

CAPITULO I

EL PROYECTO

1 CAPITULO I

1.1. PRESENTACION DEL PROYECTO

1.1.1 TITULO DEL PROYECTO

Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad municipal de aseo tarija ‘EMAT’

1.1.2 CARRERA / UNIDAD

Carrera de Ingeniería Informática

1.1.3 FACULTAD

Ciencias y tecnología

1.1.4 ÁREA/LÍNEA DE INVESTIGACIÓN PRIORIZADO

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

1.1.5 RESPONSABLE DEL PROYECTO

Carrera de Ingeniería Informática – Taller III

1.1.6 ENTIDAD ASOCIADA (S)

Entidad De Aseo Municipal Tarija ‘EMAT’

1.2 PERSONAL VINCULADO AL PROYECTO

1.2.1 DIRECTOR DEL PROYECTO

Martínez	Rueda	Mery Maya	7179280 Tja
Apellido Materno	Apellido Paterno	Nombre	CI

Estudiante Univ. Profesión	Ing. Informática Carrera o Unidad	Ciencias y Tecnología Facultad	
S/N Teléfono	78225828 Celular	meryruedamartinez@gmail.com Correo electrónico	Firma

Tabla No. 1 Director del proyecto

1.2.2 PARTICIPANTES DEL EQUIPO DE TRABAJO

Categoría	Nombres y Apellidos	Carrera/Profesión	C.I.	Firma
Director	Mery Maya Rueda Martínez	Estudiante en Ingeniería Informática	7179280Tja.	
Asesor	Ing. Silvana Paz R.	Lic. en Ingeniería Informática		
Tutor	Ing. Zulma Ricaldi	Lic. en Ingeniería Informática		

Tabla No. 2 Participantes del Equipo De Trabajo

1.2.3 EQUIPO DE TRABAJO DE: EMPRESAS / INSTITUCIONES/ ORGANIZACIONES PARTICIPANTES/COOPERANTES

Nombre: Entidad Municipal De Aseo Tarija			
Dirección: Av. Integración /Calle 8 de marzo		Teléf. Oficina: 466-37505	
Nombre y Apellidos	Cargo	C.I.	Firma
Lic. Rolando Ruiz Gallardo	Director De Entidad Municipal De Aseo Tarija EMAT	7172815	
Jhonny Mamani Rodriguez	Encargado de area de residuos solidos	7185807	

Tabla No. 3 Equipo de Trabajo Cooperante

1.2.4 ACTIVIDADES PREVISTAS PARA LOS INTEGRANTES DEL EQUIPO DE INVESTIGACIÓN

Responsable	Actividades
Director	Como Jefe de Proyecto: - Planificar y controlar el 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 de los aspectos tecnológicos para el desarrollo del proyecto. • Asesoramiento en el uso de la Metodología RUP de la herramienta UML. • Evaluación del documento del proyecto

	según agenda.
--	---------------

Tabla No. 4 Actividades Previstas Para Los Integrantes Del Equipo

1.2.4.1 UNIDADES DE GESTIÓN: ORGANIGRAMA DEL EQUIPO DEL PROYECTO

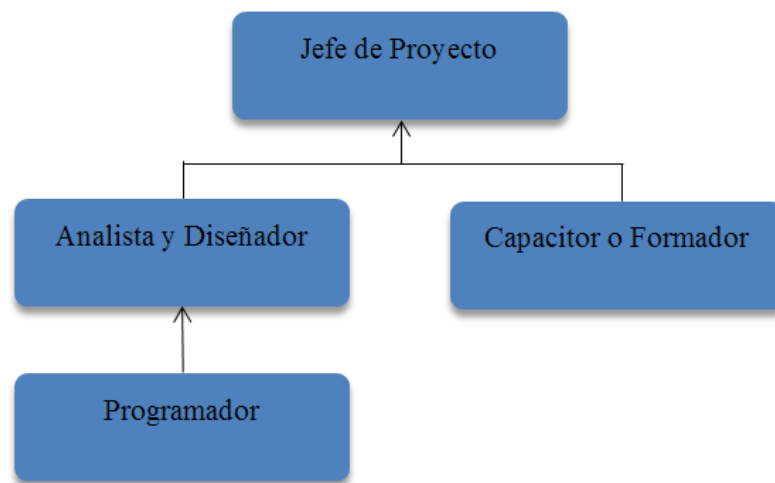


Figura 1 Organigrama del Equipo de Proyecto

1.3 DESCRIPCION DEL PROYECTO

1.3.1 RESUMEN EJECUTIVO DEL PROYECTO

La Empresa Municipal de Aseo de Tarija (EMAT) es una entidad que realiza el trabajo de limpieza de basura en beneficio de los barrios y algunas comunidades cercanas a la ciudad. La empresa de recojo de residuos sólidos fue creada en 1995 con el objetivo primordial de llevar adelante una buena gestión integral ambiental.

En el transcurso de los años la empresa fue evolucionando y mejorando el servicio, en un inicio se lograba una cobertura del 24 % de la ciudad y que ahora se tiene una cobertura del 98 %.

Las Tecnologías de la Información y la Comunicación, también conocidas como TIC, son el conjunto de tecnologías desarrolladas para gestionar información y enviarla de un lugar a otro. Abarcan un abanico de soluciones muy amplio. Incluyen las tecnologías para almacenar información y recuperarla después, enviar y recibir información de un sitio a otro, o procesar información para poder calcular resultados y elaborar informes.

El presente proyecto de investigación aplicada propone la inserción de las TIC en los procesos de control y seguimiento de la disposición final de residuos sólidos por EMAT desarrollando en primera instancia un modelo de workflow que optimice el registro de la bitácora, para que luego a través de un modelo tecnológico fortalecido con un sistema automatizado desarrollado con tecnología de punta, garantice un adecuado funcionamiento.

1.3.2 DESCRIPCIÓN Y FUNDAMENTACIÓN DEL PROYECTO

El proyecto de “Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad aseo municipal Tarija” EMAT “tiene como finalidad facilitar las tareas manuales que realiza la administración, como ser la elaboración de reportes del ingreso de basura al relleno sanitario por mes y gestión. Lo fundamental es coadyuvar a mejorar la administración de la entidad municipal de aseo Tarija.

El proyecto de “Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad aseo municipal Tarija” EMAT “

”, será logrado con el desarrollo e implementación de un sistema de control web.

El Sistema de Control Web permitirá:

- Realizar gestión de usuarios, choferes, ayudantes.
- Registro de bitácoras, vehículos, zonas, categorías, viajes, turnos,
- Elaboración de reportes del total de basura ingresada al relleno sanitario por mes, gestión y categorías.

Uno de los problemas percibidos en la parte administrativa, es la deficiente elaboración de reportes, también como consecuencia de imprecisiones se dan demoras en el cálculo y demora en la entrega de informes, generando también mal estar e inconformidad entre los encargados del área.

Existen varios problemas que contribuyen a esta deficiencia, están los llenados inadecuados e incompletos que llenan los choferes, se tiene problemas con los cálculos al realizar los reportes como consecuencia datos inexactos y demora en la revisión al momento de verificar.

La morosidad para elaborar los reportes hace que no se tenga información oportuna.

Para esto se realiza un sistema que automatice la información el cual reportara datos exactos en base al llenado de los formulados datos confiables y oportunos.

1.3.3 ARBOL DE PROBLEMAS

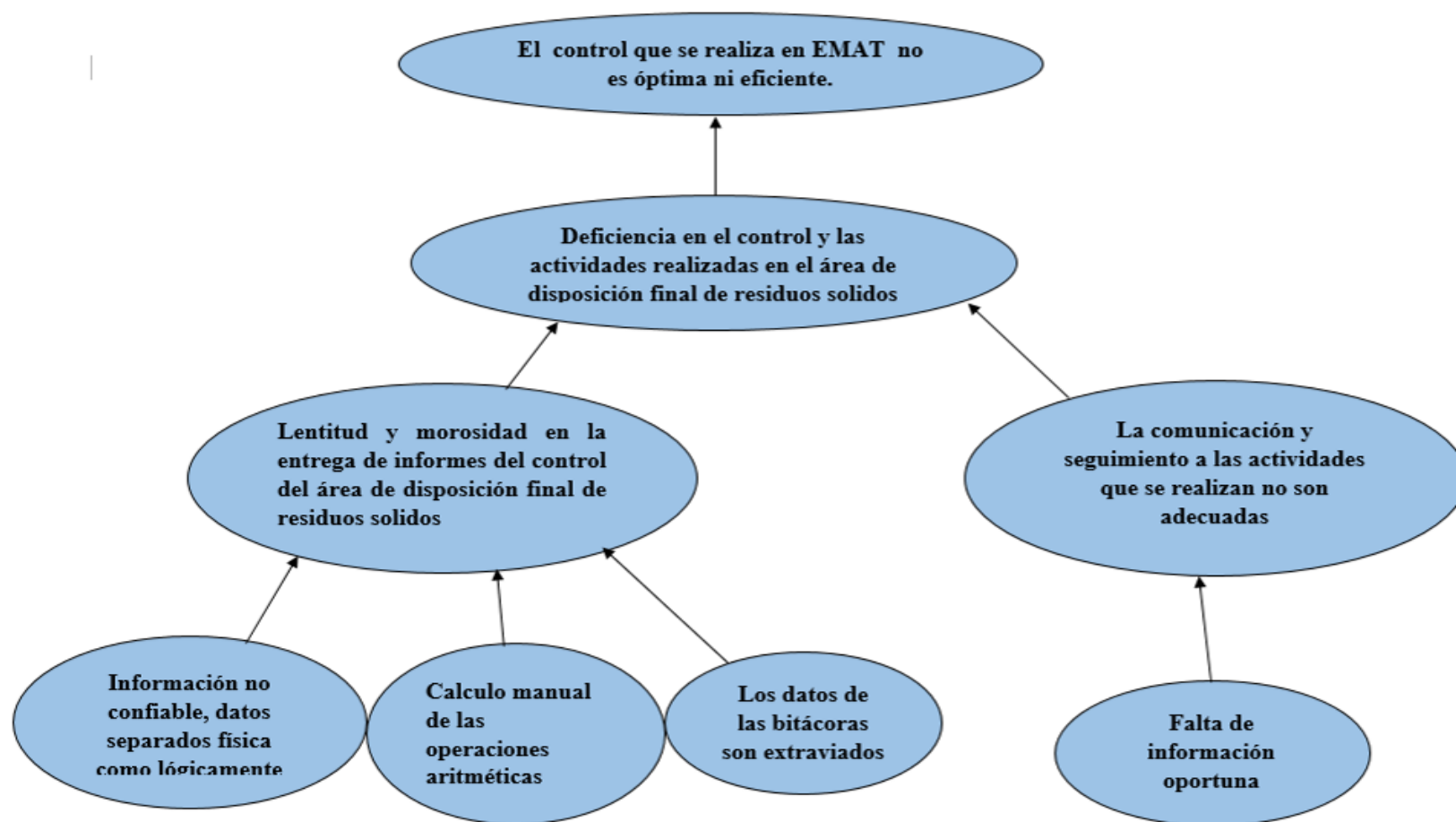


Figura No. 2 Árbol de Problemas

1.3.4 ARBOL DE OBJETIVOS

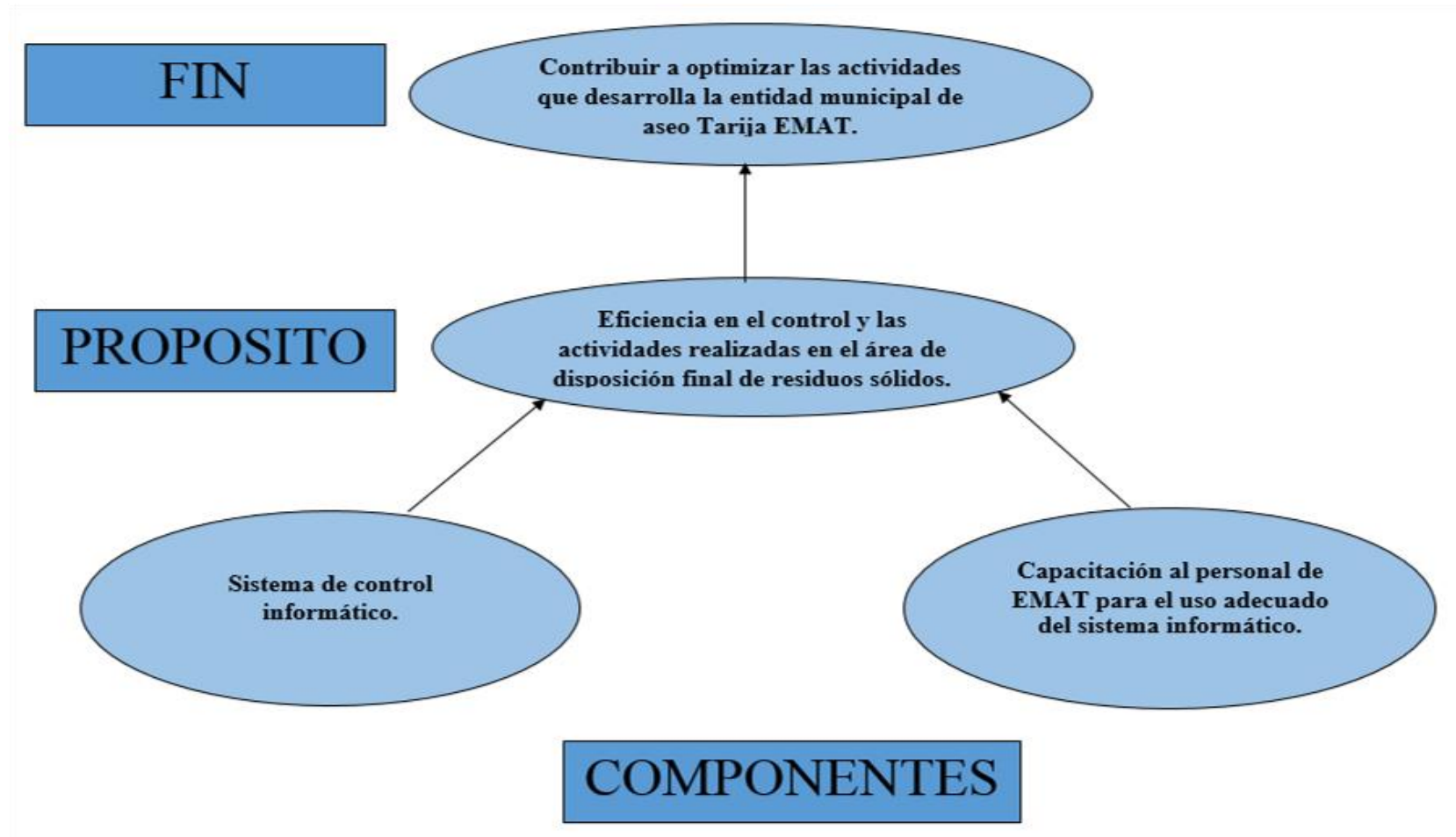


Figura No. 3 Árbol de Objetivos

1.3.5 OBJETIVOS

1.3.5.1 OBJETIVO GENERAL

Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad de aseo municipal Tarija ‘EMAT’

1.3.5.2 OBJETIVOS ESPECÍFICOS

- Desarrollar un sistema de control de para el área de disposición final de residuos sólidos.
- Capacitar el personal del área de disposición final de residuos sólidos para el manejo del Sistema.

1.3.6 METODOLOGÍA DE TRABAJO

1.3.6.1 METODOLOGÍA PARA EL DESARROLLO DE LAS APLICACIONES

COMPONENTE 1

Para el proceso de desarrollo del componente sistema 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.

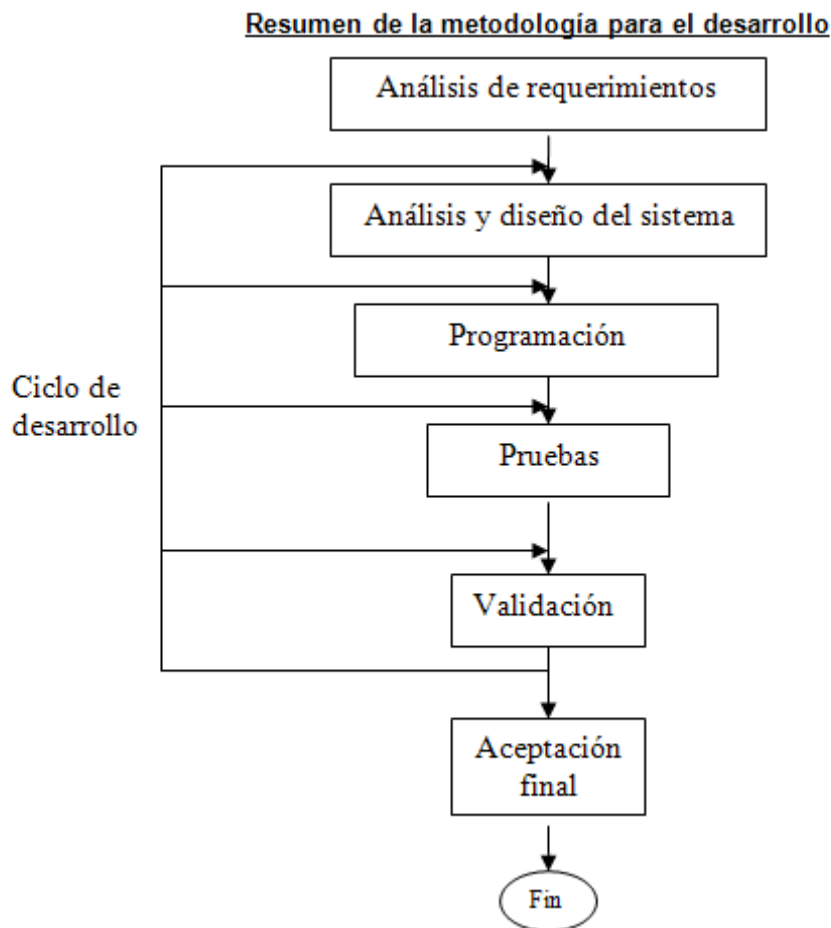


Figura 2 Resumen de la Metodología para el Desarrollo

1.3.6.2 METODOLOGÍA PARA LAS CAPACITACIONES Y/O TALLERES

COMPONETE 2

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: Capacitaciones Individuales.

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 Práctico

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.

1.3.7 RESULTADOS ESPERADOS

Los resultados esperados son los siguientes componentes.

- El proyecto tendrá un impacto importante en el desempeño administrativo en la entidad municipal de aseo Tarija EMAT., mayor eficiencia en la elaboración de reportes e informes finales para la toma de decisiones futuras.
- El sistema registrará a los trabajadores, los vehículos, las zonas por donde pasan y la cantidad de basura que ingresa al relleno sanitario expresado en toneladas se accederá a través de roles para darle mayor seguridad y evitar la manipulación de datos de forma indebida, generará reportes y elaborará planilla de informes.
- A través de la capacitación del personal se espera el uso adecuado y correcto del sistema. Asegurando el éxito del proyecto.

1.3.7.1 TRANSFERENCIA DE RESULTADOS

1.3.7.1.1 MEDIOS Y ESTRATEGIAS PARA LA TRANSFERENCIA DE RESULTADOS

Presentación final del sistema informático al Área de Disposición Final de Residuos Sólidos de la Entidad Municipal de Aseo Tarija, EMAT.

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

Socialización del producto final con el personal involucrado.

Los resultados del proyecto (componentes) serán transferidos previas pruebas y verificación del cumplimiento de normas de calidad de software y verificación de parte de la gerencia de la entidad municipal de aseo Tarija EMAT

1.3.8 GRUPO DE BENEFICIARIOS DE LOS RESULTADOS

Personal administrativo

Encargado De Área De Residuos Solidos

Director General De La Entidad Municipal De Aseo Tarija EMAT.

1.3.10 MARCO LOGICO DEL PROYECTO

Resumen Narrativo del Proyecto	Indicadores	Medios de Verificación	Supuestos
Fin Contribuir a optimizar las actividades que desarrolla la entidad municipal de aseo Tarija EMAT	Al finalizar el primer año de la implementación del proyecto para la entidad EMAT; la información que genera la unidad es correcta en un 80% y está disponible en forma oportuna.	Informe otorgado por el director general de la entidad municipal de aseo Tarija EMAT para medir la satisfacción, eficiencia y eficacia del sistema.	Se mantienen las condiciones tecnológicas y administrativas necesarias para el funcionamiento del sistema.
Propósito Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad municipal de aseo Tarija EMAT a través de las TIC.	Al finalizar el proyecto se ha automatizado el 80% de procesos del área de disposición final de residuos sólidos. La información es eficiente y óptima a la hora de tomar decisiones. $\frac{\text{No procesos automatizados}}{\text{No total de procesos}} \times 100$	Informe de conformidad con el sistema por el encargado de área, de disposición final de residuos sólidos.	Presupuesto suficiente para la implementación y uso del Sistema Web

Componentes			
I. “ Sistema Informático vía Web, desarrollado”	Al finalizar el proyecto se ha concluido el desarrollo del sistema informático en un 100 % de acuerdo a las especificaciones de requerimiento cumpliendo con lo especificado en el documento de la determinación de requerimientos de acuerdo a la norma IEEE 830.	Informe de cumplimiento del sistema firmado por el encargado de área.	Disponibilidad de Equipos de computación y material de oficina.
II. Capacitación del personal para el uso del Sistema informático	A finales de noviembre del 2015 el 80% del personal administrativo de EMAT participaron en la capacitación del uso del sistema informático en la cual se explicó el uso y las ventajas del Sistema de control del área de disposición final de residuos sólidos.	Lista de asistencia firmada por los asistentes.	El personal tendrá interés en la capacitación del Sistema

Actividades			
Sistema Informático	Actividades	Documentación del análisis y diseño del sistema.	Disponibilidad de Equipos de computación.
1) Especificación de requerimientos	A fines de febrero de la gestión 2015 se culmina con el 80 % de los requerimientos planteados.		
2) Análisis y Diseño del Sistema.	A fines de marzo de la gestión 2015 se culmina con el 80 % del análisis y diseño del sistema.	Utilización de las normas de modelado a los diagramas de análisis y diseño del sistema, mediante la metodología de desarrollo de sistemas RUP.	Existencia y disponibilidad de recursos económicos.
3) Programación del Sistema.	A fines de junio de la gestión 2015 se culmina con el 70 % de la programación del sistema.		Disponibilidad del uso de medios de transporte motorizados
4) Configuración e implementación del sistema al Servidor	A fines de agosto de la gestión 2015 se culmina con el 75 % de la configuración del sistema al servidor.	Verificación de integridad referencial de la base de datos.	
5) Elaboración del Informe final sobre el Desarrollo del Sistema.	A fines de octubre de la gestión 2015 se culmina con el 80 % de la elaboración del informe final		

Capacitación al Personal 1. Definición de metodologías de enseñanza para la socialización 2. Planificación del contenido temático de los cursos de capacitación.	A fines de mayo se definió el 80%de la metodología de capacitación para el sistema. A fines de mayo se realizó el cronograma del contenido de la capacitación al personal en un 100%	Manual de usuario. Manual de instalación.	El personal tendrá interés en la capacitación del Sistema
--	--	---	--

Tabla No. 5 Marco Lógico Del Proyecto

1.4 PRESUPUESTO / JUSTIFICACIÓN

ITEM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)
10000	SERVICIOS PERSONALES			
	12000 Empleados no Permanentes			20800
	Sub total rubro			20800
20000	SERVICIOS NO PERSONALES			
	21000. Servicios Básicos			5400
	22000. Servicios de transporte			1000
	25000. Servicios Profesionales y Comerciales			200
	Sub total rubro			6600
30000	MATERIALES Y SUMINISTROS			
	32000. Productos de Papel, Cartón e Impresos			175
	39000. Productos Varios.			130
	Sub total rubro			305
40000	ACTIVOS REALES			
	43000. Maquinaria y Equipo.			260
	49000. Otros Activos			1200
	Sub total rubro			1460
	TOTAL			29165

Tabla 6 Presupuesto / Justificación

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

Partida	Personal	Remuneración	Tiempo/meses	Total
12100	Ingeniero Junior	2600	8	20800
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 meses	Costo Total
21100	Comunicación	515	8	4120
21200	Energía Eléctrica	60	8	480
21300	Agua	70	8	560
21400	Servicios Telefónicos	30	8	240
Total				5400

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	N° de viajes	Costo unitario	Costo total
22100	Pasajes		250	4	1000
Total					

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

f) SUB GRUPO 25000. Descripción de los gastos en servicios profesionales y comerciales

Partida	Tipo de servicio profesional y comercial	Cantidad	Costo unitario	Tiempo mes	Costo total
25500	Publicidad		50	1	50
25600	Capacitación de Personal	10	15	1	150
Total					200

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

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
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.

k) SUB GRUPO 39000. Descripción del gasto en productos varios

Partida	Productos de cuero y caucho	Cantidad	Costo/Unitario	Total
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
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
Total				260

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

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
Total				1200

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

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

1.4.1 CURRICULUM VITAE

1.4.1.1 ANTECEDENTES PERSONALES

Rueda Apellido Paterno	Martinez Apellido Materno	Mery Maya Nombre	7179280 Tja. C.I.
02/12/1991 Fecha de Nacimiento	M..... F...X.... Sexo	B/ Los Chapacos C/ Bella Vista Nro. 236 Dirección	

Tabla 7 Antecedentes Personales

1.4.1.2 ANTECEDENTES ACADÉMICOS

Título obtenido	Universidad	País	Año
Bachiller en Humanidades Educación Superior.	“Unidad Educativa Liceo Tarija”.	Bolivia	2008
Ing. Informática - 1º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 2º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2009
Ing. Informática - 3º	“Universidad Autónoma Juan	Bolivia	2010

Semestre	Misael Saracho”.		
Ing. Informática - 4º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2010
Ing. Informática - 5º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2011
Ing. Informática - 6º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2011
Ing. Informática - 7º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2012
Ing. Informática - 8º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2012
Ing. Informática - 9º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2013
Ing. Informática - 10º Semestre	“Universidad Autónoma Juan Misael Saracho”.	Bolivia	2013

Tabla 8 Antecedentes Académicos

1.4.1.3 CURSOS REALIZADOS

Curso	Institución	Duración en Horas	Año
Capacitador para referéndum nacional constituyente	Corte Departamental Electoral Tarija	80	2009
Ensamblaje y mantenimiento de computadoras	Universidad Autónoma Juan	20	2010

	Misael Saracho		
Técnico en ensamblaje, mantenimiento de computadoras y en redes	Centro formación municipal la florida	600	2013
XXI Congreso Nacional de ciencias de la computación – CCBOL 2014	Universidad Autónoma Juan Misael Saracho	60	2014

Tabla 9 Cursos Realizado

CAPITULO II

COMPONENTES

CAPÍTULO II: COMPONENTES

2.1 COMPONENTE 1: SISTEMA INFORMÁTICO VÍA WEB, DESARROLLADO.

2.1.1 PATRON DE DISEÑO DE SOFTWARE

Patrón de arquitectura Modelo Vista Controlador (MVC)

El patrón de arquitectura MVC (Modelo Vista Controlador) es un patrón que define la organización independiente del Modelo (Objetos de Negocio), la Vista (interfaz con el usuario u otro sistema) y el Controlador (controlador del workflow de la aplicación la lógica interna).

El patrón de arquitectura "modelo vista controlador", es una filosofía de diseño de aplicaciones, compuesta por:

-MODELO

Contiene el núcleo de la funcionalidad (dominio) de la aplicación.

Encapsula el estado de la aplicación.

No sabe nada / independiente del Controlador y la Vista.

-VISTA

Es la presentación del Modelo.

Puede acceder al Modelo pero nunca cambiar su estado.

Puede ser notificada cuando hay un cambio de estado en el Modelo.

-CONTROLADOR

Reacciona a la petición del Cliente, ejecutando la acción adecuada y creando el modelo pertinente

Para entender cómo funciona nuestro patrón Modelo vista controlador, se debe entender la división a través del conjunto de estos tres elementos y como estos componentes se comunican unos con los otros y con otras vistas y controladores externos a el modelo principal. Para ello, es importante saber que el controlador interpreta las entradas del usuario (tanto teclado como el ratón), enviado el mensaje de acción al modelo y a la vista para que se proceda con los cambios que se consideren adecuados

2.1.2 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

2.1.3 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.

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

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

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

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

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

d) UML ofrece notación y semántica estándar:

UML prescribe una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos. Previamente, un diseño orientado a objetos podría haber sido modelado con cualquiera de las docenas 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).

2.1.4 TIPOS DE DIAGRAMAS UTILIZADOS

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.

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.

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.

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.

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.

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.

2.1.5 HERRAMIENTAS DE CONSTRUCCIÓN DE SOFTWARE

JAVA

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos y basado en clases que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra.

Java es un lenguaje originalmente desarrollado por un grupo de ingenieros de Sun, utilizado por Netscape posteriormente como base para Java script. Si bien su uso se destaca en el

Web, sirve para crear todo tipo de aplicaciones (locales, intranet o internet). Java es un lenguaje de objetos, independiente de la plataforma.

SPRING

SPRING es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.

La primera versión fue escrita por Rod Johnson, quien lo lanzó junto a la publicación de su libro *Expert One-on-One J2EE Design and Development* (Wrox Press, octubre 2002). El framework fue lanzado inicialmente bajo la licencia Apache 2.0 en junio de 2003. El primer gran lanzamiento fue la versión 1.0, que apareció en marzo de 2004 y fue seguida por otros hitos en septiembre de 2004 y marzo de 2005.

Si bien las características fundamentales de Spring Framework pueden ser usadas en cualquier aplicación desarrollada en Java, existen variadas extensiones para la construcción de aplicaciones web sobre la plataforma Java EE. A pesar de que no impone ningún modelo de programación en particular, este framework se ha vuelto popular en la comunidad al ser considerado una alternativa, sustituto, e incluso un complemento al modelo EJB (Enterprise JavaBean).

PostgreSQL

PostgreSQL es un SGBD relacional orientado a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no desarrolladores que trabajan de formas desinteresadas, libres y apoyadas por organizaciones comerciales. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

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

Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y

organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG.

NAVICAT

Navicat es una de las herramientas más fiables y rápidas para la administración de bases de datos, que te permitirán simplificar la gestión de las bases de datos y reducir los costes de administración.

Diseñado para satisfacer las necesidades de los administradores de bases de datos, desarrolladores y pequeñas y medianas empresas.

Navicat dispone de una interfaz gráfica muy intuitiva, que le permitirá crear, organizar, acceder y compartir información de forma fácil y segura.

APACHE TOMCAT

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

Tomcat es un contenedor web con soporte de servlets y JSPs. Tomcat no es un servidor de aplicaciones, como JBoss o JOnAS. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets. El motor de servlets de Tomcat a menudo se presenta en combinación con el servidor web Apache.

Tomcat puede funcionar como servidor web por sí mismo. En sus inicios existió la percepción de que el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones.

ENTERPRICE ARCHITECT

Enterprise Architect es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multi-

usuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad.

IREPORT-5.5.0

Entorno de desarrollo para crear plantillas de reportes para java en formatos como Word, Excel, html y pdf. Su entorno grafico facilita al usuario a la hora de crear las plantillas.

2.1.6 PLAN DE DESARROLLO DEL SOFTWARE

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 la universitaria: Mery Maya Rueda Martínez basado en una metodología de RationalUnifiedProcess (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.

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.

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.

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.

VISTA GENERAL DEL PROYECTO

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 que se encargan de la parte administrativa, sobre todo del área de recursos humanos.

PROPÓSITO

Con el presente proyecto se pretende contribuir a optimizar los procedimientos administrativos que permitan la utilización adecuada racional y transparente de los recursos de información oportuna y clara por EMAT, para su mejor funcionamiento.

ALCANCE

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo de EMAT un Sistema Informático que está comprendido por las siguientes gestiones:

- Gestión Personal.
- Gestión Bitácoras.
- Gestión Zonas.
- Gestión Viajes.
- Gestión Vehículos.
- Gestión Reportes.

OBJETIVO GENERAL

Contribuir a optimizar los procedimientos administrativos que permitan la utilización adecuada racional y transparente de los recursos de información instantánea y confiable, por EMAT para su mejor funcionamiento.

OBJETIVO ESPECÍFICO

- Desarrollar un Sistema Informático para la automatización de los procesos.
- Optimizar los procesos y procedimientos del control de las actividades, para la utilización adecuada, racional y transparente de la información oportuna, de EMAT, para su mejor funcionamiento y contribuir en su fortalecimiento Institucional.
- Mejorar y automatizar las transacciones del control de las actividades que se desempeña en la EMAT.
- Contribuir en la entrega de la información oportunamente al consejo administrativo para su correspondiente análisis y toma de decisiones para la próxima gestión.

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.

SUPOSICIONES

Suponemos que la entidad municipal de aseo Tarija EMAT, cuenta con:

- La empresa dispone equipos de computación, que incluye un computador e 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.
- La formación del personal encargado se lleve en la fecha determinada.
- Un servidor Web para la implementación del sistema.

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.

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.

2.1.6.1 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.

DIAGRAMA DE OBJETOS DEL NEGOCIO

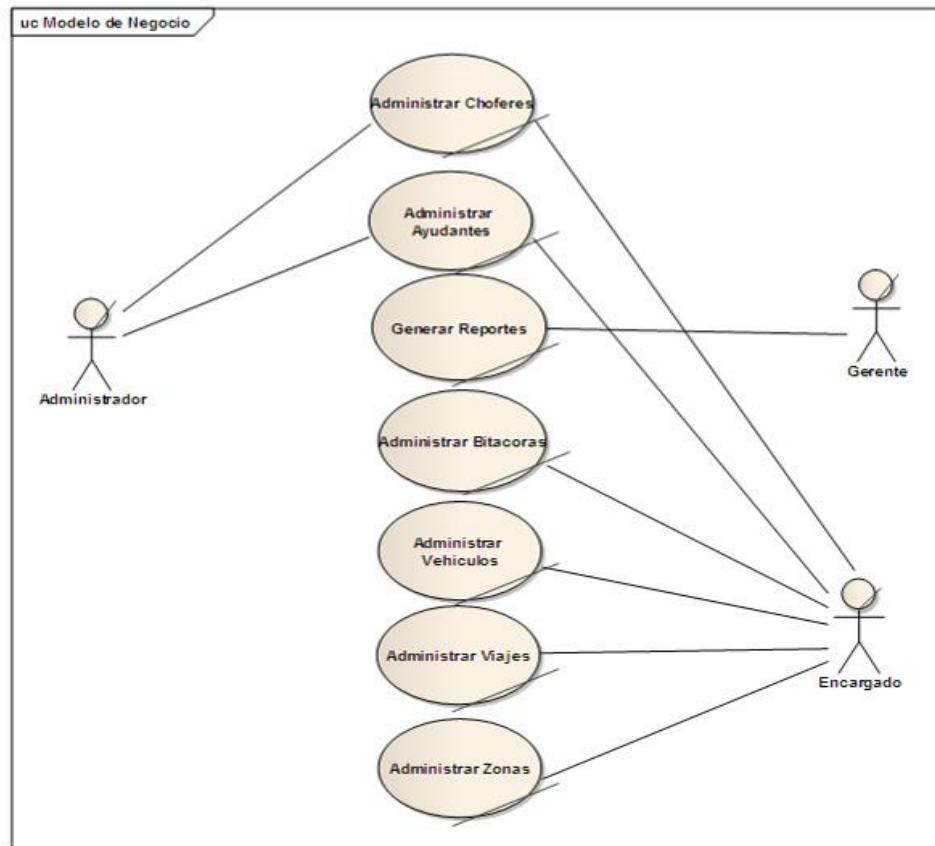


Figura No. 17 Diagrama de Objetos del Negocio

DESCRIPCIÓN DE LOS CASOS DEL USO DEL NEGOCIO

Nombre	Administrar Choferes
Descripción	El administrador y/o Encargado realizan el registro de los choferes al sistema. El administrador y/o Encargado puede modificar, eliminar y ver al personal de la empresa.
Actores	Administrador , Encargado

Tabla11 Especificacion Objeto De Negocio Administrar Choferes

Nombre	Administrar Ayudantes
Descripción	El administrador/ Encargado realizan el registro de los ayudantes. El administrador/Encargado puede modificar o eliminar al personal.
Actores	Administrador , Encargado

Tabla 12 Especificacion Objeto De Negocio Administrar Ayudantes

Nombre	Administrar Bitácoras
Descripción	El encargado realiza el registro de la bitácora al sistema. La bitácora esta previamente llenada manualmente. Se puede modificar, eliminar y ver datos de la bitácora.
Actores	Encargado

Tabla 13 Especificacion Objeto De Negocio Administrar Bitácoras

Nombre	Generar Reportes
Descripción	El gerente ve los todos los reportes que existen tales como ser: 1. Reporte de choferes 2. Reporte de ayudantes 3. Reporte de zonas 4. Reporte de vehículos 5. Reporte de bitácoras 6. Reporte de bitácoras/diurno por año

	7. Reporte de bitácoras/nocturno por año 8. Reporte de bitácoras/diurno por mes y año 9. Reporte de bitácoras/nocturno por mes y año 10. Reporte de bitácora/Total por año y categoria 11. Reporte de bitácora/total por mes , año y actegoria
Actores	Gerente

Tabla 14 Especificacion Objeto De Negocio Generar Reportes

Nombre	Administrar Vehiculos
Descripción	El encargado registra los vehículos con los que se cuentan en la empresa. Este mismo puede modificar, eliminar y ver datos del vehiculo.
Actores	Encargado

Tabla 15 Especificacion Objeto De Negocio Administrar Vehiculos

Nombre	Administrar Viajes
Descripción	El encargado realiza los registros de los viajes que se realizan por bitácora. Se cuenta con las opciones modificar, eliminar y ver datos del viaje.
Actores	Encargado

Tabla 16 Especificacion Objeto De Negocio Administrar Viajes

Nombre	Administrar Zonas
Descripción	Se registrara todas las zonas por las cuales se brindan los sericios de EMAT. Se contara con las opciones modificar, eliminar y ver datos de la zona.
Actores	Encargado

Tabla 17 Especificacion Objeto De Negocio Administrar Zonas

2.1.6.2 MODELO DE CASOS DE USO DEL SISTEMA

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

DIAGRAMAS DE CASOS DE USO

DIAGRAMA DE CASOS DE USO – GENERAL

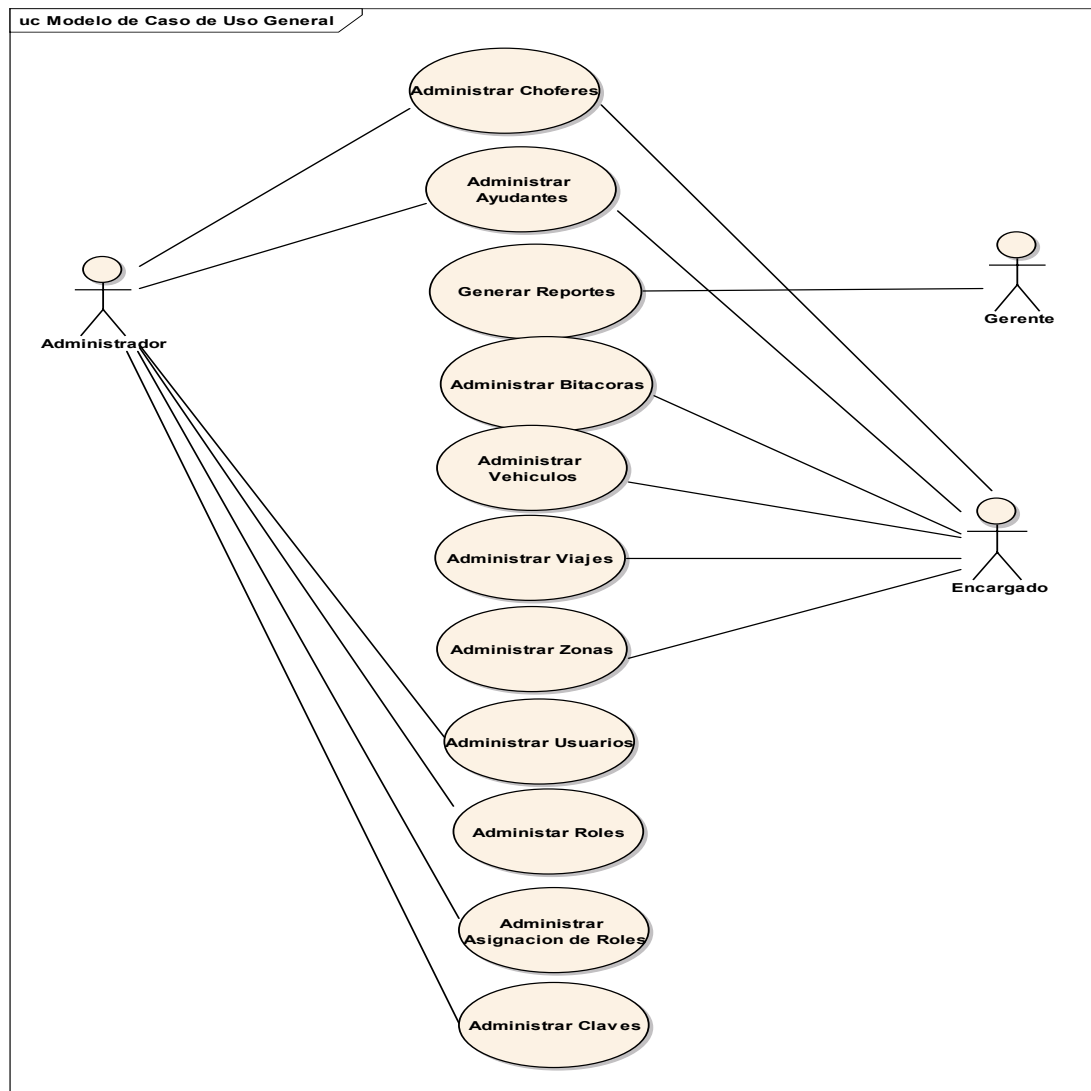


Figura No. 18 Casos De Uso General

MODELOS DE CASOS DE USOS ESPECÍFICOS

MODELO CASO DE USO ACCESO AL SISTEMA

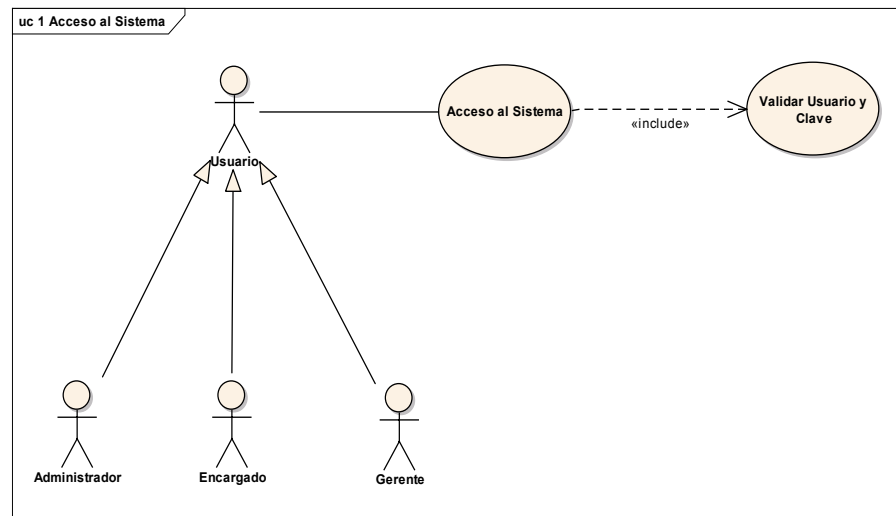


Figura 19 Modelo Caso de Uso Acceso al Sistema

MODELO CASO DE USO ADMINISTRAR USUARIOS

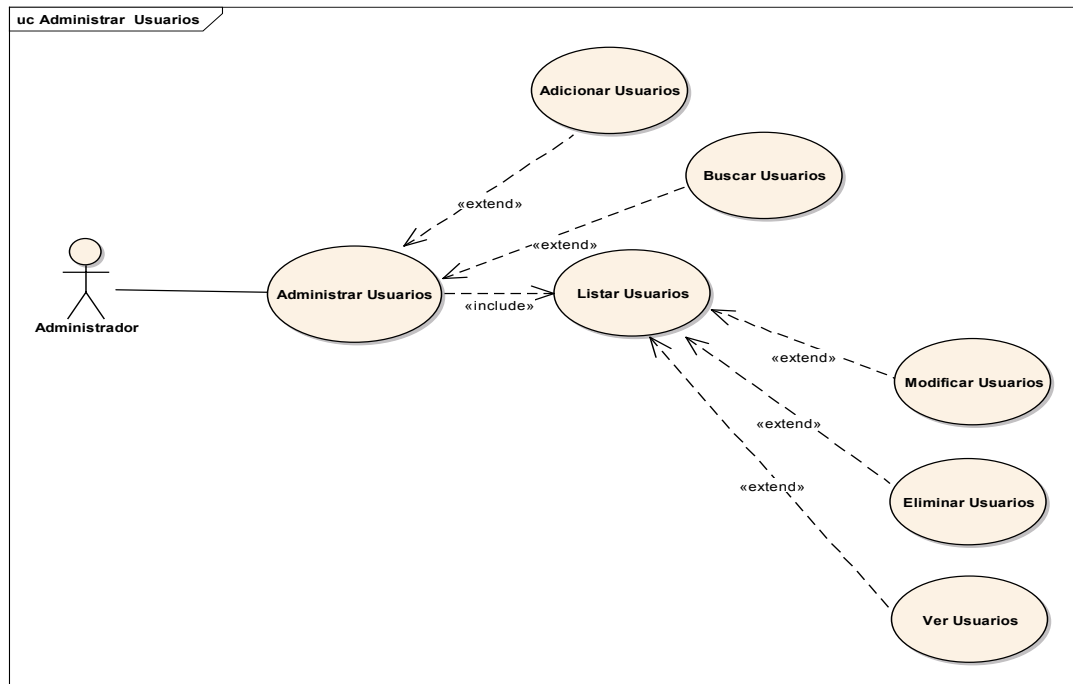


Figura 20 Modelo Caso de Uso Administrar Usuarios

MODELO CASO DE USO ADMINISTRAR ROL

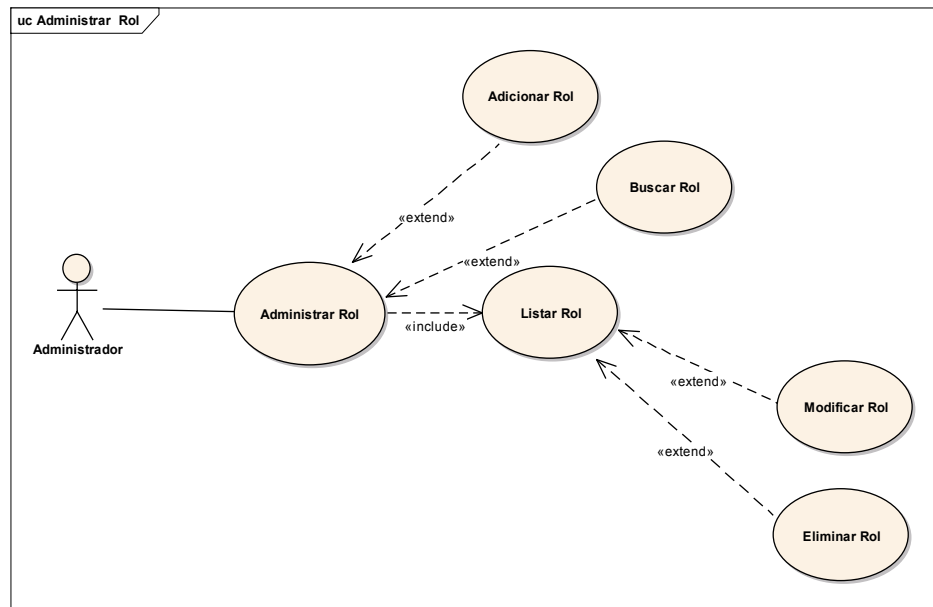


Figura 21 Modelo Caso de Uso Administrar Rol

MODELO CASO DE USO ADMINISTRAR ASIGNACION DE ROLES

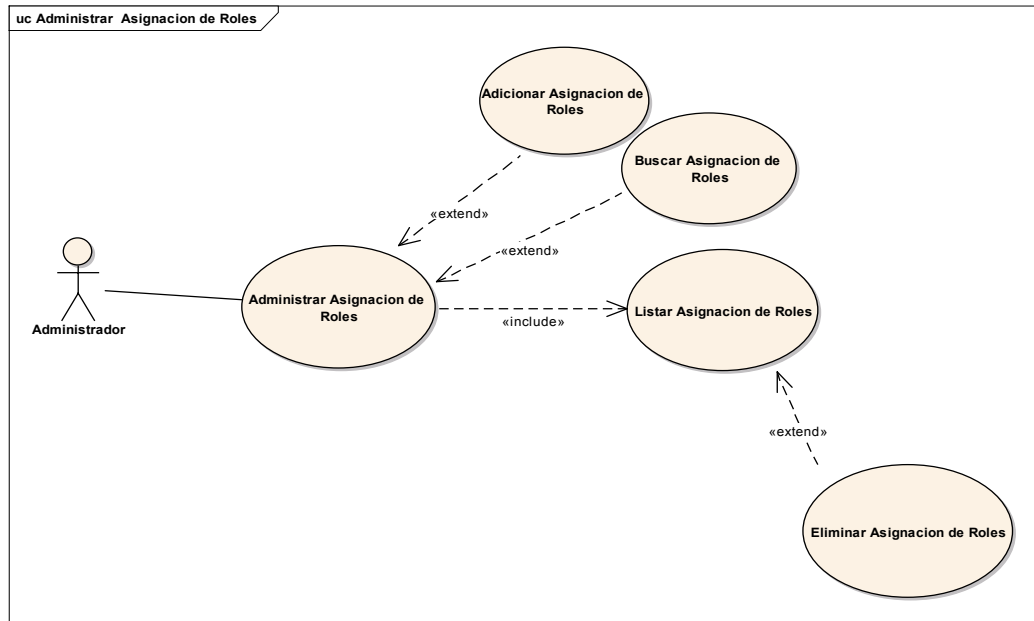


Figura 22 Modelo Caso de Uso Administrar Asignacion de Roles

MODELO CASO DE USO ADMINISTRAR CLAVES

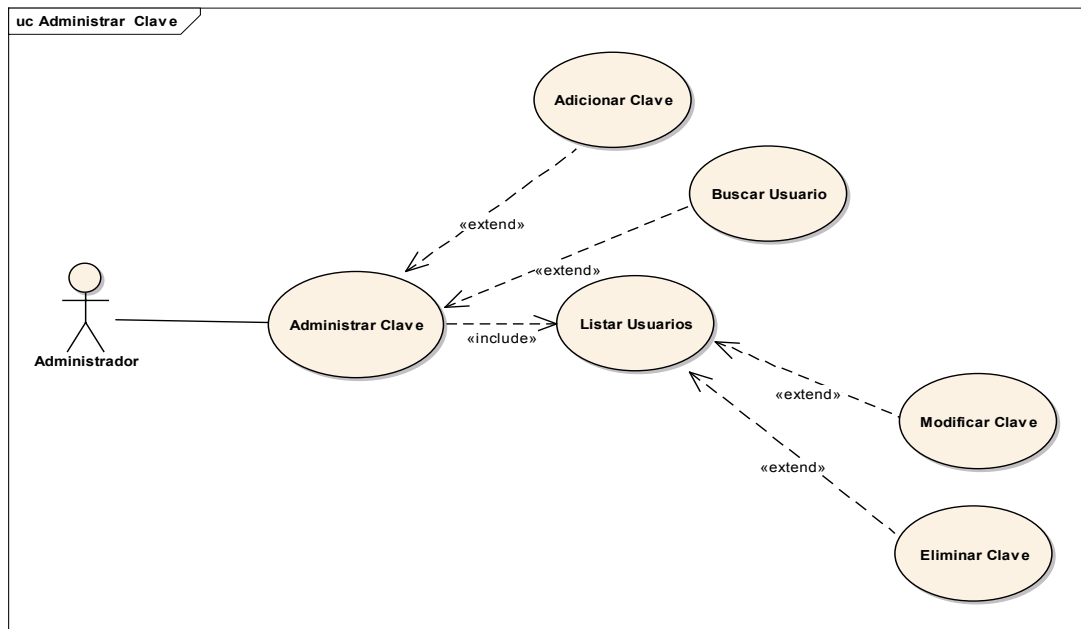


Figura 23 Modelo Caso de Uso Administrar Claves

MODELO CASO DE USO ADMINISTRAR CHOFERES

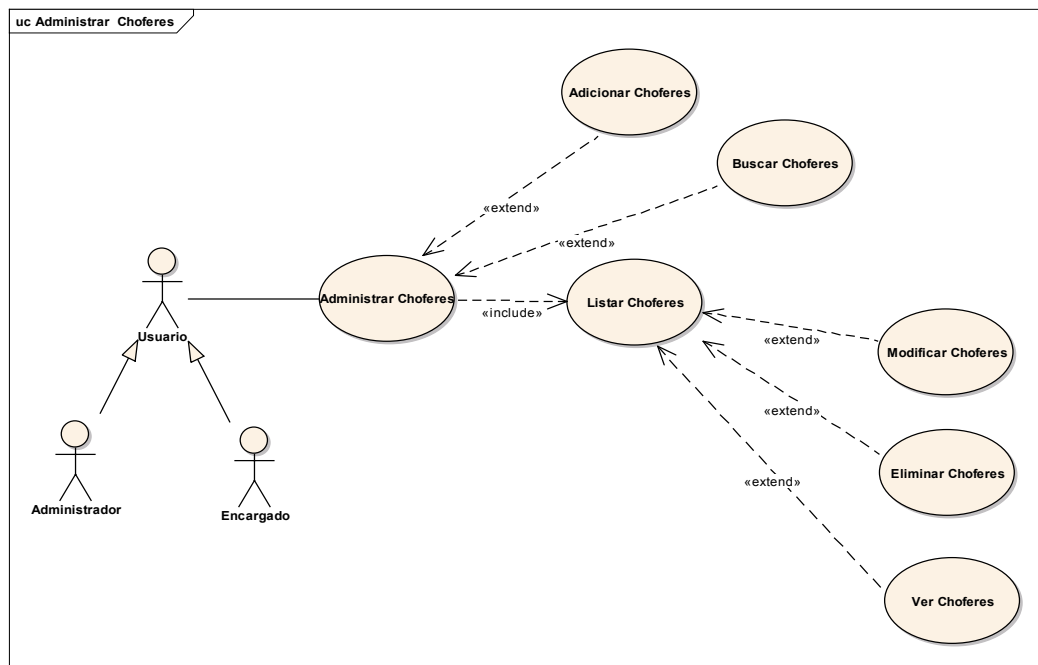


Figura 24 Modelo Caso de Uso Administrar Choferes

MODELO CASO DE USO ADMINISTRAR AYUDANTES

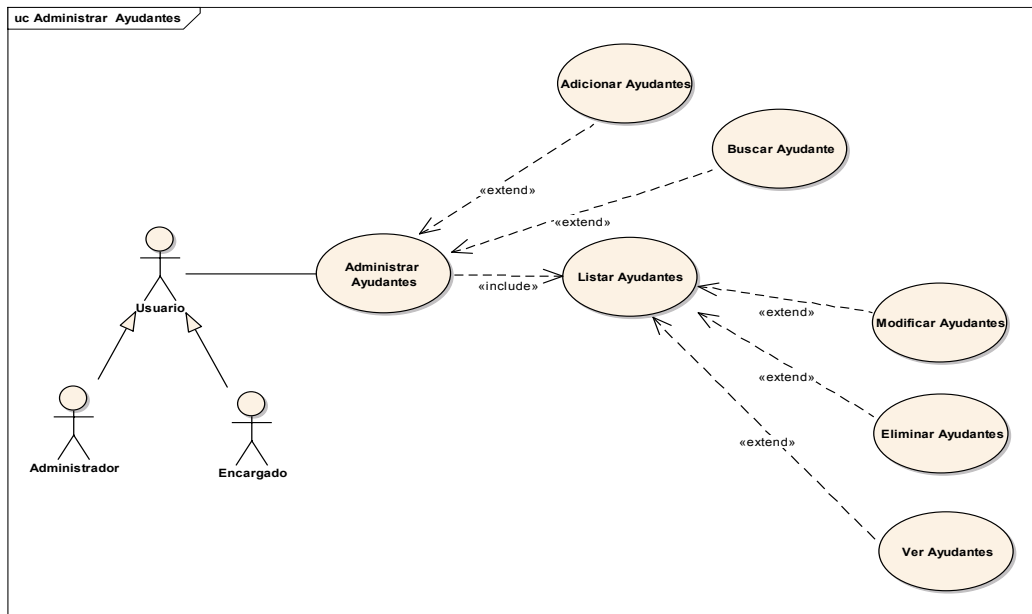


Figura 25 Modelo Caso de Uso Administrar Ayudantes

MODELO CASO DE USO ADMINISTRAR ZONAS

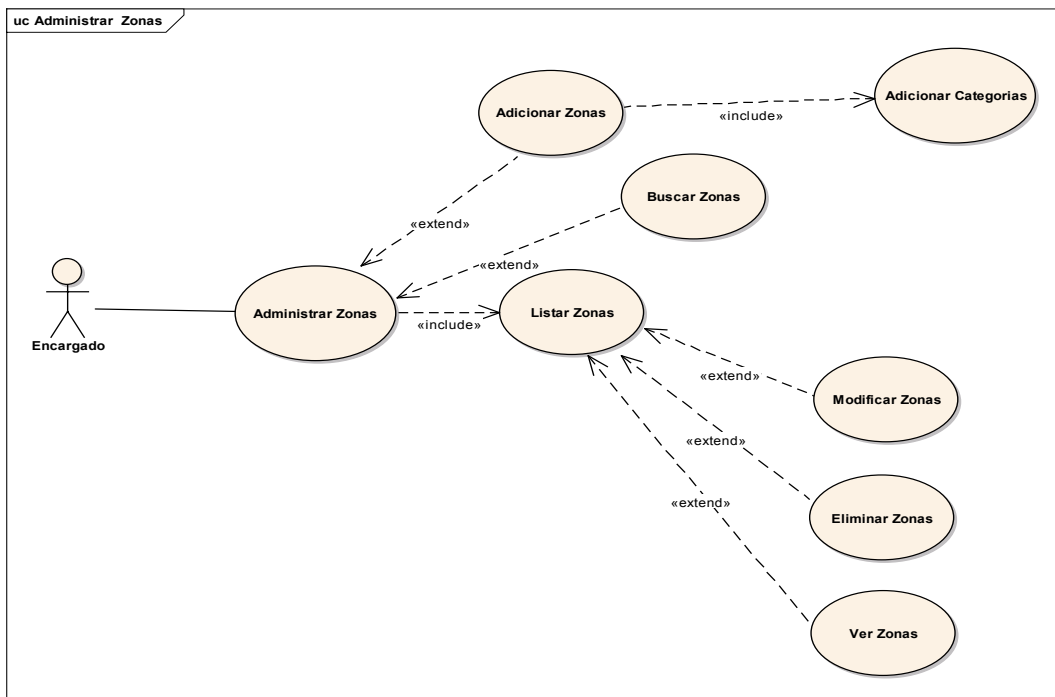


Figura 26 Modelo Caso de Uso Administrar Zonas

MODELO CASO DE USO ADMINISTRAR VEHICULOS

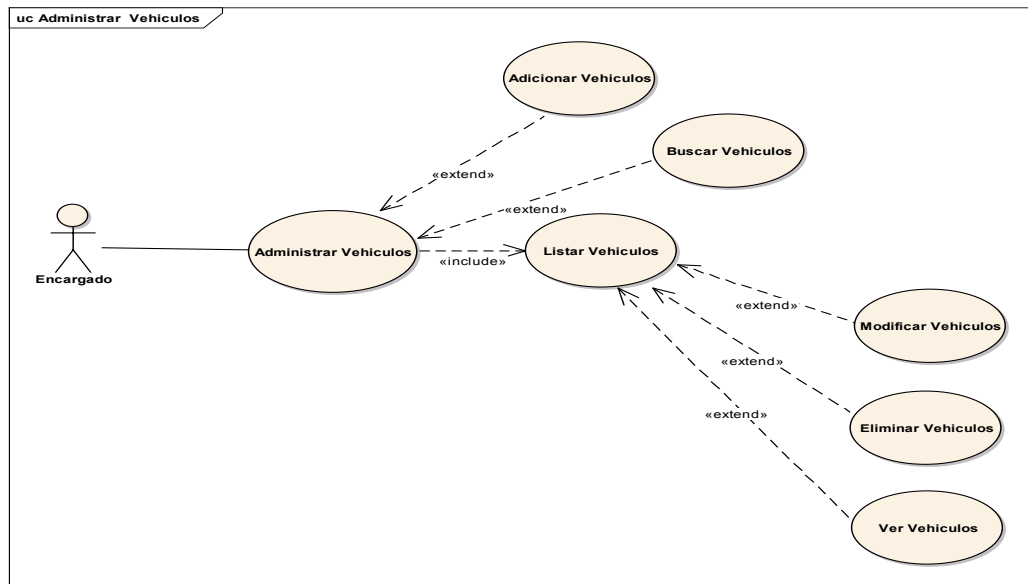


Figura 27 Modelo Caso de Uso Administrar Vehiculos

MODELO CASO DE USO ADMINISTRAR BITACORAS

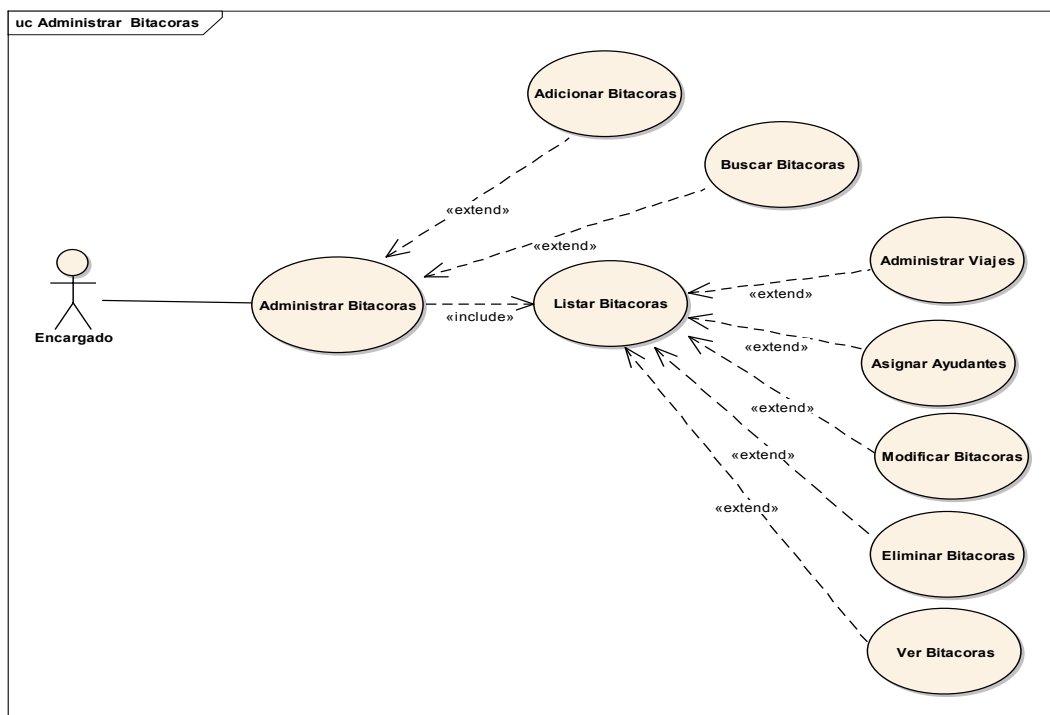


Figura 28 Modelo Caso de Uso Administrar Bitacoras

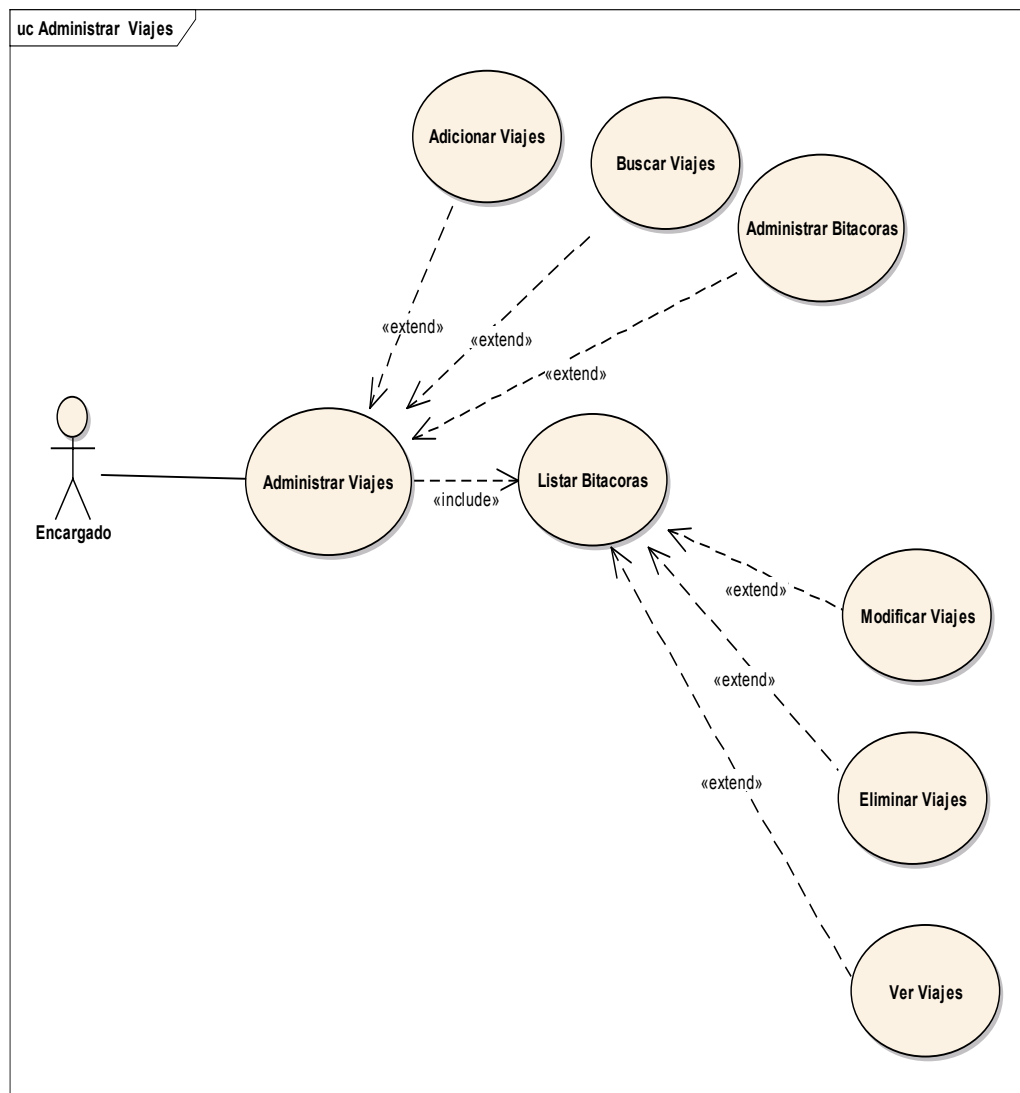
MODELO CASO DE USO ADMINISTRAR VIAJES

Figura 29 Modelo Caso de Uso Administrar Viajes

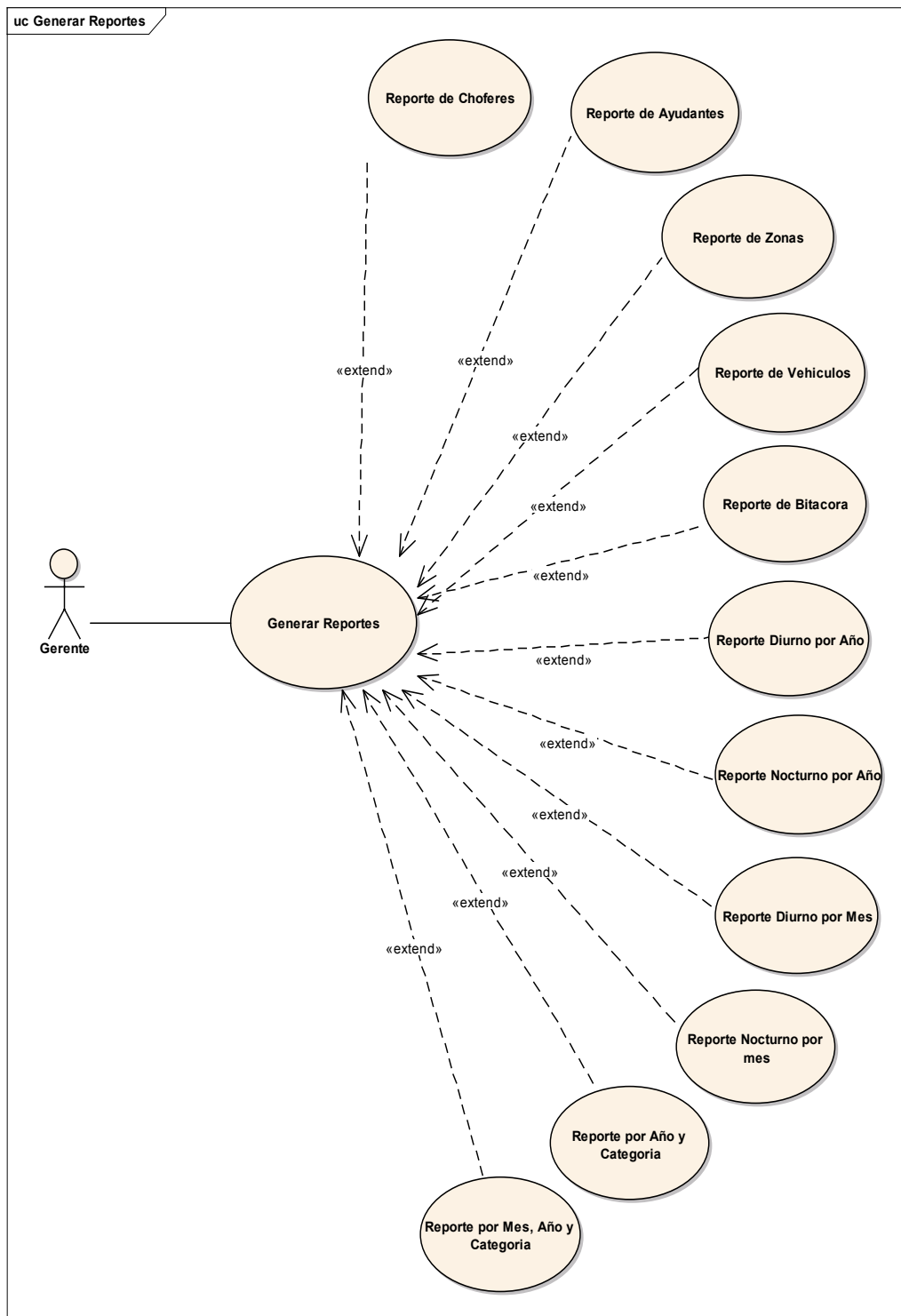
MODELO CASO DE USO GENERAR REPORTES

Figura 30 Modelo Caso de Uso Generar Reportes

2.1.6.3 ESPECIFICACIONES DE CASOS DE USO

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, 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.

ESPECIFICACIÓN DE CASO DE USO ACCESO AL SISTEMA

Caso de uso	Acceso al sistema
Actores	Administrador, Gerente, Encargado.
Tipo	Básico
Propósito	Controlar el acceso del personal autorizado para acceder al sistema.
Resumen	Este caso de uso hace referencia al control de acceso del usuario (Administrador, gerente, encargado.), autorizado al manejo del sistema, teniendo que introducir el usuario y clave de acceso, cada usuario que está autorizado para acceder al sistema consta de respectivos roles.
Precondiciones	El usuario autorizado debe introducir usuario y clave.
Flujo Principal	Se presenta al usuario la pantalla acceso al sistema 1.- El usuario introduce usuario y clave. 2.- Selecciona la opción “Acceder”. 3.- Se ejecuta el caso de uso Validar usuario y clave.

	4.- Si existe, se muestra la pantalla principal del sistema , caso contrario se lanza una excepción ;vuelve a la pantalla principal
Sub-flujos	El usuario selecciona “Salir”, se sale del sistema.
Excepciones	Ninguno

Tabla 18 Especificación Caso de Uso: Acceso al Sistema

ESPECIFICACIÓN DE CASO DE USO VALIDAR USUARIO

Caso de uso	Ingresar usuario y clave
Actores	Administrador, Gerente, Encargado.
Tipo	Inclusión
Propósito	Verificar la existencia de usuario y clave.
Resumen	Los datos introducidos por el usuario son capturados, para luego ser evaluados y poder dar así una respuesta, si el usuario y clave existen en la base de datos.
Precondiciones	El usuario debe introducir sus datos de identificación en el sistema.
Flujo Principal	1.- Buscar en la base de datos el usuario y clave.
Sub-flujos	Ninguno.
Excepciones	Ninguno.

Tabla 19 Especificación Caso de Uso: Validar usuario y clave

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR USUARIOS

Caso de uso	Administrar Usuarios
Actores	Administrador
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de Usuarios.
Resumen	Este caso de uso hace referencia a Administrar Usuarios, permitiendo visualizar una lista completa de todos los usuarios del sistema, está dado por nombre, apellido paterno y apellido materno y teléfono, dando la posibilidad de adicionar, eliminar, modificar y ver datos relacionados.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los usuarios del sistema y haber seleccionado del menú de la pantalla Principal la opción personal y luego administrar usuarios.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Usuarios con la lista de los usuarios registrados en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Usuarios”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Usuarios”.

	• Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Usuarios”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Usuarios”. • Si el usuario selecciona la opción “Ver”, se ejecuta el sub-flujo: “Ver Choferes”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Usuarios por Estado”.
Excepciones	Ninguno.

Tabla 20 Especificación Caso de Uso: Administrar Usuarios

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR USUARIOS

Caso de uso	Adicionar Usuarios
Actores	Administrador
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo usuario al sistema.

Resumen	Este caso de uso hace referencia a la creación de un nuevo usuario.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los usuarios y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Usuarios.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Usuarios. Se visualizan campos: • “Carnet de Identidad” se introduce el número de documento. • “Nombre” se introduce el nombre de la persona. • “Apellido Paterno” se introduce el apellido paterno de la persona. • “Apellido Materno” se introduce el apellido materno de la persona. • “Teléfono” se introduce el número telefónico si es que lo tiene la persona. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet”

	• “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuarios ya existe”.
--	---

Tabla 21 Especificación Caso de Uso: Adicionar Usuario

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR USUARIO

Caso de uso	Modificar Usuarios
Actores	Administrador
Tipo	Extensión
Propósito	Permite modificar los datos de un usuario del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de un usuario del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los usuarios, debe haber seleccionado de la lista un registro existente de un usuario y seleccionar la opción modificar.
Flujo	Se presenta al usuario la pantalla de Modificar Usuarios con los datos

Principal	actuales del usuario. Se visualizan campos de texto: • “Carnet de Identidad” se visualiza el número de documento a modificar • “Nombre” se visualiza el nombre de la persona a modificar. • “Apellido Paterno” se visualiza el apellido paterno del usuario a modificar. • “Apellido Materno” se visualiza el apellido materno del usuario a modificar. • “Teléfono” se visualiza el número telefónico si es que lo tiene la persona a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si la persona ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet” • “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono”

	E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuario ya existe”.
--	--

Tabla 22 Especificación Caso De Uso: Modificar Usuario

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR USUARIO

Caso de uso	Eliminar Usuario
Actores	Administrador
Tipo	Extensión
Propósito	Permite eliminar el registro de un usuario del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un usuario del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los usuarios del sistema, seleccionar de la lista un registro existente de un usuario y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar persona con el nombre del usuario a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno

Excepciones	Ninguno
--------------------	---------

Tabla 23 Especificación Caso De Uso: Eliminar Usuario

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR ROLES

Caso de uso	Administrar Roles
Actores	Administrador
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de Roles.
Resumen	Este caso de uso hace referencia a la Administrar Roles, permitiendo visualizar una lista completa de todos los roles del sistema, dando la posibilidad de adicionar, modificar, eliminar y ver datos relacionados.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los roles de la pantalla Principal y luego Administrar Roles
Flujo Principal	Se muestra al usuario la pantalla de Administrar Roles con la lista de los roles registrados en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Rol”. • Si el usuario selecciona la opción “Modificar” se ejecuta el sub-flujo:

	“Modificar Rol”. • Si el usuario selecciona la opción “Eliminar” se ejecuta el sub-flujo: “Eliminar Rol”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Rol”. • Si el usuario selecciona la opción “Ver”, se ejecuta el sub-flujo: “Ver Rol”.
Excepciones	Ninguno.

Tabla 24 Especificación Caso de Uso: Administrar Roles

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR ROL

Caso de uso	Adicionar Rol
Actores	Administrador
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo rol al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nuevo rol.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los roles y haber seleccionado la opción “Adicionar Nuevo

	Roles”: Administrar Roles.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Rol. Se visualizan campos: • “Nombre” se introduce el nombre de rol. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se muestra un mensaje: “Ingrese su nombre de rol”. E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El rol ya existe”.

Tabla 25 Especificación Caso de Uso: Adicionar Rol

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR ROL

Caso de uso	Modificar Rol
Actores	Administrador
Tipo	Extensión

Propósito	Permite modificar los datos de un Rol del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de un rol del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los roles, debe haber seleccionado de la lista un registro existente de un rol y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Rol con los datos actuales del rol. Se visualizan campos de texto: • “Nombre” se visualiza el nombre del rol a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el rol ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se muestra un mensaje: “Ingresa su nombre de rol”. E-2. Si el tipo de proceso ya existe en la base de datos se muestra un mensaje: “El rol ya existe”.

Tabla 26 Especificación Caso de Uso: Modificar Rol

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR ROL

Caso de uso	Eliminar Rol
Actores	Responsable Administrador
Tipo	Extensión
Propósito	Permite eliminar el registro de un rol del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un rol del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los roles, seleccionar de la lista un registro existente de un rol y seleccionar la opción eliminar
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar rol con el nombre del rol a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 27 Especificación Caso de Uso: Eliminar Rol

ESPECIFICACIÓN DE CASO DE USO: ASIGNAR ROLES

Caso de uso	Asignar Roles
Actores	Administrador
Tipo	Extensión
Propósito	Permite asignar el registro de uno o varios roles a un usuario del sistema.
Resumen	Este caso de uso hace referencia a la asignación de uno o varios roles a un usuario del sistema.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los roles, seleccionar de la lista un registro existente de un rol, usuario y seleccionar la opción aceptar.
Flujo Principal	Se presenta al usuario la pantalla de Asignar Roles. Con los datos actuales de todos los roles y usuarios existentes del sistema. Se visualizan los campos: • CheckBox se visualiza los nombres de los usuarios • CheckBox se visualiza los nombres de los roles que existen en el sistema • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos. • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno

Excepciones	Ninguno
--------------------	---------

Tabla 28 Especificación Caso de Uso: Asignar Roles

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR CHOFERES

Caso de uso	Administrar Choferes
Actores	Administrador / Encargado
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de Choferes.
Resumen	Este caso de uso hace referencia a Administrar Choferes, permitiendo visualizar una lista completa de todos los choferes del sistema, está dado por nombre, apellido paterno y apellido materno y teléfono, dando la posibilidad de adicionar, eliminar, modificar y ver datos relacionados.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los choferes del sistema y haber seleccionado del menú de la pantalla Principal la opción personal y luego administrar choferes.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Choferes con la lista de los choferes registrados en el sistema.

Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Choferes”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Choferes”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Choferes”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Choferes”. • Si el usuario selecciona la opción “Ver”, se ejecuta el sub-flujo: “Ver Choferes”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Choferes por Estado”.
Excepciones	Ninguno.

Tabla 29 Especificación Caso De Uso: Administrar Choferes

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR CHOFERES

Caso de uso	Adicionar choferes
--------------------	--------------------

Actores	Administrador /encargado
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo chofer al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nuevo chofer.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los choferes y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Choferes.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Choferes. Se visualizan campos: • “Carnet de Identidad” se introduce el número de documento. • “Nombre” se introduce el nombre de la persona. • “Apellido Paterno” se introduce el apellido paterno de la persona. • “Apellido Materno” se introduce el apellido materno de la persona. • “Teléfono” se introduce el número telefónico si es que lo tiene la persona. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior

Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet” • “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuarios ya existe”.

Tabla 30 Especificación Caso de Uso: Adicionar Choferes

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR CHOFERES

Caso de uso	Modificar choferes
Actores	Administrador /encargado
Tipo	Extensión
Propósito	Permite modificar los datos de un chofer del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de un chofer del sistema, para corregirlos o actualizarlos.

Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los choferes, debe haber seleccionado de la lista un registro existente de un chofer y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Choferes con los datos actuales del chofer. Se visualizan campos de texto: • “Carnet de Identidad” se visualiza el número de documento a modificar • “Nombre” se visualiza el nombre de la persona a modificar. • “Apellido Paterno” se visualiza el apellido paterno del usuario a modificar. • “Apellido Materno” se visualiza el apellido materno del usuario a modificar. • “Teléfono” se visualiza el número telefónico si es que lo tiene la persona a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si la persona ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet”

	• “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuario ya existe”.
--	--

Tabla 31 Especificación Caso de Uso: Modificar choferes

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR CHOFERES

Caso de uso	Eliminar choferes
Actores	Administrador /Encargado
Tipo	Extensión
Propósito	Permite eliminar el registro de un chofer del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un chofer del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los choferes del sistema, seleccionar de la lista un registro existente de un chofer y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar persona con el nombre del chofer a eliminar.

	El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 32 Especificación Caso De Uso: Eliminar Usuario

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR AYUDANTES

Caso de uso	Administrar ayudantes
Actores	Administrador /encargado
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de ayudantes.
Resumen	Este caso de uso hace referencia a Administrar Ayudantes, permitiendo visualizar una lista completa de todos los ayudantes del sistema, está dado por nombre, apellido paterno y apellido materno y teléfono, dando la posibilidad de adicionar, eliminar, modificar y ver datos relacionados.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los ayudantes del sistema y haber seleccionado del menú

	de la pantalla Principal la opción personal y luego Administrar Ayudantes.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Ayudantes con la lista de los ayudantes registrados en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Ayudantes”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Ayudantes”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Ayudantes”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Ayudantes”. • Si el usuario selecciona la opción “Ver”, se ejecuta el sub-flujo: “Ver Ayudantes”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Ayudantes por Estado”.
Excepciones	Ninguno.

Tabla 33 Especificación Caso De Uso: Administrar Ayudantes

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR AYUDANTES

Caso de uso	Adicionar Ayudantes
Actores	Administrador /encargado
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo Ayudante al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nuevo Ayudante.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los Ayudantes y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Ayudantes.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Ayudantes. Se visualizan campos: • “Carnet de Identidad” se introduce el número de documento. • “Nombre” se introduce el nombre de la persona. • “Apellido Paterno” se introduce el apellido paterno de la persona. • “Apellido Materno” se introduce el apellido materno de la persona. • “Teléfono” se introduce el número telefónico si es que lo tiene la persona. • “Aceptar” los datos introducidos se validan y se guardaran en la

	base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet” • “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuarios ya existe”.

Tabla 34 Especificación Caso de Uso: Adicionar Ayudantes

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR AYUDANTES

Caso de uso	Modificar Ayudantes
Actores	Administrador /encargado
Tipo	Extensión

Propósito	Permite modificar los datos de un ayudante del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de un ayudante del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar a los ayudantes, debe haber seleccionado de la lista un registro existente de un ayudante y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Ayudantes con los datos actuales del ayudante. Se visualizan campos de texto: • “Carnet de Identidad” se visualiza el número de documento a modificar • “Nombre” se visualiza el nombre de la persona a modificar. • “Apellido Paterno” se visualiza el apellido paterno del usuario a modificar. • “Apellido Materno” se visualiza el apellido materno del usuario a modificar. • “Teléfono” se visualiza el número telefónico si es que lo tiene la persona a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si la persona ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno

Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese su número de carnet” • “Ingrese su nombre” • “Ingrese su apellido paterno” • “Ingrese su apellido materno” • “Ingrese su teléfono” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “El usuario ya existe”.
--------------------	---

Tabla 35 Especificación Caso de Uso: Modificar Ayudantes

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR AYUDANTES

Caso de uso	Eliminar Ayudantes
Actores	Administrador /encargado
Tipo	Extensión
Propósito	Permite eliminar el registro de un ayudante del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un ayudante del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los ayudantes del sistema, seleccionar de la lista un registro existente de un ayudante y seleccionar la opción eliminar.

Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar persona con el nombre del usuario a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 36 Especificación Caso de Uso: Eliminar Ayudantes

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR ZONAS

Caso de uso	Administrar Zonas
Actores	Encargado
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de Zonas.
Resumen	Este caso de uso hace referencia a Administrar Zonas, permitiendo visualizar una lista completa de todas las Zonas del sistema, está dado por nombre y categoría seleccionable según a la que pertenezca, dando la posibilidad de adicionar, eliminar, modificar.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las Zonas del sistema y haber seleccionado del menú de la

	pantalla Principal la opción zonas y luego Administrar Zonas.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Zonas con la lista de las Zonas registradas en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Zonas”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Zonas”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Zonas”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Zonas”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Zonas por Estado”.
Excepciones	Ninguno.

Tabla 37 Especificación Caso de Uso: Administrar Zonas

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR ZONAS

Caso de uso	Adicionar Zonas
--------------------	-----------------

Actores	encargado
Tipo	Extensión
Propósito	Permite adicionar el registro de una nueva Zona al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nueva Zona.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las Zonas y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Zonas.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Zonas. Se visualizan campos: • “Nombre” se introduce el nombre de la persona. • CheckBox se visualiza los nombres de las categorías • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno

Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingresa el nombre” • “Seleccione la categoría” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “La Zona ya existe”.
--------------------	---

Tabla 38 Especificación Caso de Uso: Adicionar Zonas

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR ZONAS

Caso de uso	Modificar Zonas
Actores	encargado
Tipo	Extensión
Propósito	Permite modificar los datos de una Zona del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de una Zona del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las Zonas debe haber seleccionado de la lista un registro existente de una Zona y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Zonas con los datos actuales de la Zona. Se visualizan campos de texto:

	• “Nombre” se visualiza el nombre de la zona a modificar. • “Categoría” se visualiza la lista de opciones a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si la persona ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese el nombre la zona” • “Seleccione la categoría” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “la zona ya existe”.

Tabla 1 Especificación Caso De Uso: Modificar Zonas

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR ZONAS

Caso de uso	Eliminar Zonas
Actores	encargado
Tipo	Extensión

Propósito	Permite eliminar el registro de una zona del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de una zona del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar las Zonas del sistema, seleccionar de la lista un registro existente de una zona y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar zona con el nombre de la zona a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 42 Especificación Caso De Uso: Eliminar Zona

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR VEHÍCULOS

Caso de uso	Administrar Vehículos
Actores	Encargado
Tipo	Básico

Propósito	Permite visualizar y realizar diferentes acciones en una lista de los Vehículos.
Resumen	Este caso de uso hace referencia a Administrar Vehículos, permitiendo visualizar una lista completa de todos los Vehículos del sistema, está dado por placa modelo y tara, dando la posibilidad de adicionar, eliminar, modificar.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los Vehículos del sistema y haber seleccionado del menú de la pantalla Principal la opción Vehículos y luego Administrar Vehículos.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Vehículos con la lista de los Vehículos registradas en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Vehículos”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Vehículos”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Vehículos”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Vehículos”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Vehículos por Estado”.

Excepciones	Ninguno.

Tabla 41 Especificación Caso de Uso: Administrar Vehículos

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR VEHÍCULOS

Caso de uso	Adicionar Vehículos
Actores	encargado
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo Vehículos al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nuevo Vehículo
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los Vehículos y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Vehículos.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Vehículos. Se visualizan campos: • “placa” se introduce la placa del vehículo. • “modelo” se introduce el modelo del vehículo. • “tara” se introduce el peso del vehículo. • “Aceptar” los datos introducidos se validan y se guardaran en la

	base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese la placa” • “Ingrese el modelo” • “Ingrese la tara” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “el vehículo ya existe”.

Tabla 42 Especificación Caso de Uso: Adicionar Vehículos

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR VEHÍCULOS

Caso de uso	Modificar Vehículos
Actores	encargado
Tipo	Extensión
Propósito	Permite modificar los datos de un vehículo del sistema.

Resumen	Este caso de uso hace referencia a la modificación de datos de un vehículo del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los Vehículos debe haber seleccionado de la lista un registro existente de un vehículo y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Vehículos con los datos actuales del Vehículo. Se visualizan campos de texto: • “placa” se visualiza la placa del vehículo a modificar. • “modelo” se visualiza el modelo del vehículo a modificar. • “tara” se visualiza la tara del vehículo a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el vehículo ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Ingrese la placa del vehículo” • “Ingrese el modelo del vehículo” • “Ingrese la tara del vehículo” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje:

	“el vehículo ya existe”.
--	--------------------------

Tabla 43 Especificación Caso De Uso: Modificar Vehículo

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR VEHÍCULO

Caso de uso	Eliminar vehículo
Actores	encargado
Tipo	Extensión
Propósito	Permite eliminar el registro de un vehículo del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un vehículo del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los vehículos del sistema, seleccionar de la lista un registro existente de un vehículo y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar vehículo con el dato del vehículo a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.

Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 44 Especificación Caso De Uso: Eliminar Vehículo

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR BITÁCORAS

Caso de uso	Administrar Bitácoras
Actores	Encargado
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de las Bitácoras
Resumen	Este caso de uso hace referencia a Administrar Bitácoras, permitiendo visualizar una lista completa de todas las Bitácoras del sistema, está dado por chofer, vehículo, hora salida, hora llegada, combustible, odómetro salida, odómetro llegada, horometro salida, horometro llegada; dando la posibilidad de adicionar, eliminar, modificar y ver datos del mismo.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las Bitácoras del sistema y haber seleccionado del menú de la pantalla Principal la opción Registro y luego Administrar Bitácoras.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Bitácoras con la lista de las Bitácoras registradas en el sistema.

Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Bitácoras”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Bitácoras”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Bitácoras”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Bitácoras”. • Si el usuario selecciona la opción “Ver” se ejecuta el sub-flujo: “Ver Bitácoras”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Bitácoras por Estado”.
Excepciones	Ninguno.

Tabla 45 Especificación Caso de Uso: Administrar Bitácoras

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR BITÁCORAS

Caso de uso	Adicionar Bitácoras
Actores	encargado

Tipo	Extensión
Propósito	Permite adicionar el registro de una nueva Bitácora al sistema.
Resumen	Este caso de uso hace referencia a la creación de una nueva Bitácora
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las Bitácoras y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Bitácoras.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Bitácoras. Se visualizan campos: • CheckBox se selecciona un Chofer que haya hecho el recorrido • CheckBox se selecciona un Vehículo que haya hecho el recorrido • “hora salida” se introduce la hora de salida de la empresa. • “hora llegada” se introduce la hora de llegada a la empresa. • “combustible” se introduce la cantidad de combustible. • “odómetro salida” se introduce el valor del odómetro que marca al salir. • “odómetro llegada” se introduce el valor del odómetro que marca al llegar. • “horometro salida” se introduce el horometro que marca al salir. • “horometro llegada” se introduce el horometro que marca al llegar. • “Aceptar” los datos introducidos se validan y se guardaran en la

	base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Seleccione un chofer” • “Seleccione un vehículo ” • “Ingrese la hora de salida” • “Ingrese la hora de llegada” • “Ingrese el combustible” • “Ingrese el odómetro de salida” • “Ingrese el odómetro de llegada” • “Ingrese el horómetro de salida” • “Ingrese el horometro de llegada” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “la bitácora ya existe”.

Tabla 46 Especificación Caso De Uso: Adicionar Bitácora

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR BITÁCORA

Caso de uso	Modificar bitácora
--------------------	--------------------

Actores	encargado
Tipo	Extensión
Propósito	Permite modificar los datos de una bitácora del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de una bitácora del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las bitácoras debe haber seleccionado de la lista un registro existente de una bitácora y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Vehículos con los datos actuales de la bitácora. Se visualizan campos de texto: • “Seleccione un chofer” se visualiza el chofer a modificar. • “Seleccione un vehículo” se visualiza el vehículo a modificar. • “Ingrese la hora de salida” se visualiza la hora de salida a modificar. • “Ingrese la hora de llegada” se visualiza la hora de llegada a modificar • “Ingrese el combustible” se visualiza el combustible a modificar • “Ingrese el odómetro de salida” se visualiza el odómetro de salida a modificar • “Ingrese el odómetro de llegada” se visualiza el odómetro de llegada a modificar • “Ingrese el horómetro de salida” se visualiza el horómetro de

	salida a modificar • “Ingrese el horometro de llegada” se visualiza el horómetro de llegada a modificar • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si la bitácora ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Seleccione un chofer” • “Seleccione un vehículo ” • “Ingrese la hora de salida” • “Ingrese la hora de llegada” • “Ingrese el combustible” • “Ingrese el odómetro de salida” • “Ingrese el odómetro de llegada” • “Ingrese el horómetro de salida” • “Ingrese el horometro de llegada” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “la bitácora ya existe”.

Tabla 47 Especificación Caso De Uso: Modificar Bitácoras

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR BITÁCORAS

Caso de uso	Eliminar bitácoras
Actores	encargado
Tipo	Extensión
Propósito	Permite eliminar el registro de una bitácora del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de una bitácora del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar las bitácoras del sistema, seleccionar de la lista un registro existente de una bitácora y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar bitácora con el dato de la bitácora a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 48 Especificación Caso De Uso: Eliminar Bitácoras

ESPECIFICACIÓN DE CASO DE USO: VER BITÁCORAS

Caso de uso	Ver bitácoras
Actores	Encargado
Tipo	Extensión
Propósito	Permite visualizar los datos guardados de la bitácora.
Resumen	Este caso de uso hace referencia a ver de datos de la bitácora del sistema.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar las bitácoras, debe haber seleccionado de la lista un registro existente de una bitácora y seleccionar la opción ver bitácora
Flujo Principal	Se presenta al usuario la pantalla de Ver bitácoras. Se visualizan campos de texto: • “chofer” se visualiza el chofer. • “vehículo” se visualiza el vehículo. • “la hora de salida” se visualiza la hora de salida. • “hora de llegada” se visualiza la hora de llegada • “combustible” se visualiza el combustible • “odómetro de salida” se visualiza el odómetro de salida • “odómetro de llegada” se visualiza el odómetro de llegada • “horómetro de salida” se visualiza el horómetro de salida

	• “horometro de llegada” se visualiza el horómetro de llegada • “Imprimir” Manda a la opción imprimir. • “Cancelar” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 49 Especificación Caso De Uso: Ver Bitácoras

ESPECIFICACIÓN DE CASO DE USO: ADMINISTRAR VIAJES

Caso de uso	Administrar Viajes
Actores	Encargado
Tipo	Básico
Propósito	Permite visualizar y realizar diferentes acciones en una lista de Viajes
Resumen	Este caso de uso hace referencia a Administrar Viajes, permitiendo visualizar una lista completa de todas los Viajes del sistema, está dado por la zona , hora ingreso, hora salida, ingreso al relleno, salida del relleno, odómetro inicio, odómetro salida, peso neto; dando la posibilidad de adicionar, eliminar, modificar y ver datos del mismo.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los viajes del sistema y haber seleccionado del menú de la

	pantalla Principal la opción Registro-Administrar Bitácoras y luego Administrar Viajes.
Flujo Principal	Se muestra al usuario la pantalla de Administrar Viajes con la lista de los Viajes registradas en el sistema.
Sub-flujos	El usuario puede seleccionar entre las siguientes opciones: • Si el usuario selecciona la opción: “Adicionar”, se ejecuta el sub-flujo: “Adicionar Viajes”. • Si el usuario selecciona la opción “Modificar”, se ejecuta el sub-flujo: “Modificar Viajes”. • Si el usuario selecciona la opción “Eliminar”, se ejecuta el sub-flujo: “Eliminar Viajes”. • Si el usuario selecciona la opción “Buscar” se ejecuta el sub-flujo: “Buscar Viajes”. • Si el usuario selecciona la opción “Ver” se ejecuta el sub-flujo: “Ver Viajes”. • Si el usuario selecciona la opción de estado (activo o inactivo), se ejecuta el sub-flujo: “Listar Viajes por Estado”.
Excepciones	Ninguno.

Tabla 50 Especificación Caso de Uso: Administrar Viajes

ESPECIFICACIÓN DE CASO DE USO: ADICIONAR VIAJES

Caso de uso	Adicionar Viajes
Actores	encargado
Tipo	Extensión
Propósito	Permite adicionar el registro de un nuevo Viaje al sistema.
Resumen	Este caso de uso hace referencia a la creación de un nuevo Viaje
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los Viajes y haber seleccionado la opción “Adicionar” en la pantalla: Administrar Viajes.
Flujo Principal	Se presenta al usuario la pantalla de Adicionar Viajes. Se visualizan campos: • CheckBox se selecciona una zona a la que se fue hacer el recorrido • “hora ingreso” se introduce la hora de ingreso a la zona. • “hora salida” se introduce la hora de salida de la zona. • “ingreso al relleno” se introduce la hora de ingreso al relleno sanitario. • “salida del relleno” se introduce la hora de salida al relleno sanitario. • “odómetro inicio” se introduce el valor del odómetro que marca al inicio de hacer el viaje.

	• “odómetro salida” se introduce el valor del odómetro que marca al finalizar el viaje. • “peso neto” se introduce el valor total del vehículo. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el proceso ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
uw	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Seleccione una zona” • “Ingrese la hora de ingreso” • “Ingrese la hora de salida” • “ingrese la hora de ingreso al relleno” • “ingrese la hora de salida al relleno” • “Ingrese el odómetro de inicio” • “Ingrese el odómetro de salida” • “Ingrese el peso neto” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “el viaje ya existe”.

Tabla 51 Especificación Caso de Uso: Adicionar viaje

ESPECIFICACIÓN DE CASO DE USO: MODIFICAR VIAJE

Caso de uso	Modificar viaje
Actores	encargado
Tipo	Extensión
Propósito	Permite modificar los datos de un viaje del sistema.
Resumen	Este caso de uso hace referencia a la modificación de datos de un viaje del sistema, para corregirlos o actualizarlos.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los viajes debe haber seleccionado de la lista un registro existente de un viaje y seleccionar la opción modificar.
Flujo Principal	Se presenta al usuario la pantalla de Modificar Viajes con los datos actuales del viaje. Se visualizan campos de texto: • “Seleccione una zona” se visualiza la zona a modificar. • “hora ingreso” se introduce la hora de ingreso a la zona a modificar. • “hora salida” se introduce la hora de salida de la zona a modificar. • “ingreso al relleno” se introduce la hora de ingreso al relleno

	sanitario a modificar. • “salida del relleno” se introduce la hora de salida al relleno sanitario a modificar. • “odómetro inicio” se introduce el valor del odómetro a modificar. • “odómetro salida” se introduce el valor del odómetro a modificar. • “peso neto” se introduce el valor total del vehículo a modificar. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos, si los campos no están completos o son incorrectos se genera una excepción (E-1), si el viaje ya existe se genera la excepción (E-2). • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	E-1. Si los datos no están completos o son incorrectos se mostrará algunos de estos mensajes: • “Seleccione una zona” • “Ingrese la hora de ingreso” • “Ingrese la hora de salida” • “ingrese la hora de ingreso al relleno” • “ingrese la hora de salida al relleno” • “Ingrese el odómetro de inicio” • “Ingrese el odómetro de salida”

	• “Ingrese el peso neto” E-2. Si el proceso ya existe en la base de datos se muestra un mensaje: “el viaje ya existe”.
--	--

Tabla 52 Especificación Caso de Uso: Modificar Viajes

ESPECIFICACIÓN DE CASO DE USO: ELIMINAR VIAJES

Caso de uso	Eliminar Viajes
Actores	encargado
Tipo	Extensión
Propósito	Permite eliminar el registro de un Viaje del sistema.
Resumen	Este caso de uso hace referencia a la eliminación de un Viaje del sistema, eliminándolo lógicamente.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los Viajes del sistema, seleccionar de la lista un registro existente de un Viaje y seleccionar la opción eliminar.
Flujo Principal	Se muestra al usuario la pantalla de dialogo de confirmación eliminar viaje con el dato del Viaje a eliminar. El usuario puede seleccionar las siguientes opciones: • “Aceptar” El registro se elimina lógicamente de la base de datos. Retornando a la pantalla anterior. • “Cancelar” no se realiza ningún cambio, Retornando a la pantalla

	anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 53 Especificación Caso de Uso: Eliminar Viajes

ESPECIFICACIÓN DE CASO DE USO: VER VIAJES

Caso de uso	Ver Viajes
Actores	Encargado
Tipo	Extensión
Propósito	Permite visualizar los datos guardados del Viaje
Resumen	Este caso de uso hace referencia a ver de datos del Viaje del sistema.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para gestionar los Viajes, debe haber seleccionado de la lista un registro existente de un Viaje y seleccionar la opción ver Viaje
Flujo Principal	Se presenta al usuario la pantalla de Ver Viajes. Se visualizan campos de texto: • “zona” se visualiza la zona • “hora de ingreso” se visualiza la hora de ingreso del viaje

	• “hora de salida” se visualiza la hora de salida del viaje • “hora de ingreso al relleno” se visualiza la hora de ingreso al relleno • “hora de salida al relleno” se visualiza la hora de salida del relleno • “odómetro de inicio” se visualiza el odómetro de inicio del viaje • “odómetro de salida” se visualiza el odómetro de salida del viaje • “peso neto” se visualiza el peso neto del vehículo • “Imprimir” Manda a la opción imprimir. • “Cancelar” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 54 Especificación Caso de Uso: ver Viajes

ESPECIFICACIÓN DE CASO DE USO: ASIGNAR AYUDANTES

Caso de uso	Asignar Ayudantes
Actores	Encargado

Tipo	Extensión
Propósito	Permite asignar el registro de uno o varios Ayudantes a una bitácora del sistema.
Resumen	Este caso de uso hace referencia a la asignación de uno o varios Ayudantes a una bitácora del sistema.
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar las bitácoras, seleccionar de la lista un registro existente de una bitácora, y seleccionar la opción asignar ayudantes.
Flujo Principal	Se presenta al usuario la pantalla de Asignar Ayudantes. Con los datos actuales de todos los Ayudantes existentes del sistema. Se visualizan los campos: • se visualizan los nombres de los ayudantes y se marcan los que trabajaron en esa bitácora. • “Aceptar” los datos introducidos se validan y se guardaran en la base de datos. • “Cancelar” no se realiza ningún cambio. Retornando a la pantalla anterior
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 55 Especificación Caso de Uso: Asignar ayudantes

ESPECIFICACIÓN DE CASO DE USO: REPORTE CHOFERES

Caso de uso	Reporte choferes
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de los choferes existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de los choferes existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos los choferes existentes del sistema. Se visualiza • se visualiza el reporte de los choferes • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 56 Especificación Caso De Uso: Reporte Choferes

ESPECIFICACIÓN DE CASO DE USO: REPORTE AYUDANTES

Caso de uso	Reporte ayudantes
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de los ayudantes existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de los ayudantes existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos los ayudantes existentes del sistema. Se visualiza • se visualiza el reporte de los ayudantes • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 57 Especificación Caso De Uso: Reporte Ayudantes

ESPECIFICACIÓN DE CASO DE USO: REPORTE ZONAS

Caso de uso	Reporte zonas
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las zonas existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las zonas existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todas las zonas existentes del sistema. Se visualiza • se visualiza el reporte de los zonas • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 58 Especificación Caso De Uso: Reporte Zonas

ESPECIFICACIÓN DE CASO DE USO: REPORTE VEHÍCULOS

Caso de uso	Reporte vehículos
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de los vehículos existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de los vehículos existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todas los vehículos existentes del sistema. Se visualiza • se visualiza el reporte de los vehículos • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno

Excepciones	Ninguno
--------------------	---------

Tabla 59 Especificación Caso De Uso: Reporte Vehículos

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS

Caso de uso	Reporte bitácoras
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las distintas bitácoras existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las distintas bitácoras existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos las bitácoras existentes del sistema. Se visualiza • se visualiza el reporte de la bitácora • “Imprimir” Manda a la opción imprimir.

	• “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 60 Especificación Caso De Uso: Reporte Bitácora

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /DIURNO POR AÑO

Caso de uso	Reporte bitácoras /diurno por año
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las bitácoras /diurno por año existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las bitácoras /diurno por año existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos las bitácoras /diurno por año existentes del sistema.

	Se visualiza • se visualiza la opción reporte de las bitácoras /diurno por año. • se selecciona la gestión de la cual se quiere reporte. • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 61 Especificación Caso De Uso: Reporte Bitácoras /Diurno Por Año

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /NOCTURNO POR AÑO

Caso de uso	Reporte bitácoras /nocturno por año
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las bitácoras /nocturno por año existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las bitácoras /nocturno por año existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del

	sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos las bitácoras /nocturno por año existentes del sistema. Se visualiza • se visualiza la opción reporte de las bitácoras /nocturno por año. • se selecciona la gestión de la cual se quiere reporte. • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 62 Especificación Caso De Uso: Reporte Bitácoras /Nocturno Por Año

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /DIURNO POR MES Y AÑO

Caso de uso	Reporte bitácoras /diurno por mes y año
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las bitácoras /diurno por mes y año existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las

	bitácoras /diurno por mes y año existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos las bitácoras /diurno por mes y año existentes del sistema. Se visualiza • se visualiza la opción reporte de las bitácoras /diurno por mes y año. • se selecciona el mes del cual se quiere reporte. • se selecciona la gestión de la cual se quiere reporte. • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 63 Especificación Caso De Uso: Reporte Bitácoras /Diurno Por Mes Y Año

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /NOCTURNO POR MES Y AÑO

Caso de uso	Reporte bitácoras /nocturno por mes y año
--------------------	---

Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las bitácoras /nocturno por mes y año existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las bitácoras /nocturno por mes y año existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todas las bitácoras /nocturno por mes y año existentes del sistema. Se visualiza • se visualiza la opción reporte de las bitácoras /nocturno por mes y año. • se selecciona el mes del cual se quiere reporte. • se selecciona la gestión de la cual se quiere reporte. • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 64 Especificación Caso De Uso: Reporte Bitácoras /Nocturno Por Mes Y Año

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /TOTAL POR AÑO Y CATEGORÍA

Caso de uso	Reporte Bitácoras /total por año y categoría
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las Bitácoras /total por año y categoría existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las Bitácoras /total por año y categoría existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo Principal	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todos las Bitácoras /total por año y categoría existentes del sistema. Se visualiza • se visualiza la opción reporte de las Bitácoras /total por año y categoría. • se selecciona el año del cual se quiere reporte. • se selecciona la categoría de la cual se quiere reporte.

	• “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 65 Especificación Caso De Uso: Reporte Bitácoras /Total Por Año Y Categoría

ESPECIFICACIÓN DE CASO DE USO: REPORTE BITÁCORAS /TOTAL POR MES, AÑO Y CATEGORÍA

Caso de uso	Reporte Bitácoras /total por mes, año y categoría
Actores	Gerente
Tipo	Extensión
Propósito	Permite visualizar el reporte de las Bitácoras /total por mes, año y categoría existentes del sistema.
Resumen	Este caso de uso hace referencia a la visualización del reporte de las Bitácoras /total por mes, año y categoría existentes del sistema
Precondiciones	El usuario debe haber ingresado al sistema, estar autorizado para administrar los reportes , seleccionar del menú un reporte requerido del sistema
Flujo	Se presenta al usuario la pantalla de reportes de registros. Con los datos actuales de todas las Bitácoras /total por mes, año y categoría existentes

Principal	del sistema. Se visualiza • se visualiza la opción reporte de las Bitácoras /total por mes, año y categoría. • se selecciona el mes del cual se quiere reporte. • se selecciona el año del cual se quiere reporte. • se selecciona la categoría de la cual se quiere reporte. • “Imprimir” Manda a la opción imprimir. • “atrás” Manda a la pantalla anterior.
Sub-flujos	Ninguno
Excepciones	Ninguno

Tabla 66 Especificación Caso De Uso: Reporte Bitácoras /Total Por Mes, Año Y Categoría

2.1.6.4 MODELADO DE DIAGRAMA DE ACTIVIDADES

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.

DIAGRAMA DE ACTIVIDAD: ACCESO AL SISTEMA

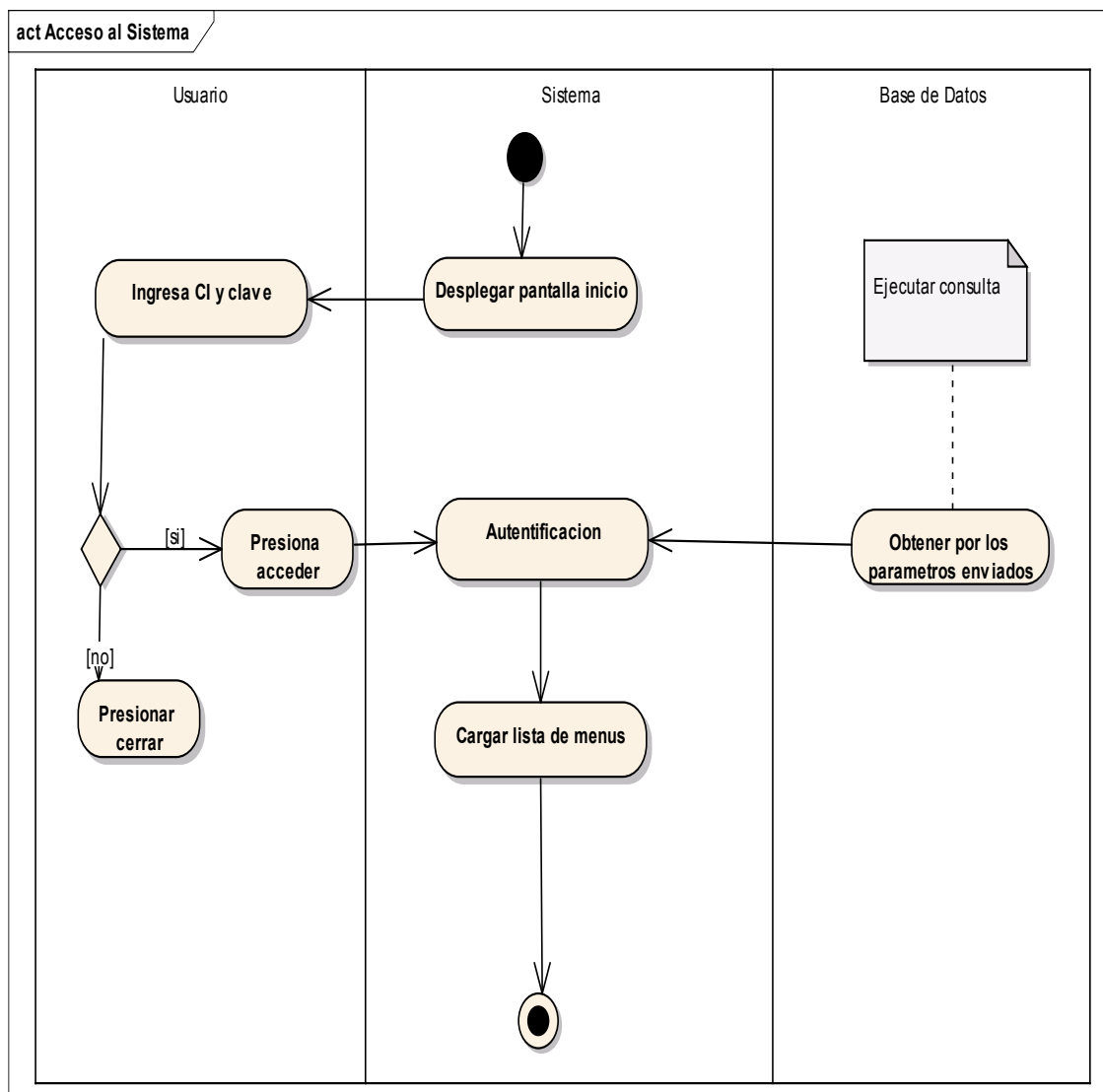


Figura 97 Diagrama de Actividad: Acceso al Sistema

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR USUARIOS

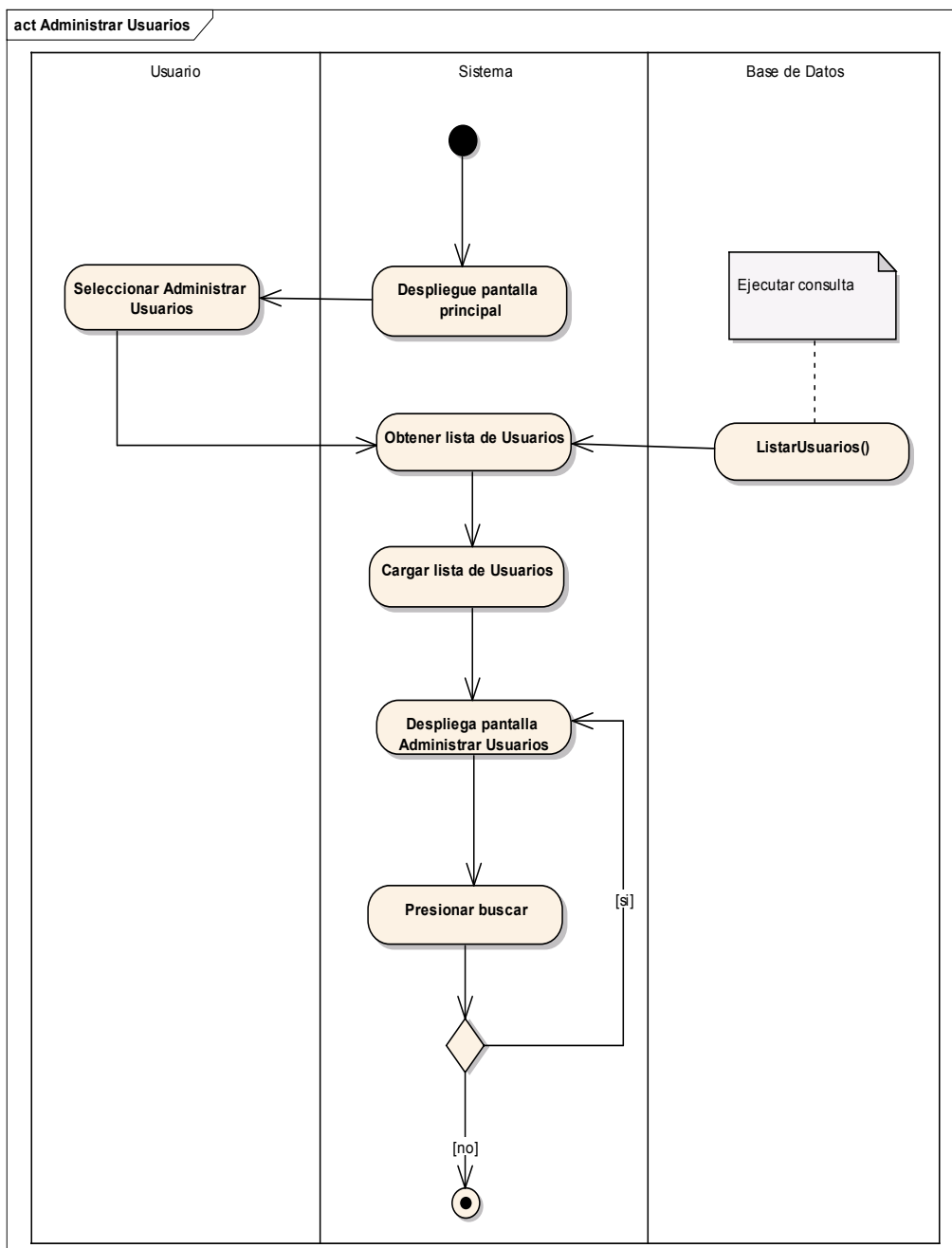


Figura 98 Diagrama de Actividad: Administrar Usuarios

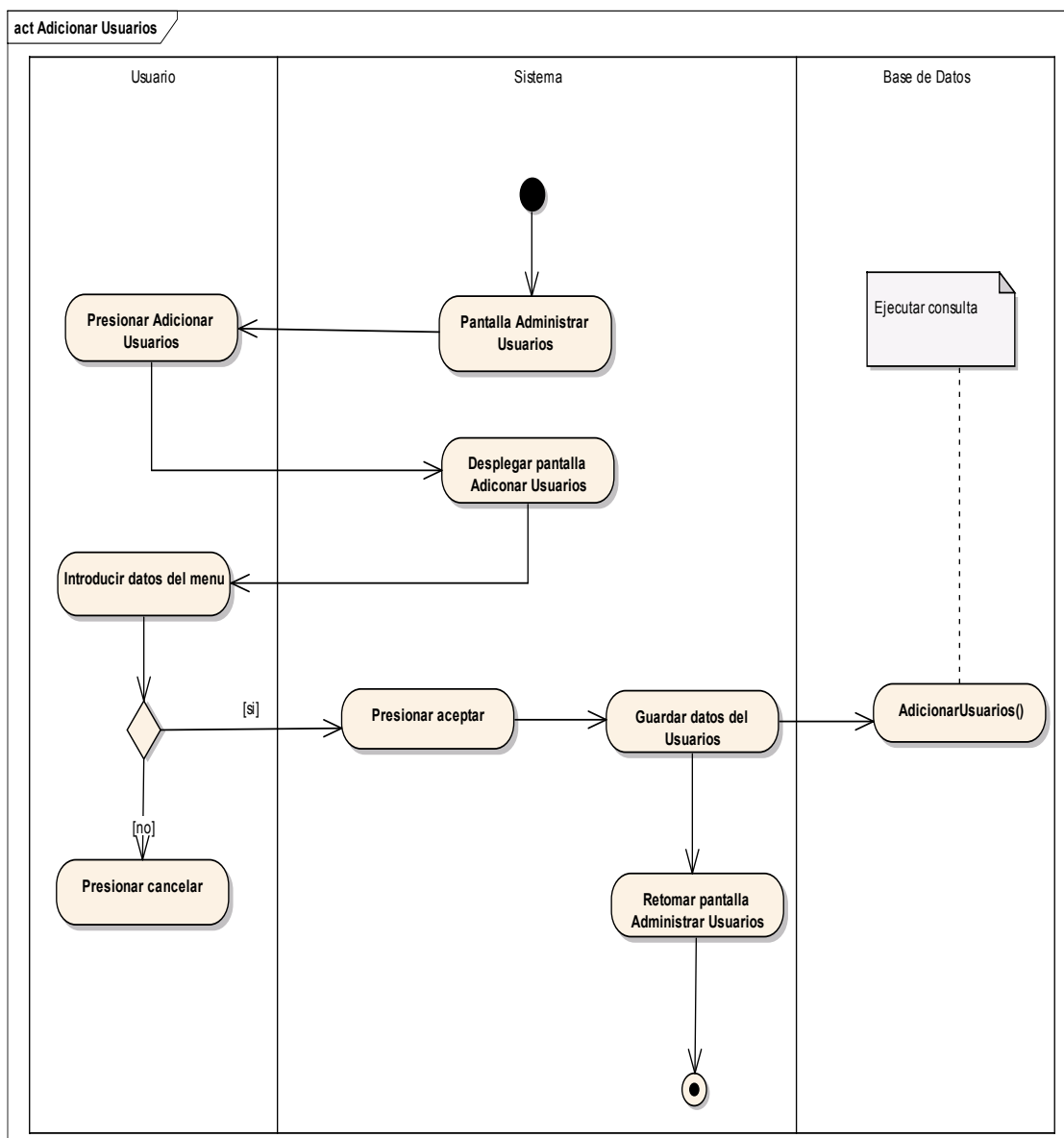
DIAGRAMA DE ACTIVIDAD: ADICIONAR USUARIOS

Figura 99 Diagrama de Actividad: Adicionar Usuarios

DIAGRAMA DE ACTIVIDAD: MODIFICAR USUARIOS

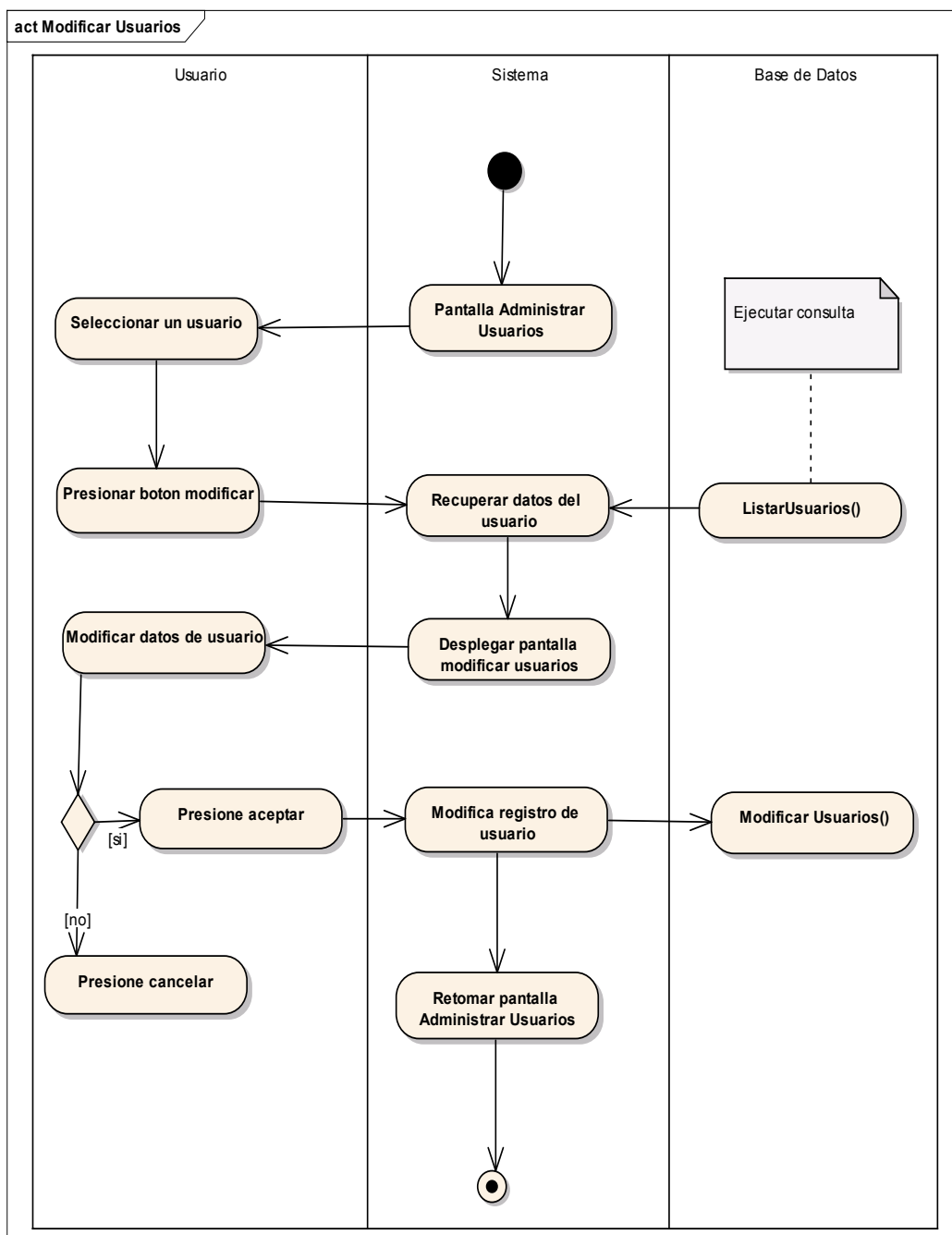


Figura 100 Diagrama de Actividad: Modificar Usuarios

DIAGRAMA DE ACTIVIDAD: ELIMINAR USUARIOS

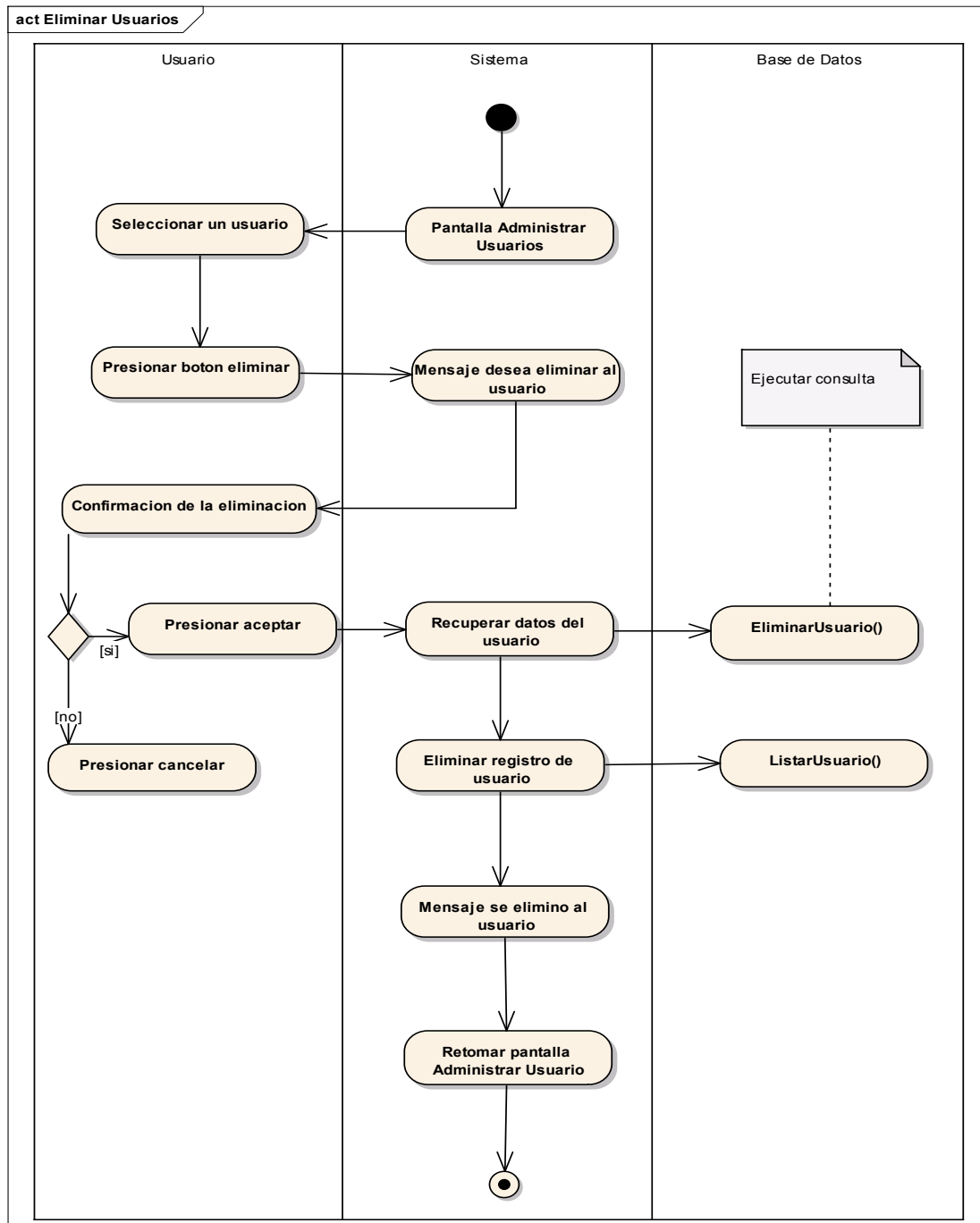


Figura 101 Diagrama de Actividad: Eliminar Usuarios

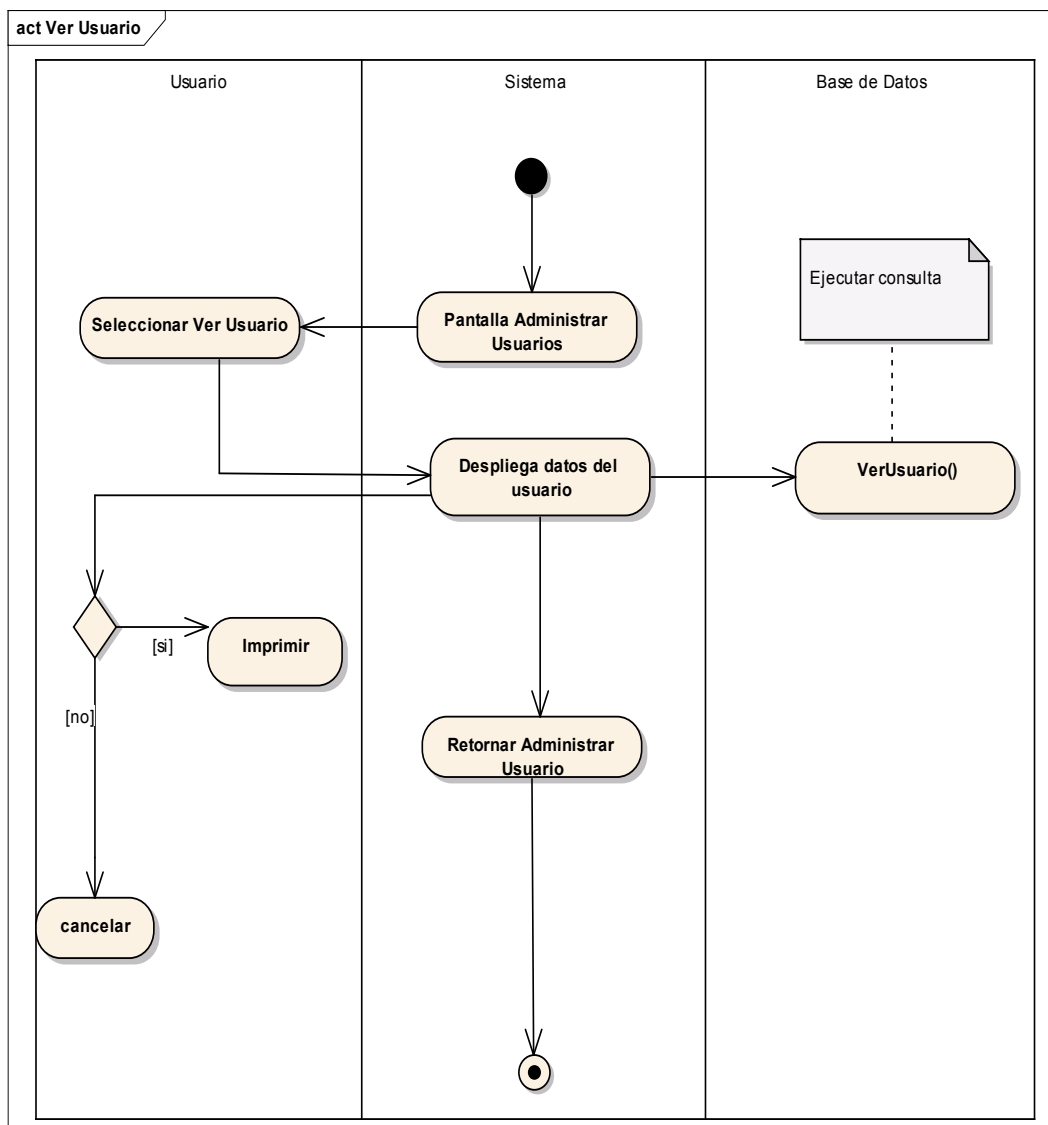
DIAGRAMA DE ACTIVIDAD: VER USUARIOS

Figura 102 Diagrama de Actividad: Ver Usuarios

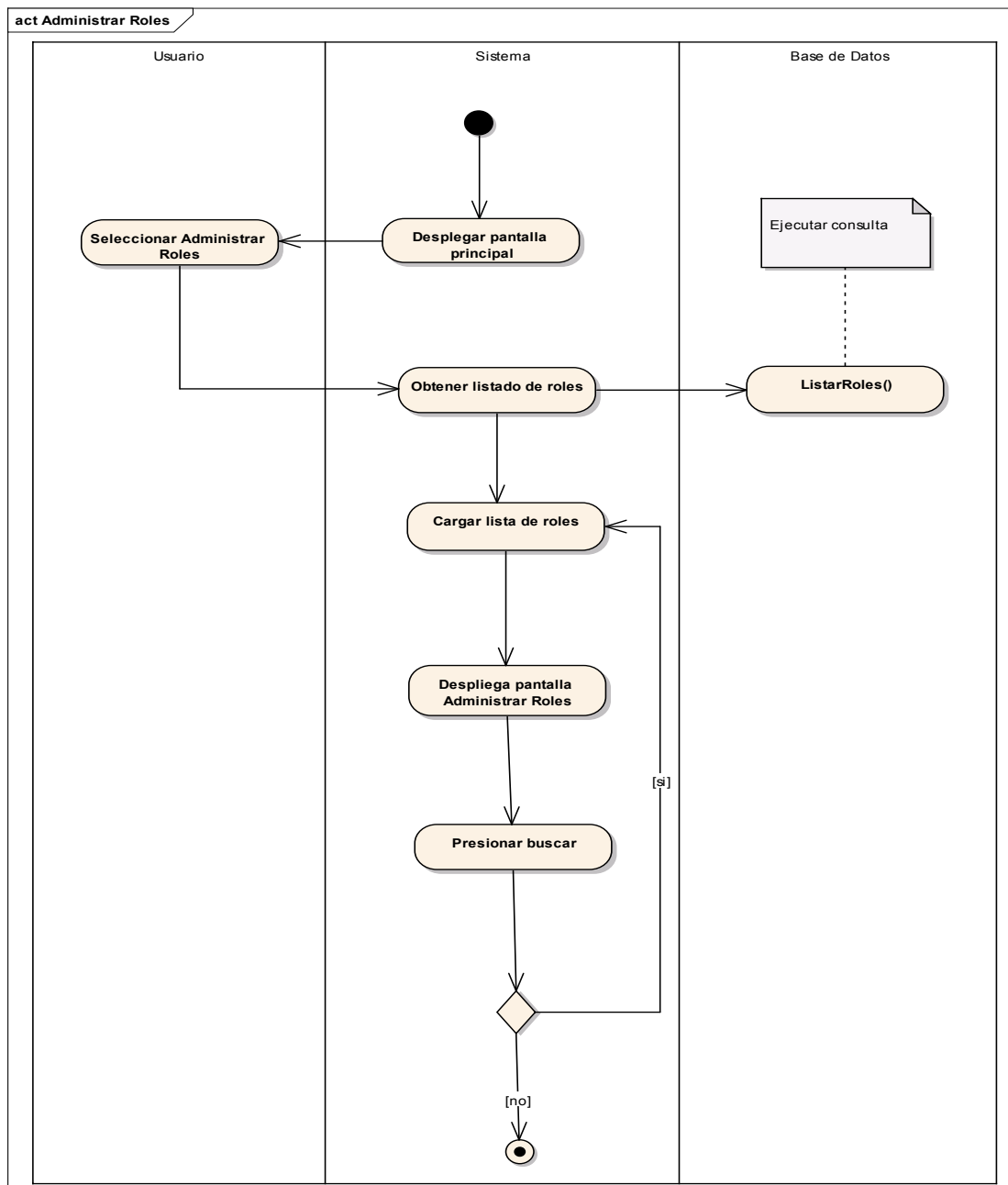
DIAGRAMA DE ACTIVIDAD: ADMINISTRAR ROLES

Figura 103 Diagrama de Actividad: Administrar Roles

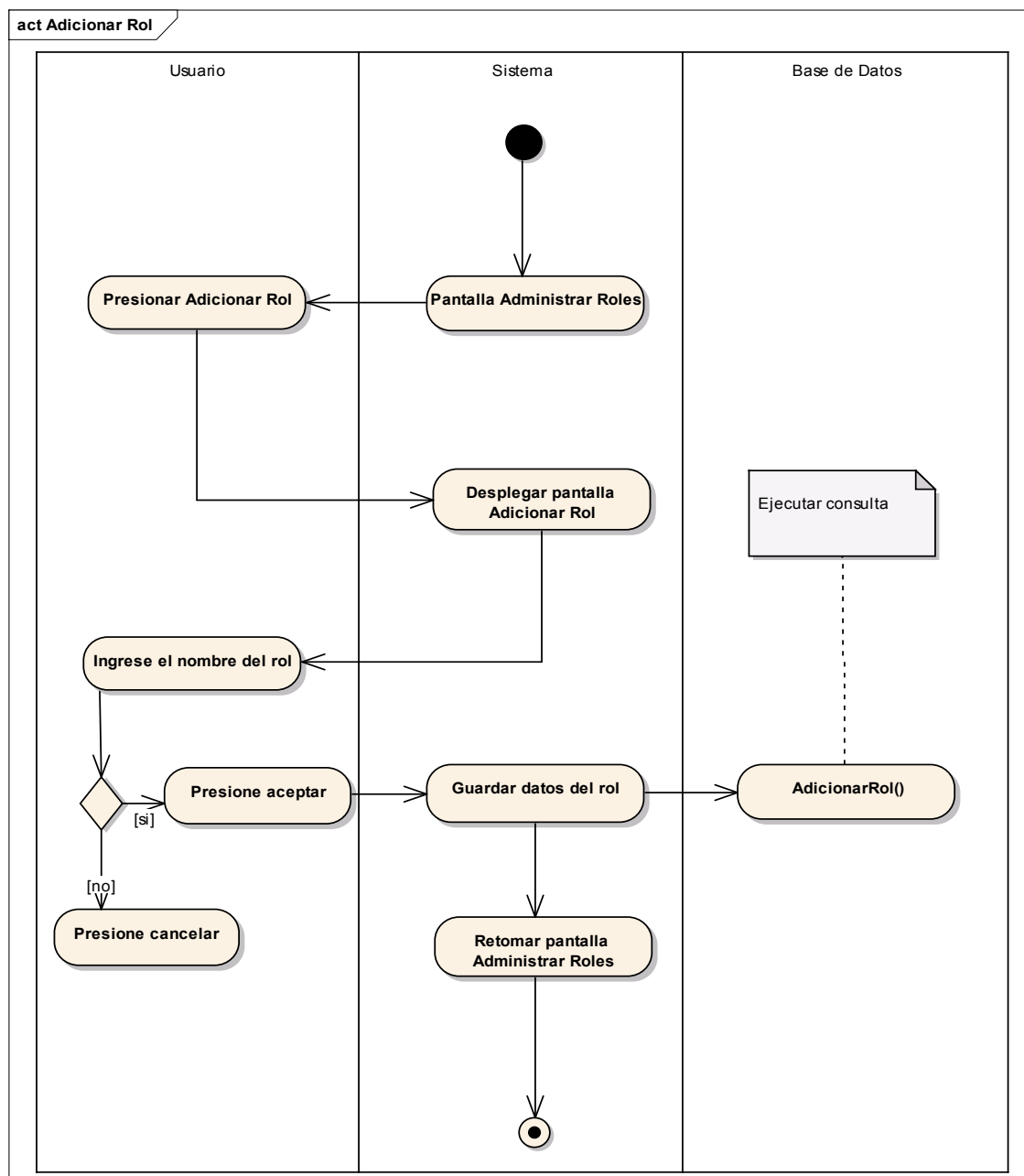
DIAGRAMA DE ACTIVIDAD: ADICIONAR ROLES

Figura 104 Diagrama de Actividad: Adicionar Roles

DIAGRAMA DE ACTIVIDAD: MODIFICAR ROLES

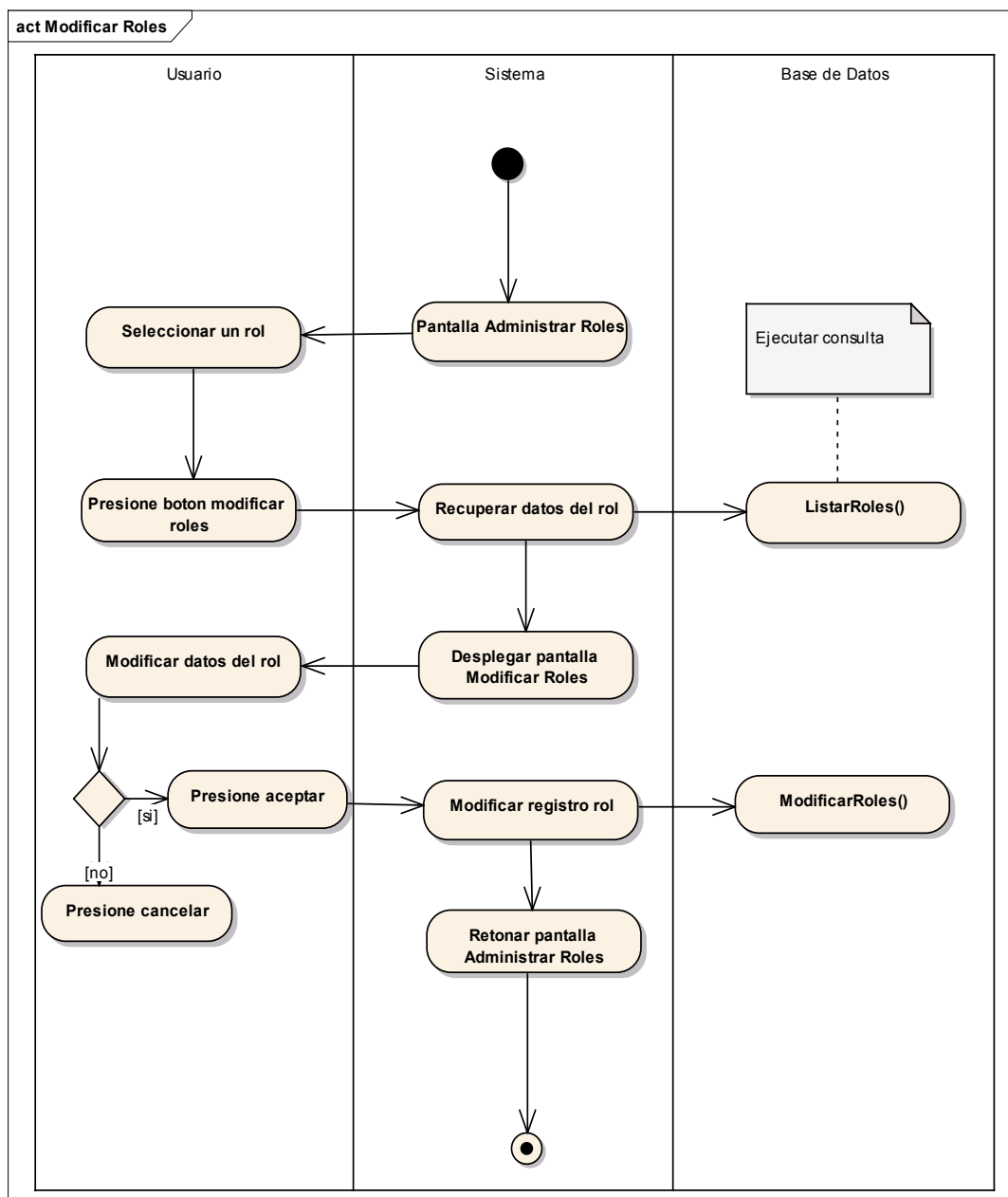


Figura 105 Diagrama de Actividad: Modificar Roles

DIAGRAMA DE ACTIVIDAD: ELIMINAR ROLES

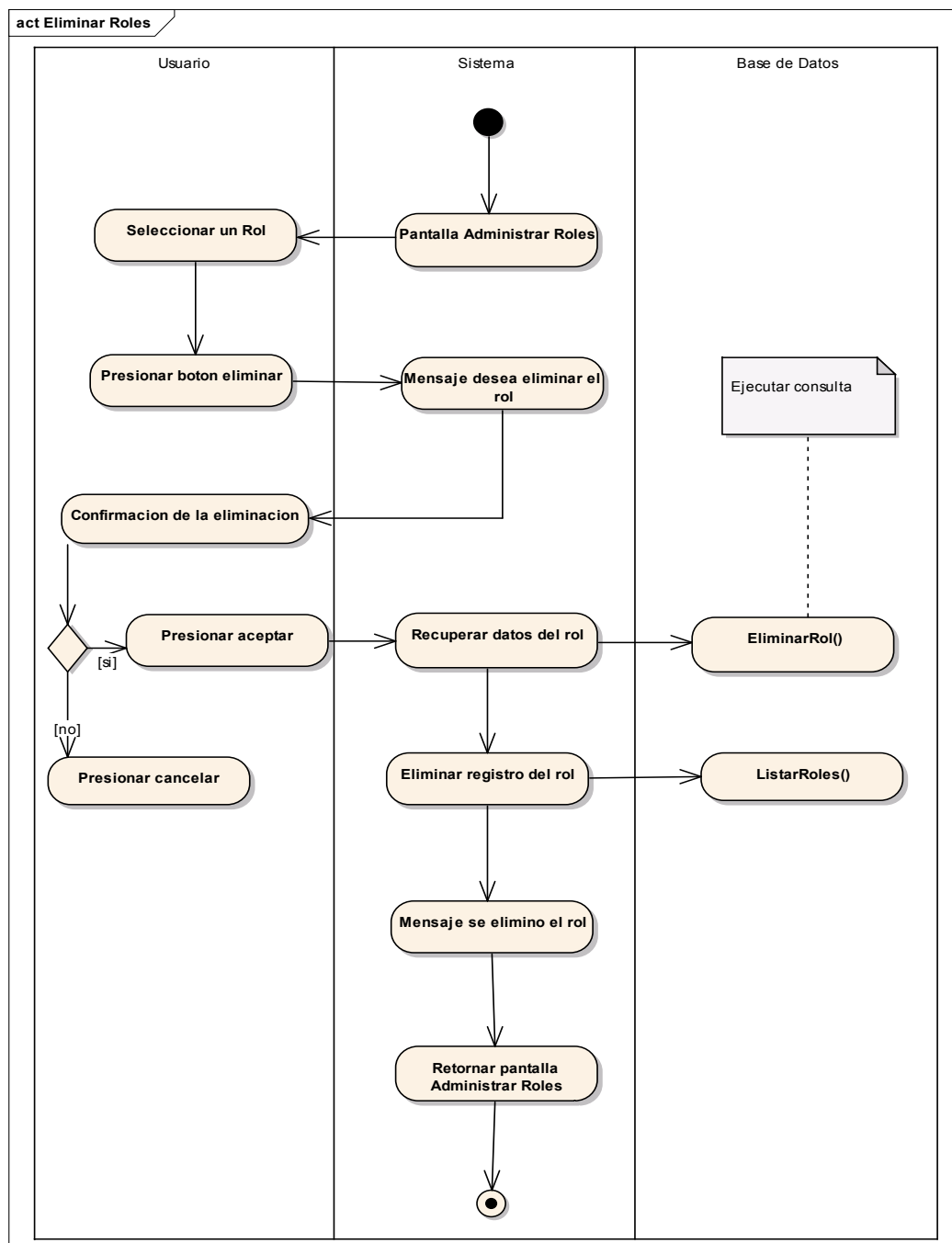


Figura 106 Diagrama de Actividad: Eliminar Roles

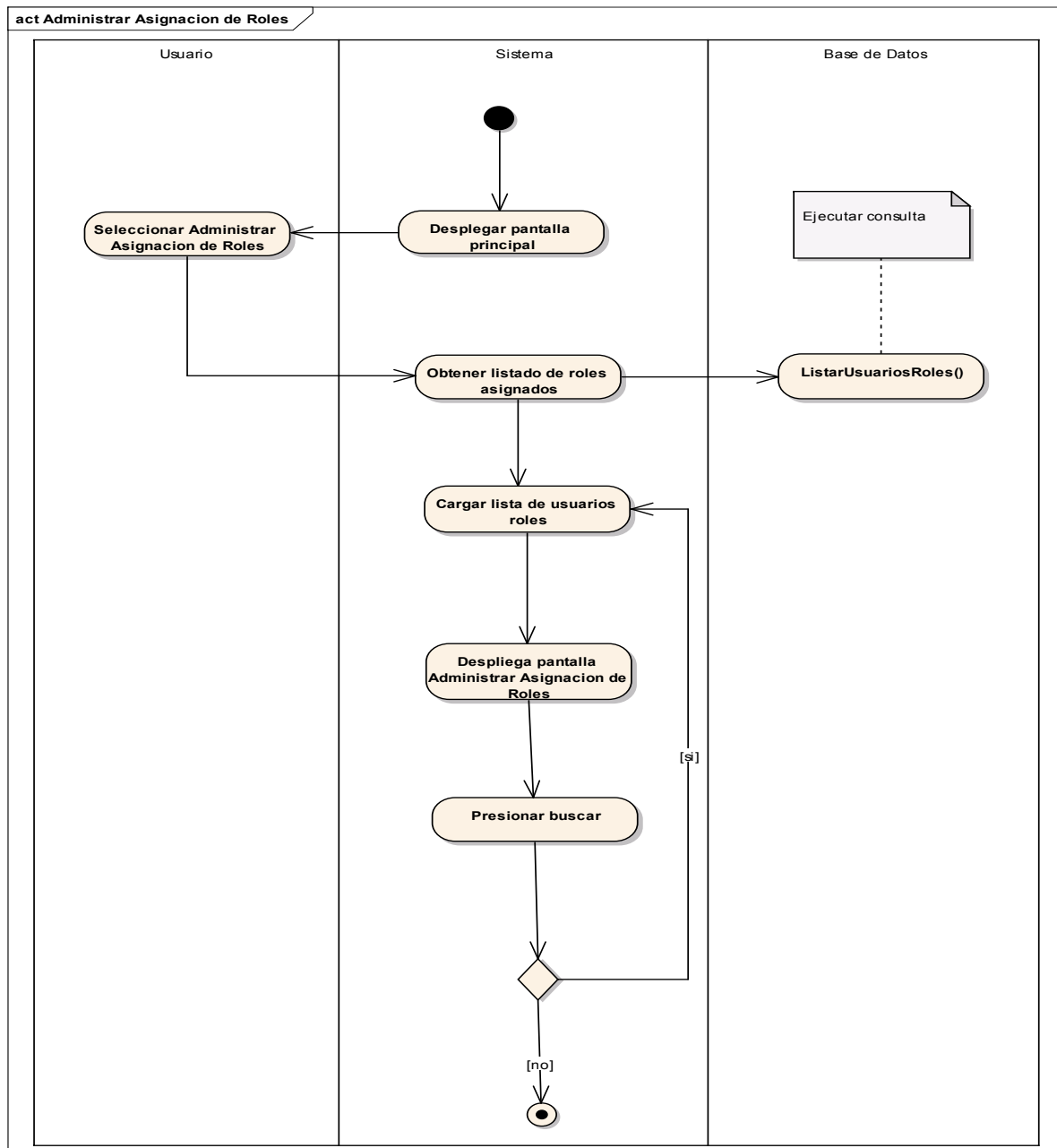
DIAGRAMA DE ACTIVIDAD: ADMINISTRAR ASIGNACION DE ROLES

Figura 107 Diagrama de Actividad: Administrar Asignacion de Roles

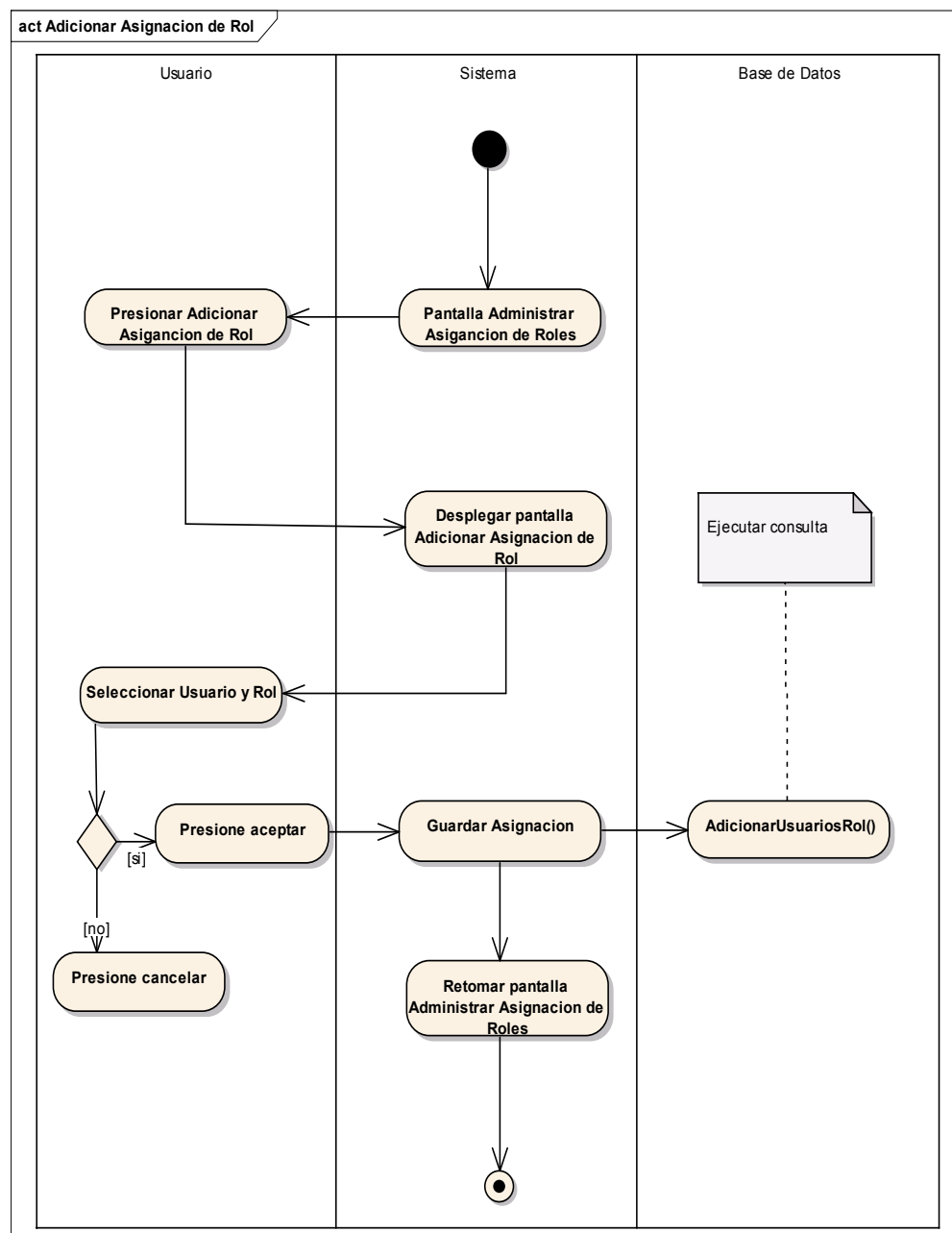
DIAGRAMA DE ACTIVIDAD: ADICIONAR ASIGNACION DE ROLES

Figura 108 Diagrama de Actividad: Adicionar Asignacion de Roles

DIAGRAMA DE ACTIVIDAD: ELIMINAR ASIGNACION DE ROLES

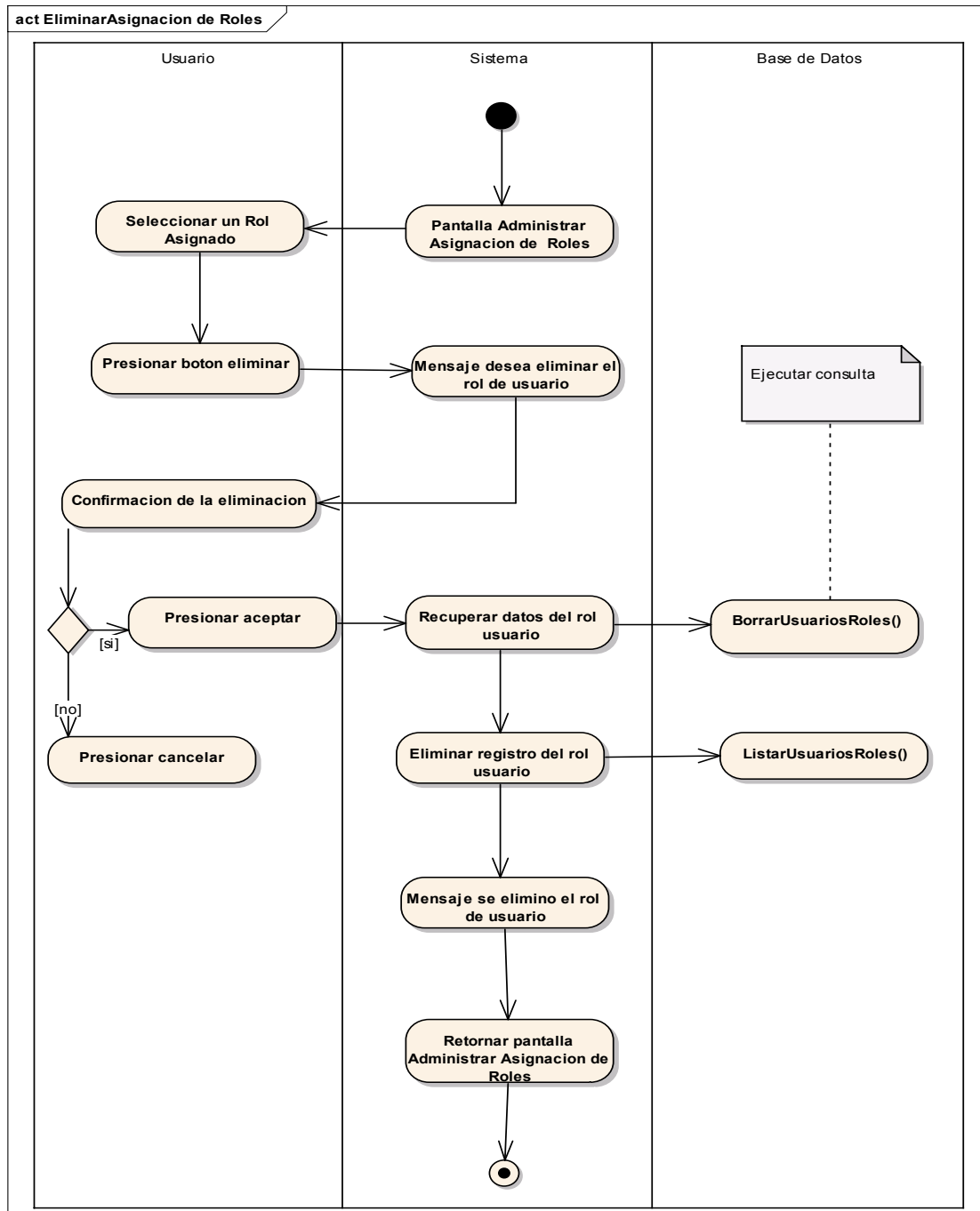


Figura 109 Diagrama de Actividad: Eliminar Asignación de Roles

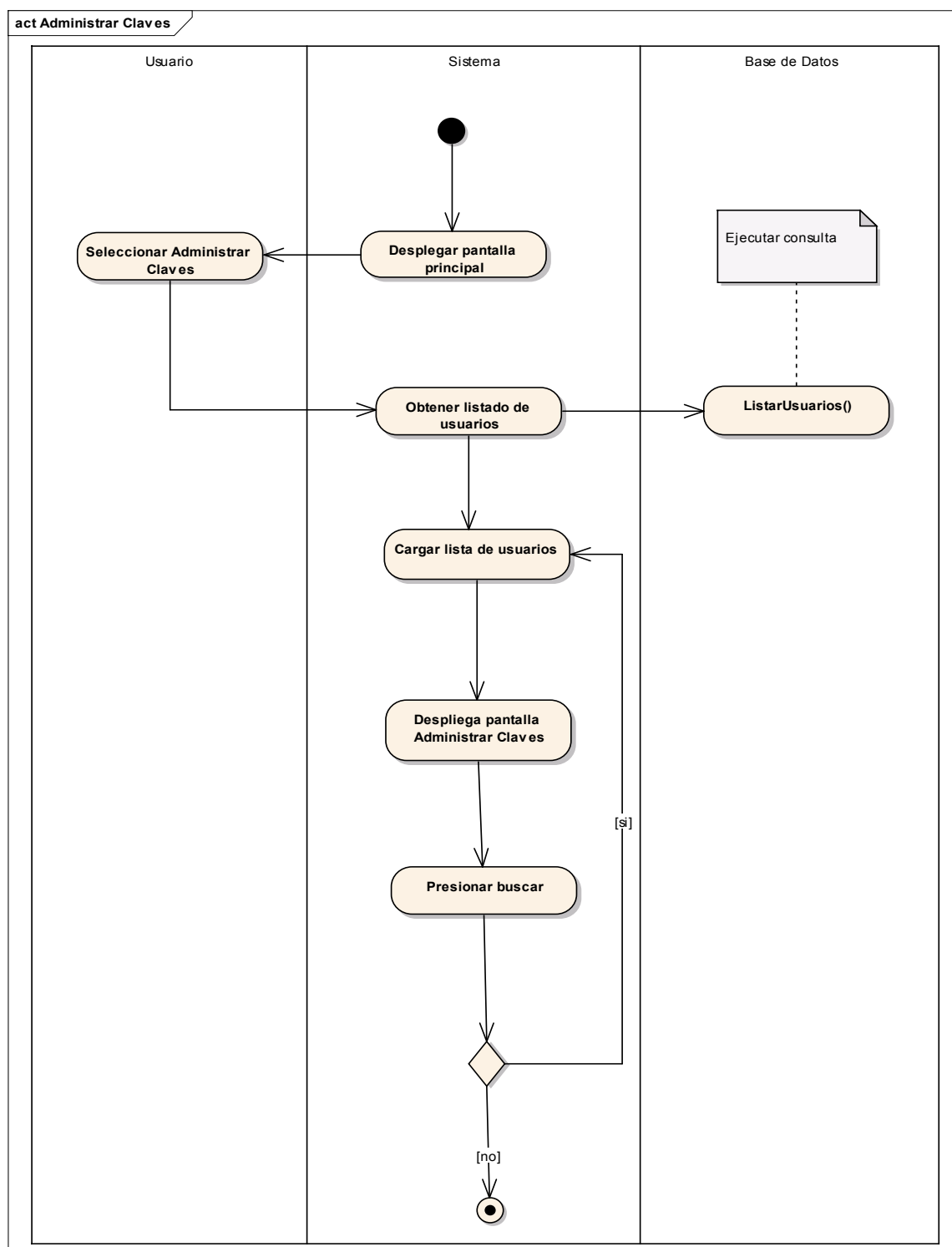
DIAGRAMA DE ACTIVIDAD: ADMINISTRAR CLAVES

Figura 110 Diagrama de Actividad: Administrar Claves

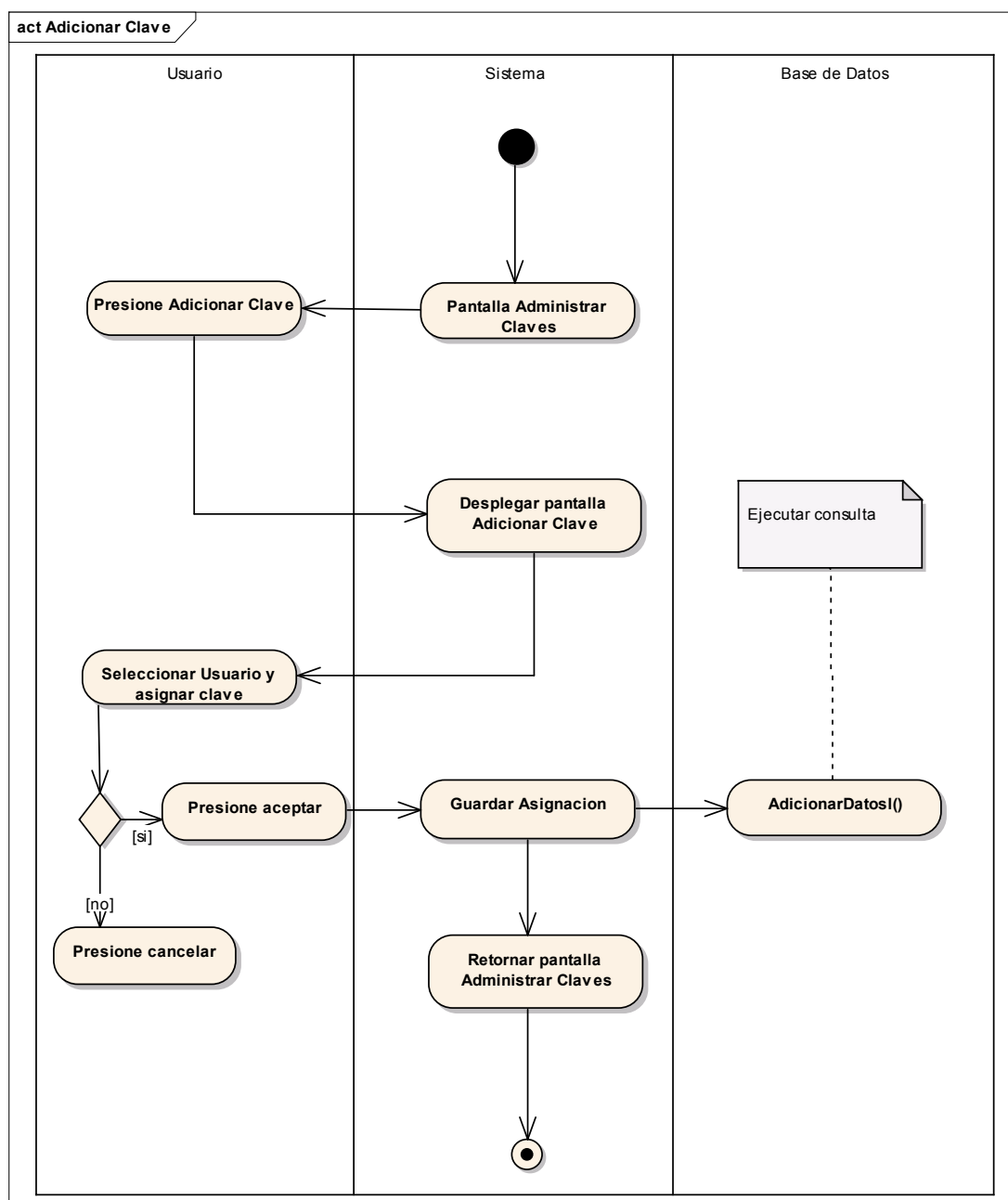
DIAGRAMA DE ACTIVIDAD: ADICIONAR CLAVES

Figura 111 Diagrama de Actividad: Adicionar Claves

DIAGRAMA DE ACTIVIDAD: ELIMINAR CLAVES

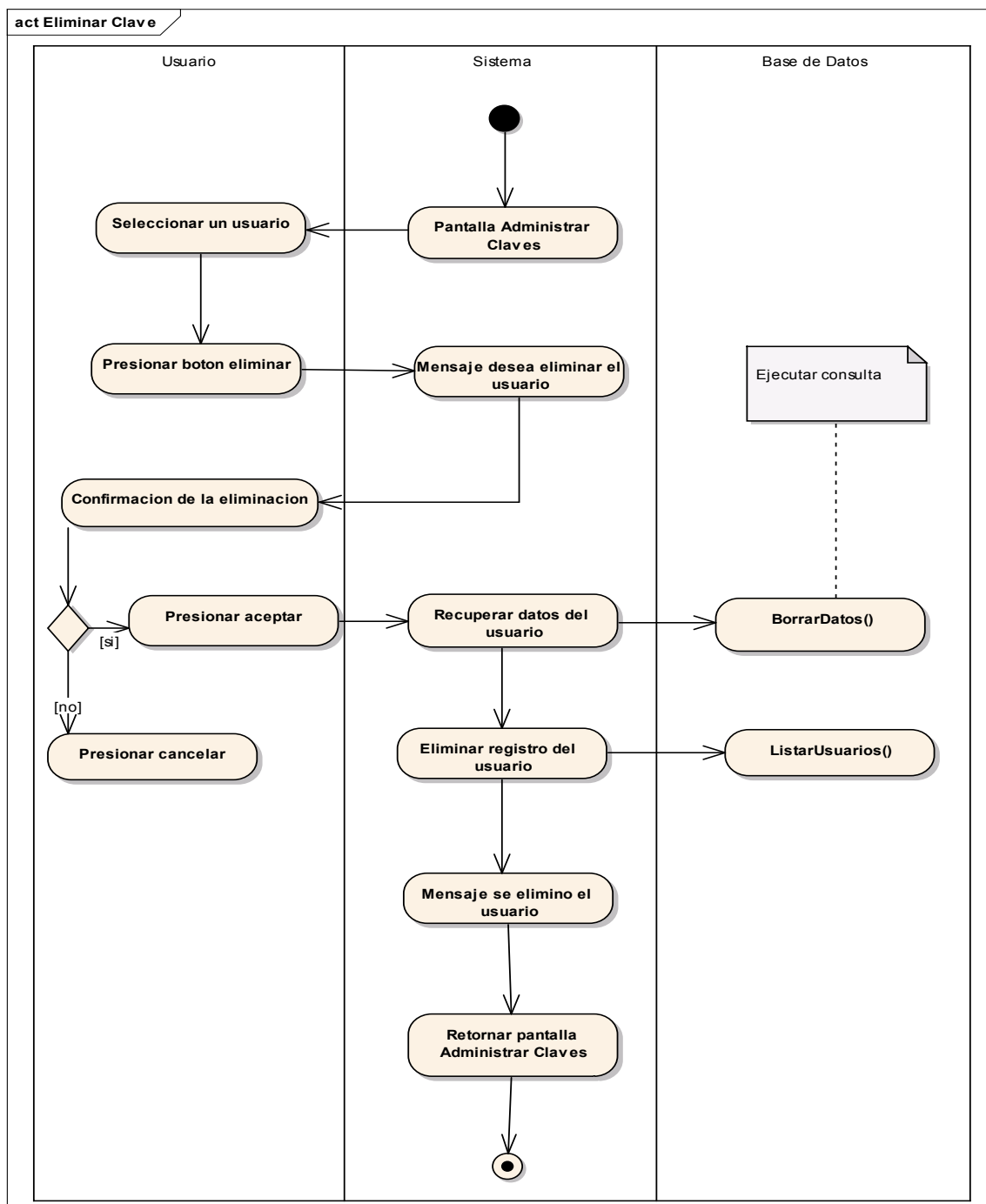


Figura 112 Diagrama de Actividad: Eliminar Claves

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR CHOFERES

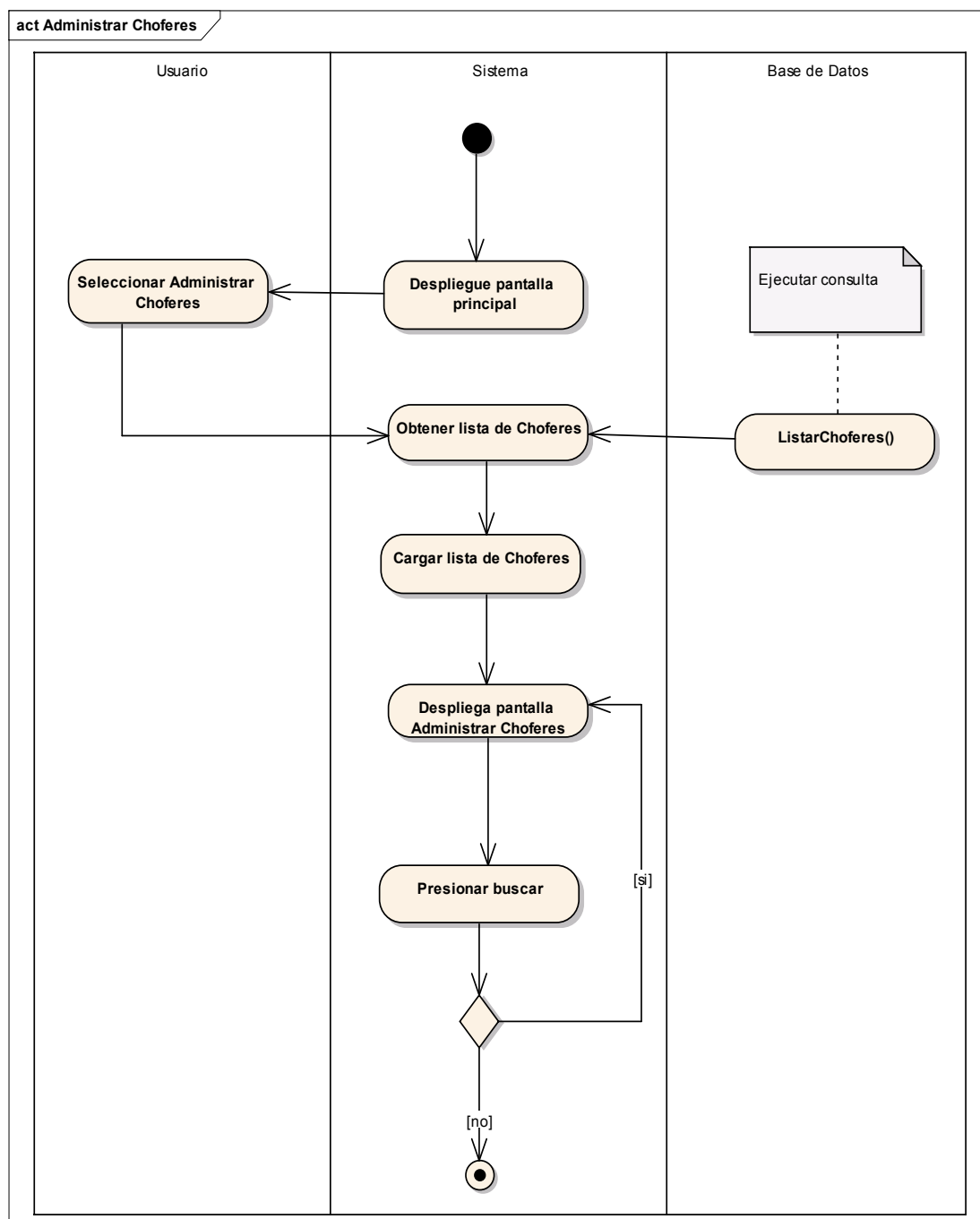


Figura 113 Diagrama de Actividad: Administrar Choferes

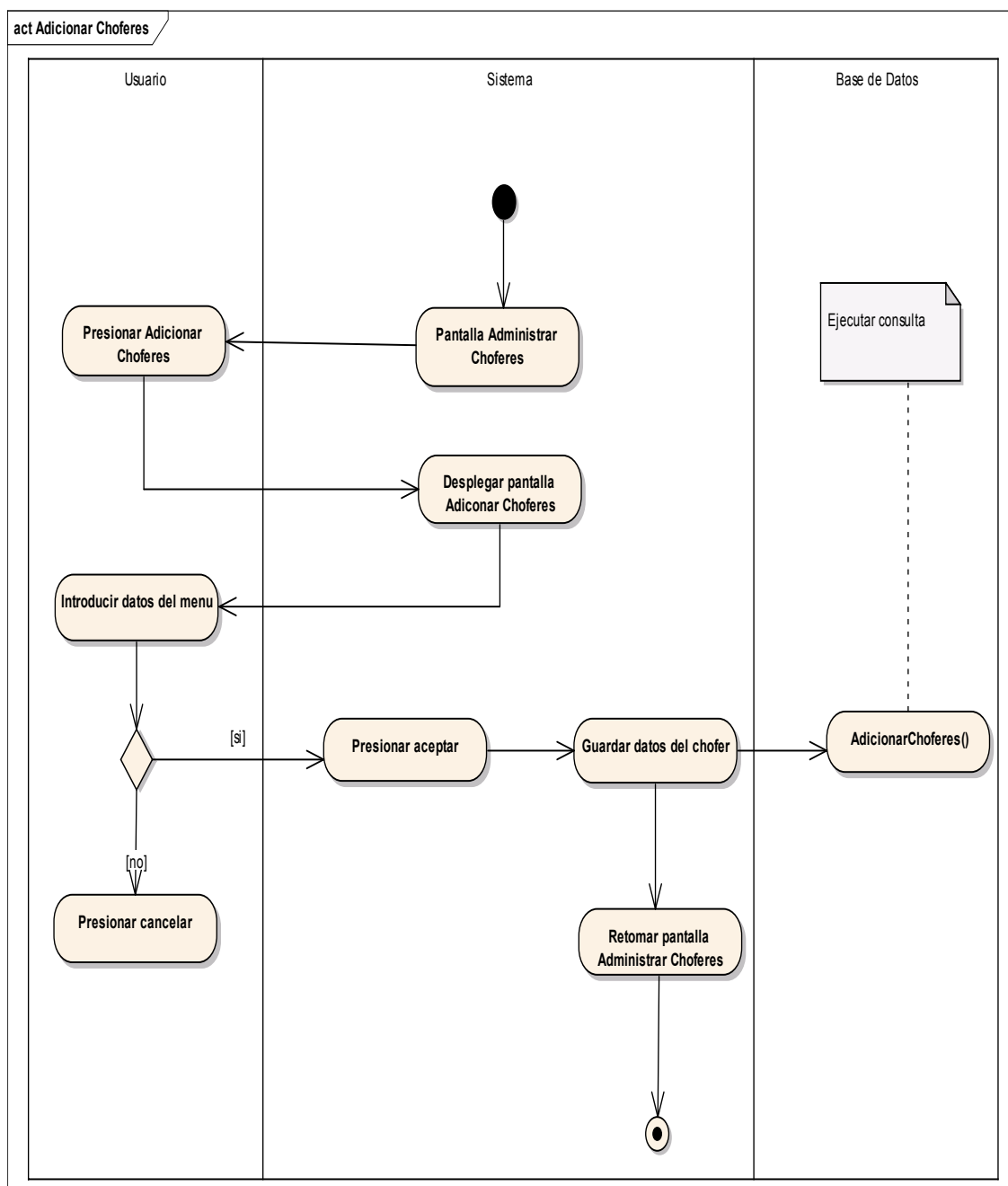
DIAGRAMA DE ACTIVIDAD: ADICIONAR CHOFERES

Figura 114 Diagrama de Actividad: Adicionar Choferes

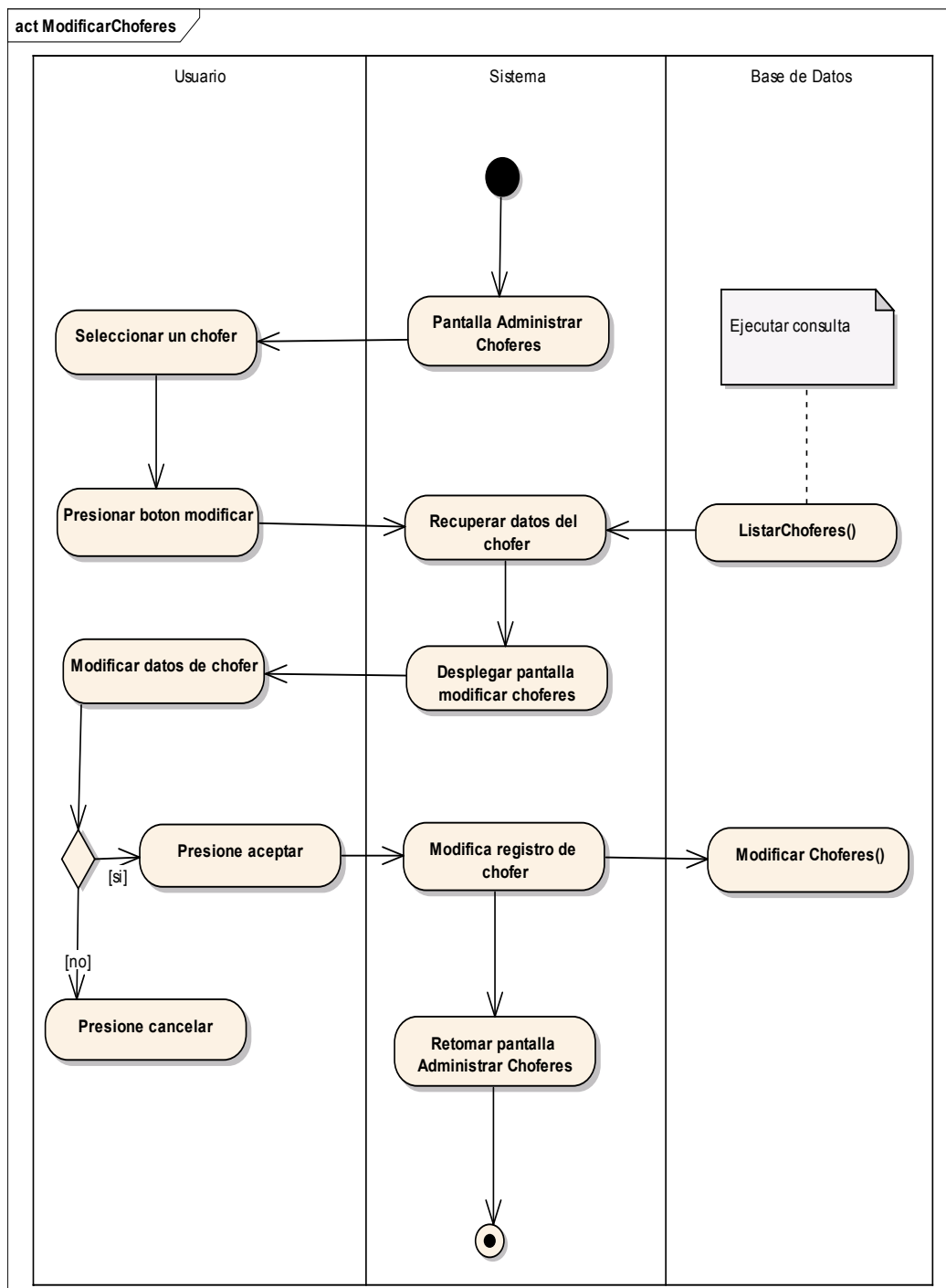
DIAGRAMA DE ACTIVIDAD: MODIFICAR CHOFERES

Figura 115 Diagrama de Actividad: Modificar Choferes

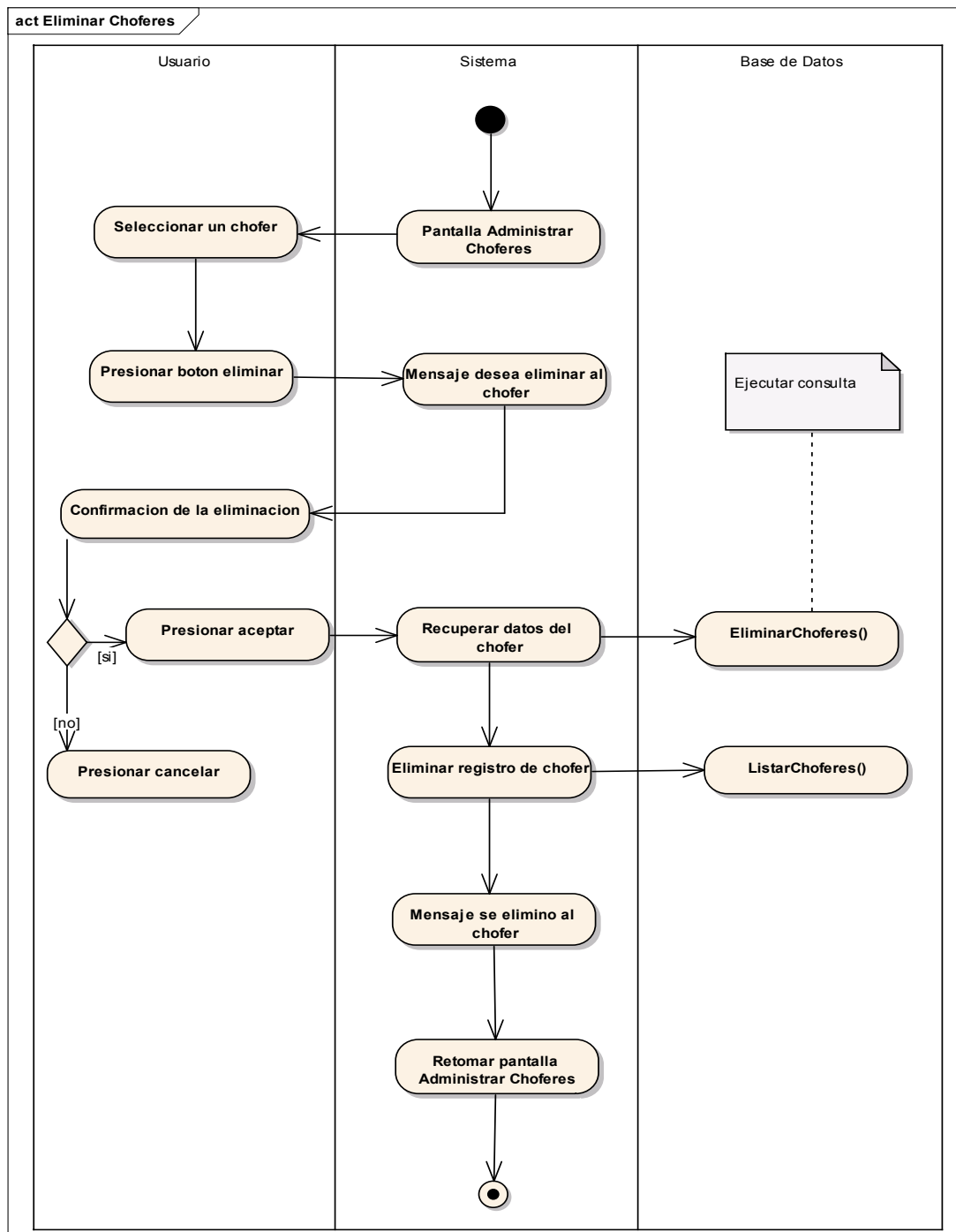
DIAGRAMA DE ACTIVIDAD: ELIMINAR CHOFERES

Figura 116 Diagrama de Actividad: Eliminar Choferes

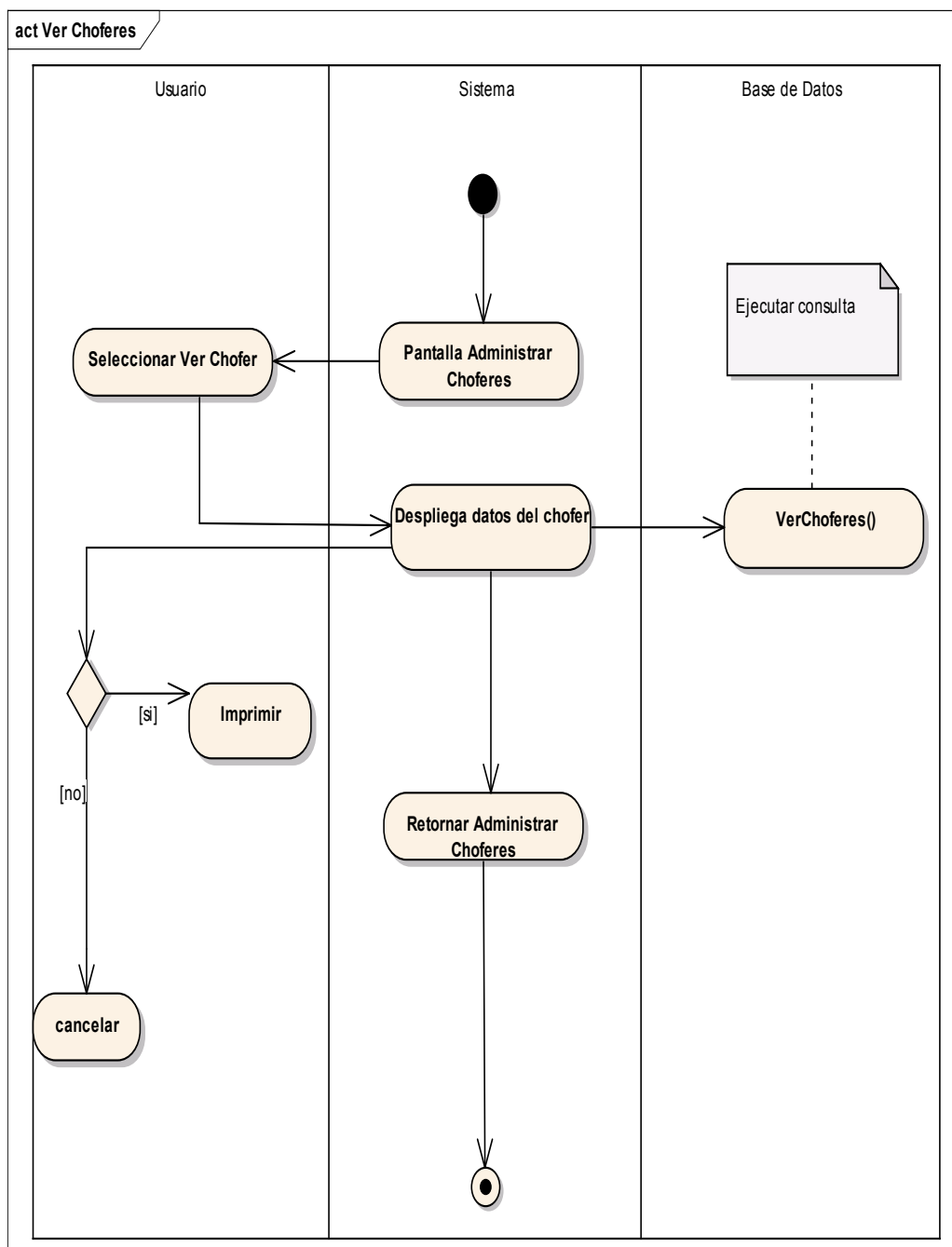
DIAGRAMA DE ACTIVIDAD: VER CHOFERES

Figura 117 Diagrama de Actividad: Ver Choferes

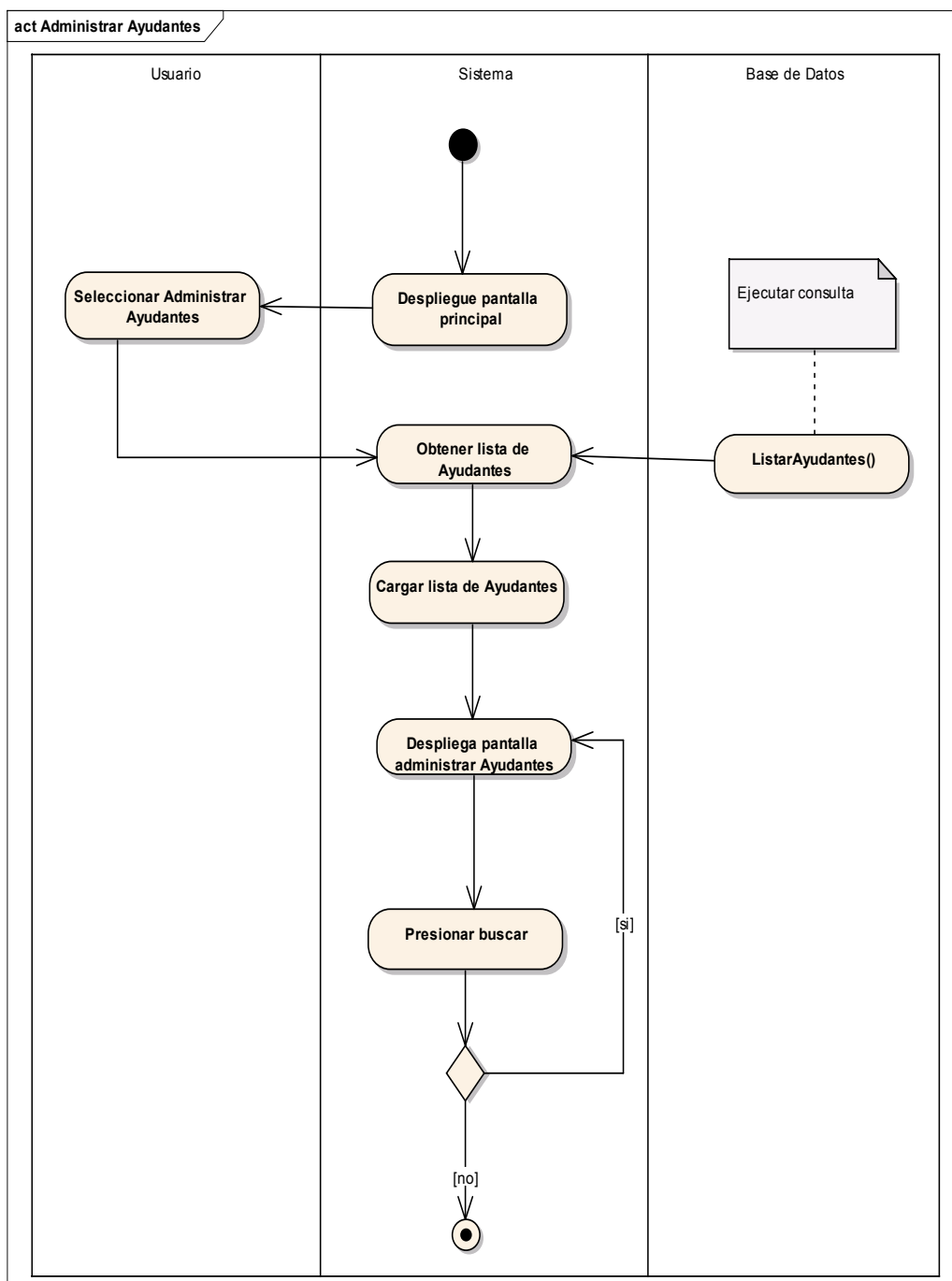
DIAGRAMA DE ACTIVIDAD: ADMINISTRAR AYUDANTES

Figura 118 Diagrama de Actividad: Administrar Ayudantes

DIAGRAMA DE ACTIVIDAD: ADICIONAR AYUDANTES

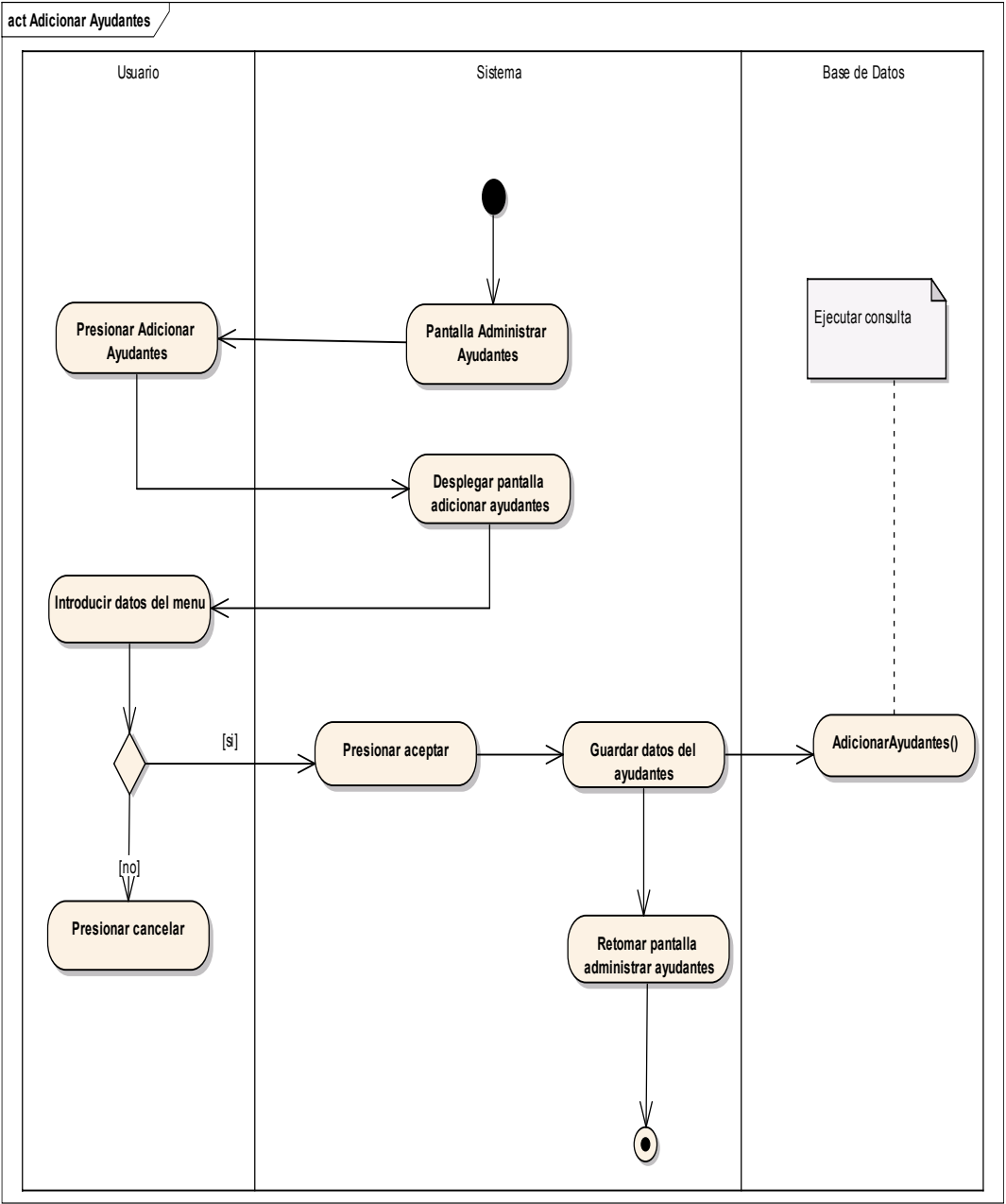


Figura 119 Diagrama de Actividad: Adicionar Ayudantes

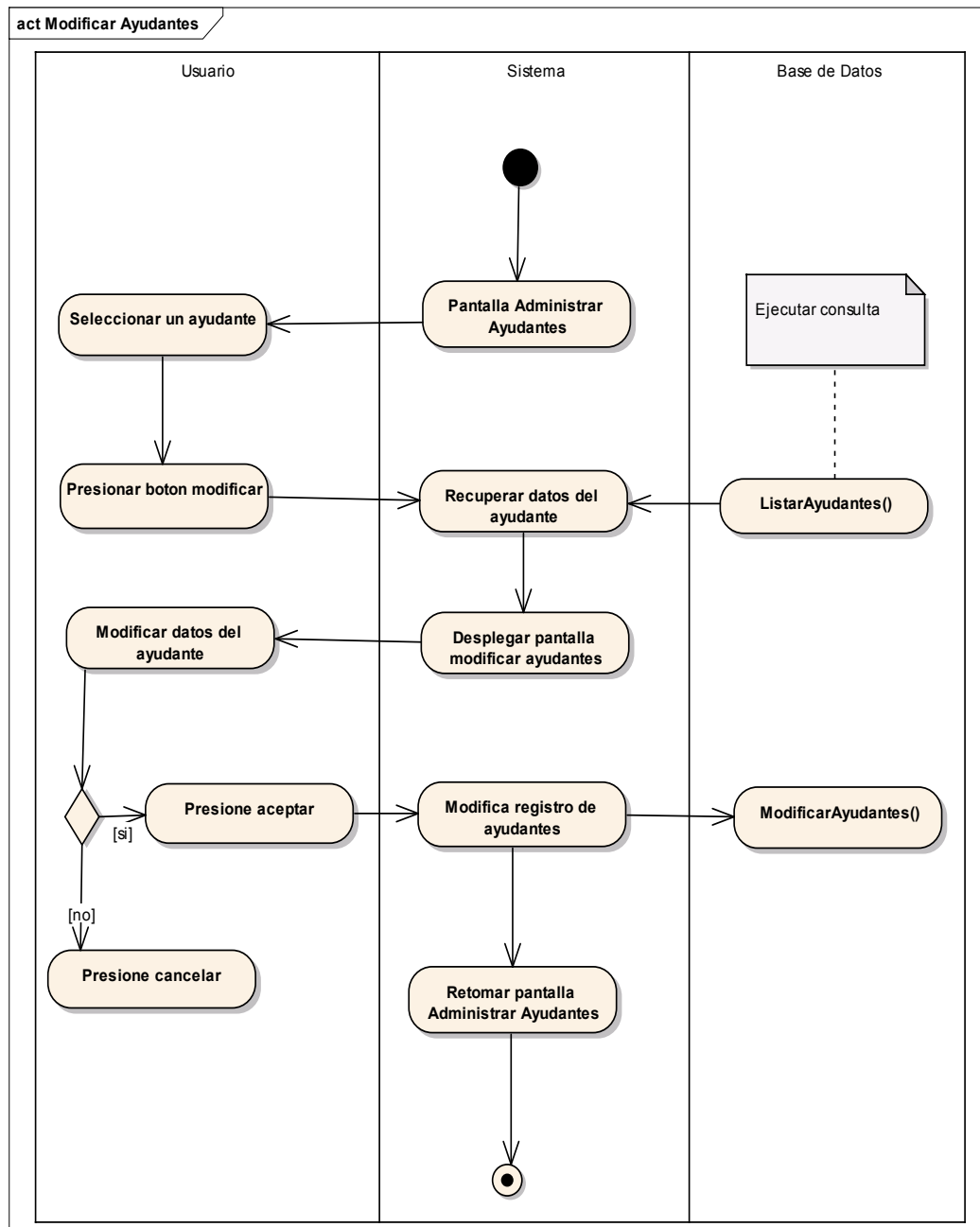
DIAGRAMA DE ACTIVIDAD: MODIFICAR AYUDANTES

Figura 120 Diagrama de Actividad: Modificar Ayudantes

DIAGRAMA DE ACTIVIDAD: ELIMINAR AYUDANTES

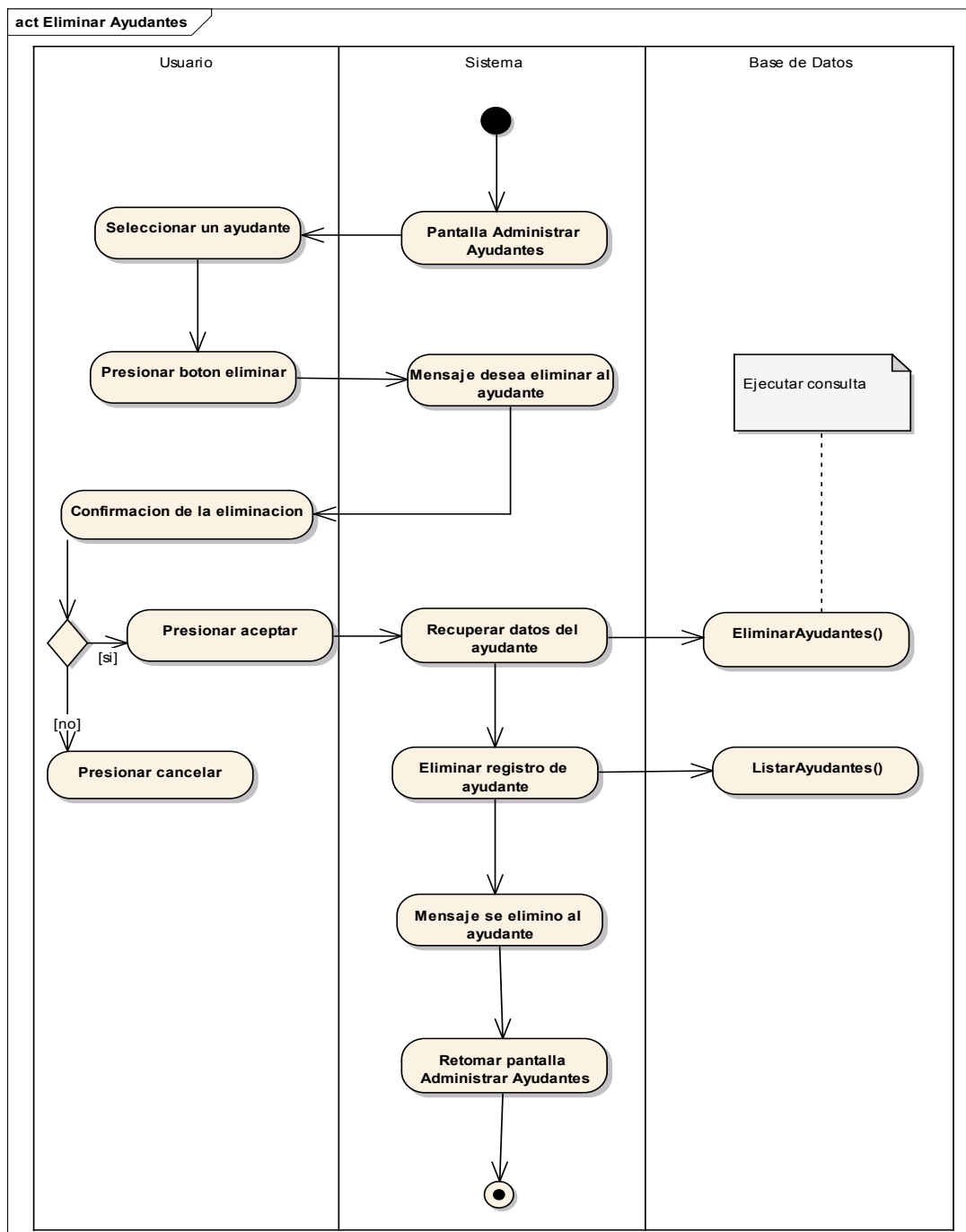


Figura 121 Diagrama de Actividad: Eliminar Ayudantes

DIAGRAMA DE ACTIVIDAD: VER AYUDANTES

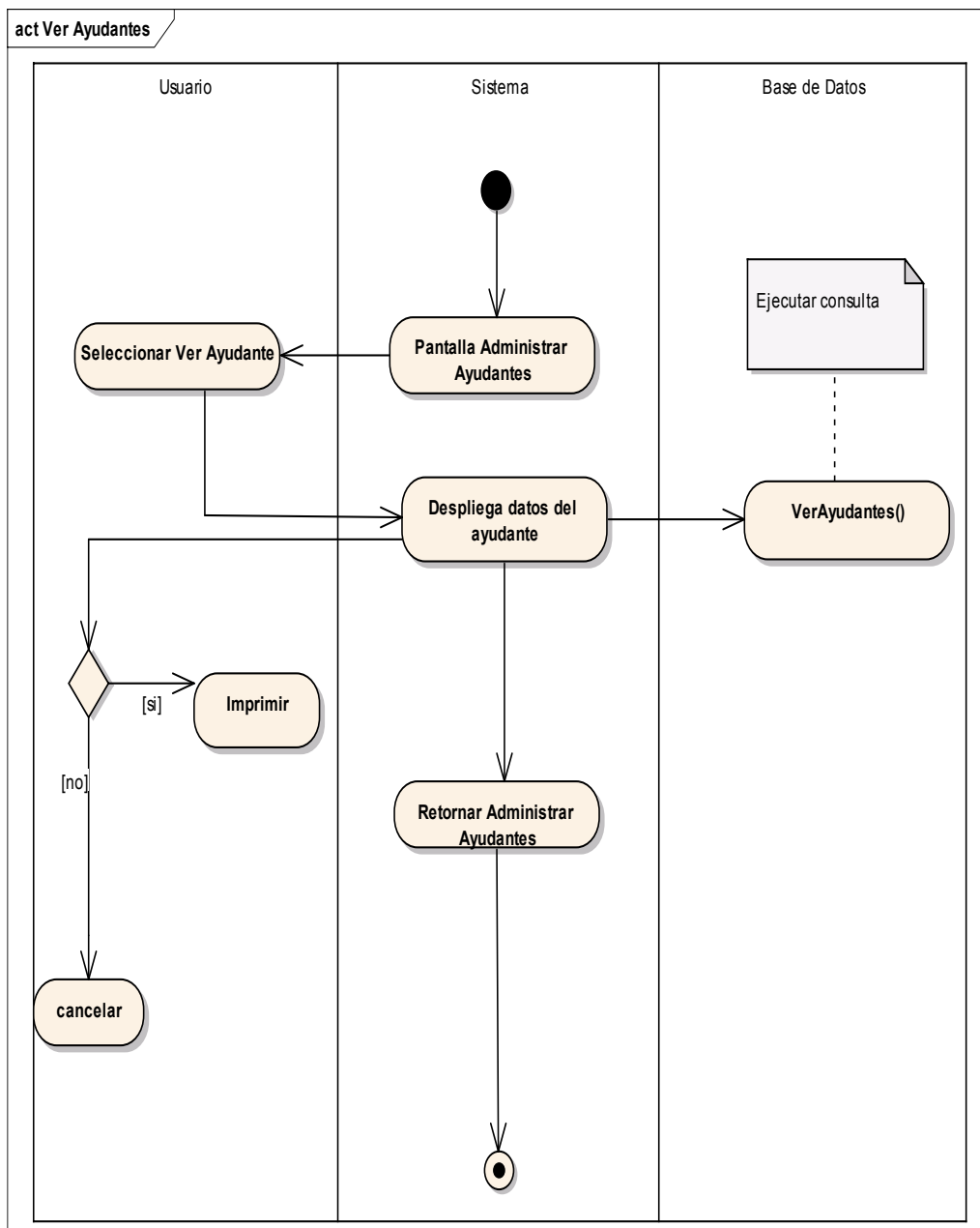


Figura 122 Diagrama de Actividad: Ver Ayudantes

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR ZONAS

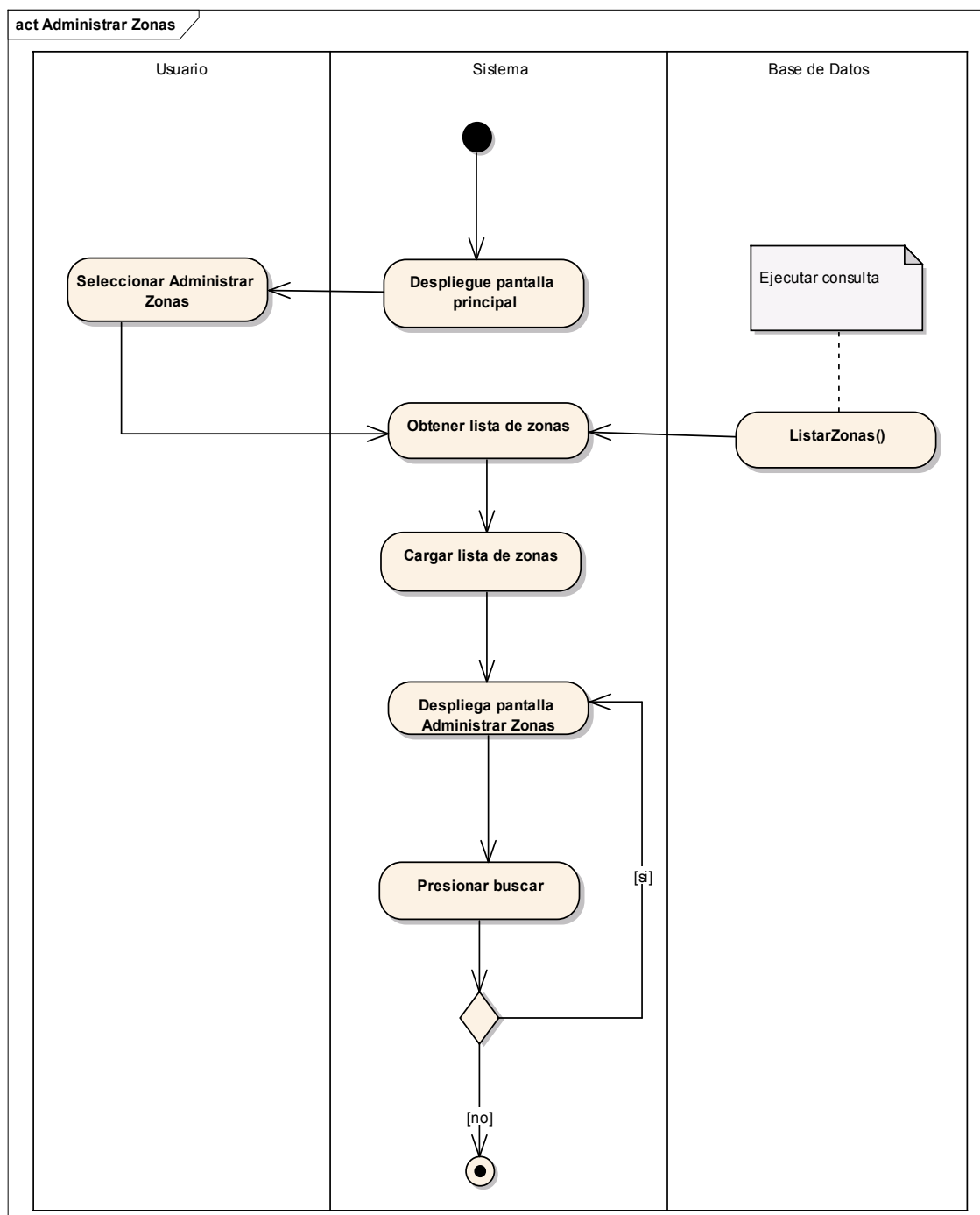


Figura 123 Diagrama de Actividad: Administrar Zonas

DIAGRAMA DE ACTIVIDAD: ADICIONAR ZONAS

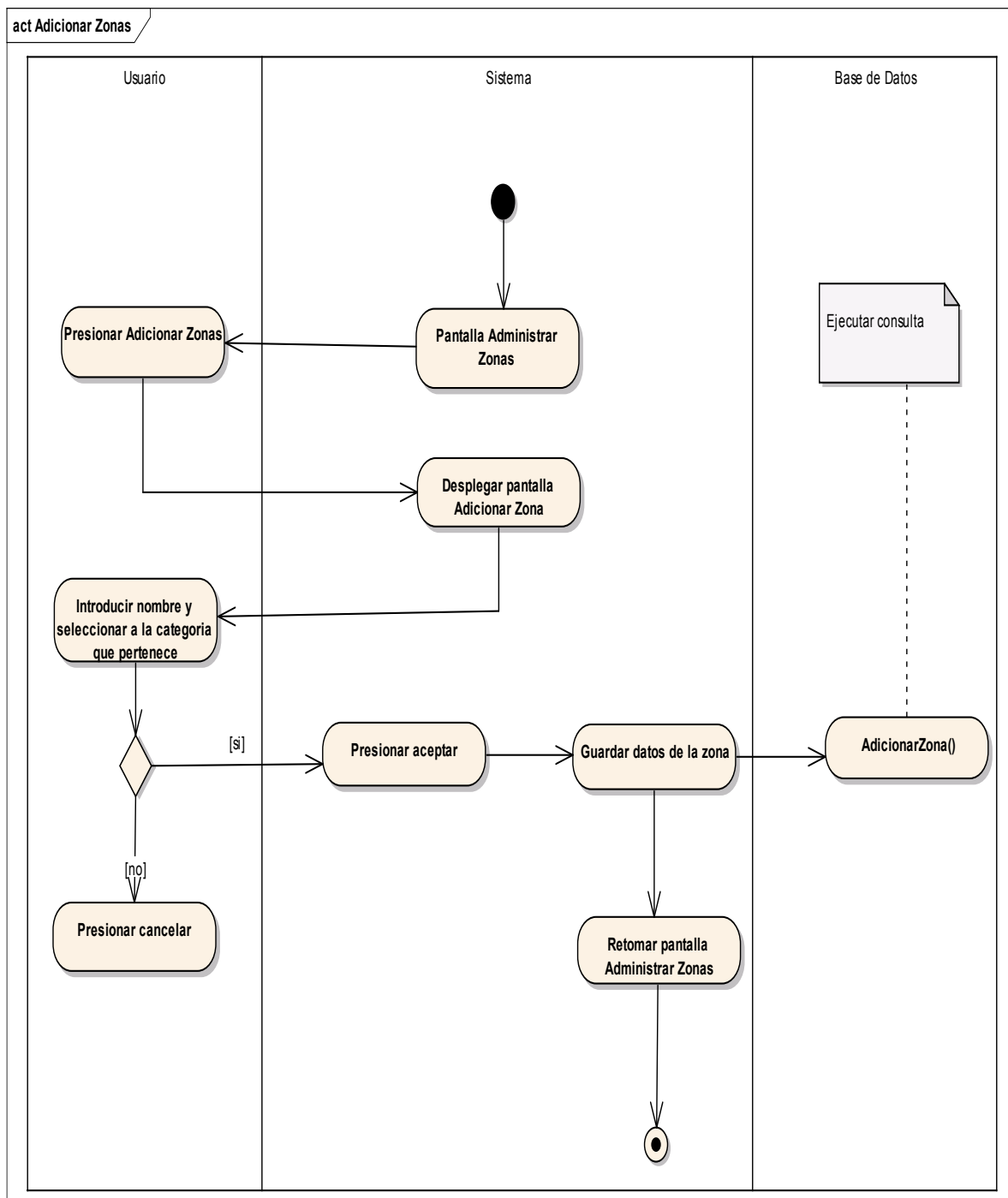


Figura 124 Diagrama de Actividad: Adicionar Zonas

DIAGRAMA DE ACTIVIDAD: MODIFICAR ZONAS

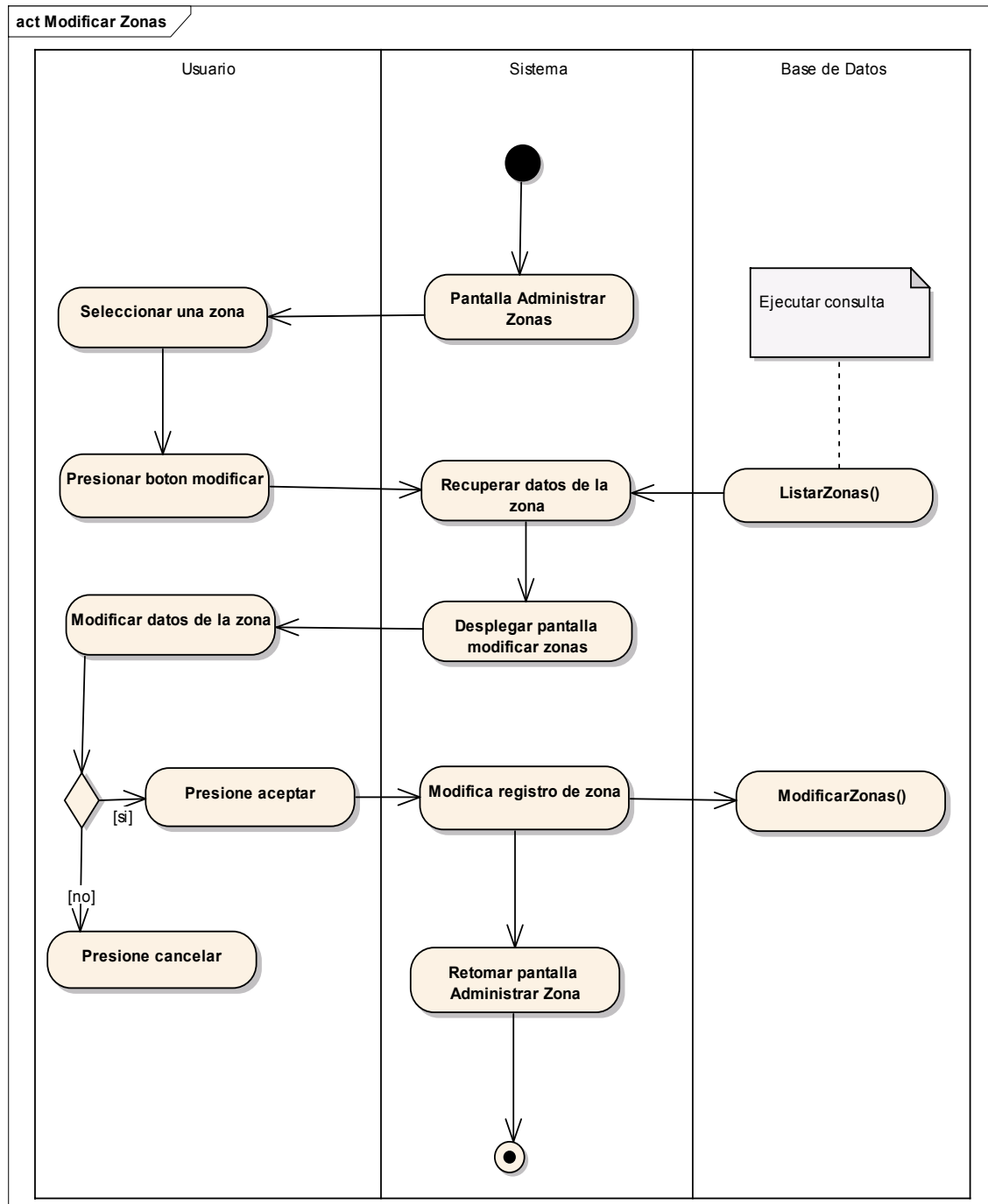


Figura 125 Diagrama de Actividad: Modificar Zonas

DIAGRAMA DE ACTIVIDAD: ELIMINAR ZONAS

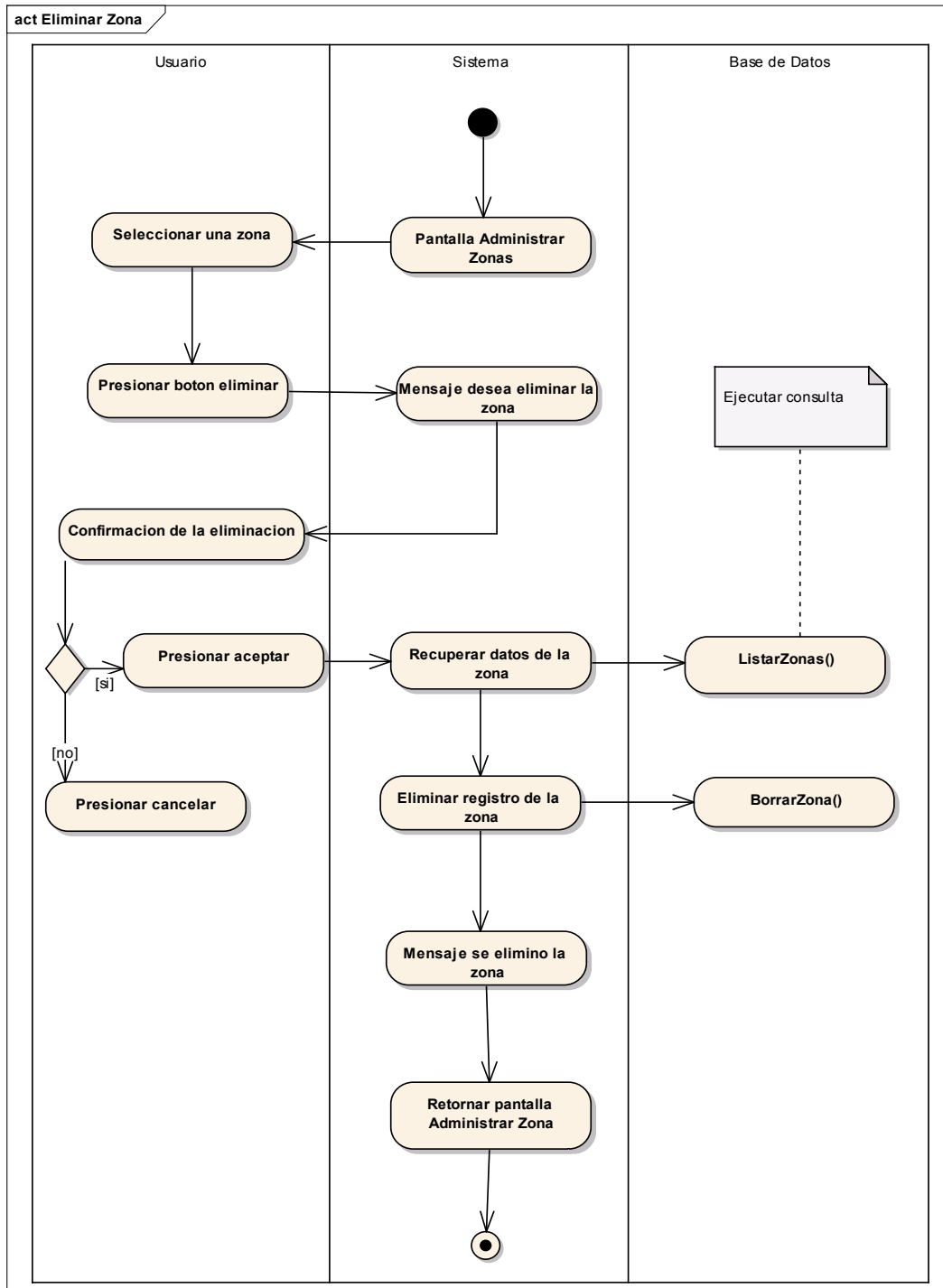


Figura 126 Diagrama de Actividad: Eliminar Zonas

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR VEHICULOS

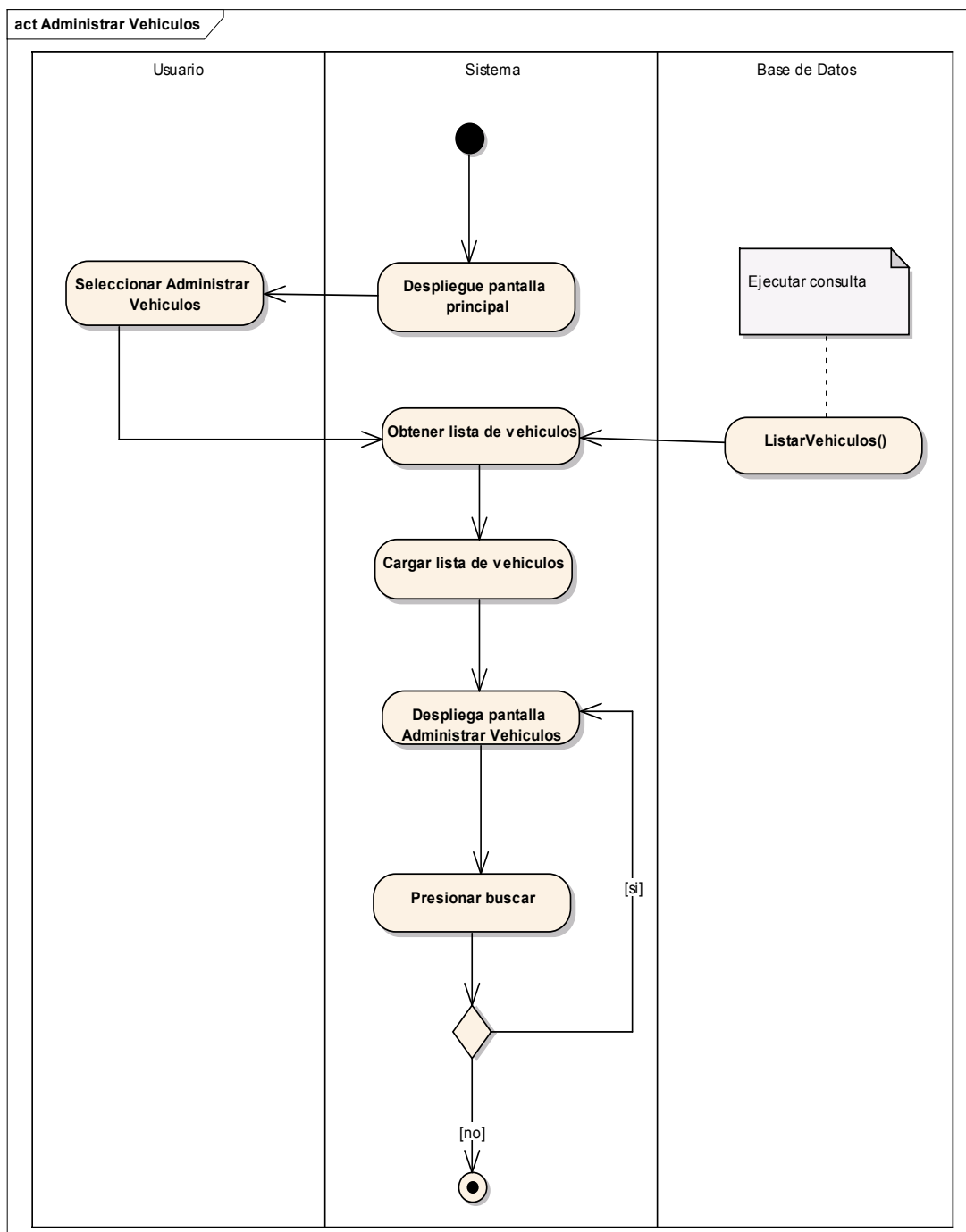


Figura 127 Diagrama de Actividad: Administrar Vehiculos

DIAGRAMA DE ACTIVIDAD: ADICIONAR VEHICULOS

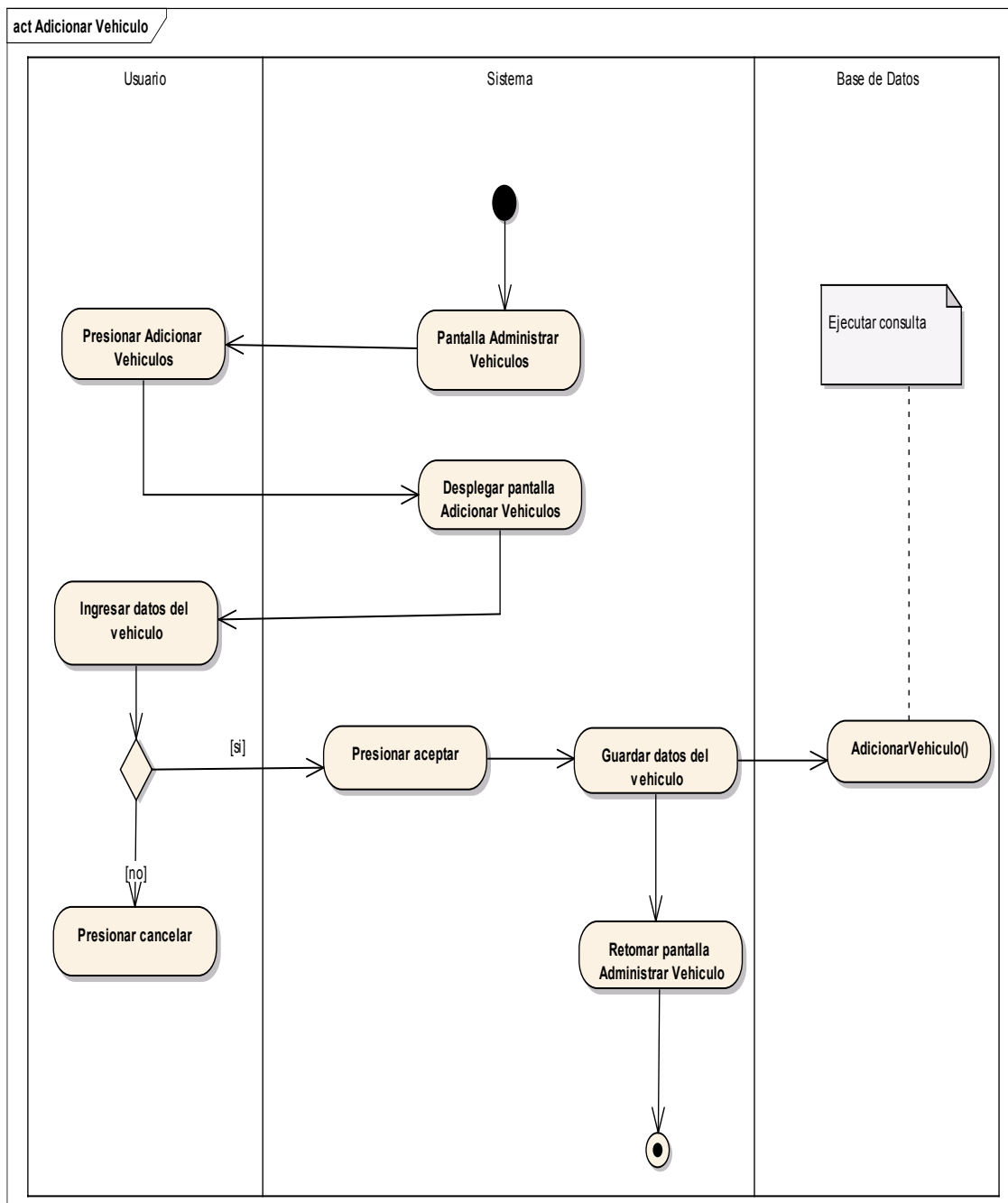


Figura 128 Diagrama de Actividad: Adicionar Vehiculos

DIAGRAMA DE ACTIVIDAD: MODIFICAR VEHICULOS

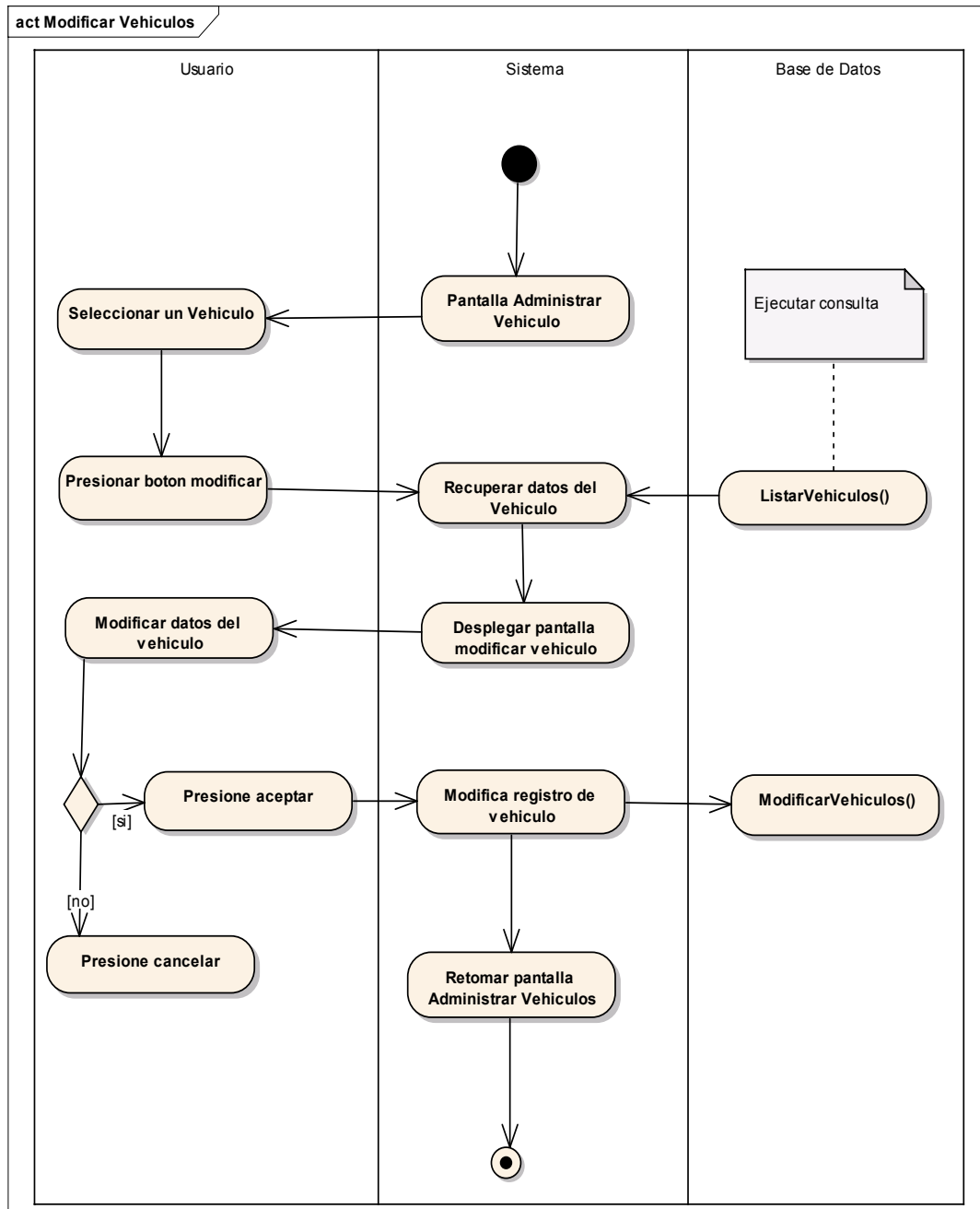


Figura 129 Diagrama de Actividad: Modificar Vehiculos

DIAGRAMA DE ACTIVIDAD: ELIMINAR VEHICULOS

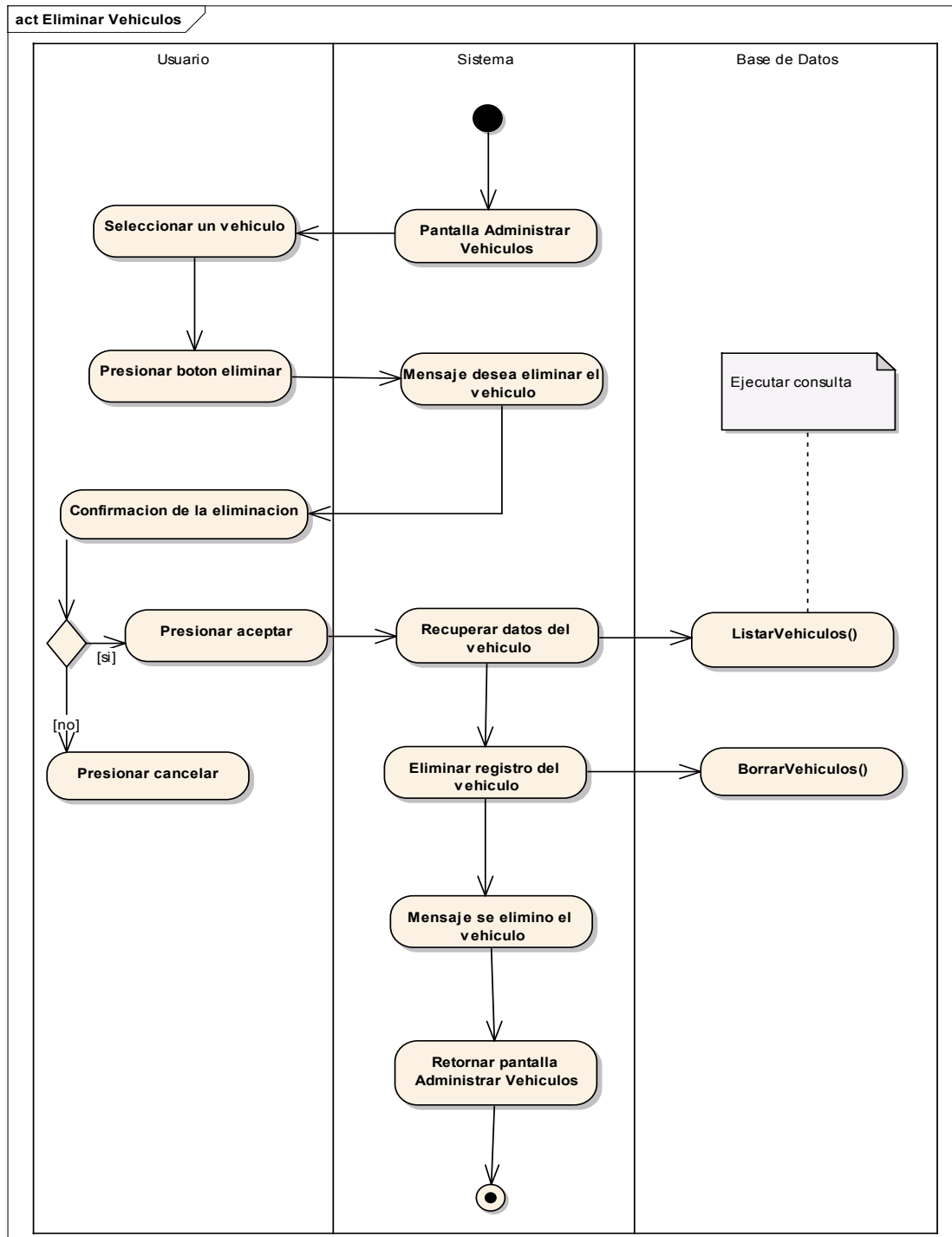


Figura 130 Diagrama de Actividad: Eliminar Vehiculos

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR BITACORAS

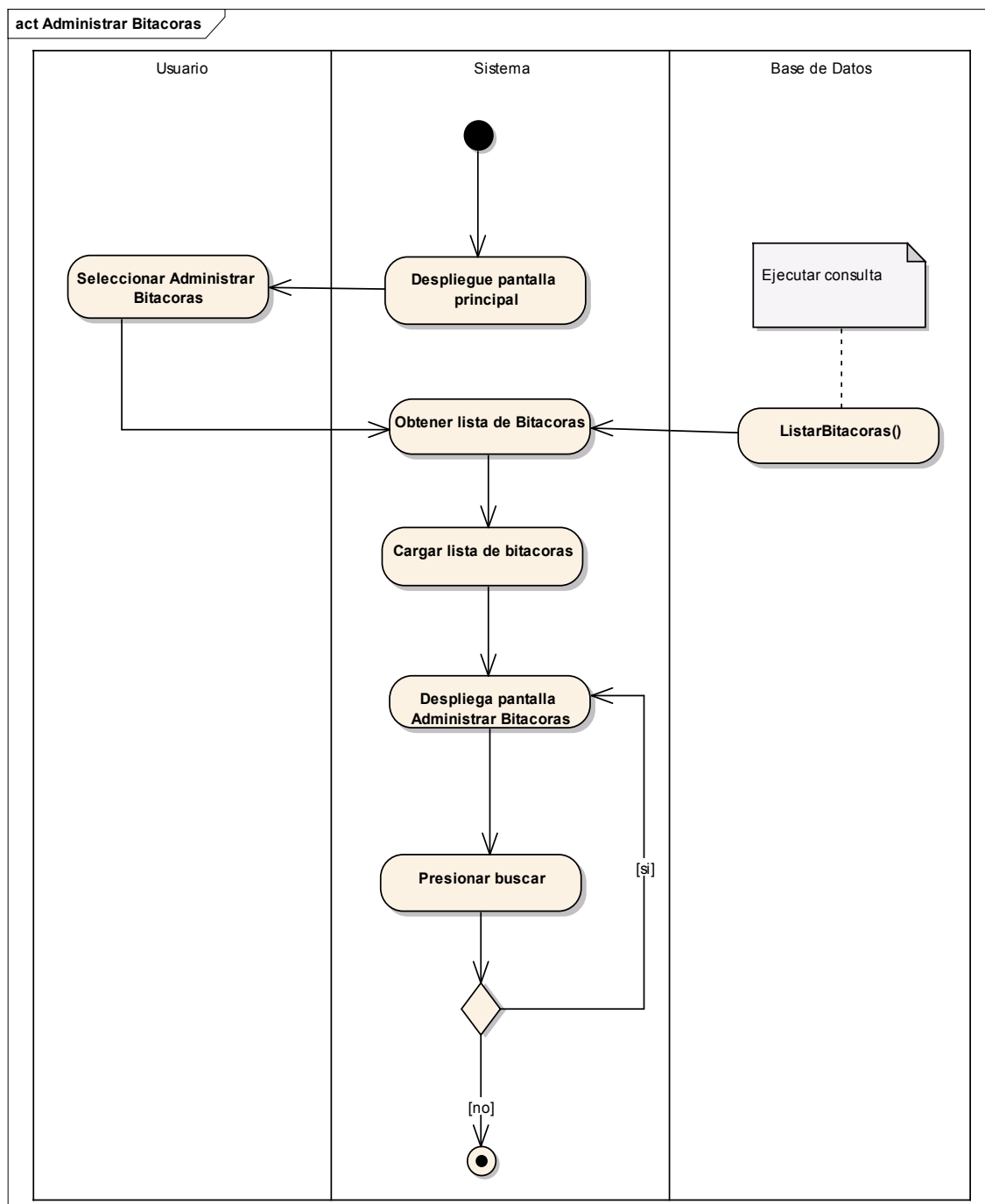


Figura 131 Diagrama de Actividad: Administrar Bitacoras

DIAGRAMA DE ACTIVIDAD: ADICIONAR BITACORAS

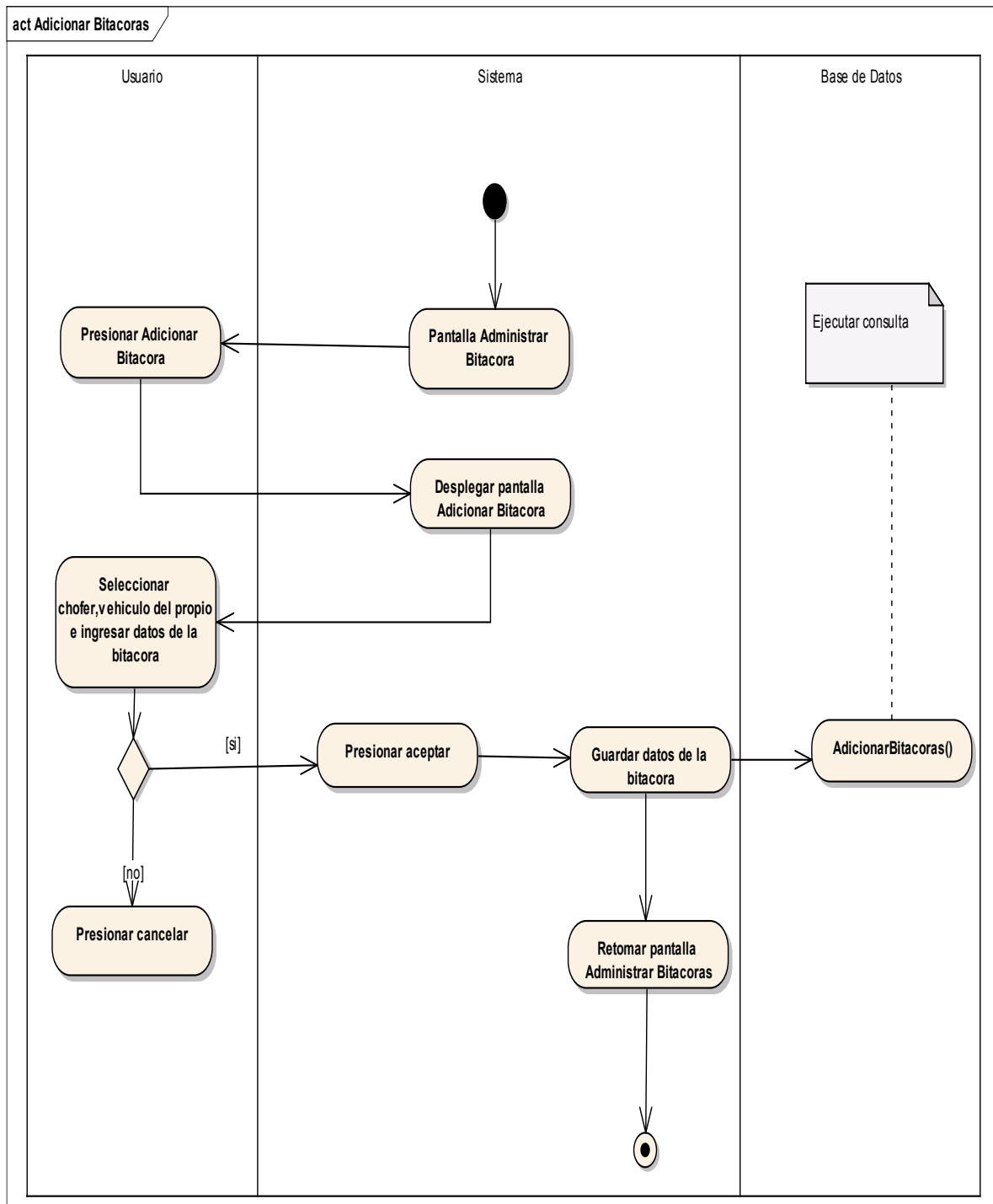


Figura 132 Diagrama de Actividad: Adicionar Bitacoras

DIAGRAMA DE ACTIVIDAD: MODIFICAR BITACORAS

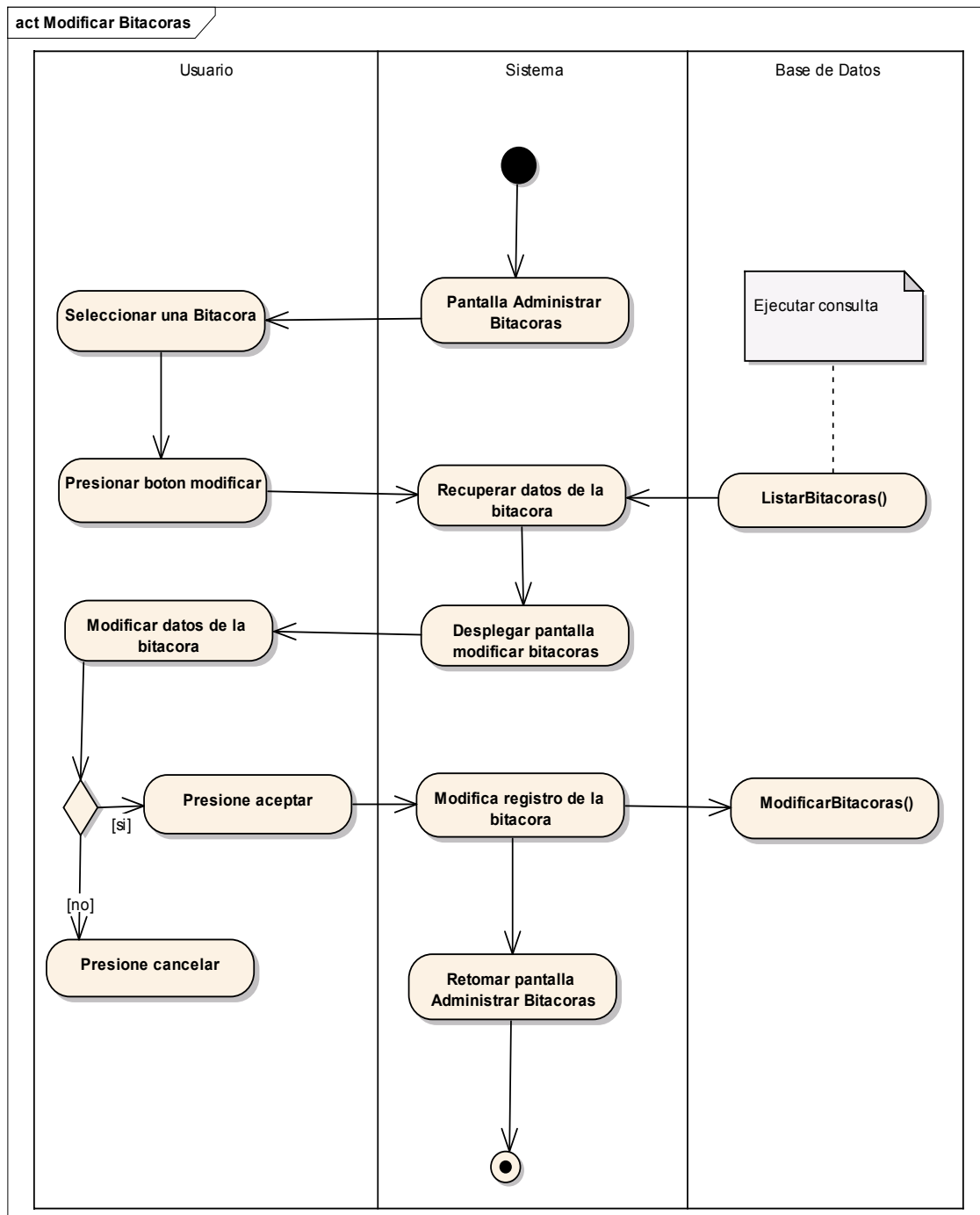


Figura 133 Diagrama de Actividad: Modificar Bitacoras

DIAGRAMA DE ACTIVIDAD: ELIMINAR BITACORAS

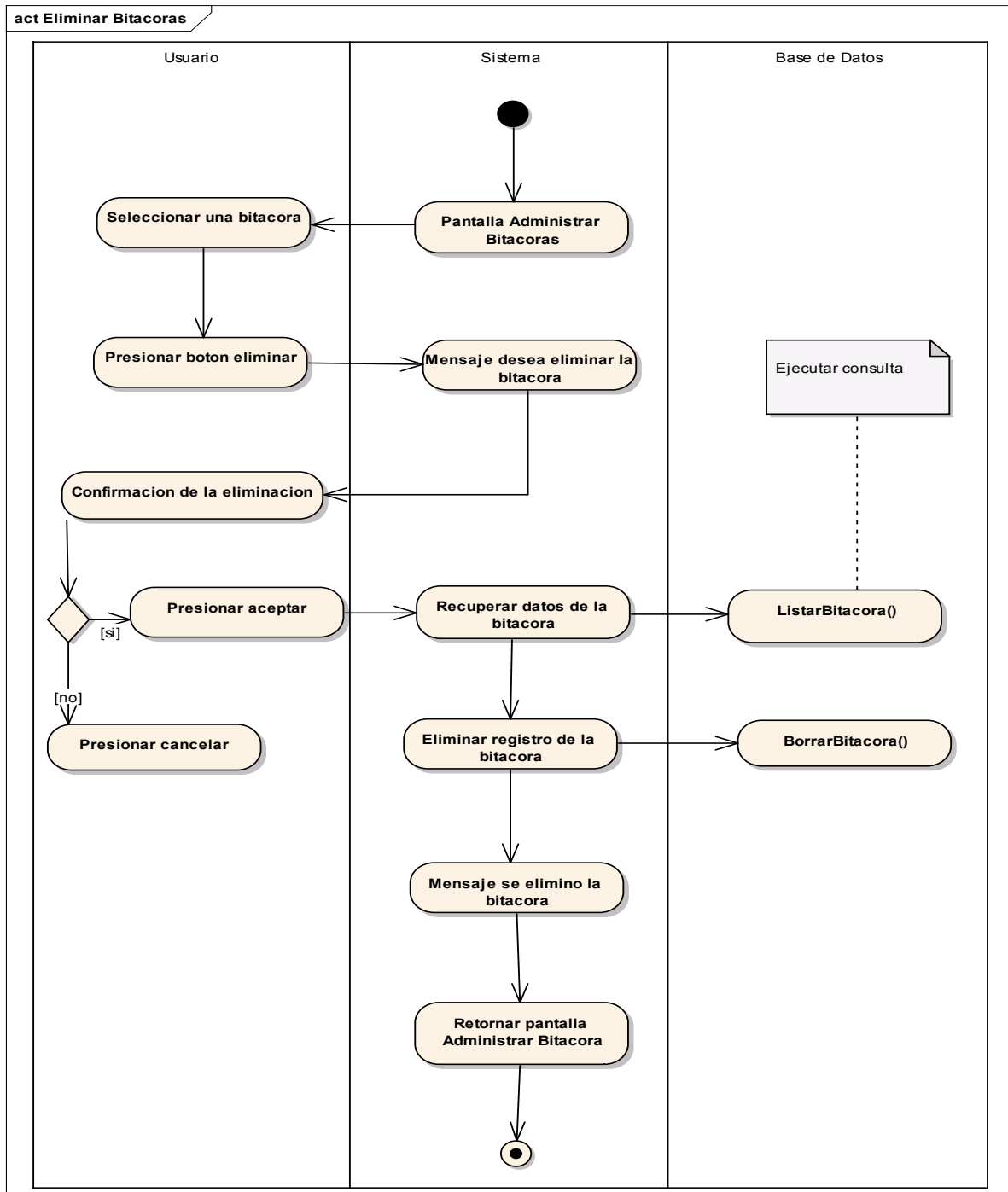


Figura 134 Diagrama de Actividad: Eliminar Bitacoras

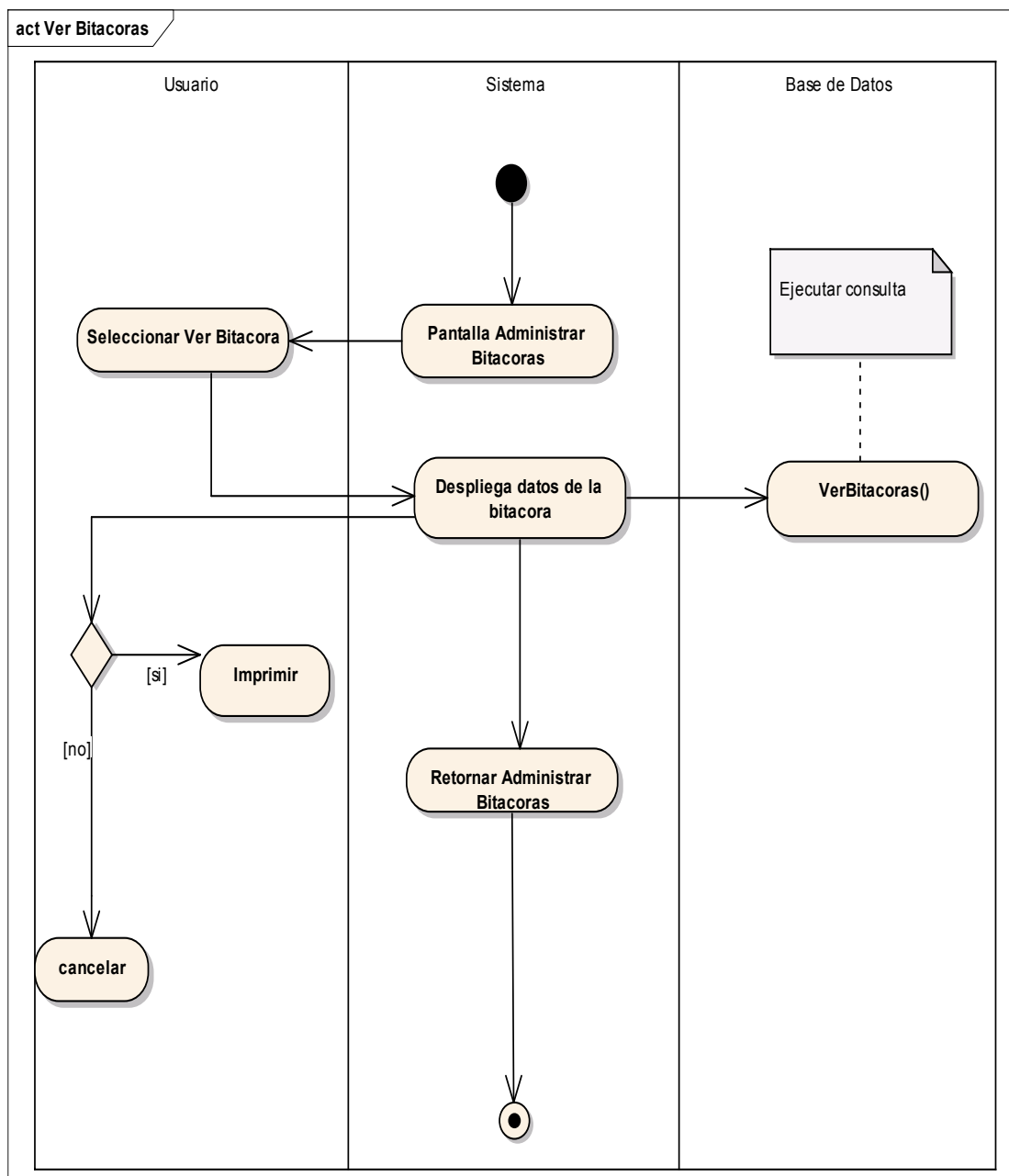
DIAGRAMA DE ACTIVIDAD: VER BITACORAS

Figura 135 Diagrama de Actividad: Ver Bitacoras

DIAGRAMA DE ACTIVIDAD: ADMINISTRAR VIAJES

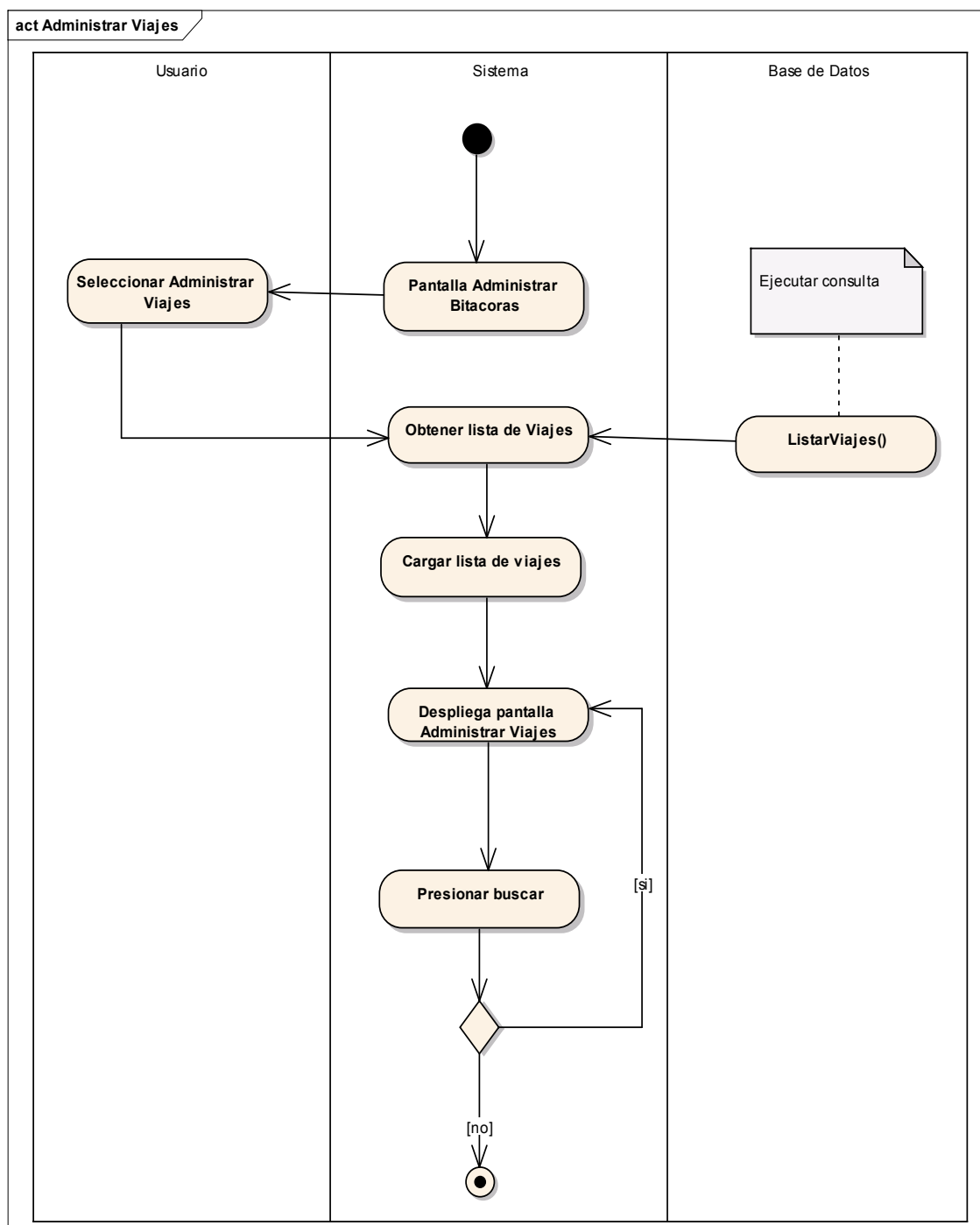


Figura 136 Diagrama de Actividad: Administrar Viajes

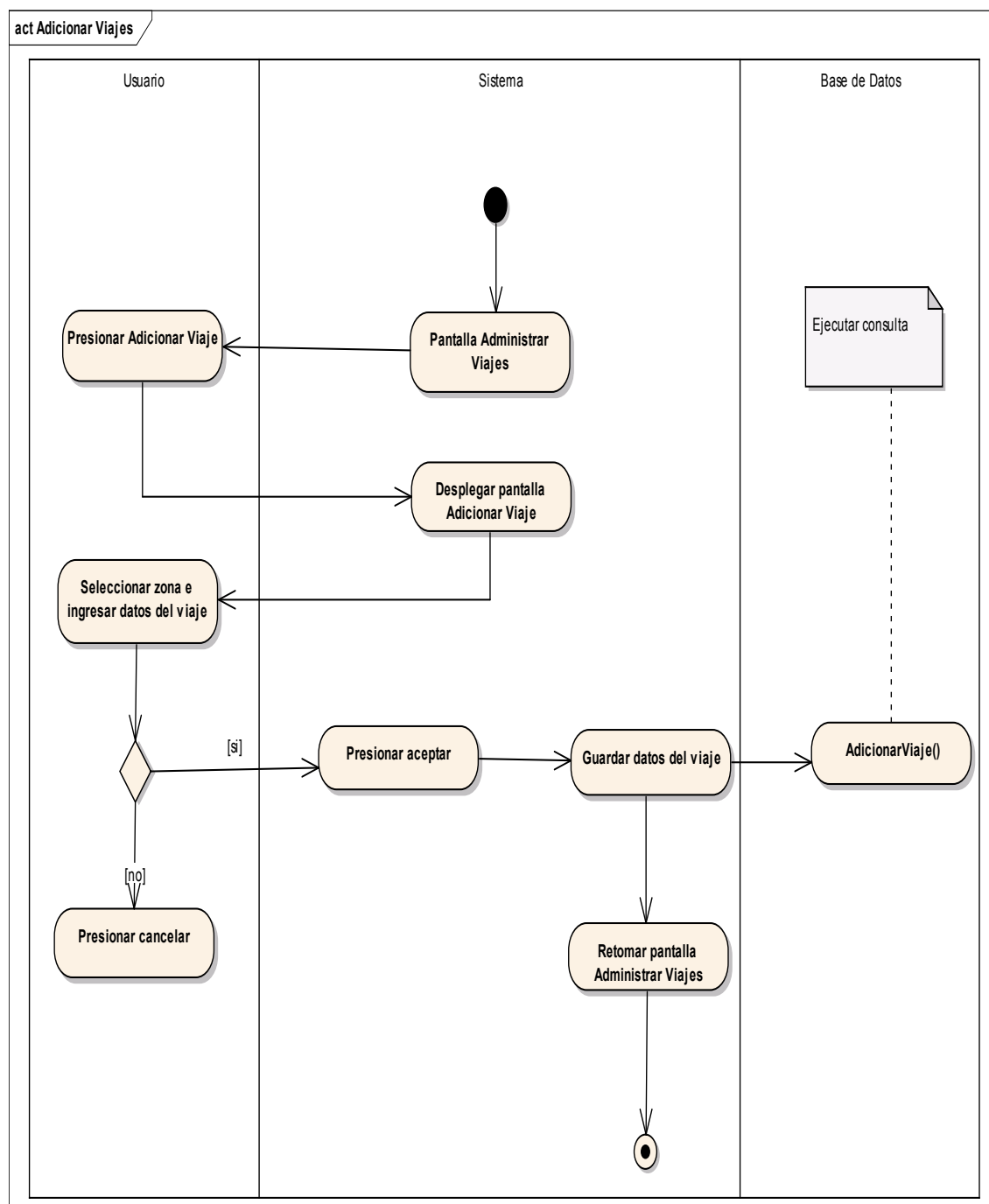
DIAGRAMA DE ACTIVIDAD: ADICIONAR VIAJES

Figura 137 Diagrama de Actividad: Adicionar Viajes

DIAGRAMA DE ACTIVIDAD: MODIFICAR VIAJES

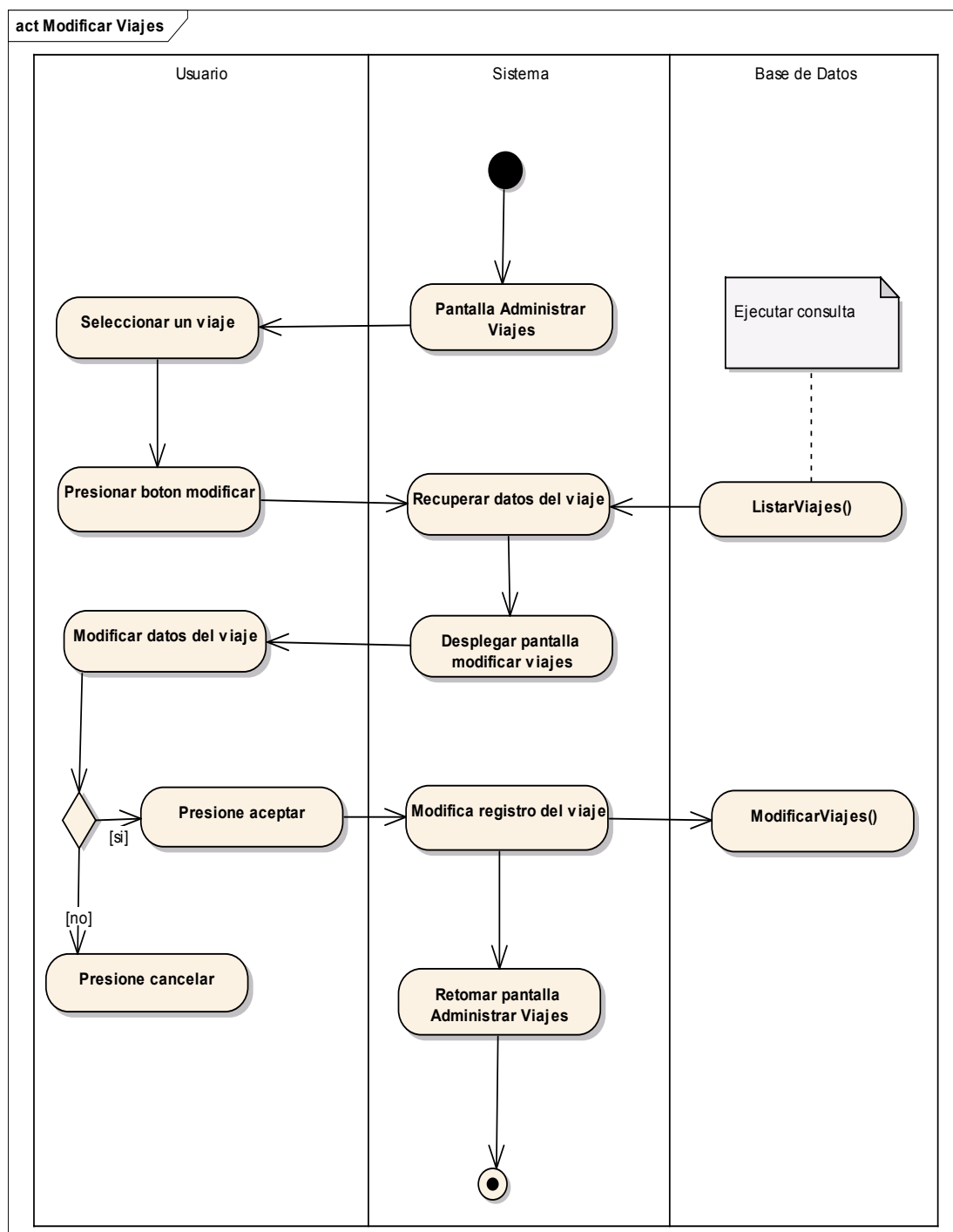


Figura 138 Diagrama de Actividad: Modificar Viajes

DIAGRAMA DE ACTIVIDAD: ELIMINAR VIAJES

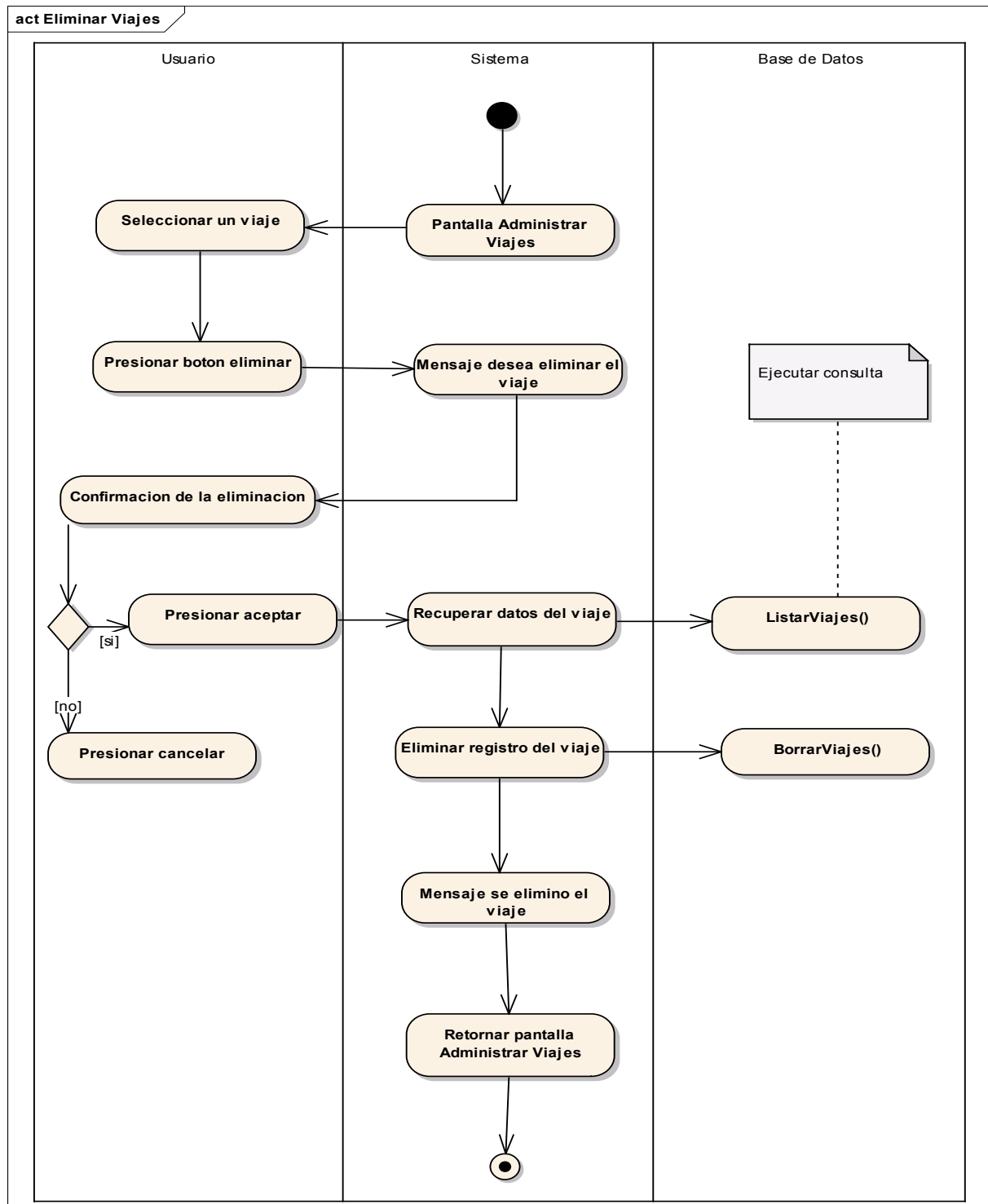


Figura 139 Diagrama de Actividad: Eliminar Viajes

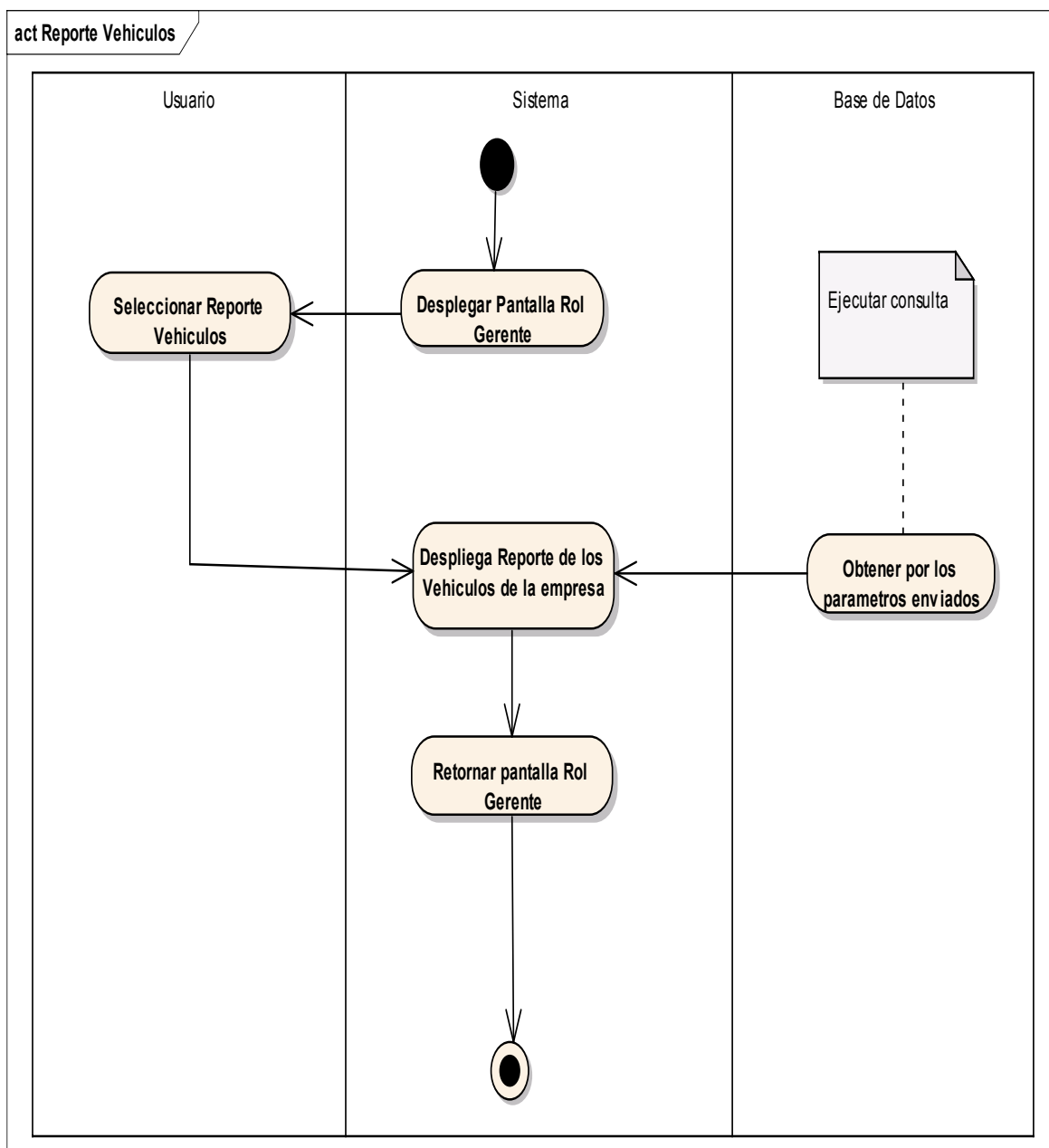
DIAGRAMA DE ACTIVIDAD: REPORTES VEHICULOS

Figura 145 Diagrama de Actividad: Reportes Vehiculos

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS

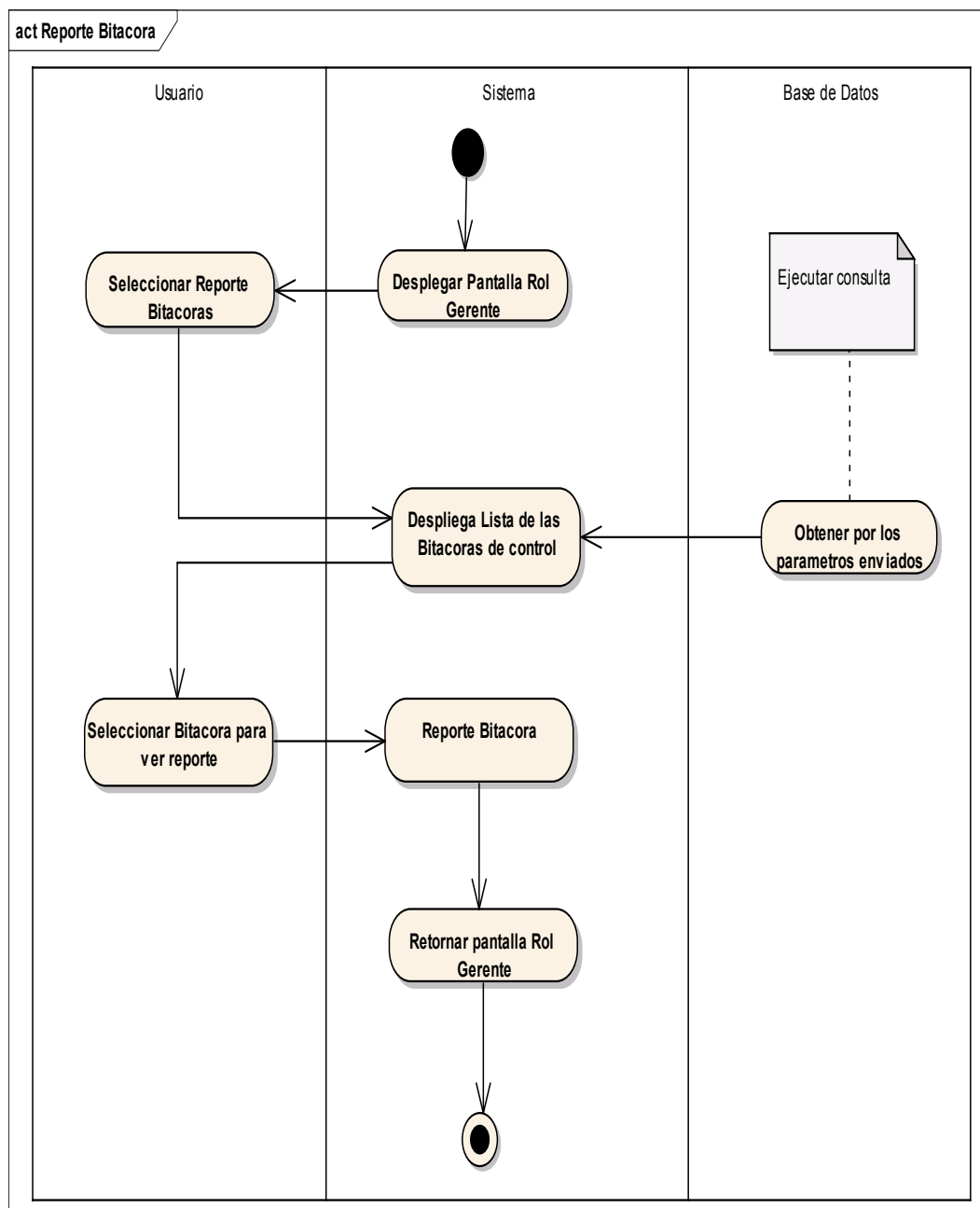


Figura 146 Diagrama de Actividad: Reportes Bitacoras

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / DIURNO POR AÑO

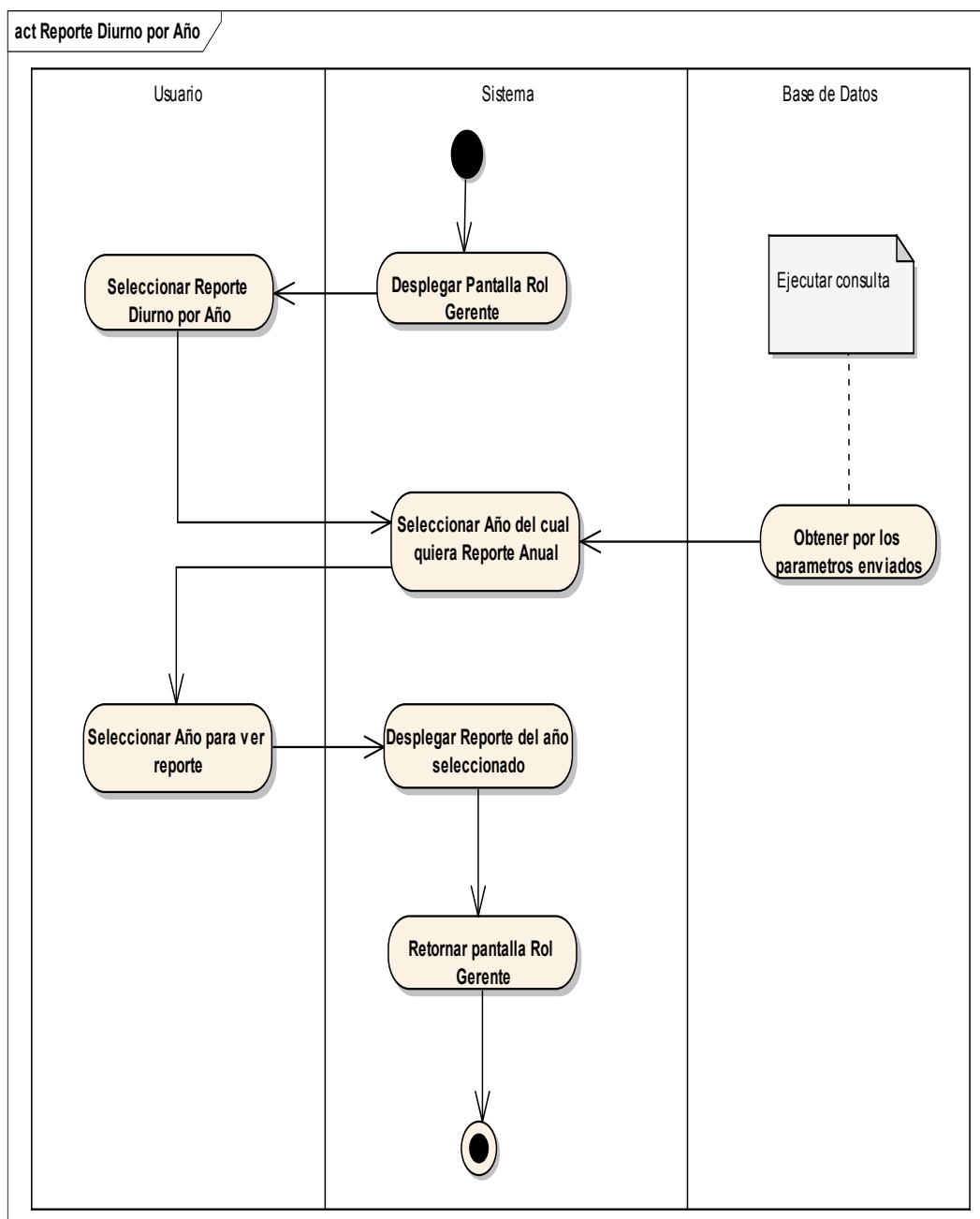


Figura 147 Diagrama de Actividad: Reportes Bitacoras / Diurno por Año

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / NOCTURNO POR AÑO

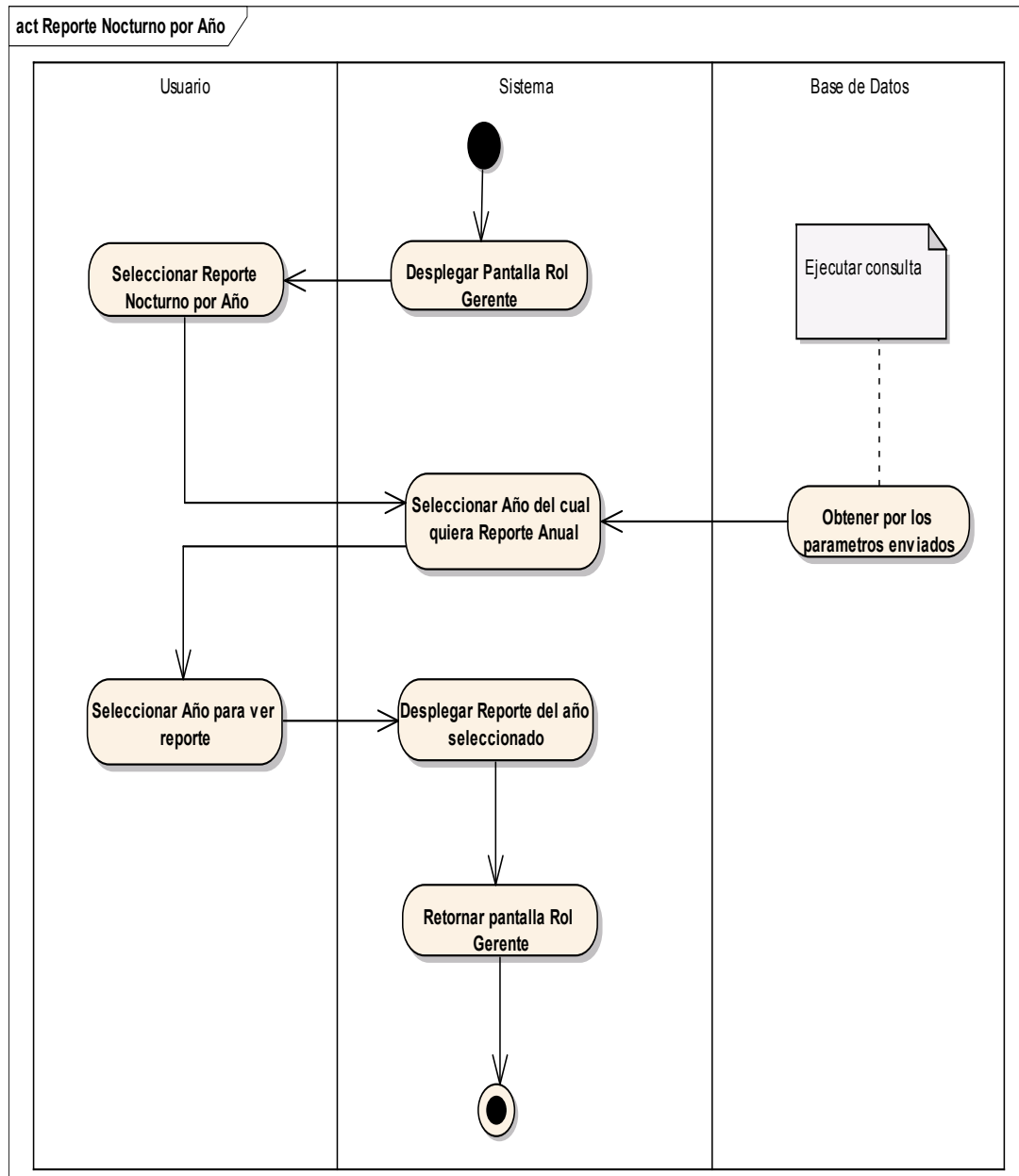


Figura 148 Diagrama de Actividad: Reportes Bitacoras / Nocturno por Año

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / DIURNO POR MES Y AÑO

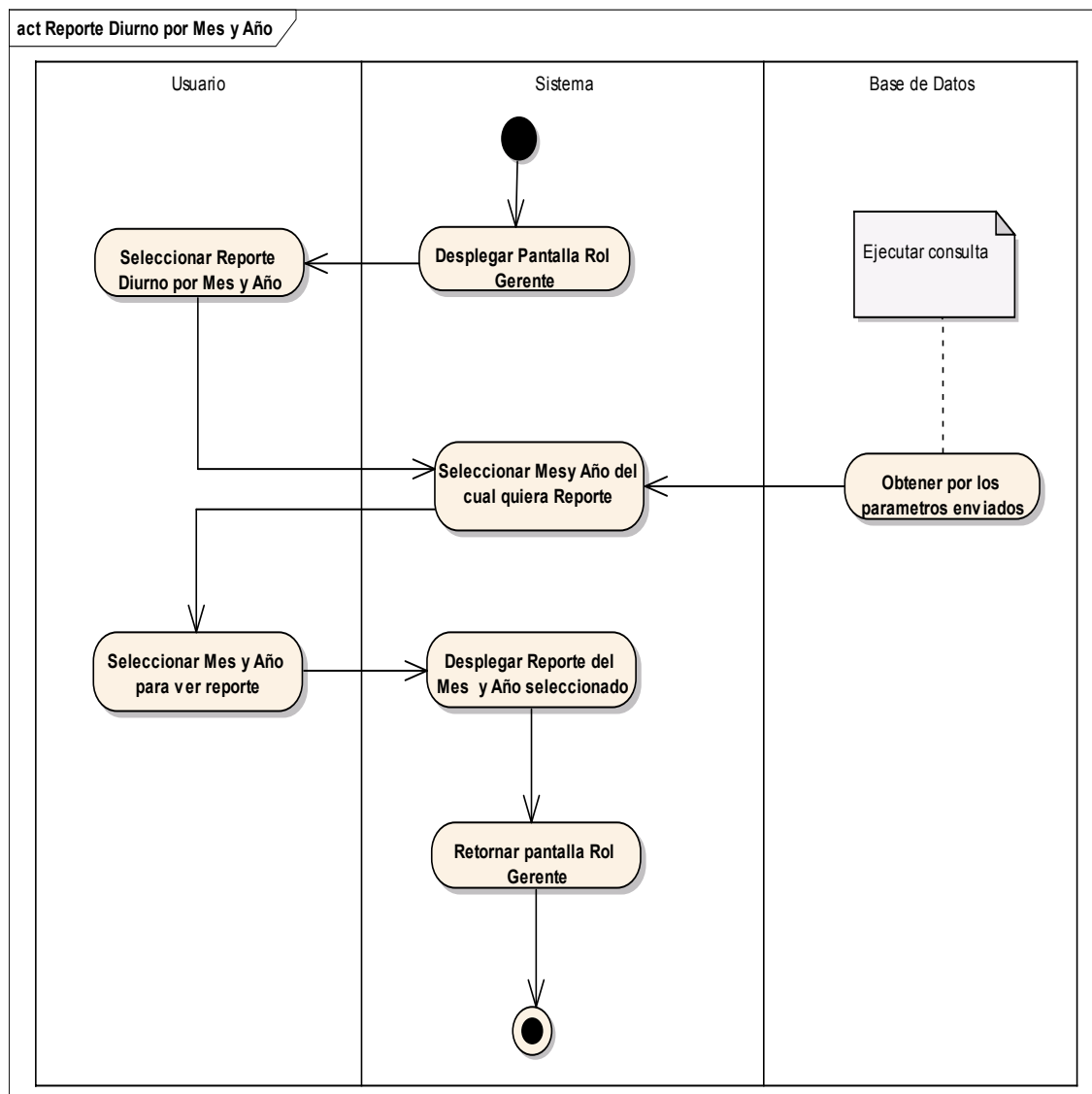


Figura 149 Diagrama de Actividad: Reportes Bitacoras / Diurno por Mes y Año

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / NOCTURNO POR MES Y AÑO

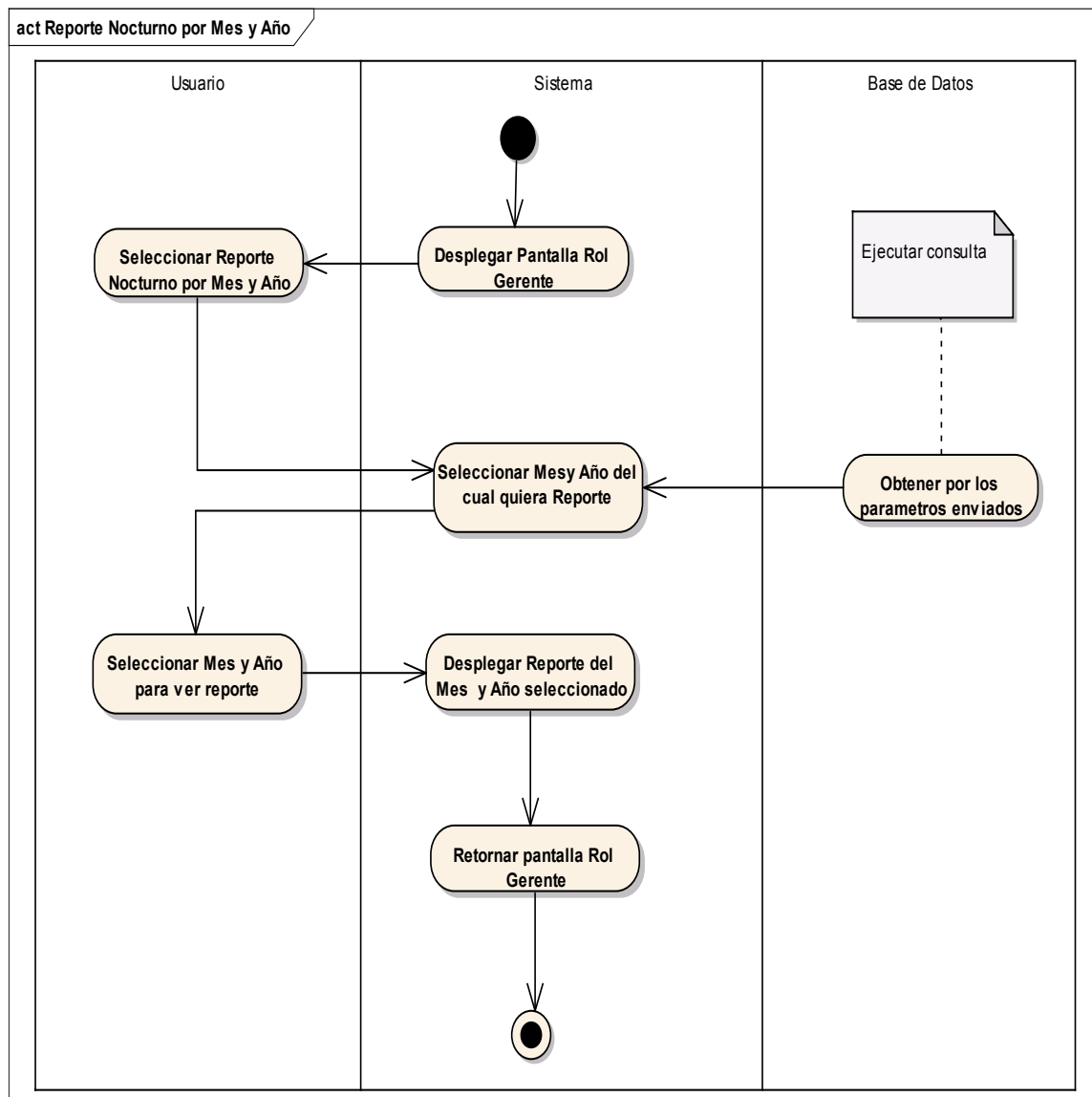


Figura 150 Diagrama de Actividad: Reportes Bitacoras / Nocturno por Mes y Año

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / TOTAL POR AÑO Y CATEGORIA

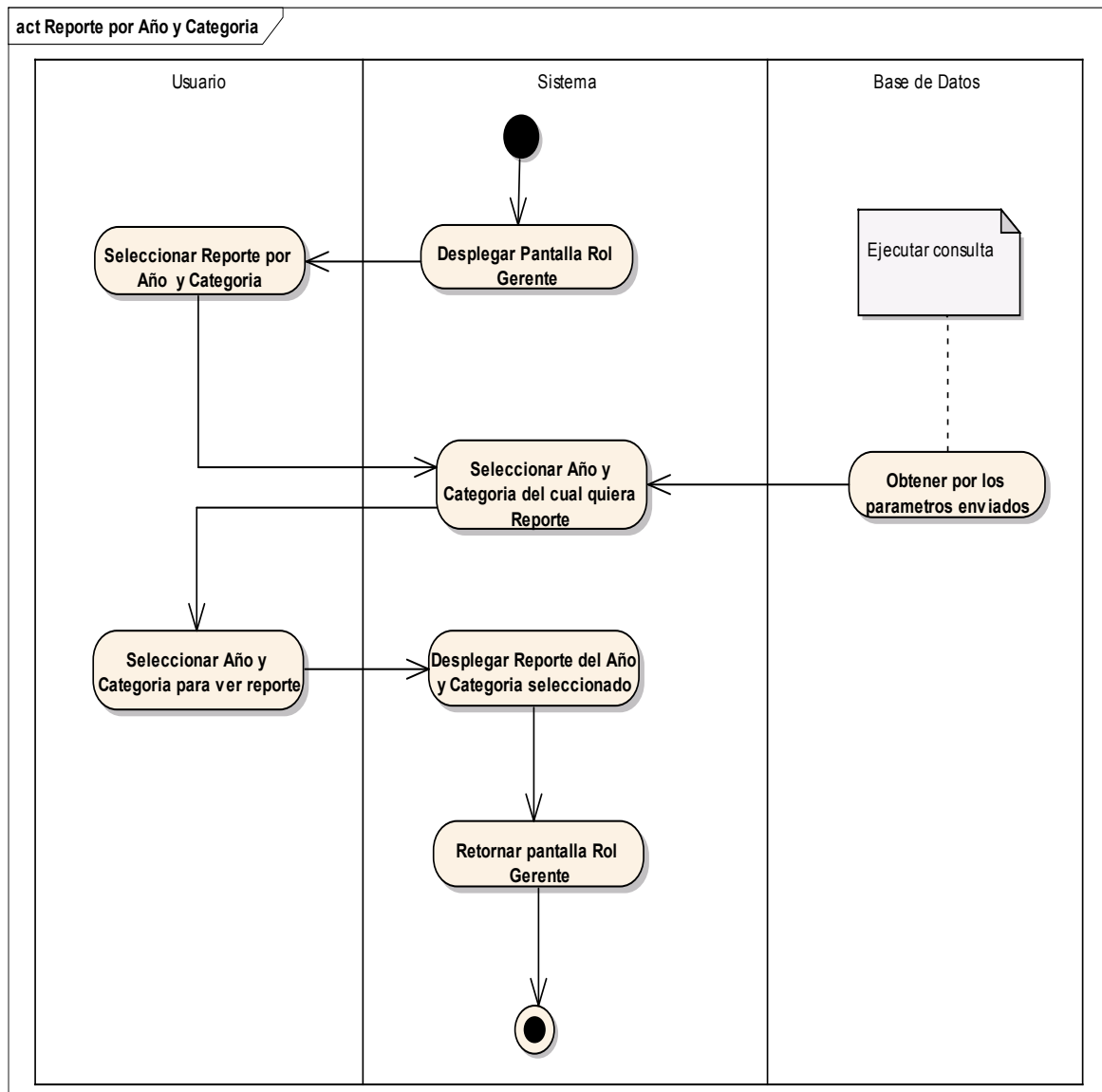


Figura 151 Diagrama de Actividad: Reportes Bitacoras / Total por Año y Categoría

DIAGRAMA DE ACTIVIDAD: REPORTES BITACORAS / TOTAL POR MES, AÑO Y CATEGORIA

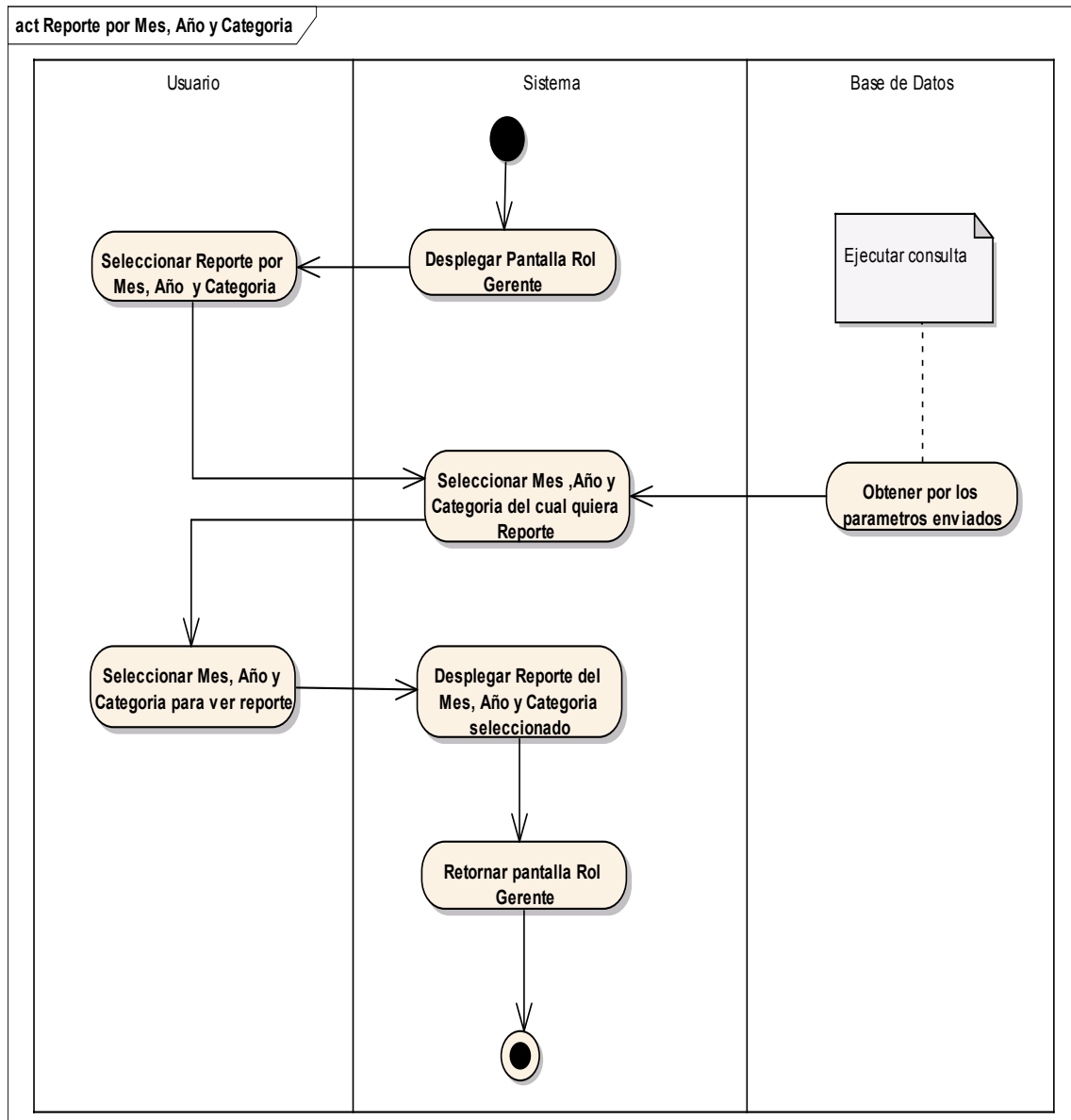


Figura 152 Diagrama de Actividad: Reportes Bitacoras / Total por Mes, Año y Categoría

2.1.6.5 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.

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.

DIAGRAMA DE SECUENCIA: ACCESO AL SISTEMA

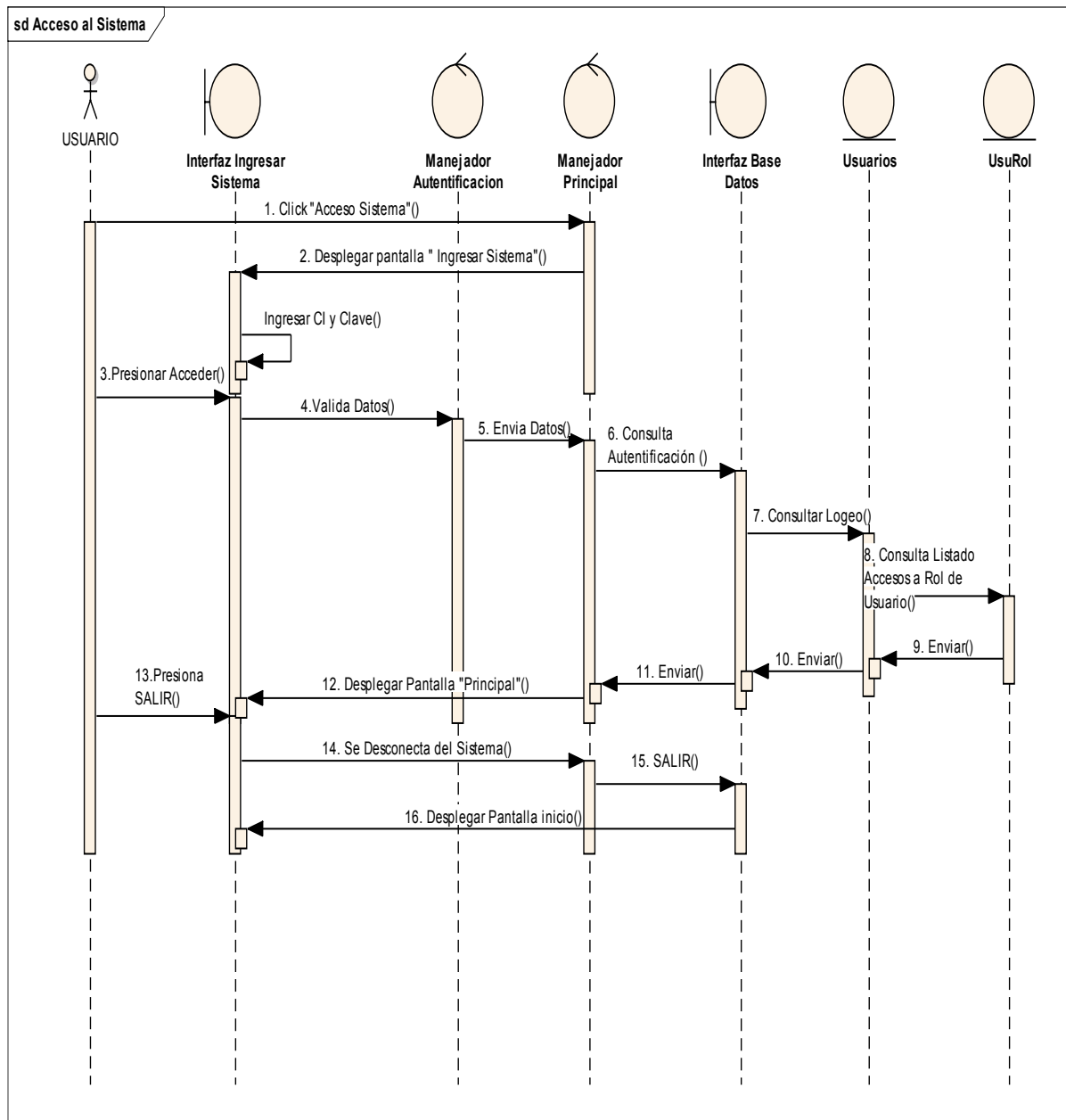


Figura 153 Diagrama de Secuencia: Ingresar al Sistema

DIAGRAMA DE SECUENCIA: ADMINISTRAR USUARIOS

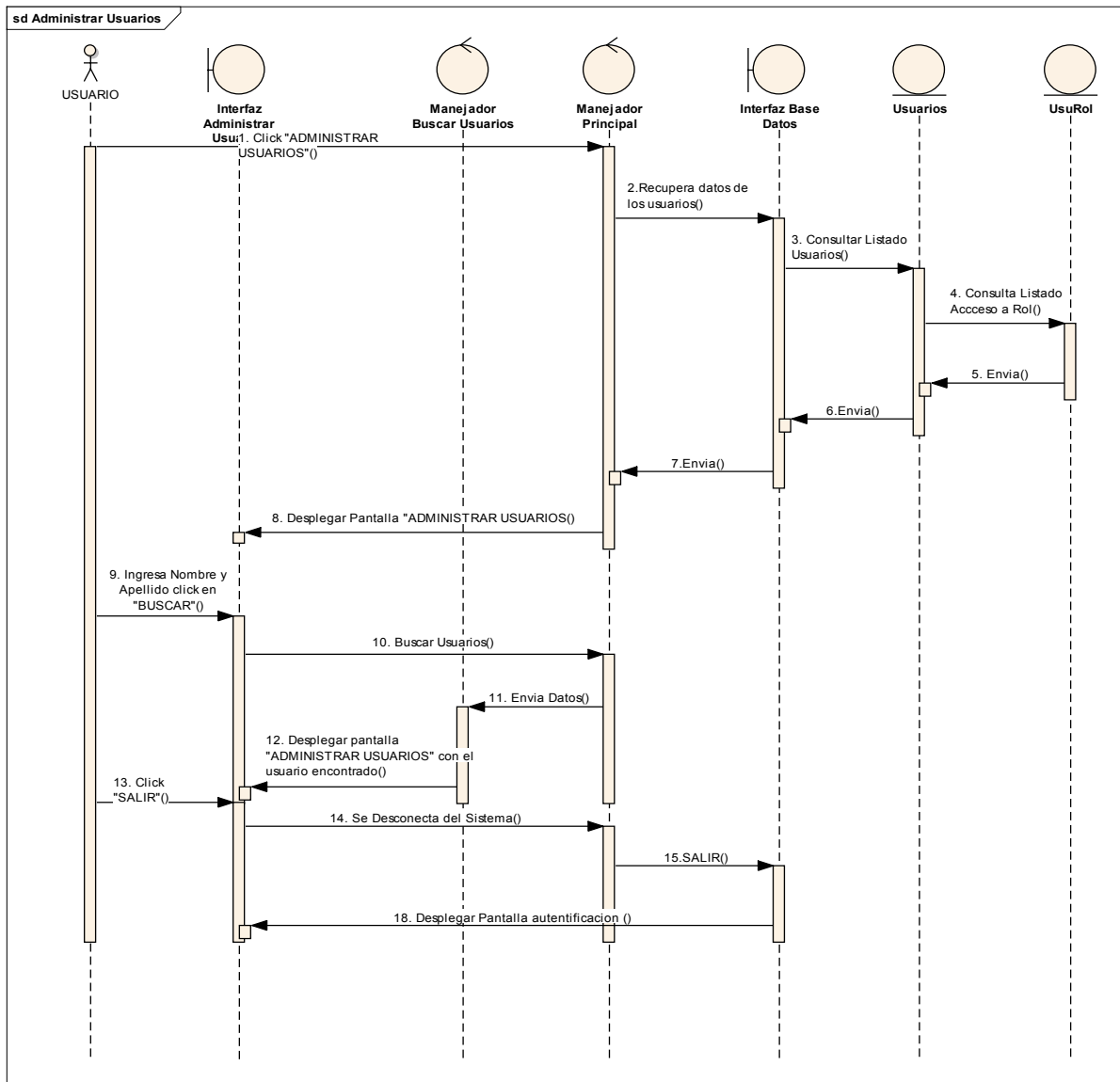


Figura 154 Diagrama de Secuencia: Administrar Usuarios

DIAGRAMA DE SECUENCIA: ADICIONAR USUARIOS

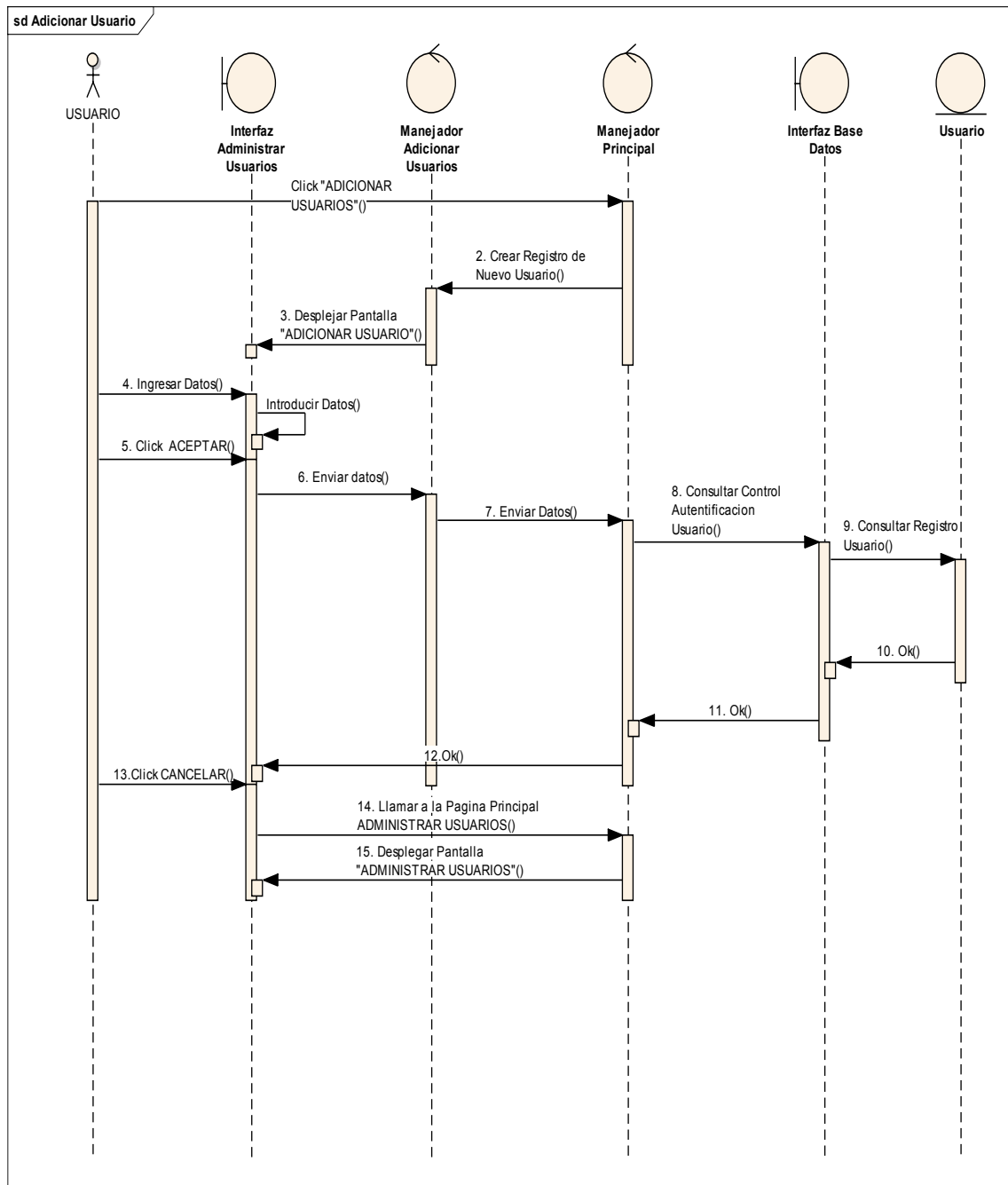


Figura 155 Diagrama de Secuencia: Adicionar Usuarios

DIAGRAMA DE SECUENCIA: MODIFICAR USUARIOS

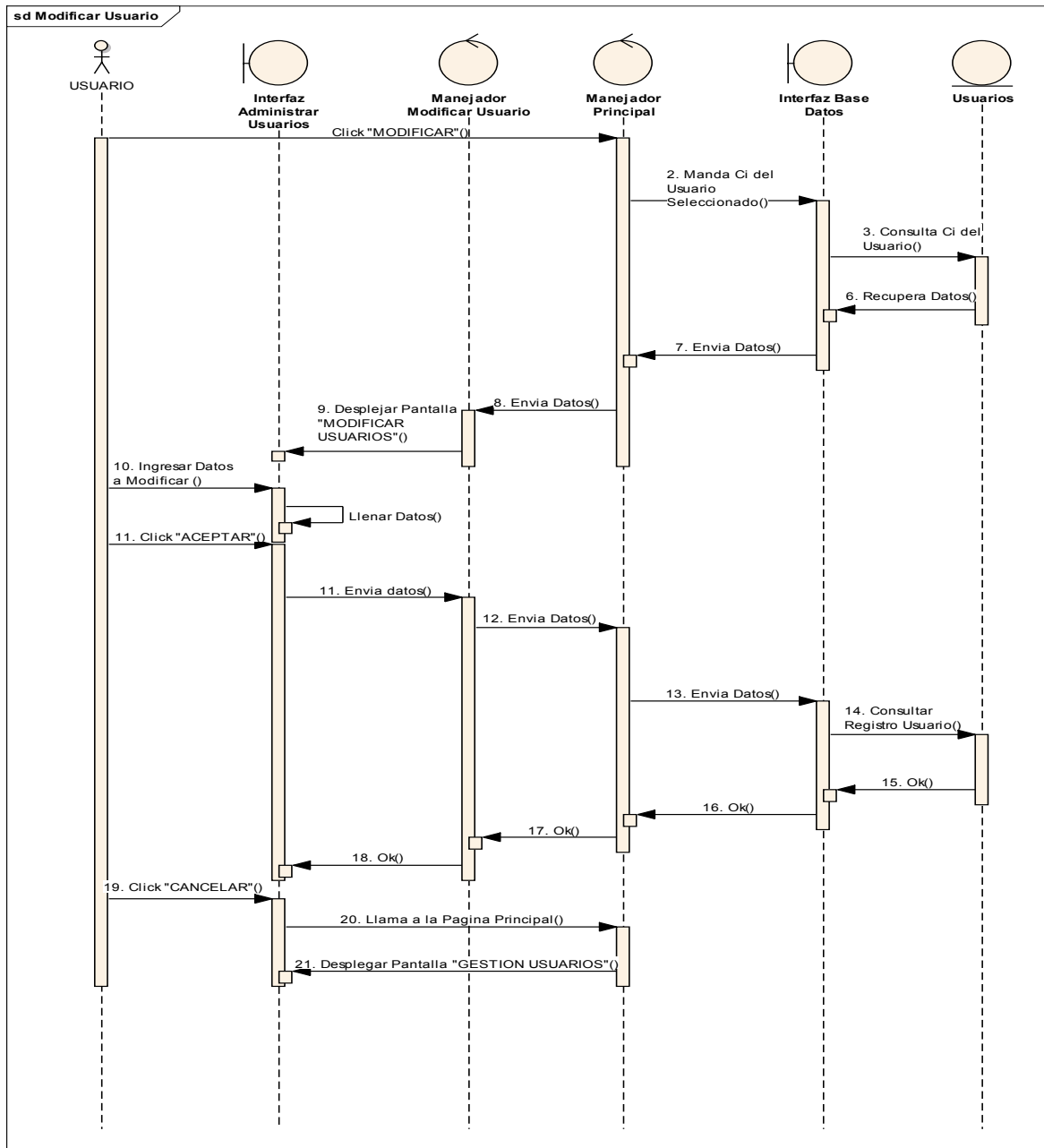


Figura 156 Diagrama de Secuencia: Modificar Usuarios

DIAGRAMA DE SECUENCIA: ELIMINAR USUARIOS

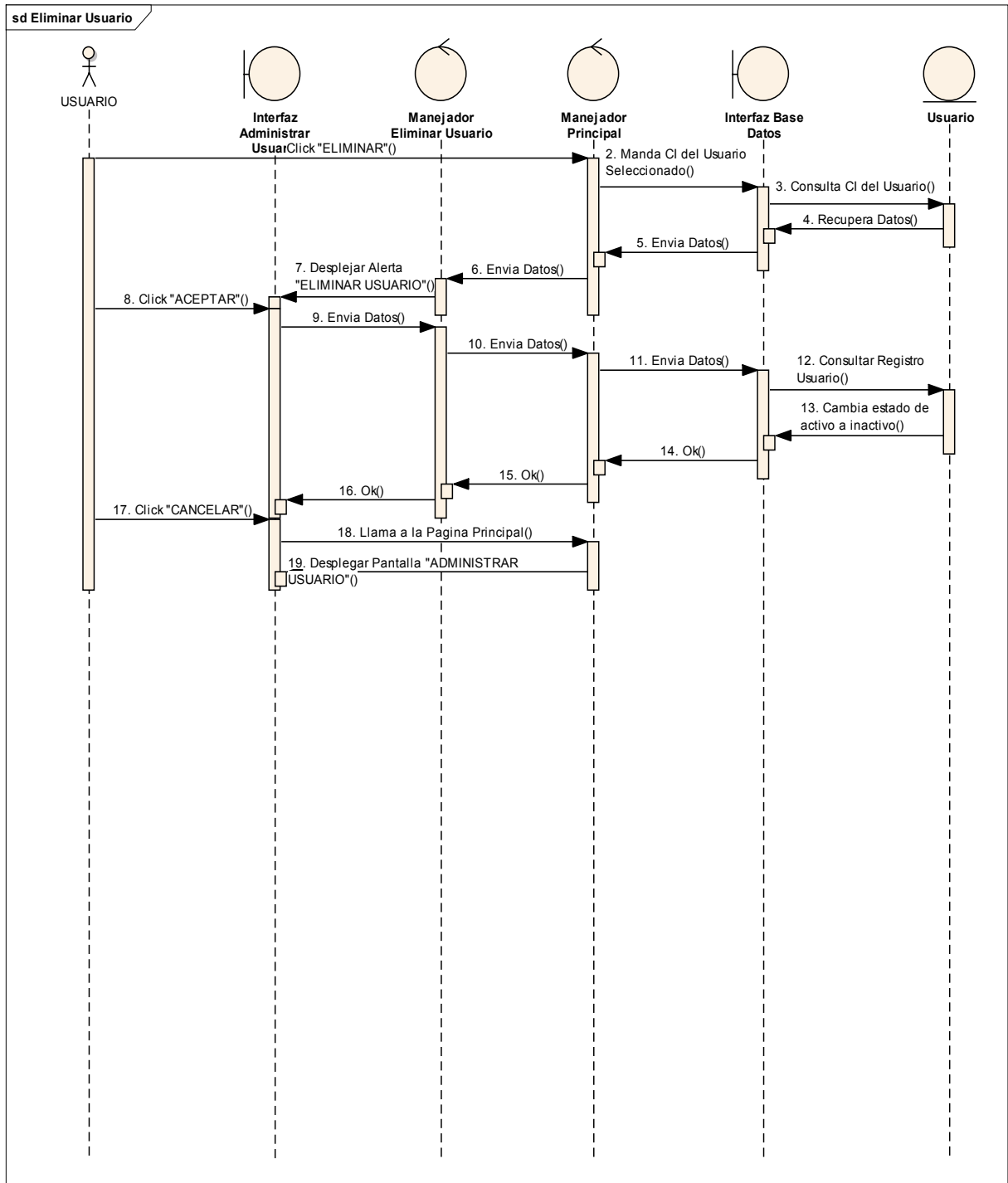


Figura 157 Diagrama de Secuencia: Eliminar Usuarios

DIAGRAMA DE SECUENCIA: ADMINISTRAR ROLES

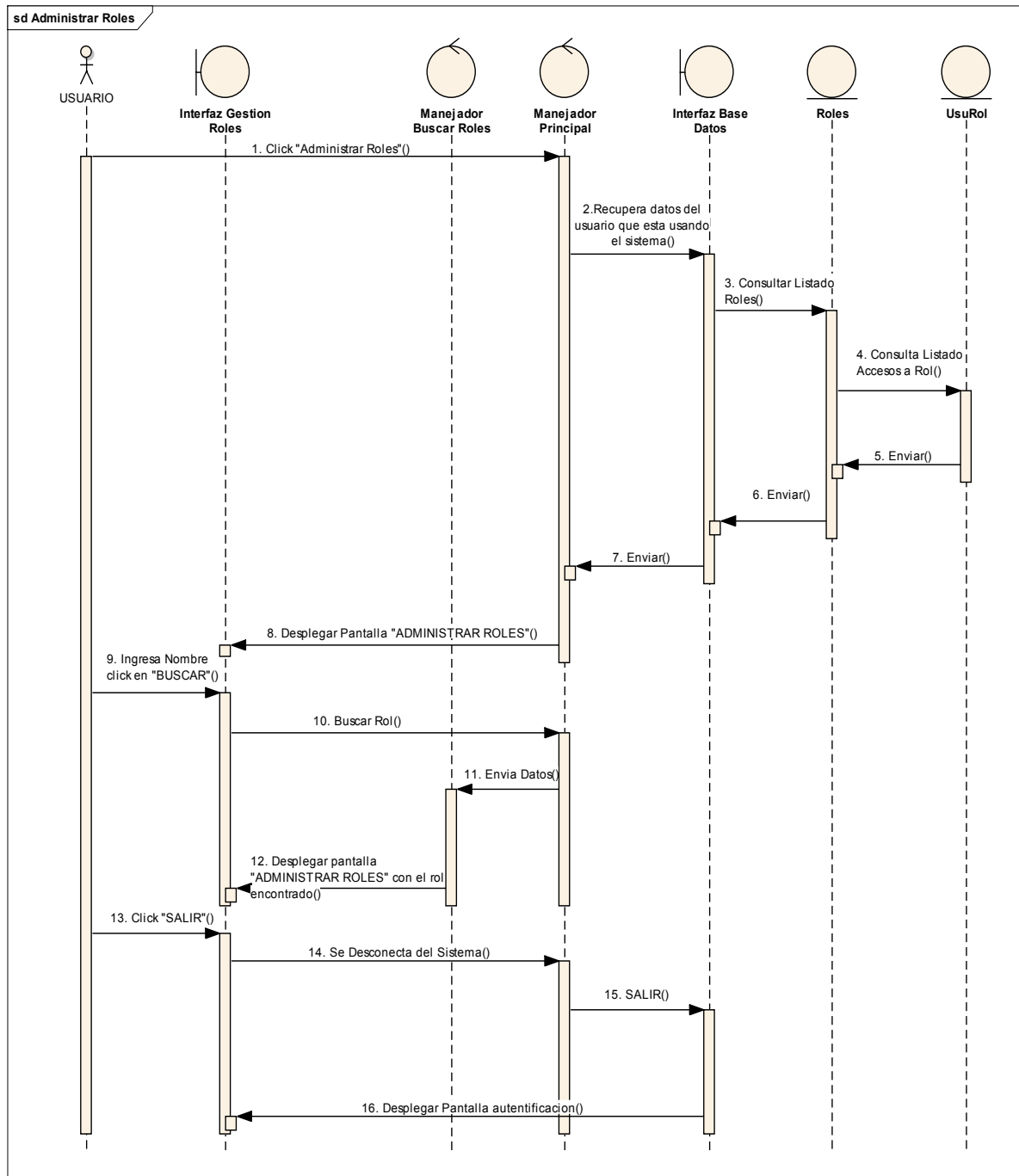


Figura 158 Diagrama de Secuencia: Administrar Roles

DIAGRAMA DE SECUENCIA: ADICIONAR ROLES

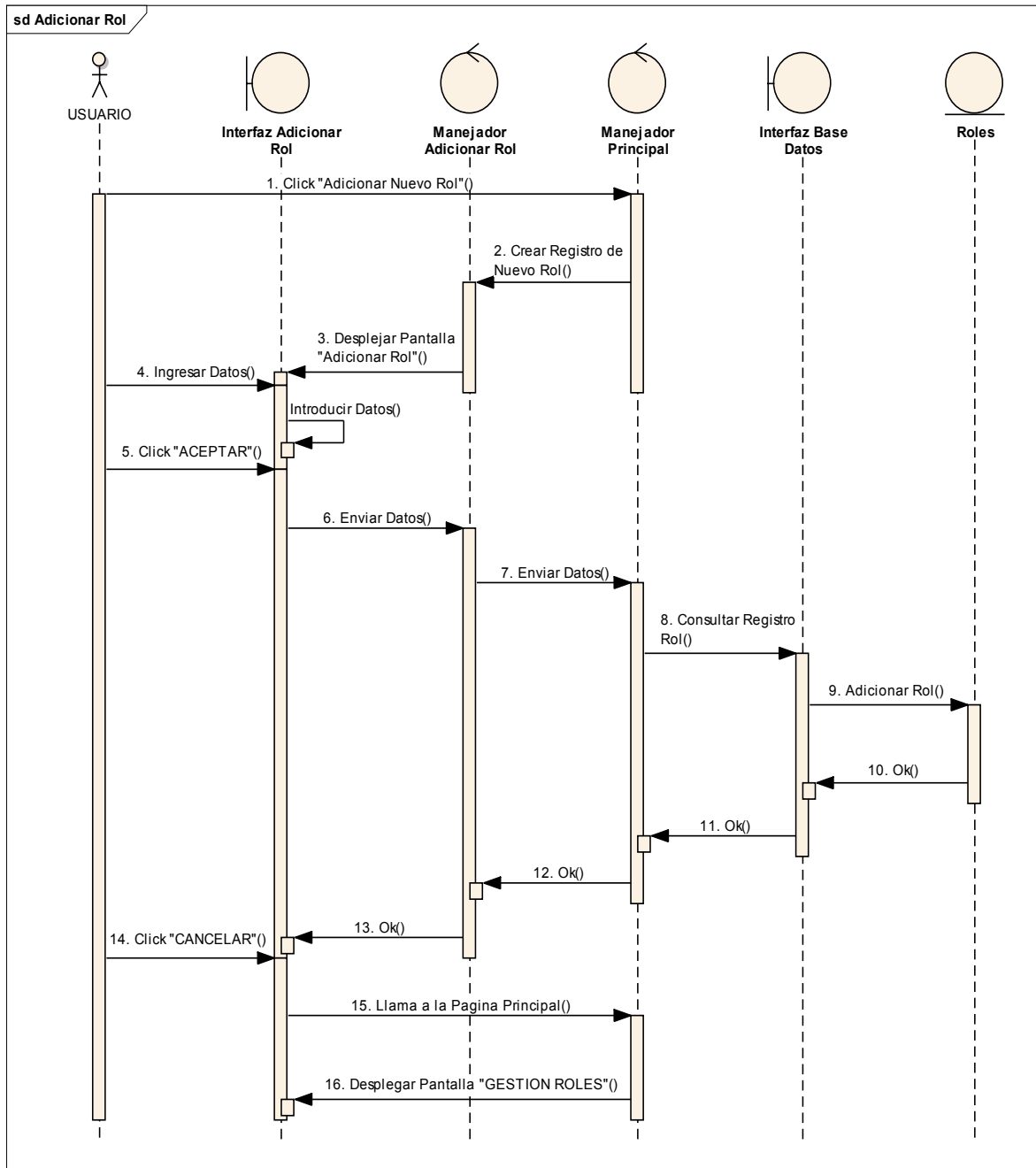


Figura 159 Diagrama de Secuencia: Adicionar Roles

DIAGRAMA DE SECUENCIA: MODIFICAR ROLES

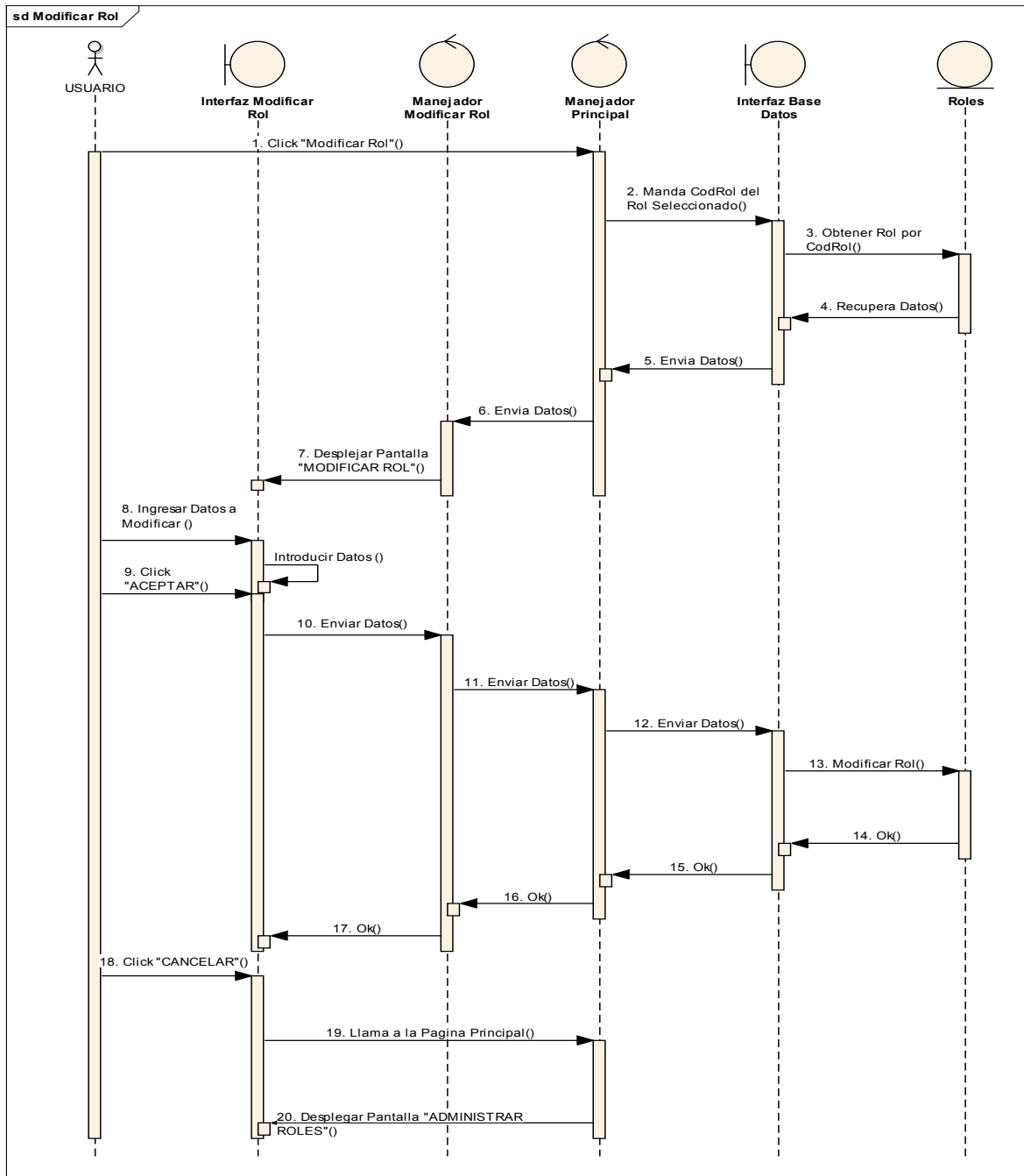


Figura 160 Diagrama de Secuencia: Modificar Roles

DIAGRAMA DE SECUENCIA: ELIMINAR ROLES

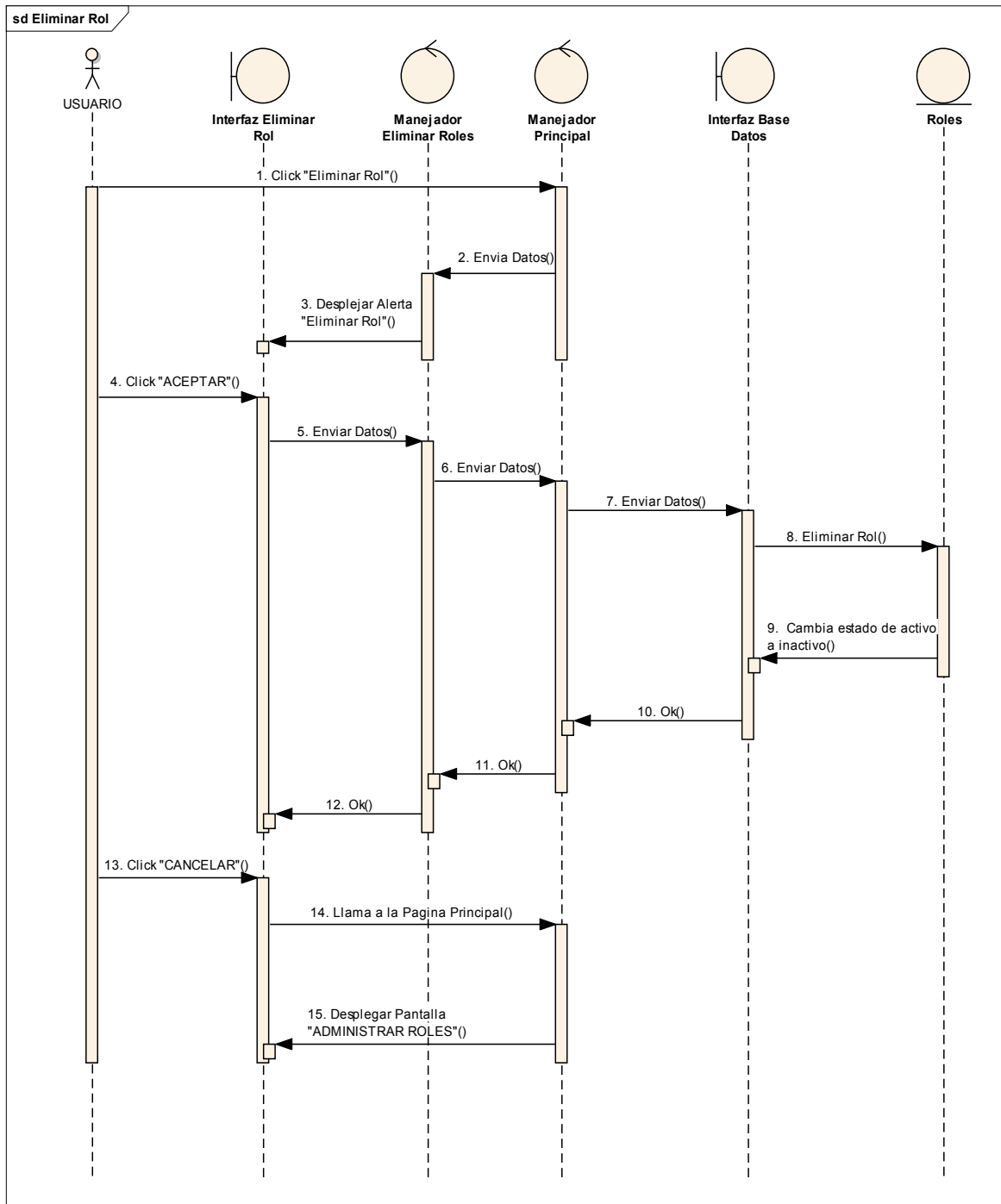


Figura 161 Diagrama de Secuencia: Eliminar Roles

DIAGRAMA DE SECUENCIA: ASIGNAR ROLES

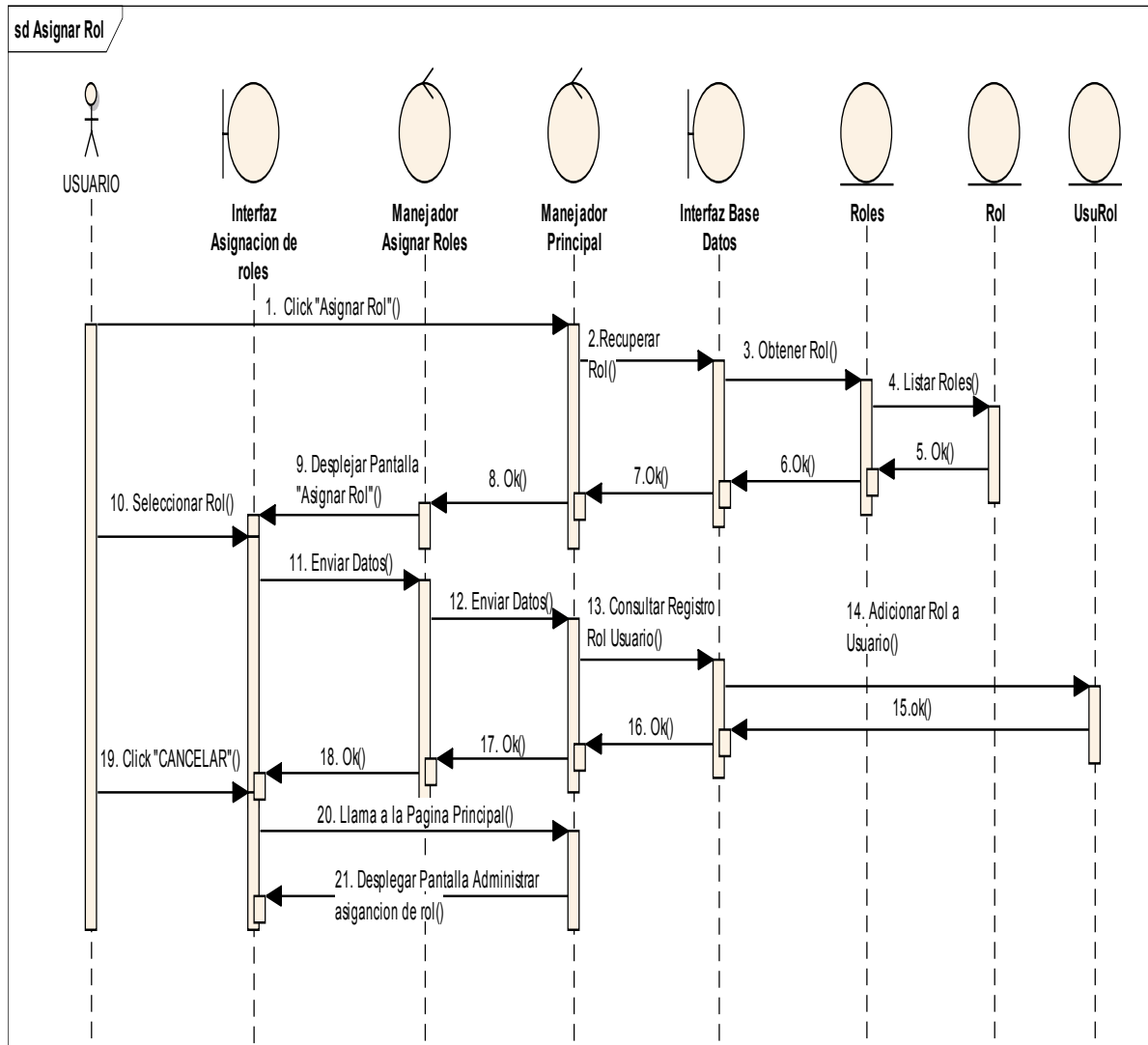


Figura 162 Diagrama de Secuencia: Asignar Roles

DIAGRAMA DE SECUENCIA: ADMINISTRAR CLAVES

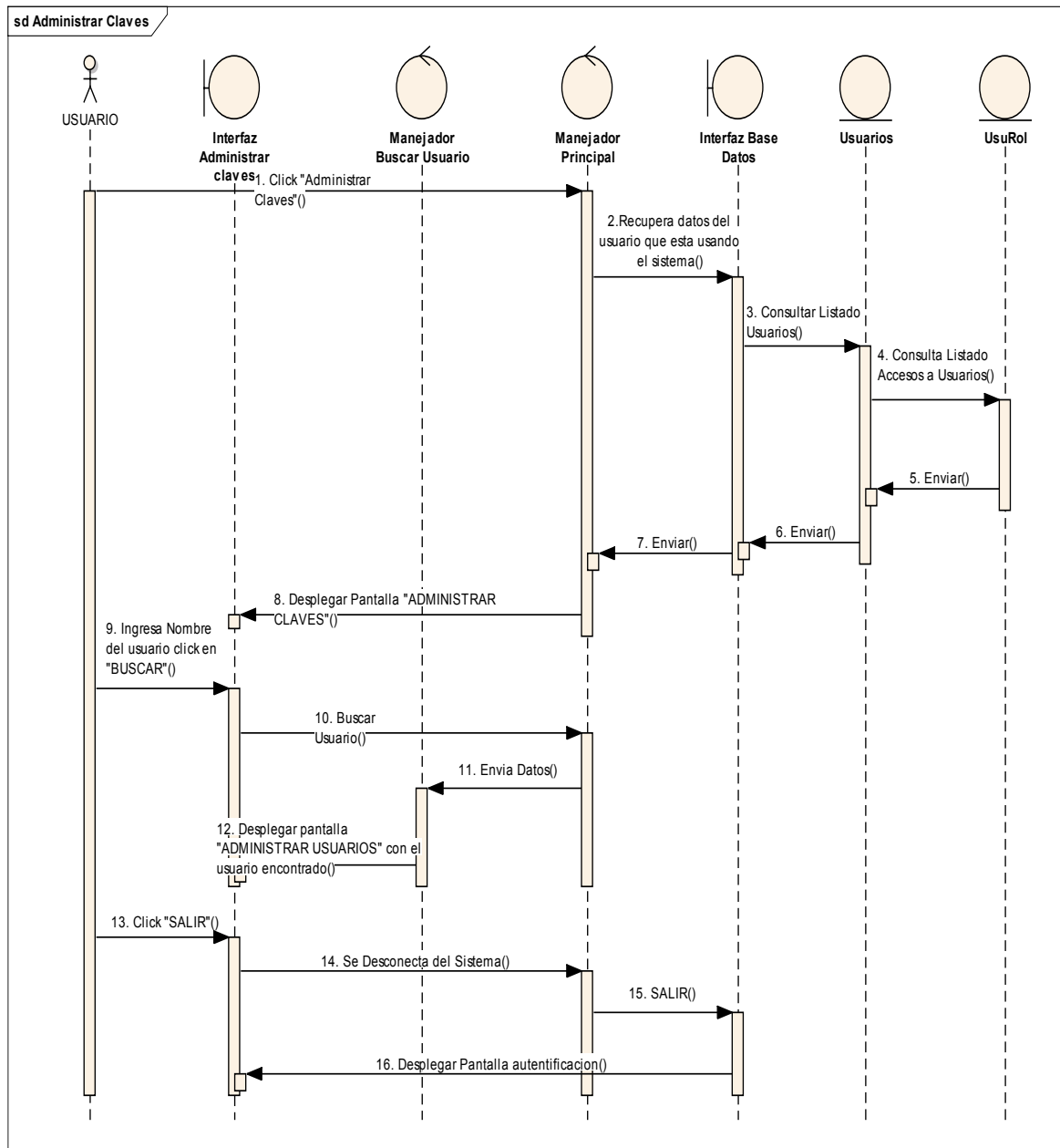


Figura 163 Diagrama de Secuencia: Administrar Claves

DIAGRAMA DE SECUENCIA: ADICIONAR CLAVES

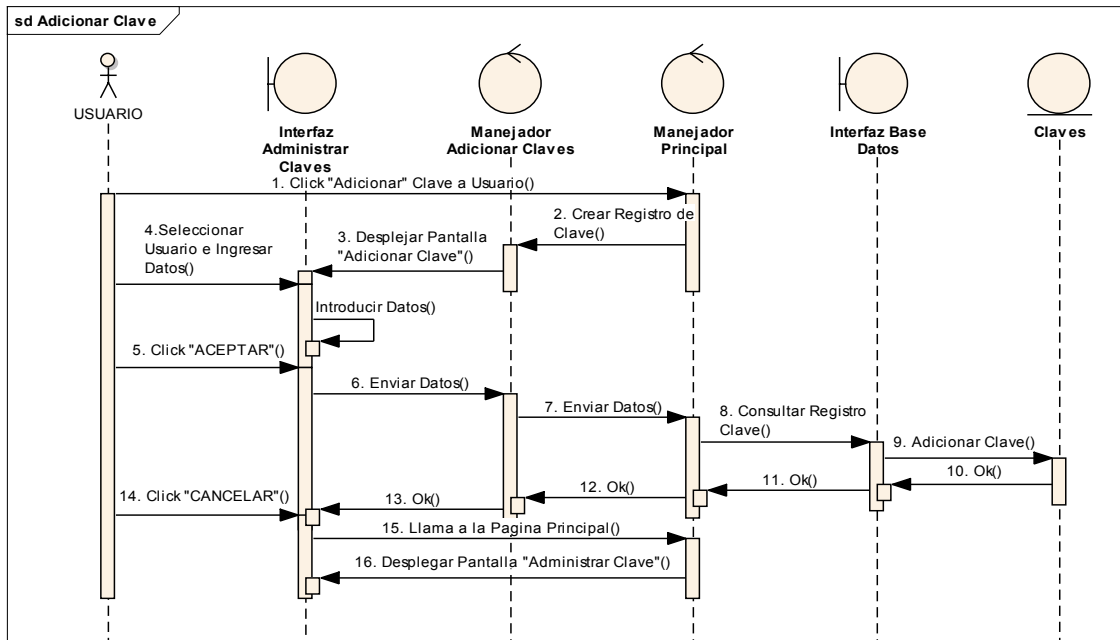


Figura 164 Diagrama de Secuencia: Adicionar Claves

DIAGRAMA DE SECUENCIA: MODIFICAR CLAVES

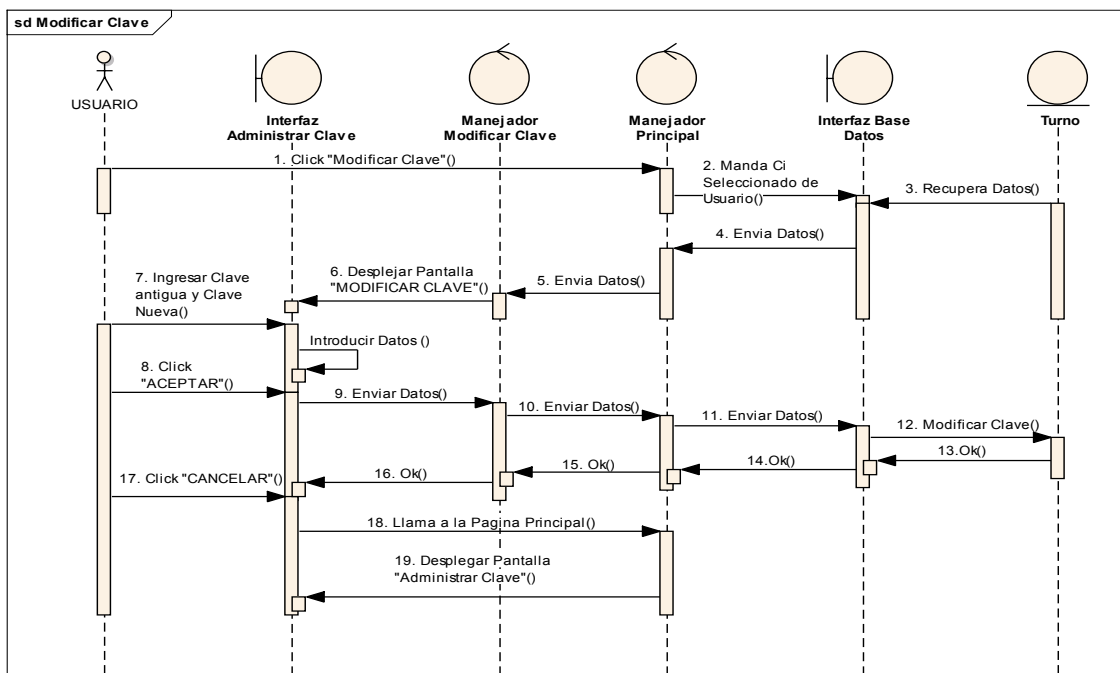


Figura 165 Diagrama de Secuencia: Modificar Claves

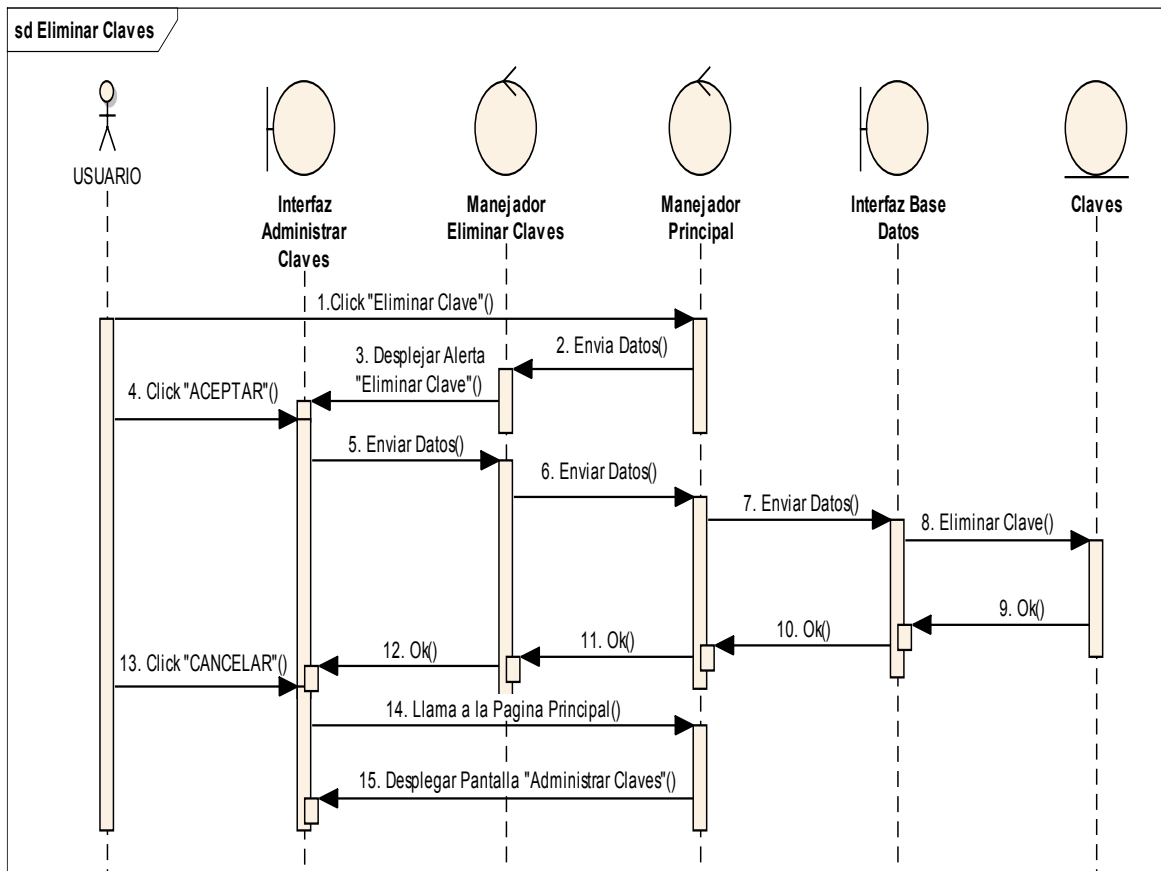
DIAGRAMA DE SECUENCIA: ELIMINAR CLAVES

Figura 166 Diagrama de Secuencia: Eliminar Claves

DIAGRAMA DE SECUENCIA: ADMINISTRAR CHOFERES

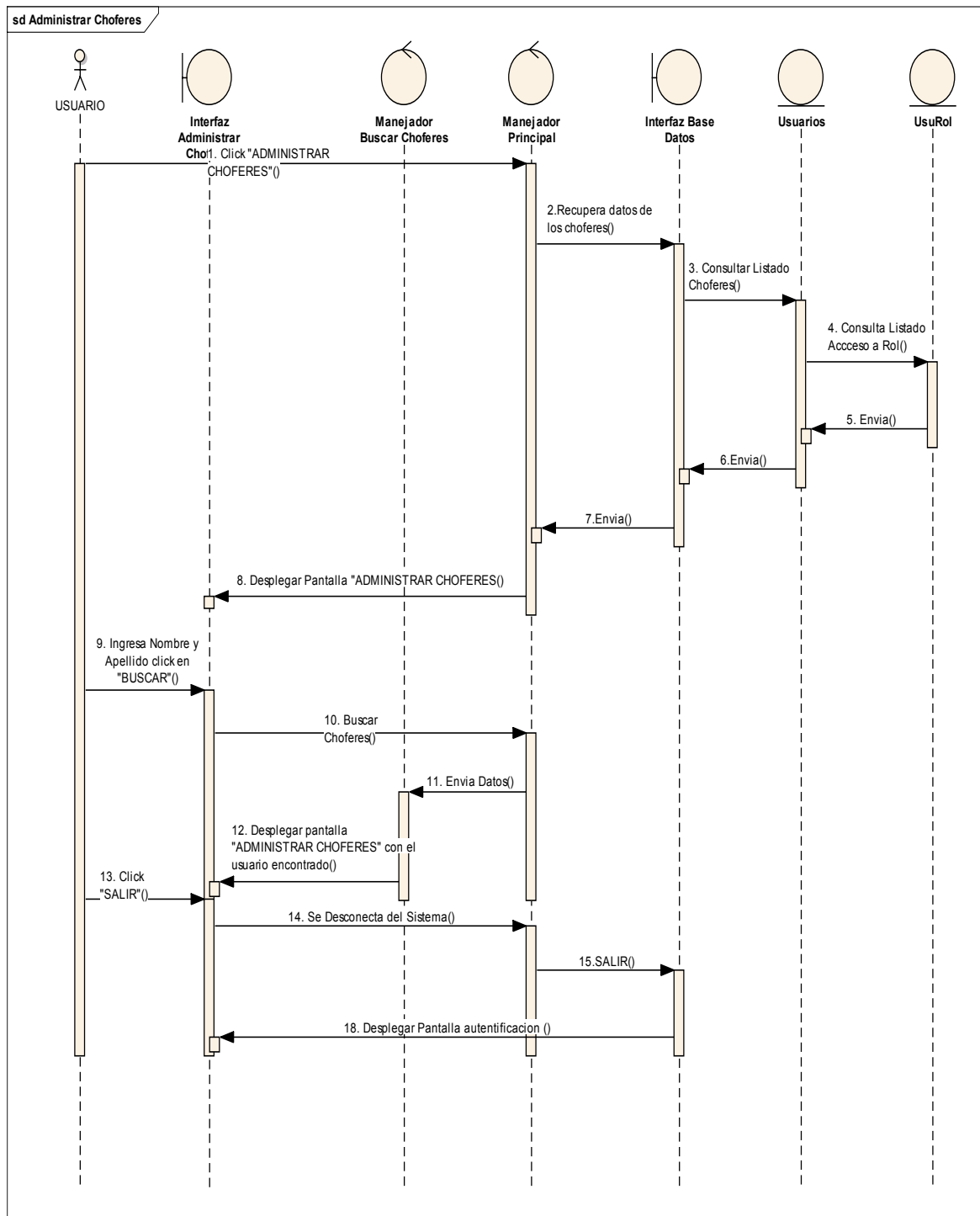


Figura 167 Diagrama de Secuencia: Administrar Choferes

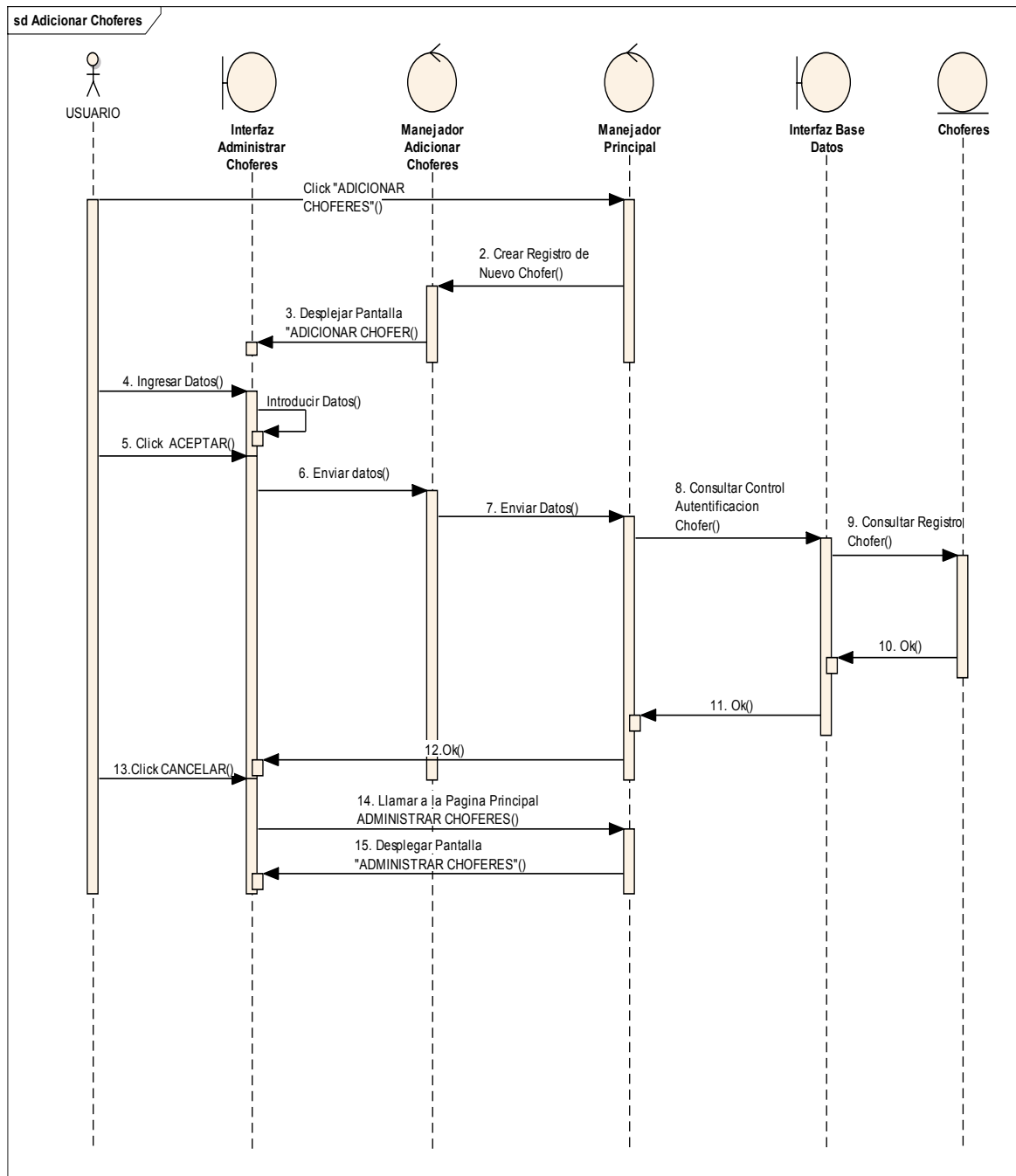
DIAGRAMA DE SECUENCIA: ADICIONAR CHOFERES

Figura 168 Diagrama de Secuencia: Adicionar Choferes

DIAGRAMA DE SECUENCIA: MODIFICAR CHOFERES

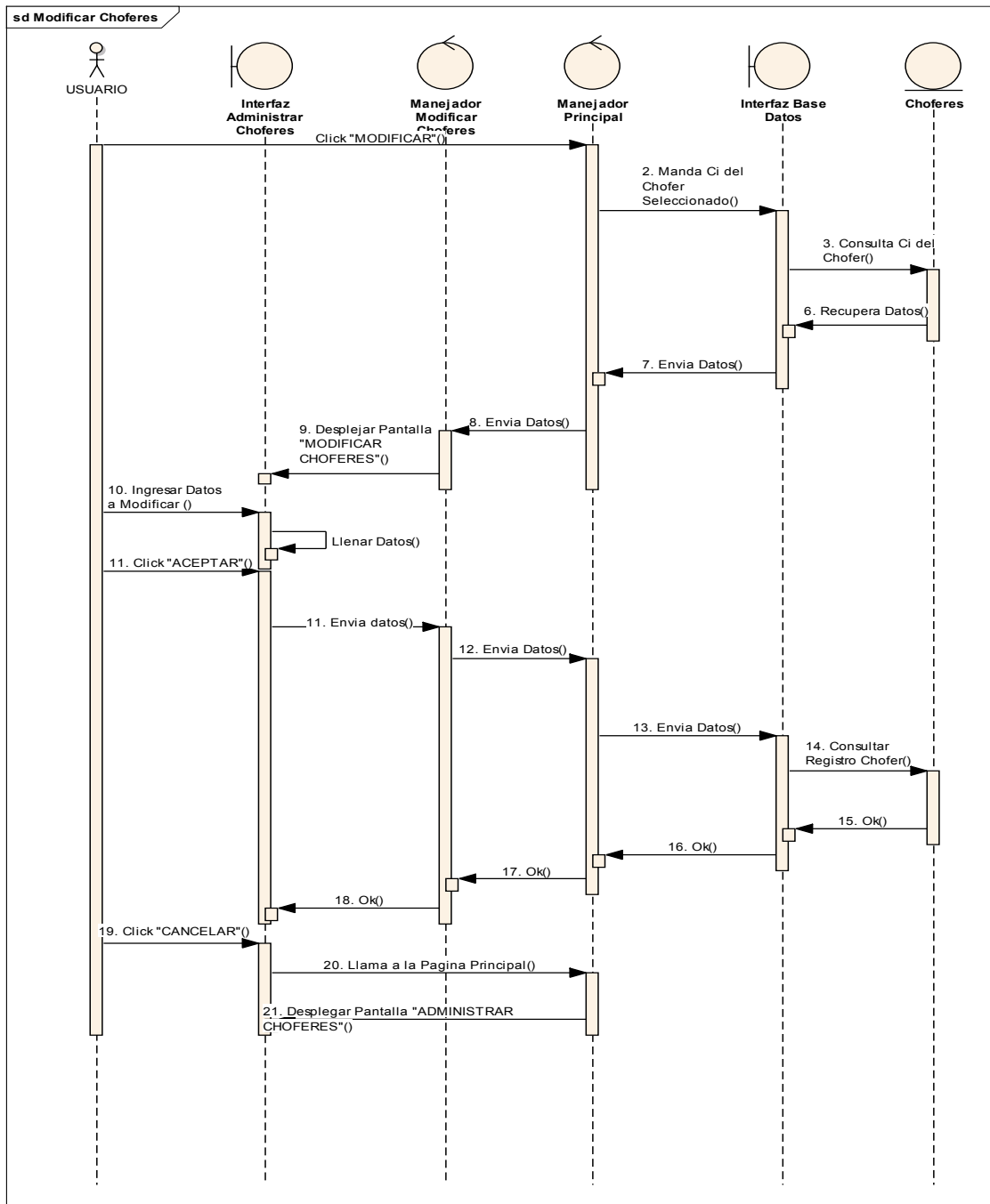


Figura 169 Diagrama de Secuencia: Modificar Choferes

DIAGRAMA DE SECUENCIA: ELIMINAR CHOFERES

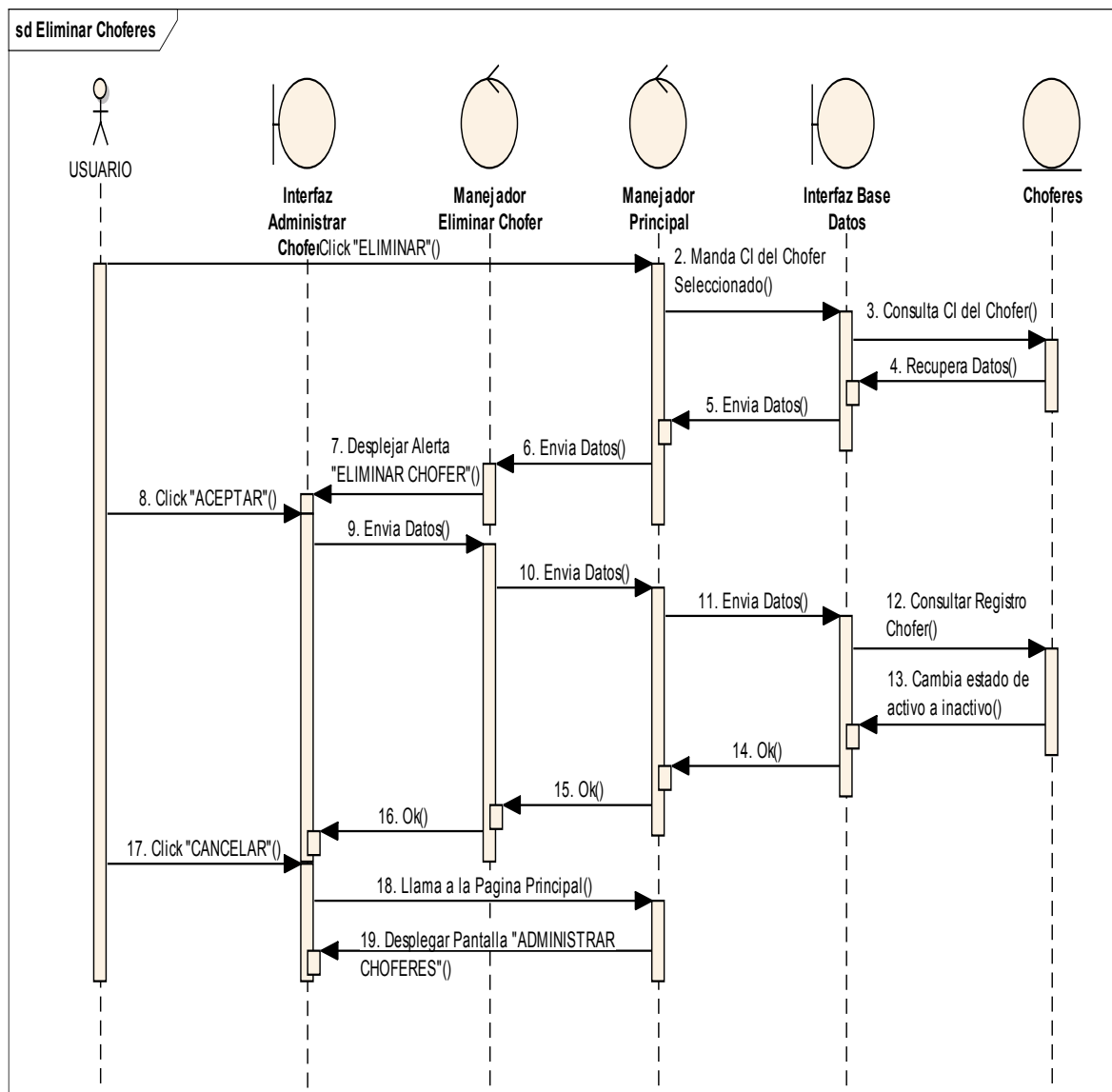


Figura 170 Diagrama de Secuencia: Eliminar Choferes

DIAGRAMA DE SECUENCIA: ADMINISTRAR AYUDANTES

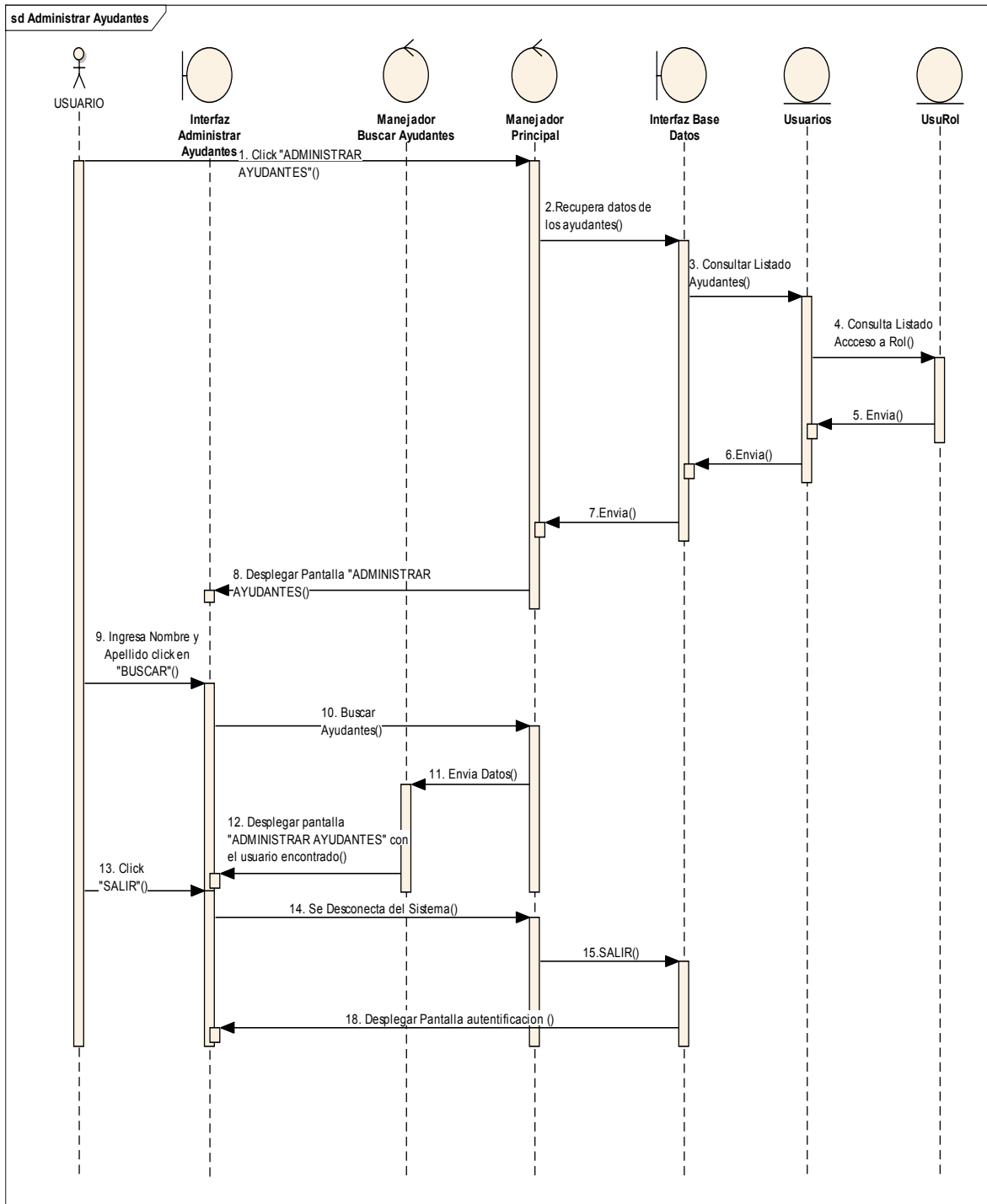


Figura 171

Diagrama de Secuencia: Administrar Ayudantes

DIAGRAMA DE SECUENCIA: ADICIONAR AYUDANTES

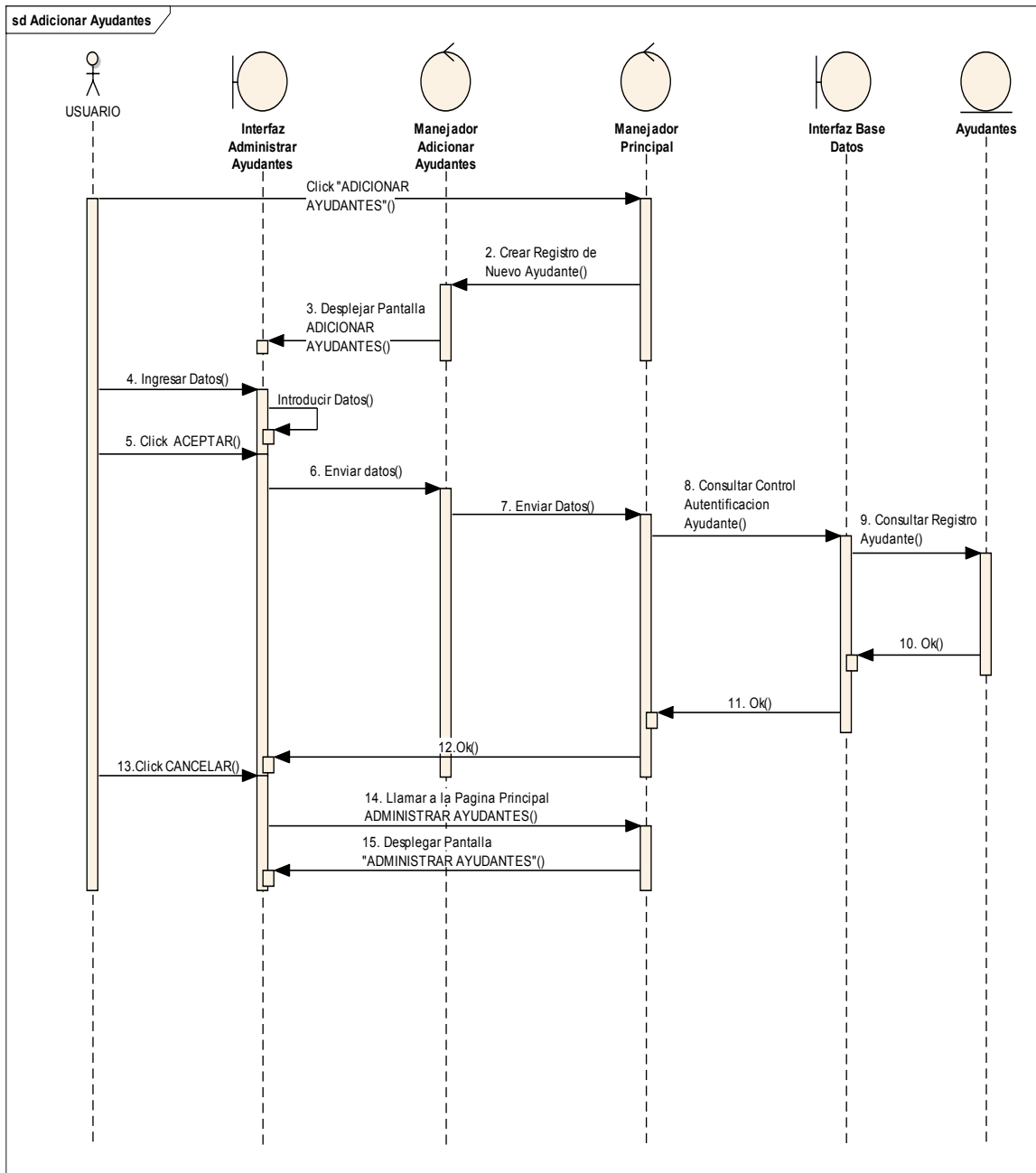


Figura 172 Diagrama de Secuencia: Adicionar Ayudantes

DIAGRAMA DE SECUENCIA: MODIFICAR AYUDANTES

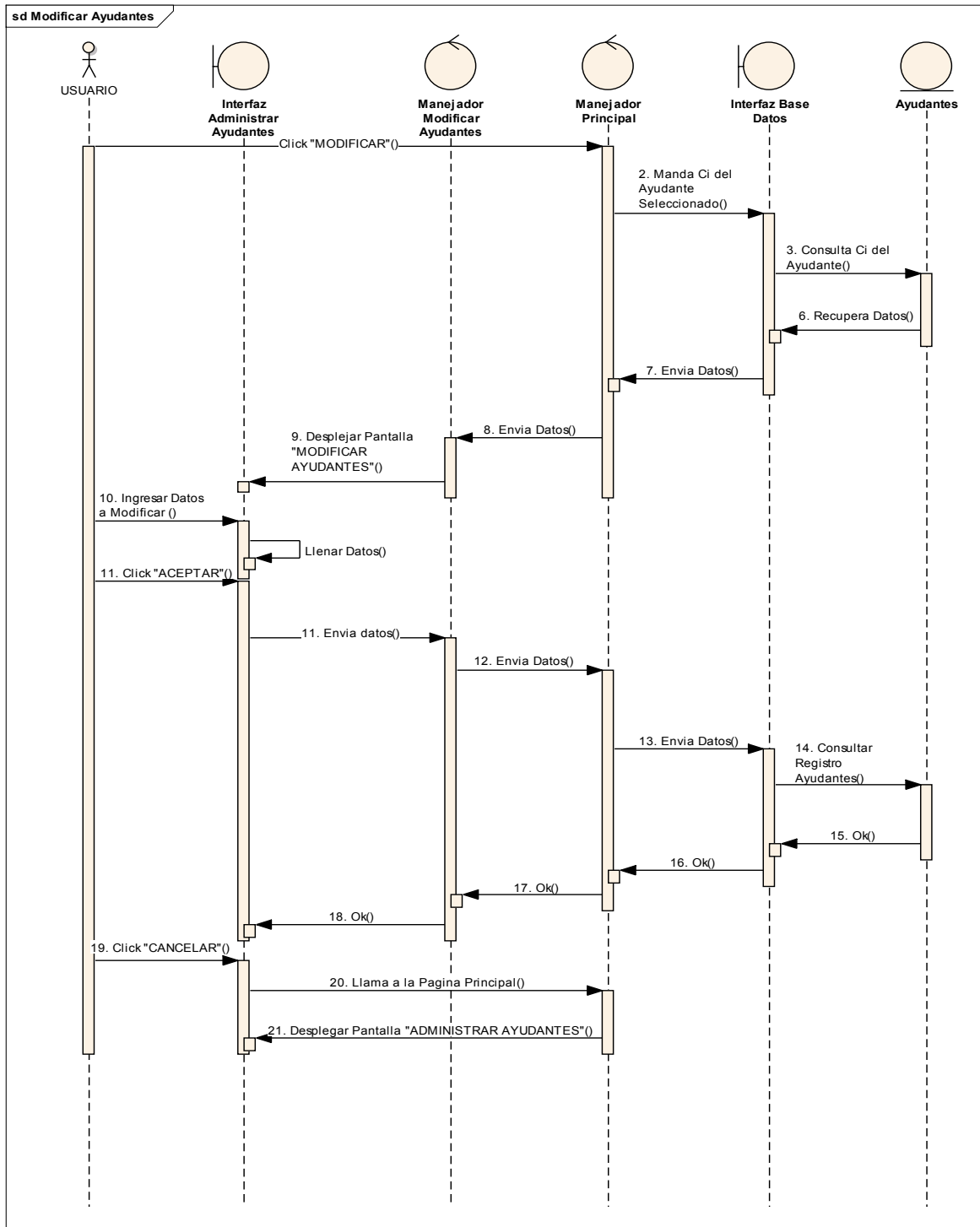


Figura 173

Diagrama de Secuencia: Modificar Ayudantes

DIAGRAMA DE SECUENCIA: ELIMINAR AYUDANTES

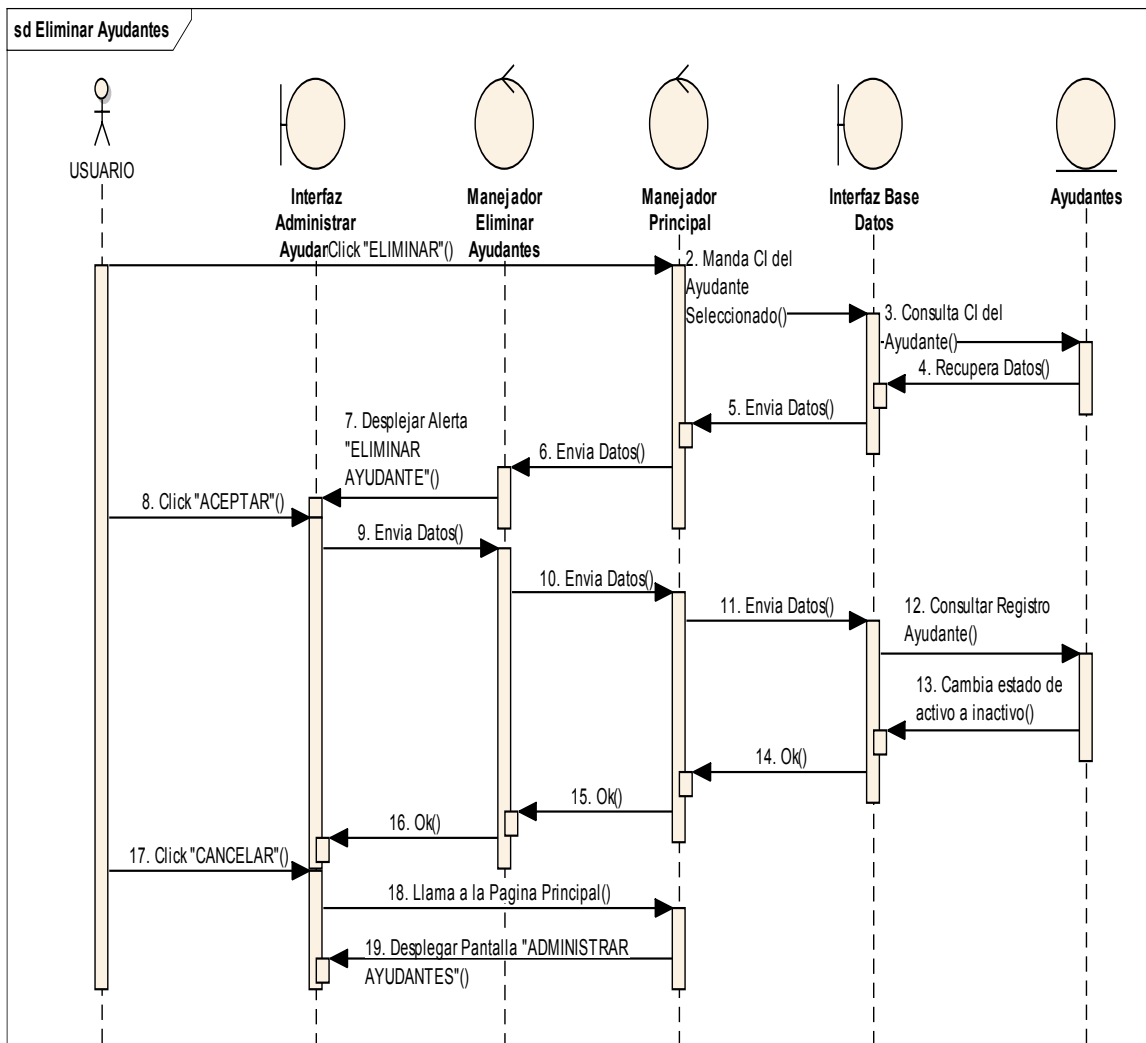


Figura 174

Diagrama de Secuencia: Eliminar Ayudantes

DIAGRAMA DE SECUENCIA: ADMINISTRAR ZONAS

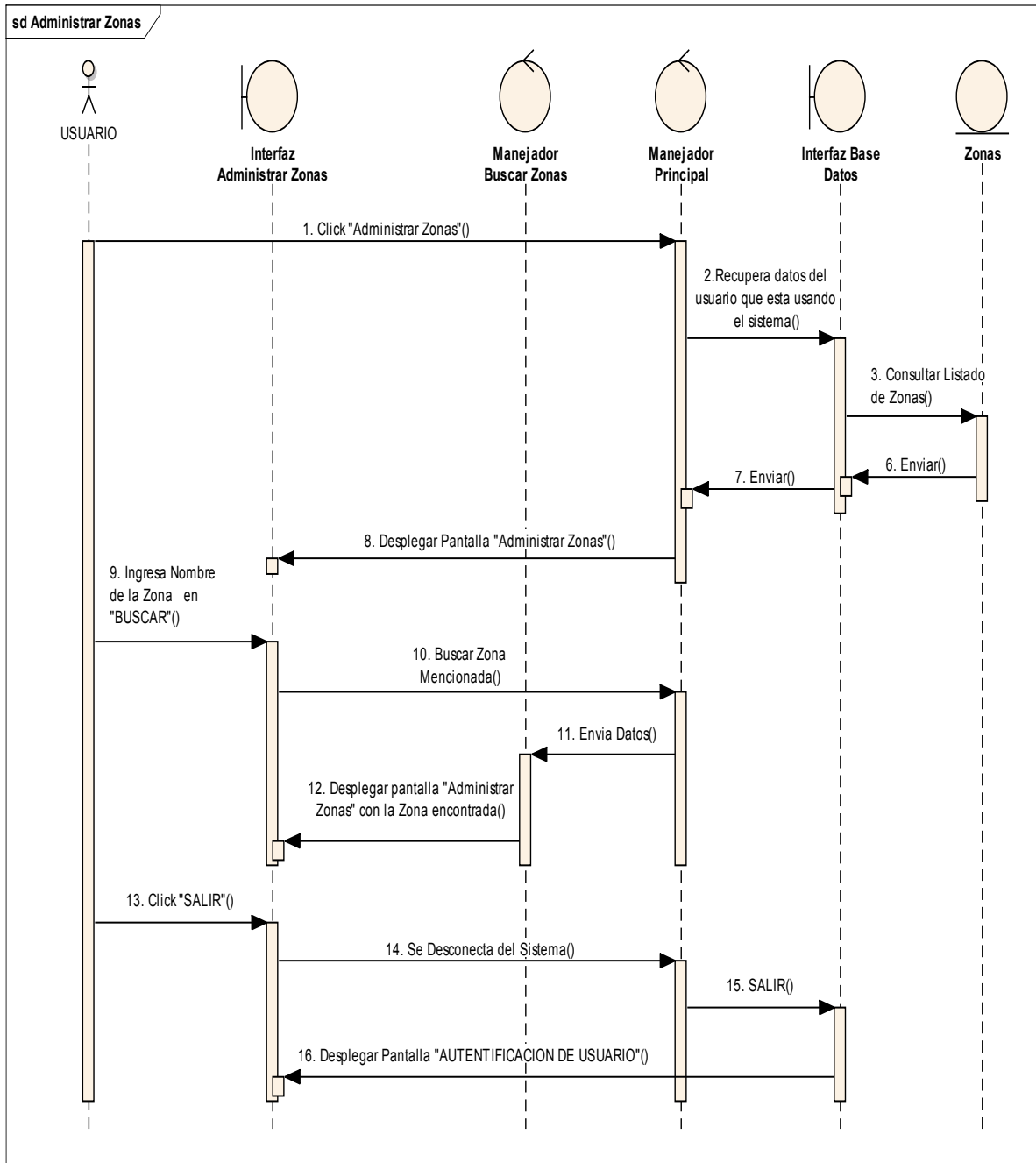


Figura 175 Diagrama De Secuencia: Administrar Zonas

DIAGRAMA DE SECUENCIA: ADICIONAR ZONAS

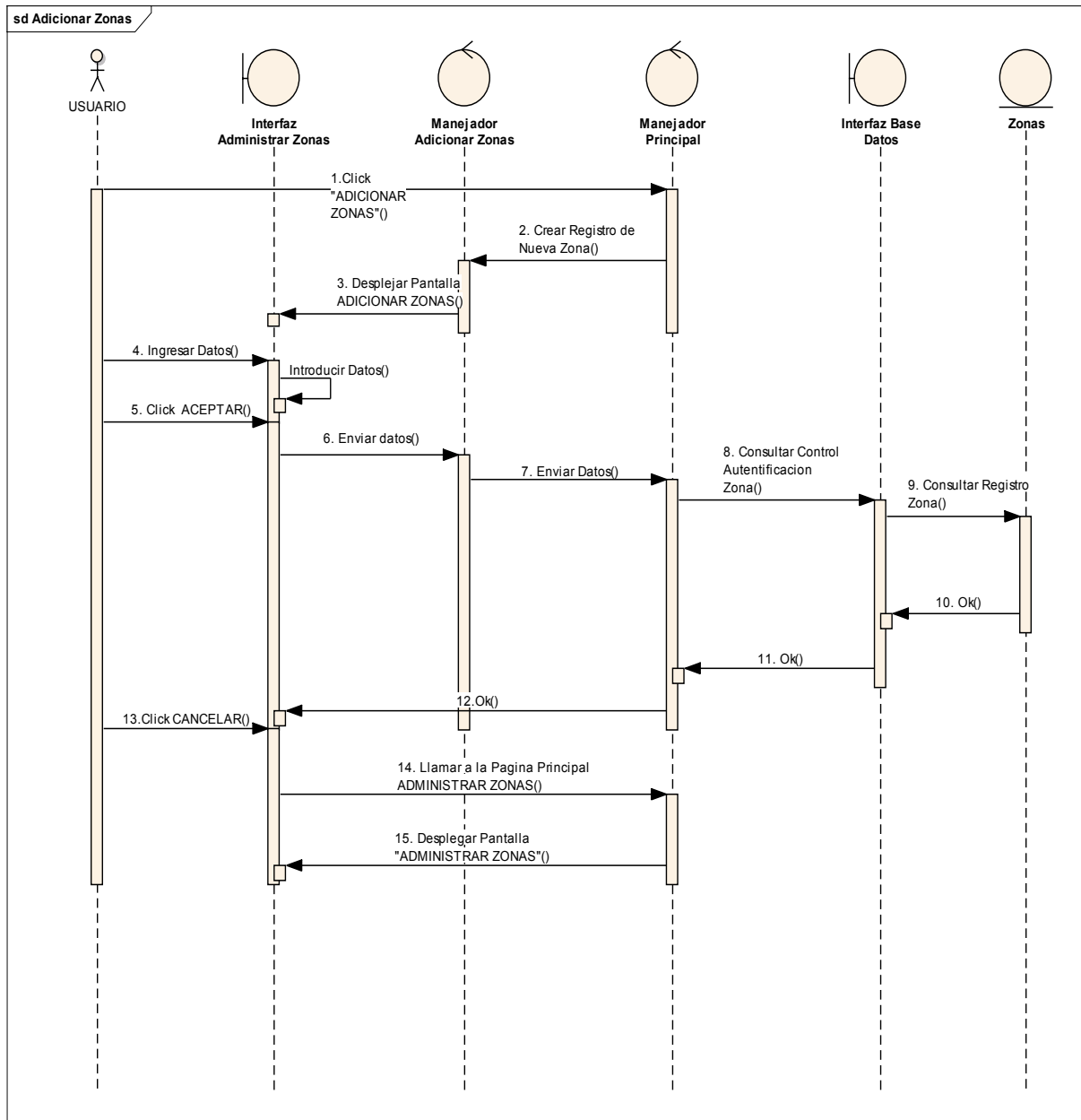


Figura 176 Diagrama de Secuencia: Adicionar Zonas

DIAGRAMA DE SECUENCIA: MODIFICAR ZONAS

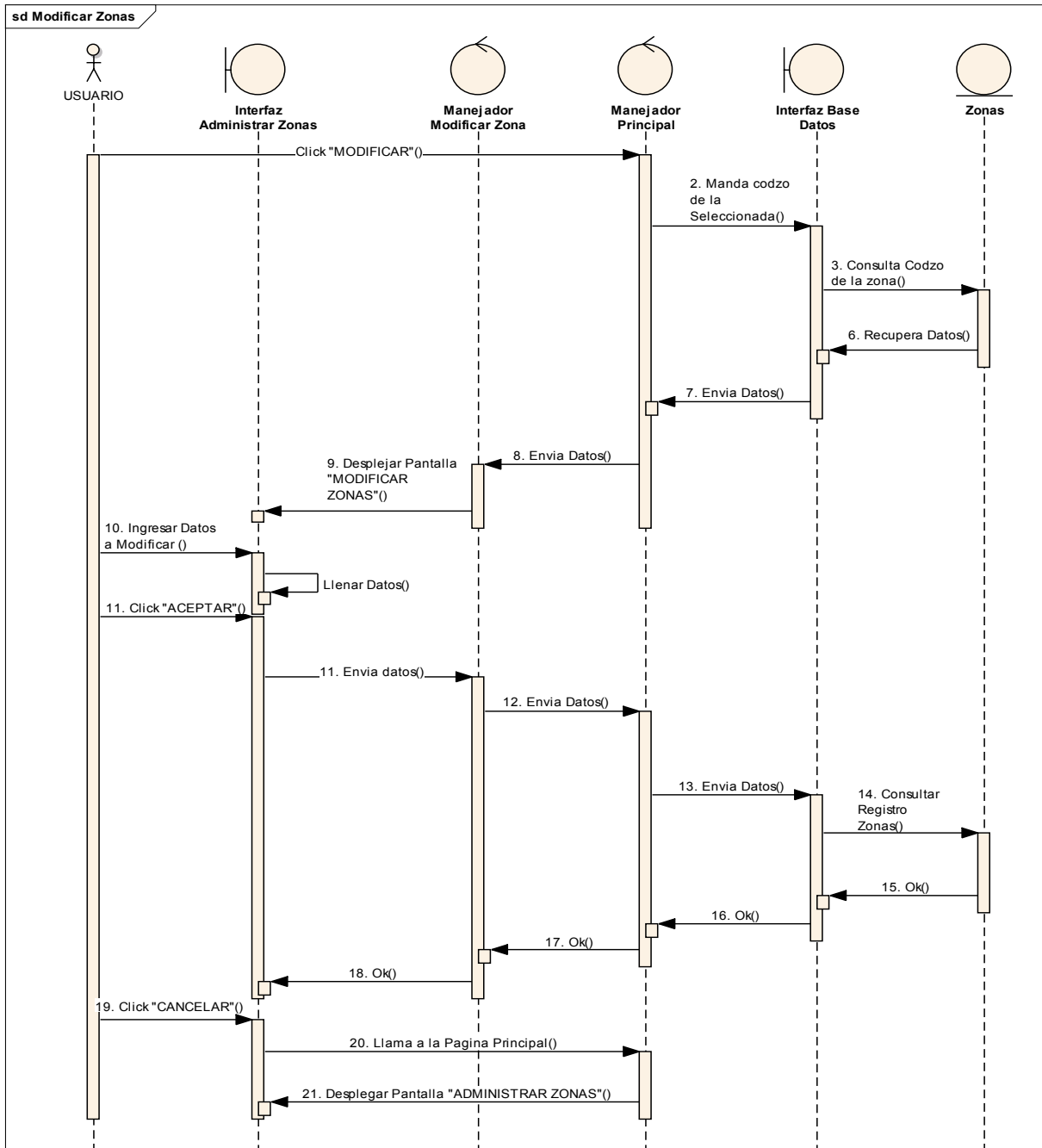


Figura 177 Diagrama de Secuencia: Modificar Zonas

DIAGRAMA DE SECUENCIA: ELIMINAR ZONAS

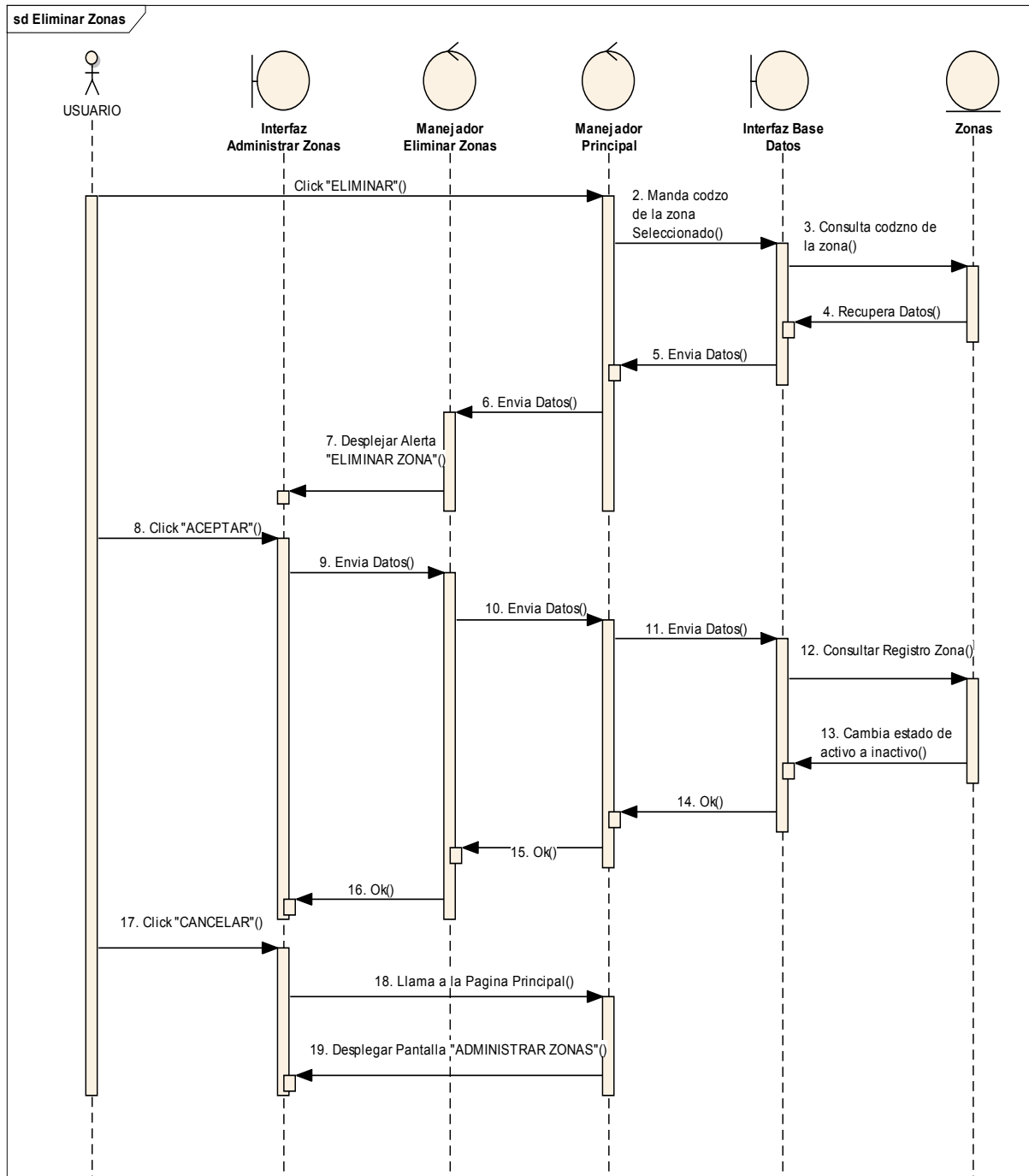


Figura 178

Diagrama de Secuencia: Eliminar Zonas

DIAGRAMA DE SECUENCIA: ADMINISTRAR VEHICULOS

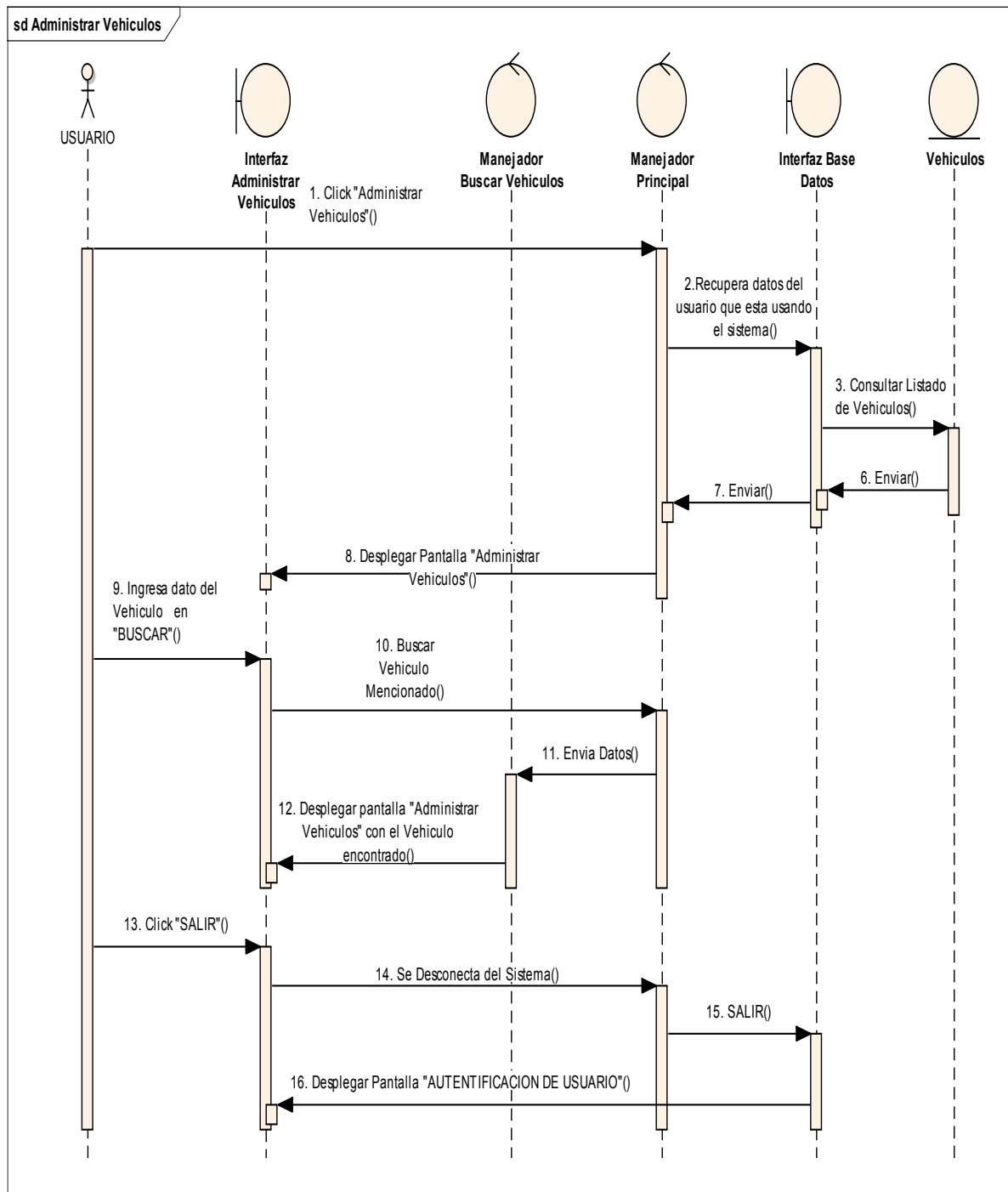


Figura 179 Diagrama de Secuencia: Administrar Vehiculos

DIAGRAMA DE SECUENCIA: ADICIONAR VEHICULOS

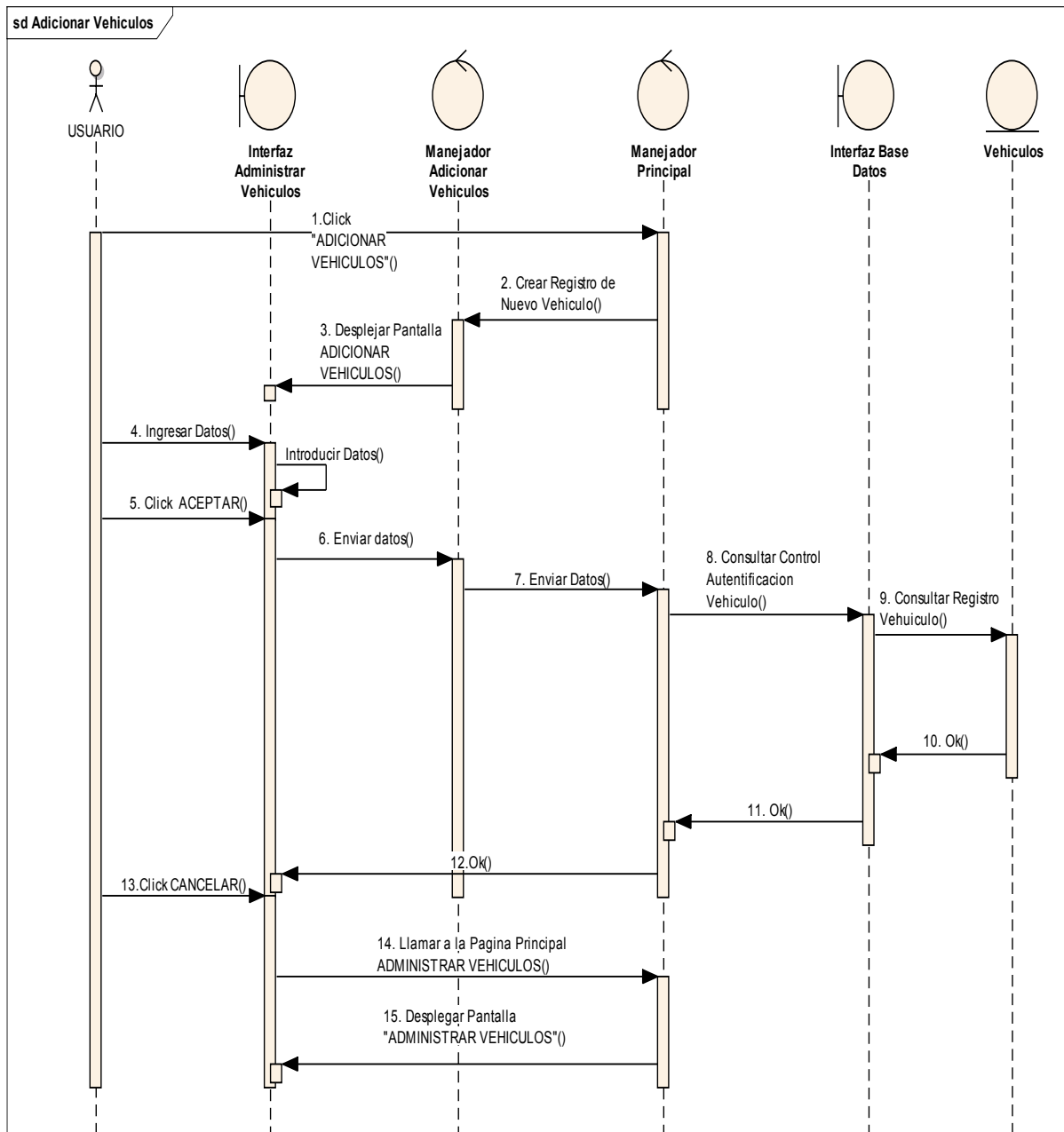


Figura 180

Diagrama De Secuencia: Adicionar Vehículos

DIAGRAMA DE SECUENCIA: MODIFICAR VEHICULOS

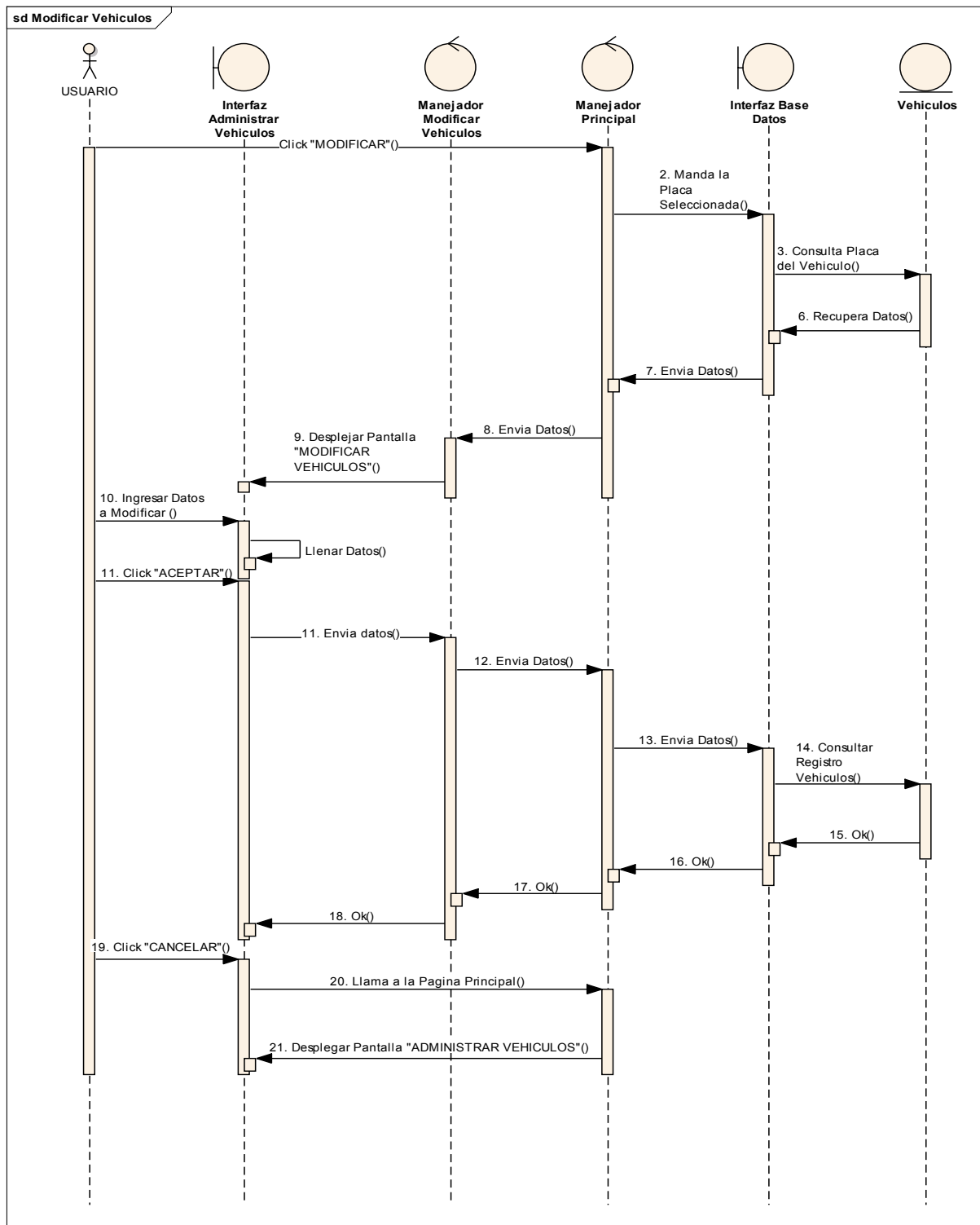


Figura 181 Diagrama De Secuencia: Modificar Vehiculo

DIAGRAMA DE SECUENCIA: ELIMINAR VEHICULOS

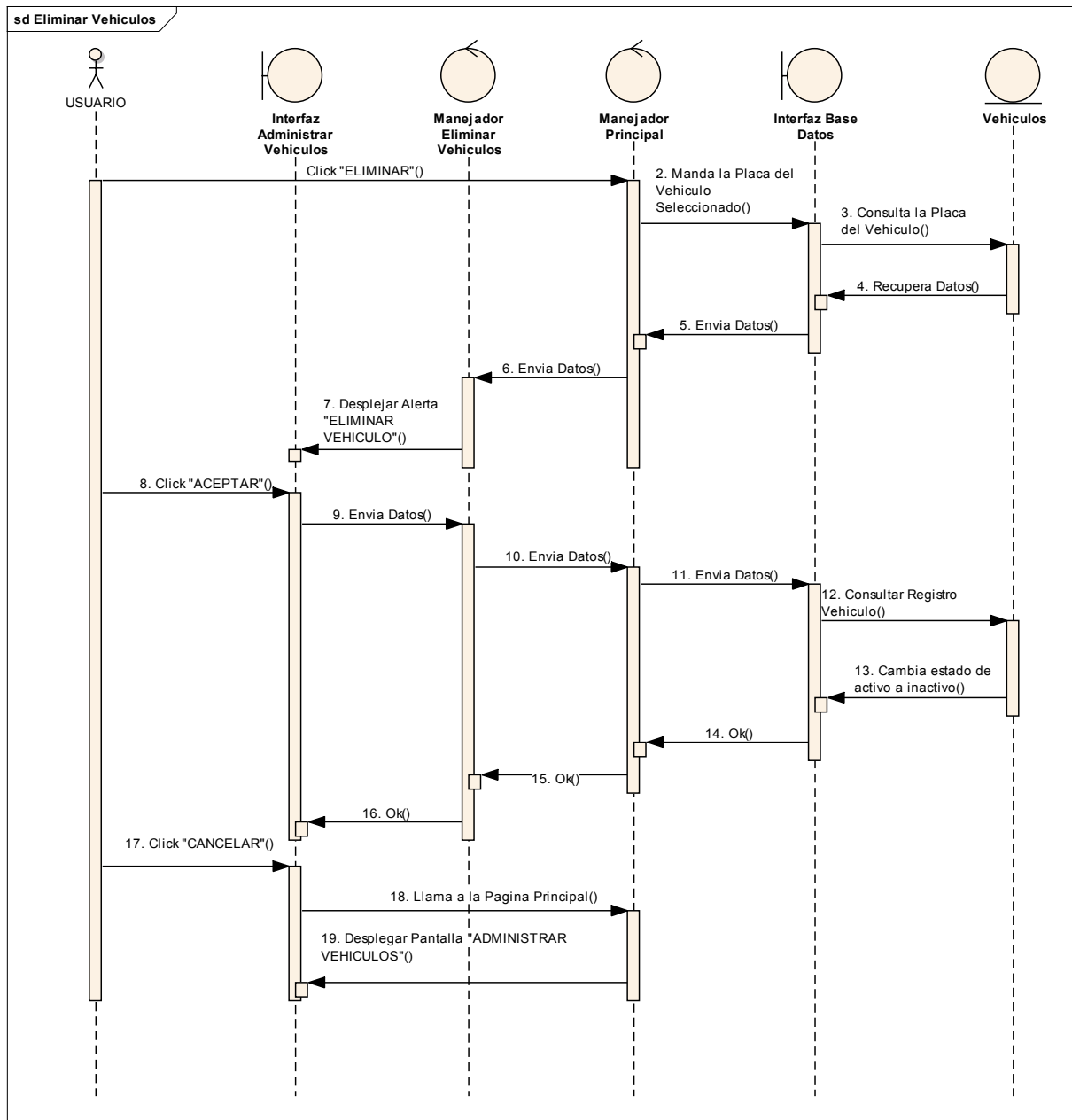


Figura 182 Diagrama De Secuencia: Eliminar Vehiculo

DIAGRAMA DE SECUENCIA: ADMINISTRAR BITACORAS

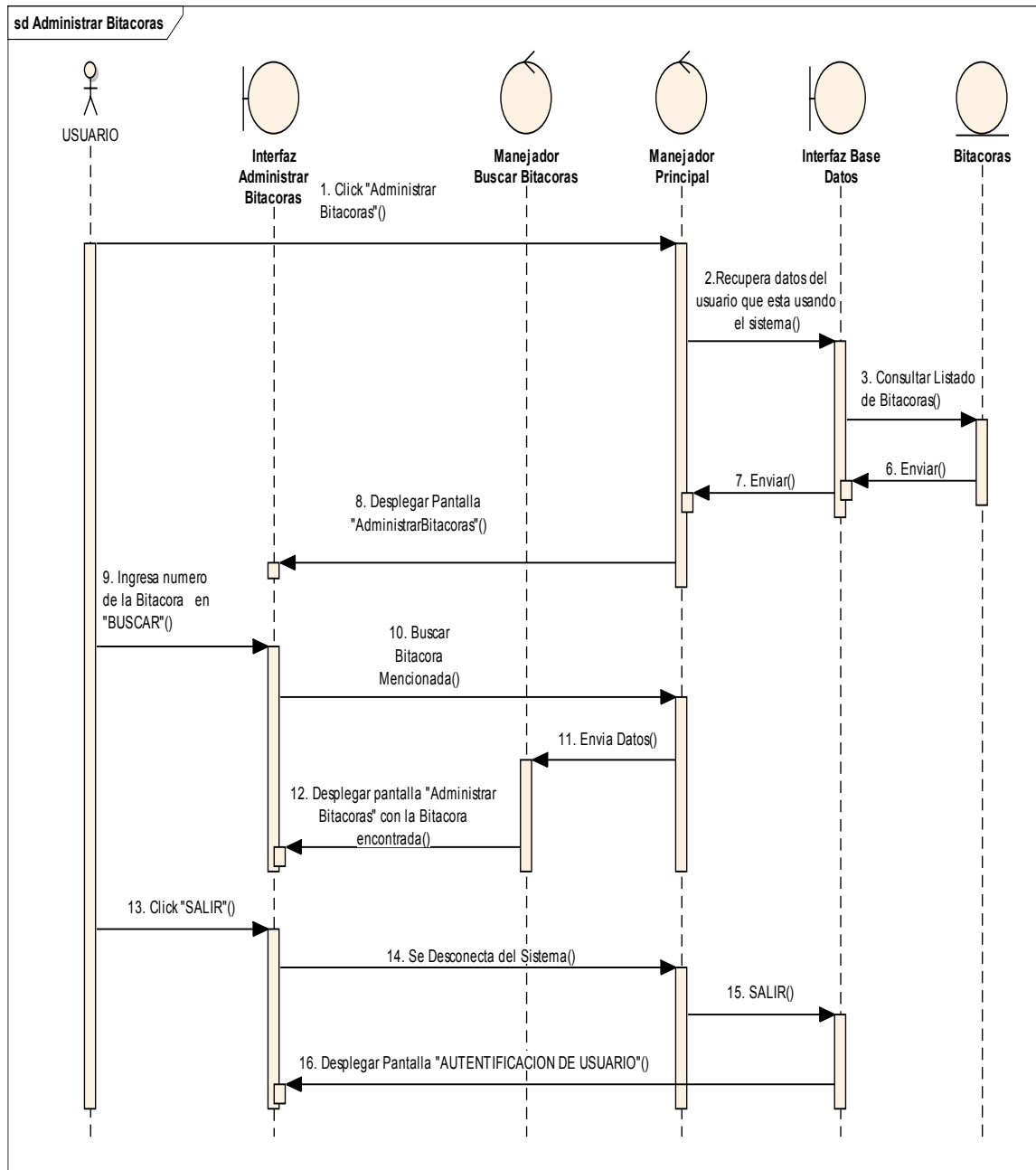


Figura 183 Diagrama De Secuencia: Administrar Bitacoras

DIAGRAMA DE SECUENCIA: ADICIONAR BITACORAS

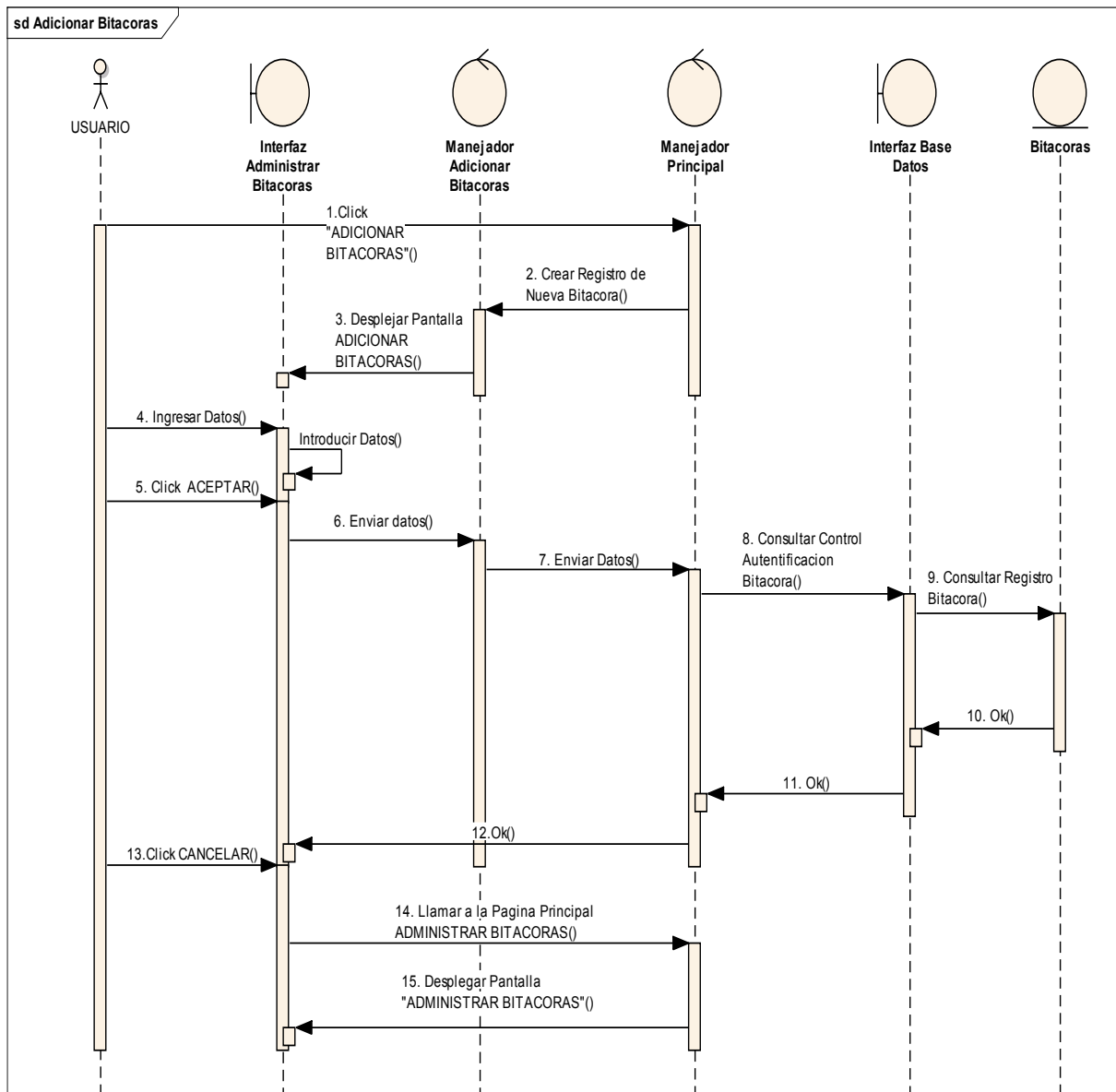


Figura 184 Diagrama De Secuencia: Adicionar Bitácoras

DIAGRAMA DE SECUENCIA: MODIFICAR BITACORAS

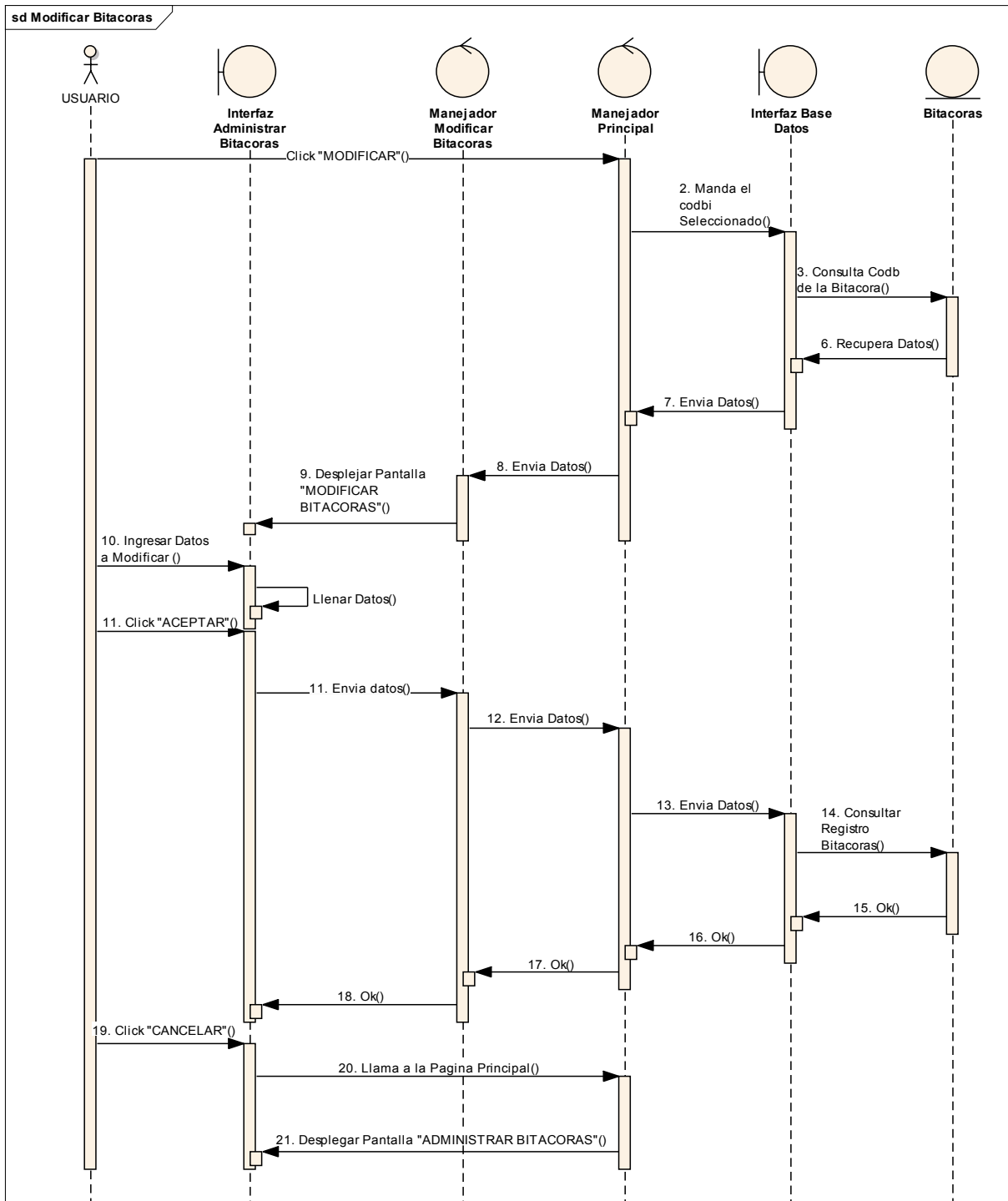


Figura 185 Diagrama De Secuencia: Modificar Bitácoras

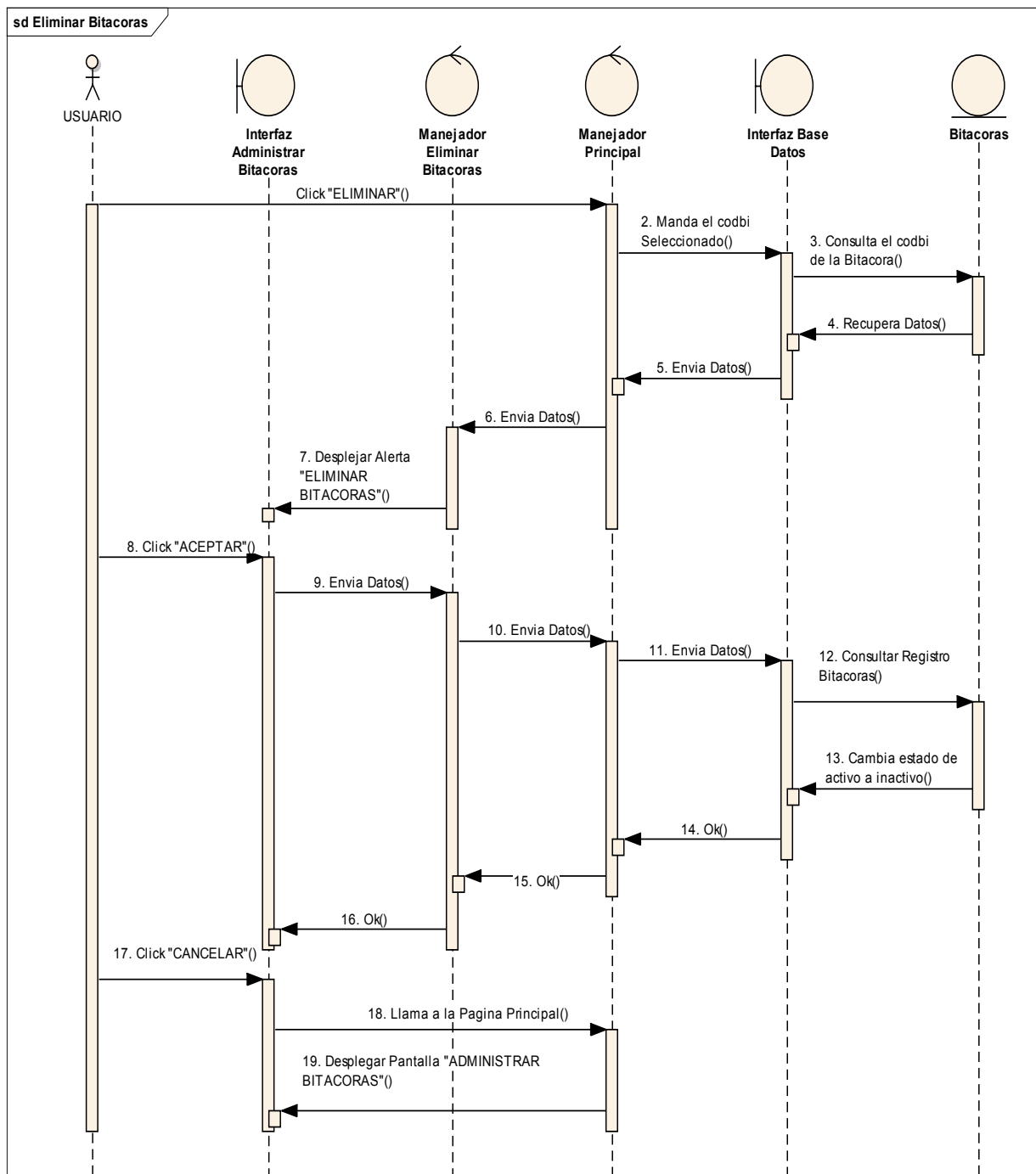
DIAGRAMA DE SECUENCIA: ELIMINAR BITACORAS

Figura 186 Diagrama De Secuencia: Eliminar Bitácoras

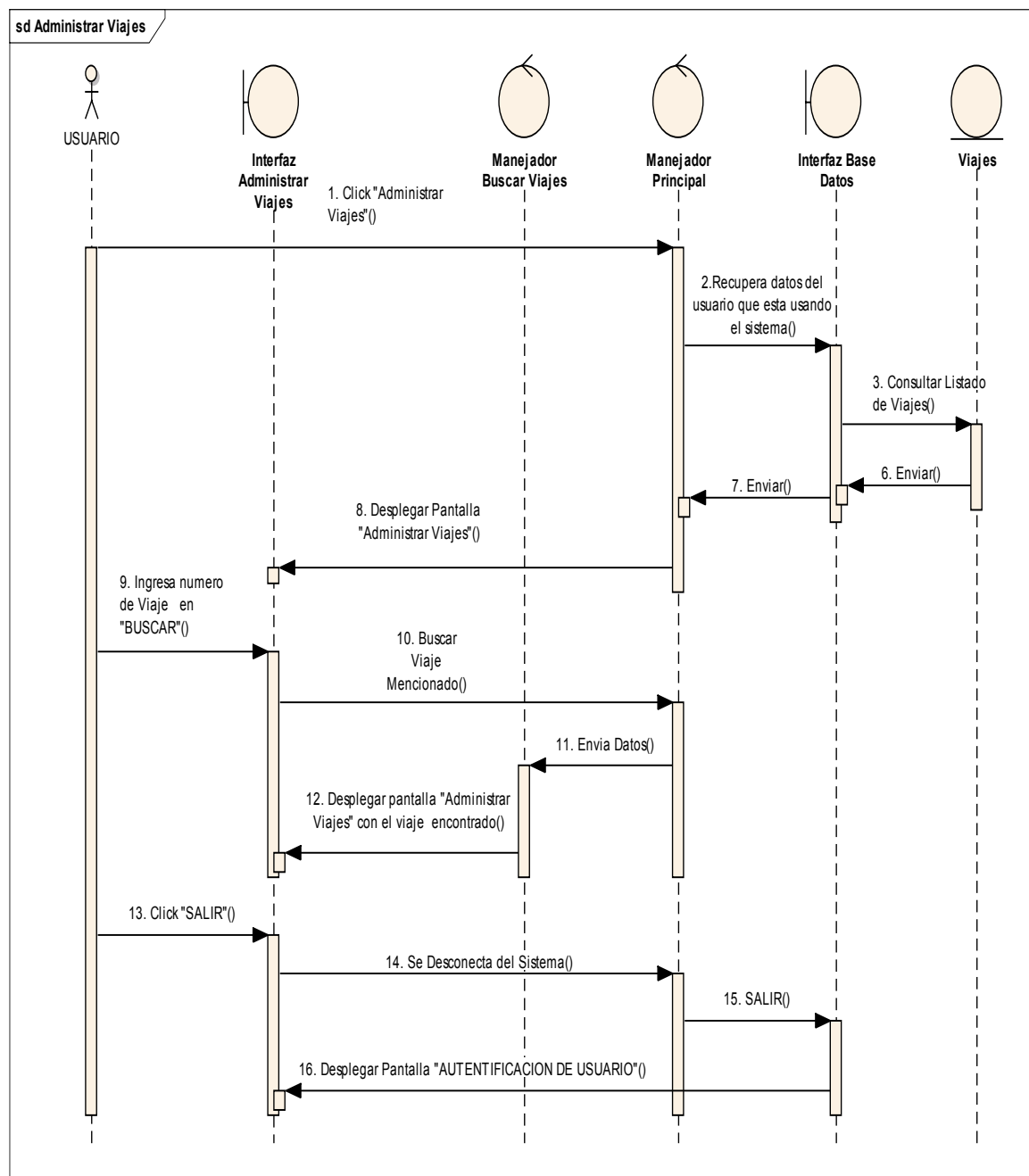
DIAGRAMA DE SECUENCIA: ADMINISTRAR VIAJES

Figura 187 Diagrama De Secuencia: Administrar Viajes

DIAGRAMA DE SECUENCIA: ADICIONAR VIAJES

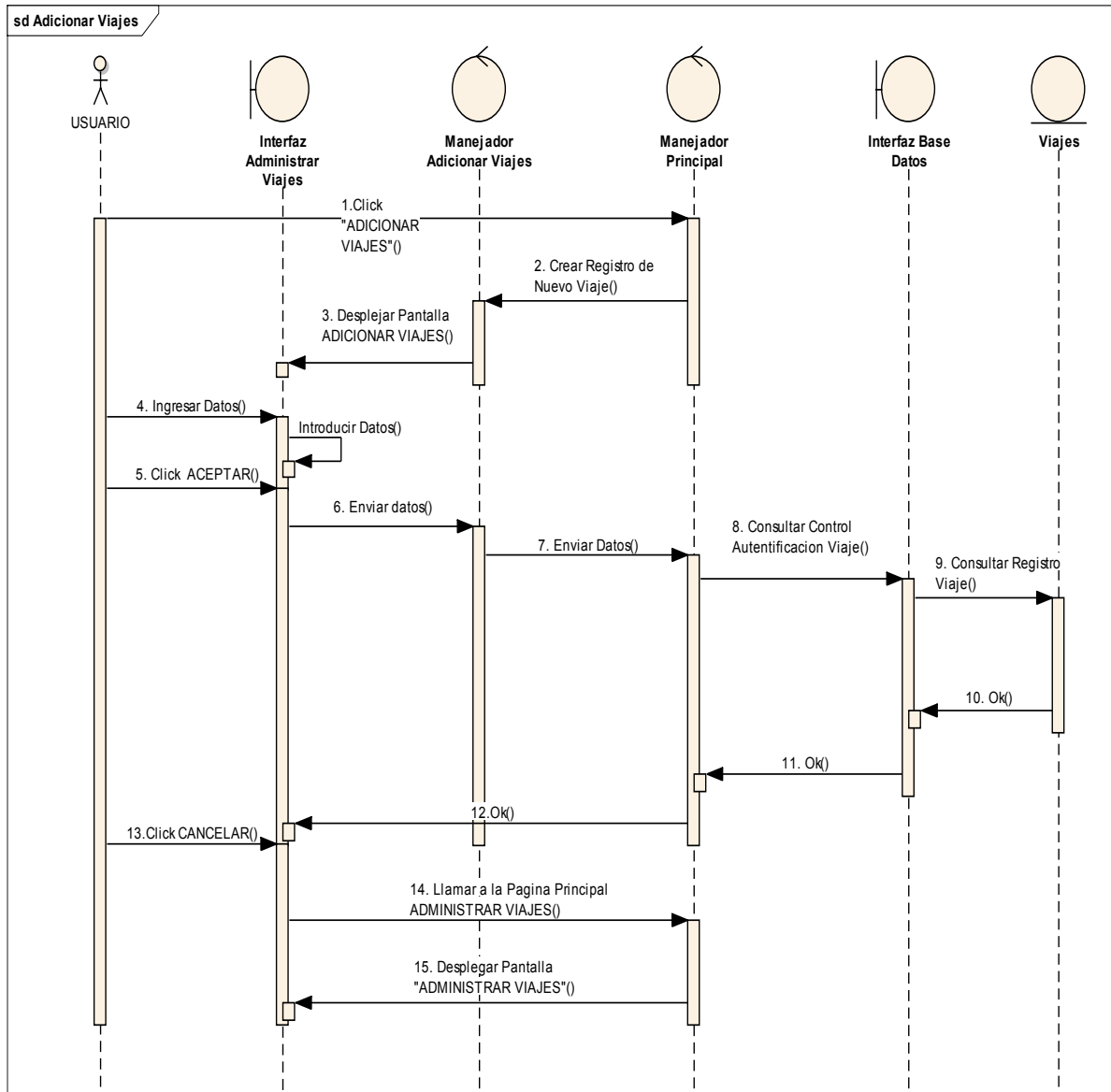


Figura 188

Diagrama De Secuencia: Adiconar Viajes

DIAGRAMA DE SECUENCIA: MODIFICAR VIAJES

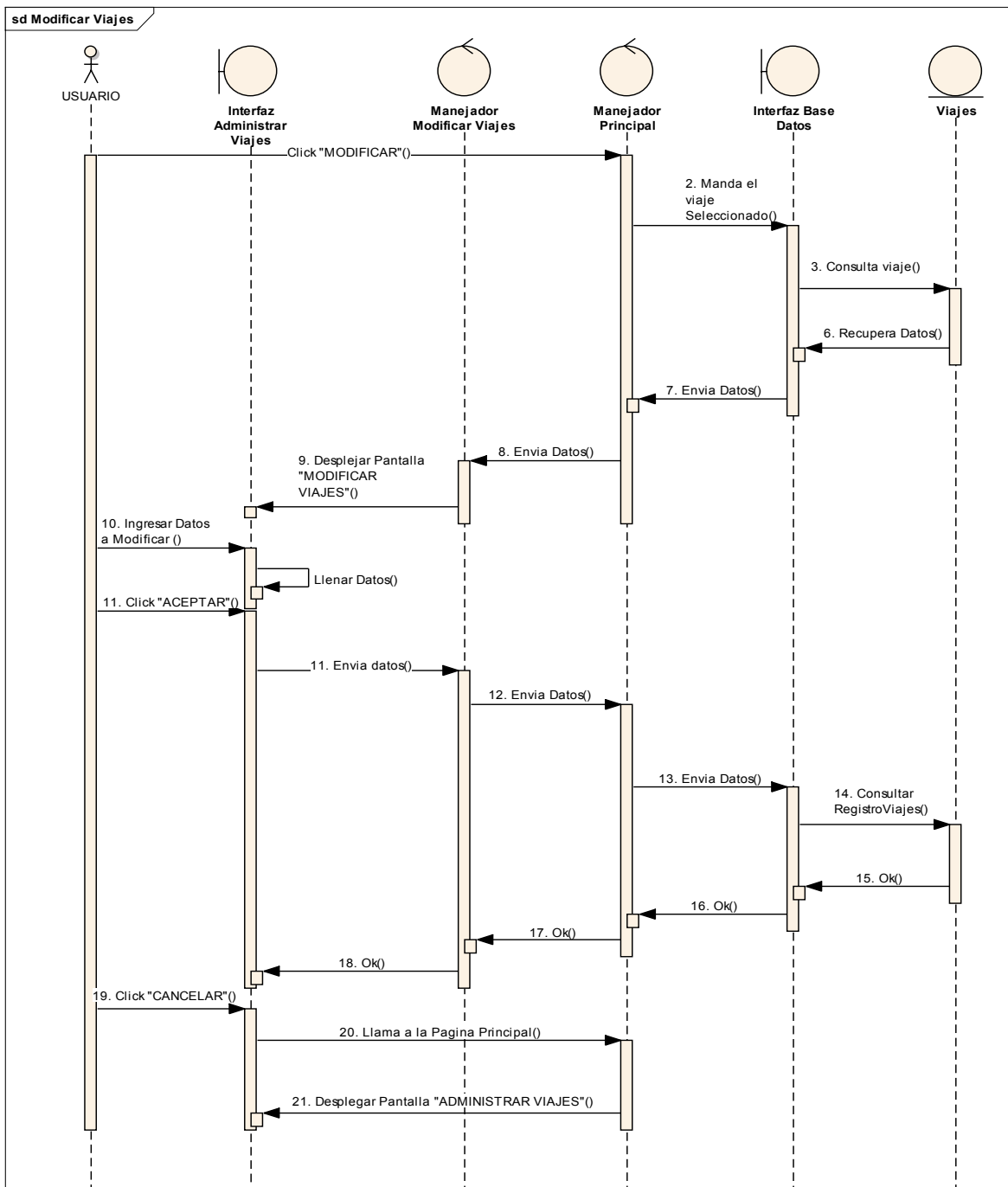


Figura 189

Diagrama De Secuencia: Modificar Viajes

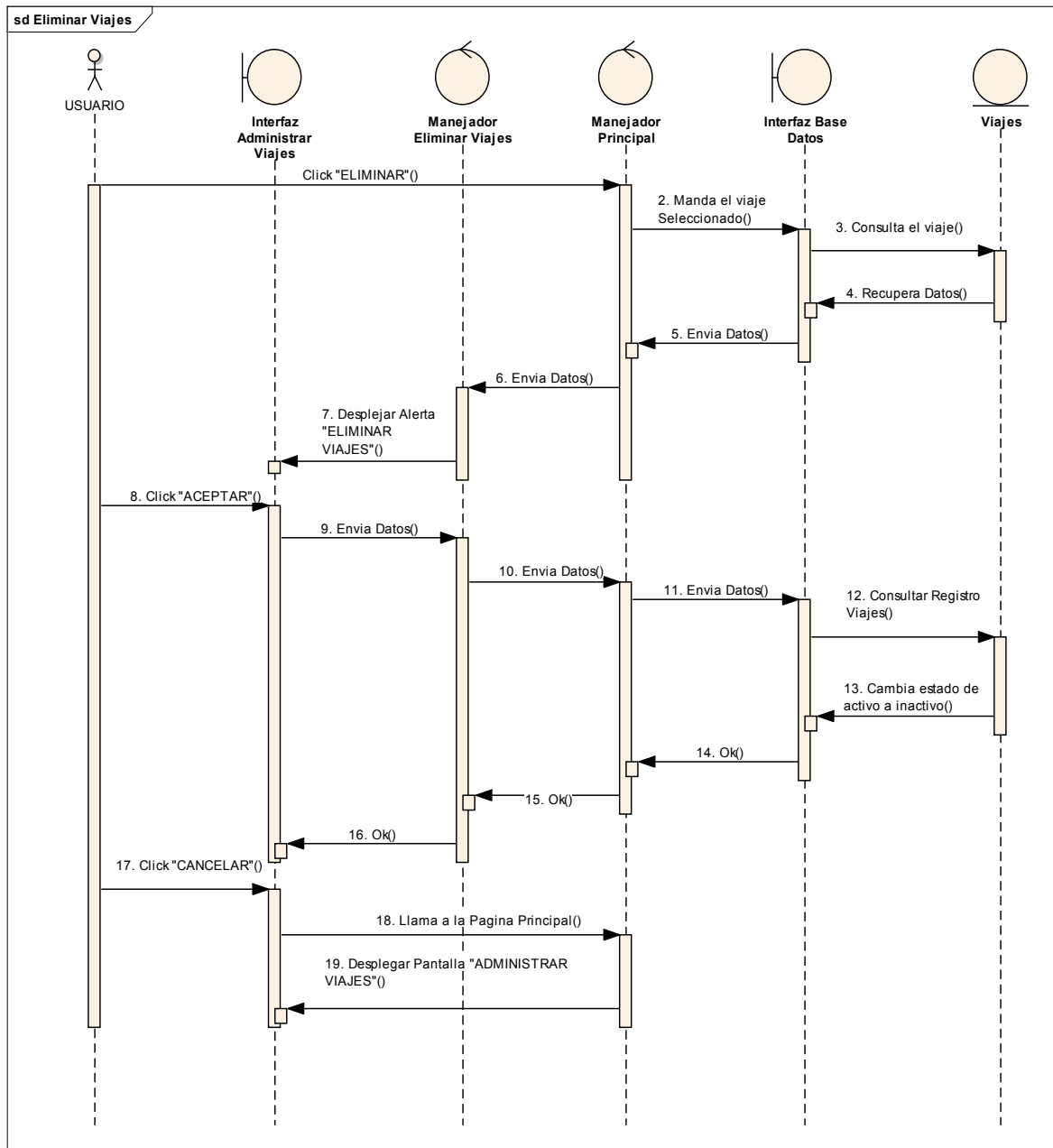
DIAGRAMA DE SECUENCIA: ELIMINAR VIAJES

Figura 190 Diagrama De Secuencia: Eliminar Viajes

DIAGRAMA DE SECUENCIA: ASIGNAR AYUDANTES

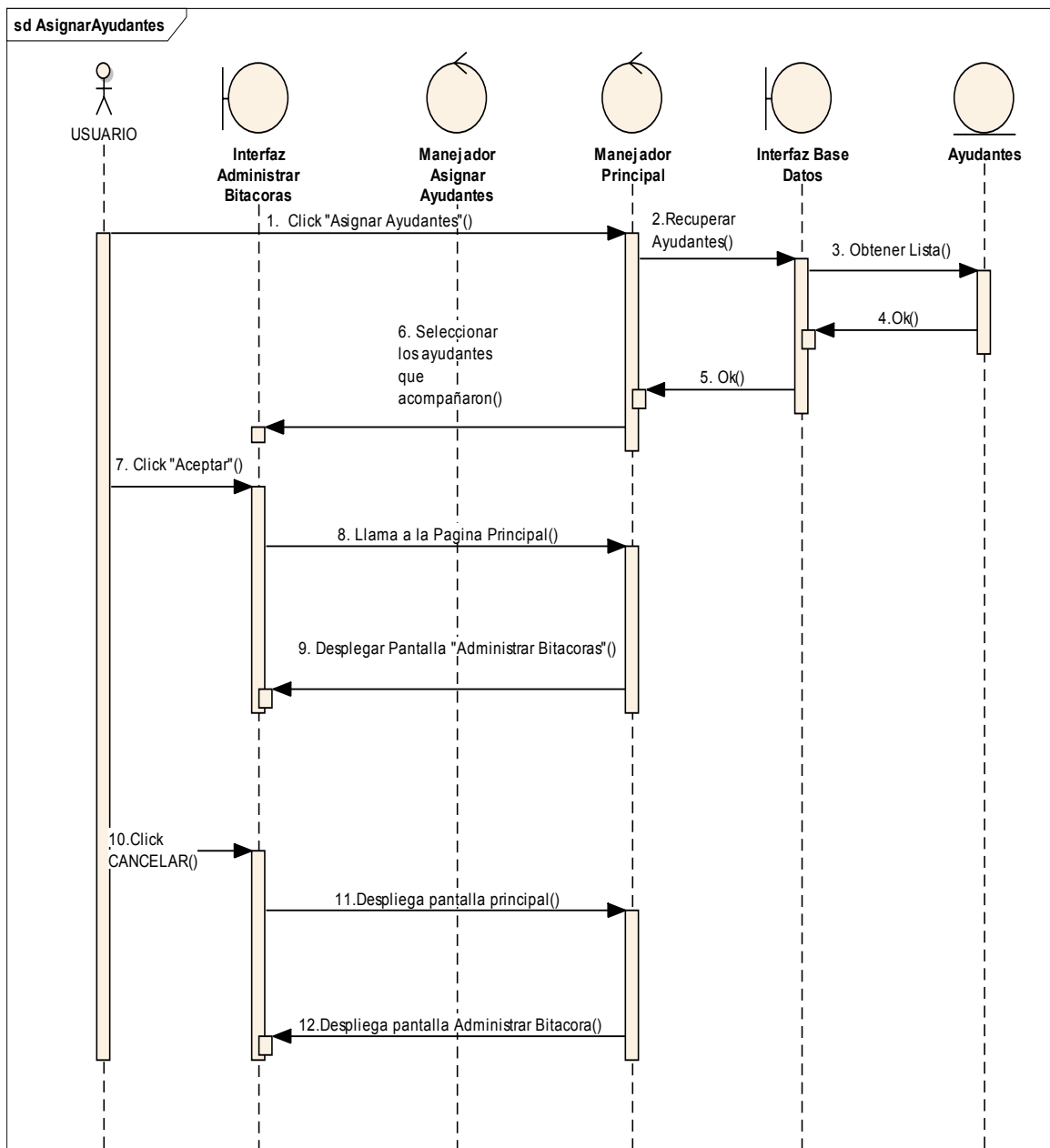


Figura 191 Diagrama De Secuencia: Asignar Ayudantes

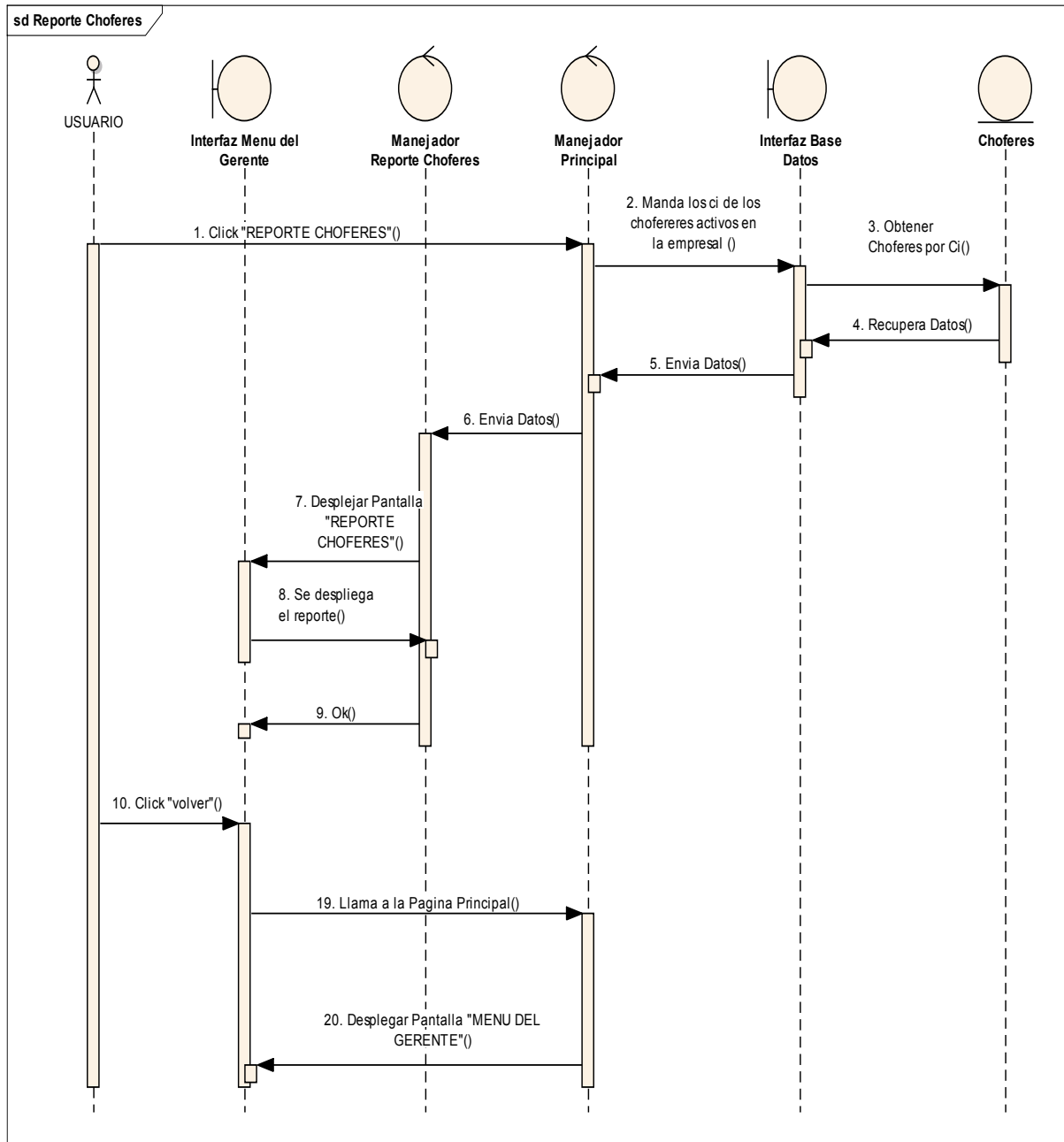
DIAGRAMA DE SECUENCIA: REPORTES CHOFERES

Figura 192 Diagrama De Secuencia: Reporte Choferes

DIAGRAMA DE SECUENCIA: REPORTES AYUDANTES

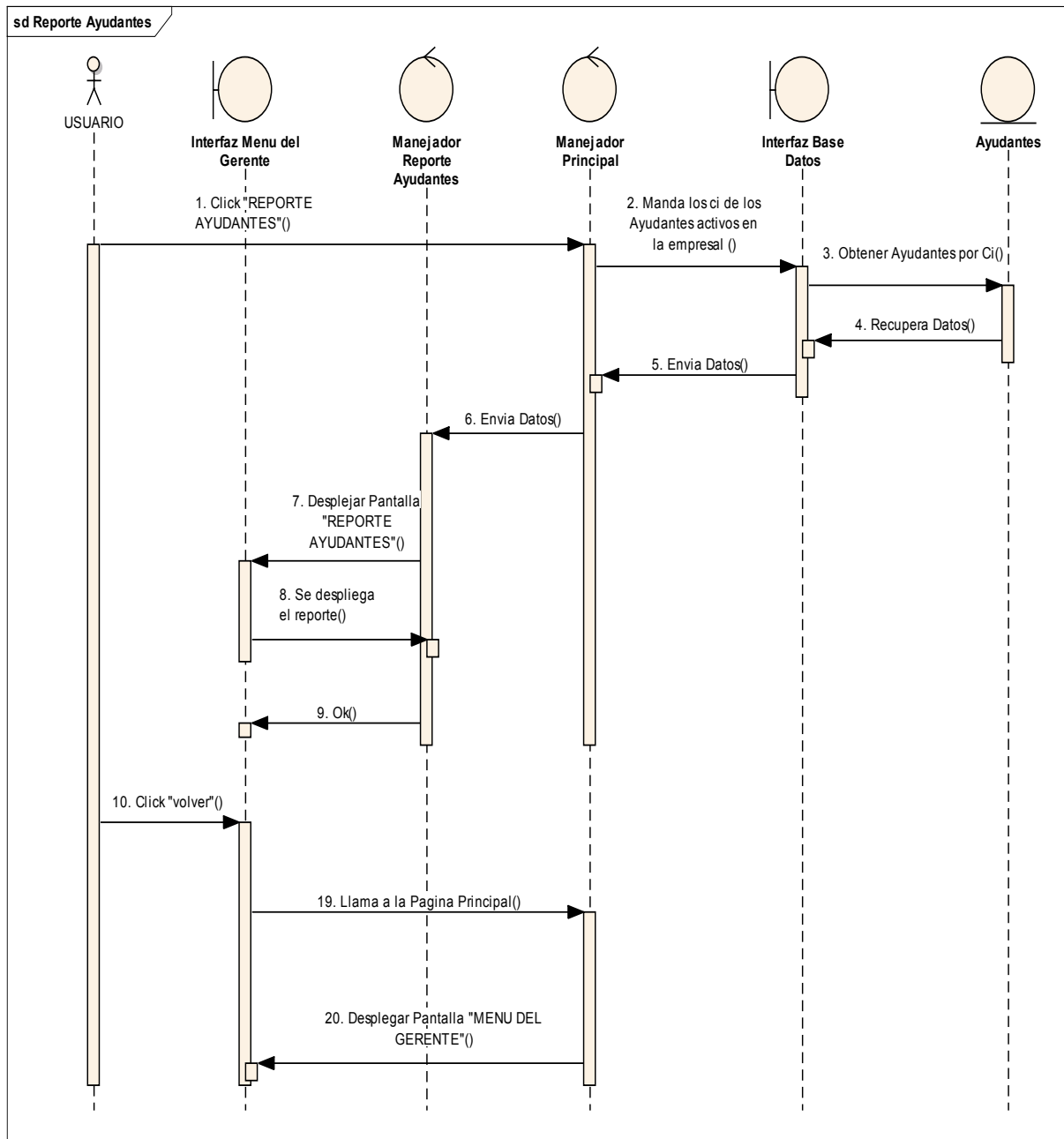


Figura 193 Diagrama De Secuencia: Reporte Ayudantes

DIAGRAMA DE SECUENCIA: REPORTES ZONAS

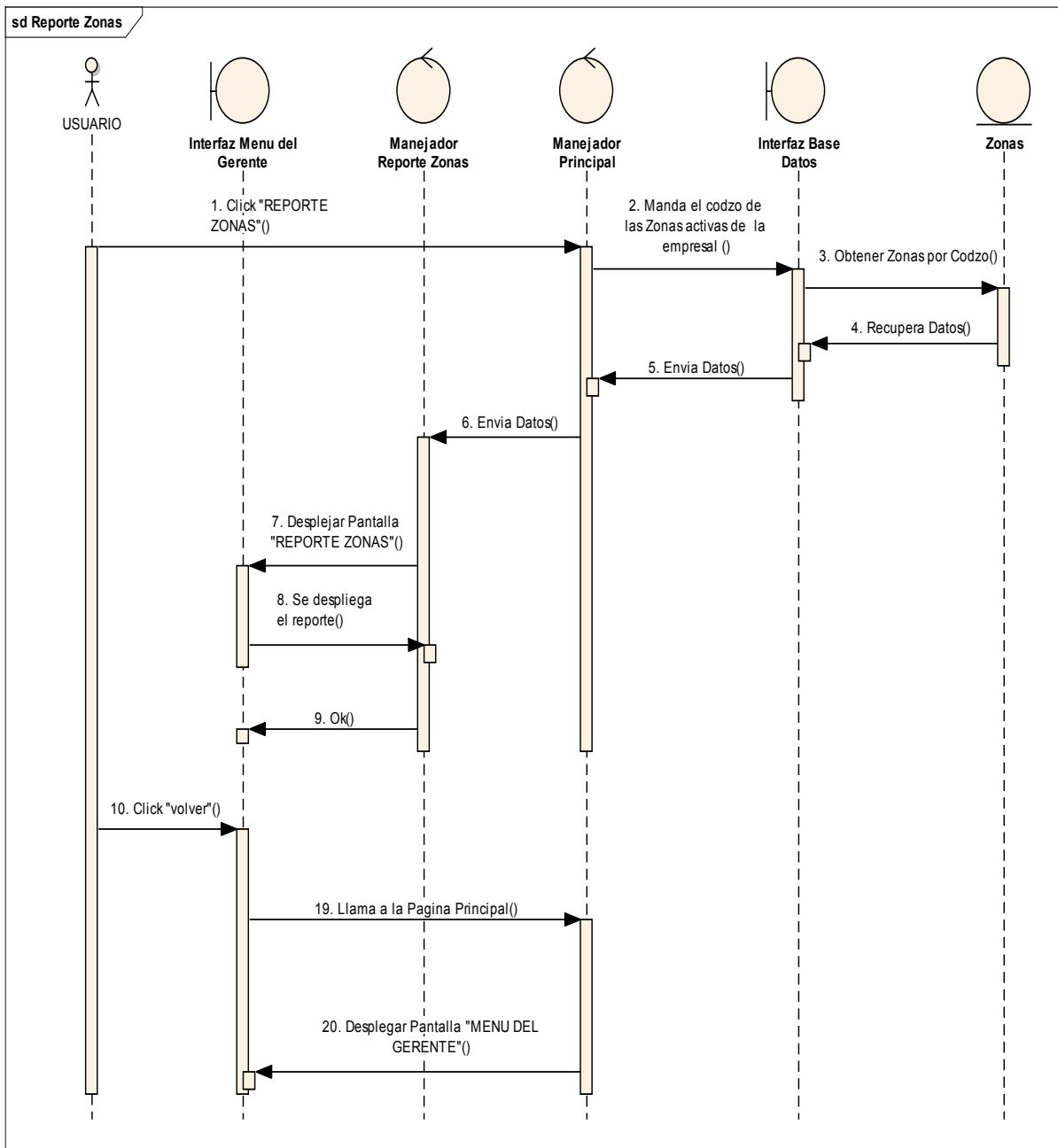


Figura 194 Diagrama De Secuencia: Reporte Zonas

DIAGRAMA DE SECUENCIA: REPORTES VEHICULOS

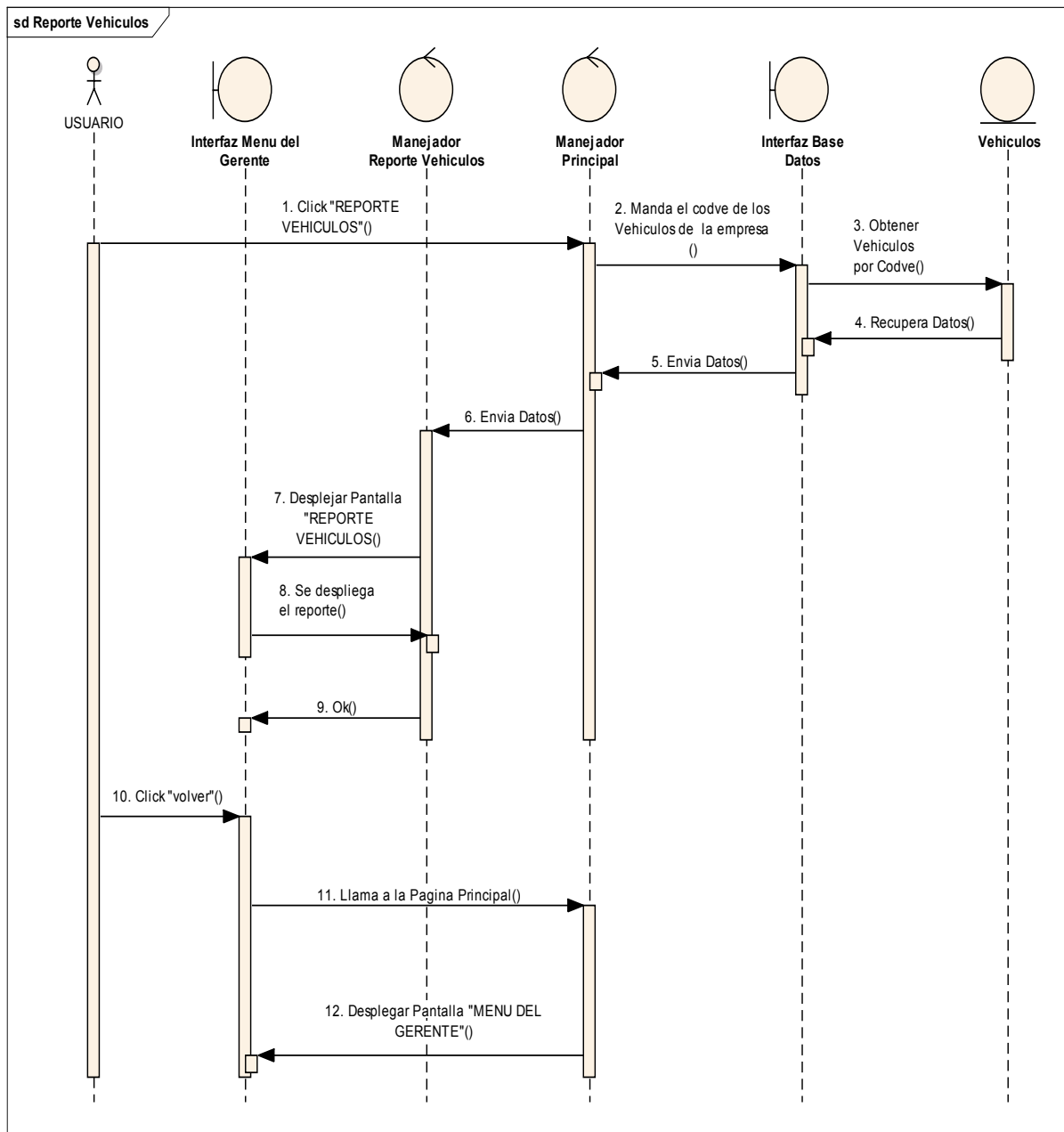


Figura 195

Diagrama De Secuencia: Reporte Vehiculos

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS

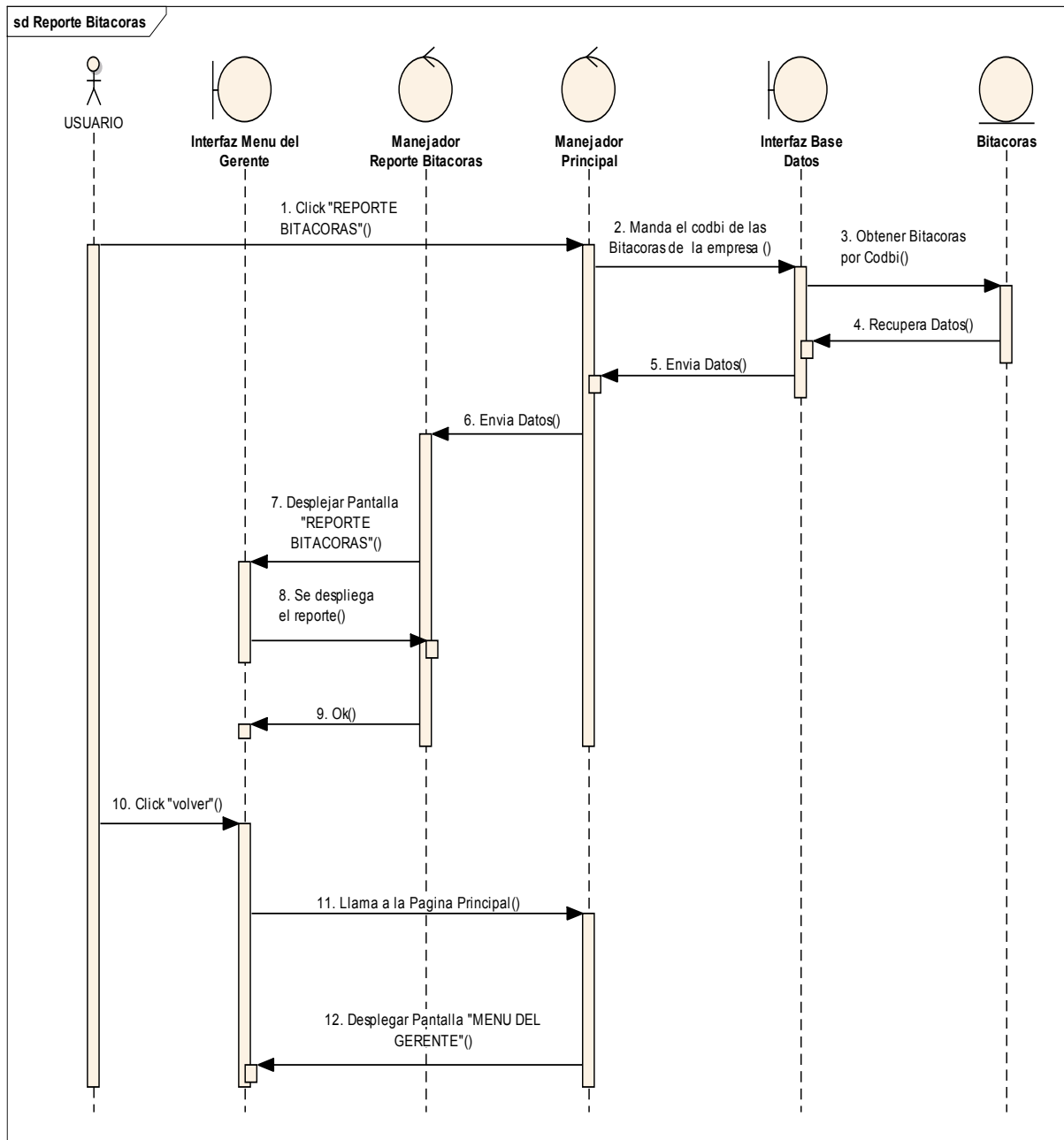


Figura 196 Diagrama De Secuencia: Reporte Bitácoras

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / DIRUNO POR AÑO

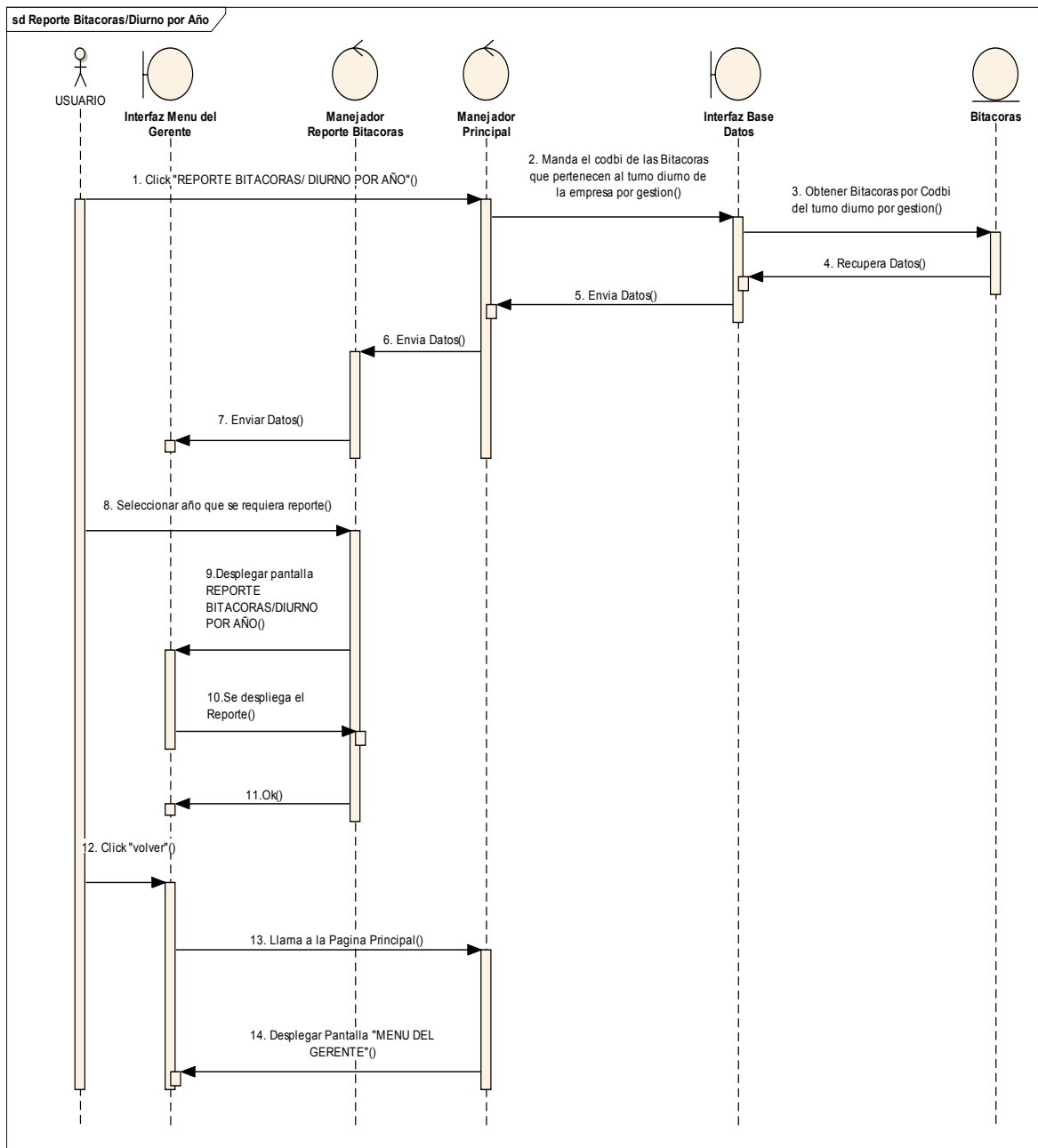


Figura 197 Diagrama De Secuencia: Reporte Bitácoras/Diurno Por Año

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / NOCTURNO POR AÑO

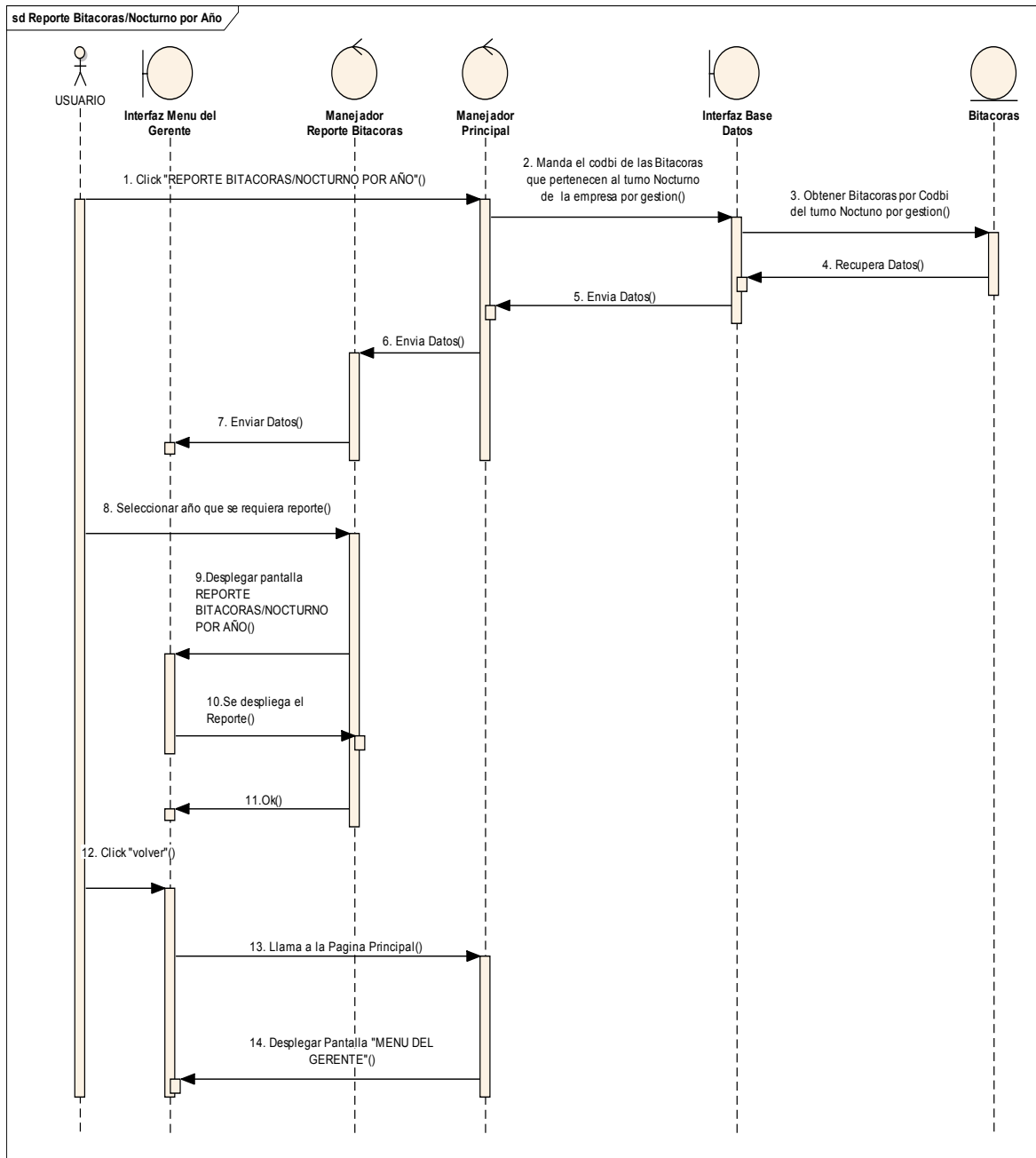


Figura 198 Diagrama De Secuencia: Reporte Bitácoras/Nocturno Por Año

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / DIRUNO POR MES Y AÑO

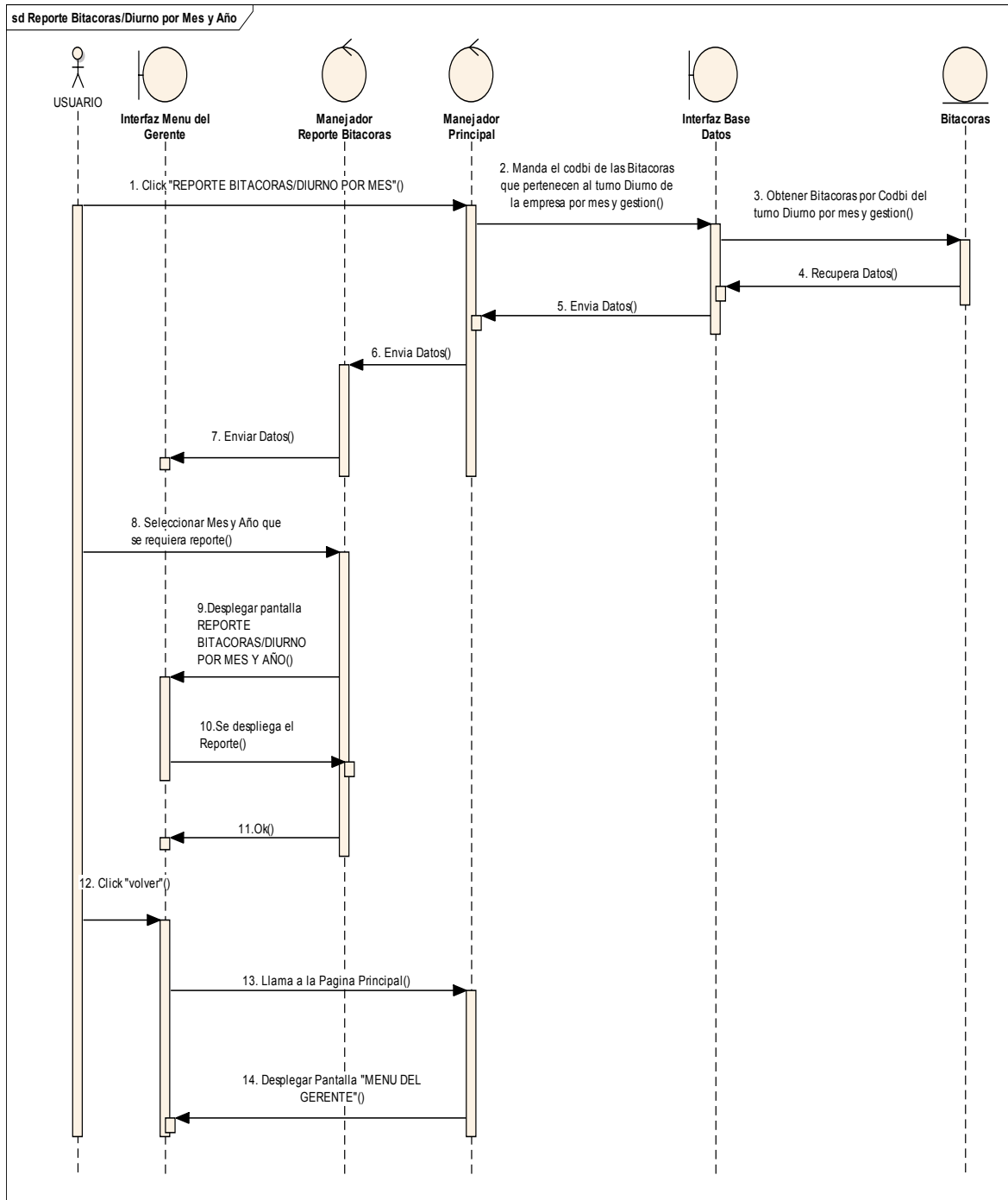


Figura 199 Diagrama De Secuencia: Reporte Bitácoras/Diurno Por Mes Y Año

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / NOCTURNO MES Y AÑO

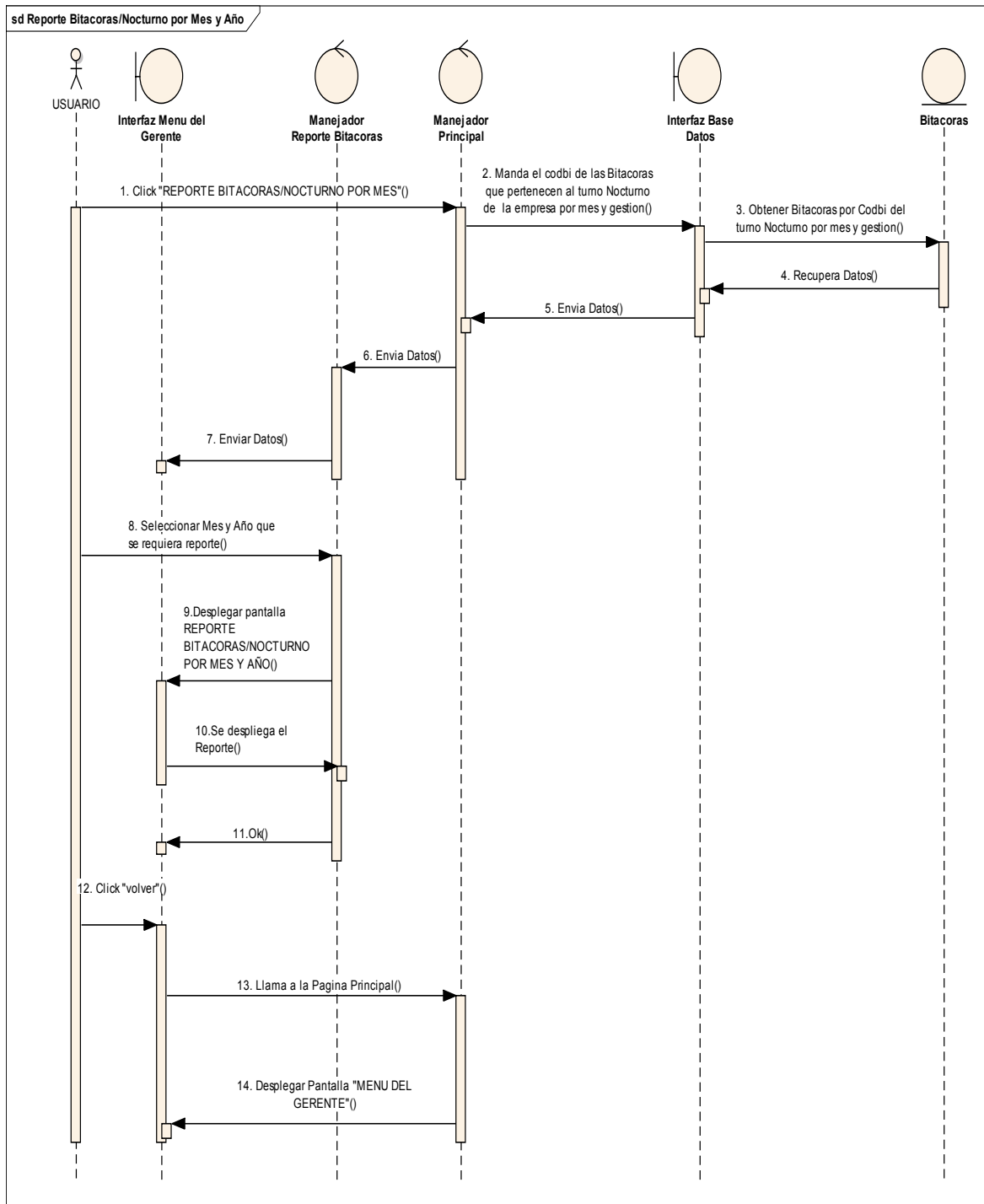


Figura 200 Diagrama De Secuencia: Reporte Bitácoras/ Nocturno Por Mes Y Año

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / TOTAL POR AÑO Y CATEGORIA

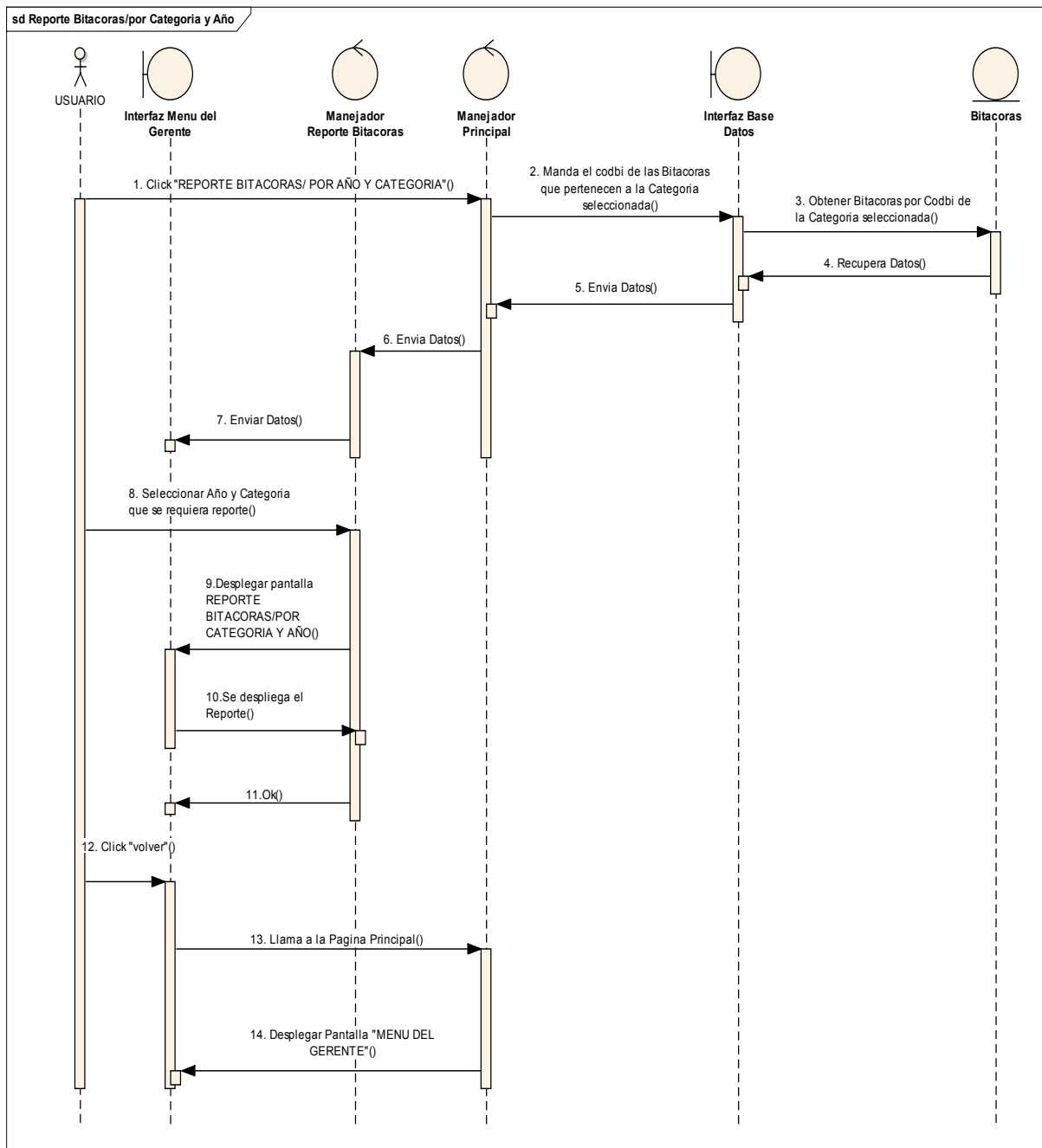


Figura 201 Diagrama De Secuencia: Reporte Bitácoras/Total Por Año Y Categoría

DIAGRAMA DE SECUENCIA: REPORTES BITACORAS / TOTAL POR MES, AÑO Y CATEGORIA

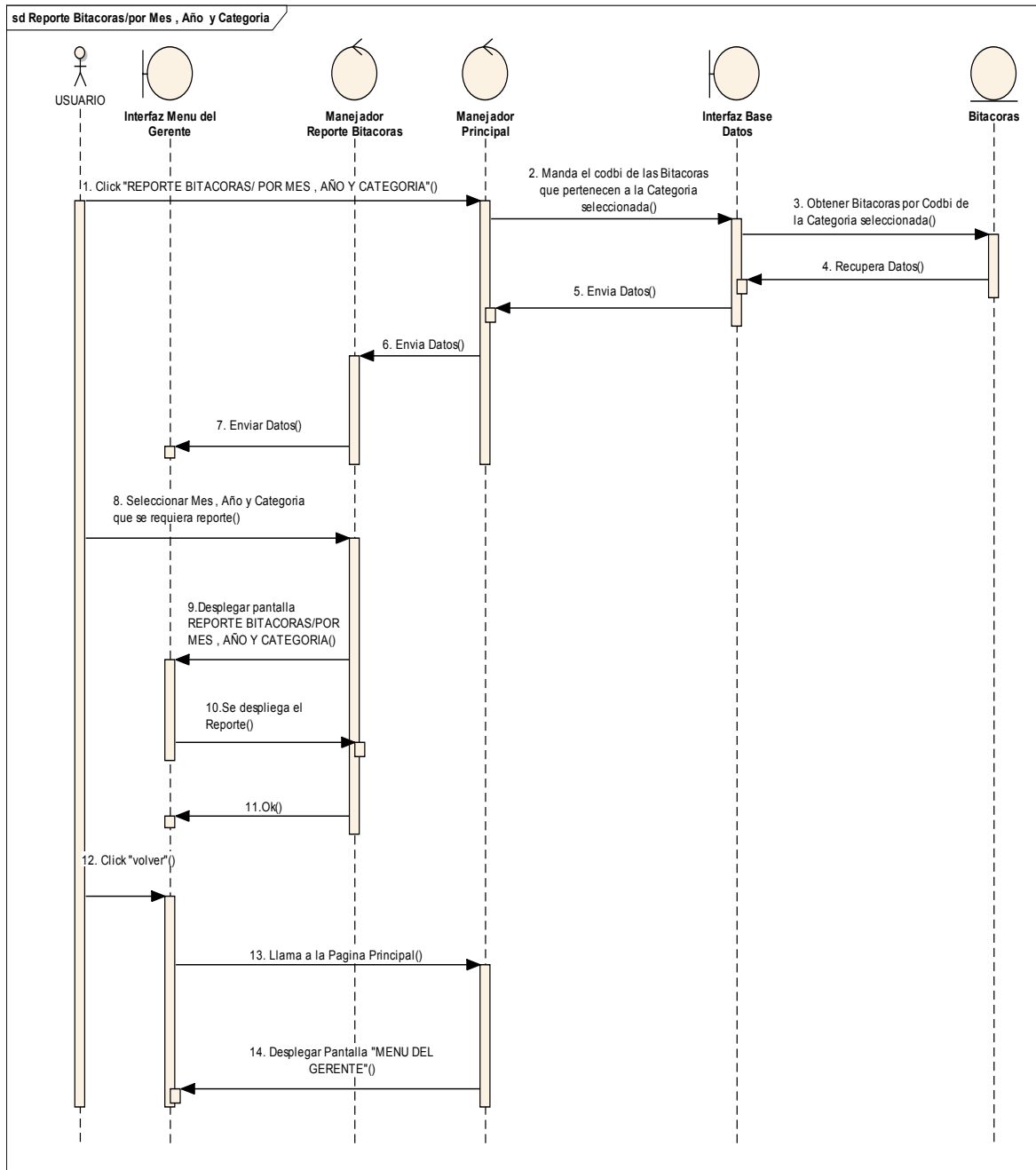


Figura 202 Diagrama De Secuencia: Reporte Bitácoras/Total Por Mes, Año Y Categoría

2.1.6.6 DIAGRAMA DE CLASES

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.

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.

2.1.2.2.3.5.3.1.3 DIAGRAMA DE CLASES

class Diagrama de Clases

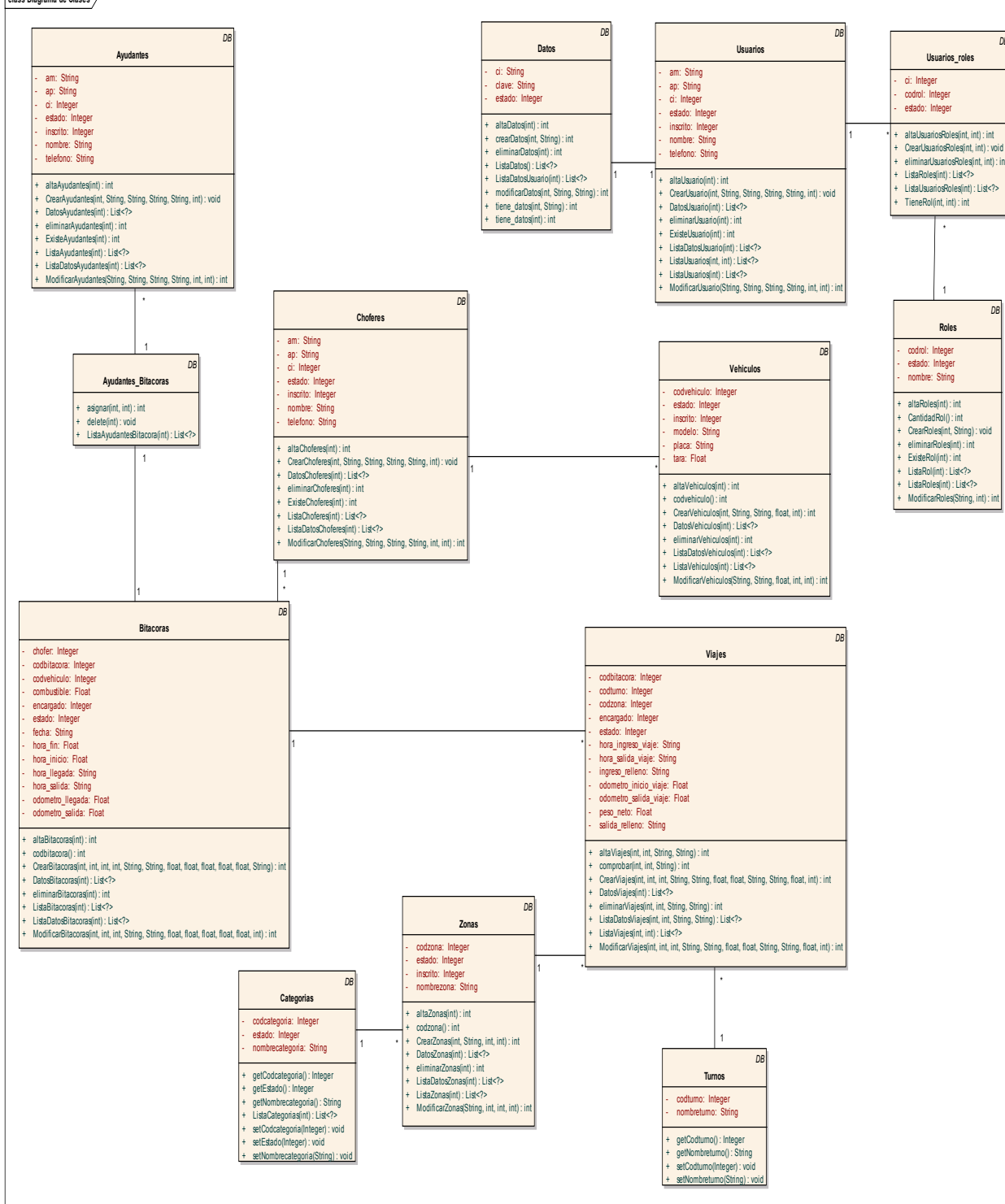


Figura 203 Diagrama de Clases

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.

OBJETO USUARIOS

OBJETO:	Usuarios
MÉTODO:	DatosUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> DatosUsuario(int ci){ return db.queryForList("SELECT * FROM Usuarios WHERE ci=?;", new Object[]{new Integer(ci)}); }</pre>	
DESCRIPCIÓN	
Método para generar datos de un usuario	

Tabla 67 Método Datos Usuario

OBJETO:	Usuarios
MÉTODO:	ListaUsuarios
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN
<pre>List<?> ListaUsuarios(int ci, int estado){ return db.queryForList("SELECT * FROM Usuarios WHERE ci<>? and estado=?;", new Object[] {new Integer(ci),new Integer(estado)}); }</pre>
DESCRIPCIÓN
Se lista todos los usuarios

Tabla 68 Método Listar Usuarios

OBJETO:	Usuarios
MÉTODO:	ListaDatosUsuarios
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatosUsuario(int ci){ return db.queryForList("SELECT * FROM Usuarios WHERE ci=?;", new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se lista todos los datos del usuario	

Tabla 69 Método Listar Datos De Usuarios

OBJETO:	Usuarios
MÉTODO:	CrearUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> CrearUsuario(int ci, String nombre, String ap, String am, String teleono, int inscrito){ String sql = "INSERT INTO usuarios(ci, nombre, ap, am, telefono,inscrito) VALUES (?, ?, ?, ?, ?, ?)"; db.update(sql, new Object[]{new Integer(ci), nombre, ap, am, teleono,new Integer(inscrito)}});} </pre>	
DESCRIPCIÓN	
Se crea usuarios.	

Tabla 70 Método Crear Usuarios

OBJETO:	Usuarios
MÉTODO:	ModificarUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

```

ModificarUsuario(String nombre, String ap, String am, String telefono,int inscrito,
int ci) {

return db.update("UPDATE usuarios SET  nombre=?, ap=?, am=?,  telefono=?,
inscrito=? WHERE ci=?;",

new Object[] {nombre,ap,am,telefono,new Integer(inscrito),new Integer(ci)});

}

```

DESCRIPCIÓN

Se modifica al usuario

Tabla 71 Método Modificar Usuarios

OBJETO:	Usuarios
MÉTODO:	eliminarUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> eliminarUsuario(int ci) { if(ci==7197942){return 0;} else{return db.update("update usuarios set estado='0' where ci=?;",new Object[] {new Integer(ci)});}} </pre>	
DESCRIPCIÓN	
Se elimina al usuario	

Tabla 72 Método Eliminar Usuarios

OBJETO:	Usuarios
MÉTODO:	altaUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaUsuario(int ci) { return db.update("update usuarios set estado='1' where ci=?",new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se genera alta de los usuarios	

Tabla 73 Método Alta Usuarios

OBJETO:	Usuarios
MÉTODO:	ExisteUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>ExisteUsuario(int ci){ String sql = "Select count(*) from usuarios where ci=?";</pre>	

return db.queryForInt(sql,new Object[] {new Integer(ci)});}
DESCRIPCIÓN
Se pregunta si existe el usuario

Tabla 74 Método Existe Usuarios

OBJETO ROLES

OBJETO:	Roles
MÉTODO:	ListaRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaRoles(int estado){ return db.queryForList("SELECT * FROM roles WHERE estado=?;", new Object[] {new Integer(estado)});}</pre>	
DESCRIPCIÓN	
Se lista los roles	

Tabla 75 Método Lista De Roles

OBJETO:	Roles
MÉTODO:	ListaRol
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaRol(int codrol){ return db.queryForList("SELECT * FROM roles WHERE codrol=?", new Object[] {new Integer(codrol)});}</pre>	
DESCRIPCIÓN	
Se lista un rol específico	

Tabla 3 Método Lista De Rol

OBJETO:	Roles
MÉTODO:	CrearRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>CrearRoles(int codrol, String nombre){ String sql = "INSERT INTO roles(codrol, nombre) VALUES (?, ?)"; db.update(sql, new Object[] {new Integer(codrol), nombre});}</pre>	
DESCRIPCIÓN	
Se crea roles	

Tabla 77 Método Crear Roles

OBJETO:	Roles
MÉTODO:	ModificarRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> ModificarRoles(String nombre, int codrol) { return db.update("UPDATE roles SET nombre=? WHERE codrol=?;", new Object[] {nombre,new Integer(codrol)}); } </pre>	
DESCRIPCIÓN	
Se modifica roles	

Tabla 78 Método Modificar Roles

OBJETO:	Roles
MÉTODO:	eliminarRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> eliminarRoles(int codrol) { return db.update("update roles set estado='0' where codrol=?;", new Object[] {new Integer(codrol)});} </pre>	

DESCRIPCIÓN
Se elimina roles

Tabla 79 Método Eliminar Roles

OBJETO:	Roles
MÉTODO:	altaRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaRoles(int codrol) { return db.update("update roles set estado='1' where codrol=?;", new Object[] {new Integer(codrol)});}</pre>	
DESCRIPCIÓN	
Se da de alta roles	

Tabla 80 Método Alta De Roles

OBJETO:	Roles
MÉTODO:	ExisteRol
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

<pre>ExisteRol(int codrol){ String sql = "Select count(*) from roles where codrol=?"; return db.queryForInt(sql,new Object[]{new Integer(codrol)});}</pre>
DESCRIPCIÓN
Se pregunta si existe rol

Tabla 81 Método Existe Rol

OBJETO:	Rol
MÉTODO:	CantidadRol
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>CantidadRol(){ String sql = "Select count(*) from roles"; return db.queryForInt(sql);}</pre>	
DESCRIPCIÓN	
Se ve la cantidad de roles que existen	

Tabla 82 Método Cantidad De Roles

OBJETO USUARIOS_ROLES

OBJETO:	Usuarios_roles
----------------	----------------

MÉTODO:	ListaRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaRoles(int ci){ return db.queryForList("SELECT * FROM usuarios_rols ur, roles r WHERE r.codrol=ur.codrol and ur.estado=1 and ci=?"; new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se lista los usuarios con roles	

Tabla 83 Método Lista Roles

OBJETO:	Usuarios_rols
MÉTODO:	ListaUsuariosRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaUsuariosRoles(int estado){ return db.queryForList("SELECT u.ci,u.nombre, u.ap ,u.am ,r.nombre as nombrerol,r.codrol FROM usuarios_rols ur, usuarios u, roles r WHERE r.codrol=ur.codrol and u.ci=ur.ci and u.estado=1 and r.estado=1 and ur.estado=?";</pre>	

<code>new Object[] {new Integer(estado)}});}</code>
DESCRIPCIÓN
Se lista los usuarios con los roles activos

Tabla 84 Método Lista Usuarios Roles

OBJETO:	Usuarios_roles
MÉTODO:	TieneRol
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>TieneRol(int codrol,int ci){ String sql = "Select count(*) from usuarios_roles where codrol=? and ci=?"; return db.queryForInt(sql,new Object[] {new Integer(codrol),new Integer(ci)}});}</pre>	
DESCRIPCIÓN	
Se pregunta si se tiene rol	

Tabla 85 Método Tiene Rol

OBJETO:	Usuarios_roles
MÉTODO:	CrearUsuariosRoles
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> CrearUsuariosRoles(int codrol, int ci){ String sql = "INSERT INTO usuarios_roles(codrol, ci) VALUES (?, ?);"; db.update(sql, new Object[] {new Integer(codrol), new Integer(ci)});} </pre>	
DESCRIPCIÓN	
Se usuarios con sus roles	

Tabla 4 Método Crear Usuarios Roles

OBJETO:	Usuarios_roles
MÉTODO:	eliminarUsuariosRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> eliminarUsuariosRoles(int codrol,int ci) { return db.update("update usuarios_roles set estado='0' where codrol=? and ci=?;", new Object[] {new Integer(codrol),new Integer(ci)}); } </pre>	
DESCRIPCIÓN	
Se elimina al usuarios con su rol respectivo	

Tabla 87 Método Eliminar Usuarios Roles

OBJETO:	Usuarios_roles
MÉTODO:	altaUsuariosRoles
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaUsuariosRoles(int codrol,int ci) { return db.update("update usuarios_roles set estado='1' where codrol=? and ci=?"; new Object[] {new Integer(codrol),new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se da alta de usuarios con sus roles	

Tabla 88 Método Alta Usuarios Roles

OBJETO DATOS

OBJETO:	Datos
MÉTODO:	tiene_datos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>tiene_datos(int ci,String clave){ String sql = "Select count(*) from datos where estado=1 and ci=? and clave=?"; return db.queryForInt(sql,new Object[] {new Integer(ci),clave});}</pre>	

DESCRIPCIÓN
Se pregunta si tiene datos y si esta activo

Tabla 89 Método Tiene Datos

OBJETO:	Datos
MÉTODO:	ListaDatos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatos(){ return db.queryForList("SELECT * FROM datos WHERE estado=1;");}</pre>	
DESCRIPCIÓN	
Se lista datos de los activos	

Tabla 90 Método Lista Datos

OBJETO:	Datos
MÉTODO:	crearDatos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

<pre> crearDatos(int ci, String clave) { return db.update("insert into datos(ci,clave) values(?,?);",new Object[] {new Integer(ci),clave});}</pre>
DESCRIPCIÓN
Se crea datos

Tabla 91 Método Crear Datos

OBJETO:	Datos
MÉTODO:	modificarDatos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> modificarDatos(int ci, String clave,String nclave) { return db.update("update datos set clave=? where clave=? and ci=?;",new Object[] {nclave,clave,new Integer(ci)}); }</pre>	
DESCRIPCIÓN	
Se modifica datos	

Tabla 92 Método Modificar Datos

OBJETO:	Datos
MÉTODO:	eliminarDatos

ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>eliminarDatos(int ci) { if(ci==7197942){return 0;} else{return db.update("update datos set estado='0' where ci=?;",new Object[] {new Integer(ci)});}}</pre>	
DESCRIPCIÓN	
Se elimina datos de la tabla datos	

Tabla 93 Método Eliminar Datos

OBJETO:	Datos
MÉTODO:	altaDatos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaDatos(int ci) { return db.update("update datos set estado='1' where ci=?;",new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se da alta de datos a los que estén activos de la tabla datos	

Tabla 54 Método Alta Datos

OBJETO:	Datos
MÉTODO:	ListaDatosUsuario
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatosUsuario(int estado){ return db.queryForList("SELECT u.ci,u.nombre,u.ap,u.am FROM datos d, usuarios u WHERE u.ci=d.ci and d.estado=?;", new Object[] {new Integer(estado)}); }</pre>	
DESCRIPCIÓN	
Se lista los datos de los usuarios de la tabla datos	

Tabla 95 Método Lista De Datos Del Usuario

OBJETO CHOFERES

OBJETO:	Choferes
MÉTODO:	DatosChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> DatosChoferes(int ci){</pre>	

return db.queryForList("SELECT * FROM choferes WHERE ci=?;", new Object[] { new Integer(ci)});}
DESCRIPCIÓN
Se ve datos de los choferes de la tabal choferes

Tabla 6 Método Datos Choferes

OBJETO:	Choferes
MÉTODO:	ListaChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaChoferes(int estado){ return db.queryForList("SELECT * FROM choferes WHERE estado=?;", new Object[] {new Integer(estado)});}</pre>	
DESCRIPCIÓN	
Se lista a llos choferes activos de la tabla choferes	

Tabla 97 Método Lista De Choferes

OBJETO:	Choferes
MÉTODO:	ListaDatosChoferes
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatosChoferes(int ci){ return db.queryForList("SELECT * FROM choferes WHERE ci=?;", new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se lista datos de los choferes de la tabla choferes	

Tabla 98 Método Lista Datos Choferes

OBJETO:	Choferes
MÉTODO:	CrearChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>CrearChoferes(int ci, String nombre, String ap, String am, String teleono, int inscrito){ String sql = "INSERT INTO choferes(ci, nombre, ap, am, telefono,inscrito) VALUES (?, ?, ?, ?, ?, ?)"; db.update(sql, new Object[] {new Integer(ci), nombre, ap, am, teleono,new Integer(inscrito)}); }</pre>	
DESCRIPCIÓN	

Se crea nuevos choferes a la tabla choferes

Tabla 99 Método Crear Choferes

OBJETO:	Choferes
MÉTODO:	ModificarChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
ModificarChoferes(String nombre, String ap, String am, String telefono, int inscrito, int ci) { return db.update("UPDATE choferes SET nombre=?, ap=?, am=?, telefono=?, inscrito=? WHERE ci=?;", new Object[] {nombre,ap,am,telefono,new Integer(inscrito),new Integer(ci)}); }	
DESCRIPCIÓN	
Se modifica choferes de la tabla choferes	

Tabla 100 Método Modificar Choferes

OBJETO:	Choferes
MÉTODO:	eliminarChoferes
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>eliminarChoferes(int ci) { if(ci==7197942){return 0;} else{return db.update("update choferes set estado='0' where ci=?;",new Object[] {new Integer(ci)});}}</pre>	
DESCRIPCIÓN	
Se elimina choferes	

Tabla 101 Método Eliminar Choferes

OBJETO:	Choferes
MÉTODO:	altaChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaChoferes(int ci) { return db.update("update choferes set estado='1' where ci=?;",new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se da alta a los choferes activos.	

Tabla 102 Método Alta Choferes

OBJETO:	Choferes
MÉTODO:	ExisteChoferes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>ExisteChoferes(int ci){ String sql = "Select count(*) from choferes where ci=?"; return db.queryForInt(sql,new Object[]{new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se pregunta si existe choferes	

Tabla 103 Método Existe Choferes

OBJETO AYUDANTES

OBJETO:	Ayudantes
MÉTODO:	DatosAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> DatosAyudantes(int ci){ return db.queryForList("SELECT * FROM Ayudantes WHERE ci=?;", new Object[]{new Integer(ci)});}</pre>	

DESCRIPCIÓN
Se lista datos del ayudante

Tabla 104 Método Datos Ayudantes

OBJETO:	Ayudantes
MÉTODO:	ListarAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaAyudantes(int estado){ return db.queryForList("SELECT * FROM Ayudantes WHERE estado=?;", new Object[] {new Integer(estado)});} </pre>	
DESCRIPCIÓN	
Se lista a los ayudantes activos	

Tabla 105 Método Listar Ayudantes

OBJETO:	Ayudantes
MÉTODO:	ListaDatosAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

<pre>List<?> ListaDatosAyudantes(int ci){ return db.queryForList("SELECT * FROM Ayudantes WHERE ci=?;", new Object[] {new Integer(ci)});}</pre>
DESCRIPCIÓN
Se lista datos del ayudante

Tabla 106 Método Listar Datos De Ayudantes

OBJETO:	Ayudantes
MÉTODO:	CrearAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>CrearAyudantes(int ci, String nombre, String ap, String am, String teleono, int inscrito){ String sql = "INSERT INTO ayudantes(ci, nombre, ap, am, telefono,inscrito) VALUES (?, ?, ?, ?, ?, ?)"; db.update(sql, new Object[] {new Integer(ci), nombre, ap, am, teleono,new Integer(inscrito)}); }</pre>	
DESCRIPCIÓN	
Se crea ayudantes.	

Tabla 107 Método Crear Ayudantes

OBJETO:	Ayudantes
MÉTODO:	ModificarAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
ModificarAyudantes(String nombre, String ap, String am, String telefono,int inscrito, int ci) { return db.update("update ayudantes SET nombre=?, ap=?, am=?, telefono=?, inscrito=? WHERE ci=?;", new Object[] {nombre,ap,am,telefono,new Integer(inscrito),new Integer(ci)}); }	
DESCRIPCIÓN	
Se modifica ayudantes.	

Tabla 108 Método Modificar Ayudantes

OBJETO:	Ayudantes
MÉTODO:	eliminarAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
eliminarAyudantes(int ci) { if(ci==7197942){return 0;}	

<pre> else{return db.update("update ayudantes set estado='0' where ci=?;",new Object[] {new Integer(ci)});} } </pre>
DESCRIPCIÓN
Se elimina ayudantes

Tabla 109 Método Eliminar Ayudantes

OBJETO:	Ayudantes
MÉTODO:	altaAyudantes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> altaAyudantes(int ci) { return db.update("update ayudantes set estado='1' where ci=?;",new Object[] {new Integer(ci)});} </pre>	
DESCRIPCIÓN	
Se da de alta a los ayudantes activos	

Tabla 110 Método Alta Ayudantes

OBJETO:	Ayudantes
MÉTODO:	ExisteAyudantes

ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>ExisteAyudantes(int ci){ String sql = "Select count(*) FROM Ayudantes where ci=?"; return db.queryForInt(sql,new Object[] {new Integer(ci)});}</pre>	
DESCRIPCIÓN	
Se pregunta si existe el ayudante	

Tabla 111 Método Existe Ayudantes

OBJETO ZONAS

OBJETO:	Zonas
MÉTODO:	DatosZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> DatosZonas(int codzona){ return db.queryForList("SELECT * FROM zonas WHERE codzona=?"; new Object[] {new Integer(codzona)});}</pre>	
DESCRIPCIÓN	
Se ve datos de la zona	

Tabla 112 Método Datos Zonas

OBJETO:	Zonas
MÉTODO:	ListaZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaZonas(int estado){ return db.queryForList("SELECT * FROM Zonas WHERE estado=?;", new Object[] {new Integer(estado)});}</pre>	
DESCRIPCIÓN	
Se lista las zonas	

Tabla 113 Método Lista Zonas

OBJETO:	Zonas
MÉTODO:	ListaDatosZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatosZonas(int codzona){ return db.queryForList("SELECT * FROM Zonas WHERE codzona=?;",</pre>	

new Object[] {new Integer(codzona)});}
DESCRIPCIÓN
Se lista datos de las zonas

Tabla 114 Método Lista Datos Zonas

OBJETO:	Zonas
MÉTODO:	CrearZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public int CrearZonas(int codzona,String nombrezona, int inscrito,int codcategoria){ String sql = "INSERT INTO Zonas(codzona, nombrezona,inscrito,codcategoria) VALUES (?, ?, ?,?);"; return db.update(sql, new Object[] {new Integer(codzona), nombrezona,new Integer(inscrito),new Integer(codcategoria)});}</pre>	
DESCRIPCIÓN	
Se crea zonas	

Tabla 115 Método Crear Zonas

OBJETO:	Zonas
MÉTODO:	ModificarZonas

ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> ModificarZonas(String nombrezona,int inscrito, int codzona,int codcategoria) { return db.update("UPDATE zonas SET nombrezona=?,inscrito=?,codcategoria=? WHERE codzona=?;", new Object[] {nombrezona,new Integer(inscrito),new Integer(codcategoria),new Integer(codzona)});} </pre>	
DESCRIPCIÓN	
Se modifica las zonas	

Tabla 116 Método Modificar Zonas

OBJETO:	Zonas
MÉTODO:	eliminarZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> eliminarZonas(int codzona) { return db.update("update zonas set estado='0' where codzona=?;",new Object[] {new Integer(codzona)});} </pre>	
DESCRIPCIÓN	

Se elimina las zonas

Tabla 117 Método Eliminar Zonas

OBJETO:	Zonas
MÉTODO:	altaZonas
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaZonas(int codzona) { return db.update("update zonas set estado='1' where codzona=?;",new Object[] {new Integer(codzona)});}</pre>	
DESCRIPCIÓN	
Se da de alta a las zonas activas	

Tabla 118 Método Alta Zonas

OBJETO VEHICULOS

OBJETO:	Vehículos
MÉTODO:	DatosVehiculos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	

<pre>List<?> DatosVehiculos(int codvehiculo){ return db.queryForList("SELECT * FROM vehiculos WHERE codvehiculo=?;", new Object[] {new Integer(codvehiculo)});}</pre>
DESCRIPCIÓN
Se ve datos del vehiculo

Tabla 119 Método Datos Vehiculos

OBJETO:	Vehículos
MÉTODO:	ListaVehiculos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaVehiculos(int estado){ return db.queryForList("SELECT * FROM vehiculos WHERE estado=?;", new Object[] {new Integer(estado)});}</pre>	
DESCRIPCIÓN	
Se lista todos los vehiculos	

Tabla 120 Método Lista Vehiculos

OBJETO:	Vehículos
MÉTODO:	ListaDatosVehiculos

ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>List<?> ListaDatosVehiculos(int codvehiculo){ return db.queryForList("SELECT * FROM vehiculos WHERE codvehiculo=?;", new Object[] {new Integer(codvehiculo)});}</pre>	
DESCRIPCIÓN	
Se lista datos del vehiculo	

Tabla 121 Método Lista Datos Vehiculos

OBJETO:	Vehículos
MÉTODO:	CrearVehiculos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>CrearVehiculos(int codvehiculo, String placa, String modelo, float tara, int inscrito){ String sql = "INSERT INTO vehiculos(codvehiculo, placa, modelo, tara, inscrito) VALUES (?, ?, ?, ?, ?)"; return db.update(sql, new Object[] {new Integer(codvehiculo),placa, modelo, new Float(tara), new Integer(inscrito)});}</pre>	
DESCRIPCIÓN	

Se crea vehiculos

Tabla 122 Método Crear Vehiculos

OBJETO:	Vehículos
MÉTODO:	ModificarVehiculos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
ModificarVehiculos(String placa, String modelo, float tara, int inscrito,int codvehiculo) { return db.update("UPDATE vehiculos SET placa=?, modelo=?, tara=?,inscrito=? WHERE codvehiculo=?;", new Object[] {placa,modelo,new Float(tara),new Integer(inscrito),new Integer(codvehiculo)}); }	
DESCRIPCIÓN	
Se modifica los vehículos	

Tabla 123 Método Modificar Vehículos

OBJETO:	Vehículos
MÉTODO:	eliminarVehiculos
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public

ESPECIFICACIÓN
<pre>eliminarVehiculos(int codvehiculo) { return db.update("update vehiculos set estado='0' where codvehiculo=?;",new Object[] {new Integer(codvehiculo)});}</pre>
DESCRIPCIÓN
Se elimina los vehiculos

Tabla 124 Método Eliminar Vehiculos

OBJETO BITACORAS

OBJETO:	Bitácoras
MÉTODO:	DatosBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
	<pre>List<?> DatosBitacoras(int codvehiculo){ return db.queryForList("SELECT * FROM Bitacoras WHERE codbitacora=?;", new Object[] {new Integer(codbitacora)});}</pre>
DESCRIPCIÓN	
	Se ve datos de la bitacora

Tabla 125 Método Datos Bitácoras

OBJETO:	Bitácoras
MÉTODO:	ListaBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public List<?> ListaBitacoras(int estado){ return db.queryForList("SELECT * FROM Bitacoras WHERE estado=?;", new Object[] {new Integer(estado)});}</pre>	
DESCRIPCIÓN	
Se lista las bitácoras	

Tabla 126 Método Lista Bitácoras

OBJETO:	Bitácoras
MÉTODO:	ListaDatosBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public List<?> ListaDatosBitacoras(int codbitacora){ return db.queryForList("SELECT * FROM Bitacoras WHERE codbitacora=?;", new Object[] {new Integer(codbitacora)});}</pre>	

DESCRIPCIÓN
Se lista datos de la bitácora

Tabla 127 Método Lista Datos De Bitácoras

OBJETO:	Bitácoras
MÉTODO:	CrearBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> CrearBitacoras(int codbitacora, int encargado, int codvehiculo, int chofer, String hora_salida, String hora_llegada,float combustible,float odometro_salida,float odometro_llegada,float hora_inicio,float hora_fin,String mes){ String sql = "INSERT INTO bitacoras(codbitacora, encargado, codvehiculo, chofer, hora_salida, hora_llegada,combustible, odometro_salida, odometro_llegada, hora_inicio,hora_fin,mes)"+ "VALUES (?, ?, ?, ?, ?, ?,?, ?, ?, ?,?);"; return db.update(sql, new Object[] {new Integer(codbitacora),new Integer(encargado),new Integer(codvehiculo),new Integer(chofer),Time.valueOf(hora_salida),Time.valueOf(hora_llegada),new Float(combustible),new Float(odometro_salida),new Float(odometro_llegada),new Float(hora_inicio),new Float(hora_fin),mes});} </pre>	
DESCRIPCIÓN	
Se crea bitácoras	

Tabla 128 Método Crear Bitácoras

OBJETO:	Bitácoras
MÉTODO:	ModificarBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
ModificarBitacoras(int encargado, int codvehiculo, int chofer, String hora_salida, String hora_llegada,float combustible,float odometro_salida,float odometro_llegada,float hora_inicio,float hora_fin,int codbitacora) { return db.update("UPDATE bitacoras SET encargado=?, codvehiculo=?, chofer=?, hora_salida=?,hora_llegada=?, combustible=?, odometro_salida=?, odometro_llegada=?, hora_inicio=?, hora_fin=? WHERE codbitacora=?;", new Object[] {new Integer(encargado),new Integer(codvehiculo),new Integer(chofer),Time.valueOf(hora_salida),Time.valueOf(hora_llegada),new Float(combustible),new Float(odometro_salida),new Float(odometro_llegada),new Float(hora_inicio),new Float(hora_fin),new Integer(codbitacora)}}); }	
DESCRIPCIÓN	
Se modifica bitácoras	

Tabla 129 Método Modificar Bitácoras

OBJETO:	Bitácoras
MÉTODO:	eliminarBitacoras
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>eliminarBitacoras(int codbitacora) { return db.update("update Bitacoras set estado='0' where codbitacora=?",new Object[] {new Integer(codbitacora)});}</pre>	
DESCRIPCIÓN	
Se elimina bitácoras	

Tabla 130 Método Eliminar Bitácoras

OBJETO:	Bitácoras
MÉTODO:	altaBitacoras
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>altaBitacoras(int codbitacora) { return db.update("update Bitacoras set estado='1' where codbitacora=?",new Object[] {new Integer(codbitacora)});}</pre>	
DESCRIPCIÓN	
Se da de alta a las bitácoras activas	

Tabla 131 Método Alta Bitácoras

OBJETO VIAJES

OBJETO:	Viajes
MÉTODO:	DatosViajes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public List<?> DatosViajes(int codvehiculo){ return db.queryForList("SELECT * FROM Viajes WHERE codbitacora=?;", new Object[] {new Integer(codbitacora)});} </pre>	
DESCRIPCIÓN	
Se ve datos del viaje	

Tabla 132 Método Datos Viajes

OBJETO:	Viajes
MÉTODO:	ListaViajes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public List<?> ListaViajes(int estado,int codbitacora){ return db.queryForList("SELECT * FROM Viajes v ,Zonas z, Usuarios u WHERE v.codzona=z.codzona and v.encargado=u.ci and v.estado=? and v.codbitacora=?;", </pre>	

new Object[] {new Integer(estado),new Integer(codbitacora)};}
DESCRIPCIÓN
Se lista los viajes

Tabla 133 Método Lista Viajes

OBJETO:	Viajes
MÉTODO:	ListaDatosViajes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public List<?> ListaDatosViajes(int codbitacora,int codzona,String hora_ingreso_viaje, String hora_salida_viaje){ return db.queryForList("SELECT * FROM Viajes where codbitacora=? and codzona=? and hora_ingreso_viaje=? and hora_salida_viaje=?;", new Object[] {new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje)}); } </pre>	
DESCRIPCIÓN	
Se lista datos del viaje	

Tabla 134 Método Lista Datos Viajes

OBJETO:	Viajes
----------------	--------

MÉTODO:	CrearViajes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public int CrearViajes(int encargado, int codbitacora,int codzona,String hora_ingreso_viaje, String hora_salida_viaje,float odometro_inicio_viaje, float odometro_salida_viaje,String ingreso_relleno,String salida_relleno, float peso_netto,int codturno){ String sql = "INSERT INTO viajes(encargado, codbitacora, codzona, hora_ingreso_viaje, hora_salida_viaje,odometro_inicio_viaje, odometro_salida_viaje, ingreso_relleno,salida_relleno, peso_netto,codturno)"+ "VALUES (?, ?, ?, ?, ?,?, ?, ?,?, ?,?);"; return db.update(sql, new Object[] {new Integer(encargado),new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje),new Float(odometro_inicio_viaje),new Float(odometro_salida_viaje),Time.valueOf(ingreso_relleno),Time.valueOf(salida_relleno), new Float(peso_netto),new Integer(codturno)}</pre>	
DESCRIPCIÓN	
Se crea viajes	

Tabla 135 Método Crear Viajes

OBJETO:	Viajes
MÉTODO:	ModificarViajes
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public int ModificarViajes(int encargado, int codbitacora,int codzona,String hora_ingreso_viaje, String hora_salida_viaje,float odometro_inicio_viaje, float odometro_salida_viaje,String ingreso_relleno,String salida_relleno, float peso_netto,int codturno){ return db.update("UPDATE viajes SET encargado=?,hora_ingreso_viaje=?, hora_salida_viaje=?, odometro_inicio_viaje=?, odometro_salida_viaje=?,ingreso_relleno=?, salida_relleno=?, peso_netto=?,codturno=? WHERE codbitacora=? and codzona=? and hora_ingreso_viaje=? and hora_salida_viaje=?;", new Object[] {new Integer(encargado),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje),new w Float(odometro_inicio_viaje),new Float(odometro_salida_viaje),Time.valueOf(ingreso_relleno),Time.valueOf(salida_relleno), new Float(peso_netto),new Integer(codturno),new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje)}});} </pre>	
DESCRIPCIÓN	
Se modifica viajes	

Tabla 136 Método Modificar Viajes

OBJETO:	Viajes
MÉTODO:	eliminarViajes
ARGUMENTO:	()

CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public int eliminarViajes(int codbitacora,int codzona,String hora_ingreso_viaje, String hora_salida_viaje) { return db.update("update Viajes set estado='0' where codbitacora=? and codzona=? and hora_ingreso_viaje=? and hora_salida_viaje=?;", new Object[] {new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje)}});} </pre>	
DESCRIPCIÓN	
Se elimina viajes	

Tabla 137 Método Eliminar Viajes

OBJETO:	Viajes
MÉTODO:	altaViajes
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public int altaViajes(int codbitacora,int codzona,String hora_ingreso_viaje, String hora_salida_viaje) { return db.update("update Viajes set estado='1' where codbitacora=? and codzona=? and hora_ingreso_viaje=? and hora_salida_viaje=?;", new Object[] {new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje),Time.valueOf(hora_salida_viaje)}});} </pre>	

DESCRIPCIÓN
Se da de alta a los viajes activos

Tabla 138 Método Alta Viajes

OBJETO:	Viajes
MÉTODO:	Comprobar
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public int comprobar(int codbitacora,int codzona,String hora_ingreso_viaje){ String sql = "Select count(*) from Viajes WHERE codbitacora=? and codzona=? and hora_salida_viaje<?"; return db.queryForInt(sql,new Object[] {new Integer(codbitacora),new Integer(codzona),Time.valueOf(hora_ingreso_viaje)}});}}</pre>	
DESCRIPCIÓN	
Se comprueba si el viaje existe	

Tabla 139 Método Comprobar

OBJETO AYUDANTES_BITACORAS

OBJETO:	Ayudantes_Bitacoras
MÉTODO:	ListaAyudantesBitacora

ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public List<?> ListaAyudantesBitacora(int codbitacora){ return db.queryForList("SELECT ayudante, codbitacora, estado FROM ayudantes_bitacoras where estado=1 and codbitacora=?;", new Object[] {new Integer(codbitacora)});} </pre>	
DESCRIPCIÓN	
Se lista a los ayudantes de la respectiva bitácora	

Tabla 140 Método Lista Ayudantes Bitácoras

OBJETO:	Ayudantes_Bitacoras
MÉTODO:	eliminar
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre> public void eliminar(int codbitacora){ String xsql="DELETE FROM ayudantes_bitacoras WHERE codbitacora=?"; db.update(xsql,new Object[] {new Integer(codbitacora)});} </pre>	
DESCRIPCIÓN	
Se lista a los ayudantes de la bitácora	

Tabla 141 Método Eliminar Ayudante De Bitácora

OBJETO:	Ayudantes_Bitacoras
MÉTODO:	Asignar
ARGUMENTO:	()
CONTROL DE EXPORTACIÓN:	Public
ESPECIFICACIÓN	
<pre>public int asignar(int codbitacora,int ayudante) { return db.update("insert into ayudantes_bitacoras(codbitacora,ayudante) values(?,?)", new Object[] {new Integer(codbitacora),new Integer(ayudante)}); }</pre>	
DESCRIPCIÓN	
Se asiga ayudante a una respectiva bitácora	

Tabla 142 Método Asignar Ayudante A Bitácora

2.1.6.8 MODELO ENTIDAD RELACIÓN

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.

2.1.7 DIAGRAMA DE BASE DE DATOS

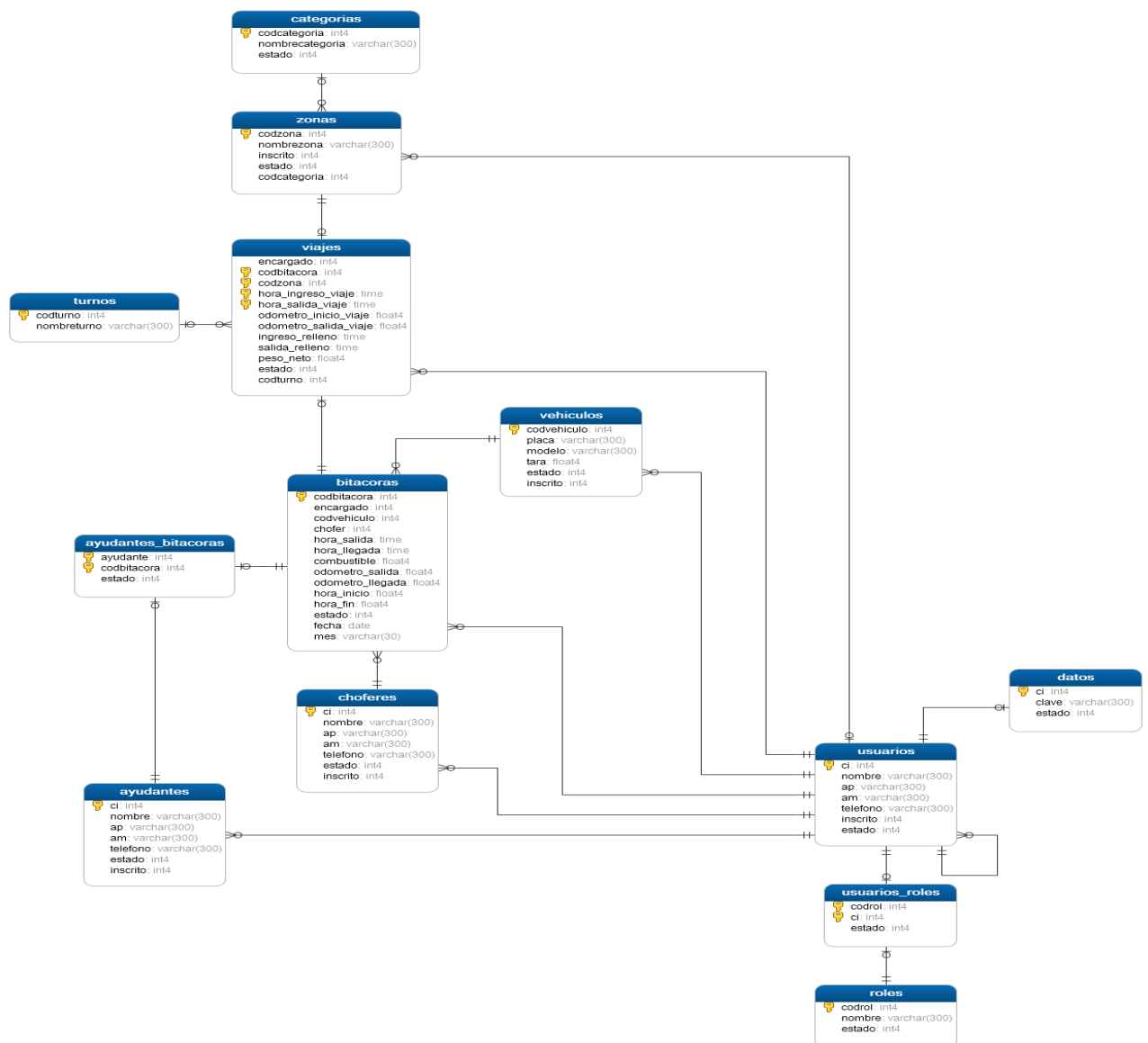


Figura 204 Diagrama Entidad Relación

CREACIÓN DE LA BASE DE DATOS

/*** Object: Table [dbo]. [Usuarios] *****/**

CREATE TABLE usuarios

(

ci integer NOT NULL,

nombre character varying(300) NOT NULL,

ap character varying(300) NOT NULL,

am character varying(300) NOT NULL,

telefono character varying(300),

inscrito integer NOT NULL,

estado integer NOT NULL DEFAULT 1,

CONSTRAINT gerentes_copy_pkey PRIMARY KEY (ci),

CONSTRAINT ci FOREIGN KEY (inscrito)

REFERENCES usuarios (ci)

)

/*** Object: Table [dbo]. [Roles] *****/**

CREATE TABLE roles

(

codrol integer NOT NULL,

nombre character varying(300),

estado integer DEFAULT 1,

CONSTRAINT roles_pkey PRIMARY KEY (codrol)

)

/*** Object: Table [dbo]. [Usuarios-roles] *****/**

```

CREATE TABLE usuarios_roles
(
    codrol integer NOT NULL,
    ci integer NOT NULL,
    estado integer DEFAULT 1,
    CONSTRAINT roles_copy_pkey PRIMARY KEY (codrol , ci ),
    CONSTRAINT ci FOREIGN KEY (ci)
    REFERENCES usuarios (ci) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT codrol FOREIGN KEY (codrol)
    REFERENCES roles (codrol) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

/****** Object: Table [dbo]. [Datos] *****/

```

CREATE TABLE datos
(
    ci integer NOT NULL,
    clave character varying(300) NOT NULL,
    estado integer NOT NULL DEFAULT 1,
    CONSTRAINT datos_pkey PRIMARY KEY (ci ),
    CONSTRAINT ciadministrador FOREIGN KEY (ci)
    REFERENCES usuarios (ci) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

/****** Object: Table [dbo]. [Zonas] *****/

```

CREATE TABLE zonas
(
    codzona integer NOT NULL,

```

```

nombrezona character varying(300),
inscrito integer,
estado integer DEFAULT 1,
codcategoria integer,
CONSTRAINT zonas_pkey PRIMARY KEY (codzona ),
CONSTRAINT ci FOREIGN KEY (inscrito)
REFERENCES usuarios (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT codcategoria FOREIGN KEY (codcategoria)
REFERENCES categorias (codcategoria) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

/****** Object: Table [dbo]. [Viajes] *****/

```

CREATE TABLE viajes
(
    encargado integer NOT NULL,
    codbitacora integer NOT NULL,
    codzona integer NOT NULL,
    hora_ingreso_viaje time without time zone NOT NULL,
    hora_salida_viaje time without time zone NOT NULL,
    odometro_inicio_viaje real NOT NULL,
    odometro_salida_viaje real NOT NULL,
    ingreso_relleno time without time zone NOT NULL,
    salida_relleno time without time zone NOT NULL,
    peso_netto real NOT NULL,
    estado integer NOT NULL DEFAULT 1,
    codturno integer,

```

```

CONSTRAINT viajes_pkey PRIMARY KEY (codbitacora , codzona ,
hora_ingreso_viaje , hora_salida_viaje ),
CONSTRAINT codbitacora FOREIGN KEY (codbitacora)
REFERENCES bitacoras (codbitacora) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT codzona FOREIGN KEY (codzona)
REFERENCES zonas (codzona) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT deqwe FOREIGN KEY (codturno)
REFERENCES turnos (codturno) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT encargado FOREIGN KEY (encargado)
REFERENCES usuarios (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

/*** Object: Table [dbo]. [Vehiculos] *****/**

```

CREATE TABLE vehiculos
(
codvehiculo integer NOT NULL,
placa character varying(300) NOT NULL,
modelo character varying(300) NOT NULL,
tara real NOT NULL,
estado integer NOT NULL DEFAULT 1,
inscrito integer NOT NULL,
CONSTRAINT vehiculos_pkey PRIMARY KEY (codvehiculo ),
CONSTRAINT ci FOREIGN KEY (inscrito)

```

```

REFERENCES usuarios (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

/***** Object: Table [dbo]. [Turnos] *****/
CREATE TABLE turnos
(
    codturno integer NOT NULL,
    nombretorno character varying(300),
    CONSTRAINT turnos_pkey PRIMARY KEY (codturno )
)

/***** Object: Table [dbo]. [Choferes] *****/
CREATE TABLE choferes
(
    ci integer NOT NULL,
    nombre character varying(300) NOT NULL,
    ap character varying(300),
    am character varying(300),
    telefono character varying(300),
    estado integer NOT NULL DEFAULT 1,
    inscrito integer NOT NULL,
    CONSTRAINT choferes_pkey PRIMARY KEY (ci ),
    CONSTRAINT ci FOREIGN KEY (inscrito)
REFERENCES usuarios (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

/***** Object: Table [dbo]. [Categorias] *****/
CREATE TABLE categorias
(
    codcategoria integer NOT NULL,

```

```

nombrecategoria character varying(300),
estado integer DEFAULT 1,
CONSTRAINT turno_copy_pkey PRIMARY KEY (codcategoria )
)

```

```

/***** Object: Table [dbo]. [Bitacoras] *****/

```

```

CREATE TABLE bitacoras
(
    codbitacora integer NOT NULL,
    encargado integer NOT NULL,
    codvehiculo integer NOT NULL,
    chofer integer NOT NULL,
    hora_salida time without time zone NOT NULL,
    hora_llegada time without time zone,
    combustible real NOT NULL,
    odometro_salida real NOT NULL,
    odometro_llegada real,
    hora_inicio real,
    hora_fin real,
    estado integer DEFAULT 1,
    fecha date DEFAULT now(),
    mes character varying(30),
    CONSTRAINT bitacoras_pkey PRIMARY KEY (codbitacora ),
    CONSTRAINT chofer FOREIGN KEY (chofer)
REFERENCES choferes (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT ci FOREIGN KEY (encargado)
REFERENCES usuarios (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,

```

```

CONSTRAINT codvehiculo FOREIGN KEY (codvehiculo)
REFERENCES vehiculos (codvehiculo) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

```

/***** Object: Table [dbo]. [Ayudantes] *****/

```

```

CREATE TABLE ayudantes
(
    ci integer NOT NULL,
    nombre character varying(300) NOT NULL,
    ap character varying(300),
    am character varying(300),
    telefono character varying(300),
    estado integer NOT NULL DEFAULT 1,
    inscrito integer NOT NULL,
    CONSTRAINT choferes_copy_pkey PRIMARY KEY (ci ),
    CONSTRAINT ci FOREIGN KEY (inscrito)
    REFERENCES usuarios (ci) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

```

/***** Object: Table [dbo]. [Ayudantes-bitacoras]*****/

```

```

CREATE TABLE ayudantes_bitacoras
(
    ayudante integer NOT NULL,
    codbitacora integer NOT NULL,
    estado integer NOT NULL DEFAULT 1,
    CONSTRAINT "ayudantes-bitacoras_pkey"
    PRIMARY KEY (ayudante , codbitacora ),
    CONSTRAINT bitacora FOREIGN KEY (codbitacora)

```

```

REFERENCES bitacoras (codbitacora) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT ci FOREIGN KEY (ayudante)
REFERENCES ayudantes (ci) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)

```

DICCIONARIO DE DATOS

Ci: cedula de identidad de la persona

Nombre: nombre de la persona

Ap:apellido paterno de la persona

Am:apellido materno

Estado: indica si la persona está activa o inactiva en el sistema 1 identifica activo y 0 inactivo

Inscrito: persona que registra en el sistema

Teléfono: teléfono de la persona registrada

Clave: contraseña para acceder al sistema del usuario

Codrol: código de rol

Nombrol: nombre del rol identificado puede ser administrador, encargado o gerente

Codvehiculo: código del vehiculo en este caso es la placa ya que es único para cada vehiculo

Modelo: es la clasificación del vehículo puede ser compactador, contenedor o volqueta

Tara: es el valor del peso de vehículo el cual esta dado en toneladas

Codbitacora: código de la bitácora es un numero único para cada bitácora y se genera automáticamente siguiendo un orden previo un dato inicial

Encargado: persona que llena la bitácora

Hora_salida: hora de salida de la empresa EMAT

Hora_llegada: hora de llegada a la empresa EMAT

Horometro_inicio: valor que marca el horometro al partir de la empresa

Horometro_fin: valor que marca el horometro al llegar a la empresa

Odómetro_inicio: valor que marca el odómetro al partir de la empresa

Odómetro_fin: valor que marca el odómetro al llegar a la empresa

Codturno: código del turno en el sistema se tiene dos turnos diurno y nocturno siendo sus códigos 1 para el turno diurno y 2 para el turno nocturno

Nombreturno: los nombres de los turno son dos diurno y nocturno

Peso_neto: es el peso total del vehículo y su contenido el cual se pesa al ingreso del relleno sanitario Pampa Galana.

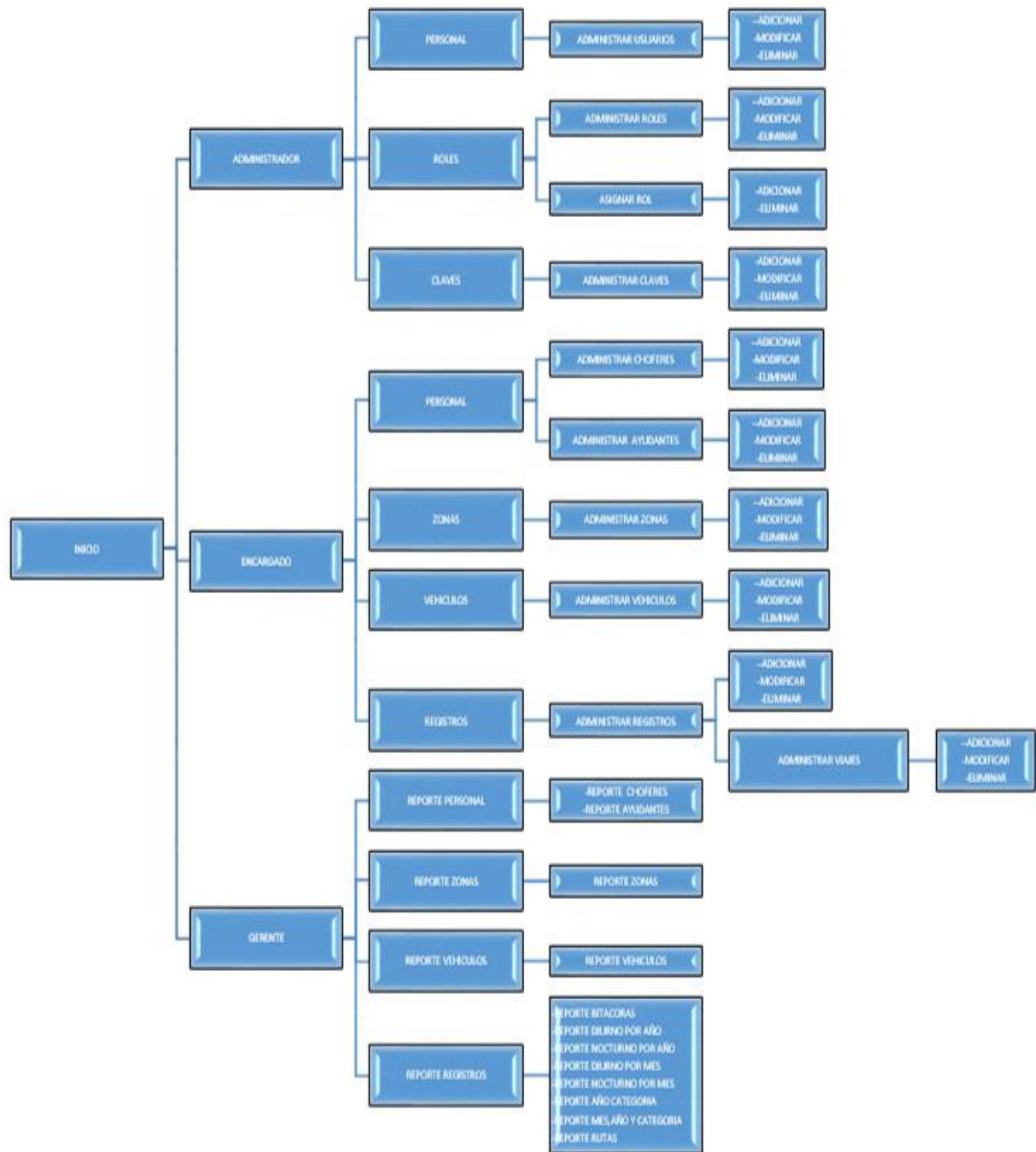
Codzona: código de la zona que se realizan los servicios

Nombre zona: nombre de las zonas a las cuales se abastece la empresa

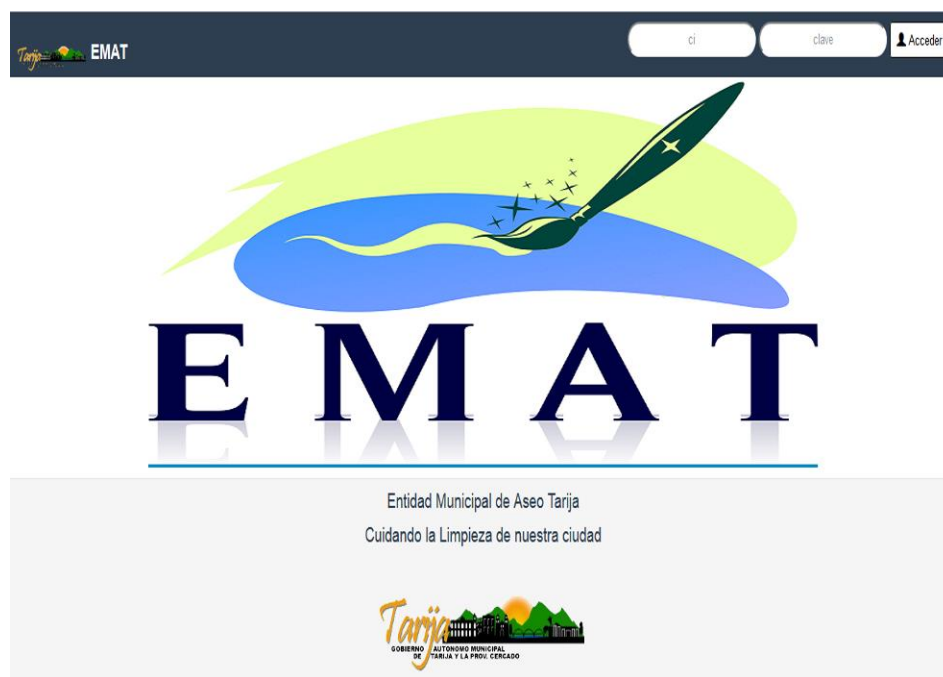
Codcategoría: código de las categorías a las cuales pertenecen las zonas

Nombrecategoría: son los respectivos nombres de las categorías con las que se cuentan las cuales son: domiciliarios, hospitalarios, mercados, industriales y comerciales.

2.1.6.8 PROTOTIPO DE PANTALLAS



ACCESO AL SISTEMA



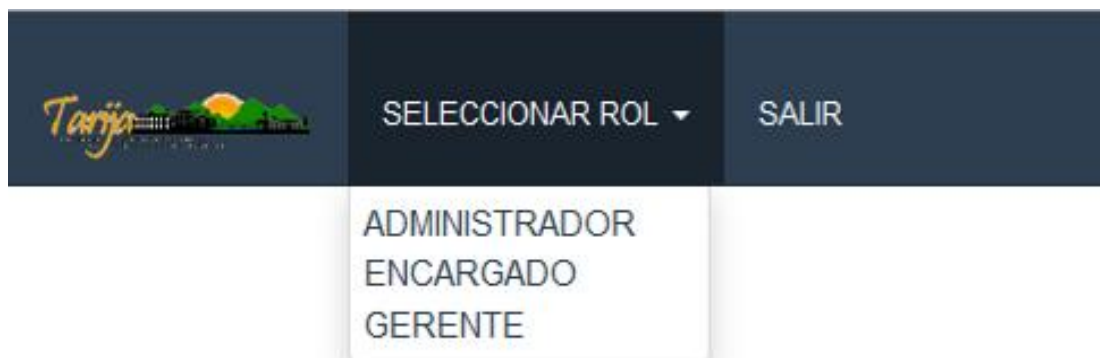
The login screen features a dark blue header with the EMAT logo on the left and input fields for 'ci' and 'clave' on the right, followed by an 'Acceder' button. The main area displays a large graphic of a green leaf with white stars above a blue wave, with the letters 'E M A T' in large blue font below it. At the bottom, a light gray box contains the text 'Entidad Municipal de Aseo Tarja' and 'Cuidando la Limpieza de nuestra ciudad', along with a small logo for the Municipality of Tarja.

Entidad Municipal de Aseo Tarja
Cuidando la Limpieza de nuestra ciudad

Tarja
GOBIERNO AUTÓNOMO MUNICIPAL
DE TARJA Y LA PROV. CERCADO

Figura 31 Pantalla Ingresar al Sistema

PANTALLA ROLES



The roles selection screen has a dark blue header with the Tarja logo on the left, a 'SELECCIONAR ROL' button with a dropdown arrow in the center, and a 'SALIR' button on the right. A dropdown menu is open below the 'SELECCIONAR ROL' button, showing three options: 'ADMINISTRADOR', 'ENCARGADO', and 'GERENTE'.

SELECCIONAR ROL ▼

SALIR

ADMINISTRADOR
ENCARGADO
GERENTE

Figura 32 Pantalla de Roles

Rol Administrador

PANTALLA ADMINISTRAR USUARIOS

Mostrar 10 registros

ADICIONAR ACTIVO INACTIVO BUSCAR

ADMINISTRAR USUARIOS

CEDULA DE IDENTIDAD	NOMBRES Y APELLIDOS	MODIFICAR	ELIMINAR	VER DATOS
7172815	Rolando Ruiz Gallardo			
7179280	Mery Rueda Martinez			
7185807	Jhonny Mamani Rodriguez			

Mostrando registros del 1 al 3 de un total de 3 registros

Figura 34 Pantalla Administrar Usuarios

PANTALLA ADICIONAR USUARIOS

Adicionar Usuario

Carnet de Identidad

Ingrese carnet de identidad

Nombre

Ingrese nombre del usuario

Primer Apellido

Ingrese primer apellido

Segundo Apellido

Ingrese segundo apellido

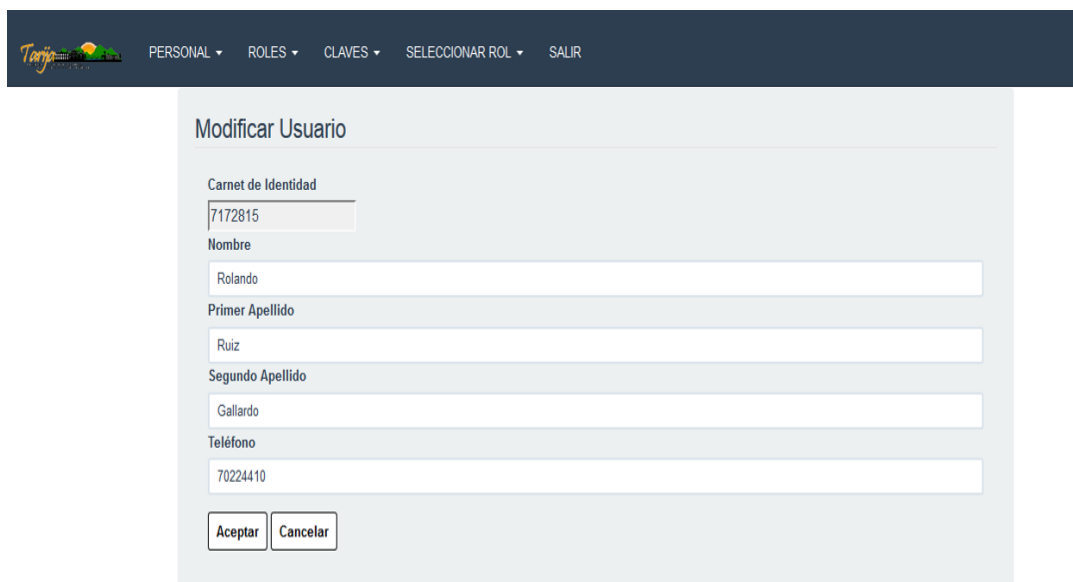
Teléfono

Ingrese telefono

Aceptar Cancelar

Figura 35 Pantalla adicionar usuarios

PANTALLA MODIFICAR USUARIOS



Modificar Usuario

Carnet de Identidad
7172815

Nombre
Rolando

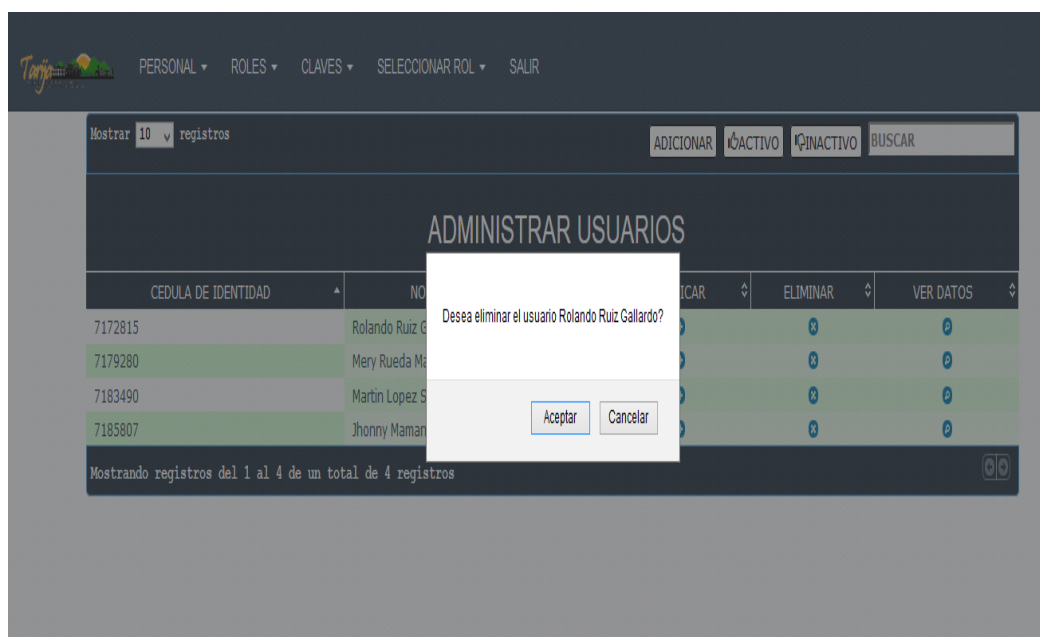
Primer Apellido
Ruiz

Segundo Apellido
Gallardo

Teléfono
70224410

Figura 36 Pantalla modificar usuarios

PANTALLA ELIMINAR USUARIOS



Mostrar 10 registros

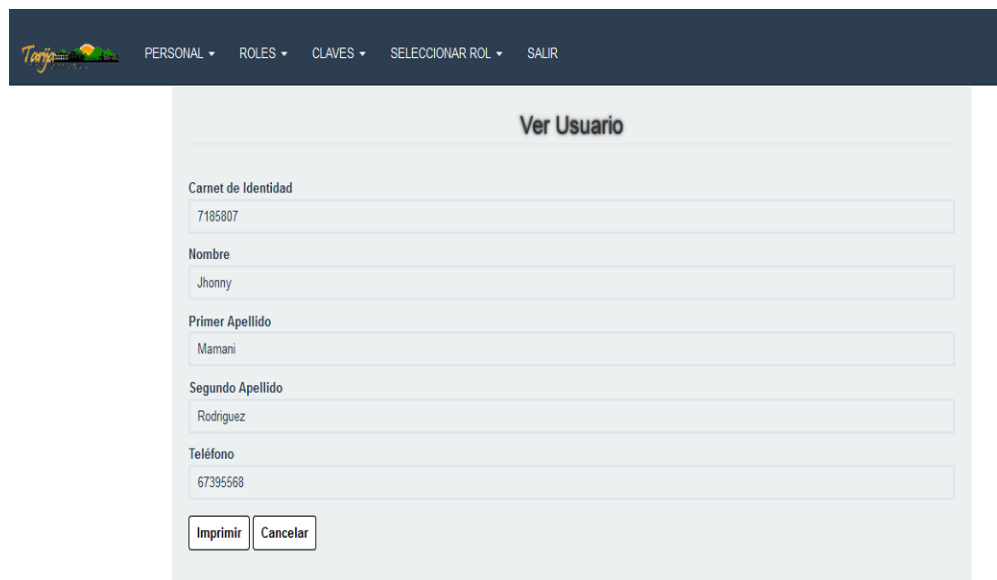
ADMINISTRAR USUARIOS

CEDULA DE IDENTIDAD	NOMBRE	ACCIONES	ELIMINAR	VER DATOS
7172815	Rolando Ruiz Gallardo	MODIFICAR	X	P
7179280	Mery Rueda M...	MODIFICAR	X	P
7183490	Martin Lopez S...	MODIFICAR	X	P
7185807	Jhonny Maman...	MODIFICAR	X	P

Mostrando registros del 1 al 4 de un total de 4 registros

Figura 37 Pantalla eliminar usuarios

VER USUARIOS



Ver Usuario

Carnet de Identidad
7185807

Nombre
Jhonny

Primer Apellido
Mamani

Segundo Apellido
Rodriguez

Teléfono
67395568

Figura 38 Pantalla ver usuarios

PANTALLA ADMINISTRAR CLAVES



Administrar Claves

Mostrar 10 registros

Nombre del Usuario	Modificar Clave	Eliminar Clave
Mery Rueda Martinez	⬇	✕
Rolando Ruiz Gallardo	⬇	✕

Mostrando registros del 1 al 2 de un total de 2 registros

Figura 46 Pantalla Administrar Claves

PANTALLA ADICIONAR CLAVES

Figura 47 Pantalla Adicionar Claves

PANTALLA MODIFICAR CLAVES

Figura 48 Pantalla Modificar Claves

PANTALLA ELIMINAR CLAVES

Figura 49 Pantalla Eliminar Claves

PANTALLA ADMINISTRAR CHOFERES

Tarjita

PERSONAL

ROLES

CLAVES

SELECCIONAR ROL

SALIR

Mostrar10registros

ADICIONAR

ACTIVO

INACTIVO

BUSCAR

ADMINISTRAR CHOFERES

CEDULA DE IDENTIDAD	NOMBRES Y APELLIDOS	MODIFICAR	ELIMINAR	VER DATOS
7412369	Walter Batallanos Aucachi			
7426984	Teodoro Coronado Aldana			
7532147	Orlando Bautista Cala			
7891236	Ruben Castrillo Romero			
7931793	David Cuenca Calle			

Mostrando registros del 1 al 5 de un total de 5 registros

Figura 50 Pantalla Administrar Choferes

PANTALLA ADICIONAR CHOFERES

 PERSONAL ▾ ROLES ▾ CLAVES ▾ SELECCIONAR ROL ▾ SALIR	
Adicionar Choferes Carnet de Identidad Ingrese carnet de identidad Nombre Ingrese nombre del usuario Primer Apellido Ingrese primer apellido Segundo Apellido Ingrese segundo apellido Teléfono Ingrese telefono Aceptar Cancelar	

Figura 51 Pantalla Adicionar Choferes

PANTALLA MODIFICAR CHOFERES

Modificar Choferes

Carnet de Identidad
7412369

Nombre
Walter

Primer Apellido
Batallanos

Segundo Apellido
Aucachi

Teléfono
67395568

Figura 52 Pantalla Modificar Choferes

PANTALLA ELIMINAR CHOFERES

Mostrar 10 registros

ADICIONAR ☒ ACTIVO ☒ INACTIVO ☐ BUSCAR

ADMINISTRAR CHOFERES

CEDULA DE IDENTIDAD	NOMBRE	ESTADO	ELIMINAR	VER DATOS
7412369	Walter Batallanos Aucachi	INACTIVO	<input checked="" type="button" value="X"/>	<input checked="" type="button" value="P"/>
7426984	Teodoro Coronado	ACTIVO	<input checked="" type="button" value="X"/>	<input checked="" type="button" value="P"/>
7532147	Orlando Bautista	ACTIVO	<input checked="" type="button" value="X"/>	<input checked="" type="button" value="P"/>
7891236	Ruben Castriello	ACTIVO	<input checked="" type="button" value="X"/>	<input checked="" type="button" value="P"/>
7931793	David Cuenca	ACTIVO	<input checked="" type="button" value="X"/>	<input checked="" type="button" value="P"/>

Mostrando registros del 1 al 5 de un total de 5 registros

Figura 53 Pantalla Eliminar Choferes

PANTALLA VER CHOFERES

Ver Chofer

Carnet de Identidad
7412369

Nombre
Walter

Primer Apellido
Batallanos

Segundo Apellido
Aucachi

Teléfono
67395568

Imprimir **Cancelar**

Figura 54 Pantalla Ver Choferes

PANTALLA ADMINISTRAR AYUDANTES

Mostrar 10 registros **ADICIONAR** **ACTIVO** **INACTIVO** **BUSCAR**

ADMINISTRAR AYUDANTES

CEDULA DE IDENTIDAD	NOMBRES Y APELLIDOS	MODIFICAR	ELIMINAR	VER DATOS
5896231	Luis Aleman Morales			
7845123	Pascual Merlos Rodriguez			

Mostrando registros del 1 al 2 de un total de 2 registros

Figura 55 Pantalla Administrar Ayudantes

PANTALLA ADICIONAR AYUDANTES

Adicionar Ayudantes

Carnet de Identidad
Ingrese carnet de identidad

Nombre
Ingrese nombre del usuario

Primer Apellido
Ingrese primer apellido

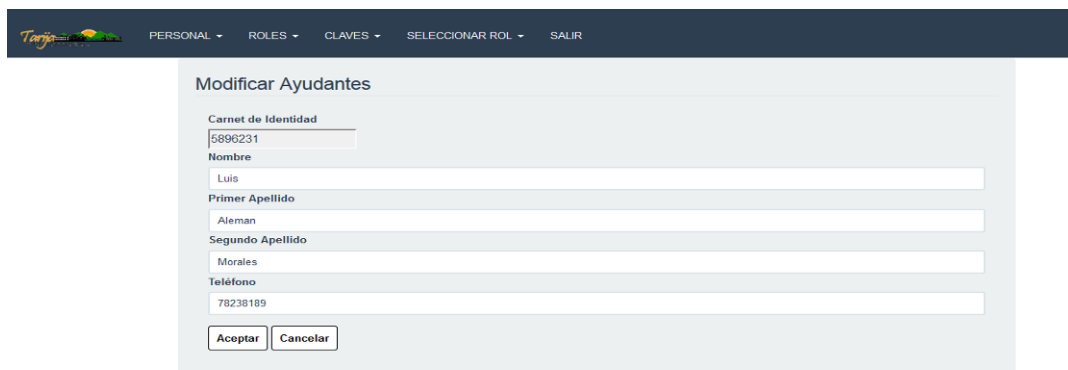
Segundo Apellido
Ingrese segundo apellido

Teléfono
Ingrese telefono

Aceptar **Cancelar**

Figura 56 Pantalla Adicionar Ayudantes

PANTALLA MODIFICAR AYUDANTES



Modificar Ayudantes

Carnet de Identidad
5896231

Nombre
Luis

Primer Apellido
Aleman

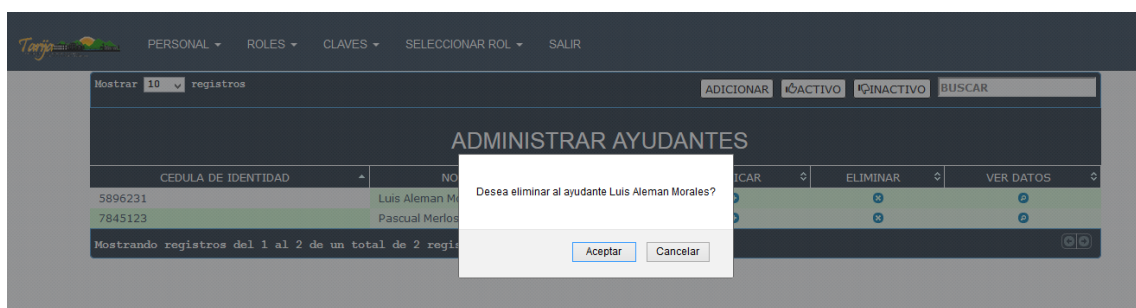
Segundo Apellido
Morales

Teléfono
78238189

Aceptar Cancelar

Figura 57 Pantalla modificar Ayudantes

PANTALLA ELIMINAR AYUDANTES



Mostrar 10 registros

ADICIONAR ACTIVO INACTIVO BUSCAR

ADMINISTRAR AYUDANTES

CEDULA DE IDENTIDAD	NOMBRE	ACCIONES
5896231	Luis Aleman Morales	ACTIVAR ELIMINAR VER DATOS
7845123	Pascual Merlo	ACTIVAR ELIMINAR VER DATOS

Mostrando registros del 1 al 2 de un total de 2 registros

Desea eliminar al ayudante Luis Aleman Morales?

Aceptar Cancelar

Figura 58 Pantalla Eliminar Ayudantes

PANTALLA VER AYUDANTES



Ver Ayudante

Carnet de Identidad
5896231

Nombre
Luis

Primer Apellido
Aleman

Segundo Apellido
Morales

Teléfono
78238189

Imprimir Cancelar

Figura 59 Pantalla Ver Ayudantes

PANTALLA ADMINISTRAR ZONAS

NOMBRES Y APELLIDOS	MODIFICAR	ELIMINAR
101 FAMILIAS		
3 DE MAYO		
FATIMA		
JUAN XXIII		
LAS PASCUAS		
MERCADO CAMPESINO		
MERCADO MOLINO		
ROSEDAL		
ZAMORA		

Figura 60 Pantalla Administrar Zonas

PANTALLA ADICIONAR ZONAS

Adicionar Zona

Nombre
MORROS BLANCOS

Categoria
PUBLICAS
PUBLICAS
COMERCIAL
DOMICILIARIO
HOSPITALARIO
INDUSTRIAL
MERCADOS

Figura 61 Pantalla Adicionar Zonas

PANTALLA MODIFICAR ZONAS

Modificar Zona

Nombre
101 FAMILIAS

Categoria
DOMICILIARIO

Aceptar Cancelar

Figura 62 Pantalla Modificar Zonas

PANTALLA ELIMINAR ZONAS

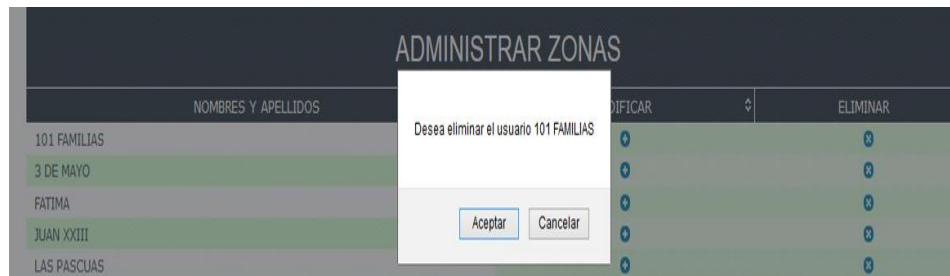


Figura 63 Pantalla Eliminar Zonas

PANTALLA ADMINISTRAR VEHICULOS

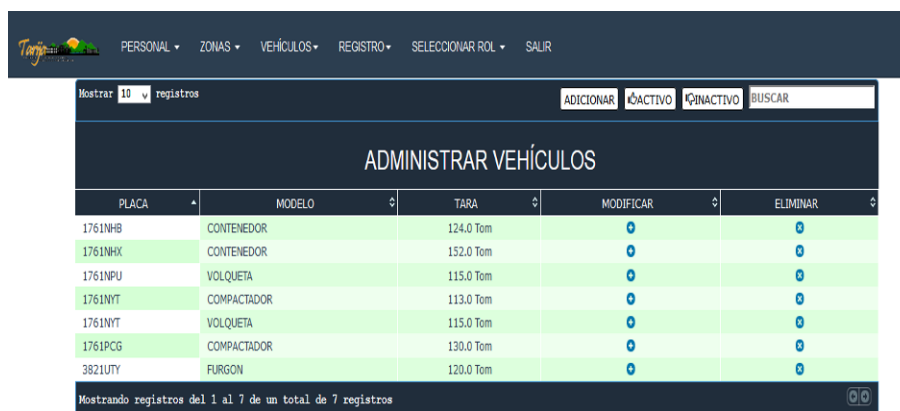


Figura 64 Pantalla Administrar Vehiculos

PANTALLA ADICIONAR VEHICULOS



Figura 65 Pantalla Adicionar Vehiculos

PANTALLA MODIFICAR VEHICULOS

Modificar Vehiculo

Placa
1761NHB

Modelo
CONTENEDOR

Tara(Toneladas)
124

Aceptar Cancelar

Figura 66 Pantalla Modificar Vehiculos

PANTALLA ELIMINAR VEHICULOS

ADMINISTRAR VEHÍCULOS

Desea eliminar el usuario 1761NHB?

Aceptar Cancelar

MODELO	MODIFICAR
CONTENEDOR	+
CONTENEDOR	+
VOLQUETA	+
COMPACTADOR	+
VOLQUETA	+
COMPACTADOR	+

Figura 67 Pantalla Eliminar Vehiculos

PANTALLA ADMINISTRAR BITACORAS

Mostrar 10 registros

ADICIONAR INACTIVO INACTIVO BUSCAR

ADMINISTRAR BITACORAS

Nº BITACORA	ADMINISTRAR VIAJES	AYUDANTES	MODIFICAR	ELIMINAR	VER DATOS
Nº Bitacora : 1	+	+	+	+	+
Nº Bitacora : 2	+	+	+	+	+
Nº Bitacora : 3	+	+	+	+	+

Mostrando registros del 1 al 3 de un total de 3 registros

Figura 68 Pantalla Administrar Bitacoras

PANTALLA ADICIONAR BITACORAS

Adicionar Bitacora

Chofer
Walter

Vehiculo
1761NYT

Hora Salida
Ingrese la Hora de Salida

Hora Llegada
Ingrese la Hora de Llegada

Combustible(Litros)
0

Odometro Salida
0

Odometro Llegada
0

Horometro Salida
0

Horometro Llegada

Figura 69 Pantalla Adicionar Bitacoras

PANTALLA MODIFICAR BITACORAS

Modificar Bitacora

Chofer
Orlando

Vehiculo
1761NYT

Hora Salida
06:30

Hora Llegada
07:45

Combustible(Litros)
21

Odometro Salida
121

Odometro Llegada
21121

Horometro Salida
121

Horometro Llegada

Figura 70 Pantalla Modificar Bitacoras

PANTALLA ELIMINAR BITACORAS

Mostrar registros

ADICIONAR

ADMINISTRAR BITACORAS

Nº BITACORA	ADMINISTRAR VIAJES	AYUDANTES	MODIFICAR	ELIMINAR	VER DATOS
Nº Bitacora : 1	✚	✚	✚	✚	✚
Nº Bitacora : 2	✚	✚	✚	✚	✚
Nº Bitacora : 3	✚	✚	✚	✚	✚

Mostrando registros del 1 al 3 de un total de 3 registros

Figura 71 Pantalla Eliminar Bitacoras

PANTALLA VER BITACORAS

PERSONAL

Ver Bitacora N 1

Chofer
Orlando

Vehiculo
1761NYT

Hora Salida
06:30

Hora Llegada
07:45

Combustible(Litros)
21.0

Odometro Salida
121.0

Odometro Llegada
21121.0

Hora Inicio

Figura 72 Pantalla Ver Bitacoras

PANTALLA ADMINISTRAR VIAJES

Mostrar 10 registros BITACORAS ADICIONAR INACTIVO ACTIVO BUSCAR

ADMINISTRAR VIAJES

N° VIAJE	ZONA DE TRABAJO	MODIFICAR	ELIMINAR	VER DATOS
N° Viaje :1	JUAN XXIII			
N° Viaje :2	3 DE MAYO			

Mostrando registros del 1 al 2 de un total de 2 registros

Figura 73 Pantalla Administrar Viajes

PANTALLA ADICIONAR VIAJES

Adicionar Viaje

Zona
JUAN XXIII

Hora Ingreso
Ingrese la hora del ingreso del viaje

Hora Salida
Ingrese la hora de salida del viaje

Ingreso Al Relleno
Ingrese la hora de ingreso

Salida Del Relleno
Ingrese la hora de salida

Odometro Inicio
0

Odometro Salida
0

Peso Neto(Toneladas)
0

Figura 74 Pantalla Adicionar Viajes

PANTALLA MODIFICAR VIAJES

Modificar Viaje

Zona: JUAN XXIII

Hora Ingreso: 06:12

Hora Salida: 08:35

Ingreso Al Relleno: 12:58

Salida Del Relleno: 13:23

Odometro Inicio: 1210

Odometro Salida: 232323

Peso Neto: 2123

Figura 75 Pantalla Modificar Viajes

PANTALLA ELIMINAR VIAJES

ADMINISTRAR VIAJES

ZONA DE TRABAJO	ELIMINAR
JUAN XXIII	[X]
3 DE MAYO	[X]

Se un total de 2 registros

Desea eliminar N° Viaje :1?

Aceptar Cancelar

Figura 76 Pantalla Eliminar Viajes

PANTALLA VER VIAJES

Ver Viaje

Zona: JUAN XXIII

Hora Ingreso: 06:12

Hora Salida: 08:35

Ingreso Al Relleno: 12:58

Salida Del Relleno: 13:23

Odometro Inicio: 1210.0

Odometro Salida: 232323.0

Peso Neto:

Introduzca formato hora 00:00

Figura 77 Pantalla Ver Viajes

PANTALLA ASIGNAR AYUDANTES

The screenshot shows the 'ADMINISTRAR BITACORAS' interface. At the top, there's a navigation bar with links: PERSONAL, ZONAS, VEHÍCULOS, REGISTRO, SELECCIONAR ROL, and SALIR. Below this, a header bar contains 'Mostrar 10 registros', 'ADICIONAR', 'ACTIVO', 'INACTIVO', and 'BUSCAR'. The main table has columns: N° BITACORA, ADMINISTRAR VIAJES, AYUDANTES, MODIFICAR, ELIMINAR, and VER DATOS. Three rows are visible, each with a bitacora number and a corresponding 'AYUDANTES' button. A modal titled 'ASIGNAR AYUDANTES BITARORA N°: 1' is open, showing a list of names (Pascual, Luis) under 'NOMBRES DE AYUDANTES' and checkboxes under 'SELECCIONAR AYUDANTE'. At the bottom of the modal are 'Aceptar' and 'Cancelar' buttons.

N° BITACORA	ADMINISTRAR VIAJES	AYUDANTES	MODIFICAR	ELIMINAR	VER DATOS
N° Bitacora : 1					
N° Bitacora : 2					
N° Bitacora : 3					

Mostrando registros del 1 al 3 de un total de 3 registros

Asignar Ayudantes

NOMBRES DE AYUDANTES

Pascual

Luis

SELECCIONAR AYUDANTE

☐

☐

Aceptar

Cancelar

Figura 78 Pantalla Asignar Ayudantes

2.1.6.9 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).

MODELADO DE DIAGRAMA DE PAQUETES

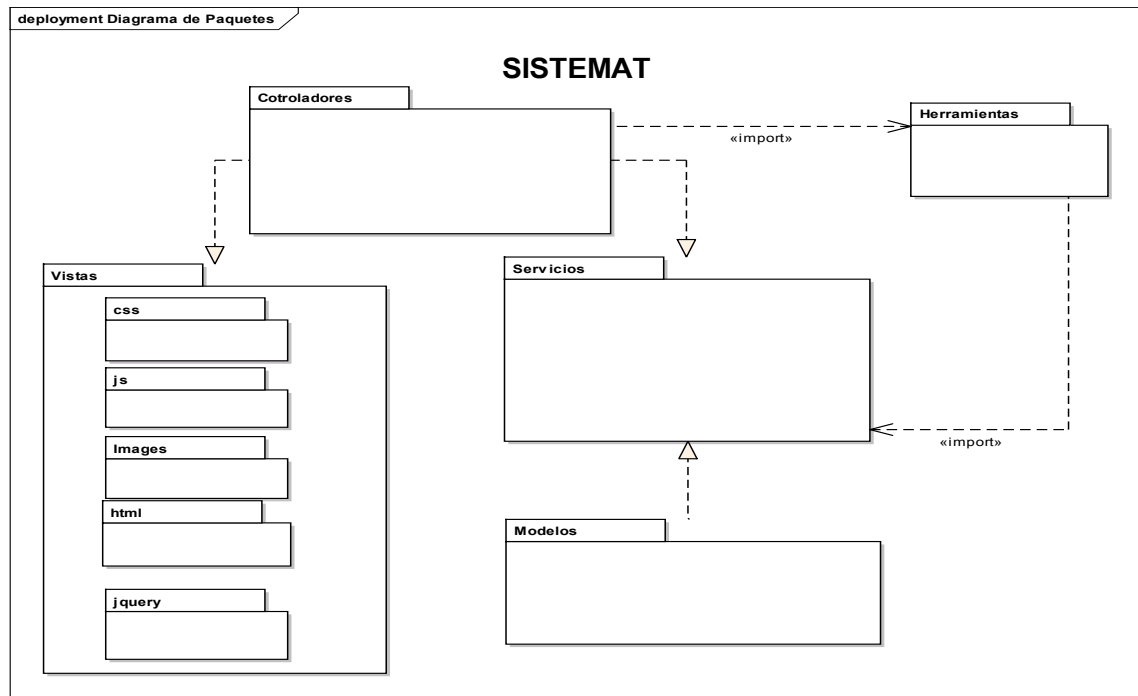


Figura 205 Diagrama de Paquetes

2.1.6.10 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.

MODELADO DE DIAGRAMA DE DESPLIEGUE

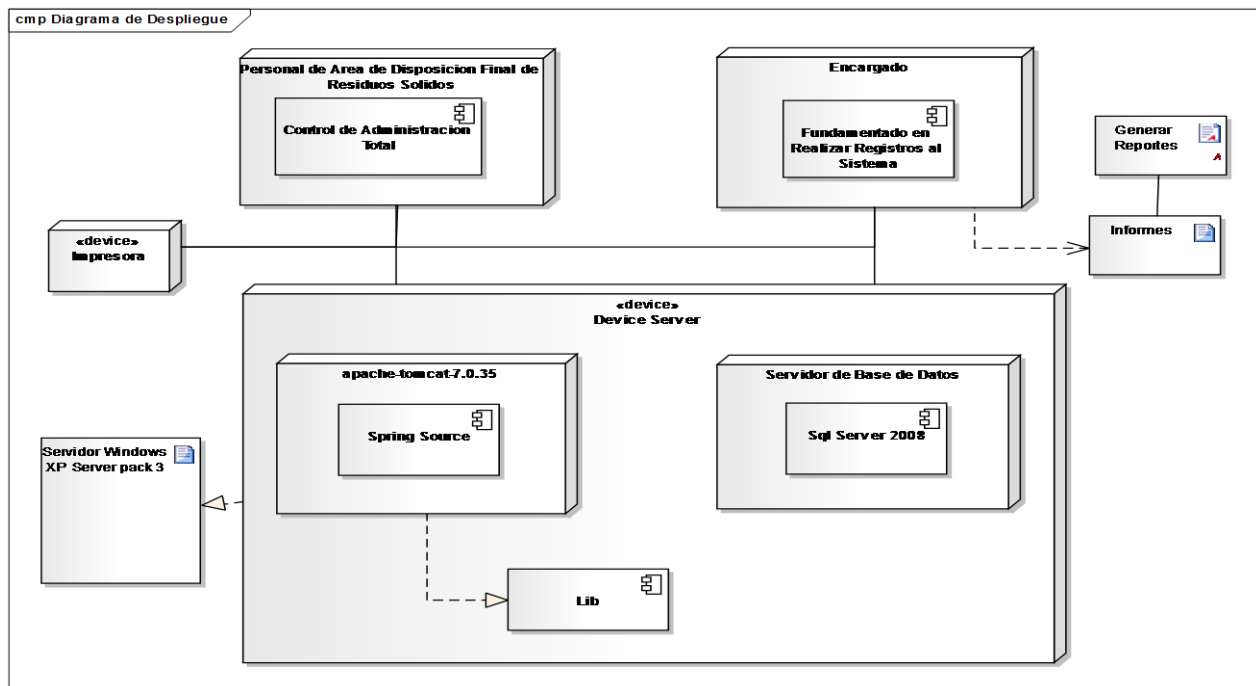


Figura 206 Diagrama De Despliegue

2.1.7 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.

PLAN DE PRUEBAS

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.

OBJETIVO

Este Plan de Pruebas fue desarrollado con el fin de guiar el proceso de pruebas al proyecto “SISTEMAT” para contribuir a optimizar los procedimientos administrativos y tecnológicos , para su mejor funcionamiento, 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.

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 Entidad Municipal de Aseo Tarija, además que sea aprobado por la comunidad de usuarios finales del sistema.

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.

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 Pentium Core en adelante
Sistema operativos XP server pack 3

Tabla 143 Hardware

SOFTWARE		
Aplicación	Versión	Descripción
Sistema Integrado Para El Área De Disposición Final De Residuos Solidos	3.0	Sistema desarrollado.
Enterprise Architect	8.0	Herramienta de Diseño y Desarrollo de Diagramas UML para RUP
PostgreSQL	9.01	Motor de Base de Datos Relacional.

Tabla 144 Software

OTROS		
Descripción	Versión	Descripción
Documentación	3.0	Documentación de Desarrollo de Software

Tabla 145 Otros

ALCANCE

El sistema debe registrar todas las bitácoras generadas día a día.

Este registro debe producir los diferentes balances y devolver reportes para tener una información oportuna.

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.

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.

2.1.8 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.

TIPOS Y TÉCNICAS DE PRUEBAS

PRUEBAS DE POOL DE CONEXIONES

Las pruebas de pooling de conexiones serán realizadas para asegurar que los datos no son corruptos, dentro de las estructuras de datos internas. Estas pruebas son realizadas de forma independiente de la Interfaz de Usuario, en forma de pruebas de caja negra.

Objetivo de la Técnica:	Verificar la las conexiones realizadas a la base de datos y determinar el rendimiento del Sistema y del Motor de Base de Datos (PostgreSQL).
Técnica:	Analizar la carga a la conexión mediante el JDBC Template, simulando usuarios conectados.
Hitos:	Rastreo de cuello de botella en la conexión a la base de datos, realizada por el sistema.
Herramientas Requeridas:	JDBC TEMPLATE.

Criterios de Éxito:	Las conexiones se realizan con fluidez y una por usuario.
----------------------------	---

Tabla 146 Pruebas de Pool de Conexiones

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 147 Pruebas Funcionales

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 148 Pruebas de Validación de Datos

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 149 Pruebas de Perfiles de Desempeño

2.1.9 EJECUCIÓN DE LAS PRUEBAS

CASOS DE PRUEBA CAJA NEGRA

❖ Pantalla Acceso al Sistema

Clases de Equivalencia Válidas y no Válidas

Condición de entrada	Tipo	Clase de Equivalencia Válida	Clase de Equivalencia no Válida
Usuario	Valor	1: Usuario=cadena alfanumérica y símbolos.	3: Usuario=en blanco.

		2: $5 \leq \text{Valor} \leq 10$	4: $10 < \text{Valor} < 5$
Clave	Valor	5: clave= cadena alfanumérica y símbolos. 6: $7 \leq \text{Valor} \leq 7$	7: Clave=en blanco 8: $7 < \text{Valor} < 7$

Casos de prueba

Nº	Clase de Equivalencia	Propósito del caso	Dato de prueba	Mensaje
1	Usuario =cadena de caracteres alfanuméricos y símbolos. $5 \leq \text{Valor} \leq 10$ Usuario =en blanco. $10 < \text{Valor} < 5$	Aceptar Aceptar Denegar Denegar	“3210abc” 123456 “” 123	El login y clave deben tener entre 5 y 10 caracteres
2	Contraseña = cadena de caracteres alfanuméricos y símbolos. $7 \leq \text{Valor} \leq 7$ Contraseña =en blanco. $7 < \text{Valor} < 7$	Aceptar Aceptar Denegar Denegar	5568752 123456 “” 123	El login y clave deben tener 7 caracteres

❖ Pantalla Adicionar y Modificar Usuario

Clases de Equivalencia Válidas y no Válidas

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. $1 \leq \text{Caracteres} \leq 30$	7. Vacío 8. Letra >30
Tipo: Apellido Paterno Tamaño: Apellido Paterno	9. Alfabético 10. $0 \leq \text{letra} \leq 30$	11. Vacío 12. Numero o carácter especial 13. Letra >30
Tipo: Apellido Materno Tamaño: Apellido Materno	14. Alfabético 15. $0 \leq \text{letra} \leq 30$	16. Numero o carácter especial 17. Letra >30
Tipo: Inscrito	18. Numero	19. Ninguno
Tipo: Estado	20. Numero	21. Ninguno
Tipo: Número telefónico	23. Numérico 24. Vacío	25. Alfabético

Prueba Adicionar y Modificar Usuario

Casos de Prueba válidas

CI	7179280
Nombre	Juan
Apellido Paterno	Pérez

Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	78225828

Casos de Pruebas Inválidas

CP1	
CI	a
Nombre	Juan
Apellido Paterno	Pérez
Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	

CI	nombre
Nombre	Juan
Apellido Paterno	Pérez
Apellido Materno	Martínez
Inscrito	111

Estado	1
Teléfono	78225858

❖ **Pantalla Adicionar y Modificar Chofer**

Clases de Equivalencia Válidas y no Válidas

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: Apellido Paterno Tamaño: Apellido Paterno	9. Alfabético 10. 0<=letra<=30	11. Vacío 12. Numero o carácter especial 13. Letra >30
Tipo: Apellido Materno Tamaño: Apellido Materno	14. Alfabético 15. 0<=letra<=30	16. Numero o carácter especial 17. Letra >30
Tipo: Inscrito	18. Numero	19. Ninguno
Tipo: Estado	20. Numero	21. Ninguno
Tipo: Número telefónico	23. Numérico 24. Vacío	25. Alfabético

Prueba Adicionar y Modificar Chofer

Casos de Prueba válidas

CI	7145612
Nombre	Jose
Apellido Paterno	Ramirez
Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	74142563

Casos de Pruebas Inválidas

CP1	
CI	u
Nombre	Jose
Apellido Paterno	Ramirez
Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	

CP1	
CI	nombre
Nombre	Jose
Apellido Paterno	Ramirez
Apellido Materno	Martínez
Inscrito	111
Estado	1
Teléfono	78225858

❖ **Pantalla Adicionar y Modificar Ayudantes**

Clases de Equivalencia Válidas y no Válidas

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: Apellido Paterno Tamaño: Apellido Paterno	9. Alfabético 10. 0<=letra<=30	11. Vacío 12. Numero o carácter especial 13. Letra >30
Tipo: Apellido Materno Tamaño: Apellido Materno	14. Alfabético 15. 0<=letra<=30	16. Numero o carácter especial 17. Letra >30

Tipo: Inscrito	18. Numero	19. Ninguno
Tipo: Estado	20. Numero	21. Ninguno
Tipo: Número telefónico	23. Numérico 24. Vacío	25. Alfabético

Prueba Adicionar y Modificar Ayudantes

Casos de Prueba válidas

CI	5124789
Nombre	Pedro
Apellido Paterno	Pérez
Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	75123698

Casos de Pruebas Inválidas

CP1	
CI	a
Nombre	Pedro
Apellido Paterno	Pérez

Apellido Materno	Martínez
Inscrito	1
Estado	1
Teléfono	76245361

CP1	
CI	nombre
Nombre	Pedro
Apellido Paterno	Pérez
Apellido Materno	Martínez
Inscrito	111
Estado	1
Teléfono	78952314

❖ Pantalla Adicionar y Modificar Roles

Clases de Equivalencia Válidas y no Válidas

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfabético 2. $1 \leq \text{Caracteres} \leq 30$	3. Vacío 4. $\text{Letra} > 30$

Prueba Adicionar Usuario y Modificar Roles

Casos de Prueba válidas

Nombre	ADMINISTRADOR
---------------	---------------

Casos de Pruebas Inválidas

Nombre	1245632
---------------	---------

❖ Pantalla Adicionar y Modificar Zona

Clases de Equivalencia Válidas y no Válidas

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Nombre Tamaño: Nombre	1. Alfabético 2. $1 \leq \text{Caracteres} \leq 30$	3. Vacío 4. $\text{Letra} > 30$

Prueba Adicionar y Modificar Zonas

Casos de Prueba válidas

Nombre	Los chapacos
---------------	--------------

Casos de Pruebas Inválidas

Nombre	
---------------	--

❖ **Pantalla Adicionar y Modificar Vehículo**

Clases de Equivalencia Válidas y no Válidas

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Placa Tamaño: Placa	1. Alfabético 2. $1 \leq \text{Caracteres} \leq 7$	3. Vacío 4. Letra > 7

Prueba Adicionar y Modificar Vehículo

Casos de Prueba válidas

Placa	1761NSP
-------	---------

Casos de Pruebas Inválidas

Placa	
-------	--

❖ **Pantalla Adicionar y Modificar Bitacoras**

Clases de Equivalencia Válidas y no Válidas

Condición de Entrada	Clases Equivalentes	Clases Equivalentes
-----------------------------	----------------------------	----------------------------

	Válidas	Inválidas
Tipo: Fecha Tamaño: Fecha	1. Numérico 2. Formato día:mes:año	3. Vacío 4. Formato año:mes:día
Tipo: combustible	5. Numérico	6.texto
Tipo: odómetro salida	18. Numero	19. texto
Tipo: odómetro llegada	20. Numero	21. texto
Tipo: horometro salida	18. Numero	19. texto
Tipo: horometro llegada	20. Numero	21. texto

Prueba Adicionar y Modificar Bitacoras

Casos de Prueba válidas

Fecha	12:05:2016
Combustible	Juan
Odómetro salida	100
Odómetro llegada	200
Horometro salida	100
Horometro llegada	200

Casos de Pruebas Inválidas

Fecha	12:05:2016
Combustible	Juan
Odómetro salida	100
Odómetro llegada	200
Horometro salida	100
Horometro llegada	

❖ **Pantalla Adicionar y Modificar Viajes**

Clases de Equivalencia Válidas y no Válidas

Condición de Entrada	Clases Equivalentes Válidas	Clases Equivalentes Inválidas
Tipo: Hora ingreso Tamaño: hora ingreso	1. Numérico 2.Formato hora:minutos:am/pm	3. Vacío 4.Formato am/pm:minutos:hora
Tipo: hora salida Tamaño: hora salida	5. Numérico 6.Formato hora:minutos:am/pm	7. Vacío 8.Formato am/pm:minutos:hora
Tipo: ingreso relleno Tamaño: ingreso relleno	9. Numérico 10.Formato hora:minutos:am/pm	11. Vacío 12.Formato am/pm:minutos:hora
Tipo: salida relleno Tamaño: salida relleno	13. Numérico 14.Formato hora:minutos:am/pm	15. Vacío 16.Formato am/pm:minutos:hora
Tipo: odómetro salida	17. Numero	18. texto

Tipo: odómetro llegada	19. Numero	20. texto
Tipo :Peso	21. Numero	22.texto

Prueba Adicionar y Modificar Viajes

Casos de Prueba válidas

Hora ingreso	07:05:AM
hora salida	10:00:AM
ingreso relleno	10:05:AM
salida relleno	10:15:AM
odómetro salida	1000
odómetro llegada	2000
Peso	14000

Casos de Pruebas Inválidas

Hora ingreso	07:05:AM
hora salida	10:00:AM
ingreso relleno	10:05:AM
salida relleno	10:15:AM
odómetro salida	
odómetro llegada	
Peso	14000

2.1.10 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 15 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 definir 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.

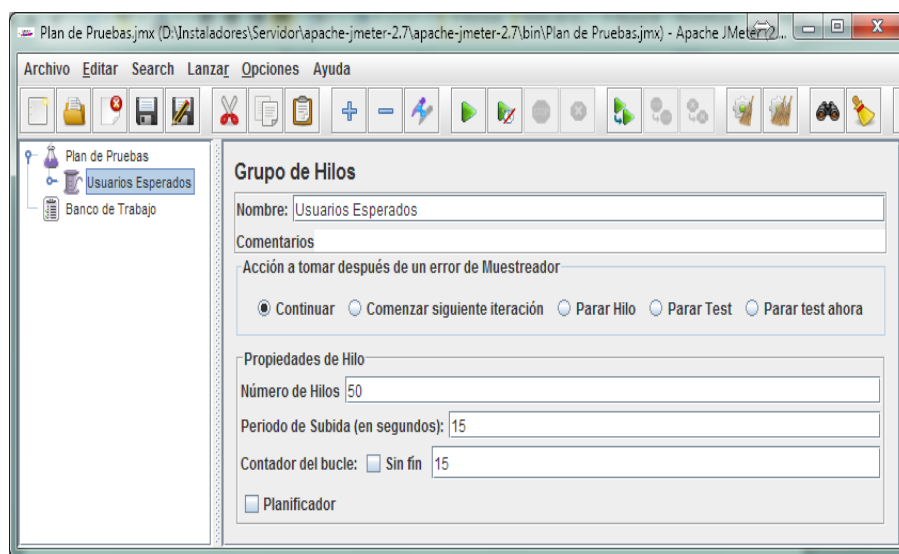


Figura 207 Grupo de Hilos: Usuarios Esperados

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 /emat/index.html, que es la página principal de nuestro sistema web.

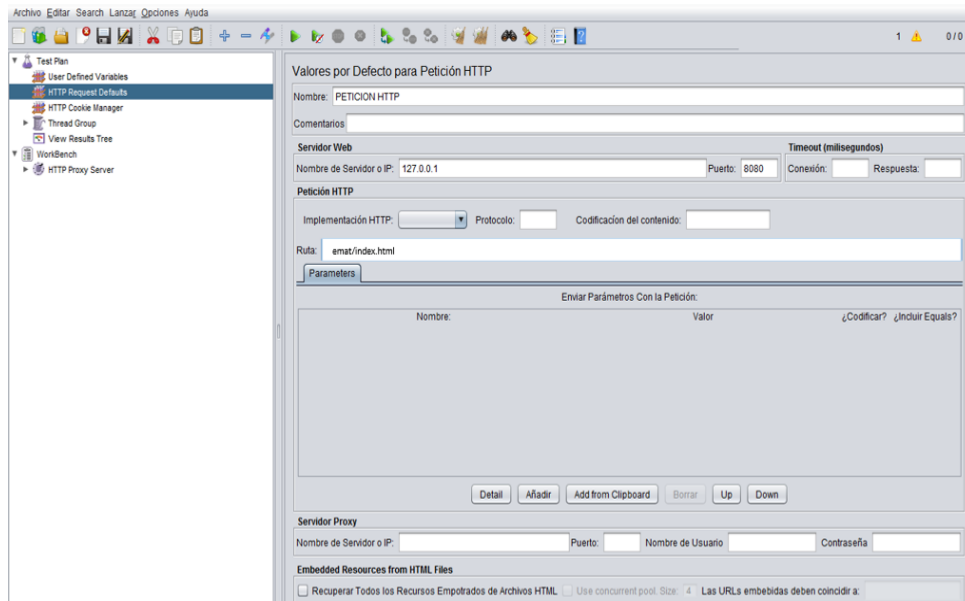


Figura 208 Petición HTTP

- 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 ejecútelo 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 209 Gráfico de Resultados VidiVirtual

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

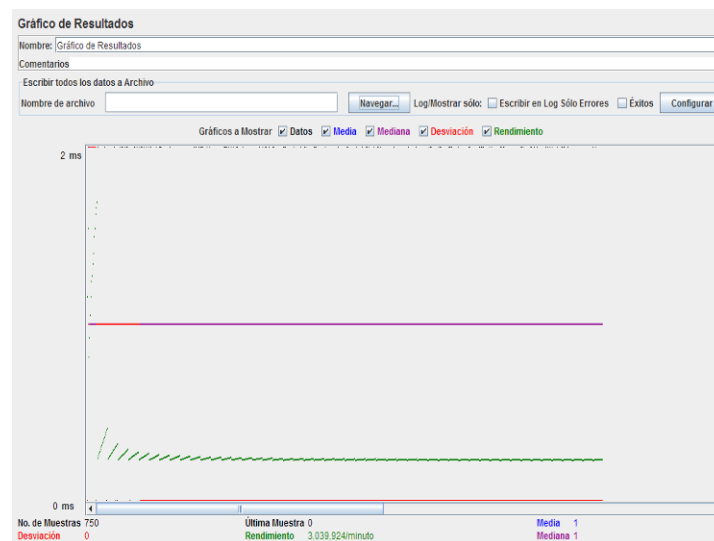


Figura 210 Gráfico de Resultados: SISTEMAT

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

2.1.11 PRUEBA DE POOL DE CONEXIONES

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

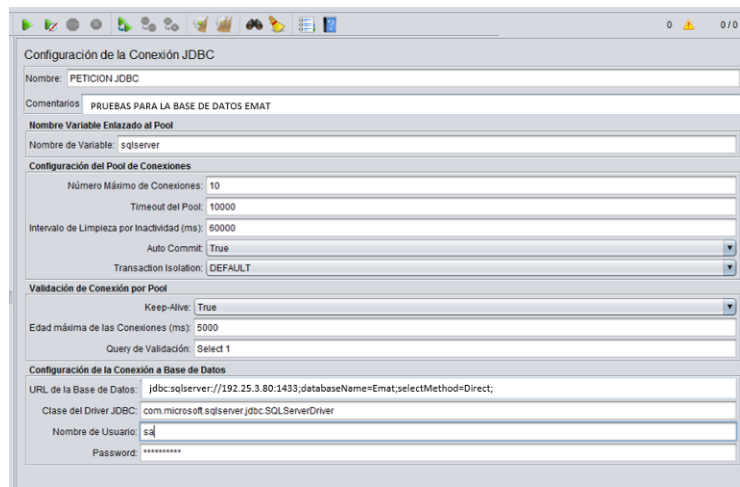


Figura 211 Configuración de la Conexión JDBC

En la cual se probará la siguiente consulta:

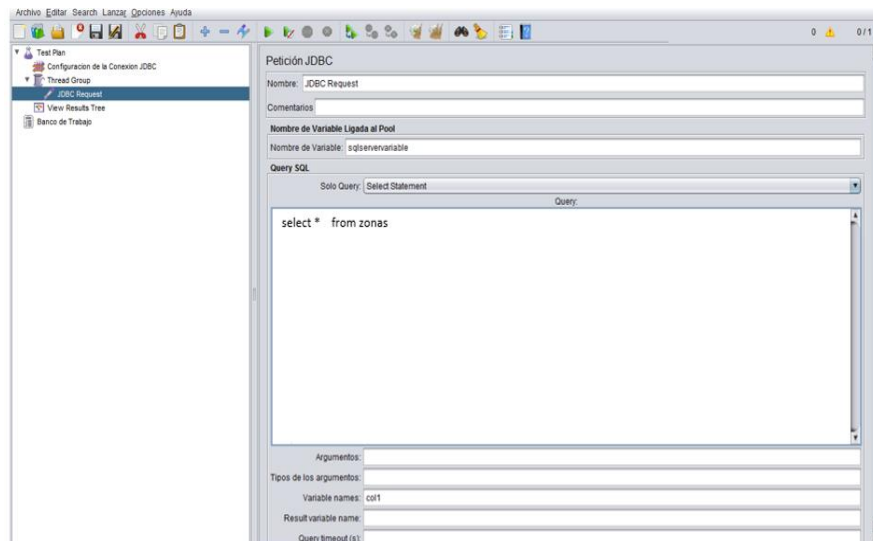


Figura 212 Petición JDBC

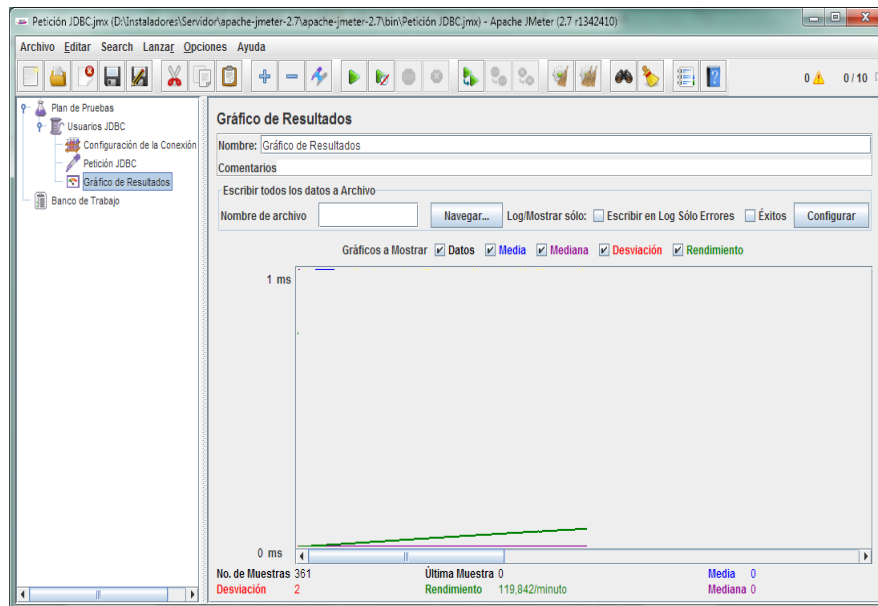


Figura 213 Gráfico de Resultado JDBC

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

2.1.12 LISTA DE RIESGOS

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.

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.

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 153 Roles Encargados de Gestionar los Riesgos

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 154 Escala de Ponderación de los Riesgos

2.1.13 GESTIÓN DEL PROCESO

Se deben identificar las necesidades de información para el proyecto de gestión “SISTEMAT”. 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.

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 157 Métricas

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 158 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 159 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 60 Puntos de Función

UTILIZACIÓN DE LAS MÉTRICAS

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.

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:

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

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

$$E = 44 * 2 = 88 \text{ horas-hombre}$$

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%	$10\% * 88 = 8,8$
Diseño	20%	$20\% * 88 = 17,6$
Programación	40%	$40\% * 88 = 35,2$
Pruebas	15%	$15\% * 88 = 13,2$
Sobrecarga (otras actividades)	15%	$15\% * 88 = 13,2$
TOTAL de esfuerzo	100%	$\Sigma = 88$

Tabla 161 Cálculo de Esfuerzo

El tiempo es de 88 horas – hombre por cada Caso de Uso.

En la cual Si realizamos el siguiente calculo

$$88 \text{ horas-hombre} / 24 \text{ horas} = 3,66 \text{ días por cada Caso de Uso}$$

Es una estimación que no tiene mucha relevancia por considerar solo los requisitos del cliente capturados en los casos de uso.

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:

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

Dónde:

PF= Punto de función

CT= Cuenta Total

Fi= Cuenta de los valores de ajuste de complejidad

Los Puntos de Función se derivan de medidas Directas del dominio de la información

Calculo de la cuenta total:

Parámetro	Cuenta	Factor de Ponderación			Subtotal
		Simple	Medio	Complejo	
Número de entradas de usuario	15	3	4	6	60
Número de salidas de usuario	20	4	5	7	140
Número de peticiones de usuario	15	3	4	6	90
Numero de archivos	3	7	10	15	2
Numero de interfaces externas	2	5	7	10	10
Cuenta Total					351

Tabla 162 Factor de ponderación

Valores de Ajuste de Complejidad

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

Tabla 163 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 164 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 165 Estimación de cantidad líneas de código en función al lenguaje de programación

En base a esta tabla se pueden establecer las líneas de código (LDC) que sería la estimación para nuestro proyecto.

El lenguaje de programación que desarrollamos para nuestro proyecto es Java entonces tendríamos unas 120 LDC por PF:

$$LDC = 120 * 418$$

$$LDC = 50160$$

Para simplificar el proceso de estimación y utilizar una forma más común para su modelo de estimación, Putman y Myers sugieren un conjunto de ecuaciones obtenidas de la ecuación del software:

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

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

Donde

E = esfuerzo en personas – mes

t = duración del proyecto en meses a años.

B = factor especial de destrezas. Para programas mayores a 60 KLDC, B > 0.39.

P = parámetro de productividad. Para aplicaciones comerciales de sistemas, P > 28000

Aplicando las ecuaciones (1) y (2) a nuestro proyecto obtenemos:

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.

2.1.14 CONTROL DE CALIDAD

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 (IEE830).

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.

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.

OBJETIVOS

- Aumentar la satisfacción del personal administrativo.
- Equilibrar el esfuerzo en múltiples demandas.

- Obtener el mejor producto.
- Disfrutar del mejoramiento del proceso administrativo.
- Disponer de métricas objetivas de valoración.
- Ahorrar tiempo y esfuerzo.

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.

SEGUIMIENTO DE LA ADMINISTRACIÓN

ID	Propósito	Sub Actividad	Detalle / Indicación
1	Criterios de Entrada	- Ninguna.	Plan de Desarrollo y proceso de Software y Personal.
2	Revisión	- Examinar estructura gerencial de la organización. - Identificar tareas de cada integrante de la gerencia. - Definir responsabilidades a cada integrante de la gerencia.	Verificar consistencia de la estructura organizacional con las responsabilidades asignadas en Plan de desarrollo de Software.
3	Criterios de Salida	- Estructura de la administración revisada.	Estructura organizacional de la gerencia óptima para el proyecto.

Tabla 171 Seguimiento de la administración

SEGUIMIENTO DE LA DOCUMENTACIÓN

ID	Propósito	Sub Actividad	Detalle / Indicación
1	Criterios de Entrada	- Ninguna.	Plan de Desarrollo de Software.

2	Revisión	- Revisión y análisis del plan de documentación. - Buscar discrepancias. - Discutir discrepancias con el gerente de proyecto.	Reportar discrepancias según documento presentado por el docente y estándares.
3	Criterios de Salida	- Documentación revisada.	Documentos de acuerdo a Estándar, y sin

Tabla 172 Seguimiento de la documentación

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 173 Seguimiento de la adherencia a los Estándares

2.1.15 GESTIÓN DE RIESGOS

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.

RELACIONADOS CON EL IMPACTO EN LA ORGANIZACIÓN

No entregar el proyecto en el tiempo estimado.

El posible no uso del software debido a factores externos.

Poco uso del software.

Mediana cantidad de documentación a entregar al Cliente.

Posibles errores en el producto.

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.

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.

RELACIONADOS CON EL ENTORNO DE DESARROLLO

Si hay herramientas de gestión de proyectos.

Hay herramientas de prueba apropiadas.

Generadores de código para la aplicación.

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.

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.

2.1.16 MEDIOS DE VERIFICACIÓN DEL COMPONENTE SISTEMA INFORMÁTICO

Informe otorgado por el director general de la entidad municipal de aseo Tarija EMAT para medir la satisfacción, eficiencia y eficacia del sistema.

Informe de conformidad con el sistema por el encargado de área, de disposición final de residuos sólidos



GOBIERNO MUNICIPAL DE TARIJA CERCADO

DIRECCIÓN MUNICIPAL DE ASEO TARIJA



CITE /DMAT/571/2015

Tarija ,18 de Noviembre del 2015

Señores**UAJMS-Carrera De Ingeniería Informática****Ref.: Informe de la conclusión del Sistema Informático**

De mi mayor consideración:

El motivo de la presente es para informarle que se culmino con el proyecto trabajando conjuntamente con la Srta. Universitaria Mery Rueda Martínez con CI 7179280, proporcionándole toda la información requerida para llevar adelante el proyecto Mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad de aseo municipal Tarija EMAT actividad planifica a principio de gestión cumpliéndose con los objetivos marcados inicialmente.

Es cuanto puedo informar, para fines de la interesada

Atentamente:



Lic. Rolando Ruiz Gallardo
DIRECTOR MUNICIPAL DE ASEO TARIJA DMAT
GOBIERNO AUTÓNOMO MUNICIPAL DE TARIJA



GOBIERNO MUNICIPAL DE TARIJA CERCADO

DIRECCIÓN MUNICIPAL DE ASEO TARIJA



CITE /DMAT/576/2015

Tarija, 25 de Noviembre del 2015

Señores**UAJMS-Carrera De Ingeniería Informática****Ref.: Informe de la conclusión de la práctica profesional desarrollo del Sistema Informático**

De mi mayor consideración:

En primer lugar agradecer la predisposición de coordinar y apoyar iniciativas institucionales dirigidas a mejorar la calidad del servicio de aseo urbano EMAT que ejecuta en nuestra ciudad.

Por otra parte hago conocer que después de todo el proceso de ejecución del proyecto por parte de la Srta. Mery Rueda Martínez con CI 7179280 con el fin de mejorar el control del área de residuos sólidos; el sistema tiene una consistencia total a lo que se requería cumple con las expectativas planteadas inicialmente, posteriormente sería bueno adoptar más áreas como el área de reciclaje.

Sin otro particular motivo me despido de usted con las condiciones más distinguidas

Atentamente:

Sr. Jhonny Mamani Rodríguez

**ENC. AREA DE DISPOSICION FINAL DE RESIDUOS SOLIDOS
GOBIERNO AUTÓNOMO MUNICIPAL DE TARIJA**

2.2 COMPONENTE 2: CAPACITACIÓN DEL PERSONAL EN EL USO DEL SISTEMA INFORMÁTICO “SISTEMAT”.

2.2.1 INTRODUCCIÓN

El objetivo de este componente es capacitar a los usuarios en el uso del sistema de control de del área de disposición final de residuos sólidos.

El propósito del proyecto es: mejorar el control y seguimiento a las actividades del área de disposición final de residuos sólidos de la entidad de aseo municipal Tarija ‘EMAT’ la capacitación en el uso del sistema informático al personal afectado por el proyecto se convierte en un componente fundamental para el logro del mismo.

El componente capacitación, se encamina hacia el siguiente objetivo: usar adecuadamente el sistema informático por el personal de la entidad “EMAT explotando las fortalezas del mismo.

La Capacitación será presencial dada la corta duración de la misma, la disponibilidad de ambientes, de materiales didácticos y la importancia de posibilitar que el alumno (usuario) reciba asesoramiento oportuno ante cualquier consulta.

2.2.2. CONTEXTO

La Capacitación se desarrollará de forma personal para que se conozca en forma independiente los alcances y beneficios que el sistema informático “Mejorar El Control Y Seguimiento A Las Actividades Del Área De Disposición Final De Residuos Sólidos”, que aportara a la entidad “EMAT” así como los cambios positivos y responsabilidades que esto implica para la entidad.

Se realizarán actividades de capacitación personalizadas de acuerdo al rol que a cada uno le compete.

En este contexto el Capacitador confeccionó la Guía para Capacitación tomando en cuenta los diferentes niveles de preparación del usuario final.

El rol del capacitador estará en función a las categorías de los usuarios según el siguiente detalle:

2.2.3. NIVEL EJECUTIVO:

- Se mostrará la importancia de la capacitación, objetivos y participación del personal.

2.2.4. USUARIOS FINALES

- Gerente general de la entidad municipal de aseo Tarija EMAT
- Encargado del área de disposición final de residuos solidos
- Otros posibles encargados auxiliares

2.2.5 PROPUESTA PEDAGÓGICA

La propuesta pedagógica a utilizar dada las características de los usuarios del sistema SISTEMAT, tendrá en cuenta sus particularidades, el rol que juega dentro de la organización y niveles de conocimiento.

Los métodos de enseñanza a utilizar pondrán su énfasis principalmente en tres teorías de aprendizajes: la cognitiva, con su máximo exponente en el constructivismo, la colaborativa, fundamentalmente para ser explotada con intensidad en la formación del personal técnico y finalmente la significativa aunque también estará presente en la formación del personal de las categorías de nivel ejecutivo y de usuarios finales.

Finalmente se pone de manifiesto el aprendizaje significativo porque el alumno tiene que incorporar los nuevos conocimientos en forma sustantiva en su estructura cognitiva. Esto se logra cuando el alumno relaciona los nuevos conocimientos con los anteriormente adquiridos; pero también es necesario que el alumno se interese por aprender lo que se le está mostrando. De esta forma el alumno no solo obtendrá resultados satisfactorios en un trabajo final, sino que será capaz de enfrentarse a diversas situaciones donde podrá aplicar los conocimientos adquiridos.

2.2.6 CONTENIDOS DE LA CAPACITACIÓN

- **Lección 1:** Acceso al sistema
- **Lección 2:** Adicionar datos
- **Lección 3:** Modificar datos
- **Lección 4:** Eliminar datos
- **Lección 5:** Generar de reportes

2.2.8 RESULTADOS ESPERADOS

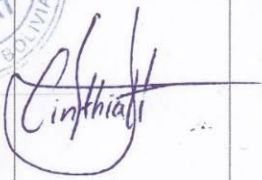
Capacitar al 90% del personal que estará encargado del manejo del sistema.

2.2.9 MEDIOS DE VERIFICACIÓN DEL COMPONENTE

Lista de asistencia firmada por los asistentes.

LISTA DE LOS ASISTENTES DE LA CAPACITACIÓN DEL SISTEMA INFORMÁTICO

Proyecto "Mejorar El Control Y Seguimiento A Las Actividades Del Área De Disposición
Final De Residuos Sólidos"

CONTENIDOS DE LA CAPACITACIÓN			
Lección 1: Acceso al sistema Lección 2: Adicionar datos Lección 3: Modificar datos Lección 4: Eliminar datos Lección 5: Generar de reportes			
CARGO	PARTICIPANTE	FECHA	FIRMA
 Director General de Aseo Urbano EMAT	Lic. Rolando Ruiz Gallardo CI 7172815	26/10/2015	
 Encargado de Área de Disposición Final de Residuos Sólidos	Sr. Jhonny Mamani Rodríguez CI 7185807	25/10/2015 	
 Encargado Auxiliar	Srta. Cinthia Hoyos 7184456	25/10/2015	

CAPÍTULO III

CONCLUSIONES

Y

RECOMENDACIONES

3.1 CONCLUSIONES

- Se ha desarrollado, el proyecto MEJORAR EL CONTROL Y SEGUIMIENTO A LAS ACTIVIDADES DEL AREA DE DISPOSICION FINAL DE RESIDUOS SOLIDOS EN LA ENTIDAD MUNICIPAL DE ASEO TARIJA EMAT. Permitiendo al personal de la empresa trabajar con datos actualizados.
- Se logró cumplir el propósito del proyecto, a través de su indicador automatizando al menos un 80% de los procesos asociados al área de disposición final de residuos sólidos de la entidad municipal de aseo Tarija EMAT.
- El primer componente agiliza los procesos del control del área de disposición final de residuos que contempla el relleno sanitario de Pampa Galana de la empresa ya que el sistema controla cada proceso y genera todos los informes necesarios para la toma de decisiones , permitiendo tener información oportuna para la gerencia .
- La elección RUP como metodología de desarrollo de software empleada, ha sido adecuada porque brindó gran flexibilidad de escalamiento en las iteraciones realizadas.
- Se aplicó el patrón de arquitectura modelo, vista controlador para el diseño del software.
- El software fue desarrollado en el lenguaje JAVA el cual es multiplataforma. Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos y basado en clases que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.

Se desarrolló bajo las herramientas:

- El Spring Framework es un framework de código abierto de desarrollo de aplicaciones para la plataforma Java.
- PostgreSQL es un SGBD relacional orientado a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto. Es ampliamente popular - Ideal para tecnologías Web, fácil de administrar, su sintaxis SQL es estándar y fácil de aprender.
- Tomcat (también llamado Jakarta Tomcat o Apache Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation.
- Enterprise Architect es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento.
- Navicat es una de las herramientas más fiables y rápidas para la administración de bases de datos, que te permitirán simplificar la gestión de las bases de datos y reducir los costes de administración.
- Ireport-5.5.0 es un entorno de desarrollo para crear plantillas de reportes para java en formatos como Word, Excel, html y pdf. Su entorno grafico facilita al usuario a la hora de crear las plantillas.
- El componente de capacitación asegura que tanto el gerente, como el personal encargado del área de disposición final de residuos sólidos de “EMAT” operarán correctamente el sistema.
- Con este proyecto, la optimización se ve reflejada en la mejora de los tiempos y la exactitud en la ejecución de procesos para el control de las bitácoras de la empresa.
- Se obtuvieron los reportes necesarios de la base de datos del sistema, destinados a mejorar el control de las actividades la empresa.
- Se consiguió capacitar de manera positiva a las personas destinadas al empleo del sistema.
- Cabe señalar que algunos procesos que no fueron requeridos por la parte interesada pero fueron incluidos para un mejor desempeño del sistema.

3.2 RECOMENDACIONES

- Es necesario realizar la supervisión de las redes y el sistema en cuanto a la seguridad, es por eso que se debe llevar a cabo la realización del plan de mantenimiento que se tiene para el sistema.
- Para optimizar el rendimiento del sistema, se recomienda implementar el sistema en equipos de marca con procesador Core 2 Duo o superior.
- 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.
- Se recomienda con el futuro implementar el sistema en Python por su simpleza y rapidez al momento de programar, es considerado como el lenguaje de programación más limpio y sencillo de escribir, además que, es un lenguaje tan popular actualmente que sistemas como MacOS y Linux lo traen pre-instalado desde que uno adquiere una computadora.
- El sistema abre la posibilidad de seguir automatizando más procesos, como ser control de ecovecindario para tener datos más precisos.
- Por último se recomienda realizar la capacitación y socialización a cada nuevo encargado para que este pueda utilizar el sistema sin mayores complicaciones y de manera correcta.