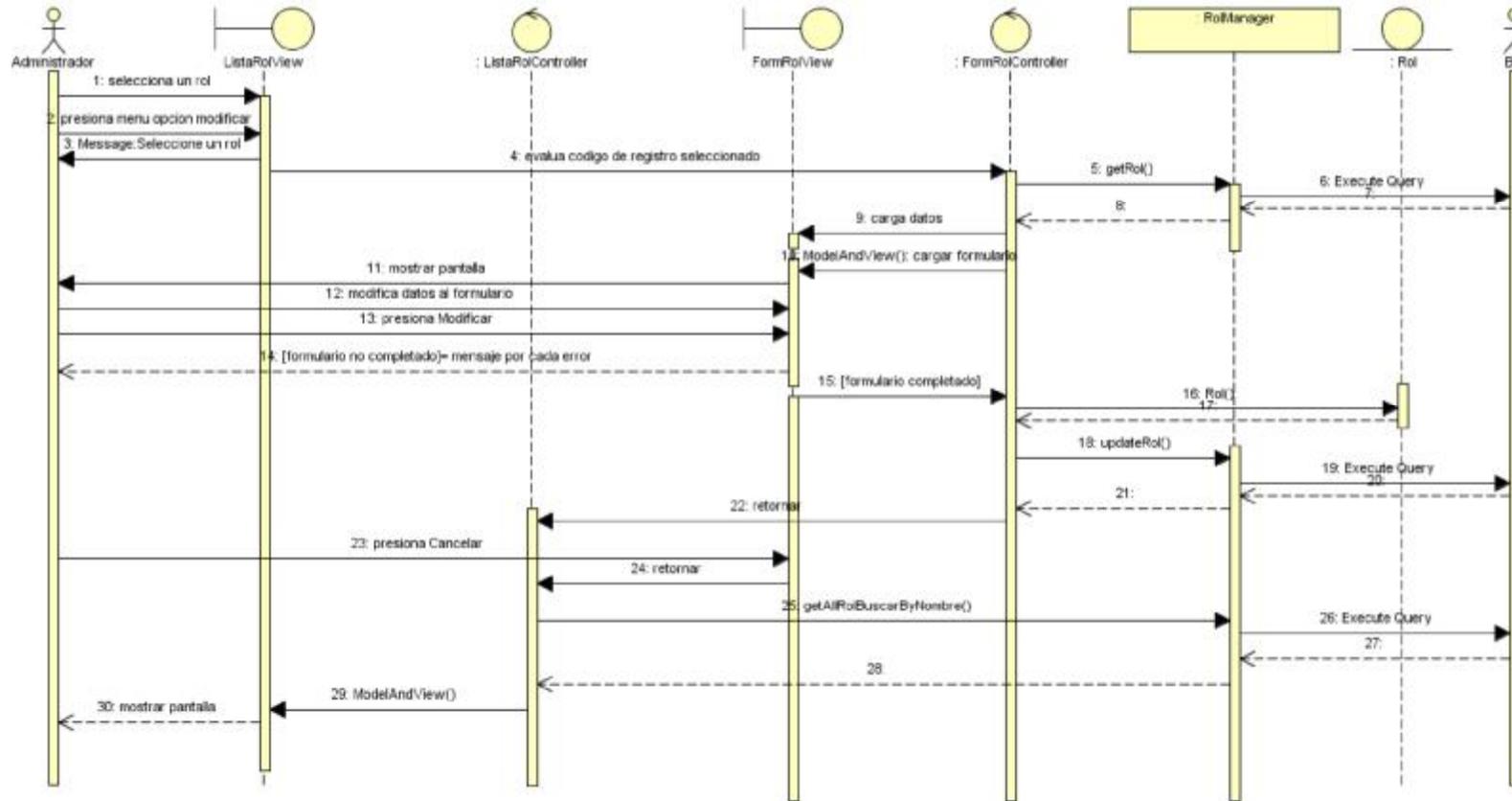


Figura 347. Diagrama de Secuencia: CU Modificar Rol

2.1.2.2.3.9.2.1.55 Diagrama de Secuencia: Caso de Uso Eliminar Rol

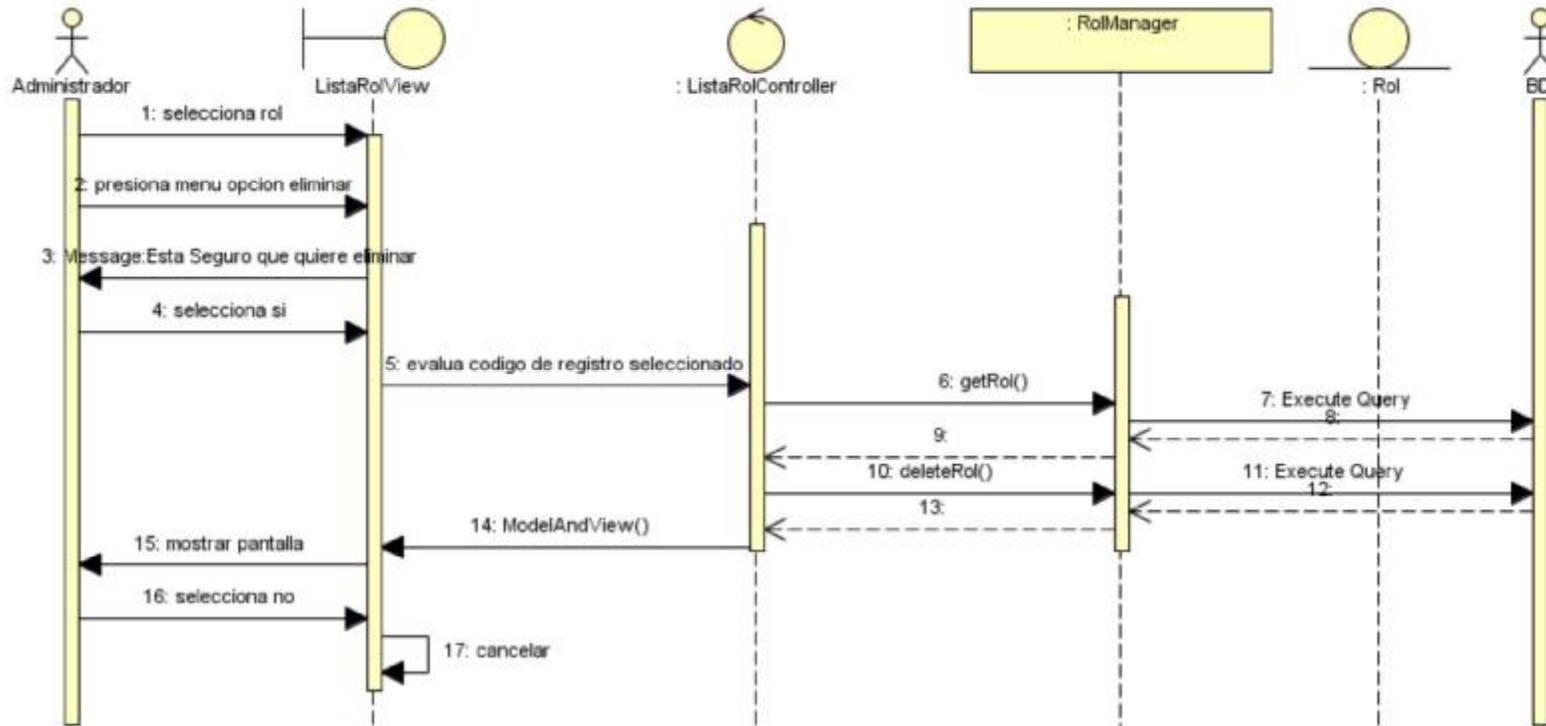


Figura 348. Diagrama de Secuencia: CU Eliminar Rol

2.1.2.2.3.9.2.1.56 Diagrama de Secuencia: Caso de Uso Cambiar Estado Rol

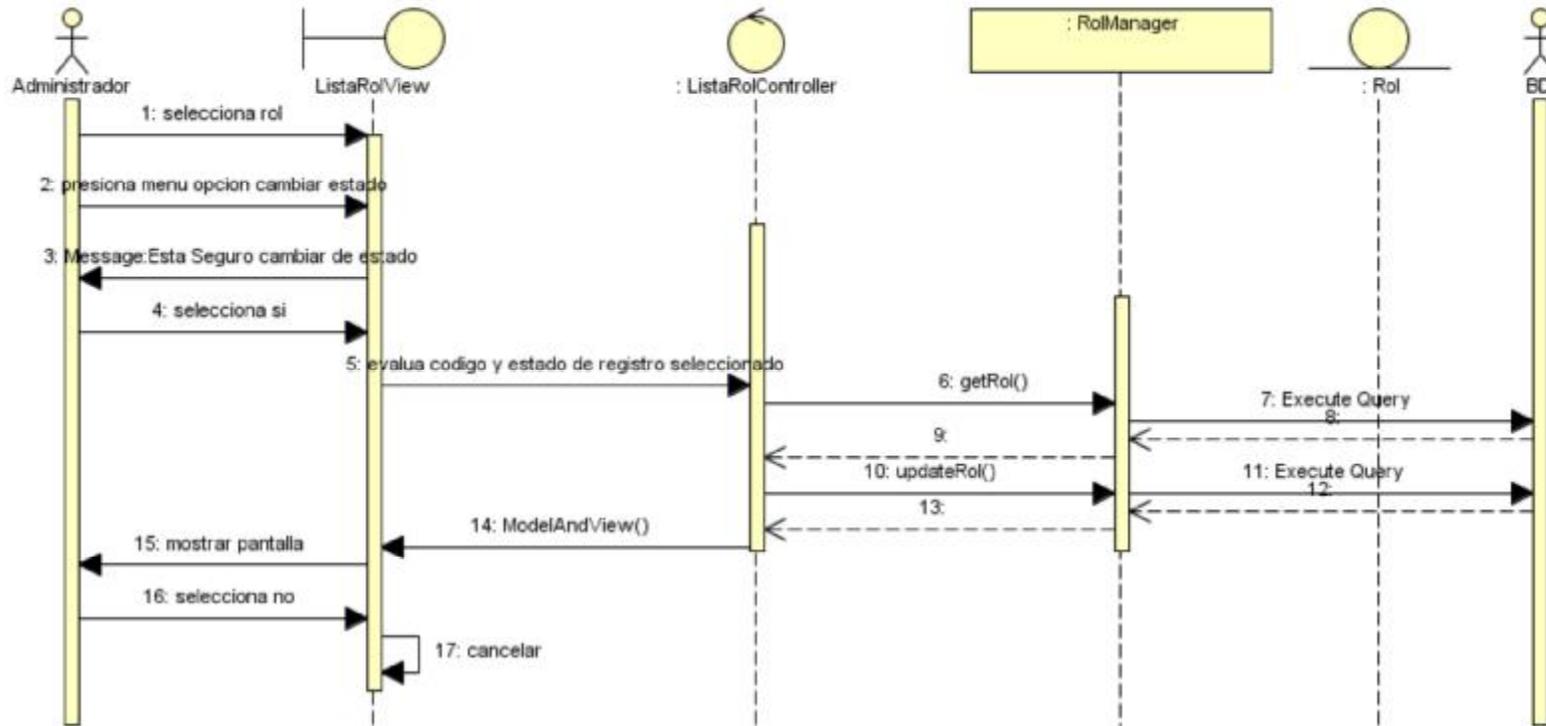


Figura 349. Diagrama de Secuencia: CU Cambiar Estado Rol

2.1.2.2.3.9.2.1.57 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Rol

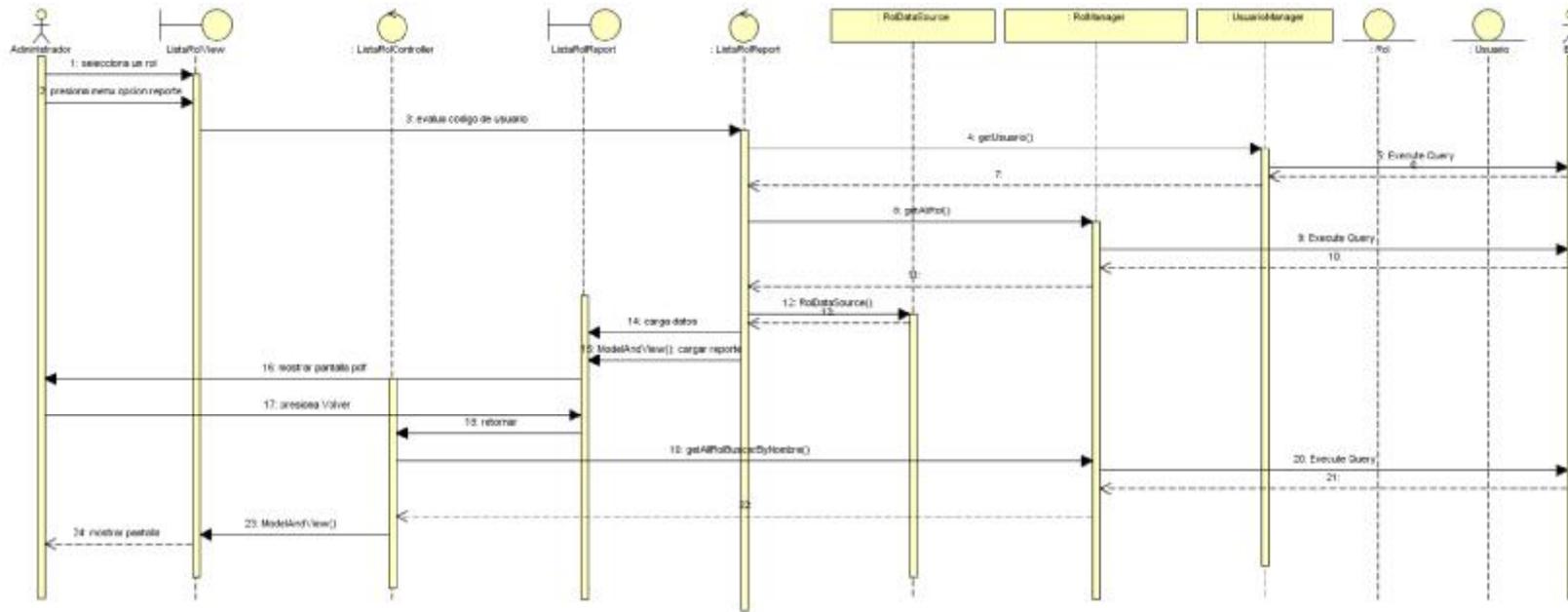


Figura 350. Diagrama de Secuencia: CU Generar Reporte Lista Rol

2.1.2.2.3.9.2.1.58 Diagrama de Secuencia: Caso de Uso Asignar Funcionalidad

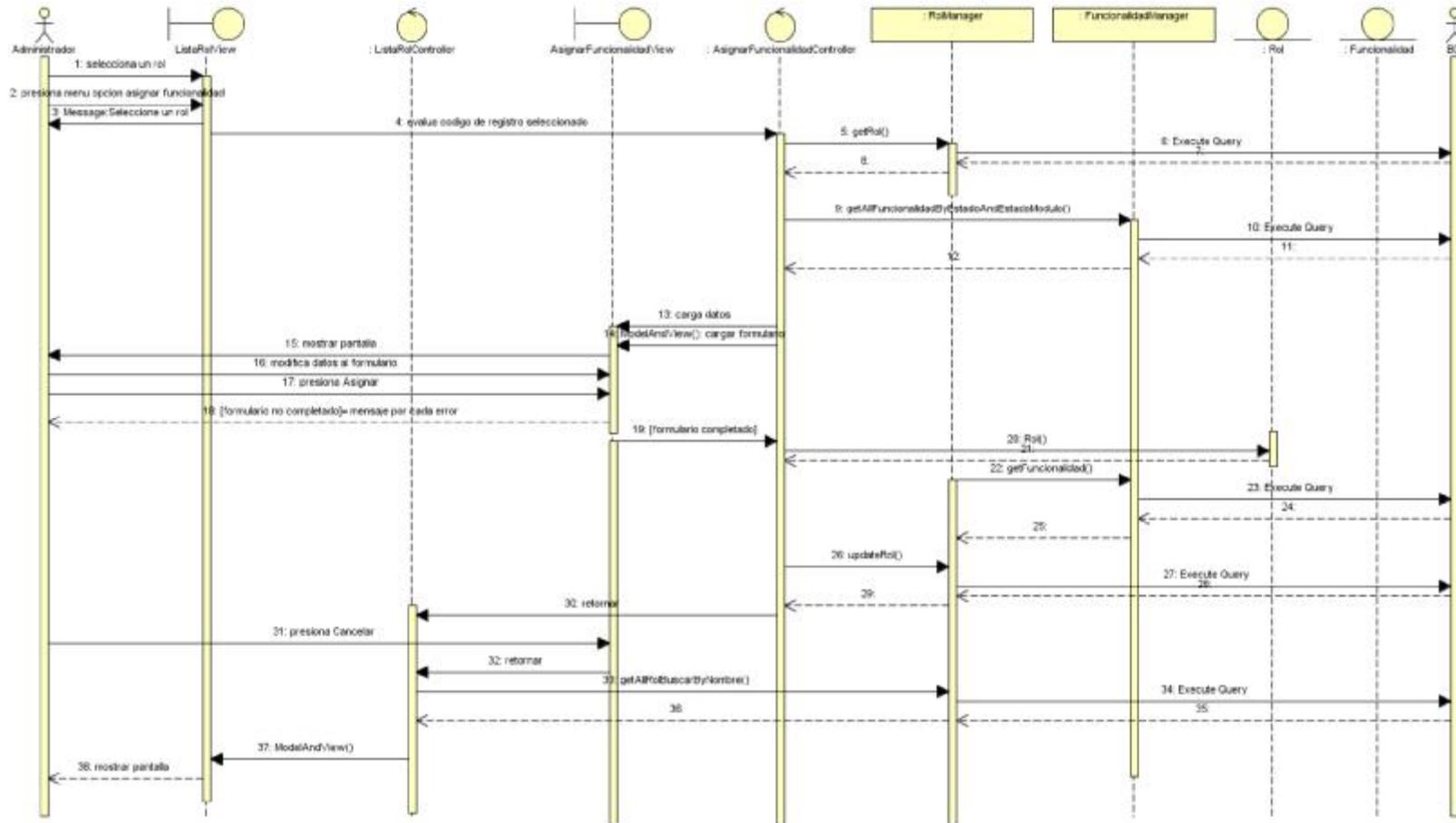


Figura 351. Diagrama de Secuencia: CU Asignar Funcionalidad

2.1.2.2.3.9.2.1.59 Diagrama de Secuencia: Caso de Uso Administrar Tipo de Empresa

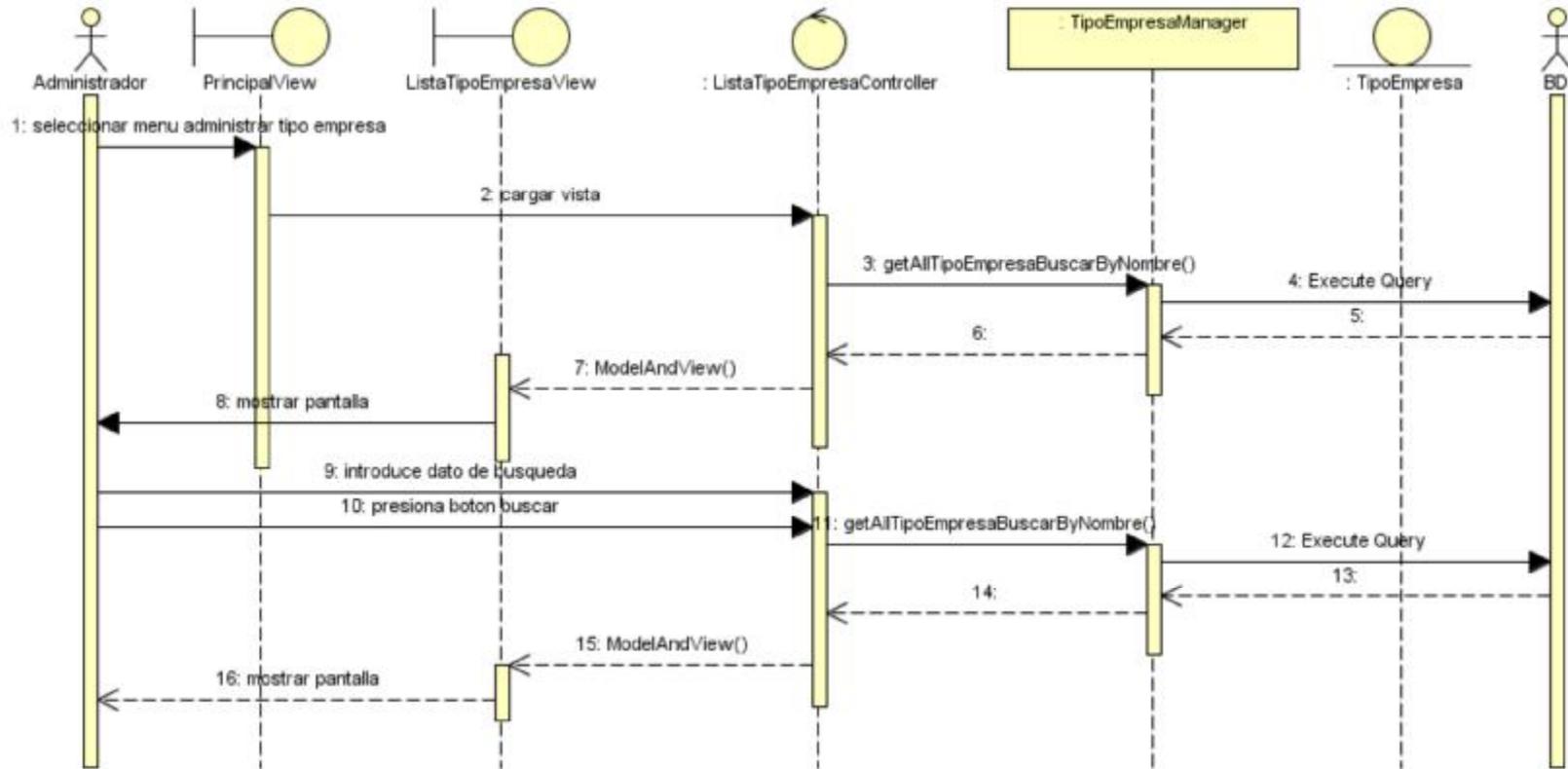


Figura 352. Diagrama de Secuencia: CU Administrar Tipo de Empresa

2.1.2.2.3.9.2.1.60 Diagrama de Secuencia: Caso de Uso Adicionar Tipo de Empresa

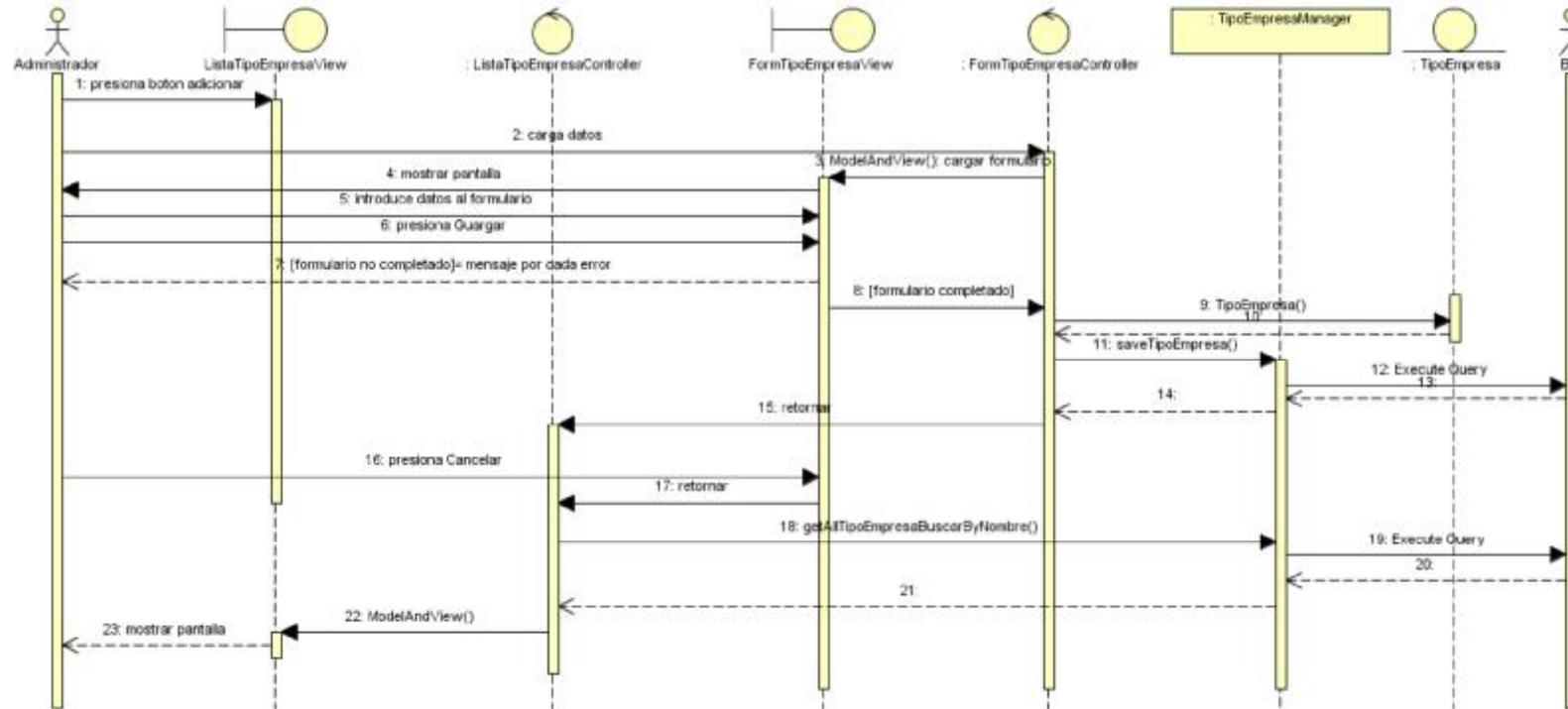


Figura 353. Diagrama de Secuencia: CU Adicionar Tipo de Empresa

2.1.2.2.3.9.2.1.61 Diagrama de Secuencia: Caso de Uso Modificar Tipo de Empresa

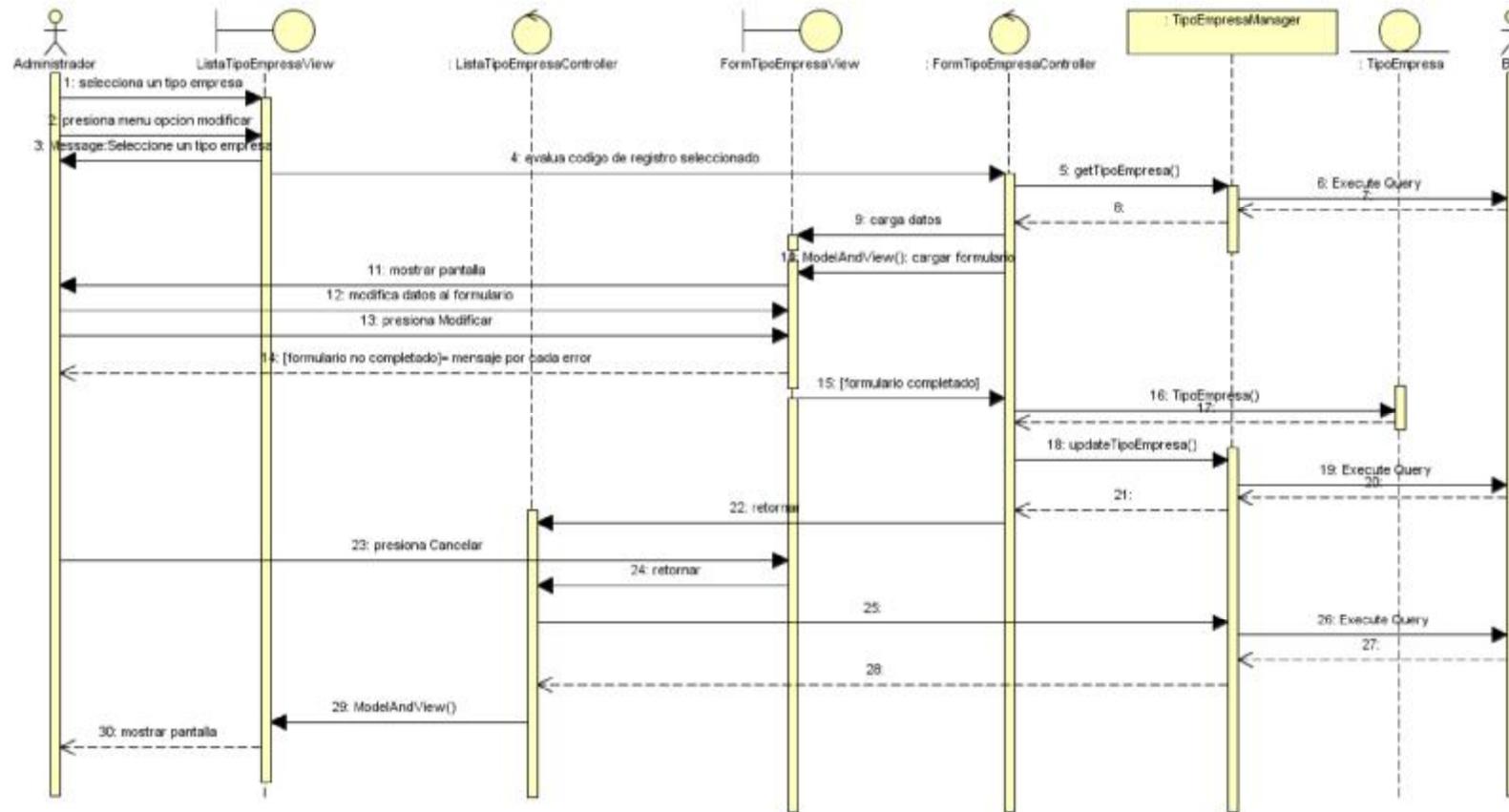


Figura 354. Diagrama de Secuencia: CU Modificar Tipo de Empresa

2.1.2.2.3.9.2.1.62 Diagrama de Secuencia: Caso de Uso Eliminar Tipo de Empresa

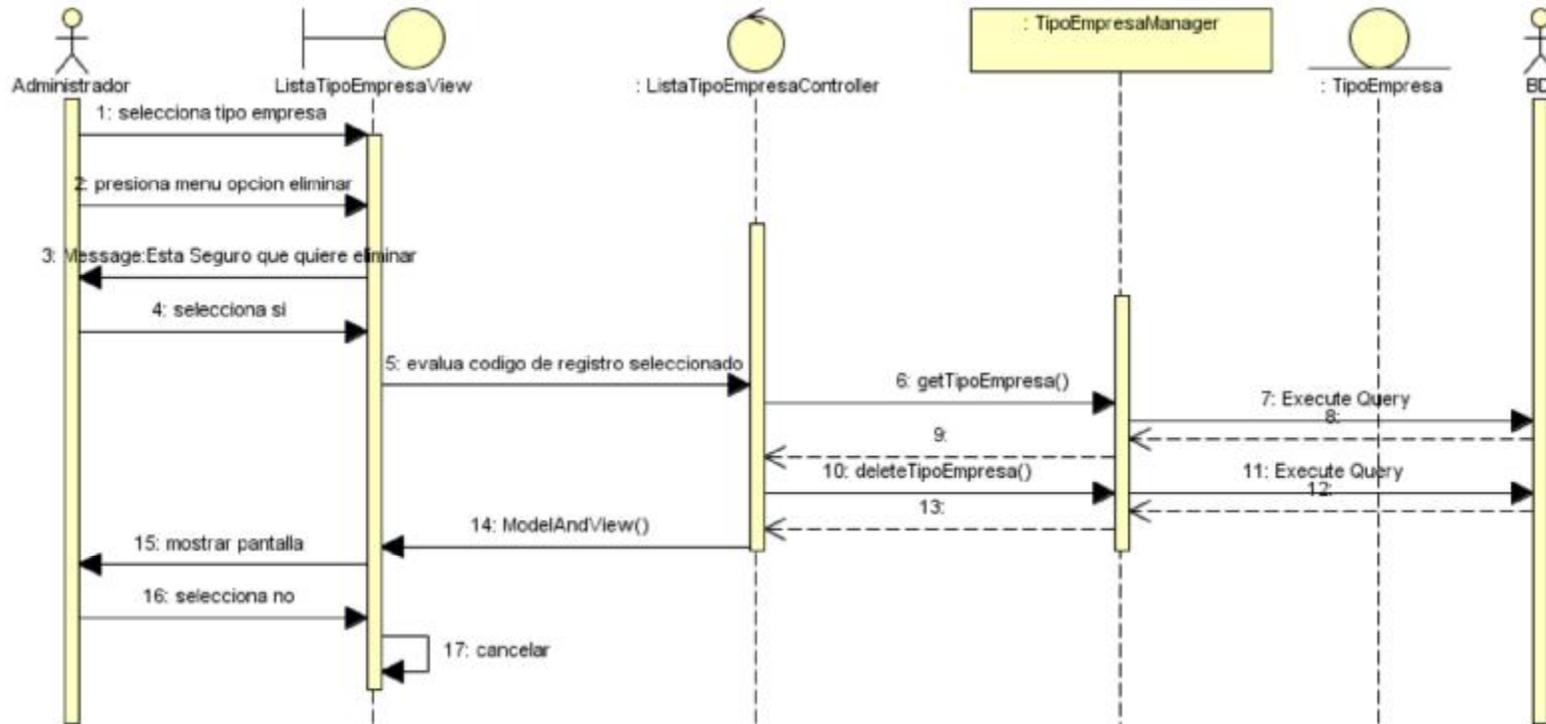


Figura 355. Diagrama de Secuencia: CU Eliminar Tipo de Empresa

2.1.2.2.3.9.2.1.63 Diagrama de Secuencia: Caso de Uso Cambiar Estado Tipo de Empresa

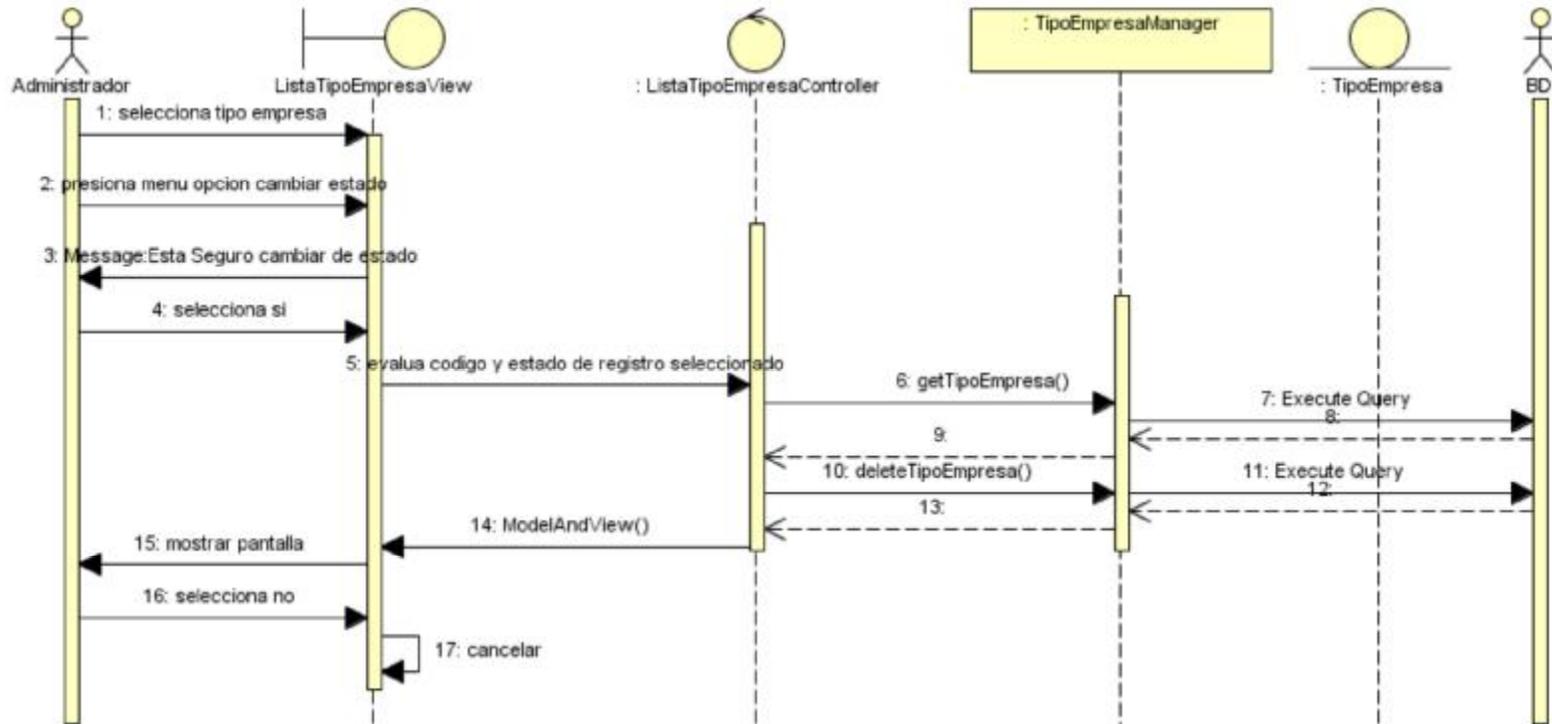


Figura 356. Diagrama de Secuencia: CU Cambiar Estado Tipo de Empresa

2.1.2.2.3.9.2.1.64 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Tipo de Empresa

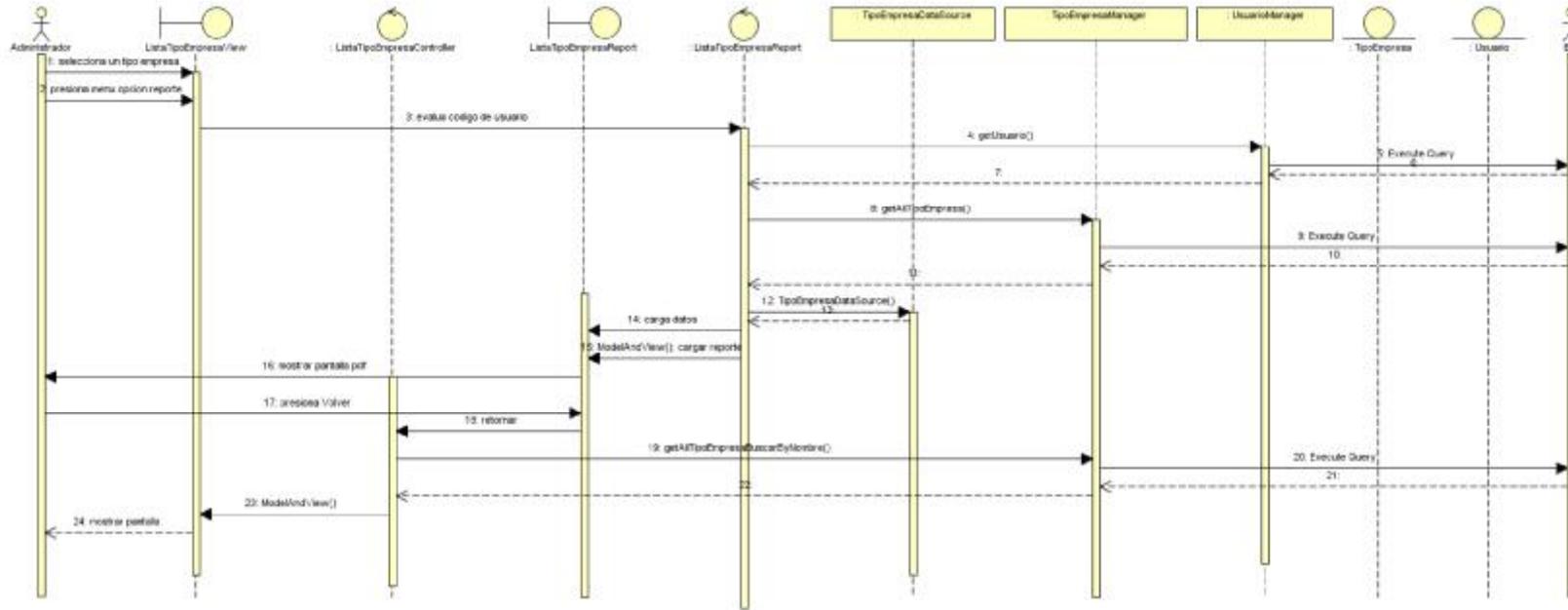


Figura 357. Diagrama de Secuencia: CU Generar Reporte Lista Tipo de Empresa

2.1.2.2.3.9.2.1.65 Diagrama de Secuencia: Caso de Uso Administrar Tipo de Seguro

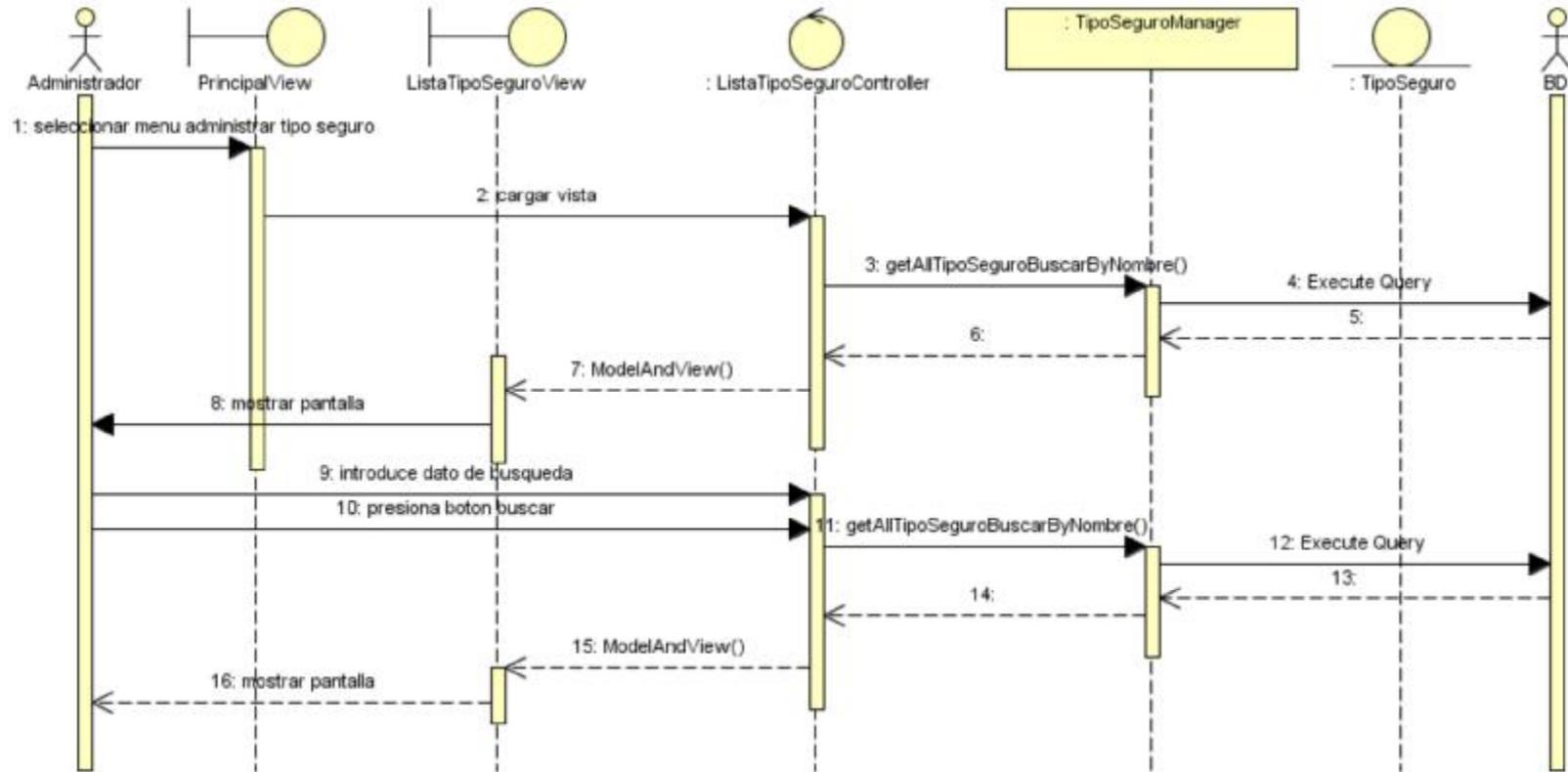


Figura 358. Diagrama de Secuencia: CU Administrar Tipo de Seguro

2.1.2.2.3.9.2.1.66 Diagrama de Secuencia: Caso de Uso Adicionar Tipo de Seguro

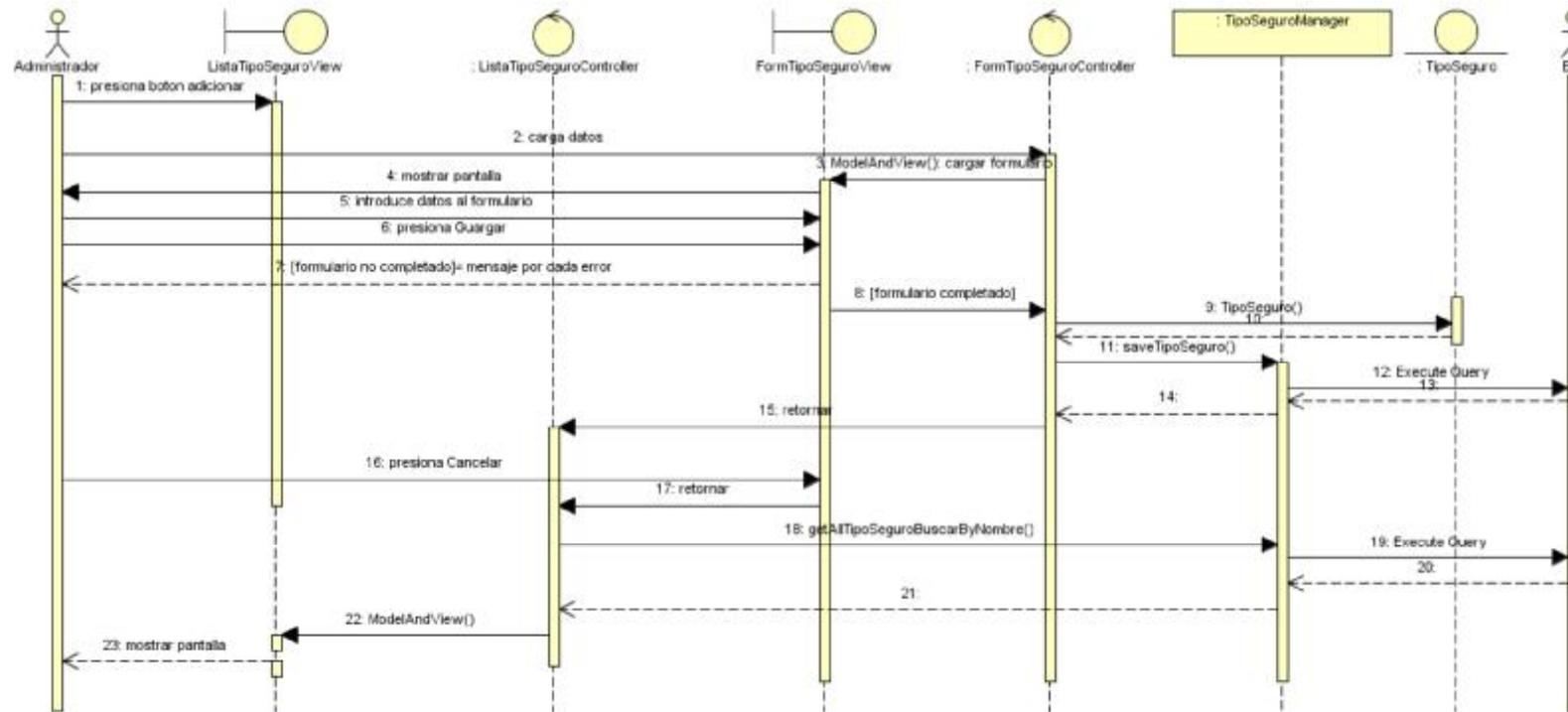


Figura 359. Diagrama de Secuencia: CU Adicionar Tipo de Seguro

2.1.2.2.3.9.2.1.67 Diagrama de Secuencia: Caso de Uso Modificar Tipo de Seguro

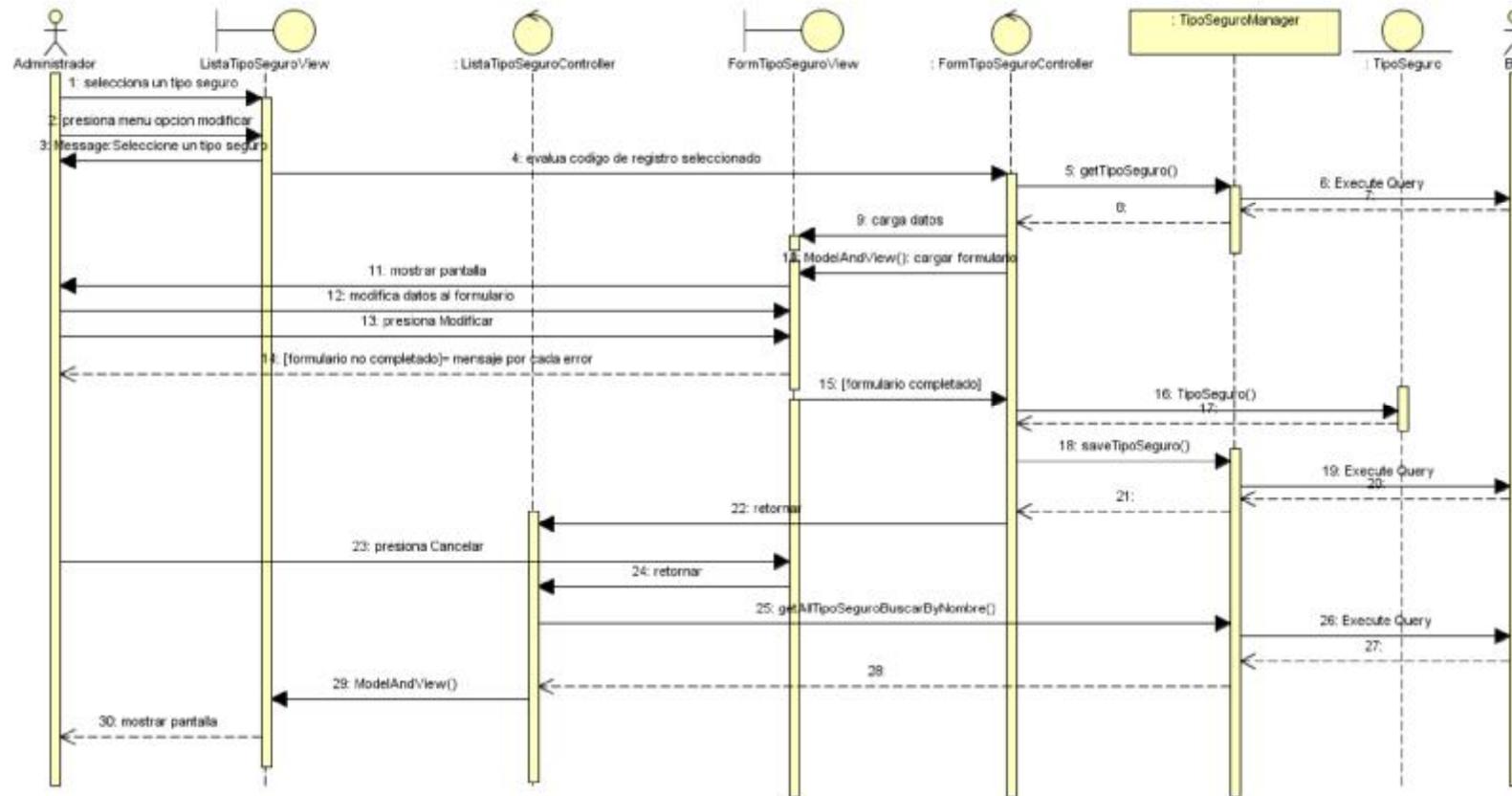


Figura 360. Diagrama de Secuencia: CU Modificar Tipo de Seguro

2.1.2.2.3.9.2.1.68 Diagrama de Secuencia: Caso de Uso Eliminar Tipo de Seguro

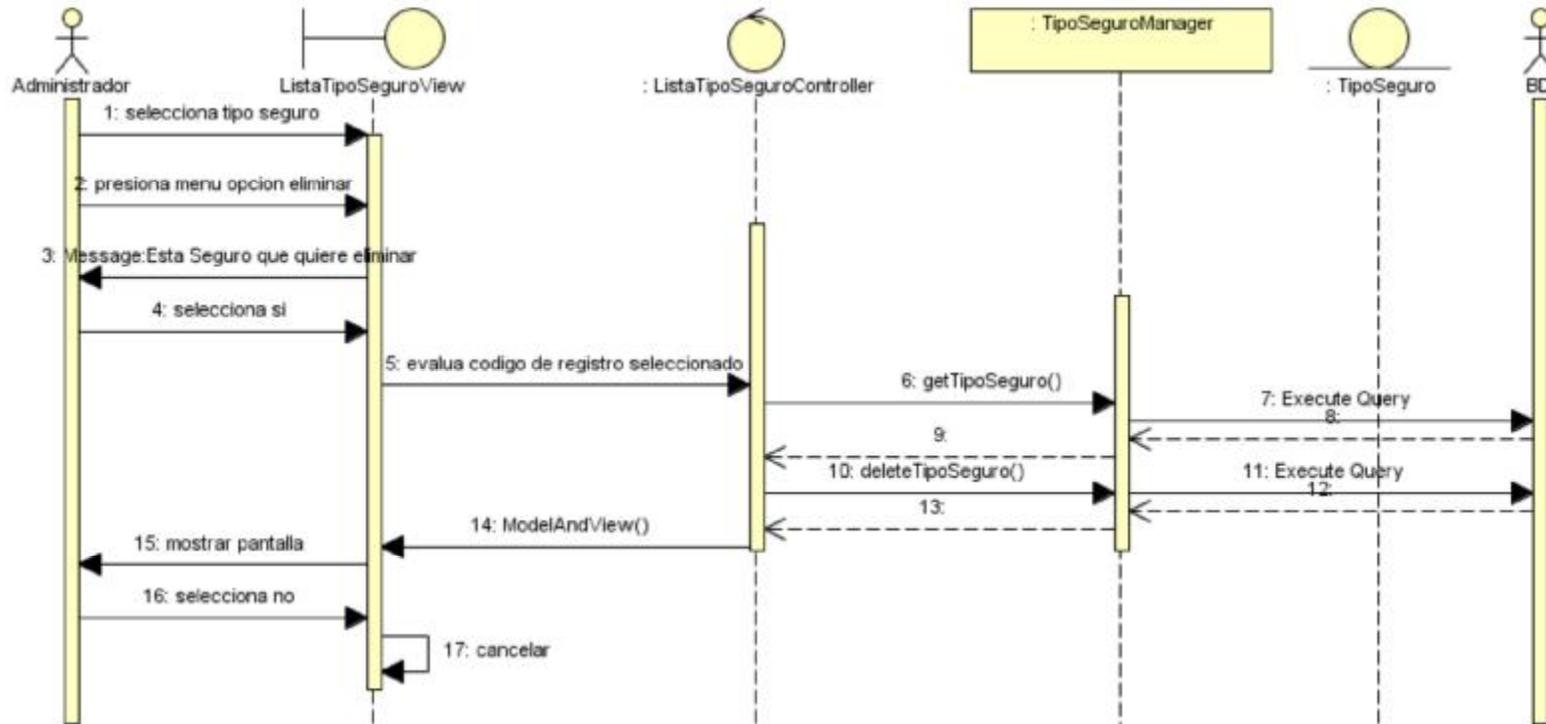


Figura 361. Diagrama de Secuencia: CU Eliminar Tipo de Seguro

2.1.2.2.3.9.2.1.69 Diagrama de Secuencia: Caso de Uso Cambiar Estado Tipo de Seguro

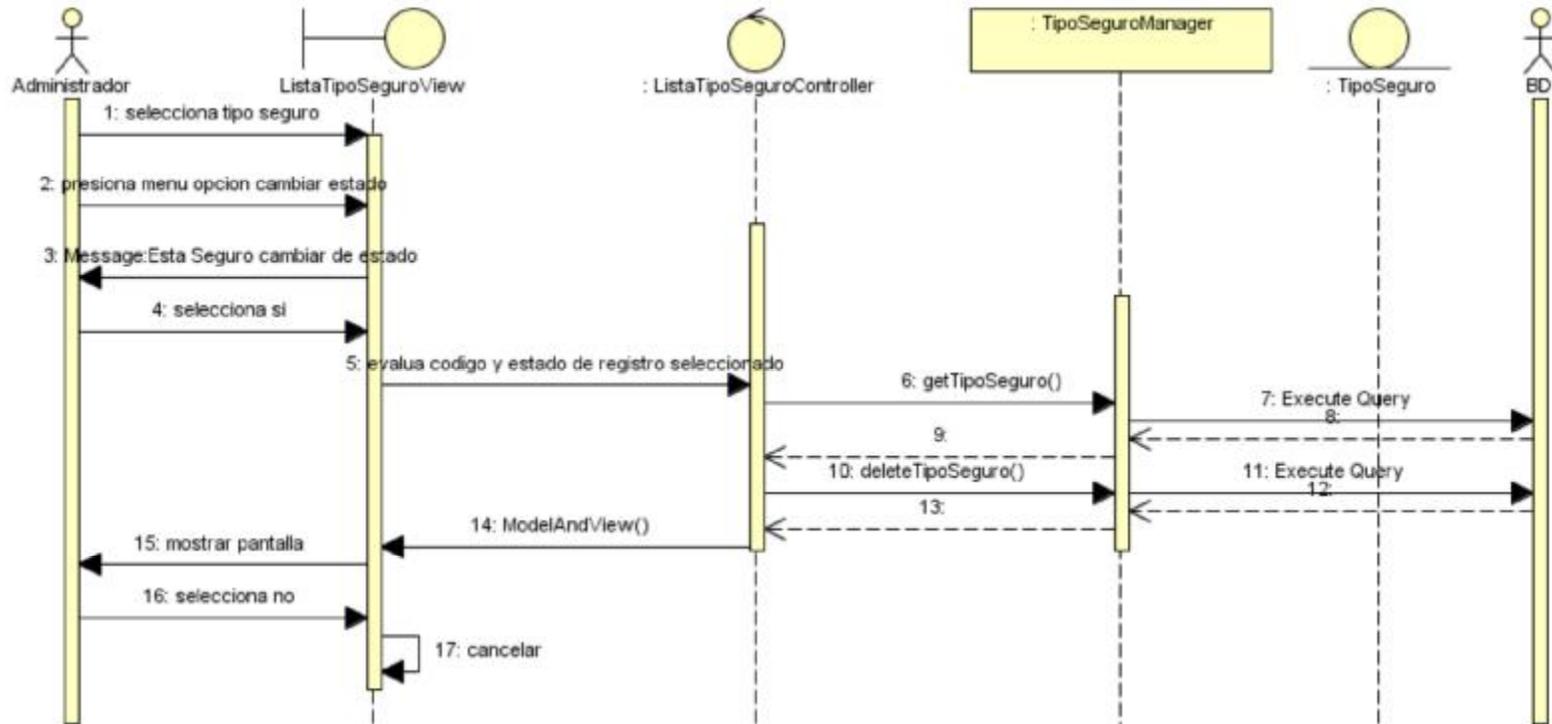


Figura 362. Diagrama de Secuencia: CU Cambiar Estado Tipo de Seguro

2.1.2.2.3.9.2.1.70 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Tipo de Seguro

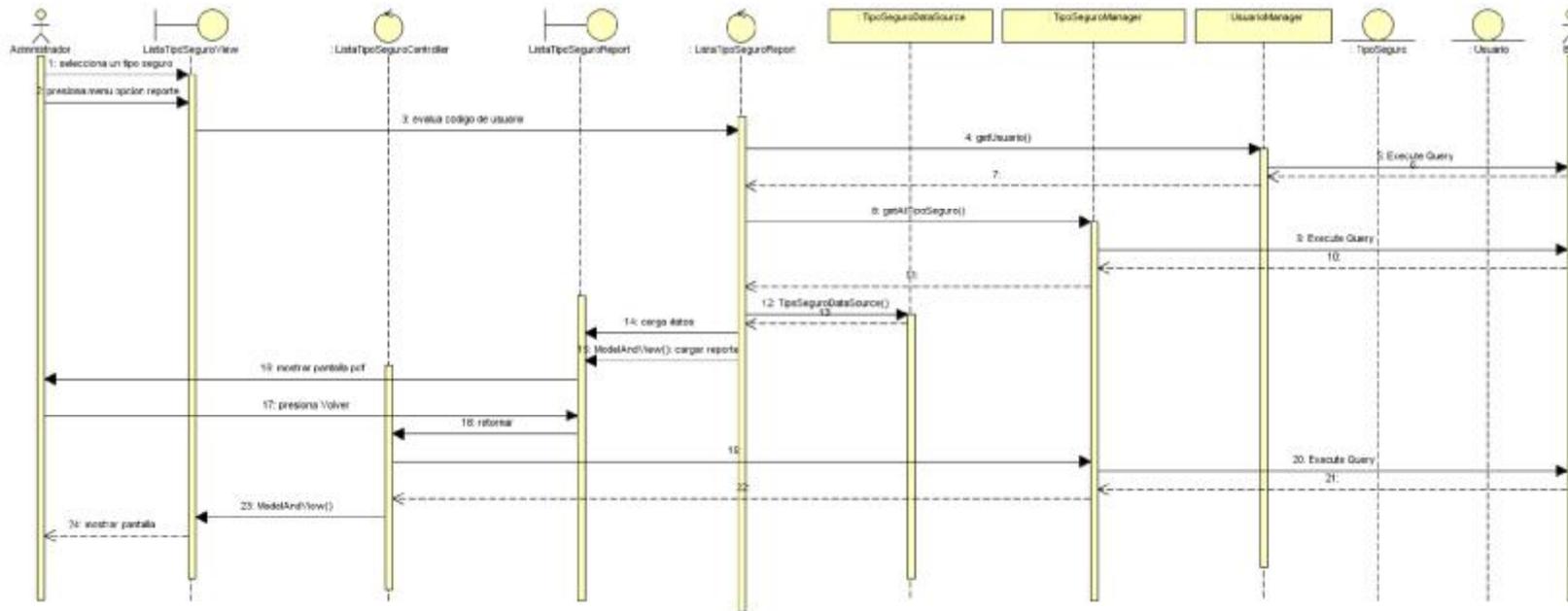


Figura 363. Diagrama de Secuencia: CU Generar Reporte Lista Tipo de Seguro

2.1.2.2.3.9.2.1.71 Diagrama de Secuencia: Caso de Uso Administrar Usuario

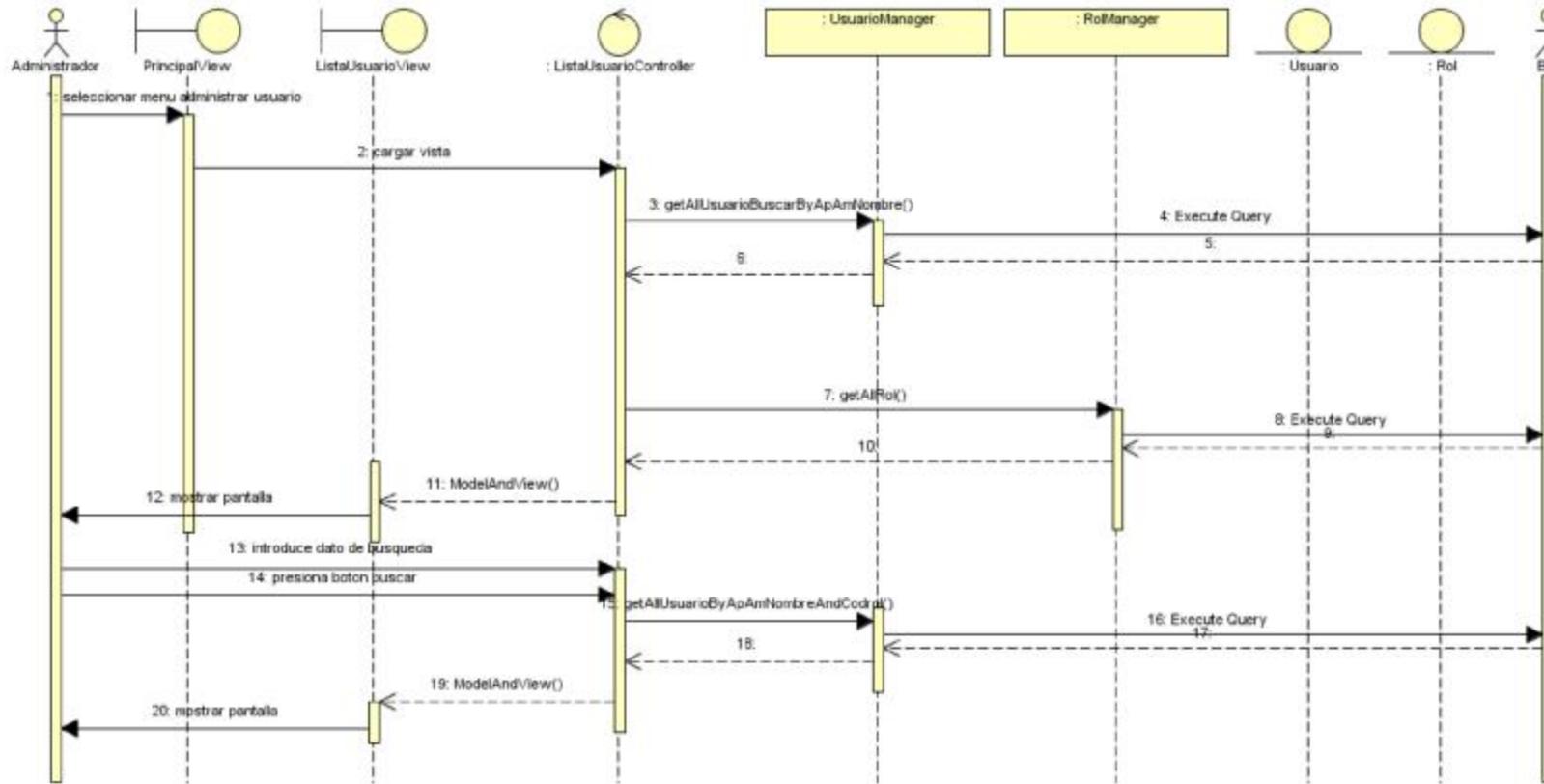


Figura 364. Diagrama de Secuencia: CU Administrar Usuario

2.1.2.2.3.9.2.1.72 Diagrama de Secuencia: Caso de Uso Adicionar Usuario

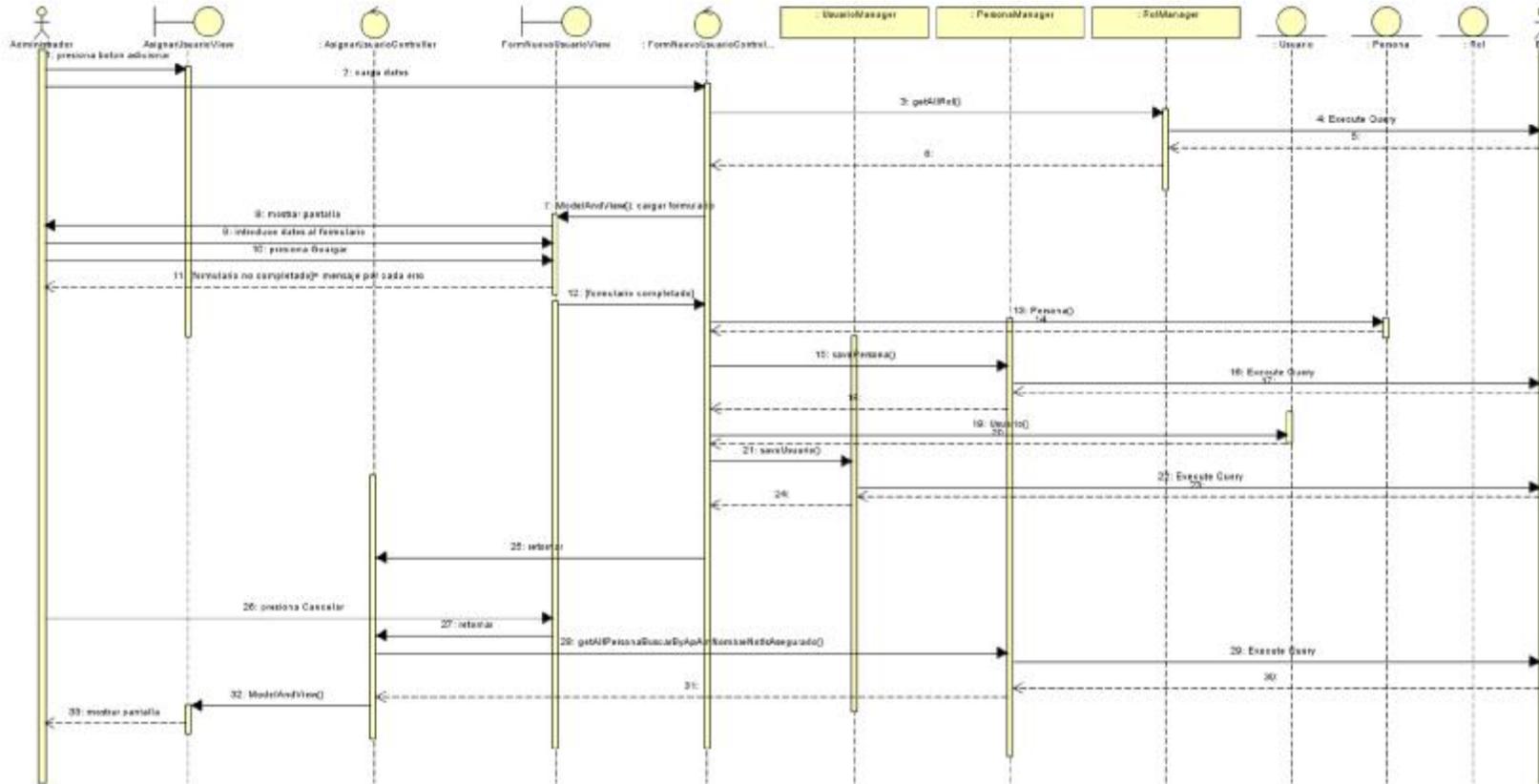


Figura 365. Diagrama de Secuencia: CU Adicionar Usuario

2.1.2.2.3.9.2.1.73 Diagrama de Secuencia: Caso de Uso Modificar Usuario

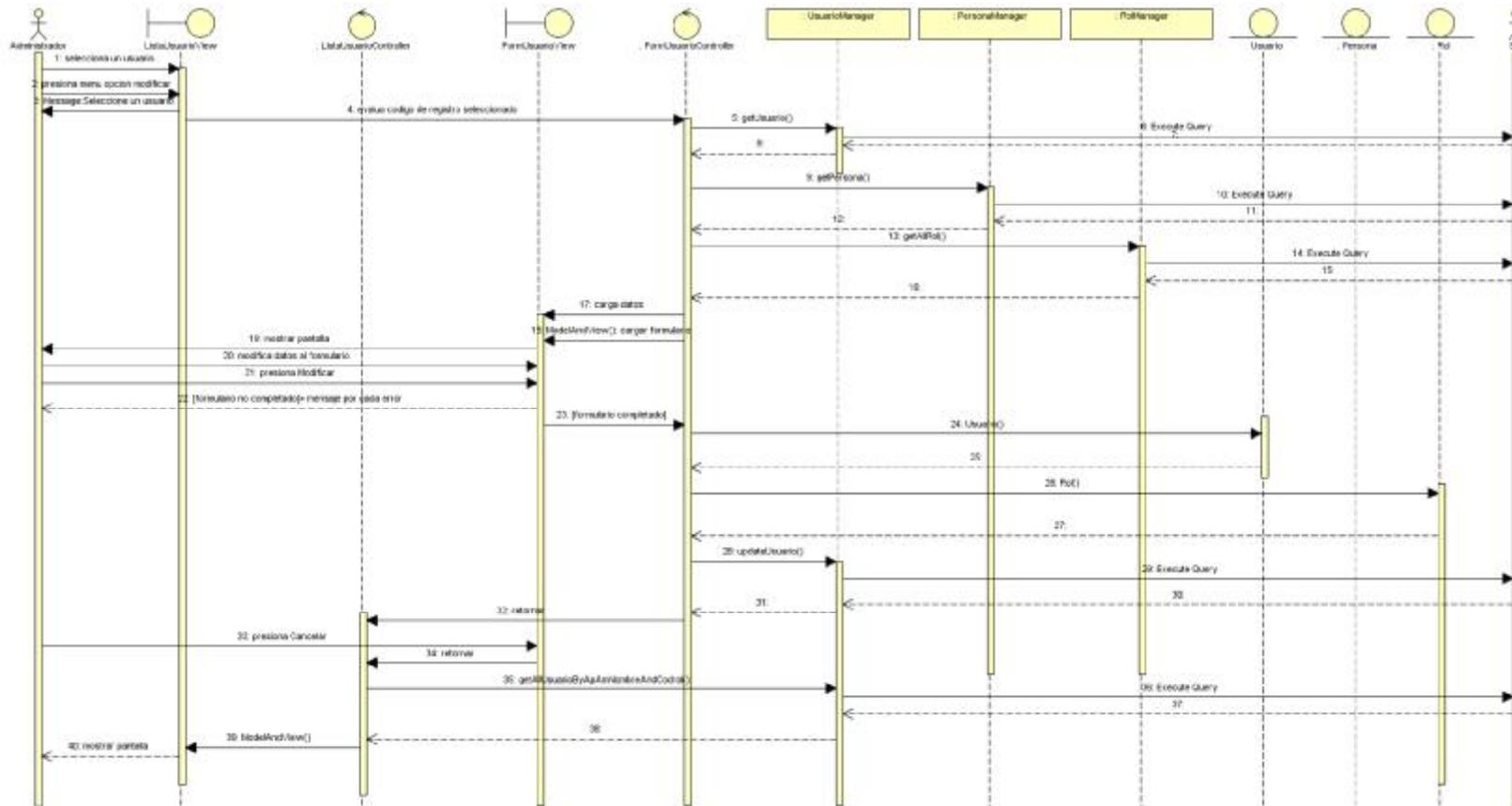


Figura 366. Diagrama de Secuencia: CU Modificar Usuario

2.1.2.2.3.9.2.1.74 Diagrama de Secuencia: Caso de Uso Eliminar Usuario

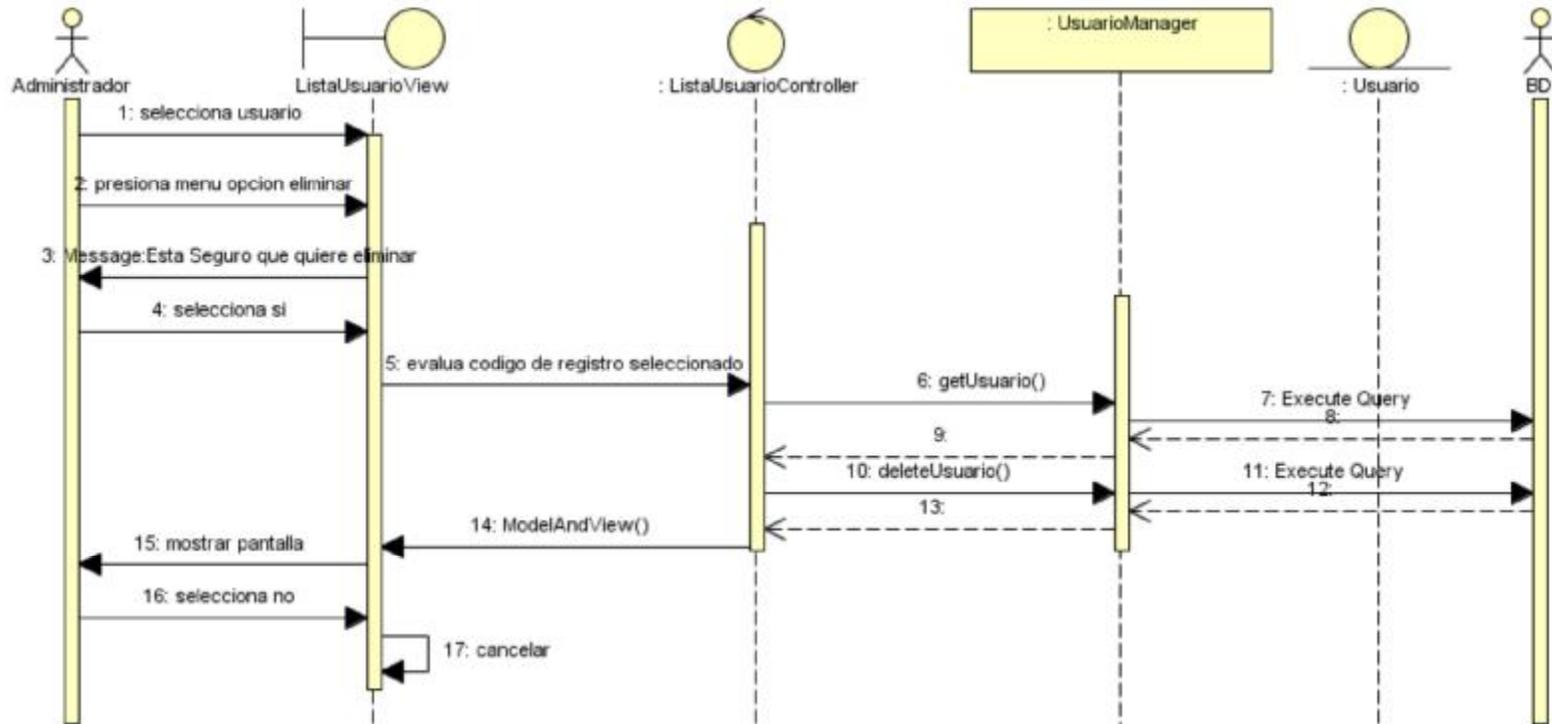


Figura 367. Diagrama de Secuencia: CU Eliminar Usuario

2.1.2.2.3.9.2.1.75 Diagrama de Secuencia: Caso de Uso Cambiar Estado Usuario

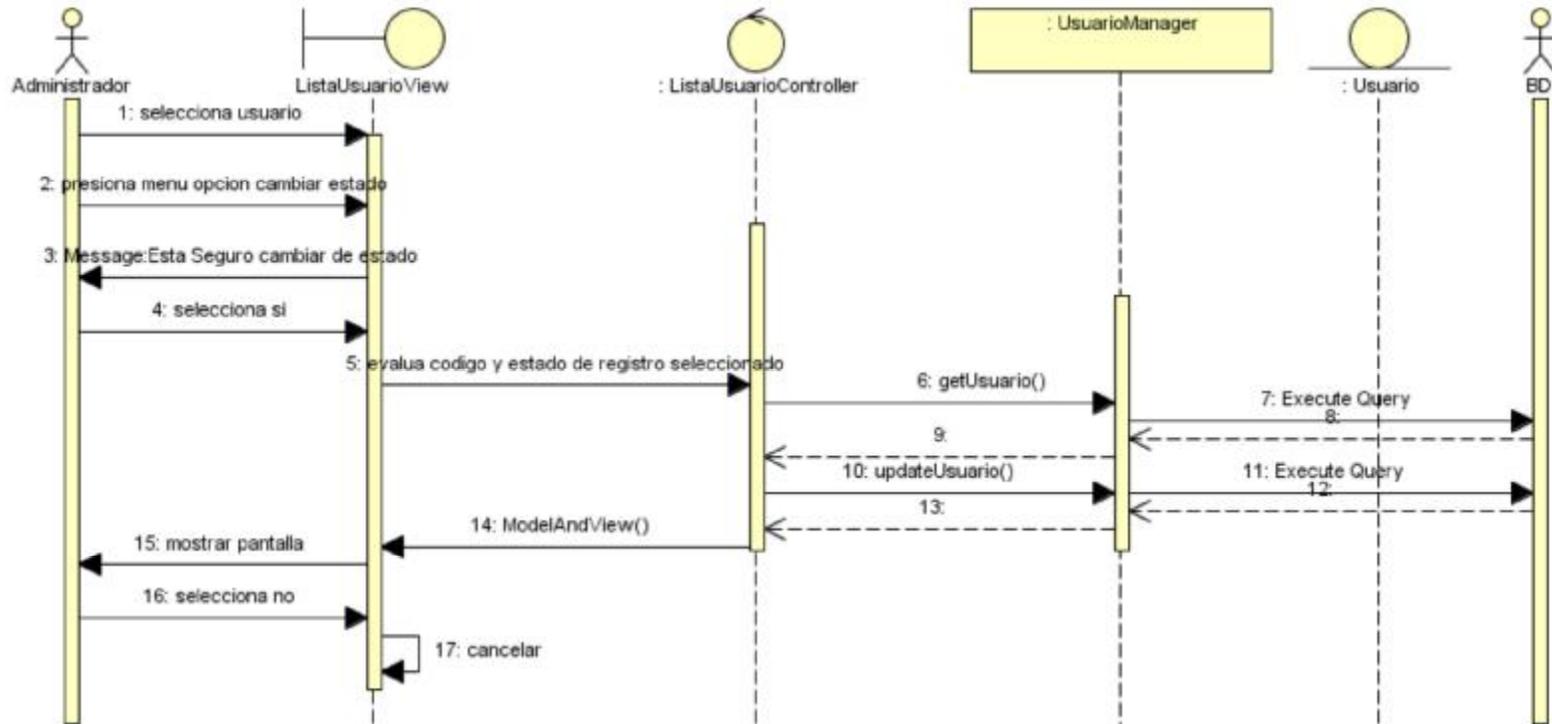


Figura 368. Diagrama de Secuencia: CU Cambiar Estado Usuario

2.1.2.2.3.9.2.1.76 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Usuario

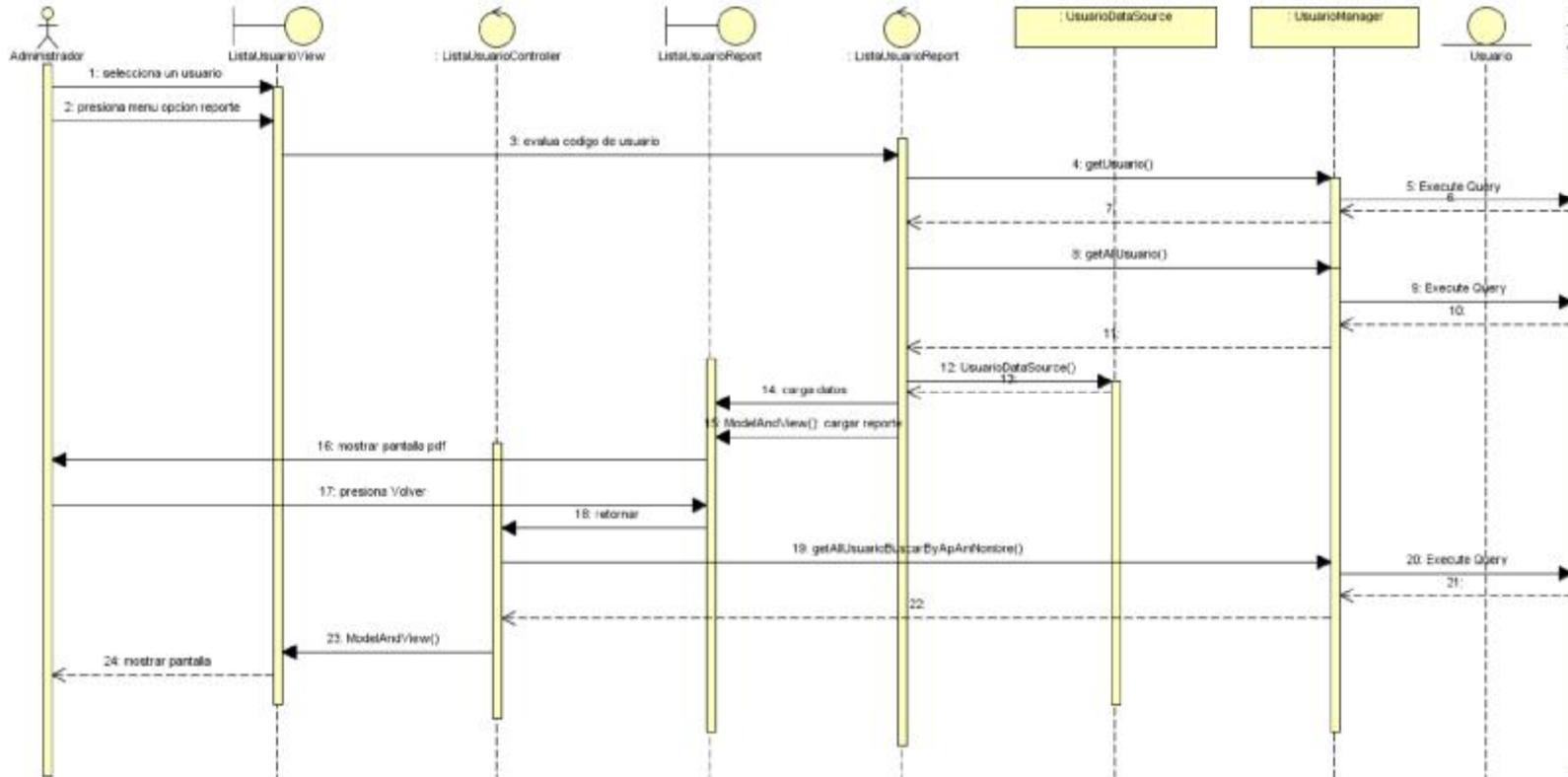


Figura 369. Diagrama de Secuencia: CU Generar Reporte Lista Usuario

2.1.2.2.3.9.2.1.77 Diagrama de Secuencia: CU Generar Reporte Ficha Usuario

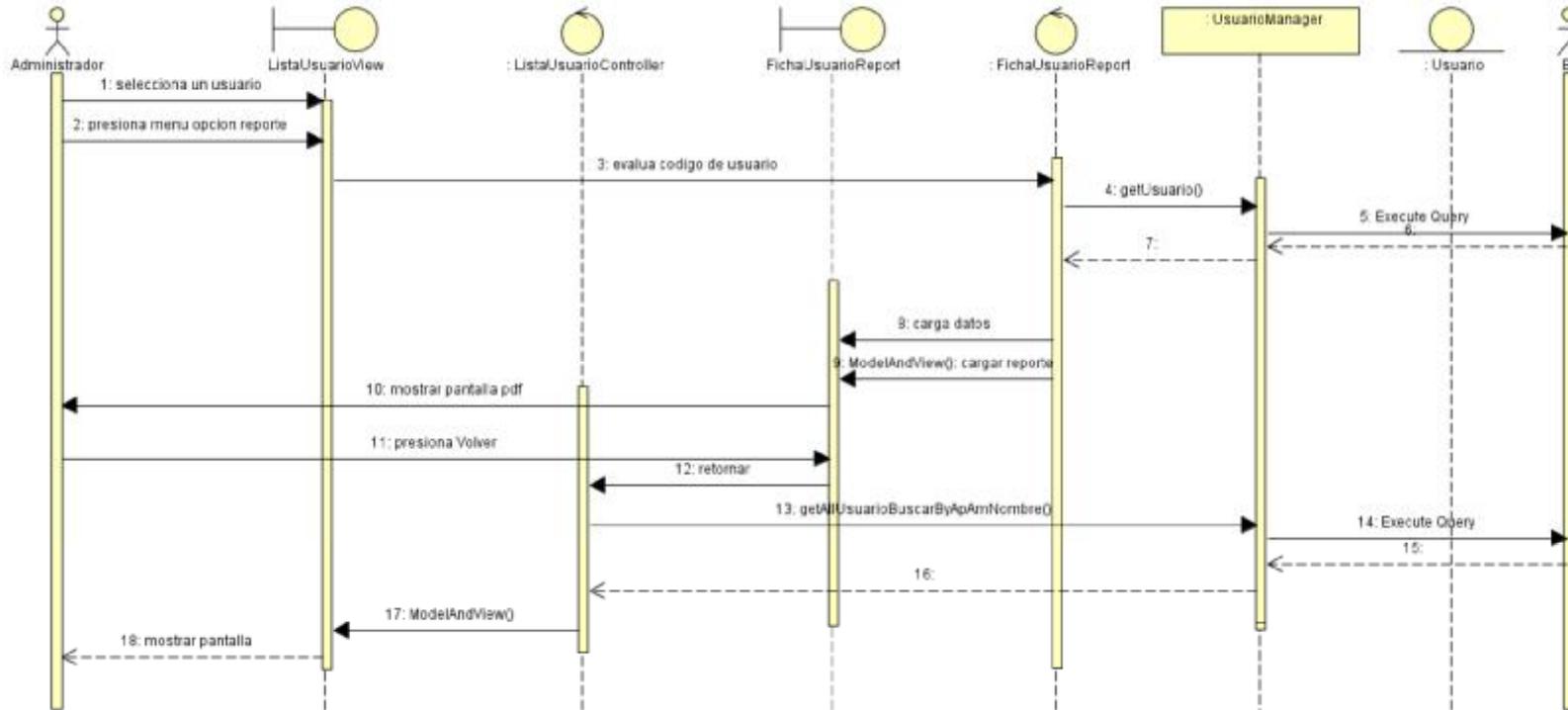


Figura 370. Diagrama de Secuencia: CU Generar Reporte Ficha Usuario

2.1.2.2.3.9.2.1.78 Diagrama de Secuencia: Caso de Uso Gestionar Personas

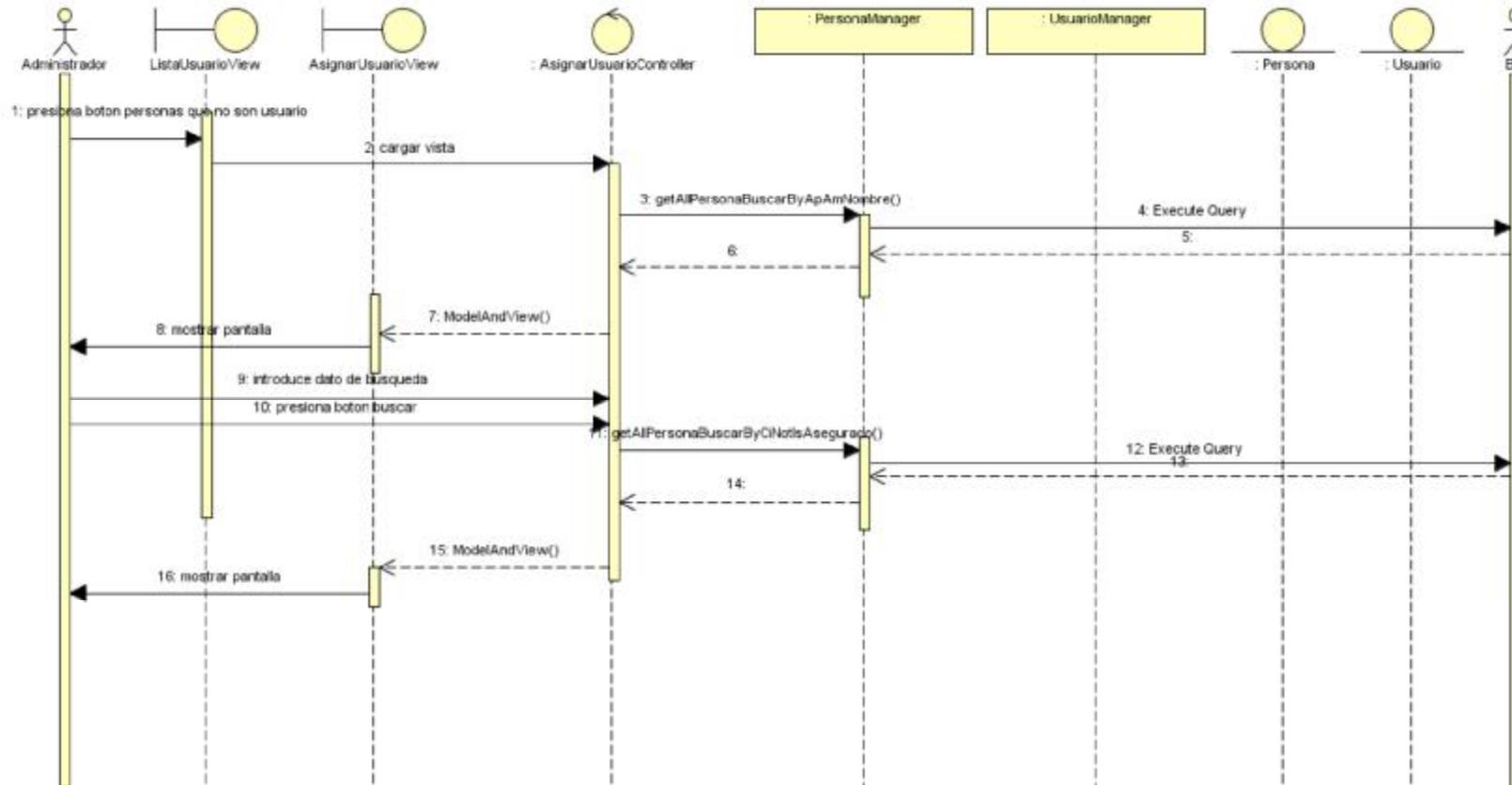


Figura 371. Diagrama de Secuencia: CU Gestionar Personas

2.1.2.2.3.9.2.1.79 Diagrama de Secuencia: Caso de Uso Asignar Login y Clave

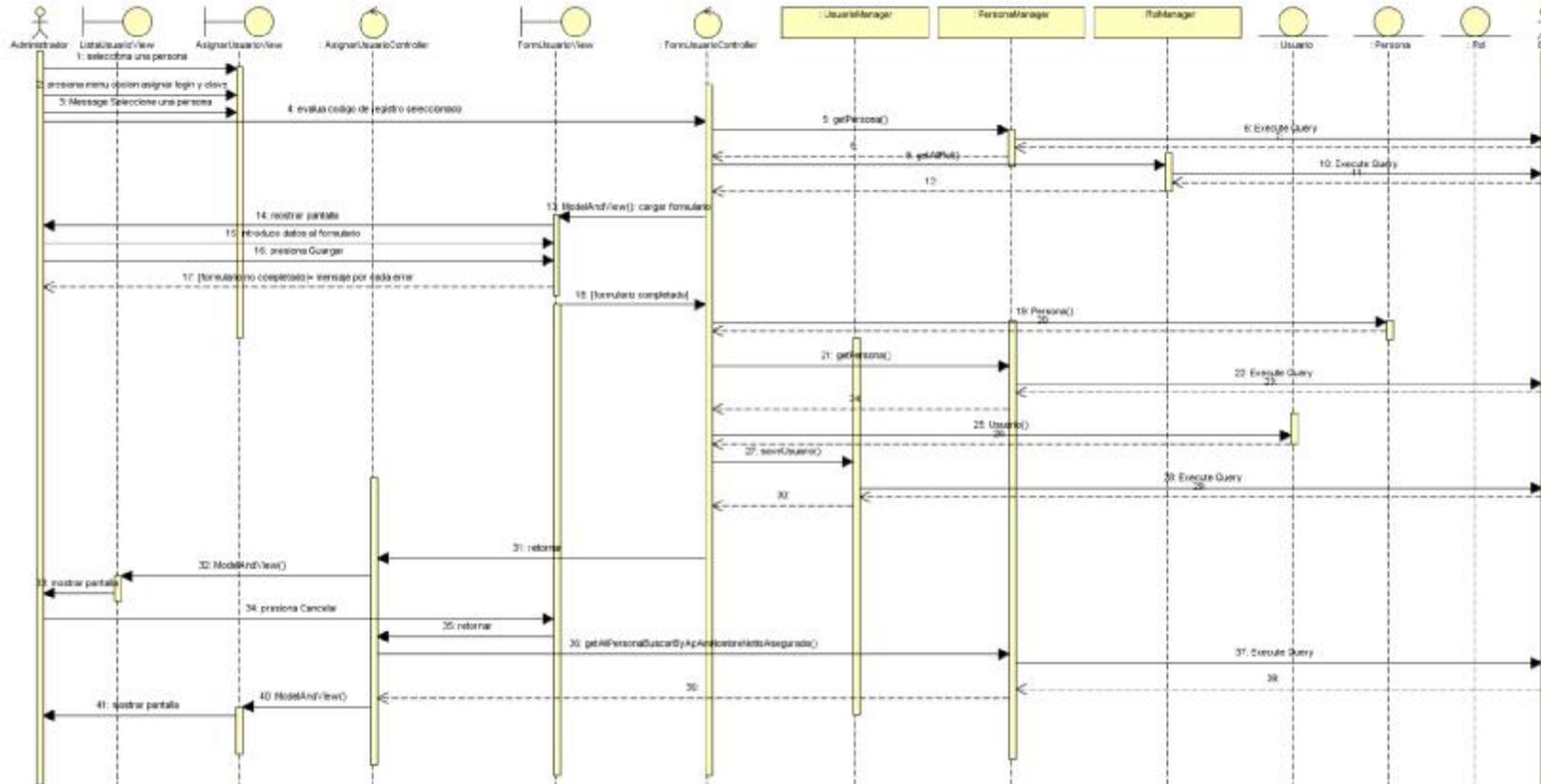


Figura 372. Diagrama de Secuencia: CU Asignar Login y Clave

2.1.2.2.3.9.2.1.80 Diagrama de Secuencia: Caso de Uso Administrar Zona

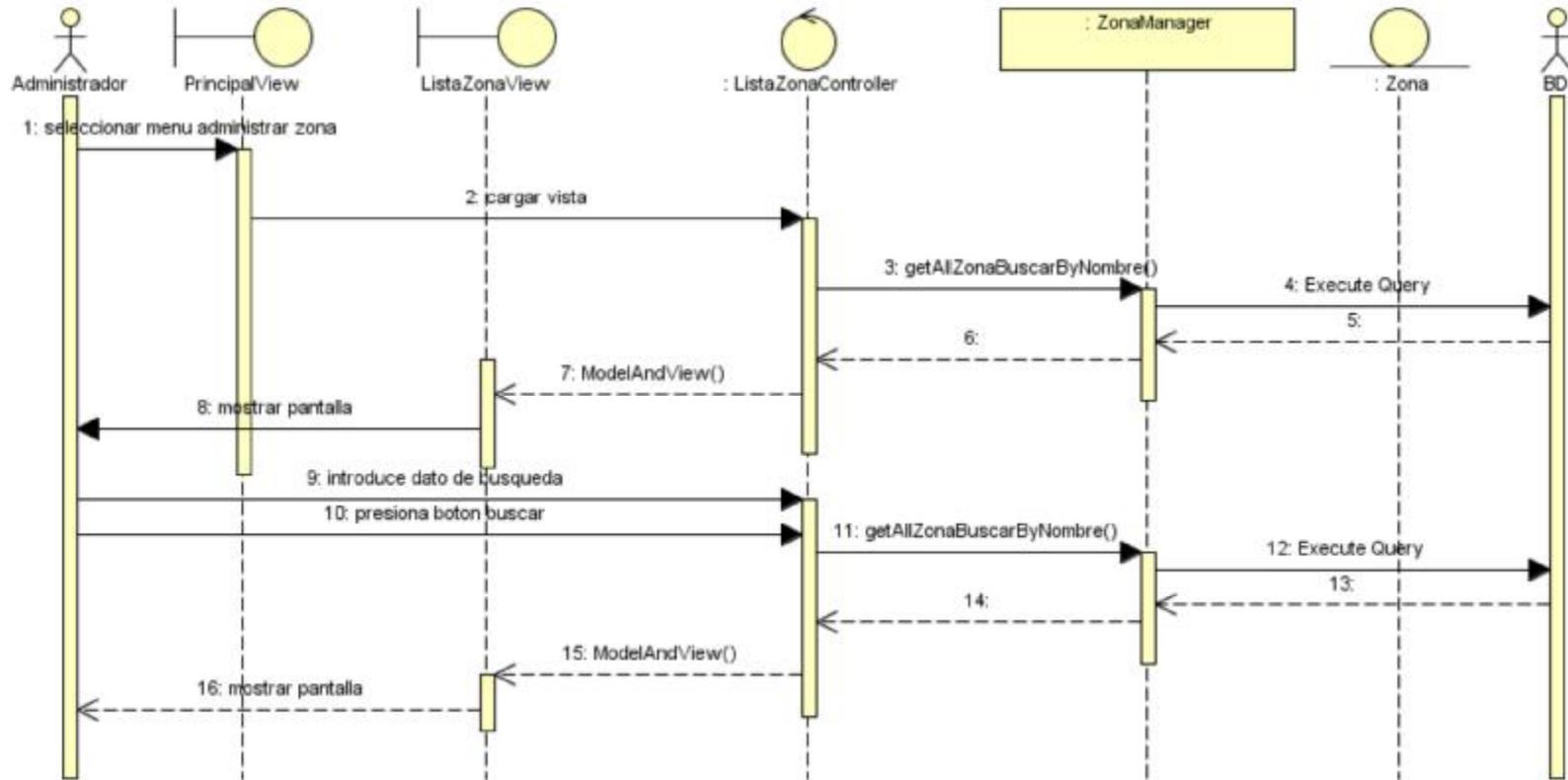


Figura 373. Diagrama de Secuencia: CU Administrar Zona

2.1.2.2.3.9.2.1.81 Diagrama de Secuencia: Caso de Uso Adicionar Zona

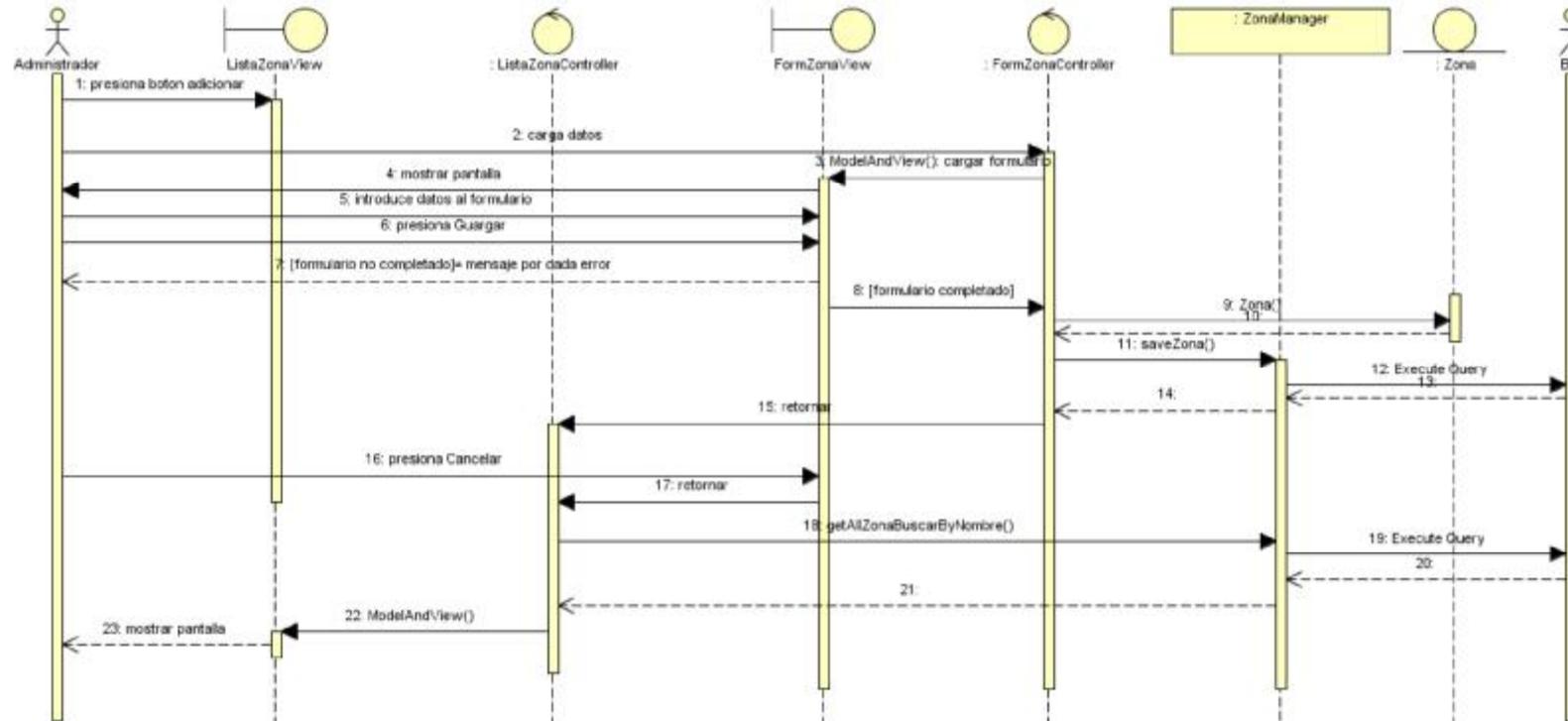


Figura 374. Diagrama de Secuencia: CU Adicionar Zona

2.1.2.2.3.9.2.1.82 Diagrama de Secuencia: Caso de Uso Modificar Zona

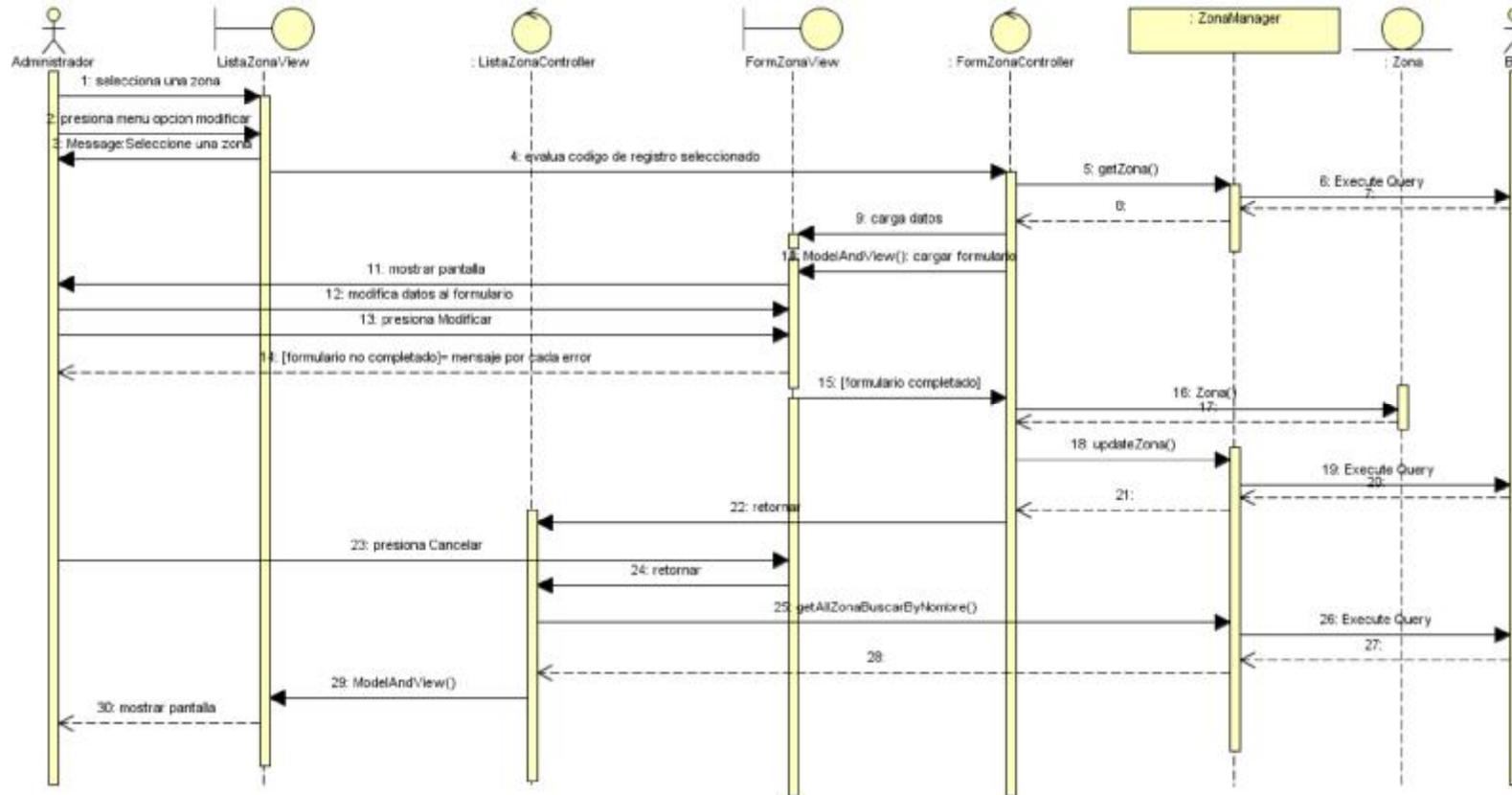


Figura 375. Diagrama de Secuencia: CU Modificar Zona

2.1.2.2.3.9.2.1.83 Diagrama de Secuencia: Caso de Uso Eliminar Zona

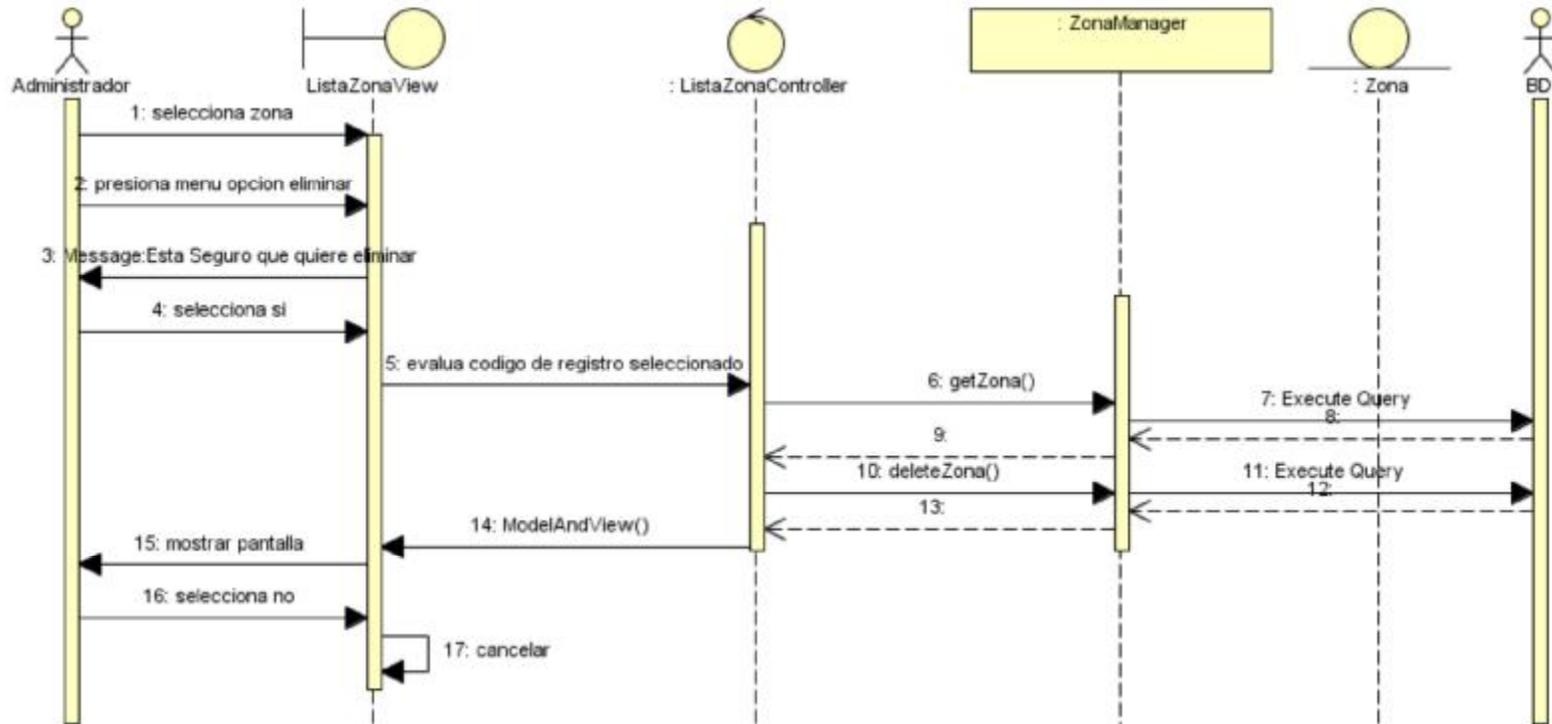


Figura 376. Diagrama de Secuencia: CU Eliminar Zona

2.1.2.2.3.9.2.1.84 Diagrama de Secuencia: Caso de Uso Cambiar Estado Zona

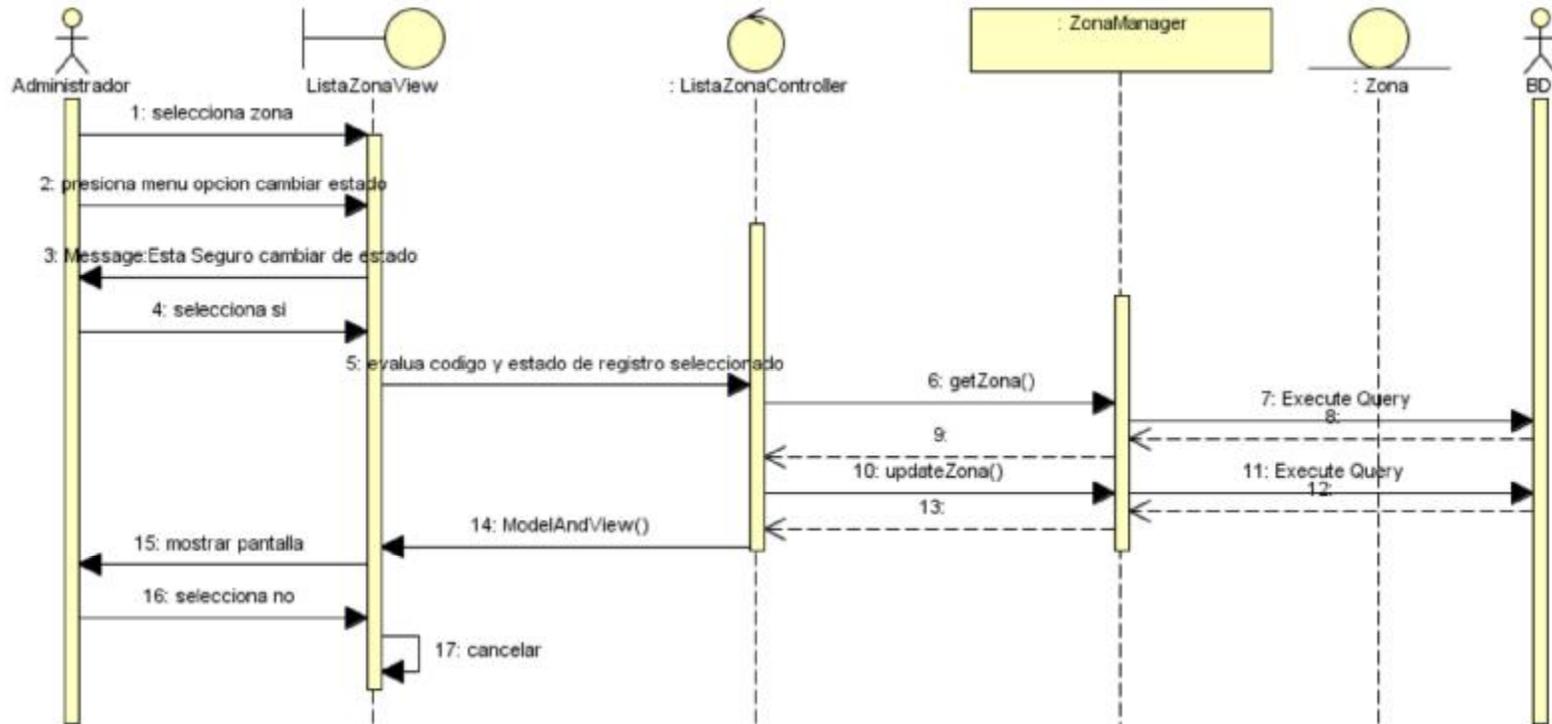


Figura 377. Diagrama de Secuencia: CU Cambiar Estado Zona

2.1.2.2.3.9.2.1.85 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Zona

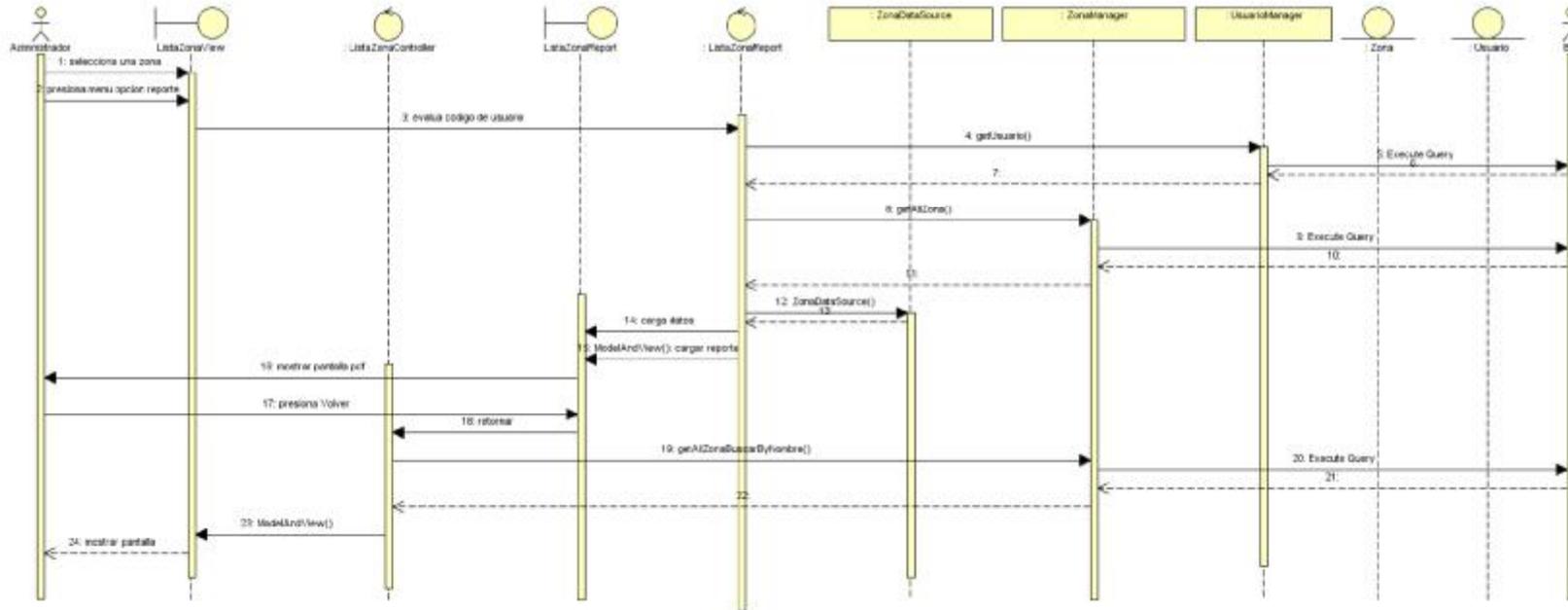


Figura 378. Diagrama de Secuencia: CU Generar Reporte Lista Zona

2.1.2.2.3.9.2.1.86 Diagrama de Secuencia: Caso de Uso Gestionar Asegurados por Tipo Seguro Afiliados

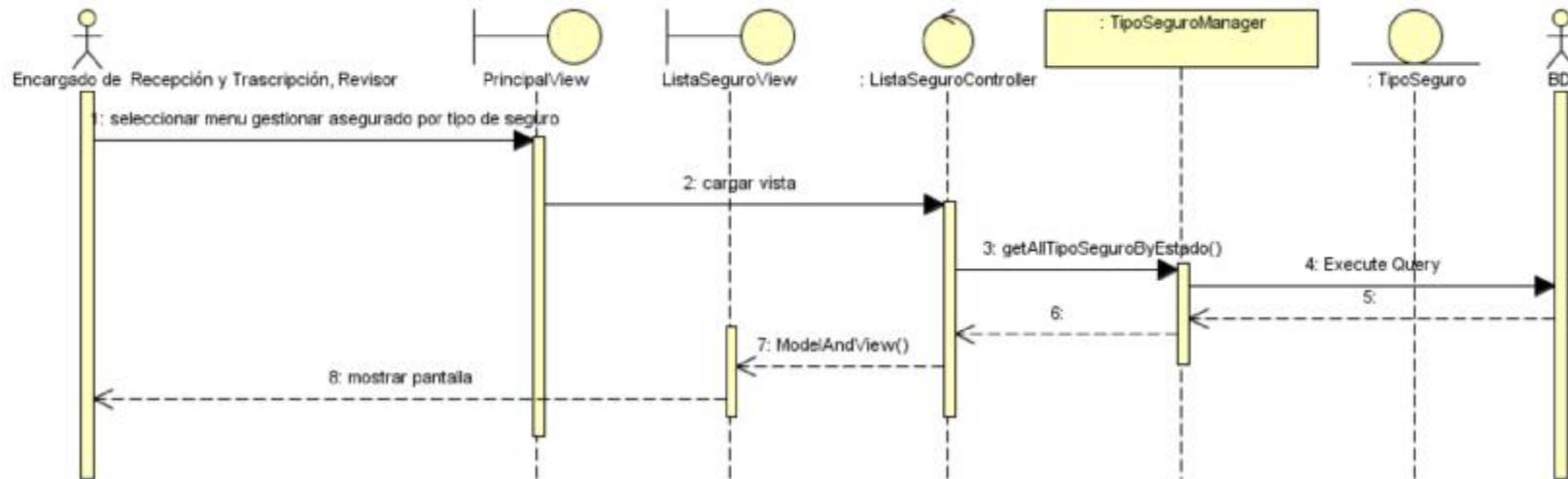


Figura 379. Diagrama de Secuencia: CU Gestionar Asegurados por Tipo Seguro Afiliados

2.1.2.2.3.9.2.1.87 Diagrama de Secuencia: Caso de Uso Gestionar Asegurados por Tipo Seguro

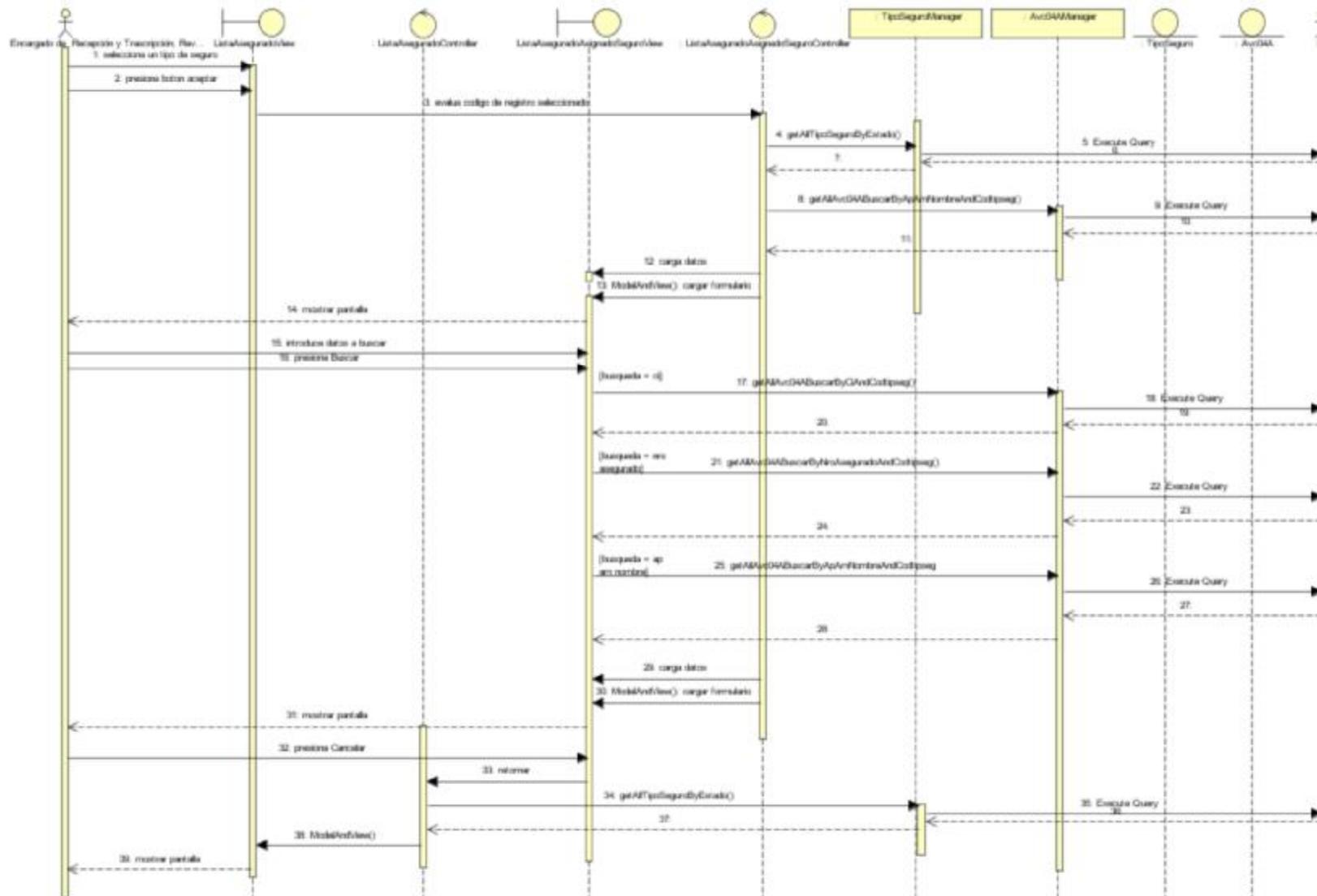


Figura 380. Diagrama de Secuencia: CU Gestionar Asegurados por Tipo Seguro

2.1.2.2.3.9.2.1.88 Diagrama de Secuencia: Caso de Uso Gestionar Personas No Aseguradas por Tipo de Seguro

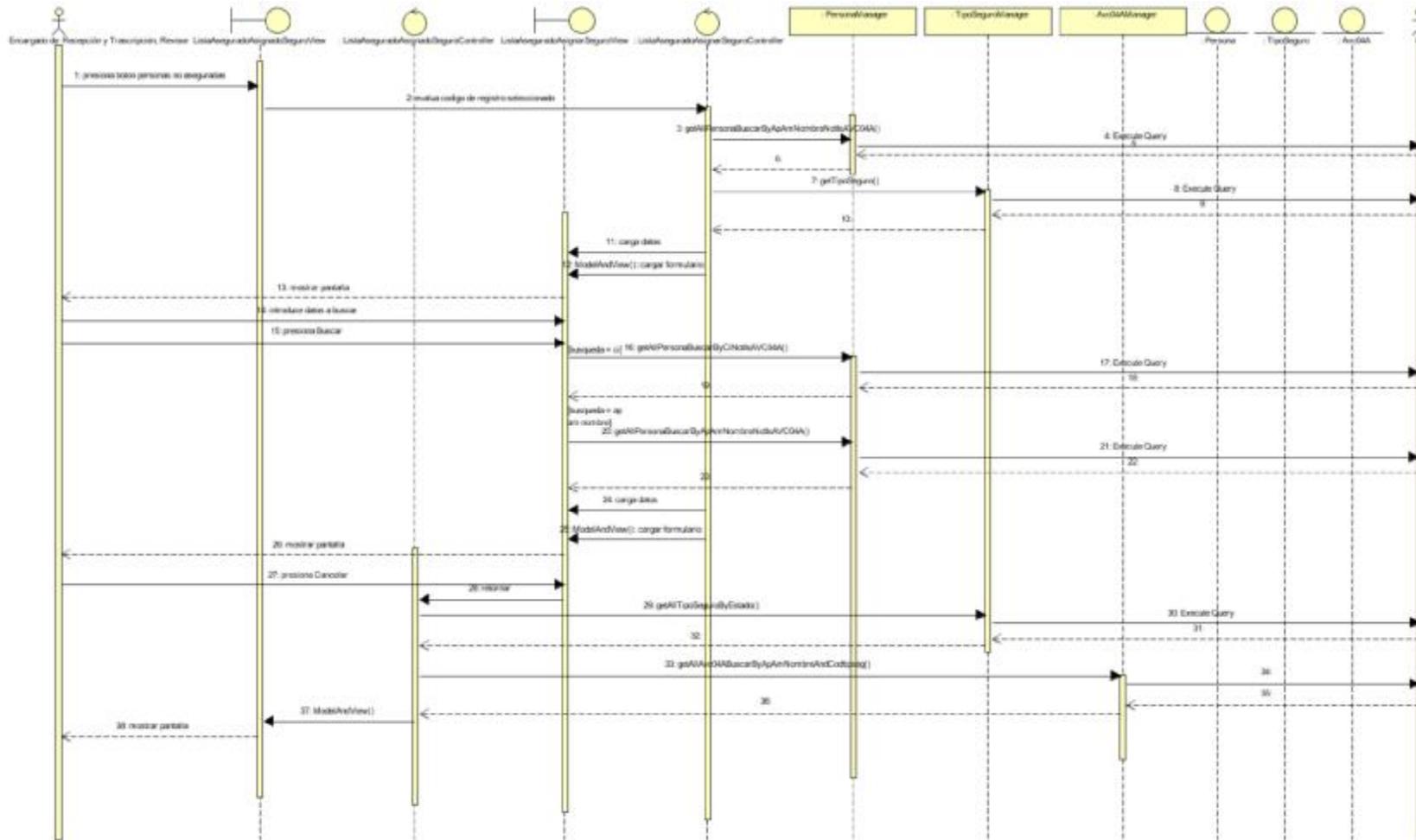


Figura 381. Diagrama de Secuencia: CU Gestionar Personas No Aseguradas por Tipo de Seguro

2.1.2.2.3.9.2.1.89 Diagrama de Secuencia: Caso de Uso Asegurar Persona por Tipo de Seguro

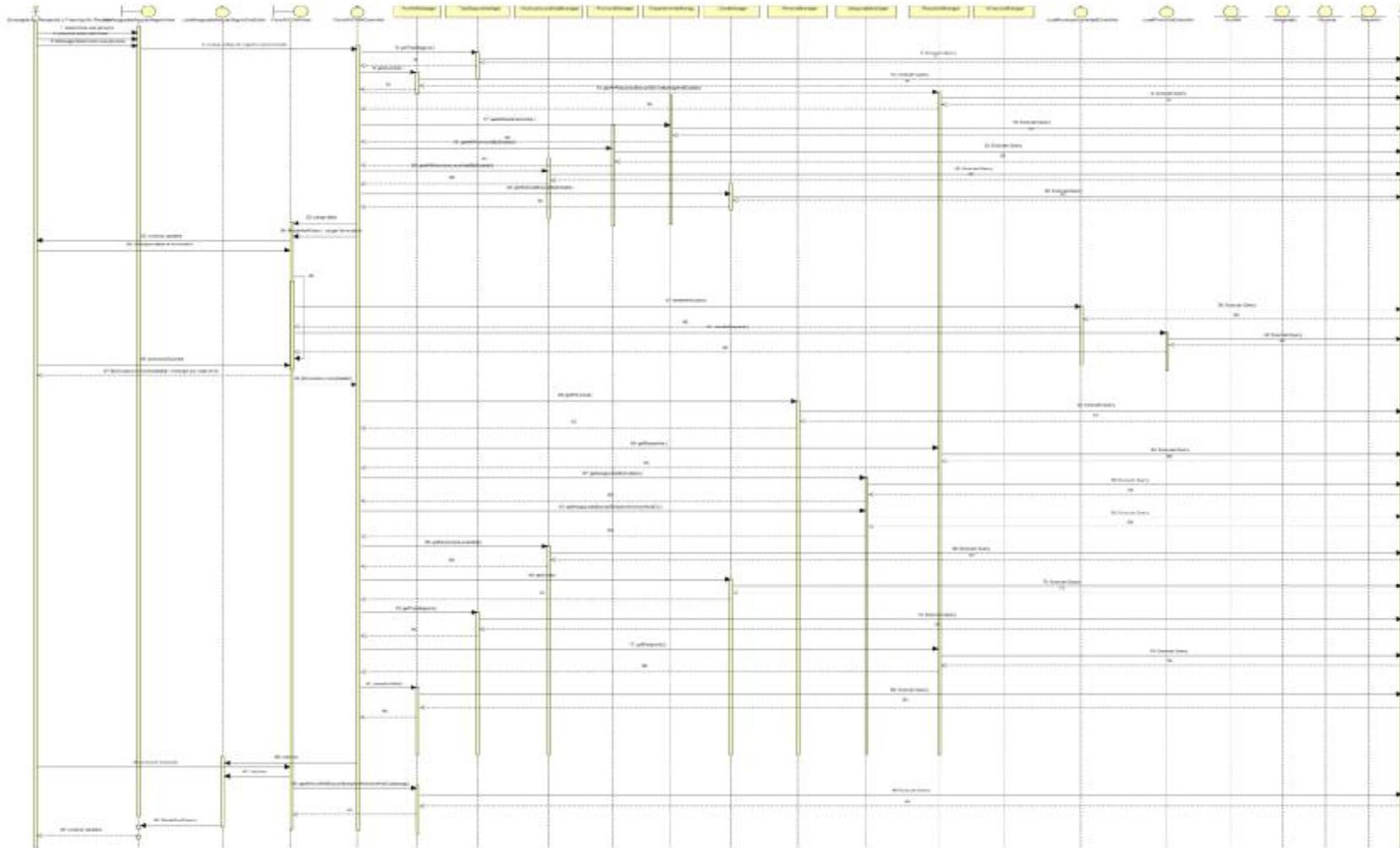


Figura 382. Diagrama de Secuencia: CU Asegurar Persona por Tipo de Seguro

Figura 383. Diagrama de Secuencia: CU Adicionar Asegurado por Tipo Seguro

2.1.2.2.3.9.2.1.91 Diagrama de Secuencia: Caso de Uso Eliminar Asegurado por Tipo Seguro

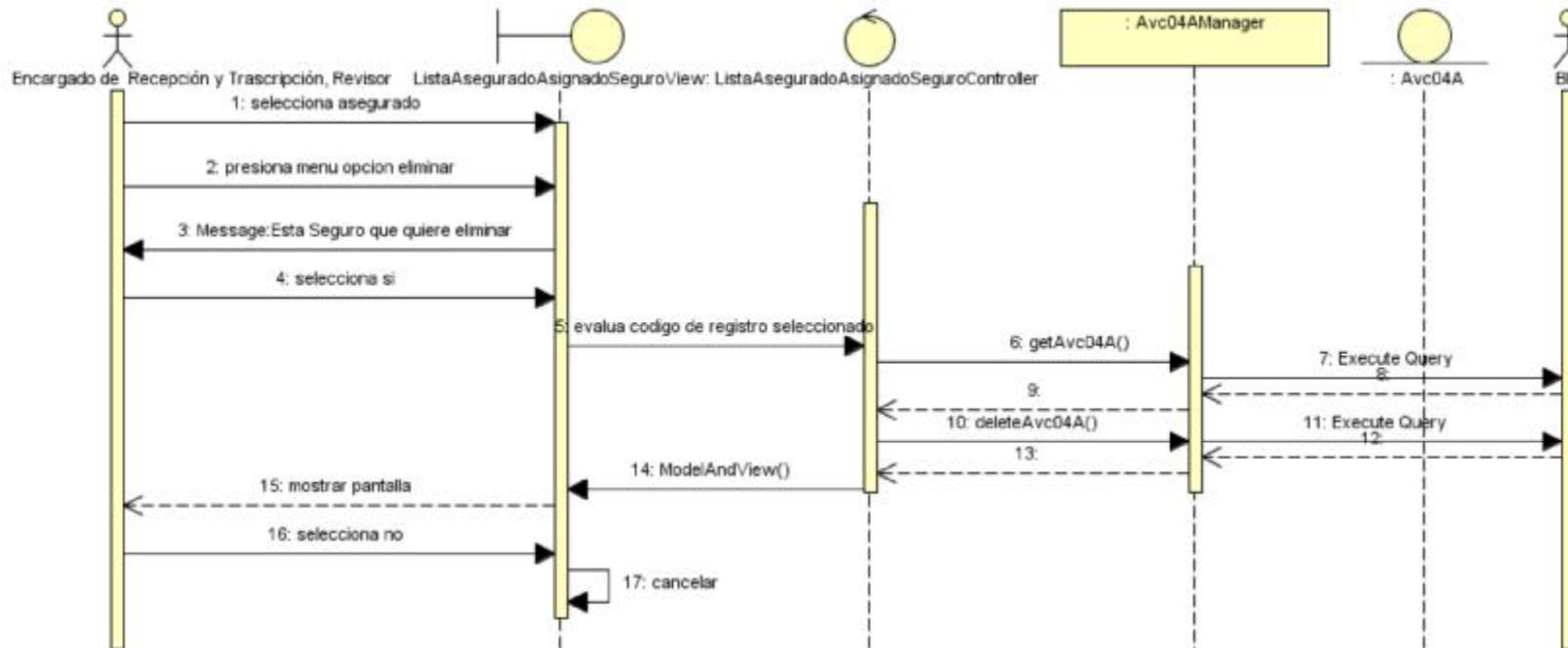


Figura 384. Diagrama de Secuencia: CU Eliminar Asegurado por Tipo Seguro

2.1.2.2.3.9.2.1.92 Diagrama de Secuencia: Caso de Uso Dar de Baja Asegurado por Tipo Seguro

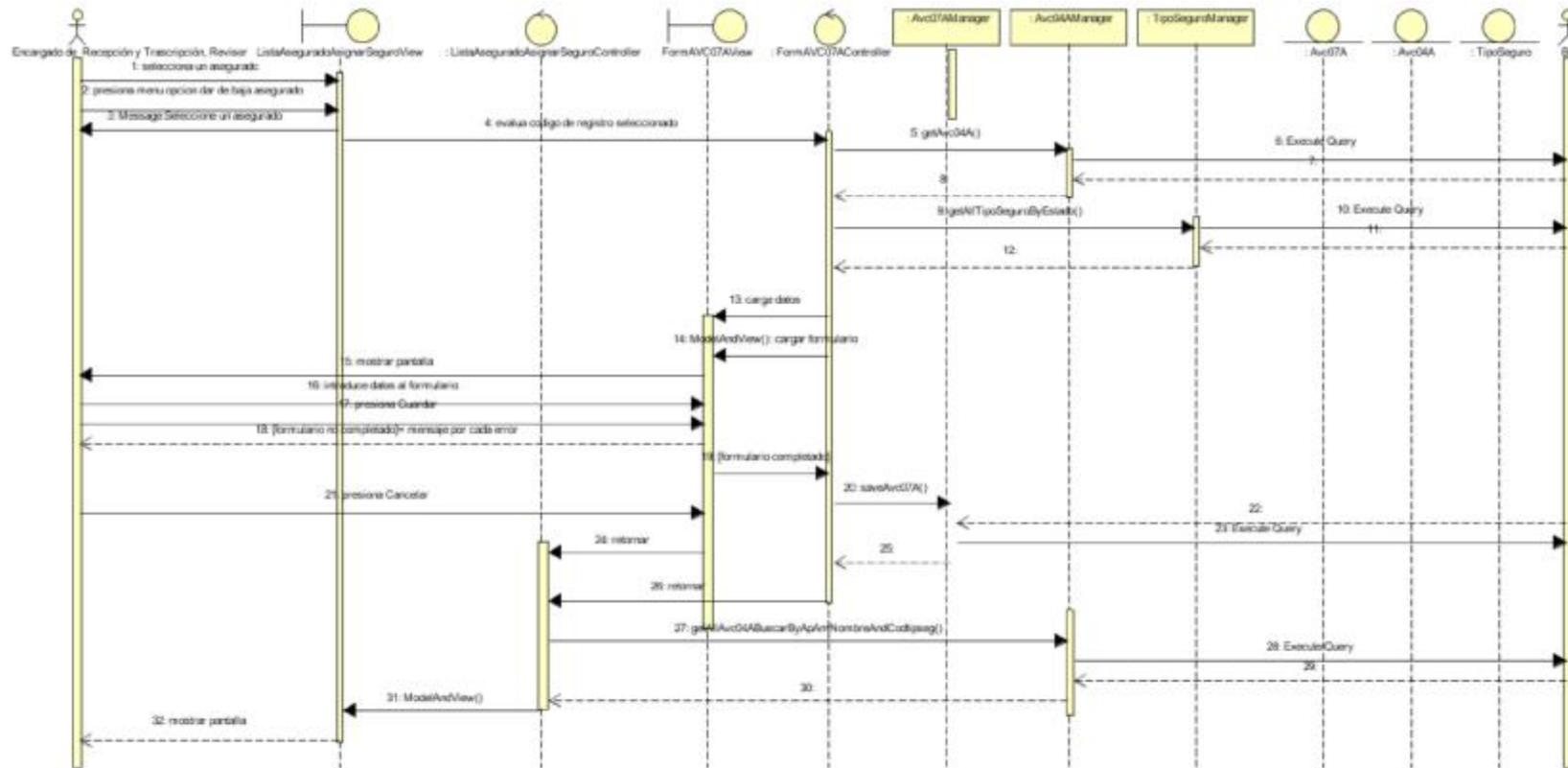


Figura 385. Diagrama de Secuencia: CU Dar de Baja Asegurado por Tipo Seguro

2.1.2.2.3.9.2.1.93 Diagrama de Secuencia: Caso de Uso Gestionar Empresas Afiliadas

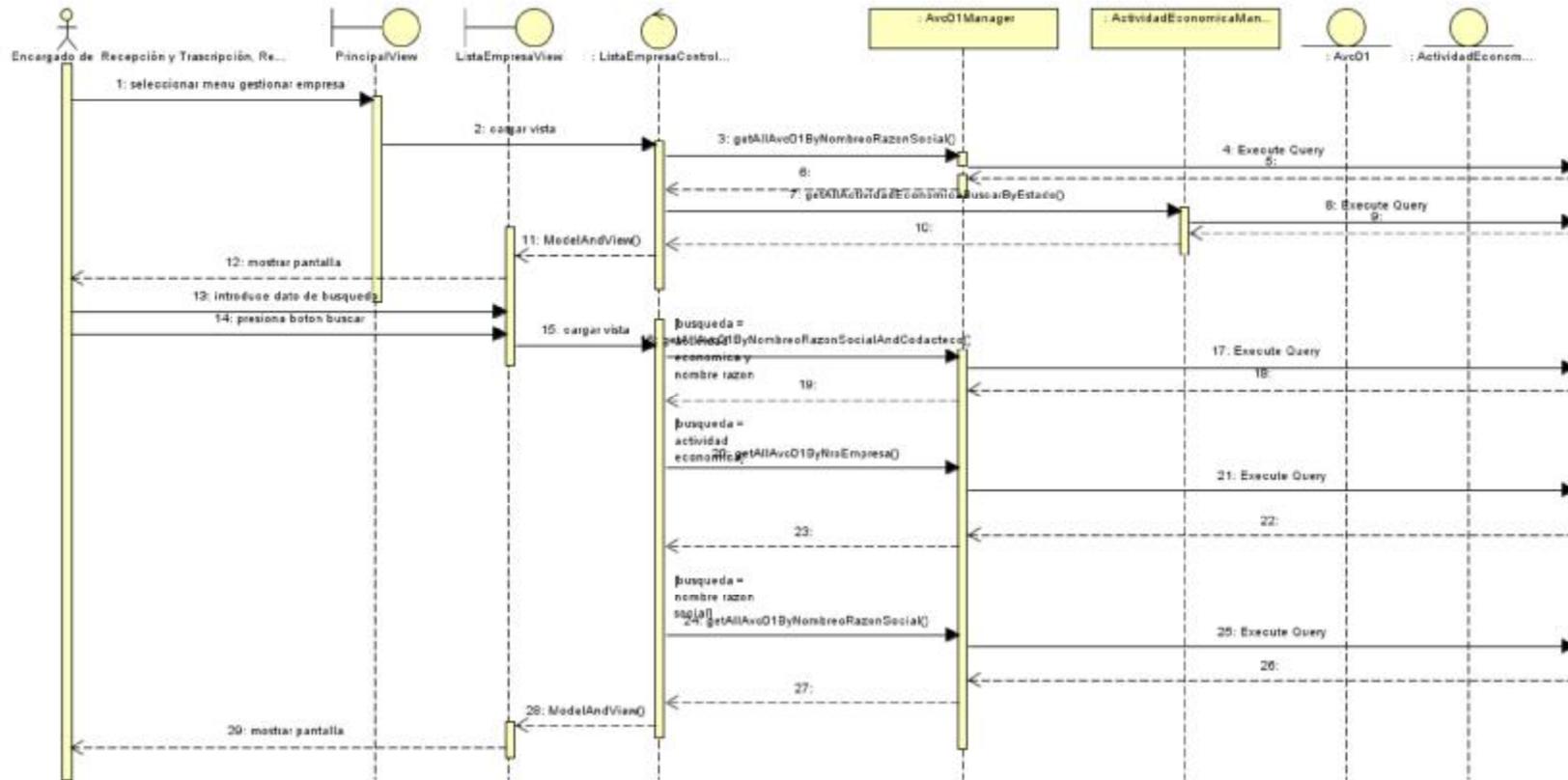


Figura 386. Diagrama de Secuencia: CU Gestionar Empresas Afiliadas

2.1.2.2.3.9.2.1.94 Diagrama de Secuencia: Caso de Uso Adicionar Empresa

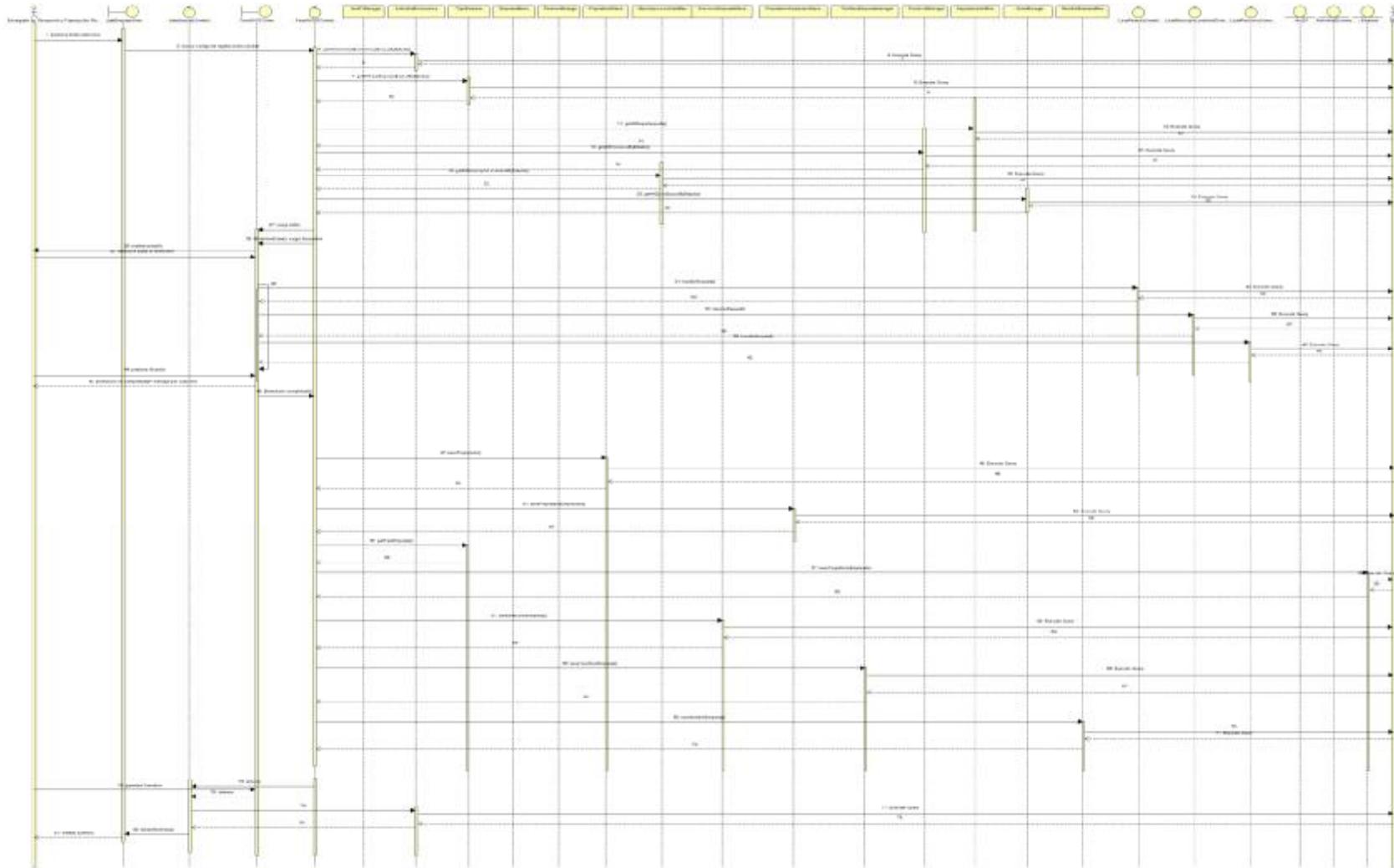


Figura 387. Diagrama de Secuencia: CU Adicionar Empresa

2.1.2.2.3.9.2.1.95 Diagrama de Secuencia: Caso de Uso Eliminar Empresa

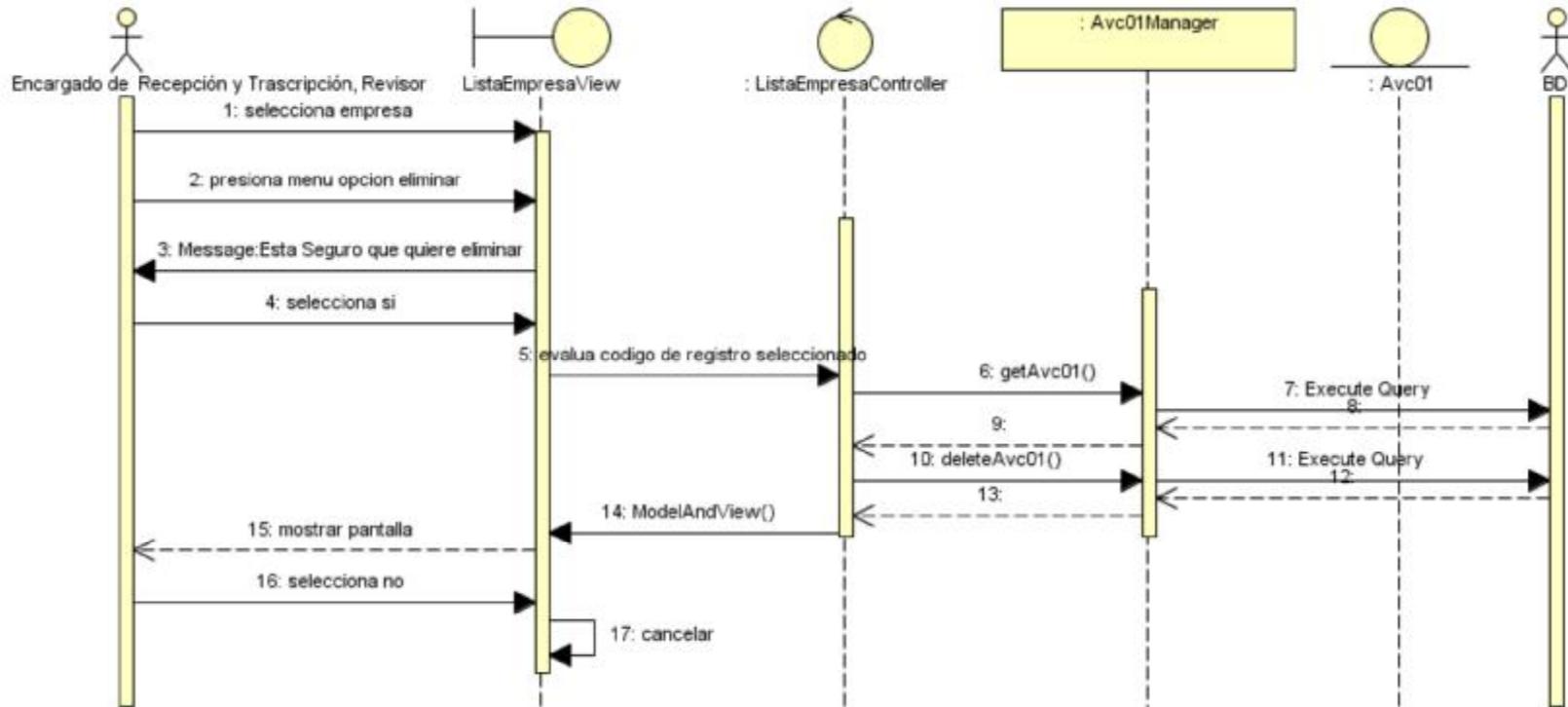


Figura 388. Diagrama de Secuencia: CU Eliminar Empresa

2.1.2.2.3.9.2.1.96 Diagrama de Secuencia: Caso de Uso Aviso Novedades Empresa

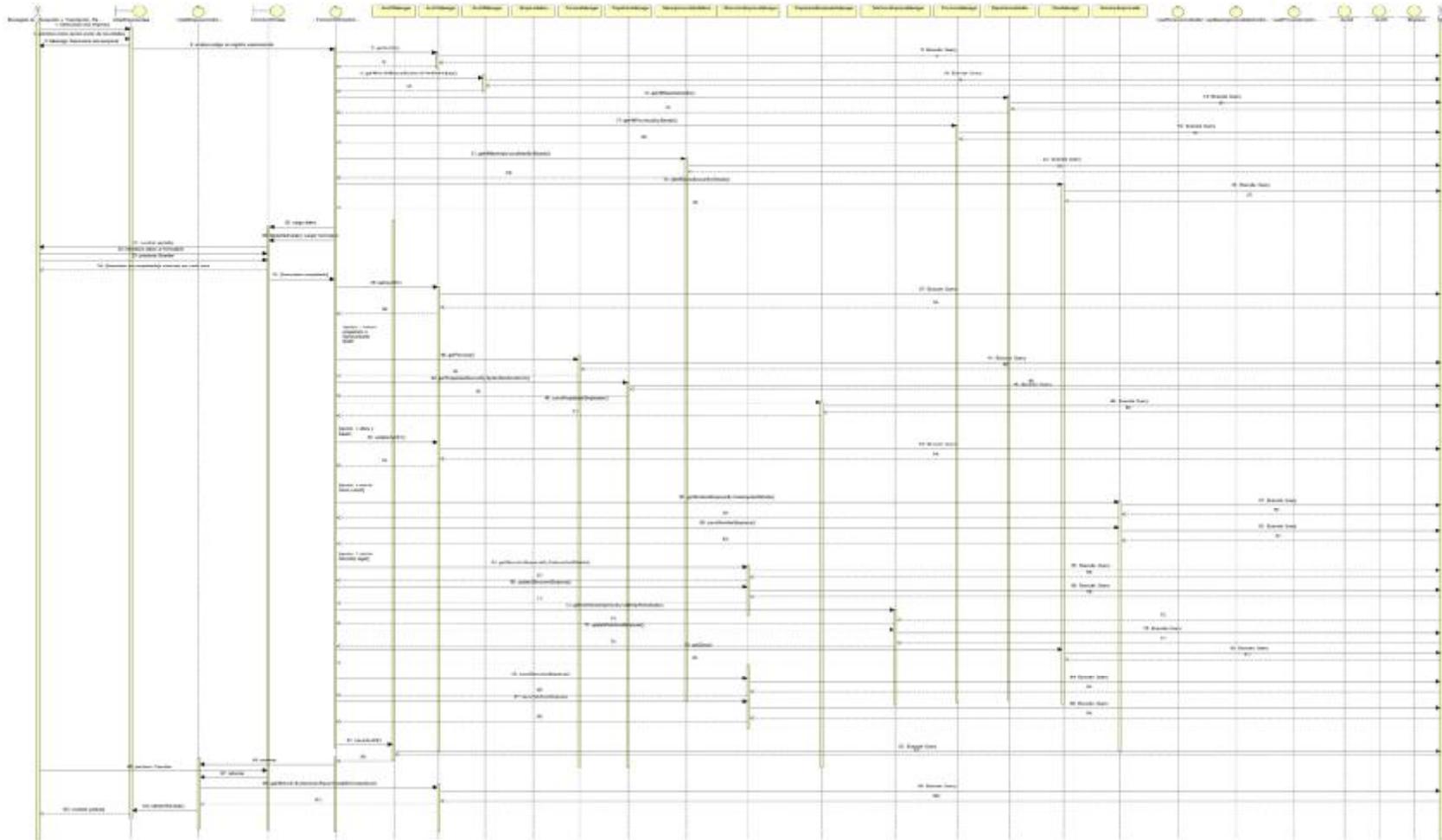


Figura 389. Diagrama de Secuencia: CU Aviso Novedades Empresa

2.1.2.2.3.9.2.1.97 Diagrama de Secuencia: Caso de Uso Dar de Alta Empresa por Norma

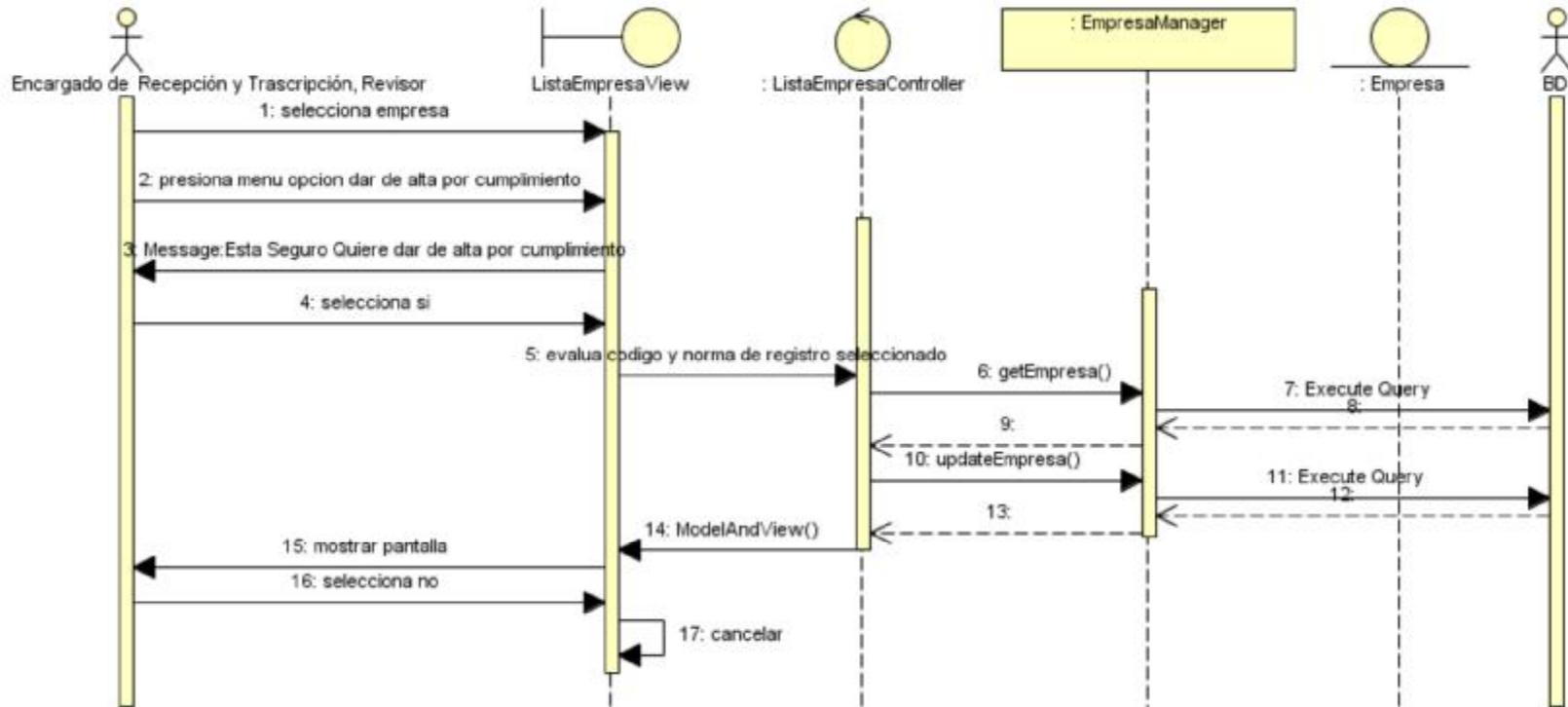


Figura 390. Diagrama de Secuencia: CU Dar de Alta Empresa por Norma

2.1.2.2.3.9.2.1.98 Diagrama de Secuencia: Caso de Uso Dar de Baja Empresa por Norma

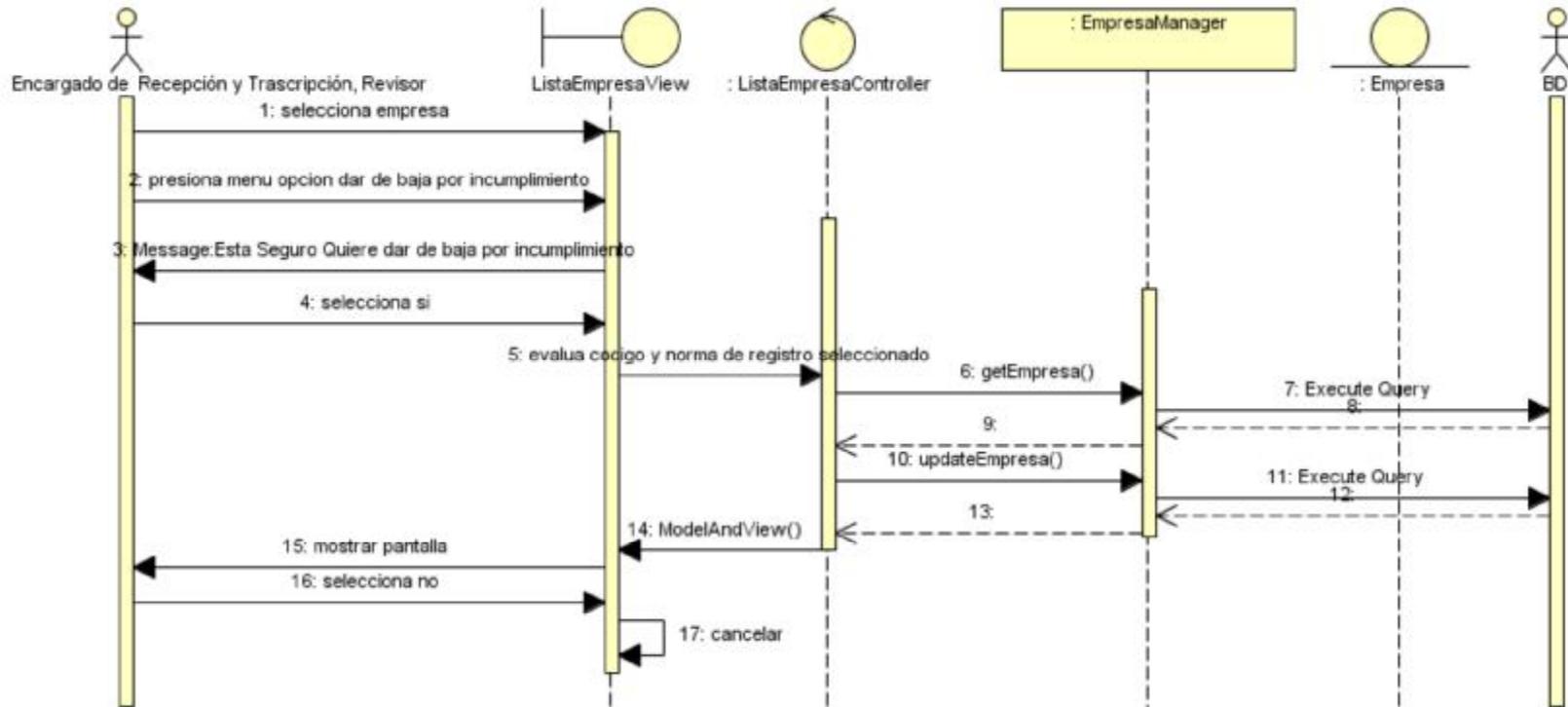


Figura 391. Diagrama de Secuencia: CU Dar de Baja Empresa por Norma

2.1.2.2.3.9.2.1.99 Diagrama de Secuencia: Caso de Uso Gestionar Asegurados Afiliados

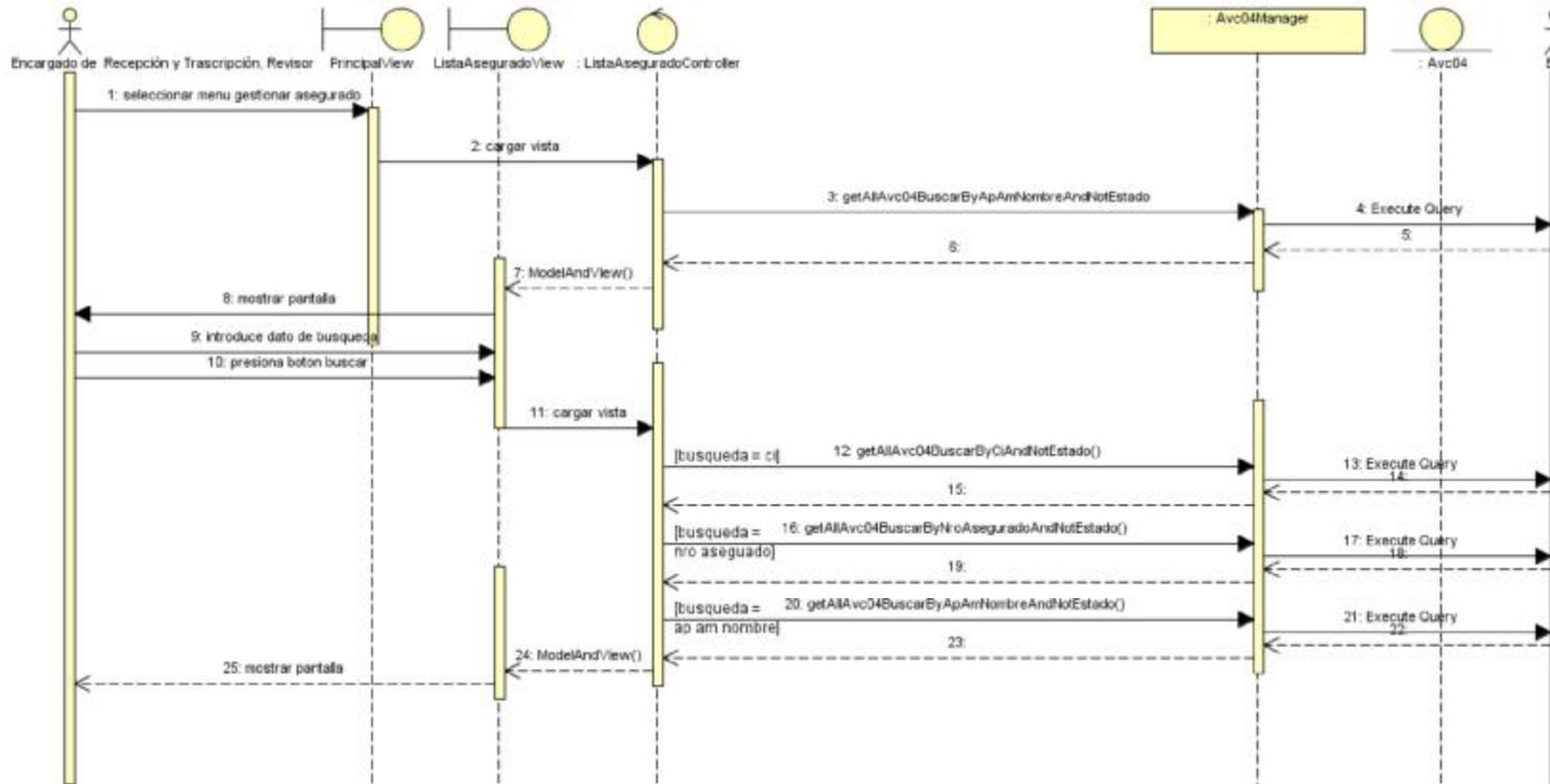


Figura 392. Diagrama de Secuencia: CU Gestionar Asegurados Afiliados

2.1.2.2.3.9.2.1.100 Diagrama de Secuencia: Caso de Uso Reingreso Asegurado

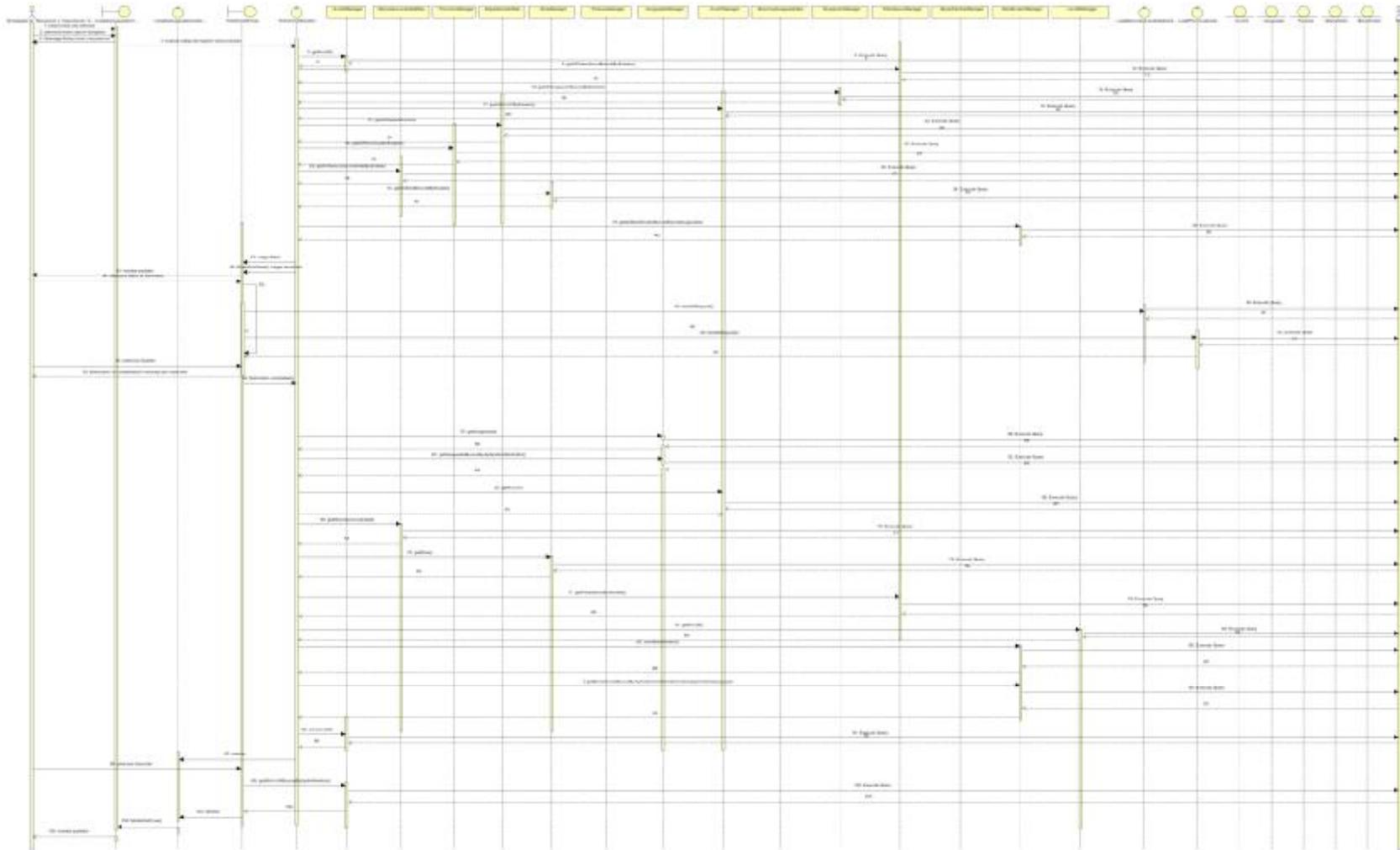


Figura 393. Diagrama de Secuencia: CU Reingreso Asegurado

2.1.2.2.3.9.2.1.101 Diagrama de Secuencia: Caso de Uso Eliminar Asegurado

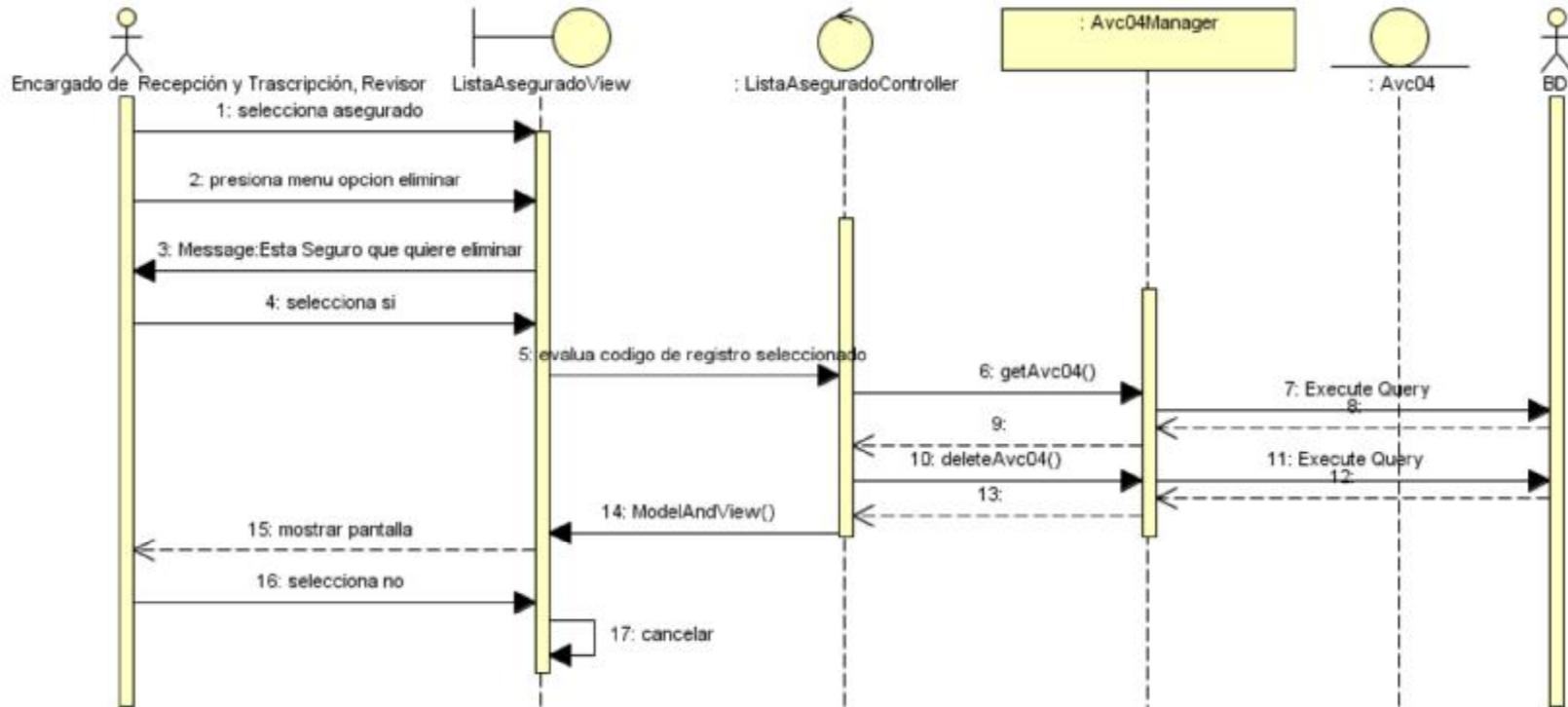


Figura 394. Diagrama de Secuencia: CU Eliminar Asegurado

2.1.2.2.3.9.2.1.102 Diagrama de Secuencia: Caso de Uso Dar de Baja Asegurado

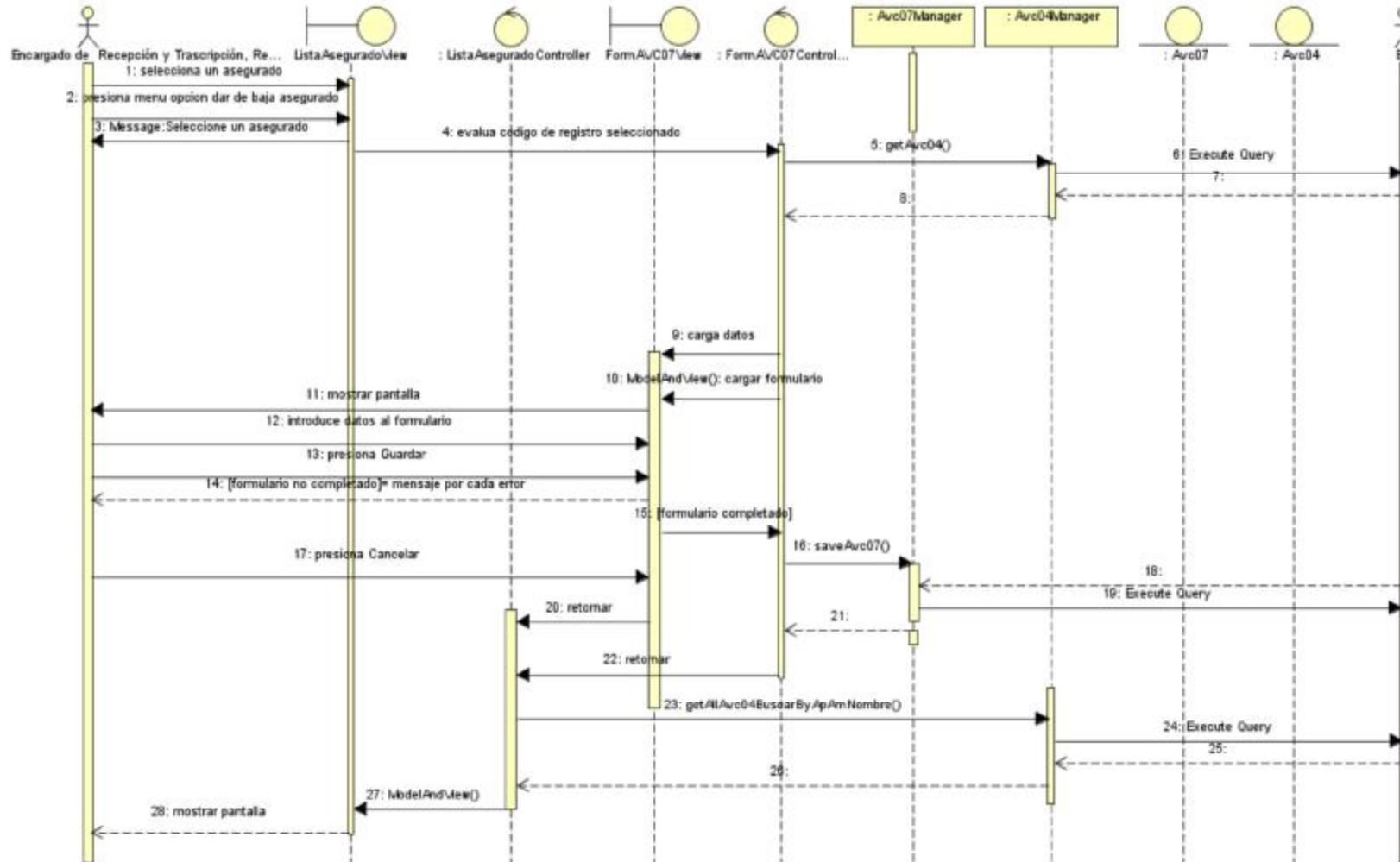


Figura 395. Diagrama de Secuencia: CU Dar de Baja Asegurado

2.1.2.2.3.9.2.1.103

Diagrama de Secuencia: Caso de Uso Dar de Alta Asegurado por Norma

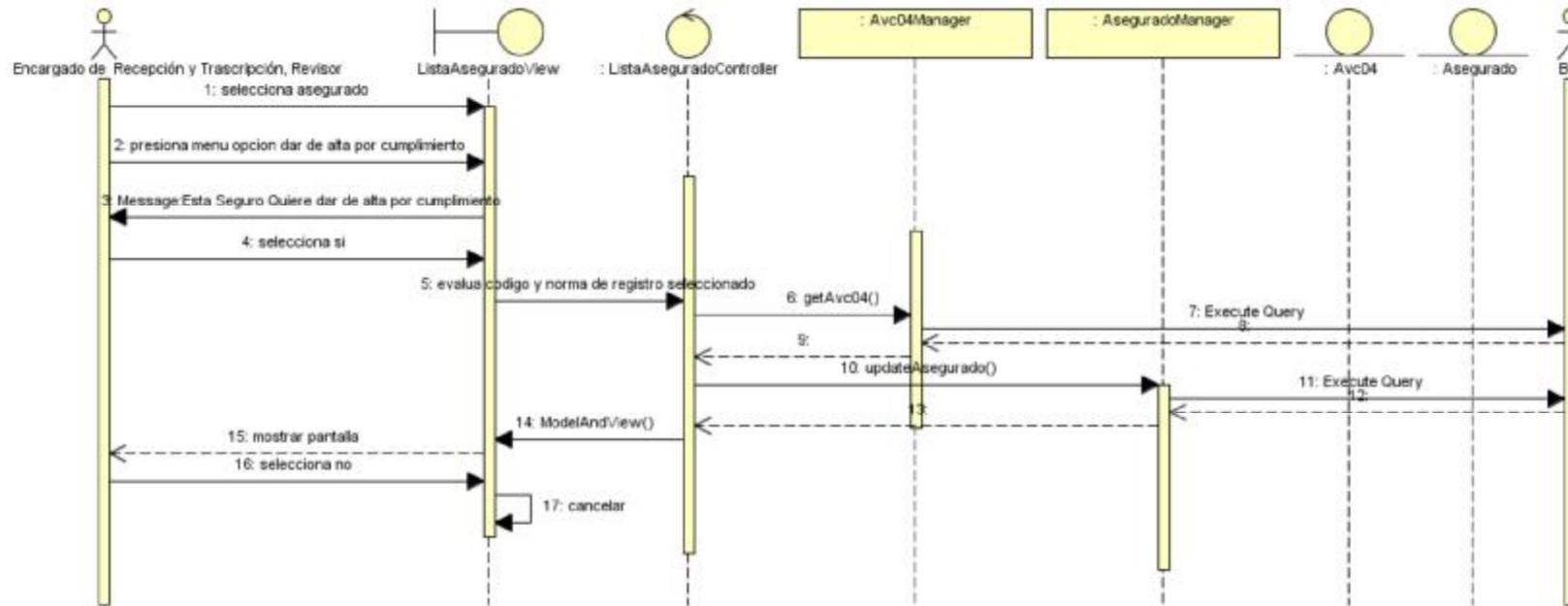


Figura 396. Diagrama de Secuencia: CU Dar de Alta Asegurado por Norma

2.1.2.2.3.9.2.1.104 Diagrama de Secuencia: Caso de Uso Dar de Baja Asegurado por Norma

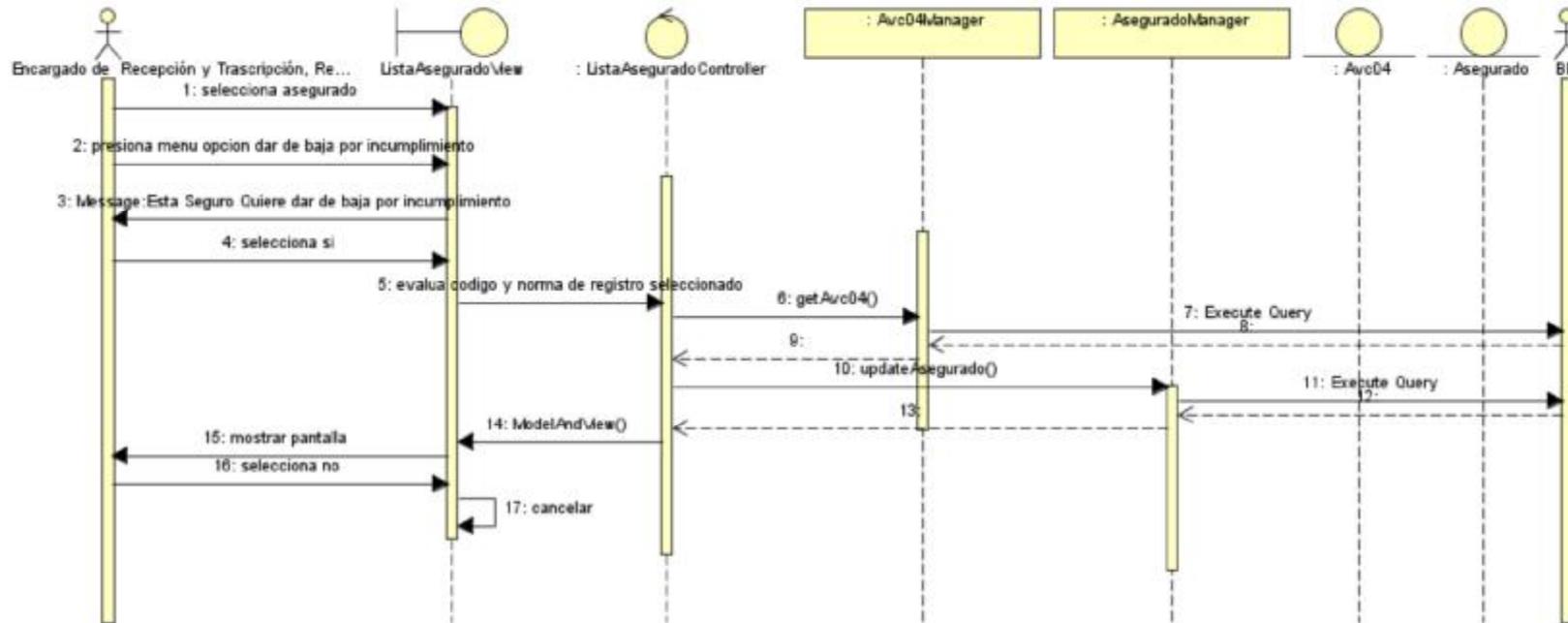


Figura 397. Diagrama de Secuencia: CU Dar de Baja Asegurado por Norma

2.1.2.2.3.9.2.1.105 Diagrama de Secuencia: Caso de Uso Gestionar Personas No Aseguradas

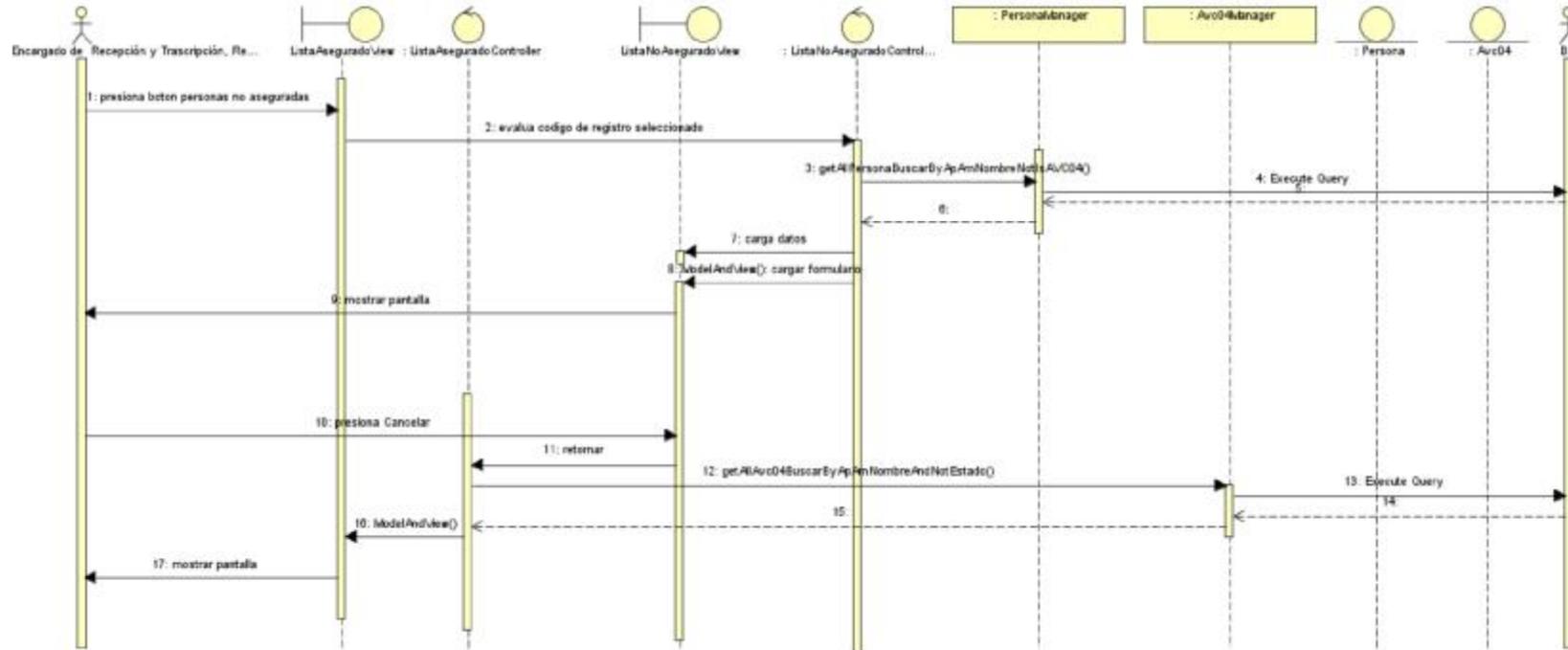


Figura 398. Diagrama de Secuencia: CU Gestionar Personas No Aseguradas

2.1.2.2.3.9.2.1.106

Diagrama de Secuencia: Caso de Uso Adicionar Asegurado

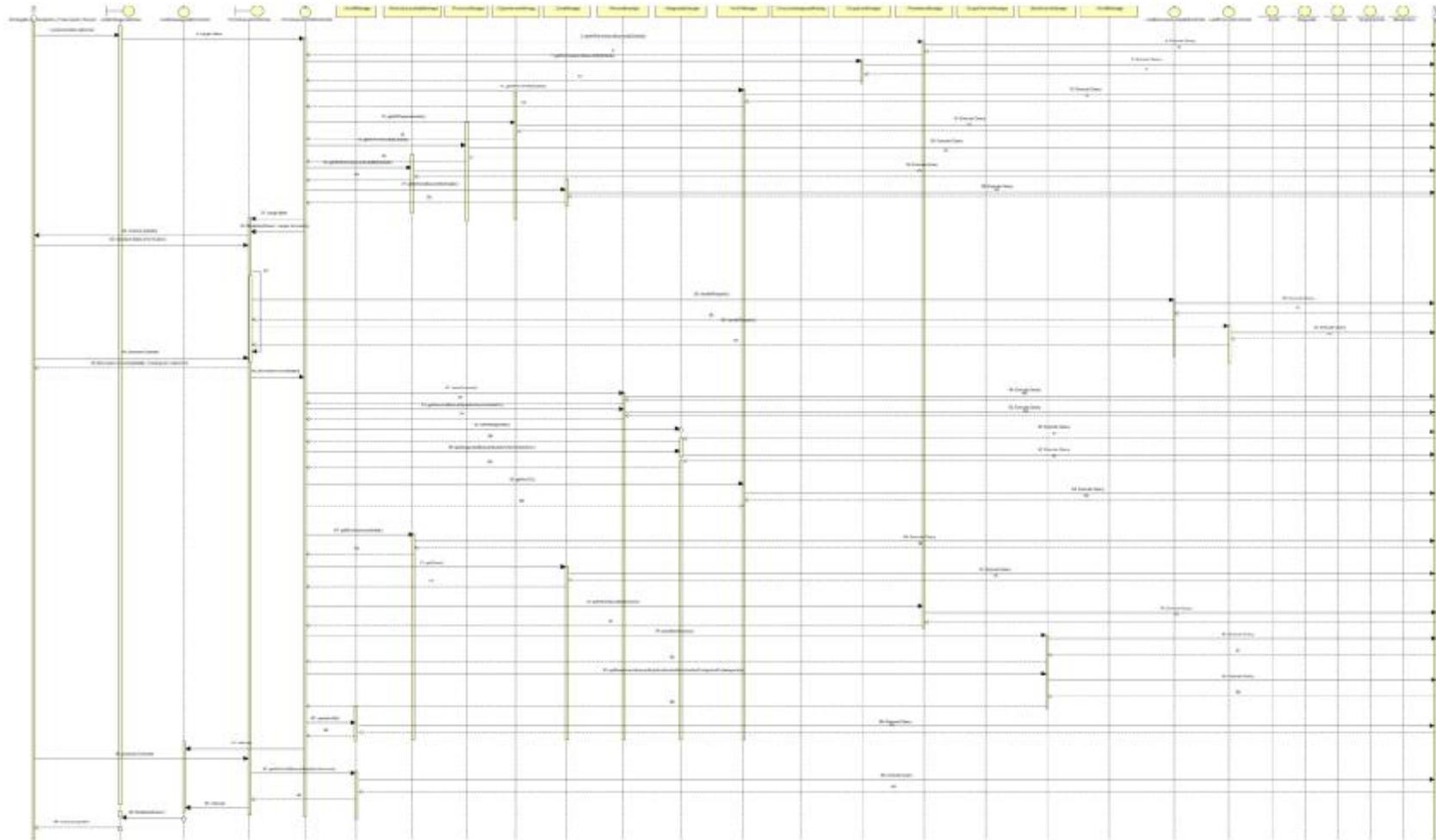


Figura 399. Diagrama de Secuencia: CU Adicionar Asegurado

2.1.2.2.3.9.2.1.107

Diagrama de Secuencia: Caso de Uso Asegurar Persona

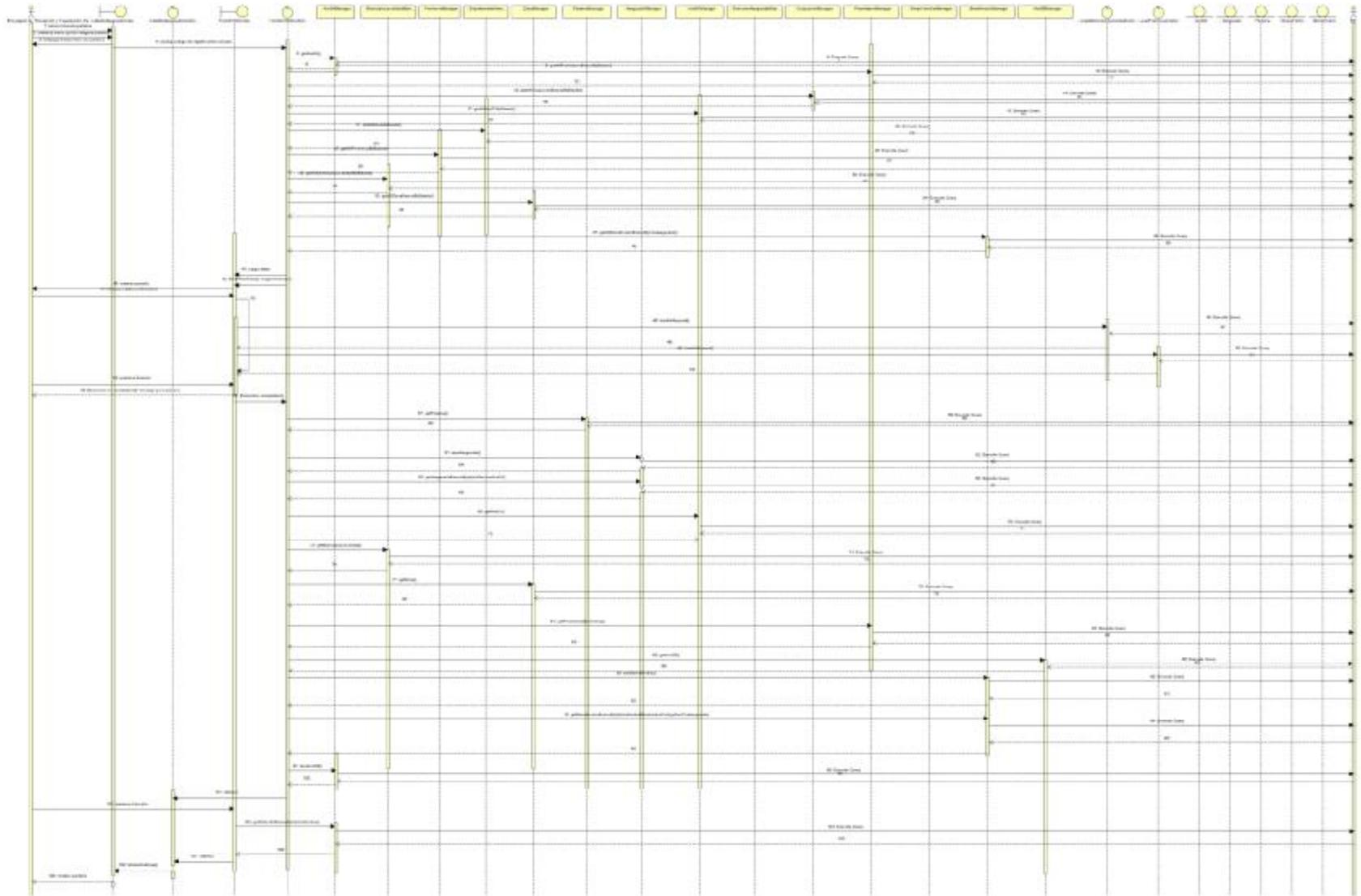


Figura 400. Diagrama de Secuencia: CU Asegurar Persona

2.1.2.2.3.9.2.1.108

Diagrama de Secuencia: Caso de Uso Gestionar Emitir Autorización Médica Externa

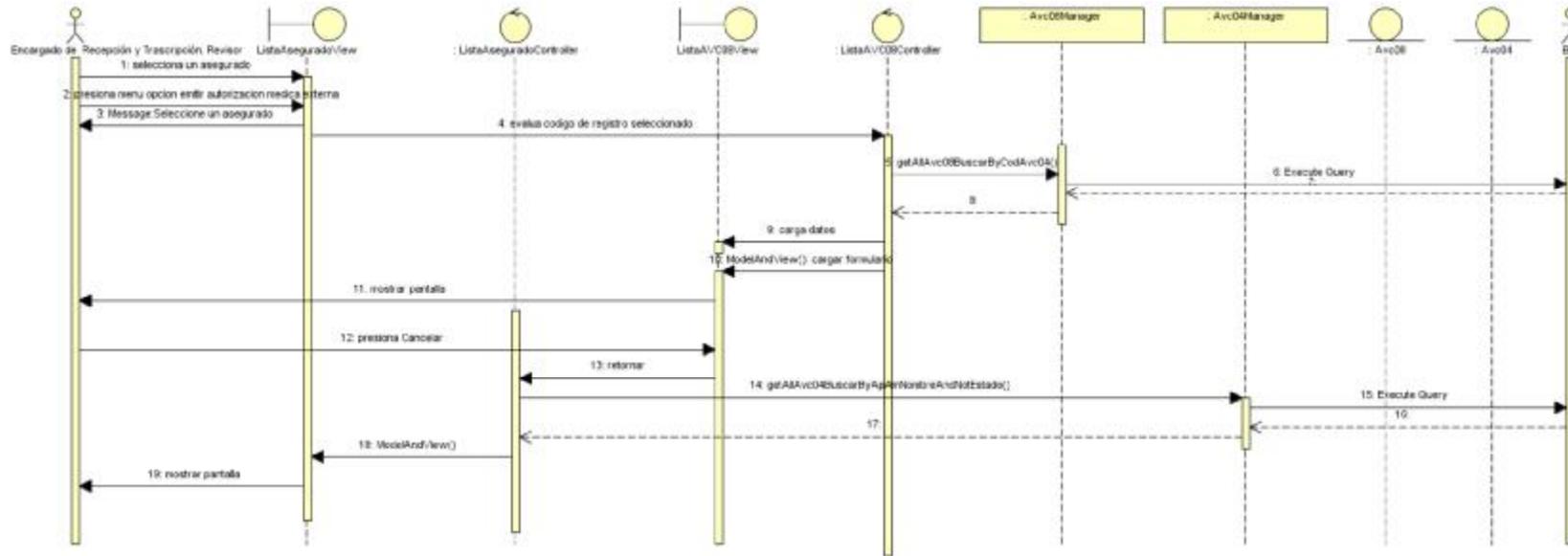


Figura 401. Diagrama de Secuencia: CU Gestionar Emitir Autorización Médica Externa

2.1.2.2.3.9.2.1.109 Diagrama de Secuencia: Caso de Uso Adicionar Autorización Médica Externa

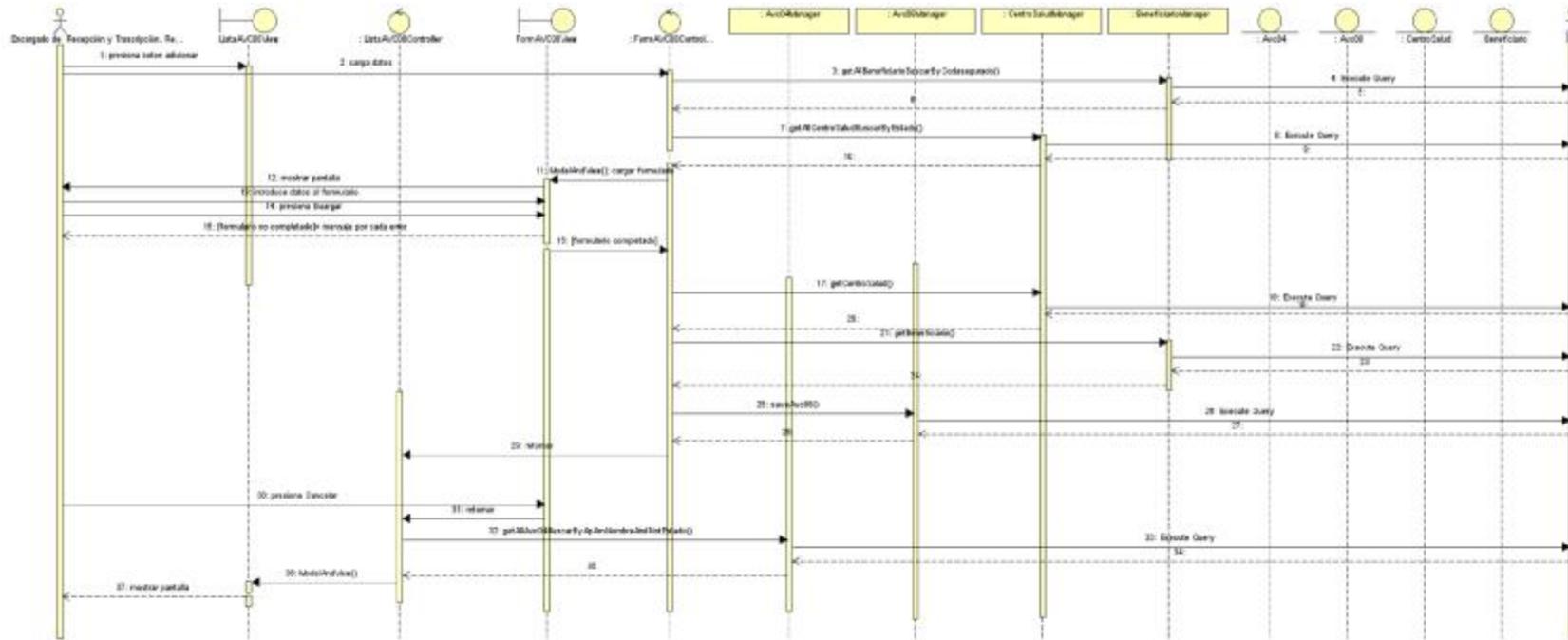


Figura 402. Diagrama de Secuencia: CU Adicionar Autorización Médica Externa

2.1.2.2.3.9.2.1.110 Diagrama de Secuencia: Caso de Uso Modificar Autorización Médica Externa

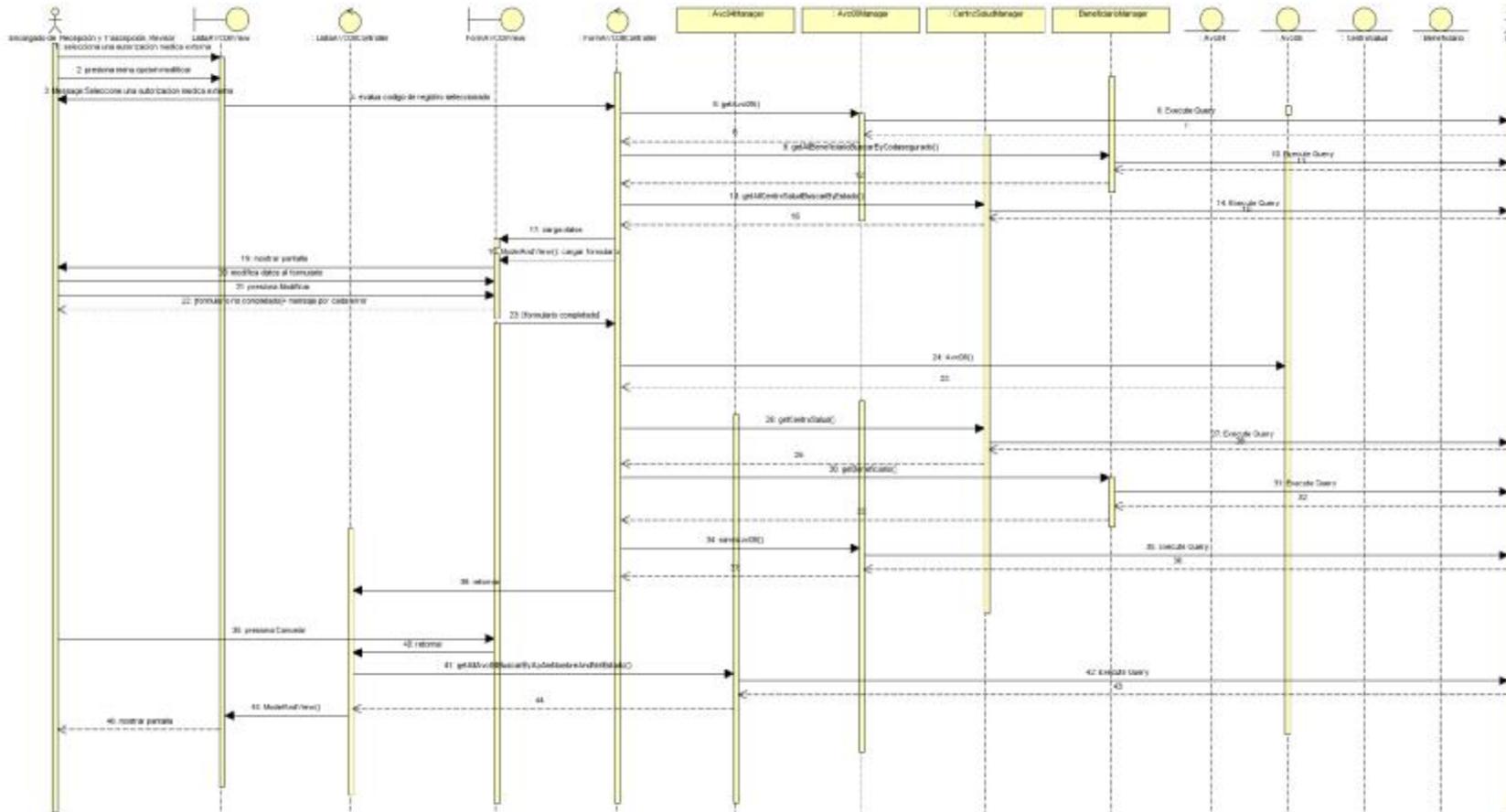


Figura 403. Diagrama de Secuencia: CU Modificar Autorización Médica Externa

2.1.2.2.3.9.2.1.111 Diagrama de Secuencia: Caso de Uso Eliminar Autorización Médica Externa

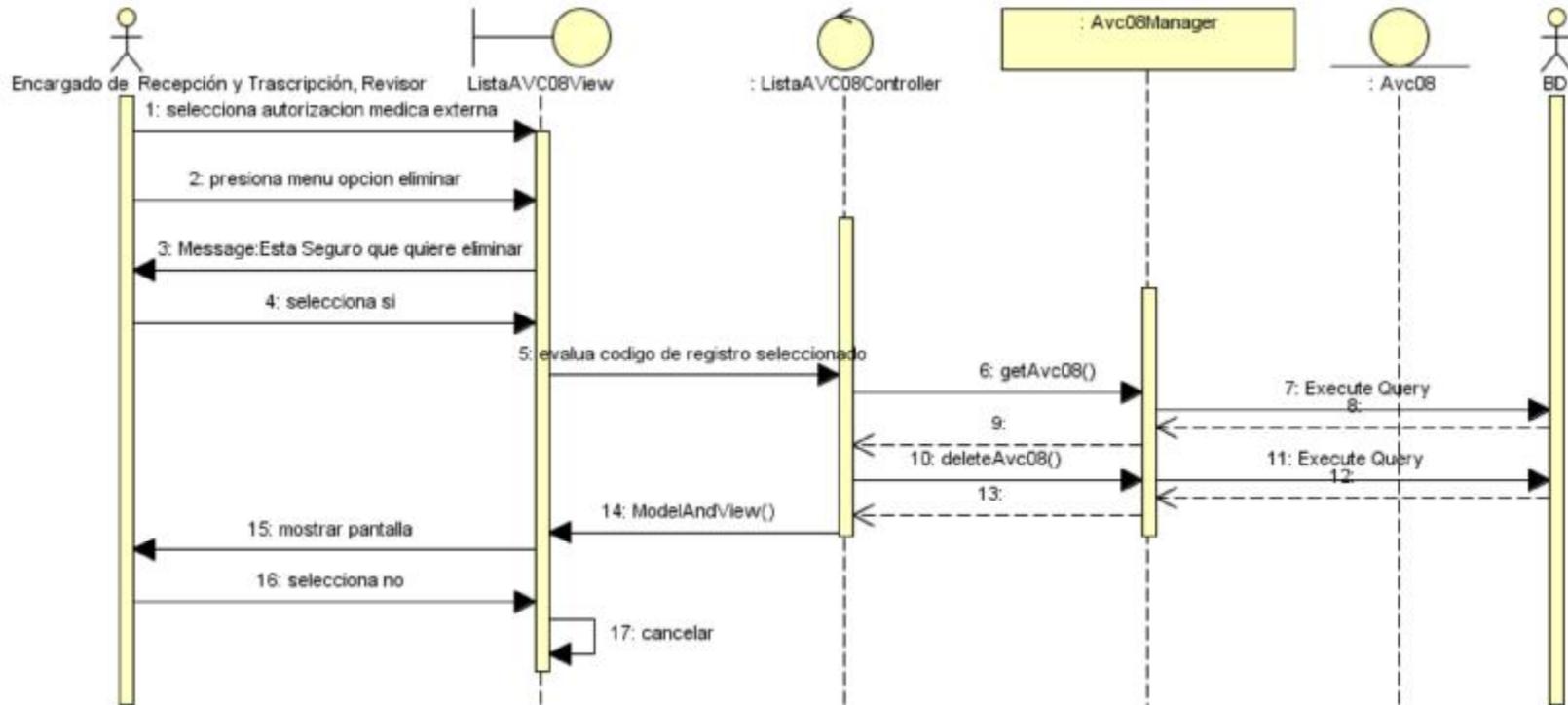


Figura 404. Diagrama de Secuencia: CU Eliminar Autorización Médica Externa

2.1.2.2.3.9.2.1.112 Diagrama de Secuencia: Caso de Uso Cambiar Estado Autorización Médica Externa

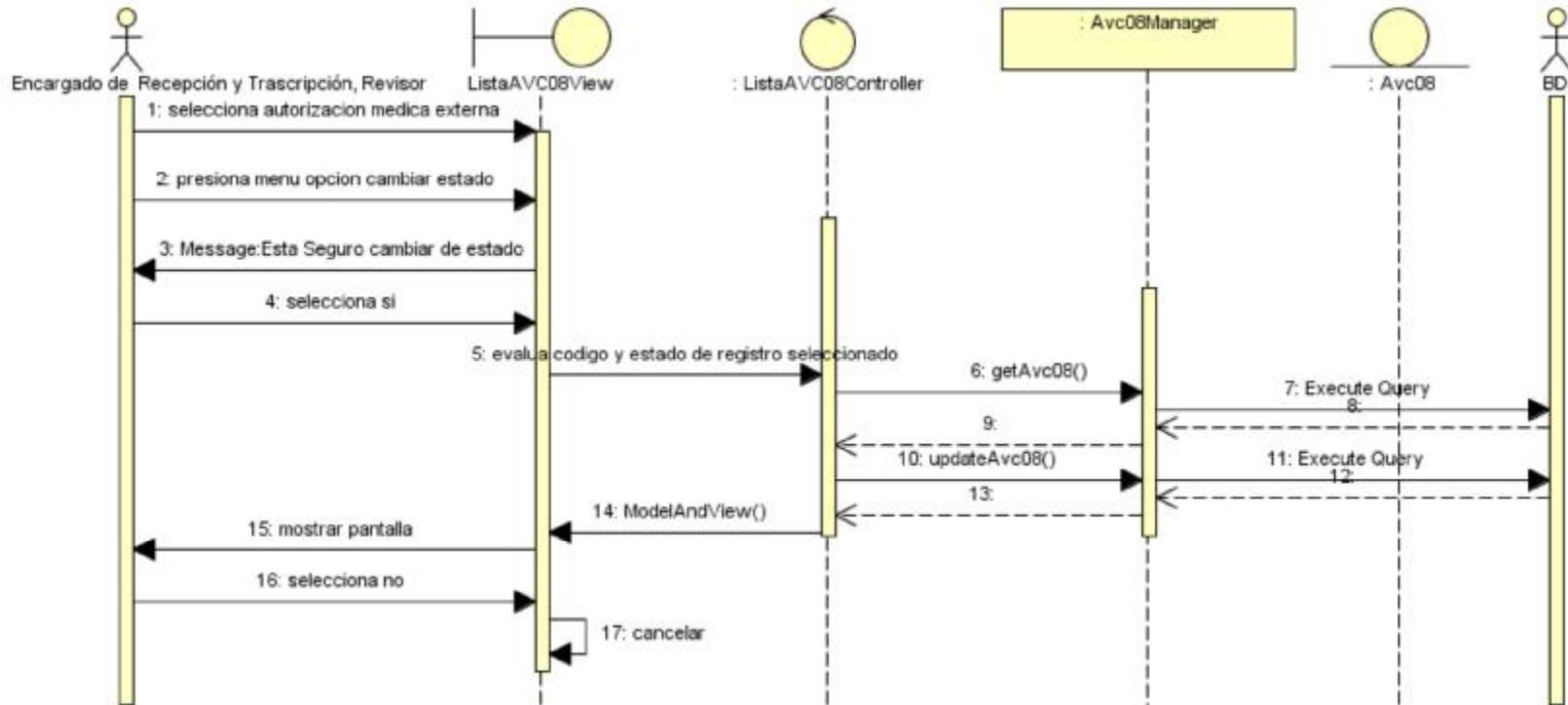


Figura 405. Diagrama de Secuencia: CU Cambiar Estado Autorización Médica Externa

2.1.2.2.3.9.2.1.113

Diagrama de Secuencia: Caso de Uso Gestionar Emitir Certificado de Incapacidad Temporal

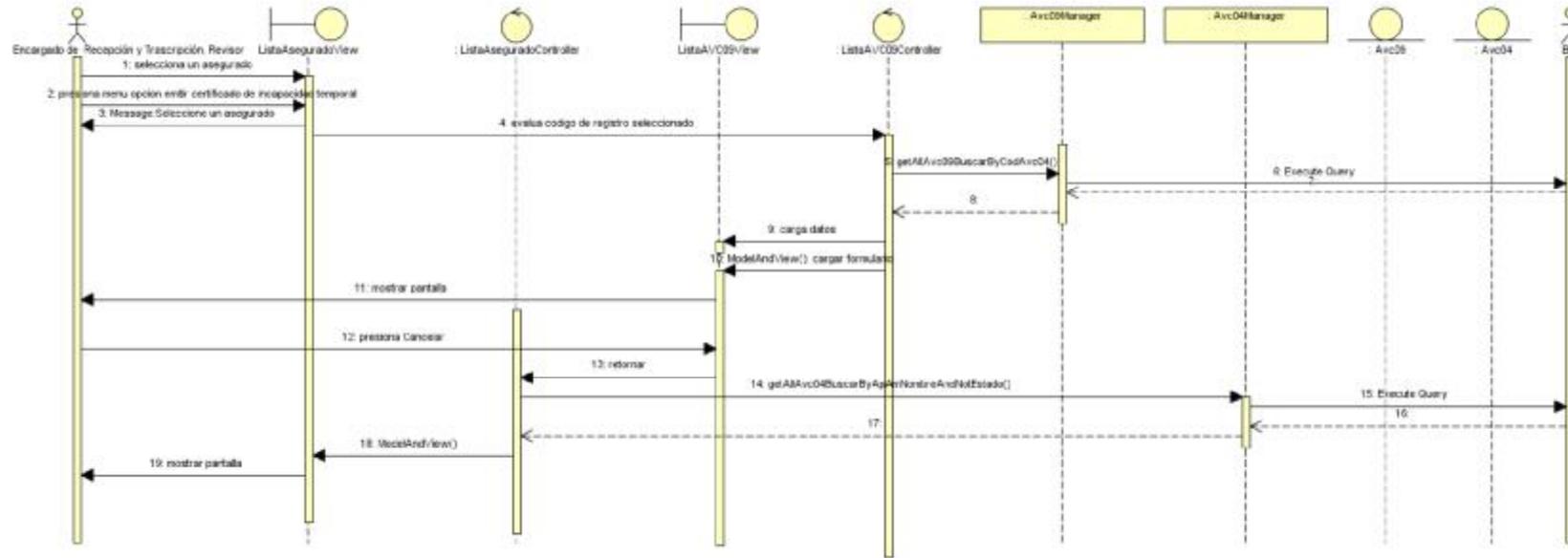


Figura 406. Diagrama de Secuencia: CU Gestionar Emitir Certificado de Incapacidad Temporal

2.1.2.2.3.9.2.1.114 Diagrama de Secuencia: Caso de Uso Adicionar Certificado de Incapacidad Temporal

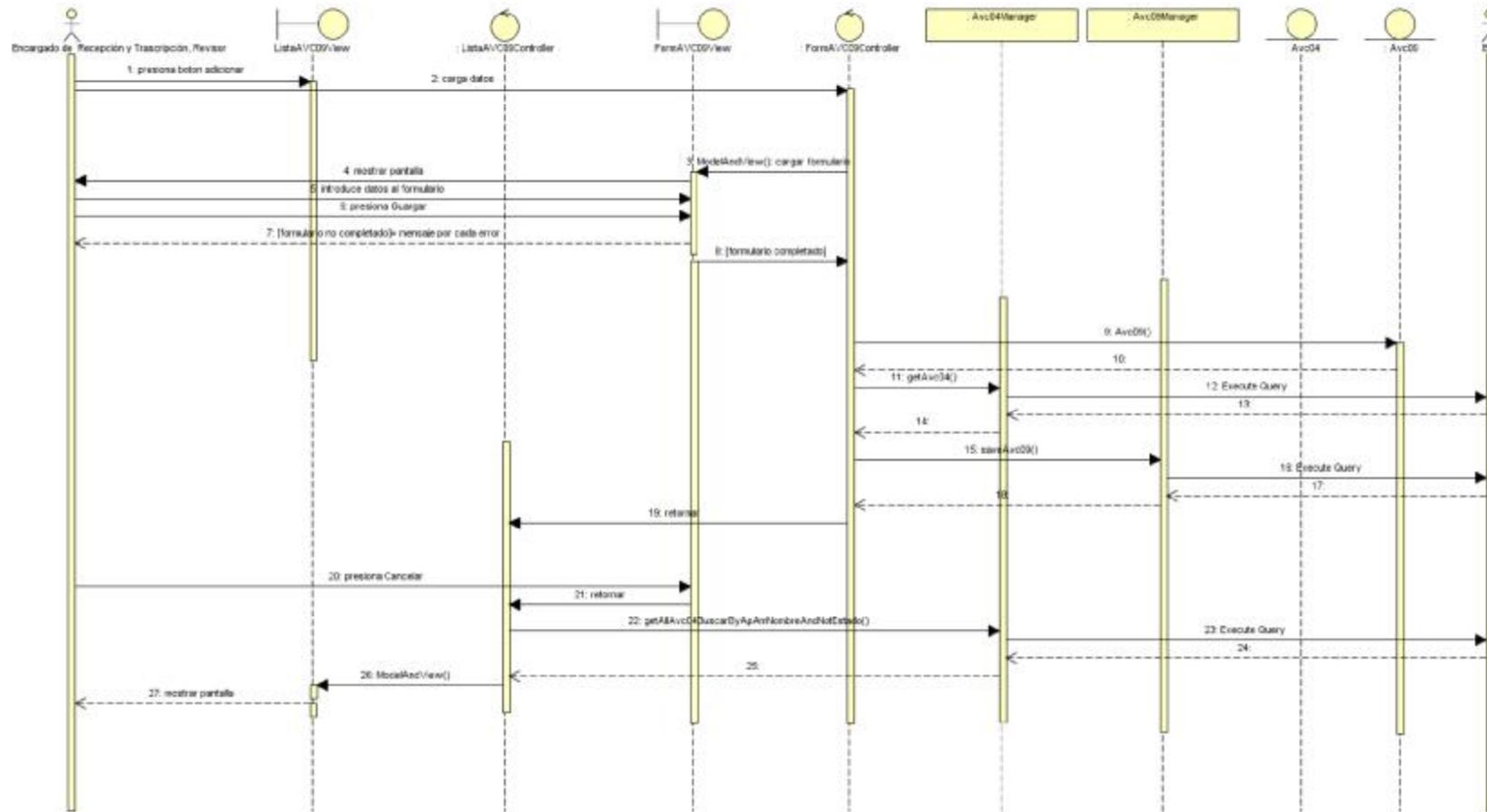


Figura 407. Diagrama de Secuencia: CU Adicionar Certificado de Incapacidad Temporal

2.1.2.2.3.9.2.1.115 Diagrama de Secuencia: Caso de Uso Modificar Certificado de Incapacidad Temporal

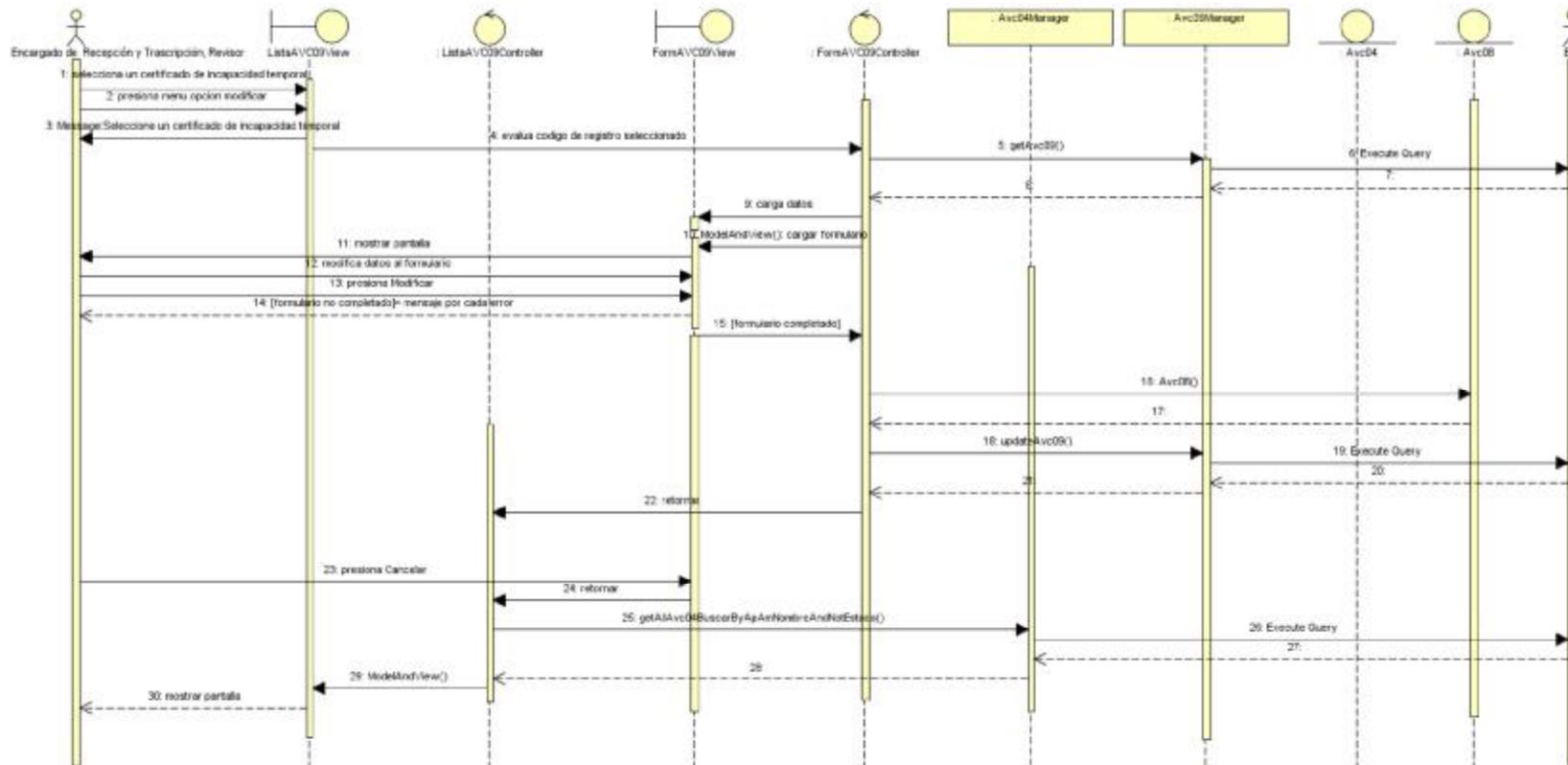


Figura 408. Diagrama de Secuencia: CU Modificar Certificado de Incapacidad Temporal

2.1.2.2.3.9.2.1.116 Diagrama de Secuencia: Caso de Uso Eliminar Certificado de Incapacidad Temporal

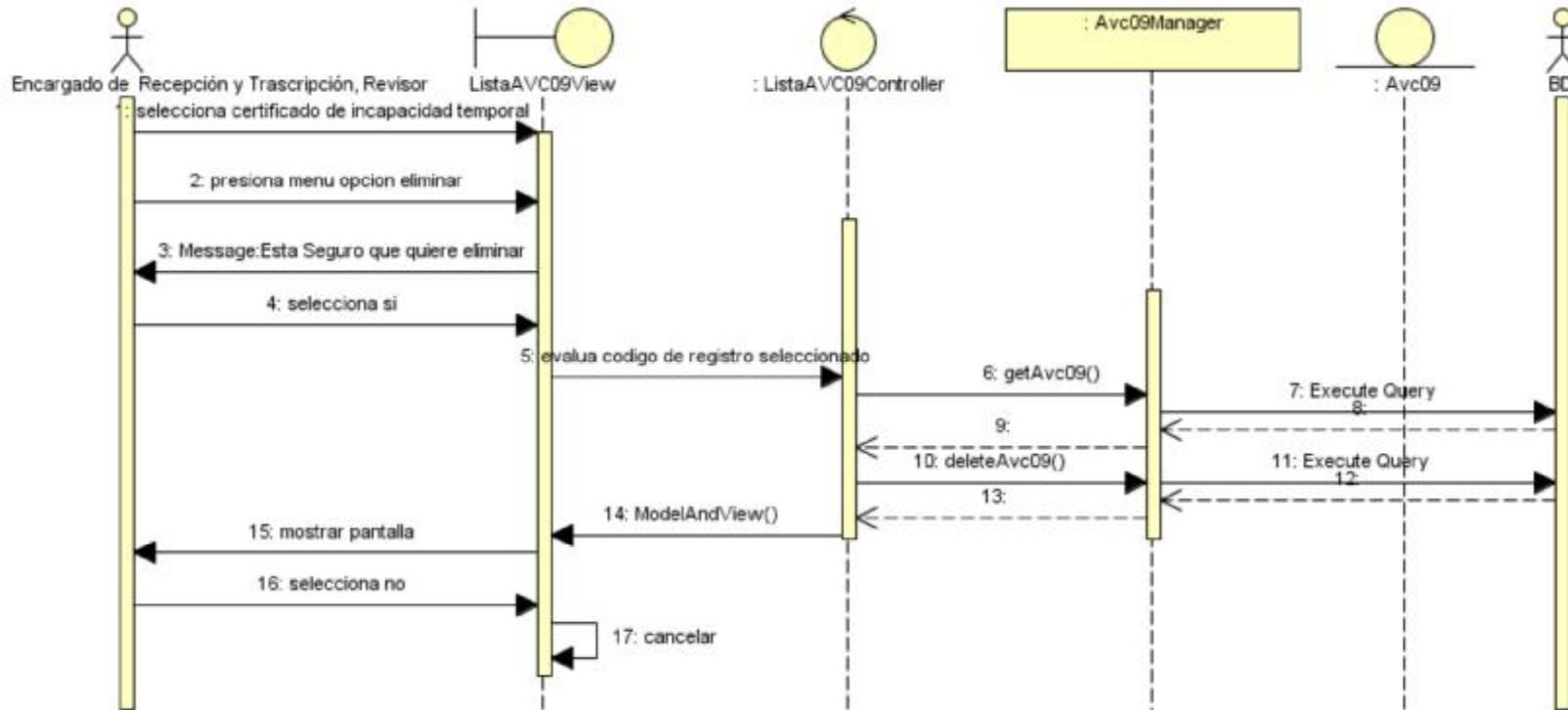


Figura 409. Diagrama de Secuencia: CU Eliminar Certificado de Incapacidad Temporal

2.1.2.2.3.9.2.1.117 Diagrama de Secuencia: Caso de Uso Cambiar Estado Certificado de Incapacidad Temporal

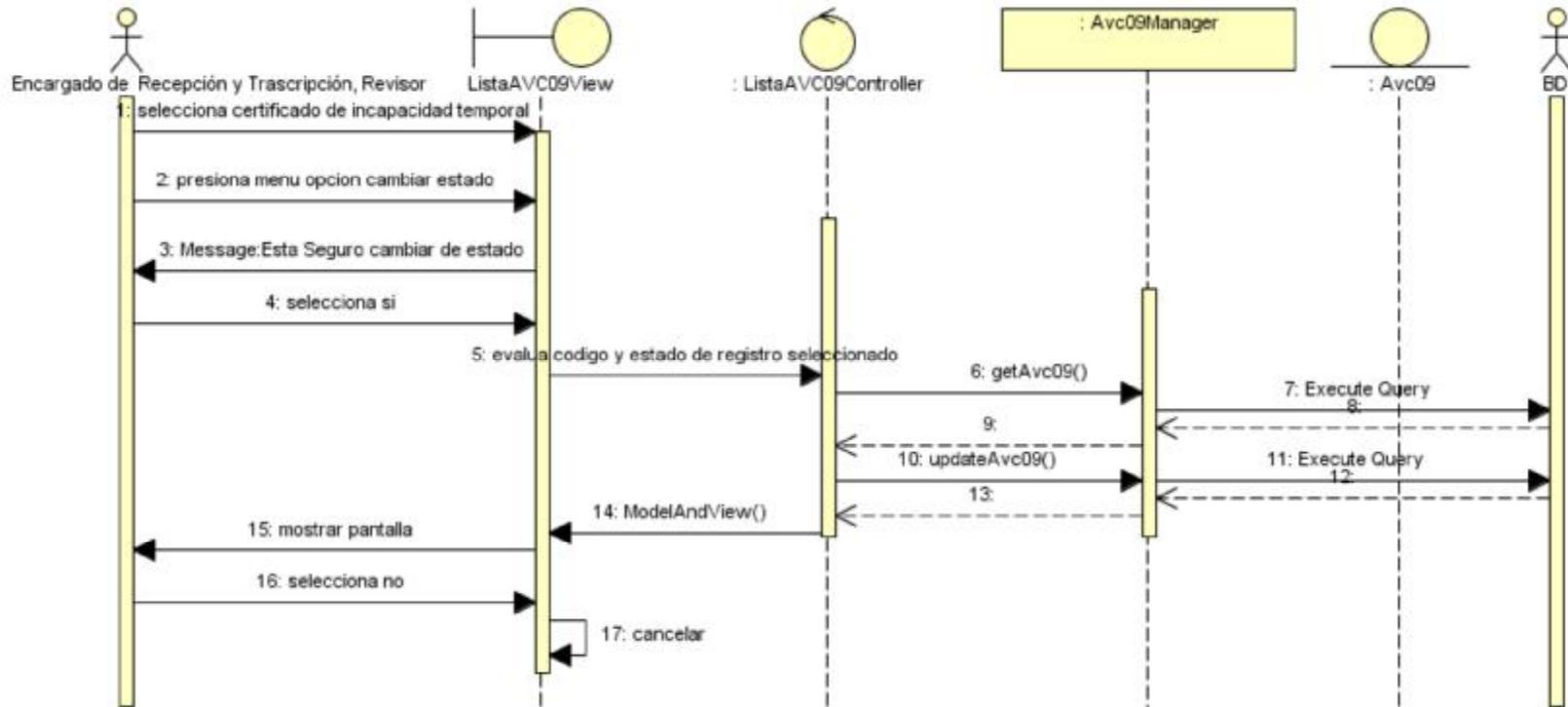


Figura 410. Diagrama de Secuencia: CU Cambiar Estado Certificado de Incapacidad Temporal

2.1.2.2.3.9.2.1.118

Diagrama de Secuencia: Caso de Uso Gestionar Altas y Bajas Beneficiarios

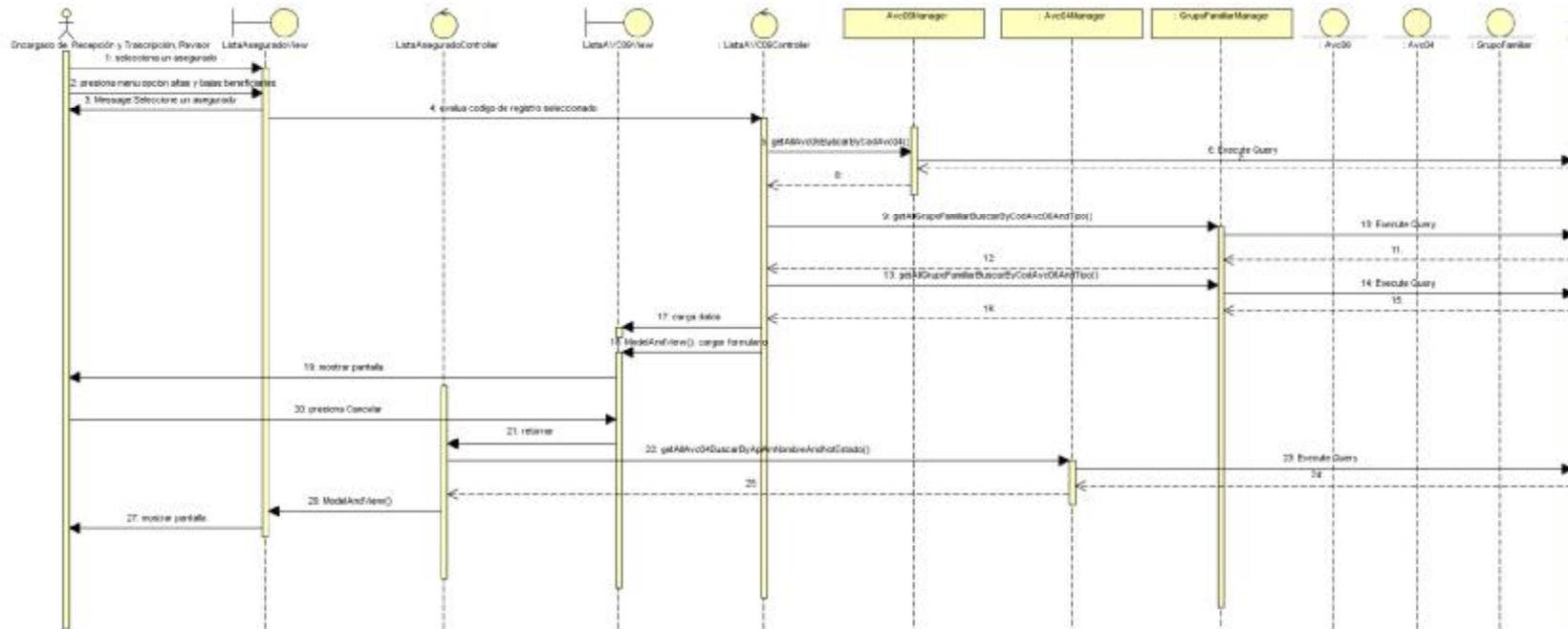


Figura 411. Diagrama de Secuencia: CU Gestionar Altas y Bajas Beneficiarios

2.1.2.2.3.9.2.1.120 Diagrama de Secuencia: Caso de Uso Eliminar Altas y Bajas Beneficiarios

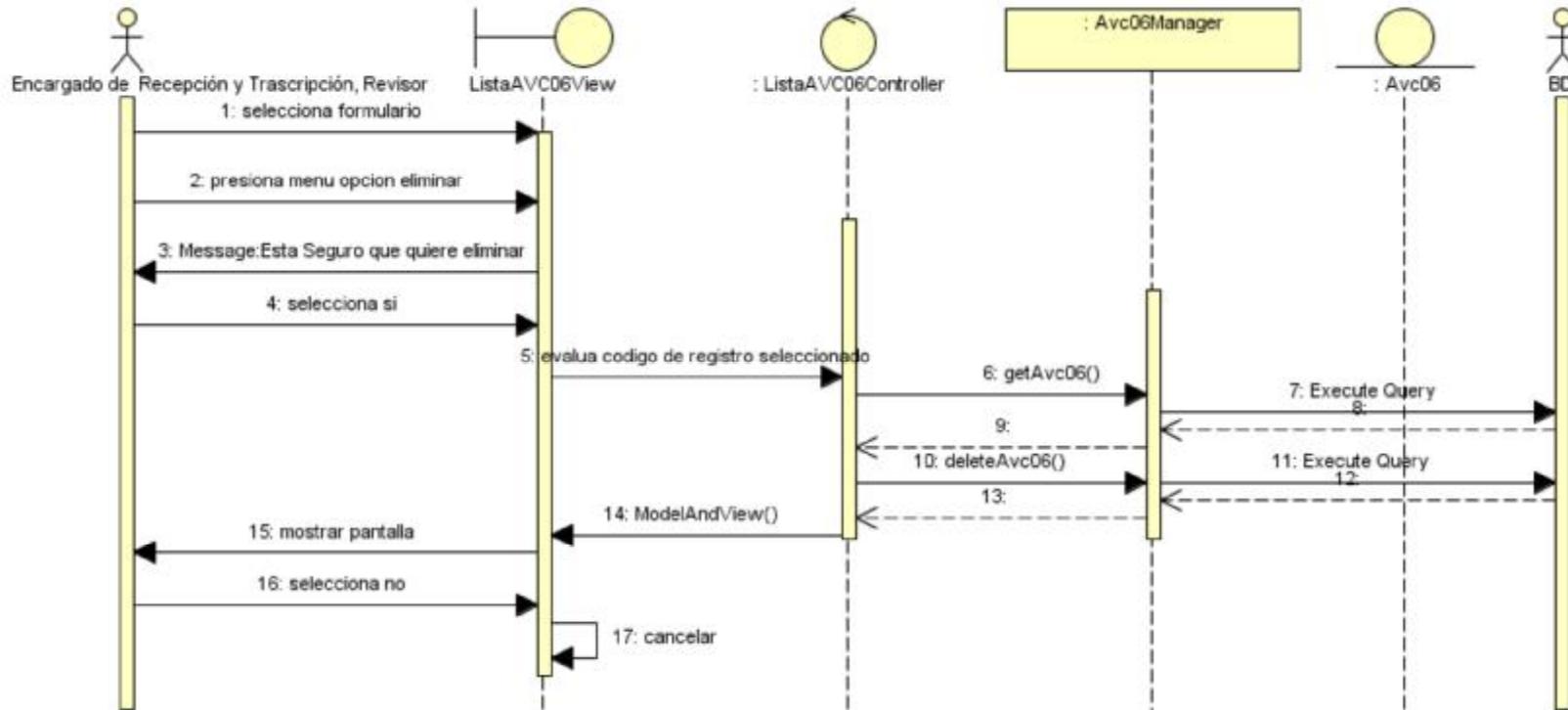


Figura 413. Diagrama de Secuencia: CU Eliminar Altas y Bajas Beneficiarios

2.1.2.2.3.9.2.1.121 Diagrama de Secuencia: Caso de Uso Cambiar Estado Altas y Bajas Beneficiarios

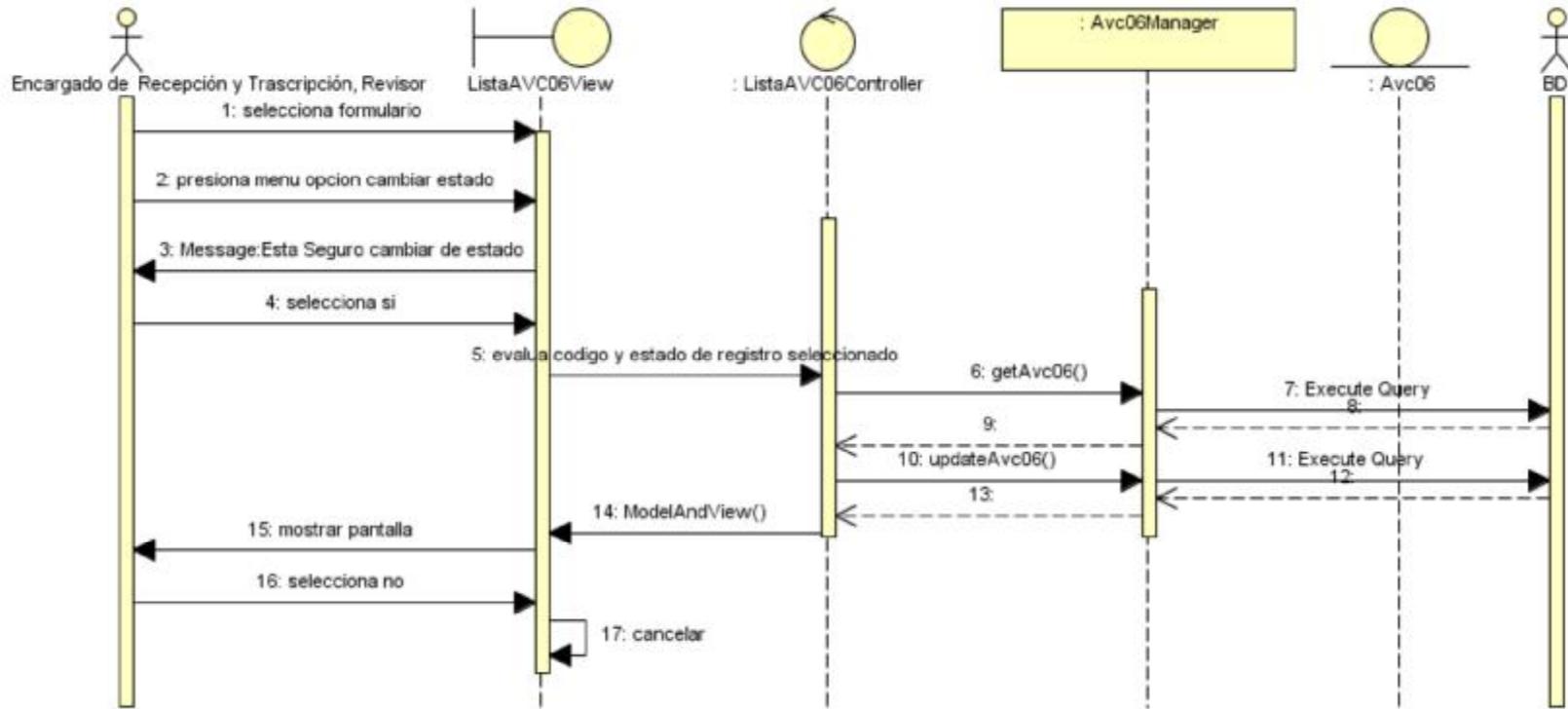


Figura 414. Diagrama de Secuencia: CU Cambiar Estado Altas y Bajas Beneficiarios

2.1.2.2.3.9.2.1.122

Diagrama de Secuencia: Caso de Uso Gestionar Beneficiarios Afiliados

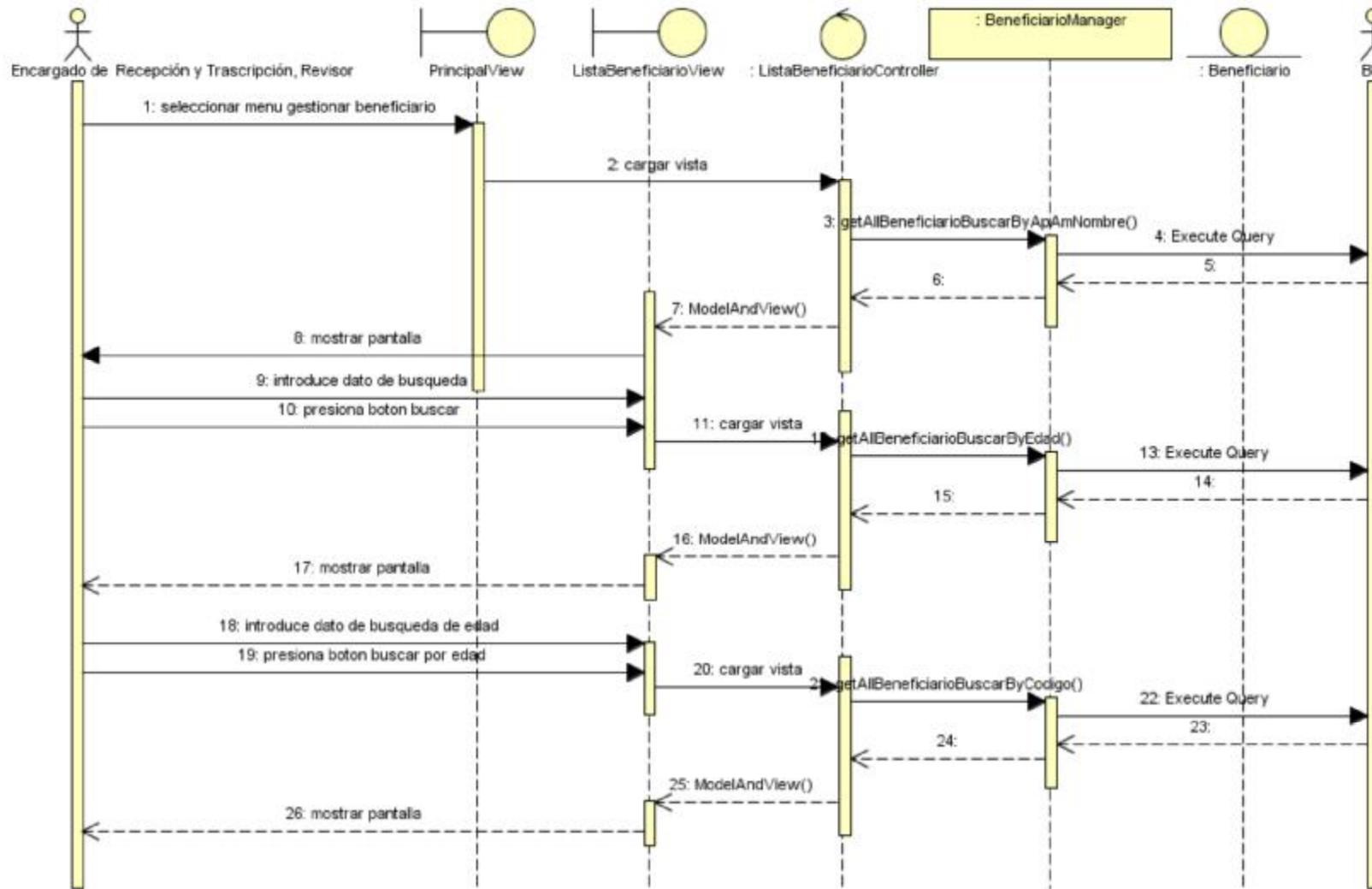


Figura 415. Diagrama de Secuencia: CU Gestionar Beneficiarios Afiliados

2.1.2.2.3.9.2.1.123 Diagrama de Secuencia: Caso de Uso Dar de Alta Beneficiario por Norma

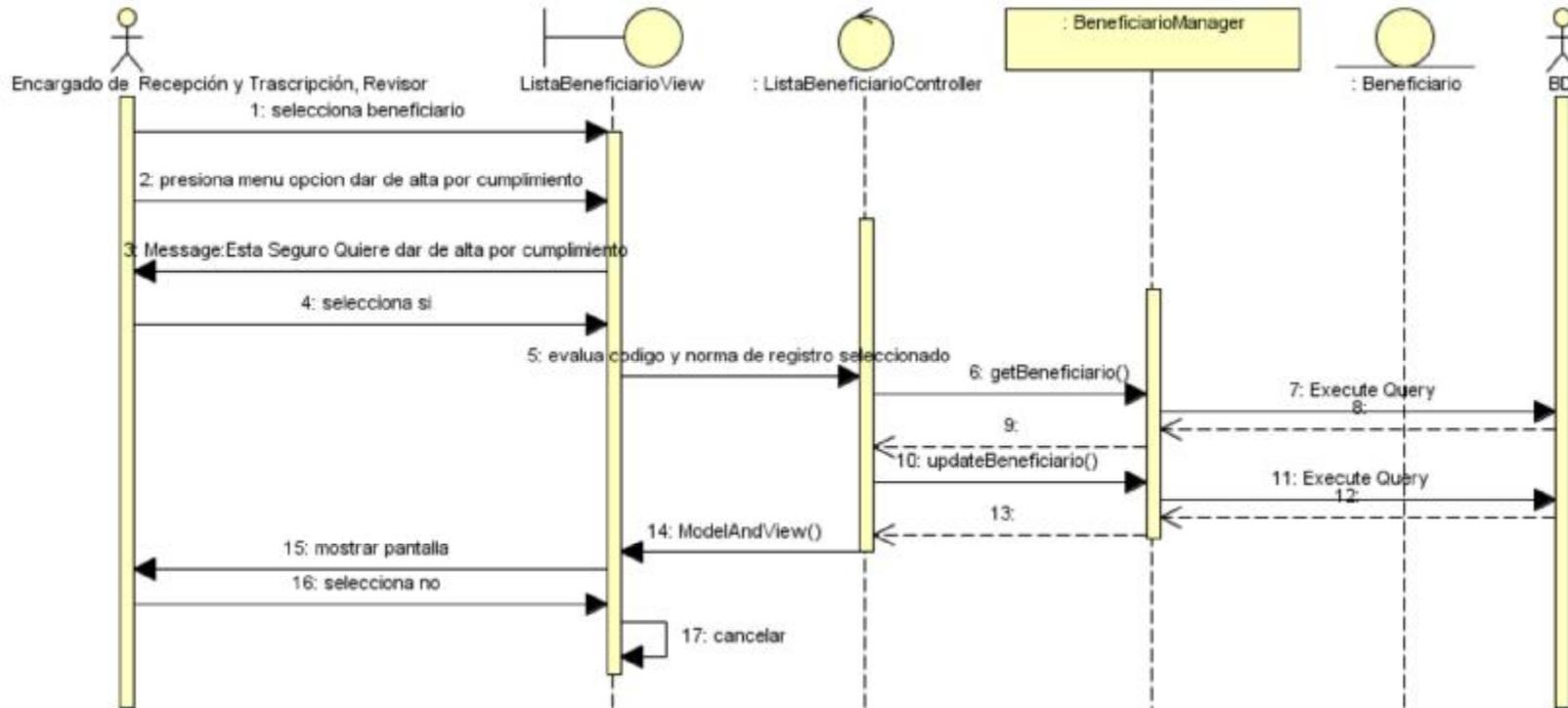


Figura 416. Diagrama de Secuencia: CU Dar de Alta Beneficiario por Norma

2.1.2.2.3.9.2.1.124 Diagrama de Secuencia: Caso de Uso Dar de Baja Beneficiario por Norma

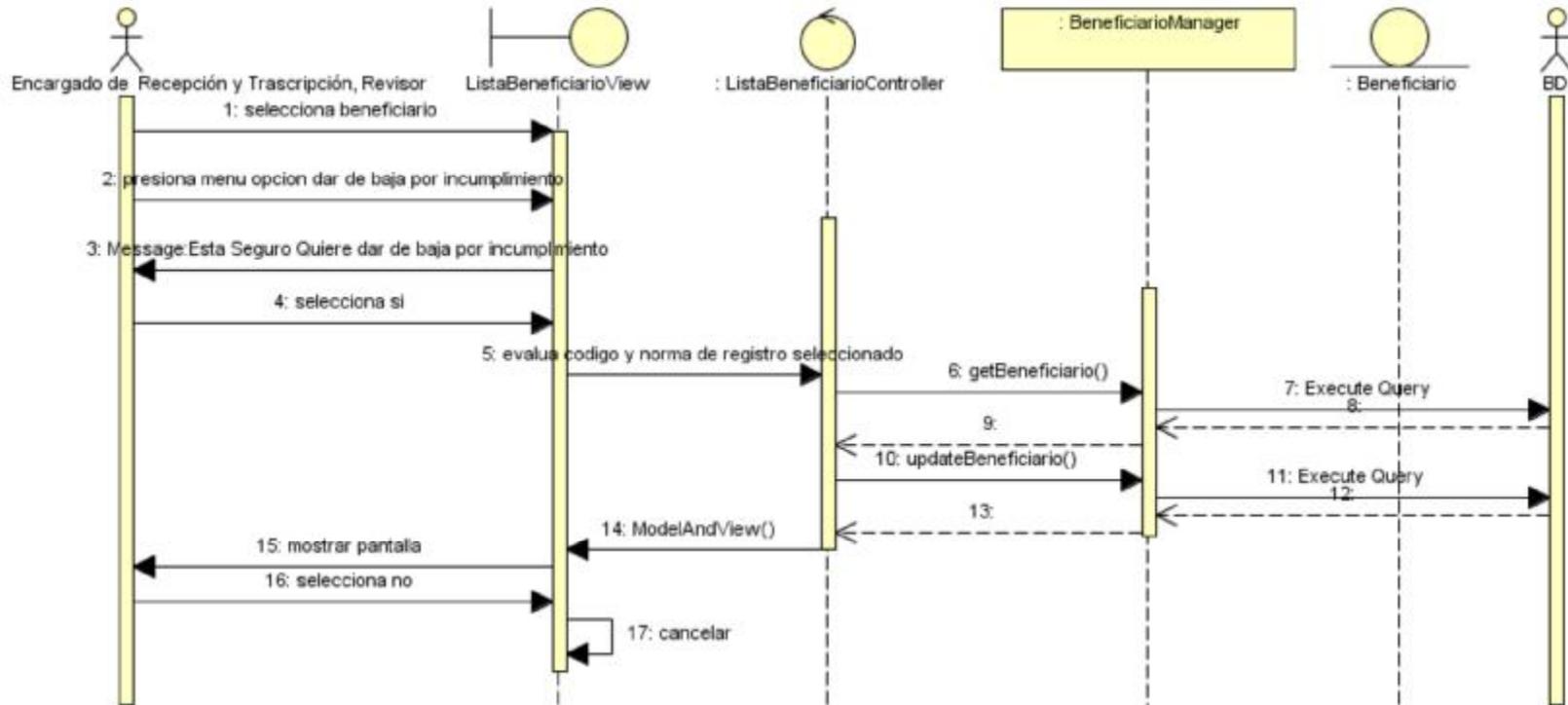


Figura 417. Diagrama de Secuencia: CU Dar de Baja Beneficiario por Norma

2.1.2.2.3.9.2.1.125 Diagrama de Secuencia: Caso de Uso Gestionar Reporte Asegurado por Tipo de Seguro

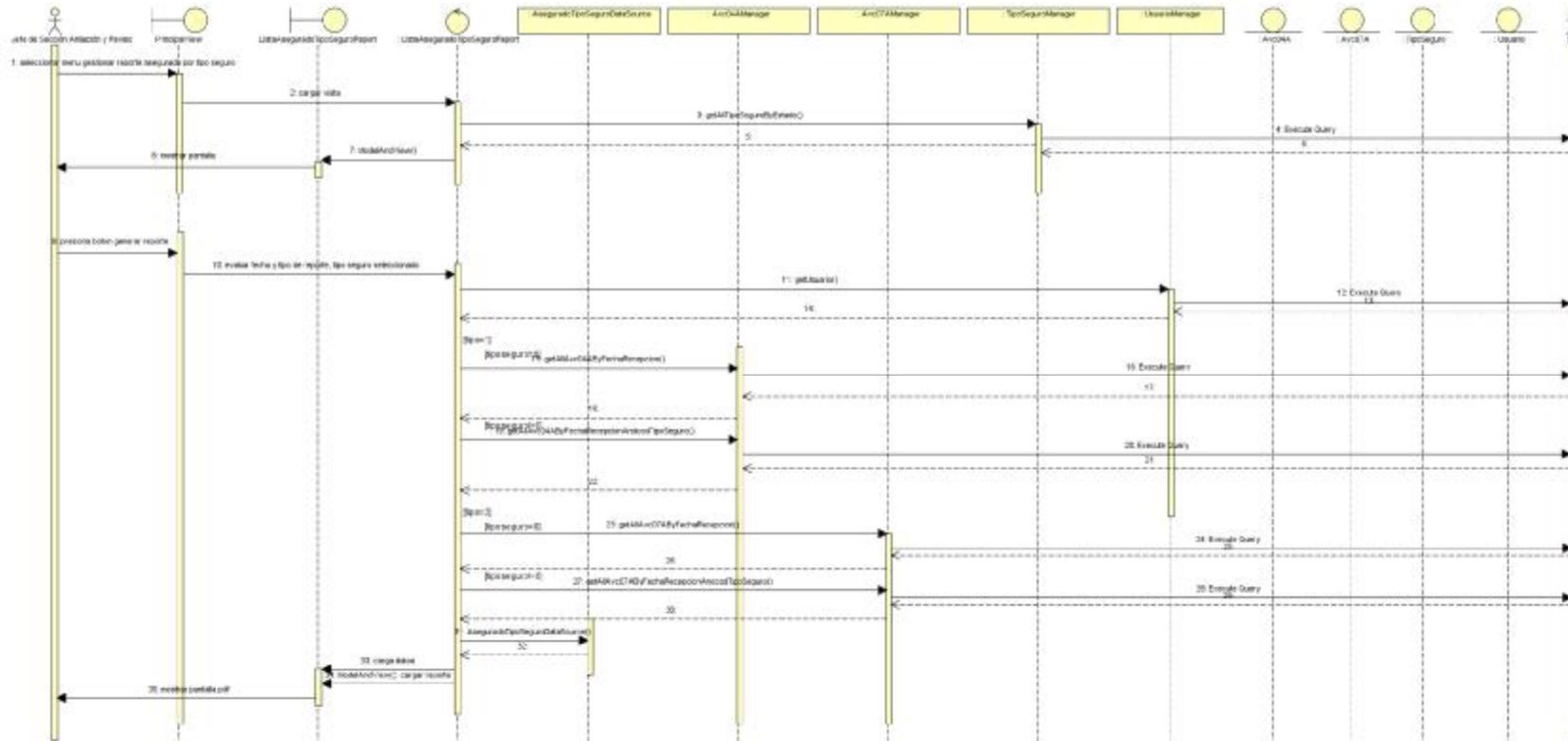


Figura 418. Diagrama de Secuencia: CU Gestionar Reporte Asegurado por Tipo de Seguro

2.1.2.2.3.9.2.1.126

Diagrama de Secuencia: Caso de Uso Gestionar Reporte Empresa

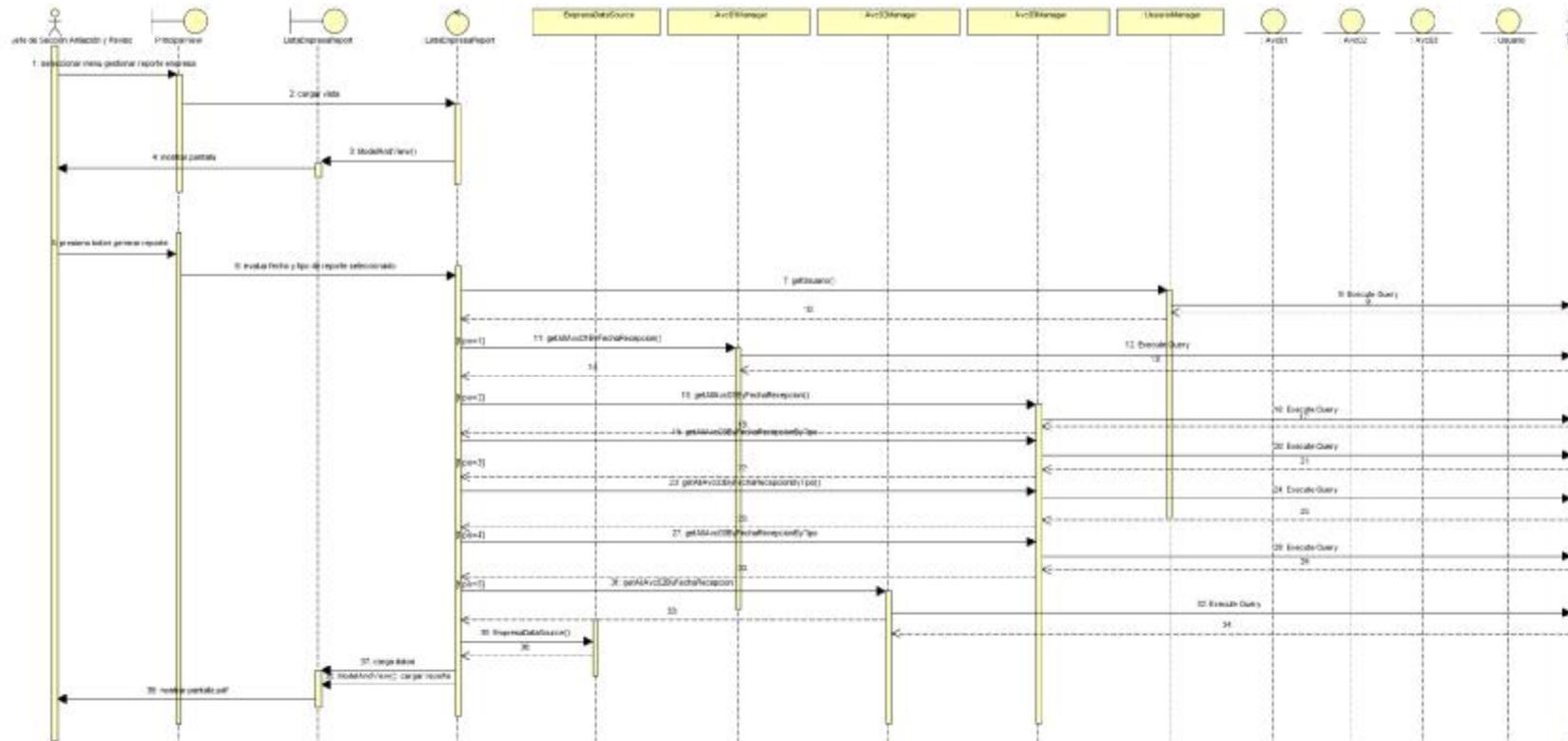


Figura 419. Diagrama de Secuencia: CU Gestionar Reporte Empresa

2.1.2.2.3.9.2.1.127

Diagrama de Secuencia: Caso de Uso Gestionar Reporte Asegurado

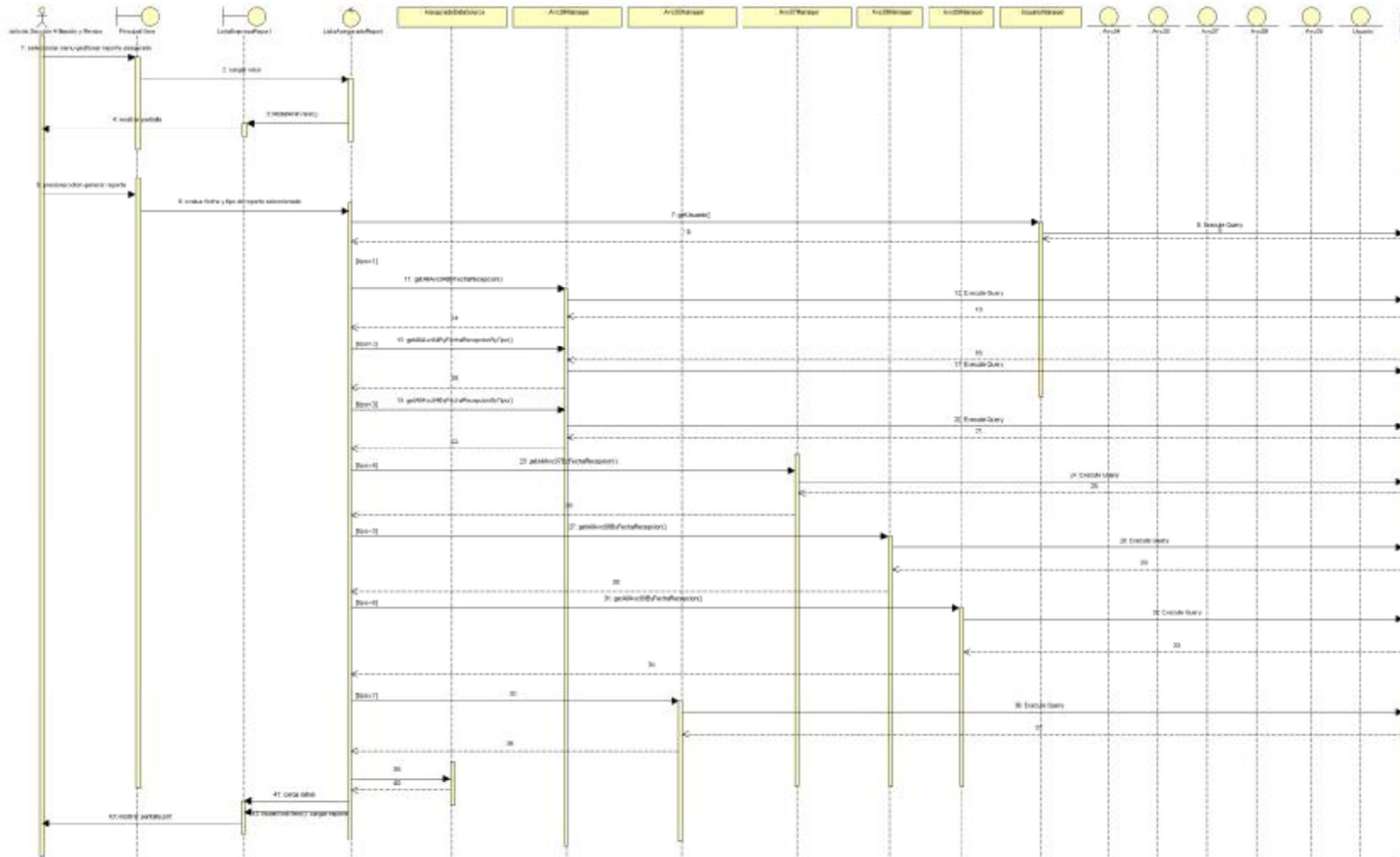


Figura 420. Diagrama de Secuencia: CU Gestionar Reporte Asegurado

2.1.2.2.3.9.2.1.128 Diagrama de Secuencia: Caso de Uso Gestionar Reporte Beneficiario

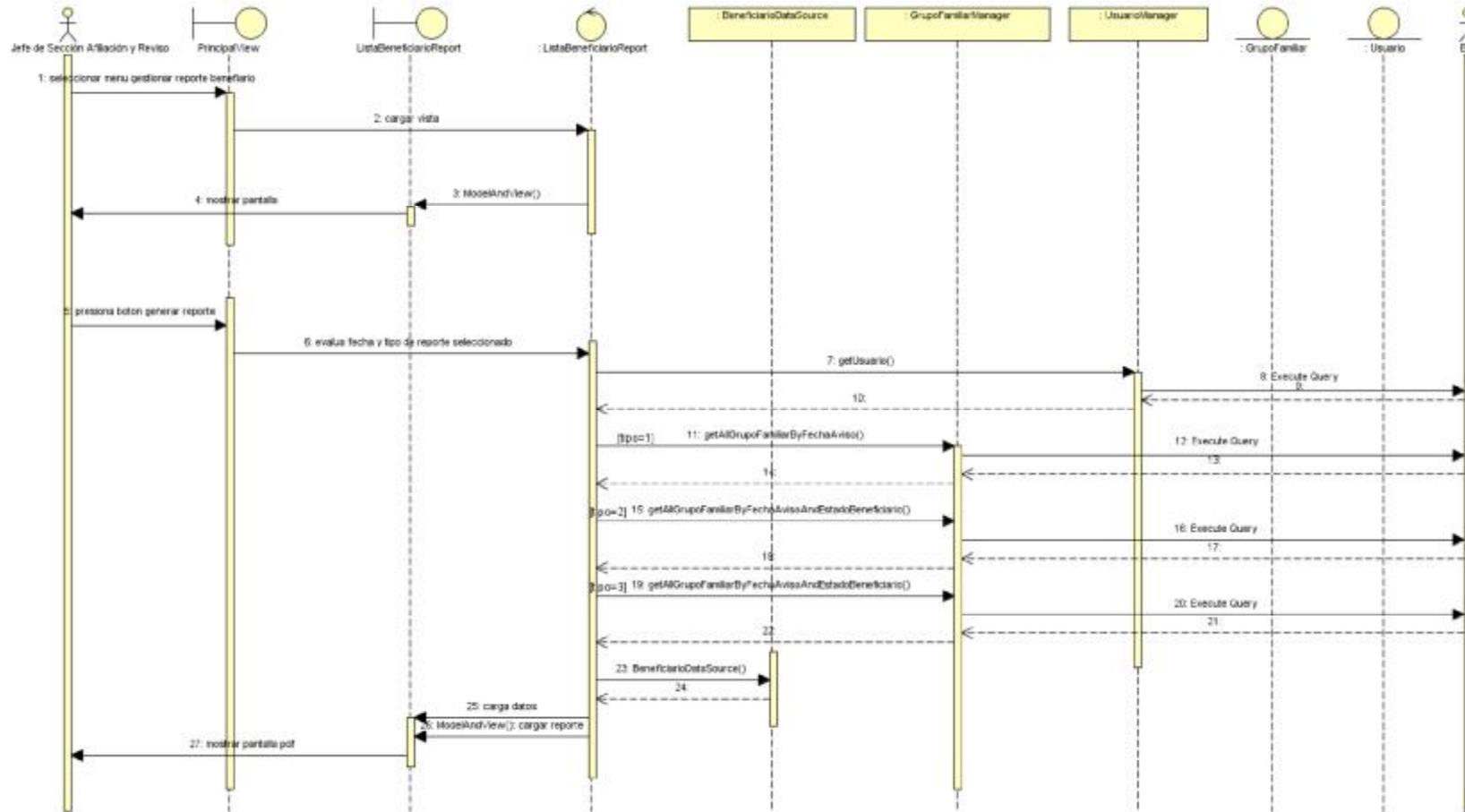


Figura 421. Diagrama de Secuencia: CU Gestionar Reporte Beneficiario

2.1.2.2.3.9.2.1.129

Diagrama de Secuencia: Caso de Uso Reporte Ficha Asegurado

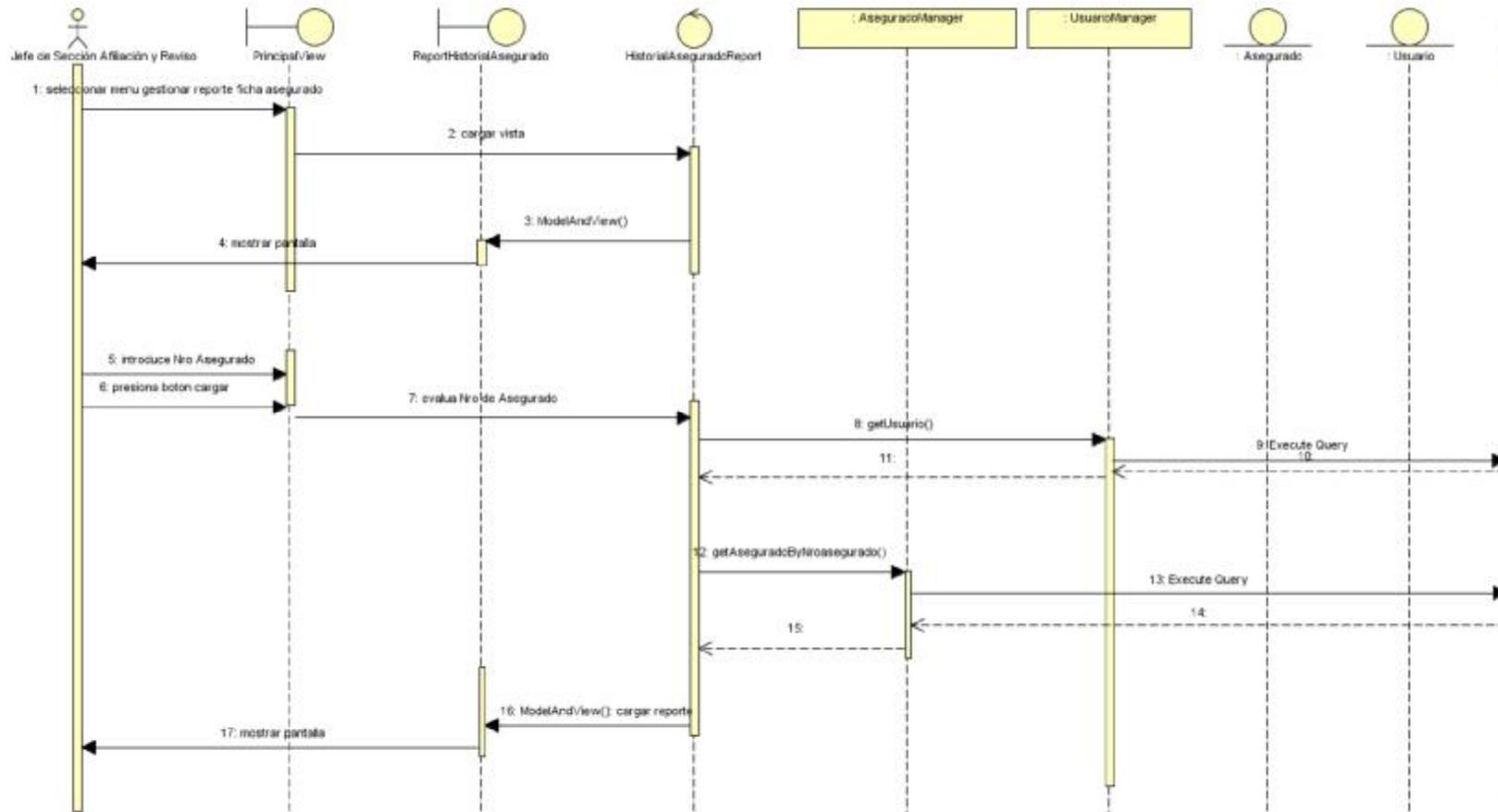


Figura 422. Diagrama de Secuencia: CU Reporte Ficha Asegurado

2.1.2.2.3.9.2.1.130

Diagrama de Secuencia: Caso de Uso Reporte Ficha Empresa

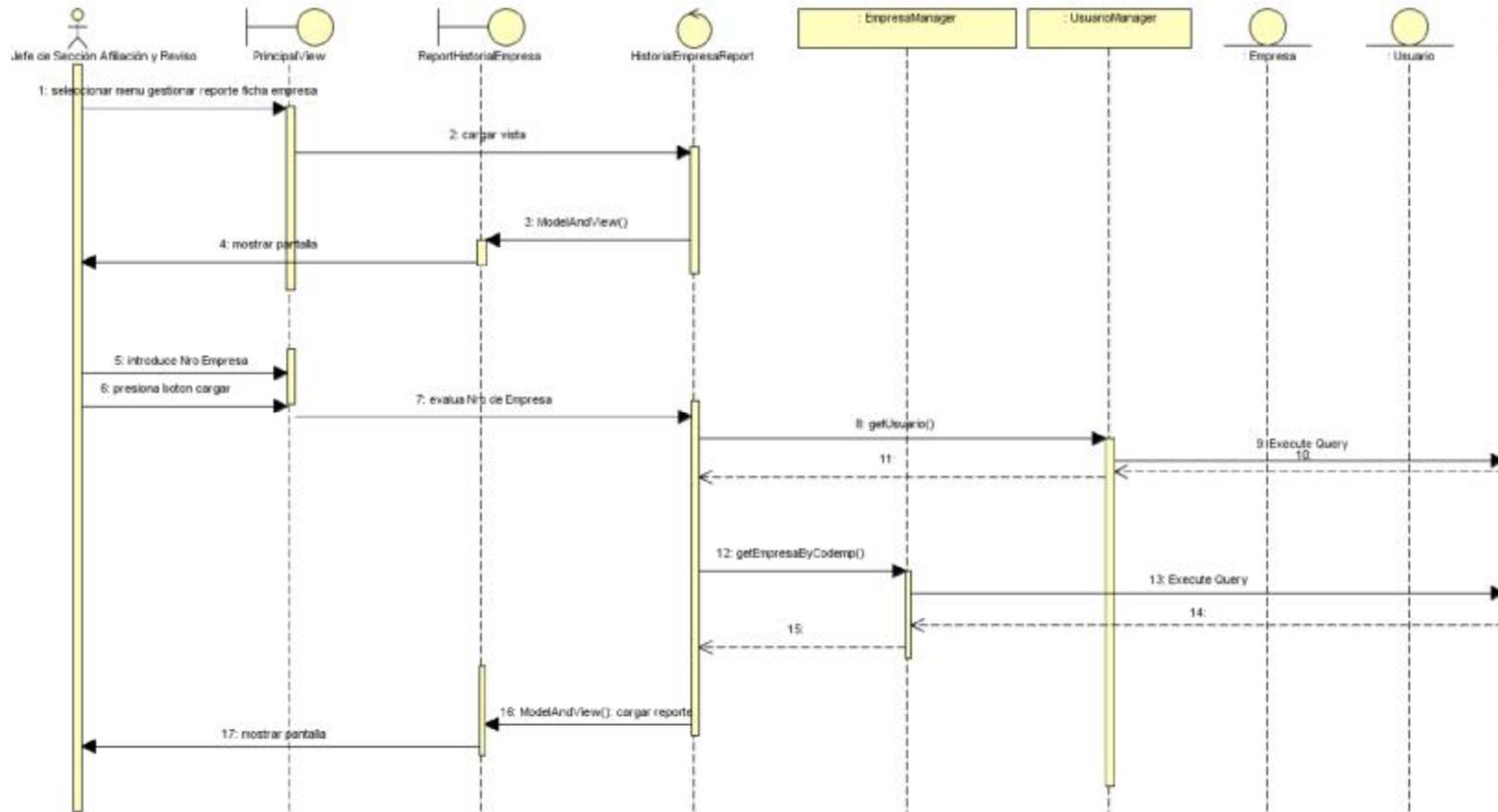


Figura 423. Diagrama de Secuencia: CU Reporte Ficha Empresa

2.1.2.2.3.9.2.1.131 Diagrama de Secuencia: Caso de Uso Reporte Ficha Asegurado por Tipo Seguro

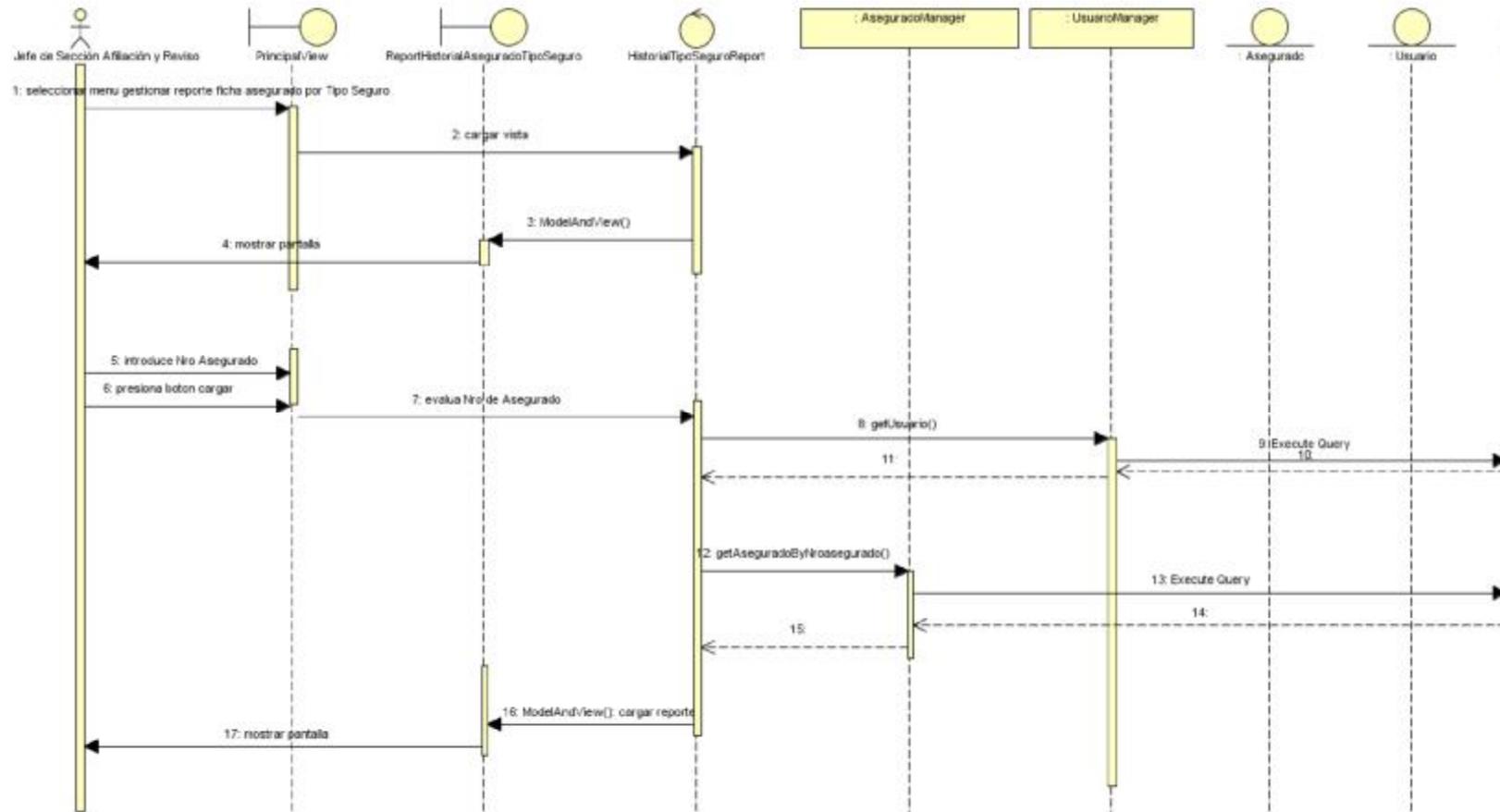


Figura 424. Diagrama de Secuencia: CU Reporte Ficha Asegurado por Tipo Seguro

2.1.2.2.3.9.2.1.132

Diagrama de Secuencia: Caso de Uso Reporte Estadístico

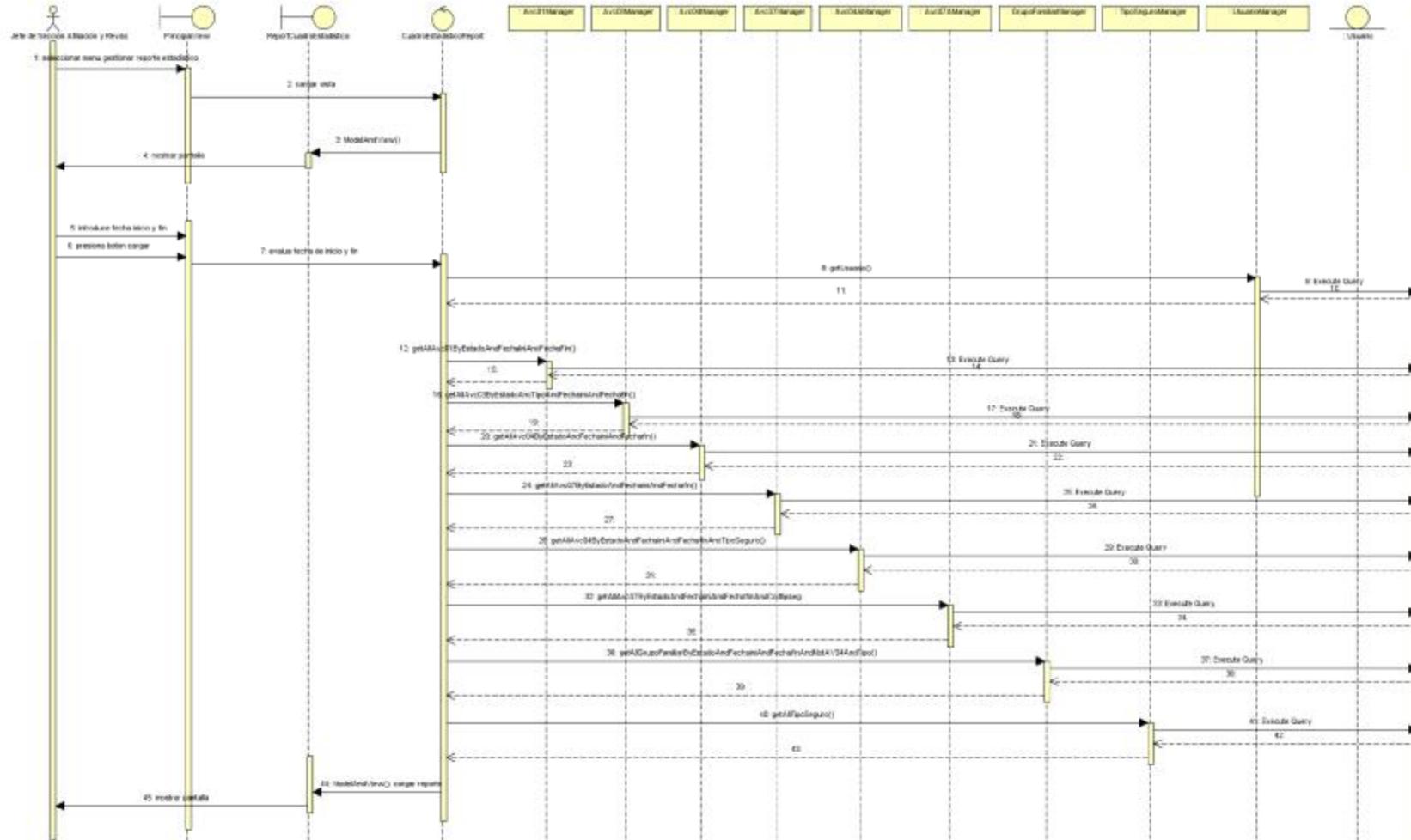


Figura 425. Diagrama de Secuencia: CU Reporte Estadístico

2.1.2.2.3.9.2.1.133 Diagrama de Secuencia: Caso de Uso Administrar Backup

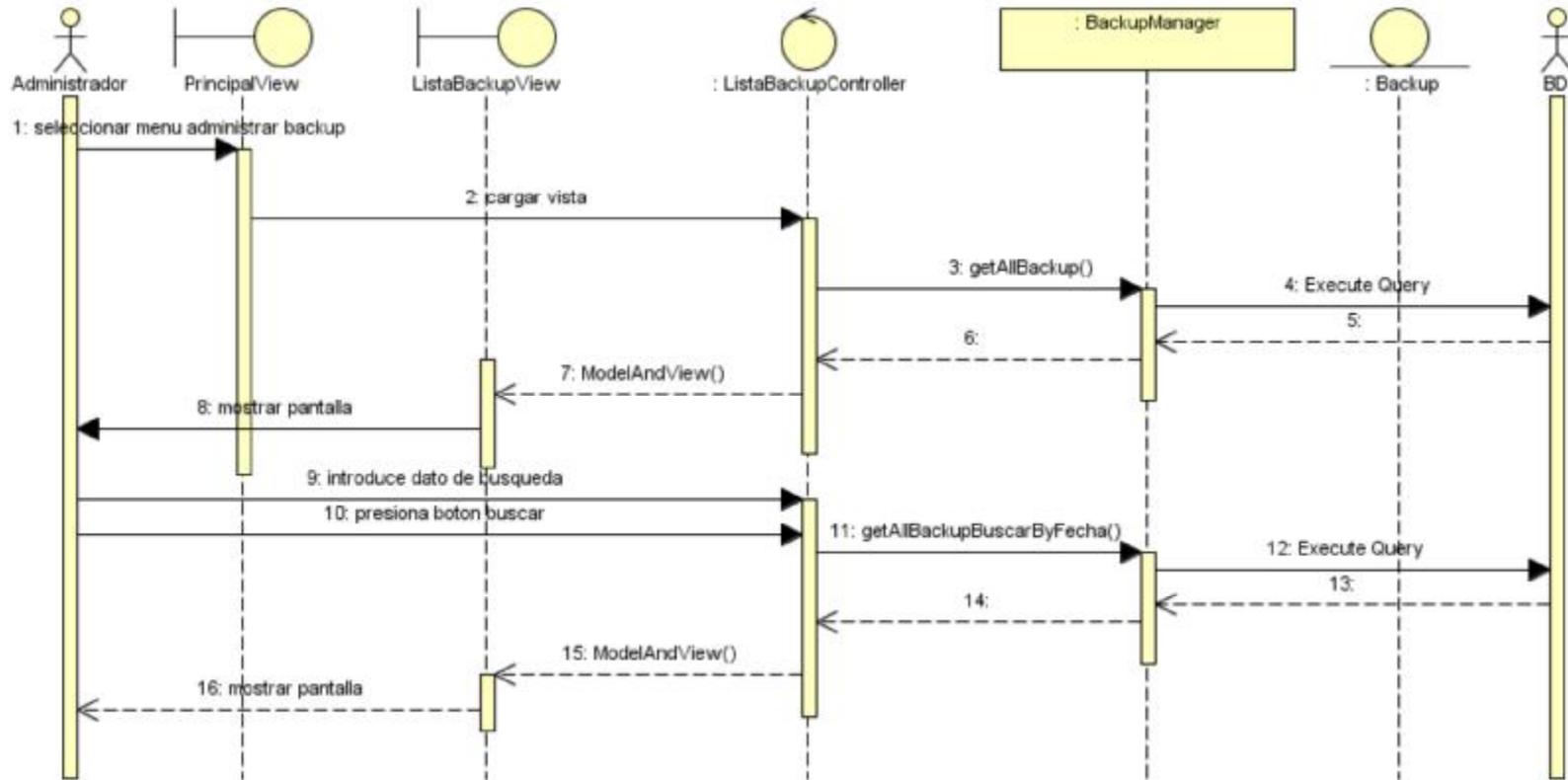


Figura 426. Diagrama de Secuencia: CU Administrar Backup

2.1.2.2.3.9.2.1.134 Diagrama de Secuencia: Caso de Uso Resguardar Backup

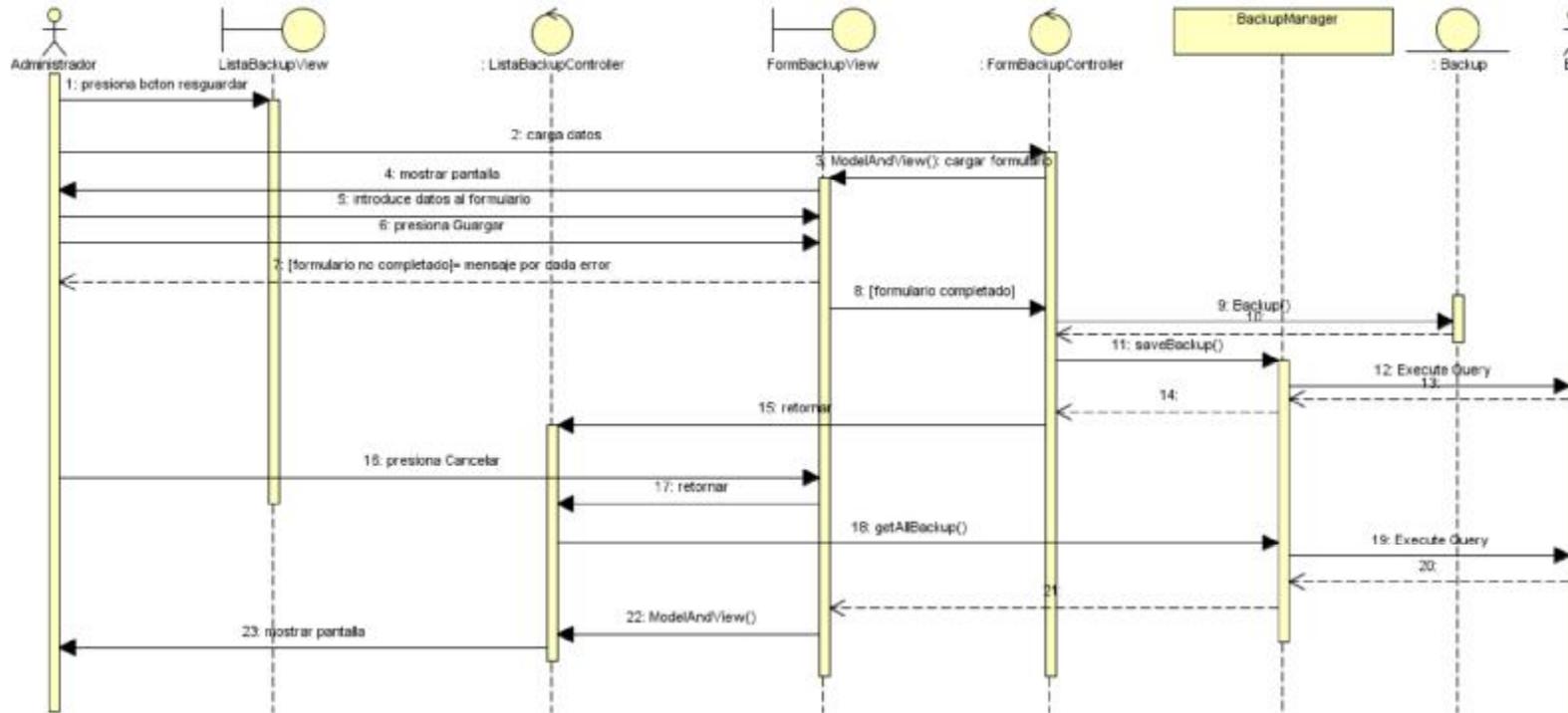


Figura 427. Diagrama de Secuencia: CU Resguardar Backup

2.1.2.2.3.9.2.1.135 Diagrama de Secuencia: Caso de Uso Restaurar Backup

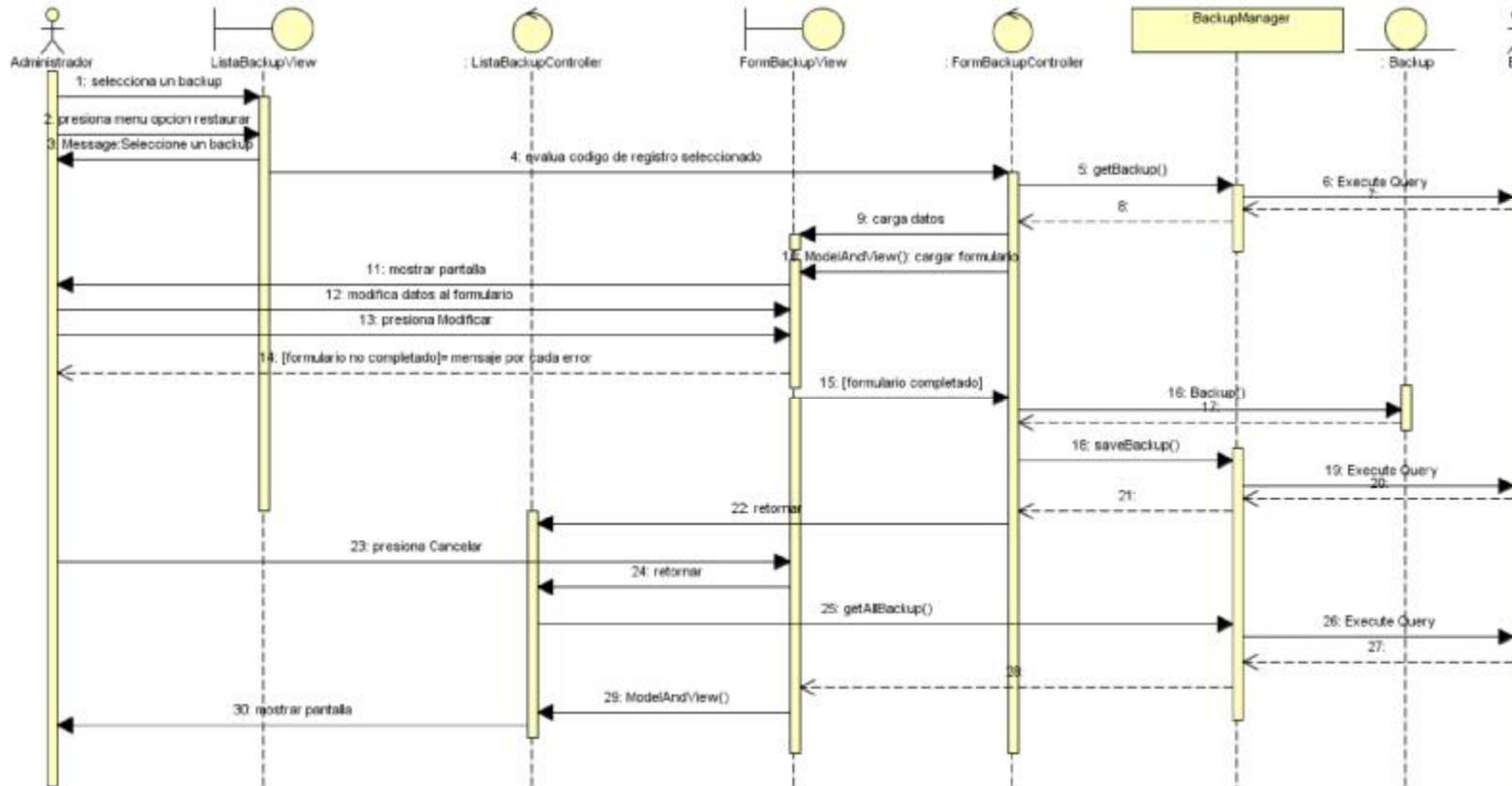


Figura 428. Diagrama de Secuencia: CU Restaurar Backup

2.1.2.2.3.9.2.1.136 Diagrama de Secuencia: Caso de Uso Eliminar Backup

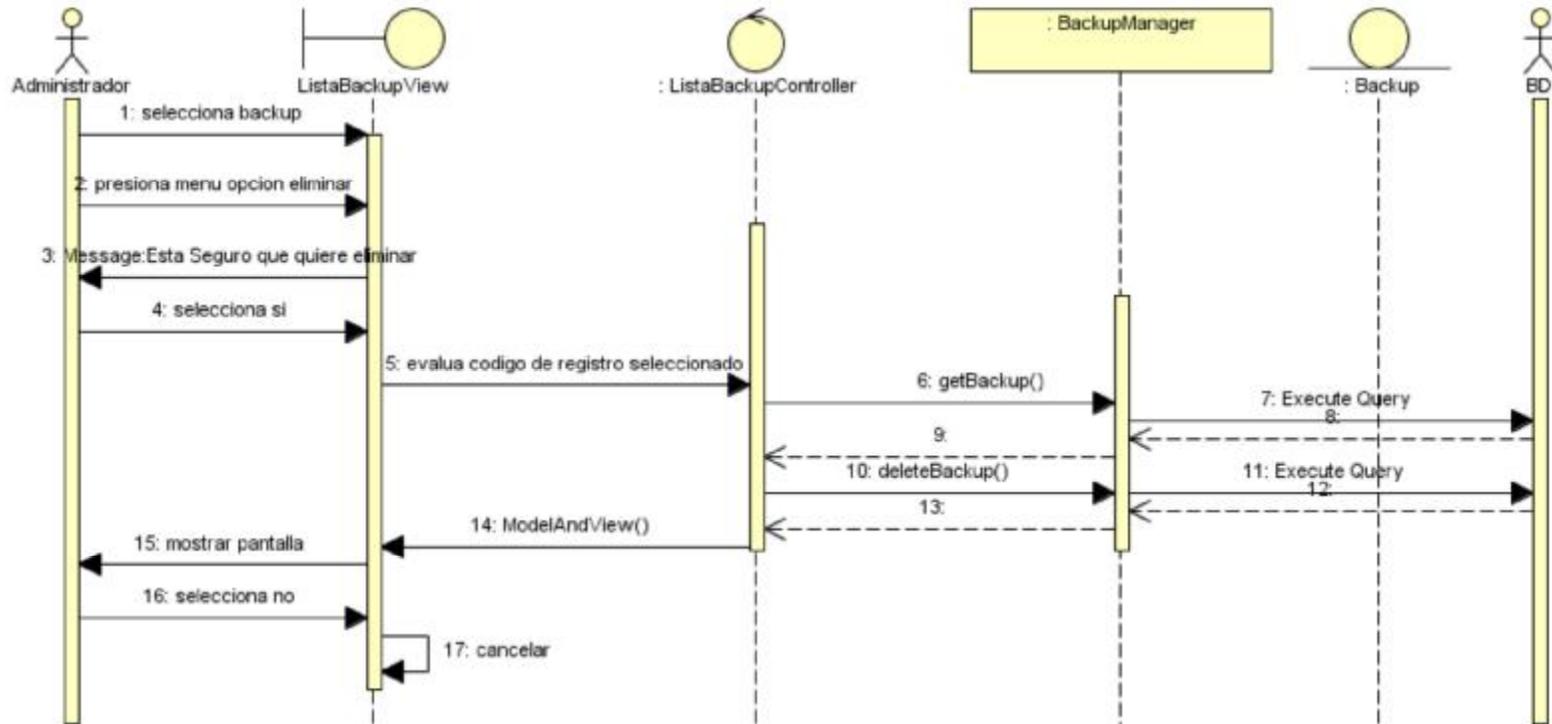


Figura 429. Diagrama de Secuencia: CU Eliminar Backup

2.1.2.2.3.9.2.1.137 Diagrama de Secuencia: Caso de Uso Generar Reporte Lista Backup

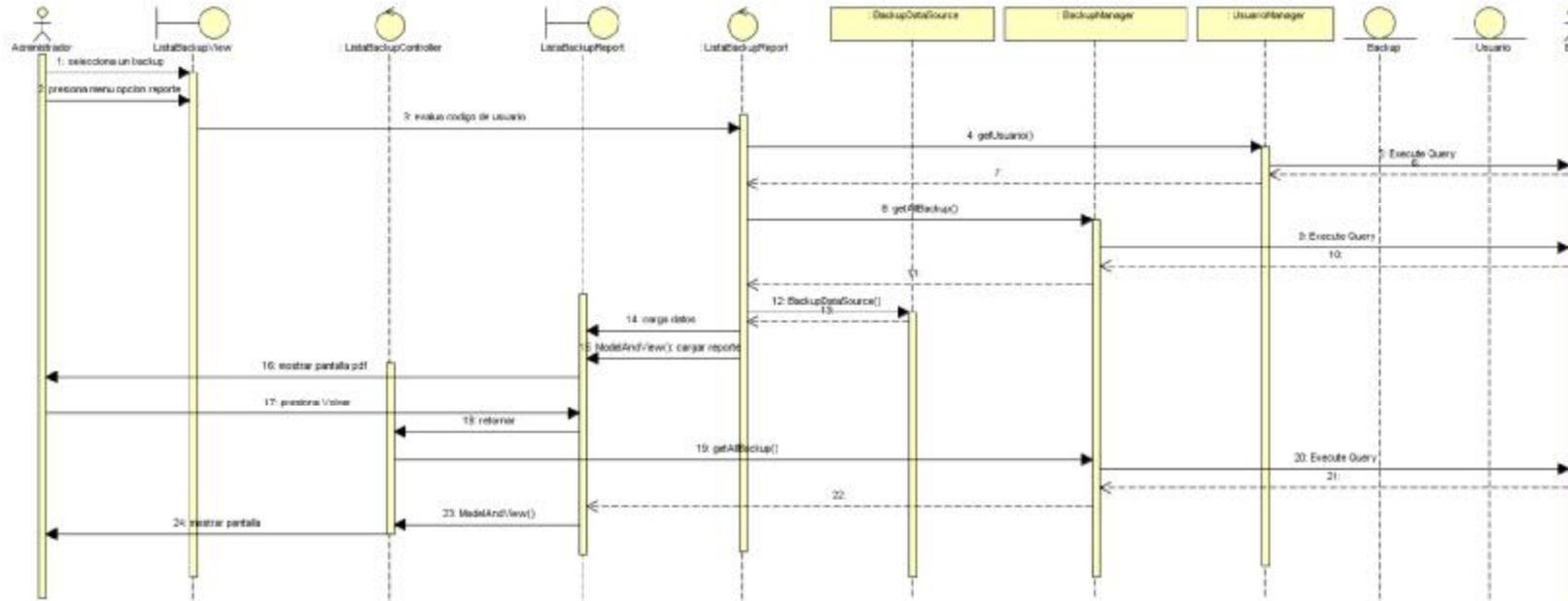


Figura 430. Diagrama de Secuencia: CU Generar Reporte Lista Backup

2.1.2.2.3.10 Modelo de Datos

Previendo que la persistencia de la información del Sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

2.1.2.2.3.10.1 Modelado de Diagrama de Clases

2.1.2.2.3.10.1.1 Introducción

El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un Diagrama de Clases representa las clases del Sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. El modelo de Casos de Uso aporta información para establecer las clases, objetos, atributos y operaciones.

2.1.2.2.3.10.1.2 Mecanismos de Abstracción

- Clasificación / Instanciación
- Composición / Descomposición
- Agrupación / Individualización
- Especialización / Generalización

La clasificación es uno de los mecanismos de abstracción más utilizados. La clase define el ámbito de definición de un conjunto de objetos, y cada objeto pertenece a una clase. Los objetos se crean por instanciación de las clases.

Cada clase se representa en un rectángulo con tres compartimientos:

- Nombre de la clase
- Atributos de la clase
- Operaciones de la clase

Los atributos de una clase no deberían ser manipulables directamente por el resto de objetos. Por esta razón se crearon niveles de visibilidad para los elementos que son:

- (-) Privado: es el más fuerte. Esta parte es totalmente invisible.
- (#) Los atributos u operaciones protegidas están visibles para las clases de herencia.
- (+) Los atributos u operaciones públicas son visibles desde otras clases y también por clases de herencia.

2.1.2.2.3.10.1.3 Diagrama de Clases

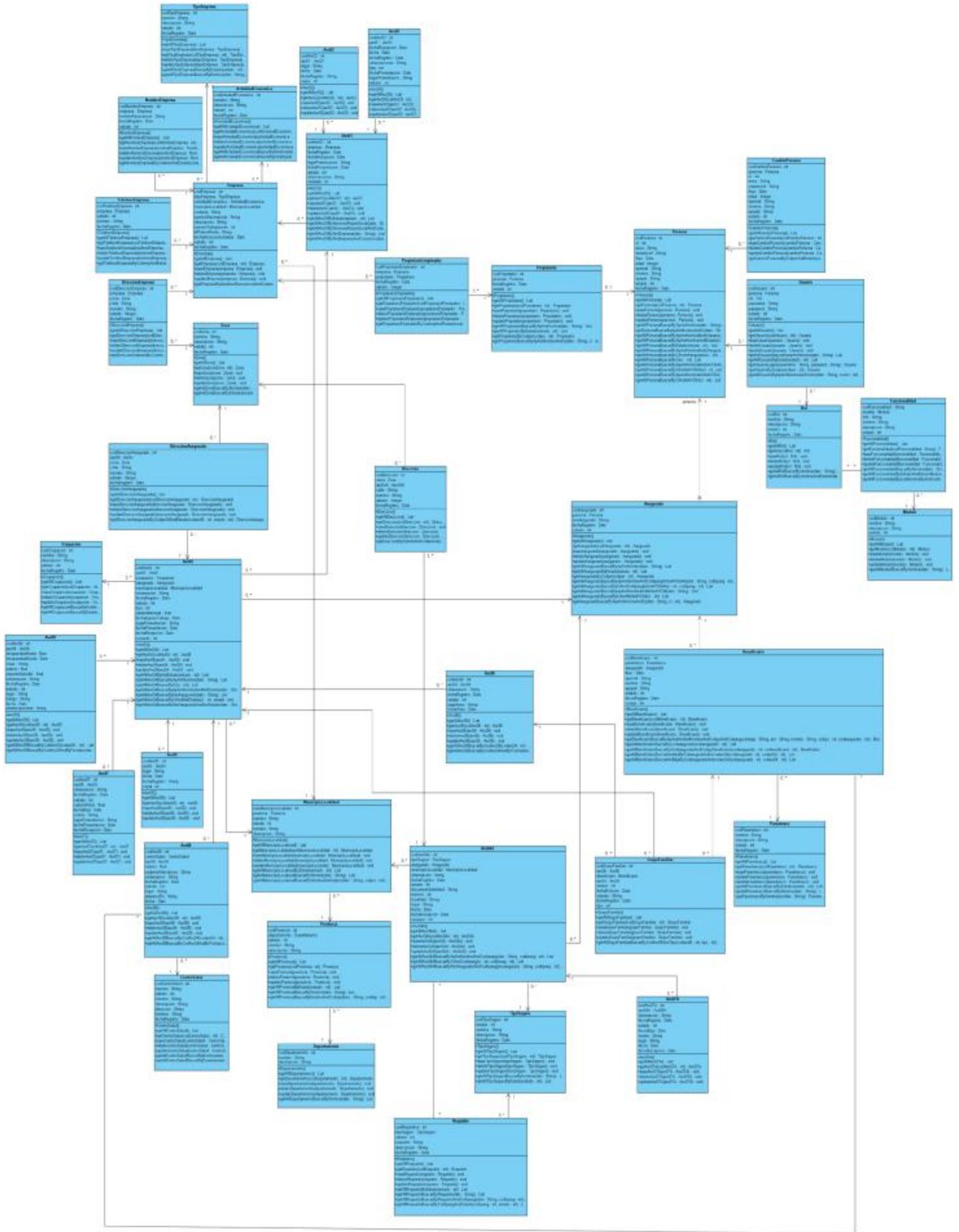


Figura 431. Diagrama de clases

2.1.2.2.3.10.1.4 Especificación de Métodos

A continuación una especificación de los métodos de los objetos utilizados en el proyecto, se mostrara el pseudo-código preliminar de cada método.

2.1.2.2.3.10.1.4.1 Objeto ActividadEconomica

OBJETO:	ActividadEconomica
MÉTODO:	getAllActividadEconomica
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(ActividadEconomica.class);	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
MÉTODO:	getActividadEconomica
ARGUMENTO:	(int codActividadEconomica)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (ActividadEconomica) getHibernateTemplate().get(ActividadEconomica.class, codActividadEconomica);	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
MÉTODO:	saveActividadEconomica
ARGUMENTO:	(ActividadEconomica actividadEconomica)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(actividadEconomica);	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
MÉTODO:	updateActividadEconomica
ARGUMENTO:	(ActividadEconomica actividadEconomica)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(actividadEconomica);	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
---------	--------------------

MÉTODO:	deleteActividadEconomica
ARGUMENTO:	(ActividadEconomica actividadEconomica)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(actividadEconomica);	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
MÉTODO:	getAllActividadEconomicaBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from ActividadEconomica where (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	ActividadEconomica
MÉTODO:	getAllActividadEconomicaBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from ActividadEconomica where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

Tabla 162. Objeto ActividadEconomica

2.1.2.2.3.10.1.4.2 Objeto Asegurado

OBJETO:	Asegurado
MÉTODO:	getAllAsegurado
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Asegurado.class);	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAsegurado
ARGUMENTO:	(int codAsegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Asegurado) getHibernateTemplate().get(Asegurado.class, codAsegurado);	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	saveAsegurado
ARGUMENTO:	(Asegurado asegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(asegurado);	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	updateAsegurado
ARGUMENTO:	(Asegurado asegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(asegurado);	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	deleteAsegurado
ARGUMENTO:	(Asegurado asegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(asegurado);	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoBuscarByApAmNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Asegurado where (upper(persona.apepat ' persona.apemat ' persona.nombre) LIKE upper('"+dato+"%')) order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Asegurado where (estado="+estado+"") order by persona.nombre asc");	
DESCRIPCIÓN	

--

OBJETO:	Asegurado
MÉTODO:	getAseguradoByCodper
ARGUMENTO:	(int codper)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from Asegurado where (persona.codPersona="+codper+") order by persona.nombre asc"); if(lista.size(>0){ return (Asegurado)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoBuscarByApAmNombreAndCodtipsegNotIsAVC04A
ARGUMENTO:	(String dato,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("Select a from Asegurado a where (a.codAsegurado NOT IN (Select avc.asegurado.codAsegurado from Avc04A avc where (avc.tipoSeguro.codTipoSeguro="+codtipseg+"))) and (upper(a.persona.apepat ' a.persona.apemat ' a.persona.nombre) LIKE upper("+dato+"%')) order by a.persona.nombre asc");</pre>	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoBuscarByCiAndCodtipsegNotIsAVC04A
ARGUMENTO:	(int ci,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("Select a from Asegurado a where (a.codAsegurado NOT IN (Select avc.asegurado.codAsegurado from Avc04A avc where (avc.tipoSeguro.codTipoSeguro="+codtipseg+"))) and (a.persona.ci="+ci+") order by a.persona.nombre asc");</pre>	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoBuscarByApAmNombreAndNotIsAVC04
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("Select a from Asegurado a where (a.codAsegurado NOT IN (Select avc.asegurado.codAsegurado from Avc04 avc)) and (upper(a.persona.apepat ' a.persona.apemat ' a.persona.nombre) LIKE upper("+dato+"%')) order by a.persona.nombre asc");</pre>	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAllAseguradoBuscarByCiAndNotIsAVC04
ARGUMENTO:	(int ci);
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select a from Asegurado a where (a.codAsegurado NOT IN (Select avc.asegurado.codAsegurado from Avc04 avc)) and (a.persona.ci="+ci+") order by a.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAseguradoBuscarByApAmNombreAndCi
ARGUMENTO:	(String dato,int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Asegurado where (persona.ci="+ci+") and (upper(persona.apemat ' persona.apemat ' persona.nombre) LIKE upper("+dato+"%")) order by persona.nombre asc"); if(lista.size(>0){ return (Asegurado)lista.get(0); } return null;	
DESCRIPCIÓN	

OBJETO:	Asegurado
MÉTODO:	getAseguradoByNroasegurado
ARGUMENTO:	(String nroasegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Asegurado where (nroAsegurado="+nroasegurado+") order by fechaRegistro asc"); if(lista.size(>0){ return (Asegurado)lista.get(0); } return null;	
DESCRIPCIÓN	

Tabla 163. Objeto Asegurado

2.1.2.2.3.10.1.4.3 Objeto Avc01

OBJETO:	Avc01
MÉTODO:	getAllAvc01
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc01.class);	
DESCRIPCIÓN	

OBJETO:	Avc01
---------	-------

MÉTODO:	getAvc01
ARGUMENTO:	(int codAvc01)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc01) getHibernateTemplate().get(Avc01.class, codAvc01);	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	saveAvc01
ARGUMENTO:	(Avc01 avc01)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc01);	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	updateAvc01
ARGUMENTO:	(Avc01 avc01)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc01);	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	deleteAvc01
ARGUMENTO:	(Avc01 avc01)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc01);	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByNombreoRazonSocial
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (upper(empresa.nombreRazonsocial) LIKE upper('"+dato+"%')) order by empresa.nombreRazonsocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByNombreoRazonSocialAndCodacteco
ARGUMENTO:	(String dato,int codacteco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (upper(empresa.nombreRazonSocial) LIKE upper(""+dato+"%')) and (empresa.actividadEconomica.codActividadEconomica="+codacteco+")order by empresa.nombreRazonSocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (estado="+estado+") order by empresa.nombreRazonSocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByNroEmpresa
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (empresa.codemp="+dato+") order by empresa.nombreRazonSocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByNroEmpresaAndCodacteco
ARGUMENTO:	(String dato,int codacteco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (empresa.codemp="+dato+") and (empresa.actividadEconomica.codActividadEconomica="+codacteco+")order by empresa.nombreRazonSocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (fechaRecepcion="+fecha+") order by empresa.nombreRazonSocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc01
MÉTODO:	getAllAvc01ByEstadoAndFechaIniAndFechaFin
ARGUMENTO:	(int estado, Date ini, Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc01 where (fechaRepcion>='"+ini+"' and (fechaRepcion<='"+fin+"'))");	
DESCRIPCIÓN	

Tabla 164. Objeto Avc01

2.1.2.2.3.10.1.4.4 Objeto Avc02

OBJETO:	Avc02
MÉTODO:	getAllAvc02
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc02.class);	
DESCRIPCIÓN	

OBJETO:	Avc02
MÉTODO:	getAvc02
ARGUMENTO:	(int codAvc02)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc02) getHibernateTemplate().get(Avc02.class, codAvc02);	
DESCRIPCIÓN	

OBJETO:	Avc02
MÉTODO:	saveAvc02
ARGUMENTO:	(Avc02 avc02)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc02);	
DESCRIPCIÓN	

OBJETO:	Avc02
MÉTODO:	updateAvc02
ARGUMENTO:	(Avc02 avc02)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().update(avc02);
DESCRIPCIÓN

OBJETO:	Avc02
MÉTODO:	deleteAvc02
ARGUMENTO:	(Avc02 avc02)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc02);	
DESCRIPCIÓN	

OBJETO:	Avc02
MÉTODO:	getAvc02ByCodavc01
ARGUMENTO:	(int codavc01)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from Avc02 where (avc01.codAvc01="+codavc01+") order by fechaRegistro asc"); if(lista.size(>0){ return (Avc02)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 165. Objeto Avc02

2.1.2.2.3.10.1.4.5 Objeto Avc03

OBJETO:	Avc03
MÉTODO:	getAllAvc03
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc03.class);	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	getAvc03
ARGUMENTO:	(int codAvc03)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc03) getHibernateTemplate().get(Avc03.class, codAvc03);	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	saveAvc03
ARGUMENTO:	(Avc03 avc03)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc03);	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	updateAvc03
ARGUMENTO:	(Avc03 avc03)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc03);	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	deleteAvc03
ARGUMENTO:	(Avc03 avc03)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc03);	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	getAllAvc03ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc03 where (fechaRecepcion='"+fecha+"'") order by avc01.empresa.nombreRazonsocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	getAllAvc03ByFechaRecepcionByTipo
ARGUMENTO:	(Date fecha, int tipo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc03 where (fechaRecepcion='"+fecha+"'")and (tipo='"+tipo+"'") order by avc01.empresa.nombreRazonsocial asc");	
DESCRIPCIÓN	

OBJETO:	Avc03
MÉTODO:	public List getAllAvc03ByFechaRecepcionByEstadoAndTipoAndFechaIniAndFechaFin
ARGUMENTO:	(int estado,int tipo, Date ini,Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().find("from Avc03 where (fechaRecepcion>='"+ini+"' and (fechaRecepcion<='"+fin+"' and (tipo='"+tipo+"' and (estado='"+estado+""));
DESCRIPCIÓN	

Tabla 166. Objeto Avc03

2.1.2.2.3.10.1.4.6 Objeto Avc04A

OBJETO:	Avc04A
MÉTODO:	getAllAvc04A
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().loadAll(Avc04A.class);
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAvc04A
ARGUMENTO:	(int codAvc04A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return (Avc04A) getHibernateTemplate().get(Avc04A.class, codAvc04A);
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	saveAvc04A
ARGUMENTO:	(Avc04A avc04A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	getHibernateTemplate().save(avc04A);
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	updateAvc04A
ARGUMENTO:	(Avc04A avc04A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().update(avc04A);
DESCRIPCIÓN

OBJETO:	Avc04A
MÉTODO:	deleteAvc04A
ARGUMENTO:	(Avc04A avc04A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc04A);	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04ABuscarByApAmNombreAndCodtipseg
ARGUMENTO:	(String dato,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (tipoSeguro.codTipoSeguro="+codtipseg+") and (upper(asegurado.persona.apepat asegurado.persona.apemat asegurado.persona.nombre) LIKE upper("+dato+"%')) order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04ABuscarByCiAndCodtipseg
ARGUMENTO:	(int ci,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (tipoSeguro.codTipoSeguro="+codtipseg+") and (asegurado.persona.ci="+ci+") order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04ABuscarByNroAseguradoAndCodtipseg
ARGUMENTO:	(String nroasegurado,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (tipoSeguro.codTipoSeguro="+codtipseg+") and (asegurado.persona.nroAsegurado="+nroasegurado+") order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04AByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (fechaRecepcion='"+fecha+"'") order by asegurado.persona.apepat asc");	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04AByFechaRecepcionAndcodTipoSeguro
ARGUMENTO:	(Date fecha,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (fechaRecepcion='"+fecha+"'") and (tipoSeguro.codTipoSeguro='"+codtipseg+"'") order by asegurado.persona.apepat asc");	
DESCRIPCIÓN	

OBJETO:	Avc04A
MÉTODO:	getAllAvc04ByEstadoAndFechainiAndFechafinAndTipoSeguro
ARGUMENTO:	(int estado,Date ini,Date fin,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04A where (fechaRecepcion>='"+ini+"'") and (fechaRecepcion<='"+fin+"'") and (estado='"+estado+"'") and (tipoSeguro.codTipoSeguro='"+codtipseg+"'");	
DESCRIPCIÓN	

Tabla 167. Objeto Avc04A

2.1.2.2.3.10.1.4.7 Objeto Avc04

OBJETO:	Avc04
MÉTODO:	getAllAvc04
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc04.class);	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAvc04
ARGUMENTO:	(int codAvc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc04) getHibernateTemplate().get(Avc04.class, codAvc04);	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	saveAvc04

ARGUMENTO:	(Avc04 avc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc04);	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	updateAvc04
ARGUMENTO:	(Avc04 avc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc04);	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	deleteAvc04
ARGUMENTO:	(Avc04 avc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc04);	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04ByNotEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (estado<>"+estado+"") order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByApAmNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (upper(asegurado.persona.apemat) asegurado.persona.apemat " asegurado.persona.nombre) LIKE upper('"+dato+"%')) order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
---------	-------

MÉTODO:	getAllAvc04BuscarByApAmNombreAndNotEstado
ARGUMENTO:	(String dato,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (estado<>"+estado+") and (upper(asegurado.persona.apepat ' ' asegurado.persona.apemat ' ' asegurado.persona.nombre) LIKE upper("'" +dato+"%')) order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByCi
ARGUMENTO:	(int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (asegurado.persona.ci="+ci+") order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByNroAsegurado
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (asegurado.nroAsegurado='"+dato+"') order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByCiAndNotEstado
ARGUMENTO:	(int ci,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (estado<>"+estado+") and (asegurado.persona.ci="+ci+") order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByNroAseguradoAndNotEstado
ARGUMENTO:	(String dato,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc04 where (estado<>"+estado+") and (asegurado.nroAsegurado='"+dato+"') order by asegurado.persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04BuscarByIdavc01AndNotIsBaja
ARGUMENTO:	(int idavc)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().find("select a from Avc04 a where (a.avc01.codAvc01="+idavc+") and (a.codAvc04 NOT IN (Select avc.avc04.codAvc04 from Avc07 avc)) order by a.asegurado.persona.nombre asc");
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().find("from Avc04 where (fechaRecepcion="+fecha+"") order by asegurado.persona.aepat asc");
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04ByFechaRecepcionByTipo
ARGUMENTO:	(Date fecha, int tipo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().find("from Avc04 where (fechaRecepcion="+fecha+"") and (tipo="+tipo+"")order by asegurado.persona.aepat asc");
DESCRIPCIÓN	

OBJETO:	Avc04
MÉTODO:	getAllAvc04ByEstadoAndFechainiAndFechafin
ARGUMENTO:	(int estado,Date ini,Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().find("from Avc04 where (fechaRecepcion>="+ini+"") and (fechaRecepcion<="+fin+"") and (estado="+estado+"");
DESCRIPCIÓN	

Tabla 168. Objeto Avc04

2.1.2.2.3.10.1.4.8 Objeto Avc05

OBJETO:	Avc05
MÉTODO:	getAllAvc05()
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	return getHibernateTemplate().loadAll(Avc05.class);
DESCRIPCIÓN	

--

OBJETO:	Avc05
MÉTODO:	getAvc05
ARGUMENTO:	(int codAvc05)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc05) getHibernateTemplate().get(Avc05.class, codAvc05);	
DESCRIPCIÓN	

OBJETO:	Avc05
MÉTODO:	saveAvc05
ARGUMENTO:	(Avc05 avc05)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc05);	
DESCRIPCIÓN	

OBJETO:	Avc05
MÉTODO:	updateAvc05
ARGUMENTO:	(Avc05 avc05)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc05);	
DESCRIPCIÓN	

OBJETO:	Avc05
MÉTODO:	deleteAvc05
ARGUMENTO:	(Avc05 avc05)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc05);	
DESCRIPCIÓN	

OBJETO:	Avc05
MÉTODO:	getAvc05ByCodavc04
ARGUMENTO:	(int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Avc05 where (avc04.codAvc04='"+codavc04+"') order by fechaRegistro asc"); if(lista.size(>0){	

<pre> return (Avc05)lista.get(0); } return null; </pre>
DESCRIPCIÓN

Tabla 169. Objeto Avc05

2.1.2.2.3.10.1.4.9 Objeto Avc06

OBJETO:	Avc06
MÉTODO:	getAllAvc06
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc06.class);	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	getAvc06
ARGUMENTO:	(int codAvc06)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc06) getHibernateTemplate().get(Avc06.class, codAvc06);	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	saveAvc06(Avc06 avc06)
ARGUMENTO:	(Avc06 avc06)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc06);	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	updateAvc06
ARGUMENTO:	(Avc06 avc06)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc06);	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	deleteAvc06
ARGUMENTO:	(Avc06 avc06)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc06);	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	getAllAvc06BuscarByCodAvc04
ARGUMENTO:	(int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc06 where (codAvc06<>1) and (avc04.codAvc04="+codavc04+") order by fechaAviso asc");	
DESCRIPCIÓN	

OBJETO:	Avc06
MÉTODO:	getAllAvc06BuscarByCodAvc04AndByFechaAviso
ARGUMENTO:	(int codavc04,String fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc06 where (codAvc06<>1) and (avc04.codAvc04="+codavc04+") and (fechaAviso="+fecha+") order by fechaAviso asc");	
DESCRIPCIÓN	

Tabla 170. Objeto Avc06

2.1.2.2.3.10.1.4.10 Objeto Avc07A

OBJETO:	Avc07A
MÉTODO:	getAllAvc07A
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc07A.class);	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	getAvc07A
ARGUMENTO:	(int codAvc07A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc07A) getHibernateTemplate().get(Avc07A.class, codAvc07A);	
DESCRIPCIÓN	

OBJETO:	Avc07A
---------	--------

MÉTODO:	saveAvc07A
ARGUMENTO:	(Avc07A avc07A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc07A);	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	updateAvc07A
ARGUMENTO:	(Avc07A avc07A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc07A);	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	deleteAvc07A
ARGUMENTO:	(Avc07A avc07A)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc07A);	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	getAllAvc07AByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc07A where (fechaRecepcion='"+fecha+"'") order by avc04A.asegurado.persona.apepat asc");	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	getAllAvc07AByFechaRecepcionAndcodTipoSeguro
ARGUMENTO:	(Date fecha,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc07A where (fechaRecepcion='"+fecha+"'") and (avc04A.tipoSeguro.codTipoSeguro='"+codtipseg+"'") order by avc04A.asegurado.persona.apepat asc");	
DESCRIPCIÓN	

OBJETO:	Avc07A
MÉTODO:	getAllAvc07ByEstadoAndFechainiAndFechaFinAndCodtipseg
ARGUMENTO:	(int estado,Date ini,Date fin,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc07A where (estado="+estado+") and (fechaRecepcion>="+ini+") and (fechaRecepcion<="+fin+") and (avc04A.tipoSeguro.codTipoSeguro="+codtipseg+)");	
DESCRIPCIÓN	

Tabla 171. Objeto Avc07A

2.1.2.2.3.10.1.4.11 Objeto Avc07

OBJETO:	Avc07
MÉTODO:	getAllAvc07
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc07.class);	
DESCRIPCIÓN	

OBJETO:	Avc07
MÉTODO:	getAvc07
ARGUMENTO:	(int codAvc07)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc07) getHibernateTemplate().get(Avc07.class, codAvc07);	
DESCRIPCIÓN	

OBJETO:	Avc07
MÉTODO:	saveAvc07
ARGUMENTO:	(Avc07 avc07)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc07);	
DESCRIPCIÓN	

OBJETO:	Avc07
MÉTODO:	updateAvc07
ARGUMENTO:	(Avc07 avc07)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc07);	
DESCRIPCIÓN	

--

OBJETO:	Avc07
MÉTODO:	deleteAvc07
ARGUMENTO:	(Avc07 avc07)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc07);	
DESCRIPCIÓN	

OBJETO:	Avc07
MÉTODO:	getAllAvc07ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc07 where (fechaRecepcion='"+fecha+"'") order by avc04.asegurado.persona.apepat asc");	
DESCRIPCIÓN	

OBJETO:	Avc07
MÉTODO:	getAllAvc07ByEstadoAndFechainiAndFechafin
ARGUMENTO:	(int estado,Date ini,Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc07 where (estado='"+estado+"' and (fechaRecepcion>='"+ini+"' and (fechaRecepcion<='"+fin+"'"));	
DESCRIPCIÓN	

Tabla 172. Objeto Avc07

2.1.2.2.3.10.1.4.12 Objeto Avc08

OBJETO:	Avc08
MÉTODO:	getAllAvc08
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc08.class);	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	getAvc08
ARGUMENTO:	(int codAvc08)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

return (Avc08) getHibernateTemplate().get(Avc08.class, codAvc08);
DESCRIPCIÓN

OBJETO:	Avc08
MÉTODO:	saveAvc08
ARGUMENTO:	(Avc08 avc08)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc08);	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	updateAvc08
ARGUMENTO:	(Avc08 avc08)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc08);	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	deleteAvc08
ARGUMENTO:	(Avc08 avc08)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc08);	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	getAllAvc08BuscarByCodAvc04
ARGUMENTO:	(int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc08 where (avc04.codAvc04="+codavc04+") order by fecha asc");	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	getAllAvc08BuscarByCodAvc04AndByFecha
ARGUMENTO:	(int codavc04,String fecha)
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc08 where (avc04.codAvc04="+codavc04+") and (fecha="+fecha+"") order by fecha asc");	
DESCRIPCIÓN	

OBJETO:	Avc08
MÉTODO:	getAllAvc08ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc08 where (fecha="+fecha+"") order by avc04.asegurado.persona.apepat asc");	
DESCRIPCIÓN	

Tabla 173. Objeto Avc08

2.1.2.2.3.10.1.4.13 Objeto Avc09

OBJETO:	Avc09
MÉTODO:	getAllAvc09
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Avc09.class);	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	getAvc09
ARGUMENTO:	(int codAvc09)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Avc09) getHibernateTemplate().get(Avc09.class, codAvc09);	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	saveAvc09
ARGUMENTO:	(Avc09 avc09)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(avc09); }	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	updateAvc09

ARGUMENTO:	(Avc09 avc09)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(avc09);	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	deleteAvc09
ARGUMENTO:	(Avc09 avc09)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(avc09);	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	getAllAvc09BuscarByCodAvc04
ARGUMENTO:	(int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc09 where (avc04.codAvc04="+codavc04+") order by fecha asc");	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	getAllAvc09BuscarByCodAvc04AndByFecha
ARGUMENTO:	(int codavc04,String fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc09 where (avc04.codAvc04="+codavc04+") and (fecha="+fecha+") order by fecha asc");	
DESCRIPCIÓN	

OBJETO:	Avc09
MÉTODO:	getAllAvc09ByFechaRecepcion
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Avc09 where (fecha="+fecha+") order by avc04.asegurado.persona.aepat asc");	
DESCRIPCIÓN	

Tabla 174. Objeto Avc09

2.1.2.2.3.10.1.4.14 Objeto Backup

OBJETO:	Backup
MÉTODO:	getAllBackup
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Backup.class);	
DESCRIPCIÓN	

OBJETO:	Backup
MÉTODO:	getBackup
ARGUMENTO:	(int codBackup)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Backup) getHibernateTemplate().get(Avc03.class, codBackup);	
DESCRIPCIÓN	

OBJETO:	Backup
MÉTODO:	saveBackup
ARGUMENTO:	(Backup backup)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(backup);	
DESCRIPCIÓN	

OBJETO:	Backup
MÉTODO:	updateBackup
ARGUMENTO:	(Backup backup)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(backup);	
DESCRIPCIÓN	

OBJETO:	Backup
MÉTODO:	deleteBackup
ARGUMENTO:	(Backup backup)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(backup);	
DESCRIPCIÓN	

--

OBJETO:	Backup
MÉTODO:	getAllBackupBuscarByTipo
ARGUMENTO:	(String tipo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Backup where (tipo='"+tipo+"'") order by fecha asc");	
DESCRIPCIÓN	

OBJETO:	Backup
MÉTODO:	getAllBackupBuscarByFecha
ARGUMENTO:	(String fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Backup where (fecha='"+fecha+"'") order by fecha asc");	
DESCRIPCIÓN	

Tabla 175. Objeto Backup

2.1.2.2.3.10.1.4.15

Objeto Beneficiario

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Beneficiario.class);	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getBeneficiario
ARGUMENTO:	(int codBeneficiario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Beneficiario) getHibernateTemplate().get(Beneficiario.class, codBeneficiario);	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	saveBeneficiario
ARGUMENTO:	(Beneficiario beneficiario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().save(beneficiario);
DESCRIPCIÓN

OBJETO:	Beneficiario
MÉTODO:	updateBeneficiario
ARGUMENTO:	(Beneficiario beneficiario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(beneficiario);	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	deleteBeneficiario
ARGUMENTO:	(Beneficiario beneficiario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(beneficiario);	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getBeneficiarioBuscarByApAndAmAndNombreAndCodigoAndCodasegurado
ARGUMENTO:	(String ap,String am,String nombre,int codigo,int codasegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from Beneficiario where (asegurado.codAsegurado="+codasegurado+") and (codigo="+codigo+") and (upper(apepat ' apemat ' nombre) LIKE upper('"+(ap+" "+am+" "+nombre)+"%')) order by codigo asc"); if(lista.size(>0){ return (Beneficiario)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByCodasegurado
ARGUMENTO:	(int codasegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Beneficiario where (asegurado.codAsegurado="+codasegurado+") order by codigo asc");	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByCodaseguradoAndCodigoBeneficiario
ARGUMENTO:	(int codasegurado,int codbeneficiario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista = getHibernateTemplate().find("from Beneficiario where (asegurado.codAsegurado="+codasegurado+") and (codigo="+codbeneficiario+") order by codigo asc"); if(lista.size(>0){ return (Beneficiario)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarIsInAltaByCodaseguradoAndcodavc04
ARGUMENTO:	(int codasegurado,int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("from Beneficiario b where (b.asegurado.codAsegurado="+codasegurado+") and ((select count(*) from GrupoFamiliar gf where (gf.beneficiario.codBeneficiario=b.codBeneficiario) and (gf.avc04.codAvc04="+codavc04+") and (gf.tipo=1)) > (select count(*) from GrupoFamiliar gf where (gf.beneficiario.codBeneficiario=b.codBeneficiario) and (gf.avc04.codAvc04="+codavc04+") and (gf.tipo=0))) order by b.codigo asc");</pre>	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarIsInBajaByCodaseguradoAndcodavc04
ARGUMENTO:	(int codasegurado,int codavc04)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("from Beneficiario b where (b.asegurado.codAsegurado="+codasegurado+") and ((select count(*) from GrupoFamiliar gf where (gf.beneficiario.codBeneficiario=b.codBeneficiario) and (gf.avc04.codAvc04="+codavc04+") and (gf.tipo=1)) = (select count(*) from GrupoFamiliar gf where (gf.beneficiario.codBeneficiario=b.codBeneficiario) and (gf.avc04.codAvc04="+codavc04+") and (gf.tipo=0))) order by b.codigo asc");</pre>	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByApAmNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>return getHibernateTemplate().find("from Beneficiario where (upper(apepat ' apemat ' nombre) LIKE upper('"+dato+"%')) order by apepat asc");</pre>	
DESCRIPCIÓN	

--

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByCodigo
ARGUMENTO:	(int codigo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Beneficiario where (codigo="+codigo+") order by apepat asc");	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByNroAsegurado
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Beneficiario where (asegurado.nroAsegurado="+dato+") order by apepat asc");	
DESCRIPCIÓN	

OBJETO:	Beneficiario
MÉTODO:	getAllBeneficiarioBuscarByEdad
ARGUMENTO:	(int edad,int norma)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> boolean estado=true; List lista=new LinkedList(); Date date=new Date(); SimpleDateFormat formatDateFormat=new SimpleDateFormat("dd/MM/yyyy"); String dato=formatDateFormat.format(date); String datofecha=""; int anio=Integer.parseInt(dato.split("/")[2])-edad; int mes=Integer.parseInt(dato.split("/")[1]); int dia=Integer.parseInt(dato.split("/")[0]); datofecha=dia+"/"+mes+"/"+anio;//1985-09-30 try { date=formatDateFormat.parse(datofecha); } catch (ParseException e1) { // TODO Auto-generated catch block e1.printStackTrace(); } while(estado){ try{ lista=getHibernateTemplate().find("from Beneficiario where (fnac<"+date+"") and (norma="+norma+") order by apepat asc");//1985-09-30 estado=false; } catch (Exception e) { dia--; estado=true; } } </pre>	
DESCRIPCIÓN	

<pre> } } return lista; </pre>
DESCRIPCIÓN

Tabla 176. Objeto Beneficiario

2.1.2.2.3.10.1.4.16 Objeto CambioPersona

OBJETO:	CambioPersona
MÉTODO:	getAllCambioPersona
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(CambioPersona.class);	
DESCRIPCIÓN	

OBJETO:	CambioPersona
MÉTODO:	getCambioPersona
ARGUMENTO:	(int codCambioPersona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (CambioPersona) getHibernateTemplate().get(CambioPersona.class, codCambioPersona);	
DESCRIPCIÓN	

OBJETO:	CambioPersona
MÉTODO:	saveCambioPersona
ARGUMENTO:	(CambioPersona cambioPersona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(cambioPersona);	
DESCRIPCIÓN	

OBJETO:	CambioPersona
MÉTODO:	updateCambioPersona
ARGUMENTO:	(CambioPersona cambioPersona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(cambioPersona);	
DESCRIPCIÓN	

OBJETO:	CambioPersona
MÉTODO:	deleteCambioPersona
ARGUMENTO:	(CambioPersona cambioPersona)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(cambioPersona);	
DESCRIPCIÓN	

OBJETO:	CambioPersona
MÉTODO:	getCambioPersonaByCodperAndEstado
ARGUMENTO:	(int codper,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from CambioPersona where (persona.codPersona="+codper+") and (estado="+estado+") order by persona.nombre asc"); if(lista.size(>0){ return (CambioPersona)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 177. Objeto CambioPersona

2.1.2.2.3.10.1.4.17 Objeto CentroSalud

OBJETO:	CentroSalud
MÉTODO:	getAllCentroSalud
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(CentroSalud.class);	
DESCRIPCIÓN	

OBJETO:	CentroSalud
MÉTODO:	getCentroSalud
ARGUMENTO:	(int codCentroSalud)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (CentroSalud) getHibernateTemplate().get(CentroSalud.class, codCentroSalud);	
DESCRIPCIÓN	

OBJETO:	CentroSalud
MÉTODO:	saveCentroSalud
ARGUMENTO:	(CentroSalud centroSalud)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(centroSalud);	
DESCRIPCIÓN	

--

OBJETO:	CentroSalud
MÉTODO:	updateCentroSalud
ARGUMENTO:	(CentroSalud centroSalud)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(centroSalud);	
DESCRIPCIÓN	

OBJETO:	CentroSalud
MÉTODO:	deleteCentroSalud
ARGUMENTO:	(CentroSalud centroSalud)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(centroSalud);	
DESCRIPCIÓN	

OBJETO:	CentroSalud
MÉTODO:	getAllCentroSaludBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from CentroSalud where (upper(nombre) LIKE upper("'" + dato + "%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	CentroSalud
MÉTODO:	getAllCentroSaludBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from CentroSalud where (estado='"+ estado +"") order by nombre asc");	
DESCRIPCIÓN	

Tabla 178. Objeto CentroSalud

2.1.2.2.3.10.1.4.18

Objeto Departamento

OBJETO:	Departamento
MÉTODO:	getAllDepartamento
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

return getHibernateTemplate().loadAll(Departamento.class);
DESCRIPCIÓN

OBJETO:	Departamento
MÉTODO:	getDepartamento
ARGUMENTO:	(int codDepartamento)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Departamento) getHibernateTemplate().get(Departamento.class, codDepartamento);	
DESCRIPCIÓN	

OBJETO:	Departamento
MÉTODO:	saveDepartamento
ARGUMENTO:	(Departamento departamento)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(departamento);	
DESCRIPCIÓN	

OBJETO:	Departamento
MÉTODO:	updateDepartamento
ARGUMENTO:	(Departamento departamento)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(departamento);	
DESCRIPCIÓN	

OBJETO:	Departamento
MÉTODO:	deleteDepartamento
ARGUMENTO:	(Departamento departamento)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(departamento);	
DESCRIPCIÓN	

OBJETO:	Departamento
MÉTODO:	getAllDepartamentoBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN
return getHibernateTemplate().find("from Departamento where (upper(nombre) LIKE upper("'" + dato + "%')) order by nombre asc");
DESCRIPCIÓN

Tabla 179. Objeto Departamento

2.1.2.2.3.10.1.4.19 Objeto DireccionAsegurado

OBJETO:	DireccionAsegurado
MÉTODO:	getAllDireccionAsegurado
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(DireccionAsegurado.class);	
DESCRIPCIÓN	

OBJETO:	DireccionAsegurado
MÉTODO:	getDireccionAsegurado
ARGUMENTO:	(int codDireccionAsegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (DireccionAsegurado) getHibernateTemplate().get(DireccionAsegurado.class, codDireccionAsegurado);	
DESCRIPCIÓN	

OBJETO:	DireccionAsegurado
MÉTODO:	saveDireccionAsegurado
ARGUMENTO:	(DireccionAsegurado direccionAsegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(direccionAsegurado);	
DESCRIPCIÓN	

OBJETO:	DireccionAsegurado
MÉTODO:	updateDireccionAsegurado
ARGUMENTO:	(DireccionAsegurado direccionAsegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(direccionAsegurado);	
DESCRIPCIÓN	

OBJETO:	DireccionAsegurado
MÉTODO:	deleteDireccionAsegurado

ARGUMENTO:	(DireccionAsegurado direccionAsegurado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(direccionAsegurado);	
DESCRIPCIÓN	

OBJETO:	DireccionAsegurado
MÉTODO:	getDireccionAseguradoByCodavc04AndEstado
ARGUMENTO:	(int codavc04,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from DireccionAsegurado where (avc04.codAvc04="+codavc04+" and (estado="+estado+") order by calle asc"); if(lista.size(>0){ return (DireccionAsegurado)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 180. Objeto DireccionAsegurado

2.1.2.2.3.10.1.4.20 Objeto Direccion

OBJETO:	Direccion
MÉTODO:	getAllDireccion
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Direccion.class);	
DESCRIPCIÓN	

OBJETO:	Direccion
MÉTODO:	getDireccion
ARGUMENTO:	(int codDireccion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Direccion) getHibernateTemplate().get(Direccion.class, codDireccion);	
DESCRIPCIÓN	

OBJETO:	Direccion
MÉTODO:	saveDireccion
ARGUMENTO:	(Direccion direccion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(direccion);	

DESCRIPCIÓN

OBJETO:	Direccion
MÉTODO:	updateDireccion
ARGUMENTO:	(Direccion direccion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(direccion);	
DESCRIPCIÓN	

OBJETO:	Direccion
MÉTODO:	deleteDireccion
ARGUMENTO:	(Direccion direccion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(direccion);	
DESCRIPCIÓN	

OBJETO:	Direccion
MÉTODO:	getDireccionByEstadoAndCodavc04a
ARGUMENTO:	(int codavc04a,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from Direccion where (avc04A.codAvc04A="+codavc04a+") and (estado="+estado+") order by calle asc"); if(lista.size(>0){ return (Direccion)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 181. Objeto Direccion

2.1.2.2.3.10.1.4.21 Objeto DireccionEmpresa

OBJETO:	DireccionEmpresa
MÉTODO:	getAllDireccionEmpresa
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(DireccionEmpresa.class);	
DESCRIPCIÓN	

OBJETO:	DireccionEmpresa
MÉTODO:	getDireccionEmpresa

ARGUMENTO:	(int codDireccionEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (DireccionEmpresa) getHibernateTemplate().get(DireccionEmpresa.class, codDireccionEmpresa);	
DESCRIPCIÓN	

OBJETO:	DireccionEmpresa
MÉTODO:	saveDireccionEmpresa
ARGUMENTO:	(DireccionEmpresa direccionEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(direccionEmpresa);	
DESCRIPCIÓN	

OBJETO:	DireccionEmpresa
MÉTODO:	updateDireccionEmpresa
ARGUMENTO:	(DireccionEmpresa direccionEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(direccionEmpresa);	
DESCRIPCIÓN	

OBJETO:	DireccionEmpresa
MÉTODO:	deleteDireccionEmpresa
ARGUMENTO:	(DireccionEmpresa direccionEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(direccionEmpresa);	
DESCRIPCIÓN	

OBJETO:	DireccionEmpresa
MÉTODO:	getDireccionEmpresaByCodempAndEstado
ARGUMENTO:	(int codemp,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from DireccionEmpresa where (empresa.codEmpresa="+codemp+")and (estado="+estado+") order by calle asc"); if(lista.size(>0){ return (DireccionEmpresa)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 182. Objeto DireccionEmpresa

2.1.2.2.3.10.1.4.22 Objeto Empresa

OBJETO:	Empresa
MÉTODO:	getAllEmpresa
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Empresa.class);	
DESCRIPCIÓN	

OBJETO:	Empresa
MÉTODO:	getEmpresa
ARGUMENTO:	(int codEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Empresa) getHibernateTemplate().get(Empresa.class, codEmpresa);	
DESCRIPCIÓN	

OBJETO:	Empresa
MÉTODO:	saveEmpresa
ARGUMENTO:	(Empresa empresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(empresa);	
DESCRIPCIÓN	

OBJETO:	Empresa
MÉTODO:	updateEmpresa
ARGUMENTO:	(Empresa empresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(empresa);	
DESCRIPCIÓN	

OBJETO:	Empresa
MÉTODO:	deleteEmpresa
ARGUMENTO:	(Empresa empresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().delete(empresa);
DESCRIPCIÓN

OBJETO:	Empresa
MÉTODO:	getEmpresaByNombreRazonsocialAndCodemp
ARGUMENTO:	(String nombreRazonsocial,String codemp)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista= getHibernateTemplate().find("from Empresa where (nombreRazonsocial='"+nombreRazonsocial+"') and (codemp='"+codemp+"') order by nombreRazonsocial asc"); if(lista.size(>0){ return (Empresa)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

OBJETO:	Empresa
MÉTODO:	getEmpresaByCodemp
ARGUMENTO:	(String codemp)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista= getHibernateTemplate().find("from Empresa where (codemp='"+codemp+"') order by nombreRazonsocial asc"); if(lista.size(>0){ return (Empresa)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 183. Objeto Empresa

2.1.2.2.3.10.1.4.23 Objeto Funcionalidad

OBJETO:	Funcionalidad
MÉTODO:	getAllFuncionalidad
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Funcionalidad.class);	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	getFuncionalidad
ARGUMENTO:	(String codFuncionalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

return (Funcionalidad) getHibernateTemplate().get(Funcionalidad.class, codFuncionalidad);
DESCRIPCIÓN

OBJETO:	Funcionalidad
MÉTODO:	saveFuncionalidad
ARGUMENTO:	(Funcionalidad funcionalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(funcionalidad);	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	updateFuncionalidad
ARGUMENTO:	(Funcionalidad funcionalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(funcionalidad);	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	deleteFuncionalidad
ARGUMENTO:	(Funcionalidad funcionalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(funcionalidad);	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	getAllFuncionalidadBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Funcionalidad where (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	getAllFuncionalidadByEstadoAndEstadoModulo
ARGUMENTO:	(int estado,int estadomod)
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN	
return getHibernateTemplate().find("from Funcionalidad where (estado="+estado+") and (modulo.estado="+estadomod+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Funcionalidad
MÉTODO:	getAllFuncionalidadBuscarNombreByAndCodmod
ARGUMENTO:	(String dato,int codmod)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Funcionalidad where (modulo.codModulo="+codmod+") and (upper(nombre) LIKE upper("'" +dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

Tabla 184. Objeto Funcionalidad

2.1.2.2.3.10.1.4.24 Objeto GrupoFamiliar

OBJETO:	GrupoFamiliar
MÉTODO:	getAllGrupoFamiliar
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(GrupoFamiliar.class);	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	getGrupoFamiliar
ARGUMENTO:	(int codGrupoFamiliar)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (GrupoFamiliar) getHibernateTemplate().get(GrupoFamiliar.class, codGrupoFamiliar);	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	saveGrupoFamiliar
ARGUMENTO:	(GrupoFamiliar grupoFamiliar)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(grupoFamiliar);	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	updateGrupoFamiliar

ARGUMENTO:	(GrupoFamiliar grupoFamiliar)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(grupoFamiliar);	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	deleteGrupoFamiliar
ARGUMENTO:	(GrupoFamiliar grupoFamiliar)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(grupoFamiliar);	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	getAllGrupoFamiliarBuscarByCodAvc06AndTipo
ARGUMENTO:	(int codavc06,int tipo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from GrupoFamiliar where (avc06.codAvc06='"+codavc06+"' and (tipo='"+tipo+"' order by fechaRegistro asc");	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	getAllGrupoFamiliarByFechaAviso
ARGUMENTO:	(Date fecha)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from GrupoFamiliar where (avc06.fechaAviso='"+fecha+"' order by beneficiario.apepat asc");	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	getAllGrupoFamiliarByFechaAvisoAndEstadoBeneficiario
ARGUMENTO:	(Date fecha,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from GrupoFamiliar where (avc06.fechaAviso='"+fecha+"' and (beneficiario.estado='"+estado+"' order by beneficiario.apepat asc");	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
---------	---------------

MÉTODO:	getAllGrupoFamiliarByEstadoAndFechainiAndFechafinAndYesAV04
ARGUMENTO:	(int estado,Date ini,Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from GrupoFamiliar where (avc04.fechaRecepcion>='"+ini+"'") and (avc04.fechaRecepcion<='"+fin+"'") and (estado='"+estado+"'") and (tipo='"+tipo+"'") and (avc06.codAvc06=1)");	
DESCRIPCIÓN	

OBJETO:	GrupoFamiliar
MÉTODO:	getAllGrupoFamiliarByEstadoAndFechainiAndFechafinAndNotAV04
ARGUMENTO:	(int estado,Date ini,Date fin)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from GrupoFamiliar where (avc06.fechaAviso>='"+ini+"'") and (avc06.fechaAviso<='"+fin+"'") and (estado='"+estado+"'") and (tipo='"+tipo+"'") and (avc06.codAvc06!=1)");	
DESCRIPCIÓN	

Tabla 185. ObjetoGrupoFamiliar

2.1.2.2.3.10.1.4.25 Objeto Módulo

OBJETO:	Modulo
MÉTODO:	getAllModulo
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Modulo.class);	
DESCRIPCIÓN	

OBJETO:	Modulo
MÉTODO:	getModulo
ARGUMENTO:	(int codModulo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Modulo) getHibernateTemplate().get(Modulo.class, codModulo);	
DESCRIPCIÓN	

OBJETO:	Modulo
MÉTODO:	getAllModuloBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Modulo where (upper(nombre) LIKE upper('" + dato + "%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Modulo
MÉTODO:	saveModulo
ARGUMENTO:	(Modulo modulo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(modulo);	
DESCRIPCIÓN	

OBJETO:	Modulo
MÉTODO:	updateModulo
ARGUMENTO:	(Modulo modulo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(modulo);	
DESCRIPCIÓN	

OBJETO:	Modulo
MÉTODO:	deleteModulo
ARGUMENTO:	(Modulo modulo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(modulo);	
DESCRIPCIÓN	

Tabla 186. ObjetoMódulo

2.1.2.2.3.10.1.4.26

Objeto MunicipioLocalidad

OBJETO:	MunicipioLocalidad
MÉTODO:	getAllMunicipioLocalidad
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(MunicipioLocalidad.class);	
DESCRIPCIÓN	

OBJETO:	MunicipioLocalidad
MÉTODO:	getMunicipioLocalidad
ARGUMENTO:	(int numMunicipioLocalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (MunicipioLocalidad) getHibernateTemplate().get(MunicipioLocalidad.class, numMunicipioLocalidad);	

DESCRIPCIÓN

OBJETO:	MunicipioLocalidad
MÉTODO:	saveMunicipioLocalidad
ARGUMENTO:	(MunicipioLocalidad municipioLocalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(municipioLocalidad);	
DESCRIPCIÓN	

OBJETO:	MunicipioLocalidad
MÉTODO:	updateMunicipioLocalidad
ARGUMENTO:	(MunicipioLocalidad municipioLocalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(municipioLocalidad);	
DESCRIPCIÓN	

OBJETO:	MunicipioLocalidad
MÉTODO:	deleteMunicipioLocalidad
ARGUMENTO:	(MunicipioLocalidad municipioLocalidad)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(municipioLocalidad);	
DESCRIPCIÓN	

OBJETO:	MunicipioLocalidad
MÉTODO:	getAllMunicipioLocalidadByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from MunicipioLocalidad where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	MunicipioLocalidad
MÉTODO:	getAllMunicipioLocalidadBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

return getHibernateTemplate().find("from MunicipioLocalidad where (upper(nombre) LIKE upper '"+dato+"%')) order by nombre asc");
DESCRIPCIÓN

OBJETO:	MunicipioLocalidad
MÉTODO:	getAllMunicipioLocalidadBuscarByNombreAndCodpro
ARGUMENTO:	(String dato,int codpro)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from MunicipioLocalidad where (provincia.codProvincia="+codpro+") and (upper(nombre) LIKE upper '"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

Tabla 187. ObjetoMunicipioLocalidad

2.1.2.2.3.10.1.4.27 Objeto NombreEmpresa

OBJETO:	NombreEmpresa
MÉTODO:	getAllNombreEmpresa
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(NombreEmpresa.class);	
DESCRIPCIÓN	

OBJETO:	NombreEmpresa
MÉTODO:	getNombreEmpresa
ARGUMENTO:	(int codNombreEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (NombreEmpresa) getHibernateTemplate().get(NombreEmpresa.class, codNombreEmpresa);	
DESCRIPCIÓN	

OBJETO:	NombreEmpresa
MÉTODO:	saveNombreEmpresa
ARGUMENTO:	(NombreEmpresa nombreEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(nombreEmpresa);	
DESCRIPCIÓN	

OBJETO:	NombreEmpresa
MÉTODO:	updateNombreEmpresa
ARGUMENTO:	(NombreEmpresa nombreEmpresa)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(nombreEmpresa);	
DESCRIPCIÓN	

OBJETO:	NombreEmpresa
MÉTODO:	deleteNombreEmpresa
ARGUMENTO:	(NombreEmpresa nombreEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(nombreEmpresa);	
DESCRIPCIÓN	

OBJETO:	NombreEmpresa
MÉTODO:	getNombreEmpresaByCodempAndEstado
ARGUMENTO:	(int codemp,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from NombreEmpresa where (empresa.codEmpresa="+codemp+")and (estado="+estado+") order by nombreRazonSocial asc"); if(lista.size(>0){ return (NombreEmpresa)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 188. ObjetoNombreEmpresa

2.1.2.2.3.10.1.4.28 Objeto Ocupacion

OBJETO:	Ocupacion
MÉTODO:	getAllOcupacion
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Ocupacion.class);	
DESCRIPCIÓN	

OBJETO:	Ocupacion
MÉTODO:	getOcupacion
ARGUMENTO:	(int codOcupacion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Ocupacion) getHibernateTemplate().get(Ocupacion.class, codOcupacion);	
DESCRIPCIÓN	

--

OBJETO:	Ocupacion
MÉTODO:	saveOcupacion
ARGUMENTO:	(Ocupacion ocupacion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(ocupacion);	
DESCRIPCIÓN	

OBJETO:	Ocupacion
MÉTODO:	updateOcupacion
ARGUMENTO:	(Ocupacion ocupacion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(ocupacion);	
DESCRIPCIÓN	

OBJETO:	Ocupacion
MÉTODO:	deleteOcupacion
ARGUMENTO:	(Ocupacion ocupacion)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(ocupacion);	
DESCRIPCIÓN	

OBJETO:	Ocupacion
MÉTODO:	getAllOcupacionBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Ocupacion where (upper(nombre) LIKE upper '"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Ocupacion
MÉTODO:	getAllOcupacionBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Ocupacion where (estado="+estado+") order by nombre asc");	

DESCRIPCIÓN

Tabla 189. ObjetoOcupacion

2.1.2.2.3.10.1.4.29 Objeto Parentesco

OBJETO:	Parentesco
MÉTODO:	getAllParentesco
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Parentesco.class);	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	getParentesco
ARGUMENTO:	(int codParentesco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Parentesco) getHibernateTemplate().get(Parentesco.class, codParentesco);	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	saveParentesco
ARGUMENTO:	(Parentesco parentesco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(parentesco);	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	updateParentesco
ARGUMENTO:	(Parentesco parentesco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(parentesco);	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	deleteParentesco
ARGUMENTO:	(Parentesco parentesco)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().delete(parentesco);
DESCRIPCIÓN

OBJETO:	Parentesco
MÉTODO:	getAllParentescoBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Parentesco where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	getAllParentescoBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Parentesco where (upper(nombre) LIKE upper("+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Parentesco
MÉTODO:	getParentescoByNombre
ARGUMENTO:	(String nombre)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from Parentesco where (upper(nombre) LIKE upper("+nombre+"%')) order by nombre asc"); if(lista.size()>0){ return (Parentesco)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 190. ObjetoParentesco

2.1.2.2.3.10.1.4.30 Objeto Persona

OBJETO:	Persona
MÉTODO:	getAllPersona
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Persona.class);	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getPersona
ARGUMENTO:	(int codPersona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Persona) getHibernateTemplate().get(Persona.class, codPersona);	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	savePersona
ARGUMENTO:	(Persona persona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(persona);	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	updatePersona
ARGUMENTO:	(Persona persona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(persona);	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	deletePersona
ARGUMENTO:	(Persona persona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(persona);	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Persona where (upper(apeat ' apemat ' nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombreAndEstado
ARGUMENTO:	(String dato, int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Persona where (estado="+estado+") and (upper(apepat ' apemat ' nombre) LIKE upper("'" +dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Persona where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombreNotIsUsuario
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select u.persona.codPersona from Usuario u)) and (upper(apepat ' apemat ' nombre) LIKE upper("'" +dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombreNotIsAsegurado
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select a.persona.codPersona from Asegurado a)) and (upper(apepat ' apemat ' nombre) LIKE upper("'" +dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getPersonaBuscarByApAmNombreAndCi
ARGUMENTO:	(String dato,int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Persona where (ci="+ci+") and (upper(apepat ' apemat ' nombre) LIKE upper("'" +dato+"%')) order by nombre asc");	

<pre> if(lista.size(>0){ return (Persona)lista.get(0); } return null; </pre>
DESCRIPCIÓN

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByCi
ARGUMENTO:	(int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Persona where (ci="+ci+"") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByCiNotIsAsegurado
ARGUMENTO:	(int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select a.persona.codPersona from Asegurado a)) and (ci="+ci+"") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombreNotIsAVC04A
ARGUMENTO:	(String dato,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select avc.asegurado.persona.codPersona from Avc04A avc where (avc.tipoSeguro.codTipoSeguro="+codtipseg+"))) and (upper(apepat ' apemat ' nombre) LIKE upper("+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByCiNotIsAVC04A
ARGUMENTO:	(int ci,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select avc.asegurado.persona.codPersona from Avc04A avc where (avc.tipoSeguro.codTipoSeguro="+codtipseg+"))) and (ci="+ci+"") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByApAmNombreNotIsAVC04
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select avc.asegurado.persona.codPersona from Avc04 avc)) and (upper(apeat ' apemat ' nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Persona
MÉTODO:	getAllPersonaBuscarByCiNotIsAVC04
ARGUMENTO:	(int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("Select p from Persona p where (p.codPersona NOT IN (Select avc.asegurado.persona.codPersona from Avc04 avc)) and (ci="+ci+"") order by nombre asc");	
DESCRIPCIÓN	

Tabla 191. ObjetoPersona

2.1.2.2.3.10.1.4.31

Objeto Propietario

OBJETO:	Propietario
MÉTODO:	getAllPropietario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Propietario.class);	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	getPropietario
ARGUMENTO:	(int codPropietario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Propietario) getHibernateTemplate().get(Propietario.class, codPropietario);	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	savePropietario
ARGUMENTO:	(Propietario propietario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(proprietario);	
DESCRIPCIÓN	

--

OBJETO:	Propietario
MÉTODO:	updatePropietario
ARGUMENTO:	(Propietario propietario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(proprietario);	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	deletePropietario
ARGUMENTO:	(Propietario propietario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(proprietario);	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	getAllPropietarioBuscarByApAmNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Propietario where (upper(persona.aepat ' persona.apemat ' persona.nombre) LIKE upper '"+dato+"%')) order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	getAllPropietarioByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Propietario where (estado="+estado+") order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Propietario
MÉTODO:	getPropietarioByCodper
ARGUMENTO:	(int codper)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Propietario where (persona.codPersona="+codper+") order by persona.nombre asc"); if(lista.size(>0){	

<pre> return (Propietario)lista.get(0); } return null; </pre>
DESCRIPCIÓN

OBJETO:	Propietario
MÉTODO:	getPropietarioBuscarByApAmNombreAndCi
ARGUMENTO:	(String dato,int ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> List lista=getHibernateTemplate().find("from Propietario where (persona.ci="+ci+") and (upper(persona.apepat ' ' persona.apemat ' ' persona.nombre) LIKE upper("+dato+"%')) order by persona.nombre asc"); if(lista.size(>0){ return (Propietario)lista.get(0); } return null; </pre>	
DESCRIPCIÓN	

Tabla 192. ObjetoPropietario

2.1.2.2.3.10.1.4.32

Objeto PropietarioEmpleador

OBJETO:	PropietarioEmpleador
MÉTODO:	getAllPropietarioEmpleador
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> return getHibernateTemplate().loadAll(PropietarioEmpleador.class); </pre>	
DESCRIPCIÓN	

OBJETO:	PropietarioEmpleador
MÉTODO:	getPropietarioEmpleador
ARGUMENTO:	(int codPropietarioEmpleador)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> return (PropietarioEmpleador) getHibernateTemplate().get(PropietarioEmpleador.class, codPropietarioEmpleador); </pre>	
DESCRIPCIÓN	

OBJETO:	PropietarioEmpleador
MÉTODO:	savePropietarioEmpleador
ARGUMENTO:	(PropietarioEmpleador propietarioEmpleador)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> getHibernateTemplate().save(propietarioEmpleador); </pre>	
DESCRIPCIÓN	

--

OBJETO:	PropietarioEmpleador
MÉTODO:	updatePropietarioEmpleador
ARGUMENTO:	(PropietarioEmpleador propietarioEmpleador)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(proprietarioEmpleador);	
DESCRIPCIÓN	

OBJETO:	PropietarioEmpleador
MÉTODO:	deletePropietarioEmpleador
ARGUMENTO:	(PropietarioEmpleador propietarioEmpleador)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(proprietarioEmpleador);	
DESCRIPCIÓN	

OBJETO:	PropietarioEmpleador
MÉTODO:	getPropietarioEmpleadorByCodempAndEstado
ARGUMENTO:	(int codemp,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from PropietarioEmpleador where (empresa.codEmpresa="+codemp+")and (estado="+estado+") order by empresa.nombreRazonsocial asc"); if(lista.size(>0){ return (PropietarioEmpleador)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 193. Objeto PropietarioEmpleador

2.1.2.2.3.10.1.4.33 Objeto Provincia

OBJETO:	Provincia
MÉTODO:	getAllProvincia
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Provincia.class);	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	getProvincia
ARGUMENTO:	(int codProvincia)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Provincia) getHibernateTemplate().get(Provincia.class, codProvincia);	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	saveProvincia
ARGUMENTO:	(Provincia provincia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(provincia);	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	updateProvincia
ARGUMENTO:	(Provincia provincia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(provincia);	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	deleteProvincia
ARGUMENTO:	(Provincia provincia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(provincia);	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	getAllProvinciaByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Provincia where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	getAllProvinciaBuscarByNombre

ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Provincia where (upper(nombre) LIKE upper(""+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Provincia
MÉTODO:	getAllProvinciaBuscarByNombreAndCoddep
ARGUMENTO:	(String dato,int coddep)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Provincia where (departamento.codDepartamento="+coddep+"") and (upper(nombre) LIKE upper(""+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

Tabla 194. Objeto Provincia

2.1.2.2.3.10.1.4.34 Objeto Requisito

OBJETO:	Requisito
MÉTODO:	getAllRequisito
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Requisito.class);	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	getRequisito
ARGUMENTO:	(int codRequisito)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Requisito) getHibernateTemplate().get(Requisito.class, codRequisito);	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	saveRequisito
ARGUMENTO:	(Requisito requisito)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(requisito);	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	updateRequisito
ARGUMENTO:	(Requisito requisito)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(requisito);	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	deleteRequisito
ARGUMENTO:	(Requisito requisito)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(requisito);	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	getAllRequisitoByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Requisito where (estado="+estado+") order by requisito asc");	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	getAllRequisitoBuscarByRequisito
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Requisito where (upper(requisito) LIKE upper("'" + dato + "%')) order by requisito asc");	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	getAllRequisitoBuscarByRequisitoAndCodtipseg
ARGUMENTO:	(String dato,int codtipseg)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Requisito where (tipoSeguro.codTipoSeguro="+codtipseg+") and (upper(requisito) LIKE upper("'" + dato + "%')) order by requisito asc");	
DESCRIPCIÓN	

OBJETO:	Requisito
MÉTODO:	getAllRequisitoBuscarByCodtipsegAndEstado
ARGUMENTO:	(int codtipseg,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Requisito where (tipoSeguro.codTipoSeguro="+codtipseg+") and (estado="+estado+") order by requisito asc");	
DESCRIPCIÓN	

Tabla 195. Objeto Requisito

2.1.2.2.3.10.1.4.35 Objeto Rol

OBJETO:	Rol
MÉTODO:	getAllRol
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Rol.class);	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	getRol
ARGUMENTO:	(int codRol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Rol) getHibernateTemplate().get(Rol.class, codRol);	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	saveRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(rol);	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	updateRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(rol);	
DESCRIPCIÓN	

DESCRIPCIÓN

OBJETO:	Rol
MÉTODO:	deleteRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(rol);	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	getAllRolBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Rol where (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	getAllRolBuscarByNombreAndEstado
ARGUMENTO:	(String dato,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Rol where (estado="+estado+") and (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

Tabla 196. Objeto Rol

2.1.2.2.3.10.1.4.36

Objeto TelefonoEmpresa

OBJETO:	TelefonoEmpresa
MÉTODO:	getAllTelefonoEmpresa
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(TelefonoEmpresa.class);	
DESCRIPCIÓN	

OBJETO:	TelefonoEmpresa
MÉTODO:	getTelefonoEmpresa
ARGUMENTO:	(int codTelefonoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

return (TelefonoEmpresa) getHibernateTemplate().get(TelefonoEmpresa.class, codTelefonoEmpresa);
DESCRIPCIÓN

OBJETO:	TelefonoEmpresa
MÉTODO:	saveTelefonoEmpresa
ARGUMENTO:	(TelefonoEmpresa telefonoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(telefonoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TelefonoEmpresa
MÉTODO:	updateTelefonoEmpresa
ARGUMENTO:	(TelefonoEmpresa telefonoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(telefonoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TelefonoEmpresa
MÉTODO:	deleteTelefonoEmpresa
ARGUMENTO:	(TelefonoEmpresa telefonoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(telefonoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TelefonoEmpresa
MÉTODO:	getTelefonoEmpresaByCodempAndEstado
ARGUMENTO:	(int codemp,int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List lista=getHibernateTemplate().find("from TelefonoEmpresa where (empresa.codEmpresa="+codemp+")and (estado="+estado+"") order by numero asc"); if(lista.size(>0){ return (TelefonoEmpresa)lista.get(0); } return null;</pre>	
DESCRIPCIÓN	

Tabla 197. Objeto TelefonoEmpresa

2.1.2.2.3.10.1.4.37

Objeto TipoEmpresa

OBJETO:	TipoEmpresa
MÉTODO:	getAllTipoEmpresa
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(TipoEmpresa.class);	
DESCRIPCIÓN	

OBJETO:	TipoEmpresa
MÉTODO:	getTipoEmpresa
ARGUMENTO:	(int codTipoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (TipoEmpresa) getHibernateTemplate().get(TipoEmpresa.class, codTipoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TipoEmpresa
MÉTODO:	saveTipoEmpresa
ARGUMENTO:	(TipoEmpresa tipoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(tipoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TipoEmpresa
MÉTODO:	updateTipoEmpresa
ARGUMENTO:	(TipoEmpresa tipoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(tipoEmpresa);	
DESCRIPCIÓN	

OBJETO:	TipoEmpresa
MÉTODO:	deleteTipoEmpresa
ARGUMENTO:	(TipoEmpresa tipoEmpresa)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(tipoEmpresa);	
DESCRIPCIÓN	

--

OBJETO:	TipoEmpresa
MÉTODO:	getAllTipoEmpresaBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from TipoEmpresa where (estado="+estado+") order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	TipoEmpresa
MÉTODO:	getAllTipoEmpresaBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from TipoEmpresa where (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

Tabla 198. ObjetoTipoEmpresa

2.1.2.2.3.10.1.4.38 Objeto TipoSeguro

OBJETO:	TipoSeguro
MÉTODO:	getAllTipoSeguro
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(TipoSeguro.class);	
DESCRIPCIÓN	

OBJETO:	TipoSeguro
MÉTODO:	getTipoSeguro
ARGUMENTO:	(int codTipoSeguro)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (TipoSeguro) getHibernateTemplate().get(TipoSeguro.class, codTipoSeguro);	
DESCRIPCIÓN	

OBJETO:	TipoSeguro
MÉTODO:	saveTipoSeguro
ARGUMENTO:	(TipoSeguro tipoSeguro)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().save(tipoSeguro);
DESCRIPCIÓN

OBJETO:	TipoSeguro
MÉTODO:	updateTipoSeguro
ARGUMENTO:	(TipoSeguro tipoSeguro)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(tipoSeguro);	
DESCRIPCIÓN	

OBJETO:	TipoSeguro
MÉTODO:	deleteTipoSeguro
ARGUMENTO:	(TipoSeguro tipoSeguro)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(tipoSeguro);	
DESCRIPCIÓN	

OBJETO:	TipoSeguro
MÉTODO:	getAllTipoSeguroBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from TipoSeguro where (upper(nombre) LIKE upper('"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	TipoSeguro
MÉTODO:	getAllTipoSeguroByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from TipoSeguro where (estado="+estado+"") order by nombre asc");	
DESCRIPCIÓN	

Tabla 199. ObjetoTipoSeguro

2.1.2.2.3.10.1.4.39

Objeto Usuario

OBJETO:	Usuario
MÉTODO:	getAllUsuario
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Usuario.class);	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getUsuario
ARGUMENTO:	(int codUsuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Usuario) getHibernateTemplate().get(Usuario.class, codUsuario);	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	saveUsuario
ARGUMENTO:	(Usuario usuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(usuario);	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	updateUsuario
ARGUMENTO:	(Usuario usuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(usuario);	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	deleteUsuario
ARGUMENTO:	(Usuario usuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(usuario);	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getAllUsuarioBuscarByApAmNombre

ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Usuario where (upper(persona.apepat ' persona.apemat ' persona.nombre) LIKE upper('"+dato+"%')) order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getAllUsuarioByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Usuario where (estado="+estado+"") order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getUsuarioLogear
ARGUMENTO:	(String username,String password)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Usuario where (password='"+password+"") and (username='"+username+"") and (estado=1) and (persona.estado=1) order by persona.nombre asc"); if(lista.size(>0)){ return (Usuario)lista.get(0); } return null;	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getAllUsuarioByApAmNombreAndCodrol
ARGUMENTO:	(String dato,int codrol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Usuario where (rol.codRol="+codrol+"") and (upper(persona.apepat ' persona.apemat ' persona.nombre) LIKE upper('"+dato+"%')) order by persona.nombre asc");	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	getUsuarioByCodper
ARGUMENTO:	(int codper)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
List lista=getHibernateTemplate().find("from Usuario where (persona.codPersona="+codper+"") order by persona.nombre asc"); if(lista.size(>0){	

<pre> return (Usuario)lista.get(0); } return null; </pre>
DESCRIPCIÓN

Tabla 200. ObjetoUsuario

2.1.2.2.3.10.1.4.40

Objeto Zona

OBJETO:	Zona
MÉTODO:	getAllZona
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().loadAll(Zona.class);	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	getZona
ARGUMENTO:	(int codZona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return (Zona) getHibernateTemplate().get(Zona.class, codZona);	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	saveZona
ARGUMENTO:	(Zona zona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(zona);	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	updateZona
ARGUMENTO:	(Zona zona)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(zona);	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	deleteZona
ARGUMENTO:	(Zona zona)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(zona);	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	getAllZonaBuscarByNombre
ARGUMENTO:	(String dato)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Zona where (upper(nombre) LIKE upper '"+dato+"%')) order by nombre asc");	
DESCRIPCIÓN	

OBJETO:	Zona
MÉTODO:	getAllZonaBuscarByEstado
ARGUMENTO:	(int estado)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
return getHibernateTemplate().find("from Zona where (estado='"+estado+"') order by nombre asc");	
DESCRIPCIÓN	

Tabla 201. ObjetoZona

2.1.2.2.3.10.2 Modelado Entidad Relación

2.1.2.2.3.10.2.1 Introducción

Cuando se utiliza una base de datos para gestionar información, se está plasmando una parte del mundo real en una serie de tablas, registros y campos ubicados en un ordenador; creándose un modelo parcial de la realidad. Antes de crear físicamente estas tablas en el ordenador se debe realizar un modelo de datos.

Se suele cometer el error de ir creando nuevas tablas a medida que se van necesitando, haciendo así el modelo de datos y la construcción física de las tablas simultáneamente. El resultado de esto acaba siendo un sistema de información parcheado, con datos dispersos que terminan por no cumplir adecuadamente los requisitos necesarios.

2.1.2.2.3.10.2.2 Modelo Entidad / Relación

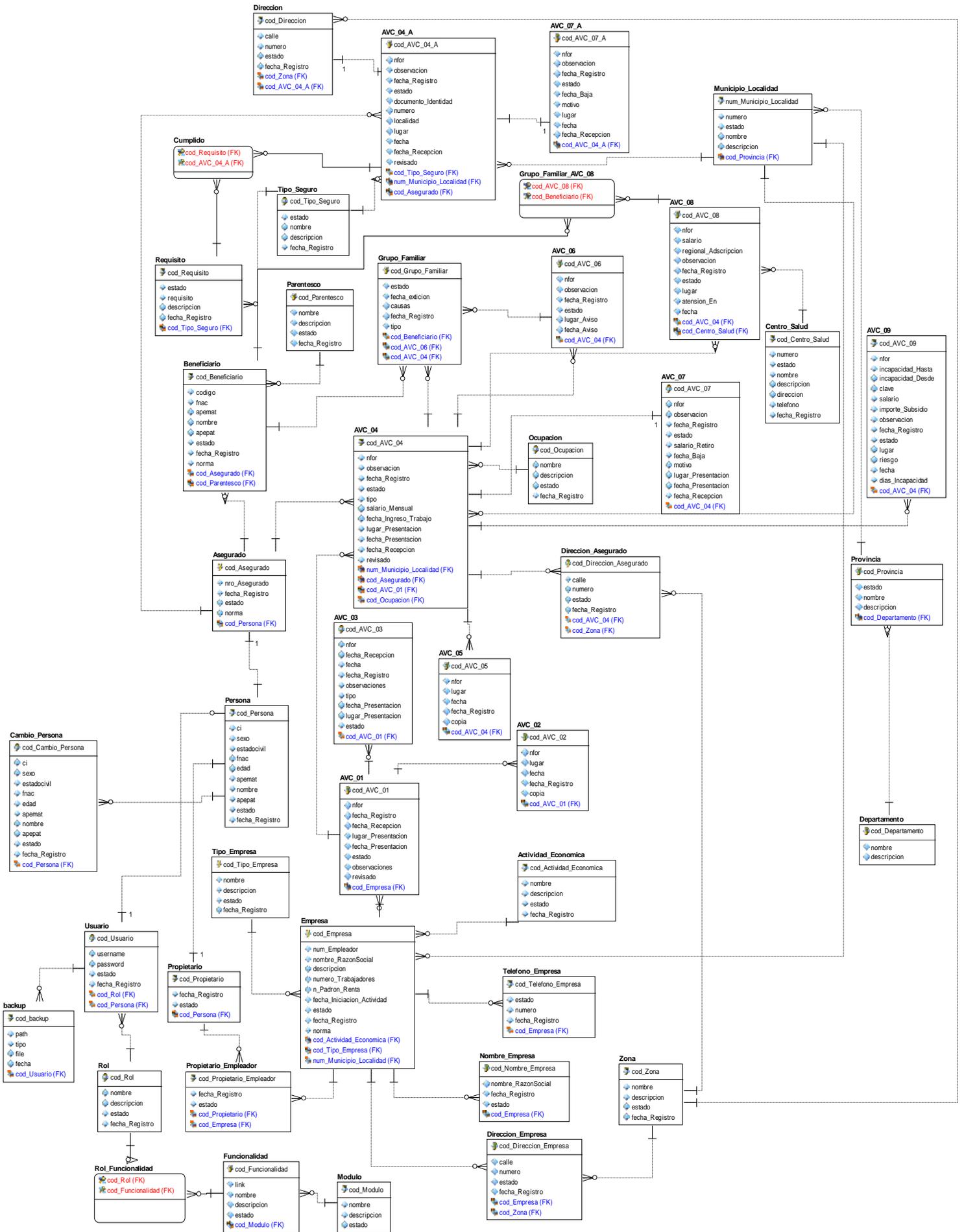


Figura 432. Modelo Entidad Relación

2.1.2.2.3.10.2.3 Creación de la Base de Datos (SQL)

```
--
-- TABLE: Actividad_Economica
--
CREATE TABLE Actividad_Economica (
  cod_Actividad_Economica integer NOT NULL,
  nombre varchar(100) NOT NULL,
  descripcion varchar(300),
  estado int4 DEFAULT 1 NOT NULL,
  fecha_Registro date NOT NULL,
  CONSTRAINT PK2 PRIMARY KEY ( cod_Actividad_Economica ));
--
-- TABLE: Asegurado
--
CREATE TABLE Asegurado (
  cod_Asegurado integer NOT NULL,
  nro_Asegurado varchar(25) NOT NULL,
  fecha_Registro date NOT NULL,
  estado int4 DEFAULT 1 NOT NULL,
  norma int4 DEFAULT 1 NOT NULL,
  cod_Persona int4 NOT NULL,
  CONSTRAINT PK21 PRIMARY KEY ( cod_Asegurado ));
--
-- TABLE: AVC_01
--
CREATE TABLE AVC_01 (
  cod_AVC_01 integer NOT NULL,
  nfor varchar(50),
  fecha_Registro date NOT NULL,
  fecha_Recepcion date NOT NULL,
  lugar_Presentacion varchar(100) NOT NULL,
  fecha_Presentacion date NOT NULL,
  estado int4 NOT NULL,
  observaciones varchar(150),
  revisado int4 DEFAULT 0 NOT NULL,
  cod_Empresa int4 NOT NULL,
  CONSTRAINT PK1 PRIMARY KEY ( cod_AVC_01 ));
--
-- TABLE: AVC_02
--
CREATE TABLE AVC_02 (
  cod_AVC_02 integer NOT NULL,
  nfor varchar(50),
  lugar varchar(100),
  fecha date,
  fecha_Registro date NOT NULL,
  copia int4 NOT NULL,
  cod_AVC_01 int4 NOT NULL,
  CONSTRAINT PK9 PRIMARY KEY ( cod_AVC_02 ));
--
-- TABLE: AVC_03
```

```

--
CREATE TABLE AVC_03 (
cod_AVC_03      integer      NOT NULL,
nfor           varchar(50),
fecha_Recepcion date      NOT NULL,
fecha         date,
fecha_Registro date      NOT NULL,
observaciones  varchar(150),
tipo          int4         NOT NULL,
fecha_Presentacion date,
lugar_Presentacion varchar(100),
estado        int4         NOT NULL,
cod_AVC_01    int4         NOT NULL,
CONSTRAINT PK12 PRIMARY KEY ( cod_AVC_03 ));
--
-- TABLE: AVC_04
--
CREATE TABLE AVC_04 (
cod_AVC_04      integer      NOT NULL,
nfor           varchar(50),
observacion     varchar(150),
fecha_Registro date      NOT NULL,
estado         int4         DEFAULT 1 NOT NULL,
tipo          int4         DEFAULT 1 NOT NULL,
salario_Mensual float4     NOT NULL,
fecha_Ingreso_Trabajo date  NOT NULL,
lugar_Presentacion varchar(50) NOT NULL,
fecha_Presentacion date  NOT NULL,
fecha_Recepcion date  NOT NULL,
revisado       int4         DEFAULT 0 NOT NULL,
num_Municipio_Localidad int4 NOT NULL,
cod_Asegurado  int4         NOT NULL,
cod_AVC_01    int4         NOT NULL,
cod_Ocupacion  int4         NOT NULL,
CONSTRAINT PK22 PRIMARY KEY ( cod_AVC_04 ));
--
-- TABLE: AVC_04_A
--
CREATE TABLE AVC_04_A (
cod_AVC_04_A    integer      NOT NULL,
nfor           varchar(50),
observacion     varchar(150),
fecha_Registro date      NOT NULL,
estado         int4         DEFAULT 1 NOT NULL,
documento_Identidad varchar(25) NOT NULL,
numero        int4         DEFAULT 1 NOT NULL,
localidad     varchar(50) NOT NULL,
lugar        varchar(50) NOT NULL,
fecha        date      NOT NULL,
fecha_Recepcion date  NOT NULL,
revisado       int4         DEFAULT 0 NOT NULL,
cod_Tipo_Seguro int4      NOT NULL,
num_Municipio_Localidad int4 NOT NULL,

```

```

        cod_Asegurado      int4      NOT NULL,
CONSTRAINT PK22_2 PRIMARY KEY ( cod_AVC_04_A ));
--
-- TABLE: AVC_05
--
CREATE TABLE AVC_05 (
cod_AVC_05      integer      NOT NULL,
nfor           varchar(50),
lugar         varchar(100),
fecha        date,
fecha_Registro date NOT NULL,
copia        int4      NOT NULL,
        cod_AVC_04      int4      NOT NULL,
CONSTRAINT PK9_1 PRIMARY KEY ( cod_AVC_05 ));
--
-- TABLE: AVC_06
--
CREATE TABLE AVC_06 (
cod_AVC_06      integer      NOT NULL,
nfor           varchar(50),
observacion   varchar(150),
fecha_Registro date NOT NULL,
estado        int4      DEFAULT 1 NOT NULL,
lugar_Aviso   varchar(50) NOT NULL,
fecha_Aviso   date NOT NULL,
        cod_AVC_04      int4      NOT NULL,
CONSTRAINT PK22_1_1 PRIMARY KEY ( cod_AVC_06 ));
--
-- TABLE: AVC_07
--
CREATE TABLE AVC_07 (
cod_AVC_07      integer      NOT NULL,
nfor           varchar(50),
observacion   varchar(150),
fecha_Registro date NOT NULL,
estado        int4      DEFAULT 1 NOT NULL,
        salario_Retiro  float4    DEFAULT 1 NOT NULL,
fecha_Baja    date NOT NULL,
motivo        varchar(150) NOT NULL,
lugar_Presentacion varchar(50) NOT NULL,
fecha_Presentacion date NOT NULL,
fecha_Recepcion date NOT NULL,
        cod_AVC_04      int4      NOT NULL,
CONSTRAINT PK22_1 PRIMARY KEY ( cod_AVC_07 ));
--
-- TABLE: AVC_07_A
--
CREATE TABLE AVC_07_A (
cod_AVC_07_A    integer      NOT NULL,
nfor           varchar(50),
observacion   varchar(150),
fecha_Registro date NOT NULL,
estado        int4      DEFAULT 1 NOT NULL,

```

```

fecha_Baja    date      NOT NULL,
motivo        varchar(150) NOT NULL,
lugar         varchar(50) NOT NULL,
fecha         date      NOT NULL,
fecha_Recepcion date    NOT NULL,
cod_AVC_04_A  int4      NOT NULL,
CONSTRAINT PK22_1_2 PRIMARY KEY ( cod_AVC_07_A ));
--
-- TABLE: AVC_08
--
CREATE TABLE AVC_08 (
cod_AVC_08    integer    NOT NULL,
nfor          varchar(50),
salario       float4     DEFAULT 0 NOT NULL,
regional_Adscpcion varchar(50),
observacion   varchar(150),
fecha_Registro date      NOT NULL,
estado        int4       DEFAULT 1 NOT NULL,
lugar         varchar(50) NOT NULL,
atension_En   varchar(150) NOT NULL,
fecha         date      NOT NULL,
cod_AVC_04    int4      NOT NULL,
cod_Centro_Salud int4    NOT NULL,
CONSTRAINT PK22_2_1 PRIMARY KEY ( cod_AVC_08 ));
--
-- TABLE: AVC_09
--
CREATE TABLE AVC_09 (
cod_AVC_09    integer    NOT NULL,
nfor          varchar(50),
incapacidad_Hasta date    NOT NULL,
incapacidad_Desde date    NOT NULL,
clave         varchar(50),
salario       float4     DEFAULT 0 NOT NULL,
importe_Subsidio float4   DEFAULT 0 NOT NULL,
observacion   varchar(150),
fecha_Registro date      NOT NULL,
estado        int4       DEFAULT 1 NOT NULL,
lugar         varchar(50) NOT NULL,
riesgo        varchar(50) NOT NULL,
fecha         date      NOT NULL,
dias_Incapacidad varchar(150) NOT NULL,
cod_AVC_04    int4      NOT NULL,
CONSTRAINT PK22_2_1_1 PRIMARY KEY ( cod_AVC_09 ));
--
-- TABLE: backup
--
CREATE TABLE backup(
cod_backup    integer    NOT NULL,
path          varchar(250) NOT NULL,
tipo          varchar(250) NOT NULL,
file          varchar(250),
fecha         date      NOT NULL,

```

```

    cod_Usuario int4      NOT NULL,
CONSTRAINT PK58 PRIMARY KEY (cod_backup));
--
-- TABLE: Beneficiario
--
CREATE TABLE Beneficiario (
cod_Beneficiario integer  NOT NULL,
codigo          int4,
fnac            date      NOT NULL,
apemat         varchar(50) NOT NULL,
nombre         varchar(70) NOT NULL,
apepat         varchar(50),
estado         int4       DEFAULT 1 NOT NULL,
fecha_Registro date      NOT NULL,
norma          int4       DEFAULT 1 NOT NULL,
    cod_Asegurado int4     NOT NULL,
    cod_Parentesco int4    NOT NULL,
CONSTRAINT PK17_2 PRIMARY KEY ( cod_Beneficiario ));
--
-- TABLE: Cambio_Persona
--
CREATE TABLE Cambio_Persona (
    cod_Cambio_Persona integer  NOT NULL,
ci          int4      NOT NULL,
sexo        varchar(10) NOT NULL,
estadocivil varchar(50) NOT NULL,
fnac        date      NOT NULL,
edad        int4      DEFAULT 0,
apemat      varchar(50) NOT NULL,
nombre      varchar(70) NOT NULL,
apepat      varchar(50),
estado      int4      DEFAULT 1 NOT NULL,
fecha_Registro date      NOT NULL,
    cod_Persona int4     NOT NULL,
CONSTRAINT PK17_1 PRIMARY KEY ( cod_Cambio_Persona ));
--
-- TABLE: Centro_Salud
--
CREATE TABLE Centro_Salud (
cod_Centro_Salud integer  NOT NULL,
numero          varchar(20),
estado          int4      DEFAULT 1 NOT NULL,
nombre          varchar(50) NOT NULL,
descripcion     varchar(300),
direccion       varchar(150),
telefono        varchar(15),
fecha_Registro date      NOT NULL,
CONSTRAINT PK31_2_1_1 PRIMARY KEY ( cod_Centro_Salud ));
--
-- TABLE: Cumplido
--
CREATE TABLE Cumplido (
cod_Requisito int4  NOT NULL,

```

```

    cod_AVC_04_A int4 NOT NULL,
CONSTRAINT PK41 PRIMARY KEY ( cod_Requisito , cod_AVC_04_A ));
--
-- TABLE: Departamento
--
CREATE TABLE Departamento (
    cod_Departamento integer NOT NULL,
nombre varchar(100) NOT NULL,
descripcion varchar(300),
CONSTRAINT PK3 PRIMARY KEY ( cod_Departamento ));
--
-- TABLE: Direccion
--
CREATE TABLE Direccion (
cod_Direccion integer NOT NULL,
calle varchar(300) NOT NULL,
numero varchar(10),
estado int4 DEFAULT 1,
fecha_Registro date NOT NULL,
cod_Zona int4 NOT NULL,
cod_AVC_04_A int4 NOT NULL,
CONSTRAINT PK7_1_1 PRIMARY KEY ( cod_Direccion ));
--
-- TABLE: Direccion_Asegurado
--
CREATE TABLE Direccion_Asegurado (
    cod_Direccion_Asegurado integer NOT NULL,
calle varchar(300) NOT NULL,
numero varchar(10),
estado int4 DEFAULT 1,
fecha_Registro date NOT NULL,
cod_AVC_04 int4 NOT NULL,
cod_Zona int4 NOT NULL,
CONSTRAINT PK7_1 PRIMARY KEY ( cod_Direccion_Asegurado ));
--
-- TABLE: Direccion_Empresa
--
CREATE TABLE Direccion_Empresa (
    cod_Direccion_Empresa integer NOT NULL,
calle varchar(300) NOT NULL,
numero varchar(10),
estado int4 DEFAULT 1,
fecha_Registro date NOT NULL,
cod_Empresa int4 NOT NULL,
cod_Zona int4 NOT NULL,
CONSTRAINT PK7 PRIMARY KEY ( cod_Direccion_Empresa ));
--
-- TABLE: Empresa
--
CREATE TABLE Empresa (
    cod_Empresa integer NOT NULL,
codemp varchar(30) NOT NULL,
nombre_RazonSocial varchar(100) NOT NULL,

```

```

descripcion          varchar(300),
numero_Trabajadores  int4      NOT NULL,
n_Padron_Renta       varchar(15),
fecha_Iniciacion_Actividad date    NOT NULL,
estado               int4      NOT NULL,
fecha_Registro       date    NOT NULL,
norma                int4      DEFAULT 1 NOT NULL,
cod_Actividad_Economica int4    NOT NULL,
cod_Tipo_Empresa     int4      NOT NULL,
num_Municipio_Localidad int4    NOT NULL,
CONSTRAINT PK11 PRIMARY KEY ( cod_Empresa ));
--
-- TABLE: Funcionalidad
--
CREATE TABLE Funcionalidad (
cod_Funcionalidad varchar(10) NOT NULL,
link              varchar(100) NOT NULL,
nombre           char(50) NOT NULL,
descripcion      varchar(250),
estado          int4      NOT NULL,
cod_Modulo      int4      NOT NULL,
CONSTRAINT PK48 PRIMARY KEY ( cod_Funcionalidad ));
--
-- TABLE: Grupo_Familiar
--
CREATE TABLE Grupo_Familiar (
cod_Grupo_Familiar integer NOT NULL,
estado             int4      DEFAULT 1 NOT NULL,
fecha_exticcion   date    NOT NULL,
causas            varchar(150),
fecha_Registro   date    NOT NULL,
tipo              int4,
cod_Beneficiario int4      NOT NULL,
cod_AVC_06       int4      NOT NULL,
cod_AVC_04       int4      NOT NULL,
CONSTRAINT PK28 PRIMARY KEY ( cod_Grupo_Familiar ));
--
-- TABLE: Grupo_Familiar_AVC_08
--
CREATE TABLE Grupo_Familiar_AVC_08 (
cod_AVC_08       int4 NOT NULL,
cod_Beneficiario int4 NOT NULL,
CONSTRAINT PK56 PRIMARY KEY ( cod_AVC_08 , cod_Beneficiario ));
--
-- TABLE: Modulo
--
CREATE TABLE Modulo (
cod_Modulo integer NOT NULL,
nombre     varchar(50) NOT NULL,
descripcion varchar(250),
estado    int4      DEFAULT 1 NOT NULL,
CONSTRAINT PK47 PRIMARY KEY ( cod_Modulo ));
--

```

```

-- TABLE: Municipio_Localidad
--
CREATE TABLE Municipio_Localidad (
  num_Municipio_Localidad integer NOT NULL,
  numero varchar(25) NOT NULL,
  estado int4 DEFAULT 1 NOT NULL,
  nombre varchar(50) NOT NULL,
  descripcion varchar(300),
  cod_Provincia int4 NOT NULL,
  CONSTRAINT PK31_2 PRIMARY KEY ( num_Municipio_Localidad ));
--
-- TABLE: Nombre_Empresa
--
CREATE TABLE Nombre_Empresa (
  cod_Nombre_Empresa integer NOT NULL,
  nombre_RazonSocial varchar(100) NOT NULL,
  fecha_Registro date NOT NULL,
  estado int4 NOT NULL,
  cod_Empresa int4 NOT NULL,
  CONSTRAINT PK8_1 PRIMARY KEY ( cod_Nombre_Empresa ));
--
-- TABLE: Ocupacion
--
CREATE TABLE Ocupacion (
  cod_Ocupacion integer NOT NULL,
  nombre varchar(50) NOT NULL,
  descripcion varchar(150),
  estado int4 DEFAULT 1 NOT NULL,
  fecha_Registro date NOT NULL,
  CONSTRAINT PK24 PRIMARY KEY ( cod_Ocupacion ));
--
-- TABLE: Parentesco
--
CREATE TABLE Parentesco (
  cod_Parentesco integer NOT NULL,
  nombre varchar(50) NOT NULL,
  descripcion varchar(150),
  estado int4 DEFAULT 1 NOT NULL,
  fecha_Registro date NOT NULL,
  CONSTRAINT PK24_1 PRIMARY KEY ( cod_Parentesco ));
--
-- TABLE: Persona
--
CREATE TABLE Persona (
  cod_Persona integer NOT NULL,
  ci int4 NOT NULL,
  sexo varchar(10) NOT NULL,
  estadocivil varchar(50) NOT NULL,
  fnac date NOT NULL,
  edad int4 DEFAULT 0,
  apemat varchar(50) NOT NULL,
  nombre varchar(70) NOT NULL,
  apepat varchar(50),

```

```

estado      int4      DEFAULT 1 NOT NULL,
fecha_Registro date    NOT NULL,
CONSTRAINT PK17 PRIMARY KEY ( cod_Persona ));
--
-- TABLE: Propietario
--
CREATE TABLE Propietario (
cod_Propietario integer NOT NULL,
fecha_Registro date    NOT NULL,
estado      int4      DEFAULT 1 NOT NULL,
cod_Persona int4      NOT NULL,
CONSTRAINT PK10 PRIMARY KEY ( cod_Propietario ));
--
-- TABLE: Propietario_Empleador
--
CREATE TABLE Propietario_Empleador (
cod_Propietario_Empleador integer NOT NULL,
fecha_Registro      date    NOT NULL,
estado      int4      DEFAULT 1,
cod_Propietario      int4      NOT NULL,
cod_Empresa      int4      NOT NULL,
CONSTRAINT PK18 PRIMARY KEY ( cod_Propietario_Empleador ));
--
-- TABLE: Provincia
--
CREATE TABLE Provincia (
cod_Provincia      integer NOT NULL,
estado      int4      DEFAULT 1 NOT NULL,
nombre      varchar(50) NOT NULL,
descripcion  varchar(300),
cod_Departamento int4      NOT NULL,
CONSTRAINT PK31_2_1 PRIMARY KEY ( cod_Provincia ));
--
-- TABLE: Requisito
--
CREATE TABLE Requisito (
cod_Requisito      integer NOT NULL,
estado      int4      DEFAULT 1 NOT NULL,
requisito      varchar(500) NOT NULL,
descripcion  varchar(300),
fecha_Registro date    NOT NULL,
cod_Tipo_Seguro int4      NOT NULL,
CONSTRAINT PK31_1 PRIMARY KEY ( cod_Requisito ));
--
-- TABLE: Rol
--
CREATE TABLE Rol (
cod_Rol      integer NOT NULL,
nombre      varchar(50) NOT NULL,
descripcion  varchar(250),
estado      int4      DEFAULT 1 NOT NULL,
fecha_Registro date    NOT NULL,
CONSTRAINT PK46 PRIMARY KEY ( cod_Rol ));

```

```

--
-- TABLE: Rol_Funcionalidad
--
CREATE TABLE Rol_Funcionalidad (
  cod_Rol      int4      NOT NULL,
  cod_Funcionalidad varchar(10) NOT NULL,
  CONSTRAINT PK50 PRIMARY KEY ( cod_Rol , cod_Funcionalidad ));
--
-- TABLE: Telefono_Empresa
--
CREATE TABLE Telefono_Empresa (
  cod_Telefono_Empresa integer NOT NULL,
  estado      int4      NOT NULL,
  numero      varchar(30) NOT NULL,
  fecha_Registro date    NOT NULL,
  cod_Empresa int4      NOT NULL,
  CONSTRAINT PK8 PRIMARY KEY ( cod_Telefono_Empresa ));
--
-- TABLE: Tipo_Empresa
--
CREATE TABLE Tipo_Empresa (
  cod_Tipo_Empresa integer NOT NULL,
  nombre      varchar(100) NOT NULL,
  descripcion  varchar(300),
  estado      int4      DEFAULT 1 NOT NULL,
  fecha_Registro date    NOT NULL,
  CONSTRAINT PK13 PRIMARY KEY ( cod_Tipo_Empresa ));
--
-- TABLE: Tipo_Seguro
--
CREATE TABLE Tipo_Seguro (
  cod_Tipo_Seguro integer NOT NULL,
  estado      int4      DEFAULT 1 NOT NULL,
  nombre      varchar(50) NOT NULL,
  descripcion  varchar(300),
  fecha_Registro date    NOT NULL,
  CONSTRAINT PK31 PRIMARY KEY ( cod_Tipo_Seguro ));
--
-- TABLE: Usuario
--
CREATE TABLE Usuario (
  cod_Usuario integer NOT NULL,
  username    varchar(50) NOT NULL,
  password    varchar(50) NOT NULL,
  estado      int4      DEFAULT 1 NOT NULL,
  fecha_Registro date    NOT NULL,
  cod_Rol     int4      NOT NULL,
  cod_Persona int4,
  CONSTRAINT PK45 PRIMARY KEY ( cod_Usuario ));
--
-- TABLE: Zona
--
CREATE TABLE Zona (

```

```

    cod_Zona    integer    NOT NULL,
nombre        varchar(100) NOT NULL,
descripcion   varchar(300),
estado        int4        DEFAULT 1 NOT NULL,
fecha_Registro date      NOT NULL,
CONSTRAINT PK6 PRIMARY KEY ( cod_Zona );
--
-- TABLE: Asegurado
--
ALTER TABLE Asegurado ADD CONSTRAINT RefPersona312
    FOREIGN KEY ( cod_Persona )
REFERENCES Persona ( cod_Persona );
--
-- TABLE: AVC_01
--
ALTER TABLE AVC_01 ADD CONSTRAINT RefEmpresa832
    FOREIGN KEY ( cod_Empresa )
REFERENCES Empresa ( cod_Empresa );
--
-- TABLE: AVC_02
--
ALTER TABLE AVC_02 ADD CONSTRAINT RefAVC_01202
    FOREIGN KEY ( cod_AVC_01 )
REFERENCES AVC_01 ( cod_AVC_01 );
--
-- TABLE: AVC_03
--
ALTER TABLE AVC_03 ADD CONSTRAINT RefAVC_01252
    FOREIGN KEY ( cod_AVC_01 )
REFERENCES AVC_01 ( cod_AVC_01 );
--
-- TABLE: AVC_04
--
ALTER TABLE AVC_04 ADD CONSTRAINT RefAsegurado352
    FOREIGN KEY ( cod_Asegurado )
REFERENCES Asegurado ( cod_Asegurado );
ALTER TABLE AVC_04 ADD CONSTRAINT RefOcupacion362
    FOREIGN KEY ( cod_Ocupacion )
REFERENCES Ocupacion ( cod_Ocupacion );
ALTER TABLE AVC_04 ADD CONSTRAINT RefMunicipio_Localidad572
    FOREIGN KEY ( num_Municipio_Localidad )
REFERENCES Municipio_Localidad ( num_Municipio_Localidad );
ALTER TABLE AVC_04 ADD CONSTRAINT RefAVC_01852
    FOREIGN KEY ( cod_AVC_01 )
REFERENCES AVC_01 ( cod_AVC_01 );
--
-- TABLE: AVC_04_A
--
ALTER TABLE AVC_04_A ADD CONSTRAINT RefAsegurado542
    FOREIGN KEY ( cod_Asegurado )
REFERENCES Asegurado ( cod_Asegurado );
ALTER TABLE AVC_04_A ADD CONSTRAINT RefTipo_Seguro762
    FOREIGN KEY ( cod_Tipo_Seguro )

```

```

REFERENCES Tipo_Seguro ( cod_Tipo_Seguro );
ALTER TABLE AVC_04_A ADD CONSTRAINT RefMunicipio_Localidad792
  FOREIGN KEY ( num_Municipio_Localidad )
  REFERENCES Municipio_Localidad ( num_Municipio_Localidad );
--
-- TABLE: AVC_05
--
ALTER TABLE AVC_05 ADD CONSTRAINT RefAVC_04332
  FOREIGN KEY ( cod_AVC_04 )
  REFERENCES AVC_04 ( cod_AVC_04 );
--
-- TABLE: AVC_06
--
ALTER TABLE AVC_06 ADD CONSTRAINT RefAVC_04882
  FOREIGN KEY ( cod_AVC_04 )
  REFERENCES AVC_04 ( cod_AVC_04 );
--
-- TABLE: AVC_07
--
ALTER TABLE AVC_07 ADD CONSTRAINT RefAVC_04382
  FOREIGN KEY ( cod_AVC_04 )
  REFERENCES AVC_04 ( cod_AVC_04 );
--
-- TABLE: AVC_07_A
--
ALTER TABLE AVC_07_A ADD CONSTRAINT RefAVC_04_A552
  FOREIGN KEY ( cod_AVC_04_A )
  REFERENCES AVC_04_A ( cod_AVC_04_A );
--
-- TABLE: AVC_08
--
ALTER TABLE AVC_08 ADD CONSTRAINT RefAVC_04492
  FOREIGN KEY ( cod_AVC_04 )
  REFERENCES AVC_04 ( cod_AVC_04 );
ALTER TABLE AVC_08 ADD CONSTRAINT RefCentro_Salud512
  FOREIGN KEY ( cod_Centro_Salud )
  REFERENCES Centro_Salud ( cod_Centro_Salud );
--
-- TABLE: AVC_09
--
ALTER TABLE AVC_09 ADD CONSTRAINT RefAVC_04502
  FOREIGN KEY ( cod_AVC_04 )
  REFERENCES AVC_04 ( cod_AVC_04 );
--
-- TABLE: backup
--
ALTER TABLE backup ADD CONSTRAINT RefUsuario1032
  FOREIGN KEY ( cod_Usuario )
  REFERENCES Usuario ( cod_Usuario );
--
-- TABLE: Beneficiario
--
ALTER TABLE Beneficiario ADD CONSTRAINT RefAsegurado972

```

```

FOREIGN KEY ( cod_Asegurado )
REFERENCES Asegurado ( cod_Asegurado );
ALTER TABLE Beneficiario ADD CONSTRAINT RefParentesco982
FOREIGN KEY ( cod_Parentesco )
REFERENCES Parentesco ( cod_Parentesco );
--
-- TABLE: Cambio_Persona
--
ALTER TABLE Cambio_Persona ADD CONSTRAINT RefPersona562
FOREIGN KEY ( cod_Persona )
REFERENCES Persona ( cod_Persona );
--
-- TABLE: Cumplido
--
ALTER TABLE Cumplido ADD CONSTRAINT RefRequisito422
FOREIGN KEY ( cod_Requisito )
REFERENCES Requisito ( cod_Requisito );
ALTER TABLE Cumplido ADD CONSTRAINT RefAVC_04_A772
FOREIGN KEY ( cod_AVC_04_A )
REFERENCES AVC_04_A ( cod_AVC_04_A );
--
-- TABLE: Direccion
--
ALTER TABLE Direccion ADD CONSTRAINT RefZona802
FOREIGN KEY ( cod_Zona )
REFERENCES Zona ( cod_Zona );
ALTER TABLE Direccion ADD CONSTRAINT RefAVC_04_A812
FOREIGN KEY ( cod_AVC_04_A )
REFERENCES AVC_04_A ( cod_AVC_04_A );
--
-- TABLE: Direccion_Asegurado
--
ALTER TABLE Direccion_Asegurado ADD CONSTRAINT RefAVC_04392
FOREIGN KEY ( cod_AVC_04 )
REFERENCES AVC_04 ( cod_AVC_04 );
ALTER TABLE Direccion_Asegurado ADD CONSTRAINT RefZona402
FOREIGN KEY ( cod_Zona )
REFERENCES Zona ( cod_Zona );
--
-- TABLE: Direccion_Empresa
--
ALTER TABLE Direccion_Empresa ADD CONSTRAINT RefZona142
FOREIGN KEY ( cod_Zona )
REFERENCES Zona ( cod_Zona );
ALTER TABLE Direccion_Empresa ADD CONSTRAINT RefEmpresa292
FOREIGN KEY ( cod_Empresa )
REFERENCES Empresa ( cod_Empresa );
--
-- TABLE: Empresa
--
ALTER TABLE Empresa ADD CONSTRAINT RefActividad_Economica32
FOREIGN KEY ( cod_Actividad_Economica )
REFERENCES Actividad_Economica ( cod_Actividad_Economica );

```

```

ALTER TABLE Empresa ADD CONSTRAINT RefTipo_Empresa42
FOREIGN KEY ( cod_Tipo_Empresa )
REFERENCES Tipo_Empresa ( cod_Tipo_Empresa );
ALTER TABLE Empresa ADD CONSTRAINT RefMunicipio_Localidad592
FOREIGN KEY ( num_Municipio_Localidad )
REFERENCES Municipio_Localidad ( num_Municipio_Localidad );
--
-- TABLE: Funcionalidad
--
ALTER TABLE Funcionalidad ADD CONSTRAINT RefModulo602
FOREIGN KEY ( cod_Modulo )
REFERENCES Modulo ( cod_Modulo );
--
-- TABLE: Grupo_Familiar
--
ALTER TABLE Grupo_Familiar ADD CONSTRAINT RefAVC_04872
FOREIGN KEY ( cod_AVC_04 )
REFERENCES AVC_04 ( cod_AVC_04 );
ALTER TABLE Grupo_Familiar ADD CONSTRAINT RefBeneficiario942
FOREIGN KEY ( cod_Beneficiario )
REFERENCES Beneficiario ( cod_Beneficiario );
ALTER TABLE Grupo_Familiar ADD CONSTRAINT RefAVC_06992
FOREIGN KEY ( cod_AVC_06 )
REFERENCES AVC_06 ( cod_AVC_06 );
--
-- TABLE: Grupo_Familiar_AVC_08
--
ALTER TABLE Grupo_Familiar_AVC_08 ADD CONSTRAINT RefAVC_08922
FOREIGN KEY ( cod_AVC_08 )
REFERENCES AVC_08 ( cod_AVC_08 );
ALTER TABLE Grupo_Familiar_AVC_08 ADD CONSTRAINT RefBeneficiario1012
FOREIGN KEY ( cod_Beneficiario )
REFERENCES Beneficiario ( cod_Beneficiario );
--
-- TABLE: Municipio_Localidad
--
ALTER TABLE Municipio_Localidad ADD CONSTRAINT RefProvincia522
FOREIGN KEY ( cod_Provincia )
REFERENCES Provincia ( cod_Provincia );
--
-- TABLE: Nombre_Empresa
--
ALTER TABLE Nombre_Empresa ADD CONSTRAINT RefEmpresa862
FOREIGN KEY ( cod_Empresa )
REFERENCES Empresa ( cod_Empresa );
--
-- TABLE: Propietario
--
ALTER TABLE Propietario ADD CONSTRAINT RefPersona152
FOREIGN KEY ( cod_Persona )
REFERENCES Persona ( cod_Persona );
--
-- TABLE: Propietario_Empleador

```

```

--
ALTER TABLE Propietario_Empleador ADD CONSTRAINT RefPropietario222
  FOREIGN KEY ( cod_Propietario )
  REFERENCES Propietario ( cod_Propietario );
ALTER TABLE Propietario_Empleador ADD CONSTRAINT RefEmpresa232
  FOREIGN KEY ( cod_Empresa )
  REFERENCES Empresa ( cod_Empresa );
--
-- TABLE: Provincia
--
ALTER TABLE Provincia ADD CONSTRAINT RefDepartamento532
  FOREIGN KEY ( cod_Departamento )
  REFERENCES Departamento ( cod_Departamento );
--
-- TABLE: Requisito
--
ALTER TABLE Requisito ADD CONSTRAINT RefTipo_Seguro412
  FOREIGN KEY ( cod_Tipo_Seguro )
  REFERENCES Tipo_Seguro ( cod_Tipo_Seguro );
--
-- TABLE: Rol_Funcionalidad
--
ALTER TABLE Rol_Funcionalidad ADD CONSTRAINT RefRol642
  FOREIGN KEY ( cod_Rol )
  REFERENCES Rol ( cod_Rol );
ALTER TABLE Rol_Funcionalidad ADD CONSTRAINT RefFuncionalidad652
  FOREIGN KEY ( cod_Funcionalidad )
  REFERENCES Funcionalidad ( cod_Funcionalidad );
--
-- TABLE: Telefono_Empresa
--
ALTER TABLE Telefono_Empresa ADD CONSTRAINT RefEmpresa82
  FOREIGN KEY ( cod_Empresa )
  REFERENCES Empresa ( cod_Empresa );
--
-- TABLE: Usuario
--
ALTER TABLE Usuario ADD CONSTRAINT RefPersona672
  FOREIGN KEY ( cod_Persona )
  REFERENCES Persona ( cod_Persona );
ALTER TABLE Usuario ADD CONSTRAINT RefRol752
  FOREIGN KEY ( cod_Rol )
  REFERENCES Rol ( cod_Rol );

```

2.1.2.2.3.10.2.4 Diccionario de Datos

	Actividad_Economica
Nombre Entidad	Actividad_Economica
Tipo Entidad	Independent
Primary Keys	cod_Actividad_Economica
Definicion	actividad económica que tiene una empresa
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK2	PrimaryKey	cod_Actividad_Economica

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Actividad_Economica	SERIAL/INTEGER	ID	Representa la clave primaria de la Actividad Económica que será generada por la base de datos
nombre	VARCHAR(100)	N	Representa nombre de la Actividad Económica
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene la Actividad Económica
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registro una dupla

Nombre Entidad	Asegurado
Tipo Entidad	Independent
Primary Keys	cod_Asegurado
Definicion	persona que es un beneficiario por parte de una empresa
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK21	PrimaryKey	cod_Asegurado

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Asegurado	SERIAL/INTEGER	ID	
nro_Asegurado	VARCHAR(25)	N	Representa la clave primaria de Asegurado que será asignado
fecha_Registro	DATE	N	Representa el día mes año en que se registro una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
norma	INTEGER	N	Representa si esta cumplió cumpliendo normativas 1: alta por cumplimiento 0: baja por incumplimiento
cod_Persona	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Persona	Non-Identifying	One to Exactly: 1	cod_Persona	NO

Nombre Entidad	AVC_01
Tipo Entidad	Independent
Primary Keys	cod_AVC_01
Definicion	Aviso de Afiliación del Empleador
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK1	PrimaryKey	cod_AVC_01

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_01	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_01 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al

fecha_Registro	DATE	N	realizar registros
fecha_Recepcion	DATE	N	Representa el día mes año en que se registro una dupla
lugar_Presentacion	VARCHAR(100)	N	Representa el día mes año de la recepción del formulario
fecha_Presentacion	DATE	N	Representa el lugar de la presentación del formulario
estado	INTEGER	N	Representa el día mes año de la presentación del formulario
observaciones	VARCHAR(150)	Y	Representa el eliminado lógico 0: inhabilitado 1: habilitado
revisado	INTEGER	N	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
cod_Empresa	INTEGER	N	Representa si cumplió todos los requisitos 1: cumplido 0: no cumplido
		N	Representa la clave primaria de Empresa que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Empresa	Non-Identifying	One To Zero or More	cod_Empresa	NO
Nombre Entidad	AVC_02			
Tipo Entidad	Independent			
Primary Keys	cod_AVC_02			
Definicion	Carnet de Empleador			
Notas				

Keys

Nombre Key	Tipo Key	Keys
PK9	PrimaryKey	cod_AVC_02

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_02	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_02 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
lugar	VARCHAR(100)	Y	Representa el lugar donde se lleno el registro
fecha	DATE	Y	Representa el día mes año de emisión
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
copia	INTEGER	N	Representa si es una copia o original el carné 1: original - 0: copia
cod_AVC_01	INTEGER	N	Representa la clave primaria del AVC_01 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_01	Non-Identifying	One To Zero or More	cod_AVC_01	NO
Nombre Entidad	AVC_03			
Tipo Entidad	Independent			
Primary Keys	cod_AVC_03			
Definicion	Aviso de Novedades del Empleador			
Notas	tipo			
	0	temporal		baja
	1	definitiva		baja

2
3 ninguno

altas

Keys

Nombre Key	Tipo Key	Keys
PK12	PrimaryKey	cod_AVC_03

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_03	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_03 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
fecha_Recepcion	DATE	N	Representa el día mes año de la recepción del formulario
fecha	DATE	Y	Representa el día mes año de emisión
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
observaciones	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
tipo	INTEGER	N	Representa 0: temporal baja 1: definitiva baja 2: altas 3: ninguno modificación de datos
fecha_Presentacion	DATE	Y	Representa el día mes año de la presentación del formulario
lugar_Presentacion	VARCHAR(100)	Y	Representa el lugar de la presentación del formulario
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
cod_AVC_01	INTEGER	N	Representa la clave primaria del AVC_01 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_01	Non-Identifying	One To Zero or More	cod_AVC_01	NO
Nombre Entidad	AVC_04	AVC_04		
Tipo Entidad	Independent			
Primary Keys	cod_AVC_04			
Definicion	Aviso de Afiliación y Reingreso del Trabajador			
Notas	tipo 0 1 reingreso			afiliación

Keys

Nombre Key	Tipo Key	Keys
PK22	PrimaryKey	cod_AVC_04

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_04	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_04 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registro una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado 2: que fue habilitado y luego rehabilitado

tipo	INTEGER	N	Representa 0: afiliación 1: reingreso
salario_Mensual	FLOAT	N	Representa el salario mensual que gana el asegurado
fecha_Ingreso_Tra bajo	DATE	N	Representa el día mes año de ingreso a su fuente de trabajo
lugar_Presentacion	VARCHAR(50)	N	Representa el lugar de la presentación del formulario
fecha_Presentacion	DATE	N	Representa el día mes año de la presentación del formulario
fecha_Recepcion	DATE	N	Representa el día mes año de la recepción del formulario
revisado	INTEGER	N	Representa si cumplio todos los requisitos 1: cumplido 0: no cumplido
num_Municipio_L ocalidad	INTEGER	N	Representa la clave primaria de Municipio_Localidad que será generada por la base de datos
cod_Asegurado	INTEGER	N	
cod_AVC_01	INTEGER	N	Representa la clave primaria del AVC_01 que será generada por la base de datos
cod_Ocupacion	INTEGER	N	Representa la clave primaria de Ocupación que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_01	Non-Identifying	One To Zero or More	cod_AVC_01	NO
Asegurado	Non-Identifying	One To Zero or More	cod_Asegurado	NO
Ocupacion	Non-Identifying	One To Zero or More	cod_Ocupacion	NO
Municipio_Localidad	Non-Identifying	One To Zero or More	num_Municipio _Localidad	NO

AVC_04_A

Nombre Entidad	AVC_04_A
Tipo Entidad	Independent
Primary Keys	cod_AVC_04_A
Definicion	Seguro de Salud para el Adulto Mayor
Notas	tipo 0 1 reingreso
	afiliación

Keys

Nombre Key	Tipo Key	Keys
PK22_2	PrimaryKey	cod_AVC_04_A

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_04_A	SERIAL/INTE GER	ID	
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
observacion	VARCHAR(15 0)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
documento_Identid ad	VARCHAR(25)	N	
numero	INTEGER	N	se asignara el numero de aseguro que es
localidad	VARCHAR(50)	N	
lugar	VARCHAR(50)	N	Representa el lugar donde se lleno el registro
fecha	DATE	N	Representa el día mes año de emisión

fecha_Recepcion	DATE	N	Representa el día mes año de la recepción del formulario
revisado	INTEGER	N	Representa si cumplió todos los requisitos 1: cumplido 0: no cumplido
cod_Tipo_Seguro	INTEGER	N	Representa la clave primaria de Tipo_Seguro que será generada por la base de datos
num_Municipio_Localidad	INTEGER	N	Representa la clave primaria de Municipio_Localidad que será generada por la base de datos
cod_Asegurado	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Tipo_Seguro	Non-Identifying	One To Zero or More	cod_Tipo_Seguro	NO
Municipio_Localidad	Non-Identifying	One To Zero or More	num_Municipio_Localidad	NO
Asegurado	Non-Identifying	One To Zero or More	cod_Asegurado	NO

Nombre Entidad AVC_05
Tipo Entidad Independent
Primary Keys cod_AVC_05
Definicion Carnet de Asegurado
Notas

Keys

Nombre Key	Tipo Key	Keys
PK9_1	PrimaryKey	cod_AVC_05

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_AVC_05	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_05 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
lugar	VARCHAR(100)	Y	Representa el lugar donde se llenó el registro
fecha	DATE	Y	Representa el día mes año de emisión
fecha_Registro copia	DATE	N	Representa el día mes año en que se registro una dupla
	INTEGER	N	Representa si es una copia o original el carné 1: original - 0: copia
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AVC_04	NO

Nombre Entidad AVC_06
Tipo Entidad Independent
Primary Keys cod_AVC_06
Definicion Aviso de Altas y Bajas de Beneficiarios
Notas
0
1 alta

baja

Keys

Nombre Key	Tipo Key	Keys
PK22_1_1	PrimaryKey	cod_AVC_06

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_06	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_06 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
lugar_Aviso	VARCHAR(50)	N	Representa el lugar de aviso donde se afilió a un nuevo beneficiario
fecha_Aviso	DATE	N	Representa el día mes año de aviso de altas y bajas
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AVC_04	NO
Nombre Entidad	AVC_07			
Tipo Entidad	Independent			
Primary Keys	cod_AVC_07			
Definicion	Aviso Baja del Asegurado			
Notas	tipo 0 1 reingreso			afiliación

Keys

Nombre Key	Tipo Key	Keys
PK22_1	PrimaryKey	cod_AVC_07

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_07	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_07 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el numero de Formulario que es único al realizar registros
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
salario_Retiro	FLOAT	N	Representa el salario que gana el asegurado cuando se da de baja del seguro
fecha_Baja	DATE	N	Representa el día mes año de baja
motivo	VARCHAR(150)	N	Representa el motivo de la baja
lugar_Presentacion	VARCHAR(50)	N	Representa el lugar de la presentación del formulario
fecha_Presentacion	DATE	N	Representa el día mes año de la presentación del formulario

fecha_Recepcion	DATE	N	Representa el día mes año de la recepción del formulario
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One to Exactly: 1	cod_AVC_04	NO
AVC_07_A				
Nombre Entidad	AVC_07_A			
Tipo Entidad	Independent			
Primary Keys	cod_AVC_07_A			
Definicion	Aviso de Baja del Asegurado para el Adulto Mayor			
Notas	tipo 0 1 reingreso			afiliación

Keys

Nombre Key	Tipo Key	Keys
PK22_1_2	PrimaryKey	cod_AVC_07_A

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_AVC_07_A	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_07 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Baja	DATE	N	Representa el día mes año de baja
motivo	VARCHAR(150)	N	Representa el motivo de la baja
lugar	VARCHAR(50)	N	Representa el lugar donde se llenó el registro
fecha	DATE	N	Representa el día mes año de emisión
fecha_Recepcion	DATE	N	Representa el día mes año de la recepción del formulario
cod_AVC_04_A	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04_A	Non-Identifying	One to Exactly: 1	cod_AVC_04_A	NO
AVC_08				
Nombre Entidad	AVC_08			
Tipo Entidad	Independent			
Primary Keys	cod_AVC_08			
Definicion	Autorización para Recibir Atención Médica en regional Distinta a la de Adscripción			
Notas	tipo 0 1 reingreso			afiliación

Keys

Nombre Key	Tipo Key	Keys
PK22_2_1	PrimaryKey	cod_AVC_08

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_AVC_08	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_08 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
salario	FLOAT	N	Representa el salario que gana el asegurado
regional_Adscripcion	VARCHAR(50)	Y	Representa el lugar de origen de la que autoriza para recibir atención en otro lugar
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
lugar	VARCHAR(50)	N	Representa el lugar donde se lleno el registro
atension_En	VARCHAR(150)	N	
fecha	DATE	N	Representa el día mes año de emisión
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos
cod_Centro_Salud	INTEGER	N	Representa la clave primaria del Centro_Salud que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AVC_04	NO
Centro_Salud	Non-Identifying	One To Zero or More	cod_Centro_Salud	NO

AVC_09

Nombre Entidad	AVC_09
Tipo Entidad	Independent
Primary Keys	cod_AVC_09
Definicion	Certificado de Incapacidad Temporal
Notas	riesgo
	1
	2
	3 maternidad
	profesional enfermedad

Keys

Nombre Key	Tipo Key	Keys
PK22_2_1_1	PrimaryKey	cod_AVC_09

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_AVC_09	SERIAL/INTEGER	ID	Representa la clave primaria del AVC_09 que será generada por la base de datos
nfor	VARCHAR(50)	Y	Representa el número de Formulario que es único al realizar registros
incapacidad_Hasta	DATE	N	Representa el día mes año de fin de su incapacidad
incapacidad_Desde	DATE	N	Representa el día mes año de inicio de su incapacidad
clave	VARCHAR(50)	Y	
salario	FLOAT	N	Representa el salario que gana el asegurado

importe_Subsidio	FLOAT	N	Representa el importe de su subsidio
observacion	VARCHAR(150)	Y	Representa las observaciones que se dieron o situaciones dadas que deben ser registradas
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
lugar	VARCHAR(50)	N	Representa el lugar donde se lleno el registro
riesgo	VARCHAR(50)	N	Representa el riesgo por el cual esta solicitando su certificado de incapacidad
fecha	DATE	N	Representa el día mes año de emisión
dias_Incapacidad	VARCHAR(150)	N	Representa los días de incapacidad para trabajar
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AVC_04	NO
Nombre Entidad	backup			
Tipo Entidad	Independent			
Primary Keys	cod_backup			
Definicion				
Notas				

Keys

Nombre Key	Tipo Key	Keys
PK58	PrimaryKey	cod_backup

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_backup	SERIAL/INTEGER	ID	Representa la clave primaria de Backup que será generada por la base de datos
path	VARCHAR(250)	N	Representa la ruta donde se guardo el Resguardo o Restauración de un archivo Backup
tipo	VARCHAR(250)	N	Representa si es de tipo Resguardo o Restauración de un Backup
file	VARCHAR(250)	Y	Representa el nombre del archivo Backup
fecha	DATE	N	Representa el día mes año en que se registró una dupla o Backup
cod_Usuario	INTEGER	N	Representa la clave primaria de Usuario que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Usuario	Non-Identifying	One To Zero or More	cod_Usuario	NO
Nombre Entidad	Beneficiario			
Tipo Entidad	Independent			
Primary Keys	cod_Beneficiario			
Definicion	cualquier persona jurídica o natural que se relaciona con la Caja Nacional de Salud			

Notas

Keys

Nombre Key	Tipo Key	Keys
------------	----------	------

PK17_2 PrimaryKey cod_Beneficiario

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Beneficiario	SERIAL/INTEGER	ID	
codigo	INTEGER	Y	
fnac	DATE	N	Representa el día mes año del nacimiento de una persona
apemat	VARCHAR(50)	N	
nombre	VARCHAR(70)	N	Representa nombre de la persona
apepat	VARCHAR(50)	Y	
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
norma	INTEGER	N	Representa si esta cumplió cumpliendo normativas 1: alta por cumplimiento 0: baja por incumplimiento
cod_Asegurado	INTEGER	N	
cod_Parentesco	INTEGER	N	Representa la clave primaria de Parentesco que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Asegurado	Non-Identifying	One To Zero or More	cod_Asegurado	NO
Parentesco	Non-Identifying	One To Zero or More	cod_Parentesco	NO

Cambio_Persona

Nombre Entidad Cambio_Persona
Tipo Entidad Independent
Primary Keys cod_Cambio_Persona
Definicion persona en la que se registra la modificacion de sus datos

Notas

Keys

Nombre Key	Tipo Key	Keys
PK17_1	PrimaryKey	cod_Cambio_Persona

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Cambio_Persona	SERIAL/INTEGER	ID	Representa la clave primaria de Cambio_Persona que será generada por la base de datos
ci	INTEGER	N	
sexo	VARCHAR(10)	N	Representa de la persona M: masculino F: femenino
estadocivil	VARCHAR(50)	N	
fnac	DATE	N	Representa el día mes año del nacimiento de una persona
edad	INTEGER	Y	Representa la edad en la que se registró en el seguro
apemat	VARCHAR(50)	N	
nombre	VARCHAR(70)	N	Representa nombre de la persona
apepat	VARCHAR(50)	Y	
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
cod_Persona	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Persona	Non-Identifying	One To Zero or More	cod_Persona	NO

Centro_Salud

Nombre Entidad Centro_Salud
Tipo Entidad Independent
Primary Keys cod_Centro_Salud
Definicion centro de salud como ser entidades hospitalarias
Notas

Keys

Nombre Key	Tipo Key	Keys
PK31_2_1_1	PrimaryKey	cod_Centro_Salud

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Centro_Salud	SERIAL/INTEGER	ID	Representa la clave primaria del Centro_Salud que será generada por la base de datos
numero	VARCHAR(20)	Y	Representa el numero de centro de salud de acuerdo al código vigente
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
nombre	VARCHAR(50)	N	Representa nombre del centro de salud
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene el centro de salud
direccion	VARCHAR(150)	Y	Representa la dirección o ubicación del centro de salud
telefono	VARCHAR(15)	Y	Representa en número de teléfono del centro de salud
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla

Cumplido

Nombre Entidad Cumplido
Tipo Entidad Dependent
Primary Keys cod_Requisito,cod_AVC_04_A
Definicion cumplimeitno de los requisitos para optar por algún tipo de seguro
Notas

Keys

Nombre Key	Tipo Key	Keys
PK41	PrimaryKey	cod_Requisito, cod_AVC_04_A

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Requisito	INTEGER	N	Representa la clave primaria de Requisitos que será generada por la base de datos
cod_AVC_04_A	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04_A	Identifying	One To Zero or More	cod_AVC_04_A	NO
Requisito	Identifying	One To Zero or More	cod_Requisito	NO

Departamento

Nombre Entidad Departamento
Tipo Entidad Independent
Primary Keys cod_Departamento
Definicion departamentos de Bolivia
Notas

Keys

Nombre Key	Tipo Key	Keys
------------	----------	------

PK3 PrimaryKey cod_Departamento

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Departamento	SERIAL/INTEGER	ID	Representa la clave primaria de Departamento que será generada por la base de datos
nombre	VARCHAR(100)	N	Representa nombre de departamento
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene el departamento

Direccion

Nombre Entidad Direccion
Tipo Entidad Independent
Primary Keys cod_Direccion
Definicion Direccion del asegurado Nro calle etc.

Notas

[Keys](#)

Nombre Key	Tipo Key	Keys
PK7_1_1	PrimaryKey	cod_Direccion

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Direccion	SERIAL/INTEGER	ID	Representa la clave primaria de Direccion_Asegurado que será generada por la base de datos
calle	VARCHAR(300)	N	
numero	VARCHAR(10)	Y	Representa el numero de la casa de su dirección
estado	INTEGER	Y	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
cod_Zona	INTEGER	N	Representa la clave primaria de Zona que será generada por la base de datos
cod_AVC_04_A	INTEGER	N	

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Zona	Non-Identifying	One To Zero or More	cod_Zona	NO
AVC_04_A	Non-Identifying	One to Exactly: 1	cod_AVC_04_A	NO

Direccion_Asegurado

Nombre Entidad Direccion_Asegurado
Tipo Entidad Independent
Primary Keys cod_Direccion_Asegurado
Definicion Dirección del asegurado Nro. calle etc.

Notas

[Keys](#)

Nombre Key	Tipo Key	Keys
PK7_1	PrimaryKey	cod_Direccion_Asegurado

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Direccion_Asegurado	SERIAL/INTEGER	ID	Representa la clave primaria de Direccion_Asegurado que será generada por la base de datos
calle	VARCHAR(300)	N	
numero	VARCHAR(10)	Y	Representa el numero de la casa de su dirección
estado	INTEGER	Y	Representa el eliminado lógico 0: inhabilitado 1:

fecha_Registro	DATE	N	habilitado
cod_AVC_04	INTEGER	N	Representa el día mes año en que se registró una dupla
cod_Zona	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos
			Representa la clave primaria de Zona que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AV C_04	NO
Zona	Non-Identifying	One To Zero or More	cod_Zon a	NO

Direccion_Empresa

Nombre Entidad	Direccion_Empresa
Tipo Entidad	Independent
Primary Keys	cod_Direccion_Empresa
Definicion	Direccion de la empresa calle zona etc.
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK7	PrimaryKey	cod_Direccion_Empresa

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Direccion_Em presa calle	SERIAL/INTE GER VARCHAR(30 0)	ID N	Representa la clave primaria de Direccion_Empesa que será generada por la base de datos
numero estado	VARCHAR(10) INTEGER	Y Y	Representa el numero de la casa de su dirección Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro cod_Empresa	DATE INTEGER	N N	Representa el día mes año en que se registró una dupla Representa la clave primaria de Empresa que será generada por la base de datos
cod_Zona	INTEGER	N	Representa la clave primaria de Zona que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Zona	Non-Identifying	One To Zero or More	cod_Zon a	NO
Empresa	Non-Identifying	One To Zero or More	cod_Em presa	NO

Empresa

Nombre Entidad	Empresa
Tipo Entidad	Independent
Primary Keys	cod_Empresa
Definicion	empresa o entidad que tiene fines de lucro en la que cuenta con empleados

Notas

Keys

Nombre Key	Tipo Key	Keys
------------	----------	------

PK11 PrimaryKey cod_Empresa

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Empresa	SERIAL/INTEGER	ID	Representa la clave primaria de Empresa que será generada por la base de datos
num_Empleador	VARCHAR(30)	N	se asignara el numero de acuerdo al código vigente
nombre_RazonSocial	VARCHAR(100)	N	Representa nombre de la razón social
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene la empresa
numero_Trabajadores	INTEGER	N	Representa el número de trabajadores de la empresa al registrarse
n_Padron_Renta	VARCHAR(15)	Y	Representa el registro de número de RUC registrónico de contribuyentes
fecha_Iniciacion_Actividad	DATE	N	Representa el día mes año de inicio de la empresa en funcionamiento
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registro una dupla
norma	INTEGER	N	Representa si esta cumplió cumpliendo normativas 1: alta por cumplimiento 0: baja por incumplimiento
cod_Actividad_Economica	INTEGER	N	Representa la clave primaria del Actividad_Economica que será generada por la base de datos
cod_Tipo_Empresa	INTEGER	N	Representa la clave primaria de Tipo_Empresa que será generada por la base de datos
num_Municipio_Localidad	INTEGER	N	Representa la clave primaria de Municipio_Localidad que será generada por la base de datos

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Actividad_Economica	Non-Identifying	One To Zero or More	cod_Actividad_Economica	NO
Tipo_Empresa	Non-Identifying	One To Zero or More	cod_Tipo_Empresa	NO
Municipio_Localidad	Non-Identifying	One To Zero or More	num_Municipio_Localidad	NO

Funcionalidad

Nombre Entidad Funcionalidad
Tipo Entidad Independent
Primary Keys cod_Funcionalidad
Definicion funcionalidades del sistema
Notas

[Keys](#)

Nombre Key	Tipo Key	Keys
PK48	PrimaryKey	cod_Funcionalidad

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Funcionalidad	VARCHAR(10)	N	Representa la clave primaria de Funcionalidad que será generada por la base de datos
link	VARCHAR(100)	N	Representa la dirección url de las funcionalidades del sistema
nombre	CHAR(50)	N	Representa nombre de la funcionalidad
descripcion	VARCHAR(25)	Y	Representa la descripción que tiene la funcionalidad

estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
cod_Modulo	INTEGER	N	Representa la clave primaria de Módulo que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Modulo	Non-Identifying	One To Zero or More	cod_Mo dulo	NO

Grupo_Familiar

Nombre Entidad	Grupo_Familiar
Tipo Entidad	Independent
Primary Keys	cod_Grupo_Familiar
Definicion	Grupo familiar de un Asegurado como ser sus hijos, etc.
Notas	estado 0 1 cargado 0 1 2 cargado avc06
	baja alta ninguno avc04

Keys

Nombre Key	Tipo Key	Keys
PK28	PrimaryKey	cod_Grupo_Familiar

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Grupo_Familiar	SERIAL/INTEGER	ID	Representa la clave primaria de Grupo_Familiar que será generada por la base de datos
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_exticcion	DATE	N	Representa el día mes año de la extinción del seguro como beneficiario
causas	VARCHAR(150)	Y	
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
tipo	INTEGER	Y	Representa 0: baja 1: alta 2:alta y baja
cod_Beneficiario	INTEGER	N	
cod_AVC_06	INTEGER	N	Representa la clave primaria del AVC_06 que será generada por la base de datos
cod_AVC_04	INTEGER	N	Representa la clave primaria del AVC_04 que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_04	Non-Identifying	One To Zero or More	cod_AVC_04	NO
Beneficiario	Non-Identifying	One To Zero or More	cod_Beneficiario	NO
AVC_06	Non-Identifying	One To Zero or More	cod_AVC_06	NO

Grupo_Familiar_AVC_08

Nombre Entidad	Grupo_Familiar_AVC_08
Tipo Entidad	Dependent
Primary Keys	cod_AVC_08,cod_Beneficiario
Definicion	

Notas

Keys

Nombre Key	Tipo Key	Keys
PK56	PrimaryKey	cod_AVC_08, cod_Beneficiario

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_AVC_08	INTEGER	N	Representa la clave primaria del AVC_08 que será generada por la base de datos
cod_Beneficiario	INTEGER	N	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
AVC_08	Identifying	One To Zero or More	cod_AVC_08	NO
Beneficiario	Identifying	One To Zero or More	cod_Beneficiario	NO

o

Modulo

Nombre Entidad	Modulo
Tipo Entidad	Independent
Primary Keys	cod_Modulo
Definicion	modulos del sistema

Notas

Keys

Nombre Key	Tipo Key	Keys
PK47	PrimaryKey	cod_Modulo

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_Modulo	SERIAL/INTEGER	ID	Representa la clave primaria de Módulo que será generada por la base de datos
nombre	VARCHAR(50)	N	Representa nombre del Módulo
descripcion	VARCHAR(250)	Y	Representa la descripción que tiene el Módulo
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado

Municipio_Localidad

Nombre Entidad	Municipio_Localidad
Tipo Entidad	Independent
Primary Keys	num_Municipio_Localidad
Definicion	municipios o localidades de una provincia perteneciente a un departamento de Bolivia

Notas

Keys

Nombre Key	Tipo Key	Keys
PK31_2	PrimaryKey	num_Municipio_Localidad

Attributes

Atributo	Tipo Dato	Null	Definicion
num_Municipio_Localidad	SERIAL/INTEGER	ID	Representa la clave primaria de Municipio_Localidad que será generada por la base de datos
numero	VARCHAR(25)	N	
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
nombre	VARCHAR(50)	N	Representa nombre del municipio o localidad
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene el municipio o localidad

cod_Provincia INTEGER N Representa la clave primaria de Provincia que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Provincia	Non-Identifying	One To Zero or More	cod_Provincia	NO

Nombre Entidad Nombre_Empresa
Tipo Entidad Independent
Primary Keys cod_Nombre_Empresa
Definicion telefono de la empresa
Notas

Keys

Nombre Key	Tipo Key	Keys
PK8_1	PrimaryKey	cod_Nombre_Empresa

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Nombre_Empresa	SERIAL/INTEGER	ID	
nombre_RazonSocial	VARCHAR(100)	N	Representa el numero de teléfono
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
cod_Empresa	INTEGER	N	Representa la clave primaria de Empresa que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Empresa	Non-Identifying	One To Zero or More	cod_Empresa	NO

Nombre Entidad Ocupacion
Tipo Entidad Independent
Primary Keys cod_Ocupacion
Definicion ocupaciones que existen en una fuente laboral
Notas

Keys

Nombre Key	Tipo Key	Keys
PK24	PrimaryKey	cod_Ocupacion

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Ocupacion	SERIAL/INTEGER	ID	Representa la clave primaria de Ocupación que será generada por la base de datos
nombre	VARCHAR(50)	N	Representa nombre de la ocupación
descripcion	VARCHAR(150)	Y	Representa la descripción que tiene la ocupación
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla

Nombre Entidad Parentesco
Tipo Entidad Independent
Primary Keys cod_Parentesco
Definicion parentesco relación que tiene con el asegurado el beneficiario

Notas[Keys](#)

Nombre Key	Tipo Key	Keys
PK24_1	PrimaryKey	cod_Parentesco

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Parentesco	SERIAL/INTEGER	ID	Representa la clave primaria de Parentesco que será generada por la base de datos
nombre	VARCHAR(50)	N	Representa nombre del parentesco
descripcion	VARCHAR(150)	Y	Representa la descripción que tiene el parentesco
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla

Persona

Nombre Entidad	Persona
Tipo Entidad	Independent
Primary Keys	cod_Persona
Definicion	cualquier persona juridica o natural que se relaciona con la Caja Nacional de Salud

Notas[Keys](#)

Nombre Key	Tipo Key	Keys
PK17	PrimaryKey	cod_Persona

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Persona	SERIAL/INTEGER	ID	
ci	INTEGER	N	Representa la clave primaria de la persona
sexo	VARCHAR(10)	N	Representa de la persona M: masculino F: femenino
estadocivil	VARCHAR(50)	N	
fnac	DATE	N	Representa el día mes año del nacimiento de una persona
edad	INTEGER	Y	Representa la edad en la que se registró en el seguro
apemat	VARCHAR(50)	N	
nombre	VARCHAR(70)	N	Representa nombre de la persona
apepat	VARCHAR(50)	Y	
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registro una dupla

Propietario

Nombre Entidad	Propietario
Tipo Entidad	Independent
Primary Keys	cod_Propietario
Definicion	persona que es propietario de una empresa

Notas[Keys](#)

Nombre Key	Tipo Key	Keys
PK10	PrimaryKey	cod_Propietario

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Propietario	SERIAL/INTEGER	ID	Representa la clave primaria de Propietario que será

fecha_Registro	GER		generada por la base de datos
estado	DATE	N	Representa el día mes año en que se registró una dupla
	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
cod_Persona	INTEGER	N	

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Persona	Non-Identifying	One to Exactly: 1	cod_Persona	NO

Nombre Entidad	Propietario_Empleador
Tipo Entidad	Independent
Primary Keys	cod_Propietario_Empleador
Definicion	
Notas	

[Keys](#)

Nombre Key	Tipo Key	Keys
PK18	PrimaryKey	cod_Propietario_Empleador

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Propietario_E mpleador	SERIAL/INTE GER	ID	
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
estado	INTEGER	Y	Representa el eliminado lógico 0: inhabilitado 1: habilitado
cod_Propietario	INTEGER	N	Representa la clave primaria de Propietario que será generada por la base de datos
cod_Empresa	INTEGER	N	Representa la clave primaria de Empresa que será generada por la base de datos

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Propietario	Non-Identifying	One To Zero or More	cod_Propietario	NO
Empresa	Non-Identifying	One To Zero or More	cod_Empresa	NO

Nombre Entidad	Provincia
Tipo Entidad	Independent
Primary Keys	cod_Provincia
Definicion	provincia perteneciente a un departamento
Notas	

[Keys](#)

Nombre Key	Tipo Key	Keys
PK31_2_1	PrimaryKey	cod_Provincia

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Provincia	SERIAL/INTE GER	ID	Representa la clave primaria de Provincia que será generada por la base de datos
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
nombre	VARCHAR(50)	N	Representa nombre de la provincia
descripcion	VARCHAR(30 0)	Y	Representa la descripción que tiene la provincia
cod_Departamento	INTEGER	N	Representa la clave primaria de Departamento que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Departamento	Non-Identifying	One To Zero or More	cod_Departamen to	NO

Requisito

Nombre Entidad	Requisito
Tipo Entidad	Independent
Primary Keys	cod_Requisito
Definicion	requisitos que se deben cumplir para optar a un tipo de seguro
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK31_1	PrimaryKey	cod_Requisito

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Requisito	SERIAL/INTEGER	ID	Representa la clave primaria de Requisitos que será generada por la base de datos
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
requisito	VARCHAR(50)	N	Representa el nombre de los requisitos
descripcion	VARCHAR(30)	Y	Representa la descripción que tiene el requisito
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
cod_Tipo_Seguro	INTEGER	N	Representa la clave primaria de Tipo_Seguro que será generada por la base de datos

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Tipo_Seguro	Non-Identifying	One To Zero or More	cod_Tipo_Segur o	NO

Rol

Nombre Entidad	Rol
Tipo Entidad	Independent
Primary Keys	cod_Rol
Definicion	roles que puede tener una usuario en el sistema
Notas	

Keys

Nombre Key	Tipo Key	Keys
PK46	PrimaryKey	cod_Rol

Attributes

Attributo	Tipo Dato	Null	Definicion
cod_Rol	SERIAL/INTEGER	ID	Representa la clave primaria de Rol que será generada por la base de datos
nombre	VARCHAR(50)	N	Representa nombre del rol
descripcion	VARCHAR(25)	Y	Representa la descripción que tiene el rol
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla

Rol_Funcionalidad

Nombre Entidad	Rol_Funcionalidad
Tipo Entidad	Dependent

Primary Keys cod_Rol,cod_Funcionalidad

Definicion

Notas

[Keys](#)

Nombre Key	Tipo Key	Keys
PK50	PrimaryKey	cod_Rol, cod_Funcionalidad

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Rol	INTEGER	N	Representa la clave primaria de Rol que será generada por la base de datos
cod_Funcionalidad	VARCHAR(10)	N	Representa la clave primaria de Funcionalidad que será generada por la base de datos

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Rol	Identifying	One To Zero or More	cod_Rol	NO
Funcionalidad	Identifying	One To Zero or More	cod_Funcionalidad	NO

Telefono_Empresa

Nombre Entidad Telefono_Empresa
Tipo Entidad Independent
Primary Keys cod_Telefono_Empresa
Definicion teléfono de la empresa

Notas

[Keys](#)

Nombre Key	Tipo Key	Keys
PK8	PrimaryKey	cod_Telefono_Empresa

[Attributes](#)

Attributo	Tipo Dato	Null	Definicion
cod_Telefono_Empresa	SERIAL/INTEGER	ID	
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
numero	VARCHAR(30)	N	Representa el numero de teléfono
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
cod_Empresa	INTEGER	N	Representa la clave primaria de Empresa que será generada por la base de datos

[Relationships](#)

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Empresa	Non-Identifying	One To Zero or More	cod_Empresa	NO

Tipo_Empresa

Nombre Entidad Tipo_Empresa
Tipo Entidad Independent
Primary Keys cod_Tipo_Empresa
Definicion tipos de empresas que se encuentran catalogadas por normativa de la Caja Nacional de Salud

Notas tabla que puede tener empresas públicas privadas etc.

[Keys](#)

Nombre Key	Tipo Key	Keys
PK13	PrimaryKey	cod_Tipo_Empresa

Attributes

Attribute	Data Type	Null	Definition
cod_Tipo_Empresa	SERIAL/INTEGER	ID	Representa la clave primaria de Tipo_Empresa que será generada por la base de datos
nombre	VARCHAR(100)	N	Representa nombre del tipo de empresa
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene el tipo de empresa
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
Nombre Entidad	Tipo_Seguro		
Tipo Entidad	Independent		
Primary Keys	cod_Tipo_Seguro		
Definicion	Tipos de seguros que existen en la Caja Nacional de Salud		
Notas			

Keys

Nombre Key	Tipo Key	Keys
PK31	PrimaryKey	cod_Tipo_Seguro

Attributes

Attribute	Data Type	Null	Definition
cod_Tipo_Seguro	SERIAL/INTEGER	ID	Representa la clave primaria de Tipo_Seguro que será generada por la base de datos
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
nombre	VARCHAR(50)	N	Representa nombre de tipo de seguro
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene el tipo de seguro
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
Nombre Entidad	Usuario		
Tipo Entidad	Independent		
Primary Keys	cod_Usuario		
Definicion	usuario Persona que accedera la sistema de acuerdo a un rol		
Notas			

Keys

Nombre Key	Tipo Key	Keys
PK45	PrimaryKey	cod_Usuario

Attributes

Attribute	Data Type	Null	Definition
cod_Usuario	SERIAL/INTEGER	ID	Representa la clave primaria de Usuario que será generada por la base de datos
userNombre	VARCHAR(50)	N	Representa las login o nombre de usuario del usuario del sistema
password	VARCHAR(50)	N	Representa las clave del usuario del sistema
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla
cod_Rol	INTEGER	N	Representa la clave primaria de Rol que será generada por la base de datos
cod_Persona	INTEGER	Y	

Relationships

Nombre Entidad	Tipo	Cardinalidad	Foreign Keys	Is SuperTipo
Persona	Non-Identifying	Zero or One to Exactly: 1	cod_Persona	NO
Rol	Non-Identifying	One To Zero or More	cod_Rol	NO

Zona

Nombre Entidad	Zona
Tipo Entidad	Independent
Primary Keys	cod_Zona
Definicion	zona ubicación de un lugar

Notas

Keys

Nombre Key	Tipo Key	Keys
PK6	PrimaryKey	cod_Zona

Attributes

Atributo	Tipo Dato	Null	Definicion
cod_Zona	SERIAL/INTEGER	ID	Representa la clave primaria de Zona que será generada por la base de datos
nombre	VARCHAR(100)	N	Representa nombre de la zona
descripcion	VARCHAR(300)	Y	Representa la descripción que tiene la zona
estado	INTEGER	N	Representa el eliminado lógico 0: inhabilitado 1: habilitado
fecha_Registro	DATE	N	Representa el día mes año en que se registró una dupla

2.1.2.2.3.11 Modelo de Implementación

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del Sistema. (Este modelo es sólo una versión preliminar al final de la Fase de Elaboración, posteriormente tiene bastante refinamiento).

2.1.2.2.3.11.1 Modelado de Diagrama de Paquetes

2.1.2.2.3.11.1.1 Diagrama de Paquetes

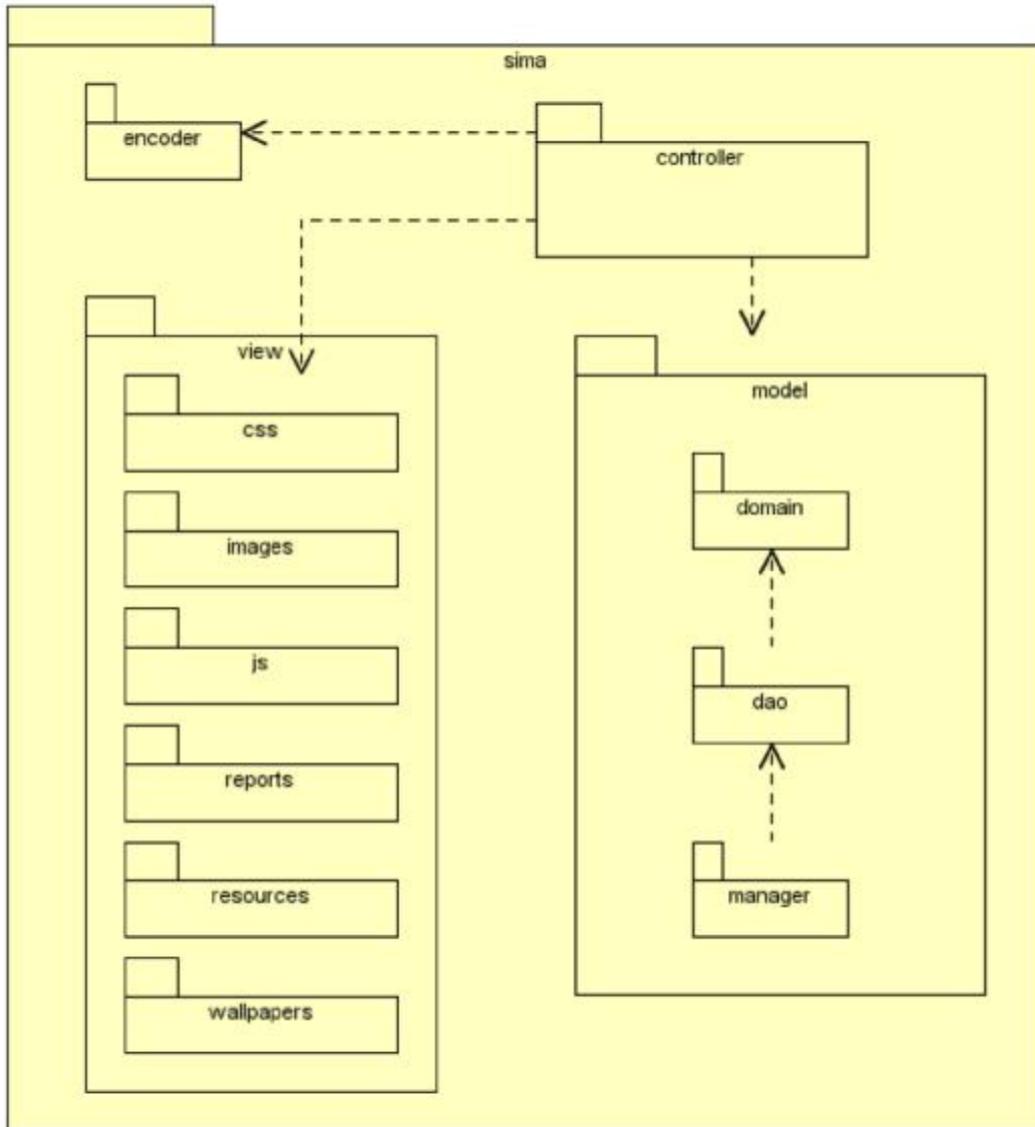


Figura 433. Diagrama de Paquete

2.1.2.2.3.11.2 Modelado de Diagrama de Componentes

2.1.2.2.3.11.2.1 Diagrama de Componentes

2.1.2.2.3.11.2.1.1 Diagrama de Componente: Ingresar al Sistema

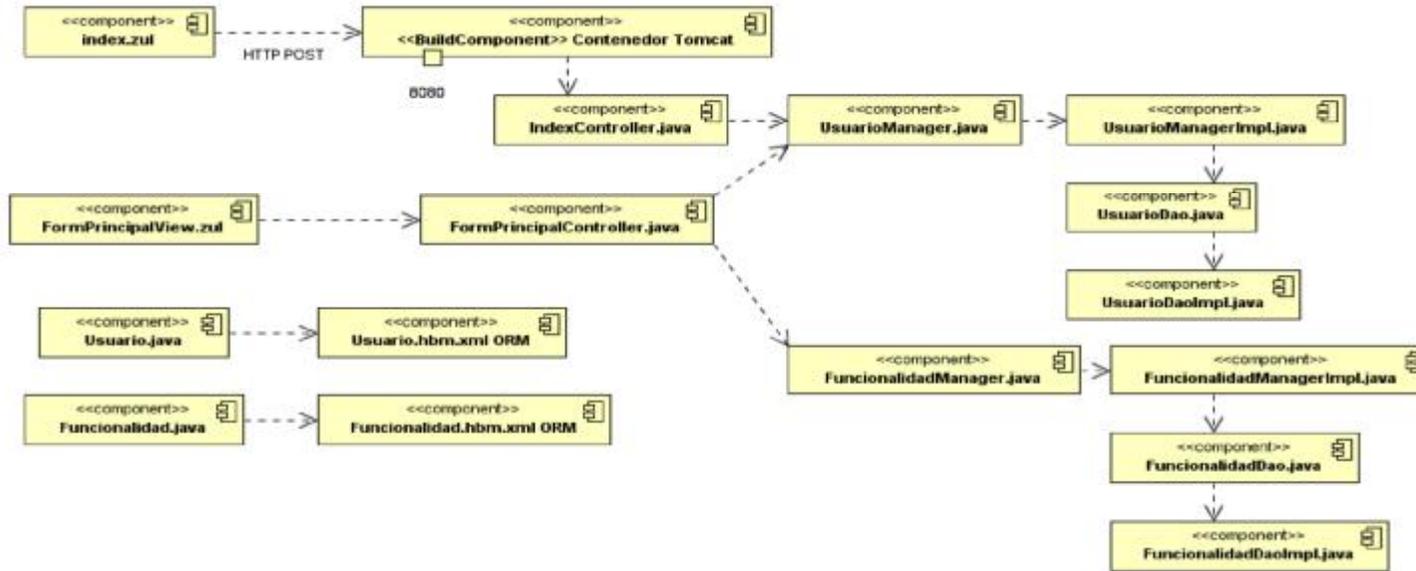


Figura 434. Diagrama de Componente: Ingresar al Sistema

2.1.2.2.3.11.2.1.2 Diagrama de Componente: Administrar Actividad Económica

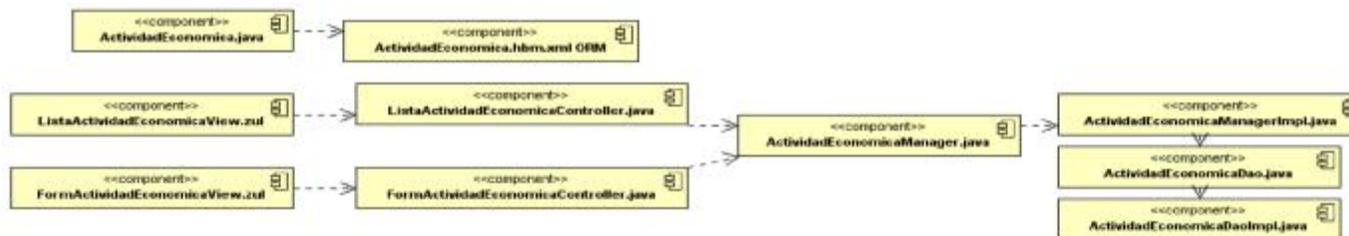


Figura 435. Diagrama de Componente: Administrar Actividad Económica

2.1.2.2.3.11.2.1.3 Diagrama de Componente: Administrar Centro de Salud

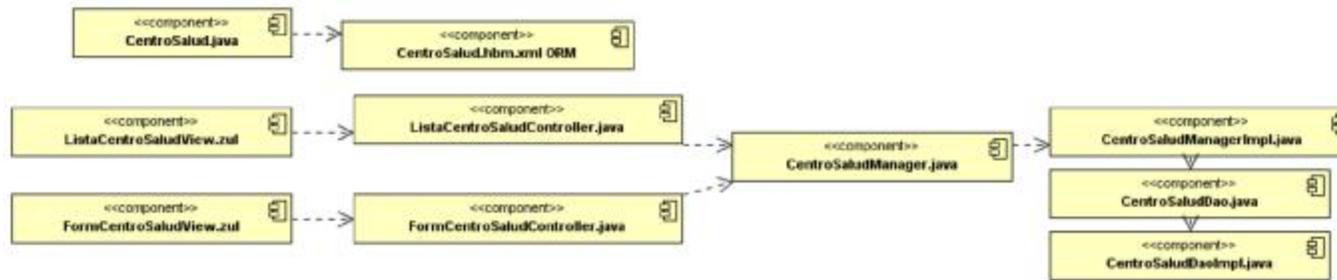


Figura 436. Diagrama de Componente: Administrar Centro de Salud

2.1.2.2.3.11.2.1.4 Diagrama de Componente: Administrar Departamento

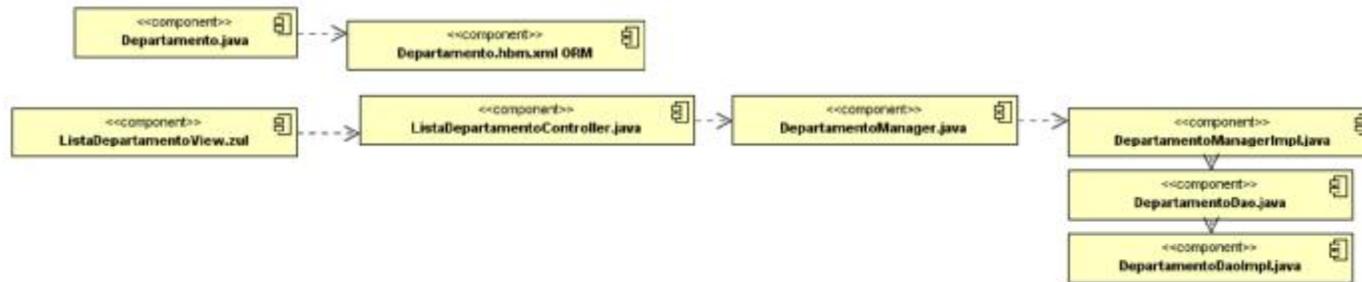


Figura 437. Diagrama de Componente: Administrar Departamento

2.1.2.2.3.11.2.1.5 Diagrama de Componente: Administrar Funcionalidad

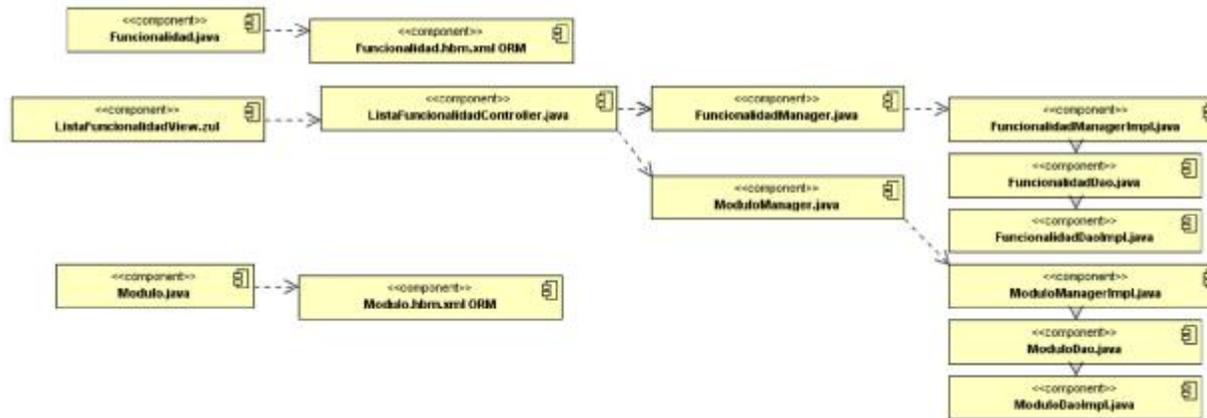


Figura 438. Diagrama de Componente: Administrar Funcionalidad

2.1.2.2.3.11.2.1.6 Diagrama de Componente: Administrar Módulo

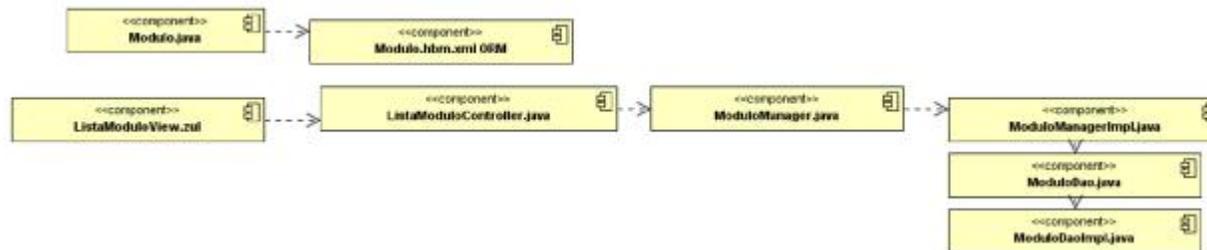


Figura 439. Diagrama de Componente: Administrar Módulo

2.1.2.2.3.11.2.1.7 Diagrama de Componente: Administrar Municipio Localidad

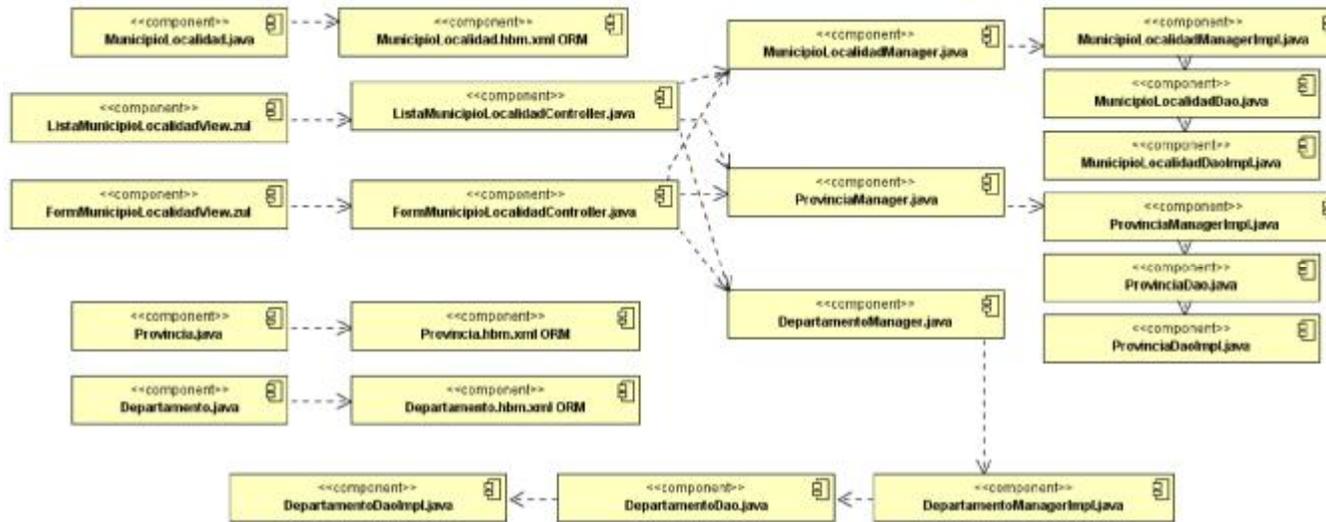


Figura 440. Diagrama de Componente: Administrar Municipio Localidad

2.1.2.2.3.11.2.1.8 Diagrama de Componente: Administrar Ocupación

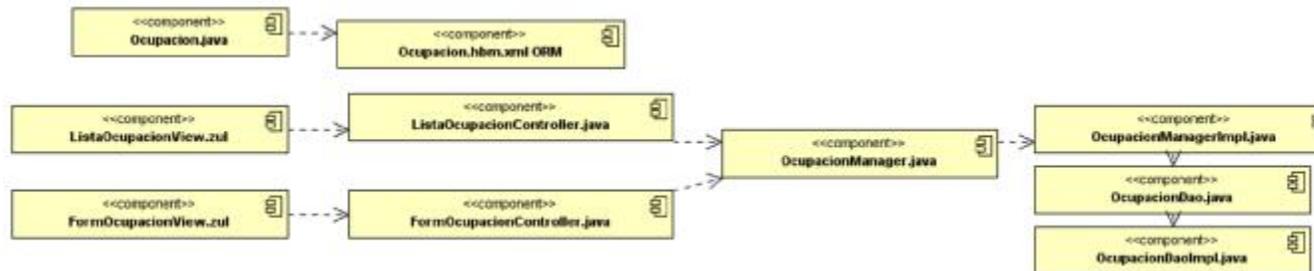


Figura 441. Diagrama de Componente: Administrar Ocupación

2.1.2.2.3.11.2.1.9 Diagrama de Componente: Administrar Parentesco

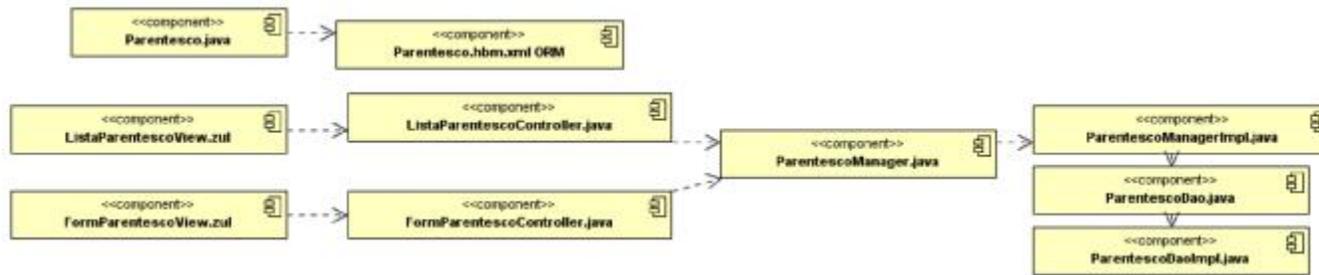


Figura 442. Diagrama de Componente: Administrar Parentesco

2.1.2.2.3.11.2.1.10 Diagrama de Componente: Administrar Provincia

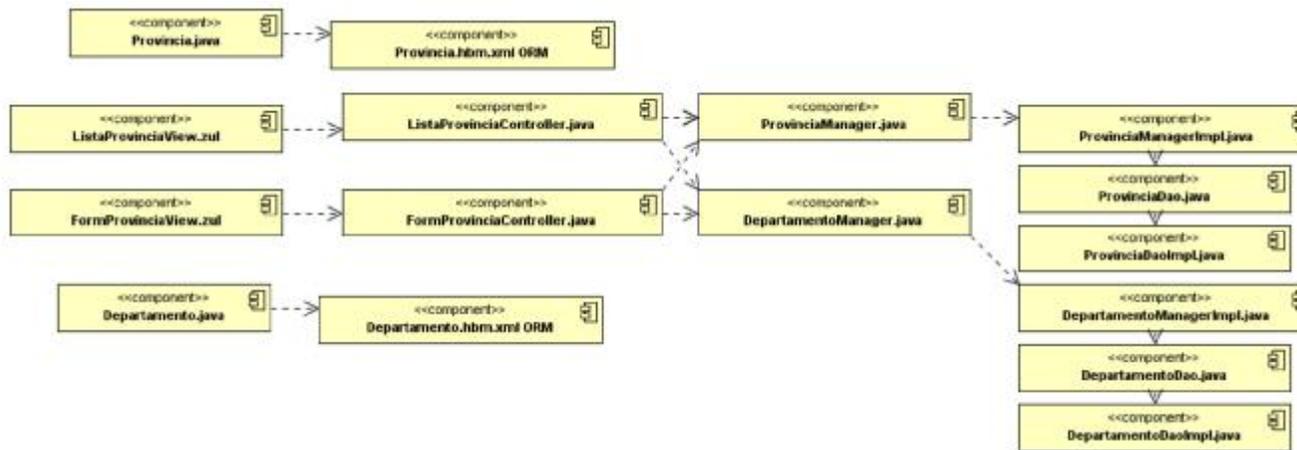


Figura 443. Diagrama de Componente: Administrar Provincia

2.1.2.2.3.11.2.1.11 Diagrama de Componente: Administrar Requisito

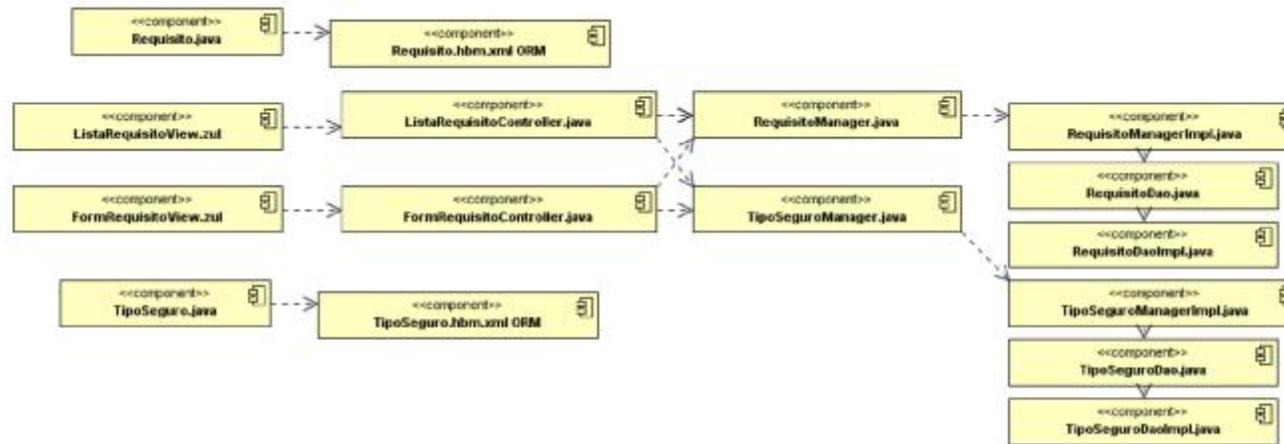


Figura 444. Diagrama de Componente: Administrar Requisito

2.1.2.2.3.11.2.1.12 Diagrama de Componente: Administrar Rol

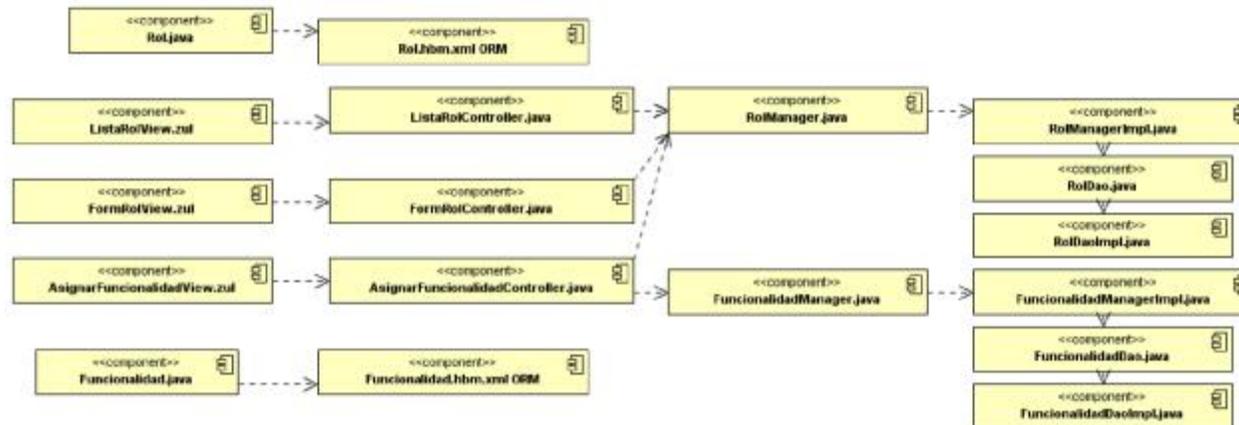


Figura 445. Diagrama de Componente: Administrar Rol

2.1.2.2.3.11.2.1.13 Diagrama de Componente: Administrar Tipo de Empresa

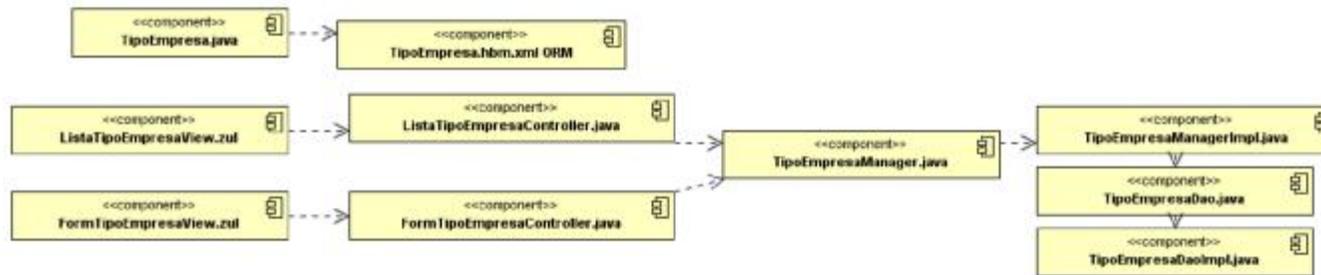


Figura 446. Diagrama de Componente: Administrar Tipo de Empresa

2.1.2.2.3.11.2.1.14 Diagrama de Componente: Administrar Tipo de Seguro

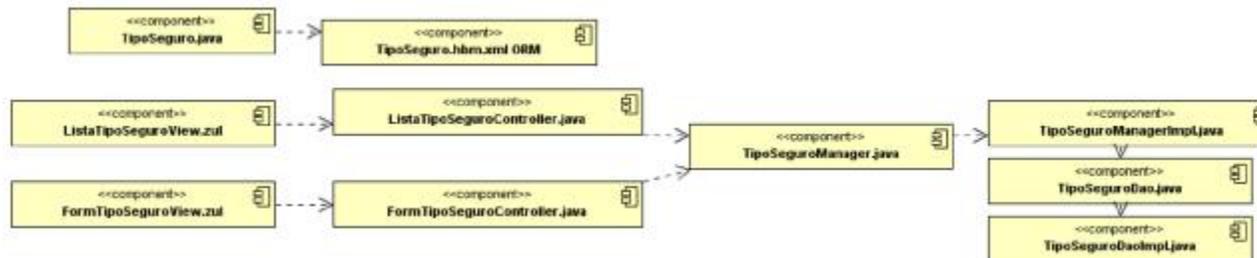


Figura 447. Diagrama de Componente: Administrar Tipo de Seguro

2.1.2.2.3.11.2.1.15 **Diagrama de Componente: Administrar Usuario**

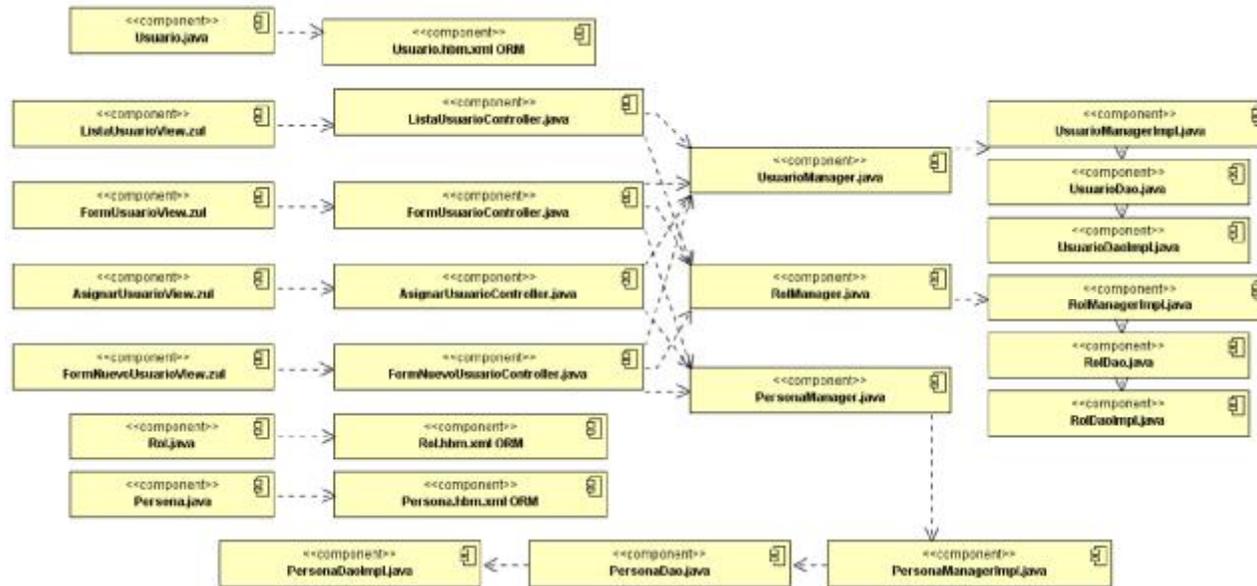


Figura 448. Diagrama de Componente: Administrar Usuario

2.1.2.2.3.11.2.1.16 **Diagrama de Componente: Administrar Zona**

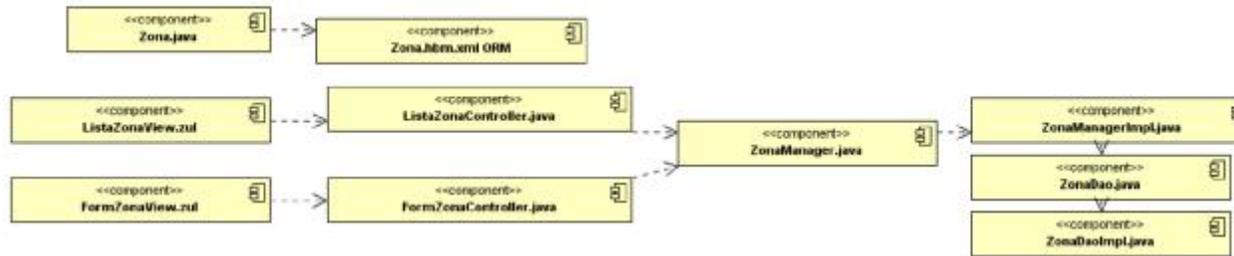


Figura 449. Diagrama de Componente: Administrar Zona

2.1.2.2.3.11.2.1.17

Diagrama de Componente: Gestión Empresa

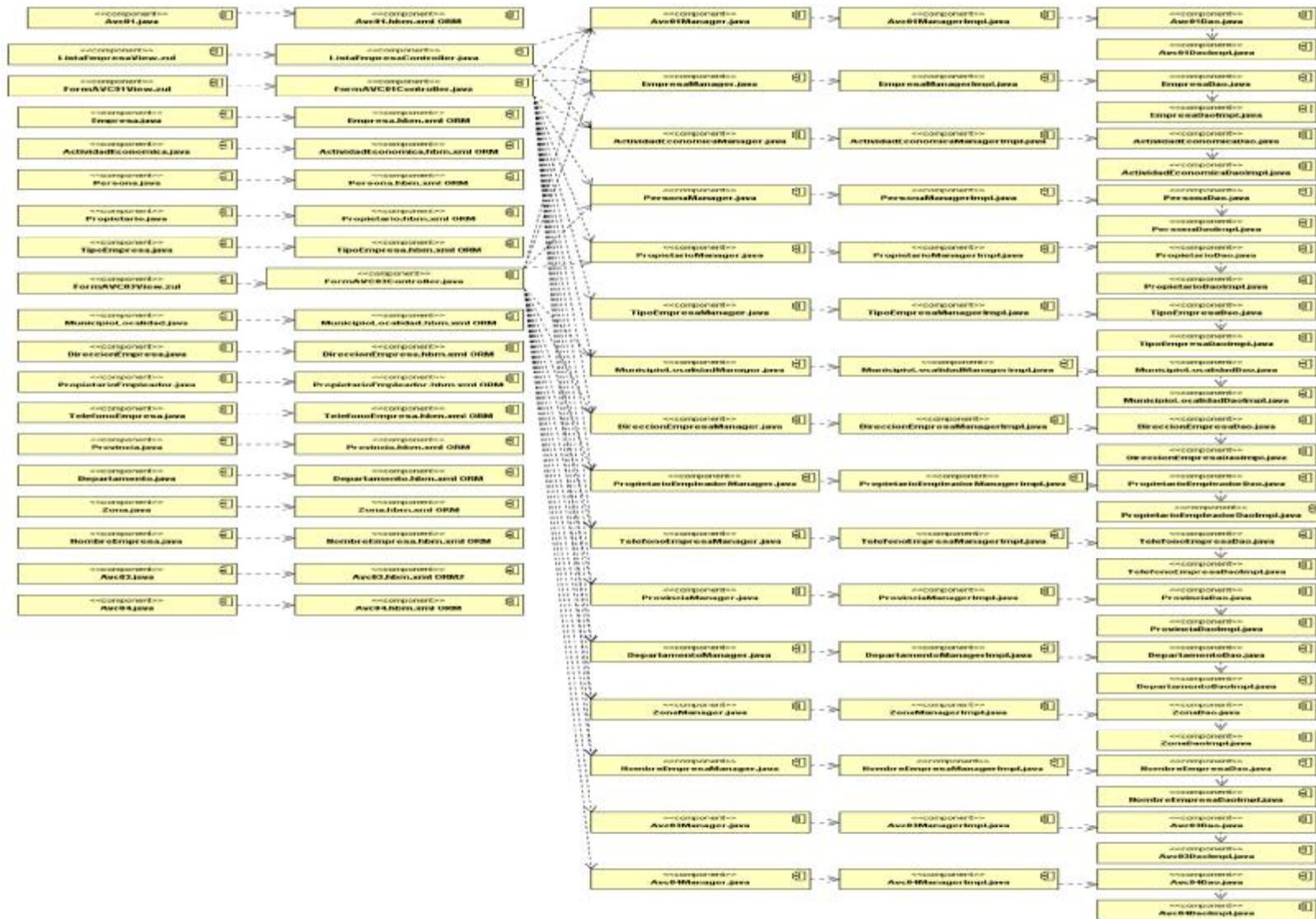


Figura 450. Diagrama de Componente: Gestión Empresa

2.1.2.2.3.11.2.1.18

Diagrama de Componente: Gestión Asegurado por Tipo de Seguro

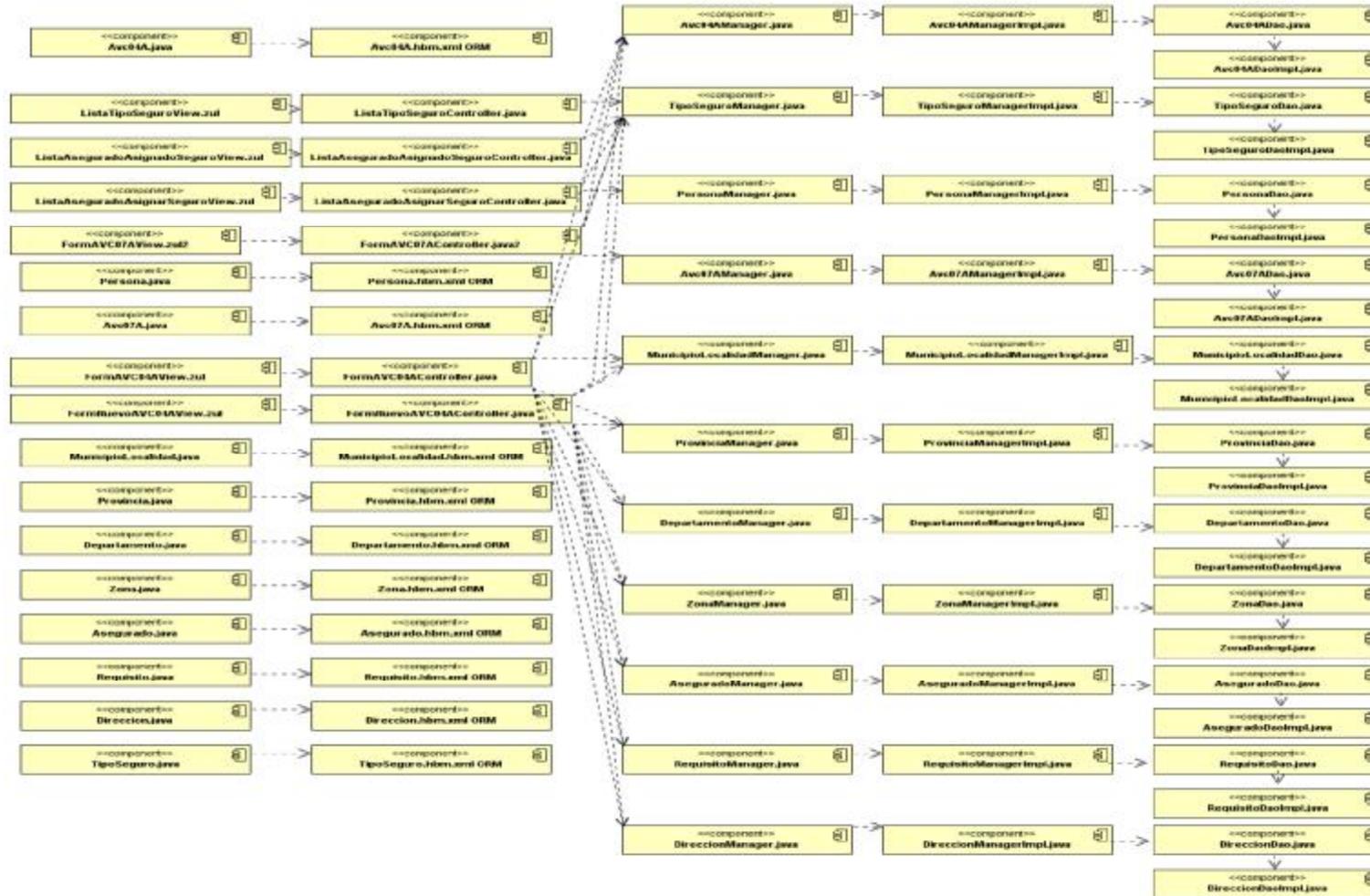


Figura 451. Diagrama de Componente: Gestión Asegurado por Tipo de Seguro

Figura 452. Diagrama de Componente: Gestión Asegurado

2.1.2.2.3.11.2.1.20

Diagrama de Componente: Gestión Certificado Incapacidad Temporal

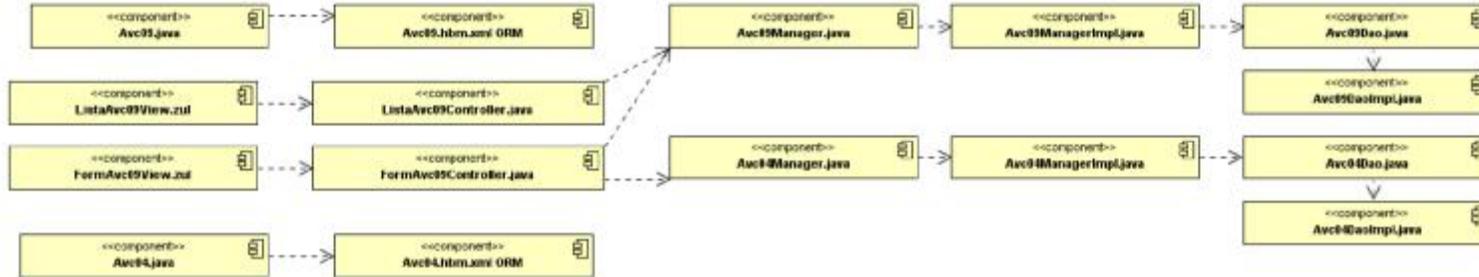


Figura 453. Diagrama de Componente: Gestión Certificado Incapacidad Temporal

2.1.2.2.3.11.2.1.21

Diagrama de Componente: Gestión Autorización Médica Externa

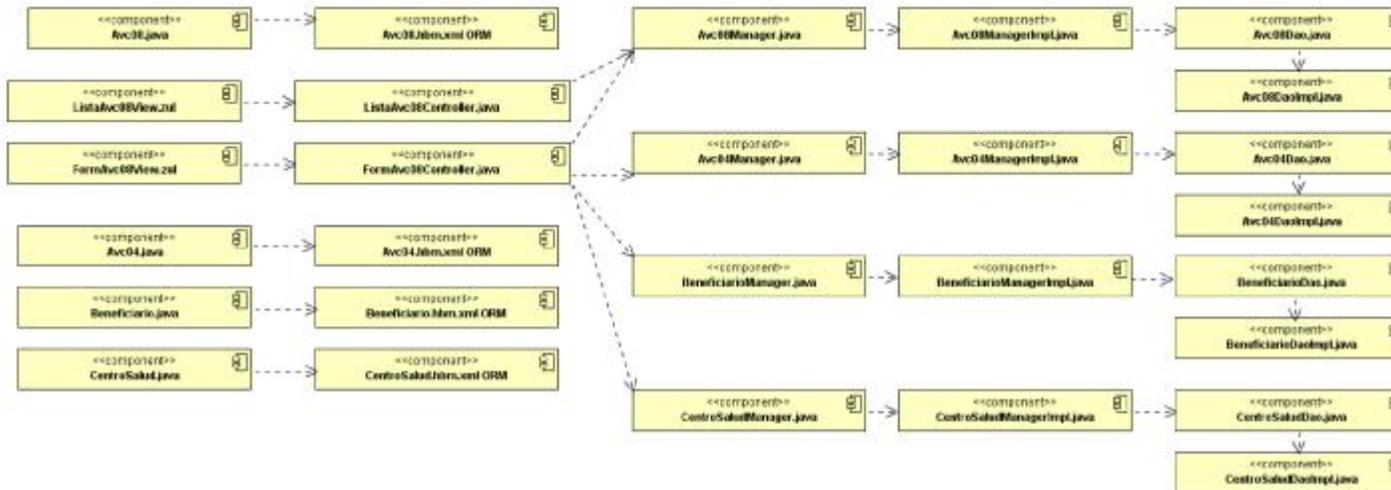


Figura 454. Diagrama de Componente: Gestión Autorización Médica Externa

2.1.2.2.3.11.2.1.22 **Diagrama de Componente: Gestión Altas y Bajas Beneficiarios**

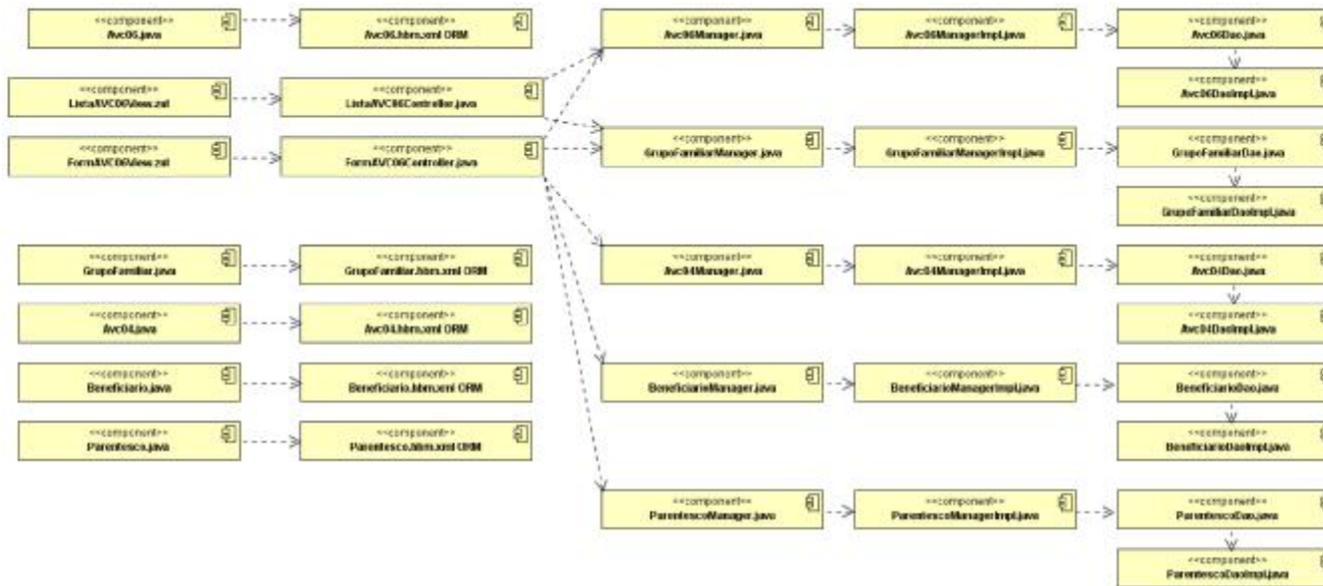


Figura 455. Diagrama de Componente: Gestión Altas y Bajas Beneficiarios

2.1.2.2.3.11.2.1.23 **Diagrama de Componente: Gestión Beneficiario**

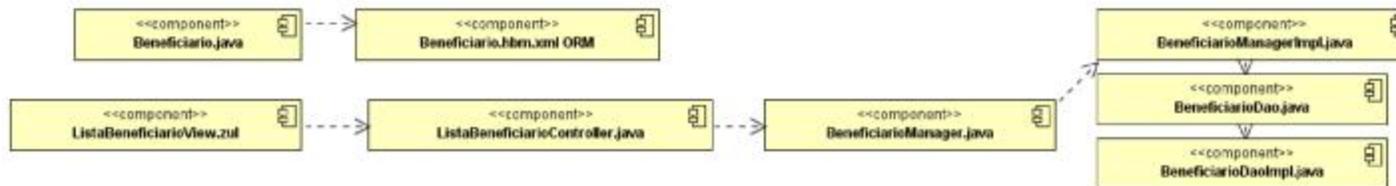


Figura 456. Diagrama de Componente: Gestión Beneficiario

2.1.2.2.3.11.2.1.24 Diagrama de Componente: Gestión Reporte



Figura 457. Diagrama de Componente: Gestión Reporte

2.1.2.2.3.11.2.1.25 **Diagrama de Componente: Gestión Reporte Ficha**

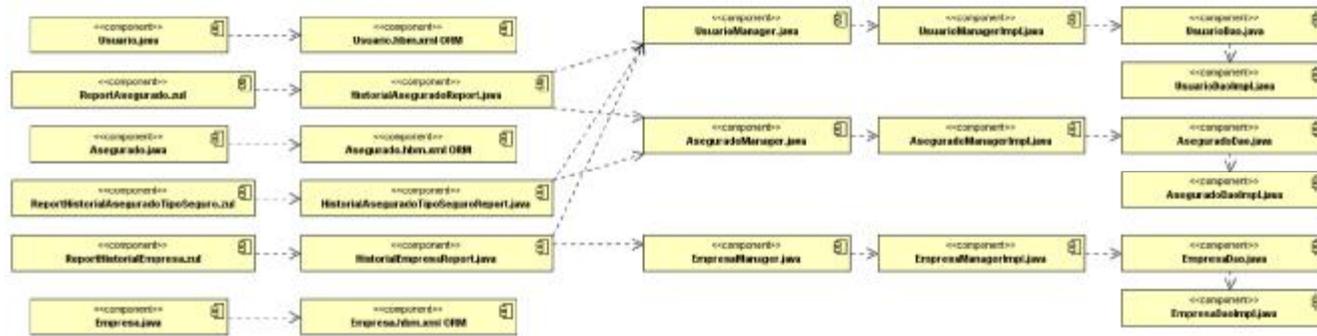


Figura 458. Diagrama de Componente: Gestión Reporte Ficha

2.1.2.2.3.11.2.1.26

Diagrama de Componente: Gestión Reporte Estadístico

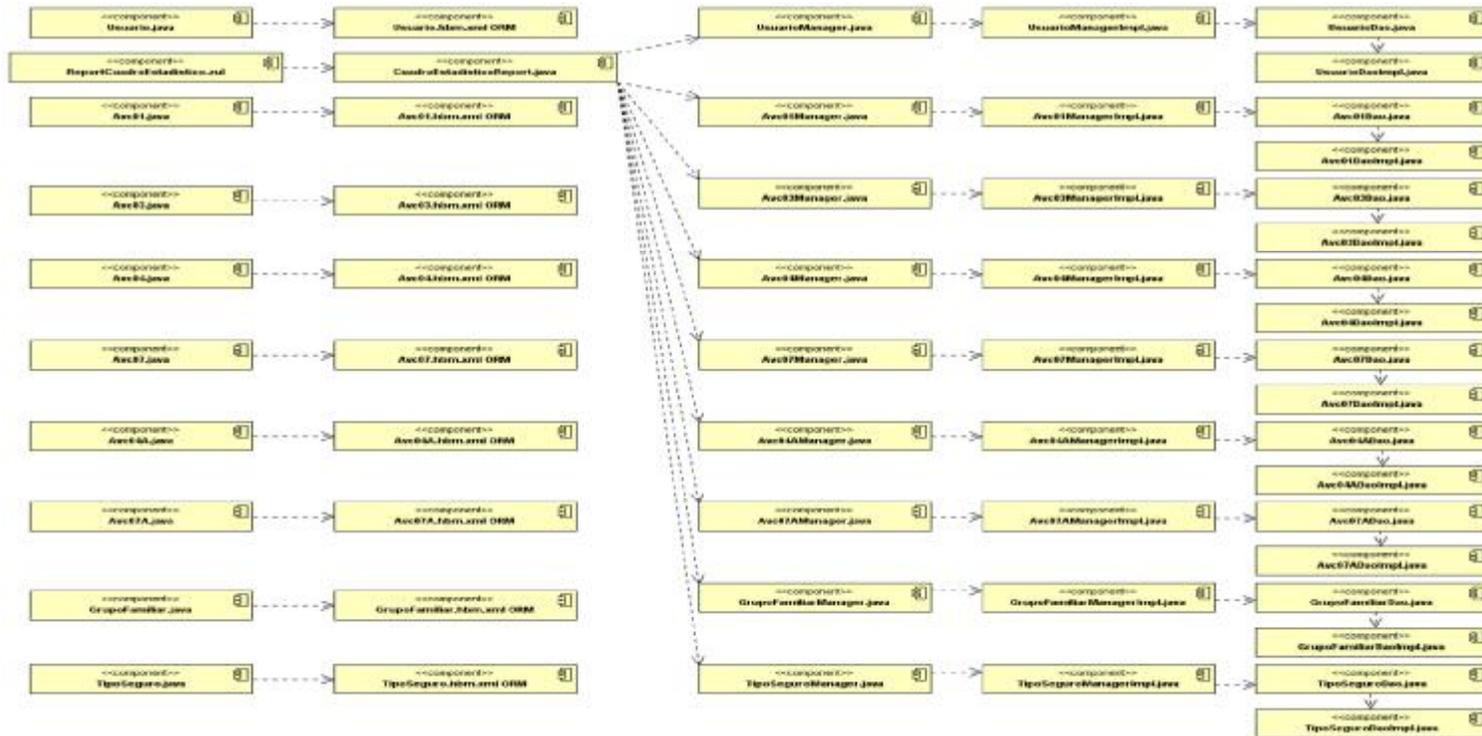


Figura 459. Diagrama de Componente: Gestión Reporte Estadístico

2.1.2.2.3.11.2.1.27 Diagrama de Componente: Administrar Backup

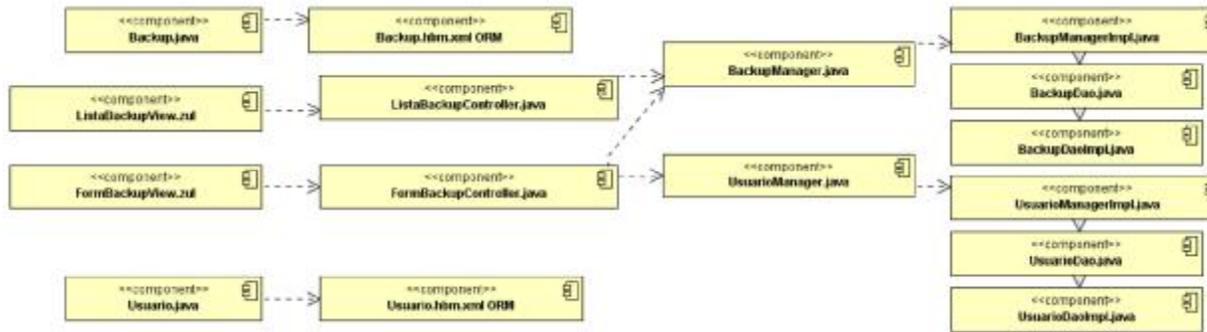


Figura 460. Diagrama de Componente: Administrar Backup

2.1.2.2.3.12 Modelo de Despliegue

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

2.1.2.2.3.12.1 Modelado de Diagrama de Despliegue

2.1.2.2.3.12.1.1 Diagrama de Despliegue

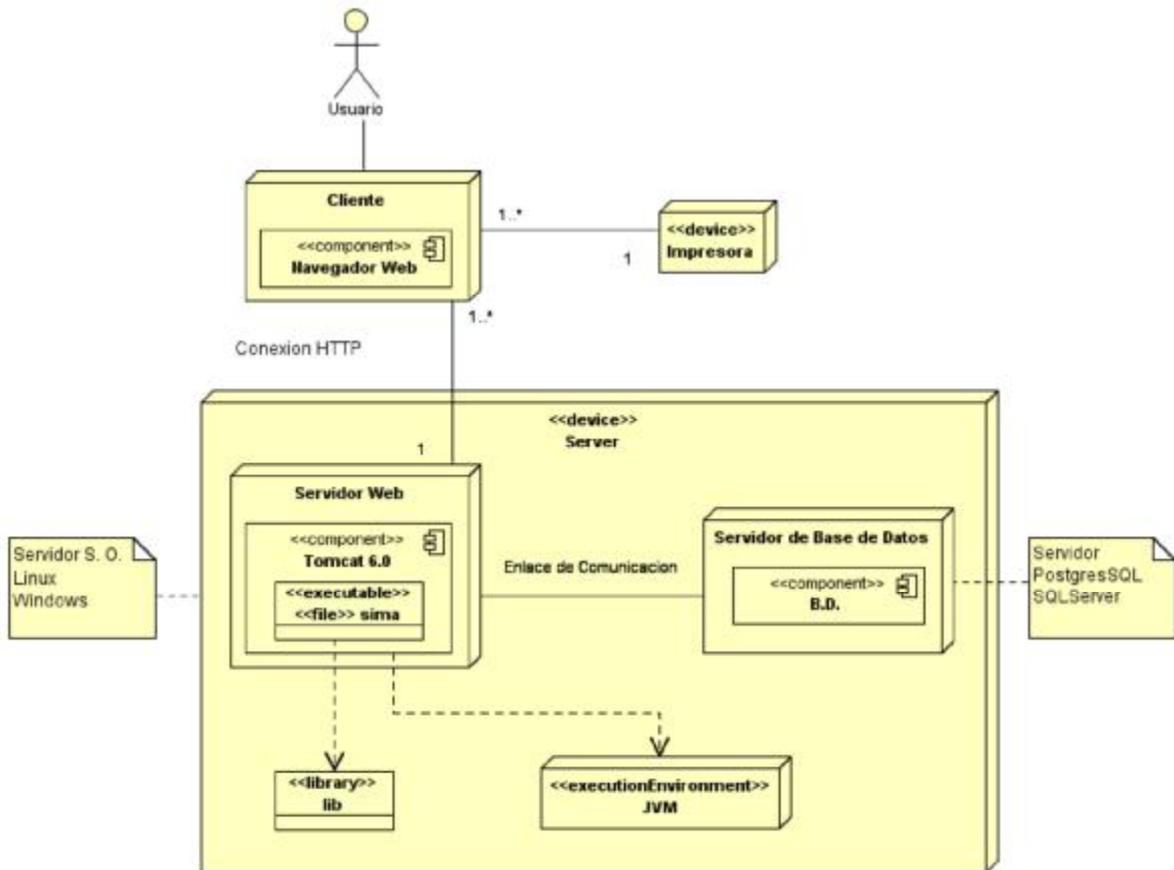


Figura 461. Diagrama de Despliegue

2.1.2.2.3.13 Casos de Prueba

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

2.1.2.2.3.13.1 Plan de Pruebas

2.1.2.2.3.13.1.1 Descripción de Aspectos Generales

Esta sección establece el alcance y el objetivo del Plan de Pruebas. Es aquí donde se describen los aspectos fundamentales del esfuerzo que se hará para probar cada uno de los módulos que conforman el Sistema descrito en este Plan de Desarrollo de Software, independiente las características y tamaño que ésta pueda tener.

2.1.2.2.3.13.1.1.1 Objetivo

Este Plan de Pruebas fue desarrollado con el fin de guiar el proceso de pruebas al proyecto SIMA Sistema Informático para el Mejoramiento de Afiliación desarrollado por nuestro grupo de trabajo, esto con el fin de asegurar una excelente calidad del software desarrollado, encontrando para ello errores que puedan perjudicar en el funcionamiento de este, los cuales una vez corregidos, se podrán elaborar un documento que presente las evaluaciones correspondientes en el plazo más corto posible.

2.1.2.2.3.13.1.1.2 Entorno o Marco

La necesidad de aplicar este plan de pruebas radica en la necesidad de obtener un software de calidad, que cumpla con los requerimientos de la Caja Nacional de Salud Regional Tarija, además que sea aprobado por la comunidad de usuarios finales des sistema.

2.1.2.2.3.13.1.1.3 Arquitectura Técnica

Para la realización de las pruebas se tomará en cuenta los módulos desarrollados en la fase de construcción y los diagramas realizados en la fase de elaboración.

2.1.2.2.3.13.1.1.4 Especificaciones del Software y Hardware

Corresponde a una lista individualizada de todo el hardware y el software que utiliza la aplicación, incluyendo proveedores y versiones.

HARDWARE
Computadora con la suficiente capacidad de soportar el software Los requisitos de hardware del Servidor Web son los siguientes: <ul style="list-style-type: none">• Procesador P-IV de 2.4Ghz en adelante.• Memoria RAM de 512 Mb en adelante.• Lector de CD: 4x.• Monitor: Color de 15".• Espacio libre en la unidad de disco rígido: 140 Mb aproximadamente.• Conexión a Internet ADSL. O en su Defecto Conexión Intranet.

Tabla 202. Hardware

SOFTWARE		
Aplicación	Versión	Descripción

Sistema Informático para el Mejoramiento de Afiliación	0.3	Sistema desarrollado por el grupo de trabajo.
Rational Rose Enterprise Edition	7.6	Herramienta de Diseño y Desarrollo de Diagramas UML para RUP
Visual Paradigm for UML Enterprise Edition	6.4	Herramienta de Diseño y Desarrollo de Diagramas UML para RUP
PostgreSQL	8.3	Motor de Base de Datos Relacional.

Tabla 203. Software

OTROS		
Descripción	Versión	Descripción
Documentación	0.3	Documentación de Desarrollo de Software

Tabla 204. Otros

2.1.2.2.3.13.1.1.5 Alcance

Este plan describe las pruebas del Sistema, que serán aplicadas a los componentes del Sistema Informático para el Mejoramiento de Afiliación.

Se asume que antes de probar cada módulo habrá una revisión informal y sólo el código que ha sido revisado como exitoso será probado.

Las unidades de prueba serán realizadas a través del programa manejador de pruebas que ejecuta el chequeo de los límites y las pruebas básicas de caja negra.

Las siguientes medidas de desempeño serán probadas:

Validación correcta de las Entradas de Dato.

Tiempo de respuesta de carga del Sitio Web.

Tiempo de respuesta y Pool de conexiones a la Base de Datos.

2.1.2.2.3.13.1.1.6 Referencias

Glosario del Proyecto.

Plan de Desarrollo de Software.

2.1.2.2.3.13.2 Descripción de las Pruebas Planeadas

2.1.2.2.3.13.2.1 Descripción de las Pruebas Incluidas

Las siguientes pruebas serán realizadas a todo el Sistema de Informático:

Pruebas de Pool de Conexiones.

Pruebas Funcionales.

Pruebas de Validación de Datos.

Pruebas de Perfiles de desempeño.

2.1.2.2.3.13.3 Planeación de la Ejecución de las Pruebas

La siguiente es una descripción de los tipos y técnicas de pruebas que se harán sobre el Sistema Informático, junto con la manera en la que éstas se ejecutarán.

2.1.2.2.3.13.4 Tipos y Técnicas de Pruebas

2.1.2.2.3.13.4.1 Pruebas de Pool de Conexiones

Las pruebas de pooling de conexiones serán realizadas para asegurar que los datos no son corruptos, dentro de las estructuras de datos internas. Estas pruebas son realizadas de forma independiente de la Interfaz de Usuario, en forma de pruebas de caja negra.

Objetivo de la Técnica:	Verificar la las conexiones realizadas a la base de datos y determinar el rendimiento del Sistema y del Motor de Base de Datos (PostgreSQL).
Técnica:	Analizar la carga a la conexión mediante el JDBC, simulando usuarios conectados.
Hitos:	Rastreo de cuello de botella en la conexión a la base de datos, realizada por el sistema.
Herramientas Requeridas:	Apache JMeter.
Criterios de Éxito:	Las conexiones se realizan con fluidez y una por usuario.

Tabla 205. Pruebas de Pool de Conexiones

2.1.2.2.3.13.4.2 Pruebas Funcionales

Las pruebas funcionales serán realizadas para verificar que todos los requerimientos funcionales se cumplen satisfactoriamente. Estas serán cumplidas a través de pruebas de caja negra.

Objetivo de la Técnica:	Verificar requerimientos funcionales del sistema.
Técnicas:	Verificar que los requerimientos funcionales expuestos en los casos de uso se cumplen.
Hitos:	Casos de Uso del Sistema Informático.
Herramientas Requeridas:	Herramientas de backup y recuperación. Herramientas de Instalación y Monitoreo (registro, disco duro, CPU, memoria, etc.) Herramientas de Generación de Datos
Criterio de éxito:	Los siguientes elementos son probados exitosamente: <ul style="list-style-type: none">• Todos los casos de uso del Sistema Informático. La prueba funcional fue realizada juntamente con la persona encargada de su Administración.

Tabla 206. Pruebas Funcionales

2.1.2.2.3.13.4.3 Pruebas de Validación de Datos

Las pruebas de validación de datos verifican la interacción del usuario con el software y la información que introduce. El objetivo de las pruebas de la interfaz de usuario es asegurar que dicha interfaz proporciona al usuario el acceso y validación a todos los datos introducidos por el Usuario con motivo de dar Integridad la información.

Objetivo de la Técnica:	Verificar y probar la Interfaz de entrada Usuario- Sistema Verificar todas las posibles combinaciones de caracteres para entradas del sistema.
Técnica:	Crear o modificar pruebas para cada ventana para verificar todas las entradas de datos.
Hitos:	El testear verificara todas las entradas del Sistema.
Herramientas Requeridas:	-
Criterio de Éxito:	Todas las entradas han sido verificadas mediante combinaciones de caracteres para dar con los posibles errores del Sistema.

Tabla 207. Pruebas de Validación de Datos

2.1.2.2.3.13.4.4 Pruebas de Perfiles de Desempeño

Los perfiles de desempeño prueban tiempos de respuesta, tasas de transacción y otros requerimientos sensibles al tiempo son medidos y evaluados. El objetivo de los perfiles de desempeño es verificar el desempeño con el que los requerimientos se han cumplido. Un perfil de desempeño es implementado y ejecutado para refinar el desempeño y comportamiento de los elementos objetivo de las pruebas.

Objetivo de la Técnica:	Determinar el comportamiento del objeto de la prueba bajo: <ul style="list-style-type: none">• Carga de trabajo anticipada.• Peor caso de la carga de trabajo.
Técnica:	Concurrencia de visita al Sistema Informático. Prueba de Stress al servidor Web.
Hitos:	Cálculo de tiempos de software.
Herramientas requeridas:	JMeter
Criterio de éxito:	Los tiempos de respuesta son los esperados para el número de usuarios que acceden al Sistema Informático.

Tabla 208. Pruebas de Perfiles de Desempeño

2.1.2.2.3.13.5 Ejecución de las Pruebas

2.1.2.2.3.13.5.1 Demostración y Resultados de la Prueba de Validación de Datos

2.1.2.2.3.13.5.1.1 Partición Equivalente : Casos de Prueba Caja Negra

Interfaz: Ingresar al Sistema

Usuario: Alfanumérico 50 caracteres

Clave: Alfanumérico 50 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Usuario	1_ Alfanumérico	3_ Vacío
Tamaño: Usuario	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Clave	4_ Alfanumérico	6_ Vacío
Tamaño: Clave	5_ 1<=Caracteres<=50	7_ Caracteres >50

Tabla 209. Partición Equivalente. Interfaz: Ingresar al Sistema

Casos de Prueba válidas

CP1	
Usuario	Clave
ADMINISTRADOR	ADMINISTRADOR

Cubre las clases de equivalencias válidas: 1-2-4-5

Casos de Pruebas Inválidas

CP2	
Usuario	Clave
ADMINISTRADOR	

Cubre las clases de equivalencias inválidas: 3

CP3	
Usuario	Clave
	ADMINISTRADOR

Cubre las clases de equivalencias inválidas: 6

Interfaz: Formulario (ABM) Actividad Económica

Nombre Alfabético 100 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfanumérico	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=100	4_ Caracteres >100
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >300
Tamaño: Descripción	6_ 0<=Caracteres<=300	

Tabla 210. Partición Equivalente. Interfaz: Formulario (ABM) Actividad Económica

Casos de Prueba válidas

CP1

Nombre	Descripción
MINERA	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Centro Salud

Numero Numérico 20 caracteres

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 300 caracteres

Dirección Alfanumérico 150 caracteres

Teléfono Alfanumérico 15 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Numero	1_ Numérico	3_ Vacío
Tamaño: Numero	2_ 1<=Caracteres<=20	4_ Caracteres >20
Tipo: Nombre	5_ Alfabético	7_ Vacío
Tamaño: Nombre	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo Descripción	9_ Alfanumérico	11_ Caracteres >300
Tamaño Descripción	10_ 0<=Caracteres<=300	
Tipo Dirección	12_ Alfanumérico	14_ Vacío
Tamaño Dirección	13_ 1<=Caracteres<=150	15_ Caracteres >150
Tipo: Teléfono	16_ Alfanumérico	18_ Cualquier otra cosa
Tamaño: Teléfono	17_ 1<=Caracteres<=15	19_ Caracteres >15

Tabla 211. Partición Equivalente. Interfaz: Formulario (ABM) Centro Salud

Casos de Prueba válidas

CP1				
Numero	Nombre	Descripción	Dirección	Teléfono
15	HOSPITAL OBRERO		CALLE POTOSÍ	66-95632

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-12-13-16-17

Casos de Pruebas Inválidas

CP2				
Numero	Nombre	Descripción	Dirección	Teléfono
	HOSPITAL OBRERO		CALLE POTOSÍ	66-95632

Cubre las clases de equivalencias inválidas: 3

CP3				
Numero	Nombre	Descripción	Dirección	Teléfono
15			CALLE POTOSÍ	66-95632

Cubre las clases de equivalencias inválidas: 7

CP4				
Numero	Nombre	Descripción	Dirección	Teléfono
15	HOSPITAL OBRERO			66-95632

Cubre las clases de equivalencias inválidas: 14

CP5				
Numero	Nombre	Descripción	Dirección	Teléfono
15	HOSPITAL OBRERO		CALLE POTOSÍ	66-95632

Cubre las clases de equivalencias inválidas: 18

Interfaz: Formulario (ABM) Municipio Localidad

Numero Numérico 25 caracteres

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Numero	1_ Numérico	3_ Vacío
Tamaño: Numero	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Nombre	5_ Alfabético	7_ Vacío
Tamaño: Nombre	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo Descripción	9_ Alfanumérico	11_ Caracteres >300
Tamaño Descripción	10_ 0<=Caracteres<=300	

Tabla 212. Partición Equivalente. Interfaz: Formulario (ABM) Municipio Localidad

Casos de Prueba válidas

CP1		
Numero	Nombre	Descripción
125	YACUIBA	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10

Casos de Pruebas Inválidas

CP2		
Numero	Nombre	Descripción
	YACUIBA	

Cubre las clases de equivalencias inválidas: 3

CP3		
Numero	Nombre	Descripción

125		
-----	--	--

Cubre las clases de equivalencias inválidas: 7

Interfaz: Formulario (ABM) Ocupación

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >150
Tamaño: Descripción	6_ 0<=Caracteres<=150	

Tabla 213. Partición Equivalente. Interfaz: Formulario (ABM) Ocupación

Casos de Prueba válidas

CP1	
Nombre	Descripción
INGENIERO CIVIL	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Parentesco

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >150
Tamaño: Descripción	6_ 0<=Caracteres<=150	

Tabla 214. Partición Equivalente. Interfaz: Formulario (ABM) Parentesco

Casos de Prueba válidas

CP1	
Nombre	Descripción
HUJO(A)	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Provincia

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ $1 \leq \text{Caracteres} \leq 50$	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >300
Tamaño: Descripción	6_ $0 \leq \text{Caracteres} \leq 300$	

Tabla 215. Partición Equivalente. Interfaz: Formulario (ABM) Provincia

Casos de Prueba válidas

CP1	
Nombre	Descripción
GRAN CHACO	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Requisito

Nombre Requisito Alfabético 50 caracteres

Descripción Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Requisito	1_ Alfabético	3_ Vacío
Tamaño: Nombre Requisito	2_ $1 \leq \text{Caracteres} \leq 50$	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >150
Tamaño: Descripción	6_ $0 \leq \text{Caracteres} \leq 150$	

Tabla 216. Partición Equivalente. Interfaz: Formulario (ABM) Requisito

Casos de Prueba válidas

CP1	
Nombre Requisito	Descripción
CERTIFICADO TRABAJO	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre Requisito	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Rol

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 250 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >250
Tamaño: Descripción	6_ 0<=Caracteres<=250	

Tabla 217. Partición Equivalente. Interfaz: Formulario (ABM) Rol

Casos de Prueba válidas

CP1	
Nombre	Descripción
ADMINISTRADOR	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Tipo Empresa

Interfaz: Formulario (ABM) Tipo Empresa

Nombre Alfabético 100 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
----------------------	-----------------------------	-------------------------------

Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=100	4_ Caracteres >100
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >300
Tamaño: Descripción	6_ 0<=Caracteres<=300	

Tabla 218. Partición Equivalente. Interfaz: Formulario (ABM) Tipo Empresa

Casos de Prueba válidas

CP1	
Nombre	Descripción
INDIVIDUAL	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Tipo Seguro

Nombre Alfabético 50 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >300
Tamaño: Descripción	6_ 0<=Caracteres<=300	

Tabla 219. Partición Equivalente. Interfaz: Formulario (ABM) Tipo Seguro

Casos de Prueba válidas

CP1	
Nombre	Descripción
ADULTO MAYOR	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Usuario

Ci Numérico
 Nombre Alfabético 70 Caracteres
 Ape. Pat Alfabético 50 Caracteres
 Ape. Mat Alfabético 50 Caracteres
 Fecha Nac. Date
 Edad Numérico
 Username Alfanumérico 50 Caracteres
 Password Alfanumérico 50 Caracteres
 Password Verificar Alfanumérico 50 Caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Ci	1_ Numérico	2_ Vacío
Tipo: Nombre Tamaño: Nombre	3_ Alfabético 4_ 1<=Caracteres<=70	5_ Vacío 6_ Caracteres >70
Tipo: Ape. Pat Tamaño: Ape. Pat	7_ Alfabético 8_ 0<=Caracteres<=50	9_ Cualquier Cosa 10_ Caracteres >50
Tipo: Ape. Mat Tamaño: Ape. Mat	11_ Alfabético 12_ 1<=Caracteres<=50	13_ Vacío 14_ Caracteres >50
Tipo: Fecha Nac.	15_ Date	16_ Vacío
Tipo: Edad	17_ Numérico	18_ Vacío
Tipo: Username Tamaño: Username	19_ Alfanumérico 20_ 1<=Caracteres<=50	21_ Vacío 22_ Caracteres >50
Tipo: Password Tamaño: Password	23_ Alfanumérico 24_ 1<=Caracteres<=50	25_ Vacío 26_ Caracteres >50
Tipo: Password Verificar Tamaño: Password Verificar	27_ Alfanumérico 28_ 1<=Caracteres<=50	29_ Distinto de Password 30_ Caracteres >50

Tabla 220.Partición Equivalente. Interfaz: Formulario (ABM) Usuario

Casos de Prueba válidas

CP1								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ	10/11/1970	40	*****	*****	*****

Cubre las clases de equivalencias válidas: 1-3-4-7-8-11-12-15-17-19-20-23-24-27-28

Casos de Pruebas Inválidas

CP2								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar

	ALBERTO	GUERRERO	MARQUEZ	10/11/1970	40	*****	*****	*****
--	---------	----------	---------	------------	----	-------	-------	-------

Cubre las clases de equivalencias inválidas: 3

CP3								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158		GUERRERO	MÁRQUEZ	10/11/1970	40	*****	*****	*****

Cubre las clases de equivalencias inválidas: 5

CP4								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	DJK@#%	MÁRQUEZ	10/11/1970	40	*****	*****	*****

Cubre las clases de equivalencias inválidas: 9

CP5								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO		10/11/1970	40	*****	*****	*****

Cubre las clases de equivalencias inválidas: 13

CP6								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ		40	*****	*****	*****

Cubre las clases de equivalencias inválidas: 16

CP7								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ	10/11/1970		*****	*****	*****

Cubre las clases de equivalencias inválidas: 18

CP8								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ	10/11/1970	40		*****	*****

Cubre las clases de equivalencias inválidas: 21

CP9								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ	10/11/1970	40	*****		*****

Cubre las clases de equivalencias inválidas: 25

CP10								
Ci	Nombre	Ape. Pat	Ape. Mat	Fecha Nac.	Edad	Username	Password	Password Verificar
78954158	ALBERTO	GUERRERO	MARQUEZ	10/11/1970	40	*****	*****	*****

Cubre las clases de equivalencias inválidas: 29

Interfaz: Asignar Login y Clave

Username Alfanumérico 50 Caracteres

Password Alfanumérico 50 Caracteres

Password Verificar Alfanumérico 50 Caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Username	1_ Alfanumérico	3_ Vacío
Tamaño: Username	2_ 1<=Caracteres<=50	4_ Caracteres >50
Tipo: Password	5_ Alfanumérico	7_ Vacío
Tamaño: Password	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo: Password Verificar	9_ Alfanumérico	11_ Distinto de Password
Tamaño: Password Verificar	10_ 1<=Caracteres<=50	12_ Caracteres >50

Tabla 221. Partición Equivalente. Interfaz: Asignar Login y Clave

Casos de Prueba válidas

CP1		
Username	Password	Password Verificar
*****	*****	*****

Cubre las clases de equivalencias válidas: 1-5-9

Casos de Pruebas Inválidas

CP		
Username	Password	Password Verificar
	*****	*****

Cubre las clases de equivalencias inválidas: 3

CP3		
Username	Password	Password Verificar
*****		*****

Cubre las clases de equivalencias inválidas: 5

CP4		
Username	Password	Password Verificar
*****	*****	*****

Cubre las clases de equivalencias inválidas: 9

Interfaz: Formulario (ABM) Zona

Nombre Alfabético 100 caracteres

Descripción Alfanumérico 300 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=100	4_ Caracteres >100
Tipo: Descripción	5_ Alfanumérico	7_ Caracteres >300
Tamaño: Descripción	6_ 0<=Caracteres<=300	

Tabla 222. Partición Equivalente. Interfaz: Formulario (ABM) Zona

Casos de Prueba válidas

CP1	
Nombre	Descripción
CENTRAL	

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario (ABM) Asegurado con Tipo de Seguro (FormAVC04A)

Nro. Formulario Numérico 25 caracteres

Nombre Alfabético 70 caracteres

Ap. Paterno Alfabético 50 caracteres

Ap. Materno Alfabético 50 caracteres

Numero Asegurado Numérico 25 caracteres

Fecha Nac. Date

Sexo Alfabético 10 caracteres

Edad Numérico

Estado Civil Alfabético 50 caracteres

Documento Identidad Alfabético 25 caracteres

Numero Numérico

Numero Numérico

Calle Alfanúmero 100 caracteres

Localidad Alfabético 50 caracteres

Numero Municipio Numérico
Lugar Alfabético 50 caracteres
Fecha Date
Fecha Recepción Date
Observaciones Alfabético150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Nombre	5_ Alfabético	7_ Vacío
Tamaño: Nombre	6_ 1<=Caracteres<=70	8_ Caracteres >70
Tipo: Ap. Paterno	9_ Alfabético	11_ Cualquier otra cosa
Tamaño Ap. Paterno	10_ 0<=Caracteres<=50	
Tipo: Ap. Materno	12_ Alfabético	14_ Vacío
Tamaño: Ap. Materno	13_ 1<=Caracteres<=50	15_ Caracteres >50
Tipo: Numero Asegurado	16_ Numérico	18_ Vacío
Tamaño: Numero Asegurado	17_ 1<=Caracteres<=25	29_ Caracteres >25
Tipo: Fecha Nac.	20_ Date	21_ Vacío
Tipo: Sexo	22_ Alfabético	24_ Vacío
Tamaño: Sexo	23_ 1<=Caracteres<=10	25_ Caracteres >10
Tipo: Edad	26_ Numérico	27_ Vacío
Tipo: Estado Civil	28_ Alfabético	30_ Vacío
Tamaño: Estado Civil	29_ 1<=Caracteres<=50	31_ Caracteres >50
Tipo: Documento Identidad	32_ Alfabético	34_ Vacío
Tamaño: Documento Identidad	33_ 1<=Caracteres<=25	35_ Caracteres >25
Tipo: Numero	36_ Numérico	37_ Vacío
Tipo: Numero	38_ Numérico	39_ Vacío
Tipo: Calle	40_ Alfanumérico	42_ Vacío
Tamaño: Calle	41_ 1<=Caracteres<=100	43_ Caracteres >100
Tipo: Localidad	44_ Alfabético	46_ Vacío
Tamaño: Localidad	45_ 1<=Caracteres<=50	47_ Caracteres >50
Tipo: Numero Municipio	48_ Numérico	49_ Vacío
Tipo: Lugar	50_ Alfabético	52_ Vacío
Tamaño: Lugar	51_ 1<=Caracteres<=50	53_ Caracteres >50
Tipo: Fecha	54_ Date	55_ Vacío
Tipo: Fecha Recepción	56_ Date	57_ Vacío
Tipo: Observaciones	58_ Alfanumérico	60_ Caracteres >150

Tamaño: Observaciones	59_0<=Caracteres<=150
-----------------------	-----------------------

Tabla 223. Partición Equivalente. Interfaz: Formulario (ABM) Asegurado con Tipo de Seguro (FormAVC04A)

Casos de Prueba válidas

CPI																		
Nro. Formulario	Nom bre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
8	ROBE RTO	MEND IETA	ARTE AGA	333-333-333	10/10/1973	MASCU LINO	37	CAS ADO	CI	333	33	CA LLE LA PAZ	CIUDA D TARIJ A	123	TAR IJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-12-13-16-17-20-22-23-26-28-29-32-33-36-38-40-41-44-45-48-50-51-54-56-58-59

Casos de Pruebas Inválidas

CP2																		
Nro. Formulario	Nom bre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
	ROBE RTO	MEND IETA	ARTE AGA	333-333-333	10/10/1973	MASCU LINO	37	CAS ADO	CI	333	33	CA LLE LA PAZ	CIUDA D TARIJ A	123	TAR IJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 3

CP3																		
Nro. Formulario	Nom bre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
8		MEND IETA	ARTE AGA	333-333-333	10/10/1973	MASCU LINO	37	CAS ADO	CI	333	33	CA LLE LA PAZ	CIUDA D TARIJ A	123	TAR IJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas 7

CP4																		
Nro. Formulario	Nom bre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones

8	ROBERTO	JAE	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	
---	---------	-----	---------	-------------	------------	-----------	----	--------	----	-----	----	--------------	---------------	-----	--------	------------	------------	--

Cubre las clases de equivalencias inválidas: 11

CP5																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Aseguro	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Munici pio	Lugar	Fecha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA		333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 14

CP6																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Aseguro	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Munici pio	Lugar	Fecha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA		10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 18

CP7																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Aseguro	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Munici pio	Lugar	Fecha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333		MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 21

CP8																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Aseguro	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Munic ipio	Lugar	Fecha	Fecha Recep ción	Observa ciones

8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973		37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	
---	---------	----------	---------	-------------	------------	--	----	--------	----	-----	----	--------------	---------------	-----	--------	------------	------------	--

Cubre las clases de equivalencias inválidas: 24

CP9																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO		CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 27

CP 10																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37		CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 30

CP 11																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO		333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 34

CP 12																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Nume ro Asegu rado	Fech a Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Num ero	Num ero	Cal le	Local idad	Nume ro Muni cipio	Lu gar	Fech a	Fecha Recep ción	Observa ciones

8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI		33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	
---	---------	----------	---------	-------------	------------	-----------	----	--------	----	--	----	--------------	---------------	-----	--------	------------	------------	--

Cubre las clases de equivalencias inválidas: 37

CP 13																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Edad	Estado Civil	Documento Identidad	Numero	Numero	Calle	Localidad	Numero Municipio	Lugar	Fecha	Fecha Recepción	Observaciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333		CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 39

CP 14																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Edad	Estado Civil	Documento Identidad	Numero	Numero	Calle	Localidad	Numero Municipio	Lugar	Fecha	Fecha Recepción	Observaciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33		CIUDAD TARIJA	123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 42

CP 15																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Edad	Estado Civil	Documento Identidad	Numero	Numero	Calle	Localidad	Numero Municipio	Lugar	Fecha	Fecha Recepción	Observaciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ		123	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 46

CP 16																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Edad	Estado Civil	Documento Identidad	Numero	Numero	Calle	Localidad	Numero Municipio	Lugar	Fecha	Fecha Recepción	Observaciones

8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA		TARIJA	30/07/2010	20/08/2010	
---	---------	----------	---------	-------------	------------	-----------	----	--------	----	-----	----	--------------	---------------	--	--------	------------	------------	--

Cubre las clases de equivalencias inválidas: 49

CP 17																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Muni cipio	Lu gar	Fecha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123		30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 52

CP 18																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Muni cipio	Lug ar	Fec ha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA		20/08/2010	

Cubre las clases de equivalencias inválidas: 55

CP 19																		
Nro. Formulario	Nombre	Ap. Pater no	Ap. Mate rno	Numero Asegurado	Fecha Nac.	Sexo	Ed ad	Esta do Civi l	Docu mento Identid ad	Numero	Numero	Cal le	Local idad	Numero Muni cipio	Lu gar	Fecha	Fecha Recep ción	Observa ciones
8	ROBERTO	MENDIETA	ARTEAGA	333-333-333	10/10/1973	MASCULINO	37	CASADO	CI	333	33	CALLE LA PAZ	CIUDAD TARIJA	123	TARIJA	30/07/2010		

Cubre las clases de equivalencias inválidas: 57

Interfaz: Formulario Baja Asegurado (FormAVC07A)

Nro. Formulario Numérico 25 caracteres

Fecha Baja. Date

Motivo Baja 250 caracteres

Lugar Alfabético 50 caracteres

Fecha Presentación Date

Fecha Recepción Date

Observaciones Alfabético 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Fecha Baja	5_ Date	6_ Vacío
Tipo: Motivo Baja	7_ Alfanumérico	9_ Vacío
Tamaño: Motivo Baja	8_ 1<Caracteres<250	10_ Caracteres >250
Tipo: Lugar	11_ Alfabético	13_ Vacío
Tamaño: Lugar	12_ 1<=Caracteres<=50	14_ Caracteres >50
Tipo: Fecha Presentación	15_ Date	16_ Vacío
Tipo: Fecha Recepción	17_ Date	18_ Vacío
Tipo: Observaciones	19_ Alfanumérico	21_ Caracteres >150
Tamaño: Observaciones	20_ 0<=Caracteres<=150	

Tabla 224. Partición Equivalente. Interfaz: Formulario Baja Asegurado (FormAVC07A)

Casos de Prueba válidas

CP1						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones
8	30/07/2010	RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias válidas: 1-5-7-11-15-17-19

Casos de Pruebas Inválidas

CP2						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones
	30/07/2010	RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 3

CP3						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones
8			TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 9

CP4						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones

8		RENUNCIA		30/07/2010	20/08/2010	
---	--	----------	--	------------	------------	--

Cubre las clases de equivalencias inválidas: 13

CP5						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones
8	30/07/2010	RENUNCIA	TARIJA		20/08/2010	

Cubre las clases de equivalencias inválidas: 16

CP5						
Nro. Formulario	Fecha Baja	Motivo Baja	Lugar	Fecha Presentación	Fecha Recepción	Observaciones
8	30/07/2010	RENUNCIA	TARIJA	30/07/2010		

Cubre las clases de equivalencias inválidas: 18

Interfaz: Formulario (ABM) Empresa (FormAVC01)

- Nro. Formulario Numérico 25 caracteres
- Nombre o Razón Social Empleador Alfanumérico 100 caracteres
- Numero Empleador Numérico
- CI Numérico 25 caracteres
- Nombre Alfabético 70 caracteres
- Ap. Paterno Alfabético 50 caracteres
- Ap. Materno Alfabético 50 caracteres
- Fecha Nac. Date
- Edad Numérico
- Numero Numérico
- Calle Alfanúmero 100 caracteres
- Teléfono Numérico 30 caracteres
- Fecha inicio Actividad Data
- Numero Trabajadores Numérico
- Nro. Padrón Renta Numérico
- Lugar Presentación Alfanumérico 50 caracteres
- Fecha Presentación Date
- Fecha Recepción Date
- Observaciones Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25

Tipo: Nombre o Razón Social Empleador	5_ Alfanumérico	7_ Vacío
Tamaño: Nombre o Razón Social Empleador	6_ 1<=Caracteres<=100	8_ Caracteres >100
Numero Empleador	9_ Numérico	10_ Vacío
Tipo: CI	11_ Numérico	13_ Vacío
Tamaño: CI	12_ 1<=Caracteres<=25	14_ Caracteres >25
Tipo: Nombre	15_ Alfabético	17_ Vacío
Tamaño: Nombre	16_ 1<=Caracteres<=70	18_ Caracteres >70
Tipo: Ap. Paterno	19_ Alfabético	21_ Cualquier otra cosa
Tamaño: Ap. Paterno	20_ 0<=Caracteres<=50	
Tipo: Ap. Materno	22_ Alfabético	24_ Vacío
Tamaño: Ap. Materno	23_ 1<=Caracteres<=50	25_ Caracteres >50
Tipo: Fecha Nac.	26_ Date	27_ Vacío
Tipo: Edad	28_ Numérico	29_ Vacío
Tipo: Numero	30_ Numérico	31_ Vacío
Tipo: Calle	32_ Alfanumérico	34_ Vacío
Tamaño: Calle	33_ 1<=Caracteres<=100	35_ Caracteres >100
Tipo: Teléfono	36_ Numérico	37_ Cualquier cosa
Tamaño Teléfono	1<= caracteres<=30	
Tipo: Fecha inicio Actividad	38_ Date	39_ Vacío
Tipo: Numero Trabajadores	40_ Numérico	41_ Vacío
Tipo: Nro.Padrón Renta Numérico	42_ Numérico	43_ Vacío
Tipo: Lugar Presentación	44_ Alfanumérico	46_ Vacío
Tamaño: Lugar Presentación	45_ 1<=Caracteres<=50	47_ Caracteres >50
Tipo: Fecha Presentación	48_ Date	49_ Vacío
Tipo: Fecha Recepción	50_ Date	51_ Vacío
Tipo: Observaciones	52_ Alfanumérico	54_ Caracteres >150
Tamaño Observaciones	53_ 0<=Caracteres<=150	

Tabla 225. Partición Equivalente. Interfaz: Formulario (ABM) Empresa (FormAVC01)

Casos de Prueba válidas

CP 1																		
Nro. Formulari	Nom bre o Razó	Num ero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici	Num ero Trab	Nro. Padr ón	Luga r prese	Fech a prese	Fech a Rece	Obse rvaci ones

o	n											o	ajado	Rent	ntaci	ntaci	pción	
9	SETA R	111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30	145	HERIB ERTO TRIG O	662389 4	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-11-12-15-16-19-20-22-23-26-28-30-32-33-36-38-40-42-44-45-48-50-52-53

Casos de Pruebas Inválidas

CP 2																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Actividad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntación	Fech a prese ntación	Fech a Rece pción	Obse rvaciones
	SETA R	111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30	145	HERIB ERTO TRIG O	662389 4	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias inválidas: 3

CP 3																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Actividad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntación	Fech a prese ntación	Fech a Rece pción	Obse rvaciones
9		111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30	145	HERIB ERTO TRIG O	662389 4	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias inválidas: 7

CP 4																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Actividad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntación	Fech a prese ntación	Fech a Rece pción	Obse rvaciones

	l											idad						
9	SETAR		66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60	7788922		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 10

CP 5																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111		ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60	7788922		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 13

CP 6																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111	66785695		RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60	7788922		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 17

CP 7																		
Nro. Formulario	Nom bre o Razón Social	Numero	CI	Nom bre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111	66785695	ALFREDO	RO@#%&	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60	7788922		20/08/2010	10/10/1973	

Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Numero	Calle	Teléf ono	Fech a Inici o Activ idad	Numero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntaci ón	Fech a prese ntaci ón	Fech a Rece pción	Obse rvaci ones
9	SETA R	111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30		HERIB ERTO TRIG O	662389 4	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias inválidas: 31

CP 12																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Numero	Calle	Teléf ono	Fech a Inici o Activ idad	Numero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntaci ón	Fech a prese ntaci ón	Fech a Rece pción	Obse rvaci ones
9	SETA R	111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30	145		662389 4	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias inválidas: 34

CP 13																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Numero	Calle	Teléf ono	Fech a Inici o Activ idad	Numero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntaci ón	Fech a prese ntaci ón	Fech a Rece pción	Obse rvaci ones
9	SETA R	111	667856 95	ALFR EDO	RODR IGUEZ	ARCE	10/10/ 1980	30	145	HERIB ERTO TRIG O	66IERI F@\$	10/10/ 2000	60	778892 2		20/08/ 2010	10/10/ 1973	

Cubre las clases de equivalencias inválidas: 37

CP 14																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Numero	Calle	Teléf ono	Fech a Inici o Activ idad	Numero Trab ajado res	Nro. Padr ón Rent a	Luga r prese ntaci ón	Fech a prese ntaci ón	Fech a Rece pción	Obse rvaci ones

	1											idad						
9	SETAR	111	66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894		60	7788922		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 39

CP 15																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111	66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000		7788922		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 41

CP 16																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111	66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60			20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 43

CP 17																		
Nro. Formulario	Nombre o Razón Social	Numero	CI	Nombre	Ap. Pater no	Ap. Mate rno	Fech a Nac.	Edad	Num ero	Calle	Teléf ono	Fech a Inici o Activ idad	Num ero Trab ajado res	Nro. Padr ón Rent a	Luga r presentaci ón	Fech a presentaci ón	Fech a Rece pci ón	Obse rvaci ones
9	SETAR	111	66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30	145	HERIBERTO TRIGO	6623894	10/10/2000	60	7788922		20/08/2010	10/10/1973	

Nro. Formulario Numérico 25 caracteres

Lugar Presentación Alfanumérico 50 caracteres

Fecha Presentación Date

Fecha Recepción Date

Observaciones Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Lugar Presentación	5_ Alfanumérico	7_ Vacío
Tamaño: Lugar Presentación	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo: Fecha Presentación	9_ Date	10_ Vacío
Tipo: Fecha Recepción	11_ Date	12_ Vacío
Tipo: Observaciones	13_ Alfanumérico	14_ Caracteres >150
Tamaño Observaciones	15_ 0<=Caracteres<=150	

Tabla 226. Partición Equivalente. Interfaz: Formulario Aviso de Novedades (FormAVC03)

Casos de Prueba válidas

CP 1				
Nro. Formulario	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	TARIJA	20/08/2010	10/10/1973	

Cubre las clases de equivalencias válidas: 1-5-9-11-13

Casos de Pruebas Inválidas

CP 2				
Nro. Formulario	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
	TARIJA	20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 3

CP 3				
Nro. Formulario	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9		20/08/2010	10/10/1973	

Cubre las clases de equivalencias inválidas: 7

CP 4				
Nro. Formulario	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	TARIJA		10/10/1973	

Cubre las clases de equivalencias inválidas: 10

CP 5				
Nro. Formulario	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
	TARIJA	20/08/2010		

Cubre las clases de equivalencias inválidas: 12

CI Numérico 25 caracteres

Nombre Alfabético 70 caracteres

Ap. Paterno Alfabético 50 caracteres

Ap. Materno Alfabético 50 caracteres

Fecha Nac. Date

Edad Numérico

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: CI	1_ Numérico	3_ Vacío
Tamaño: CI	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Nombre	5_ Alfabético	7_ Vacío
Tamaño: Nombre	6_ 1<=Caracteres<=70	8_ Caracteres >70
Tipo: Ap. Paterno	9_ Alfabético	
Tamaño: Ap. Paterno	10_ 0<=Caracteres<=50	

Tipo: Ap. Materno	11_ Alfabético	13_ Vacío
Tamaño: Ap. Materno	12_ 1<=Caracteres<=50	14_ Caracteres >50
Tipo: Fecha Nac.	15_ Date	16_ Vacío
Tipo: Edad	17_ Numérico	18_ Vacío

Tabla 227. Partición Equivalente. Interfaz: Formulario Nuevo Propietario o Representante Legal

Casos de Prueba válidas

CP 1					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30

Cubre las clases de equivalencias válidas: 1-5-9-11-15-17

Casos de Pruebas Inválidas

CP 2					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	30

Cubre las clases de equivalencias inválidas: 3

CP 3					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
66785695		RODRIGUEZ	ARCE	10/10/1980	30

Cubre las clases de equivalencias inválidas: 7

CP 4					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
66785695	ALFREDO	RODRIGUEZ		10/10/1980	30

Cubre las clases de equivalencias inválidas: 13

CP 5					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
66785695	ALFREDO	RODRIGUEZ	ARCE		30

Cubre las clases de equivalencias inválidas: 16

CP 6					
CI	Nombre	Ap. Paterno	Ap. Materno	Fecha Nac.	Edad
66785695	ALFREDO	RODRIGUEZ	ARCE	10/10/1980	

Cubre las clases de equivalencias inválidas: 18

Nombre o Nueva Razón Social Alfanumérico 100 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre o Nueva Razón	1_ Alfanumérico	3_ Vacío

Social Tamaño: Nombre o Nueva Razón Social	2_ 1<=Caracteres<=100	4_ Caracteres >100
---	-----------------------	--------------------

Tabla 228. Partición Equivalente. Interfaz: Formulario Nueva Razón Social

Casos de Prueba válidas

CP 1
Nombre o Razón Social
SETAR SRL.

Cubre las clases de equivalencias válidas: 1

Casos de Pruebas Inválidas

CP 2
Nombre o Razón Social

Cubre las clases de equivalencias inválidas: 3

The screenshot shows a web form titled 'Nuevo Domicilio Legal'. It contains four input fields: 'Zona' (a dropdown menu with 'CENTRAL' selected), 'Numero', 'Calle', and 'Telefono'. Each field has a light blue border and a small white box for text entry.

Numero Alfanúmero 20 caracteres

Calle Alfanúmero 100 caracteres

Teléfono Alfanúmero 30 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Numero	1_ Alfanumérico 2_ 1<=Caracteres<=20	3_ Vacío 4_ Caracteres >20
Tipo: Calle Tamaño: Calle	5_ Alfanumérico 6_ 1<=Caracteres<=100	7_ Vacío 8_ Caracteres >100
Tipo: Teléfono Tamaño Teléfono	9_ Alfanumérico 10_ 1<=Caracteres<=30	11_ Vacío 12_ Caracteres >30

Tabla 229. Partición Equivalente. Interfaz: Formulario Nuevo Domicilio Legal

Casos de Prueba válidas

CP 1		
Numero	Calle	Teléfono
145	HERIBERTO TRIGO	6623894

Cubre las clases de equivalencias válidas: 1-5-9

Casos de Pruebas Inválidas

CP 1

Numero	Calle	Teléfono
	HERIBERTO TRIGO	6623894

Cubre las clases de equivalencias inválidas: 3

CP 1		
Numero	Calle	Teléfono
145		6623894

Cubre las clases de equivalencias inválidas: 7

CP 1		
Numero	Calle	Teléfono
145	HERIBERTO TRIGO	

Cubre las clases de equivalencias inválidas: 11

Interfaz: Formulario (ABM-Reingreso) Asegurados (FormAVC04)

Nro. Formulario Numérico 25 caracteres

Documento Identidad Alfanumérico 25 caracteres

Nombre Alfabético 70 caracteres

Ap. Paterno Alfabético 50 caracteres

Ap. Materno Alfabético 50 caracteres

Numero Asegurado Numérico

Fecha Nacimiento Date

Edad Numérico

Calle Alfanúmero 100 caracteres

Numero Numérico

Numero Municipio Numérico

Salario Mensual Numérico

Fecha Ingreso Trabajo Data

Número del Empleador Numérico

Lugar Presentación Alfanumérico 50 caracteres

Fecha Presentación Date

Fecha Recepción Date

Observaciones Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formularios	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Documento Identidad	5_ Alfanumérico	7_ Vacío
Tamaño: Documento Identidad	6_ 1<=Caracteres<=25	8_ Caracteres >25

Tipo: Nombre	9_ Alfabético	11_ Vacío
Tamaño: Nombre	10_ 1<=Caracteres<=70	12_ Caracteres >70
Tipo: Ap. Paterno	13_ Alfabético	15_ Cualquier otra cosa
Tamaño: Ap. Paterno	14_ 0<=Caracteres<=50	
Tipo: Ap. Materno	16_ Alfabético	18_ Vacío
Tamaño: Ap. Paterno	17_ 1<=Caracteres<=50	19_ Caracteres >50
Tipo:: Numero Asegurado	20_ Numérico	21_ Vacío
Tipo: Fecha Nac.	22_ Date	23_ Vacío
Tipo: Edad	24_ Numérico	25_ Vacío
Tipo: Calle	26_ Alfanumérico	28_ Vacío
Tamaño: Calle	27_ 1<=Caracteres<=100	29_ Caracteres >100
Tipo: Numero	30_ Numérico	31_ Vacío
Tipo: Numero Municipio	32_ Numérico	33_ Vacío
Tipo: Salario Mensual	34_ Numérico	35_ Vacío
Tipo: Fecha Ingreso Trabajo	36_ Date	37_ Vacío
Tipo: Numero del Empleador	38_ Numérico	39_ Vacío
Tipo: Lugar Presentación	40_ Alfanumérico	42_ Vacío
Tipo: Lugar Presentación	41_ 1<=Caracteres<=50	43_ Caracteres >50
Tipo: Fecha Presentación	44_ Date	45_ Vacío
Tipo: Fecha Recepción	46_ Date	47_ Vacío
Tipo: Observaciones	48_ Alfanumérico	50_ Caracteres >150
Tamaño: Observaciones	49_ 0<=Caracteres<=150	

Tabla 230. Partición Equivalente. Interfaz: Formulario (ABM-Reingreso) Asegurados (FormAVC04)

Casos de Prueba válidas

CP 1																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Paterno	Ap. Materno	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Emplador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA	ORDONES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-13-14-16-17-20-22-24-26-27-30-32-34-36-38-40-41-44-46-48-49

Casos de Pruebas Inválidas

CP 2																	
Nro.	Docu	Nom	Ap.	Ap.	Nume	Fecha	Edad	Calle	Nume	Nume	Salari	Fecha	Núme	Lugar	Fecha	Fecha	Obser

Formulario	Documento de identidad	Nombre	Paterno	Materno	Numero Asegurado	Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
	CEDULA	ISMAEL	SURITA	ORDONES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 3

CP 3																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Paterno	Ap. Materno	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9		ISMAEL	SURITA	ORDONES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 7

CP 4																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Paterno	Ap. Materno	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA		SURITA	ORDONES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 11

CP 5																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Paterno	Ap. Materno	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	ADJ@#	ORDONES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 15

CP 6																	
------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA		111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 18

CP 7																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA	ORDONES		10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 21

CP 8																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA	ORDONES	111		40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 23

CP 9																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empleador	Lugar presentación	Fecha presentación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA	ORDONES	111	10/10/1970		COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 25

CP 10																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40		365	332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 28

CP 11																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA		332	2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 31

CP 12																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365		2000	10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 33

CP 13																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Número del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332		10/10/1995	60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 35

CP 14																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332	2000		60	TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 37

CP 15																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995		TARIJA	20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 39

CP 16																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60		20/08/1998	10/10/2000	

Cubre las clases de equivalencias inválidas: 42

CP 17																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Rece pción	Obser vacio nes
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA		10/10/2000	

Cubre las clases de equivalencias inválidas: 45

CP 18																	
Nro. Formulario	Documento de identidad	Nombre	Ap. Pater no	Ap. Mater no	Numero Asegurado	Fecha Nac.	Edad	Calle	Numero	Numero Municipio	Salario Mensual	Fecha Ingreso Trabajo	Numero del Empl eador	Lugar prese ntación	Fecha prese ntación	Fecha Recepción	Observaciones
9	CEDULA	ISMAEL	SURITA	ORDOÑES	111	10/10/1970	40	COCHABAMBA	365	332	2000	10/10/1995	60	TARIJA	20/08/1998		

Cubre las clases de equivalencias inválidas: 47



A. P. Alfabético 50 caracteres

A. M. Alfabético 50 caracteres

Nombre Alfabético 70 caracteres

Fecha Nac. Date

Fecha Ext. Date

Código Numérico

Causas Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre	1_ Alfabético	3_ Vacío
Tamaño: Nombre	2_ 1<=Caracteres<=70	4_ Caracteres >70
Tipo: A. P.	5_ Alfabético	7_ Vacío
Tamaño: A. P.	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo: A. M.	9_ Alfabético	11_ Vacío
Tamaño: A. M.	10_ 1<=Caracteres<=50	12_ Caracteres >50

Tipo: Fecha Nac.	13_ Date	14_ Vacío
Tipo: Fecha Ext.	15_ Date	16_ Vacío
Tipo: Causas	17_ Alfanumérico	19_ Caracteres >150
Tamaño: Causas	18_ 0<=Caracteres<=150	

Tabla 231. Partición Equivalente. Interfaz: Formulario Beneficiario

Casos de Prueba válidas

CP 1					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
ISMAEL	SURITA	ORDOÑES	10/10/1990	10/10/2015	

Cubre las clases de equivalencias válidas: 1-5-8-12-14-16

Casos de Pruebas Inválidas

CP 2					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
	SURITA	ORDOÑES	10/10/1990	10/10/2015	

Cubre las clases de equivalencias inválidas: 3

CP 3					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
ISMAEL		ORDOÑES	10/10/1990	10/10/2015	

Cubre las clases de equivalencias inválidas: 7

CP 4					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
ISMAEL	SURITA		10/10/1990	10/10/2015	

Cubre las clases de equivalencias inválidas: 11

CP 5					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
ISMAEL	SURITA	ORDOÑES		10/10/2015	

Cubre las clases de equivalencias inválidas: 14

CP 6					
Nombre	A. P.	A. M.	Fecha Nac.	Fecha Ext.	Causas
ISMAEL	SURITA	ORDOÑES	10/10/1990		

Cubre las clases de equivalencias inválidas: 16

Interfaz: Formulario Dar de Baja Asegurado (FormAVC07)

Nro. Formulario Numérico 25 caracteres

Salario Mensual Numérico

Fecha Baja Date

Motivo Baja 250 caracteres

Lugar Presentación Alfabético 50 caracteres

Fecha Presentación Date

Fecha Recepción Date

Observaciones Alfabético 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Salario Mensual	5_ Numérico	6_ Vacío
Tamaño: Salario Mensual		
Tipo: Fecha Baja	7_ Date	8_ Vacío
Tipo: Motivo Baja	9_ Alfanumérico	11_ Vacío
Tamaño: Motivo Baja	10_ 1<Caracteres<250	12_ Caracteres >250
Tipo: Lugar Presentación	13_ Alfabético	15_ Vacío
Tamaño: Lugar Presentación	14_ 1<=Caracteres<=50	16_ Caracteres >50
Tipo: Fecha Presentación	17_ Date	18_ Vacío
Tipo: Fecha Recepción	19_ Date	20_ Vacío
Tipo: Observaciones	21_ Alfanumérico	23_ Caracteres >150
Tamaño: Observaciones	22_ 0<=Caracteres<=150	

Tabla 232. Partición Equivalente. Interfaz: Formulario Dar de Baja Asegurado (FormAVC07)

Casos de Prueba válidas

CP 1							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000	30/07/2010	RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias válidas: 1-5-7-11-15-17-19

Casos de Pruebas Inválidas

CP 2							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
	5000	30/07/2010	RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 3

CP 3							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8		30/07/2010	RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 6

CP 4							
------	--	--	--	--	--	--	--

Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000		RENUNCIA	TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 8

CP 5							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000	30/07/2010		TARIJA	30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 11

CP 6							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000	30/07/2010	RENUNCIA		30/07/2010	20/08/2010	

Cubre las clases de equivalencias inválidas: 15

CP 1							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000	30/07/2010	RENUNCIA	TARIJA		20/08/2010	

Cubre las clases de equivalencias inválidas: 18

CP 1							
Nro. Formulario	Salario Mensual	Fecha Baja	Motivo Baja	Lugar Presentación	Fecha Presentación	Fecha Recepción	Observaciones
8	5000	30/07/2010	RENUNCIA	TARIJA	30/07/2010		

Cubre las clases de equivalencias inválidas: 18

Interfaz: Formulario (ABM) Certificado Incapacidad Temporal (FormAVC09)

Nro. Formulario Numérico 25 caracteres

Incapacidad desde Date

Incapacidad hasta Date

Días incapacidad Numeral

Clave Numérico

Salario Numérico

Importe subsidio Numérico

Lugar Alfanumérico 50 caracteres

Fecha Date

Observaciones Alfanumerico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
----------------------	-----------------------------	-------------------------------

Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Incapacidad Desde	5_ Date	6_ Vacío
Tipo: Incapacidad Hasta	7_ Date	8_ Vacío
Tipo: Días Incapacidad	9_ Numérico	10_ Vacío
Tipo: Clave	11_ Numérico	12_ Vacío
Tipo: Salario	13_ Numérico	14_ Vacío
Tipo: Importe Subsidio	15_ Numérico	16_ Vacío
Tipo: Lugar	17_ Alfanumérico	19_ Vacío
Tamaño: Lugar	18_ 1<=Caracteres<=50	20_ Caracteres >50
Tipo: Fecha	21_ Date	22_ Vacío
Tipo: Observaciones	23_ Alfanumérico	25_ Caracteres >150
Tamaño: Observaciones	24_ 0<=Caracteres<=150	

Tabla 233. Partición Equivalente. Interfaz: Formulario (ABM) Certificado Incapacidad Temporal (FormAVC09)

Casos de Prueba válidas

CP 1									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20	1456	2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias válidas: 1-2-5-7-9-11-13-15-17-18-21-23-24

Casos de Pruebas Inválidas

CP 2									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
	10/10/2010	30/10/210	20	1456	2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 3

CP 3									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12		30/10/210	20	1456	2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 6

CP 4									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010		20	1456	2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 8

CP 5									
------	--	--	--	--	--	--	--	--	--

Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210		1456	2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 10

CP 6									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20		2000	150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 12

CP 7									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20	1456		150	TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 14

CP 8									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20	1456	2000		TARIJA	09/10/2010	

Cubre las clases de equivalencias inválidas: 16

CP 9									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20	1456	2000	150		09/10/2010	

Cubre las clases de equivalencias inválidas: 19

CP 10									
Nro. Formulario	Incapacidad Desde	Incapacidad Hasta	Días Incapacidad	Clave	Salario	Importe Subsidio	Lugar	Fecha	Observaciones
12	10/10/2010	30/10/210	20	1456	2000	150	TARIJA		

Cubre las clases de equivalencias inválidas: 22

Interfaz: Formulario (ABM) Autorización Médica Externa (FormAVC08)

Nro. Formulario Numérico 25 caracteres

Regional de Adscripción Alfanumérico 50 caracteres

Salario Numérico

Atención Médica En Alfanumérico 150 caracteres

Lugar Alfanumérico 50 caracteres

Fecha Date

Observaciones Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacio
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Regional de Adscripción	5_ Alfanumérico	7_ Vacio
Tamaño: Regional de Adscripción	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo: Salario	9_ Numérico	10_ Vacio
Tipo: Atención Médica En	11_ Alfanumérico	13_ Vacio
Tamaño: Atención Médica En	12_ 1<=Caracteres<=150	14_ Caracteres >150
Tipo: Lugar	15_ Alfanumérico	17_ Vacio
Tamaño: Lugar	16_ 1<=Caracteres<=50	18_ Caracteres >50
Tipo: Fecha	19_ Date	20_ Vacio
Tipo: Observaciones	21_ Alfanumérico	23_ Caracteres >150
Tamaño: Observaciones	22_ 0<=Caracteres<=150	

Tabla 234. Partición Equivalente. Interfaz: Formulario (ABM) Autorización Médica Externa (FormAVC08)

Casos de Prueba válidas

CP 1						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15	HOSPITAL OBRERO	2500	CLÍNICA SANTA LUCIA	TARIJA	10/09/2005	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-11-12-15-16-19

Casos de Pruebas Inválidas

CP 2						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
	HOSPITAL OBRERO	2500	CLÍNICA SANTA LUCIA	TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 3

CP 3						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15		2500	CLÍNICA SANTA LUCIA	TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 7

CP 4						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15	HOSPITAL OBRERO		CLÍNICA SANTA LUCIA	TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 10

CP 5						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15	HOSPITAL OBRERO	2500		TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 13

CP 6						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15	HOSPITAL OBRERO	2500	CLÍNICA SANTA LUCIA		10/09/2005	

Cubre las clases de equivalencias inválidas: 17

CP 7						
Nro. Formulario	Regional de Adscripción	Salario	Atención Médica en	Lugar	Fecha	Observaciones
15	HOSPITAL OBRERO	2500	CLÍNICA SANTA LUCIA	TARIJA		

Cubre las clases de equivalencias inválidas: 20

Interfaz: Formulario (ABM) Altas y Bajas Beneficiarios (FormAVC06)

Nro. Formulario Numérico 25 caracteres

Lugar Alfanumérico 50 caracteres

Fecha Date

Observaciones Alfanumérico 150 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nro. Formulario	1_ Numérico	3_ Vacío
Tamaño: Nro. Formulario	2_ 1<=Caracteres<=25	4_ Caracteres >25
Tipo: Lugar	5_ Alfanumérico	7_ Vacío
Tamaño: Lugar	6_ 1<=Caracteres<=50	8_ Caracteres >50
Tipo: Fecha	9_ Date	10_ Vacío
Tipo: Observaciones	11_ Alfanumérico	13_ Caracteres >150
Tamaño: Observaciones	12_ 0<=Caracteres<=150	

Tabla 235. Partición Equivalente. Interfaz: Formulario (ABM) Altas y Bajas Beneficiarios (FormAVC06)

Casos de Prueba válidas

CP 1			
Nro. Formulario	Lugar	Fecha	Observaciones
1456	TARIJA	10/09/2005	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-11-12

Casos de Pruebas Inválidas

CP 2			
Nro. Formulario	Lugar	Fecha	Observaciones
	TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 3

CP 3			
Nro. Formulario	Lugar	Fecha	Observaciones
1456		10/09/2005	

Cubre las clases de equivalencias inválidas: 7

CP 4			
Nro. Formulario	Lugar	Fecha	Observaciones
1456	TARIJA		

Cubre las clases de equivalencias inválidas: 10

Interfaz: Formulario (ABM) Backup

Usuario de la Base de Datos Alfanumérico 250 caracteres

Clave de la Base de Datos Alfanumérico 250 caracteres

Nombre del archivo Alfanumérico 250 caracteres

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Usuario de la BDD	1_ Alfanumérico	3_ Vacío
Tamaño: Usuario de la BDD	2_ 1<=Caracteres<=250	4_ Caracteres >250
Tipo: Clave de la BDD	5_ Alfanumérico	7_ Vacío
Tamaño: Clave de la BDD	6_ 1<=Caracteres<=250	8_ Caracteres >250
Tipo: Nombre del archivo	1_ Alfanumérico	3_ Vacío
Tamaño: Nombre del Archivo	2_ 1<=Caracteres<=250	4_ Caracteres >250

Tabla 236. Partición Equivalente. Interfaz: Formulario (ABM) Backup

Casos de Prueba válidas

CP 1			
Nro. Formulario	Lugar	Fecha	Observaciones
1456	TARIJA	10/09/2005	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-11-12

Casos de Pruebas Inválidas

CP 2			
Nro. Formulario	Lugar	Fecha	Observaciones
	TARIJA	10/09/2005	

Cubre las clases de equivalencias inválidas: 3

2.1.2.2.3.13.5.2 Prueba de Perfil de Desempeño

El componente principal de JMeter es denominado *Plan de Prueba* o *Test Plan*, en él se definen todos los aspectos relacionados con una prueba de carga, como : parámetros empleados por requisición, tipo de reportes a generarse con los resultados obtenidos, la posible reutilización de requisiciones compuestas por usuarios, entre otros aspectos.

A continuación se ilustra paso a paso un *Plan de Prueba* utilizado para simular una carga de 50 usuarios solicitando la página principal en determinado sitio.

Estando en la interface principal de JMeter, en la columna izquierda debe observar un icono llamado Test Plan -- selecciónelo -- al llevar a cabo este paso, en la ventana derecha aparecerán varias opciones, aquella encontrada en la parte superior le permite asignar un nombre a su plan de prueba, defínalo a su criterio. El resto de las opciones representan funcionalidades avanzadas que no serán descritas para esta simulación.

Definido el nombre, colóquese nuevamente en el icono Test Plan y oprima el botón derecho de su "mouse", del menú generado seleccione la opción Add -- Thread Group; para efectos prácticos un "Thread Group" es considerado el grupo de usuarios que desea simular para su aplicación.

Ahora seleccione el icono Thread Group recientemente creado. Al llevar a cabo este paso la ventana derecha mostrará la siguiente serie de opciones:

- name: utilizado para definir un nombre más descriptivo sobre el grupo de usuarios, como : "Usuarios Esperados".
- number of threads: equivale al número de usuarios que se desean simular, en este caso utilizaremos 50.
- Ramp-up period: es el lapso de tiempo en segundos que se desea tener entre cada grupo de usuarios ("Thread Group"), utilizaremos 15.
- Forever: utilizado para indicar si la simulación para grupos de usuarios ("Thread Group") será llevada a cabo infinitamente, esto es, si selecciona esta opción indica que desea simular 50 usuarios, esperar 15 segundos ("Ramp-up period"), simular otros 50 usuarios y así sucesivamente. Para esta prueba es recomendable defina 10 ciclos para simular un total de 500 usuarios en cuestión de 150 segundos.
- Scheduler: finalmente, esta opción permite definir tiempos de arranque para determinados grupos de usuarios ("Thread Group"), para efectos prácticos de esta simulación no será seleccionada esta opción y se iniciará la prueba a nuestra discreción.

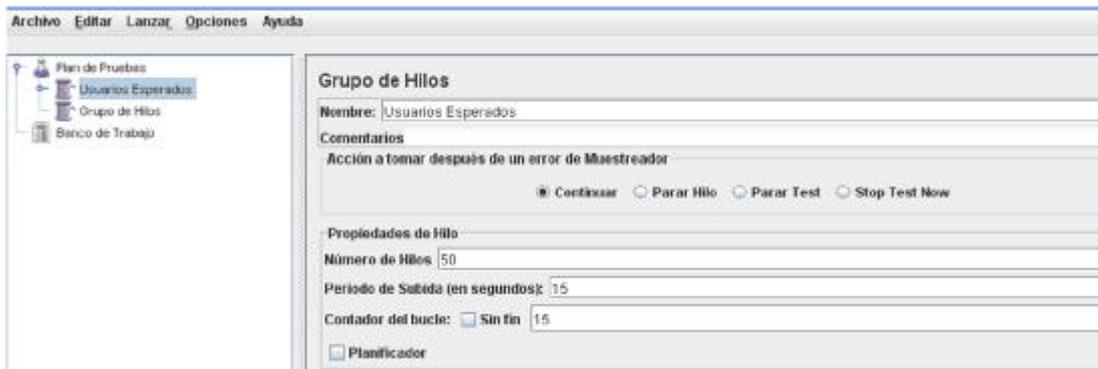


Figura 462. Grupo de Hilos: Usuarios Esperados

Una vez definidas las características del grupo de usuarios ("Thread Group"), colóquese nuevamente en este icono de la columna izquierda y ahora seleccione la opción Add -- Sampler -- HTTP Request. Lo anterior genera un icono denominado HTTP Request utilizado para definir las requisiciones de simulación. Si selecciona este último icono aparecerán las siguientes opciones en la ventana derecha:

- Server Name or IP: empleado para definir la dirección I.P o nombre del servidor donde será llevada a cabo la prueba de carga, se utilizará 127.0.0.1 para indicar un servidor local.
- port number: define el puerto TCP de operación del servidor, será empleado 8080.
- path: utilizado para definir la ruta de acceso para llevar a cabo la prueba, será definido /sima/index.html, que es la página principal de nuestro sistema web.

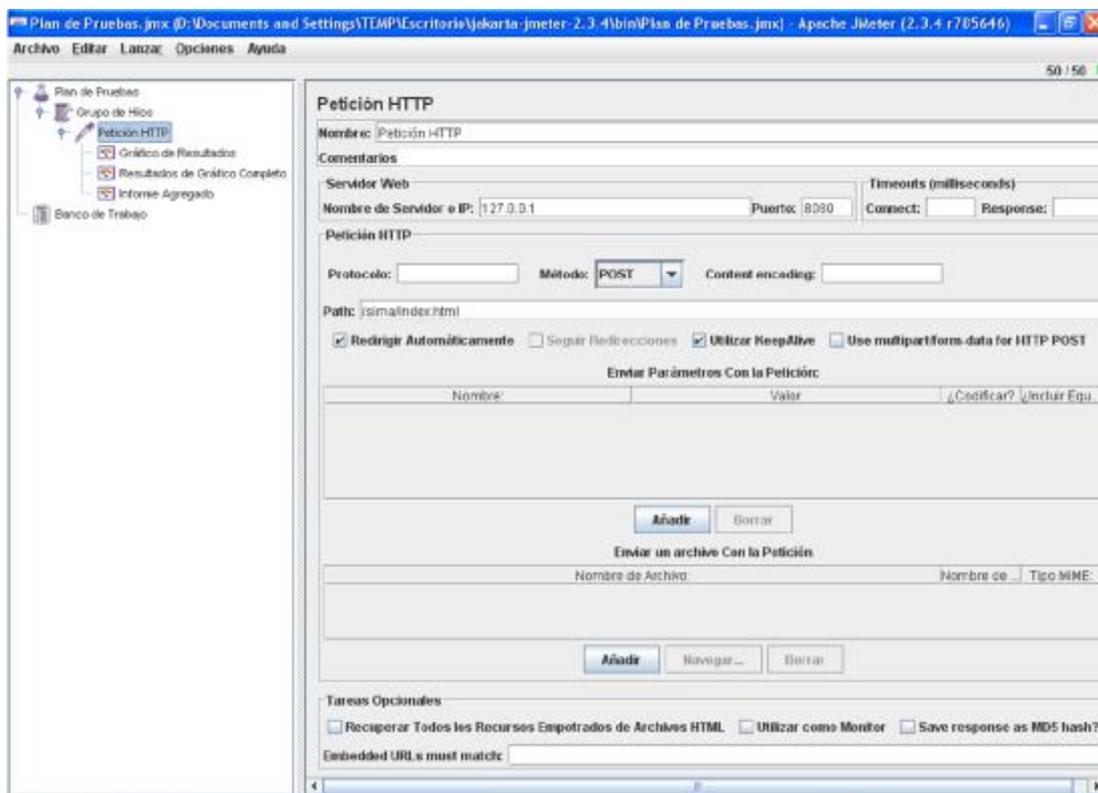


Figura 463. Petición HTTP

- Ahora debe colocarse en el icono de HTTP Request y oprimir el botón derecho de su "mouse", del menú generado seleccione la opción Add -- Listener -- Graph Results, así como Add -- Listener -- View Results in Table.
- Finalmente guarde su plan de prueba ("Plan Pruebas") y ejecútelo seleccionando la opción Run -- Start del menú superior. Si se coloca en el icono Graph Results mientras se está llevando a cabo la simulación, puede observar como es construida interactivamente la gráfica de resultados.



Figura 464. Grafico de Resultado

El Resultado Indica que la Página Web SIMA soporta 54,926 Usuarios en un minuto.

Label	# Muestras	Media	Mediana	Linea de 90%	Min	Máx	% Error	Rendimiento	Kb/sec
Pedición HTTP	211	10729	8754	18774	531	189858	0,00%	54,9/min	4,8
TOTAL	211	10729	8754	18774	531	189858	0,00%	54,9/min	4,8

Figura 465. Informe Agregado

2.1.2.2.3.13.5.3 Prueba de Pool de Conexiones

La prueba también se realiza con la herramienta JMeter, y su configuración es la siguiente:

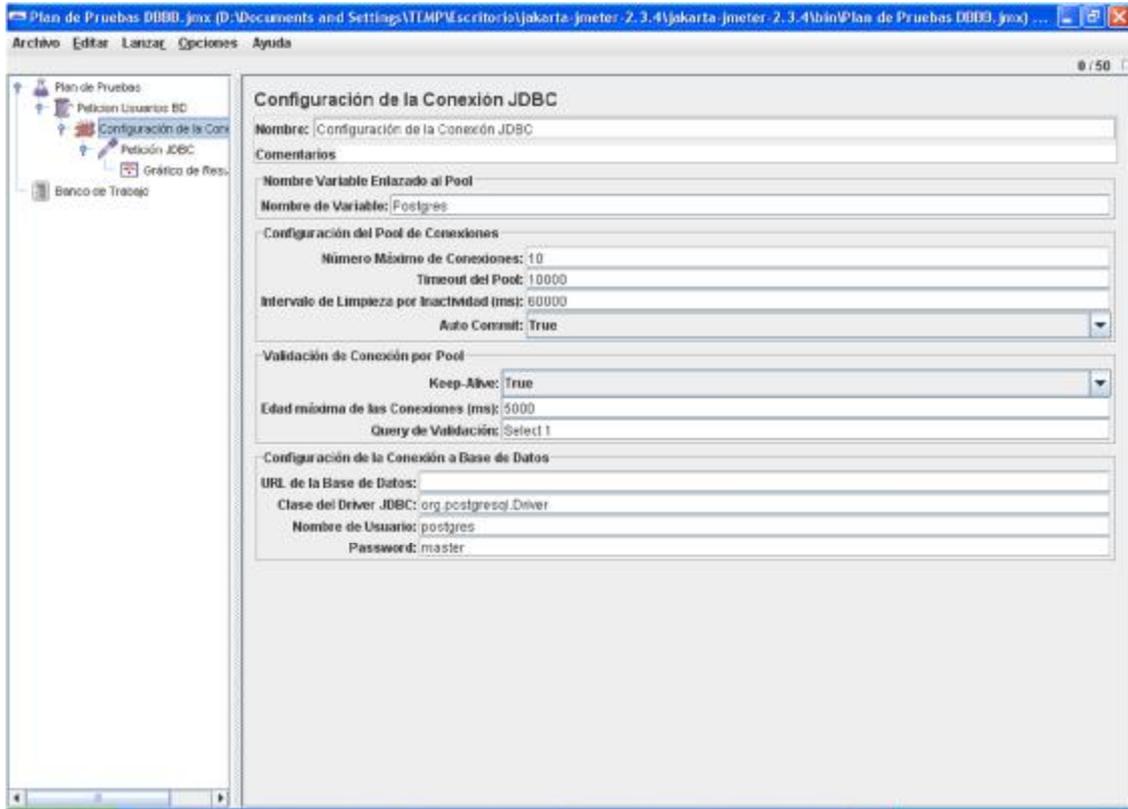


Figura 466. Configuración de la Conexión JDBC

En la cual se probará la siguiente consulta:



Figura 467. Petición JDBC

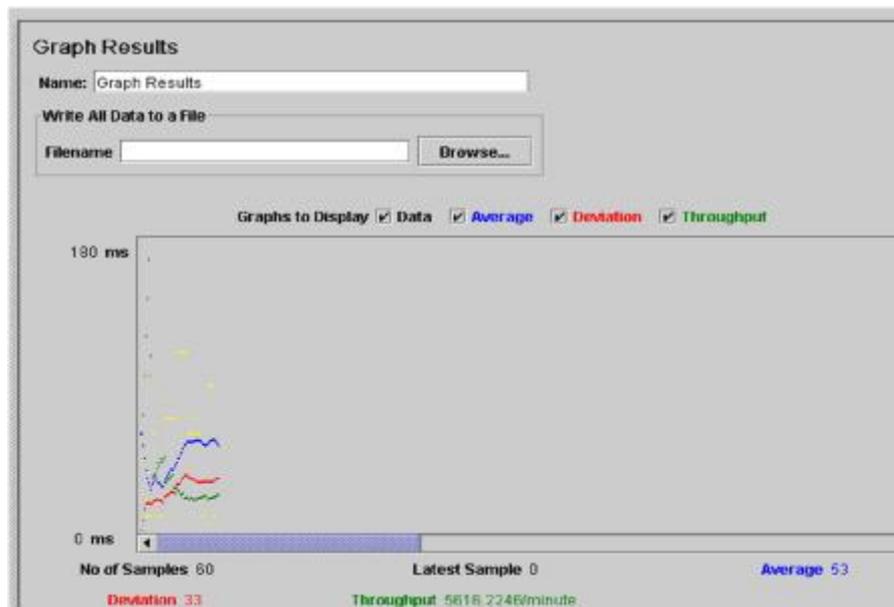


Figura 468. Grafico de Resultado JDBC

Por conexión se puede realizar la consulta a una velocidad de 5616.2246 consultas por minuto.

2.1.2.2.3.14 Solicitud de Cambio

2.1.2.2.3.14.1 Introducción

Cuando construimos software de computadora surgen cambios. Debido a esto, necesitamos controlarlos eficazmente, como un conjunto de actividades diseñadas para controlar el cambio identificando los productos del trabajo que probablemente cambien estableciendo relaciones entre ellos definiendo mecanismos para gestionar distintas versiones de estos productos controlando los cambios realizados y auditando e informando de los cambios realizados.

Los cambios propuestos para los artefactos se formalizan mediante este documento.

Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación de impacto, y se asegura que sean conocidos por el director del proyecto. Los cambios se establecen respecto de la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. En nuestro caso al final de cada iteración se establecerá una baseline o Línea Base.

2.1.2.2.3.14.2 Descripción

Las solicitudes de cambio se presentan de acuerdo a las presentaciones parciales de proyecto. Cada línea base presenta los siguientes campos:

Nombre: en este campo se anota el nombre de aquel elemento del cual se desea la modificación,

actualización o mejoramiento.

Tipo: se indica el tipo de documento del que se desea el cambio.

Fecha: se indica la fecha de solicitud del cambio.

Fase / Iteración: se indica la fase / iteración en la que se realizó el cambio.

Fecha de Actualización: se indica la fecha que se realizó el cambio.

Encargado: se indica el nombre del rol responsable de autorizar, realizar y verificar el cambio una vez finalizado este.

Informe del Cambio: se describe los resultados obtenidos provenientes de la modificación del artefacto.

2.1.2.2.3.14.3 Control de Solicitudes de Cambio

2.1.2.2.3.14.3.1 Primera Línea Base

Nro.	(Nombre) Elemento o Artefacto	Tipo	Fecha	Fase/Iteración	Fecha de Actualización	Encargado	Información del cambio
1	Documento de Desarrollo del Software.	Documento	14/04/2010	Inicio/1º Iteración	31/05/2010	Director de Proyecto	Se corrigió el alcance, propósito, y objetivos, complementando de acuerdo a las necesidades.
2	BPMN.	Documento	14/04/2010	Inicio/1º Iteración	31/05/2010	Director de Proyecto	Se redefinió los modelados de procesos del negocio incorporando nuevos procesos.
3	Modelo de Casos de Uso del Negocio	Documento	14/04/2010	Inicio/1º Iteración	31/05/2010	Director de Proyecto	Se redefinió modelo de casos de uso del negocio desde la perspectiva de los actores externos del sistema, usuarios.
4	Modelo de Objetos del Negocio	Documento	14/04/2010	Inicio/1º Iteración	31/05/2010	Director de Proyecto	Se realizaron las correcciones convenientes conforme a los modelos de casos de uso del negocio.
5	Especificación de Requisitos de Software	Documento	14/04/2010	Inicio/1º Iteración	31/05/2010	Director de Proyecto	Se realizaron los cambios incorporados a los requerimientos.

Tabla 237. Primera Línea Base

2.1.2.2.3.14.3.2 Segunda Línea Base

Nro.	(Nombre) Elemento o Artefacto	Tipo	Fecha	Fase/Iteración	Fecha de Actualización	Encargado	Información del cambio
1	Documento de Desarrollo del Software.	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se corrigió el alcance, propósito, y objetivos, complementando de

							acuerdo a las necesidades.
2	Modelo de Casos de Uso del Negocio	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se redefinió los modelados de procesos del negocio incorporando nuevos procesos.
3	Modelo de Objetos del Negocio	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se redefinió modelo de casos de uso del negocio desde la perspectiva de los actores externos del sistema, usuarios.
4	Especificación de Requisitos de Software	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se realizaron las correcciones convenientes conforme a los modelos de casos de uso del negocio.
5	Diagrama de Clases	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se redefinió el diagrama de clases de acuerdo a los nuevos requerimientos incorporados.
6	Modelo Entidad/Relación	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se redefinió el modelo entidad / relación de acuerdo a los nuevos requerimientos incorporados.
7	Glosario	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se corrigió los términos empleados en el proyecto de acuerdo desde el punto de vista administrativo.
8	Visión	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se corrigió la visión del producto desde la perspectiva del cliente.
9	Casos de Uso del Sistema	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se definió el sistema desde la perspectiva de

							los actores externos del sistema y se agregaron nuevos casos de uso para una mejor especificación del sistema.
10	Especificación de Casos de Uso	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se realizaron las correcciones convenientes conforme a la nueva descripción de los casos de uso.
11	Prototipo de Interfaces de Usuario	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se definió nuevas interfaces externas para un mejor manejo e interactividad con el usuario.
12	Diagramas de Actividades	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se detalló la realización de cada diagrama de casos de uso nuevamente, donde se reflejan los actores, la información que se maneja y los flujos de trabajo en los nuevos diagramas de actividades corregidos.
13	Diagramas de Secuencia	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el modelo de Actividad y los diagramas de casos de uso.
14	Diagramas de Colaboración	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el modelo de Secuencia.
15	Gestión de Riesgos	Documento	31/05/2010	Elaboración/1º Iteración	03/08/2010	Director de Proyecto	Se actualizó y corrigió el documento gestión de riesgos, haciendo una mayor especificación de

							cada riesgo y estableciendo de mejor manera las acciones preventivas.
--	--	--	--	--	--	--	---

Tabla 238. Segunda Línea Base

2.1.2.2.3.14.3.3 Tercera Línea Base

Nro.	(Nombre) Elemento o Artefacto	Tipo	Fecha	Fase/Iteración	Fecha de Actualización	Encargado	Información del cambio
1	Documento de Desarrollo del Software.	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se corrigió el alcance, propósito, y objetivos, complementando de acuerdo a las necesidades.
2	Especificación de Requisitos de Software	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizaron los cambios incorporados a los requerimientos.
3	Diagrama de Clases	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se redefinió el diagrama de clases de acuerdo a los nuevos requerimientos incorporados i/o a los cambios incorporados en la programación.
4	Modelo Entidad/Relación	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se redefinió el modelo entidad / relación de acuerdo a los nuevos requerimientos incorporados.
5	Diccionario de Datos	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el modelo entidad / relación.
6	Especificación de Métodos del Diagrama de Clases	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizaron las correcciones convenientes conforme

							al diagrama de clases.
7	SQL de la Base de Datos	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se redefinió el SQL de la base de Datos de acuerdo a los nuevos requerimientos incorporados.
8	Glosario	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se corrigió los términos empleados en el proyecto de acuerdo desde el punto de vista tecnológico.
9	Casos de Uso del Sistema	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se definió el sistema desde la perspectiva de los actores externos del sistema y se agregaron nuevos casos de uso para una mejor especificación del sistema.
10	Especificación de Casos de Uso	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se realizaron las correcciones convenientes conforme a la nueva descripción de los casos de uso.
11	Prototipo de Interfaces de Usuario	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se definió nuevas interfaces externas para un mejor manejo e interactividad con el usuario.
12	Diagramas de Actividades	Documento	03/08/2010	Construcción/1° Iteración	04/10/2010	Director de Proyecto	Se detalló la realización de cada diagrama de casos de uso nuevamente, donde se reflejan los actores, la información que se maneja y los flujos de trabajo en los nuevos diagramas de actividades

							corregidos.
13	Diagramas de Secuencia	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el modelo de Actividad y los diagramas de casos de uso.
14	Diagramas de Colaboración	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el modelo de Secuencia.
15	Diagramas de Componente	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se realizó conforme a los cambios realizados en el desarrollo de los archivos generados del código fuente.
16	Gestión de Riesgos	Documento	03/08/2010	Construcción/1º Iteración	04/10/2010	Director de Proyecto	Se actualizó y corrigió el documento gestión de riesgos, haciendo una mayor especificación de cada riesgo y estableciendo de mejor manera las acciones preventivas.

Tabla 239. Tercera Línea Base

2.1.2.2.3.14.3.4 Control de Versiones del Sistema

Nro.	(Nombre) Elemento o Artefacto	Tamaño	Fecha	Fase/Iteración	Fecha de Actualización	Versión	Línea Base
1	Prototipo Sistema	960 KB	20/06/2010	Construcción/1º Iteración	03/08/2010	0.2	Si
2	Prototipo Sistema	1,52 MB	04/08/2010	Transición/1º Iteración	04/10/2010	0.3	No

Tabla 240. Control de versiones del sistema

2.1.2.2.3.15 Lista de Riesgos

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

2.1.2.2.3.15.1 Introducción

2.1.2.2.3.15.1.1 Alcance

Este plan es emprendido ante la fase de elaboración para asegurar que ninguno de los riesgos identificados sea direccionado durante la misma fase de elaboración. Apenas el plan de gestión de riesgos haya sido documentado, el proceso de prevención de riesgos estará ocupado para monitorear y controlar la probabilidad y el impacto de los riesgos sobre el proyecto.

2.1.2.2.3.15.2 Proceso de Manipulación de Riesgos

En esta sección se establecen todos los aspectos que son necesarios para que cada riesgo que se pueda identificar dentro del proyecto en desarrollo sea debidamente documentado, evaluado y mitigado tanto como sea necesario. Para ello se establecen los procedimientos que en este proyecto deben ser ejecutados al momento de identificar un determinado riesgo, el mismo está acorde a las políticas de la organización.

2.1.2.2.3.15.2.1 Roles Encargados de Gestionar los Riesgos

Se establecen los roles participantes para el manejo de riesgos en su proyecto, los cuales van a ser los encargados de gestionar los riesgos y conforme a cada uno se señalan cuáles son sus funciones específicas.

Roles para el Manejo de Riesgos	Funciones
Jefe del Proyecto	Debe establecer un plan donde se debe contemplar los riesgos que sean identificados para el proyecto, adicionalmente dicho plan puede contener las descripciones, análisis, prioridades y estrategias que sirvan para minimizar el impacto que los riesgos puedan llegar a tener. Asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios. Mantiene al equipo del proyecto enfocado en los objetivos. Gestión de riesgos. Planificación y control del proyecto. Identificar y valorar preliminarmente los riesgos encontrados.

Tabla 241. Roles Encargados de Gestionar los Riesgos

2.1.2.2.3.15.3 Escala de Ponderación de los Riesgos

Se establecen un sistema de puntuación que sirva para establecer la calificación a los diversos riesgos, como propuesta tenemos:

Puntaje	Calificación	Descripción General
0% - 20%	Muy Bajo	Riesgo muy improbable, la probabilidad de ocurrencia es bajísima y es muy posible que no ocurra.
20% - 40%	Bajo	Riesgo de baja probabilidad de ocurrencia, es posible que la no ocurrencia de este.
40% - 60%	Moderado	Riesgo medianamente probable, puede ocurrir como puede no ocurrir.
60% - 80%	Alto	Riesgo muy probable, podría ocurrir, es muy probable la ocurrencia en el proceso de desarrollo.
80% - 100%	Muy alto	Riesgo altamente probable para ocurrir dado que las circunstancias que generarían este evento son muy probables.

Tabla 242. Escala de Ponderación de los Riesgos

2.1.2.2.3.15.4 Lista de Riesgos

Categoría de Riesgo			Identificación de Riesgos		Análisis de Riesgo		Planificación de Riesgos		Supervisión de Riesgos	
Proyecto	Producto	Negocio	N°	Riesgo	Ponderación	Efecto	Estrategias		Plan de contingencia	Políticas de Supervisión
							Anulación	Disminución o Preventiva		
*	*	*	1	La ausencia del Director del proyecto o que se desvíe del objetivo de terminar el proyecto.	20%	No poder llevar a cabo el proyecto y reprobar la materia.	-	*Ser consciente por sí mismo y trazarse el objetivo para poder llevar a su término el proyecto.	-	Realizar un control y seguimiento a los objetivos del proyecto.
	*		2	Que el personal cambie en la institución.	20%	Que no se brinde la misma ayuda al equipo de desarrollo.	-	*Conversaciones con el nuevo personal.	-	Realizar un control y seguimiento a posibles cambios del personal de la institución.
*	*		3	Los usuarios rehúsan utilizar el	20%	Los usuarios no quieren que	-	*En conversación con los	-	Realizar un control y seguimiento

				sistema.		su sistema sea automatizado.		usuarios finales, indicar las ventajas que presentara el software que se desarrollara. *Preparar un manual de usuario comprensible e que pueda estimular a los usuarios a usar el sistema a desarrollar.		a los usuarios.
	*		4	Los usuarios no tengan conocimiento en el manejo del sistema.	50%	El sistema sea mal utilizado o solo utilizado en parte.	-	*Aconseja al usuario interesado a cerca de capacitación para realizarse en el presente o futuro. *Prever para que el usuario final tenga el conocimiento o suficiente para poder manejar el sistema	-	Realizar un control y seguimiento a los usuarios en el manejo del sistema.
*	*	*	5	Realizar una mala estimación del tamaño del software.	50%	El producto no será entregado en la fecha acordada.	-	*Compara con otros proyectos al nuestro para tener una vista general de una estimación del tamaño del proyecto.	-	Realizar un control y seguimiento a las estimaciones del proyecto.

								*Realizar lectura de recursos bibliográficos disponibles.		
*	*	*	6	Realizar una mala estimación de los riesgos.	25%	El software no llegue a satisfacer las necesidades del usuario.	-	*Obtener un mayor conocimiento de las acciones se realizan en la empresa cliente, las acciones que realizara nuestro equipo de desarrollo y tener en claro el objetivo y el alcance de nuestro proyecto. *Obtener ayuda de expertos, libros y otros recursos que puedan ayudarnos en el desarrollo del plan de gestión de riesgos.	-	Realizar un control y seguimiento a los a los posibles riesgos.
*	*		7	No tener buen conocimiento en el manejo de las herramientas de trabajo.	30%	El software pueda tener fallas y no pueda ser entregado a tiempo.	-	*Utilizar herramientas conocidas. *Asegurarse que las personas miembros del equipo de	-	Realizar un control y seguimiento a las herramientas con las que se va a trabajar.

								desarrollo cuentan con el conocimiento necesario y suficiente acerca del uso de las herramientas a utilizar en el proyecto.		
*			8	Las Interfaces y el diseño de las pantallas no sean correctos.	30%	Las interfaces para el usuario lleguen a ser desagradables.	-	<p>*Establecer con anticipación las interfaces y el entorno gráfico a utilizar.</p> <p>*Hacer uso de herramientas que puedan otorgar una mejor apariencia a las interfaces.</p> <p>*Obtener conocimiento o de buenas normas para el diseño de interfaces.</p>	-	Realizar un control y seguimiento a las interfaces o pantallas que vayan de acuerdo a la facilidad del usuario.
*	*	*		No llegar a la culminación del sistema en la fecha indicada.	40%	La empresa quede perjudicada.	-	*Estimar la calendarización del proyecto haciendo uso de técnicas, métodos y herramientas adecuadas que nos faciliten una correcta	-	Realizar un control y seguimiento al calendario del proyecto.

								estimación.		
*	*	9	El coste del proyecto llegue a aumentarse durante el desarrollo.	25%	El proyecto puede cancelarse temporal o definitivamente, por el incremento del precio para el usuario.	-	-	*Hacer uso de técnicas de Ingeniería de Software para realizar una buena estimación del coste. *Obtener información actualizada del coste del desarrollo proyectos similares: personal, recursos de hardware y otros.	-	Realizar un control y seguimiento a los costes del proyecto.
	*	10	La institución no cuente con el Hardware y Software necesarios.	20%	El sistema no pueda ser implementado.	-	-	*Aconsejar y plantear equipo de hardware y recursos de software necesario para el funcionamiento del sistema con anticipación	-	Realizar un control y seguimiento al software y hardware de la institución.
*	*	11	El personal está enfermo y no disponible en momentos críticos en el desarrollo del proyecto.	40%	La empresa quede perjudicada y no se presente el proyecto en la fecha acordada (retraso).	-	-	-	-	-
*	*	12	Posible Falla en algún dispositivo de almacenamiento físico al resguardar de documentación del	15%	Perdida de toda la información avanzada del proyecto.	-	-	Realizar respaldos y copias de seguridad en dispositivos de almacenami	Acudir a software de recuperación de datos.	Establecer mantenimiento constante de los ordenadores en los que

				proyecto				ento externo.		se desarrolla el proyecto.
*	*		13	Se proponen cambios en los requerimientos que requieren rehacer el diseño.	60%	El producto no será entregado en la fecha acordada por los cambios a realizar.	-	*Revisar y hacer un análisis de cada uno de los requerimientos establecidos y llegar a un acuerdo con el cliente.	-	Realizar un control y seguimiento a cada uno de los requerimientos.

Tabla 243. Lista de Riesgos

2.1.2.2.4 Evolución del Plan de Desarrollo del Software

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

2.1.2.3 Organización el Proyecto

2.1.2.3.1 Participantes en el Proyecto

Director del Proyecto: Omar Elías Yana Cerezo

2.1.2.3.2 Interfaces Externas

El entorno gráfico del sistema está pensado en los usuarios, se tomará en cuenta la comodidad y la facilidad de manejo.

El sistema cuenta con una interfaz diseñada en base a módulos debido a que el sistema es muy complejo, requiere bastante precisión, y sobre todo para otorgar una mejor comunicación en el proceso de desarrollo entre el personal responsable del mismo.

Todas las funcionalidades del sistema requieren de una validación de seguridad debido a que se maneja información de afiliados. La tediosa carga de datos que se realiza bajo una conexión de Intranet, debe ser totalmente transparente para el Usuario, para esto la interfaz debe simular a una Interfaz de un Sistema de Escritorio.

El rendimiento de las interfaces nombradas con anterioridad recae sobre todo en el lenguaje a utilizar y la versión del mismo.

El sistema contará de manera general con:

- Pantalla de Acceso.
- Pantalla de Menú Principal.

2.1.2.3.3 Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

Puesto	Responsabilidad
Jefe de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.
Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

Tabla 244. Roles y Responsabilidades

2.1.2.4 Gestión del Proceso

2.1.2.4.1 Estimaciones del Proyecto

2.1.2.4.1.1 Introducción

El objetivo principal del plan de medidas es generar información que sea útil para gestionar el proyecto y tomar decisiones en base a datos objetivos.

2.1.2.4.1.2 Propósito

Se tomarán medidas sobre el proyecto de gestión “SIMA” Sistema Informático para Mejoramiento de Afiliación que se está desarrollando.

2.1.2.4.1.3 Alcance

Este plan se limita al proyecto de gestión “SIMA” Sistema Informático para Mejoramiento de Afiliación.

2.1.2.4.1.4 Visión General

Se van a identificar la información que se desea obtener y qué métricas miden esa información, describiendo con detalle algunas de ellas.

2.1.2.4.2 Objetivos y subobjetivos de la Gestión

Se deben identificar las necesidades de información para el proyecto de gestión “SIMA” Sistema Informático para Mejoramiento de Afiliación. Varias de estas medidas son necesarias para gestionar riesgos, otras para gestionar el proyecto midiendo tiempos y recursos, otras para comprobar los costes y otras para comprobar la calidad, por ejemplo del diseño realizado antes de comenzar la implementación. Puede haber más casos en los que se necesite hacer medidas, pero como ejemplo, en este documento sólo veremos algunas métricas representativas.

2.1.2.4.3 Métricas

Categoría de información	de	Concepto medible	Métrica
Tiempo y progreso		Tiempo y esfuerzo de desarrollo	COCOMO
Tamaño del producto		Tamaño físico	Número de líneas de código
		Tamaño Funcional	Puntos de función Número de casos de uso

Tabla 245. Métricas

2.1.2.4.4 Descripción de las Métricas

Nombre	COCOMO
Definición	Métrica de tiempo y esfuerzo de desarrollo
Objetivo	Ayudar a realizar la planificación temporal y de recursos, así como a realizar la gestión del proyecto.
Proceso de Análisis	Aplicar tablas y formulas del método COCOMO para obtener estimaciones de esfuerzo y tiempo.

Tabla 246. Cocomo

Nombre	Basado en Casos de Uso
Definición	Métrica de tamaño, Esfuerzo y Funcionalidad
Objetivo	Medir el tamaño de Producto y Funcional de la aplicación y Estimar Esfuerzo
Proceso de Análisis	Aplicar tablas con valores y hacer cálculos siguiendo el método de análisis de estimación basada en Casos de Uso.

Tabla 247. Basado en Casos de Uso

Nombre	Puntos de Función
Definición	Métrica de tamaño y complejidad
Objetivo	Medir el tamaño y complejidad de la aplicación

Proceso de Análisis	Aplicar tablas con valores y hacer cálculos siguiendo el método de análisis de puntos de función.
----------------------------	---

Tabla 248. Puntos de Función

2.1.2.4.5 Utilización de las Métricas

2.1.2.4.5.1 COCOMO

El tamaño del software varía de unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles de líneas (medio).

Se utilizan dos ecuaciones para determinar el esfuerzo de personal y el tiempo de desarrollo. El coste es:

$$K_m = 2.4 S_k^{1.05}$$

Dónde:

K_m se expresa en personas-mes.

S_k es el tamaño expresado en miles de líneas de código fuente.

El tiempo de desarrollo se da por:

$$t_d = 2.5 K_m^{0.38}$$

Donde K_m se obtiene de la ecuación anterior y t_d es el tiempo de desarrollo en meses.

El N° de Personas Medio se da por:

$$P_e = K_m / t_d$$

Donde K_m se obtiene de la ecuación anterior y t_d se obtiene de la ecuación anterior y P_e es el número de personas promedio.

$$K_m = (2.4)(15)^{1.05} = 41.22 \text{ personas-mes}$$

$$T_d = 2.5 (41.22)^{0.38} = 10.27 \text{ mes}$$

$$P_e = 41.22 / 10.27 = 4.01 \text{ personas}$$

Lo cual se estima que el proyecto se llevará a cabo en 10 meses con 4 personas, Por tanto, el proyecto se realizará en 250 días sin considerar los días domingos y feriados. En este proyecto sin embargo lo desarrolla sólo una persona que es el Jefe de Proyecto que tiene que realizar la función de las cuatro personas.

2.1.2.4.5.2 Basado en Casos de Uso

Se cuenta con los puntos de Casos de Uso ajustados y se calcula que para cada punto de Casos de Uso se requieren 2 horas-hombre. Por lo tanto el esfuerzo se obtiene aplicando la siguiente fórmula:

UCP = Punto de Caso de Uso

E = UCP * 2 = horas-hombre

E = 132 * 2 = 264 horas-hombre

Usar la contabilización de los factores ambientales para ajustar las horas-hombre que se requieren por punto de Caso de Uso, este valor de esfuerzo según la siguiente tabla representa a la actividad de programación, por tanto se debe realizar un nuevo cálculo agregando las actividades restantes relacionadas con el desarrollo del software.

Por lo tanto debería calcularse el esfuerzo de la siguiente manera:

Actividad	Porcentaje	Horas-hombre
Análisis	10%	10%*264=26.4
Diseño	20%	20%*264=52.8
Programación	40%	40%*264=105.6
Pruebas	15%	15%*264=39.6
Sobrecarga (otras actividades)	15%	15%*264=39.6
TOTAL de esfuerzo	100%	$\Sigma = 264$

Tabla 249. Calculo de esfuerzo

El tiempo es de 264 horas – hombre por cada Caso de Uso.

En la cual si realizamos el siguiente cálculo

$264 \text{ horas-hombre} / 24 \text{ horas} = 11 \text{ días}$ por cada Caso de Uso

Es una estimación que no tiene mucha relevancia por considerar sólo los requisitos del cliente capturados en los casos de uso.

2.1.2.4.5.3 Puntos de Función

Los puntos de función que obtienen utilizando una función empírica basando en medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software.

Los puntos de función se calculan mediante la siguiente fórmula:

$$PF=CT *(0.65+0.1*\sum Fi)$$

Dónde:

PF= Punto de función

CT= Cuenta Total

Fi= Cuenta de los valores de ajuste de complejidad

Los Puntos de Función se derivan de medidas directas del dominio de la información

Calculo de la cuenta total:

Parámetro	Cuenta	Factor de Ponderación			Subtotal
		Simple	Medio	Complejo	
Número de entradas de usuario	15	3	4	6	60
Número de salidas de usuario	20	4	5	7	140
Número de peticiones de usuario	15	3	4	6	90
Numero de archivos	3	7	10	15	2
Numero de interfaces externas	2	5	7	10	10
Cuenta Total					351

Tabla 250. Factor de Ponderación

Valores de Ajuste de Complejidad.

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

Tabla 251. Valores de Ajuste de Complejidad

Nº	Parámetros a Evaluar	Valor
1	Comunicación de datos	5
2	Funciones de procesamiento distribuidos	2
3	Objetivos de Performance	4
4	Ejecución del sistema en un entorno operativo utilitario	4
5	Transacciones de datos sobre múltiples entradas	1
6	Entrada de datos	5
7	Copia de seguridad y recuperación de datos fiable	5
8	Actualización de archivos en forma interactiva	5
9	Complejidad de procesamiento interno	3
10	Reusabilidad de código	4
11	Facilidad de instalación	2
12	Facilidad operacional	5
13	Soporte de múltiples instalaciones	4
14	Facilidad de cambio y manejo	5
	S-Total	54

Tabla 252. Parámetros a Evaluar

$$PF = \text{Cuenta-Total} * [0.65 + (0.01 * \text{S-Total})]$$

$$PF = 351 * [0.65 + (0.01 * 54)]$$

$$PF = 417.69$$

$$PF = 418$$

La siguiente tabla proporciona estimaciones del número de líneas de código que se necesitan para construir un punto de función en varios lenguajes de programación:

Lenguaje	LDC/PF
Ensamblador	320
C	128
Java	120
Fortran	105
Ada	70
4GL	20
Lenguajes de Iconos	6

Tabla 253. Estimación de Cantidad Líneas de Código en Función al Lenguaje de Programación
En base a esta tabla se pueden establecer las líneas de código (LDC) que sería la estimación para nuestro proyecto.

El lenguaje de programación que desarrollamos para nuestro proyecto es Java entonces tendríamos unas 120 LDC por PF:

$$LDC = 120 * 418$$

$$LDC = 50160$$

Para simplificar el proceso de estimación y utilizar una forma más común para su modelo de estimación, Putman y Myers sugieren un conjunto de ecuaciones obtenidas de la ecuación del software:

$$t = 8.14 (LDC / P)^{0.43} \quad (1)$$

$$E = 180 B t^3 \quad (2)$$

Donde

E = esfuerzo en personas – mes

t = duración del proyecto en meses a años.

B = factor especial de destrezas. Para programas mayores a 60 KLDC, B > 0.39.

P = parámetro de productividad. Para aplicaciones comerciales de sistemas, P > 28000

Aplicando las ecuaciones (1) y (2) a nuestro proyecto obtenemos:

Dónde: B = 1.03 y P = 28000 para nuestro caso.

$$t = 8.14 (50160 / 28000)^{0.43}$$

$$t = 10,46 \text{ meses.}$$

El tiempo estimado el proyecto es de 10 meses.

$$E = 180 * 1.03 * (0.82)^3$$

$$E = 102.22 \text{ personas – mes.}$$

Entonces 102.22 personas – mes. / 10 mes.= 10.22 personas.

El esfuerzo estimado el proyecto según el análisis de Punto de Función es de 10 personas por mes. Pero vale la pena mencionar que sólo una persona desarrolla el Sistema.

2.1.2.4.6 Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

2.1.2.4.6.1 Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar).

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	50
Fase de Elaboración	1	64
Fase de Construcción	1	61
Fase de Transición	1	53

Tabla 254. Plan de las Faces

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente /usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso

	<p>correspondientes a requisitos que serán implementados en la primera reléase de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana.</p>
Fase de Construcción	<p>Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una reléase a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la reléase 2.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.</p>
Fase de Transición	<p>En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.</p>

Tabla 255. Hitos

2.1.2.4.6.2 Calendario del Proyecto

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.

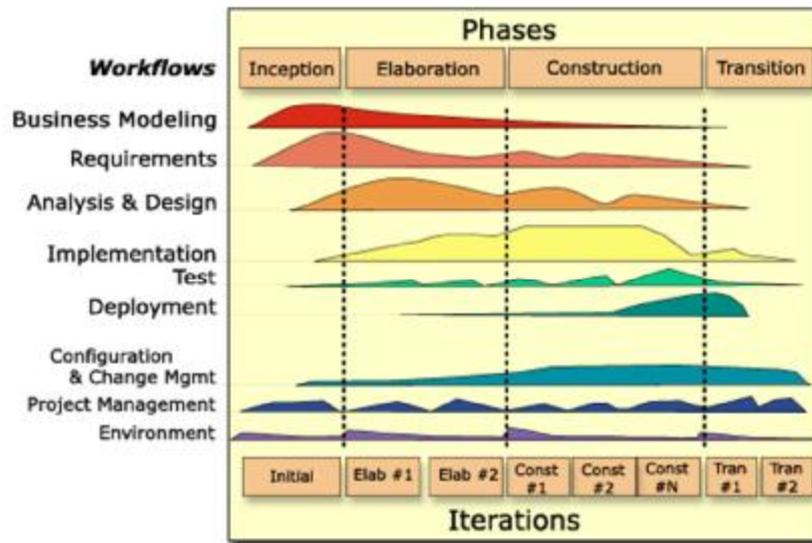


Figura 469. Calendario del Proyecto

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

Disciplinas / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		
Requisitos		
Glosario		
Visión		
Modelo de Casos de Uso		siguiente fase
Especificación de Casos de Uso		siguiente fase
Especificaciones Adicionales		siguiente fase
Análisis/Diseño		
Modelo de Análisis/Diseño		siguiente fase
Modelo de Datos		siguiente fase
Implementación		
Prototipos de Interfaces de Usuario		siguiente fase

Modelo de Implementación		siguiente fase
Pruebas		
Casos de Pruebas Funcionales		siguiente fase
Despliegue		
Modelo de Despliegue		siguiente fase
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones		
Ambiente	Durante todo el proyecto	

Tabla 256. Disciplina / Artefactos Fase de Inicio

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		aprobado
Requisitos		
Glosario		aprobado
Visión		aprobado
Modelo de Casos de Uso		
Especificación de Casos de Uso		
Especificaciones Adicionales		
Análisis / Diseño		
Modelo de Análisis / Diseño		Revisar en cada iteración
Modelo de Datos		Revisar en cada iteración
Implementación		
Prototipos de Interfaces de Usuario		Revisar en cada iteración
Modelo de Implementación		Revisar en cada iteración

Pruebas		
Casos de Pruebas Funcionales		Revisar en cada iteración
Despliegue		
Modelo de Despliegue		Revisar en cada iteración
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones		Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Tabla 257. Disciplina / Artefactos Fase de Elaboración

2.1.2.4.7 Seguimiento y Control del Proyecto

2.1.2.4.7.1 Gestión de Requisitos

Ver Anexo: Especificación de Requisitos de Software Norma IEEE835

2.1.2.4.7.2 Control de Plazos

Fase	Nº de Iteración	Inicio	Fin
Inicio	Primera	14-04-2010	31-05-2010
Elaboración	Primera	01-06-2010	03-08-2010
Construcción	Primera	04-08-2010	04-10-2010
Transición	Primera	05-10-2010	29-11-2010

Tabla 258. Control de Plazos

2.1.2.4.7.3 Control de Calidad

2.1.2.4.7.3.1 Introducción

El control de calidad es un modelo planeado y sistemático de todas las acciones necesarias para proporcionar la confianza de que el artículo o producto se ajuste a los requisitos técnicos establecidos (IEE83).

La preparación de un plan de control de calidad del software para cada proyecto de software es una de las principales responsabilidades del grupo de control de calidad del software.

El control de calidad realizará las siguientes funciones:

Durante el análisis y diseño, se presentaran un plan de verificación del software y un plan de prueba de aceptación. El plan de verificación describe los métodos que se ocuparán para revisar que los

documentos de diseño satisfagan los requisitos, y que el código fuente sea consistente con las especificaciones de requisitos y con la documentación del diseño.

El plan de prueba del código fuente es un componente importante del plan de verificación del software.

El plan de prueba de aceptación incluye casos de prueba, resultados esperados y capacidades demostradas por cada caso de prueba. A menudo, el personal de control de calidad trabajará con el cliente para desarrollar un solo plan de prueba de aceptación. En otros casos el cliente desarrollará un plan de prueba de aceptación independiente del plan de control de calidad. De cualquier forma, el personal de control de calidad debe desarrollar un plan de prueba de aceptación doméstico.

Al terminar los planes de verificación y de aceptación se realizará una revisión de verificación del software para evaluar cuan adecuados son los planes.

Durante la evolución del producto, se realizarán auditorías en el proceso para verificar que los productos de trabajo sean consistentes y estén completos.

Los elementos que sufrirán auditoría por consistencia incluyen especificaciones de interfaces para hardware, software y personas; diseño interno contra especificaciones funcionales; código fuente contra documentación. En la práctica, sólo ciertas porciones críticas del Sistema pueden someterse a auditorías intensivas.

Antes de la entrega del Sistema, se realiza una auditoría funcional y una auditoría física.

La primera reconfirma el cumplimiento de todos los requisitos. La auditoría física verifica que el código fuente y todos los documentos asociados estén completos, sean consistentes tanto internamente, como uno con otro, y que estén listos para enviarse. El resumen de verificación del software se repara para describir los resultados de todas las revisiones, auditorías y pruebas efectuadas por el personal de control de calidad, a través del ciclo de desarrollo.

Dicho personal, a veces se encarga de los acuerdos para los recorridos, inspecciones, y revisiones de logros principales. Además, el personal de control de calidad conduce el proyecto póstumo, escribe es documento del legado del proyecto, y proporciona una custodia a largo plazo de los registros del proyecto.

El grupo de control de calidad trabajará con el grupo de desarrollo para obtener el plan de pruebas y el código fuente, que especifica los objetivos de las pruebas, los criterios para la terminación de las pruebas, el plan de integración del Sistema, los métodos que se usaran en módulos particulares, además, entradas de prueba particulares y resultados esperados.

El plan de pruebas de código fuente tiene cuatro tipos de pruebas: pruebas de función, de desempeño, de tensión, y estructuradas. Las dos primeras se basan en las especificaciones de requisitos y se

diseñaron para demostrar que el sistema satisface sus requisitos los que a su vez se deben redactar en términos cuantificables y que se puedan probar.

Los casos de prueba funcional especifican condiciones operativas comunes valores de entradas comunes y resultados esperados comunes, también prueban el comportamiento dentro, sobre, y más allá de las fronteras funcionales.

Las pruebas de desempeño se proyectan para verificar el tiempo de respuesta, rendimiento, la utilización de memorias primarias y secundarias y las tasas de tráfico en los canales de datos y los enlaces de comunicación.

Las pruebas de tensión se diseñan para sobrecargar un sistema de varias maneras.

Las pruebas de estructura se relacionan con el examen de la lógica interna de procesamiento de un sistema de software. Las rutinas particulares llamadas y las rutas lógicas recorridas a lo largo de las rutinas son los objetos importantes.

2.1.2.4.7.3.2 Propósito

- Detectar problemas.
- Delimitar el área problemática.
- Estimar factores que probablemente provoquen el problema.
- Determinar si el efecto tomado como problema es verdadero o no.
- Prevenir errores debido a omisión, rapidez o descuido.
- Confirmar los efectos de mejora.
- Detectar desfases.
- Realizar pruebas en cada versión.

2.1.2.4.7.3.3 Objetivos

- Aumentar la satisfacción del cliente.
- Equilibrar el esfuerzo en múltiples demandas.
- Obtener el mejor producto.
- Disfrutar de una ventaja competitiva.
- Disponer de métricas objetivas de valoración.
- Ahorrar tiempo y dinero.

2.1.2.4.7.3.4 Resumen de las Actividades de Garantía de Calidad

Listado de las actividades de garantía de la calidad que se llevarán a cabo durante el proyecto.

2.1.2.4.7.3.4.1 Seguimiento de la Administración del SQA

ID	Propósito	Sub Actividad	Detalle / Indicación
1	Criterios de Entrada	- Ninguna.	Plan de Desarrollo y proceso de Software y Personal.
2	Revisión	- Examinar estructura gerencial de la organización. - Identificar tareas de cada integrante de la gerencia. - Definir responsabilidades a cada integrante de la gerencia	Verificar consistencia de la estructura organizacional con las responsabilidades asignadas en Plan de desarrollo de Software.
3	Criterios de Salida	- Estructura de la administración revisada.	Estructura organizacional de la gerencia óptima para el proyecto.

Tabla 259. Seguimiento de la Administración del SQA

2.1.2.4.7.3.4.2 Seguimiento de la Documentación

ID	Propósito	Sub Actividad	Detalle / Indicación
1	Criterios de Entrada	- Ninguna.	Plan de Desarrollo de Software.
2	Revisión	- Revisión y análisis del plan de documentación. - Buscar discrepancias. - Discutir discrepancias con el gerente de proyecto.	Reportar discrepancias según documento presentado por el docente y estándares.
3	Criterios de Salida	- Documentación revisada.	Documentos de acuerdo a Estándar, y sin discrepancias.

Tabla 260. Seguimiento de la Documentación

2.1.2.4.7.3.4.3 Seguimiento de la Adherencia a los Estándares

ID	Propósito	Sub Actividad	Detalle / Indicación
1	Criterios de Entrada	- Ninguna.	Documentos, plan de desarrollo de Software.
2	Documentación	- Monitorear adherencias de los documentos a los estándares.	Chequear documento desarrollado basado en la metodología RUP.
3	Diseño	- Monitorear adherencias del diseño a los estándares.	Chequear documento UML del sistema. Revisar de acuerdo a Patrones de
4	Codificación	- Monitorear adherencias de la codificación a los estándares.	Revisar de acuerdo a Patrones de Diseño MVC.
5	Métricas	- Revisar la métrica definida.	Revisar de acuerdo al estándar Puntos de Función y otros.

6	Criterios de Salida	- Proceso de Documentación revisado. - Proceso de Diseño revisado. - Métricas definidas revisadas.	Discrepancias reportadas y solucionadas. Documentos de acuerdo a estándares.
---	---------------------	--	--

Tabla 261. Seguimiento de la Adherencia a los Estándares

2.1.2.4.7.4 Gestión de Riesgos

2.1.2.4.7.4.1 Relacionados con el Tamaño del Producto.

- Es posible no poder llegar a terminar todos los componentes del proyecto debido a que estos son muchos o grandes.

2.1.2.4.7.4.2 Relacionados con el Impacto en la Organización.

- No entregar el proyecto en el tiempo estimado.
- El posible no uso del software debido a los requerimientos de equipos.
- Poco uso del software.
- Gran cantidad de documentación a entregar al cliente.
- Los límites legales y gubernamentales en cuanto al desarrollo y funciones del proyecto.
- Posibles errores en el producto y costos asociados.

2.1.2.4.7.4.3 Relacionados con el Tipo de Cliente.

- Si se tiene la disponibilidad de tiempo para la especificación formal de requerimientos.
- Si están dispuestos a participar de las pruebas o revisiones.
- Si se relacionará de forma ágil con el grupo de desarrollo.

2.1.2.4.7.4.4 Relacionados con la Definición del Proceso de Producción.

- El software a ser utilizado en el control del desarrollo del proyecto.
- Las herramientas a ser utilizadas en el análisis y diseño.

2.1.2.4.7.4.5 Relacionados con el Entorno de Desarrollo.

- Si hay herramientas de gestión de proyectos.
- Hay herramientas de prueba apropiadas.
- Generadores de código para la aplicación.

2.1.2.4.7.4.6 Relacionados con la Tecnología.

- Es una nueva tecnología.
- El hardware con el que debe interactuar es nuevo o cumple las expectativas.
- La base de datos a ser utilizada ha sido probada y tiene la funcionalidad y rendimiento.
- Las interfaces son especializadas.
- La necesidad de nuevos componentes.

2.1.2.4.7.4.7 Relacionados con la Experiencia y Tamaño del Equipo.

- Es el mejor personal disponible.
- Los miembros tienen las técnicas apropiadas.
- Hay suficiente gente disponible.
- El personal está comprometido a lo largo de desarrollo del proyecto.
- Tiene el personal las expectativas correctas del trabajo.

2.1.2.4.7.5 Gestión de Configuración

Se llevará a cabo una gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirían la gestión de las solicitudes de cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios a todos los participantes del proyecto.

2.1.3 Medios de Verificación del Componente Sistema Informático para el Mejoramiento de Afiliación

Ver Anexo I

2.2 Componente 2: Capacitación del Personal Caja Nacional Salud - Regional Tarija, en el Área de las TIC's y el Sistema Informático Implementado.

2.2.1 Introducción

De acuerdo al modo de vida actual del mundo, el término Capacitación y Sistemas de Información están modificando la forma de trabajo de las empresas, los Sistemas de Información ayudan a acelerar los procesos, por tanto, las organizaciones que los implantan logran ventajas competitivas al adoptarlos en sus funciones. Pero si el personal no está debidamente capacitado la producción de las empresa será deficiente y en mucho casos incompleto.

La capacitación se refiere a las metodologías que se usan para proporcionar a las personas dentro de la empresa, las habilidades necesarias que necesitan para realizar su trabajo de una manera más eficiente, esto contempla desde pequeños cursos que le permitan al usuario entender el funcionamiento básico del sistema nuevo, hasta capacitación más profunda y avanzada a bases de prácticas y material didáctico como libros (Tutoriales) y otros.

La capacitación es un proceso que lleva a la mejora continua de la producción y con esto implantar nuevas formas de trabajo, como en este caso el manejo adecuado del Sistema Informático desarrollado y Conocimientos en el área de las TIC's.

2.2.2 Aspectos Pedagógicos de la Capacitación

2.2.2.1 Definición de Capacitación

La capacitación es un proceso educacional de carácter estratégico aplicado de manera organizada y sistémica, mediante el cual los colaboradores adquieren o desarrollan conocimientos y habilidades específicas relativas al trabajo, y modifica sus actitudes frente a los quehaceres de la organización, el puesto o el ambiente laboral.

2.2.2.2 Objetivos de la Capacitación

- Preparar a los colaboradores para la ejecución de las diversas tareas y responsabilidades de la organización.
- Proporcionar oportunidades para el continuo desarrollo personal, no sólo en sus cargos actuales sino también para otras funciones para las cuales el colaborador puede ser considerado.
- Cambiar la actitud de los colaboradores, con varias finalidades, entre las cuales están crear un clima más propicio y armoniosos entre los colaboradores, aumentar su motivación y hacerlos más receptivos a las técnicas de supervisión y gerencia.

2.2.2.3 Importancia de la Capacitación

Permite:

- Consolidación en la integración de los miembros de la organización.
- Mayor identificación con la cultura organizacional.
- Disposición desinteresada por el logro de la misión empresarial.
- Entrega total de esfuerzo por llegar a cumplir con las tareas y actividades.
- Mayor retorno de la inversión.
- Alta productividad.
- Promueve la creatividad, innovación y disposición para el trabajo.
- Mejora el desempeño de los colaboradores.
- Desarrollo de una mejor comunicación entre los miembros de una organización.
- Reducción de costos.
- Aumento de la armonía, el trabajo en equipo y por ende de la cooperación y coordinación.
- Obtener información de fuente confiable, como son los colaboradores.

2.2.2.4 Beneficios de la Capacitación

Como beneficia la capacitación a las organizaciones:

- Conduce a rentabilidad más alta y a actitudes más positivas.
- Mejora el conocimiento del puesto a todos los niveles.
- Eleva la moral de la fuerza de trabajo.
- Ayuda al personal a identificarse con los objetivos de la organización.
- Crea mejor imagen.
- Mejora la relación jefes-subordinados.
- Es un auxiliar para la comprensión y adopción de políticas.
- Se agiliza la toma de decisiones y la solución de problemas.
- Promueve al desarrollo con vistas a la promoción.
- Contribuye a la formación de líderes y dirigentes.
- Incrementa la productividad y calidad del trabajo.
- Ayuda a mantener bajos los costos.
- Elimina los costos de recurrir a consultores externos.

2.2.3 Metodología

La metodología utilizada para esta capacitación i/o taller se basa en el modelo de experiencia “Aprender Haciendo”, logrando de esta manera la motivación de ser constructores de su propio conocimiento. Involucra trabajos de completa actividad en talleres de construcción de contenidos, Laboratorios, Guías, donde se incentiva el trabajo creativo y práctico, experimentándose con procesos

o fenómenos a partir de ideas o propuestas teóricas previamente formuladas.

Es una metodología que busca desarrollar su sensibilidad frente a problemas reales, estudiar alternativas de solución y evaluar sus implicancias en conjunto con la utilización de tecnologías.

2.2.3.1 Estrategias Didácticas

2.2.3.1.1 Descubriendo la Tecnología

2.2.3.1.1.1 Tipo: Sesiones Individuales y Grupales.

2.2.3.1.1.1.1 Descripción:

Se desarrollan actividades guiadas donde los capacitadores hacen uso del material guía y van descubriendo por sí solos las potencialidades de los programas y herramientas y que luego ellos mismos documentan haciendo énfasis en cuáles de las experiencias descubiertas son de principal apoyo para su práctica pedagógica.

2.2.3.2 Aspectos Prácticos

2.2.3.2.1 Actividad Teórico- Práctica:

Propicia la modalidad del curso teórico con una actividad de la práctica en relación a la temática de estudio. Lo teórico y lo práctico se dan simultáneamente en forma conjunta e interrelacionada y es dirigida directamente por el capacitador participante.

2.2.3.2.2 Entrenamiento Virtual:

Formación en la que los elementos teóricos y lo prácticos no se da simultáneamente sino separadamente y en entornos virtuales*. Para este sistema se utiliza el seguimiento de Tutores Facilitadores para cada área temática.

2.2.3.2.3 Talleres de Formación Continua:

Inclusión de la formación virtual* constante comprometiendo a los capacitadores a la continuidad de su capacitación a través de talleres y cursos de especialización.

** La formación virtual será desarrollada a través del sistema informático desarrollado o terminado, con el fin de que puedan conocer los sistemas informáticos o programas que utilizarán a futuro.*

2.2.3.3 Mecanismos de Trabajo Durante la Capacitación:

2.2.3.3.1 Fase Inicial:

Análisis de la Realidad Social de la Región: donde los Facilitadores intercambian diferentes criterios con los docentes logrando tener una visión clara de las necesidades pedagógicas de los participantes enfocando la capacitación a la Presentación de Temas Individuales.

Diagnóstico de Iniciación: donde se recolecta información mediante evaluaciones de conocimientos que nos permitirá medir el nivel que poseen los participantes en la utilización de las aplicaciones.

Motivación y Sensibilidad: donde los Facilitadores realizan dinámicas para Motivar y sensibilizar sobre la importancia de desarrollar capacidades en el manejo de las TIC's, y de esta manera los capacitadores puedan aplicar la herramienta y aportar en su conjunto al desarrollo de su comunidad.

2.2.3.3.2 Fase de Capacitación:

Investigación y documentación: se promueve en todas las estrategias didácticas aplicadas a la investigación continua y descubrimiento de la tecnología reconociendo su valor e identificando su potencial para ser utilizados en su preparación de material, todo esto desarrollado por los mismos capacitadores.

Defensa de Trabajos: donde los capacitadores trabajan en grupos o en forma individual presentando sus productos y siendo ellos mismos quienes se evalúan y corrigen para el enriquecimiento en conjunto el conocimiento y experiencia.

Talleres y Ponencias de Proyectos: donde los capacitadores presentan trabajos y exponen proyectos que van surgiendo a lo largo de la capacitación y los cuales son motivo de medición de conocimiento y desarrollo de destreza.

2.2.3.3.3 Fase de Finalización:

Medición de aprovechamiento: donde los Facilitadores realizan las evaluaciones de trabajos en forma individualizada como a su vez de término de grupo capacitado.

2.2.3.4 Modalidades Prácticas de Capacitación

El Plan de Capacitación podrá usar otras modalidades que se incorporen según las necesidades:

2.2.3.4.1 Inducción

Su objeto es la ambientación inicial al medio social y físico donde trabaja y se programa para todo colaborador nuevo.

2.2.3.4.2 Capacitación en el Puesto de Trabajo

Se desarrollara en el propio puesto de trabajo y mientras el interesado ejecuta sus tareas.

2.2.3.4.3 Cursos Internos

Consistirían en eventos de capacitación sobre técnicas y/o temas académicos, científicos, tecnología, u otro tema de interés empresarial, los mismos que se organizarán en la Sede Central de la empresa, con la participación en algunos casos de las sucursales. Entiéndase que para denominarlos como tal debe tener como mínimo 40 horas de duración.

2.2.3.4.4 Seminarios / Talleres

Son eventos de corta duración, alrededor de 14 horas en tres fechas y sobre temas puntuales que sirvan para reforzar o difundir aspectos técnicos o administrativos, en otros eventos generalmente concurrirán funcionarios de la sede central.

2.2.3.4.5 Cursos de Actualización

Los cursos de actualización generalmente se programan o se realizan en universidades, escuelas especializadas, institutos y otras instituciones comprometidas con la actualización, y desarrollo permanente de conocimientos de los colaboradores, estos eventos más recomendados son los seminarios y talleres en los que se dictan técnicas nuevas para personal de nivel jerárquico de la empresa.

2.2.3.5 Medios de Capacitación

Dentro de los más principales tenemos:

2.2.3.5.1 Conferencia

Permite llegar a una gran cantidad de personas y transmitir un amplio contenido de información o enseñanza. Se puede emplear como explicación preliminar antes de demostraciones prácticas. Por ejemplo, es útil al impartir las medidas de seguridad, organización de planta, etc.

2.2.3.5.2 Manuales de Capacitación

Manuales de capacitación u otros impresos, diagramas que permiten la exposición repetida, es útil aplicación de secuencias largas o procedimientos complicados que no pueden retenerse en una sola presentación. Pueden combinarse con conferencias y prácticas de tareas reales.

2.2.3.5.3 Videos

Puede sustituir a las conferencias o demostraciones formales, permite la máxima utilización de instructores más capaces. Los cortes, empalmes o en la cámara lenta son útiles para incidir en demostraciones de realidad. Ayudan a la comprensión de ideas abstractas y en la modificación de actitudes.

La grabación y proyección en videos de los colaboradores sujetos a capacitación, son un medio muy eficaz, sobre todo cuando se trata de mejorar la calidad del servicio.

2.2.3.5.4 Simuladores

Dan al aprendiz la posibilidad de participación y práctica repetida mediante la adquisición de habilidades necesarias en el trabajo real, se usan también sustitutos del equipo real. Pueden aislar y combinar las diferentes partes críticas o peligrosas del trabajo.

2.2.3.5.5 Realización Efectiva del Trabajo

El nuevo colaborador aprende mientras trabaja, bajo la guía de un instructor, es útil en la transmisión de habilidades, de experiencia ensayo y error. Su limitación es que no siempre el buen colaborador es buen instructor. Puede durar pocos días o meses.

En general, el período de aprendizaje brinda preparación para una gran variedad de especialidades, cubriendo múltiples actividades.

2.2.3.5.6 Discusión de Grupos e Interacción Social

Comprende el desarrollo de habilidades interpersonales requeridas por tareas ejecutivas y de supervisión como vías de solución de problemas mediante grupos de discusión, dirección de debates y contratos con personas para el manejo directivo de problemas reales de supervisión. Se usa mayormente en formación de ejecutivos.

2.2.3.5.7 Entrevistas para la Solución de Problemas

Se orienta básicamente al asesoramiento de colaboradores. Los supervisores encargados de capacitación mantienen periódicamente estas estrategias para mejorar la eficiencia en el trabajo de cada individuo. Se usa generalmente en la capacitación de directivos.

2.2.3.5.8 Técnicas Grupales

Consiste en ejercicios vivenciales, dinámicas grupales como los juegos de roles, psicodramas, Phillips 66, lluvias de ideas, y otros que pueden ser valiosos elementos para llevar a cabo la capacitación de acuerdo a los objetivos planteados.

2.2.4 Determinación de la Estructura y la Metodología del Componente Capacitación

Para determinar la estructura y la metodología de la capacitación, existe una cantidad de cuestiones interrelacionadas que requiere solución como pueden ser:

- Cuál será la mejor estructura para el programa de capacitación y cuándo se deberá instrumentar. (Momento Adecuado para Impartir la Capacitación);
- Qué se necesitará para llevar a cabo una presentación exitosa. (Materiales de Referencia con Propósitos de Capacitación);
- Qué temas deberán cubrir las sesiones de capacitación y cómo deberán organizarse (Estilo para Impartir la Capacitación);
- Qué instalaciones y recursos adicionales se necesitarán. (Entorno para la Capacitación);
- Cómo se medirán los resultados de la capacitación. (Evaluación del Conocimiento).

2.2.5 Ingeniería de Requerimientos

2.2.5.1 Propuesta de Solución para el Curso de Capacitación.

2.2.5.1.1 Introducción

La efectividad de la capacitación puede ser considerablemente influenciada por el estilo de impartición y los métodos de presentación empleados para este fin.

El entorno físico en el cual se llevará a cabo la capacitación tiene una importante repercusión en la eficacia de la misma. Es por lo mismo que se llevará a cabo en las instalaciones de la Caja Nacional de Salud ya que el grado de respuesta a la capacitación puede disminuirse significativamente si las necesidades básicas no se han organizado satisfactoriamente.

Las personas que se motivan y desean ser mejores a través del proceso de capacitación requieren de una herramienta que los apoye a lograr su nueva actitud en la organización con procedimientos rápidos y fáciles.

Con lo expuesto anteriormente se realizarán dos sesiones de capacitación,

Primer sesión, el 26 de noviembre, será la capacitación en el uso de las TIC's, en la que asistirán el personal administrativo y médico de la Caja Nacional de Salud - Regional Tarija.

Segunda sesión, el 06 de diciembre al 10 de diciembre, será la capacitación en el manejo del Sistema Informático desarrollado denominado "SIMA" Sistema Informático para el Mejoramiento de Afiliación, en la que asistirán el personal de la Sección de Afiliación y el Encargado de Sistemas.

2.2.5.1.2 Objetivos de la Capacitación

En cuanto al objetivo, éste apunta a: Capacitar al Personal de la Sección de Afiliación, Encargado de Sistemas en el Manejo Correcto del Sistema Web para el Mejoramiento en los Procesos de Registro en la Sección de Afiliación de la Caja Nacional de Salud – Regional Tarija

2.2.5.1.3 Medios A utilizar

2.2.5.1.3.1 Aspectos Técnicos

- 1 Data Displaye.
- 1 Computador con el Sistema Desarrollado.
- Diapositivas de exposición hechas en PowerPoint.
- Contenido en Documento Impreso.

2.2.5.1.3.2 Aspectos Logísticos

- Refrigerios

2.2.5.1.4 Estructura del Curso de Capacitación

2.2.5.1.4.1 Capacitación en Uso de las TIC's

Fecha Realizar	Módulos de Aprendizaje del Curso	Horas C/Módulo	Nº de Sesiones	Dirigido A
26/11/2010	Módulo 1. La Internet	25 min.	1	Personal de la Caja Nacional de Salud Regional Tarija.
	Módulo 2. Sistemas de Información	25 min.	1	
	Módulo 3. Introducción a Sistemas Web	10 min.	1	
	Totales	1 Hora, 00 min	3 Sesiones	

Tabla 262. Capacitación en Uso de las TIC's

2.2.5.1.4.2 Capacitación en el uso del Sistema Informático Desarrollado

Fecha Realizar	Módulos de Aprendizaje del Curso	Horas C/Módulo	Nº de Sesiones	Dirigido A
06/12/2010	Módulo 4. Presentación y Manejo del Sistema Informático SIMA.	45 min.	1	Personal de la Caja Nacional de Salud Regional Tarija y en Especial al Personal de la Sección de Afiliación.
07/12/2010	Capacitación en el Manejo del Sistema Informático SIMA	45 min.	1	Encargado de Sistemas y Personal de la Sección de Afiliación.
08/11/210	Capacitación en el Manejo del Sistema Informático SIMA	45 min.	1	Encargado de Sistemas y Personal de la Sección de Afiliación.
09/11/210	Capacitación en el Manejo del Sistema Informático SIMA	45 min.	1	Encargado de Sistemas y Personal de la Sección de Afiliación.
10/11/210	Capacitación en el Manejo del Sistema Informático SIMA	45 min.	1	Encargado de Sistemas y Personal de la Sección de Afiliación.
	Totales	3 Hora, 45 min	5 Sesiones	

Tabla 263. Capacitación en el uso del Sistema Informático Desarrollado
2.2.5.1.5 El Contenido de la Capacitación se Realizó de Acuerdo al Siguiete

Cronograma:

Fecha	Hora	Actividad	Expositor
26 de Noviembre del 2010	18:15 – 18:40	Módulo 1: La Internet.	Omar Elías Yana Cerezo
26 de Noviembre del 2010	18:45 – 19:10	Módulo 2: Sistemas de Información.	Omar Elías Yana Cerezo
26 de Noviembre del 2010	19:20 – 19:30	Módulo 3: Introducción a Sistemas Web.	Omar Elías Yana Cerezo
06 de Diciembre del 2010	18:25 – 19:10	Módulo 4: Presentación y Manejo del Sistema Informático SIMA.	Omar Elías Yana Cerezo

Tabla 264. Cronograma de Contenido de Capacitación.

2.2.5.1.6 Contenido del Curso de Capacitación

Nº	ACTIVIDAD	RESPONSABLE	HORA INICIO	HORA FIN	DURACIÓN	MEDIOS A UTILIZAR
1	Presentación y Bienvenida al Curso de Capacitación.	Encargado de Sistemas de la C.N.S.	06:00 pm	06:10 pm	10 min.	
2	Explicación de la Importancia de la Capacitación.	Jefe del Proyecto	06:10 pm	06:15 pm	5 min.	Diapositivas Data Display Computador
3	La Internet	Jefe del Proyecto	06:15 pm	06:40 pm	25 min.	Diapositivas Data Display Computador
4	Intervalo de Descanso	Jefe del Proyecto	06:40 pm	06:45 pm	5 min.	
5	Sistemas de Información.	Jefe del Proyecto	06:45 pm	07:10 pm	25 min.	Diapositivas Data Display Computador
6	Refrigerio	Jefe del Proyecto y Colaboradores	07:10 pm	07:20 pm	10 min.	
7	Introducción a Sistemas Web.	Jefe del Proyecto	07:20 pm	07:30 pm	10 min.	Diapositivas Data Display Computador
8	Consultas	Encargado de Sistema, Personal Médico y Personal de la Sección de	07:30 pm	07:45 pm	15 min.	Diapositivas Data Display Computador

		Afiliación.				
9	Entrega del Formulario de Evaluación de la Capacitación	Jefe del Proyecto y Colaboradores	07:45 pm	07:50 pm	5 min.	Formulario de Evaluación de Capacitación
10	Llenado de los Formularios de Evaluación	Encargado de Sistema, Personal Médico y Personal de la Sección de Afiliación.	07:50 pm	08:05 pm	15 min.	Formulario de Evaluación de Capacitación
11	Recogida de los Formularios de Evaluación	Jefe del Proyecto y Colaboradores	08:05 pm	08:10 pm	5 min.	Formulario de Evaluación de Capacitación
12	Clausura del Curso de Capacitación	Encargado de Sistemas de la C.N.S. y Jefe del Proyecto.	08:10 pm	08:05 pm	5 min.	
Nº	ACTIVIDAD	RESPONSABLE	HORA INICIO	HORA FIN	DURACIÓN	MEDIOS A UTILIZAR
1	Presentación y Bienvenida al Curso de Capacitación.	Encargado de Sistemas de la C.N.S.	06:00 pm	06:10 pm	10 min.	
2	Explicación de la Importancia del Sistema Desarrollado	Jefe del Proyecto	06:10 pm	06:25 pm	15 min.	Diapositivas Data Display Computador Sistema Desarrollado
3	Presentación del Sistema Informático para Mejoramiento de Afiliación y Explicación del Módulo al Personal Correspondiente.	Jefe del Proyecto	06:25 pm	07:10 pm	45 min.	Diapositivas Data Display Computador Sistema Desarrollado
6	Refrigerio	Jefe del Proyecto y Colaboradores	07:10 pm	07:20 pm	10 min.	
8	Consultas	Encargado de Sistema, Personal Médico y Personal de la Sección de Afiliación.	07:20 pm	07:35 pm	15 min.	Diapositivas Data Display Computador Sistema Desarrollado

12	Clausura del Curso de Capacitación	Encargado de Sistemas de la C.N.S. y Jefe del Proyecto.	07:35 pm	07:45 pm	10 min.	
----	------------------------------------	---	----------	----------	---------	--

Tabla 265. Contenido del Curso de Capacitación

2.2.5.1.7 Medios de Verificación para el Componente Capacitación

Ver Anexo II

III. Capítulo 3: Conclusiones y Recomendaciones

3.1 Conclusiones

En base al trabajo realizado y a la experiencia que se fue recopilando a medida que se desarrolló este proyecto, permite llegar a las siguientes conclusiones:

- El número de organizaciones, instituciones que hacen uso de las TIC's (Tecnologías de Información y Comunicación) para la gestión y automatización de sus procesos administrativos, están en incremento.
- Se pudo constatar que la información es un factor necesario para el proceso de toma de decisiones en las organizaciones, instituciones y sobre todo en las empresas que buscan satisfacer las necesidades de servicios de calidad.
- En conclusión, habiendo desarrollado de manera adecuada las actividades previstas como ser : recolección de requerimientos, análisis y diseño, diseño de la base de datos, programación del sistema; realizado capacitaciones en el uso de las TIC's y manejo del sistema, en la cual nos han permitido producir los componentes; es decir el sistema mismo y la Capacitación del Personal de la C. N. S. – Regional Tarija, de acuerdo a todo lo planificado, habiendo reflejado con ello el propósito del proyecto que es Mejorar los “Procesos de Registro en la Sección de Afiliación de la C. N. S. – Regional Tarija” y con ello se ha contribuido significativamente en el fin del proyecto que es “Contribuir a mejorar la calidad de servicios a la población asegurada de la Caja Nacional de Salud – Regional Tarija.

3.2 Recomendaciones

- Se recomienda a la Institución, realizar las gestiones necesarias para el estudio y desarrollo de sistemas para las distintas secciones o departamentos de la C.N.S. – Regional Tarija, con lo cual se llegaría a tener información centralizada, mediante sistemas parcialmente integrados.
- Se recomienda implementar o instalar una Infraestructura Tecnológica referente a las TIC's, bajo el cumplimiento de estándares de calidad.
- Se recomienda la implementación de red de comunicación de fibra óptica, cableado de datos y un servidor propio.

- Se recomienda que el RR. HH. en el área informático debe cumplir con el Perfil adecuado en conocimiento de las nuevas metodologías de desarrollo y en el manejo de las últimas Tecnologías.