

I. Capítulo I

1.1 Presentación del Proyecto

1.1.1 Título del proyecto

“Mejoramiento del control de ventas y difusión de información de la empresa de comida rápida VidiVidi: “VidiVidi Virtual”.

1.1.2 Carrera / Unidad

Ingeniería Informática

1.1.3 Facultad

Ciencias y Tecnología

1.1.4 Duración del proyecto

10 meses

1.1.5 Área/línea de investigación priorizado

Tecnologías de la Información y Comunicación - Desarrollo de Sistemas Informáticos – Sistema Web

1.1.6 Responsable del proyecto

Carrera de Ingeniería Informática – Taller III

1.1.7 Entidad asociada (s)

Empresa de comida rápida VidiVidi

1.2 Personal vinculado al proyecto

1.2.1 Director del proyecto

Céspedes	Gira	Alex Daniel	5815471 Tja
Apellido Materno	Apellido Paterno	Nombre	CI
Estudiante Univ.	Ing. Informática	Ciencias y Tecnología	

Profesión	Carrera o Unidad	Facultad	
66 31238 Teléf. Oficina	72954119 Celular	alex_dcg@hotmail.es Correo electrónico	Firma

Tabla 1 Director del Proyecto

1.2.2 Participantes del equipo de trabajo

Categoría	Nombres y Apellidos	Carrera/Profesión	C.I.	Firma
Director	Alex Daniel Céspedes Gira	Estudiante en Ing. Informática	5815471 Tja	
Asesor	Lic. Efraín Torrejón	Lic. en Ingeniería Informática	1337531	

Tabla 2 Participantes del equipo de trabajo

1.2.3 Actividades previstas para los integrantes del equipo de investigación

Responsable	Actividades
Director	Como Jefe de Proyecto: - Planificar y controlar del cronograma del proyecto. - Asignar y gestionar recursos y prioridades a los distintos componentes y actividades del proyecto. - Coordinar las iteraciones entre el equipo de trabajo y los usuarios del proyecto.

	- Mantener al equipo del proyecto enfocado en los objetivos. - Establecer un conjunto de prácticas que aseguren la calidad e integridad del proyecto. - Supervisar el desarrollo del proyecto. Como Analista de Sistemas: - Capturar la especificación y validación de requisitos interactuando con los usuarios. - Elaborar el Análisis y Diseño del Sistema. Como Programador: • Realizar la Programación del Sistema. - Elaborar las pruebas funcionales del Sistema. Como Capacitador: - Formar al personal en el uso de las TIC para el manejo del producto final.
Asesor	- Asesoramiento en los aspectos tecnológicos para el desarrollo del Proyecto. - Asesoramiento en el uso de las Metodologías de desarrollo. - Evaluación del documento del proyecto.

Tabla 3 Actividades para los integrantes del equipo de investigación

1.2.3.1 Unidades de Gestión: Organigrama del equipo del proyecto

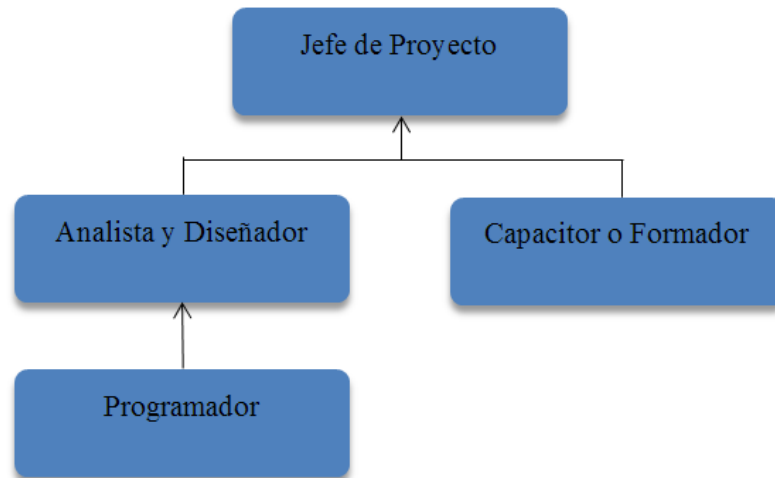


Figura 1 Organigrama del Equipo de Proyecto

1.3 Descripción del Proyecto

1.3.1 Resumen ejecutivo del proyecto

VidiVidi es una empresa ubicada en la ciudad de Tarija, que brinda el servicio de comida rápida. En la actualidad cuenta con tres puntos de atención hábiles, ubicados en los barrios: (Barrio Juan XXIII sobre la calle Font (sucursal 1), calle Ingavi entre Daniel Campos y Sucre (sucursal 2) y avenida Jaime Paz Zamora frente al B/ San Gerónimo (sucursal 3).

El servicio básicamente consiste en la dispensación de una serie de productos elaborados de comida rápida, donde se atiende a la clientela presente en cada local, siendo la excepción la Sucursal 2, que cuenta con la atención de pedidos telefónicos, este tipo de servicio continuará, por lo tanto el sistema deberá poder registrar los pedidos locales, telefónicos y vía Web.

El propósito de la administración es, por un lado, mejorar el manejo de información correspondiente a las ventas y pedidos de productos, salvando los problemas emergentes del crecimiento de la empresa y por otro, dada la rápida expansión de la

competencia de comida rápida en la ciudad de Tarija, se debe buscar nuevas estrategias para mejorar la difusión de los servicios y productos que brinda la empresa VidiVidi a la población.

El presente proyecto se enfoca al desarrollo de un sistema informático escritorio-Web que permitirá al personal de la empresa VidiVidi, mejorar el control de ventas y brindar información de los productos a los clientes con la posibilidad de realizar pedidos vía WEB.

1.3.2 Descripción, Fundamentación y Justificación del Proyecto

La empresa de venta de comida rápida VidiVidi nace en abril de 2008, en un ambiente ubicado en la zona central de la ciudad de Tarija con un concepto empresarial de comida al paso, para la alta afluencia peatonal de esa zona durante las horas de oficina.

VidiVidi entra en el mercado ofertando únicamente súper panchos (hotdogs), de 22 cm de largo, con más de 12 salsas y acompañamientos.

Tras una respuesta favorable de la demanda y en busca de una expansión y consolidación de marca, la empresa, en septiembre de 2008 abre una nueva sucursal ubicada en el barrio Juan XXIII de la ciudad de Tarija, con un concepto enfocado en las familias.

A seis meses de la inauguración de la sucursal en Juan XXIII, se percibe una respuesta muy favorable de la demanda específicamente en esa ubicación.

Dado que el ambiente en la zona central no contaba con el espacio para la incorporación de los nuevos productos ofrecidos en Juan XXIII, se desafía el traslado a un ambiente más amplio en la misma cuadra, que permitiría una mayor capacidad de producción y brindar ambientes más cómodos y agradables para el cliente.

En el último mes se consolidó la apertura de una nueva sucursal en la Av. Héroes de la Independencia, el local más grande de la cadena de VidiVidi, con una capacidad de 25 mesas y una zona exclusiva para los niños.

Después de 5 meses, debido a la baja concurrencia de clientes en la última sucursal instalada, trasladaron la misma a la Av. Las Américas frente al barrio San Gerónimo (Vidi Auto), incursionando un estacionamiento de automóviles exclusivo para realizar compras.

Los tres ambientes cuentan con la maquinaria, mueblería y personal adecuado para brindar un servicio rápido. Los ambientes se centran con alto énfasis en el diseño, la estética y la imagen empresarial.

Pronto alcanzaron un alto nivel de ventas y aunque el menú era limitado, el éxito era creciente. VidiVidi ofrece a sus clientes un menú estándar en todos sus locales. Los alimentos se preparan de acuerdo a las leyes locales de sanidad y control de higiene y calidad.

El personal de la empresa de Vidi Vidi registra los pedidos en plantillas de papel, con la existencia de algunos errores en desmedro de un óptimo funcionamiento y demorando en la entrega del producto final.

La realización de reportes implica la revisión de toda la documentación de ventas realizadas, mediante un conteo de los productos vendidos, ha provocado que se emitan reportes de baja fiabilidad, y se ha observado un alto índice de incoherencia, registros mal elaborados, ilegibles y repetidos.

Tomando en cuenta la creciente expansión del uso del Internet por la población en general, el presente proyecto de investigación pretende crear un portal Web con la información necesaria para difundir la información de la empresa y facilitar los pedidos de los clientes a través de una tienda virtual.

Con este enfoque, el presente proyecto de investigación pretende la creación de una aplicación informática que posibilite el registro de pedidos locales, telefónicos y Web; realice el registro de ventas y genere reportes diarios y por periodos de tiempo de acuerdo a las necesidades de los usuarios.

1.3.2.1 Análisis de causas de problemas

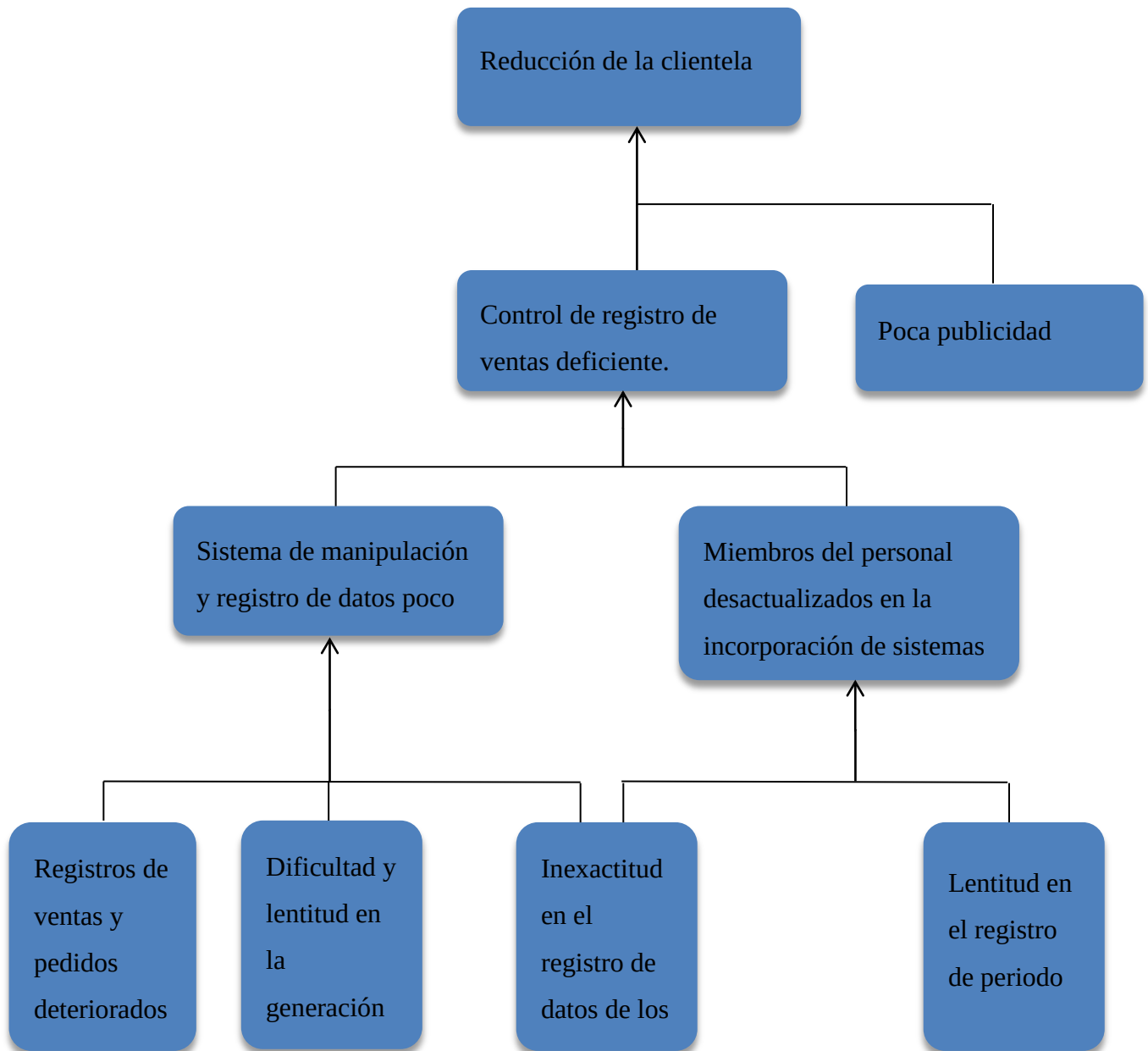


Figura 2 Árbol de Problemas

1.3.2.2 Análisis de Objetivos

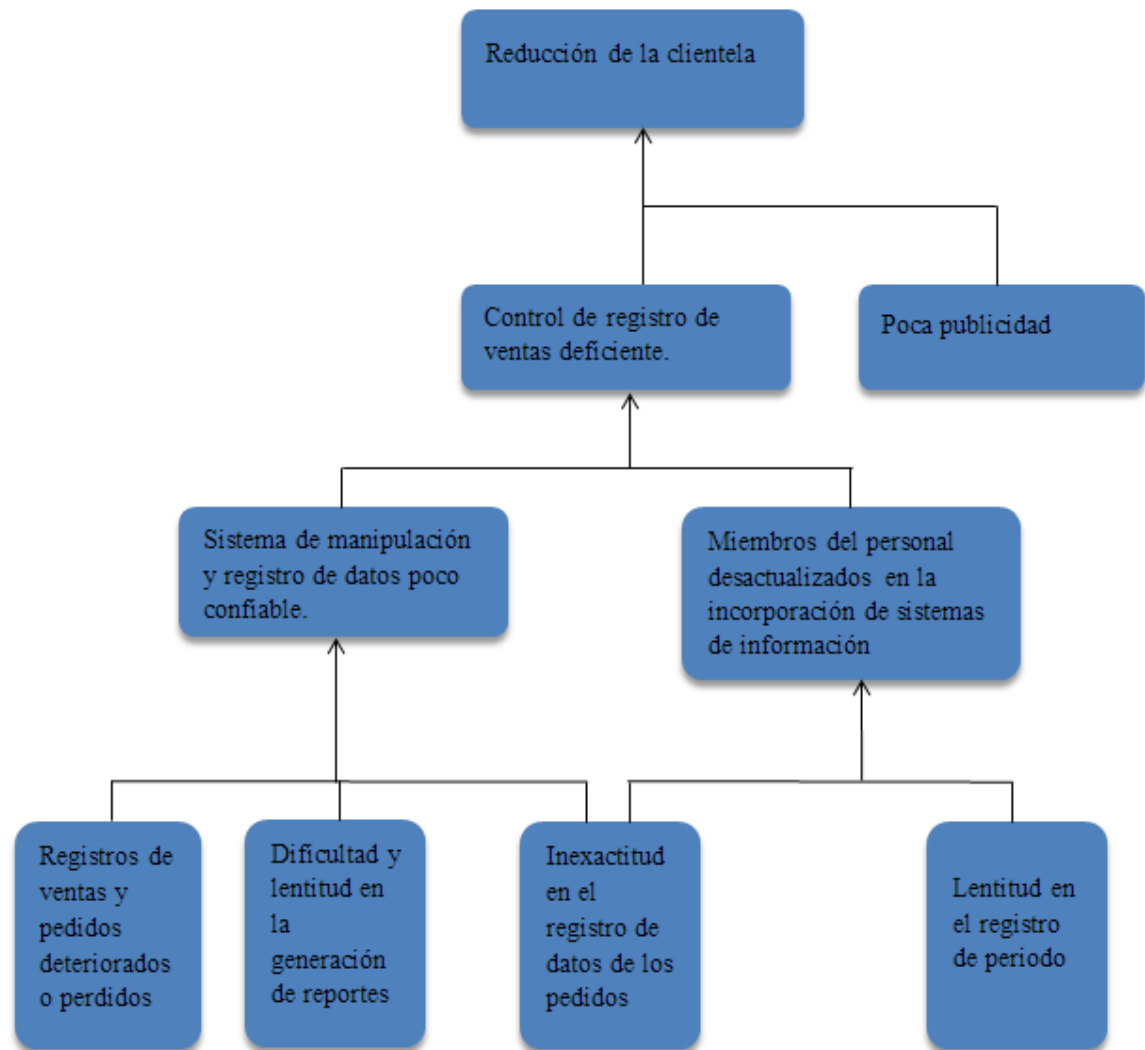


Figura 3 Análisis de Objetivos

1.3.3 Objetivos

1.3.3.1 Objetivo General

Mejorar el control de ventas y la difusión de información de la empresa de comida rápida VidiVidi.

1.3.3.2 Objetivo Específico

- Sistema Informático de gestión de ventas y pedidos vía Web, desarrollado.
- Socialización y Capacitación del personal para el uso del Sistema VidiVirtual.

1.3.4 Metodología

La metodología adapta para el desarrollo de este proyecto consiste de:

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:** Traslado de las necesidades del negocio a un sistema automatizado.
- **Análisis y Diseño:** Traslado de los requerimientos dentro de la arquitectura de software.
- **Programación e Implementación:** Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- **Pruebas:** Asegurándose que 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 probadas y en actual vigencia .

Pruebas y Validación: Antes de desarrollar las pruebas se procederá a la introducción de datos, luego 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.

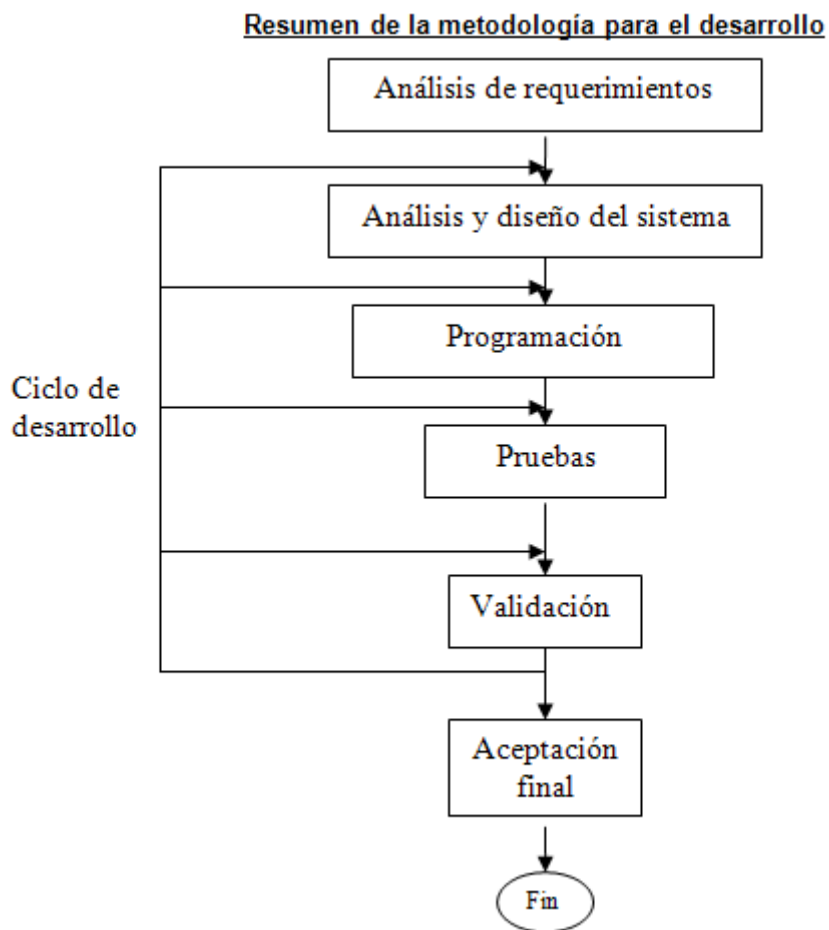


Figura 4 Resumen de la Metodología para el Desarrollo

1.3.4.2 Metodología para las Capacitaciones y/o Talleres

La metodología utilizada para esta capacitación y/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, 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 busca desarrollar la sensibilidad frente a situaciones, 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 el capacitador hace uso del material guía adecuado, fomentando que los participantes descubran por sí solos las funcionalidades y potencialidades del sistema.

Aspectos Prácticos

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

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.

1.3.5 Resultados esperados

Con el Sistema de Ventas de VidiVidi se espera poder realizar las transacciones de pedidos, ventas y reportes de una forma óptima, mejorando la precisión y velocidad de atención al cliente para satisfacción de sí mismos y de la gerencia.

La difusión de los servicios vía WEB y la posibilidad de realizar los pedidos por esta vía, busca incrementar la clientela a través de un portal atractivo, que refleje el concepto de calidad del servicio que VidiVidi ofrece.

1.3.6 Transferencia de resultados

1.3.6.1 Medios y estrategias para la transferencia de Resultados

Presentación final del sistema informático al directivo de la empresa VidiVidi.

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

Socialización del producto final con el personal involucrado.

Se realizará un convenio entre la Universidad Autónoma Juan Misael Saracho y la empresa VidiVidi con el fin de ceder los derechos de uso.

1.3.7 Grupo de beneficiarios

Directores

Podrá obtener reportes al día y en determinados periodos de tiempo; conocer el estado de operación de cada una de las sucursales.

Clientes

Serán atendidos de manera rápida y efectiva, disminuyendo aquellos errores provocados principalmente por los registros de pedidos incorrectos.

La población que goce de una conexión a la Internet, podrá acceder a la información acerca de VidiVidi (productos, costos, promociones, anuncios, direcciones de las sucursales) que podrá ser consultada desde cualquier lugar y por lo tanto también podrán hacer pedidos vía Web.

Empleados

Podrán registrar los pedidos de Clientes de manera intuitiva y sencilla, sin necesidad de realizar cálculos mentales y/o manuales evitando discrepancias con el cliente.

Podrán desarrollar reportes de gastos y ventas de manera automática evitando fallos de cálculos y pérdidas de tiempo.

1.3.8 Bibliografía consultada

[1] Empresa de Comida Rápida Vidi-Vidi

[2] http://html.rincondelvago.com/analisis-y-diseno-de-sistemas_1.html;

- [3] Grady Booch; James Rumbaugh; Ivar Jacobson; “El lenguaje unificado de modelado ”; Addison Wesley Iberoamericana año 2000; Paginas: 93 – 133; 203 – 210
- [4] <http://www.slideshare.net/ahaazd/mml-base-para-matriz-de-indicadores-para-resultados>
- [5] http://es.wikibooks.org/wiki/Inform%C3%A1tica_Educacional/Seguimiento_y_Control_del_Proyecto;
- [6] Addison Wesley Managing Iterative Software Development Projects Jun 2006 eBook
- [7] Ing. Córdoba Cepal: LA METODOLOGIA DEL MARCO LOGICO COMO BASE PARA CONSTRUIR LA MATRIZ DE INDICADORES. Diciembre 2008.
- [8] <http://www.slideshare.net/ahaazd/mml-base-para-matriz-de-indicadores-para-resultados>
- [9] Hugo Camacho; Luis Camaro; Rafael Cascante; Héctor Sainz: El Enfoque del Marco Lógico: 10 Casos Prácticos: Acciones de Desarrollo y Cooperación A.D.C. Marzo 2004.
- [10] <http://www.sc.ehu.es/jiwdocoj/mmis/cocomo.htm>;

1.3.9 Matriz de Marco Lógico

RESUMEN NARRATIVO	INDICADORES VERIFICABLES OBJETIVAMENTE	MEDIOS DE VERIFICACIÓN	SUPUESTOS
Fin Incrementar la clientela de la empresa de comida rápida “VidiVidi”.	Al haber finalizado e implementado el proyecto las ventas habrán mejorado en un 20% por lo tanto los ingresos se han mejorado en comparación con el año base 2011.	Reporte de registro de ventas realizado en el periodo indicado.	La empresa VidiVidi hace uso continuo del sistema VidiVirtual en cada sucursal. Se actualiza el portal WEB.

Propósito			
Control de ventas y difusión de información, mejorado.	Cumplido dos años de haber finalizado el proyecto, el número de errores en los procesos asociados a la venta (pedido, entrega y cobro) se redujo en un 80% tomando como base un informe de ventas del año 2011, antes de la puesta en marcha del proyecto.	Tabla comparativa del porcentaje de errores de la gestión 2011 y 2012. Certificación de Aprobación y Cumplimiento del Proyecto Mejoramiento del control de ventas y difusión de información de la empresa VidiVidi firmada por el Gerente.	Disposición del directivo para distintas pruebas Presupuesto suficiente para la implementación y uso del Sistema Web

Componentes			
“ Sistema Informático de gestión de ventas y pedidos vía Web, desarrollado”	Al finalizar el proyecto se ha concluido el desarrollo del sistema informático de acuerdo a las especificaciones de requerimiento cumpliendo al menos con un 90% de la norma IEE	Certificado de cumplimiento del sistema firmado por el docente de Taller III. Lista de asistencia previamente firmada por los asistentes.	El personal tendrá interés en la capacitación del Sistema
Socialización y Capacitación del personal para el uso del Sistema VidiVirtual	E 830 (Ver Anexo I)	Certificado de Capacitación al personal de VidiVidi firmado por el gerente general.	

	A finales de julio del 2012 el 80% del personal de VidiVidi participaron de una conferencia de socialización en la cual se explicó el uso y las ventajas del Sistema VidiVirtual A finales de julio de 2012 se ha capacitado a los tres cajeros de las sucursales de VidiVidi en el uso del sistema VidiVirtual con una duración de 1 hora. A finales de julio de 2012	Lista de asistencia previamente firmada por el gerente. Certificado de Capacitación al gerente de VidiVidi firmado por el gerente general. Fotografías de la capacitación.	
--	---	---	--

	se ha capacitado al gerente general de la empresa VidiVidi en el uso del sistema VidiVirtual con una duración de 1 hora																																																																																																																															
Actividades Sistema Informático 1) Especificación de requerimientos 2) Análisis y Diseño del Sistema. 3) Programación del Sistema.	Actividades <table><tr><th>ITEM</th><th>RUBROS</th><th>Aporte Universidad</th><th>Otro Aporte</th><th>TOTAL (Bs.)</th></tr><tr><td>10000</td><td>SERVICIOS PERSONALES</td><td></td><td></td><td></td></tr><tr><td></td><td>12000 Empleados no Permanentes</td><td></td><td></td><td></td></tr><tr><td></td><td>Sub total rubro</td><td></td><td></td><td></td></tr><tr><td>20000</td><td>SERVICIOS NO PERSONALES</td><td></td><td></td><td></td></tr><tr><td></td><td>21000. Servicios Básicos</td><td></td><td></td><td>1800</td></tr><tr><td></td><td>22000. Servicios de transporte</td><td></td><td></td><td></td></tr><tr><td></td><td>23000. Alquileres</td><td></td><td></td><td></td></tr><tr><td></td><td>24000. Mantenimiento y reparaciós</td><td></td><td></td><td></td></tr><tr><td></td><td>25000. Servicios Profesionales y Comerciales</td><td></td><td></td><td>3100</td></tr><tr><td></td><td>Sub total rubro</td><td></td><td></td><td>4900</td></tr><tr><td>30000</td><td>MATERIALES Y SUMINISTROS</td><td></td><td></td><td></td></tr><tr><td></td><td>31000. Alimentos y Productos Forestales</td><td></td><td></td><td>170</td></tr><tr><td></td><td>32000. Productos de Papel, Cartón e Impresos</td><td></td><td></td><td>320</td></tr><tr><td></td><td>33000. Textiles y Vestuario.</td><td></td><td></td><td></td></tr><tr><td></td><td>34000. Productos Químicos, Combustibles y Lubricantes</td><td></td><td></td><td></td></tr><tr><td></td><td>39000. Productos Varios.</td><td></td><td></td><td>980</td></tr><tr><td></td><td>Sub total rubro</td><td></td><td></td><td>1170</td></tr><tr><td>40000</td><td>ACTIVOS REALES</td><td></td><td></td><td></td></tr><tr><td></td><td>43000. Maquinaria y Equipo</td><td></td><td></td><td>800</td></tr><tr><td></td><td>46000. Descripciön de estudios y proyectos para inversión</td><td></td><td></td><td></td></tr><tr><td></td><td>49000. Otros Activos</td><td></td><td></td><td>1200</td></tr><tr><td></td><td>Sub total rubro</td><td></td><td></td><td>2000</td></tr><tr><td></td><td>TOTAL</td><td></td><td></td><td>8070</td></tr><tr><td></td><td>TOTAL + 40% Incentivo</td><td></td><td></td><td>11298</td></tr></table>	ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)	10000	SERVICIOS PERSONALES					12000 Empleados no Permanentes					Sub total rubro				20000	SERVICIOS NO PERSONALES					21000. Servicios Básicos			1800		22000. Servicios de transporte					23000. Alquileres					24000. Mantenimiento y reparaciós					25000. Servicios Profesionales y Comerciales			3100		Sub total rubro			4900	30000	MATERIALES Y SUMINISTROS					31000. Alimentos y Productos Forestales			170		32000. Productos de Papel, Cartón e Impresos			320		33000. Textiles y Vestuario.					34000. Productos Químicos, Combustibles y Lubricantes					39000. Productos Varios.			980		Sub total rubro			1170	40000	ACTIVOS REALES					43000. Maquinaria y Equipo			800		46000. Descripciön de estudios y proyectos para inversión					49000. Otros Activos			1200		Sub total rubro			2000		TOTAL			8070		TOTAL + 40% Incentivo			11298	Informe de Desembolso por parte del Director del Proyecto. Documentación del análisis y diseño del sistema. Utilización de las normas de modelado a los diagramas de	Disponibilidad de Equipos de computación. Disponibilidad de un Web hosting. Existencia y disponibilidad de recursos
ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)																																																																																																																												
10000	SERVICIOS PERSONALES																																																																																																																															
	12000 Empleados no Permanentes																																																																																																																															
	Sub total rubro																																																																																																																															
20000	SERVICIOS NO PERSONALES																																																																																																																															
	21000. Servicios Básicos			1800																																																																																																																												
	22000. Servicios de transporte																																																																																																																															
	23000. Alquileres																																																																																																																															
	24000. Mantenimiento y reparaciós																																																																																																																															
	25000. Servicios Profesionales y Comerciales			3100																																																																																																																												
	Sub total rubro			4900																																																																																																																												
30000	MATERIALES Y SUMINISTROS																																																																																																																															
	31000. Alimentos y Productos Forestales			170																																																																																																																												
	32000. Productos de Papel, Cartón e Impresos			320																																																																																																																												
	33000. Textiles y Vestuario.																																																																																																																															
	34000. Productos Químicos, Combustibles y Lubricantes																																																																																																																															
	39000. Productos Varios.			980																																																																																																																												
	Sub total rubro			1170																																																																																																																												
40000	ACTIVOS REALES																																																																																																																															
	43000. Maquinaria y Equipo			800																																																																																																																												
	46000. Descripciön de estudios y proyectos para inversión																																																																																																																															
	49000. Otros Activos			1200																																																																																																																												
	Sub total rubro			2000																																																																																																																												
	TOTAL			8070																																																																																																																												
	TOTAL + 40% Incentivo			11298																																																																																																																												

4) Configuración e implementación del sistema al Servidor 5) Elaboración del Informe final sobre el Desarrollo del Sistema. Capacitación al Personal 1. Definición de metodologías de enseñanza para la socialización 2. Planificación del contenido temático de los cursos de		análisis y diseño del sistema, mediante la metodología de desarrollo de sistemas RUP. Verificación de integridad referencial de la base de datos. Manual de usuario. Manual de instalación.	económicos. Disponibilidad del uso de medios de transporte motorizados Disponibilidad de línea telefónica.
--	--	---	---

capacitación.			
---------------	--	--	--

Tabla 4 Matriz de Marco Lógico

1.4 Presupuesto / Justificación

ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)
10000	SERVICIOS PERSONALES			
	12000 Empleados no Permanentes			
	Sub total rubro			
20000	SERVICIOS NO PERSONALES			
	21000. Servicios Básicos			1800
	22000. Servicios de transporte			
	23000. Alquileres			
	24000. Mantenimiento y reparación			
	25000. Servicios Profesionales y Comerciales			3100
	Sub total rubro			4900
30000	MATERIALES Y SUMINISTROS			
	31000. Alimentos y Productos Forestales			170
	32000. Productos de Papel, Cartón e Impresos			320
	33000. Textiles y Vestuario.			
	34000. Productos Químicos, Combustibles y Lubricantes			
	39000. Productos Varios.			980
	Sub total rubro			1170
40000	ACTIVOS REALES			
	43000. Maquinaria y Equipo			800
	46000. Descripción de estudios y proyectos para inversión			
	49000. Otros Activos			1200
	Sub total rubro			2000
	TOTAL			8070
	TOTAL + 40% Incentivo			11298

Tabla 5 Presupuesto / Justificación

GRUPO 10000. SERVICIOS PERSONALES**a) SUB GRUPO 12000. Empleados no Permanentes**

Partida	Personal	Remuneración	Tiempo/meses	Total
12100	Personal Eventual			
Total				

* Se refiere a gastos para remunerar a personas sujetas a contrato dependientes según la necesidad de cada entidad

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	100	8	800
21200	Energía Eléctrica	60	8	480
21300	Agua			
21400	Servicios Telefónicos	250	8	2000
Total				3280

* 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
25200	Estudios e Investigaciones				
25500	Publicidad		150	1	150
25600	Certificados de participación	20	2		40
	Empastados	3	30		90
	Gigantografías	5	10		50
25700	Capacitación de Personal	20	10	1	200
25800	Estudios e Investigaciones Para Proyectos de Inversión				
Total					530

* 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 gastos Alimentos y Productos Agroforestales

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
31110	Refrigerios y Gastos Administrativos			
31200	Alimento para Animales			
31300	Productos Agroforestales y Pecuarios			
Total				

* Se refiere a la adquisición de materiales y bienes como: alimentos y productos agroforestales, bebidas para personas, 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 Hojas Bond	5	35	175
32200	Productos de Artes Gráficas, Papel y Cartón			
32300	Libros y Revistas			
32400	Textos de Enseñanza			
32500	Periódicos			
Total				175

* 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			
	Dvd en Blanco	10	2,5	25
	Cartucho de tinta Negra	3	25	75
	Agenda Personal	1	30	30
39700	Útiles y Materiales Eléctricos			
39800	Otros Repuestos y Accesorios			
Total				130

*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 Impresora	1	260	260
43200	Maquinaria y Equipo de Producción			
43300	Equipos de Transporte, Tracción y Elevación			
43400	Equipo Médico y de Laboratorio			
43600	Equipo Educacional y Recreativo			
43700	Otra Maquinaria y Equipo			
Total				260

* 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			
	Servidor Web	1	1200	1200
49200	Compra de Bienes Muebles Existentes (Usados)			
49300	Semovientes y otros Animales			
49900	Otros Activos			
Total				1200

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

1.4.1 Curriculum Viate

1.4.1.1 Antecedentes personales

Céspedes Apellido Paterno	Gira Apellido Materno	Alex Daniel Nombre	5815471 Tja C.I.
04/02/1989 Fecha de Nacimiento	M...X... F..... Sexo	B/ 6 de Agosto C/ Libertad Nro. 1327 Dirección	

Tabla 6 Antecedentes Personales

1.4.1.2 Antecedentes académicos

Título obtenido	Universidad	País	Año
Diplomado en Pedagogía y Didáctica en la Educación Superior.	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2007
Ing. Informática - 1º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2007
Ing. Informática - 2º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2007
Ing. Informática - 3º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2008
Ing. Informática - 4º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2008
Ing. Informática - 5º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 6º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 7º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2010
Ing. Informática - 8º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2010
Ing. Informática - 9º	“Universidad Autónoma Juan	Bolivia	2011

Semestre	Misael Saracho”.		
Ing. Informática - 10º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2011

Tabla 7 Antecedentes Académicos

1.4.1.3 Participación en proyectos de Investigación

Título proyecto	Institución	Cargo	Año

Tabla 8 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 9 Publicaciones Realizadas

1.4.1.5 Antecedentes en Ayudantías

Carrera	Semestre	Año
Ing. Informática	3º Semestre	2008
Ing. Informática	4º Semestre	2009
Ing. Informática	5º Semestre	2009
Ing. Informática	5º Semestre	2010

Ing. Informática	7º Semestre	2010
Ing. Informática	8º Semestre	2009
Ing. Informática	9º Semestre	2010

Tabla 10 Antecedentes en Ayudantías

1.4.1.6 Cursos Realizados

Curso	Institución	Duración en Hrs	Año
Técnico Superior en Ingles	Centro Boliviano Americano	750	2006
Seminario de oratoria	Institución Boliviana de Capacitación	20	2006
Relaciones Humanas para el Desarrollo Personal.	Escuela de liderazgo y desarrollo humano.	20	2006
XIV Congreso Nacional de ciencias de la computación – Conferencias.	Universidad Mayor de San Andrés.	20	2007

Tabla 11 Cursos Realizados

II. Capítulo II

2.1 Componente 1: Sistema Informático para el Mejoramiento de Ventas

2.1.1 Marco Teórico

2.1.1.1 Introducción

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.

A modo de resumen de este antecedente de marco teórico, 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 Empresa de Comida Rápida Vidi Vidi

2.1.1.2.1.1 Misión

Empresa tarijeña dedicada a la elaboración de productos de comida rápida orientados a satisfacer la necesidad de alimentación de la población con productos de calidad traducida en la higiene y sabor de los mismos y apoyada en un servicio de atención especializada en el marco de la responsabilidad social.

2.1.1.2.1.2 Visión

Cadena de restaurantes de comida rápida líder en el sur del país posicionados en la mente del consumidor con altos estándares de calidad en cuanto a higiene, sabor, rapidez y servicio especializado a nuestros clientes.

2.1.1.2.1.3 Objetivo General

Consolidarnos en el mercado Tarijeño de Comida Rápida como una empresa reconocida por la calidad en cuanto a rapidez del servicio, higiene y sabor de nuestros productos de manera que se pueda colmar eficientemente los requerimientos del cliente.

2.1.1.2.1.4 Matriz Foda

El FODA se representa a través de una matriz de doble entrada, llamada matriz FODA, en la que el nivel horizontal se analiza los factores positivos y los negativos.

En la lectura vertical se analizan los factores internos y por tanto controlables del programa o proyecto y los factores externos, considerados no controlables.

Las Fortalezas son todos aquellos elementos internos y positivos que diferencian al programa o proyecto de otros de igual clase.

Las Oportunidades son aquellas situaciones externas, positivas, que se generan en el entorno y que una vez identificadas pueden ser aprovechadas

Las Debilidades son problemas internos, que una vez identificados y desarrollando una adecuada estrategia pueden y deben eliminarse.

Las Amenazas son situaciones negativas, externas al programa o proyecto, que pueden atentar contra éste, por lo que llegado al caso, puede ser necesario diseñar una estrategia adecuada para poder sortearla.

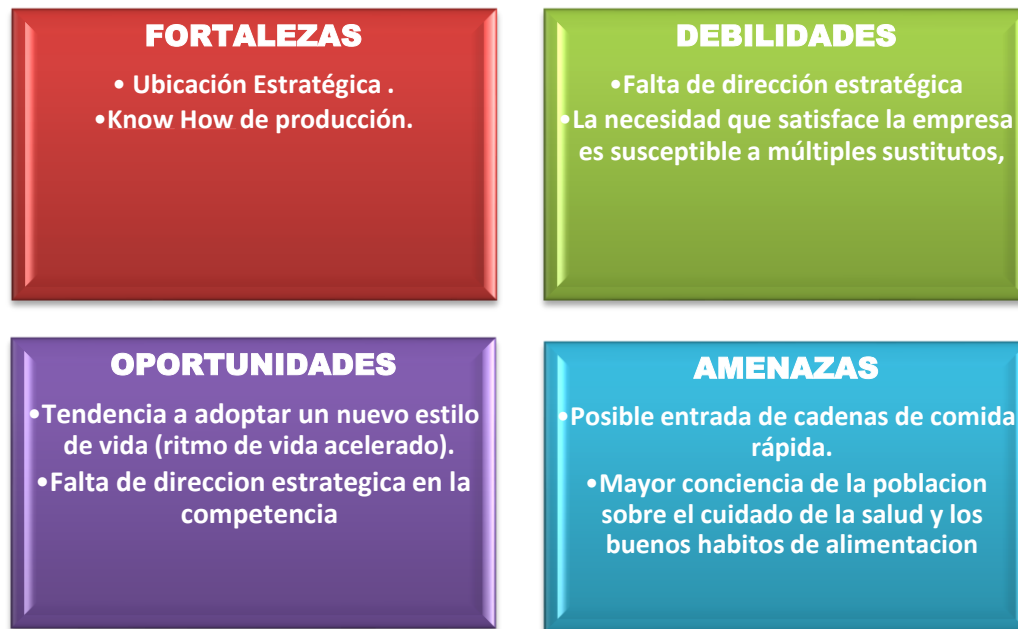


Figura 5 Matriz FODA

2.1.1.2.1.5 Diagnóstico Estratégico

El análisis realizado permite detectar que VidiVidi se encuentra compitiendo en un mercado complejo ya que la necesidad que satisface la empresa es susceptible a diversos sustitutos, convirtiéndose esto en una falencia de la empresa.

El análisis es útil para que la empresa conozca qué características deba tomar en cuenta para afrontar estratégicamente y cuáles son sus principales fortalezas que

motiven a la empresa seguir adelante y mejorar, optimizar y estandarizar los servicios y productos que ofrece.

Es importante destacar que el ritmo de vida acelerado de las personas favorece a este tipo de empresa que por la rapidez de la elaboración de sus productos el sistema de pedidos permite agilizar el tiempo de espera y brindar un servicio rápido y eficiente.

Es imprescindible que la empresa cuente con talento humano calificado que le permita desarrollar sus actividades de la mejor manera logrando así la satisfacción plena de quienes confían en el servicio y los productos que VidiVidi ofrece.

Además que el Know How de la empresa para la elaboración de los productos le permite crear una ventaja competitiva que se transforma en la percepción aceptable que las personas tiene de la empresa.

Debido al crecimiento del departamento en los últimos años Tarija se ha visto beneficiada con el turismo de las personas que en épocas de vacaciones y cuando el clima es favorable visitan esta ciudad y la cantidad de personas en el departamento aumenta significativamente a finales de año, por tanto VidiVidi tiene la oportunidad de llegar a este publico debido a la estética de sus locales y la calidad (higiene y sabor) de sus productos.

2.1.1.3 Metodología del Desarrollo

2.1.1.3.1 Metodología RUP (Rational 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 en cuál de los flujos se hace mayor énfasis según la fase, cabe destacar el flujo de trabajo concerniente al negocio.

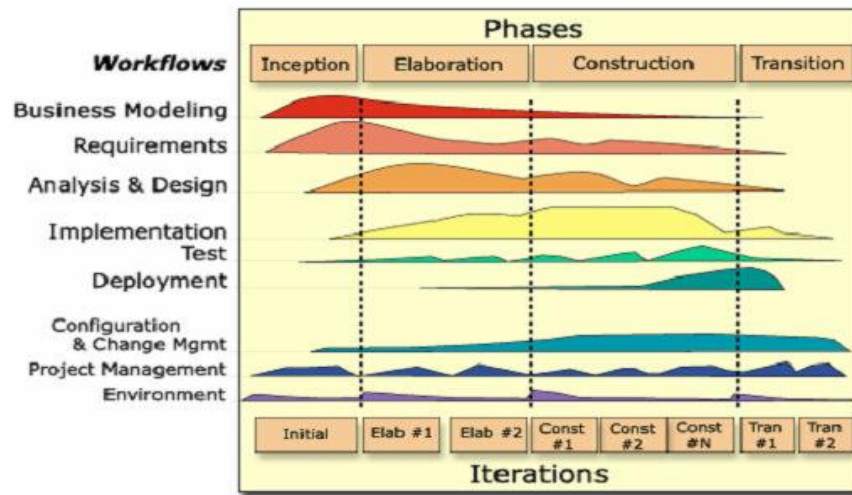


Figura 6 Fases en el Ciclo de Desarrollo de RUP

- **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.

2.1.1.3.2 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.

A 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 la docena de 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).

2.1.1.3.2.1 Tipos de Diagramas Utilizados

2.1.1.3.2.1.1 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.

2.1.1.3.2.1.2 Diagramas de Casos de Uso

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.

2.1.1.3.2.1.3 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.

2.1.1.3.2.1.4 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.

2.1.1.3.2.1.5 Diagramas 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.

2.1.1.3.2.1.6 Diagramas 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.

2.1.1.4 Herramientas de Construcción de Software

2.1.1.4.1 Eclipse Galileo

Eclipse 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.

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).

Eclipse es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación 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 7 Eclipse Galileo

2.1.1.4.2 Flex Builder

Flex Builder les permite a los diseñadores y desarrolladores disponer y crear aplicaciones Flex de una forma más productiva. Los diseñadores pueden crear con agilidad prototipos para las interfaces de aplicaciones, disponiendo interfaces de usuarios Flex de una manera rápida y fácil y conectándolas a fuentes de datos del sistema central. Los desarrolladores pueden entonces codificar y depurar aplicaciones Flex de una forma productiva usando el código sugerido y las herramientas de depuración. Con Flex Builder, todos los miembros de su equipo de trabajo podrán producir con más eficiencia aplicaciones Flex dinámicas de Internet.



Figura 8 Flex Builder

2.1.1.4.3 Action Script 3.0

Es el lenguaje de programación de la Plataforma Adobe Flash. Originalmente desarrollado como una forma para que los desarrolladores programen de forma más interactiva. La programación con ActionScript permite mucha más eficiencia en las aplicaciones de la plataforma Flash para construir animaciones de todo tipo, desde simples a complejas, ricas en datos e interfaces interactivas.

Según el equipo de desarrollo detrás de ActionScript 3.0, los objetivos que se persiguen podrían resumirse en los siguientes términos:

- **Seguridad**

El código será más fácil de mantener, eliminando ambigüedades y permitiendo una escritura más clara y concisa.

- **Simplicidad**

El lenguaje será lo suficientemente intuitivo como para que el programador pueda escribir código sin tener que consultar el manual de referencia constantemente.

- **Performance**

El lenguaje habilitará a los programadores para escribir complejos programas que funcionen de forma más eficiente y responsable.

- **Compatibilidad**

Como dialecto de la especificación ECMAScript, el lenguaje estará provisto de una mayor compatibilidad con los estándares de la industria. Se pretende lograr un lenguaje más coherente y unificado.

2.1.1.4.4 Log4j

Log4j es una biblioteca open source desarrollada en Java por la Apache Software Foundation que permite a los desarrolladores de software elegir la salida y el nivel de granularidad de los mensajes o “logs” a tiempo de ejecución y no a tiempo de compilación como es comúnmente realizado. La configuración de salida y granularidad de los mensajes es realizada a tiempo de ejecución mediante el uso de archivos de configuración externos.

2.1.1.4.5 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.

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



Figura 9 Visual Paradigm

2.1.1.4.6 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 Págs (JSP) de Sun Microsystems.

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 Java WebServer (actualmente sustituido por iPlanet). Este el modo por defecto usado por Tomcat.

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

Contenedores de Servlets dentro-de-Proceso

El contenedor Servlets es una combinación de un plugin para el servidor web y una implementación de contenedor Java. El plugin 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 servlets, el plugin 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.).

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 desplaguemos Tomcat tendremos que decidir cómo usarlo, y, si seleccionamos las opciones 2 o 3, también necesitaremos instalar un adaptador de servidor web.

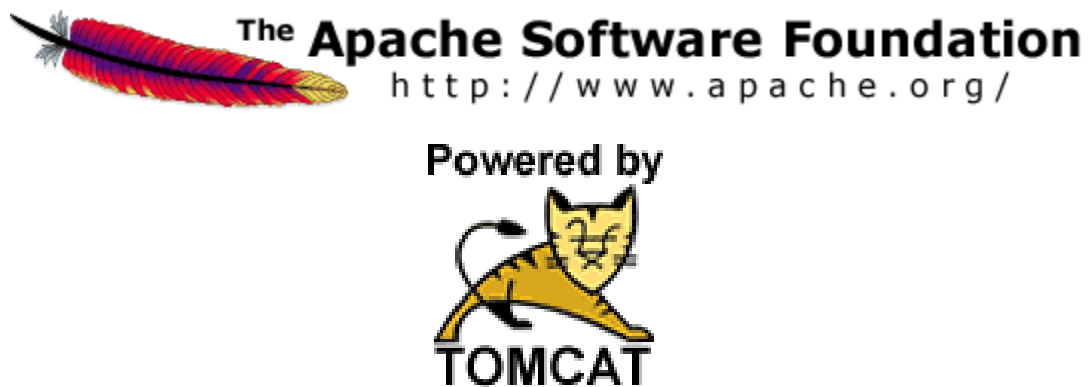


Figura 10 Apache Tomcat

2.1.1.4.7 MySQL

MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB. MySQL AB es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL.

MySQL es un sistema de administración de bases de datos. MySQL usa la licencia GPL (Licencia Pública General GNU), para definir qué es lo que se puede y no se puede hacer con el software para diferentes situaciones.

El software de bases de datos MySQL consiste de un sistema cliente/servidor que se compone de un servidor SQL multihilo, varios programas clientes y bibliotecas, herramientas administrativas, y una gran variedad de interfaces de programación (APIs). Se puede obtener también como una biblioteca multihilo que se puede enlazar dentro de otras aplicaciones para obtener un producto más pequeño, más rápido, y más fácil de manejar.



Figura 11 MySQL

2.1.1.5 Técnica

2.1.1.5.1 Adobe Flash

Flash es la tecnología más comúnmente utilizada en el Web que permite la creación de animaciones vectoriales. El interés en el uso de gráficos vectoriales es que éstos permiten llevar a cabo animaciones de poco peso, es decir, que tardan poco tiempo en ser cargadas por el navegador.

Como puede que sepáis, existen dos tipos de gráficos:

Los gráficos vectoriales, en los cuales una imagen es representada a partir de líneas (o vectores) que poseen determinadas propiedades. La calidad de este tipo de gráficos no depende del zoom o del tipo de resolución con el cual se esté mirando el gráfico. Por mucho que nos acerquemos, el gráfico no se pixelea, ya que el ordenador traza automáticamente las líneas para ese nivel de acercamiento.

Las imágenes en mapa de bits, este tipo de gráficos se asemejan a una especie de cuadrícula en la cual cada uno de los cuadrados (píxeles) muestra un color determinado. La información de estos gráficos es guardada individualmente para cada píxel y es definida por las coordenadas y color de dicho píxel. Este tipo de gráficos son dependientes de la variación del tamaño y resolución, pudiendo perder calidad al modificar sucesivamente sus dimensiones.

2.1.1.5.2 Flex

Flex es simplemente un Framework orientado al desarrollo de RIA'S. Hablando en plata, es una "herramienta" que nos da la facilidad de desarrollar ciertos tipo de cosas, en concreto las RIAS.

Flex se basa en componentes que implementan mucha funcionalidad básica y avanzada. Funcionalidad necesaria para cualquier aplicación, y que nos costaría muchas horas desarrollar con Flash. Ya que en Flash, a parte de sus componentes, lo tenemos que hacer todo a mano, es mucho más "artesano". Por poner algún ejemplo, estamos hablando de sistemas de navegación, posicionamiento en pantalla, diseños líquidos, deep linking, validación de formularios, comunicación con bases de datos, Flash remoting, servicios web, y un largo etcétera que Flash no incorpora por defecto. Y en definitiva es eso, una ayuda para desarrollar, que también tiene sus limitaciones. En Flex no se diseña ni se dibuja, ni se anima en línea de tiempo. Por eso los diseñadores están lejos de usarlo. Pero si alguna vez has tenido que programar un proyecto medianamente grande, seguro que entenderás perfectamente las ventajas que

comento.

Para dejar algo “bonito” en Flex se usan métodos como el skineado de componentes, adjuntar hojas de estilo CSS, etc. Nos da menos libertad para crear efectos visuales, pero nos facilita la vida en otros sentidos. Además, dentro de una aplicación Flex podemos incrustar cualquier cosa hecha con Flash. Desde assets gráficos hasta aplicaciones enteritas.

2.1.1.5.3 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

2.1.1.5.4 Spring

El Spring Framework es un framework de código abierto de desarrollo de aplicaciones para la plataforma Java.

A pesar de que Spring Framework no obliga a usar un modelo de programación en particular, se ha popularizado en la comunidad de programadores en Java al ser considerado como una alternativa y sustituto del modelo de Enterprise JavaBeans. Por su diseño el framework ofrece mucha libertad a los desarrolladores en Java y soluciones muy bien documentadas y fáciles de usar para las prácticas comunes en la industria.

Mientras que las características fundamentales de este framework pueden emplearse en cualquier aplicación hecha en Java, existen muchas extensiones y mejoras para construir aplicaciones basadas en web por encima de la plataforma empresarial de Java (Java Enterprise Platform).

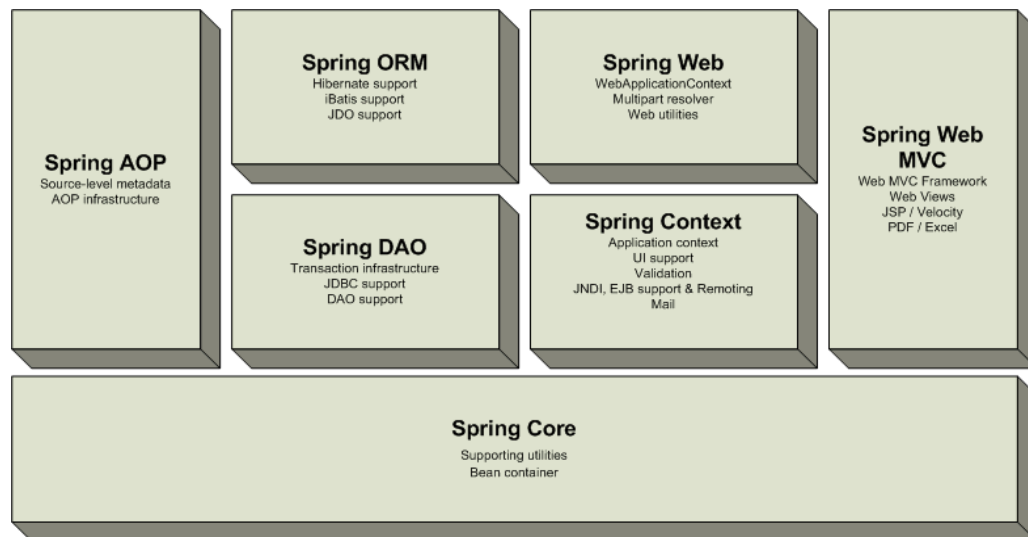


Figura 12 Arquitectura de Spring

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.

2.1.1.5.5 BlazeDS

Es un servidor remoto, hecho con JAVA para la interacción entre los lenguajes ActionScript 3.0 y JAVA, sirve como un puente de comunicación que se encarga de “traducir” de un lenguaje a otro, para que la interacción entre lenguajes y tecnologías sea transparente y no sea necesario hacer trucos raros a la hora de enviar información de una lado para otro. Permite a los desarrolladores conectar aplicaciones Adobe Flex y Adobe AIR con entornos empresariales Java.

2.1.1.5.6 Cairngorm

Cairngorm es un framework estructural para el desarrollo de RIAs que fomenta el uso de determinados patrones y potencia la escalabilidad y el crecimiento de las aplicaciones.

Para afrontar el desarrollo de cualquier tipo de aplicación medianamente grande y en

el que vayan a co-desarrollar múltiples desarrolladores es imprescindible seguir unas pautas y estrategias comunes, conocer con exactitud el alcance e impacto de cada uno de los pasos que se den. Cairngorm, al ser un framework estructural y transversal a cualquier aplicativo, permite que desarrolladores centrados en distintos ámbitos de aplicaciones trabajen con una misma metodología y unas mismas bases que les permita tener una movilidad y un know-how entre distintos proyectos.

El hecho que se trate de un framework transversal tiene múltiples implicaciones:

- Un desarrollador que conozca cairngorm puede incorporarse a cualquier proyecto basado en cairngorm minimizando la curva de aprendizaje inicial.
- El know-how adquirido en otros proyectos basados en Cairngorm es aplicable de forma directa a cualquier otro proyecto.
- En un equipo de desarrolladores, y aunque no hayan trabajado con anterioridad juntos, la metodología es conocida por todos. El lenguaje ya está dominizado y los conceptos son comunes a todos los integrantes. Con esto se consigue un mayor grado de afinidad y agiliza el planteamiento de los problemas.
- Se fomenta la reutilización real de código de forma transversal a los proyectos sin hacer copiar y pegar (para hacer mínimos retoques de integración)

2.1.1.5.7 Spring 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.

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.

Hibernate no solo 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.

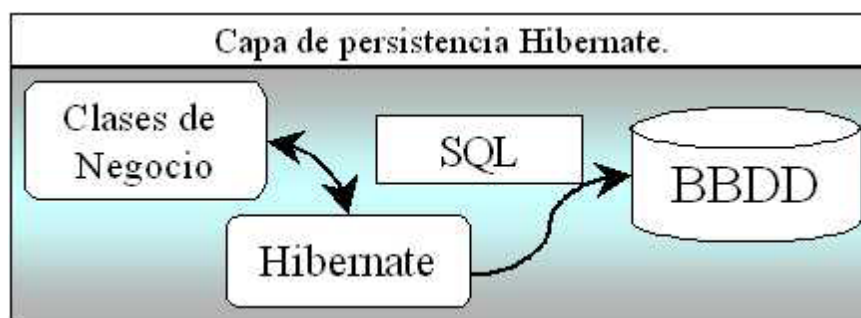


Figura 13 Arquitectura de Hibernate

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.
- Consultas 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 2 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

2.1.1.6 Base de Datos

2.1.1.6.1 Componentes principales de una Base de Datos

Datos: Los datos son la Base de Datos propiamente dicha.

Hardware: Se refiere a los dispositivos de almacenamiento en donde reside la base de datos así como los dispositivos periféricos (Unidad de Control, Canales de Comunicación, etc.) necesarios para su uso.

Software: Está constituido por un conjunto de programas que se conoce como Sistema Manejador de Base de Datos (DBMS), manejando éste todas las solicitudes formuladas por los usuarios a la base de datos.

Usuarios: Normalmente identificándose 3 tipos:

- El programador de aplicaciones.
- El usuario Final.
- El Administrador de la Base de Datos quien se encarga del control general del Sistema de Base de Datos.

2.1.1.6.2 Herramienta MySql Administrator para el manejo de la BD

MySQL Administrador es el nuevo software de administración de servidores de Bases de Datos de MySQL que ha creado MySQL AB. Se trata de un software multiplataforma, que por el momento se encuentra disponible para Linux y Microsoft Windows y que cuenta con un entorno gráfico de usuario muy intuitivo.

Este nuevo producto suple las carencias que tiene MySQL Control Center en el área de Administración de servidores. MySQL Control Center en estos momentos está deprecado, ha quedado obsoleto y no sigue desarrollándose. Se ha sustituido por el conjunto de programas MySQL Administrator y MySQL Query Browser.

Algunas utilidades accesibles desde la ventana principal del programa son:

- Service control: Inicio y detención de servidores (sólo accesible si se ha conectado con un servidor MySQL en la máquina local).

- Startup variables: Configuración del servidor y las variables de inicio (sólo accesible si se ha conectado con un servidor MySQL en la máquina local).
- User Administration: Para la gestión de usuarios y permisos.
- Server connections: Visualiza y gestiona las conexiones abiertas con el servidor de bases de datos.
- Health: Información sobre la carga del servidor
- Server Logs: El historial de logs del servidor.
- Replication Status: Con información de los sistemas replicados.
- Backup: Para hacer una copia de seguridad de las bases de datos.
- Restore: Para restaurar las copias de seguridad.
- Catalogs: Para mostrar las bases de datos, visualizar, crear y editar las tablas.

2.1.1.7 Sistema de Información Automatizado

2.1.1.7.1 La 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.

2.1.1.7.2 La World Wide 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 ordenes 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.

2.1.1.8 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 del Software

2.1.2.1 Introducción

Este Plan de Desarrollo del Software es una versión corregida para ser incluida en la propuesta elaborada como 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 será desarrollado por el universitario: Alex Daniel Céspedes Gira basado en una metodología de Rational Unified Process (RUP) 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, adicionalmente se esbozarán las fases posteriores de Construcción para dar una visión global de todo proceso.

El enfoque del desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes,

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 dar a conocer 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 lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
- Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.
- El docente para evaluar el cumplimiento del proyecto.

2.1.2.1.2 Alcance

Con el Plan de Desarrollo del Software se pretende analizar y elaborar un proyecto de gran magnitud abarcando todas las fases requeridas en la terminología RUP. Dichas fases tendrán un cronograma de cada una de las actividades a realizar. También se especificará los detalles de construcción del proyecto para los distintos roles que cumplen los desarrolladores.

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 reuniones, entrevistas y charlas que se han realizado con los usuarios destino de la empresa de comida rápida VidiVidi.

2.1.2.2.1.1 Propósito

Con el presente proyecto se pretende Mejorar “el control de ventas y difusión de información de la empresa de comida rápida Vidi-Vidi”.

2.1.2.2.1.2 Alcance

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo de VidiVirtual un Sistema Informático que está comprendido por los siguientes Módulos:

- Modulo Administrador.
- Modulo Pedidos y Ventas.
- Modulo Reportes.

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.
- Desarrollar un manual de usuario.

2.1.2.2.1.3 Objetivo

2.1.2.2.1.3.1 Objetivo General

Mejorar el “control de ventas y difusión de información de la empresa de comida rápida Vidi-Vidi”.

2.1.2.2.1.3.2 Objetivo Específico

- Obtener un sistema para el uso en la gestión de ventas y difusión de información.
- Crear una base de datos de acuerdo a las necesidades de la empresa.
- Brindar información al cliente acerca de productos, promociones, ubicación de nuestros puntos de venta, etc.
- Implementar una interfaz para que la gente pueda hacer pedidos vía Web.

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 empresa VidiVidi, Regional Tarija cuenta con:

- La empresa dispone equipos de computación, que incluye un computador, impresora en cada punto de venta.
- Equipo de computación. En el que incluye un computador, impresora.
- Total disponibilidad de la gerencia para proporcionar información.
- Personal con conocimientos básicos de computación.

- Presupuesto suficiente para la implementación del Sistema.
- Interés de la institución para utilizar el Sistema.
- Tener una línea telefónica habilitada para poder recibir y confirmar pedidos.
- Persona encargada que sea mayor de edad con brevet y tenga un medio de transporte adecuado para ir a entregar los pedidos a domicilio.
- Medio de transporte que se ajuste a las necesidades para dejar
- La formación del personal encargado se lleve en la fecha determinada.
- Un servidor Web para la implementación del sistema.
- El administrador deberá contar con el conocimiento necesario para el manejo del sistema.
- Contar con un servidor Web que no presente las características mínimas para el correcto funcionamiento del sistema.

2.1.2.2.2 Restricciones

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

- El administrador deberá contar con el conocimiento necesario para el manejo del sistema.
- Contar con un servidor Web que presente las características mínimas para el correcto funcionamiento del sistema.
- El sistema será restringido, sólo usuarios privilegiados podrán acceder al sistema.
- Para la manipulación de la base de datos, sólo podrán acceder el personal autorizado.
- El sistema no abarca el control de la asistencia de los Empleados de la institución y no contemplará Inventario.

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 preciso 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 Diagrama de Caso de Uso del Negocio

2.1.2.2.3.2.1 Modelo de casos de Uso del Cajero

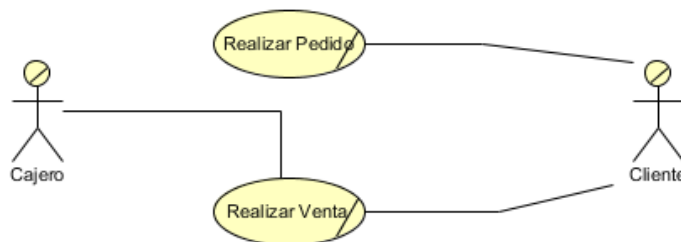


Figura 14 Modelo de casos de Uso del Cajero

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 dinámica de los Casos de Uso de Negocio.

2.1.2.2.3.3.3 Alcance

Describe los procesos del negocio.

Identificar y definir los objetos de 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: Realizar Pedido

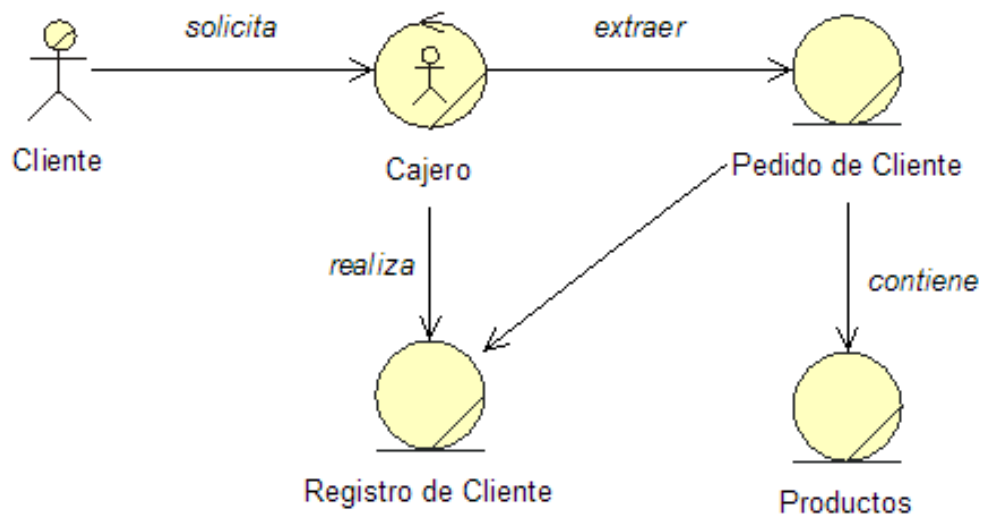


Figura 15 Modelo de Objeto del Negocio: Realizar Pedido

2.1.2.2.3.3.4.2 Modelo de Objeto del Negocio: Realizar Venta

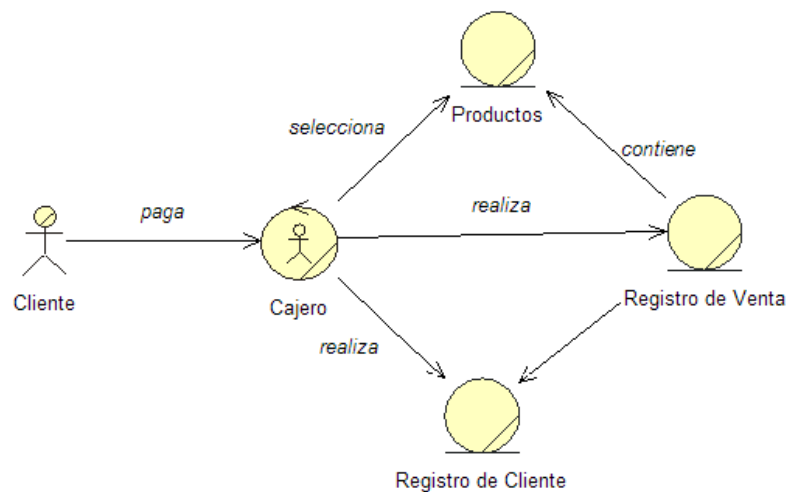


Figura 16 Modelo de Objeto del Negocio: Realizar Venta

2.1.2.2.3.4 Glosario

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

2.1.2.2.3.4.1 Introducción

El presente documento recoge los términos manejados durante la elaboración del proyecto de desarrollo del Sistema Informático para Mejoramiento de Afiliación. Se trata de un diccionario informal de datos y de definiciones de la nomenclatura que se maneja en la construcción del Sistema.

2.1.2.2.3.4.2 Propósito

El propósito del presente documento es definir la terminología manejada en el proyecto a desarrollar, también sirve como guía de consulta para la clarificación de los puntos conflictivos o poco esclarecidos del proyecto.

2.1.2.2.3.4.3 Alcance

El alcance del presente documento se extiende a todo el proyecto en desarrollo.

2.1.2.2.3.4.4 Organización del Glosario

El presente documento está organizado por definiciones de términos ordenados de forma ascendente según ordenación alfabética tradicional.

2.1.2.2.3.4.5 Definiciones

2.1.2.2.3.4.5.1 Administrador

Es aquella persona que se dedica a mantener y operar un sistema de cómputo.

2.1.2.2.3.4.5.2 Backup (copia de respaldo, copia de seguridad)

Copia de ficheros o datos de forma que estén disponibles en caso de que un fallo produzca la pérdida de los originales. Esta sencilla acción evita numerosos, y a veces irremediables, problemas si se realiza de forma habitual y periódica.

2.1.2.2.3.4.5.3 Base de Datos

Se encarga de almacenar información acerca de administradores, usuarios, y todo referente a las actividades de los usuarios, con el ordenador y componentes del sistema.

2.1.2.2.3.4.5.4 Cliente

Es la persona que podrá hacer uso del sistema en desarrollo, limitando dicho uso a una tienda Virtual e información adicional.

2.1.2.2.3.4.5.5 Diagrama

Representación gráfica de un conjunto de elementos, representando la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones).

2.1.2.2.3.4.5.6 Empleador

Persona natural o jurídica a quien se presta servicio o por cuya cuenta u orden se efectúa un trabajo, mediante contrato individual o colectivo, público o privado.

2.1.2.2.3.4.5.7 Foda

Es una herramienta de análisis estratégico, que permite analizar elementos internos o externos de programas y proyectos.

2.1.2.2.3.4.5.8 Interfaz de Usuario

La interfaz de usuario (IU) es uno de los componentes más importantes de cualquier sistema computacional, pues funciona como el vínculo entre el humano y la máquina. La interfaz de usuario es un conjunto de protocolos y técnicas para el intercambio de información entre una aplicación computacional y el usuario. La IU es responsable de solicitar comandos al usuario, y de desplegar los resultados de la aplicación de una manera comprensible. La IU no es responsable de los cálculos de la aplicación, ni del almacenamiento, recuperación y transmisión de la información.

2.1.2.2.3.4.5.9 Modelo

Abstracción que describe el sistema bajo estudio. (Un modelo puede consistir en diagramas más los textos, notaciones o aclaraciones necesarias para entenderlos).

2.1.2.2.3.4.5.10 Negocio

Satisfacer las necesidades de alimentación y socialización ofreciendo diversos productos de comida rápida, diferenciados en cuanto al sabor e higiene, brindando un ambiente confortable y familiar a través de un sistema de atención personalizada mediante el asesoramiento al cliente.

2.1.2.2.3.4.5.11 Producto

Es el producto alimenticio que la empresa VidiVidi produce para servirlos a sus clientes.

2.1.2.2.3.4.5.12 Operador de Base de Datos

Persona con acceso, a las interioridades de la BD.

2.1.2.2.3.4.5.13 RIA

RIA (Aplicaciones Ricas de Internet) es un nuevo tipo de aplicación con más ventajas que las tradicionales aplicaciones Web, las cuales pueden usarse como aplicaciones de escritorio.

2.1.2.2.3.4.5.14 RUP

RUP (Proceso Unificado de Rational) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

2.1.2.2.3.4.5.15 SQL

Structured Query Language - Lenguaje Estructurado de Consultas. Lenguaje estándar interactivo para la consulta y actualización de bases de datos.

2.1.2.2.3.4.5.16 Usuario

El usuario de un producto informático (bien sea hardware o software), es la persona a la que va destinada dicho producto una vez que ha superado las fases de desarrollo correspondientes.

2.1.2.2.3.5 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.

2.1.2.2.3.5.1 Introducción

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

2.1.2.2.3.5.2 Propósito

- Comprender la estructura y la dinámica del sistema desarrollado.
- Identificar el nivel de complejidad del sistema.
- Identificar posibles mejoras.

2.1.2.2.3.5.3 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).

Diagramas de Casos de Uso.

2.1.2.2.3.5.4 Diagramas de Caso de Uso

2.1.2.2.3.5.4.1 Casos de Usos del Sistema General

2.1.2.2.3.5.4.1.1 Modelo Casos de Uso del Módulo Administrador General

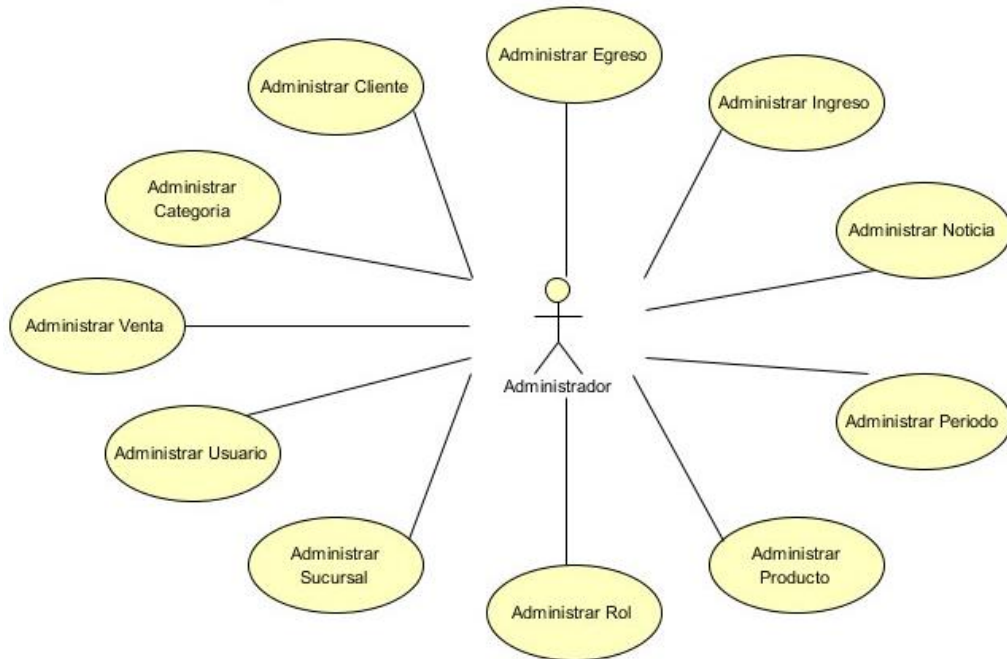


Figura 17 Modelo Casos de Uso del Módulo Administrador General

2.1.2.2.3.5.4.1.2 Modelo Casos de Uso del Módulo Venta

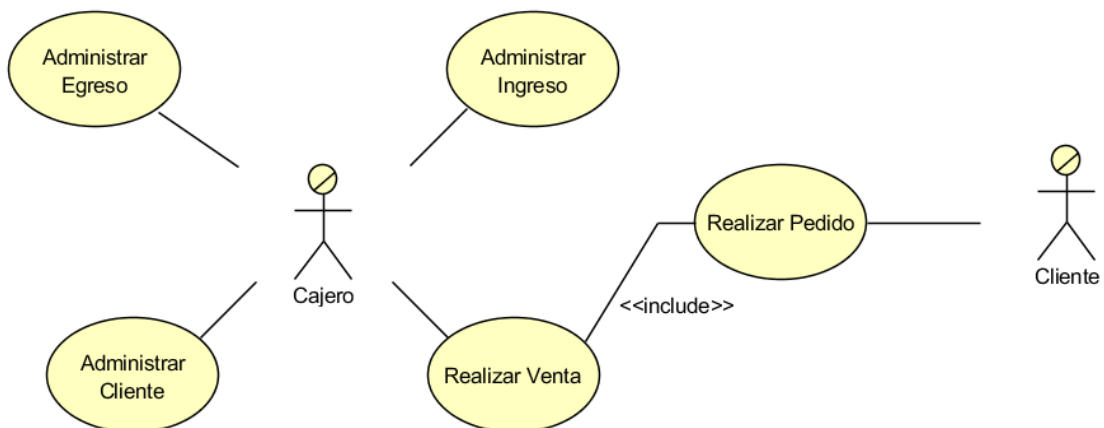


Figura 18 Modelo Casos de Uso del Módulo Venta

2.1.2.2.3.5.4.1.3 Modelo Caso de Uso del Módulo Reporte

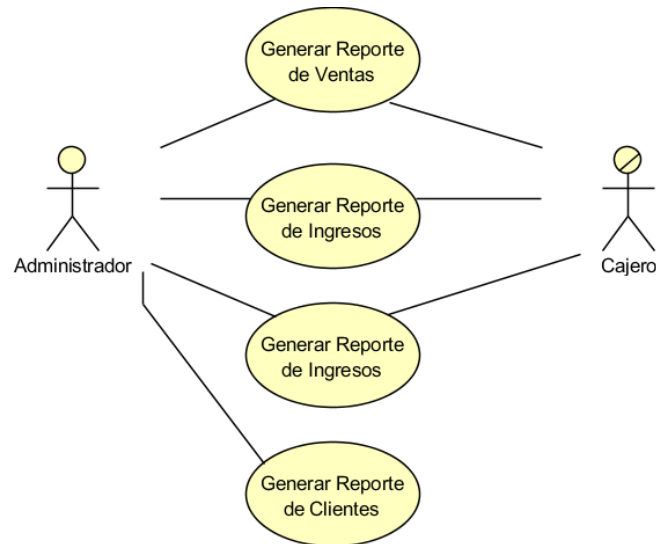


Figura 19 Modelo Caso de Uso del Módulo Reporte

2.1.2.2.3.5.4.2 Casos de Usos del Sistema Específicos

2.1.2.2.3.5.4.2.1 Modelo Caso de Uso Ingresar al Sistema

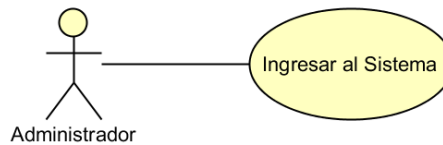


Figura 20 Modelo Caso de Uso Ingresar al Sistema

2.1.2.2.3.5.4.2.2 Modelo Caso de Uso Gestionar Categoría

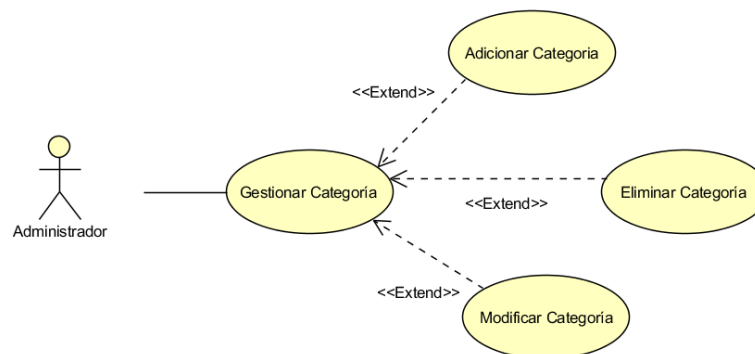


Figura 21 Modelo Caso de Uso Gestionar Categoría

2.1.2.2.3.5.4.2.3 Modelo Caso de Uso Gestionar Cliente

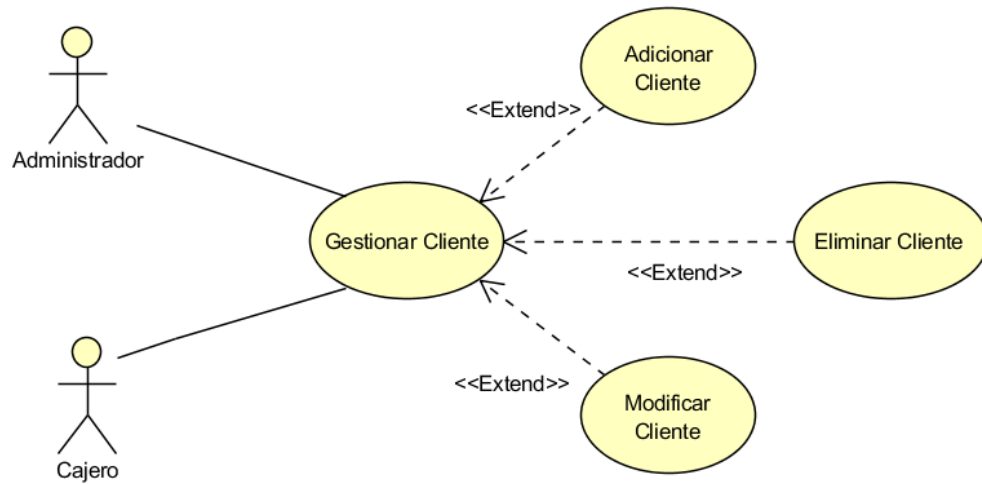


Figura 22 Modelo Caso de Uso Gestionar Cliente

2.1.2.2.3.5.4.2.4 Modelo Caso de Uso Gestionar Egreso

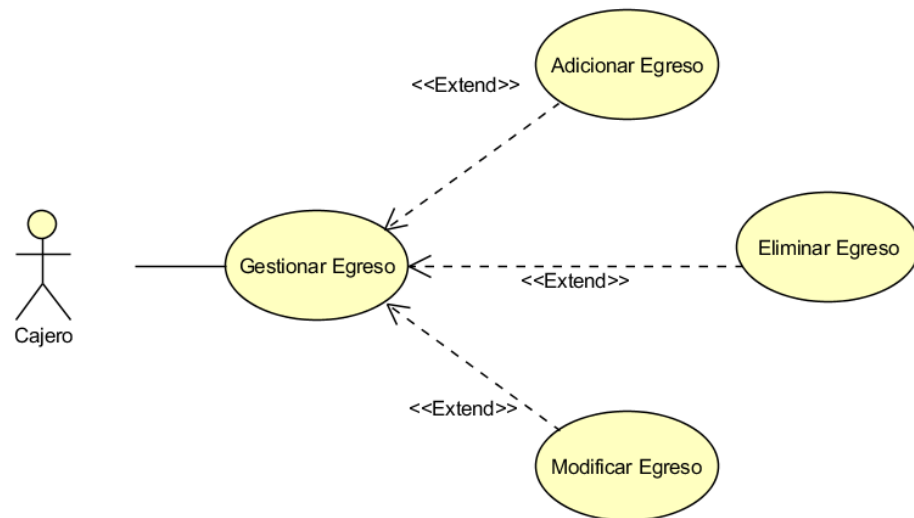


Figura 23 Modelo Caso de Uso Gestionar Egreso

2.1.2.2.3.5.4.2.5 Modelo Caso de Uso Gestionar Ingreso

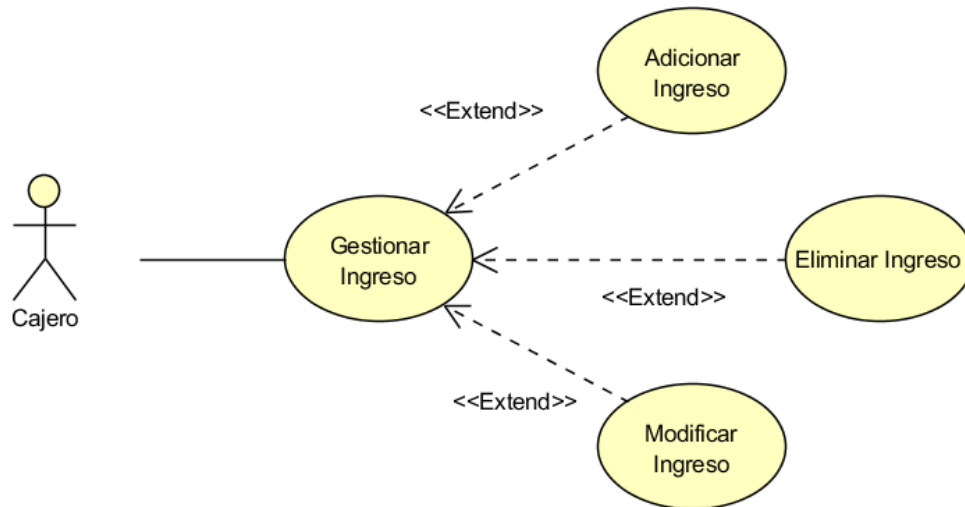


Figura 24 Modelo Caso de Uso Gestionar Ingreso

2.1.2.2.3.5.4.2.6 Modelo Caso de Uso Gestionar Noticia

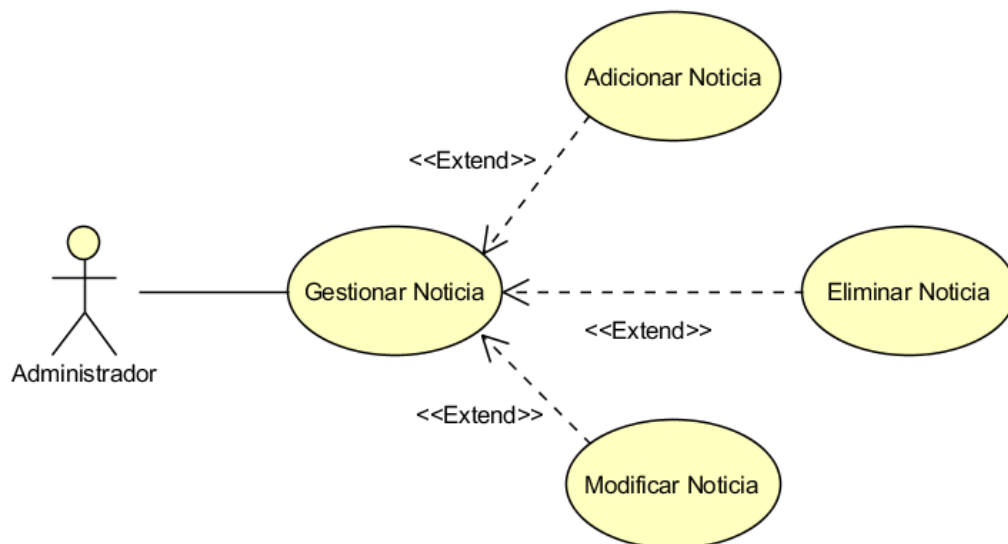


Figura 25 Modelo Caso de Uso Gestionar Noticia

2.1.2.2.3.5.4.2.7 Modelo Caso de Uso Gestionar Periodo

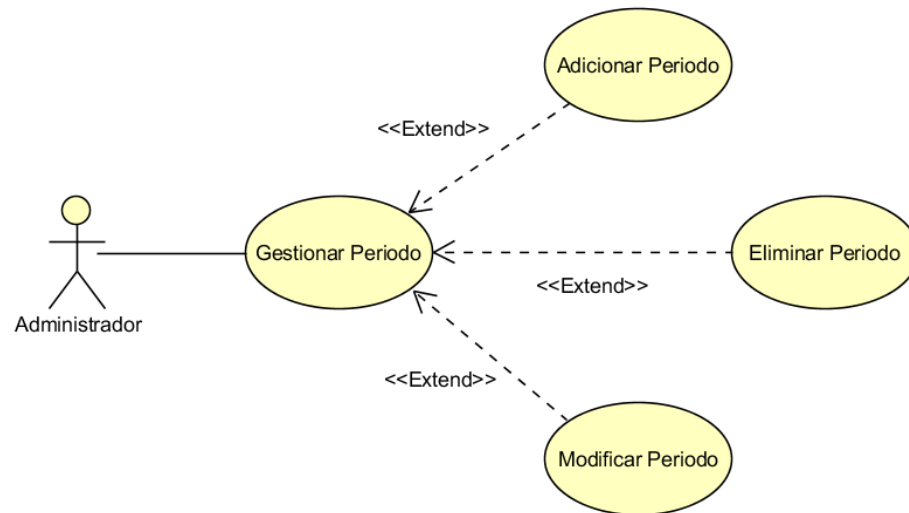


Figura 26 Modelo Caso de Uso Gestionar Periodo

2.1.2.2.3.5.4.2.8 Modelo Caso de Uso Gestionar Producto

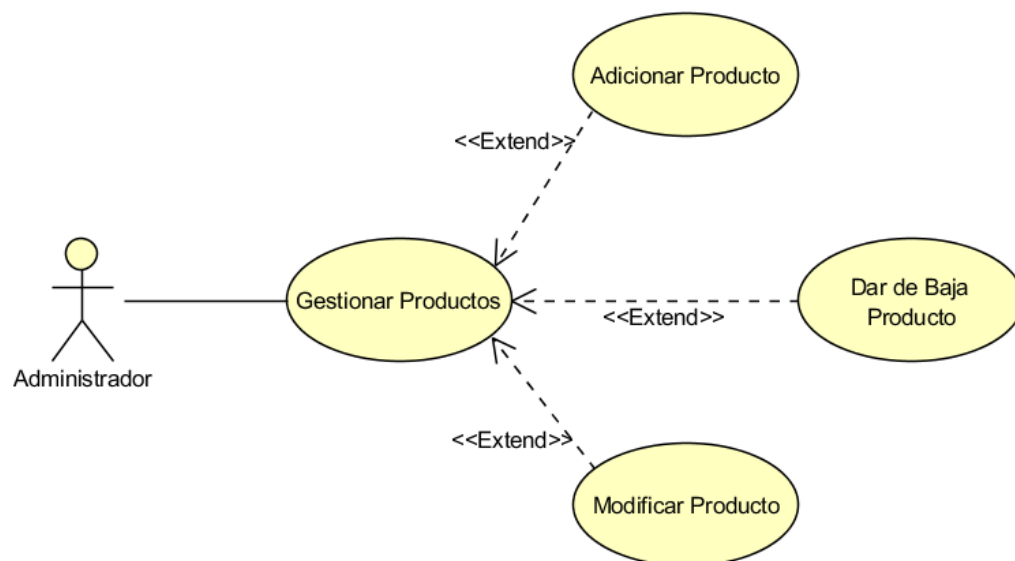


Figura 27 Modelo Caso de Uso Gestionar Producto

2.1.2.2.3.5.4.2.9 Modelo Caso de Uso Gestionar Rol

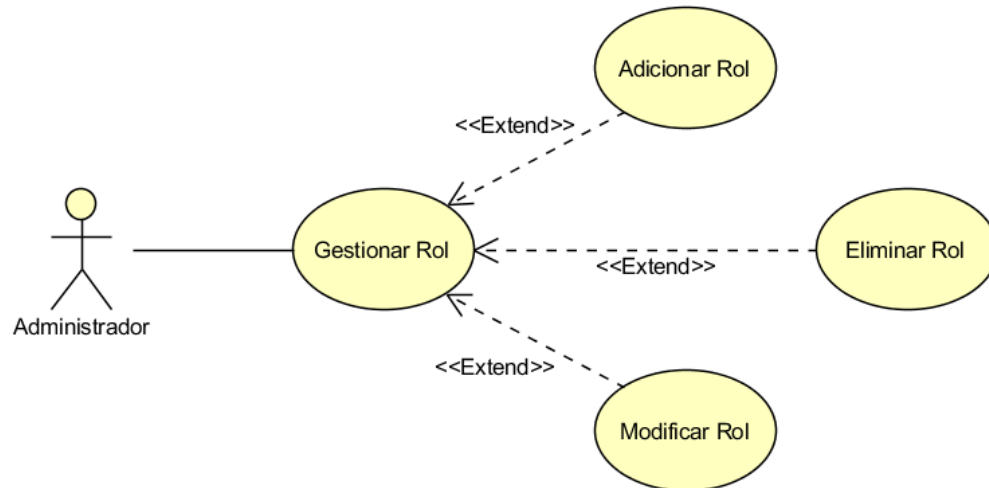


Figura 28 Modelo Caso de Uso Gestionar Rol

2.1.2.2.3.5.4.2.10 Modelo Caso de Uso Gestionar Sucursal

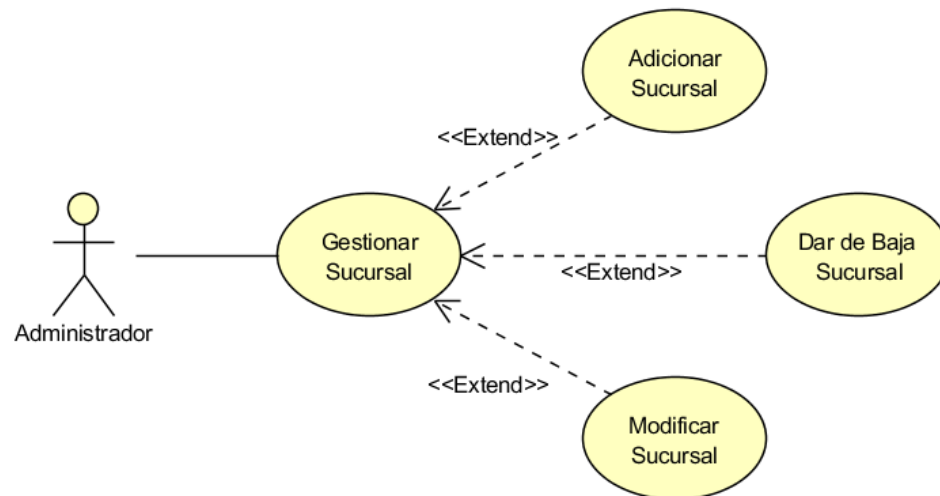


Figura 29 Modelo Caso de Uso Gestionar Sucursal

2.1.2.2.3.5.4.2.11 Modelo Caso de Uso Gestionar Usuario

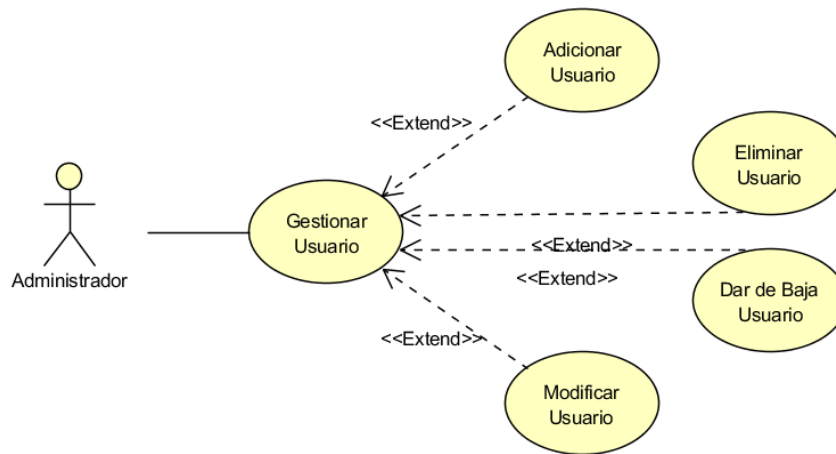


Figura 30 Modelo Caso de Uso Gestionar Usuario

2.1.2.2.3.5.4.2.12 Modelo Caso de Uso Gestionar Venta



Figura 31 Modelo Caso de Uso Gestionar Venta

2.1.2.2.3.6 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.

2.1.2.2.3.6.1 Introducción

2.1.2.2.3.6.1.1 Propósito

El propósito de éste documento es recoger, definir y analizar las necesidades más importantes y las características del desarrollo del Sistema Informático encargado del mejoramiento del control de ventas y difusión de la información de la empresa, 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.

2.1.2.2.3.6.1.2 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, noticias y promociones.
- Contar con la información centralizada en una sola organización.
- Brindar una tienda virtual a la clientela.

2.1.2.2.3.6.1.3 Definiciones, Acrónimos y Abreviaciones

- **RUP:** Son las siglas de Rational Unified Process. Se trata de una metodología para describir el proceso de desarrollo de software.
- **UML:** Son las siglas de Lenguaje Unificado de Modelación. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software.
- **COCOMO:** Acrónimo en inglés de Modelo Constructivo de Costes. Es un modelo que permite realizar estimaciones y planificaciones de proyectos de sistemas de información.
- **LAN:** Las siglas de Red de Área Local.

- **SQA:** Acrónimo en inglés de Aseguramiento de la Calidad del Software.

2.1.2.2.3.6.1.4 Posicionamiento

2.1.2.2.3.6.1.4.1 Oportunidad de Negocio

El negocio no satisface únicamente la necesidad de alimentación (comida y bebida) si no también la necesidad de socialización, al brindar al cliente la oportunidad de compartir momentos agradables en compañía de amigos, familiares, etc.

Por lo tanto las ventajas de implementar un sistema informático el cual también registre y lleve un control seguro de las ventas y/o pedidos que se hagan vía Web nos asegurarán la integridad, veracidad y la organización de nuestra información.

2.1.2.2.3.6.1.4.2 Sentencia que define el problema

El problema de	Inadecuado proceso de manejo o administración de la información y control de Ventas.
afecta a	Empresa de comida rápida VidiVidi.
El impacto asociado es	Genera retardos y errores en los balances de cierre de caja.
Una solución adecuada sería	Diseñar un Sistema informático destinado al Control de Ventas y difusión de la información.

Tabla 12 Sentencia que define el problema

2.1.2.2.3.6.1.4.3 Sentencia que define la posición del producto

para	VidiVidi
quienes	Gerente General, empleados y población en general de Tarija

El nombre del producto	“Vidi Virtual”
que	Gestiona oportuna y efectivamente la información de pedidos y ventas.
Nuestro producto	Mejorar y automatizar el control de Ventas y Pedidos mediante una interfaz sencilla y amigable. Permitir al cliente realizar compras vía Web, enterarse de promociones

Tabla 13 Sentencia que define la posición del producto

2.1.2.2.3.6.1.5 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 éstos perciben para enfocar la solución propuesta hacia ellos. No describe sus requisitos específicos ya que éstos se capturan mediante otro artefacto. En lugar de esto proporciona la justificación de por qué estos requisitos son necesarios.

2.1.2.2.3.6.1.5.1 Perfil de los Stakeholders

2.1.2.2.3.6.1.5.2 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 14 Perfil Stakeholder: Universidad Autónoma Juam Misael Saracho

2.1.2.2.3.6.1.5.3 Empresa de Comida Rápida Vidi Vidi

Representante	Lic. Gonzalo Castellanos
Descripción	Gerente General de VidiVidi
Tipo	Administrador
Responsabilidades	Administrador del Sistema VidiVirtual
Grado de participación	Velar por que se cumplan requisitos y funcionalidades, y que el proyecto sea ejecutado.

Tabla 15 Perfil Stakeholder: Empresa Vidi Vidi

2.1.2.2.3.6.1.5.4 Docente

Representante	Lic. Efraín Simeón Torrejón
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 16 Perfil Stakeholder: Docente

2.1.2.2.3.6.1.6 Perfil del Usuario

2.1.2.2.3.6.1.6.1 Administrador

Representante	Administrador
Descripción	Administrador de los recursos del sistema.
Tipo	Usuario.
Responsabilidades	Responsable del control de roles y acceso, usuarios del sistema; además es el responsable de administrar todos los componentes de la empresa
Grado de participación	Usuario principal del Sistema

Tabla 17 Perfil Stakeholder: Administrador

2.1.2.2.3.6.1.6.2 Cajero

Representante	Cajero
Descripción	Usuario con ciertos atributos- permisos de ejecución de la aplicación.
Tipo	Usuario causal.
Responsabilidades	Es el encargado de realizar ventas y administrar los pedidos.

Grado de participación	Usar el Sistema, buscar debilidades del mismo.
-------------------------------	--

Tabla 18 Perfil Stakeholder: Usuario

2.1.2.2.3.6.1.6.3 Cliente

Representante	Cliente
Descripción	Usuario que podrá realizar pedidos vía Web con la necesidad de identificarse
Tipo	Usuario final
Responsabilidades	Es el usuario que interactúa directamente con la tienda virtual
Grado de participación	Usar el Sistema, buscar debilidades del mismo.

Tabla 19 Perfil Stakeholder: Cliente

2.1.2.2.3.6.2 Descripción Global del Proyecto

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 “VidiVirtual”.

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.

2.1.2.2.3.7 Especificaciones de Casos de Usos

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, post-condiciones, 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.

2.1.2.2.3.8 Introducción

Las Especificaciones de los casos de Uso es una descripción detallada de los casos de uso del sistema.

2.1.2.2.3.9 Propósito

Interpretar y describir los casos de uso.

2.1.2.2.3.10 Alcance

Describe los procesos internos de los casos de uso.

Detalla los flujos de los casos de uso según lo establecido por la organización.

2.1.2.2.3.11 Especificación de los Casos de Uso

2.1.2.2.3.11.1 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 y Cajero.	
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	Flujo Alternativo: 1. Si el actor no llena el formulario de acceso, se muestra un mensaje donde se informa que los campos son requeridos. 2. Si el usuario no existe se muestra mensaje de error y vuelve a mostrar la pantalla principal.

la Pantalla Principal de Administración.	
Pos condiciones: Ninguno	

Tabla 20 Especificación Caso de Uso: Ingresar al Sistema

2.1.2.2.3.11.2 Especificación de Caso de Uso del Administrador

2.1.2.2.3.11.2.1 Especificación Caso de Uso Gestionar Categoría

Caso de uso:	Administrar Categoría		
Descripción: Permite Listar, Adicionar, Modificar y Eliminar a la Categoría.			
Actores: Administrador.			
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.			
Flujo Normal: 1. El sistema muestra Listado de las Categorías registrados en el sistema. 2. El actor realiza la Búsqueda de los Clientes en el Listado. 3. El sistema muestra las Categorías que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.		Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.	
Pos condiciones: Ninguno			

Tabla 21 Especificación Caso de Uso: Gestionar Categoría

2.1.2.2.3.11.2.2 Especificación de Caso de Uso Adicionar Categoría

Caso de uso:	Adicionar Categoría
---------------------	---------------------

Descripción: Permite registrar nueva Categoría en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos de la Categoría 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Categoría	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Categoría. 2. 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 22 Especificación Caso de Uso: Adicionar Categoría

2.1.2.2.3.11.2.3 Especificación de Caso de Uso Eliminar Categoría

Caso de uso:	Eliminar Categoría
Descripción: Permite eliminar una Categoría registrada en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal:	Flujo Alternativo:

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 Categoría. 4. El sistema regresa a la pantalla de Gestionar Categoría.	1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 23 Especificación Caso de Uso: Eliminar Categoría

2.1.2.2.3.11.2.4 Especificación de Caso de Uso Modificar Categoría

Caso de uso:	Modificar Categoría		
Descripción: Permite modificar datos de una Categoría en el sistema.			
Actores: Administrador.			
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema la Categoría			
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos de la Categoría. 3. El actor realiza los cambios de datos de la Categoría en el formulario. 4. Selecciona botón Guardar. 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 Categoría.		Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Categoría. 2. 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: Se Modificó Satisfactoriamente			

Tabla 24 Especificación Caso de Uso: Modificar Categoría

2.1.2.2.3.11.2.5 Especificación de Caso de Uso Gestionar Cliente

Caso de uso:	Gestionar Cliente		
Descripción: Permite Listar, Adicionar, Modificar y Eliminar al Cliente.			
Actores: Administrador y Cajero.			
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.			
Flujo Normal: 1. El sistema muestra Listado de los Clientes registrados en el sistema. 2. El actor realiza la Búsqueda de los Clientes en el Listado. 3. El sistema muestra los Clientes que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.		Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.	
Pos condiciones: Ninguno			

Tabla 25 Especificación Caso de Uso: Gestionar Cliente

2.1.2.2.3.11.2.6 Especificación de Caso de Uso Adicionar Cliente

Caso de uso:	Adicionar Cliente
Descripción: Permite registrar nuevo Cliente en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	

Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos del Cliente 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Cliente.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Cliente. 2. 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 26 Especificación Caso de Uso: Adicionar Cliente

2.1.2.2.3.11.2.7 Especificación de Caso de Uso Eliminar Cliente

Caso de uso:	Eliminar Cliente
Descripción: Permite eliminar un Cliente registrado en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
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 Cliente. 4. El sistema regresa a la pantalla de Gestionar Cliente.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.

Pos condiciones: Se eliminó Satisfactoriamente

Tabla 27 Especificación Caso de Uso: Eliminar Cliente

2.1.2.2.3.11.2.8 Especificación de Caso de Uso Modificar Cliente

Caso de uso:	Modificar Cliente		
Descripción: Permite modificar datos de un Cliente en el sistema.			
Actores: Administrador, Cajero y Cliente.			
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Cliente			
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Cliente. 3. El actor realiza los cambios de datos del Cliente en el formulario. 4. Selecciona botón Guardar. 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 Cliente.		Flujo Alternativo: 2. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Cliente. 3. 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: Se Modificó Satisfactoriamente			

Tabla 28 Especificación Caso de Uso: Modificar Cliente

2.1.2.2.3.11.2.9 Especificación de Caso de Uso Gestionar Egreso

Caso de uso:	Gestionar Egreso
Descripción: Permite Listar, Adicionar, Modificar y Eliminar al Egreso.	

Actores: Administrador y Cajero.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de los Egresos registrados en el sistema. 2. El actor realiza la Búsqueda de los Egresos en el Listado. 3. El sistema muestra los Egresos que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 29 Especificación Caso de Uso: Gestionar Egreso

2.1.2.2.3.11.2.10 Especificación de Caso de Uso Adicionar Egreso

Caso de uso:	Adicionar Egreso
Descripción: Permite registrar nuevo Egreso en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos del Egreso en el formulario.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Egreso. 2. Se muestra un mensaje donde se informa que los campos son requeridos y se

4. Selecciona botón Guardar. 5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El sistema guarda datos del formulario y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Egreso.	vuelve al paso 3 para que el actor pueda ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 30 Especificación Caso de Uso: Adicionar Egreso

2.1.2.2.3.11.2.11 Especificación de Caso de Uso Eliminar Egreso

Caso de uso:	Eliminar Egreso
Descripción: Permite eliminar un Egreso registrado en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
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 Egreso. 4. El sistema regresa a la pantalla de Gestionar Egreso.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 31 Especificación Caso de Uso: Eliminar Egreso

2.1.2.2.3.11.2.12 Especificación de Caso de Uso Modificar Egreso

Caso de uso:	Modificar Egreso
Descripción: Permite modificar datos de un Egreso en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Egreso.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Egreso. 3. El actor realiza los cambios de datos del Egreso en el formulario. 4. Selecciona botón Guardar. 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 Egreso.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Egreso. 2. 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: Se Modificó Satisfactoriamente	

Tabla 32 Especificación Caso de Uso: Modificar Egreso

2.1.2.2.3.11.2.13 Especificación de Caso de Uso Gestionar Ingreso

Caso de uso:	Gestionar Ingreso
Descripción: Permite Listar, Adicionar, Modificar y Eliminar al Ingreso.	
Actores:	

Administrador y Cajero.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de los Ingresos registrados en el sistema. 2. El actor realiza la Búsqueda de los Ingresos en el Listado. 3. El sistema muestra los Ingresos que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 33 Especificación Caso de Uso: Gestionar Ingreso

2.1.2.2.3.11.2.14 Especificación de Caso de Uso Adicionar Ingreso

Caso de uso:	Adicionar Ingreso
Descripción: Permite registrar nuevo Ingreso en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos del Ingreso en el formulario. 4. Selecciona botón Guardar.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Ingreso. 2. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda

5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El sistema guarda datos del formulario y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Ingreso.	ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 34 Especificación Caso de Uso: Adicionar Ingreso

2.1.2.2.3.11.2.15 Especificación de Caso de Uso Eliminar Ingreso

Caso de uso:	Eliminar Ingreso
Descripción: Permite eliminar un Ingreso registrado en el sistema.	
Actores: Administrador y Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
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 Ingreso. 4. El sistema regresa a la pantalla de Gestionar Ingreso.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 35 Especificación Caso de Uso: Eliminar Ingreso

2.1.2.2.3.11.2.16 Especificación de Caso de Uso Modificar Ingreso

Caso de uso:	Modificar Ingreso		
Descripción: Permite modificar datos de un Ingreso en el sistema.			
Actores: Administrador y Cajero.			
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Ingreso.			
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Ingreso. 3. El actor realiza los cambios de datos del Ingreso en el formulario. 4. Selecciona botón Guardar. 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 Ingreso.		Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Ingreso. 2. 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: Se Modificó Satisfactoriamente			

Tabla 36 Especificación Caso de Uso: Modificar Ingreso

2.1.2.2.3.11.2.17 Especificación de Caso de Uso Gestionar Noticia

Caso de uso:	Gestionar Noticia
Descripción:	
Permite Listar, Adicionar, Modificar y Eliminar a la Noticia.	
Actores:	

Administrador.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de las Noticias registrados en el sistema. 2. El actor realiza la Búsqueda de las Noticias en el Listado. 3. El sistema muestra las Noticias que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 37 Especificación Caso de Uso: Gestionar Noticia

2.1.2.2.3.11.2.18 Especificación de Caso de Uso Adicionar Noticia

Caso de uso:	Adicionar Noticia
Descripción: Permite registrar nueva Noticia en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos de la Noticia en el formulario. 4. Selecciona botón Guardar.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Noticia. 2. Se muestra un mensaje donde se informa que los campos son requeridos y se vuelve al paso 3 para que el actor pueda

5. Se valida datos del formulario que son obligatorios estén introducidos. 6. El sistema guarda datos del formulario y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Noticia.	ingresar la información que falta.
Pos condiciones: Ninguno	

Tabla 38 Especificación Caso de Uso: Adicionar Noticia

2.1.2.2.3.11.2.19 Especificación de Caso de Uso Eliminar Noticia

Caso de uso:	Eliminar Noticia
Descripción: Permite eliminar un Ingreso registrado en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
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 la Noticia. 4. El sistema regresa a la pantalla de Gestionar Noticias.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 39 Especificación Caso de Uso: Eliminar Noticia

2.1.2.2.3.11.2.20 Especificación de Caso de Uso Modificar Noticia

Caso de uso:	Modificar Noticia
Descripción: Permite modificar datos de una Noticia en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema la Noticia.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos de la Noticia. 3. El actor realiza los cambios de datos de la Noticia en el formulario. 4. Selecciona botón Guardar. 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 Noticia.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Noticia. 2. 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: Se Modificó Satisfactoriamente	

Tabla 40 Especificación Caso de Uso: Modificar Noticia

2.1.2.2.3.11.2.21 Especificación de Caso de Uso Dar Baja Noticia

Caso de uso:	Dar Baja Noticia
Descripción: Permite dar de Baja a una Noticia existente en el sistema.	

Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor selecciona un Noticia 2. El actor presiona botón Dar Baja. 3. El sistema le muestra un mensaje de confirmación. 4. El actor Selecciona la opción Sí. 5. El sistema guarda datos del formulario y muestra mensaje: “Se dio de Baja”	Flujo Alternativo: 1. Selecciona opción No, se regresa a la pantalla Gestionar Noticia.
Pos condiciones: Ninguno	

Tabla 41 Especificación Caso de Uso: Dar Baja Noticia

2.1.2.2.3.11.2.22 Especificación de Caso de Uso Gestionar Periodo

Caso de uso:	Gestionar Periodo
Descripción: Permite Listar, Adicionar, Modificar y Eliminar el Periodo.	
Actores: Administrador.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de los Periodos registrados en el sistema. 2. El actor realiza la Búsqueda de los Periodos en el Listado. 3. El sistema muestra los Periodos que cumple(n) con los Criterios de Búsqueda.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.

4. El actor selecciona una fila del Listado.	
5. El actor selecciona la opción que requiere.	
Pos condiciones: Ninguno	

Tabla 42 Especificación Caso de Uso: Gestionar Periodo

2.1.2.2.3.11.2.23 Especificación de Caso de Uso Adicionar Periodo

Caso de uso:	Adicionar Periodo		
Descripción:			
Permite registrar nuevo Periodo en el sistema.			
Actores:			
Administrador.			
Precondiciones:			
Actor debe haber iniciado sesión			
Flujo Normal:		Flujo Alternativo:	
1. El actor presiona botón Adicionar.		1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Periodo.	
2. El sistema le muestra formulario.		2. 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.	
3. El actor introduce datos del Periodo 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 y muestra mensaje: “Se añadió Satisfactoriamente”			
7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Periodo.			
Pos condiciones: Ninguno			

Tabla 43 Especificación Caso de Uso: Adicionar Periodo

**2.1.2.2.3.11.2.24 Especificación de Caso de Uso Eliminar
Periodo**

Caso de uso:	Eliminar Periodo
Descripción: Permite eliminar un Periodo registrado en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
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 Periodo. 4. El sistema regresa a la pantalla de Gestionar Periodo.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 44 Especificación Caso de Uso: Eliminar Periodo

**2.1.2.2.3.11.2.25 Especificación de Caso de Uso Modificar
Periodo**

Caso de uso:	Modificar Periodo
Descripción: Permite modificar datos de un Periodo en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema la Noticia.	

Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Periodo. 3. El actor realiza los cambios de datos del Periodo en el formulario. 4. Selecciona botón Guardar. 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 Periodo.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Periodo. 2. 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: Se Modificó Satisfactoriamente	

Tabla 45 Especificación Caso de Uso: Modificar Periodo

2.1.2.2.3.11.2.26 Especificación de Caso de Uso Gestionar Producto

Caso de uso:	Gestionar Producto
Descripción: Permite Listar, Adicionar, Modificar y Eliminar el Producto.	
Actores: Administrador.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de los Productos registrados en el sistema. 2. El actor realiza la Búsqueda de los Periodos en el Listado. 3. El sistema muestra los Productos que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.

Listado. 5. El actor selecciona la opción que requiere.	
Pos condiciones: Ninguno	

Tabla 46 Especificación Caso de Uso: Gestionar Producto

2.1.2.2.3.11.2.27 Especificación de Caso de Uso Adicionar Producto

Caso de uso:	Adicionar Producto		
Descripción: Permite registrar nuevo Producto en el sistema.			
Actores: Administrador.			
Precondiciones: Actor debe haber iniciado sesión			
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos del Producto 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Producto.		Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Producto. 2. 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 Producto

**2.1.2.2.3.11.2.28 Especificación de Caso de Uso Dar de Baja
Producto**

Caso de uso:	Dar Baja Producto
Descripción: Permite dar de Baja a un Producto existente en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 6. El actor selecciona una Producto 7. El actor presiona botón Dar Baja. 8. El sistema le muestra un mensaje de confirmación. 9. El actor Selecciona la opción Sí. 10. El sistema guarda datos del formulario y muestra mensaje: “Se dio de Baja”	Flujo Alternativo: 2. Selecciona opción No, se regresa a la pantalla Gestionar Producto.
Pos condiciones: Ninguno	

Tabla 48 Especificación de Caso de Uso: Dar Baja Producto

**2.1.2.2.3.11.2.29 Especificación de Caso de Uso Modificar
Producto**

Caso de uso:	Modificar Producto
Descripción: Permite modificar datos de un Producto en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Producto.	

Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Producto. 3. El actor realiza los cambios de datos del Producto en el formulario. 4. Selecciona botón Guardar. 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 Producto.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Producto. 2. 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: Se Modificó Satisfactoriamente	

Tabla 49 Especificación Caso de Uso: Modificar Producto

2.1.2.2.3.11.2.30 Especificación de Caso de Uso Gestionar Rol

Caso de uso:	Gestionar Rol
Descripción: Permite Listar, Adicionar, Modificar y Eliminar el Rol.	
Actores: Administrador.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de los Roles registrados en el sistema. 2. El actor selecciona una fila del Listado. 3. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 50 Especificación Caso de Uso: Gestionar Rol

2.1.2.2.3.11.2.31

Especificación de Caso de Uso Adicionar Rol

Caso de uso:	Adicionar Rol		
Descripción: Permite registrar nuevo Rol en el sistema.			
Actores: Administrador.			
Precondiciones: Actor debe 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Rol.		Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Rol. 2. 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 51 Especificación Caso de Uso: Adicionar Rol

2.1.2.2.3.11.2.32

Especificación de Caso de Uso Eliminar Rol

Caso de uso:	Eliminar Rol
Descripción:	
Permite eliminar un Rol registrado en el sistema.	
Actores:	
Administrador.	
Precondiciones:	

Actor debe haber iniciado sesión	
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 Gestionar Rol.	Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.
Pos condiciones: Se eliminó Satisfactoriamente	

Tabla 52 Especificación Caso de Uso: Eliminar Rol

2.1.2.2.3.11.2.33 Especificación de Caso de Uso Modificar Rol

Caso de uso:	Modificar Rol
Descripción: Permite modificar datos de un Rol en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Rol.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Rol. 3. El actor realiza los cambios de datos del Rol en el formulario. 4. Selecciona botón Guardar. 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 Rol.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Rol. 2. 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: Se Modificó Satisfactoriamente

Tabla 53 Especificación Caso de Uso: Modificar Rol

2.1.2.2.3.11.2.34 Especificación de Caso de Uso Gestionar Sucursal

Caso de uso:	Gestionar Sucursal
Descripción: Permite Listar, Adicionar, Modificar y Dar de Baja la Sucursal.	
Actores: Administrador.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de las Sucursales registrados en el sistema. 2. El actor selecciona una fila del Listado. 3. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 54 Especificación Caso de Uso: Gestionar Sucursal

2.1.2.2.3.11.2.35 Especificación de Caso de Uso Adicionar Sucursal

Caso de uso:	Adicionar Sucursal
Descripción: Permite registrar una nueva Sucursal en el sistema.	
Actores: Administrador.	

Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 2. El sistema le muestra formulario. 3. El actor introduce datos de la Sucursal 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Sucursal.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Sucursal. 2. 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 55 Especificación Caso de Uso: Adicionar Sucursal

2.1.2.2.3.11.2.36 Especificación de Caso de Uso Dar de Baja Sucursal

Caso de uso:	Dar Baja Sucursal
Descripción: Permite dar de Baja a un Sucursal existente en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 11. El actor selecciona una Sucursal 12. El actor presiona botón Dar Baja. 13. El sistema le muestra un mensaje de confirmación. 14. El actor Selecciona la opción Sí.	Flujo Alternativo: 3. Selecciona opción No, se regresa a la pantalla Gestionar Sucursal.

15. El sistema guarda datos del formulario y muestra mensaje: “Se dio de Baja”	
Pos condiciones: Ninguno	

Tabla 56 Especificación Caso de Uso: Dar Baja Sucursal

2.1.2.2.3.11.2.37 Especificación de Caso de Uso Modificar Sucursal

Caso de uso:	Modificar Sucursal
Descripción: Permite modificar datos de una Sucursal en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema la Sucursal.	
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos de la Sucursal. 3. El actor realiza los cambios de datos de la Sucursal en el formulario. 4. Selecciona botón Guardar. 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 Sucursal.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Sucursal. 2. 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: Se Modificó Satisfactoriamente	

Tabla 57 Especificación Caso de Uso: Modificar Sucursal

2.1.2.2.3.11.2.38 Especificación de Caso de Uso Gestionar Usuario

Caso de uso:	Gestionar Usuario		
Descripción: Permite Listar, Adicionar, Modificar y Dar de Baja al Usuario.			
Actores: Administrador.			
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.			
Flujo Normal: 1. El sistema muestra Listado de los Usuarios registradas en el sistema. 2. El actor realiza la Búsqueda de los Usuarios en el Listado. 3. El sistema muestra los Usuarios que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.		Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.	
Pos condiciones: Ninguno			

Tabla 58 Especificación Caso de Uso: Gestionar Usuario

2.1.2.2.3.11.2.39 Especificación de Caso de Uso Adicionar Usuario

Caso de uso:	Adicionar Usuario
Descripción: Permite registrar un nuevo Usuario en el sistema.	
Actores: Administrador.	

Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona botón Adicionar. 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 y muestra mensaje: “Se añadió Satisfactoriamente” 7. El actor presiona botón Cerrar y vuelve a la pantalla Gestionar Usuario.	Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Usuario. 2. 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 Usuario

2.1.2.2.3.11.2.40 Especificación de Caso de Uso Dar de Baja Usuario

Caso de uso:	Dar Baja Usuario
Descripción: Permite dar de Baja a un Usuario existente en el sistema.	
Actores: Administrador.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 16. El actor selecciona Usuario 17. El actor presiona botón Dar Baja. 18. El sistema le muestra un mensaje de confirmación. 19. El actor Selecciona la opción Sí.	Flujo Alternativo: 4. Selecciona opción No, se regresa a la pantalla Gestionar Usuario.

20. El sistema guarda datos del formulario y muestra mensaje: “Se dio de Baja”	
Pos condiciones: Ninguno	

Tabla 60 Especificación de Caso de Uso Dar de Baja Usuario

2.1.2.2.3.11.2.41 Especificación de Caso de Uso Modificar Usuario

Caso de uso:	Modificar Usuario		
Descripción: Permite modificar datos de un Usuario en el sistema.			
Actores: Administrador.			
Precondiciones: Actor debe haber iniciado sesión, Estar registrado en el sistema el Rol.			
Flujo Normal: 1. El actor presiona en el menú la opción Modificar. 2. El sistema le muestra formulario con datos del Usuario. 3. El actor realiza los cambios de datos del Usuario en el formulario. 4. Selecciona botón Guardar. 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 Usuario.		Flujo Alternativo: 1. Selecciona botón Cerrar, se regresa a la pantalla Gestionar Usuario. 2. 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: Se Modificó Satisfactoriamente			

Tabla 61 Especificación Caso de Uso: Modificar Usuario

**2.1.2.2.3.11.2.42 Especificación de Caso de Uso Gestionar
Venta Web**

Caso de uso:	Gestionar Venta Web		
Descripción:			
Permite Listar Ventas.			
Actores:			
Administrador y Cajero.			
Precondiciones:			
Actor no tiene privilegios para realizar ninguna acción.			
Flujo Normal:		Flujo Alternativo:	
1. El sistema muestra Listado de las Ventas Web registradas en el sistema.		1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.	
2. El actor realiza la Búsqueda de las Ventas Web en el Listado.			
3. El sistema muestra las Ventas que cumple(n) con los Criterios de Búsqueda.			
4. El actor selecciona una fila del Listado.			
5. El actor selecciona la opción que requiere.			
Pos condiciones: Ninguno			

Tabla 62 Especificación Caso de Uso: Gestionar Venta Web

**2.1.2.2.3.11.2.43 Especificación de Caso de Uso Adicionar
Venta Web**

Caso de uso:	Adicionar Venta Web
Descripción: Permite registrar una nueva Venta en el sistema.	
Actores:	

Cliente.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor entra a la tienda Virtual. 2. El sistema le muestra formulario. 3. El actor selecciona los productos deseados y presiona botón Continuar. 4. El actor debe identificarse como Cliente. 5. Se abre un nuevo formulario de nueva Venta. 6. Se presiona el botón Realizar Pedido y muestra mensaje: "Pedido: # Lo llamaremos para confirmar su pedido, Espere por favor"	Flujo Alternativo: 1. Introduce su CI en el formulario y presiona el botón Buscar 2. Si no encuentra al Cliente selecciona la opción Registrar Cliente Nuevo y se abrirá un formulario. 3. Se valida datos del formulario que son obligatorios estén introducidos 4. Presiona el botón Registrar
Pos condiciones: Esperar llamada de VidiVidi para confirmar pedido	

Tabla 63 Especificación Caso de Uso: Adicionar Venta Web

2.1.2.2.3.11.2.44 Especificación de Caso de Uso Confirmar Venta Web

Caso de uso:	Confirmar Venta Web
Descripción: Permite garantizar la veracidad de una Venta Web.	
Actores: Administrador, Cajero.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor presiona en el menú la opción Confirmar Venta.	Flujo Alternativo: 1. Si no confirma la venta, vuelve a la pantalla Gestionar Ventas Web.

2. El sistema le muestra una pantalla de validación. 3. Si confirma la Venta el sistema muestra un mensaje de “Venta Confirmada”. 4. El sistema regresa a la pantalla de Gestionar Ventas Web.	
Pos condiciones: Ninguno	

Tabla 64 Especificación Caso de Uso: Confirmar Venta Web

2.1.2.2.3.11.2.45 Especificación de Caso de Uso Confirmar Entrega de Venta Web

Caso de uso:	Confirmar Entrega de Venta Web		
Descripción:			
Permite garantizar la entrega de una Venta Web.			
Actores:			
Administrador, Cajero.			
Precondiciones:			
Actor debe haber iniciado sesión			
Flujo Normal:		Flujo Alternativo:	
1. El actor presiona en el menú la opción Confirmar Entrega Venta.		1. Si no confirma la venta, vuelve a la pantalla Gestionar Ventas Web.	
2. El sistema le muestra una pantalla de validación.			
3. Si confirma la Entrega de Venta el sistema muestra un mensaje de “Entrega Confirmada”.			
4. El sistema regresa a la pantalla de Gestionar Ventas Web.			
Pos condiciones: Ninguno			

Tabla 65 Especificación Caso de Uso: Confirmar Entrega de Venta Web

**2.1.2.2.3.11.2.46 Especificación de Caso de Uso Eliminar
Venta Web**

Caso de uso:	Eliminar Venta Web		
Descripción: Permite eliminar un Rol registrado en el sistema.			
Actores: Administrador y Cajero.			
Precondiciones: Actor debe haber iniciado sesión			
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 la Venta. 4. El sistema regresa a la pantalla de Gestionar Venta.		Flujo Alternativo: 1. Por no el sistema no realizará ninguna acción.	
Pos condiciones: Se eliminó Satisfactoriamente			

Tabla 66 Especificación Caso de Uso: Confirmar Eliminar de Venta Web

**2.1.2.2.3.11.2.47 Especificación de Caso de Uso Confirmar
Entrega de Venta**

Caso de uso:	Confirmar Entrega de Venta
Descripción:	
Permite garantizar que un Pedido fue entregado	
Actores:	
Administrador, Cajero.	
Precondiciones:	
Actor debe haber iniciado sesión	

Flujo Normal: 1. El actor presiona en el menú la opción Confirmar Entrega (Venta). 2. El sistema le muestra una pantalla de validación. 3. Si confirma que la Venta fue entregada el sistema muestra un mensaje de “Venta Confirmada”. 4. El sistema regresa a la pantalla de Gestionar Ventas Web.	Flujo Alternativo: 1. Si no confirma la venta, vuelve a la pantalla Gestionar Ventas Web.
Pos condiciones: Ninguno	

Tabla 67 Especificación Caso de Uso: Confirmar Entrega de Venta Web

2.1.2.2.3.11.2.48 Especificación de Caso de Uso Gestionar Venta Local

Caso de uso:	Gestionar Venta Local
Descripción: Permite Listar Ventas de tipo Local.	
Actores: Administrador y Cajero.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de las Ventas Local registradas en el sistema. 2. El actor realiza la Búsqueda de las Ventas Locales en el Listado. 3. El sistema muestra las Ventas que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.

Pos condiciones: Ninguno

Tabla 68 Especificación Caso de Uso: Gestionar Venta Local

**2.1.2.2.3.11.2.49 Especificación de Caso de Uso Adicionar
Venta Local**

Caso de uso:	Adicionar Venta Local
Descripción: Permite registrar una nueva Venta en el sistema.	
Actores: Cliente.	
Precondiciones: Actor debe haber iniciado sesión	
Flujo Normal: 1. El actor entra a la tienda Virtual. 2. El sistema le muestra formulario. 3. El actor selecciona los productos deseados y presiona botón Continuar. 4. El actor debe identificarse como Cliente. 5. Se abre un nuevo formulario de nueva Venta. 6. Se presiona el botón Realizar Pedido y muestra mensaje: "Pedido: # Lo llamaremos para confirmar su pedido, Espere por favor"	Flujo Alternativo: 1. Introduce su CI en el formulario y presiona el botón Buscar 2. Si no encuentra al Cliente selecciona la opción Registrar Cliente Nuevo y se abrirá un formulario. 3. Se valida datos del formulario que son obligatorios estén introducidos 4. Presiona el botón Registrar
Pos condiciones: Esperar llamada de VidiVidi para confirmar pedido	

Tabla 69 Especificación Caso de Uso: Adicionar de Venta Local

**2.1.2.2.3.11.2.50 Especificación de Caso de Uso Gestionar
Venta Telefónica**

Caso de uso:	Gestionar Venta Telefónica
---------------------	----------------------------

Descripción: Permite Listar Ventas de tipo Telefónica.	
Actores: Administrador y Cajero.	
Precondiciones: Actor no tiene privilegios para realizar ninguna acción.	
Flujo Normal: 1. El sistema muestra Listado de las Ventas Local registradas en el sistema. 2. El actor realiza la Búsqueda de las Ventas Locales en el Listado. 3. El sistema muestra las Ventas que cumple(n) con los Criterios de Búsqueda. 4. El actor selecciona una fila del Listado. 5. El actor selecciona la opción que requiere.	Flujo Alternativo: 1. El actor decide no confirmar selección y el sistema no realiza ninguna operación.
Pos condiciones: Ninguno	

Tabla 70 Especificación de Caso de Uso Gestionar Venta Telefónica

2.1.2.2.3.12 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.

2.1.2.2.3.13 Prototipos de Interfaces de Usuario

2.1.2.2.3.13.1 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

2.1.2.2.3.13.2 Propósito

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

2.1.2.2.3.13.3 Alcance

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

2.1.2.2.3.13.4 Prototipo de Pantallas

2.1.2.2.3.13.4.1 Sistema Web Vidi Virtual

2.1.2.2.3.13.4.2 Sistema de Escritorio Vidi Virtual POS

2.1.2.2.3.13.4.2.1 Pantalla Ingresar al Sistema



Figura 32 Pantalla Ingresar al Sistema

2.1.2.2.3.13.4.2.2 Pantalla Principal



Figura 33 Pantalla Principal

2.1.2.2.3.13.4.2.3 Pantalla Gestionar Cliente

ID	CI	Nombre	Ap Paterno	Ap Materno	Opciones
0	123	Juan Jose	Lopez	Araoz	Detalle Modificar Eliminar
1	51478294	Paola Andrea	Garzon	Castellanos	Detalle Modificar Eliminar
2	5815471	Alex	Cespedes	Gira	Detalle Modificar Eliminar
3	4321	Pablo	Castellanos	Lopez	Detalle Modificar Eliminar
4	3423	Fernando	Martinez	Martinez	Detalle Modificar Eliminar
5	1112342	Luis	lopez	claire	Detalle Modificar Eliminar
6	23434433	asd	asda	dfs	Detalle Modificar Eliminar
7	123456	as	as	asd	Detalle Modificar Eliminar
8	1234567	asd	asd	das	Detalle Modificar Eliminar

Figura 34 Pantalla Gestionar Cliente

2.1.2.2.3.13.4.2.4 Pantalla Nuevo Cliente

Adicionar Cliente

Id: 34

Ci:

Nombre:

Ap Paterno:

Ap Materno:

Telefono:

Correo:

Direccion:

Guardar Cerrar

Figura 35 Pantalla Adicionar Cliente

2.1.2.2.3.13.4.2.5 Pantalla Modificar Cliente



Modificar Cliente

Ci: 4321

Nombre: Pablo

Ap Paterno: Castellanos

Ap Materno: Lopez

Telefono:

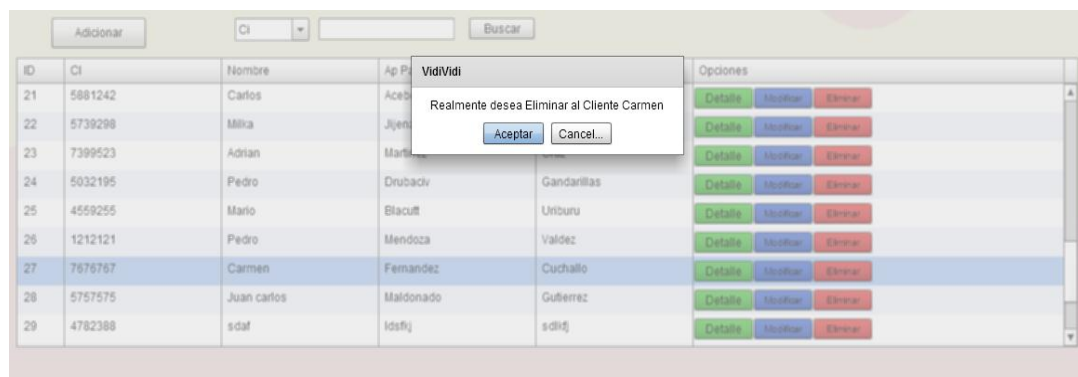
Correo: pablor@gmail.com

Direccion:

Guardar Cerrar

Figura 36 Pantalla Modificar Cliente

2.1.2.2.3.13.4.2.6 Pantalla Eliminar Cliente



ID	CI	Nombre	Ap P	Ap M	Opciones
21	5881242	Carlos	Acebo		Detalle Actualizar Eliminar
22	5738298	Mika	Jyer		Detalle Actualizar Eliminar
23	7399523	Adrian	Marin		Detalle Actualizar Eliminar
24	5032195	Pedro	Drubach	Gandarillas	Detalle Actualizar Eliminar
25	4559255	Mario	Blacuff	Uniburu	Detalle Actualizar Eliminar
26	1212121	Pedro	Mendoza	Valdez	Detalle Actualizar Eliminar
27	7676767	Carmen	Fernandez	Cuchallo	Detalle Actualizar Eliminar
28	5757575	Juan carlos	Maldonado	Guierrez	Detalle Actualizar Eliminar
29	4782388	scaf	Idafij	sdlifj	Detalle Actualizar Eliminar

Figura 37 Pantalla Eliminar Cliente

2.1.2.2.3.13.4.2.7 Pantalla Gestionar Egreso

ID	Fecha	Cuenta	Detalle	Monto	Opciones
32	Sun Jul 15 02:43:23 GMT-0400	Caja chica	1 Litro de Aceite	22	Detalle Modificar Eliminar
33	Sun Jul 15 02:43:39 GMT-0400	Caja chica	Taxi del Personal	13	Detalle Modificar Eliminar
34	Sun Jul 15 02:44:02 GMT-0400	Caja chica	Insumos de Limpieza	18	Detalle Modificar Eliminar

Figura 38 Pantalla Gestionar Egreso

2.1.2.2.3.13.4.2.8 Pantalla Adicionar Egreso

Id: 35

Cuenta:

Monto:

Detalle:

Guardar Cerrar

Figura 39 Pantalla Adicionar Egreso

2.1.2.2.3.13.4.2.9 Pantalla Modificar Egreso

Cuenta: Caja chica

Monto: 13

Detalle: Taxi del Personal

Guardar Cerrar

2.1.2.2.3.13.4.2.10 Pantalla Eliminar Egreso

Adm. Egreso Volver

Adicionar 15/07/2012

ID	Fecha	Cuenta	Detalle	Opciones
32	Sun Jul 15 02:43:23 GMT-0400	Caja chica	1 L	Detalle Modificar Eliminar
33	Sun Jul 15 02:43:39 GMT-0400	Caja chica	Taxi	Detalle Modificar Eliminar
34	Sun Jul 15 02:44:02 GMT-0400	Caja chica	...	Detalle Modificar Eliminar

VidiVidi
Realmente desea Eliminar al Egreso Caja chica
Aceptar Cancelar

Figura 40 Pantalla Eliminar Egreso

2.1.2.2.3.13.4.2.11 Pantalla Gestionar Ingreso

Inicio Ventas Clientes Egreso Ingreso Reporte Diario

Adm. Ingreso Volver

Adicionar 15/07/2012

ID	Fecha	Cuenta	Monto	Opciones
35	Sun Jul 15 02:49:19 GMT-0400 2012	Caja chica	100	Detalle Modificar Eliminar

vidividi
comida rápida

Figura 41 Pantalla Gestionar Ingreso

2.1.2.2.3.13.4.2.12 Pantalla Adicionar Ingreso

Adicionar Ingreso

Id: 35

Cuenta:

Monto:

Guardar Cerrar

Figura 42 Pantalla Adicionar Ingreso

2.1.2.2.3.13.4.2.13 Pantalla Modificar Ingreso

Modificar Ingreso

Cuenta: Caja chica

Monto: 100

Guardar Cerrar

Figura 43 Pantalla Modificar Ingreso

2.1.2.2.3.13.4.2.14 Pantalla Reporte Venta Diaria

15/07/2012

Exportar a Pdf

Exportar

Fecha: 15/7/2012

Fecha	ID	Monto	Descuento	Cliente
Sun Jul 15 03:16:13 GMT-0400 20	89	21	0	1822213
Sun Jul 15 03:17:37 GMT-0400 20	91	14	0	5815471

Nro de Ventas: 2

Total: 35 Bs.

Figura 44 Pantalla Reporte Venta Diaria

2.1.2.2.3.13.4.2.15 Pantalla Reporte Ingreso / Egreso

ID	Cuenta	Detalle	Monto
32	Caja chica	1 Litro de Aceite	22
33	Caja chica	Taxi del Personal	13
34	Caja chica	Insumos de Limpieza	18

Figura 45 Pantalla Reporte Ingreso / Egreso

2.1.2.2.3.13.4.2.16 Pantalla Gestionar Venta Web

Id	Fecha	Usuario	Cliente	Monto	Observacion	Opciones
51	Mon Jun 25 10:51:58 GI	1	7113016	21		Detalle Eliminar Confirmar Entregar
62	Mon Jun 25 17:08:08 GI	1	123423	30		Detalle Eliminar Confirmar Entregar

Figura 46 Pantalla Gestionar Venta Web

2.1.2.2.3.13.4.2.17 Pantalla Gestionar Venta Telefónica

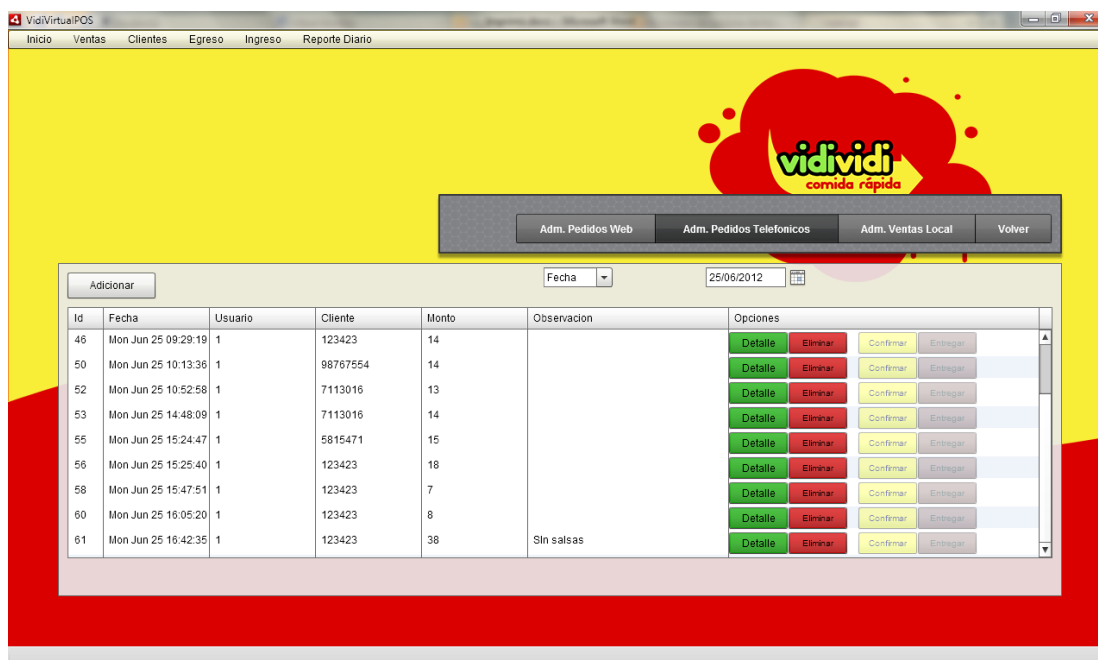


Figura 47 Pantalla Gestionar Venta Telefónica

2.1.2.2.3.13.4.2.18 Pantalla Gestionar Venta Local

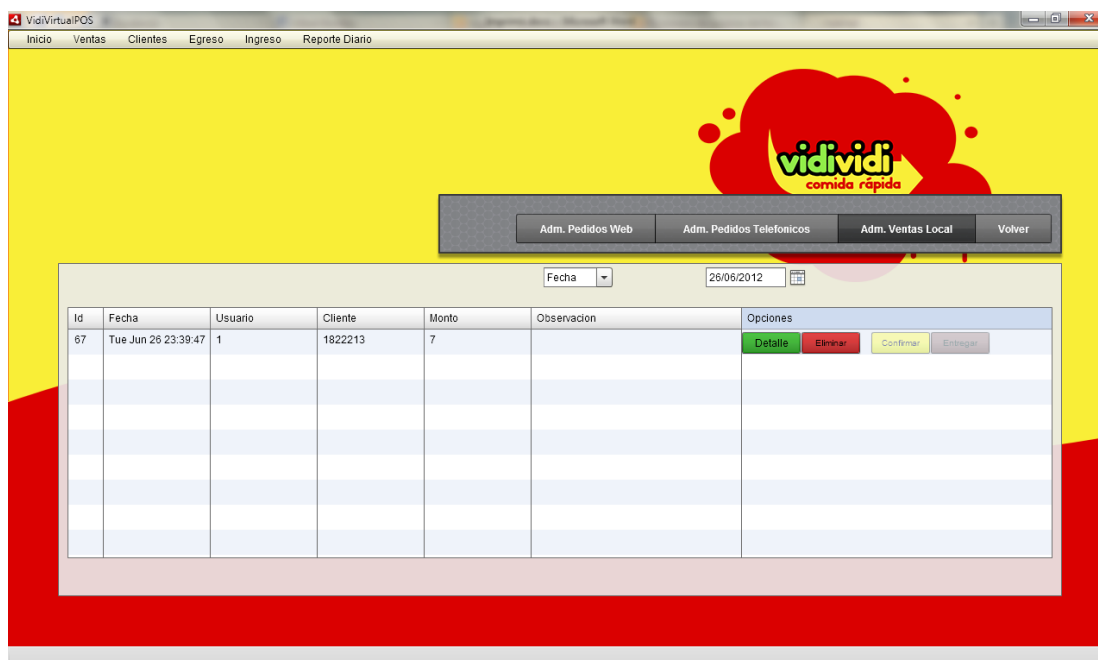


Figura 48 Pantalla Gestionar Venta Local

2.1.2.2.3.13.4.2.19 Pantalla Seleccionar Producto

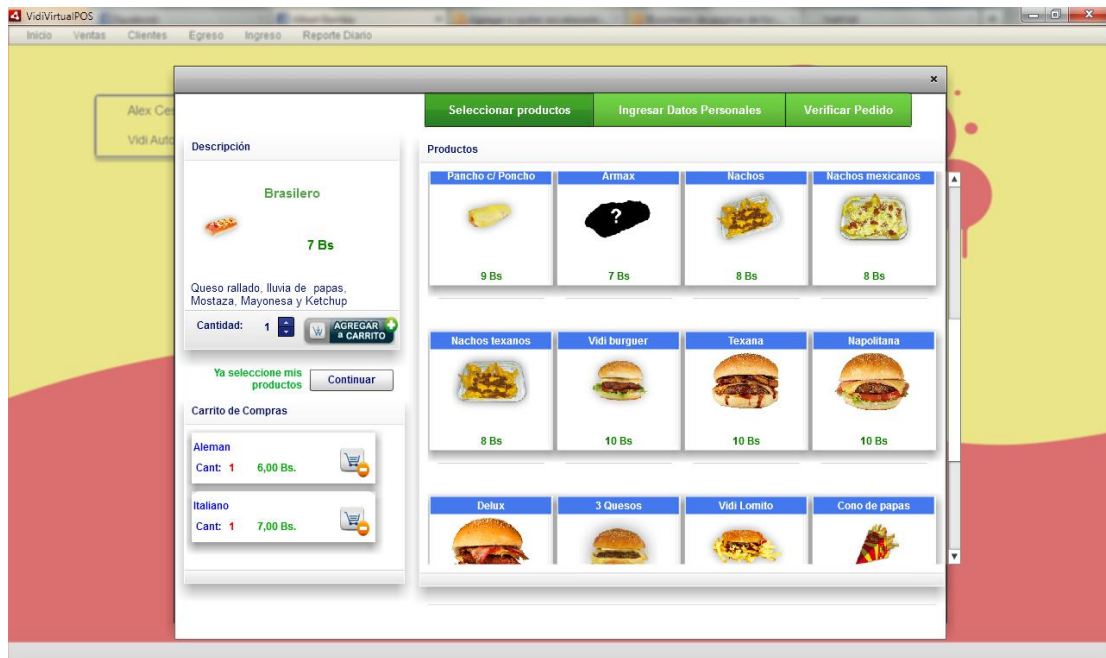


Figura 49 Pantalla Seleccionar Producto

2.1.2.2.3.13.4.2.20 Pantalla Adicionar Cliente

Figura 50 Pantalla Adicionar Cliente

2.1.2.2.3.13.4.2.21 Pantalla Confirmar Venta

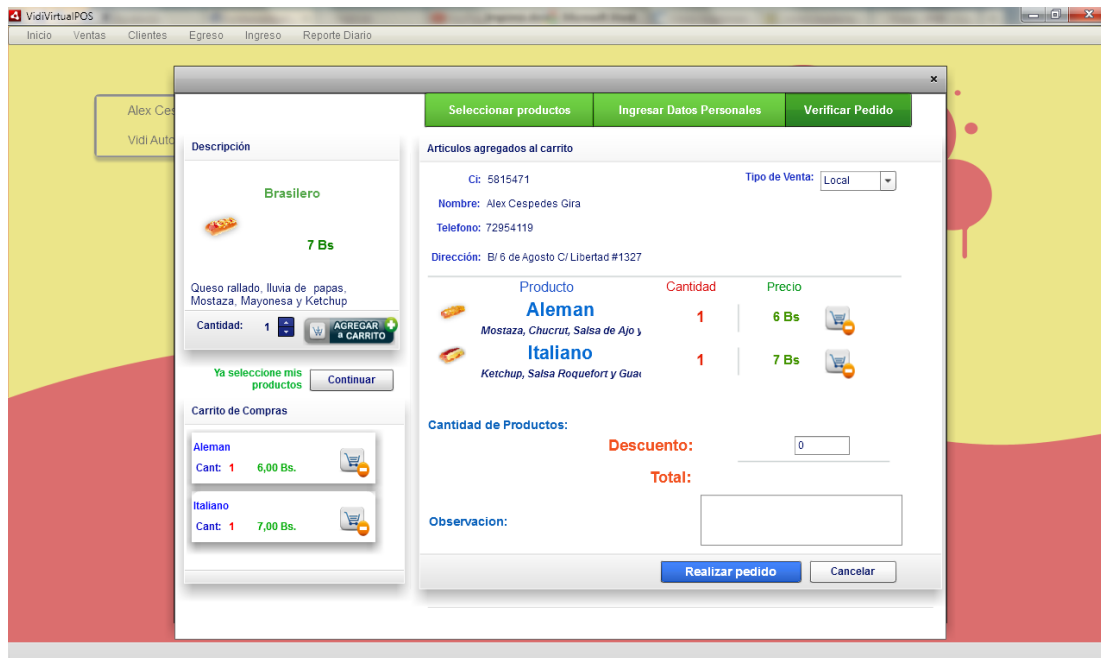


Figura 51 Pantalla Confirmar Venta

2.1.2.2.3.14 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.

2.1.2.2.3.14.1 Modelado de Diagrama de Actividades

2.1.2.2.3.14.1.1 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.

2.1.2.2.3.14.1.2 Propósito

Comprender la estructura y dinámica del sistema deseado.

Identificar posibles mejoras del sistema.

2.1.2.2.3.14.1.3 Alcance

Describir los procesos del sistema y 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.

2.1.2.2.3.14.1.4 Diagrama de Actividades

2.1.2.2.3.14.1.4.1 Diagrama de Actividad: Caso de Uso Ingresar al Sistema

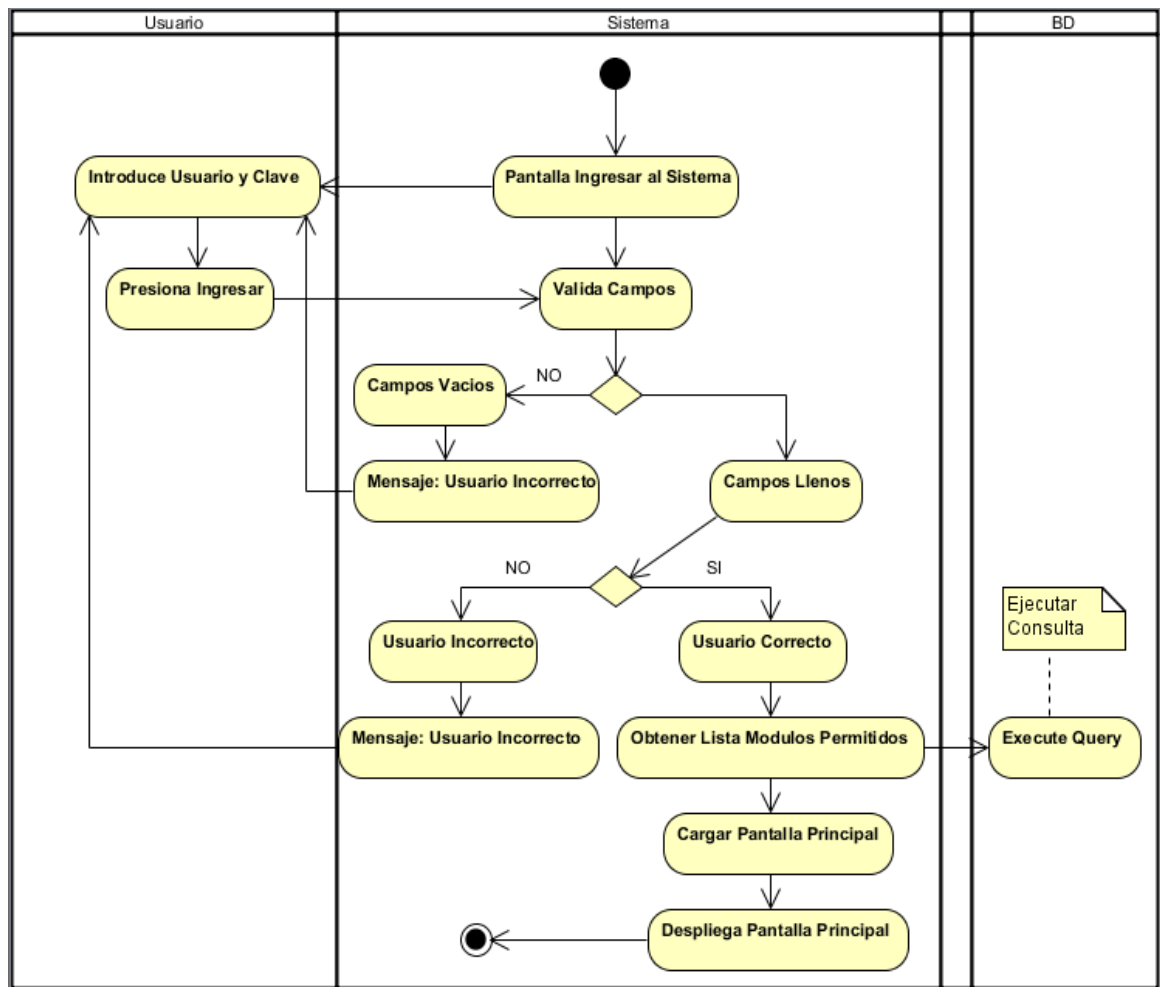


Figura 52 Diagrama de Actividad: CU Adicionar Ingresar al Sistema

2.1.2.2.3.14.1.4.2 Diagrama de Actividad: Caso de Uso Gestionar Categoría

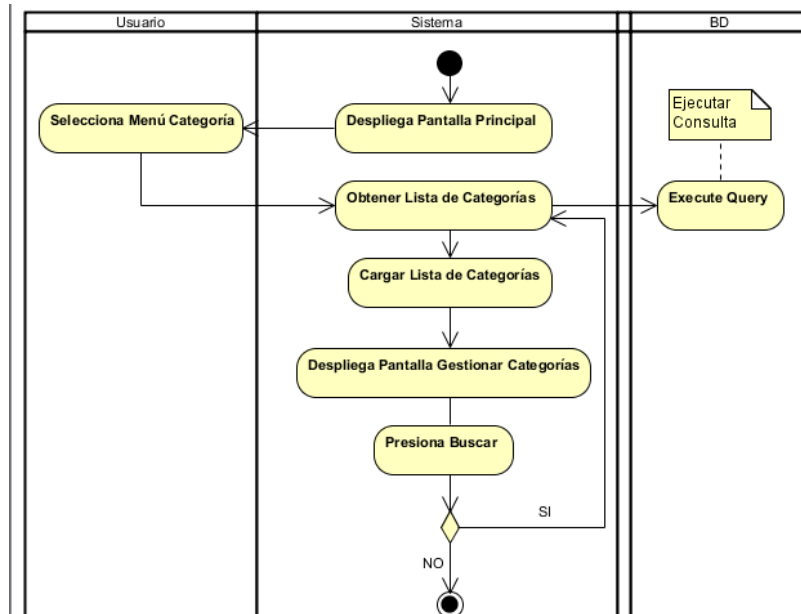


Figura 53 Diagrama de Actividad: CU Gestionar Categoría

2.1.2.2.3.14.1.4.3 Diagrama de Actividad: Caso de Uso Adicionar Categoría

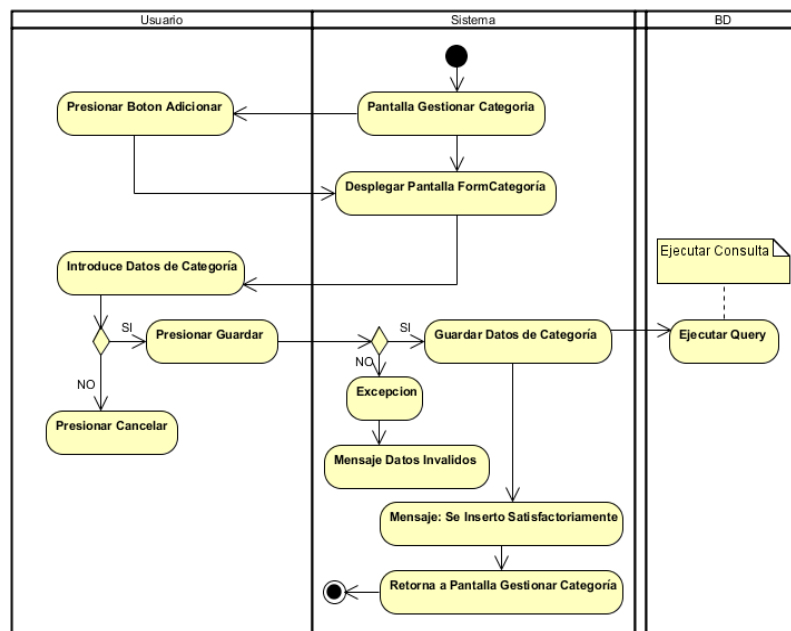


Figura 54 Diagrama de Actividad: CU Adicionar Categoría

2.1.2.2.3.14.1.4.4 Diagrama de Actividad: Caso de Uso Eliminar Categoría

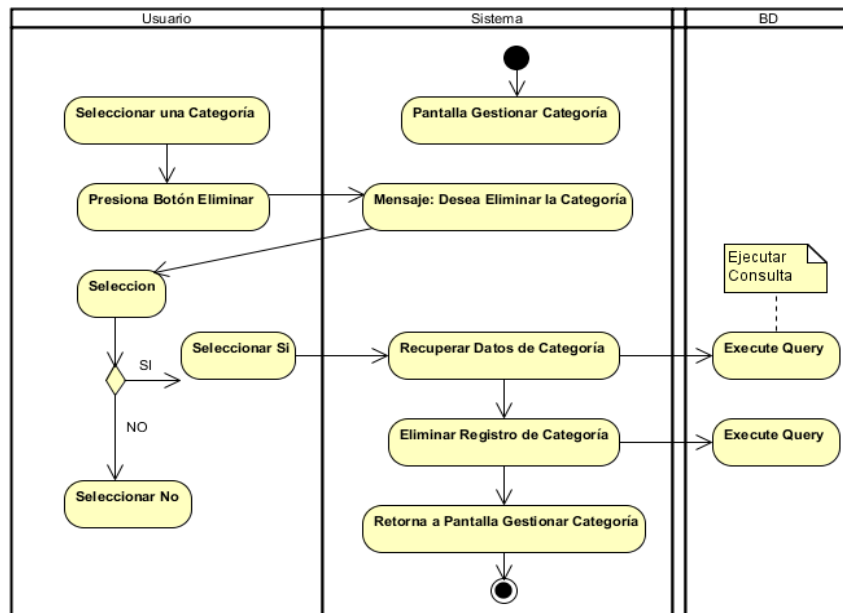


Figura 55 Diagrama de Actividad: CU Eliminar Categoría

2.1.2.2.3.14.1.4.5 Diagrama de Actividad: Caso de Uso Modificar Categoría

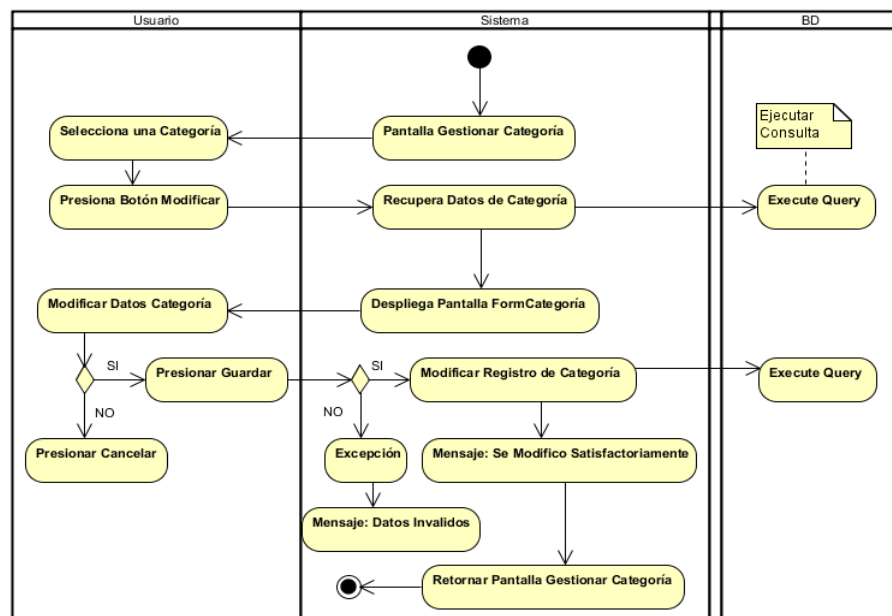


Figura 56 Diagrama de Actividad: CU Modificar Categoría

2.1.2.2.3.14.1.4.6 Diagrama de Actividad: Caso de Uso Gestionar Cliente

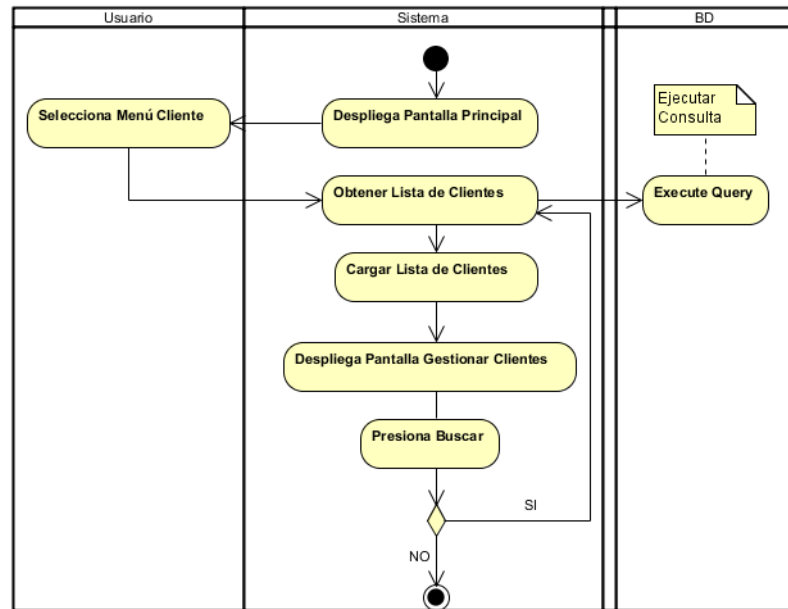


Figura 57 Diagrama de Actividad: CU Gestionar Cliente

2.1.2.2.3.14.1.4.7 Diagrama de Actividad: Caso de Uso Adicionar Cliente

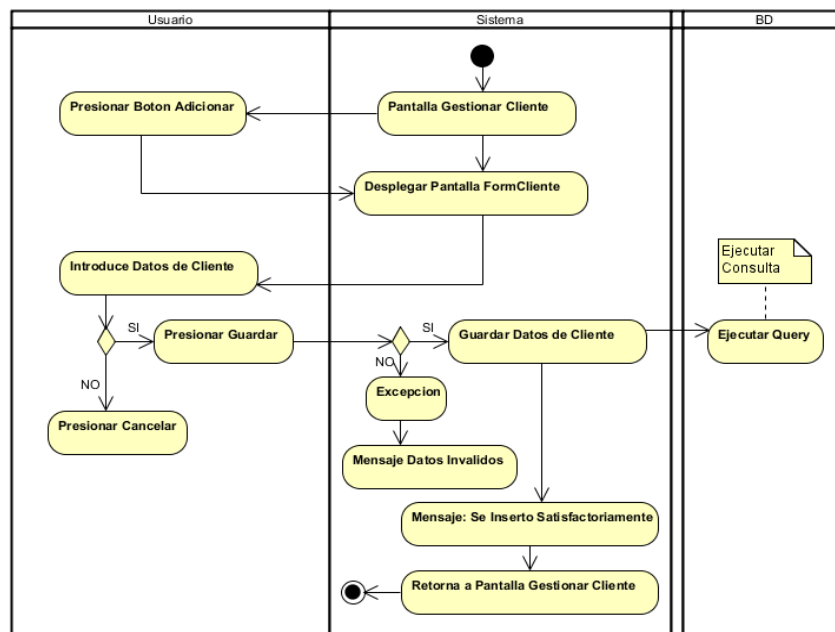


Figura 58 Diagrama de Actividad: CU Adicionar Cliente

2.1.2.2.3.14.1.4.8 Diagrama de Actividad: Caso de Uso Eliminar Cliente

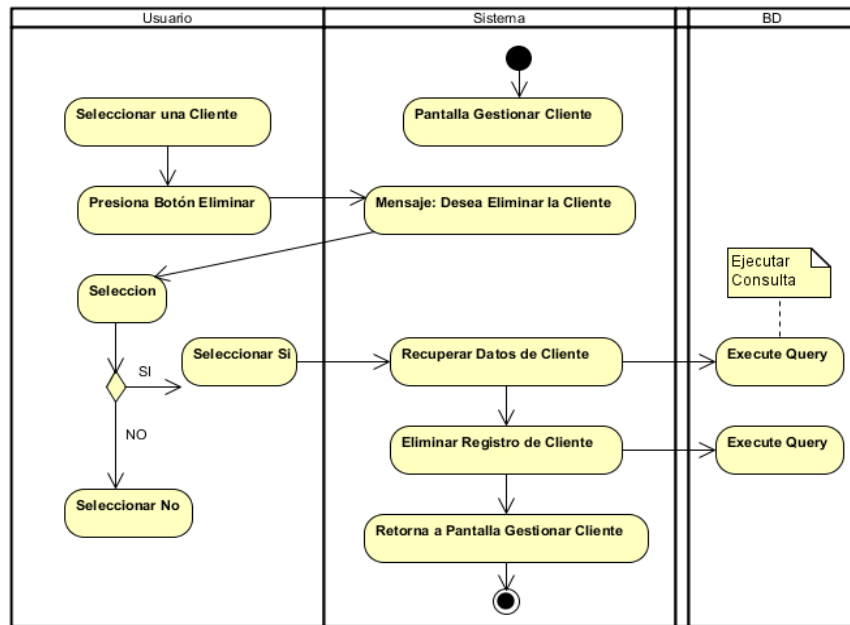


Figura 59 Diagrama de Actividad: CU Eliminar Cliente

2.1.2.2.3.14.1.4.9 Diagrama de Actividad: Caso de Uso Modificar Cliente

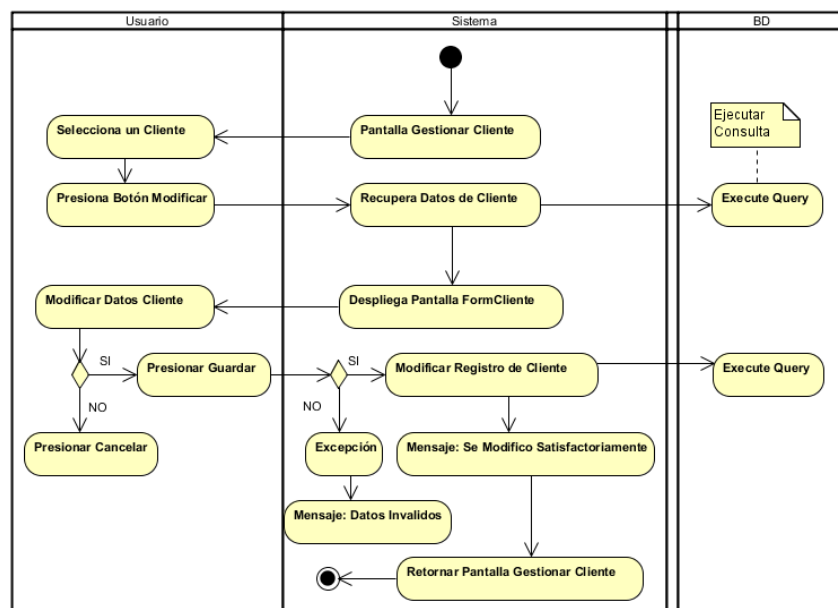


Figura 60 Diagrama de Actividad: CU Modificar Cliente

2.1.2.2.3.14.1.4.10 Diagrama de Actividad: Caso de Uso Gestionar Egreso

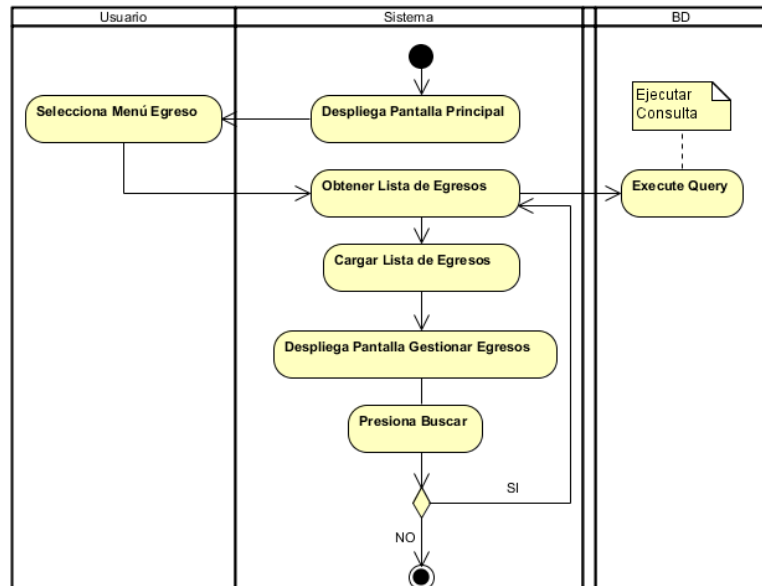


Figura 61 Diagrama de Actividad: CU Gestionar Egreso

2.1.2.2.3.14.1.4.11 Diagrama de Actividad: Caso de Uso Adicionar Egreso

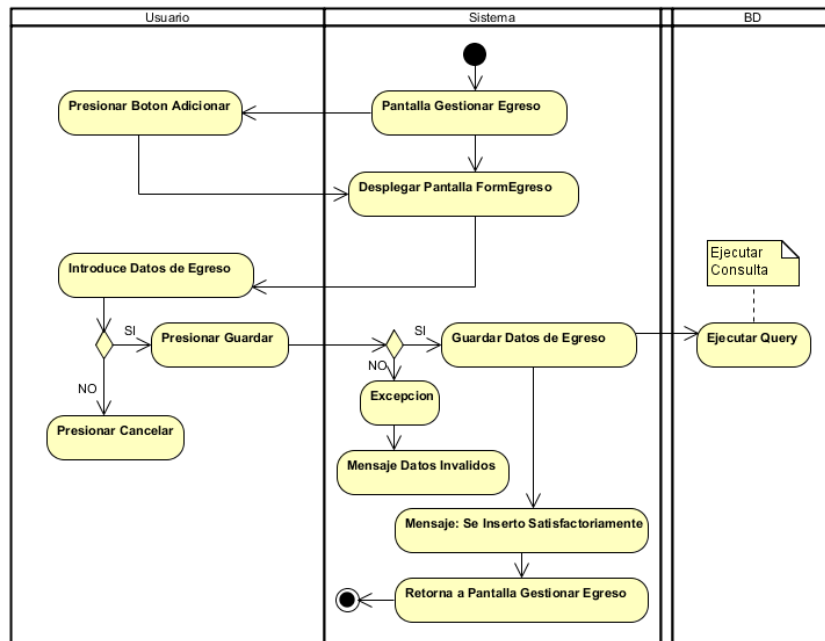


Figura 62 Diagrama de Actividad: CU Adicionar Egreso

2.1.2.2.3.14.1.4.12 Diagrama de Actividad: Caso de Uso Eliminar Egreso

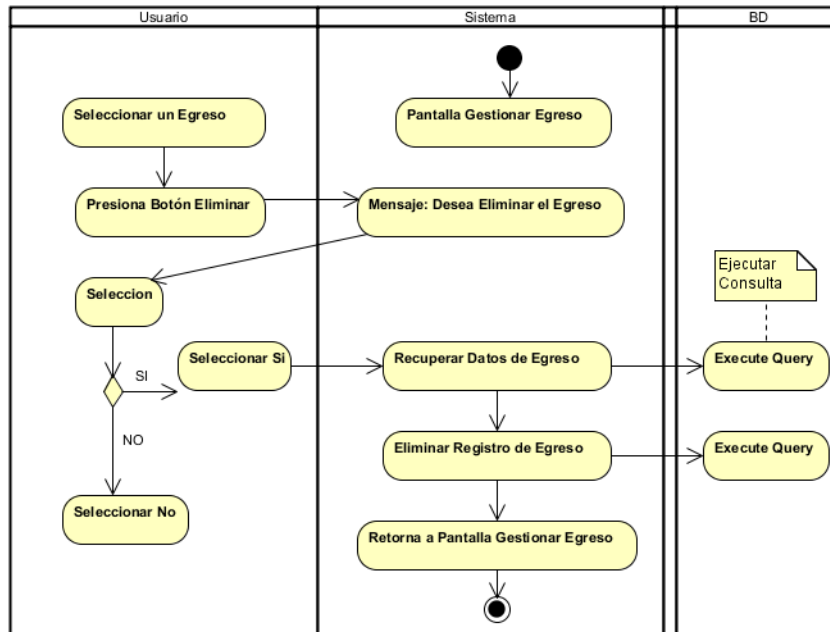


Figura 63 Diagrama de Actividad: CU Eliminar Egreso

2.1.2.2.3.14.1.4.13 Diagrama de Actividad: Caso de Uso Modificar Egreso

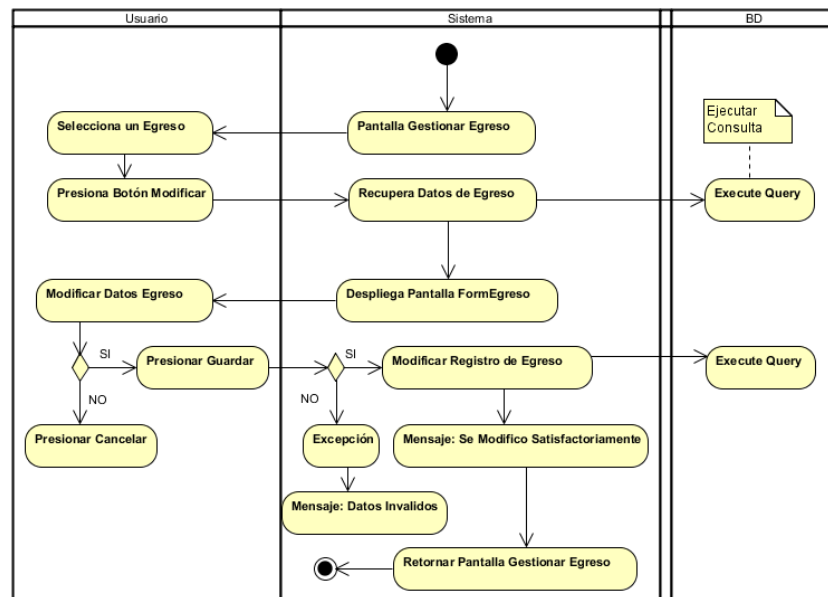


Figura 64 Diagrama de Actividad: CU Modificar Egreso

2.1.2.2.3.14.1.4.14 Diagrama de Actividad: Caso de Uso Gestionar Ingreso

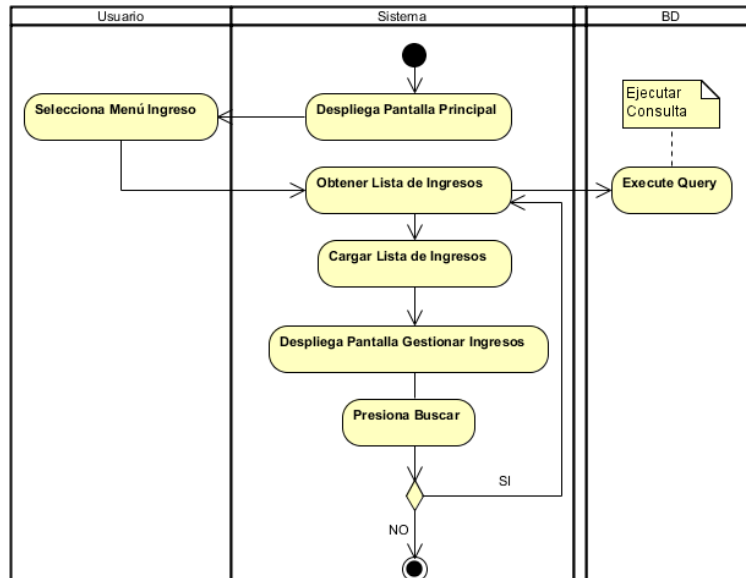


Figura 65 Diagrama de Actividad: CU Modificar Ingreso

2.1.2.2.3.14.1.4.15 Diagrama de Actividad: Caso de Uso Adicionar Ingreso

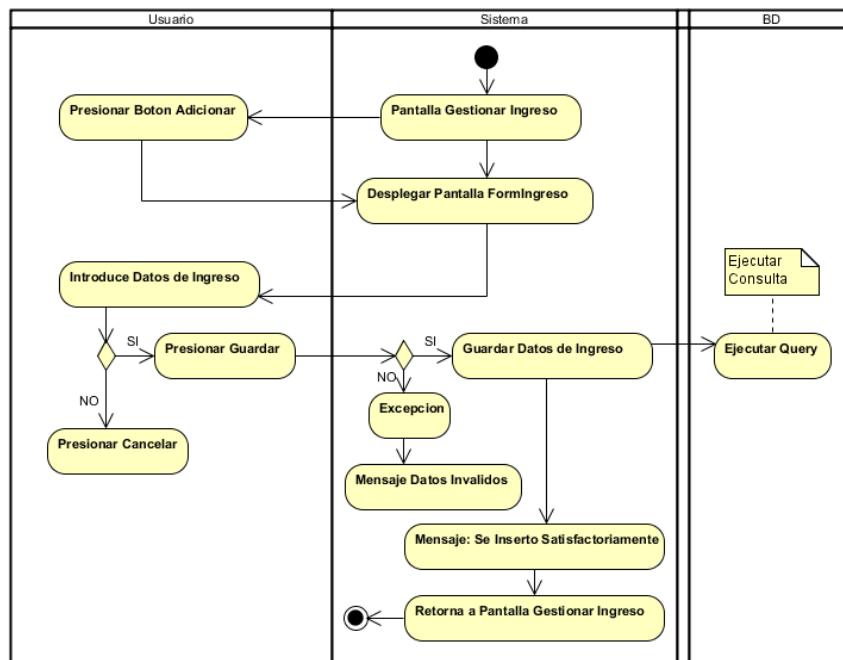


Figura 66 Diagrama de Actividad: CU Adicionar Ingreso

2.1.2.2.3.14.1.4.16 Diagrama de Actividad: Caso de Uso Eliminar Ingreso

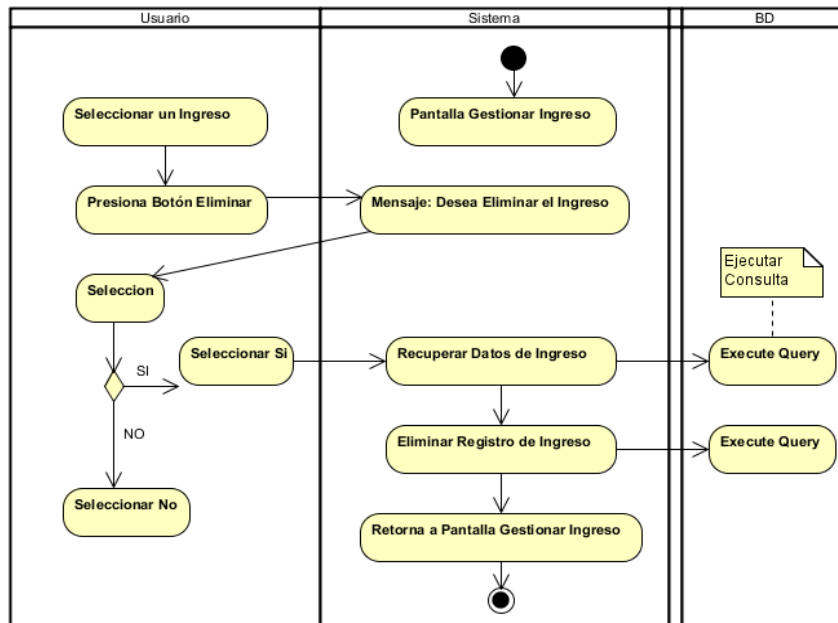


Figura 67 Diagrama de Actividad: CU Eliminar Ingreso

2.1.2.2.3.14.1.4.17 Diagrama de Actividad: Caso de Uso Modificar Ingreso

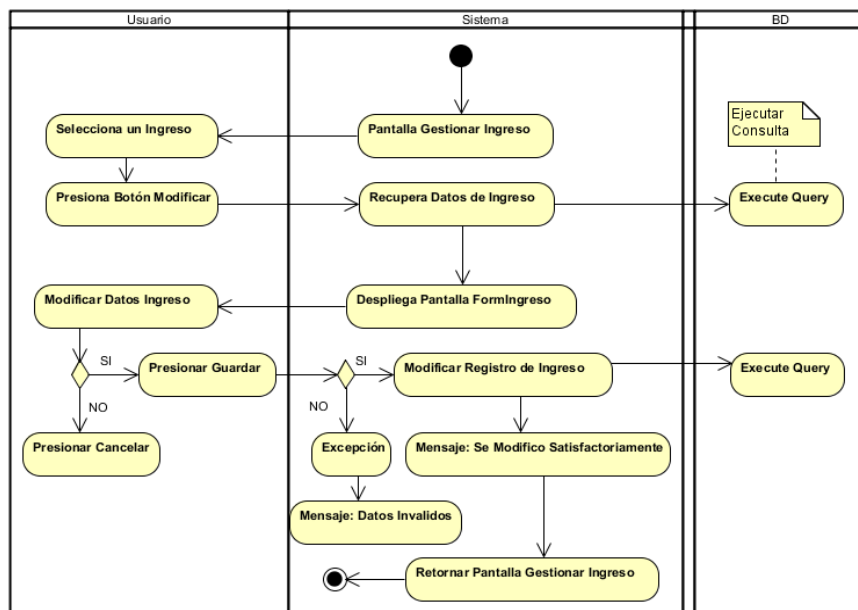


Figura 68 Diagrama de Actividad: CU Modificar Ingreso

2.1.2.2.3.14.1.4.18 Diagrama de Actividad: Caso de Uso Gestionar Noticia

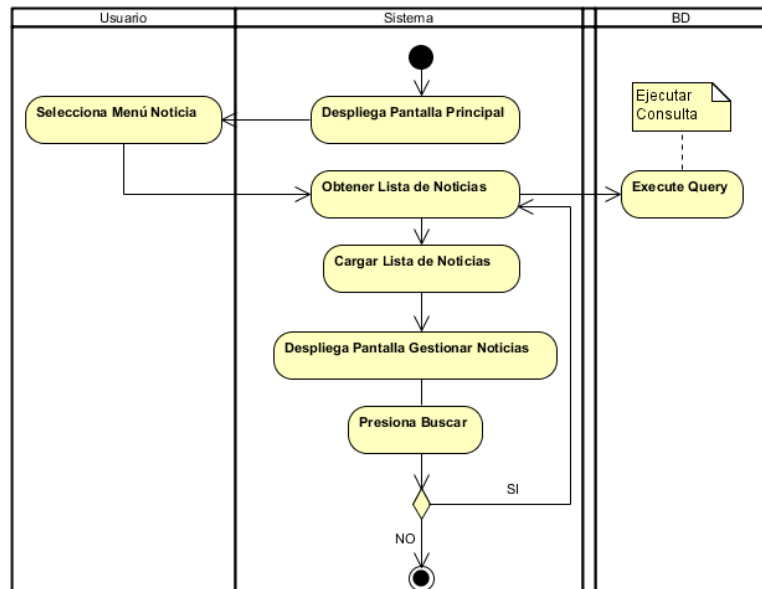


Figura 69 Diagrama de Actividad: CU Gestionar Noticia

2.1.2.2.3.14.1.4.19 Diagrama de Actividad: Caso de Uso Adicionar Noticia

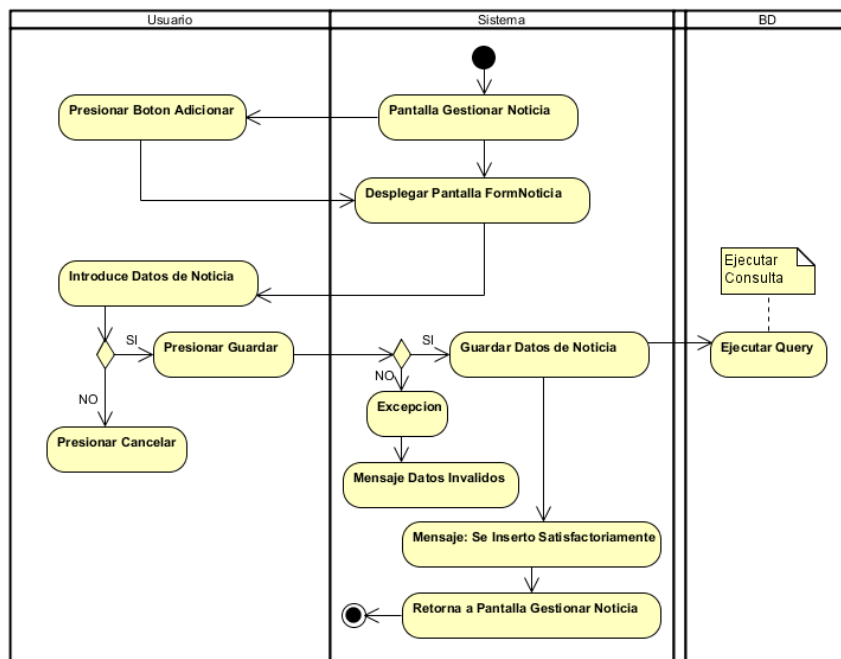


Figura 70 Diagrama de Actividad: CU Adicionar Noticia

2.1.2.2.3.14.1.4.20 Diagrama de Actividad: Caso de Uso Eliminar Noticia

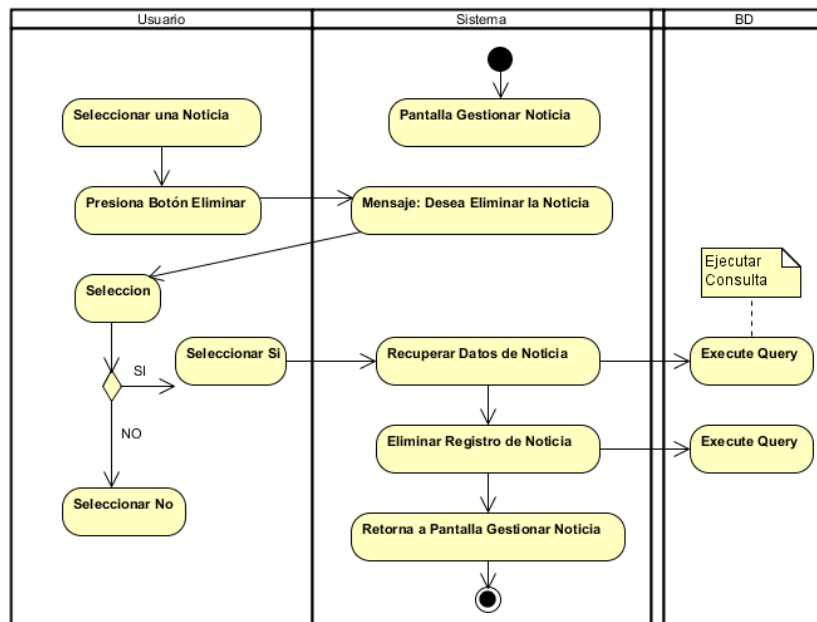


Figura 71 Diagrama de Actividad: CU Eliminar Noticia

2.1.2.2.3.14.1.4.21 Diagrama de Actividad: Caso de Uso Modificar Noticia

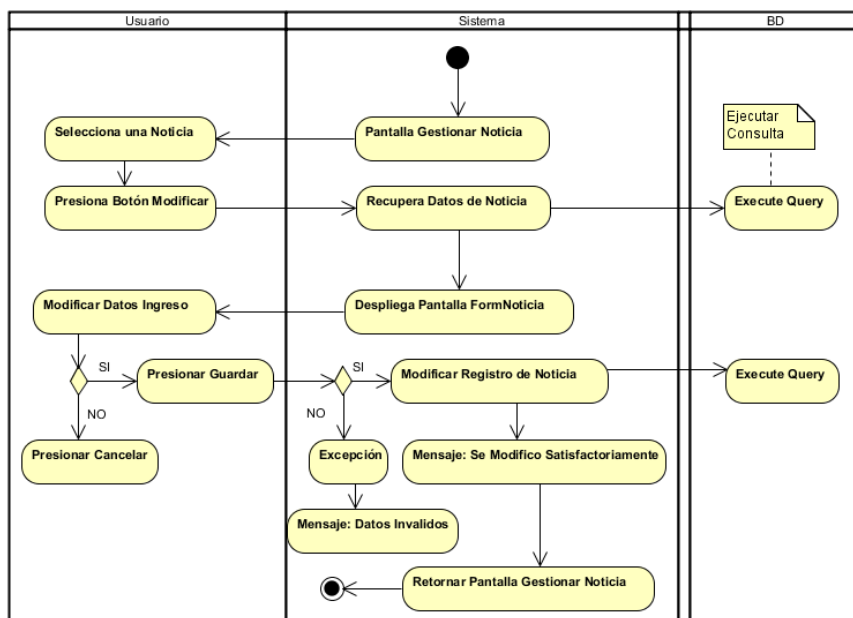


Figura 72 Diagrama de Actividad: CU Modificar Noticia

2.1.2.2.3.14.1.4.22 Diagrama de Actividad: Caso de Uso Gestionar Periodo

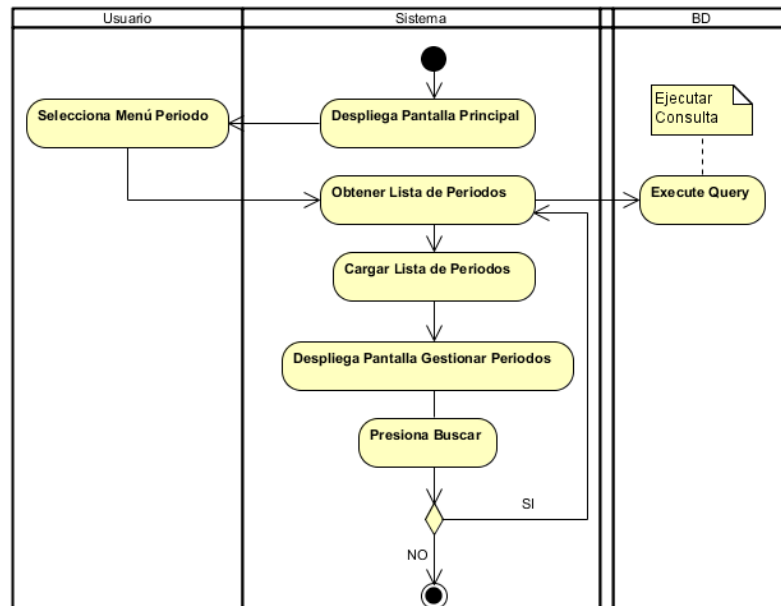


Figura 73 Diagrama de Actividad: CU Gestionar Periodo

2.1.2.2.3.14.1.4.23 Diagrama de Actividad: Caso de Uso Adicionar Periodo

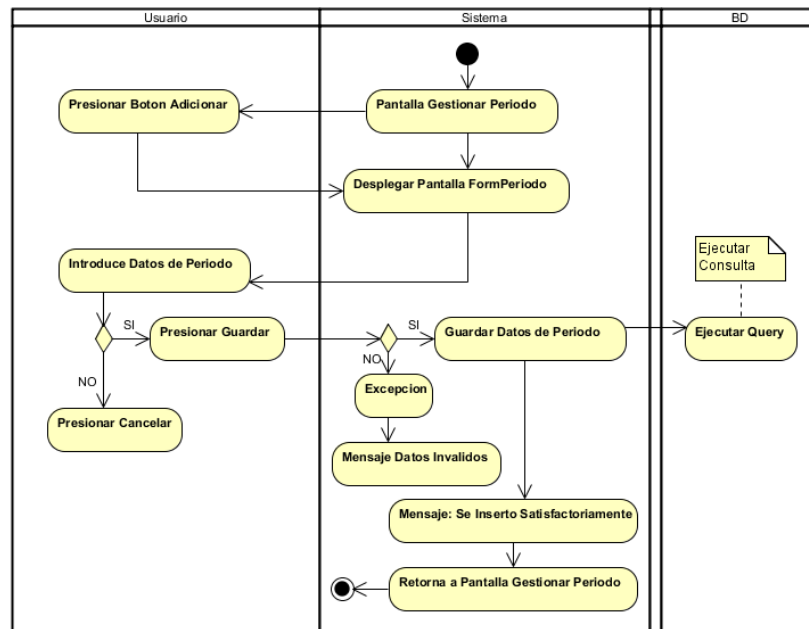


Figura 74 Diagrama de Actividad: CU Adicionar Periodo

2.1.2.2.3.14.1.4.24 Diagrama de Actividad: Caso de Uso Eliminar Periodo

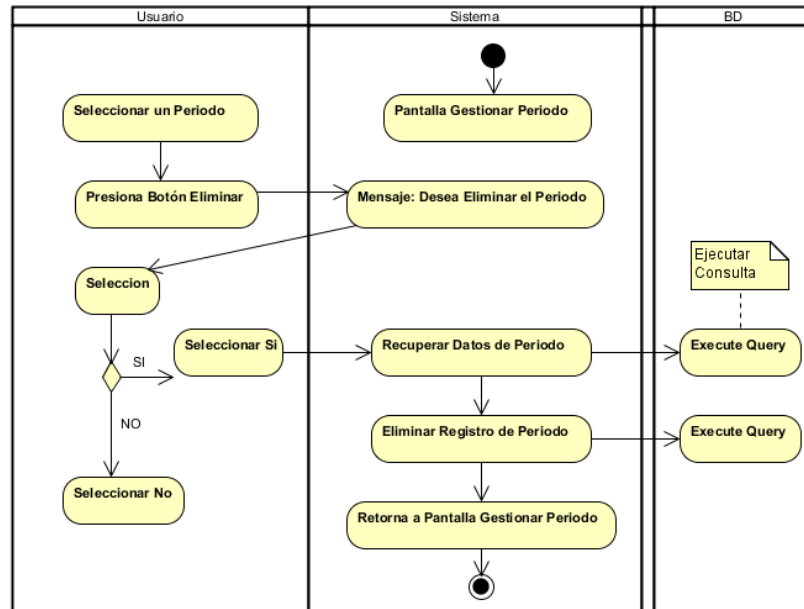


Figura 75 Diagrama de Actividad: CU Eliminar Periodo

2.1.2.2.3.14.1.4.25 Diagrama de Actividad: Caso de Uso Modificar Periodo

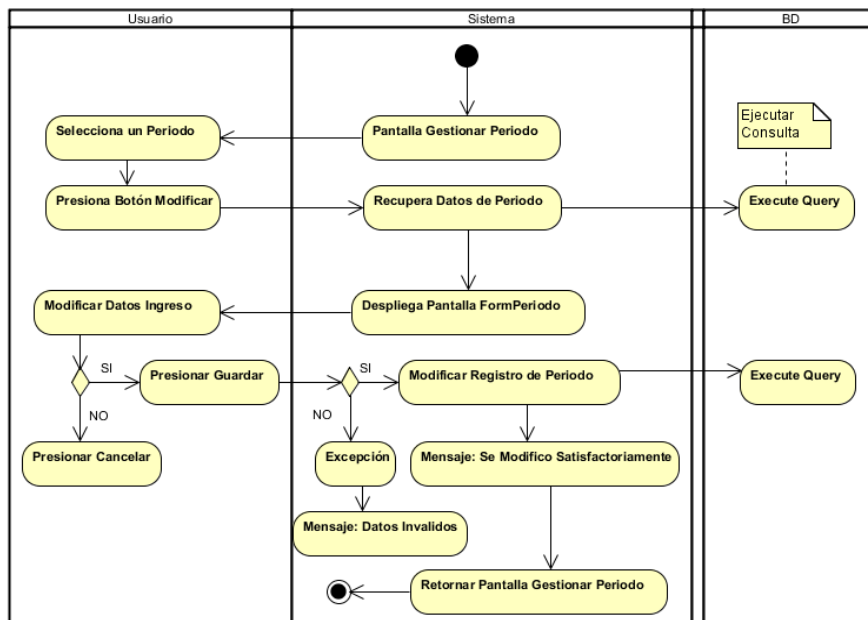


Figura 76 Diagrama de Actividad: CU Modificar Periodo

2.1.2.2.3.14.1.4.26 Diagrama de Actividad : Caso de Uso Gestionar Producto

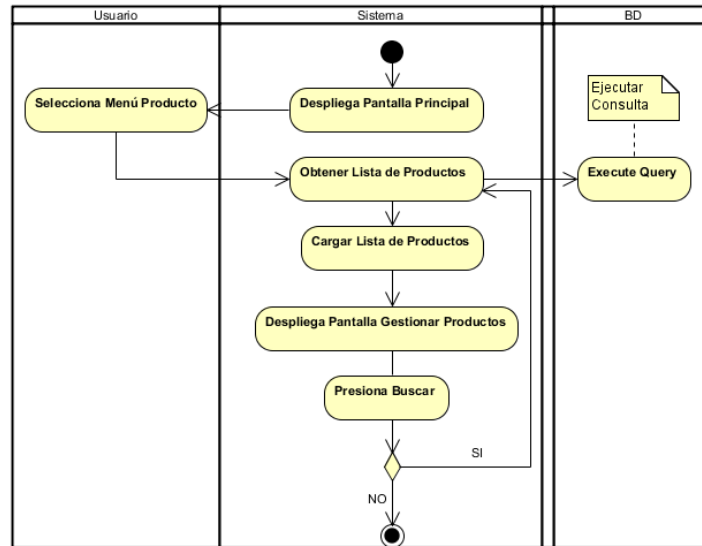


Figura 77 Diagrama de Actividad: CU Gestionar Producto

2.1.2.2.3.14.1.4.27 Diagrama de Actividad : Caso de Uso Adicionar Producto

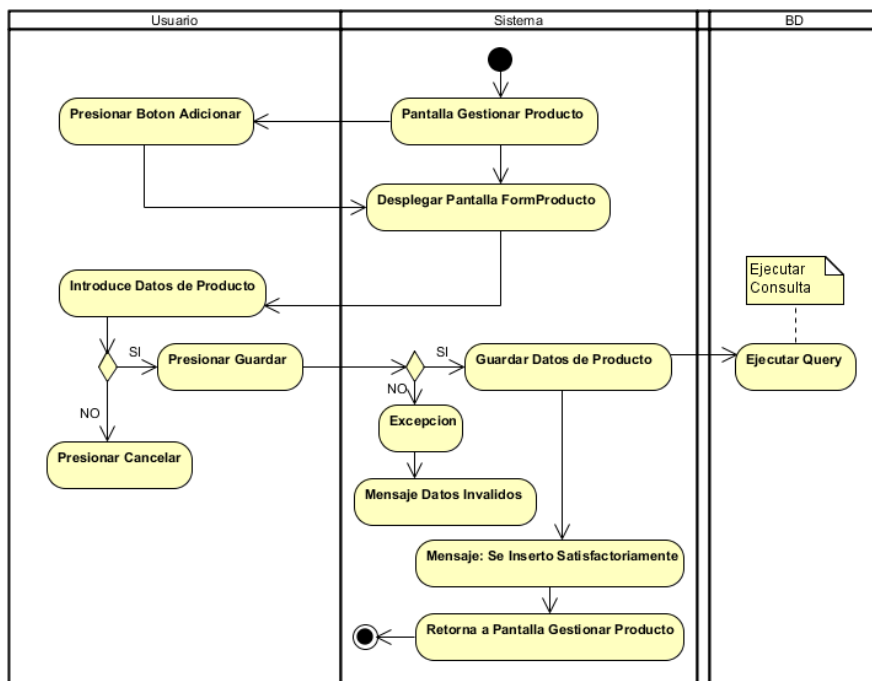


Figura 78 Diagrama de Actividad: CU Adicionar Producto

2.1.2.2.3.14.1.4.28 Diagrama de Actividad : Caso de Uso Dar de Baja Producto

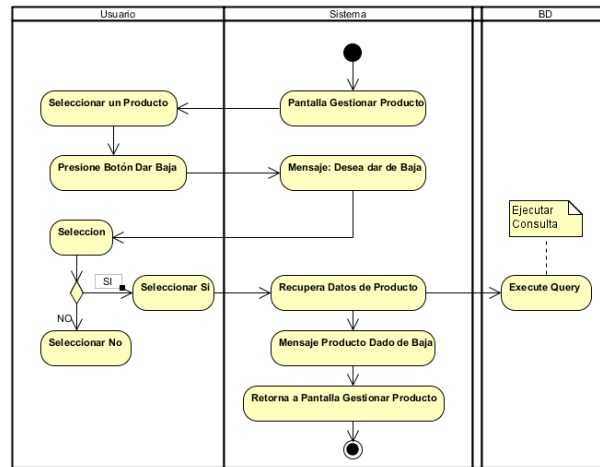


Figura 79 Diagrama Actividad: CU Dar de Baja Producto

2.1.2.2.3.14.1.4.29 Diagrama de Actividad : Caso de Uso Modificar Producto

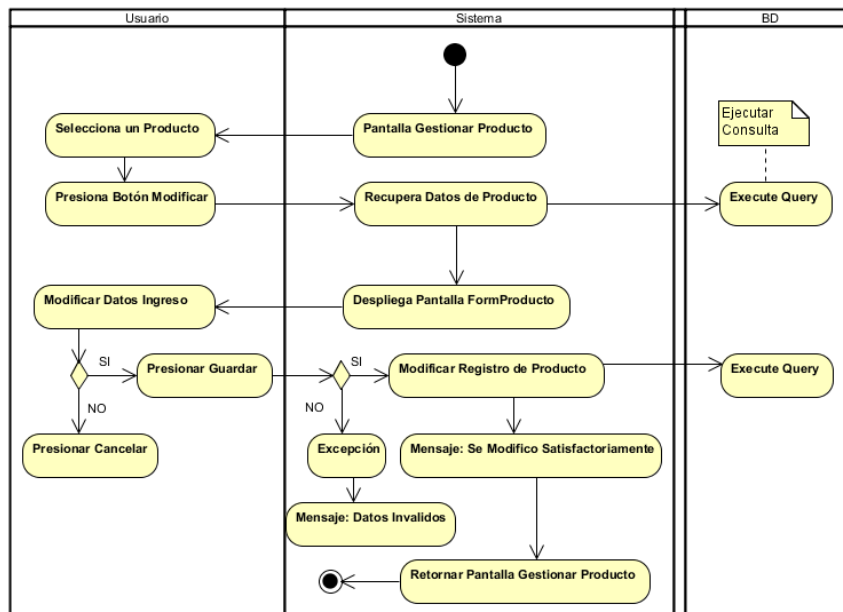


Figura 80 Diagrama de Actividad: CU Modificar Producto

2.1.2.2.3.14.1.4.30 Diagrama de Actividad : Caso de Uso Gestionar Rol

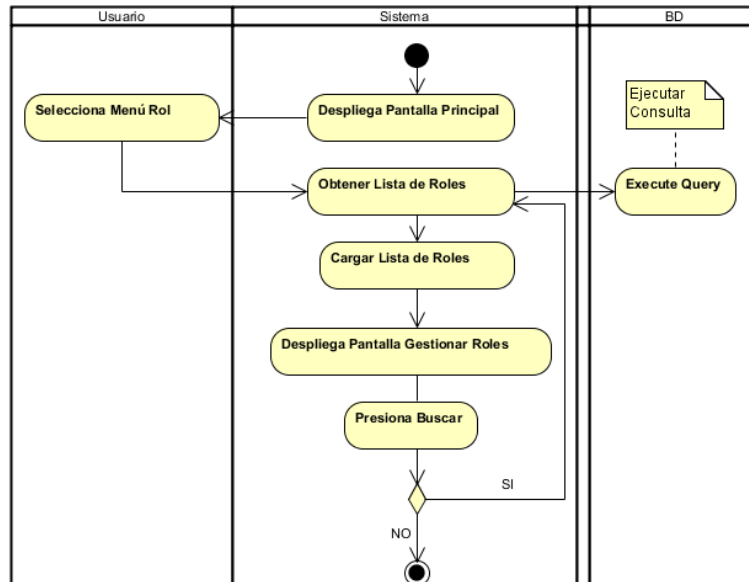


Figura 81 Diagrama de Actividad: CU Gestionar Rol

2.1.2.2.3.14.1.4.31 Diagrama de Actividad: Caso de Uso Adicionar Rol

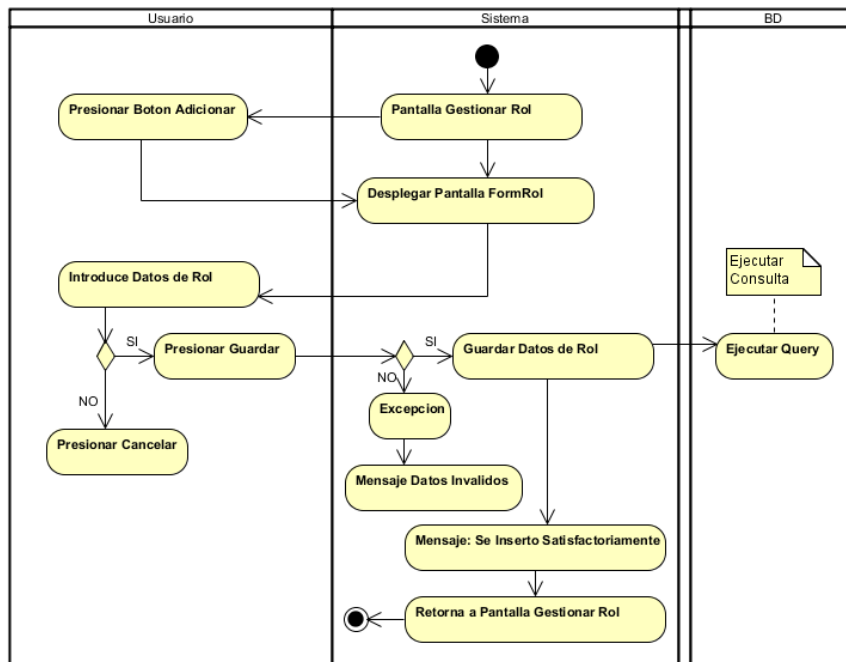


Figura 82 Diagrama de Actividad: CU Adicionar Rol

2.1.2.2.3.14.1.4.32 Diagrama de Actividad: Caso de Uso Eliminar Rol

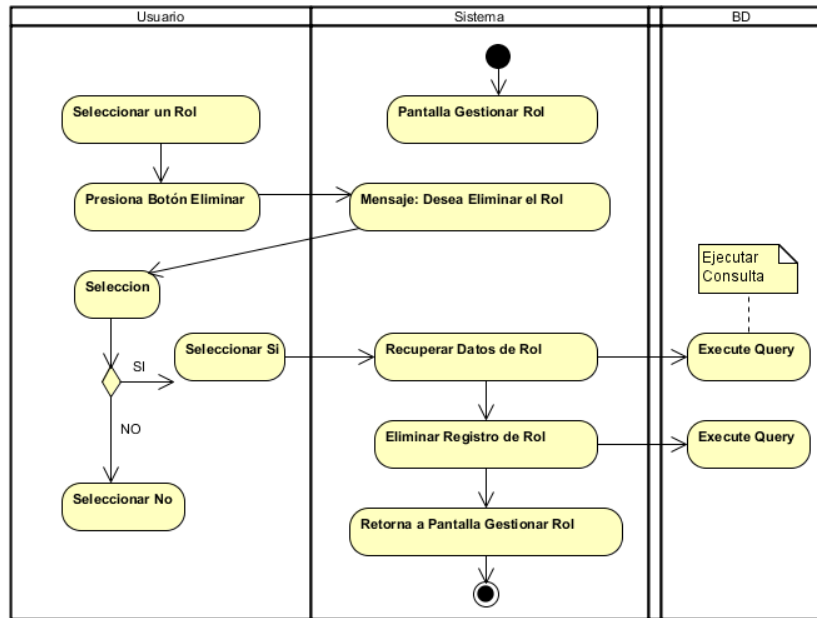


Figura 83 Diagrama de Actividad: CU Eliminar Rol

2.1.2.2.3.14.1.4.33 Diagrama de Actividad: Caso de Uso Modificar Rol

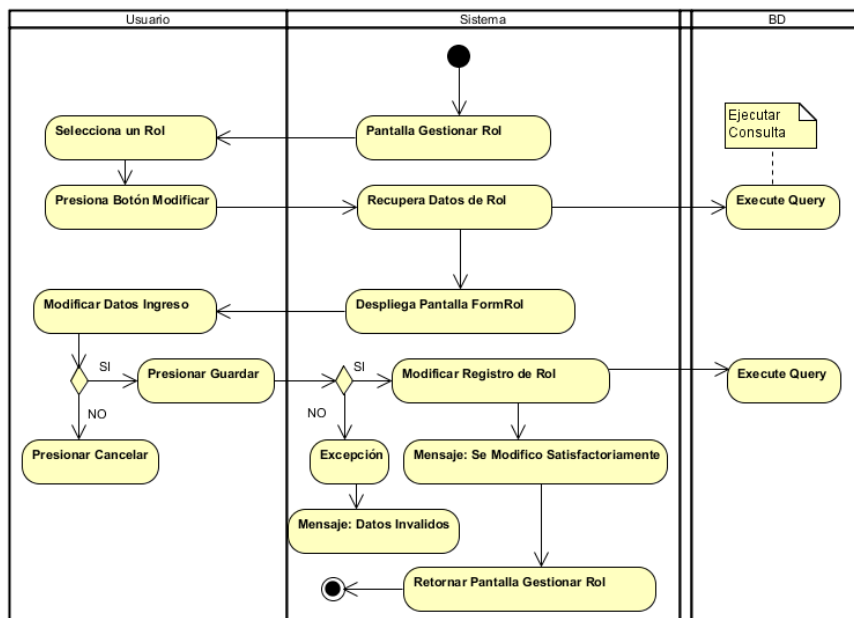


Figura 84 Diagrama de Actividad: CU Modificar Rol

2.1.2.2.3.14.1.4.34 Diagrama de Actividad: Caso de Uso Gestionar Sucursal

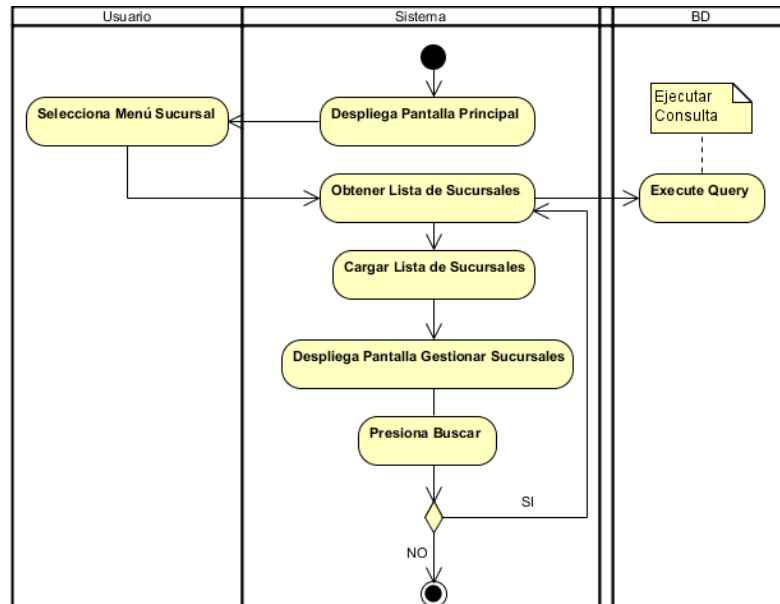


Figura 85 Diagrama de Actividad: CU Gestionar Sucursal

2.1.2.2.3.14.1.4.35 Diagrama de Actividad: Caso de Uso Adicionar Sucursal

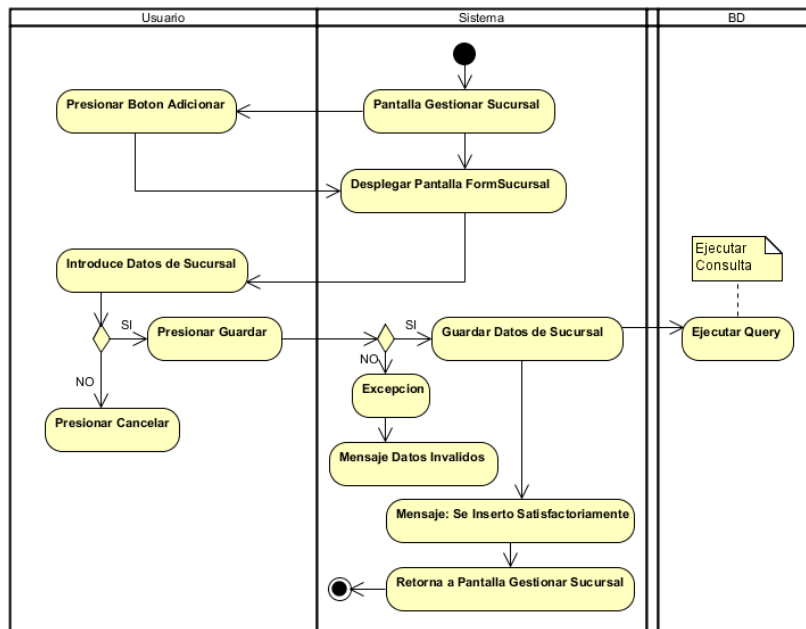


Figura 86 Diagrama de Actividad: CU Adicionar Sucursal

2.1.2.2.3.14.1.4.36 Diagrama de Actividad: Caso de Uso Dar de Baja Sucursal

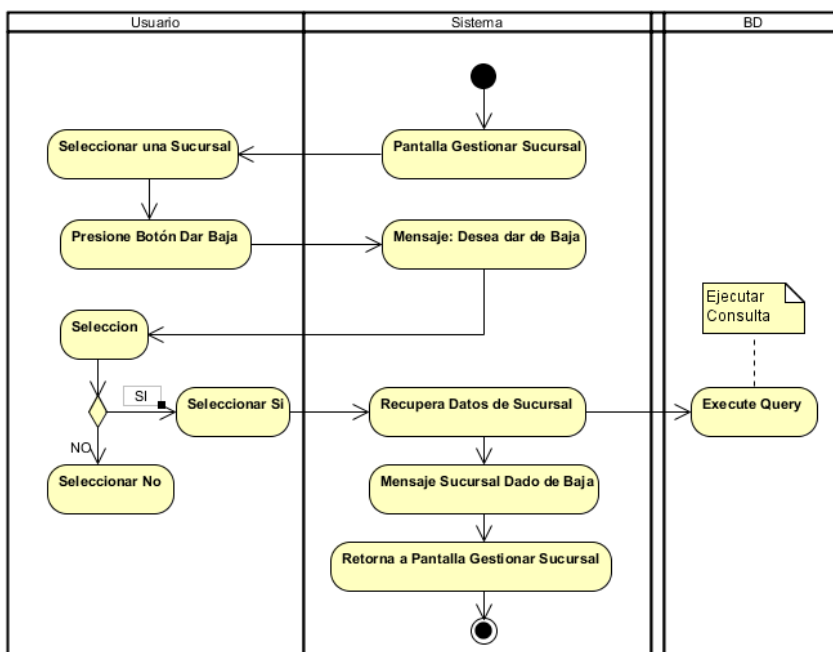


Figura 87 Diagrama de Actividad: CU Dar de Baja Sucursal

2.1.2.2.3.14.1.4.37 Diagrama de Actividad: Caso de Uso Modificar Sucursal

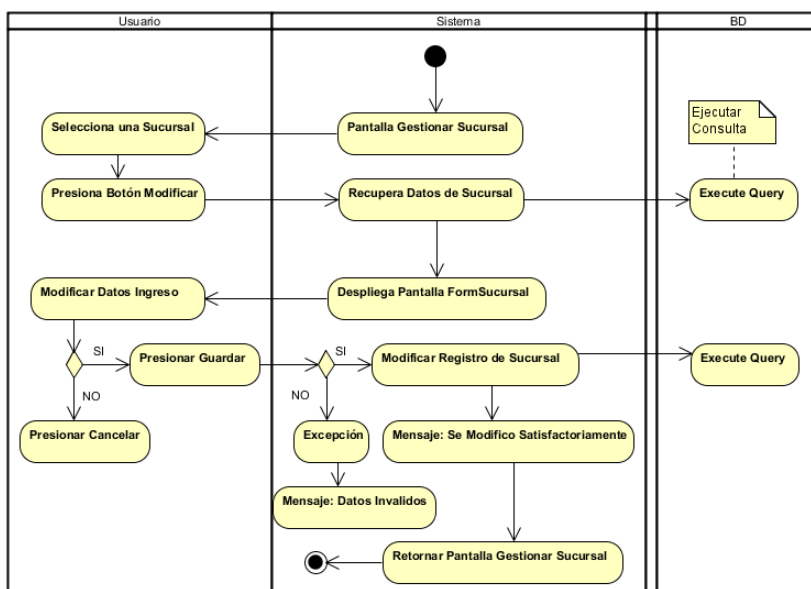


Figura 88 Diagrama de Actividad: CU Modificar Sucursal

2.1.2.2.3.14.1.4.38 Diagrama de Actividad: Caso de Uso Gestionar Usuario

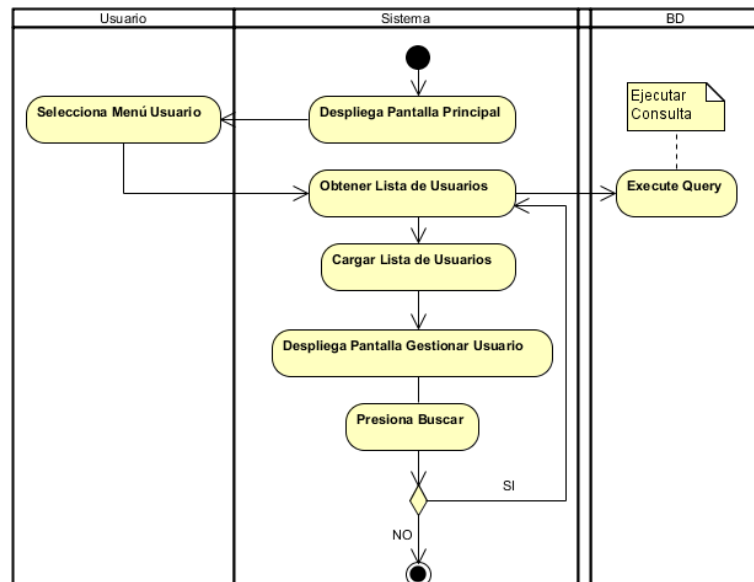


Figura 89 Diagrama de Actividad: CU Gestionar Usuario

2.1.2.2.3.14.1.4.39 Diagrama de Actividad: Caso de Uso Adicionar Usuario

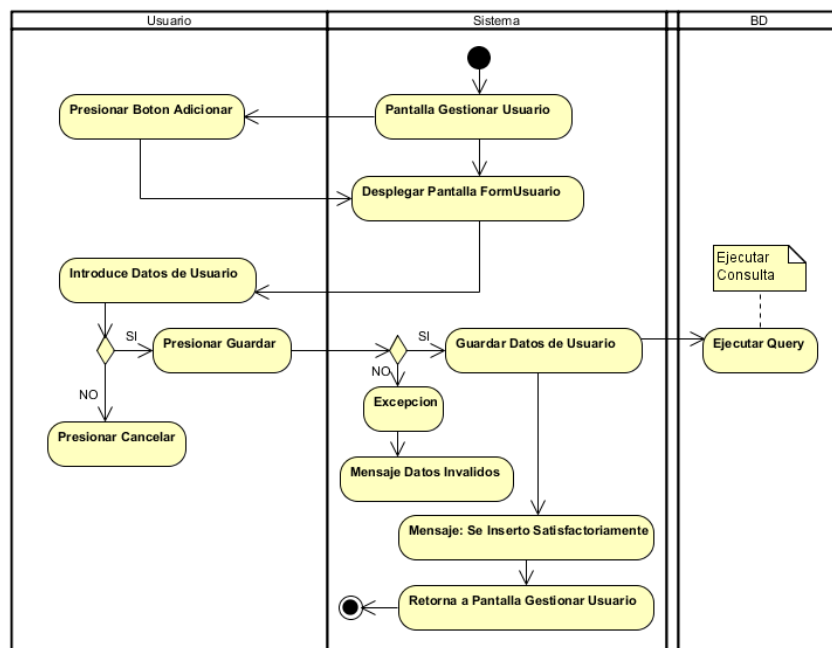


Figura 90 Diagrama de Actividad: CU Adicionar Usuario

2.1.2.2.3.14.1.4.40 Diagrama de Actividad: Caso de Uso Dar de Baja Usuario

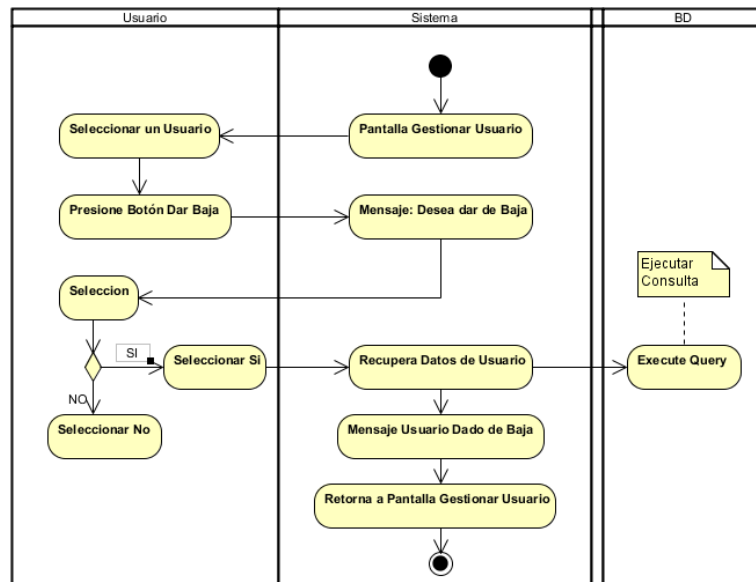


Figura 91 Diagrama de Actividad: CU Dar de Baja Usuario

2.1.2.2.3.14.1.4.41 Diagrama de Actividad: Caso de Uso Modificar Usuario

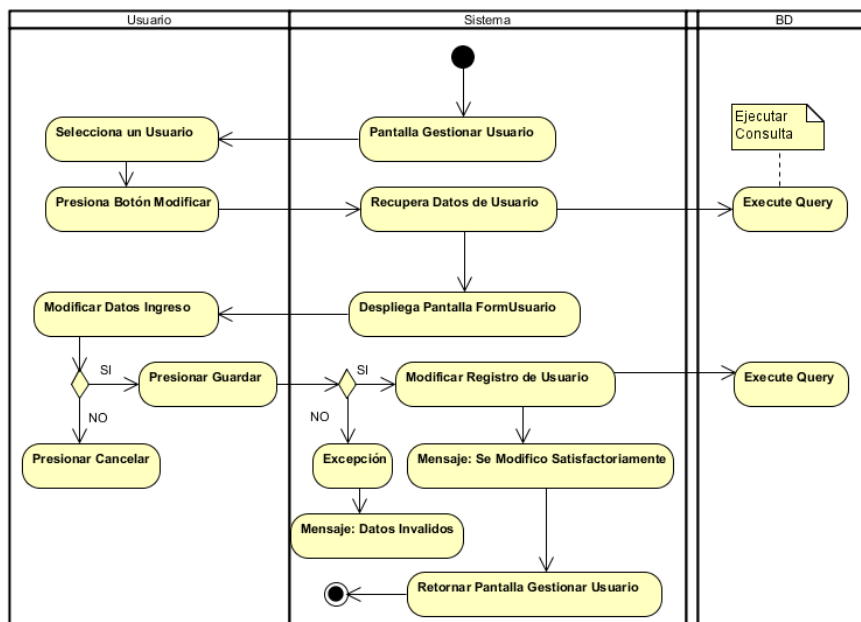


Figura 92 Diagrama de Actividad: CU Modificar Usuario

2.1.2.2.3.14.1.4.42 Diagrama de Actividad: Caso de Uso Gestionar Venta Web

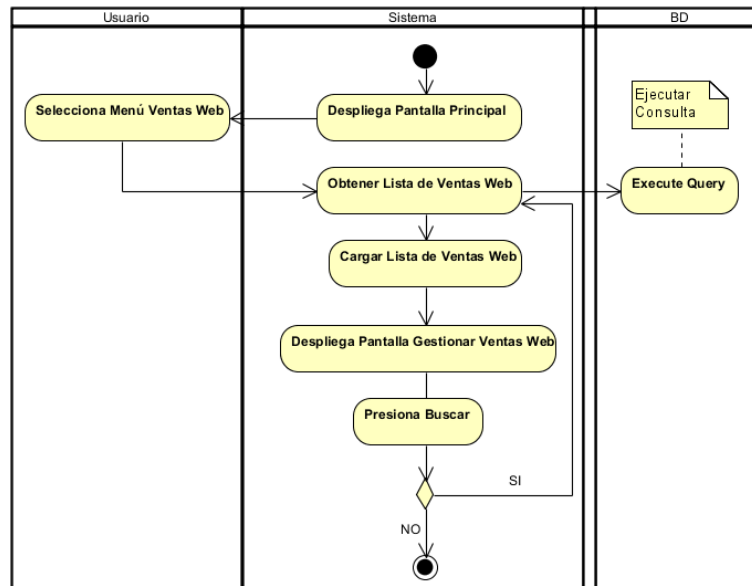


Figura 93 Diagrama de Actividad: CU Gestionar Venta Web

2.1.2.2.3.14.1.4.43 Diagrama de Actividad: Caso de Uso Eliminar Venta Web

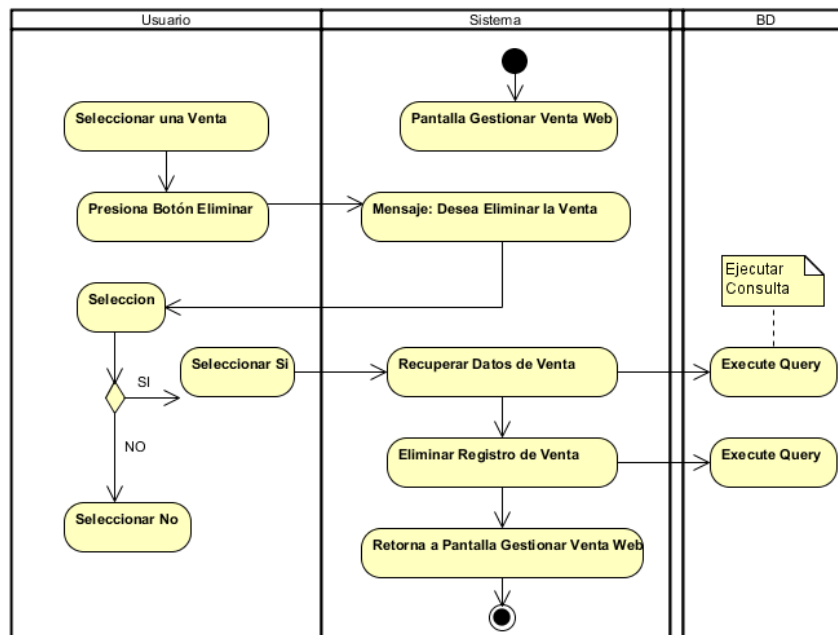


Figura 94 Diagrama de Actividad: CU Eliminar Venta Web

2.1.2.2.3.14.1.4.44 Diagrama de Actividad: Caso de Uso Adicionar Venta Web

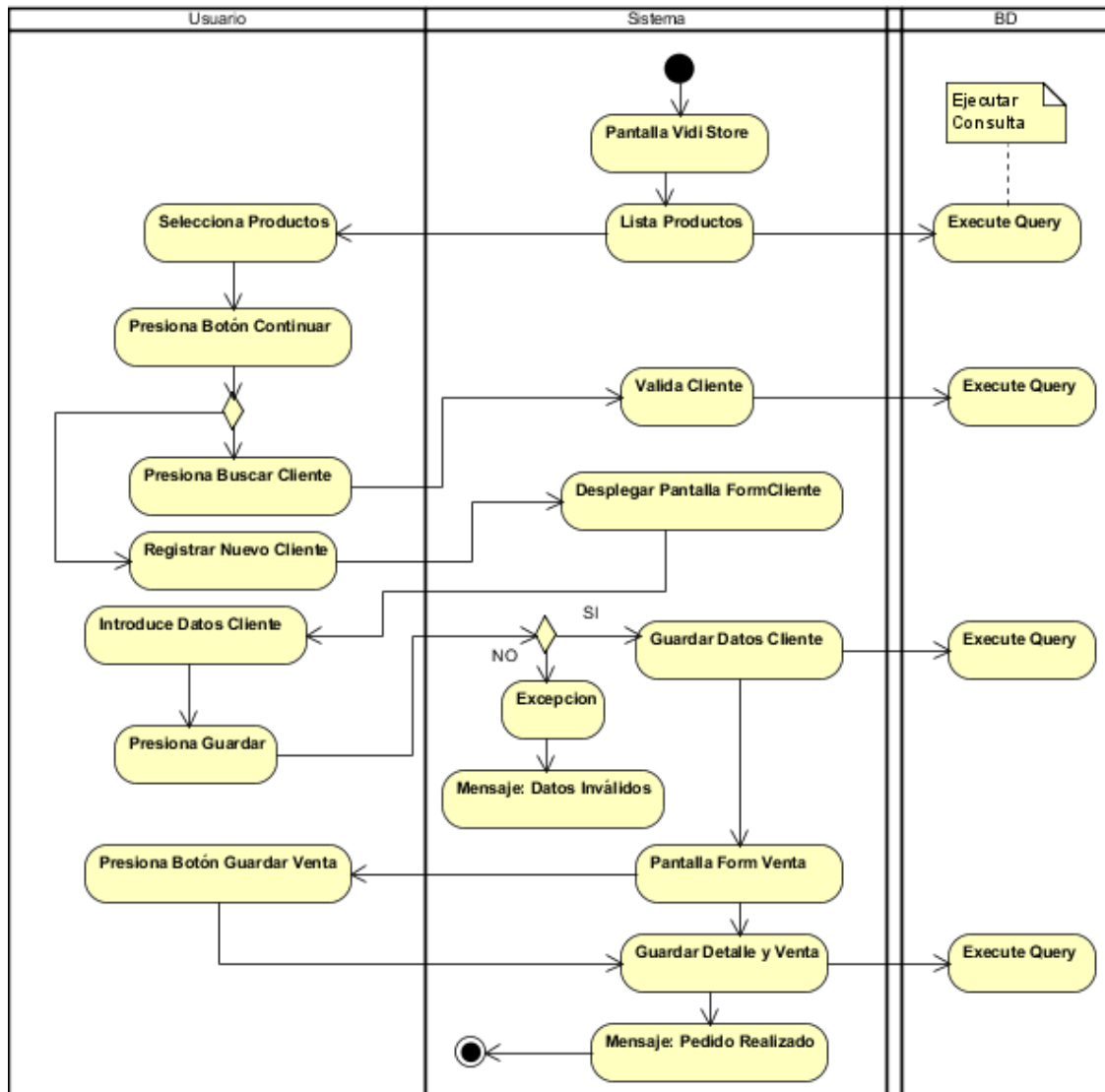


Figura 95 Diagrama de Actividad: CU Adicionar Venta Web

2.1.2.2.3.14.1.4.45 Diagrama de Actividad: Caso de Uso Confirmar Venta Web

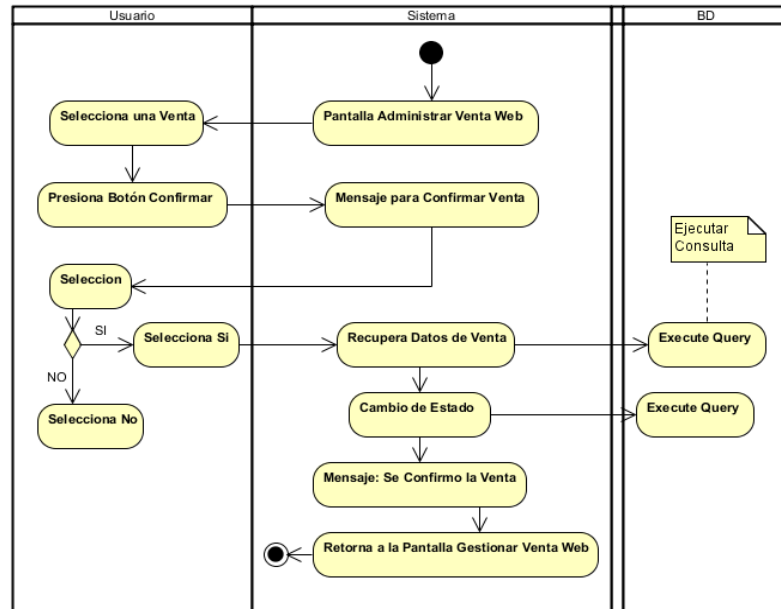


Figura 96 Diagrama de Actividad: CU Confirmar Venta Web

2.1.2.2.3.14.1.4.46 Diagrama de Actividad: Caso de Uso Confirmar Entrega de Venta Web

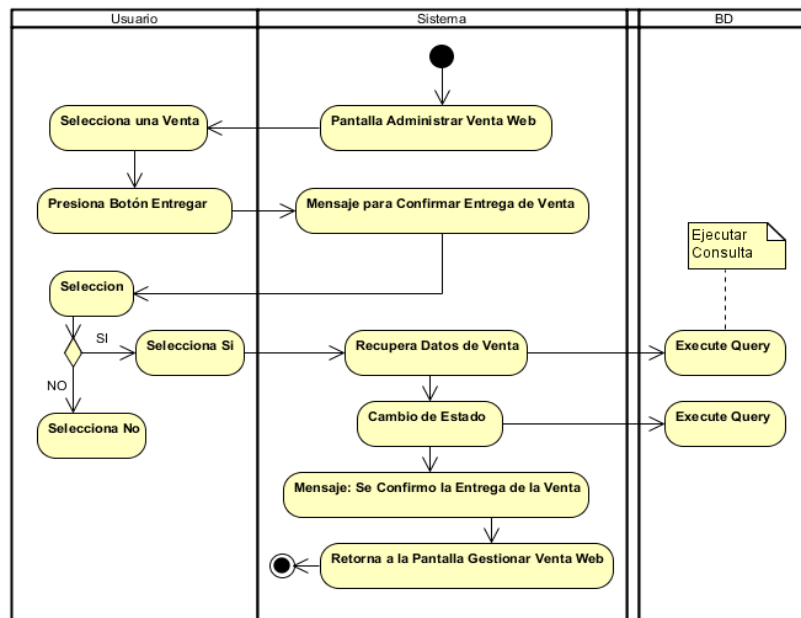


Figura 97 Diagrama de Actividad: CU Confirmar Entrega Venta Web

2.1.2.2.3.14.1.4.47 Diagrama de Actividad

2.1.2.2.3.14.1.4.48 Diagrama de Actividad

2.1.2.2.3.14.1.4.49 Diagrama de Actividad

2.1.2.2.3.14.1.4.50 Diagrama de Actividad

2.1.2.2.3.14.1.4.51 Diagrama de Actividad

2.1.2.2.3.14.1.4.52 Diagrama de Actividad

2.1.2.2.3.14.2 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 cuanto 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.

2.1.2.2.3.14.2.1 Diagrama de Secuencia

2.1.2.2.3.14.2.1.1 Diagrama de Secuencia: Caso de Uso Ingresar al Sistema

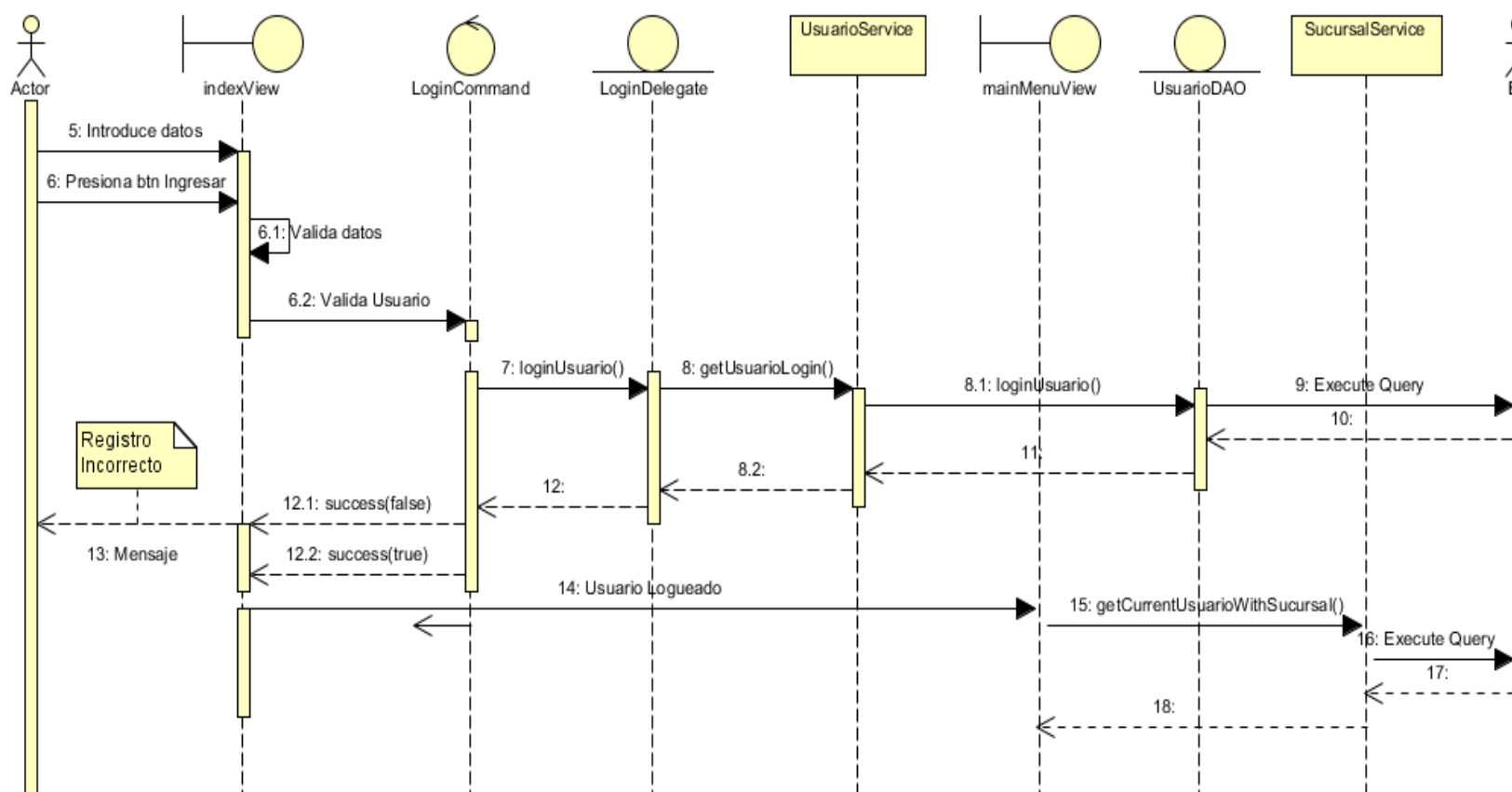


Figura 98 Diagrama de Secuencia: Caso de Uso Ingresar al Sistema

2.1.2.2.3.14.2.1.2 Diagrama de Secuencia: Caso de Uso Gestionar Categoría

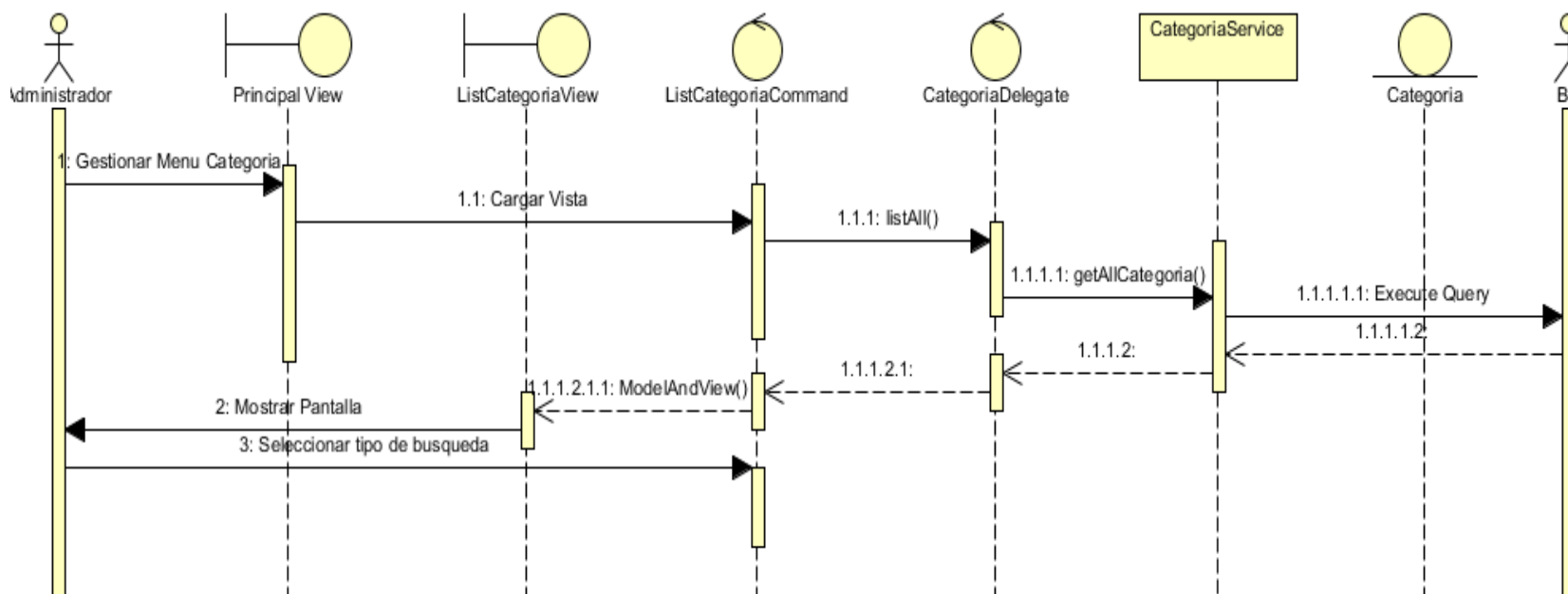


Figura 99 Diagrama de Secuencia: Caso de Uso Gestionar Categoría

2.1.2.2.3.14.2.1.3 Diagrama de Secuencia: Caso de Uso Adicionar Categoría

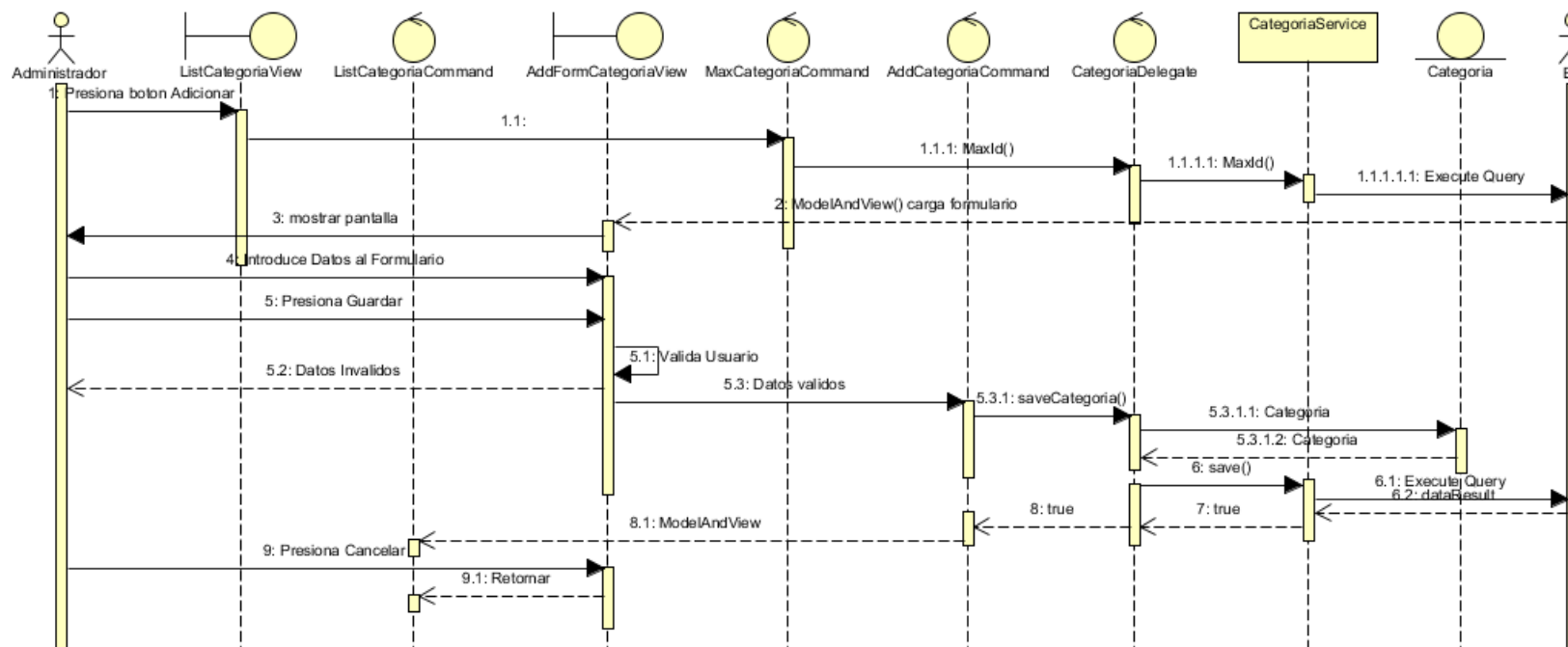


Figura 100 Diagrama de Secuencia: Caso de Uso Adicionar Categoría

2.1.2.2.3.14.2.1.4 Diagrama de Secuencia: Caso de Uso Eliminar Categoría

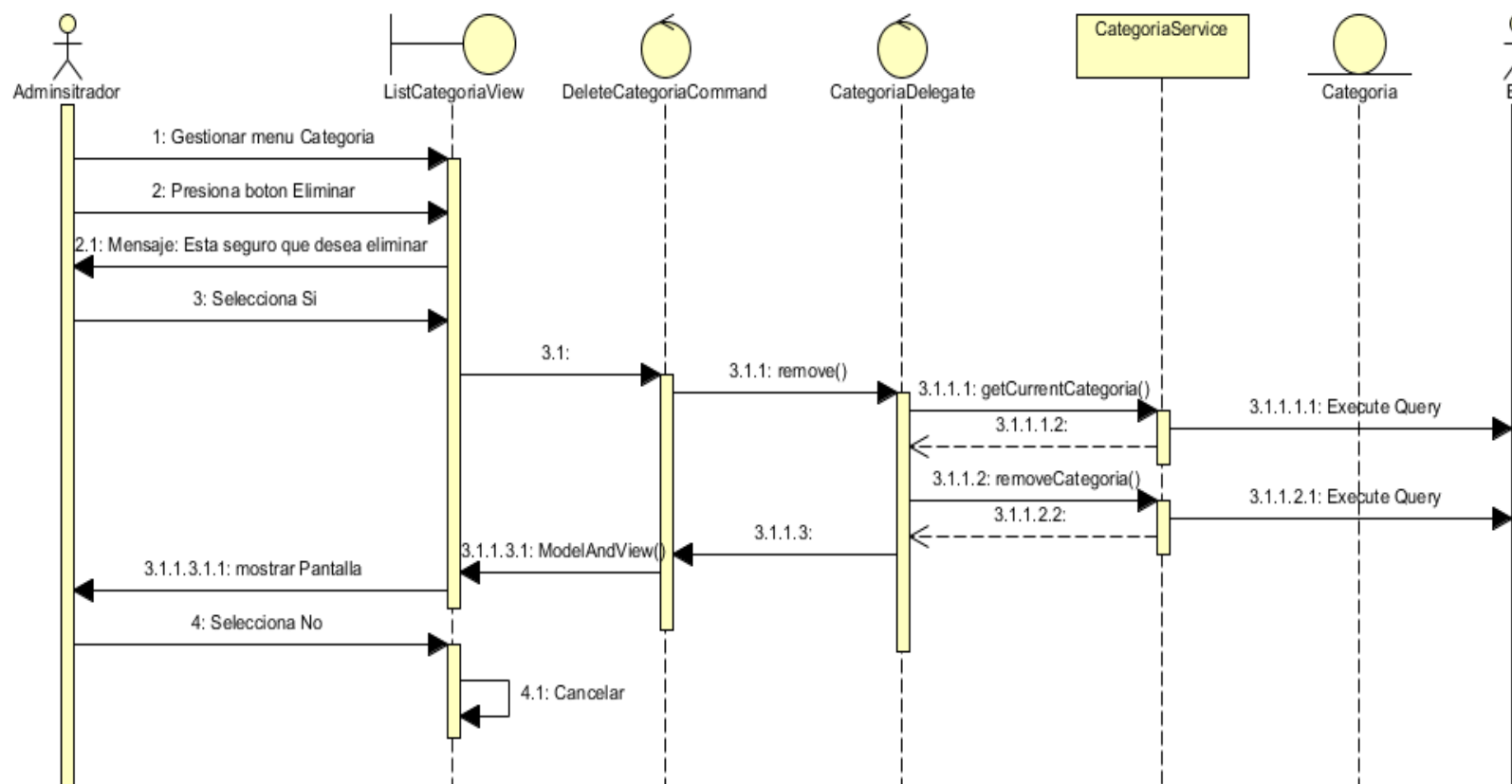


Figura 101 Diagrama de Secuencia: Caso de Uso Eliminar Categoría

2.1.2.2.3.14.2.1.5 Diagrama de Secuencia: Caso de Uso Modificar Categoría

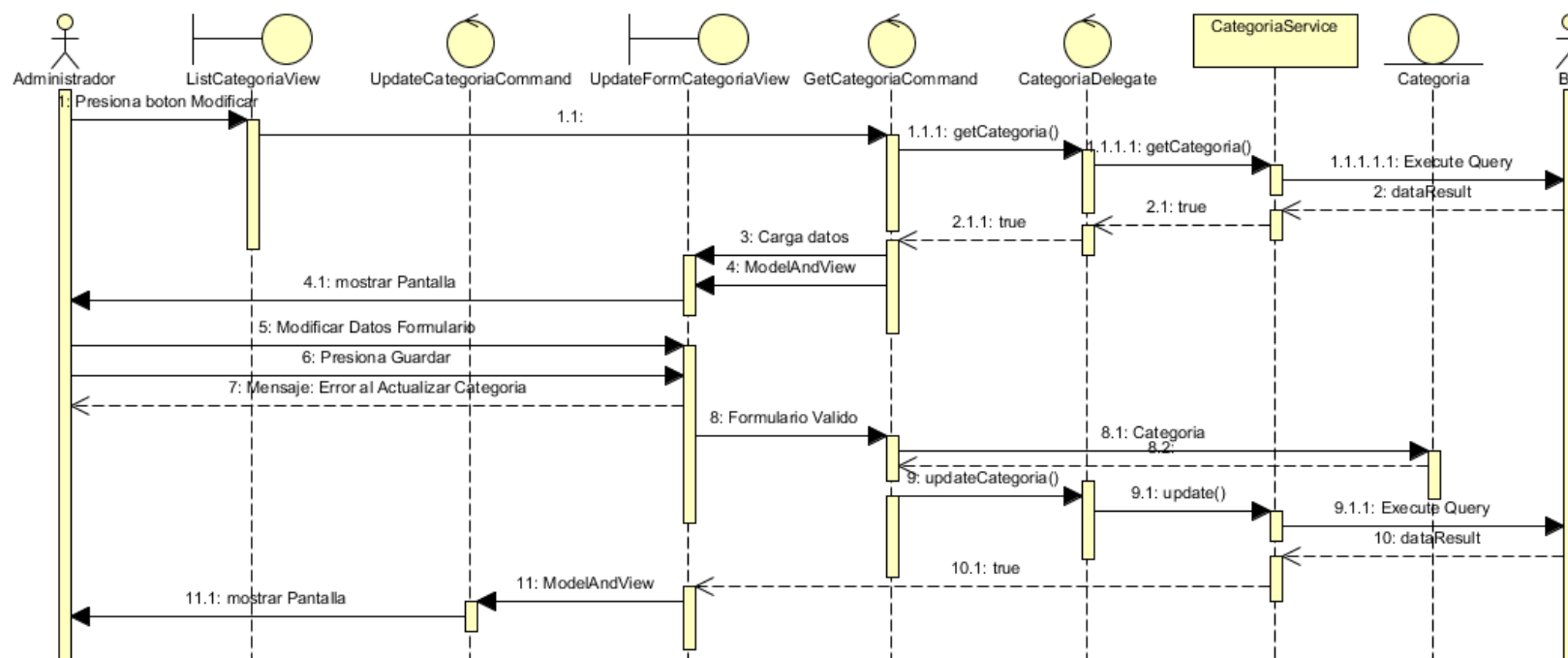


Figura 102 Diagrama de Secuencia: Caso de Uso Modificar Categoría

2.1.2.2.3.14.2.1.6 Diagrama de Secuencia: Caso de Uso Gestionar Cliente

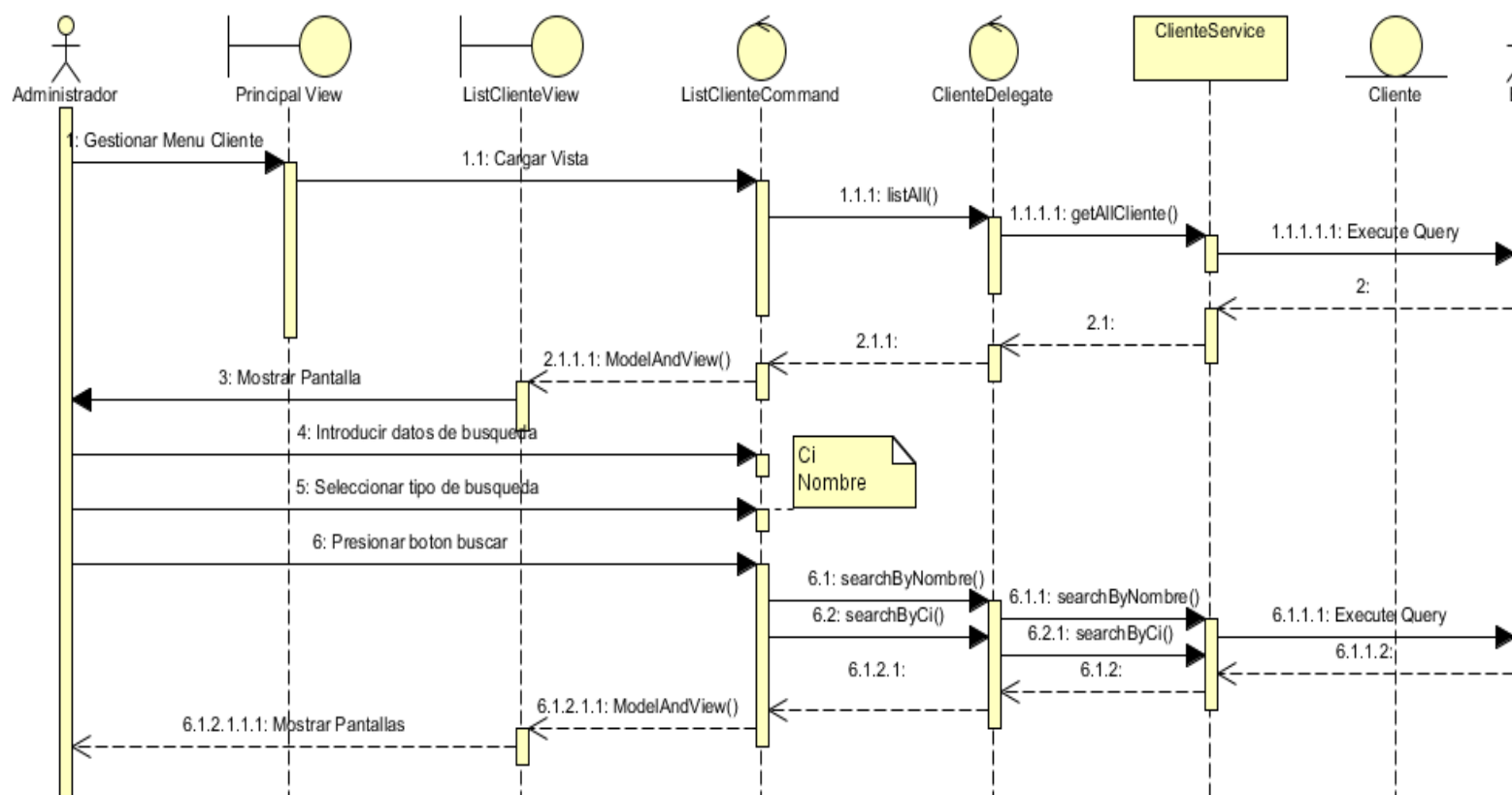


Figura 103 Diagrama de Secuencia: Caso de Uso Gestionar Cliente

2.1.2.2.3.14.2.1.7 Diagrama de Secuencia: Caso de Uso Adicionar Cliente

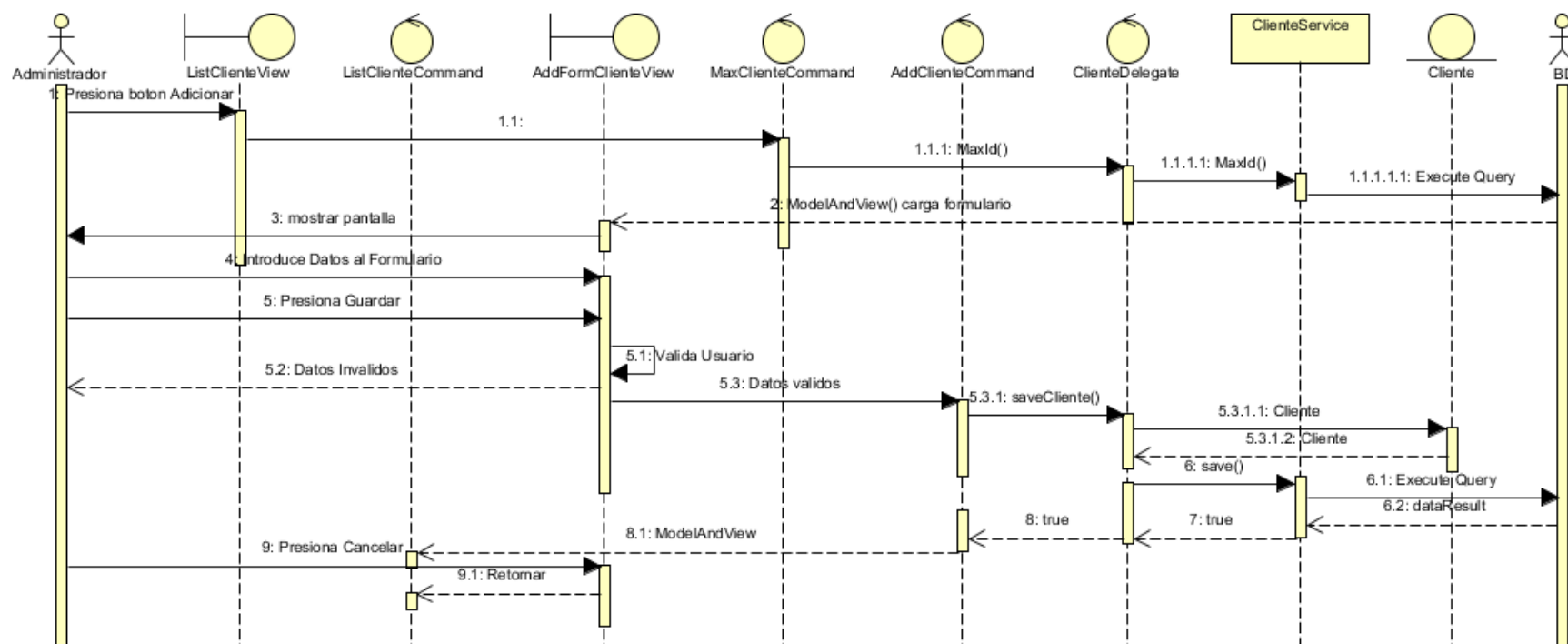


Figura 104 Diagrama de Secuencia: Caso de Uso Adicionar Cliente

2.1.2.2.3.14.2.1.8 Diagrama de Secuencia: Caso de Uso Eliminar Cliente

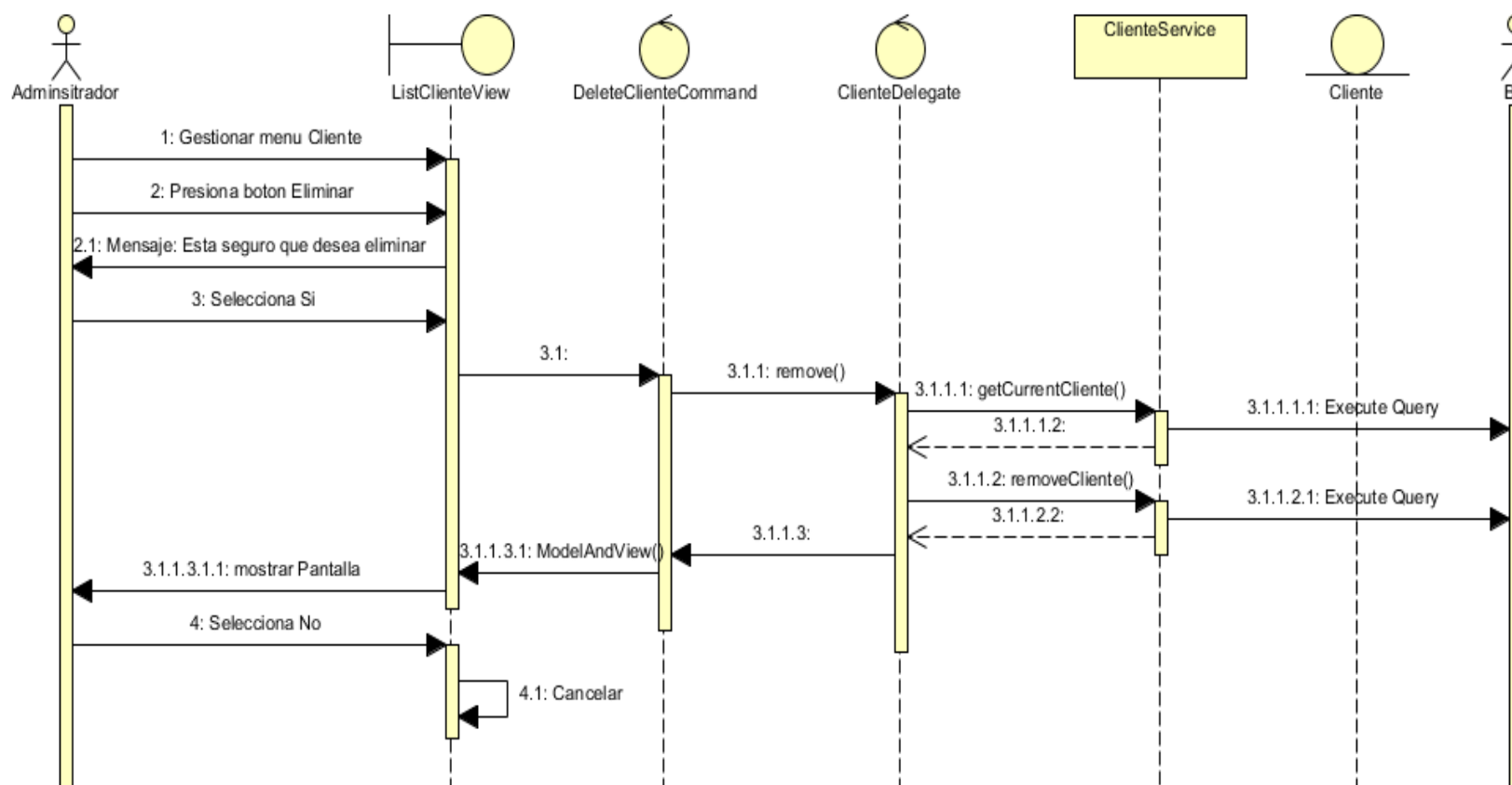


Figura 105 Diagrama de Secuencia: Caso de Uso Eliminar Cliente

2.1.2.2.3.14.2.1.9 Diagrama de Secuencia: Caso de Uso Modificar Cliente

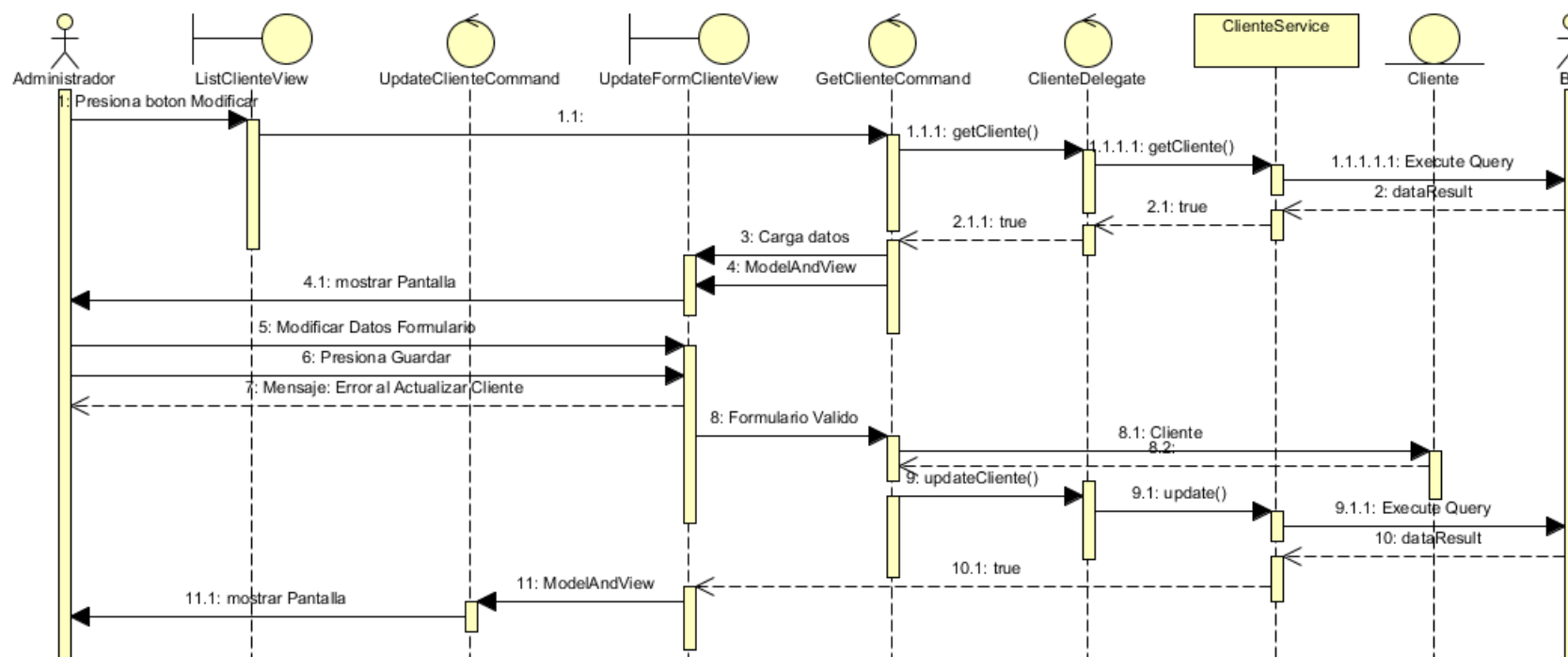


Figura 106 Diagrama de Secuencia: Caso de Uso Modificar Cliente

2.1.2.2.3.14.2.1.10 Diagrama de Secuencia: Caso de Uso Gestionar Egreso

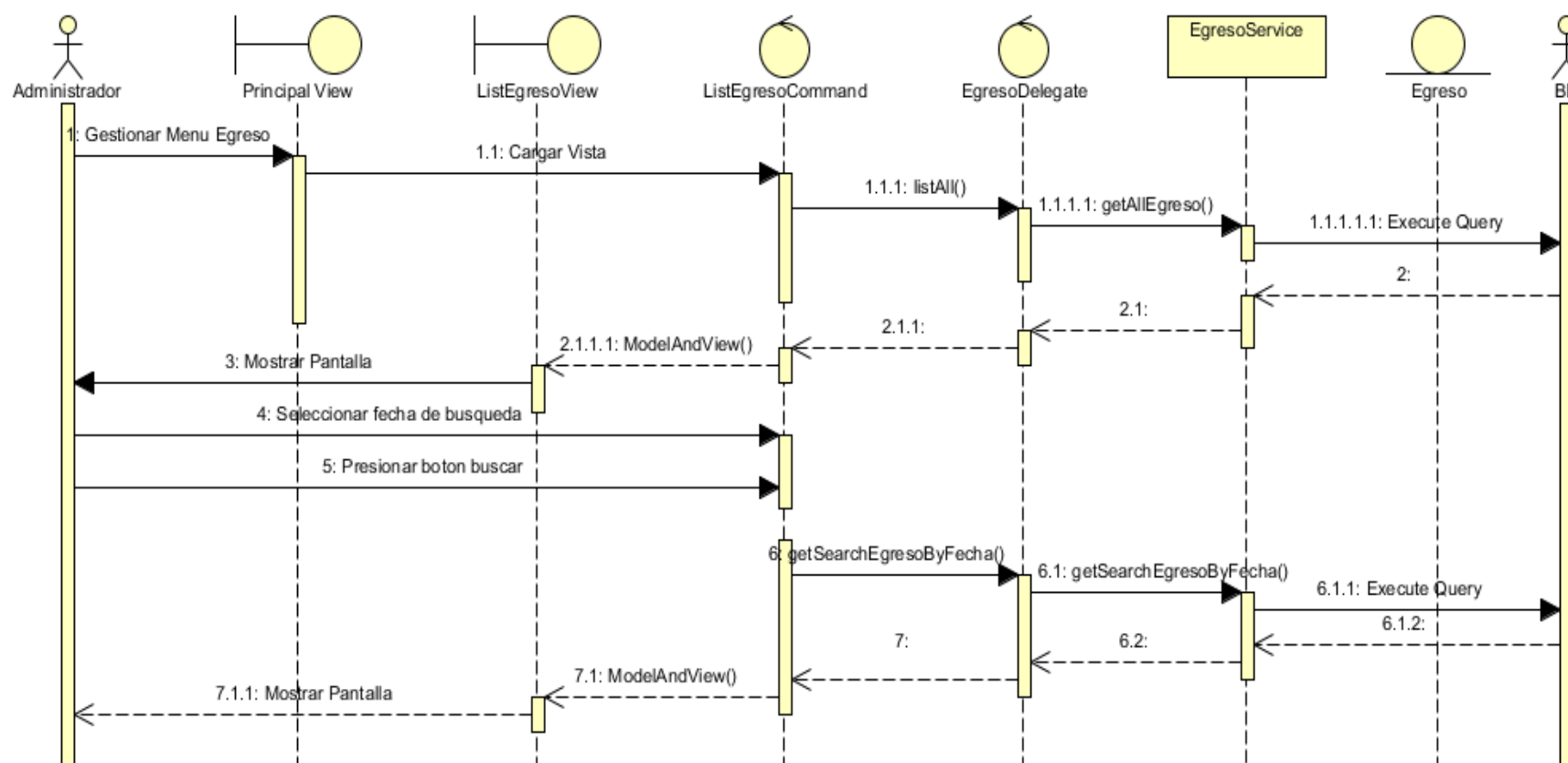


Figura 107 Diagrama de Secuencia: Caso de Uso Gestionar Egreso

2.1.2.2.3.14.2.1.11 Diagrama de Secuencia: Caso de Uso Adicionar Egreso

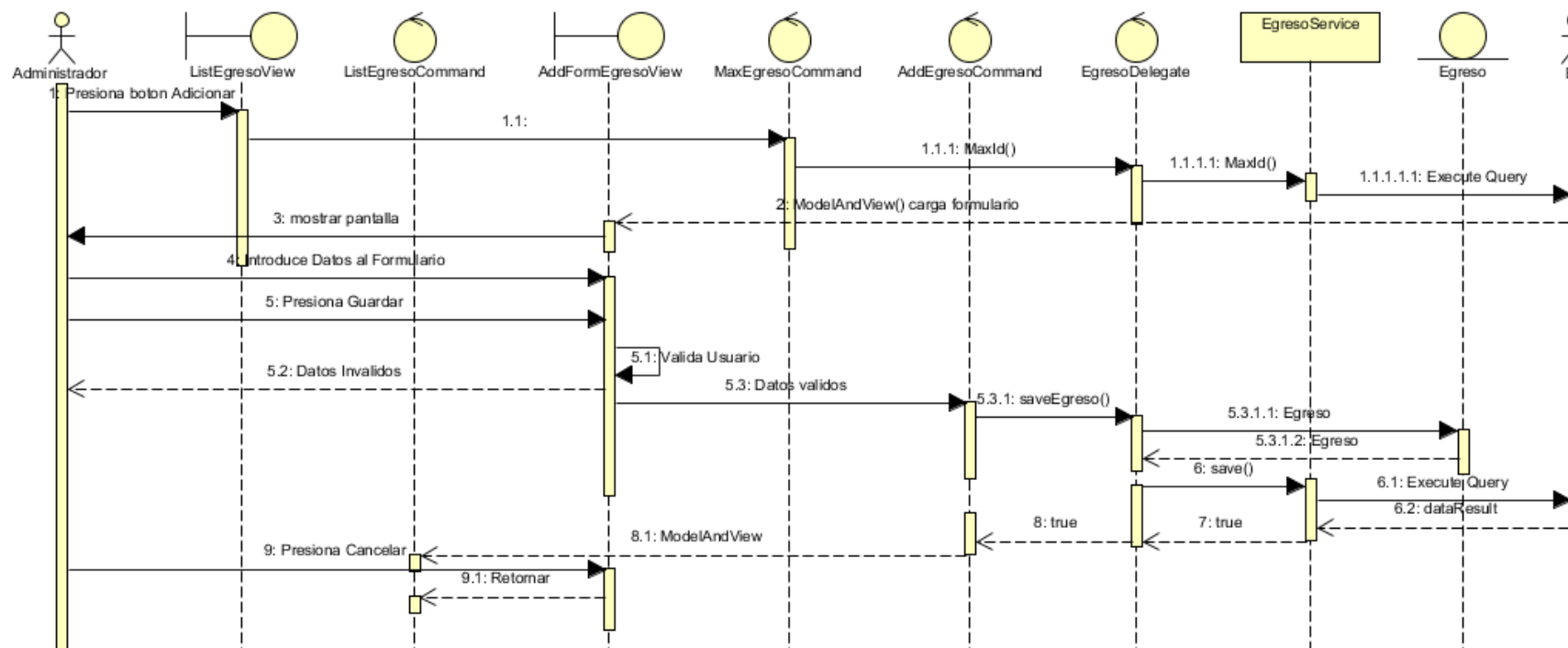


Figura 108 Diagrama de Secuencia: Caso de Uso Adicionar Egreso

2.1.2.2.3.14.2.1.12 Diagrama de Secuencia: Caso de Uso Eliminar Egreso

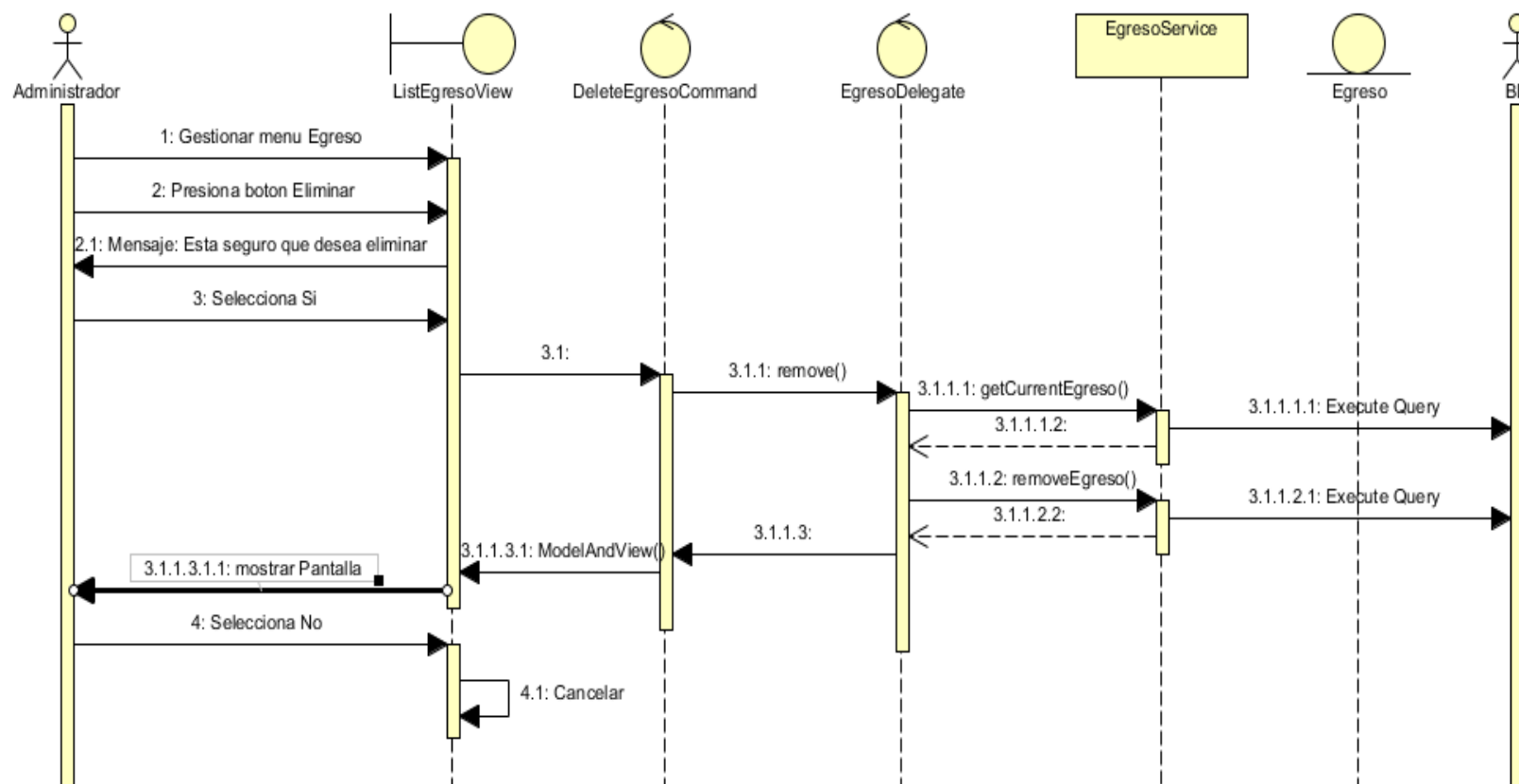


Figura 109 Diagrama de Secuencia: Caso de Uso Eliminar Egreso

2.1.2.2.3.14.2.1.13 Diagrama de Secuencia: Caso de Uso Modificar Egreso

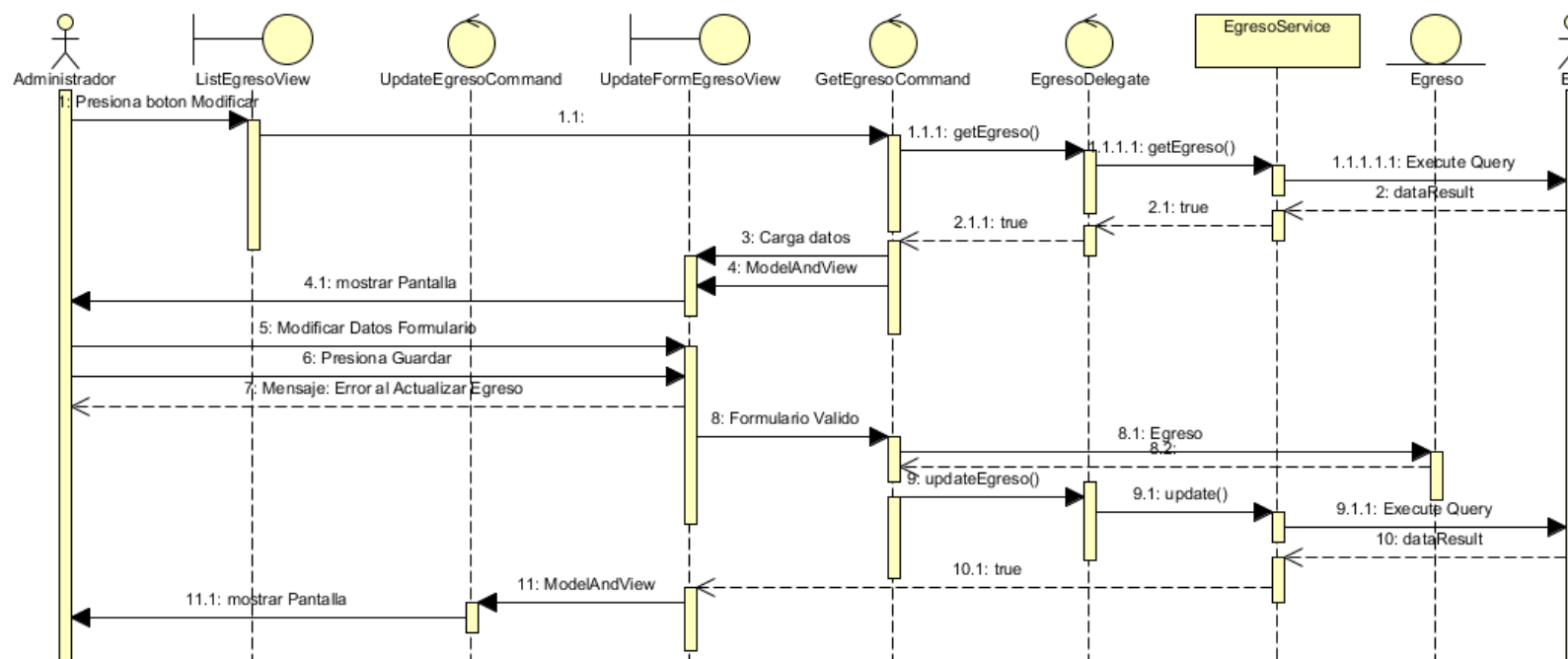


Figura 110 Diagrama de Secuencia: Caso de Uso Modificar Egreso

2.1.2.2.3.14.2.1.14 Diagrama de Secuencia: Caso de Uso Gestionar Ingreso

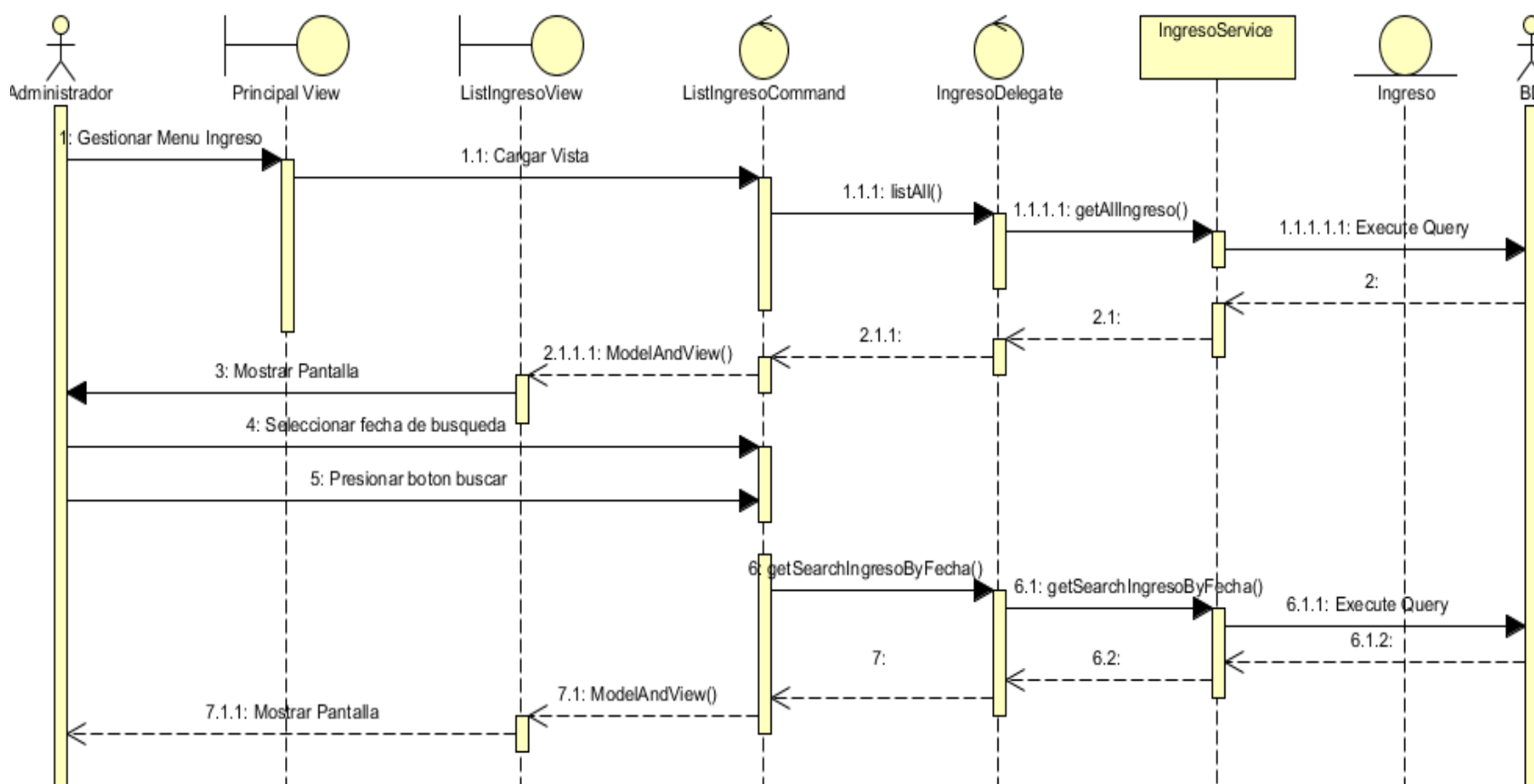


Figura 111 Diagrama de Secuencia: Caso de Uso Gestionar Ingreso

2.1.2.2.3.14.2.1.15 Diagrama de Secuencia: Caso de Uso Agregar Ingreso

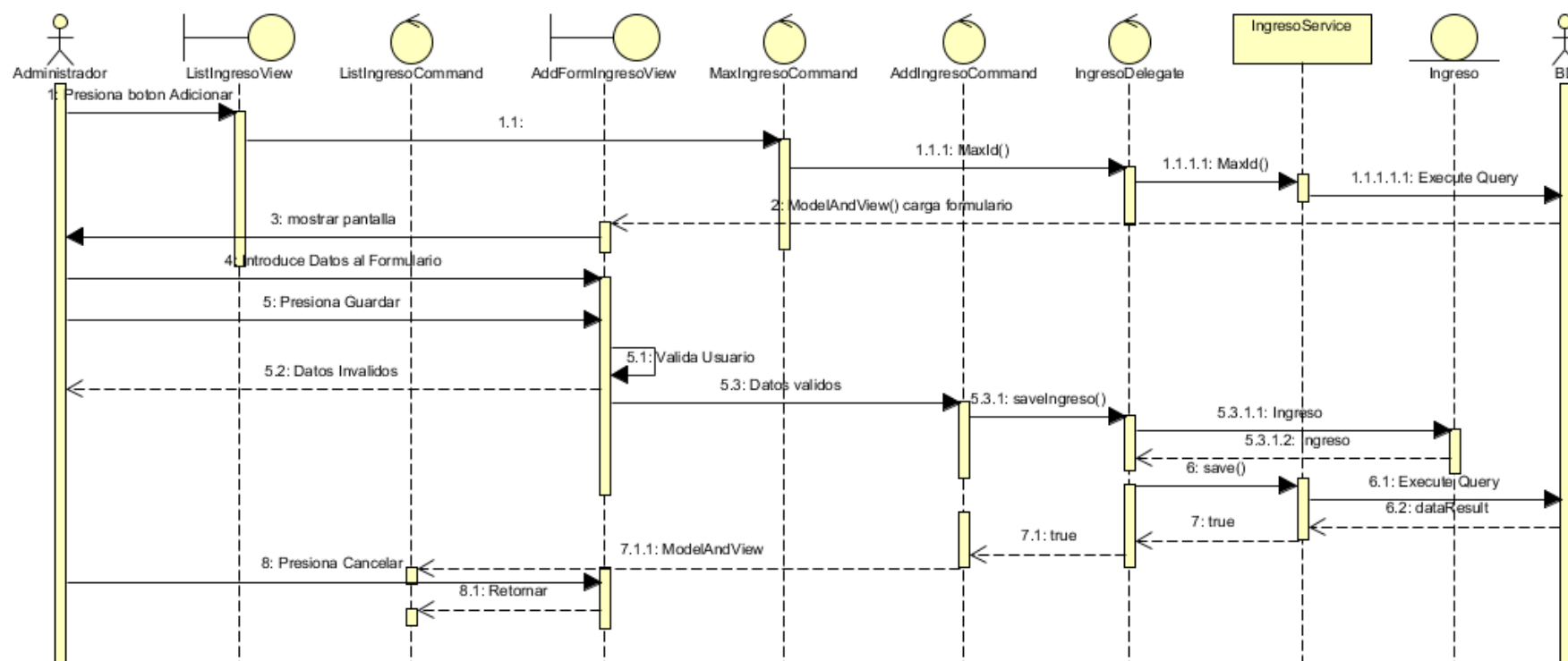


Figura 112 Diagrama de Secuencia: Caso de Uso Agregar Ingreso

2.1.2.2.3.14.2.1.16 Diagrama de Secuencia: Caso de Uso Eliminar Ingreso

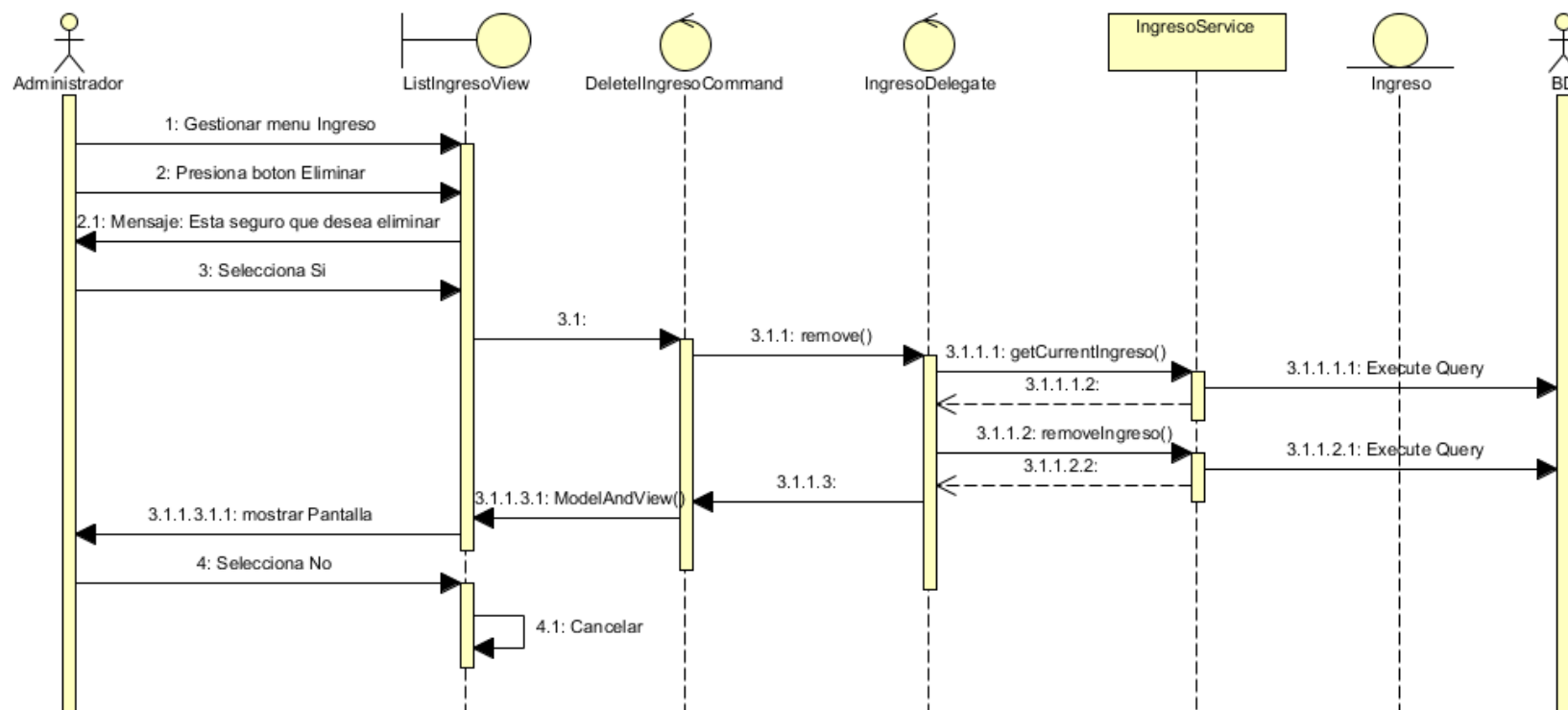


Figura 113 Diagrama de Secuencia: Caso de Uso Eliminar Ingreso

2.1.2.2.3.14.2.1.17 Diagrama de Secuencia: Caso de Uso Modificar Ingreso

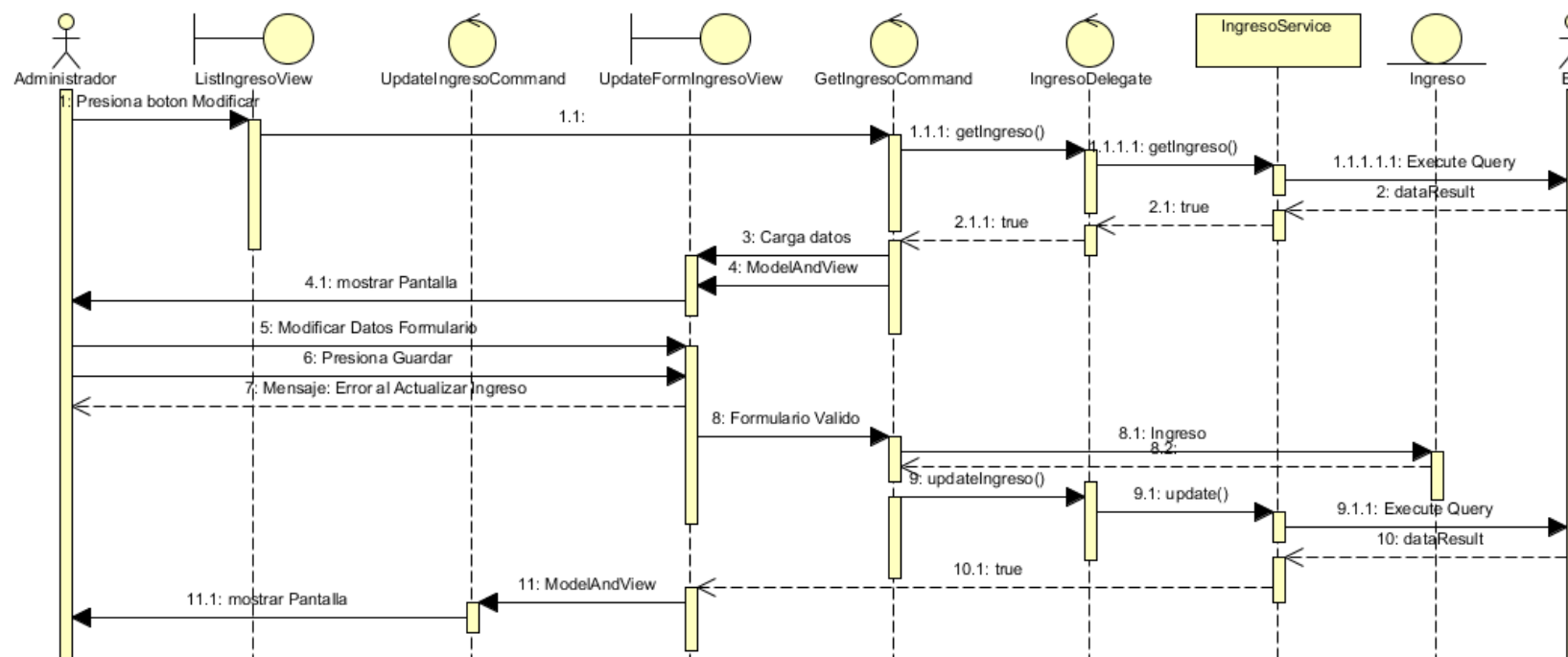


Figura 114 Diagrama de Secuencia: Caso de Uso Modificar Ingreso

2.1.2.2.3.14.2.1.18 Diagrama de Secuencia: Caso de Uso Gestionar Noticia

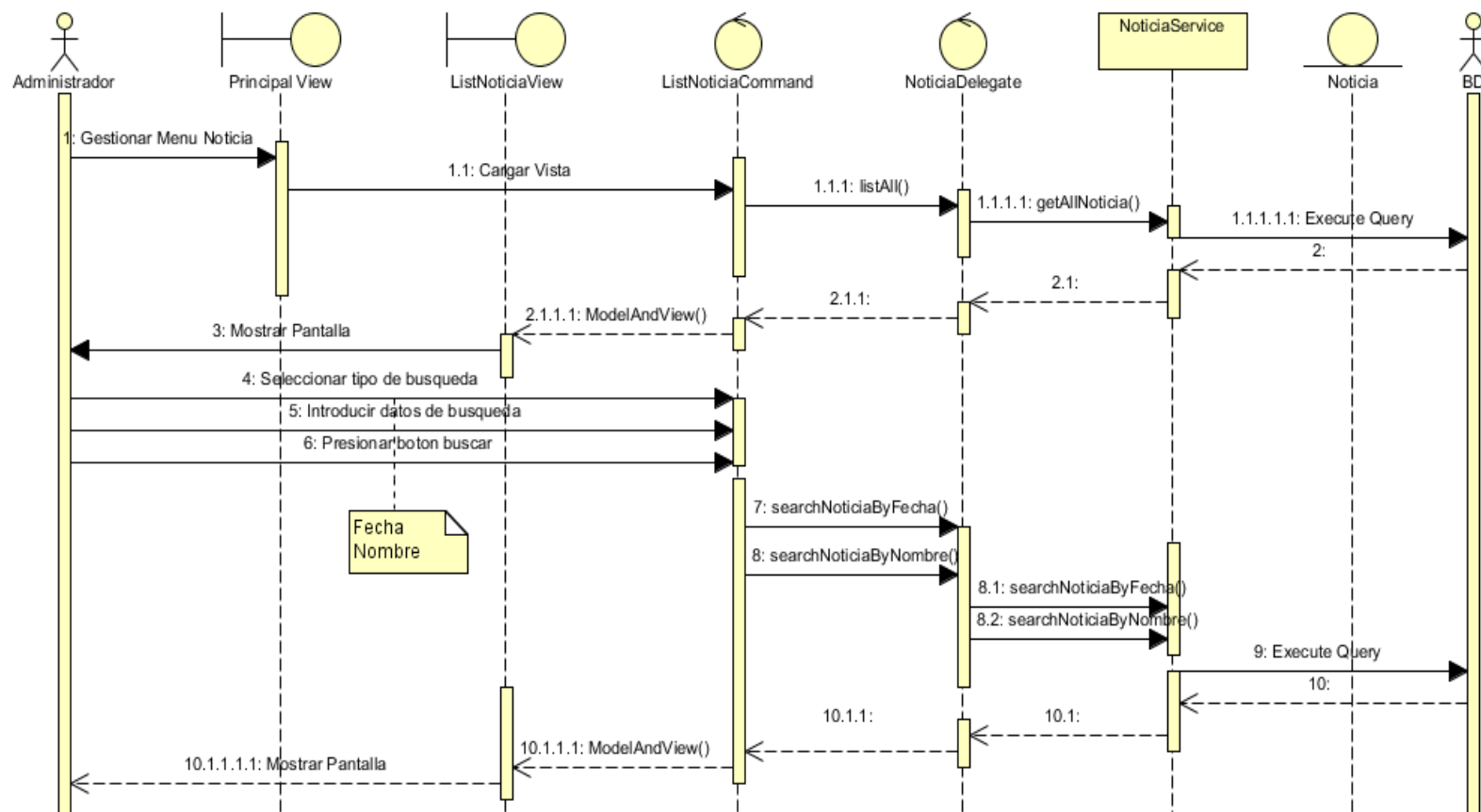


Figura 115 Diagrama de Secuencia: Caso de Uso Gestionar Noticia

2.1.2.2.3.14.2.1.19 Diagrama de Secuencia: Caso de Uso Adicionar Noticia

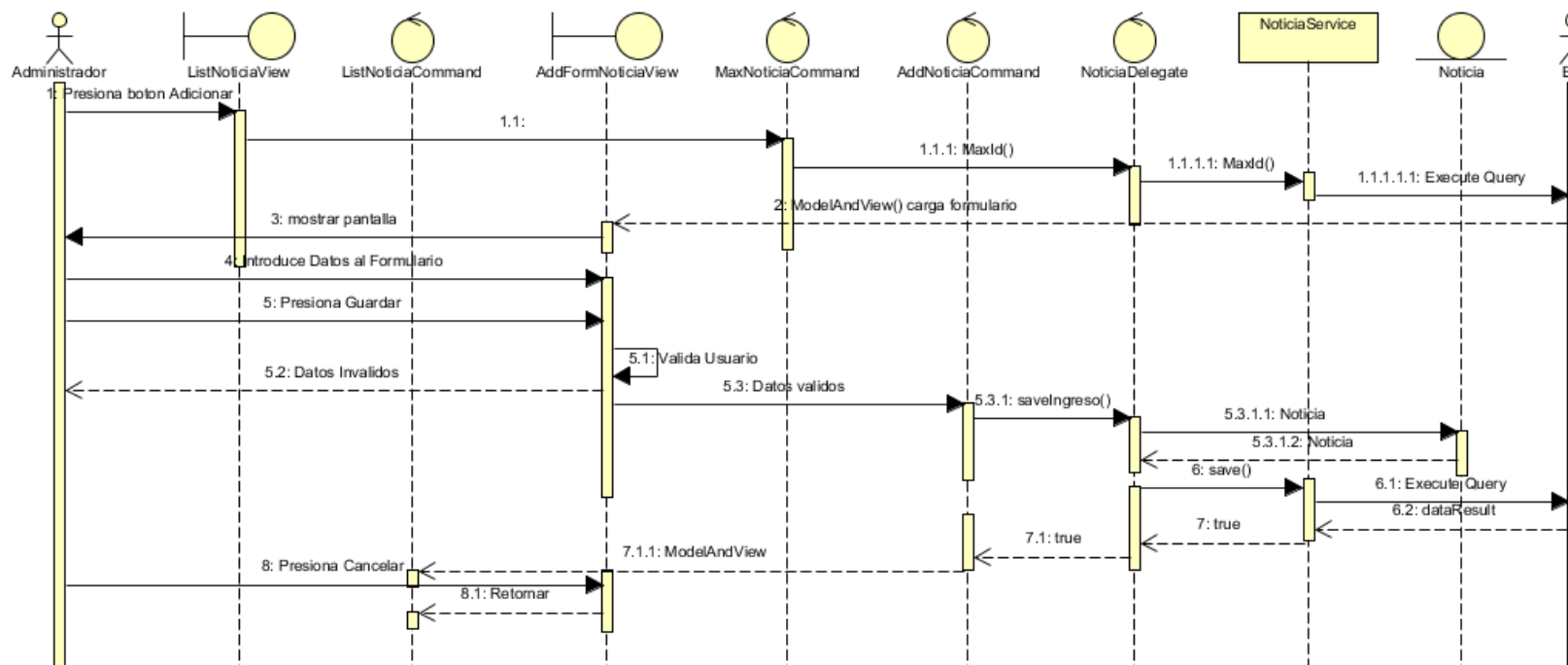


Figura 116 Diagrama de Secuencia: Caso de Uso Adicionar Noticia

2.1.2.2.3.14.2.1.20 Diagrama de Secuencia: Caso de Uso Eliminar Noticia

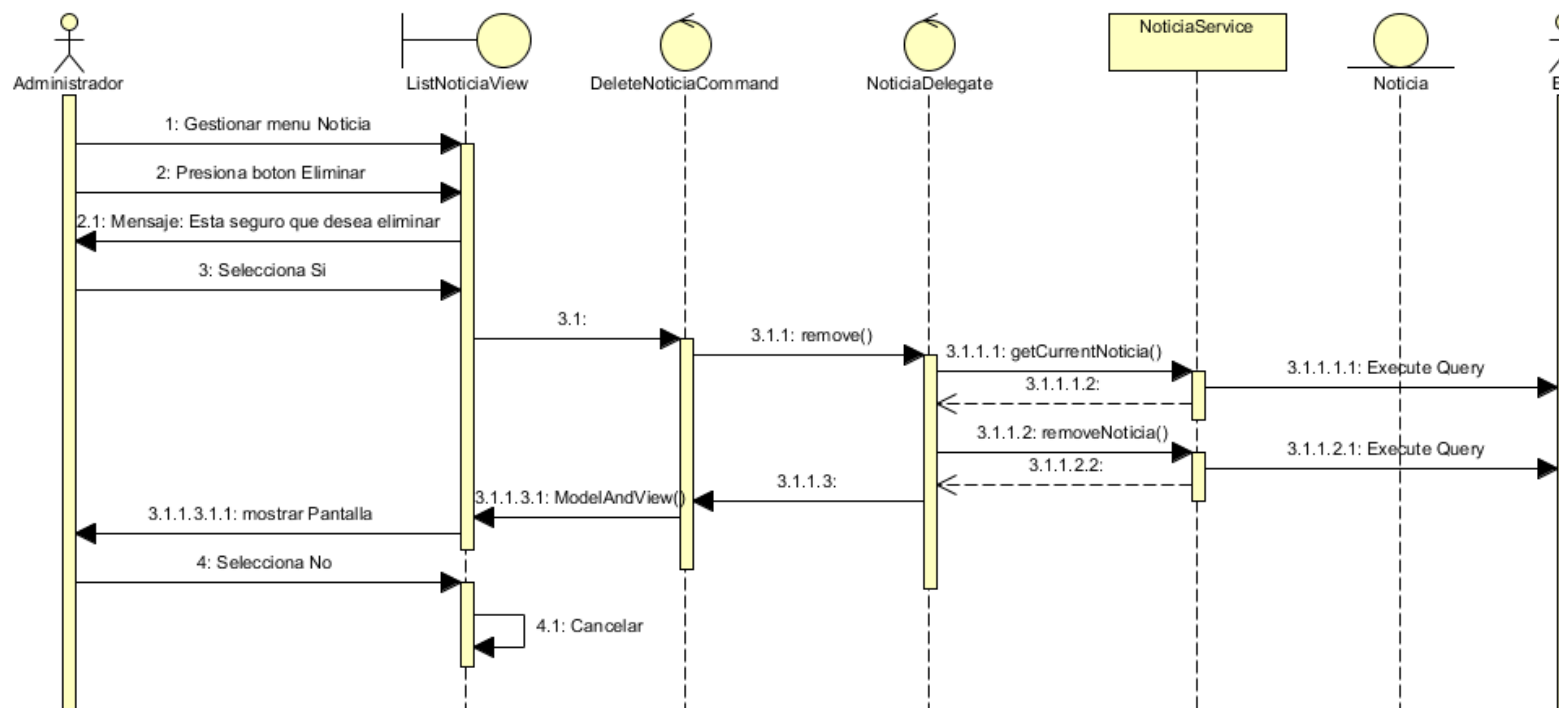


Figura 117 Diagrama de Secuencia: Caso de Uso Eliminar Noticia

2.1.2.2.3.14.2.1.21 Diagrama de Secuencia: Caso de Uso Modificar Noticia

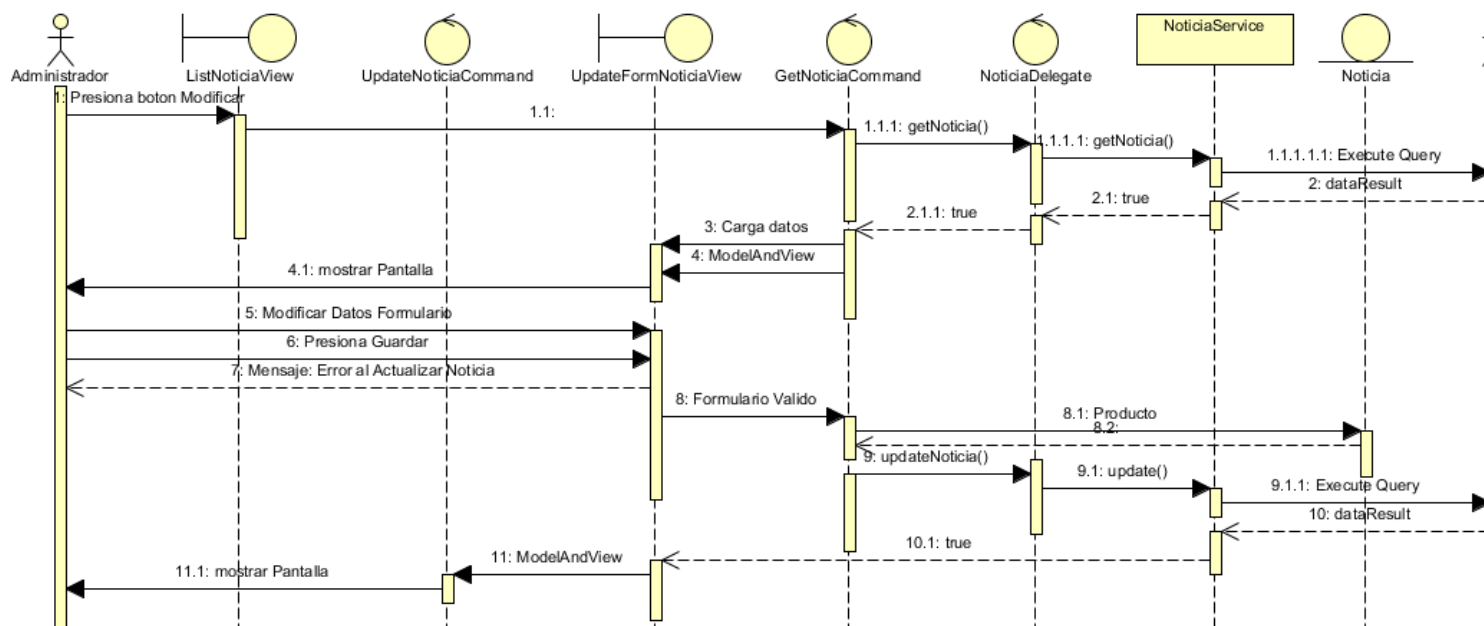


Figura 118 Diagrama de Secuencia: Caso de Uso Modificar Noticia

2.1.2.2.3.14.2.1.22 Diagrama de Secuencia: Caso de Uso Dar Baja Noticia

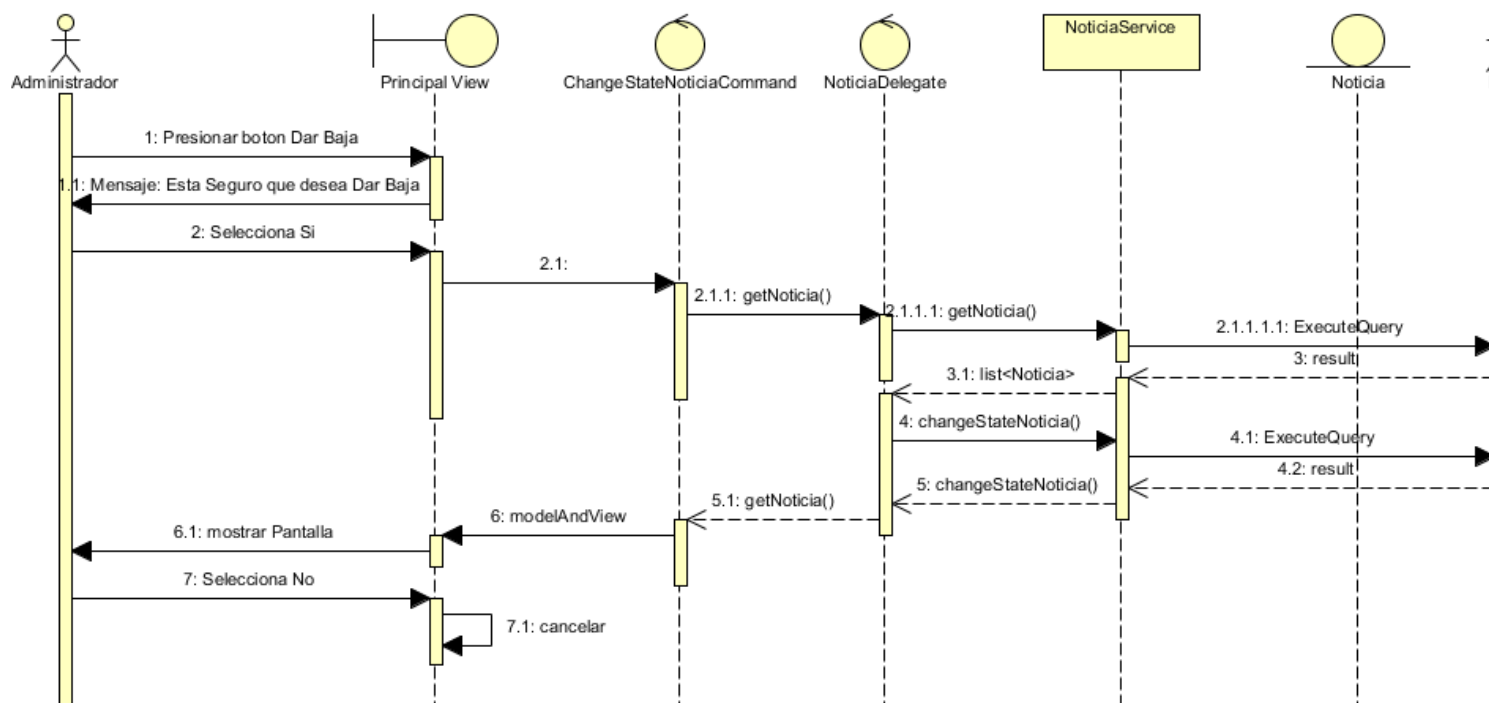


Figura 119 Diagrama de Secuencia: Caso de Uso Dar Baja Noticia

2.1.2.2.3.14.2.1.23 Diagrama de Secuencia: Caso de Uso Gestionar Periodo

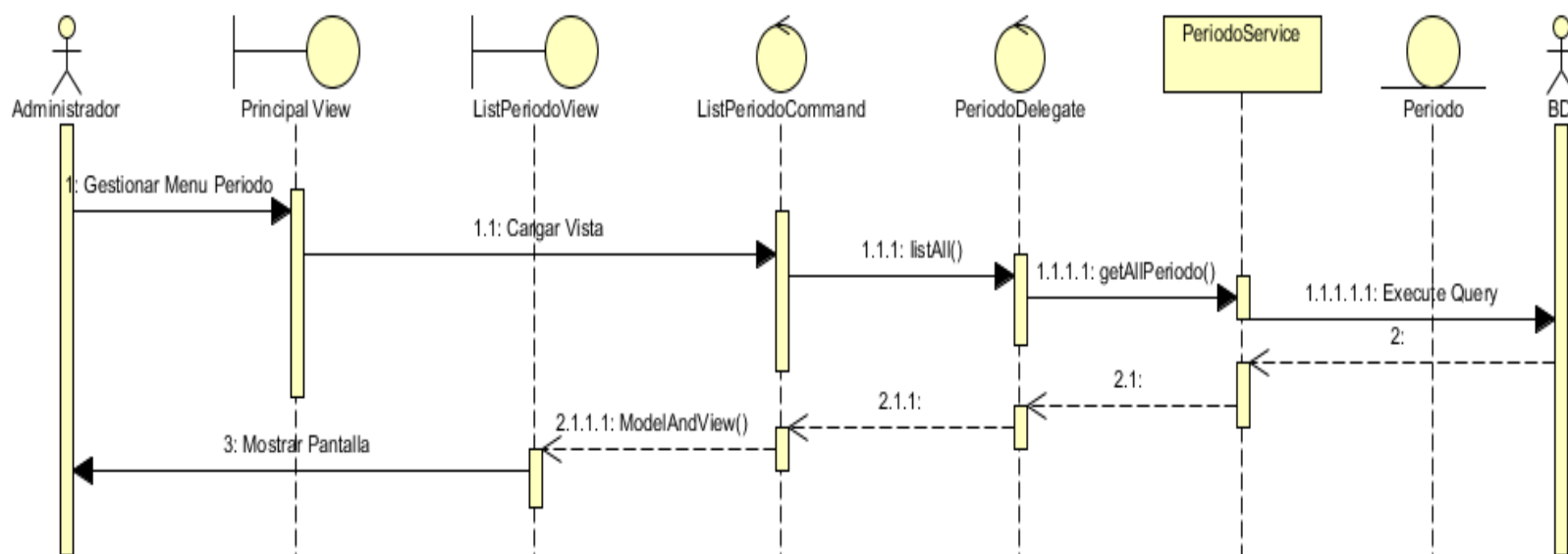


Figura 120 Diagrama de Secuencia: Caso de Uso Gestionar Periodo

2.1.2.2.3.14.2.1.24 Diagrama de Secuencia: Caso de Uso Adicionar Periodo

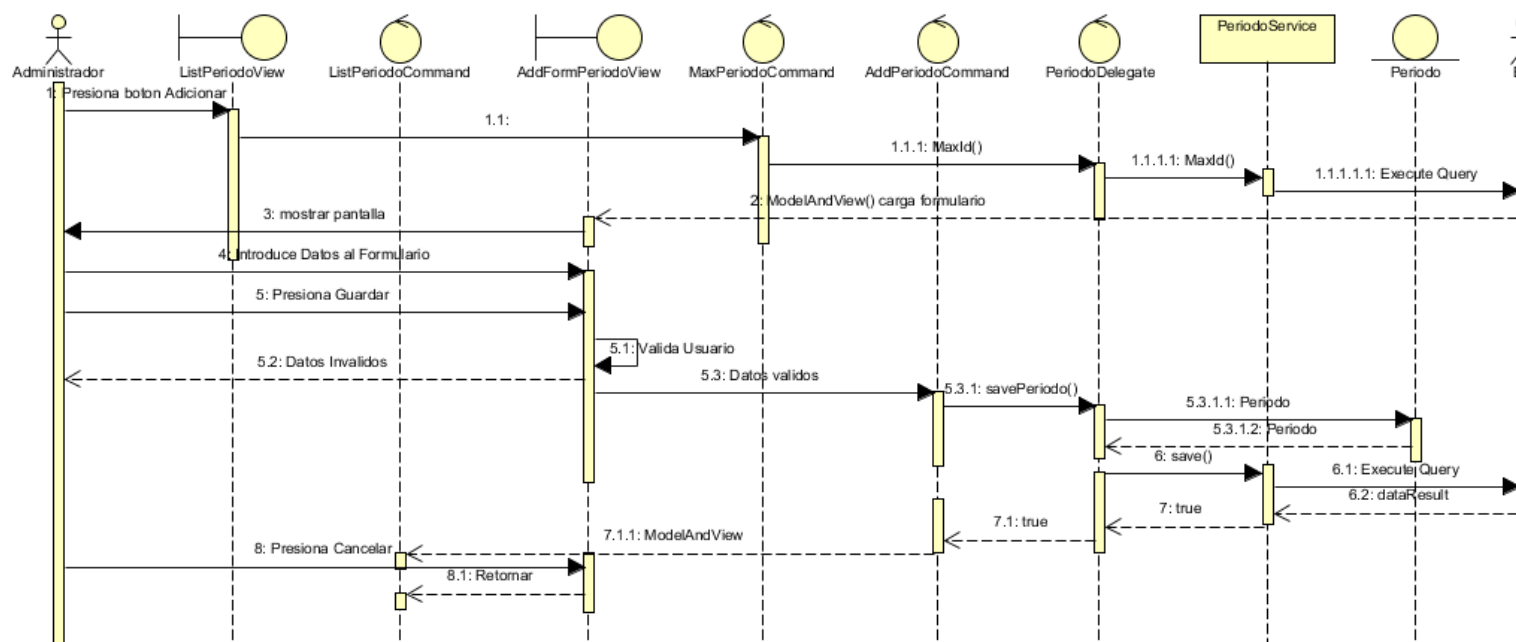


Figura 121 Diagrama de Secuencia: Caso de Uso Adicionar Periodo

2.1.2.2.3.14.2.1.25 Diagrama de Secuencia: Caso de Uso Eliminar Periodo

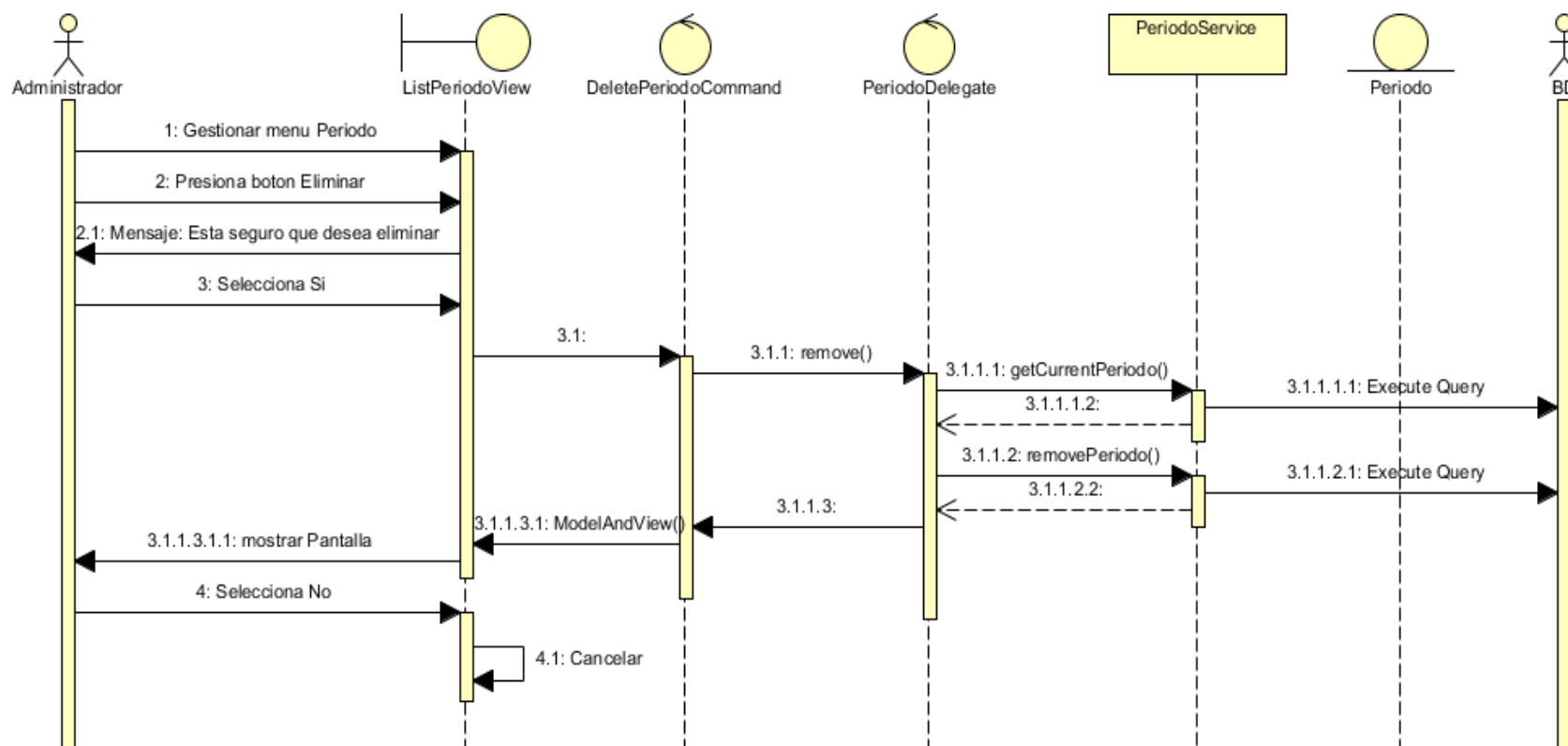


Figura 122 Diagrama de Secuencia: Caso de Uso Eliminar Periodo

2.1.2.2.3.14.2.1.26 Diagrama de Secuencia: Caso de Uso Modificar Periodo

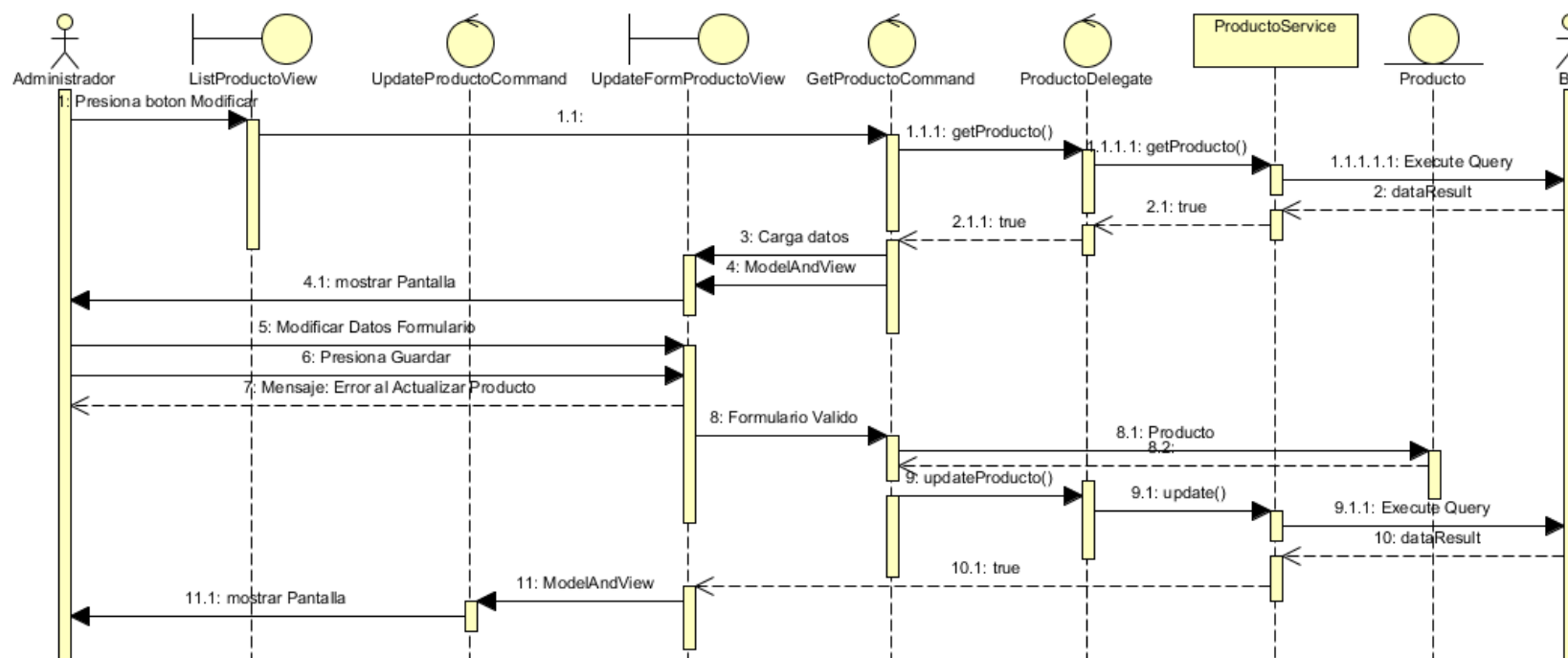


Figura 123 Diagrama de Secuencia: Caso de Uso Modificar Periodo

2.1.2.2.3.14.2.1.27 Diagrama de Secuencia: Caso de Uso Gestionar Producto

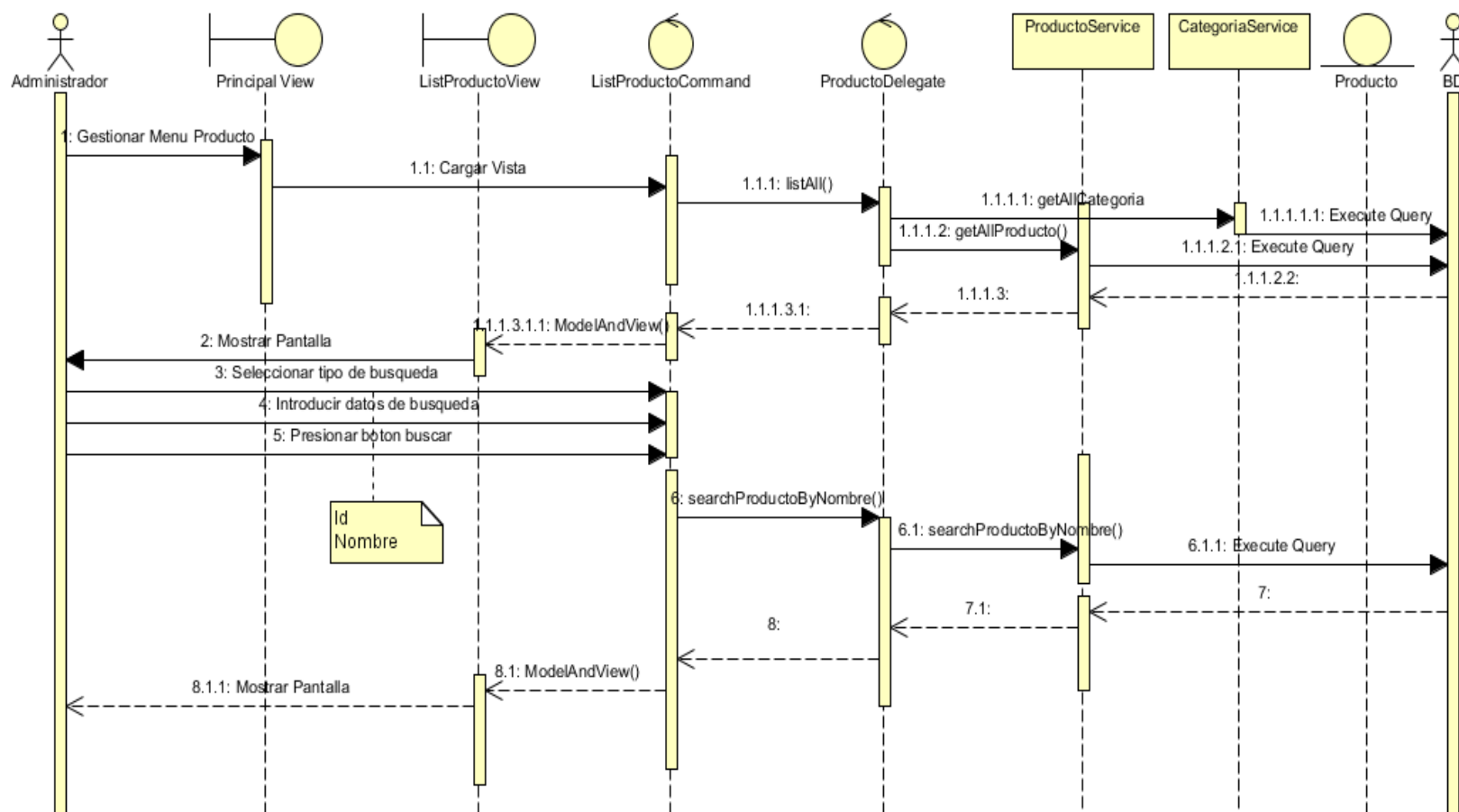


Figura 124 Diagrama de Secuencia: Caso de Uso Gestionar Producto

2.1.2.2.3.14.2.1.28 Diagrama de Secuencia: Caso de Uso Adicionar Producto

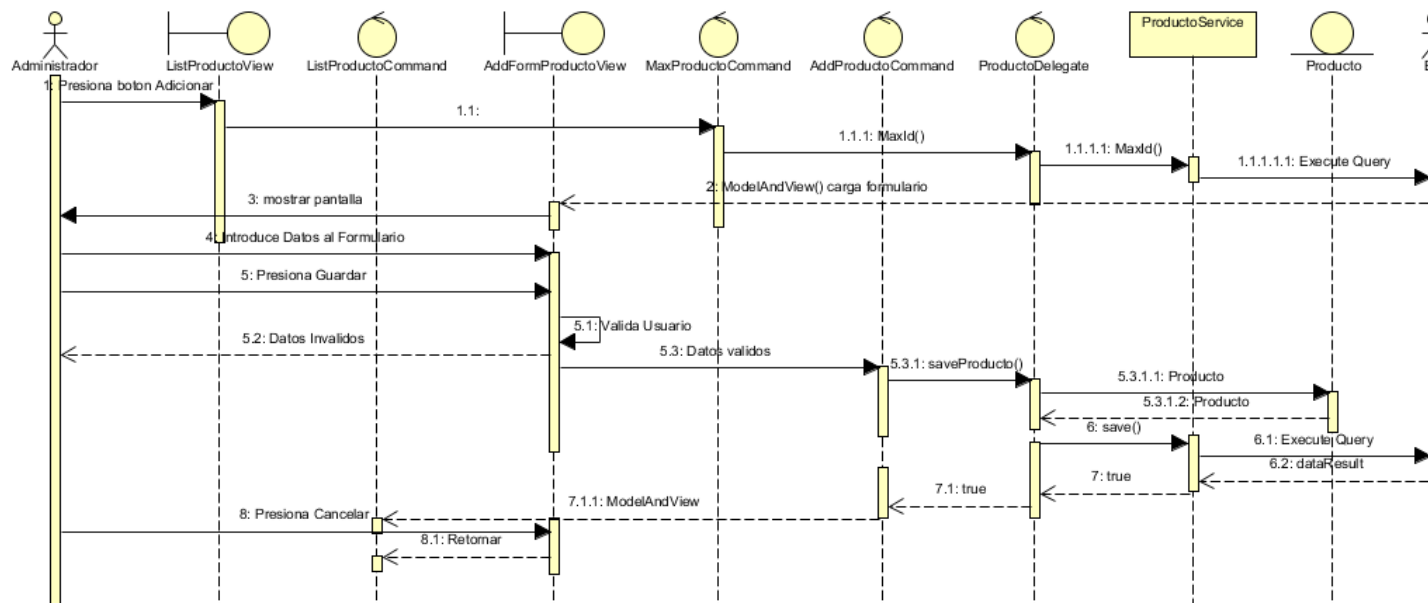


Figura 125 Diagrama de Secuencia: Caso de Uso Adicionar Producto

2.1.2.2.3.14.2.1.29 Diagrama de Secuencia: Caso de Uso Dar Baja Producto

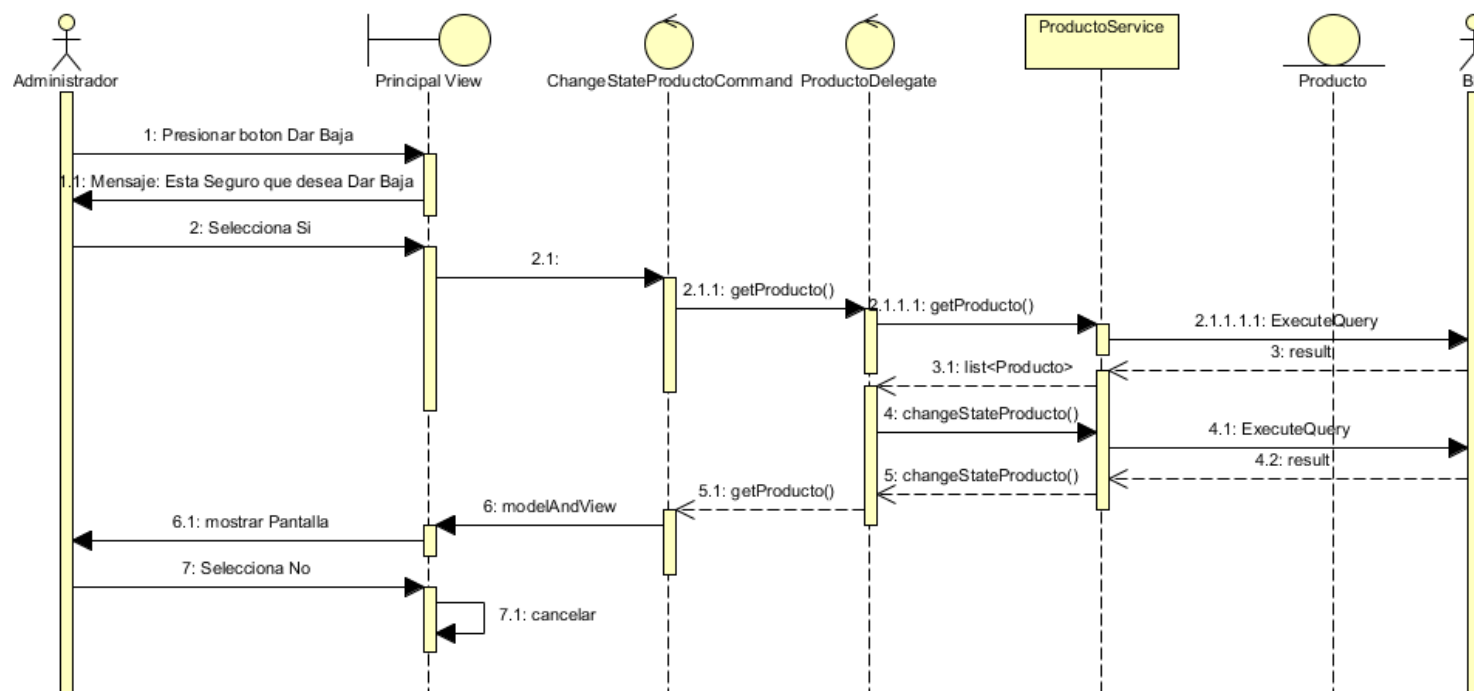


Figura 126 Diagrama de Secuencia: Caso de Uso Dar Baja Producto

2.1.2.2.3.14.2.1.30 Diagrama de Secuencia: Caso de Uso Modificar Producto

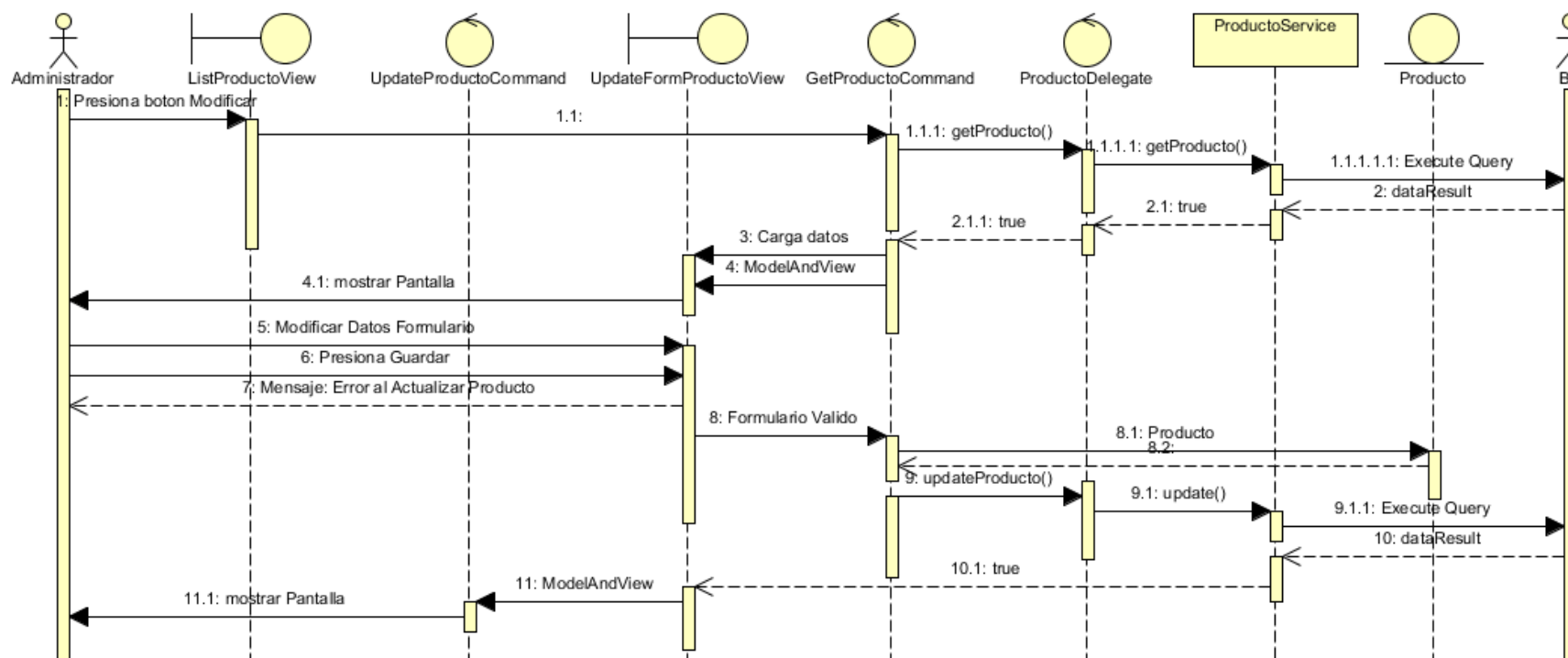


Figura 127 Diagrama de Secuencia: Caso de Uso Modificar Producto

2.1.2.2.3.14.2.1.31 Diagrama de Secuencia: Caso de Uso Gestionar Rol

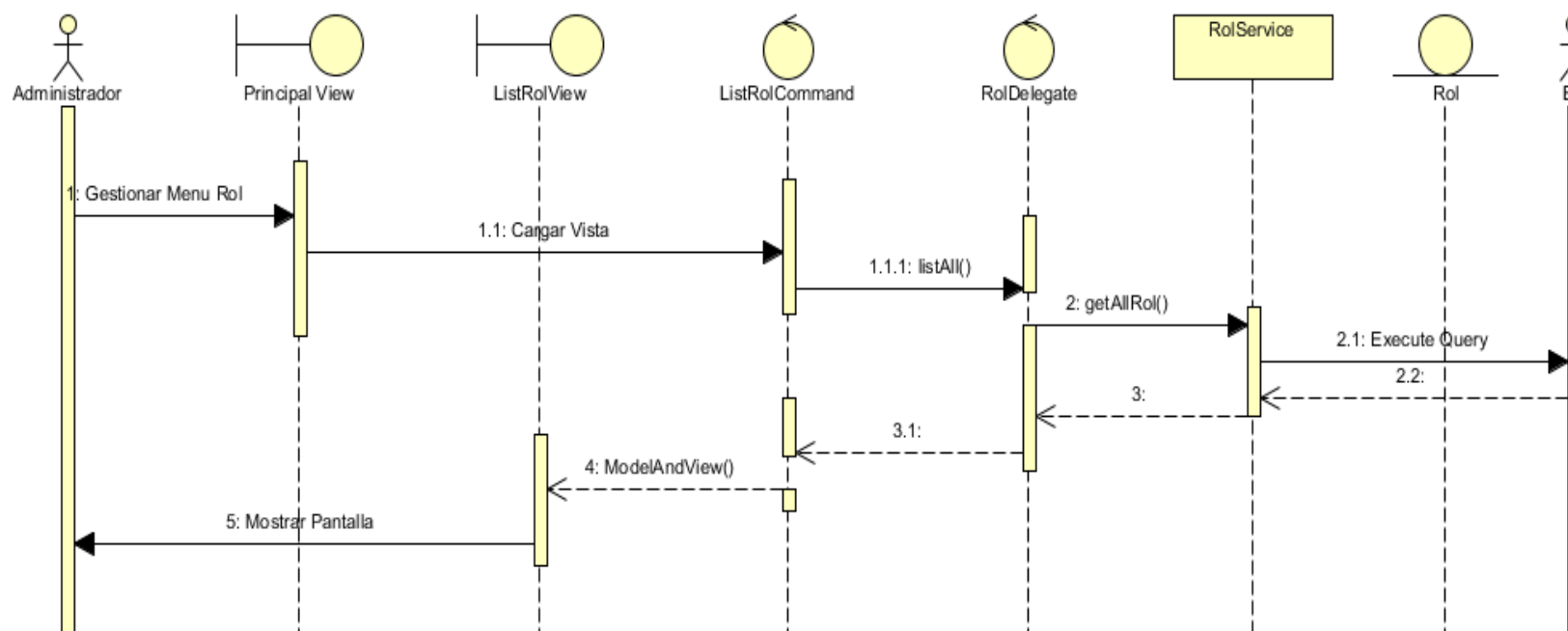


Figura 128 Diagrama de Secuencia: Caso de Uso Gestionar Rol

2.1.2.2.3.14.2.1.32 Diagrama de Secuencia: Caso de Uso Adicionar Rol

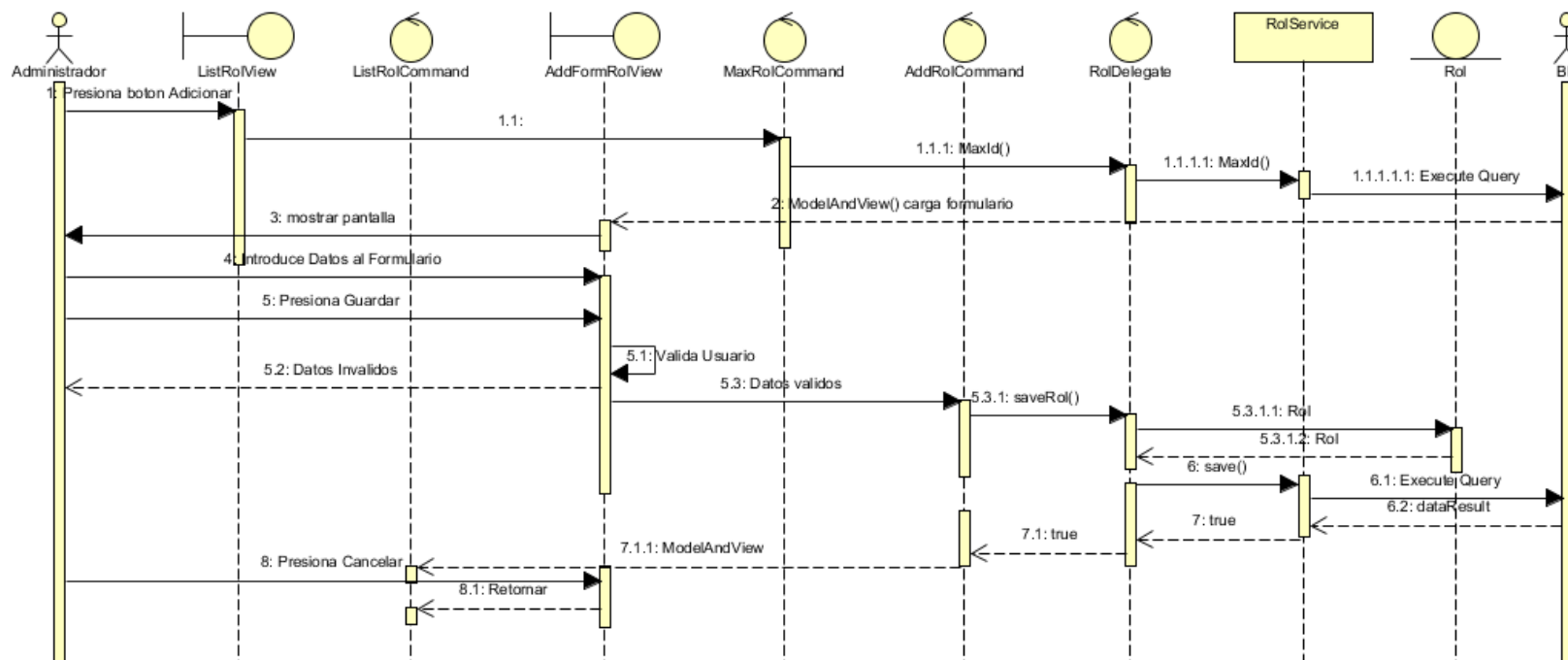


Figura 129 Diagrama de Secuencia: Caso de Uso Adicionar Rol

2.1.2.2.3.14.2.1.33 Diagrama de Secuencia: Caso de Uso Eliminar Rol

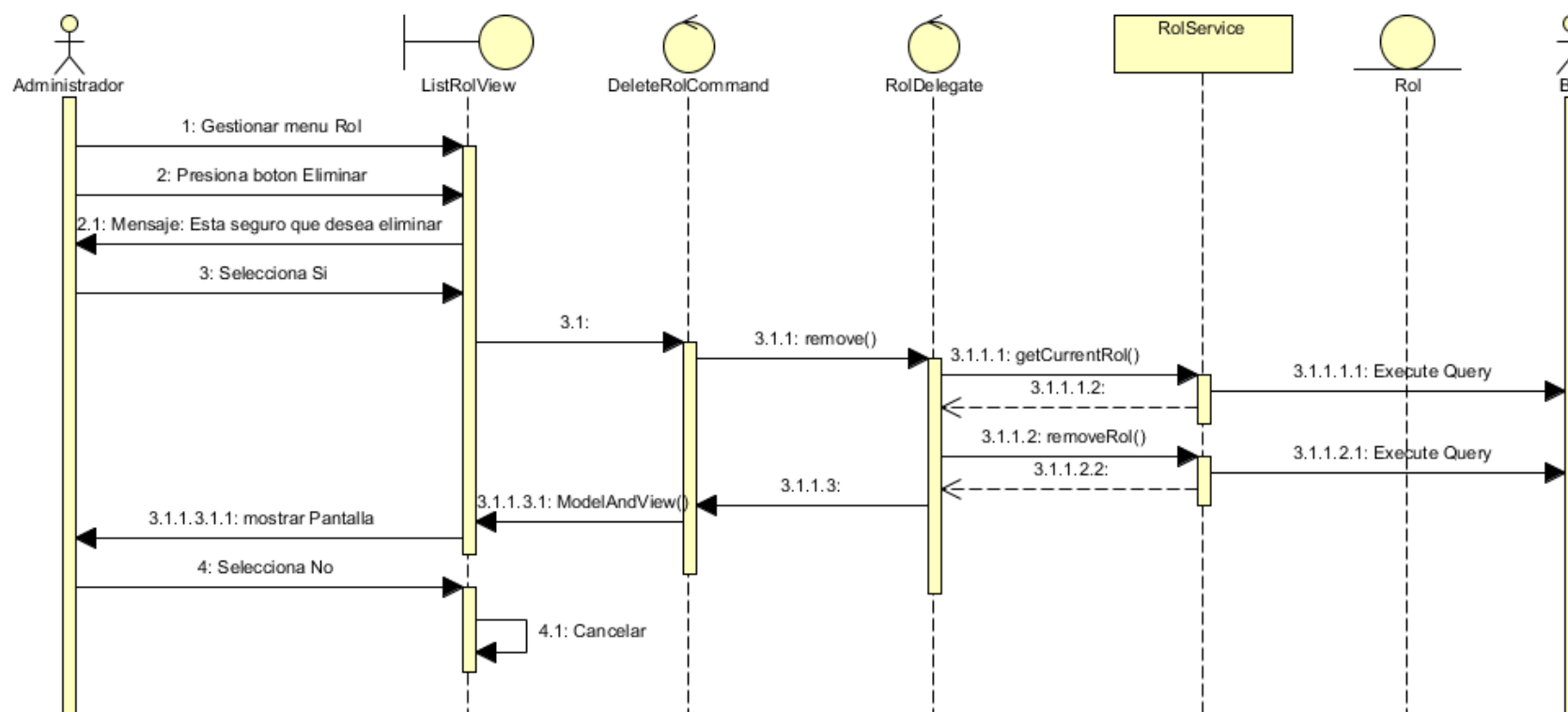


Figura 130 Diagrama de Secuencia: Caso de Uso Eliminar Rol

2.1.2.2.3.14.2.1.34 Diagrama de Secuencia: Caso de Uso Modificar Rol

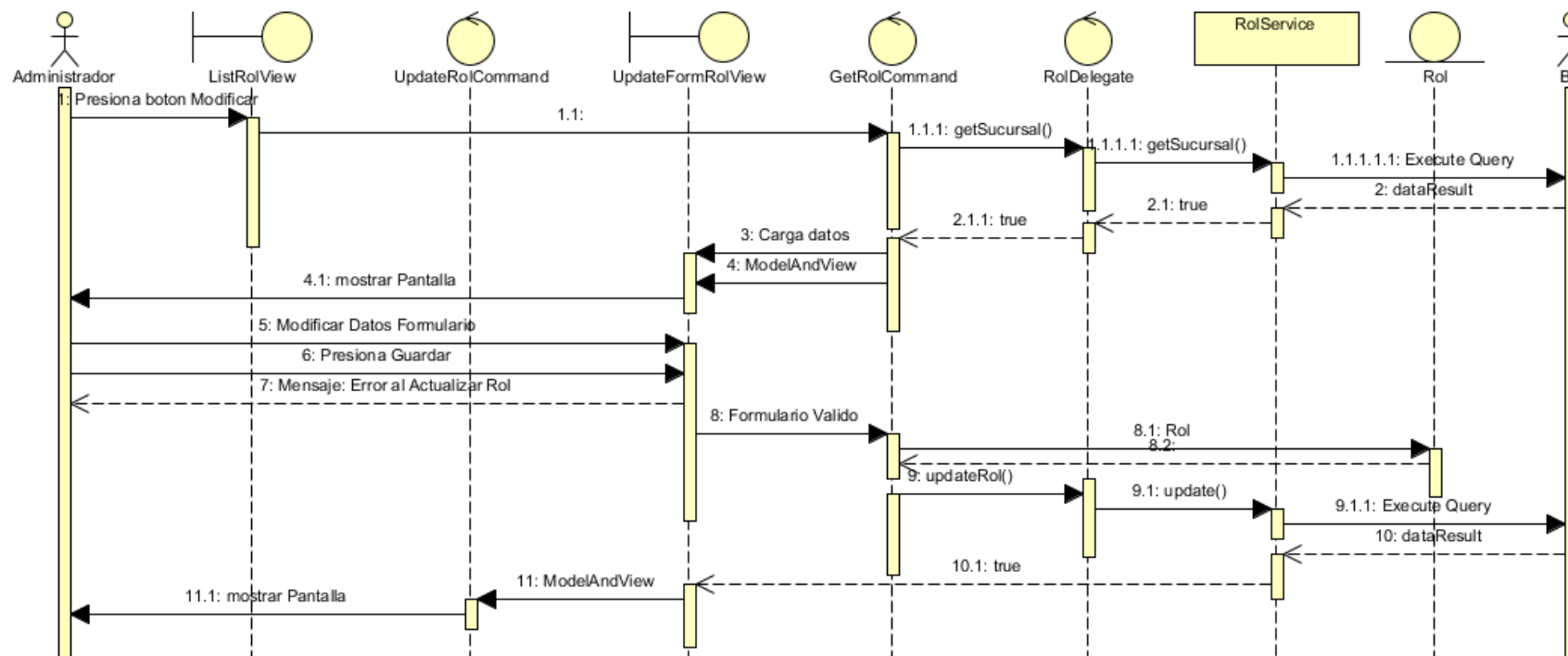


Figura 131 Diagrama de Secuencia: Caso de Uso Modificar Rol

2.1.2.2.3.14.2.1.35 Diagrama de Secuencia: Caso de Uso Gestionar Sucursal

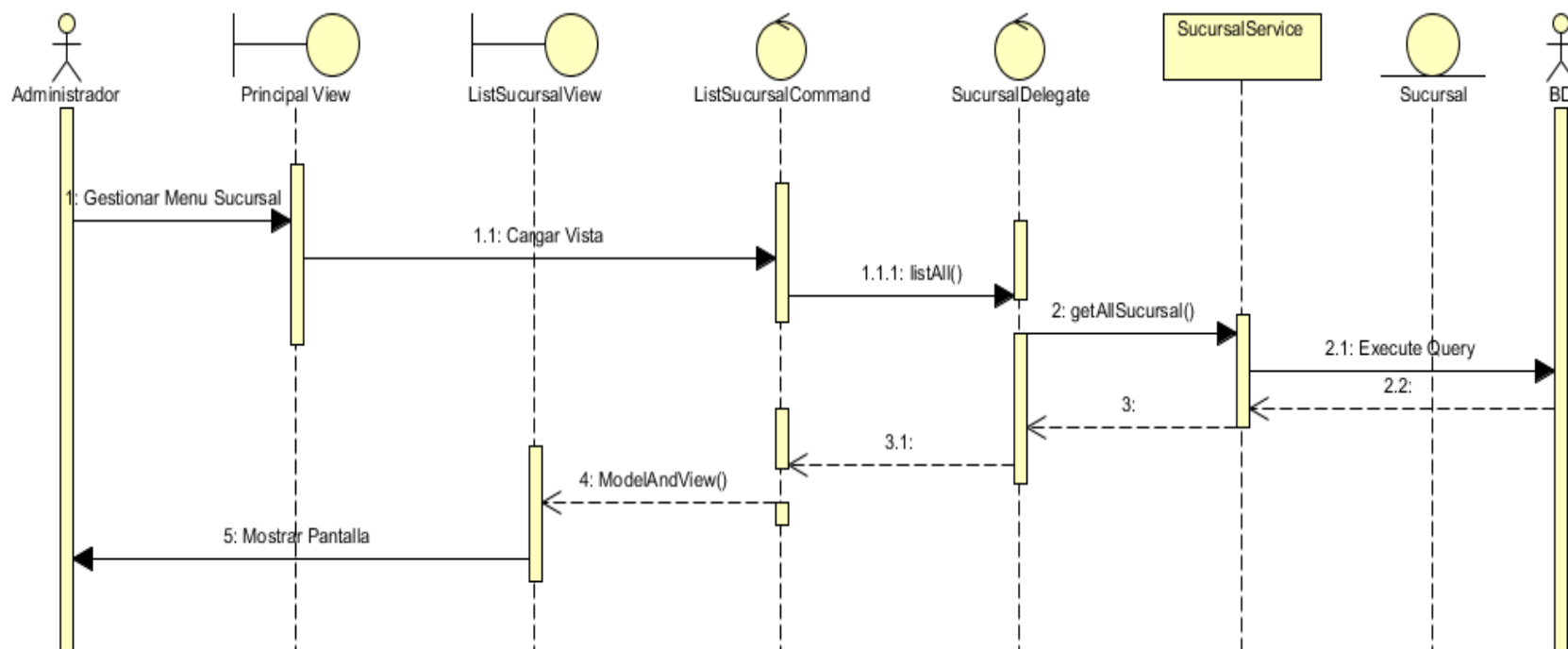


Figura 132 Diagrama de Secuencia: Caso de Uso Gestionar Sucursal

2.1.2.2.3.14.2.1.36 Diagrama de Secuencia: Caso de Uso Adicionar Sucursal

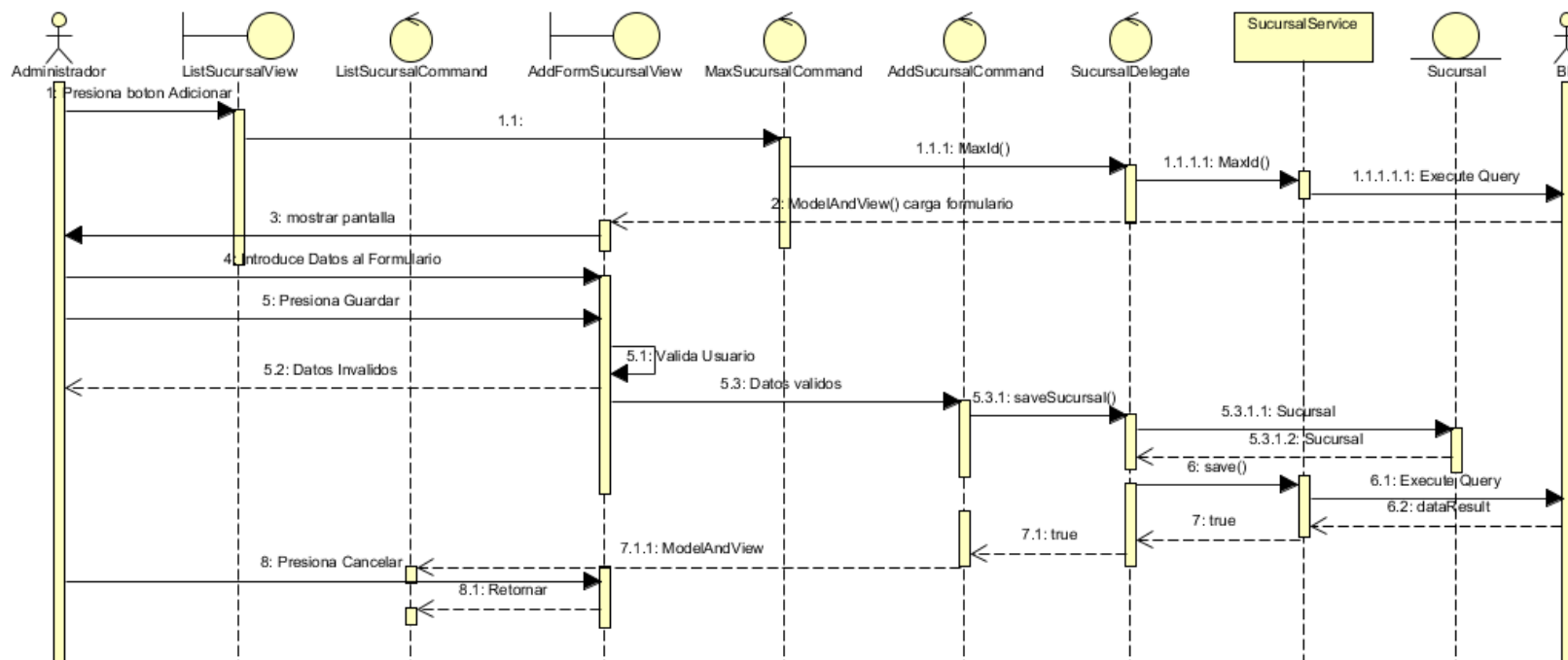


Figura 133 Diagrama de Secuencia: Caso de Uso Adicionar Sucursal

2.1.2.2.3.14.2.1.37 Diagrama de Secuencia: Caso de Uso Dar Baja Sucursal

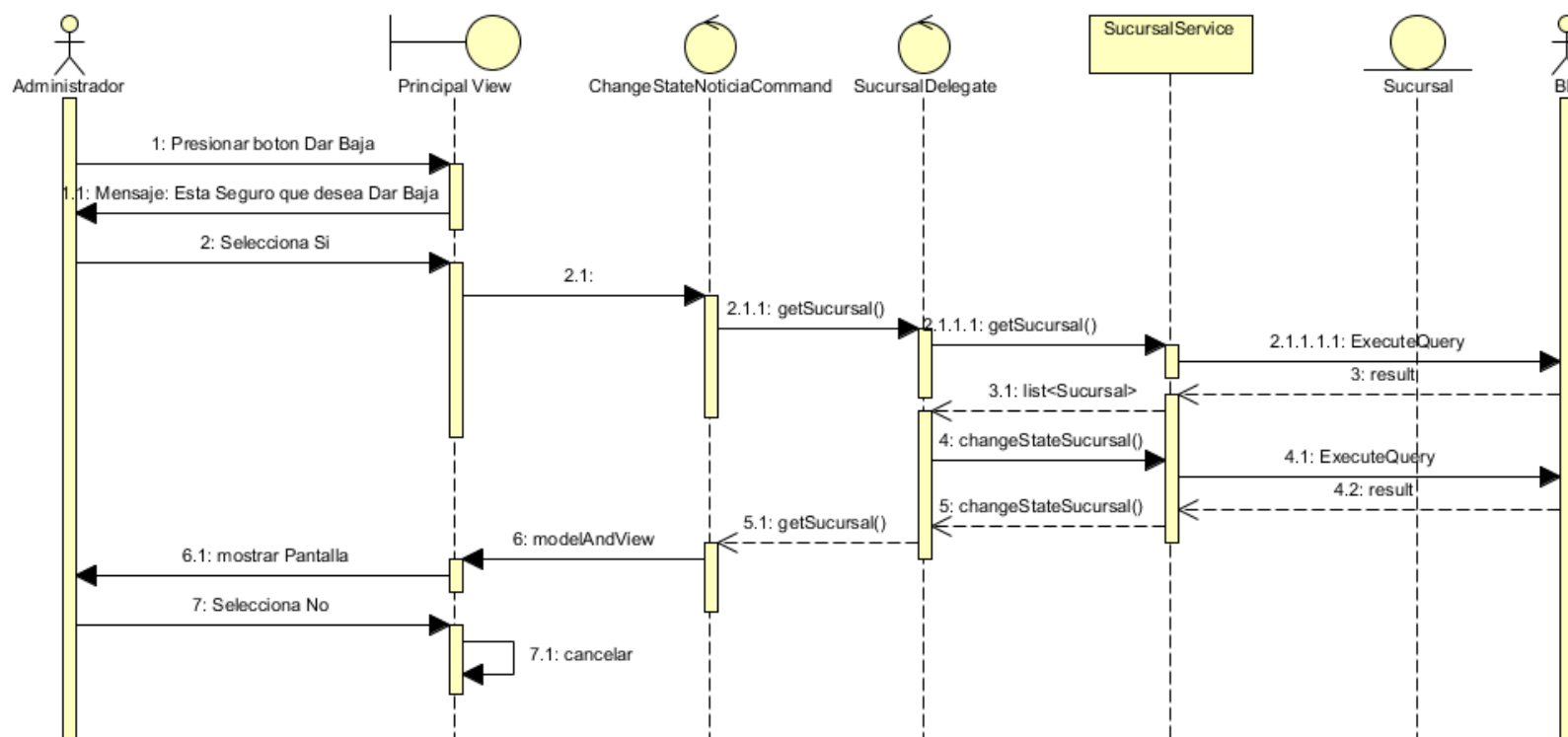


Figura 134 Diagrama de Secuencia: Caso de Uso Dar Baja Sucursal

2.1.2.2.3.14.2.1.38 Diagrama de Secuencia: Caso de Uso Modificar Sucursal

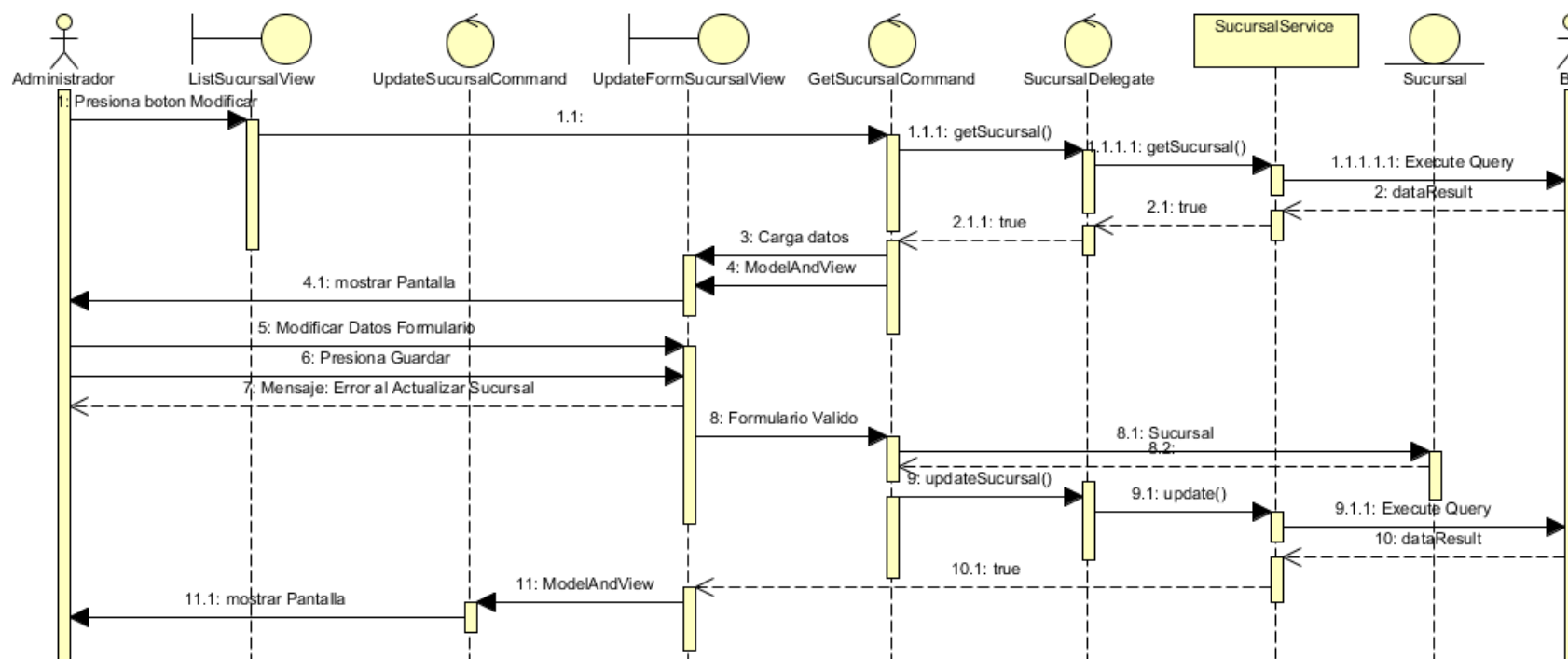


Figura 135 Diagrama de Secuencia: Caso de Uso Modificar Sucursal

2.1.2.2.3.14.2.1.39 Diagrama de Secuencia: Caso de Uso Gestionar Usuario

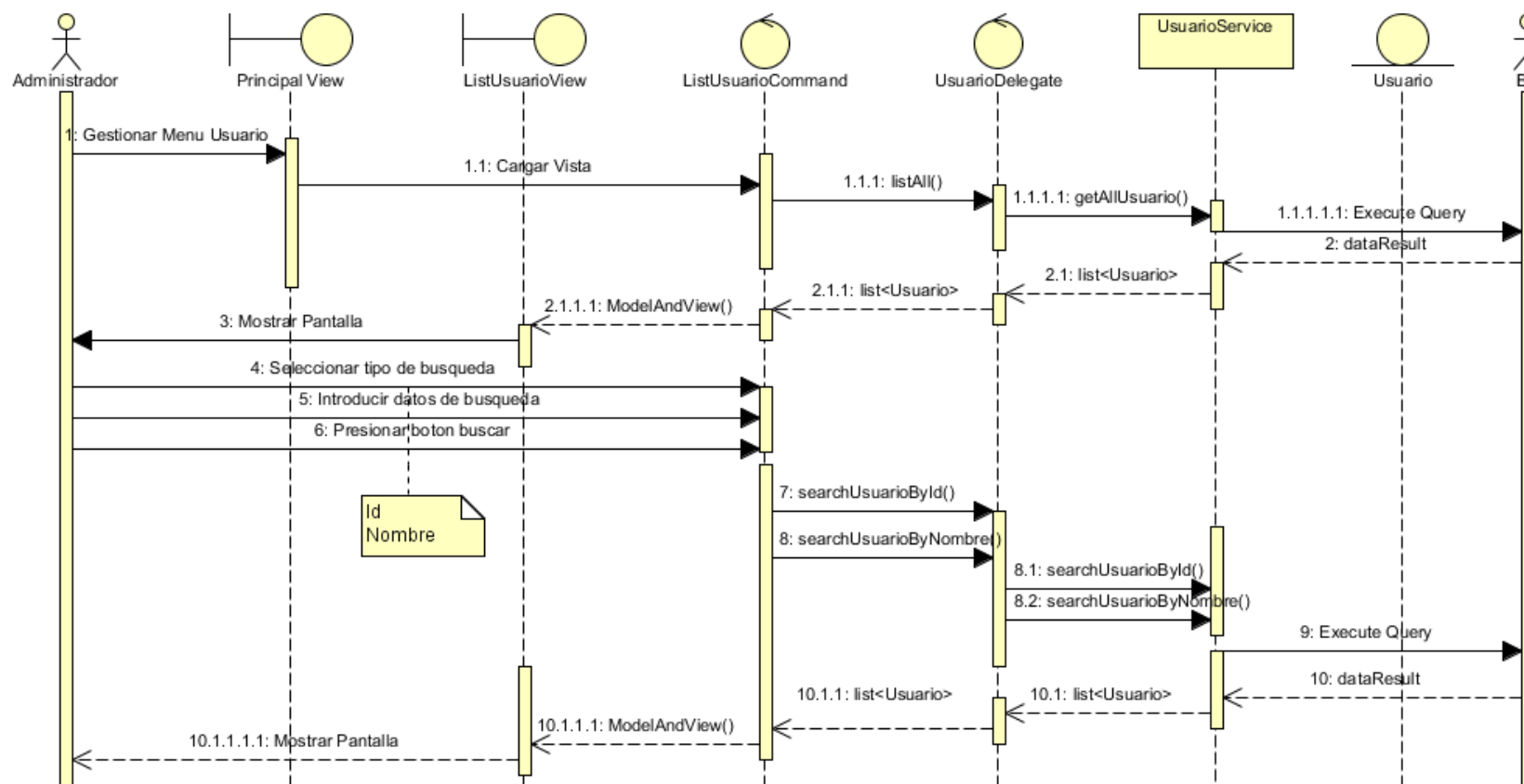


Figura 136 Diagrama de Secuencia: Caso de Uso Gestionar Usuario

2.1.2.2.3.14.2.1.40 Diagrama de Secuencia: Caso de Uso Adicionar Usuario

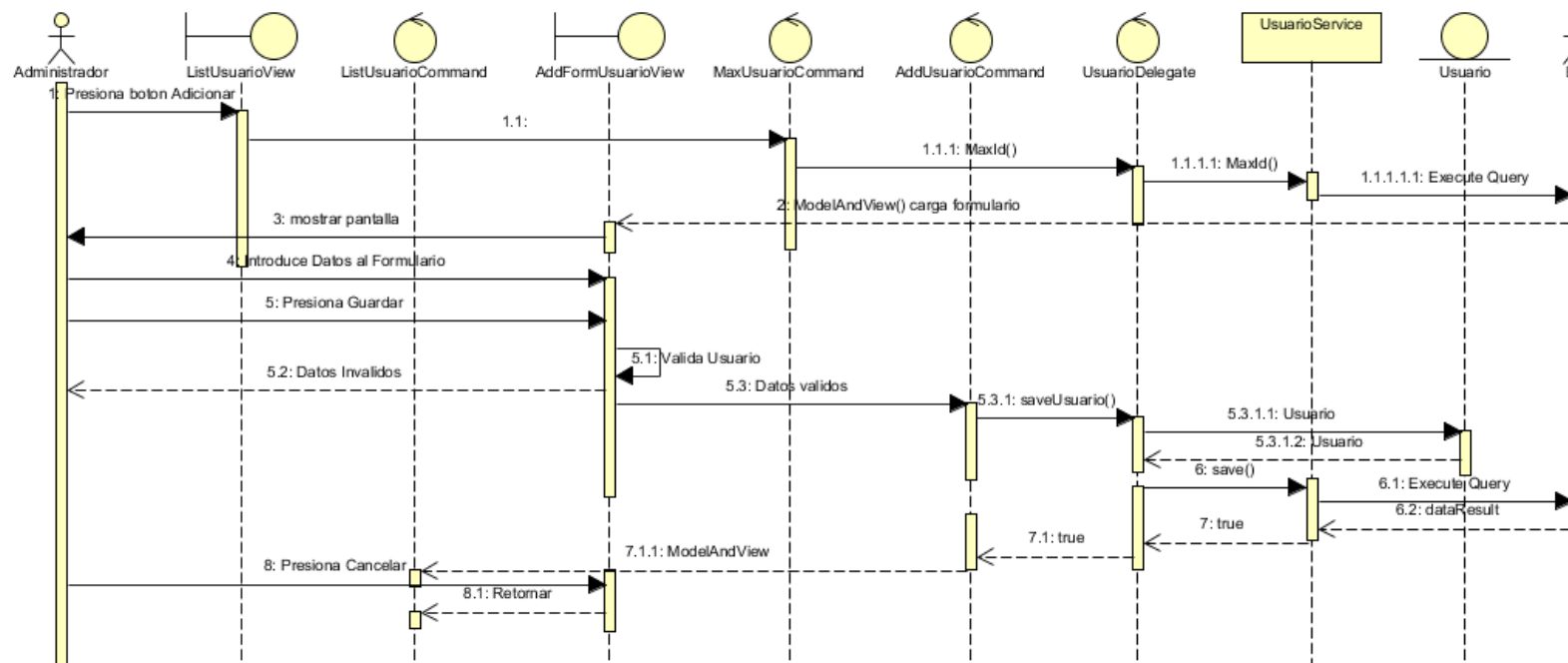


Figura 137 Diagrama de Secuencia: Caso de Uso Adicionar Usuario

2.1.2.2.3.14.2.1.41 Diagrama de Secuencia: Caso de Uso Dar Baja Usuario

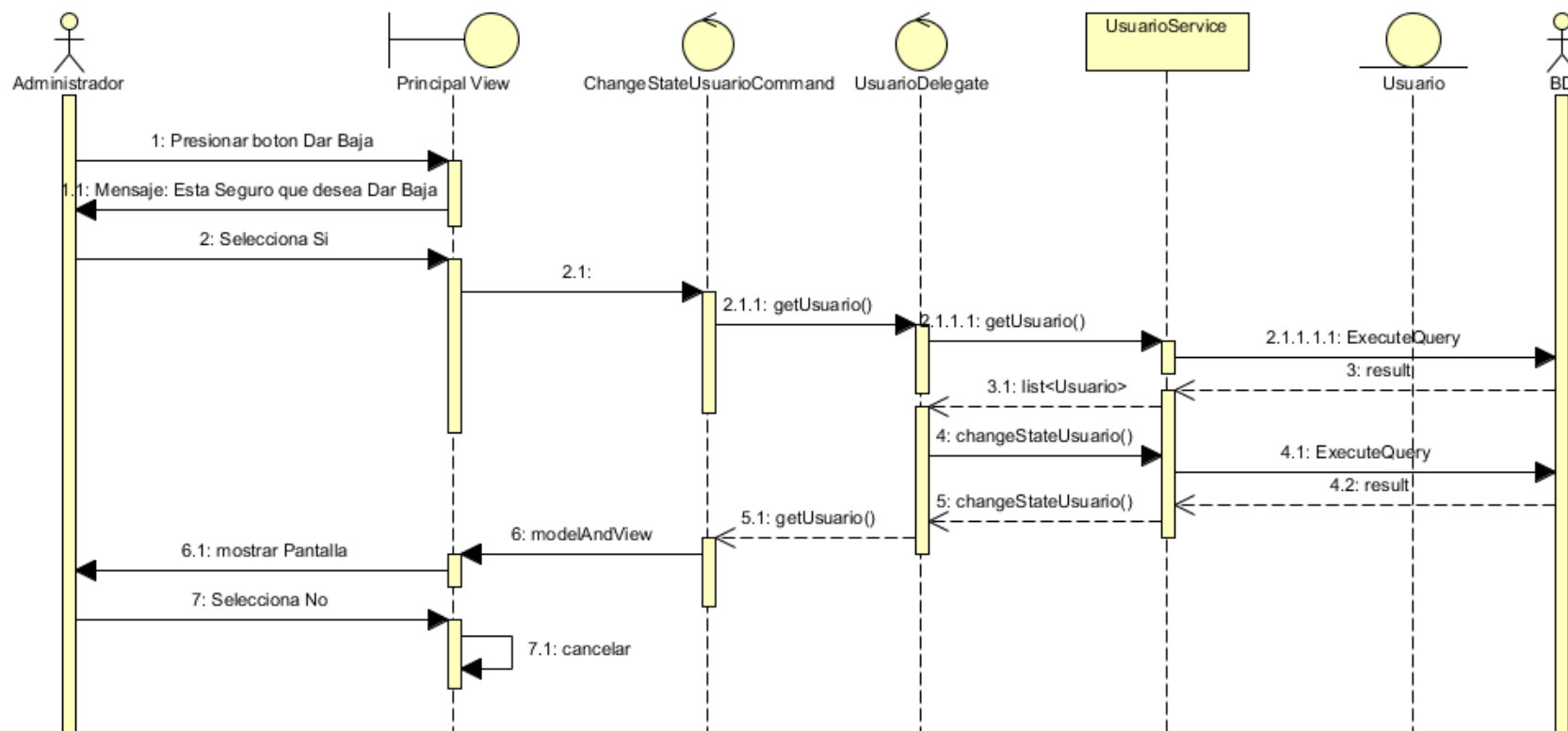


Figura 138 Diagrama de Secuencia: Caso de Uso Dar Baja Usuario

2.1.2.2.3.14.2.1.42 Diagrama de Secuencia: Caso de Uso Modificar Usuario

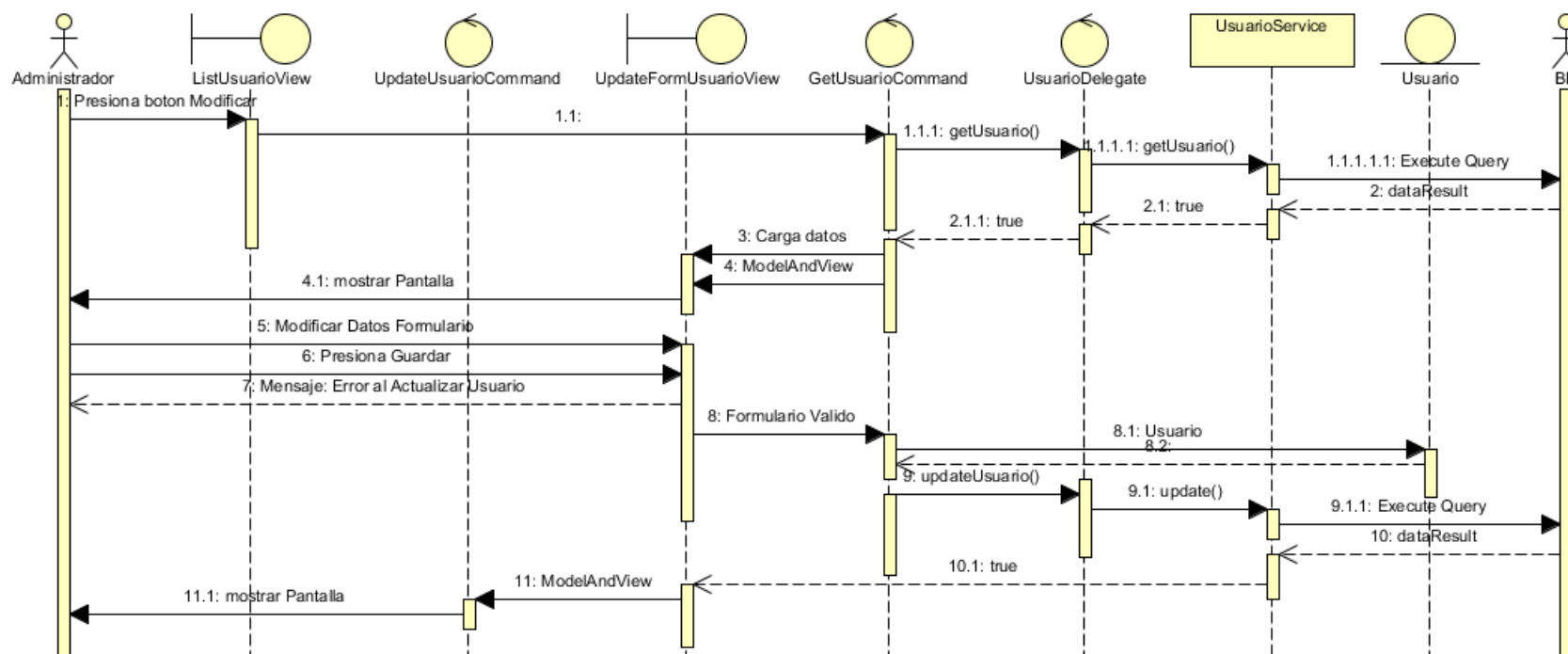


Figura 139 Diagrama de Secuencia: Caso de Uso Modificar Usuario

2.1.2.2.3.14.2.1.43 Diagrama de Secuencia: Caso de Uso Gestionar Venta

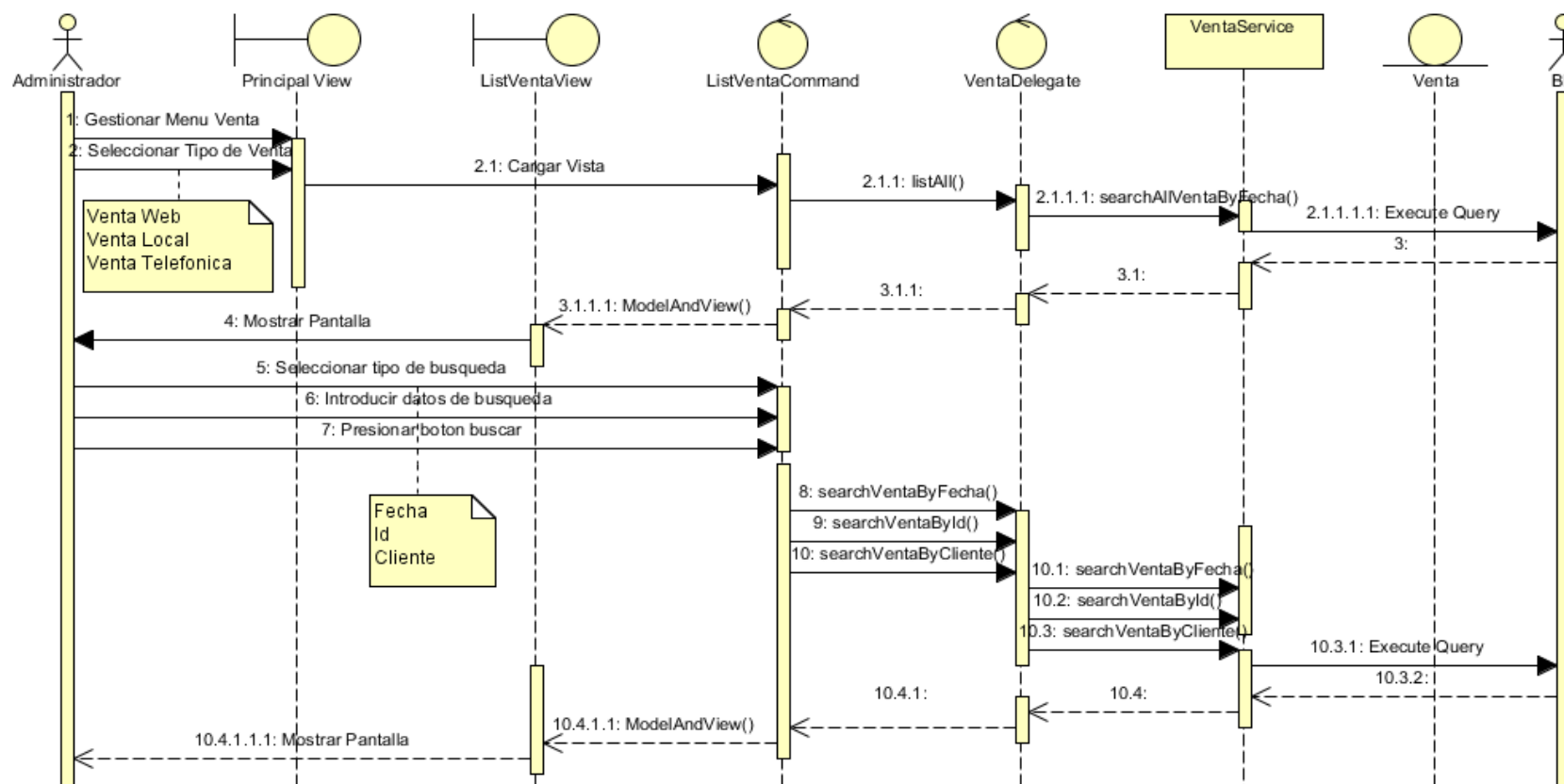


Figura 140 Diagrama de Secuencia: Caso de Uso Gestionar Venta

2.1.2.2.3.14.2.1.44 Diagrama de Secuencia: Caso de Uso Eliminar Venta Web

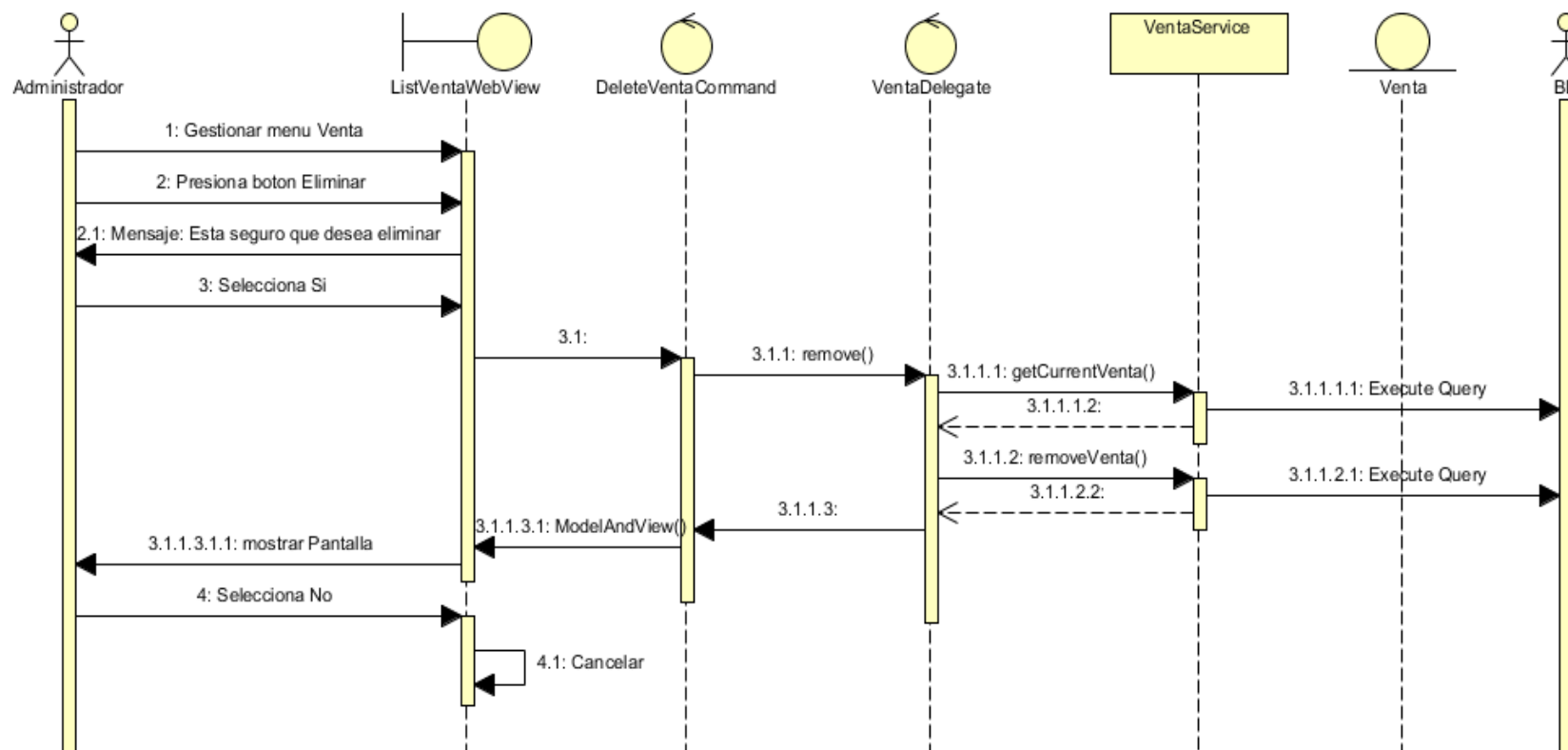


Figura 141 Diagrama de Secuencia: Caso de Uso Eliminar Venta Web

2.1.2.2.3.14.2.1.45 Diagrama de Secuencia: Caso de Uso Agregar Venta Web

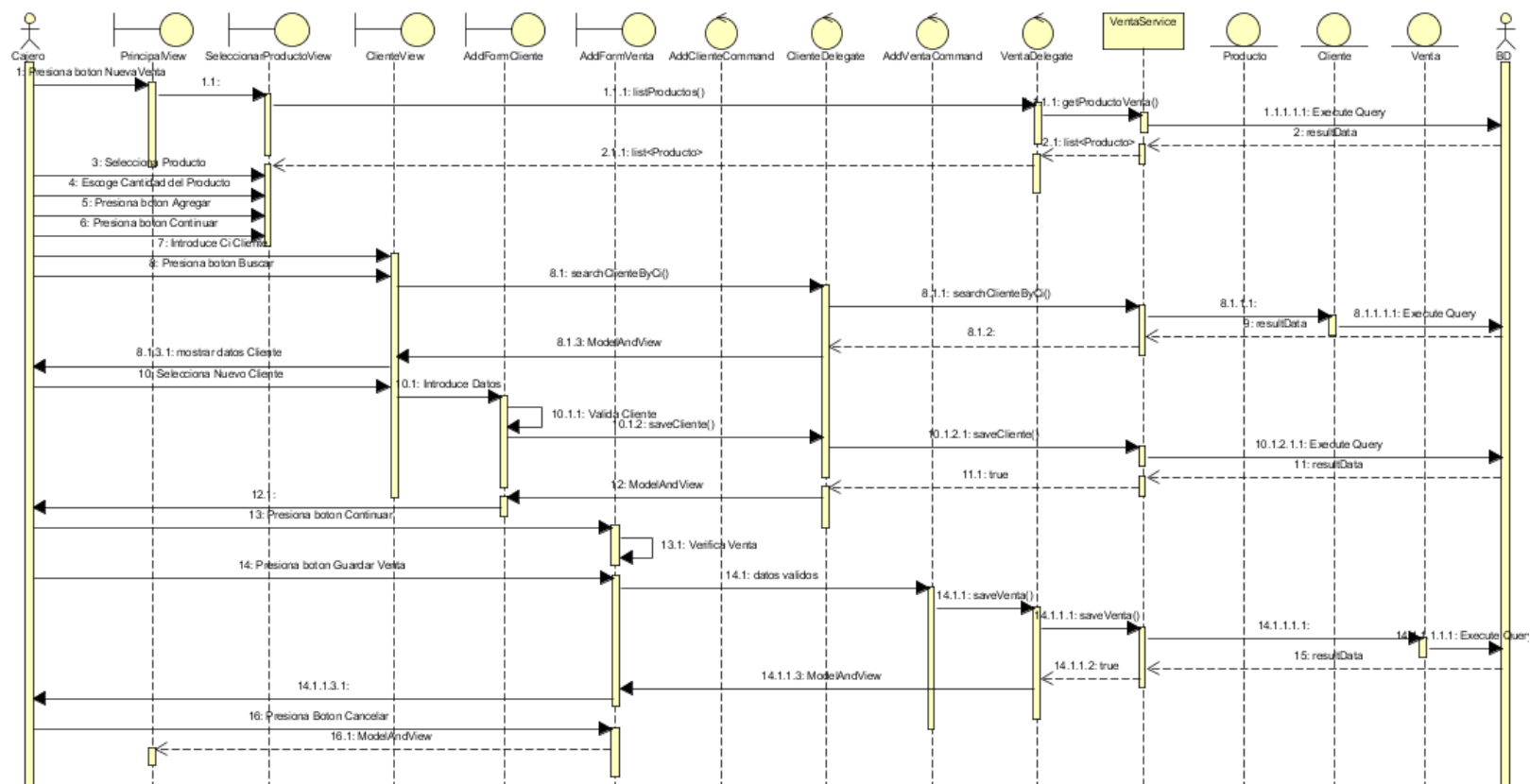


Figura 142 Diagrama de Secuencia: Caso de Uso Adicionar Venta Web

2.1.2.2.3.14.2.1.46 Diagrama de Secuencia: Caso de Uso Confirmar Venta Web

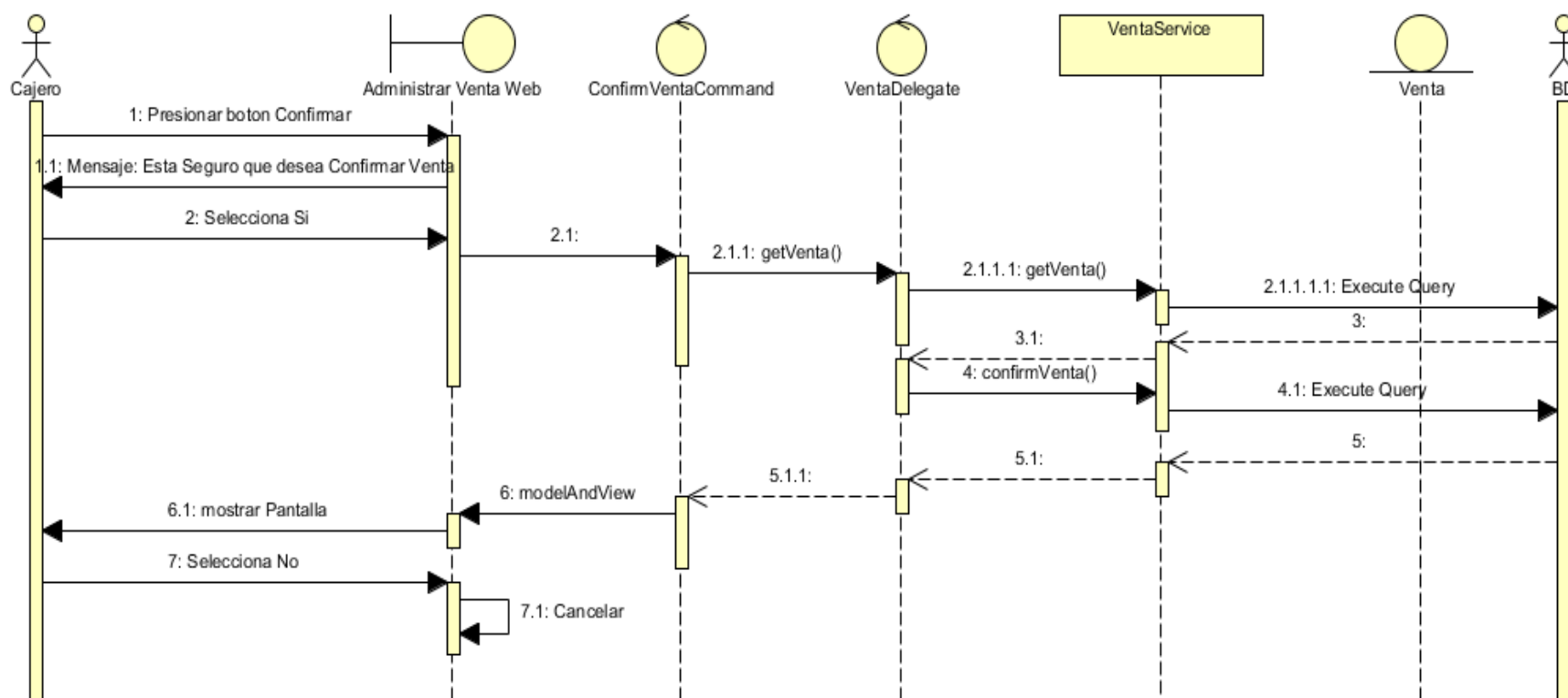


Figura 143 Diagrama de Secuencia: Caso de Uso Confirmar Venta Web

2.1.2.2.3.14.2.1.47 Diagrama de Secuencia: Caso de Uso Confirmar Entrega Venta Web

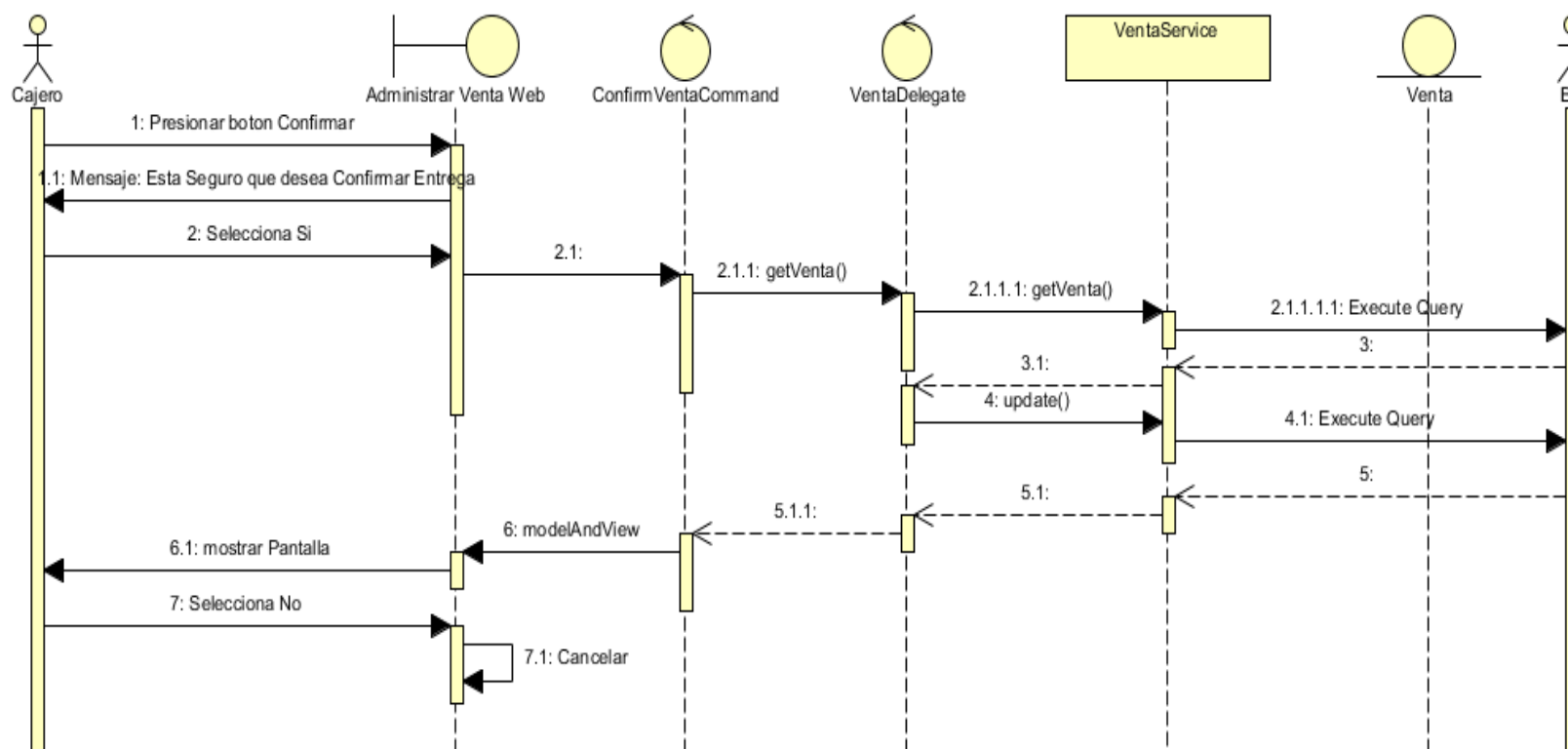


Figura 144 Diagrama de Secuencia: Caso de Uso Confirmar Entrega Venta Web

2.1.2.2.3.15 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.).

2.1.2.2.3.15.1 Modelado de Diagrama de Clases

2.1.2.2.3.15.1.1 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.

2.1.2.2.3.15.1.2 Mecanismos de Abstracción

- Clasificación / Instanciación
- Composición / Descomposición
- Agrupación / Individualización
- Especialización / Generalización

La clasificación es una 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.

2.1.2.2.3.15.1.3 Diagrama de Clases

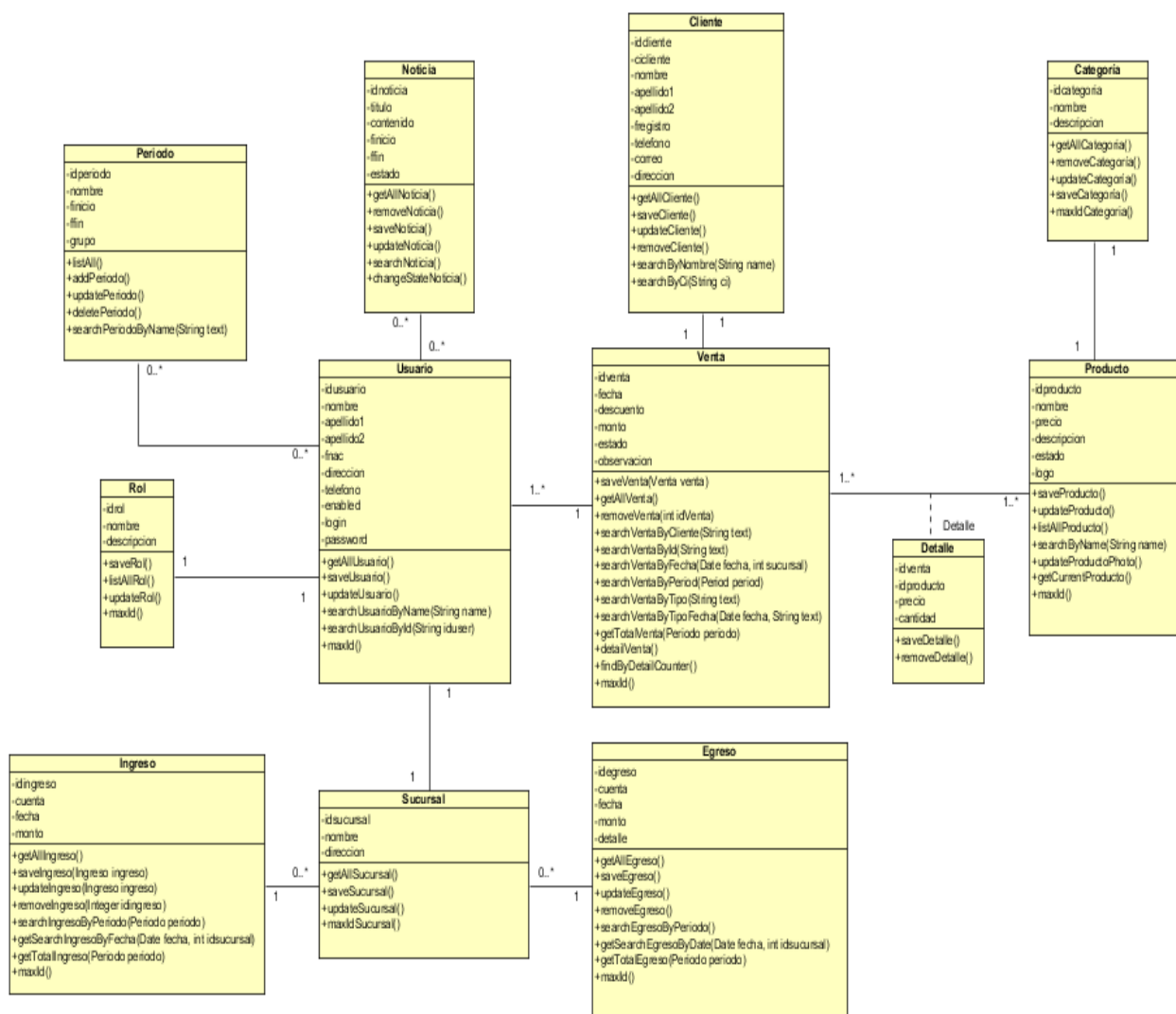


Figura 145 Diagrama de Clases

2.1.2.2.3.15.1.4 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.

2.1.2.2.3.15.1.4.1 Objeto Categoría

OBJETO:	Categoría
MÉTODO:	<u>getAllCategoría</u>
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Categoria.getName()	
DESCRIPCIÓN	

OBJETO:	Categoría
MÉTODO:	<u>removeCategoría</u>
ARGUMENTO:	(Categoria categoria)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Categoria categoria)	
DESCRIPCIÓN	

OBJETO:	Categoría
MÉTODO:	<u>updateCategoría</u>
ARGUMENTO:	(Categoria categoria)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Categoria categoria)	
DESCRIPCIÓN	

OBJETO:	Categoría
MÉTODO:	<u>saveCategoría</u>
ARGUMENTO:	(Categoria categoria)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Categoria categoria)	

DESCRIPCIÓN

OBJETO:	Categoría
MÉTODO:	maxIdCategoria
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idcategoria) from vidividi.categoria p").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 71 Objeto Categoría

2.1.2.2.3.15.1.4.2 Objeto Cliente

OBJETO:	Cliente
MÉTODO:	getAllCliente
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Cliente.getName())	
DESCRIPCIÓN	

OBJETO:	Cliente
MÉTODO:	saveCliente
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Cliente cliente)	
DESCRIPCIÓN	

OBJETO:	Cliente
MÉTODO:	updateCliente
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Cliente cliente)	
DESCRIPCIÓN	

OBJETO:	Cliente
---------	---------

MÉTODO:	removeCliente
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Cliente cliente)	
DESCRIPCIÓN	

OBJETO:	Cliente
MÉTODO:	searchByNombre
ARGUMENTO:	(String nombre)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Cliente.class).add(Restrictions.like("nombre",value,MatchMode.ANYWHERE))	
DESCRIPCIÓN	

OBJETO:	Cliente
MÉTODO:	searchByCi
ARGUMENTO:	(String Ci)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Cliente.class).add(Restrictions.like("cicliente",Integer.parseInt(value))	
DESCRIPCIÓN	

Tabla 72 Objeto Cliente

2.1.2.2.3.15.1.4.3 Objeto Detalle

OBJETO:	Detalle
MÉTODO:	saveDetalle
ARGUMENTO:	(Detalle detalle)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Detalle detalle)	
DESCRIPCIÓN	

OBJETO:	Detalle
MÉTODO:	removeDetalle

ARGUMENTO:	(Detalle detalle)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Detalle detalle)	
DESCRIPCIÓN	

Tabla 73 Objeto Detalle

2.1.2.2.3.15.1.4.4 Objeto Egreso

OBJETO:	Egreso
MÉTODO:	getAllEgreso
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Egreso.getName())	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	saveEgreso
ARGUMENTO:	(Egreso egreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Egreso egreso)	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	removeEgreso
ARGUMENTO:	(Egreso egreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Egreso egreso)	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	updateEgreso
ARGUMENTO:	(Egreso egreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().update(Egreso egreso)
DESCRIPCIÓN

OBJETO:	Egreso
MÉTODO:	searchEgresoByPeriodo
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Egreso.class).add(Restrictions.between("fecha", period.getFinicio(), period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	getSearchEgresoByDate
ARGUMENTO:	(Date fecha, int idsucursal)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Egreso.class).add(Restrictions.between("fecha", period.getFinicio(), period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	getTotalEgreso
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select sum(monto) from egreso").list().get(0).toString()	
DESCRIPCIÓN	

OBJETO:	Egreso
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idegreso) from egreso p").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 74 Objeto Egreso

2.1.2.2.3.15.1.4.5 Objeto Ingreso

OBJETO:	Ingreso
MÉTODO:	getAllIngreso
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Ingreso.getName())	
DESCRIPCIÓN	

OBJETO:	Ingreso
MÉTODO:	saveIngreso
ARGUMENTO:	(Ingreso ingreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Ingreso ingreso)	
DESCRIPCIÓN	

OBJETO:	Ingreso
MÉTODO:	removeIngreso
ARGUMENTO:	(Ingreso ingreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Ingreso ingreso)	
DESCRIPCIÓN	

OBJETO:	Ingreso
MÉTODO:	updateIngreso
ARGUMENTO:	(Ingreso ingreso)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Ingreso ingreso)	
DESCRIPCIÓN	

OBJETO:	Ingreso
---------	---------

MÉTODO:	searchIngresoByPeriodo
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Ingreso.class).add(Restrictions.between("fecha",period.getFinicio(),period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Ingreso
MÉTODO:	getSearchIngresoByDate
ARGUMENTO:	(Date fecha, int idsucursal)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Ingreso.class).add(Restrictions.between("fecha",period.getFinicio(),period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Ingreso
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idingreso) from ingreso p").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 75 Objeto Ingreso

2.1.2.2.3.15.1.4.6 Objeto Noticia

OBJETO:	Noticia
MÉTODO:	getAllNoticia
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Noticia.getName())	
DESCRIPCIÓN	

OBJETO:	Noticia
---------	---------

MÉTODO:	saveNoticia
ARGUMENTO:	(Noticia noticia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Noticia noticia)	
DESCRIPCIÓN	

OBJETO:	Noticia
MÉTODO:	removeNoticia
ARGUMENTO:	(Noticia noticia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Noticia noticia)	
DESCRIPCIÓN	

OBJETO:	Noticia
MÉTODO:	updateNoticia
ARGUMENTO:	(Noticia noticia)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Noticia noticia)	
DESCRIPCIÓN	

2.1.2.2.3.15.1.4.7 Objeto Periodo

OBJETO:	Periodo
MÉTODO:	getAllPeriodo
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Periodo.getName())	
DESCRIPCIÓN	

OBJETO:	Periodo
MÉTODO:	savePeriodo
ARGUMENTO:	(Periodo periodo)

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Periodo periodo)	
DESCRIPCIÓN	

OBJETO:	Periodo
MÉTODO:	RemovePeriodo
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Periodo periodo)	
DESCRIPCIÓN	

OBJETO:	Periodo
MÉTODO:	updatePeriodo
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Periodo periodo)	
DESCRIPCIÓN	

OBJETO:	Periodo
MÉTODO:	searchPeriodoByName
ARGUMENTO:	(Periodo periodo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Periodo.class).add(Restrictions.like("nombre",value,MatchMode.ANYWHERE))	
DESCRIPCIÓN	

2.1.2.2.3.15.1.4.8 Objeto Producto

OBJETO:	Producto
MÉTODO:	getAllProducto
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().find("from " + Producto.getName()
DESCRIPCIÓN

OBJETO:	Producto
MÉTODO:	saveProducto
ARGUMENTO:	(Producto Producto)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Producto Producto)	
DESCRIPCIÓN	

OBJETO:	Producto
MÉTODO:	remove Producto
ARGUMENTO:	(Producto Producto)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Producto producto)	
DESCRIPCIÓN	

OBJETO:	Producto
MÉTODO:	updateProducto
ARGUMENTO:	(Producto producto)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Producto producto)	
DESCRIPCIÓN	

OBJETO:	Producto
MÉTODO:	searchByName
ARGUMENTO:	(String name)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Producto producto)	
DESCRIPCIÓN	

OBJETO:	Producto
---------	----------

MÉTODO:	updateProductoPhoto
ARGUMENTO:	(Producto remoteProducto,byte fileIcon[],Boolean photoChange)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().updateProducto(Producto producto)	
DESCRIPCIÓN	

OBJETO:	Producto
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("s elect max(p.idproducto) from producto").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 76 Objeto Producto

2.1.2.2.3.15.1.4.9 Objeto Rol

OBJETO:	Rol
MÉTODO:	getAllRol
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Rol.getName()	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	saveRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Rol rol)	
DESCRIPCIÓN	

OBJETO:	Rol
---------	-----

MÉTODO:	removeRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Rol Rol)	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	updateRol
ARGUMENTO:	(Rol rol)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Rol rol)	
DESCRIPCIÓN	

OBJETO:	Rol
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idrol) from rol").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 77 Objeto Rol

2.1.2.2.3.15.1.4.10 Objeto Sucursal

OBJETO:	Sucursal
MÉTODO:	getAllSucursal
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Sucursal.getName())	
DESCRIPCIÓN	

OBJETO:	Sucursal
MÉTODO:	saveSucursal

ARGUMENTO:	(Sucursal sucursal)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Sucursal sucursal)	
DESCRIPCIÓN	

OBJETO:	Sucursal
MÉTODO:	removeSucursal
ARGUMENTO:	(Sucursal sucursal)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Sucursal sucursal)	
DESCRIPCIÓN	

OBJETO:	Sucursal
MÉTODO:	searchSucursalById
ARGUMENTO:	(String idsucursal)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Sucursal.class).add(Restrictions.like("idsucursal",Integer.parseInt(value))).list()	
DESCRIPCIÓN	

OBJETO:	Sucursal
MÉTODO:	searchSucursalByUsuario
ARGUMENTO:	(String idusuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select s.idsucursal,s.nombre,s.telefono,s.direccion,s.estado from sucursal s,usuario u where u.idsucursal=s.idsucursal && u.idusuario="+idusuario).setResultTransformer(Transformers.aliasToBean(Sucursal.class)).uniqueResult()	
DESCRIPCIÓN	

OBJETO:	Sucursal
MÉTODO:	maxId
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idsucursal) from sucursal ").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 78 Objeto Sucursal

2.1.2.2.3.15.1.4.11 Objeto Usuario

OBJETO:	Usuario
MÉTODO:	getAllUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().find("from " + Usuario.getName())	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	saveUsuario
ARGUMENTO:	(Usuario usuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Usuario usuario)	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	updateUsuario
ARGUMENTO:	(Usuario usuario)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().update(Usuario usuario)	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	searchUsuarioByName
ARGUMENTO:	(String name)
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Usuario.class).add(Restrictions.like("nombre",value,MatchMode.ANYWHERE))
DESCRIPCIÓN

OBJETO:	Usuario
MÉTODO:	searchUsuarioById
ARGUMENTO:	(String iduser)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Usuario.class).setFetchMode("rol", FetchMode.JOIN).add(Restrictions.like("idusuario", Integer.parseInt(value))).list();	
DESCRIPCIÓN	

OBJETO:	Usuario
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select max(p.idusuario) from usuario").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 79 Objeto Usuario

2.1.2.2.3.15.1.4.12 Objeto Venta

OBJETO:	Venta
MÉTODO:	getAllVenta
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().save(Venta venta)	
DESCRIPCIÓN	

OBJETO:	Venta
---------	-------

MÉTODO:	removeVenta
ARGUMENTO:	(int idVenta)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().delete(Venta venta)	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	searchVentaByCliente
ARGUMENTO:	(String text)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
returngetHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria (Venta.class).add(Restrictions.eq("idcliente", Integer.valueOf(text))).list();	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	searchVentaById
ARGUMENTO:	(String text)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Venta.class).add(Restrictions.eq("idventa",Integer.valueOf(text))).list();	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	searchVentaByFecha
ARGUMENTO:	(Date fecha, int sucursal,int tipo)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Venta.class).add(Restrictions.between("fecha",fechaInicio,fechaFin)).add(Restrictions.eq("estado", tipo)).add(Restrictions.eq("idsucursal", idsucursal)).list()	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	searchVentaByPeriod
ARGUMENTO:	(Period period)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Venta.class).add(Restrictions.between("fecha",period.getFinicio(), period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	searchVentaByTipoFecha
ARGUMENTO:	(Date fecha, String text)
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Venta.class).add(Restrictions.eq("idtipo",type)).add(Restrictions.eq("idventa",Integer.valueOf(text))).add(Restrictions.between("fecha",initDate,fechaFin)).add(Restrictions.eq("idsucursal",idsucursal)).add(Restrictions.eq("idusuario",idusuario)).list()	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	getTotalVenta(Periodo periodo)
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createCriteria(Venta.class).add(Restrictions.between("fecha", period.getFinicio(), period.getFfin())).list()	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	detailVenta
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("select p.idproducto,p.nombre,p.logo,d.precio,d.cantidad from producto p,detalle d " + "where d.idventa="+value+"&d.idproducto=p.idproducto").setResultTransformer(Transformers.aliasToBean(ProductoDetail.class)).list()
DESCRIPCIÓN

OBJETO:	Venta
MÉTODO:	findByDetailCounter()
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("S ELECT idproducto,cantidad,precio FROM detalle d,venta v where v.fecha >"+initDate+" ' && " + "v.fecha<"+ endDate + " ' && v.idventa = d.idventa && v.estado='3' && v.idsucursal="+idsucursal).setResultTransformer(Transformers.aliasToBean(Product oDetailCounter.class)).list()	
DESCRIPCIÓN	

OBJETO:	Venta
MÉTODO:	maxId
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
getHibernateTemplate().getSessionFactory().getCurrentSession().createSQLQuery("s elect max(p.idventa) from venta ").list().get(0).toString()	
DESCRIPCIÓN	

Tabla 80 Objeto Venta

2.1.2.2.3.15.2 Modelo Entidad Relación

2.1.2.2.3.15.2.1 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.

2.1.2.2.3.15.2.2 Modelo Entidad / Relación

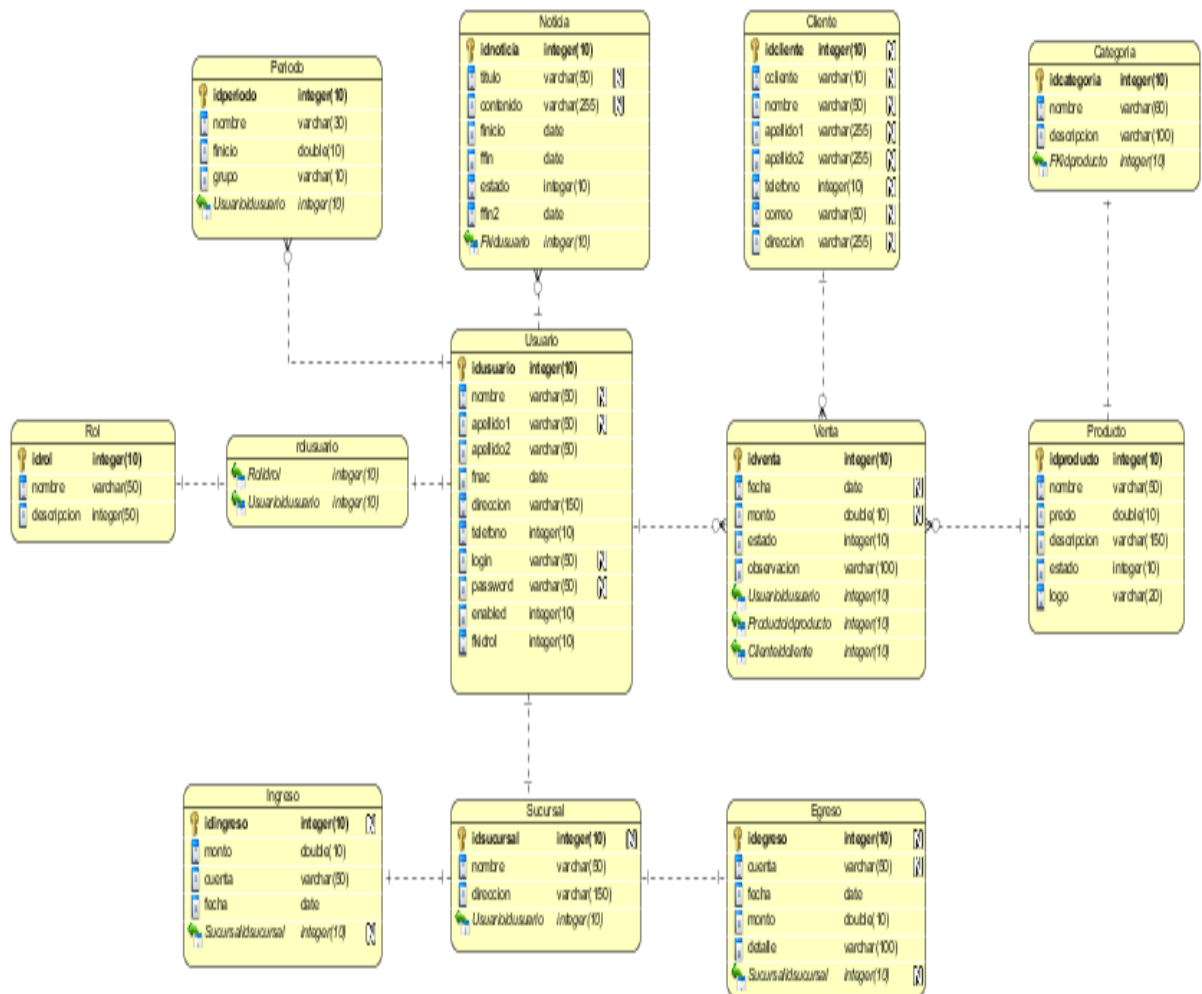


Figura 146 Diagrama Entidad Relación

2.1.2.2.3.15.2.3 Creación de la Base de Datos

```
-- MySQL Administrator dump 1.4
--
-----
-- Server version      5.1.53-community
--
-- Create schema vidividi
--

CREATE DATABASE IF NOT EXISTS vidividi;
USE vidividi;

--
-- Definition of table `categoria`
--
CREATE TABLE `categoria` (
  `idcategoria` int(10) unsigned NOT NULL,
  `nombre` varchar(30) DEFAULT NULL,
  `descripcion` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`idcategoria`) USING BTREE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `categoria`
--
--
-- Definition of table `cliente`
--
CREATE TABLE `cliente` (
  `idcliente` int(10) unsigned NOT NULL,
  `cicliente` int(11) DEFAULT NULL,
  `nombre` varchar(30) DEFAULT NULL,
  `apellido1` varchar(30) DEFAULT NULL,
  `apellido2` varchar(30) DEFAULT NULL,
  `fregistro` datetime DEFAULT NULL,
  `telefono` varchar(15) DEFAULT NULL,
  `correo` varchar(30) DEFAULT NULL,
  `direccion` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`idcliente`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `cliente`
--
--
-- Definition of table `detalle`
CREATE TABLE `detalle` (
```

```

`idventa` int(10) unsigned NOT NULL,
`idproducto` int(10) unsigned NOT NULL,
`precio` double DEFAULT NULL,
`cantidad` int(11) DEFAULT NULL,
PRIMARY KEY (`idventa`,`idproducto`) USING BTREE,
KEY `FK_detalle_producto` (`idproducto`),
CONSTRAINT `FK_detalle_producto` FOREIGN KEY (`idproducto`)
REFERENCES `producto` (`idproducto`),
CONSTRAINT `FK_detalle_venta` FOREIGN KEY (`idventa`) REFERENCES
`venta` (`idventa`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 ROW_FORMAT=FIXED;
--
-- Definition of table `egreso`
--
CREATE TABLE `egreso` (
  `idegreso` int(10) unsigned NOT NULL,
  `cuenta` varchar(50) NOT NULL,
  `fecha` datetime NOT NULL,
  `monto` double,
  `detalle` varchar(200) ,
  `idsucursal` int(11) NOT NULL,
  PRIMARY KEY (`idegreso`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `egreso`
--
--
-- Definition of table `ingreso`
--
CREATE TABLE `ingreso` (
  `idingreso` int(10) unsigned NOT NULL,
  `cuenta` varchar(45) DEFAULT NULL,
  `monto` double DEFAULT NULL,
  `idsucursal` int(10) unsigned DEFAULT NULL,
  `fecha` datetime DEFAULT NULL,
  PRIMARY KEY (`idingreso`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `ingreso`
--
--
-- Definition of table `noticia`
--
CREATE TABLE `noticia` (

```



```

`idnoticia` int(10) unsigned NOT NULL,
`título` varchar(45) DEFAULT NULL,
`contenido` varchar(500) DEFAULT NULL,
`finicio` datetime DEFAULT NULL,
`ffin` datetime DEFAULT NULL,
`estado` int(10) DEFAULT NULL,
`idusuario` int(10) DEFAULT NULL,
PRIMARY KEY (`idnoticia`),
KEY `FK_noticia_1` (`idusuario`),
CONSTRAINT `FK_noticia_1` FOREIGN KEY (`idusuario`) REFERENCES
`usuario` (`idusuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Definition of table `periodo`
--
CREATE TABLE `periodo` (
  `idperiodo` int(10) NOT NULL,
  `nombre` varchar(255) DEFAULT NULL,
  `finicio` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `ffin` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `grupo` int(11) DEFAULT NULL,
  PRIMARY KEY (`idperiodo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `periodo`
--
--
-- Definition of table `producto`
--
CREATE TABLE `producto` (
  `idproducto` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nombre` varchar(30) DEFAULT NULL,
  `precio` double DEFAULT NULL,
  `descripcion` varchar(100) DEFAULT NULL,
  `estado` int(10) unsigned DEFAULT NULL,
  `idcategoria` int(10) unsigned DEFAULT NULL,
  `logo` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`idproducto`) USING BTREE,
  KEY `idcategoriaFK` (`idcategoria`),
  CONSTRAINT `idcategoriaFK` FOREIGN KEY (`idcategoria`) REFERENCES
`categoria` (`idcategoria`)
) ENGINE=InnoDB AUTO_INCREMENT=25 DEFAULT CHARSET=latin1;
--
-- Dumping data for table `producto`

```

```

--
--
-- Definition of table `rol`
--
CREATE TABLE `rol` (
  `idrol` int(11) NOT NULL,
  `nombre` varchar(100) NOT NULL,
  `descripcion` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`idrol`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 ROW_FORMAT=DYNAMIC;
--
-- Dumping data for table `rol`
--
--
-- Definition of table `rol_usuario`
--
CREATE TABLE `rol_usuario` (
  `idrol` int(11) NOT NULL DEFAULT '0',
  `idusuario` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`idrol`,`idusuario`),
  KEY `FK4B3314BE466D6A91` (`idrol`),
  KEY `FK_rol_usuario_2` (`idusuario`),
  CONSTRAINT `FK4B3314BE466D6A91` FOREIGN KEY (`idrol`)
REFERENCES `rol` (`idrol`) ON DELETE CASCADE,
  CONSTRAINT `FK_rol_usuario_2` FOREIGN KEY (`idusuario`) REFERENCES
`usuario` (`idusuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 ROW_FORMAT=FIXED;
--
-- Dumping data for table `rol_usuario`
--
--
-- Definition of table `sucursal`
--
CREATE TABLE `sucursal` (
  `idsucursal` int(10) unsigned NOT NULL,
  `nombre` varchar(30) DEFAULT NULL,
  `telefono` varchar(10) DEFAULT NULL,
  `direccion` varchar(80) DEFAULT NULL,
  `estado` int(11) DEFAULT NULL,
  PRIMARY KEY (`idsucursal`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `sucursal`
--
--

```

```

-- Definition of table `usuario`
--
CREATE TABLE `usuario` (
  `idusuario` int(10) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `apellido1` varchar(45) DEFAULT NULL,
  `apellido2` varchar(45) DEFAULT NULL,
  `fnac` datetime DEFAULT NULL,
  `direccion` varchar(45) DEFAULT NULL,
  `telefono` varchar(45) DEFAULT NULL,
  `enabled` tinyint(1) DEFAULT NULL,
  `login` varchar(45) CHARACTER SET latin1 COLLATE latin1_bin DEFAULT
NULL,
  `password` varchar(45) CHARACTER SET latin1 COLLATE latin1_bin
DEFAULT NULL,
  `idsucursal` int(10) unsigned DEFAULT NULL,
  PRIMARY KEY (`idusuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 ROW_FORMAT=DYNAMIC;
--
-- Dumping data for table `usuario`
--
-- Definition of table `venta`
--
CREATE TABLE `venta` (
  `idventa` int(10) unsigned NOT NULL,
  `fecha` datetime DEFAULT NULL,
  `descuento` double DEFAULT NULL,
  `monto` double DEFAULT NULL,
  `idcliente` int(10) unsigned DEFAULT NULL,
  `iddetalle` int(11) unsigned DEFAULT NULL,
  `idtipo` int(10) unsigned DEFAULT NULL,
  `idusuario` int(10) unsigned NOT NULL,
  `estado` int(10) unsigned DEFAULT NULL,
  `observacion` varchar(255) DEFAULT NULL,
  `idsucursal` int(10) unsigned DEFAULT NULL,
  PRIMARY KEY (`idventa`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 ROW_FORMAT=FIXED;
--
-- Dumping data for table `venta`
--

```

2.1.2.2.3.15.2.4 Diccionario de Datos

2.1.2.2.3.15.2.4.1 Categoría

Key			
Nombre Key	Tipo Key	Keys	
PK1	PrimaryKey	idCategoría	
Attributes			
Atributo	Tipo Dato	Null	Definición
idCategoría	SERIAL/INTEGER	ID	Representa la clave primaria de la Categoría que será generada por la base de datos
nombre	VARCHAR(30)	N	Representa nombre de la Categoría
descripción	VARCHAR(50)	N	Representa la descripción que tiene la Categoría

2.1.2.2.3.15.2.4.2 Cliente

Key			
Nombre Key	Tipo Key	Keys	
PK2	PrimaryKey	idCliente	
Attributes			
Atributo	Tipo Dato	Null	Definición
idCliente	SERIAL/INTEGER	ID	Representa la clave primaria de la Categoría que será generada por la base de datos
nombre	VARCHAR(30)	N	Representa nombre de la Categoría
apellido1	VARCHAR(30)	N	Representa el primer apellido del cliente
apellido2	VARCHAR(30)	N	Representa el primer apellido del cliente
fregistro	DATE	Y	Representa la fecha en la que se registró el cliente
telefono	VARCHAR(15)	N	Representa el número de teléfono del cliente
correo	VARCHAR(30)	Y	Representa el email del cliente
dirección	VARCHAR(100)	N	Representa la dirección de vivienda del

00) cliente

2.1.2.2.3.15.2.4.3 Detalle

Key

Nombre Key	Tipo Key	Keys
PK8	PrimaryKey	idProducto
PK13	PrimaryKey	idVenta

Attributes

Atributo	Tipo Dato	Null	Definición
idproducto	SERIAL/INT EGER	ID	Representa la clave primaria del producto que se guarda en una venta
idventa	SERIAL/INT EGER	ID	Representa la clave primaria de la Venta la cual se guardará en la base de datos
precio	DOUBLE	N	Representa el precio del producto con el que se efectuó la venta
cantidad	INTEGER	N	Representa la cantidad que se compró de determinado producto

2.1.2.2.3.15.2.4.4 Egreso

Key

Nombre Key	Tipo Key	Keys
PK3	PrimaryKey	idEgreso

Attributes

Atributo	Tipo Dato	Null	Definición
idEgreso	SERIAL/INT EGER	ID	Representa la clave primaria del Egreso que será generada por la base de datos
cuenta	VARCHAR(50)	N	Representa el nombre del Egreso
fecha	DATE	N	Representa la fecha en la que se efectuó el Egreso
monto	DOUBLE	N	Representa el valor del monto del Egreso
detalle	VARCHAR(50)	N	Representa la descripción del Egreso

2.1.2.2.3.15.2.4.5 Ingreso

Key

Nombre Key	Tipo Key	Keys
PK4	PrimaryKey	idIngreso

Attributes

Atributo	Tipo Dato	Null	Definición
idingreso	SERIAL/INT EGER	ID	Representa la clave primaria del Ingreso que será generada por la base de datos
cuenta	VARCHAR(5 0)	N	Representa el nombre del Ingreso
monto	DATE	N	Representa el valor del monto del Ingreso
fecha	DOUBLE	N	Representa la fecha en la que se efectuó el Ingreso

2.1.2.2.3.15.2.4.6 Noticia**Key**

Nombre Key	Tipo Key	Keys
PK5	PrimaryKey	idNoticia

Attributes

Atributo	Tipo Dato	Null	Definición
idnoticia	SERIAL/INT EGER	ID	Representa la clave primaria de la Noticia que será generada por la base de datos
titulo	VARCHAR(3 0)	N	Representa nombre del título de la Noticia
contenido	VARCHAR(5 0)	N	Representa el contenido que tiene la Noticia
fregistro	DATE		Representa la fecha en la cual se publicará la Noticia
ffin	DATE		Representa la fecha de expiración de la Noticia publicada
estado	INTEGER		Representa el estado de la Noticia, 1=Habilitado 0=Inhabilitado

2.1.2.2.3.15.2.4.7 Periodo**Key**

Nombre Key	Tipo Key	Keys
PK6	PrimaryKey	idPeriodo

Attributes

Atributo	Tipo Dato	Null	Definición
idperiodo	SERIAL/INT	ID	Representa la clave primaria del Periodo

finicio	EGER		que será generada por la base de datos
finicio	DATE	N	Representa la fecha de inicio del Periodo
finicio	DATE	N	Representa la fecha de fin del Periodo
grupo	INTEGER		

2.1.2.2.3.15.2.4.8 Producto

Key

Nombre Key	Tipo Key	Keys
PK7	PrimaryKey	idProducto

Attributes

Atributo	Tipo Dato	Null	Definición
idproducto	SERIAL/INT EGER	ID	Representa la clave primaria del Producto que será generada por la base de datos
nombre	VARCHAR(30)	N	Representa nombre de la Categoría
precio	DOUBLE	N	Representa el precio referencial del Producto
descripción	VARCHAR(100)	N	Representa los ingredientes que tiene el Producto
estado	INTEGER	N	Representa el estado del Producto, 1=Habilitado 0=Inhabilitado

2.1.2.2.3.15.2.4.9 Rol

Key

Nombre Key	Tipo Key	Keys
PK8	PrimaryKey	idRol

Attributes

Atributo	Tipo Dato	Null	Definición
idRol	SERIAL/INT EGER	ID	Representa la clave primaria del Rol que será generada por la base de datos
nombre	VARCHAR(100)	N	Representa nombre del Rol
descripción	VARCHAR(100)	N	Representa la descripción que tiene el Rol

2.1.2.2.3.15.2.4.10 Sucursal

Key

Nombre Key	Tipo Key	Keys
PK9	PrimaryKey	idSucursal

Attributes

Atributo	Tipo Dato	Null	Definición
idsucursal	SERIAL/INTEGER	ID	Representa la clave primaria de la Sucursal que será generada por la base de datos
nombre	VARCHAR(30)	N	Representa el nombre de la Sucursal
teléfono	VARCHAR(10)	N	Representa la descripción que tiene la Categoría
dirección	VARCHAR(80)		Representa la dirección de la Sucursal
estado	INTEGER		Representa el estado de la Sucursal, 1=Habilitado 0=Inhabilitado

2.1.2.2.3.15.2.4.11 Usuario

Key

Nombre Key	Tipo Key	Keys
PK10	PrimaryKey	idUsuario

Attributes

Atributo	Tipo Dato	Null	Definición
idusuario	SERIAL/INTEGER	ID	Representa la clave primaria del Usuario que será generada por la base de datos
nombre	VARCHAR(45)	N	Representa el nombre del Usuario
apellido1	VARCHAR(45)	N	Representa el primer apellido del Usuario
apellido2	VARCHAR(45)		Representa el segundo apellido del Usuario
fnac	DATE		Representa la fecha de nacimiento del Usuario
dirección	VARCHAR(45)		Representa la dirección de vivienda del Usuario
teléfono	VARCHAR(45)		Representa el teléfono de contacto del Usuario
estado	INTEGER		Representa el estado del Usuario,

login	VARCHAR(45)	1=Habilitado 0=Inhabilitado Representa el identificador con el que el usuario se logueara
password	VARCHAR(45)	Representa la contraseña asignada para el Usuario

2.1.2.2.3.15.2.4.12 Venta

Key

Nombre Key	Tipo Key	Keys
PK11	PrimaryKey	idVenta

Attributes

Attributo	Tipo Dato	Null	Definicion
idventa	SERIAL/INT EGER	ID	Representa la clave primaria de la Venta que será generada por la base de datos
Fecha	VARCHAR(45)	N	Representa la fecha en la que se efectuó la Venta
Descuento	VARCHAR(45)	N	Representa el descuento realizado para la Venta
monto	VARCHAR(45)		Representa el importe total de la Venta
Idtipo	DATE		Representa el tipo de Venta, 1=Web 2=Telefónica 3=Local
Estado	VARCHAR(45)		Representa el estado de la Venta, 1= Venta Pendiente 2= Venta Confirmada 3= Confirmación de Entrega de la Venta
Observación	VARCHAR(45)		Representa algunas observaciones que tiene la Venta

2.1.2.2.3.16 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).

2.1.2.2.3.16.1 Modelado de Diagrama de Paquetes

2.1.2.2.3.16.1.1 Diagrama de Paquetes

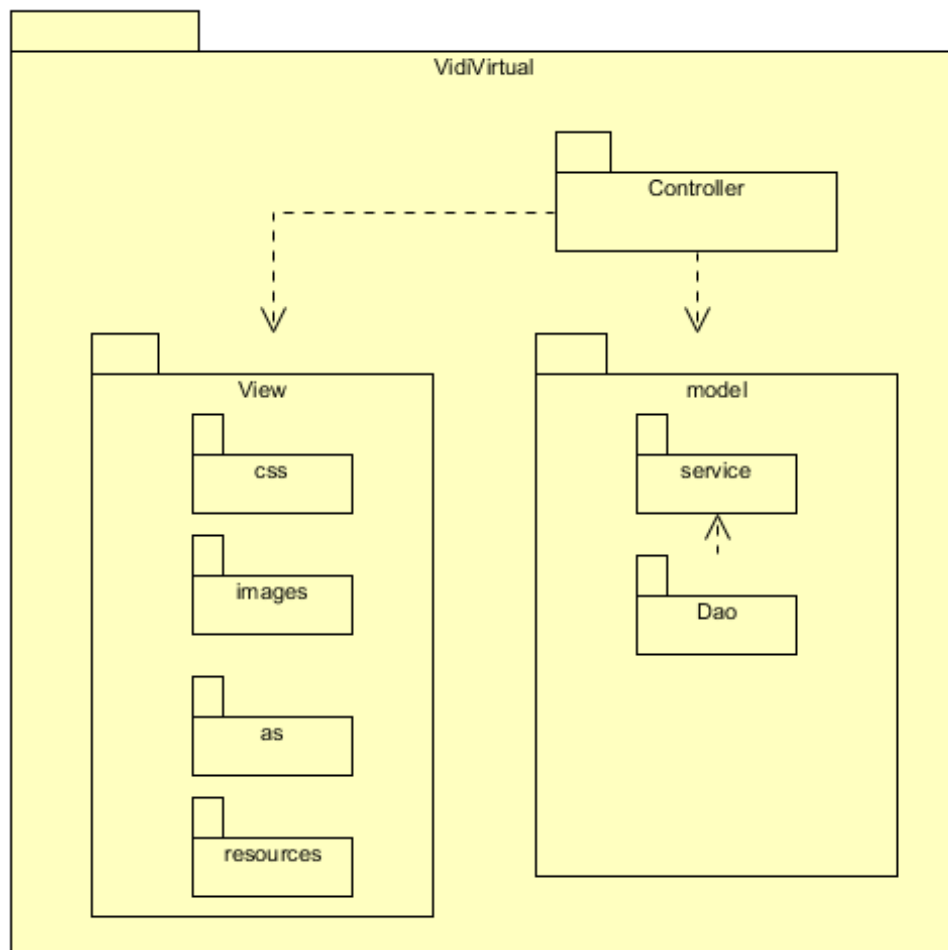


Figura 147 Diagrama de Paquetes

2.1.2.2.3.17 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.

2.1.2.2.3.17.1 Modelado de Diagrama de Despliegue

2.1.2.2.3.17.1.1 Diagrama de despliegue

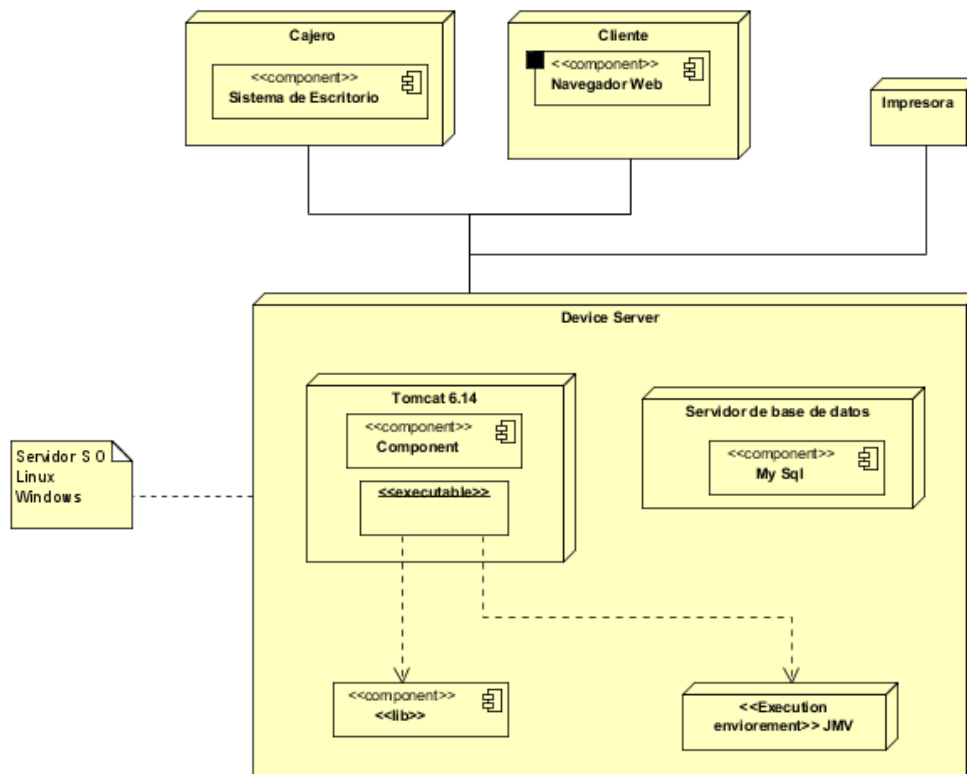


Figura 148 Diagrama de despliegue

2.1.2.2.3.18 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.

2.1.2.2.3.18.1 Plan de Pruebas

2.1.2.2.3.18.1.1 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.

2.1.2.2.3.18.1.1.1 Objetivo

Este Plan de Pruebas fue desarrollado con el fin de guiar el proceso de pruebas al proyecto Vidi Virtual para el Mejoramiento en el control de ventas y difusión de información 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 podrá elaborar un documento que presente las evaluaciones correspondientes en el plazo más corto posible.

2.1.2.2.3.18.1.1.2 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 empresa de comida rápida VidiVidi, además que sea aprobado por la comunidad de usuarios finales del sistema.

2.1.2.2.3.18.1.1.3 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.

2.1.2.2.3.18.1.1.4 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

Tabla 81 Hardware

SOFTWARE		
Aplicación	Versión	Descripción
Sistema para el Mejoramiento del control de ventas y difusión de información	0.3	Sistema desarrollado por el grupo de trabajo.
Visual Paradigm for UML Community Edition	8.0	Herramienta de Diseño y Desarrollo de Diagramas UML para RUP
MySql	5.1.53	Motor de Base de Datos Relacional.

Tabla 82 Software

OTROS		
Descripción	Versión	Descripción
Documentación	0.3	Documentación de Desarrollo de Software

Tabla 83 Otros

2.1.2.2.3.18.1.1.5 Alcance

Este plan describe las pruebas del sistema, que serán aplicadas a los componentes del Sistema Informático para el Mejoramiento del control de ventas y difusión de información.

Se asume que antes de probar cada módulo habrá una revisión informal y solo 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 Datos.

Tiempo de respuesta de carga del Sitio Web.

Tiempo de respuesta y Pool de conexiones a la Base de Datos.

2.1.2.2.3.18.1.1.6 Referencias

Glosario del Proyecto.

Plan de Desarrollo de Software.

2.1.2.2.3.18.2 Descripción de las Pruebas Planeadas

2.1.2.2.3.18.2.1 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.

2.1.2.2.3.18.3 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.

2.1.2.2.3.18.4 Tipos y Técnicas de pruebas

2.1.2.2.3.18.4.1 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 (MySQL).
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 84 Pruebas de Pool de Conexiones

2.1.2.2.3.18.4.2 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: - Todos los casos de uso del Sistema Informático. - La prueba funcional fue realizada juntamente con la persona encargada de su Administración.

Tabla 85 Pruebas Funcionales

2.1.2.2.3.18.4.3 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 a 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 86 Pruebas de Validación de Datos

2.1.2.2.3.18.4.4 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: - 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 87 Pruebas de Perfiles de Desempeño

2.1.2.2.3.18.4.5 Ejecución de las Pruebas

2.1.2.2.3.18.4.5.1 Demostración y Resultados de la prueba de

Validación de Datos

2.1.2.2.3.18.4.5.2 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 Tamaño: Usuario	1. Alfanumérico 2. $1 \leq \text{Caracteres} \leq 50$	3. Vacío 4. Caracteres > 50
Tipo: Clave Tamaño: Clave	5. Alfanumérico 6. $1 \leq \text{Caracteres} \leq 50$	7. Vacío 8. Caracteres > 50

Tabla 88. Caso de Prueba Caja Negra: Ingresar al Sistema

Casos de Prueba válidas

CP1	
Usuario	Clave
administrador	administrador

Cubre las clases de equivalencias válidas: 1-2-5-6

Casos de Pruebas Inválidas

CP2	
Usuario	Clave
admin	

Cubre las clases de equivalencias inválidas: 7

CP3	
Usuario	Clave
	admin

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario ABM Gestionar Categoría

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 > 150
Tamaño: Descripción	6. $0 \leq \text{Caracteres} \leq 150$	

Tabla 89 Caso de Prueba Caja Negra: Gestionar Categoría

Casos de Prueba válidas

CP1	
Nombre	Descripción
Hamburguesa	Distintos tipos de hamburguesa

Cubre las clases de equivalencias válidas: 1-3-5-7

Casos de Pruebas Inválidas

CP2	
Nombre	Descripción
	Distintos tipos de hamburguesa

Cubre las clases de equivalencias inválidas: 2

CP3	
Nombre	Descripción
>35	Distintos tipos de hamburguesa

Cubre las clases de equivalencias inválidas: 4

Interfaz: Formulario ABM Gestionar Cliente

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: CI Tamaño: CI	1. Numero 2. 7	3. Letra o carácter especial 4. !=7
Tipo: Nombre Tamaño: Nombre	5. Alfabético 6. 1<=Caracteres<=30	7. Vacío 8. Letra>30
Tipo: Apellido1 Tamaño: Apellido1	9. Alfabético 10. 0<=letra<=30	11. Vacío 12. Numero o carácter especial 13. Letra >30
Tipo: Apellido2 Tamaño: Apellido2	14. Alfabético 15. 0<=letra<=30	16. Numero o carácter especial 17. Letra >30
Tipo: Fecha de registro	18. Fecha	19. Ninguno
Tipo: Teléfono	20. Numérico	21. Alfabético 22. Vacío
Tipo: Correo	23. Alfanumérico 24. Vacío	25. Ninguno

Tabla 90 Caso de Prueba Caja Negra: Gestionar Cliente

Casos de Prueba válidas

CP1						
CI	Nombre	Apellido1	Apellido2	Fecha	Teléfono	Correo
71538249	Juan José	López	Araoz	10/07/2012	7123456	

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-14-15-18-20-23-24

Casos de Pruebas Inválidas

CP2						
CI	Nombre	Apellido1	Apellido2	Fecha	Teléfono	Correo
71538249	Juan José		Araoz	10/07/2012	7123456	

Cubre las clases de equivalencias inválidas: 11

CP3						
CI	Nombre	Apellido1	Apellido2	Fecha	Teléfono	Correo
12345678	Juan José	López	Araoz	10/07/2012	7123456	

Cubre las clases de equivalencias inválidas: 4

Interfaz: Formulario ABM Gestionar Egreso

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Cuenta Tamaño: Cuenta	1. Alfanumérico 2. $0 \leq \text{letra} \leq 35$	3. Vacio 4. Carácter especial 5. $\text{letra} > 35$
Tipo: Fecha Tamaño: Fecha	6. Fecha	7. Vacío
Tipo: Monto Tamaño: Monto	8. Numérico	9. letra o carácter especial 10. Numero > 0 11. Vacío

Tipo: Detalle	12. Alfabético	14. Vacío
Tamaño: Detalle	13. $0 \leq \text{letra} \leq 250$	15. Numero o carácter especial
		16. Letra >250

Tabla 91 Caso de Prueba Caja Negra: Gestionar Egreso

Casos de Prueba válidas

CP1			
Cuenta	Fecha	Monto	Detalle
Caja Chica	10/07/2012	17	Aceite

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-14-15-18-20-23-24

Casos de Pruebas Inválidas

CP2			
Cuenta	Fecha	Monto	Detalle
Caja Chica	10/07/2012	0	Aceite

Cubre las clases de equivalencias inválidas: 10

CP3			
Cuenta	Fecha	Monto	Detalle
	10/07/2012		Aceite

Cubre las clases de equivalencias inválidas: 3,11

Interfaz: Formulario ABM Gestionar Ingreso

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
----------------------	-----------------------------	-------------------------------

Tipo: Cuenta	1. Alfanumérico	3.Vacio
Tamaño: Cuenta	2. $0 \leq \text{letra} \leq 35$	4. Carácter especial
		5. $\text{letra} > 35$
Tipo: Fecha	6. Fecha	7. Vacío
Tamaño: Fecha		
Tipo: Monto	8. Numérico	9. letra o carácter especial
Tamaño: Monto		10. Numero > 0
		11. Vacío

Tabla 92 Caso de Prueba Caja Negra: Gestionar Ingreso

Casos de Prueba Válidas

CP1		
Cuenta	Fecha de gasto	Monto
Caja Chica	15/05/2012	17

Cubre las clases de equivalencias válidas: 1-2-6-8

Casos de Prueba Inválidas

CP2		
Cuenta	Fecha de gasto	Monto
Caja Chica	15/05/2012	0

Cubre las clases de equivalencias inválidas: 10

CP3		
Cuenta	Fecha de gasto	Monto
	15/05/2012	17

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario ABM Gestionar Noticia

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Titulo Tamaño: Titulo	1. Alfanumérico 2. $0 \leq \text{letra} \leq 100$	3. Vacio 4. $\text{letra} > 100$ 5. $\text{letra} = 0$
Tipo: Contenido Tamaño: Contenido	6. Alfanumérico 7. $0 \leq \text{letra} \leq 1500$	8. Vacio 9. $\text{letra} = 0$
Tipo: Fecha Inicio Tamaño: Fecha Inicio	10. Fecha	Ninguno
Tipo: Fecha Fin Tamaño: Fecha Fin	11. Fecha	12. Vacío
Tipo: Estado Tamaño: Estado	13. Numérico	Ninguno

Tabla 93 Caso de Prueba Caja Negra: Gestionar Noticia

Casos de Prueba Válidas

CP2				
Titulo	Contenido	Fecha Inicio	Fecha Fin	Estado
	A tiempo de saludarle a toda la cliente mencionarle que nuestra nueva sucursal será inaugurada el 1ro de agosto del presente año	10/07/2012	02/08/2012	1

Cubre las clases de equivalencias inválidas: 3

Casos de Prueba Inválidas

CP1				
Título	Contenido	Fecha Inicio	Fecha Fin	Estado
Inauguración		10/07/2012	02/08/2012	1

Cubre las clases de equivalencias inválidas: 8

Interfaz: Formulario ABM Gestionar Periodo

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfanumérico 2. 0<letra<=255	3.Vacio 4. letra = 0
Tipo: Fecha Inicio Tamaño: Fecha Inicio	5. Fecha	6. Otro Valor
Tipo: Fecha Fin Tamaño: Fecha Fin	7. Fecha	8. Otro Valor
Tipo: Grupo Tamaño: Grupo	9. Alfanumérico	10. Vacío

Casos de Prueba Válidas

CP1			
Nombre	Fecha Inicio	Fecha Fin	Grupo
Año	01/01/2012	31/12/2012	2012

Cubre las clases de equivalencias inválidas: 1-2-5-7-9

Casos de Prueba Inválidas

CP2			
Nombre	Fecha Inicio	Fecha Fin	Grupo
		10/07/2012	2012

Cubre las clases de equivalencias inválidas: 3-6

CP3			
Nombre	Fecha Inicio	Fecha Fin	Grupo
Año	01/01/2012		

Cubre las clases de equivalencias inválidas: 8-10

Interfaz: Formulario ABM Gestionar Producto

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfanumérico 2. $0 < \text{letra} \leq 30$	3. Vacio
Tipo: Precio Tamaño: Precio	5. Numérico	5. Letra o carácter 6. Numero=0
Tipo: Descripción Tamaño: Descripción	7. Alfanumérico	8. Vacío
Tipo: Estado Tamaño: Estado	9. Numérico	Ninguno
Tipo: Logo Tamaño: Logo	10. Cadena	Ninguno

Tabla 94 Caso de Prueba Caja Negra: Gestionar Producto

Casos de Prueba Válidas

CP1				
Nombre	Precio	Descripción	Estado	Logo
Hamburguesa	15.0	Hamburguesa doble	1	Hamburguesa.jpg

Cubre las clases de equivalencias inválidas: 1-2-5-7-9-10

Casos de Prueba Inválidas

CP2				
Nombre	Precio	Descripción	Estado	Logo
Hamburguesa	0	Hamburguesa doble	1	Hamburguesa.jpg

Cubre las clases de equivalencias inválidas: 6

CP3				
Nombre	Precio	Descripción	Estado	Logo
	15.0	Hamburguesa doble	1	Hamburguesa.jpg

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario ABM Gestionar Rol

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfanumérico 2. $0 < \text{letra} \leq 30$	3. Vacio
Tipo: Descripción Tamaño: Descripción	4. Alfanumérico	Ninguno

Tabla 95 Caso de Prueba Caja Negra: Gestionar Rol

Casos de Prueba Válidas

CP1	
Nombre	Descripción
Administrador	Administra y Maneja toda la red de VidiVidi

Cubre las clases de equivalencias inválidas: 1-2-4

Casos de Prueba Inválidas

CP2

Nombre	Descripción
Hamburguesa	

Cubre las clases de equivalencias inválidas: 3

Interfaz: Formulario ABM Gestionar Sucursal

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfanumérico 2. $0 < \text{letra} \leq 30$	3. Vacio
Tipo: Teléfono Tamaño: Teléfono	4. Numérico	5. Letra o carácter 6. Numero=0
Tipo: Dirección Tamaño: Dirección	7. Alfanumérico	8. Vacío
Tipo: Estado Tamaño: Estado	9. Numérico	Ninguno

Tabla 96 Caso de Prueba Caja Negra: Gestionar Sucursal

Casos de Prueba Válidas

CP1			
Nombre	Teléfono	Dirección	Estado
Centro	49090	C/ Ingavi Nro 432	1

Cubre las clases de equivalencias inválidas: 1-2-4-7-9

Casos de Prueba Inválidas

CP1			
Nombre	Teléfono	Dirección	Estado
	49090	C/ Ingavi Nro 432	1

Cubre las clases de equivalencias inválidas: 3

CP1			
Nombre	Teléfono	Dirección	Estado
Centro	49090		0

Cubre las clases de equivalencias inválidas: 8

Interfaz: Formulario ABM Gestionar Usuario

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfabético 2. 1<=Caracteres<=30	3. Vacío 4. Letra>30
Tipo: Apellido1 Tamaño: Apellido1	5. Alfabético 6. 0<=letra<=30	7. Vacío 8. Numero o carácter especial 12. Letra >30
Tipo: Apellido2 Tamaño: Apellido2	9. Alfabético	Ninguno
Tipo: Fecha de nacimiento	10. Fecha	11. Vacío
Tipo: Dirección	12. Alfanumérico 14. 0<=letra<=100	13. Alfanumérico
Tipo: Teléfono	15. Numérico	16. Alfabético Vacío
Tipo: Estado	17. 0 o 1	18. numero i= 1 19. numero i=0
Tipo: Login	20. Alfanumérico	22. Vacío

Tamaño: Login	21. 5<=Caracteres<=20	23. Caracteres >20
Tipo: Password	24. Alfanumérico	26. Vacío
Tamaño: Password	25. 5<=Caracteres<=20	27. Caracteres >20

Casos de Prueba Válidas

CP1									
CI	Nom bre	Apelli do1	Apelli do2	Fecha	Direcc ión	Teléfono	Esta do	Log in	Passw ord
71538249	Alex Danie l	Céspe des	Gira	10/07/2012	C/ La Madri d Nro 503	72954119	1	alex cg	Admin

Cubre las clases de equivalencias válidas: 1-2-5-6-9-10-12-14-15-17-20-21

Casos de Prueba Inválidas

CP1									
CI	Nom bre	Apelli do1	Apelli do2	Fecha	Direcc ión	Teléfono	Esta do	Log in	Passw ord
71538249	Alex Danie l	Céspe des	Gira	10/07/2012	C/ La Madri d Nro 503	72954119	1	alex cg	Admin

Cubre las clases de equivalencias inválidas: 3

CP1						
CI	Nombre	Apellido1	Apellido2	Fecha	Teléfono	Correo
71538249	Juan José	López	Araoz	10/07/2012	7123456	

Cubre las clases de equivalencias inválidas: 8

Interfaz: Formulario ABM Gestionar Venta

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Fecha Tamaño: Date	1. Fecha	2. Ninguno
Tipo: Descuento Tamaño: Double	3. Numérico	4. Vacío 5. Numero o carácter especial
Tipo: Monto Tamaño: Double	6. Numérico	7. Vacío 8. Numero o carácter especial
Tipo: Observación Tamaño: Observación	9. Alfanumérico 10. $0 \leq \text{Caracteres} \leq 250$	11. Caracteres >250

Casos de Prueba Válidas

CP1			
Fecha	Descuento	Monto	Observación
2012-06-05 23:10:42	10	34	

Cubre las clases de equivalencias válidas: 1-3-6-9

Casos de Prueba Inválidas

CP1			
Fecha	Descuento	Monto	Observación
2012-06-05 23:10:42	dedo	\$	

Cubre las clases de equivalencias inválidas: 5-8

2.1.2.2.3.18.4.5.3 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 (ilustrada anteriormente), 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 Añadir -- Grupo de Hilos; para efectos prácticos un "Grupo de Hilos" es considerado el grupo de usuarios que desea simular para su aplicación.

Ahora seleccione el icono Grupo de Hilos recientemente creado. Al llevar a cabo este paso la ventana derecha mostrará la siguiente serie de opciones:

Nombre: Utilizado para definir un nombre más descriptivo sobre el grupo de usuarios, como: "Usuarios Esperados".

Número de hilos: Equivale al número de usuarios que se desean simular, en este caso utilizaremos 50.

Periodo de subida: Es el lapso de tiempo en segundos que se desea tener entre cada grupo de usuarios ("Grupo de Hilos"), utilizaremos 15.

Contador de Bucles: Utilizado para indicar si la simulación para grupos de usuarios ("Grupo de Hilos") será llevada a cabo infinitamente, esto es, si selecciona esta opción indica que desea simular 50 usuarios, esperar 15 segundos ("Periodo de subida"), simular otros 50 usuarios y así sucesivamente. Para esta prueba es recomendable defina 15 ciclos para simular un total de 750 usuarios en cuestión de 150 segundos.

Planificador: Finalmente, esta opción permite definir tiempos de arranque para determinados grupos de usuarios ("Grupo de Hilo "), para efectos prácticos de esta simulación no será seleccionada esta opción y se iniciará la prueba a nuestra discreción.

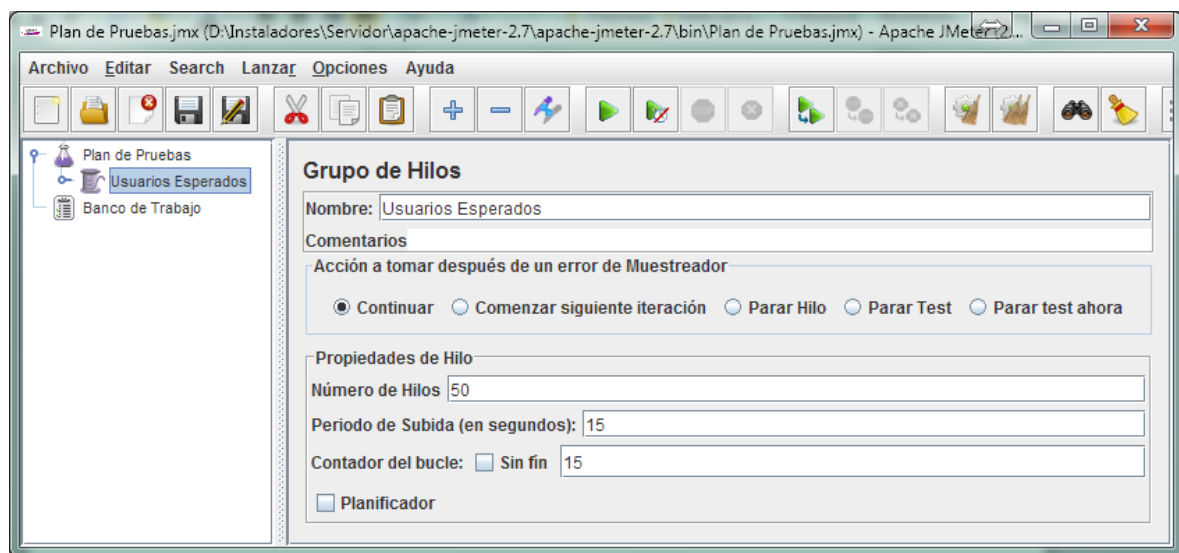


Figura 149 Grupo de Hilos: Usuarios Esperados

Una vez definidas las características del grupo de usuarios ("Grupo de Hilos"), colóquese nuevamente en este icono de la columna izquierda y ahora seleccione la opción Añadir -- Muestreador – Petición HTTP. Lo anterior genera un icono denominado Petición HTTP utilizado para definir las requisiciones de simulación. Si selecciona este último icono aparecerán las siguientes opciones en la ventana derecha:

Nombre de Servidor o 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.

Número de Puerto: Define el puerto TCP de operación del servidor, será empleado 8080.

Ruta: Utilizado para definir la ruta de acceso para llevar a cabo la prueba, será definido /VidiVirtual/index.html, que es la página principal de nuestro sistema web.

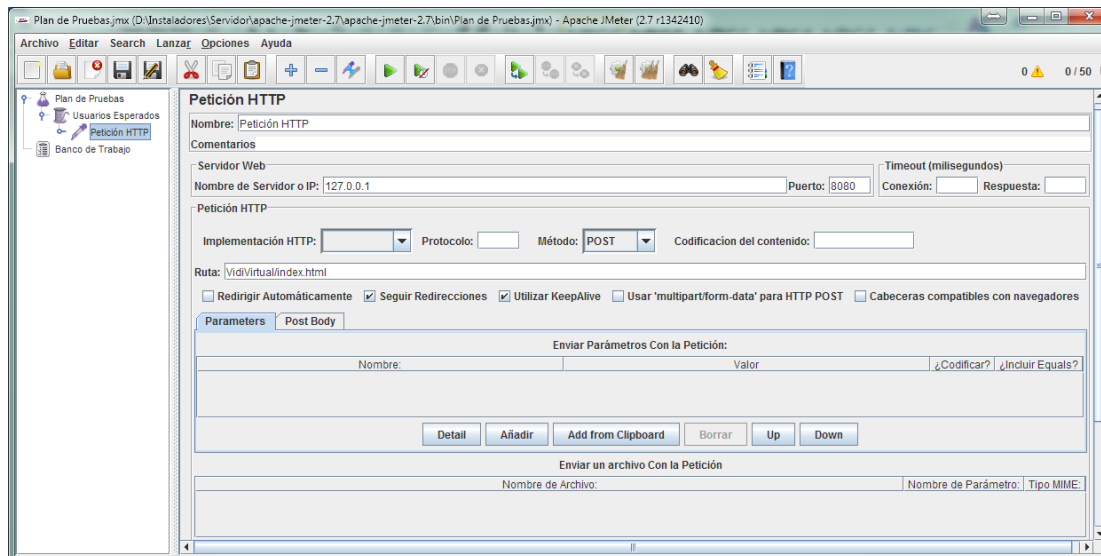


Figura 150 Petición HTTP

- Ahora debe colocarse en el icono de HTTP Request y oprimir el botón derecho de su "mouse", del menú generado seleccione la opción Añadir -- Receptor – Gráfico de Resultados, así como Añadir -- Receptor – Vista de Resultados en Árbol
- Finalmente guarde su plan de prueba ("Plan Pruebas") y ejecútelos seleccionando la opción Arrancar del menú superior. Si se coloca en el icono Gráfica de Resultados mientras se está llevando a cabo la simulación, puede observar como es construida interactivamente la gráfica de resultados.

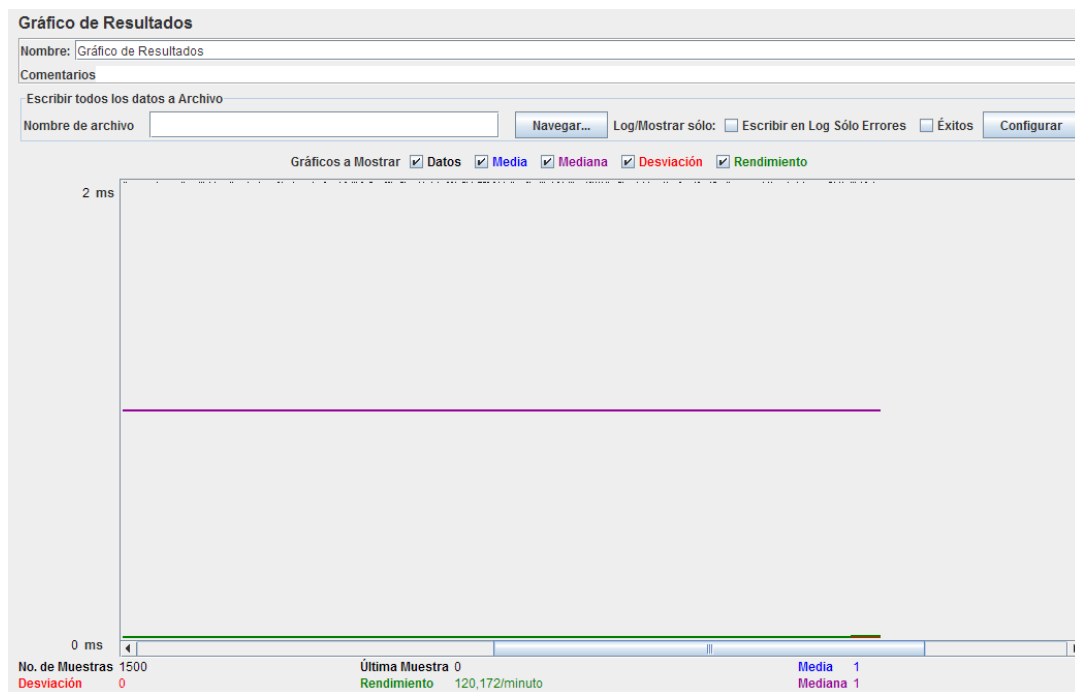


Figura 151 Gráfico de Resultados VidiVirtual

El Resultado Indica que la Página Web VidiVirtual soporta 120,172 Usuarios en un minuto.

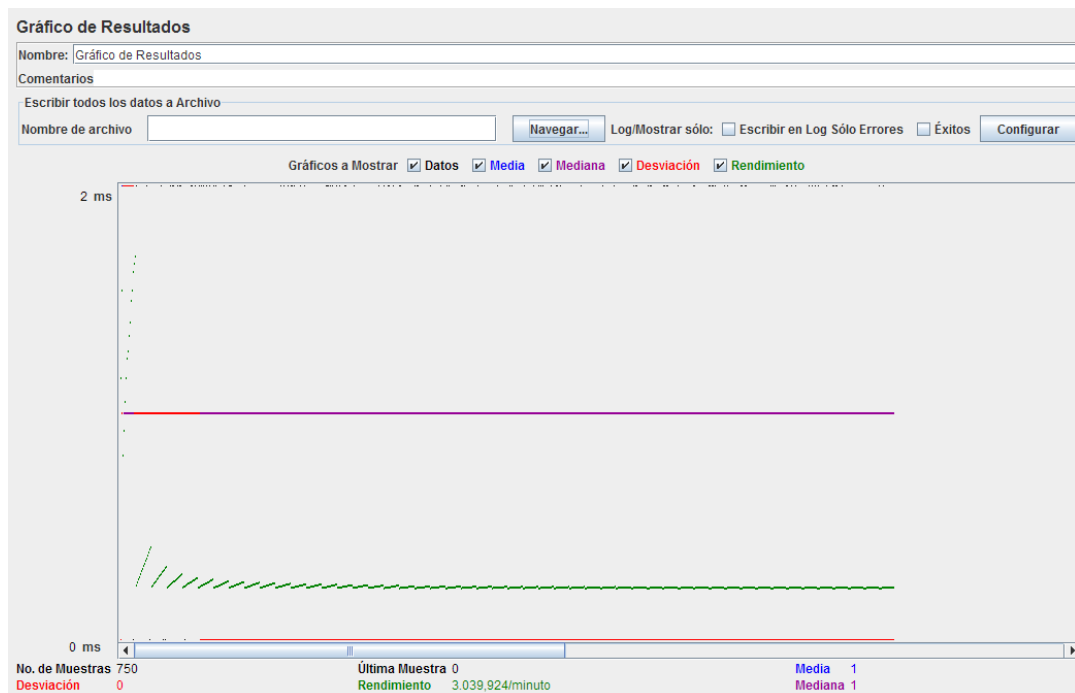


Figura 152 Gráfico de Resultados: VidiStore

El Resultado Indica que la Página Web VidiVirtual soporta 3,039 Usuarios en un minuto.

2.1.2.2.3.18.4.5.4 Prueba de Pool de Conexiones

La prueba también se realiza con la herramienta JMeter, y su configuración es la siguiente:

Configuración de la Conexión JDBC	
Nombre:	Configuración de la Conexión JDBC
Comentarios	
Nombre Variable Enlazado al Pool	
Nombre de Variable:	MySQL
Configuración del Pool de Conexiones	
Número Máximo de Conexiones:	10
Timeout del Pool:	10000
Intervalo de Limpieza por Inactividad (ms):	60000
Auto Commit:	True
Transaction Isolation:	DEFAULT
Validación de Conexión por Pool	
Keep-Alive:	True
Edad máxima de las Conexiones (ms):	5000
Query de Validación:	Select 1
Configuración de la Conexión a Base de Datos	
URL de la Base de Datos:	jdbc:mysql://localhost:3306/vidividi
Clase del Driver JDBC:	com.mysql.jdbc.Driver
Nombre de Usuario:	root
Password:	●●●●

Figura 153 Configuración de la Conexión JDBC

En la cual se probara la siguiente consulta:

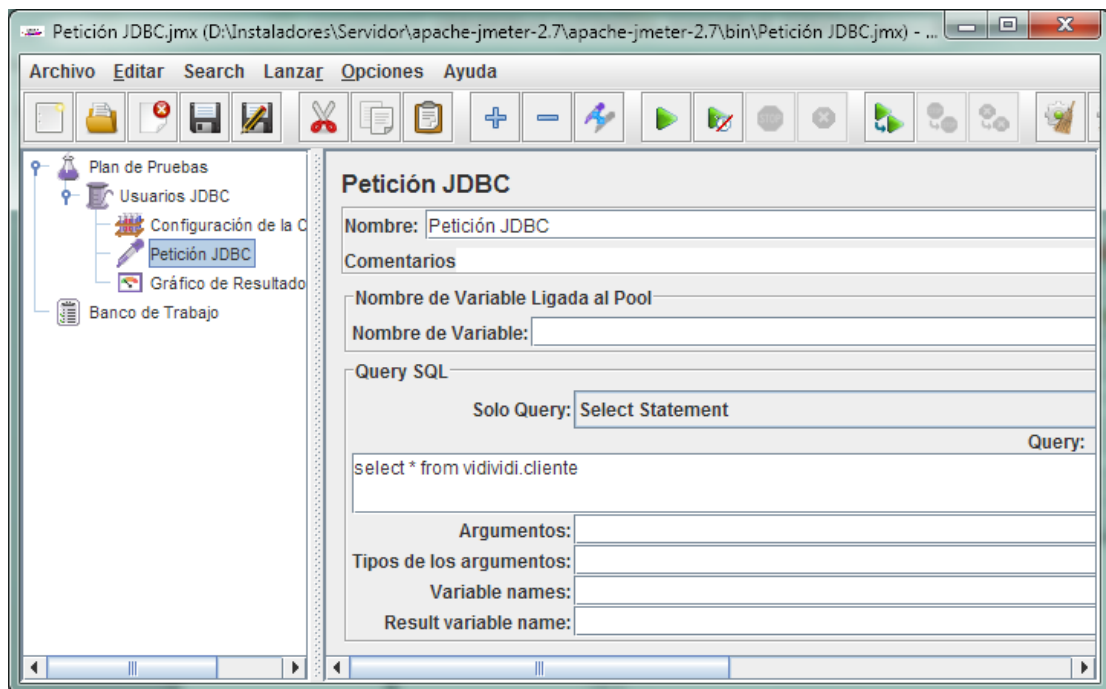


Figura 154 Petición JDBC

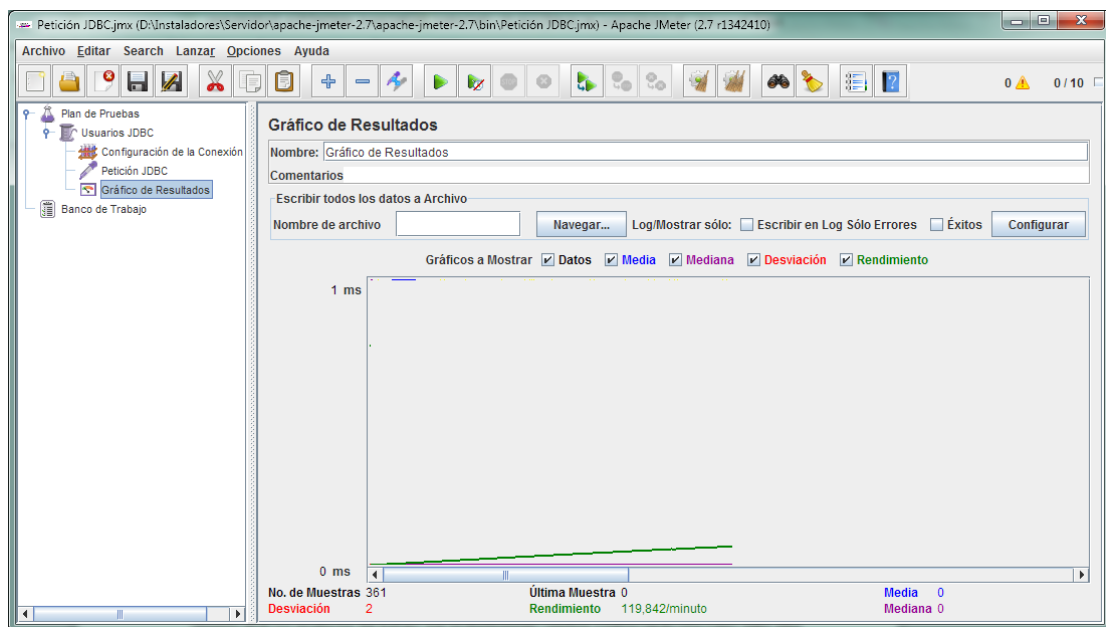


Figura 155 Gráfico de Resultado JDBC

Por conexión se puede realizar la consulta a una velocidad de 119,842 consultas por minuto.

2.1.2.2.3.19 Solicitud de Cambio

2.1.2.2.3.19.1 Introducción

Cuando construimos software de computadora surgen cambios. Debido a esto, necesitamos controlarlos eficazmente, en la cual es 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 éstos 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.

2.1.2.2.3.19.2 Descripción

Las solicitudes de cambio de 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.

2.1.2.2.3.19.3 Control de Solicitudes de Cambio

2.1.2.2.3.20 Lista de Riesgos

2.1.2.2.3.20.1 Introducción

2.1.2.2.3.20.1.1 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.

2.1.2.2.3.20.2 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.

2.1.2.2.3.20.3 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 97 Roles Encargados de Gestionar los Riesgos

2.1.2.2.3.20.4 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 98 Escala de Ponderación de los Riesgos

2.1.2.2.3.20.5 Lista de Riesgos

Categoría de Riesgo			Identificación de Riesgos		Análisis de Riesgo		Planificación de Riesgos			Supervisión de Riesgos
Proyecto	Producto	Negocio	Nº	Riesgo	Ponderación	Efecto	Estrategias		Plan de contingencia	Políticas de Supervisión
							Anulación	Disminución o Preventiv		

								a		
*	*	*	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 o 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 o a posibles cambios del personal de la institución.
*	*		3	Los usuarios rehúsan utilizar el	20%	Los usuarios no quieren que su sistema	-	*En conversación con los usuarios finales,	-	Realizar un control y seguimiento o a los

				sistema.		sea automatizado.		indicar las ventajas que presentara el software que se desarrollar a.		usuarios.
								*Preparar un manual de usuario comprensible que pueda estimular a los usuarios a usar el sistema a desarrollar .		
	*		4	Los usuarios no tengan conocimiento en el manejo del	50%	El sistema sea mal utilizado o solo utilizado en parte.	-	*Aconseja al usuario interesado a cerca de capacitación para	-	Realizar un control y seguimiento a los usuarios

				sistema.				realizarse en el presente o futuro.		en el manejo del sistema.
								*Prever para que el usuario final tenga el conocimiento suficiente para poder manejar el 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 o 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	-	Realizar un control y seguimiento a los posibles riesgos.

								de nuestro proyecto.		
								Obtener ayuda de expertos, libros y otros recursos que puedan ayudarnos en el desarrollo del plan de gestión de 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.	-	*Establecer con anticipación las interfaces y el entorno gráfico a utilizar. *Hacer uso de herramientas	-	Realizar un control y seguimiento a las interfaces o pantallas que vayan de acuerdo a la facilidad del usuario.

								as que puedan otorgar una mejor apariencia a las interfaces.		
								*Obtener conocimiento de buenas normas para el diseño de interfaces.		
*	*	*		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	-	Realizar un control y seguimiento al calendario del proyecto.

								que nos faciliten una correcta estimación .		
*		*	9	El coste del proyecto llegue a aumentarse durante el desarrollo.	25%	El proyecto puede cancelarse temporal o definitiva mente, por el increment o del precio para el usuario.	-	*Hacer uso de técnicas de Ingeniería de Software para realizar una buena estimación del coste. *Obtener informació n actualizad a del coste del desarrolló proyectos similares: personal,	-	Realizar un control y seguimient o a los costes del proyecto.

								recursos de hardware y otros.		
	*		10	La institución no cuenta 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	40%	La empresa quede perjudicada y no se presente el proyecto en la fecha	-	-	Buscar temporalmente un cajero	-

				el desarrollo del proyecto.		acordada (retraso).				
*	*		12	Posible Falla en algún dispositivo de almacenamiento físico al resguardar de documentación del proyecto	15%	Perdida de todo la información avanzada del proyecto.	-	Realizar respaldos y copias de seguridad en dispositivos de almacenamiento externo.	Acudir a software de recuperación de datos.	Establecer mantenimiento constante de los ordenadores en los que 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	-	Realizar un control y seguimiento a cada uno de los requerimientos.

								cliente.		
*	*		14	Posible corte de corriente eléctrica y/o desconexión a Internet	30%	Existe un porcentaje de error en el registro de pedidos y perjudicar á al seguimien to del flujo de caja	-	Revisar minuciosa mente y manualme nte cada registro asociado con las ventas, egresos e ingresos	Volver al método anterior de registro manual de pedidos en planilla s	Contar con un motor generador de corriente eléctrica

Tabla 99 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 del Proyecto

2.1.2.3.1 Participantes del Proyecto

Director del Proyecto: Alex Daniel Céspedes Gira

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 clientes y ventas. La tediosa carga de datos que se realiza bajo una conexión de Internet, 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 contara 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	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante

Sistemas	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 100 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 “VidiVirtual” Sistema Informático para Mejoramiento de Ventas que se está desarrollando.

2.1.2.4.1.3 Alcance

Este plan se limita al proyecto de gestión “VidiVirtual” Sistema Informático para Mejoramiento de Ventas.

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 sub-objetivos de la gestión

Se deben identificar las necesidades de información para el proyecto de gestión “Vidi Virtual” Mejoramiento en el control de ventas y difusión de información. 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	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 101 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 102 Métrica: 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 103 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 104 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:

$$K_m = 2.4 S_k^{1.05}$$

Donde:

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 llevara 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 solo 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:

$$UCP = \text{Punto de Caso de Uso}$$

$$E = UCP * 2 = \text{horas-hombre}$$

$$E = 132 * 2 = 264 \text{ horas-hombre}$$

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 105 Cálculo de Esfuerzo

El tiempo es de 264 horas – hombre por cada Caso de Uso.

En la cual Si realizamos el siguiente calculo

$$264 \text{ horas-hombre} / 24 \text{ horas} = 11 \text{ días por cada Caso de Uso}$$

Es una Estimación que no tiene mucha relevancia por considerar solo 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:

$$PF = CT * (0.65 + 0.1 * \sum Fi)$$

Donde:

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 106 Factor de ponderación

Valores de Ajuste de Complejidad

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significati vo	Esencial

Tabla 107 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 108 Parámetros a evaluar

$$PF = \text{Cuenta-Total} * [0.65 + (0.01 * S\text{-Total})]$$

$$PF = 351 * [0.65 + (0.01 * 54)]$$

$$PF = 417.69$$

$$PF = 418$$

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 109 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:

$$\text{LDC} = 120 * 418$$

$$\text{LDC} = 50160$$

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:

$$t = 8.14 (\text{LDC} / P)^{0.43} \quad (1)$$

$$E = 180 B t^3 \quad (2)$$

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:

Donde: 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 solo 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 110 Plan de las Fases

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 correspondientes a requisitos que serán implementados en la primera release 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.
Fase de Construcción	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.
Fase de Transición	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.

Tabla 111 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.

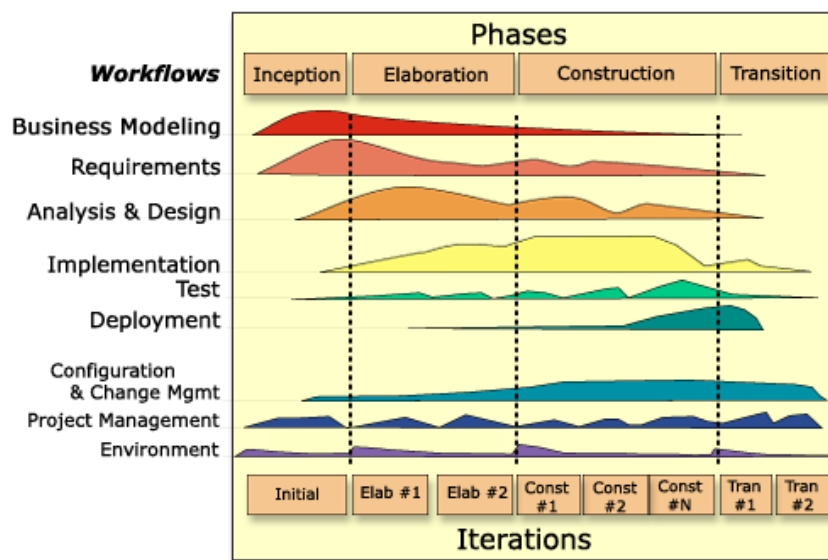


Figura 156 Calendario del proyecto

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 112 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 113 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 I: Especificación de Requisitos de Software según la norma IEEE830

2.1.2.4.7.2 Control de Plazos

Fase	Nº de Iteración	Inicio	Fin
Inicio	Primera	14-04-2011	31-05-2011
Elaboración	Primera	01-09-2011	03-04-2012

Construcción	Primera	03-04-2012	04-05-2012
Transición	Primera	05-05-2012	06-06-2012

Tabla 114 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 ocuparan 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 trabajara con el cliente para desarrollar un solo plan de prueba de aceptación. En otros casos el cliente desarrollara 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 realizara una revisión de verificación del software para evaluar cuan adecuados son los planes.

Durante la evolución del producto, se realizaran Auditorias en el proceso para verificar que los productos de trabajo sean consistentes y estén completos.

Los elementos que sufrirán auditoria por consistencia incluyen especificaciones de interfaces para hardware, software y personas; diseño interno contra especificaciones funcionales; código fuente contra documentación. En la práctica, solo ciertas porciones criticas del sistema pueden someterse a auditorias intensivas.

Antes de la entrega del sistema, se realizan una auditoria funcional y una auditoria 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, auditorias 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 trabajara 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	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 115 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	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 116 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.
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.	Discrepancias reportadas y solucionadas. Documentos de acuerdo a estándares.

Tabla 117 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 llevara 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 estas 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 Ventas

2.2 Componente 2: Socialización y Capacitación del Personal de la Empresa de Comida Rápida VidiVidi, 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 avanzadas 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 “Vidi Virtual” que ayuda a agilizar la centralización de la información de las ventas y su difusión.

2.2.2 Objetivos

2.2.2.1 Objetivo General

Lograr la participación y aportación por parte de las personas, haciendo uso de las nuevas tecnologías de Información y Comunicación

2.2.2.2 Objetivo Específico

- Fomentar la utilización del Sitio Web desarrollado.
- Preparar a los colaboradores para la ejecución de las diversas tareas y responsabilidades de la organización.

2.2.3 Aspectos Pedagógicos de la Socialización

2.2.3.1 Introducción

Recordando que la socialización es un proceso mediante el cual el individuo adopta los elementos socioculturales de su medio ambiente y los integra a su personalidad para adaptarse a la sociedad, por este motivo se pretende socializar el Proyecto con el objeto de generar flujos de comunicación y participación de la sociedad en nuestra cultura.

De esta forma el proyecto denominado “VidiVirtual” se fortalece y se garantiza un tiempo de vida prolongado.

2.2.3.2 Objetivos

Hacer conocer a la Población sobre la existencia del proyecto

Fomentar la utilización y participación de la sociedad en el Sitio Web diseñado.

2.2.3.3 Metodologías y Estrategias

La socialización se la realizara mediante Propaganda y Publicidad.

Las metodologías utilizadas están relacionadas a la situación económica y al impacto que tendrán en la sociedad, por que tomo la decisión de utilizar las siguientes:

Publicidad Directa

Es la que se pone en contacto (sin agentes intermediarios) con el consumidor.

Publicidad Indirecta

En cambio esta no se dirige personalmente a cada consumidor, sino que utiliza los medios de comunicación masiva que hacen llegar su mensaje a todo un sector de la comunidad.

Es este caso se utilizará afiches, y radio.

2.2.4 Aspectos Pedagógicos de la Capacitación

2.2.4.1 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.

2.2.4.2 Importancia de la Capacitación

Permite que:

- 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.

2.2.4.3 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

2.2.4.4 Metodología

Para realizar la capacitación acerca del uso del Sistema Informático “Vidi Virtual”, se hizo uso de técnicas de capacitación aplicada en la empresa, más propiamente la técnica conocida como capacitación en el puesto. Esta técnica contempla que una persona aprenda una responsabilidad mediante su desempeño real en el Sistema Informático “VidiVirtual” ya sea un *Administrador* o *Usuario*, aplicando esta metodología la capacitación se realizó en las instalaciones de la empresa VidiVidi.

Lo pasos que se contemplaron para realizar una capacitación en el puesto son:

- Preparación del aprendiz.
- Presentación del Sistema Informático “Vidi Virtual”
- Seguimiento

2.2.4.4.1 Contenido

El contenido de la capacitación se subdivide en 3 módulos la cuales se detallan a continuación:

- **Módulo 1: La Internet**

En esta unidad de aprendizaje se realiza una exposición, en donde el capacitador explica: sobre La Internet mediante los siguientes puntos:

1. Introducción
2. ¿De dónde proviene?
3. Estructura de La red Internet
4. Estrategia "Cliente/Servidor"
5. Internet y sociedad
6. ¿Qué nos ofrece La Internet?
7. Página Web
8. Tipos de Sitios Web
9. Como acceder a la Web
10. La Web social o Web 2.0

- **Módulo 2: Presentación del Sistema Informático Vidi Virtual**

En esta unidad de aprendizaje se hace la presentación del Sistema Informático “Vidi Virtual”.

- **Módulo 3: Capacitación del Sistema Vidi Virtual**

En esta unidad de aprendizaje se explica los pasos necesarios para administrar el sitio Web y ventas.

El contenido de la capacitación se realizó de acuerdo al siguiente cronograma:

Fecha	Hora	Actividad	Expositor
9 de Julio del 2012	15:00 – 16:30	Presentación de los participantes y Objetivo de	Alex D. Céspedes Gira

		la capacitación. Módulo 1: La Internet Módulo 2: Presentación y Manejo del Sistema Informático Vidi Virtual	
10 de Julio del 2012	15:00 – 16:00	Módulo 3: Administración del sistema Vidi Virtual	Alex D. Céspedes Gira

Tabla 118 Cronograma de la Capacitación

2.2.4.4.2 Medios a utilizar

2.2.4.4.2.1 Aspectos Técnicos

- 1 DataDisplay o monitor LCD.
- 1 Computador con el Sistema Desarrollado.
- Diapositivas de Exposición hechas en PowerPoint.

2.2.4.4.2.2 Aspectos Logísticos

- Material digital informativo (cds,dvds)

2.2.4.5 Conclusiones

Una vez realizado la capacitación y según la ejecución del contenido de la capacitación, se pudo diferenciar tipos de cambio en el personal, como por ejemplo el desarrollo de destreza y conocimiento de manera directa respecto al tema de Internet; además se desarrolló un cambio de actitud negativa por actitudes más favorables y positivas entre el personal de la Empresa de Comida Rápida VidiVidi en relación de la difusión de la información por medio de Internet

2.2.4.6 Medios de Verificación

- Certificado de capacitación en tecnologías de información y comunicación (TIC) para el mejoramiento del control de ventas y difusión de información de

la empresa de comida rápida VidiVidi”, firmados por el Departamento de Informática y por el director del proyecto.

- Fotografías tomadas durante las capacitaciones.
- La lista de Asistencia debidamente firmadas por los asistentes en cada seminario.

III. Capítulo III: Conclusiones y Recomendaciones

3.1 Conclusiones

Habiéndose culminado el presente proyecto, se puede concluir lo siguiente:

Se logró cumplir el propósito del proyecto, a través de su indicador reducido el número de errores en un 80% en los procesos asociados a la venta (pedido, entrega y cobro).

Por limitaciones de equipamiento y de alquiler de un hosting, aún queda pendiente la implementación del sistema, se puede afirmar el cumplimiento de todos los requerimientos de la empresa.

El primer componente combina un sistema WEB y uno de escritorio, integrados en mismo framework Spring, que gestionan la misma base de datos, con lo cual se garantiza la integridad y consistencia de datos de ambos módulos mejorando y agilizando el registro de pedidos, control de egresos e ingresos, proporcionando un portal Web para la mayor difusión de la información.

La elección RUP como metodología de desarrollo de software empleada, ha sido adecuada porque brindó gran flexibilidad de escalamiento en las iteraciones realizadas y se puede afirmar, en base a esta experiencia, que es muy apropiada para sistemas pequeños y de mediano alcance.

Todas las herramientas de software descritas en el documento han sido de gran utilidad habiendo facilitado el desarrollo del sistema en sus diferentes etapas. Es de destacar el uso de frameworks como Spring, Hibernate y Flex, que brindan un marco de desarrollo basado en patrones de diseño, lo cual garantiza la calidad del software.

Por último, se puede afirmar que las Tecnologías de Información y Comunicación (TIC) han sido definitivamente útiles y determinantes en el mejoramiento del control de ventas y difusión de la información de la empresa de comida rápida “VidiVidi”.

El componente de capacitación asegura que tanto el administrador del sistema, como el personal de “VidiVidi” operarán correctamente el sistema, reduciendo los errores prácticamente a cero.

3.2 Recomendaciones

Es altamente recomendable que la empresa adquiriera su propio servidor, el cual podría ser configurado y personalizado de acuerdo a las necesidades de la empresa.

Para optimizar el rendimiento del sistema, se recomienda implementar el sistema en equipos de marca con procesador Core 2 Duo o superior, con sistema operativo Linux preferentemente o Windows alternativamente. El Hosting deberá ser estar bajo la plataforma Linux.

Para la instalación, configuración, mantenimiento y realizaciones de copias de seguridad, se recomienda contratar un profesional informático, que, con conocimiento de causa, realizará estas actividades sin problemas y con el éxito asegurado.

Para la implementación y explotación del proyecto se recomienda realizar periódicamente copias de respaldo de información (backups), con el motivo de evitar pérdidas de información.

Se recomienda ampliar el sistema desarrollado para contemplar nuevos módulos que abarquen inventarios, contabilidad, administración de personal, considerando que el diseño es flexible y sujeto de escalamiento.

Por último se recomienda realizar la capacitación y socialización a cada nuevo cajero para que este pueda utilizar el sistema sin mayores complicaciones y de manera correcta.