

UNIVERSIDAD AUTONOMA “JUAN MISAEL SARACHO”

Facultad de Ciencias y Tecnología

Ingeniería Informática

CAPÍTULO I - EL PROYECTO

I.1. Formulario del Proyecto

I.1.1. Título del proyecto

Optimizar los procesos de transacción en la administración del restaurante la "Carreta".

I.1.2. Área de trabajo

Análisis y diseño de un sistemas de información transaccional para Restaurantes

I.1.3. Responsable del Proyecto

Carrera de Ingeniería Informática – Taller III – Grupo 1.

I.1.4. Entidades Asociadas

- Universidad Autónoma Juan Misael Saracho (UAJMS).
- Restaurant “La Carreta” (Pescadería).

I.1.5. Duración del Proyecto

Ocho meses y medio.

I.1.6. Grupo Responsable del Proyecto

Nahuel Nelson Tolay Gerez – Estudiante de la Carrera de Ingeniería Informática.

I.1.7. Grupo Responsable del Proyecto

| | | | |
|------------------------|----------|-------------------------|--------|
| Apellido | Apellido | Nombre | C.I. |
| Tolay | Gerez | Nahuel Nelson | 713654 |
| Carrera | | Facultad | |
| Ingeniería Informática | | Ciencias y Tecnología | |
| Telf. | Celular | Correo Electrónico | Firma |
| ----- | 7299531 | nahuelnelson@hotmail.co | |
| | | | |
| | | | |

I.1.7. Resumen

El presente proyecto tiene como fin contribuir a fortalecer y optimizar la gestión administrativo del restaurante la "Carreta", restaurante dedicado a prestar un esmerado servicio con una cocina de gran calidad. Las especialidades se basan en la cocina tradicional de variedad de pescados, que presta sus servicios hace ya más de 7 años. Actualmente solo cuenta con una sola sucursal en la ciudad de Tarija.

El propósito del proyecto es optimizar los procesos de transacción en la administración del restaurante la "Carreta", para ello es importante desarrollar un sistema informático y la capacitación del sistema al personal de la microempresa.

Para lograr el propósito del proyecto es necesario cumplir con los siguientes componentes:

1. Desarrollo e implementación de un sistema de gestión administrativa para restaurantes.

Es de vital importancia el incorporar un sistema automatizado al Restaurante la "Carreta" para el registro de clientes, empleados y transacciones de compra-venta, que permita de esta manera acceder a información actualizada y confiable de los distintos registros dentro del negocio, dicho sistema se encargará de proporcionar al personal una notable mejora en cuanto a rapidez en la atención a clientes y certeza en los datos que proporcionan.

Entre las múltiples prestaciones que tendrá el sistema se pueden mencionar las siguientes:

- Registrar transacciones de compra-venta.
- Registro de empleados.
- Registro de clientes.
- Registro de proveedores.
- Registro pago de servicios y control de ganancias.

2. Un curso de capacitación para el manejo del sistema finalizado e implementado a empleados del restaurante.

Siendo necesario efectuar programas de capacitación del sistema al personal de la del restaurante la "Carreta" con el objetivo de que obtengan el conocimiento suficiente sobre las características que ofrece el sistema para que luego manejen el sistema de manera correcta, evitando problemas posteriores.

I.2. El Proyecto

I.2.1. Objetivo General

Optimizar los procesos de transacción en la administración del restaurante la "Carreta".

I.2.2. Objetivos Específicos

- Sistema de gestión administrativa para restaurantes finalizado e instalado.
- Curso de capacitación a empleados del restaurante para el manejo del sistema finalizado e implementado.

I.2.3. Equipo del Proyecto, Actividades y Recursos

| Responsable * | Actividades |
|-------------------------------------|---|
| <p>Director</p> <p>Nahuel Tolay</p> | <p>El director planificara y controlara la ejecución del proyecto estableciendo distintos tipos de recursos con respecto a prácticas que permitan asegurar la calidad del software a desarrollarse, supervisando constantemente el mismo, elaborando el análisis y diseño del sistema, la base de datos, la codificación del sistema en un lenguaje de programación, estableciendo las respectivas pruebas y posterior implementación en el entorno de la empresa.</p> <p>En resumen desarrollara las siguientes actividades:</p> <ul style="list-style-type: none"> • Producir los componentes <ul style="list-style-type: none"> I. Sistema de gestión administrativa para restaurantes elaborado. <ul style="list-style-type: none"> a. Planificación b. Diseño y Prototipo desarrollado en Java del sistema y base de datos. c. Implementación del Prototipo y posterior corrección del mismo. d. Implementación final. e. Recopilación y organización de los contenidos del sistema. f. Diagramado e impresión. II. Capacitación de empleados finalizada. <ul style="list-style-type: none"> a. Proyección y definición de la metodología de enseñanza que se adoptará y del cronograma de capacitación para las jornadas de capacitación. b. Elaboración de material pedagógico para las jornadas de capacitación a las personas que operaran el sistema. c. Llevar a cabo las jornadas de capacitación a las personas que operaran el sistema de gestión culminado. • Llevar control del proyecto • Realizar el seguimiento del proyecto |

Presupuesto

| ITEM | RUBROS | Aporte Universidad | Otro Aporte | TOTAL (Bs.) |
|-------|--|-----------------------|-------------|----------------|
| 10000 | SERVICIOS PERSONALES | | | |
| | 12000 Empleados no Permanentes | | | |
| | Sub total rubro | | | |
| 20000 | SERVICIOS NO PERSONALES | | | |
| | 21000. Servicios Básicos | | | 1450 |
| | 22000. Servicios de transporte | | | 945 |
| | 23000. Alquileres | | | 4200 |
| | 24000. Mantenimiento y reparación | | | 1200 |
| | 25000. Servicios Profesionales y Comerciales | | | 11800 |
| | Sub total rubro | | | 19595 |
| 30000 | MATERIALES Y SUMINISTROS | | | |
| | 31000. Alimentos y Productos Forestales | | | 1800 |

| | | | | |
|--------------|--|--|--|---------------------|
| | 32000. Productos de Papel, Cartón e Impresos | | | 1355 |
| | 33000. Textiles y Vestuario. | | | |
| | 34000. Productos Químicos, Combustibles y Lubricantes | | | |
| | 39000. Productos Varios. | | | 160 |
| | Sub total rubro | | | 3315 |
| 40000 | ACTIVOS REALES | | | |
| | 43000. Maquinaria y Equipo. | | | 2000 |
| | 46000. Descripción de estudios y proyectos para inversión | | | |
| | 49000. Otros Activos | | | |
| | Sub total rubro | | | 2000 |
| | TOTAL | | | <u>24910</u> |
| | TOTAL + 40% Incentivo | | | 34874 |

I.2.4. Cuadro de Involucrados

| Grupos | Intereses | Problemas Percibidos | Recursos y Mandatos |
|---------------------|--|--|--|
| Gerente-Propietario | Mejorar el manejo administrativo restaurante | <ul style="list-style-type: none"> • Pérdida de ganancias • No existen inventarios de insumos y productos • Mala toma de decisiones a nivel administrativo • Pérdida de clientela • Gestión administrativa deficiente • Falta de puntualidad empleados • Nivel de ventas bajo | <p>R. Disponibilidad por pagar e implementar un sistema para un servicio mejorado en el restaurante</p> <p>R. Productos e insumos del día</p> |
| Cajero | Poder prestar un servicio de cuenta impreso exacto a la hora de realizar cobros a los clientes | <ul style="list-style-type: none"> • Cuentas mal calculadas • Cuentas mal registradas • Quejas de los clientes • | <p>R. Disponibilidad de manejar un sistema que ayude a mejorar la eficiencia del local</p> <p>R. Material de escritorio</p> <p>M. Ofrecer un estado de cuentas claro</p> |
| Mozos | Contar con material para registrar pedidos de forma manual | <ul style="list-style-type: none"> • Descuento de salarios • Retraso al servir platos pedidos • Falta de libretas para anotaciones | <p>M. Brindar una excelente atención hacia el cliente</p> <p>M. Especificar bien los pedidos</p> |
| Clientes | Pagar un precio exacto del consumo de bebidas y comidas pedidas por el cliente | <ul style="list-style-type: none"> • Cobros indebidos • Demora en atención | |
| Cocineros/as | Mejorar el conocimiento de pedidos de platos en el sector de la cocina | <ul style="list-style-type: none"> • Exceso de platos servidos • Compra de insumos insuficiente • Desorden de pedidos | <p>R. Indumentaria para cocinar</p> <p>M. Preparar los distintos tipos de platos según pedidos realizados</p> <p>M. Mantener la higiene</p> |

I.2.5. Árbol de Problemas

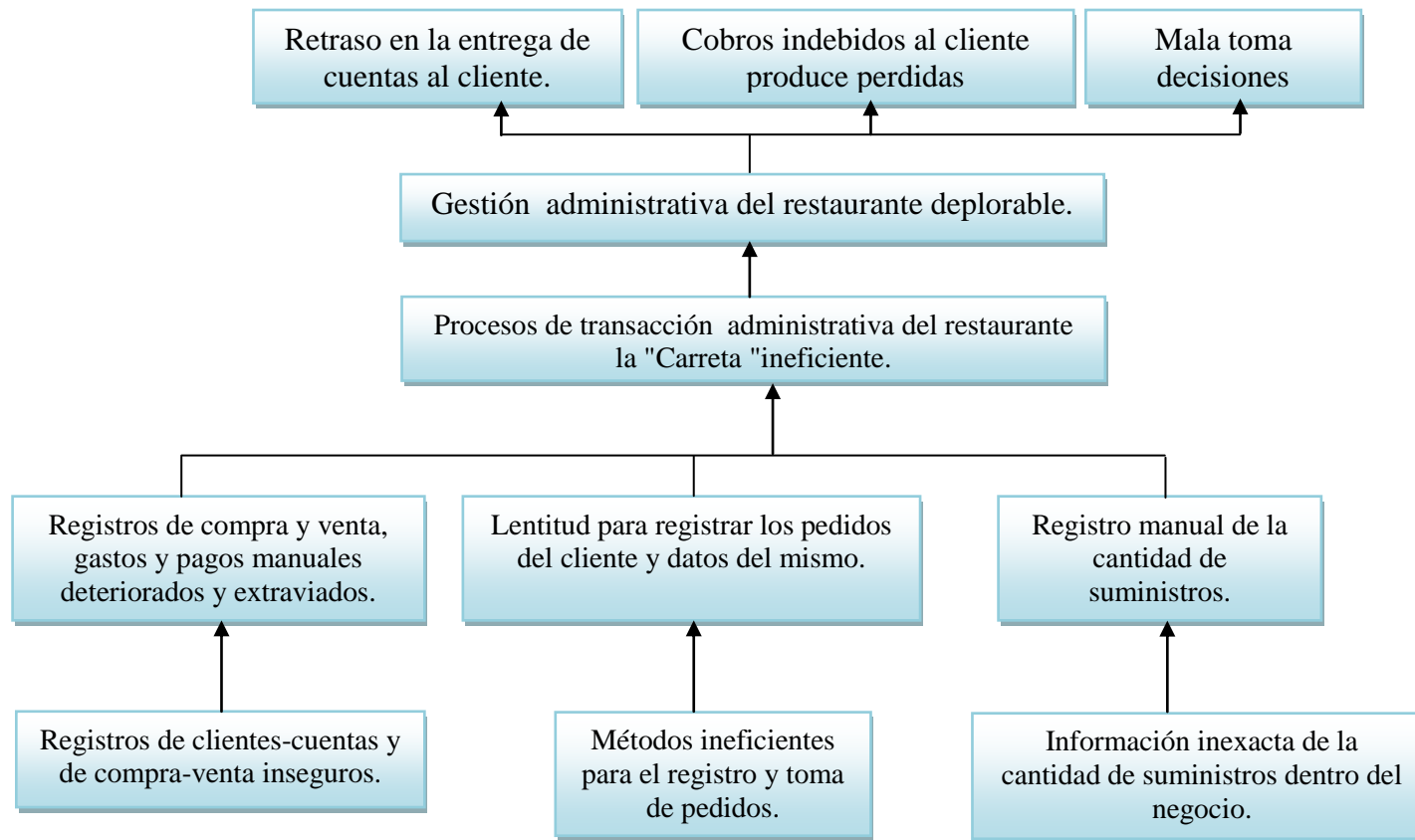


Figura 1. Árbol de Problemas

I.2.6. Árbol de Objetivos

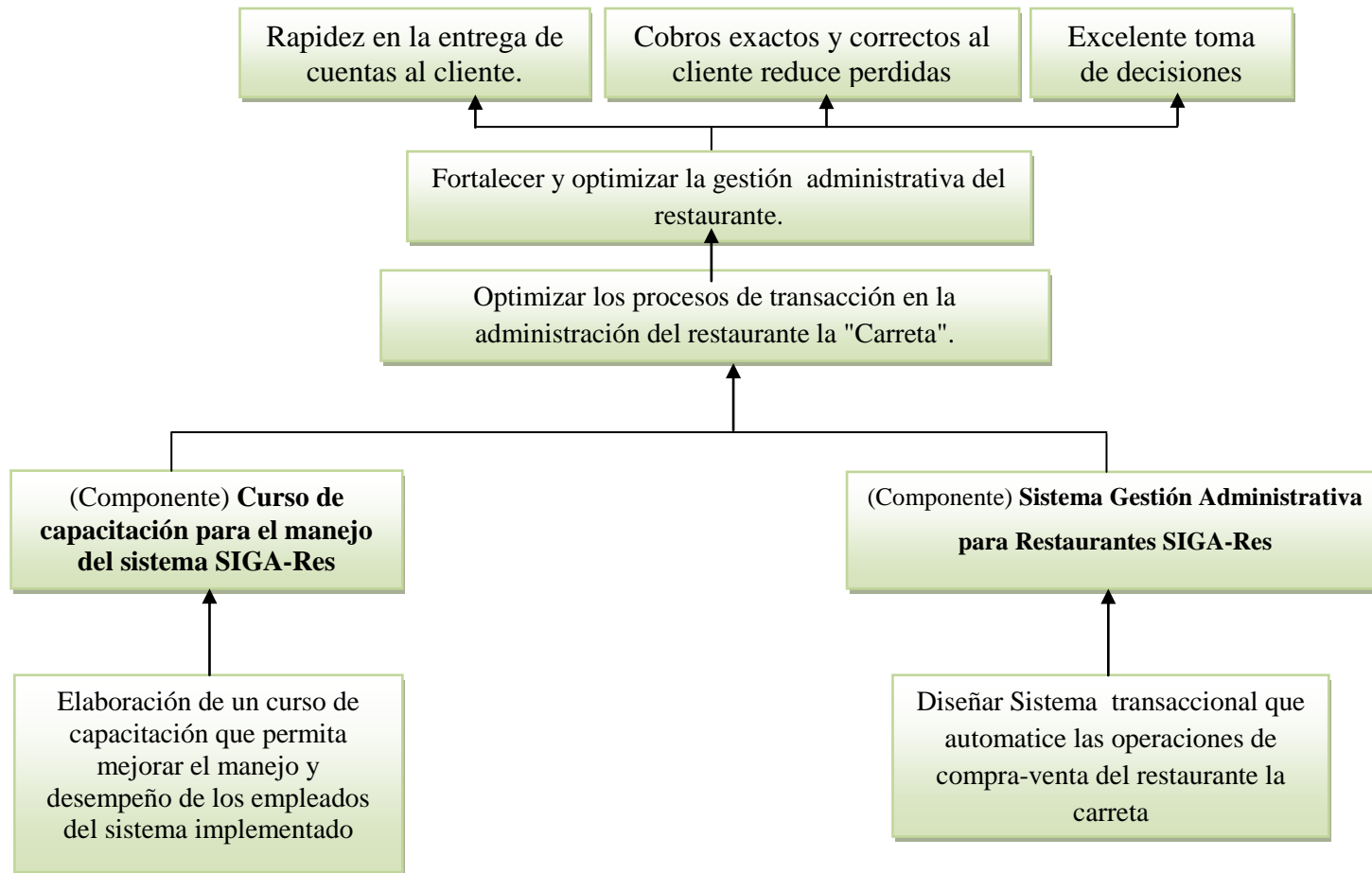


Figura 2. Árbol de Objetivos

I.2.6. Matriz de Marco Lógico

| Resumen Narrativo | Indicadores | Medios de Verificación | Supuestos |
|--|---|---|---|
| <p>Fin</p> <p>Contribuir a fortalecer y optimizar la gestión administrativo del restaurante la "Carreta".</p> | <ul style="list-style-type: none"> Al final del año 2011 aproximadamente el 80% de las actividades administrativas han mejorado gradualmente en cuanto a la agilización de sus procesos. | <ul style="list-style-type: none"> Informe de satisfacción por parte del gerente - propietario del restaurant “La Carreta” dándonos a conocer su conformidad por la confiabilidad y nivel de seguridad que le brindo, los distintos tipos de reportes de ganancias del sistema para tomar decisiones correctas con fines de lucro para su negocio. | <ul style="list-style-type: none"> Gerente y dueño del restaurant “La Carreta” aprueba gozoso el proyecto del desarrollo de un sistema para su negocio. |
| <p>Propósito</p> <p>Optimizar los procesos de transacción en la administración del restaurante la "Carreta".</p> | <ul style="list-style-type: none"> Al concluir el proyecto, la forma de manejar los registros de personal, clientes y transacciones de compra, venta, cuentas, gastos en servicios y suministros es incrementada a un 62.5% de eficiencia.(formula de eficiencia ver pág. 278) | <ul style="list-style-type: none"> Informe de conformidad por el gerente - propietario del restaurant hacia el director del proyecto certificando que el sistema optimizo y mejoro el nivel de ganancias, aumento de clientela y satisfacción verbal de | <ul style="list-style-type: none"> Presupuesto total dispuesto para el desarrollo e implementación del nuevo sistema de gestión administrativa para el restaurant “La Carreta”. La mayoría de los empleados asistieron al |

| | | | |
|--|---|--|---|
| | <ul style="list-style-type: none"> • La tasa de información errónea en los registros de cuentas se disminuirá en al menos un 50%, tomando como base los reportes erróneos de la gestión 2010. | <p>clientes para con los servicios del restaurant.</p> | <p>curso de capacitación dictado por el director del proyecto.</p> |
| <p>Componentes</p> <p>1. Sistema de gestión administrativa para restaurantes finalizado e instalado.</p> | <p>1.1. El 19/05/10 se tendrá concluido en un 95% la fase de Inicio.</p> <p>1.2. El 18/06/10 se tendrá concluido en un 90% la fase de Elaboración.</p> <p>1.3. El 12/10/11 se tendrá concluido hasta 95% la fase de Construcción.</p> <p>1.4. El 24/02/11 se realizan las últimas pruebas y se realiza la entrega del producto.</p> | <p>1.1.1 Documento Visión: Culminación de la fase de Inicio del Proyecto aprobado por los Docentes de Taller III.</p> <p>1.2.1 Documento Visión: Culminación de la fase de Elaboración del Proyecto aprobado por los Docentes de Taller III.</p> <p>1.3.1 Documento Visión: Culminación de la fase de Construcción del Proyecto aprobado por los Docentes de Taller</p> | <ul style="list-style-type: none"> • Gerente proporciona toda la información necesaria de las distintas transacciones y movimientos realizados dentro del restaurant “La Carreta” que permite el avance progresivo en el desarrollo del sistema. |

| | | | |
|---|--|---|---|
| <p>2. Curso de capacitación para el manejo del sistema finalizado e implementado a empleados del restaurante.</p> | <p>2.1 El 10/03/11 se realiza el curso de capacitación a con la presentación del manejo del sistema y sus componentes, el 99% del personal involucrado fue capacitado en el manejo del sistema y sus componentes.</p> | <p>III.</p> <p>1.4.1 Documento Visión: Culminación de la fase de Transición del Proyecto aprobado por los Docentes de Taller III.</p> <p>1.5.1 Carta de conformidad en la implantación del sistema otorgado por el gerente propietario del restaurante "La Carreta".</p> <p>2.1.1 Copias del material didáctico utilizado en la capacitación de los usuarios del Sistema.</p> | |
| <p>Actividades</p> <p>III. Sistema de gestión administrativa para restaurantes elaborado.</p> <p>g. Planificación</p> <p>h. Diseño y Prototipo desarrollado</p> | <p>Informe detallado del presupuesto en el punto 3</p> <p>COSTO TOTAL PROYECTO 24910</p> | <ul style="list-style-type: none"> Informe de la ejecución presupuestaria. | <p>Los desembolsos de dinero se realizan de acuerdo a lo pactado.</p> <p>Disponibilidad de recursos</p> |

| | | | |
|--|-----------|--|--|
| <p>en Java del sistema y base de datos.</p> <ul style="list-style-type: none"> i. Implementación del Prototipo y posterior corrección del mismo. j. Implementación final. k. Recopilación y organización de los contenidos del sistema. l. Diagramado e impresión. <p>IV. Capacitación de empleados finalizada.</p> <ul style="list-style-type: none"> d. Proyección y definición de la metodología de enseñanza que se adoptará y del cronograma de capacitación para las jornadas de capacitación. e. Elaboración de material pedagógico para las jornadas de capacitación a las personas que operaran el sistema. f. Llevar a cabo las jornadas de capacitación a las personas que operaran el sistema de gestión culminado. | <p>Bs</p> | | <p>económicos para el desarrollo del sistema gestión administrativa para el restaurant “La Carreta”.</p> |
|--|-----------|--|--|

Tabla 1. Matriz de Marco Lógico

UNIVERSIDAD AUTONOMA "JUAN MISAEL SARACHO"

Facultad de Ciencias y Tecnología

Ingeniería Informática

CAPÍTULO II - COMPONENTES

II.1. Sistema de Gestión Administrativa para Restaurantes

II.1.1. Lineamientos Generales

II.1.1.1. Introducción

En el transcurso de los últimos años la sociedad vive una época de acceso a tecnologías de información que han caracterizado el aumento de la demanda de sistemas de información para el manejo de negocios a nivel administrativo que son los tipos de sistemas más buscados e implementados hoy en día.

Debido a esto es que muchos restaurantes que aun siguen en el manejo de su administración mediante papeleo y archivos documentados se ven en la necesidad de actualizarse, dado que el problema principal va mas allá de solo actualizarse es considerable reconocer que los problemas reales de los restaurantes son en sí el desconocimiento de los suministros tanto de productos e insumos, nivel de ganancias bajo por no contar con exactos controles de cuentas y cobros, que no exista un nivel de confianza estable en toma de decisiones para poder determinar si es viable seguir o no el rubro de restaurantes ,debido al desconocimiento total o parcial del movimiento económico dentro del mismo siendo el caso de estudio el Restaurante “la Carreta”.

El proyecto planteado por el director en desarrollo de software, fue realizar un trabajo de carácter económico dirigido específicamente a la gestión administrativa de restaurantes ya que debido a los problemas anteriormente mencionados se consideran necesarios cubrir mediante la implementación de un sistema que gestione la administración en los restaurantes de manera eficiente.

Por eso se planteo la elaboración de un sistema administrativo que permita cubrir los problemas y necesidades más importantes dentro del negocio de los restaurantes, mejorando y optimizando el nivel de eficiencia administrativo y económico, aumentando el conocimiento del nivel de ventas, gastos y ganancias, a futuro tras la implementación del sistema para excelentes tomas de decisiones que permitan invertir y hacer crecer el restaurante. El diseño y desarrollo se utilizara las metodologías de UML, dentro de las

aplicaciones para el desarrollo del sistema se utilizaran Eclipse, Postgres para el desarrollo de la base de datos y la codificación mediante lenguaje JAVA.

II.1.1.2. Misión de la Empresa

Somos una empresa dedicada a ofrecer los mejores platos de pescados de la zona de Tarija brindando la satisfacción de nuestros clientes al máximo otorgando la mejor calidad y variedad de productos, un excelente servicio, además de la mejor atención en donde nuestro compromiso principal es crear experiencias agradables para nuestros clientes a través del deleite de nuestros platos típicos.

II.1.1.3. Visión de la Empresa

Ser una empresa sobresaliente en la gastronomía de pescados típicos del departamento de Tarija, ser recocidos por la calidad de nuestros productos y la calidez de nuestros colaboradores.

II.1.1.4. Organigrama de la Empresa

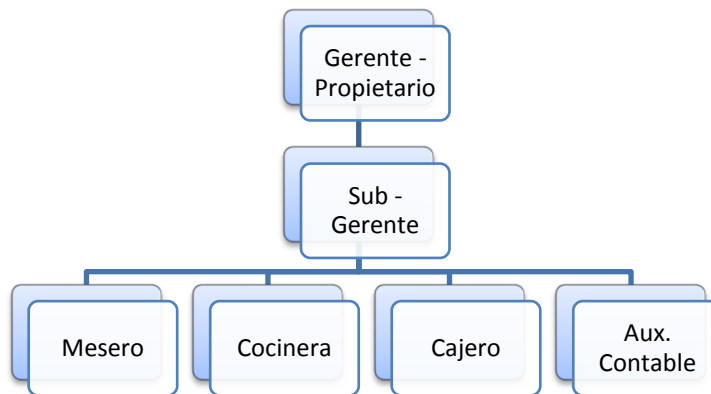


Figura 1. Organigrama de la Empresa

II.1.2. Propuesta

II.1.2.1 Objetivos de la Propuesta

II.1.2.1.1 Objetivo General

Diseñar, Desarrollar e Implementar un Sistema Transaccional enfocado en el área de los restaurantes con el propósito de optimizar los procesos de transacción en la administración del restaurante la "Carreta".

II.1.2.1.2 Objetivos Específicos

- Recabar información de parte del personal y gerente propietario del Restaurante “La Carreta” para obtener una visión clara de los principales problemas que existen ah ser resueltos.
- Usar métodos adecuados para la recopilación de información.
- Hacer uso de las herramientas necesarias y adecuadas que ayuden al desarrollo del Software planteado.
- Hacer uso del modelo orientado a objetos y luego rediseñarlo en el modelo entidad-relación para posteriormente diseñar la base de datos del sistema que se lograra desarrollar.
- Crear una GUI (Interfaz Gráfica de Usuario) amigable, con la que el usuario pueda interactuar de manera fácil y sencilla.
- Realizar las pruebas que sean necesarias para tener una garantía de que el sistema desarrollado cumplirá con el objetivo propuesto.
- Modelar el Sistema con diagramas UML.
- Realizar la codificación del Sistema con lenguaje de programación Java.

- Utilizar el motor de base de datos PostgreSQL que mantenga un registro de toda la información que usará el Sistema para cumplir el objetivo general.

II.1.2.2 Descripción del Problema

II.1.2.2.1 Título descriptivo del Problema

Diseñar un Software de Gestión Transaccional para Optimizar los procesos de transacción en la administración del restaurante la "Carreta".

II.1.2.2.2 Situación Problemática

Ahondando el caso de estudio para el cual se está desarrollando el proyecto. Por parte del Restaurante "La Carreta" se describe el siguiente problema:

- Perdida de ganancias
- Inexistencia de inventarios de insumos y productos
- Mala toma de decisiones
- Perdida de clientela
- Cuentas mal calculadas
- Pedidos mal registrados
- Cobros indebidos
- Nivel de ventas bajo

II.1.2.2.3. Planteamiento del Problema

Dentro del Restaurante "La Carreta", particularmente en caja y gerencia no existe una herramienta propia diseñada única y particularmente para dicho restaurante que le sirva de ayuda y pueda facilitar la Gestión Administrativa, que necesariamente el gerente necesita

para poder tener un conocimiento completo de las mayorías de las ventas, compras y demás para poder tomar decisiones que se realizan diariamente para los suministros necesarios del negocio.

Por lo visto anteriormente debemos denotar que:

- Se debe diseñar un Software centrado fundamentalmente en la Gestión administrativa transaccional del Restaurante “La Carreta”.
- Optimizar los procesos de transacción en la administración del restaurante la "Carreta".

II.1.2.3. Propuesta de Solución

II.1.2.3.1. Solución Propuesta

Diseño, Elaboración e Implementación de un Sistema de Gestión que implementará esencialmente las posibilidades de administrar de manera sencilla mejorando claramente la calidad del servicio, por otra parte el Sistema permitirá el manejo de la información necesaria que es imprescindible para el buen manejo económico del negocio con respecto a la mayoría de las decisiones que se toman para poder suministrar y administrar el restaurante.

Este componente se dará por terminando con la implementación y entrega del producto, lo cual se logra al culminar con éxito las 4 fases de la metodología RUP.

II.1.2.3.2. Metodología para el Desarrollo Del Software

Para el Desarrollo del Sistema se empleará como base la Metodología RUP (Proceso Racional Unificado), este proceso maneja seis principios claves:

- Adaptación del Proceso.- El proceso deberá adaptarse a las características propias de la organización. El tamaño del mismo, así como las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.

- Balancear Prioridades.- Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. *Debe encontrarse un balance que satisfaga los deseos de todos.*
- Colaboración entre Equipos.- El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados, etc.
- Demostrar valor interactivamente.- Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.
- Elevar el nivel de abstracción.- Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas por nombrar algunos. Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.
- Enfocarse en la Calidad.- El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción.

Por otra parte esta Metodología consta de cuatro fases:

1. Fase de Inicio.- Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto.
 - Definición del problema
 - Definición del contenido
 - Requisitos de Considerar
 - Modelado del negocio

2. Fase de Elaboración.- Se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
 - Modelado de la Base de Datos
 - Diagramas UML
 - Prototipos
3. Fase de Construcción.- Se concentra en la elaboración de un producto totalmente operativo y eficiente, así como también la elaboración de un manual de usuario e instalación.
 - Programación del Sistema
 - Documentación del código
 - Elaboración de manuales
4. Fase de Transición.- Se Instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.
 - Pruebas finales
 - Implementación

II.1.2.3.3. Beneficios para el Cliente

Los principales beneficios serán para el cliente, puesto que el Sistema que se desarrollará e implementará contará con las características que permitan al usuario calcular de manera exacta y correcta la mayoría de las cuentas de una forma rápida y sencilla.

II.1.2.3.4. ¿Como ayuda el sistema al restaurante la Carreta?

Dado que el Sistema a desarrollarse tiene como principal característica permitir un mayor control administrativo de las transacciones, permitirá al Restaurante mejorar los procesos

optimizando el control adecuado de la información, evitando la pérdida de datos, las morosas búsquedas por información requerida, etc.

II.1.2.3.5. Beneficios Directos

Los beneficios directos recaen sobre el titular del Restaurante y el personal que hará uso del Sistema para lograr su propósito.

- Gerente General
- Cajero

II.1.2.3.6. Beneficios Indirectos

Los beneficios indirectos recaen sobre terceros que no estén involucrados de forma directa como el titular del Restaurante.

- Empleados
- Proveedores
- Mozos

II.1.3. Ingeniería de Requerimientos

II.1.3.1. ¿Qué es la Ingeniería de Requerimientos?

Ingeniería.- En base a conocimientos adquiridos a lo largo de los recientes años de estudio realizados se puede llegar a la conclusión de que “Ingeniería es la aplicación de conocimientos en ciencia y tecnología para resolver los diferentes problemas a los cuales está sometida la humanidad en general”.

Requerimiento.- De forma general se podría decir que un requerimiento sería las solicitudes que se le hace a alguna persona.

Si juntamos estas dos definiciones y las aplicaciones a lo que es el Desarrollo de Sistemas se podría decir que la Ingeniería de Requerimientos es “Aplicar conocimientos informáticos

para dar solución a las diferentes solicitudes planteadas por una persona (usuario) sobre las características que debe poseer un cierto Sistema a ser desarrollado”.

Según [Ortas 1997] la Ingeniería de Requerimientos se define como “conjunto de actividades en las cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución (o en casos a veces más de una)”.

II.1.3.2. ¿Por qué se utiliza Ingeniería de Requerimientos?

La Ingeniería de Requerimientos se utiliza para definir todas las actividades involucradas en el descubrimiento, documentación y mantenimiento de los requerimientos para un producto determinado. El uso del término “ingeniería” implica que se deben utilizar técnicas sistemáticas y repetibles para asegurar que los requerimientos del sistema estén completos y sean consistentes y relevantes.

II.1.3.3. Propósito de Requerimientos

- Permite que los analistas expliquen cómo han entendido lo que el cliente y usuarios pretenden del sistema.
- Indican a los diseñadores que funcionalidad y características va a tener el sistema resultante.
- Establecen para los desarrolladores la especificación del comportamiento del sistema.
- Indican a los testadores que demostraciones llevar a cabo para que el cliente se convenza de que el sistema que se le está entregando, está de acuerdo a lo que se había solicitado.

II.1.3.4. Requerimientos de Usuario

Se especifican en un lenguaje entendible por los usuarios del sistema que no tienen conocimientos técnicos, generalmente se expresan usando lenguaje natural, tablas y

diagramas que todos puedan entender, Son suficientes para que los usuarios entiendan o sepan que esperar del sistema en desarrollo.

II.1.3.4.1. Requerimientos Funcionales

A continuación se describen las funcionalidades que debe proporcionar el sistema.

- **Gestionar Pedidos**

Req 01 Registrar Nuevo Pedido

El usuario podrá registrar los diferentes tipos de pedidos de los clientes previamente habiendo escogido el número de mesa de la lista que muestre el sistema que serán almacenados en la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 02 Eliminar Datos Pedido

El usuario podrá eliminar el pedido que se haya realizado eliminando el pedido completo de la base de datos

Necesidad: Obligatoria

Prioridad: Alta

Req 03 Modificar Pedido

El sistema desplegará los datos del pedido correspondiente del cliente y mesa, donde el usuario podrá modificar los tipos de platos y bebidas que fueron escogidos anteriormente para posteriormente el usuario pueda guardarlos en la base de datos del sistema

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Mesas**

Req 04 Registrar Nueva Mesa

El usuario podrá registrar una nueva mesa con sus respectivas características que posteriormente se guardaran en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 05 Modificar Datos Mesa

El usuario podrá modificar los datos de la mesa que se haya escogido de la lista desplegada para posteriormente guardarlos en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 06 Eliminar Datos Mesa

El usuario podrá eliminar los datos de la mesa de la base de datos del sistema que se haya escogido de la lista desplegada por el sistema.

Necesidad: Obligatoria

Prioridad: Alta

• **Gestionar Precios**

Req 07 Eliminar Plato

El usuario podrá eliminar los detalles del plato escogidos previamente de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 08 Modificar Plato

El usuario podrá modificar los detalles del plato escogidos previamente de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 09 Registrar Nuevo Plato

El usuario podrá registrar los detalles del plato para posteriormente guardarlos dentro de la base de datos del sistema

Necesidad: Obligatoria

Prioridad: Alta

Req 10 Eliminar Bebida

El usuario podrá eliminar los detalles de la bebida escogidos previamente de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 11 Modificar Bebida

El usuario podrá modificar los detalles de la bebida escogidos previamente de la lista desplegada por el sistema de la base de datos

Necesidad: Obligatoria

Prioridad: Alta

Req 12 Registrar Nueva Bebida

El usuario podrá registrar un nuevo detalle de bebida que posteriormente se guardaran en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

• **Gestionar Reservas**

Req 13 Eliminar Reserva

El usuario podrá eliminar la reserva que se escogerá previamente el número de la lista desplegada correspondiente.

Necesidad: Obligatoria

Prioridad: Alta

Req 14 Modificar Reserva

El usuario podrá modificar los datos de la reserva realizada, previamente escogido del número de reserva de la lista desplegada correspondiente.

Necesidad: Obligatoria

Prioridad: Alta

Req 15 Registrar Nueva Reserva

El usuario podrá registrar una nueva reserva previamente habiendo escogido el número de mesa.

Necesidad: Obligatoria

Prioridad: Alta

• **Gestionar Clientes**

Req 16 Eliminar Cliente

El usuario podrá eliminar los datos del cliente que escogerá de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 17 Modificar Cliente

El usuario podrá modificar los datos del cliente previamente habiéndolo escogido de la lista desplegada por el sistema de la base de datos

Necesidad: Obligatoria

Prioridad: Alta

Req 18 Registrar Nuevo Cliente

El usuario podrá registrar los datos del cliente en el sistema que serán guardados en la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Gastos**

Req 19 Modificar Suministro

El usuario podrá modificar los datos de suministro escogidos de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 20 Registrar Nuevo Suministro

El usuario podrá registrar los datos un nuevo suministro para posteriormente guardarlos en la base de datos del sistema

Necesidad: Obligatoria

Prioridad: Alta

Req 21 Eliminar Suministro

El usuario podrá eliminar del detalle de suministro escogido de la lista desplegada por el sistema de la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 22 Eliminar Servicio

El usuario podrá eliminar los datos del servicio de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 23 Modificar Servicio

El usuario podrá modificar los datos del servicio de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 24 Registrar Nueva Servicio

El usuario podrá registrar los datos de un nuevo servicio que posteriormente se guardaran en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

• **Gestionar Proveedores**

Req 25 Eliminar Proveedor

El usuario podrá eliminar los datos de un proveedor escogido de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 26 Modificar Proveedor

El usuario podrá modificar los datos de un proveedor escogido de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 27 Registrar Nuevo Proveedor

El usuario podrá registrar los datos de un nuevo proveedor que posteriormente se guardaran en la base de datos del sistema

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Empleados**

Req 28 Registrar Nuevo Empleado

El usuario podrá registrar los datos de un nuevo empleado para posteriormente guardarlos en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 29 Modificar Empleado

El usuario podrá modificar los datos de un empleado escogido de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 30 Eliminar Empleado

El usuario podrá eliminar los datos de un empleado escogido de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Cargos**

Req 31 Registrar Nuevo Cargo

El usuario podrá registrar un nuevo cargo con sus datos correspondientes en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 32 Modificar Cargo

El usuario podrá modificar los datos de un cargo de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 33 Eliminar Cargo

El usuario podrá eliminar los datos de un cargo de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 34 Registrar Nuevo Salario

El usuario podrá registrar un nuevo salario correspondiente a un cargo que se guardara en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 35 Modificar Salario

El usuario podrá modificar el salario de un cargo escogido previamente de la lista desplegada por el sistema de la base de datos.

Necesidad: Obligatoria

Prioridad: Alta

Req 36 Eliminar Salario

El usuario podrá eliminar los datos de salario de un cargo de la lista desplegada por el sistema.

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Usuarios**

Req 37 Registrar Nuevo Usuario

El usuario podrá registrar los datos de un nuevo usuario para posteriormente ser guardados en la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

Req 38 Modificar Datos Usuario

El usuario podrá modificar los datos de un usuario de la lista desplegada por el sistema de la base de datos del sistema.

Necesidad: Obligatoria

Prioridad: Alta

- **Gestionar Reportes**

Req 39 Generar Reportes

El usuario podrá generar una lista de reportes que el sistema le permitirá imprimir.

Necesidad: Obligatoria

Prioridad: Alta

II.1.3.4.2. Requisitos comunes de los interfaces

- Interfaces de usuario

Req 40 El sistema visualizara al usuario una pantalla de inicio en la cual el mismo deberá de validar su login y password para poder ingresar a las funciones del sistema, mostrándole la principal función la cual es gestionar consumos del cliente donde en la parte superior de la pantalla un menú con las diferentes funciones restantes del sistema

- Interfaces de hardware

No se han definido

- Interfaces de software

Req 41 El sistema operativo en el que operara el software tendrá que ser obligatoriamente Windows NT, Windows XP o Windows 7.

Req 42 La herramienta que se utilizara para la BD definitivamente será PostgreSQL 8.2.6

Req 43 Eclipse será la herramienta que se utilizara para el desarrollo del sistema

- Interfaces de comunicación

No se han definido.

II.1.3.4.3. Requisitos comunes de los interfaces

- Requisitos de rendimiento

Req 44 El tiempo de respuesta en las operaciones que se utilicen en el sistema por el usuario tendrá un tiempo de espera menor a 10 segundos

- Seguridad

Req 45 El acceso al sistema se realizara previa validación de usuario y contraseña comprobado en la base de datos para el posterior uso del software

Req 46 El sistema contara de dos tipos de usuarios a los cuales se les permitirá únicamente el acceso a las funciones que le correspondan

- Fiabilidad

Req 47 La taza de error en las operaciones realizadas por el usuario deberán de ser de un 0% demostrando el éxito de los procesos

Req 48 En caso de algún error en la introducción de los datos en algunos procesos el sistema avisara mediante mensajes que los valores introducidos son incorrectos

- Disponibilidad

Req 49 La accesibilidad del sistema será de un 100%, que dependerá del uso frecuente que el usuario le dé al software

- Mantenibilidad

Req 50 El mantenimiento del sistema será el mínimo

- Portabilidad

Por las herramientas utilizadas y el entorno elegido, la herramienta no es portable a otros sistemas operativos.

II.1.3.4.4. Requerimientos Externos

- La información que se almacena no debe ser accesible por personas ajenas a la Empresa.
- No se deberá poder tener al acceso al Sistema desde fuera de la Red Local de la Empresa.

II.1.4. Marco Teórico

II.1.4.1. Historia del Software

En el transcurso de la carrera los estudiantes fueron testigos de los grandes y sorprendentes avances que la tecnología ha desarrollado en los últimos años. El estudiar la especialidad de Licenciatura en Informática mantiene estrecha relación junto con el desarrollo.

Estos avances que se desarrollaron tan rápido en la realidad, cada uno de ellos refleja investigaciones de mucho tiempo. Las nuevas tecnologías demuestran con hechos y resultados lo importante que son actualmente; convirtiéndose en estándares para poder competir en el mundo real.

El factor más relevante en el desarrollo de tecnologías ha sido el Software, ya que ha facilitado y agilizado varios procesos. Mas allá de convertirse en una característica primordial que deben tener las Organizaciones, es un hecho el que tengan que implementarlas para ser las mejores organizaciones a nivel mundial.

Este trabajo presenta el empiezo, desarrolló y los distintos rumbos del Software actualmente.

PRIMERA ERA

Durante los primeros años de la era de la computadora, el software se contemplaba como un añadido. Desde entonces el campo se ha desarrollado tremendamente. La programación de computadoras era un arte de andar por casa para el que existían pocos métodos sistemáticos. El desarrollo del software se realizaba virtualmente sin ninguna planificación, hasta que los planes comenzaron a descalabrarse y los costos a correr. Los programadores trataban de hacer las cosas bien, y con un esfuerzo heroico, a menudo salían con éxito. Los problemas a ser resueltos eran principalmente de una naturaleza técnica, el énfasis estaba en expresar algoritmos conocidos eficazmente en algún lenguaje de programación.

En estos primeros años lo normal era que el hardware fuera de propósito general. Por otra parte, el software se diseñaba a medida para cada aplicación y tenía una distribución relativamente pequeña. El software como producto estaba en su infancia. La mayoría del software se desarrollaba y era utilizado por la misma persona u organización. La misma persona lo escribía, lo ejecutaba y, si fallaba, lo depuraba. Debido a que la movilidad en el trabajo era baja, los ejecutivos estaban seguros de que esa persona estará allí cuando se encontrara algún error. Debido a este entorno personalizado del software, el diseño era un proceso implícito, realizado en la mente de alguien, y la documentación normalmente no existía.

A lo largo de los primeros años se aprendió mucho sobre la implementación de sistemas informáticos, pero relativamente poco sobre la ingeniería de las computadoras. Sin embargo, se debe reconocer que durante esa era se desarrollaron muchos sistemas informáticos excepcionales. Algunos de ellos todavía siguen siendo utilizados hoy y, por sus características, siguen siendo admirados con toda justicia.

SEGUNDA ERA

La segunda era en la evolución de los sistemas de computadora se extiende desde la mitad de la década de los sesenta hasta finales de los setenta. La multiprogramación y los sistemas multiusuario introdujeron nuevos conceptos de interacción hombre – máquina. Las técnicas interactivas abrieron un nuevo mundo de aplicaciones y nuevos niveles de sofisticación del

hardware y del software. Los sistemas de tiempo real podían recoger, analizar y transformar datos de múltiples fuentes, controlando así los procesos y produciendo salidas en milisegundos en lugar de en minutos. Los avances en los dispositivos de almacenamiento en línea condujeron a la primera generación de sistemas de gestión de bases de datos.

La segunda era se caracterizó también por el establecimiento del software ya se desarrollaba para tener una amplia distribución en un mercado multidisciplinario. Los programas se distribuían para computadoras grandes y para minicomputadoras, a cientos e incluso a miles de usuarios. Los patronos de la industria, del gobierno y de la universidad se aprestaban a desarrollar el mejor paquete de software y ganar así mucho dinero.

Conforme crecía el número de sistemas informáticos, comenzaron a extenderse las bibliotecas de software de computadora. Las casas desarrollaban proyectos en los que se producían programas de decenas de miles de sentencias fuente. Los productos de software comprados al exterior incorporaban cientos de miles de nuevas sentencias. Una nube negra apareció en el horizonte. Todos esos programas, todas esas sentencias fuente tenían que ser corregidos cuando se detectaban fallos, modificados cuando cambiaban los requisitos de los usuarios o adaptados a nuevos dispositivos hardware que se hubieran adquirido. Estas actividades se llamaron colectivamente mantenimiento del software. El esfuerzo gastado en el mantenimiento del software comenzó a absorber recursos en una medida alarmante. Aún peor, la naturaleza personalizada de muchos programas los hacía virtualmente imposibles de mantener.

Había comenzado una crisis del software.

TERCERA ERA

La tercera era en la evolución de los sistemas de computadora comenzó a mediados de los años setenta y continuó más allá de una década. El sistema distribuido, múltiples computadoras, cada una ejecutando funciones concurrentemente y comunicándose con alguna otra, incrementó notablemente la complejidad de los sistemas informáticos. Las

redes de área local y de área global, las comunicaciones digitales de alto ancho de banda y creciente demanda de acceso instantáneo a los datos, supusieron una fuerte presión sobre los desarrolladores del software. Aún más, los sistemas y el software que lo permitían continuaron residiendo dentro de la industria y de la academia. El uso personal era extraño.

La conclusión de la tercera era se caracterizó por la llegada y amplio uso de los microprocesadores. El microprocesador ha producido un extenso grupo de productos inteligentes, desde productos inteligentes, desde automóviles hasta hornos microondas, desde robots industriales a equipos de diagnóstico de suero sanguíneo, pero ninguno ha sido más importante que la computadora personal. En menos de una década, las computadoras llegarán a ser fácilmente accesibles al público.

CUARTA ERA

La cuarta era de la evolución de sistemas informáticos se aleja de las computadoras individuales y de los programas de computadoras, dirigiéndose al impacto colectivo de las computadoras individuales y de los programas de computadoras, dirigiéndose al impacto colectivo de las computadoras y del software. Potentes máquinas personales controladas por sistemas operativos sofisticados, en redes globales y locales, acompañadas por aplicaciones de software avanzadas se han convertido en la norma. Las arquitecturas informáticas están cambiando de entornos centralizados de grandes computadoras a entornos descentralizados cliente/servidor. Las redes de información en todo el mundo proporcionan una infraestructura que iguala a expertos y políticos en pensar sobre una super utopía de información y una conexión del ciberespacio. De hecho internet se puede observar como un software al que pueden acceder usuarios individuales.

La industria del software ya es la cuna de la economía del mundo. Las decisiones tomadas por gigantes de la industria tales como Microsoft arriesgan billones de dólares. A medida que la cuarta generación progresa, han comenzado a surgir nuevas tecnologías. Las tecnologías orientadas a objetos están desplazando rápidamente los enfoques de desarrollo de software más convencionales en muchas áreas de aplicaciones. Aunque las predicciones

de las computadoras de quinta generación continúan eludiéndonos, las técnicas de cuarta generación para el desarrollo del software están cambiando en forma en que la comunidad del software construye programas informáticos. Los sistemas expertos y el software de inteligencia artificial han salido del laboratorio para entrar en aplicaciones prácticas de una gran variedad de problemas del mundo real. El software de redes neuronales artificiales junto con la aplicación de lógica difusa ha abierto posibilidades excitantes para el reconocimiento de patrones y habilidades de procesamiento de información de carácter humano. La programación de realidad virtual y los sistemas multimedia ofrecen formas radicalmente diferentes de comunicar información al usuario final. Los algoritmos genéricos ofrecen el potencial para el software que reside dentro de las computadoras biológicas masivamente en paralelo.

II.1.4.2. Metodología RUP para Desarrollo de Software

RUP es un proceso para el desarrollo de un proyecto de un software que define claramente quien, cómo, cuándo y qué debe hacerse en el proyecto. Como 3 características esenciales está dirigido por los Casos de Uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere, está centrado en la arquitectura: que Relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden, y es iterativo e incremental: donde divide el proyecto en mini proyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada.

Como filosofía RUP maneja 6 principios clave:

Adaptación del proceso

El proceso deberá adaptarse a las características propias de la organización. El tamaño del mismo, así como las regulaciones que lo condicionen, influirán en su diseño específico.

También se deberá tener en cuenta el alcance del proyecto.

Balancear prioridades

Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.

Colaboración entre equipos

El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados, etc.

Demostrar valor iterativamente

Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.

Elevar el nivel de abstracción

Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas por nombrar algunos. Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.

Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción.

El ciclo de vida de RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

En las iteraciones de cada fase se hacen diferentes esfuerzos en diferentes actividades

Inicio: Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto.

Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.

Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.

Transición: se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

II.1.4.3. Los Diagramas del Lenguaje Unificado de Modelado

UML por sus siglas en Inglés Unified Modeling Language (Lenguaje Unificado de Modelado) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante resaltar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software entregando gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, sólo se diagrama la realidad de una

utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

Los Diagramas

Un diagrama es la representación gráfica de un conjunto de elementos, visualizando la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones). Los diagramas se dibujan para visualizar el sistema desde diferentes perspectivas, de forma que un diagrama es una proyección de un sistema. En teoría un diagrama puede contener cualquier combinación de elementos y relaciones, sin embargo en la práctica sólo surge un pequeño número de combinaciones, a continuación se describirán los diagramas más usados y los que se emplearán en el desarrollo del Proyecto.

Diagrama de Actividad.- Muestra los procesos de alto nivel de la organización. Incluye flujo de datos, o un modelo de la lógica compleja dentro del sistema.

Diagrama de Clases.- Exhibe una colección de elementos del modelo estático, tales como clases y tipos, sus contenidos y sus relaciones.

Diagrama de Componentes.- Muestra los componentes de una aplicación, sistema o empresa. Se ven los componentes, sus interrelaciones, interacciones y sus interfaces públicas.

Diagrama de Despliegue.- Exhibe la ejecución de la arquitectura del sistema. Incluye nodos, ambientes operativos sea de hardware o software, así como las interfaces que las conectan.

Diagrama de Secuencia.- Modela la secuencia lógica, a través del tiempo, de los mensajes entre las instancias.

Diagrama de Estado.- Es una condición durante la vida de un objeto, de forma que cuando dicha condición se satisface se lleva a cabo alguna acción o se espera por un evento.

Diagramas de Casos de Uso.- No forman parte de la llamada Fase de Diseño, sino parte de la fase de Análisis, respondiendo el interrogante ¿Qué?, de forma que al ser parte del análisis ayuda a describir que es lo que el sistema debe hacer.

II.1.4.4. Programación Orientada a Objetos

La Programación Orientada a Objetos (P.O.O.) tal vez uno de los métodos de programación más usada hoy en día debido a su facilidad de entendimiento, pero ¿qué es lo que hace que este tipo de programación sea tan usado? Se diría que es la simplicidad del mismo al ver todo como objetos, tal como sucede en el Mundo real en el que estamos acostumbrados a que cada cosa es un objeto y esto es lo que permite diferencias entre objetos dentro de las clases que se usa al emplear este tipo de programación como por ejemplo, permite diferenciar un objeto auto de un objeto casa porque estos estarían en clases distintas.

II.1.4.5. Lenguaje de Programación

Un lenguaje de programación es una construcción mental del ser humano para expresar programas. Está constituido por un grupo de reglas gramaticales, un grupo de símbolos utilizables, un grupo de términos con sentido único y una regla principal que resume las demás. Para que ésta construcción mental sea operable en un computador debe existir otro programa que controle la validez o no de lo escrito. A éste se le llama traductor.

Los lenguajes pueden ser de alto o bajo nivel. En los de bajo nivel las instrucciones son simples y cercanas al funcionamiento de la máquina, como por ejemplo el código máquina y el ensamblador. En los lenguajes de alto nivel hay un alto grado de abstracción y el lenguaje es más próximo a los humanos.

Los programas escritos en un lenguaje deben ser "entendidos" por los circuitos físicos de la máquina para poder ser ejecutados. Según que esta traducción se realice produciendo previamente o no una versión independiente de la herramienta utilizada los lenguajes se

llaman *compilados* o *interpretados*. En el primer caso a la versión independiente producida se le conoce como código o programa ejecutable, no es entendible para el usuario y usualmente se le identifica con la extensión de nombre .exe (en sistemas Windows). En el segundo caso no se construye otra versión razón por la cual ha de distribuirse el programa original llamado código o programa fuente y el usuario debe poseer también la herramienta con la cual se ha programado y que interpreta ese código.

II.1.4.6. Java

Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

Las aplicaciones Java están típicamente compiladas en un *bytecode*, aunque la compilación en código máquina nativo también es posible. En el tiempo de ejecución, el *bytecode* es normalmente interpretado o compilado a código nativo para la ejecución, aunque la ejecución directa por hardware del *bytecode* por un procesador Java también es posible.

¿Por qué Java?

Entre las razones por las cuales se ha optado por este lenguaje de programación para el desarrollo del software anteriormente mencionado podríamos señalar las siguientes:

- El grado de experiencia con el que se cuenta al crear código fuente en este Lenguaje de Programación.
- La gran cantidad de ayuda e información que se logra encontrar en libros, internet, etc. sobre este Lenguaje de Programación.
- Porque es un Lenguaje de Programación Orientado a Objetos.

- Su independencia de Sistema Operativo, que por ser así el Software diseñado podría ser usando tanto en Windows (comprobado) como Linux (no comprobado).

II.1.4.7. El Modelo Entidad – Relación

El modelo entidad-relación es un estándar creado con el fin de contribuir al diseño de Base de Datos relacionales, cuyos principales componentes como su nombre lo indica son las entidades y relaciones.

Entidad.- Se trata de un objeto del que se recoge información de interés de cara a la base de datos, estas pueden ser fuertes, que no dependen de otras entidades para existir y débiles que dependen de otras entidades sino las mismas no tienen sentido.

Dentro de las entidades se encuentran los atributos, los cuales serían los datos a guardarse, estos son creados en base a los requerimientos del usuario.

Relación.- Es la asociación entre dos o más entidades, esta recibe un nombre acorde a las entidades que se están enlazando. Existen tres tipos de relaciones:

- 1:1. Uno a uno, a una ocurrencia de una entidad le corresponderá como máximo una sola ocurrencia de la entidad asociada.
- 1:N. Uno a muchos, a cada ocurrencia de una entidad pueden haber varias ocurrencias de la entidad asociada.
- N:M. Muchos a muchos. a cada ocurrencia de una entidad puede tener varias ocurrencias de la entidad asociada y viceversa.

II.1.4.8. Base de Datos Relacional

Una base de datos relacional es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores, y en donde todas las operaciones de la base de datos operan sobre estas tablas.

Estas bases de datos son percibidas por los usuarios como una colección de relaciones normalizadas de diversos grados que varían con el tiempo.

El modelo relacional representa un sistema de bases de datos en un nivel de abstracción un tanto alejado de los detalles de la máquina subyacente, de la misma manera como, por ejemplo, un lenguaje del tipo de PL/1 representa un sistema de programación con un nivel de abstracción un tanto alejado de los detalles de la máquina subyacente. De hecho, el modelo relacional puede considerarse como un lenguaje de programación más bien abstracto, orientado de manera específica hacia las aplicaciones de bases de datos.

II.1.4.9. PostgreSQL

PostgreSQL es un servidor de base de datos relacional orientada a objetos de libre distribución, liberado bajo la licencia BSD.

Como muchos otros proyectos de Código Abierto, 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.

¿Por qué PostgreSQL?

- Tienes licencia libre BSD
- Garantía de integridad en los datos
- Escalabilidad y rendimiento bajo grandes cargas de trabajo
- Experiencia con la que se cuenta al usar este motor de Base de Datos

II.1.4.10. Arquitectura Cliente – Servidor

La arquitectura consiste básicamente que dentro de un programa, el cliente realice peticiones a otro programa, el servidor. Aunque esta idea se puede aplicar a programas que

se ejecutan sobre una sola computadora es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema. La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. Los tipos específicos de servidores incluyen los servidores Web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

¿En qué se aplicará esta arquitectura?

Este tipo de arquitectura podrá ser aplicado al implementar un servidor de Base de Datos, el cual será PostgreSQL, el mismo que podrá ser configurado en una PC para que la misma se convierta en un servidor de Base de Datos, dando la facilidad de que varios usuarios se puedan conectar a este servidor como clientes desde distintos puntos de la red LAN existente dentro de la Empresa.

II.1.5. Requerimientos del Sistema

II.1.5.1. Análisis

II.1.5.1.1. Casos de Uso

II.1.5.1.1.1. Casos de Uso del Negocio

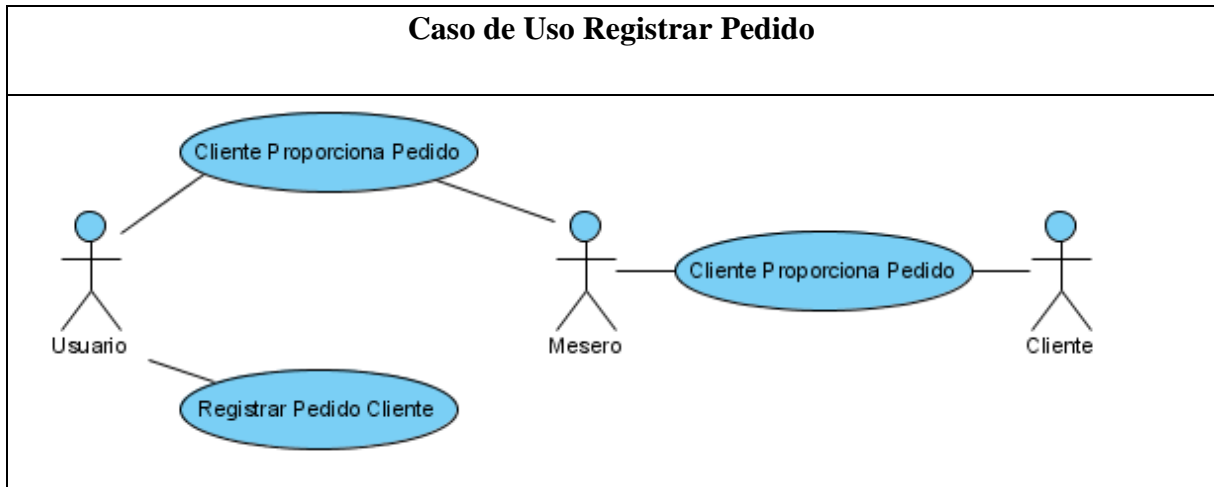


Figura 2 Caso de Uso Registrar Pedido

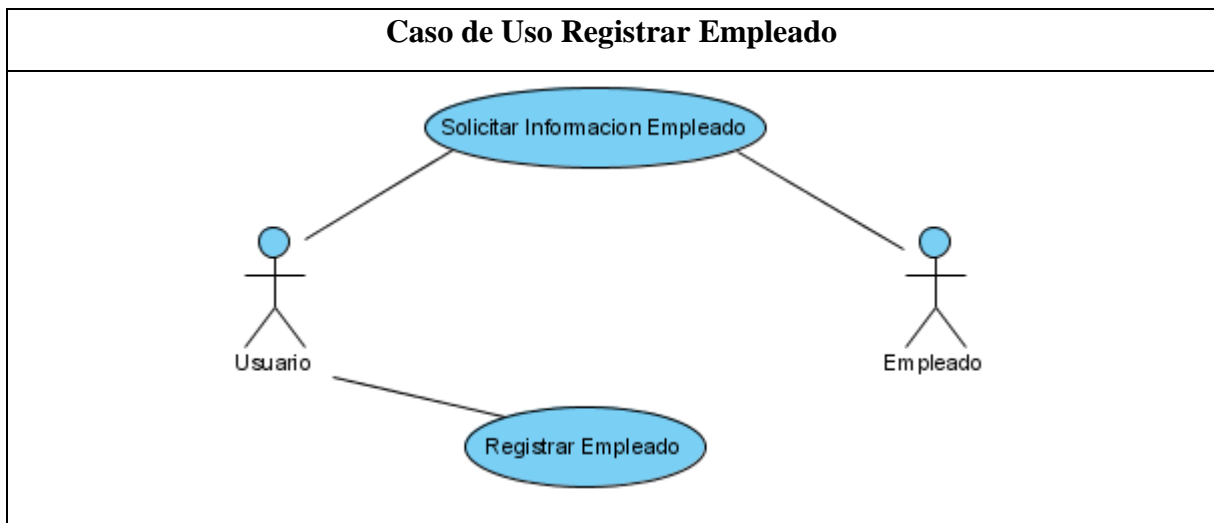


Figura 3 Caso de Uso Registrar Empleado



Figura 4 Caso de Uso Registrar Proveedor

II.1.5.1.1.2. Casos de Uso del Sistema

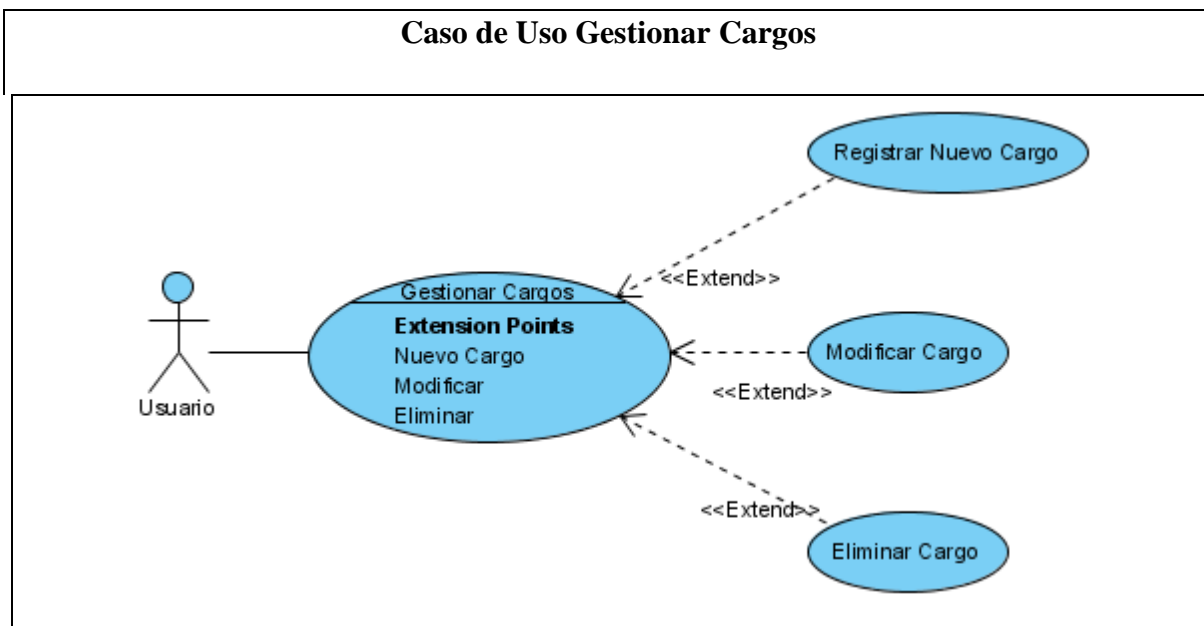


Figura 5 Caso de Uso Gestionar Cargos

| Caso de Uso Gestionar Cargos |
|--|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Cargos que serán asignados a los empleados dentro del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Empleados y luego la opción “Gestionar Cargos” para poder acceder a una lista con el detalle de los cargos registrados. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Cargos”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Cargos” muestra una lista de los “Cargos Registrados” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Cargo. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Cargo” con el manejador de la pantalla “Nuevo Cargo”.</p> <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos del Cargo que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Cargo” con el manejador de la pantalla “Modificar Cargo”.</p> <p>P3. El usuario elegirá la opción “Eliminar” para eliminar los datos del Cargo que ha sido seleccionado. Al elegir esta opción se ejecutará el</p> |

caso de uso “Eliminar Cargo”.

P4. El usuario elegirá la opción “Atrás” para cerrar esta pantalla.

Excepciones:

E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los cargos registradas no se activarán las opciones Modificar y Eliminar.

Caso de Uso Registrar Nuevo Cargo

Actores: Usuario.

Tipo: Extensión.

Propósito: Registrar los datos de un nuevo Cargo en la base de datos del Sistema.

Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Cargo” para guardar los datos ingresados en el formulario de registro.

Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Cargos” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Cargo” con su respectivo formulario de registro.

Flujo principal:

La ventana “Nuevo Cargo”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo cargo.

P1. En la ventana “Nuevo Cargo” el usuario ingresará los datos del

| |
|---|
| <p>Cargo a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Cargos”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|--|
| Caso de Uso Modificar Cargo |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Cargo Registrado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del cargo que se ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Cargos” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Cargo” con un formulario que contendrá los datos actuales del cargo ejecutándose el manejador del caso de uso “Modificar Cargo”. |

| |
|---|
| <p>Flujo principal:</p> <p>La ventana “Modificar Cargo” muestra los datos del cargo que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Cargo” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del cargo, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Cargos”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|---|
| Caso de Uso Eliminar Cargo |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Eliminar datos de un Cargo Registrado. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los |

| |
|--|
| datos del cargo que ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Cargos” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del cargo. |
| <p>Flujo principal:</p> <p>El manejador “Eliminar Cargo” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del cargo de la base de datos del Sistema.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro del cargo, ejecutado por el manejador “Eliminar Cargo”.</p> <p>S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del cargo, ejecutado por el manejador “Eliminar Cargo”.</p> |

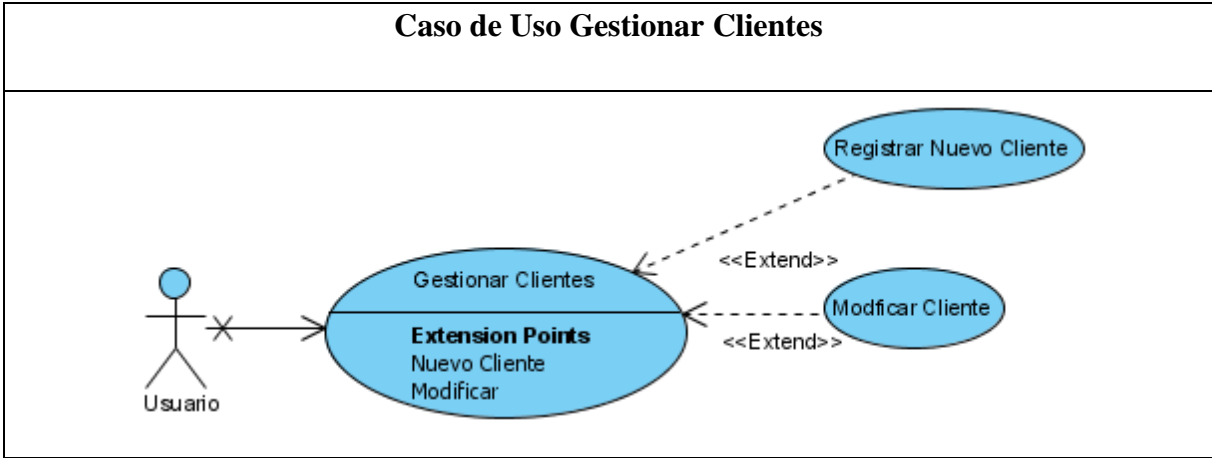


Figura 6 Caso de Uso Gestionar Clientes

| Caso de Uso Gestionar Cliente |
|---|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Cliente que están registrados dentro del Sistema que son respectivos consumidores del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Cliente ejecutándose la opción Gestionar “Clientes” para poder acceder a una lista con los detalles de los clientes registrados. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Clientes”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Cliente” muestra una lista de los “Clientes Registrados” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Cliente. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Cliente” con el manejador de la pantalla “Nuevo Cliente”.</p> <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos del Cliente que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Cliente” con el manejador de la pantalla “Modificar Cliente”.</p> |

Excepciones:

E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los clientes registradas no se activarán la opción Modificar.

| Caso de Uso Registrar Nuevo Cliente |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Cliente en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Cliente” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Clientes” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Cliente” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nuevo Cliente”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo cliente.</p> <p>P1. En la ventana “Nuevo Cliente” el usuario ingresará los datos del Cliente a registrarse.</p> |

| |
|---|
| <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Clientes”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|---|
| <p>Caso de Uso Modificar Cliente</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Modificar los datos de un Cliente Registrado.</p> |
| <p>Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del cliente que se ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Clientes” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Cliente” con un formulario que contendrá los datos actuales del cliente ejecutándose el manejador del caso de uso “Modificar Cliente”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Cliente” muestra los datos del cliente que se desea</p> |

modificar en sus campos correspondientes.

P1. En la ventana “Modificar Cliente” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.

P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del cliente, se muestra el mensaje “Datos modificados correctamente”.

Flujos secundarios:

S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Clientes”.

Excepciones:

E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.

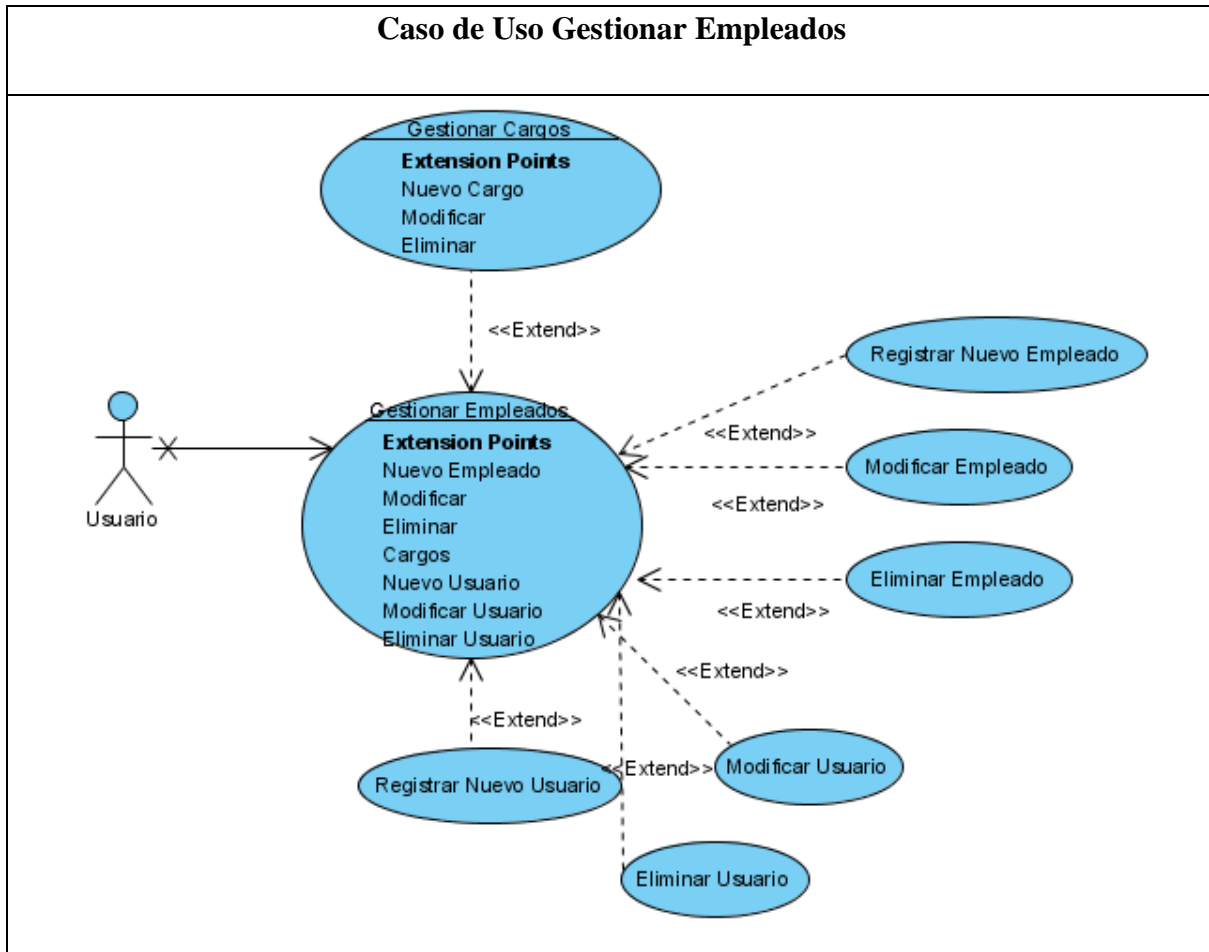


Figura 7 Caso de Uso Gestionar Empleados

| Caso de Uso Gestionar Empleados |
|--|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Empleados que están registrados dentro del Sistema que trabajan dentro del Restaurante. |

Resumen: El Usuario elegirá la Pestaña Empleados ejecutándose la opción “Empleados” para poder acceder a una lista con los detalles de los clientes registrados.

Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Empleados”.

Flujo principal:

El manejador “Gestionar Empleados” muestra una lista de los “Empleados Registrados” en la base de datos del Sistema.

- P1.** El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Empleado. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Empleado” con el manejador de la pantalla “Nuevo Empleado”.
- P2.** El usuario elegirá la opción “Modificar” para cambiar los datos del Empleado que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Empleado” con el manejador de la pantalla “Modificar Empleado”.
- P3.** El usuario elegirá la opción “Eliminar” para eliminar los datos del Empleado que ha sido seleccionado. Al elegir esta opción se ejecutará el caso de uso “Eliminar Empleado”.
- P4.** El usuario elegirá la opción “Crear Usuario” para registrar los datos de un nuevo Usuario del empleado que haya sido seleccionado de la lista. Al elegir esta opción se ejecutara el caso de uso “Registrar Nuevo Usuario” con el manejador de la pantalla “Nuevo Usuario”.
- P5.** El usuario elegirá la opción “Modificar Usuario” para cambiar los

datos del Usuario que ha sido seleccionado. Al elegir esta opción se ejecutará el caso de uso “Modificar Usuario” con el manejador de la pantalla “Modificar Usuario”.

P6. El usuario elegirá la opción “Gestionar Cargos” para ingresar a la pantalla “Gestionar Cargos”. Al elegir esta opción se ejecutara el caso de uso Gestionar Cargos con su respectivo manejador.

Excepciones:

E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los empleados registradas no se activarán las opciones Modificar, Eliminar, Modificar Usuario y Crear Usuario.

| Caso de Uso Registrar Nuevo Empleado |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Empleado en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Empleado” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Empleados” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Empleado” con su respectivo formulario de registro. |

| |
|--|
| <p>Flujo principal:</p> <p>La ventana “Nuevo Empleado”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo empleado.</p> <p>P1. En la ventana “Nuevo Empleado” el usuario ingresará los datos del Empleado a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Empleados”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|---|
| Caso de Uso Modificar Empleado |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Empleado Registrado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del empleado que se ha seleccionado. |

| |
|---|
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Empleados” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Empleado” con un formulario que contendrá los datos actuales del empleado ejecutándose el manejador del caso de uso “Modificar Empleado”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Empleado” muestra los datos del empleado que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Empleado” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del empleado, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Empleados”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |
| <p>Caso de Uso Eliminar Empleado</p> |
| <p>Actores: Usuario.</p> |

| |
|---|
| Tipo: Extensión. |
| Propósito: Eliminar datos de un Empleado Registrado. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del empleado que ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Empleado” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del empleado. |
| Flujo principal: El manejador “Eliminar Empleado” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del empleado de la base de datos del Sistema. |
| Flujos secundarios: S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro del empleado, ejecutado por el manejador “Eliminar Empleado”. S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del empleado, ejecutado por el manejador “Eliminar Empleado”. |

| Caso de Uso Registrar Nuevo Usuario |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Usuario en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Usuario” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Empleados” y seleccionado la opción “Crear”. Se despliega la pantalla “Nuevo Usuario” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nuevo Usuario”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo usuario para el empleado seleccionado previamente de la lista.</p> <p>P1. En la ventana “Nuevo Usuario” el usuario ingresará los datos del nuevo Usuario a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla</p> |

| |
|--|
| anterior “Gestionar Empleados”. |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|--|
| Caso de Uso Modificar Usuario |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Usuario Registrado de un Empleado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del usuario del empleado que se ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Empleados” y seleccionado la opción “Modificar Usuario”. Se despliega la pantalla “Modificar Usuario” con un formulario que contendrá los datos actuales del usuario del empleado ejecutándose el manejador del caso de uso “Modificar Usuario”. |
| <p>Flujo principal:</p> <p>La ventana “Modificar Usuario” muestra los datos del usuario del empleado que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Usuario” el usuario podrá cambiar todos</p> |

| |
|---|
| <p>los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del usuario del empleado, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Empleados”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

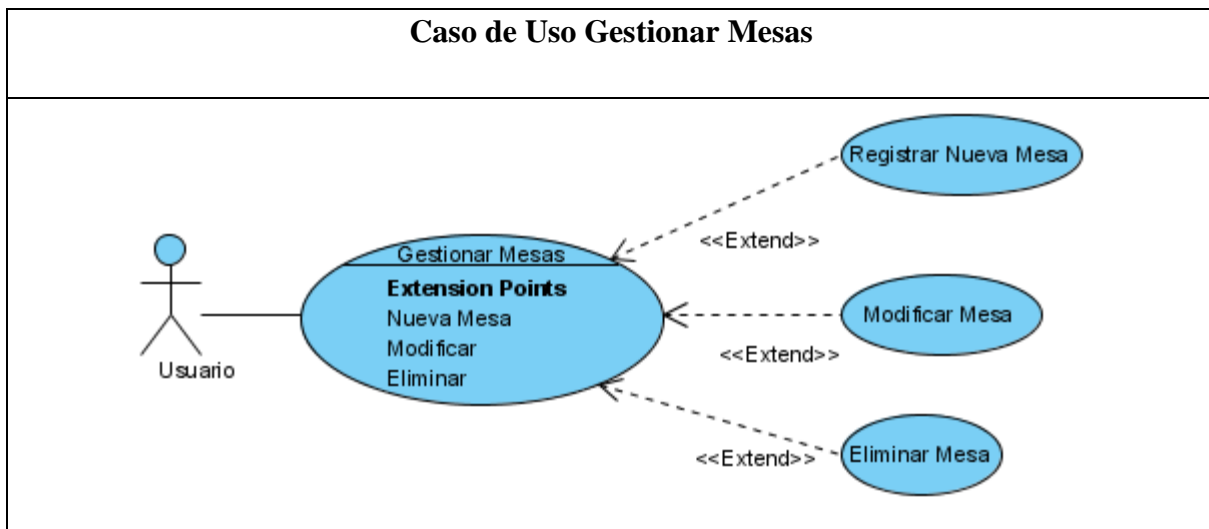


Figura 8 Caso de Uso Gestionar Mesas

| Caso de Uso Gestionar Mesas |
|--|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de las diferentes Mesas que están registradas dentro del Sistema que son respectivas del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Pedidos ejecutándose la pestaña “Pedidos” para poder acceder a una lista con los detalles de los pedidos registrados, en donde se encontrara la opción “Gestionar Mesas” para poder acceder a una lista con los detalles de las mesas registradas. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Mesas”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Mesas” muestra una lista de los “Mesas Registradas” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nuevo” para registrar los datos de una nueva Mesa. Al elegir esta opción se ejecutará el caso de uso “Registrar Nueva Mesa” con el manejador de la pantalla “Nueva Mesa”.</p> <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos de la Mesa que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Mesa” con el manejador de la pantalla “Modificar Mesa”.</p> <p>P3. El usuario elegirá la opción “Eliminar” para eliminar los datos de la Mesa que ha sido seleccionada. Al elegir esta opción se ejecutará el</p> |

| |
|---|
| caso de uso “Eliminar Mesa”. |
| <p>Excepciones:</p> <p>E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de las mesas registradas no se activarán las opciones Modificar y Eliminar.</p> |

| |
|---|
| Caso de Uso Registrar Nueva Mesa |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de una nueva Mesa en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nueva Mesa” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Mesas” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nueva Mesa” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nueva Mesa”, muestra al usuario un formulario con los datos a introducirse para el registro de una nueva mesa.</p> <p>P1. En la ventana “Nueva Mesa” el usuario ingresará los datos de la Mesa a registrarse.</p> |

| |
|---|
| <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Mesas”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|---|
| <p>Caso de Uso Modificar Mesa</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Modificar los datos de una Mesa Registrada.</p> |
| <p>Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos de la mesa que se ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Mesas” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Mesa” con un formulario que contendrá los datos actuales de la mesa ejecutándose el manejador del caso de uso “Modificar Mesa”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Mesa” muestra los datos de la mesa que se desea</p> |

| |
|---|
| <p>modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Mesa” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario de la mesa, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Mesas”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|---|
| Caso de Uso Eliminar Mesa |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Eliminar datos de una Mesa Registrada. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos de la mesa que ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Mesas” y |

| |
|---|
| <p>seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos de la mesa.</p> |
| <p>Flujo principal:</p> <p>El manejador “Eliminar Mesa” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro de la mesa de la base de datos del Sistema.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro de la mesa, ejecutado por el manejador “Eliminar Mesa”.</p> <p>S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro de la mesa, ejecutado por el manejador “Eliminar Mesa”.</p> |

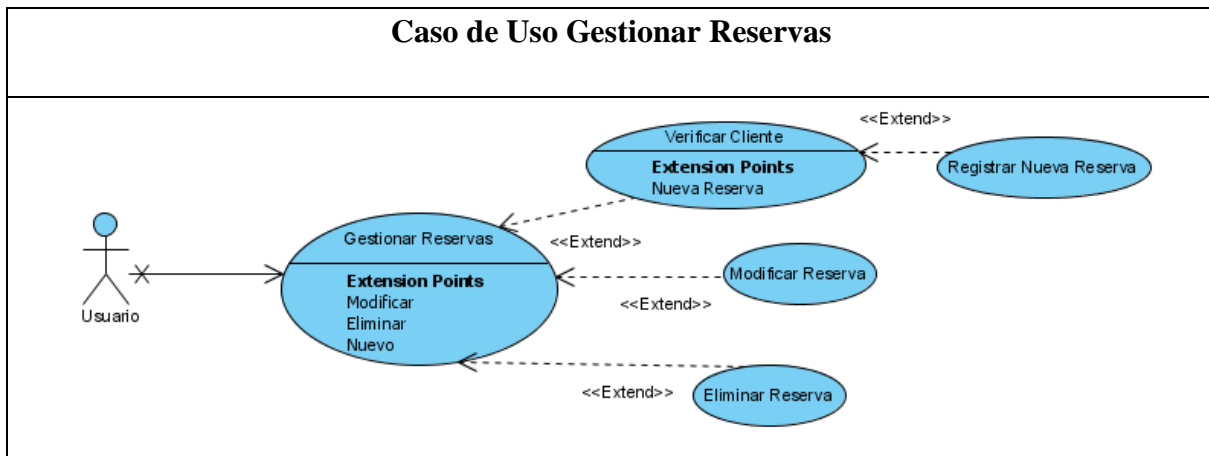


Figura 9 Caso de Uso Gestionar Reservas

| Caso de Uso Gestionar Reservas |
|--|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Reservas que están registradas dentro del Sistema que son respectivos pedidos de los clientes del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Reservas ejecutándose la pestaña “Reservas” para poder acceder a una lista con los detalles de los reservas registradas. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Reservas”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Reservas” muestra una lista de las “Reservas Registradas” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nueva” para registrar los datos de una nueva Reserva. Al elegir esta opción se ejecutará el caso de uso “Verificar Cliente” con el manejador de la pantalla “Verificar Cliente”.</p> <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos de la Reserva que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Reserva” con el manejador de la pantalla “Modificar Reserva”.</p> <p>P3. El usuario elegirá la opción “Eliminar” para eliminar los datos de la Reserva que ha sido seleccionado. Al elegir esta opción se ejecutará</p> |

el caso de uso “Eliminar Reserva”.

P4. El usuario elegirá la opción “Confirmar” para confirmar los datos de la Reserva que ha sido seleccionada.

Excepciones:

E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de las reservas registradas no se activarán las opciones Modificar y Eliminar.

Caso de Uso Verificar Cliente

Actores: Usuario.

Tipo: Extensión.

Propósito: Mostrar una lista de detalles de los clientes registrados en la base de datos del Sistema.

Resumen: El Usuario seleccionará el botón “Siguiente” de la pantalla “Verificar Cliente” tras haber seleccionado al cliente de la lista, para registrar datos del cliente con la nueva reserva.

Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Reservas” y seleccionado la opción “Nueva Reserva”. Se despliega la pantalla “Verificar Cliente” con su respectiva lista de los clientes del Restaurante.

Flujo principal:

La ventana “Verificar Cliente”, muestra al usuario una lista con los datos

detallados de los clientes a seleccionar.

P1. En la ventana “Verificar Cliente” el usuario seleccionara los datos de un Cliente a registrarse junto con la reserva respectiva.

P2. El usuario pulsará el botón “Siguiete”. Al elegir esta opción se ejecutará el caso de uso “Registrar Nueva Reserva” con el manejador de la pantalla “Nueva Reserva”.

Flujos secundarios:

S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Reservas”.

S2. El usuario pulsara el botón “Registrar Nuevo” para ejecutar el caso de uso Registrar Nuevo Cliente con el manejador de la pantalla “Nuevo Cliente”

| Caso de Uso Registrar Nueva Reserva |
|---|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de una nueva Reserva en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nueva Reserva” para guardar los datos seleccionados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Verificar Cliente” y |

| |
|--|
| <p>seleccionado la opción “Siguiete”. Se despliega la pantalla “Nueva Reserva” con su respectivo formulario de registro.</p> |
| <p>Flujo principal:</p> <p>La ventana “Nueva Reserva”, muestra al usuario un formulario con los datos a seleccionarse para el registro de una nueva reserva.</p> <p>P1. En la ventana “Nueva Reserva” el usuario seleccionara los datos de la Nueva Reserva a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos seleccionados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Reservas”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no selecciona todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|---|
| Caso de Uso Modificar Reserva |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de una Reserva Registrada. |

| |
|---|
| <p>Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos de la reserva que se ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Reservas” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Reserva” con un formulario que contendrá los datos actuales de la reserva ejecutándose el manejador del caso de uso “Modificar Reserva”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Reserva” muestra los datos del proveedor que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Reserva” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario de la reserva, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Reservas”.</p> |

| |
|--|
| Caso de Uso Eliminar Reserva |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Eliminar datos de una Reserva Registrada. |

| |
|--|
| <p>Resumen: El usuario seleccionará la opción “Eliminar Reserva” para eliminar los datos del Proveedor que ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Reservas” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos de la reserva.</p> |
| <p>Flujo principal:</p> <p>El manejador “Eliminar Reserva” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro de la reserva de la base de datos del Sistema.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro de la reserva, ejecutado por el manejador “Eliminar Reserva”.</p> <p>S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro de la reserva, ejecutado por el manejador “Eliminar Reserva”.</p> |

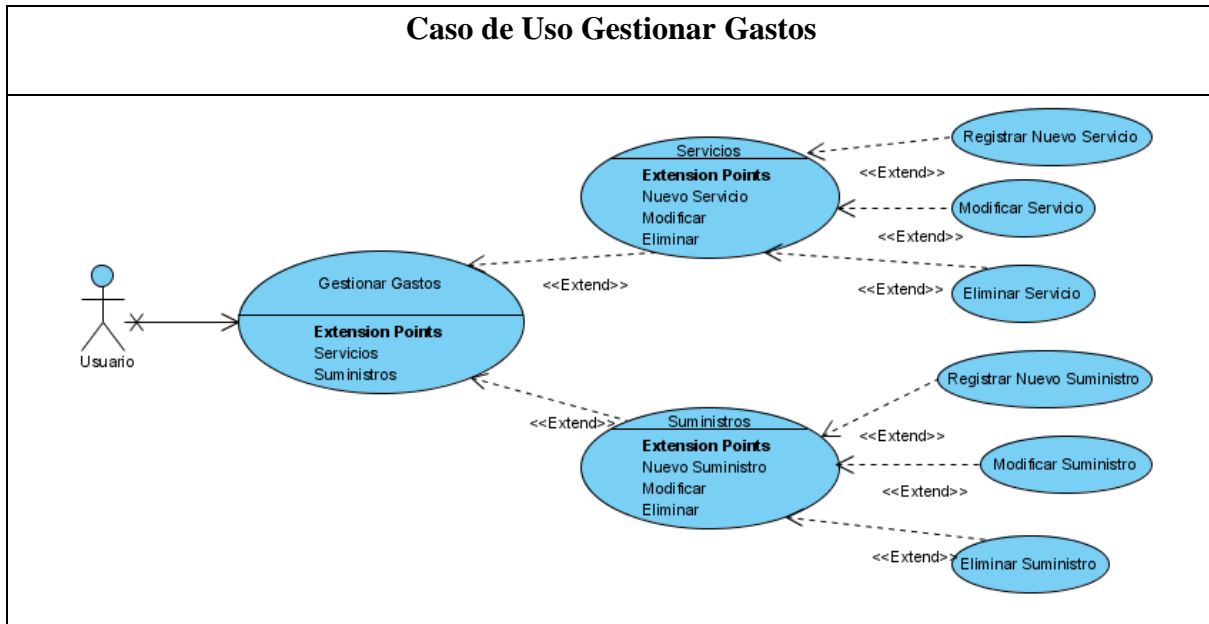


Figura 10 Caso de Uso Gestionar Gastos

| Caso de Uso Gestionar Gastos |
|---|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de las diferentes Gastos que están registradas dentro del Sistema que son respectivas del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Registrar Gastos ejecutándose la opción “Gestionar Gastos” para poder acceder a una lista con los detalles de los gastos registrados, en donde se seleccionara la opción “Suministro” o “Servicios” para poder acceder a una lista con los detalles de los gastos registradas. |

Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Gastos”.

Flujo principal:

El manejador “Gestionar Gastos” muestra una lista de los “Gastos Registrados” en la base de datos del Sistema de acuerdo al tipo de gasto seleccionado.

- P1.** El usuario elegirá la opción “Servicios” para listar los datos en una lista. Al elegir esta opción se ejecutará el caso de uso “Gestionar Servicios”.
- P2.** El usuario elegirá la opción “Suministros” para listar los datos en una lista. Al elegir esta opción se ejecutará el caso de uso “Gestionar Suministros”.
- P3.** El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Servicio al seleccionar la opción Servicios. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Servicio” con el manejador de la pantalla “Nuevo Servicio”.
- P4.** El usuario elegirá la opción “Modificar” para cambiar los datos del Servicio que ha sido seleccionado de la lista al haberse seleccionado la opción Servicios. Al elegir esta opción se ejecutará el caso de uso “Modificar Servicio” con el manejador de la pantalla “Modificar Servicio”.
- P5.** El usuario elegirá la opción “Eliminar” para eliminar los datos del Servicio que ha sido seleccionada de la lista al haberse seleccionado la opción Servicios. Al elegir esta opción se ejecutará el caso de uso “Eliminar Servicio”.

P6. El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Suministro al seleccionar la opción Suministros. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Suministro” con el manejador de la pantalla “Nuevo Suministro”.

P7. El usuario elegirá la opción “Modificar” para cambiar los datos del Suministro que ha sido seleccionado de la lista al haberse seleccionado la opción Suministro. Al elegir esta opción se ejecutará el caso de uso “Modificar Suministro” con el manejador de la pantalla “Modificar Suministro”.

P8. El usuario elegirá la opción “Eliminar” para eliminar los datos del Suministro que ha sido seleccionada de la lista al haberse seleccionado la opción Suministros. Al elegir esta opción se ejecutará el caso de uso “Eliminar Suministro”.

Excepciones:

E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los gastos registrados no se activarán las opciones Modificar y Eliminar.

| |
|--|
| Caso de Uso Registrar Nuevo Suministro |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Suministro en la base de datos del Sistema. |

| |
|--|
| <p>Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Suministro” para guardar los datos ingresados en el formulario de registro.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Suministros” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Suministro” con su respectivo formulario de registro.</p> |
| <p>Flujo principal:</p> <p>La ventana “Nuevo Suministro”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo suministro.</p> <p>P1. En la ventana “Nuevo Suministro” el usuario ingresará los datos del Suministro a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Gastos”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| Caso de Uso Modificar Suministro |
|---|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Suministro Registrado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del suministro que se ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Suministros” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Suministro” con un formulario que contendrá los datos actuales del suministro ejecutándose el manejador del caso de uso “Modificar Suministro”. |
| <p>Flujo principal:</p> <p>La ventana “Modificar Suministro” muestra los datos del suministro que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Suministro” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del suministro, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Gastos”.</p> |

Excepciones:

- E1.** El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.

Caso de Uso Eliminar Suministro

Actores: Usuario.

Tipo: Extensión.

Propósito: Eliminar datos de un Suministro Registrado.

Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del suministro que ha seleccionado.

Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Suministros” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del suministro.

Flujo principal:

El manejador “Eliminar Suministro” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del suministro de la base de datos del Sistema.

Flujos secundarios:

- S1.** El usuario pulsará el botón “Sí” de la ventana de confirmación si

está seguro para proceder a eliminar el registro del suministro, ejecutado por el manejador “Eliminar Suministro”.

S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del suministro, ejecutado por el manejador “Eliminar Suministro”.

| Caso de Uso Registrar Nuevo Servicio |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Servicio en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Servicio” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Servicio” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Servicio” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nuevo Servicio”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo servicio.</p> <p>P1. En la ventana “Nuevo Servicio” el usuario ingresará los datos del Servicio a registrarse.</p> |

| |
|---|
| <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Gastos”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|--|
| <p>Caso de Uso Modificar Servicio</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Modificar los datos de un Servicio Registrado.</p> |
| <p>Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del servicio que se ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Servicio” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Servicio” con un formulario que contendrá los datos actuales del servicio ejecutándose el manejador del caso de uso “Modificar Servicio”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Servicio” muestra los datos del servicio que se</p> |

| |
|---|
| <p>desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Servicio” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del servicio, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Gastos”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|---|
| Caso de Uso Eliminar Servicio |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Eliminar datos de un Servicio Registrado. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del servicio que ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar |

Servicios” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del servicio.

Flujo principal:

El manejador “Eliminar Servicio” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del servicio de la base de datos del Sistema.

Flujos secundarios:

S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro del servicio, ejecutado por el manejador “Eliminar Servicio”.

S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del servicio, ejecutado por el manejador “Eliminar Servicio”.

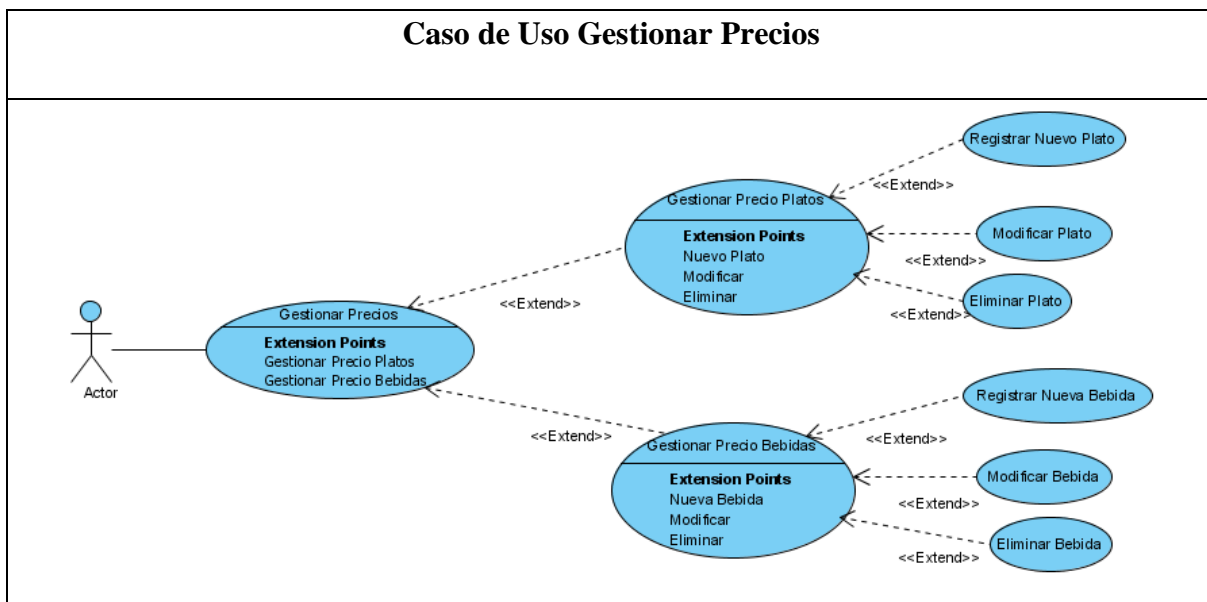


Figura 11 Caso de Uso Gestionar Precios

| Caso de Uso Gestionar Precios |
|---|
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de las diferentes Precios que están registradas o se registraran dentro del Sistema que son respectivas del Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Pedidos ejecutándose la opción Gestionar Pedidos para poder acceder a una lista con los detalles de los pedidos registrados, en donde se seleccionara la opción “Gestionar Precios” para poder acceder a una lista de tipos precios registrados. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Precios”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Precios” muestra una lista de los tipos de precios registrados en la base de datos del Sistema de acuerdo al tipo de precio seleccionado.</p> <p>P1. El usuario seleccionara “Platos” eligiendo a la vez la opción “Aceptar”. Al elegir esta opción se ejecutará el caso de uso “Gestionar Precios Platos” con el manejador de la pantalla “Gestionar Precio Platos”.</p> <p>P2. El usuario seleccionara “Bebidas” eligiendo a la vez la opción “Aceptar”. Al elegir esta opción se ejecutará el caso de uso “Gestionar Precios Bebidas” con el manejador de la pantalla</p> |

“Gestionar Precio Bebidas”.

P3. El usuario elegirá la opción “Cancelar”. Al elegir esta opción se volverá al caso de uso “Gestionar Pedidos” con su respectiva pantalla.

| Caso de Uso Gestionar Precio Platos |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Gestionar el registro de las diferentes Precios Platos que están registradas dentro del Sistema que son respectivas del Restaurante. |
| Resumen: El Usuario elegirá la opción Platos ejecutándose la opción “Gestionar Precio Platos” para poder acceder a una lista con los detalles de los precio de los platos registrados. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Precios Platos”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Precio Platos” muestra una lista de los “Platos Disponibles” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nuevo” para registrar los datos de una nuevo precio plato. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Plato” con el manejador de la pantalla “Nuevo Plato”.</p> |

| |
|--|
| <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos del plato que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Plato” con el manejador de la pantalla “Modificar Plato”.</p> <p>P3. El usuario elegirá la opción “Eliminar” para eliminar los datos del plato que ha sido seleccionada de la lista. Al elegir esta opción se ejecutará el caso de uso “Eliminar Plato”.</p> <p>P4. El usuario elegirá la opción “Atrás” para volver al caso de Uso Gestionar Precios con su respectiva pantalla.</p> |
| <p>Excepciones:</p> <p>E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los platos registrados no se activarán las opciones Modificar y Eliminar.</p> |
| <p>Caso de Uso Gestionar Precio Bebidas</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Gestionar el registro de las diferentes Precios Bebidas que están registradas dentro del Sistema que son respectivas del Restaurante.</p> |
| <p>Resumen: El Usuario elegirá la opción Bebidas ejecutándose la opción “Gestionar Precio Platos” para poder acceder a una lista con los detalles de los precio de los platos registrados.</p> |
| <p>Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Precios Bebidas”.</p> |

Flujo principal:

El manejador “Gestionar Precio Bebidas” muestra una lista de los “Precio de las Bebidas Registrados” en la base de datos del Sistema.

- P1.** El usuario elegirá la opción “Nueva” para registrar los datos de una nuevo precio bebida. Al elegir esta opción se ejecutará el caso de uso “Registrar Nueva Bebida” con el manejador de la pantalla “Nueva Bebida”.
- P2.** El usuario elegirá la opción “Modificar” para cambiar los datos de la bebida que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Bebida” con el manejador de la pantalla “Modificar Bebida”.
- P3.** El usuario elegirá la opción “Eliminar” para eliminar los datos de la bebida que ha sido seleccionada de la lista. Al elegir esta opción se ejecutará el caso de uso “Eliminar Bebida”.
- P4.** El usuario elegirá la opción “Atrás” para volver al caso de Uso Gestionar Precios con su respectiva pantalla.

Excepciones:

- E1.** Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de las bebidas registradas no se activarán las opciones Modificar y Eliminar.

Caso de Uso Registrar Nuevo Plato

Actores: Usuario.

| |
|---|
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Plato en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Plato” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precio Platos” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nueva Plato” con su respectivo formulario de registro. |
| Flujo principal: La ventana “Nuevo Plato”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo plato. P1. En la ventana “Nuevo Plato” el usuario ingresará los datos del Plato a registrarse. P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”. |
| Flujos secundarios: S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Precios Platos”. |
| Excepciones: E1. Si el usuario no introduce todos los datos requeridos en el |

formulario de registro no se activará la opción “Guardar”.

| Caso de Uso Modificar Plato |
|---|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Plato Registrado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del plato que se ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precios Platos” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Plato” con un formulario que contendrá los datos actuales del plato ejecutándose el manejador del caso de uso “Modificar Plato”. |
| <p>Flujo principal:</p> <p>La ventana “Modificar Plato” muestra los datos del plato que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Plato” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del plato, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla</p> |

anterior “Gestionar Precios Platos”.

Excepciones:

E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.

Caso de Uso Eliminar Plato

Actores: Usuario.

Tipo: Extensión.

Propósito: Eliminar datos de un Plato Registrado.

Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del plato que ha seleccionado.

Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precios Platos” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del Plato.

Flujo principal:

El manejador “Eliminar Plato” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del plato de la base de datos del Sistema.

Flujos secundarios:

S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si

está seguro para proceder a eliminar el registro del plato, ejecutado por el manejador “Eliminar Plato”.

S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del plato, ejecutado por el manejador “Eliminar Plato”.

| Caso de Uso Registrar Nueva Bebida |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nueva Bebida en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nueva Bebida” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precio Bebidas” y seleccionado la opción “Nueva”. Se despliega la pantalla “Nueva Bebida” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nueva Bebida”, muestra al usuario un formulario con los datos a introducirse para el registro de un nueva bebida.</p> <p>P1. En la ventana “Nueva Bebida” el usuario ingresará los datos de la Bebida a registrarse.</p> |

| |
|---|
| <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Precios Bebidas”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.</p> |

| |
|---|
| <p>Caso de Uso Modificar Bebida</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Modificar los datos de una Bebida Registrado.</p> |
| <p>Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos de la bebida que se ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precios Bebidas” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Bebida” con un formulario que contendrá los datos actuales de la bebida ejecutándose el manejador del caso de uso “Modificar Bebida”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Bebida” muestra los datos de la bebida que se desea</p> |

| |
|---|
| <p>modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Bebida” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario de la bebida, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Precios Bebidas”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|---|
| Caso de Uso Eliminar Bebida |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Eliminar datos de una Bebida Registrado. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos de la bebida que ha seleccionado. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Precios |

Bebidas” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos de la Bebida.

Flujo principal:

El manejador “Eliminar Bebida” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro de la bebida de la base de datos del Sistema.

Flujos secundarios:

- S1.** El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro de la bebida, ejecutado por el manejador “Eliminar Bebida”.
- S2.** El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro de la bebida, ejecutado por el manejador “Eliminar Bebida”.

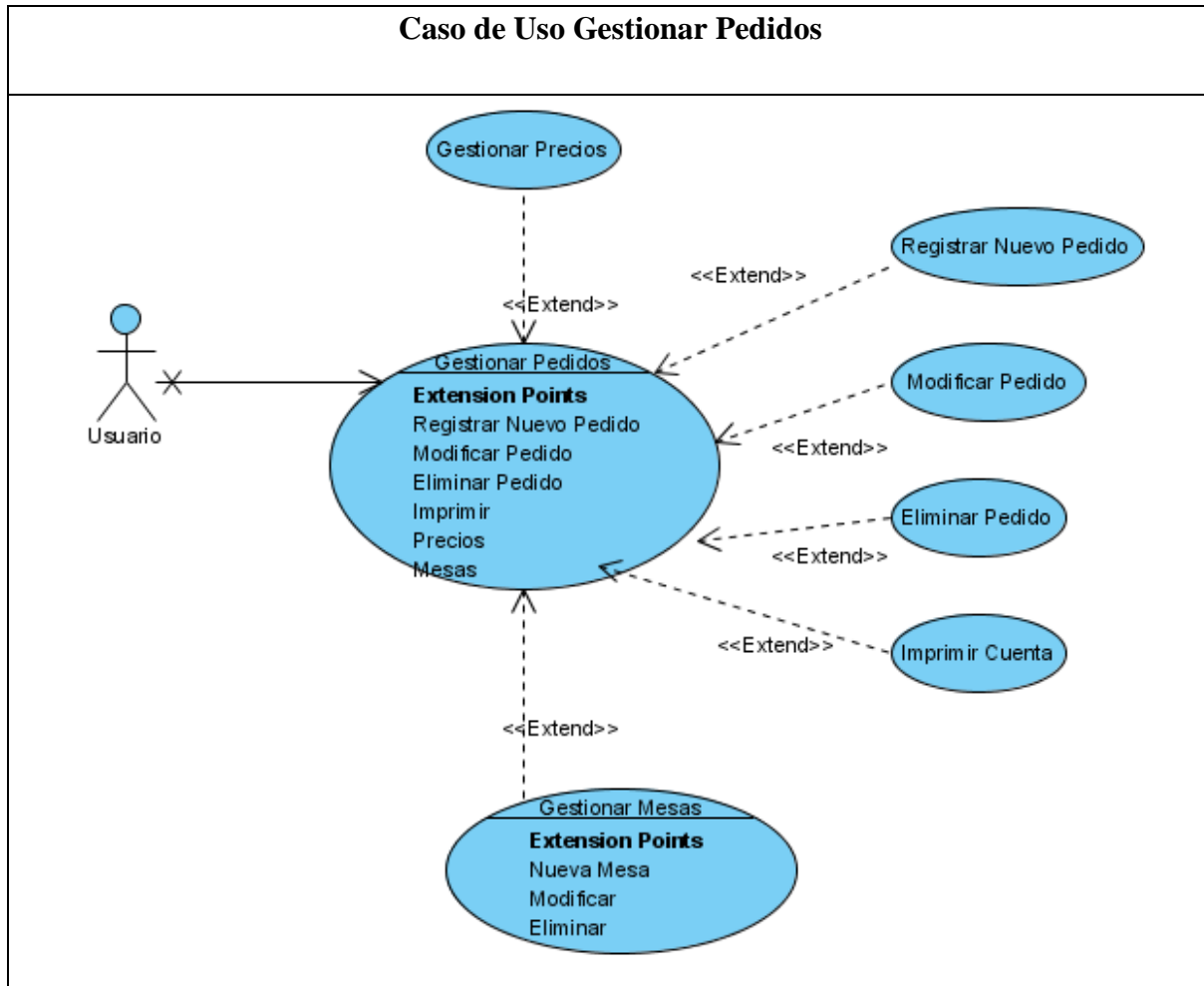


Figura 12Caso de Uso Gestionar Pedidos

| |
|--|
| Caso de Uso Gestionar Pedidos |
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Pedidos que se registran dentro del Sistema que pertenecen a los respectivos pedidos de los clientes del Restaurante. |

Resumen: El Usuario elegirá la Pestaña Pedidos ejecutándose la opción “Pedidos” para poder acceder a una lista de las mesas disponibles, ocupadas y reservadas con los detalles de los pedidos de los clientes posteriormente registrados o no del sistema.

Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Pedidos”.

Flujo principal:

El manejador “Gestionar Pedidos” muestra una lista de las “Mesas Registradas” en la base de datos del Sistema con sus respectivas características.

- P1.** El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Pedido, habiendo seleccionado previamente el numero de mesa de la lista desplegada. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Pedido” con el manejador de la pantalla “Nuevo Pedido”.
- P2.** El usuario elegirá la opción “Modificar” para cambiar los datos del Pedido que ha sido seleccionado de la mesa de la lista desplegada. Al elegir esta opción se ejecutará el caso de uso “Modificar Pedido” con el manejador de la pantalla “Modificar Pedido”.
- P3.** El usuario elegirá la opción “Eliminar” para eliminar los datos del Pedido que ha sido seleccionado de la mesa donde se encuentra registrado el pedido. Al elegir esta opción se ejecutará el caso de uso “Eliminar Pedido”.
- P4.** El usuario elegirá la opción “Gestionar Mesas” para ingresar a la pantalla “Gestionar Mesas”. Al elegir esta opción se ejecutara el

| |
|--|
| <p>caso de uso Gestionar Mesas con su respectivo manejador.</p> <p>P5. El usuario elegirá la opción “Gestionar Precios” para ingresar a la pantalla “Gestionar Precios”. Al elegir esta opción se ejecutara el caso de uso Gestionar Precios con su respectivo manejador.</p> |
| <p>Excepciones:</p> <p>E1. Si el Usuario no selecciona ningún dato de la lista de mesas que contiene los detalles de los pedidos registrados no se activarán las opciones Modificar y Eliminar.</p> |

| |
|---|
| <p>Caso de Uso Registrar Nuevo Pedido</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Registrar los datos de un nuevo Pedido en la base de datos del Sistema.</p> |
| <p>Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo” para guardar los datos seleccionados de las listas desplegadas de los tipos de platos y bebidas a consumirse.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Pedidos” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Pedido” con su respectivo listas de selección de platos y bebidas para generar el pedido.</p> |
| <p>Flujo principal:</p> |

| |
|---|
| <p>La ventana “Nuevo Pedido”, muestra al usuario una lista con los datos a seleccionarse e introducirse para el registro de un nuevo pedido.</p> <p>P1. En la ventana “Nuevo Pedido” el usuario seleccionara los datos de las listas desplegadas que se ingresarán del Pedido a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos seleccionados, se muestra el mensaje, “Datos guardados correctamente” y el estado de la mesa pasara a "OCUPADA".</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Pedidos”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no selecciona todos los datos requeridos no se activará la opción “Guardar”.</p> |

| |
|--|
| Caso de Uso Modificar Pedido |
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Modificar los datos de un Pedido Registrado. |
| Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del pedido que se ha seleccionado del numero de mesa. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Pedidos” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar |

| |
|---|
| <p>Pedido” con un formulario que contendrá los datos actuales del pedido ejecutándose el manejador del caso de uso “Modificar Pedido”.</p> |
| <p>Flujo principal:</p> <p>La ventana “Modificar Pedido” muestra los datos del pedido que se desea modificar en sus campos correspondientes.</p> <p>P1. En la ventana “Modificar Pedido” el usuario podrá cambiar todos los datos que se visualizan.</p> <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del pedido, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Pedidos”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|------------------------------------|
| Caso de Uso Eliminar Pedido |
| Actores: Usuario. |
| Tipo: Extensión. |

| |
|---|
| Propósito: Eliminar datos de un Pedido Registrado. |
| Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del pedido que se ha seleccionado del numero de mesa de la lista desplegada. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Pedidos” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del pedido del numero de mesa seleccionada. |
| Flujo principal: El manejador “Eliminar Pedido” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del pedido de la base de datos del Sistema. |
| Flujos secundarios: S1. El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro del pedido, ejecutado por el manejador “Eliminar Pedido”. S2. El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del pedido, ejecutado por el manejador “Eliminar Pedido”. |

| Caso de Uso Imprimir Cuenta |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Imprimir un reporte del pedido registrado en el sistema. |
| Resumen: El Usuario seleccionará el botón “Imprimir Cuenta” de la pantalla “Gestionar Pedidos” para generar e imprimir un reporte con el respectivo monto de la cuenta registrada en el sistema. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Pedidos” y seleccionado la opción “Imprimir Cuenta”. Si el pedido tiene un cliente asignado se imprime la cuenta, caso contrario se despliega la pantalla “Verificar Cliente” con una lista de los clientes a seleccionar para posteriormente imprimir la cuenta. |
| Flujo principal: La ventana “Verificar Cliente”, muestra al usuario una lista con los datos de los clientes que seleccionaran para el registro del cliente con el pedido y posterior impresión de la cuenta. P1. En la ventana “Verificar Cliente” el usuario seleccionara de una lista desplegada al cliente que será registrado con la respectiva cuenta. P2. El usuario pulsará el botón “Siguiente” para imprimir la cuenta del cliente seleccionado. |

| |
|---|
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Pedidos”.</p> |
| <p>Excepciones:</p> <p>E1. Si el usuario no selecciona al cliente de la lista desplegada no se activará la opción “Siguiente”.</p> |

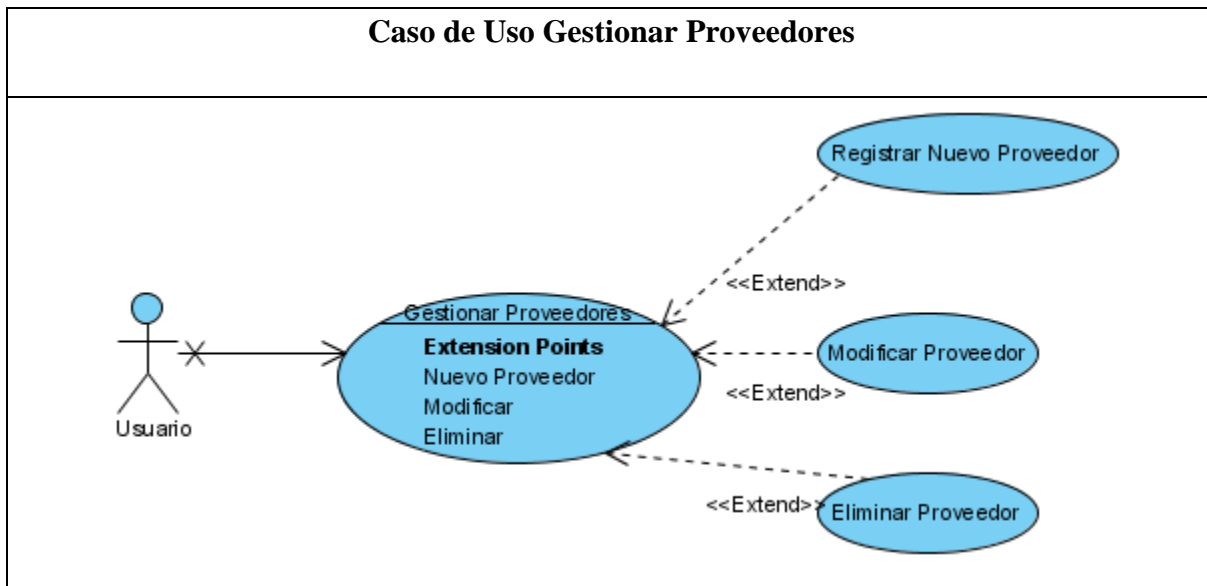


Figura 13 Caso de Uso Gestionar Proveedores

| |
|--|
| Caso de Uso Gestionar Proveedores |
| Actores: Usuario. |
| Tipo: Asociación. |
| Propósito: Gestionar el registro de los diferentes Proveedores que están |

| |
|--|
| registrados dentro del Sistema que trabajan con el Restaurante. |
| Resumen: El Usuario elegirá la Pestaña Proveedores ejecutándose la opción “Gestionar Proveedores” para poder acceder a una lista con los detalles de los proveedores registrados. |
| Precondiciones: El Usuario ejecutará el caso de uso “Gestionar Proveedores”. |
| <p>Flujo principal:</p> <p>El manejador “Gestionar Proveedores” muestra una lista de los “Proveedores Registrados” en la base de datos del Sistema.</p> <p>P1. El usuario elegirá la opción “Nuevo” para registrar los datos de un nuevo Proveedor. Al elegir esta opción se ejecutará el caso de uso “Registrar Nuevo Proveedor” con el manejador de la pantalla “Nuevo Proveedor”.</p> <p>P2. El usuario elegirá la opción “Modificar” para cambiar los datos del Proveedor que ha sido seleccionado de la lista. Al elegir esta opción se ejecutará el caso de uso “Modificar Proveedor” con el manejador de la pantalla “Modificar Proveedor”.</p> <p>P3. El usuario elegirá la opción “Eliminar” para eliminar los datos del Proveedor que ha sido seleccionado. Al elegir esta opción se ejecutará el caso de uso “Eliminar Proveedor”.</p> |
| <p>Excepciones:</p> <p>E1. Si el Usuario no selecciona ningún dato de la lista que contiene los detalles de los proveedores registrados no se activarán las opciones</p> |

Modificar y Eliminar.

| Caso de Uso Registrar Nuevo Proveedor |
|--|
| Actores: Usuario. |
| Tipo: Extensión. |
| Propósito: Registrar los datos de un nuevo Proveedor en la base de datos del Sistema. |
| Resumen: El Usuario seleccionará el botón “Guardar” de la pantalla “Nuevo Proveedor” para guardar los datos ingresados en el formulario de registro. |
| Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Proveedores” y seleccionado la opción “Nuevo”. Se despliega la pantalla “Nuevo Proveedor” con su respectivo formulario de registro. |
| <p>Flujo principal:</p> <p>La ventana “Nuevo Proveedor”, muestra al usuario un formulario con los datos a introducirse para el registro de un nuevo proveedor.</p> <p>P1. En la ventana “Nuevo Proveedor” el usuario ingresará los datos del Proveedor a registrarse.</p> <p>P2. El usuario pulsará el botón “Guardar” para almacenar los datos ingresados en el formulario, se muestra el mensaje, “Datos guardados correctamente”.</p> |
| Flujos secundarios: |

S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Proveedores”.

Excepciones:

E1. Si el usuario no introduce todos los datos requeridos en el formulario de registro no se activará la opción “Guardar”.

Caso de Uso Modificar Proveedores

Actores: Usuario.

Tipo: Extensión.

Propósito: Modificar los datos de un Proveedor Registrado.

Resumen: El usuario seleccionará la opción “Modificar”, para modificar los datos del proveedor que se ha seleccionado.

Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Proveedores” y seleccionado la opción “Modificar”. Se despliega la pantalla “Modificar Proveedor” con un formulario que contendrá los datos actuales del empleado ejecutándose el manejador del caso de uso “Modificar Proveedor”.

Flujo principal:

La ventana “Modificar Proveedor” muestra los datos del proveedor que se desea modificar en sus campos correspondientes.

P1. En la ventana “Modificar Proveedor” el usuario podrá cambiar todos los datos excepto la clave primaria que no se visualizara.

| |
|---|
| <p>P2. El usuario pulsará el botón “Modificar” para guardar los datos que se cambiaron en el formulario del proveedor, se muestra el mensaje “Datos modificados correctamente”.</p> |
| <p>Flujos secundarios:</p> <p>S1. El usuario pulsará el botón “Cancelar” para regresar a la pantalla anterior “Gestionar Proveedores”.</p> |
| <p>Excepciones:</p> <p>E1. El botón “Modificar” permanecerá activo mientras se estén ingresando todos los datos requeridos, si se detecta la falta de datos el botón “Modificar” pasará a estar deshabilitado hasta que el usuario complete el formulario.</p> |

| |
|--|
| <p>Caso de Uso Eliminar Proveedor</p> |
| <p>Actores: Usuario.</p> |
| <p>Tipo: Extensión.</p> |
| <p>Propósito: Eliminar datos de un Proveedor Registrado.</p> |
| <p>Resumen: El usuario seleccionará la opción “Eliminar” para eliminar los datos del proveedor que ha seleccionado.</p> |
| <p>Precondiciones: Que se haya ejecutado el caso de uso “Gestionar Proveedores” y seleccionado la opción “Eliminar”. Se desplegará una ventana de confirmación para verificar que se desea eliminar los datos del proveedor.</p> |

Flujo principal:

El manejador “Eliminar Proveedor” muestra al usuario en su respectiva pantalla una ventana de confirmación para proceder con la eliminación del registro del proveedor de la base de datos del Sistema.

Flujos secundarios:

- S1.** El usuario pulsará el botón “Sí” de la ventana de confirmación si está seguro para proceder a eliminar el registro del proveedor, ejecutado por el manejador “Eliminar Proveedor”.
- S2.** El usuario pulsará el botón “No” de la ventana de confirmación para cancelar la eliminación del registro del proveedor, ejecutado por el manejador “Eliminar Proveedor”.

II.1.5.1.2. Diagramas de Secuencia

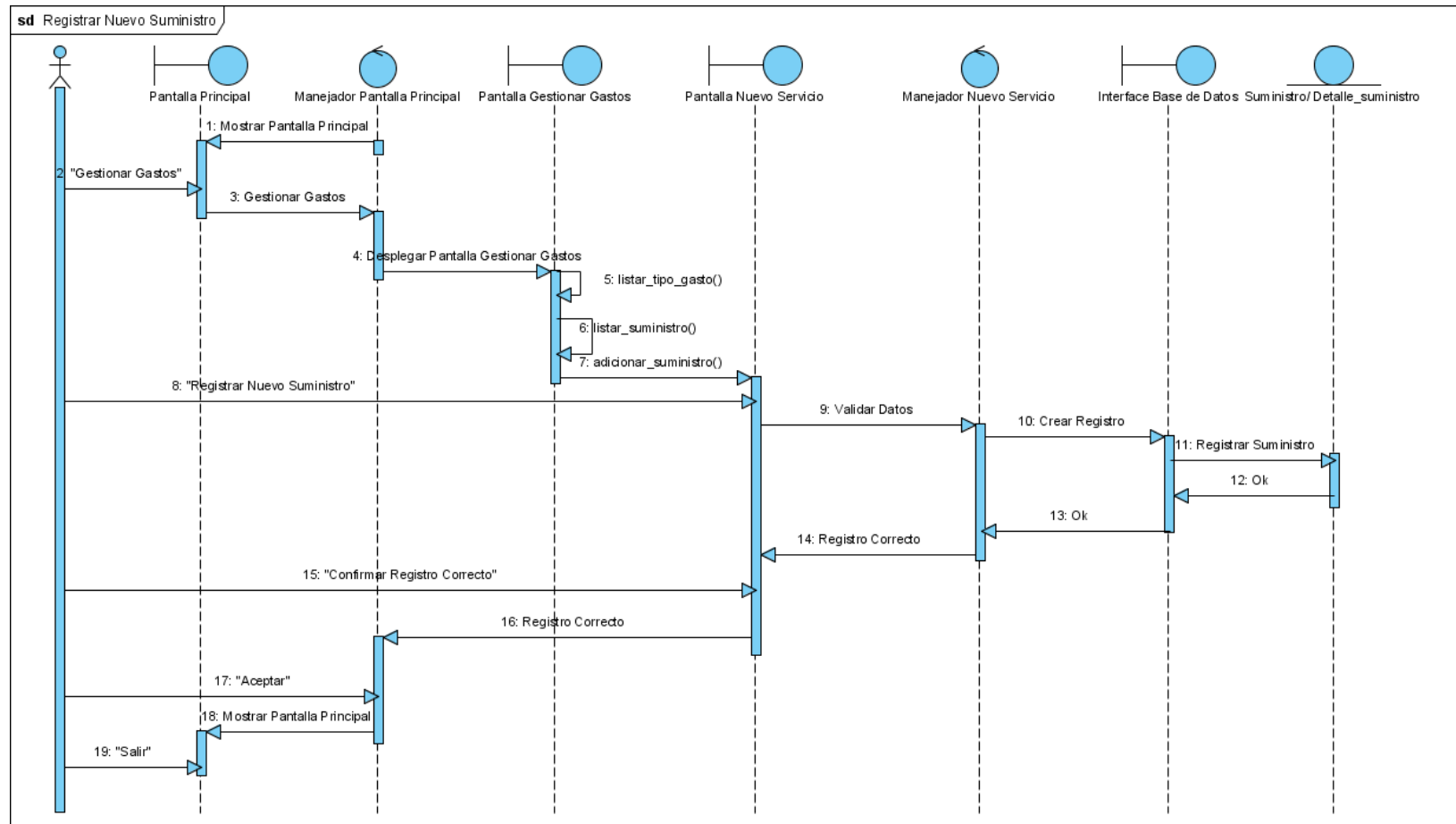


Figura 14 Diagrama Registrar Nuevo Suministro

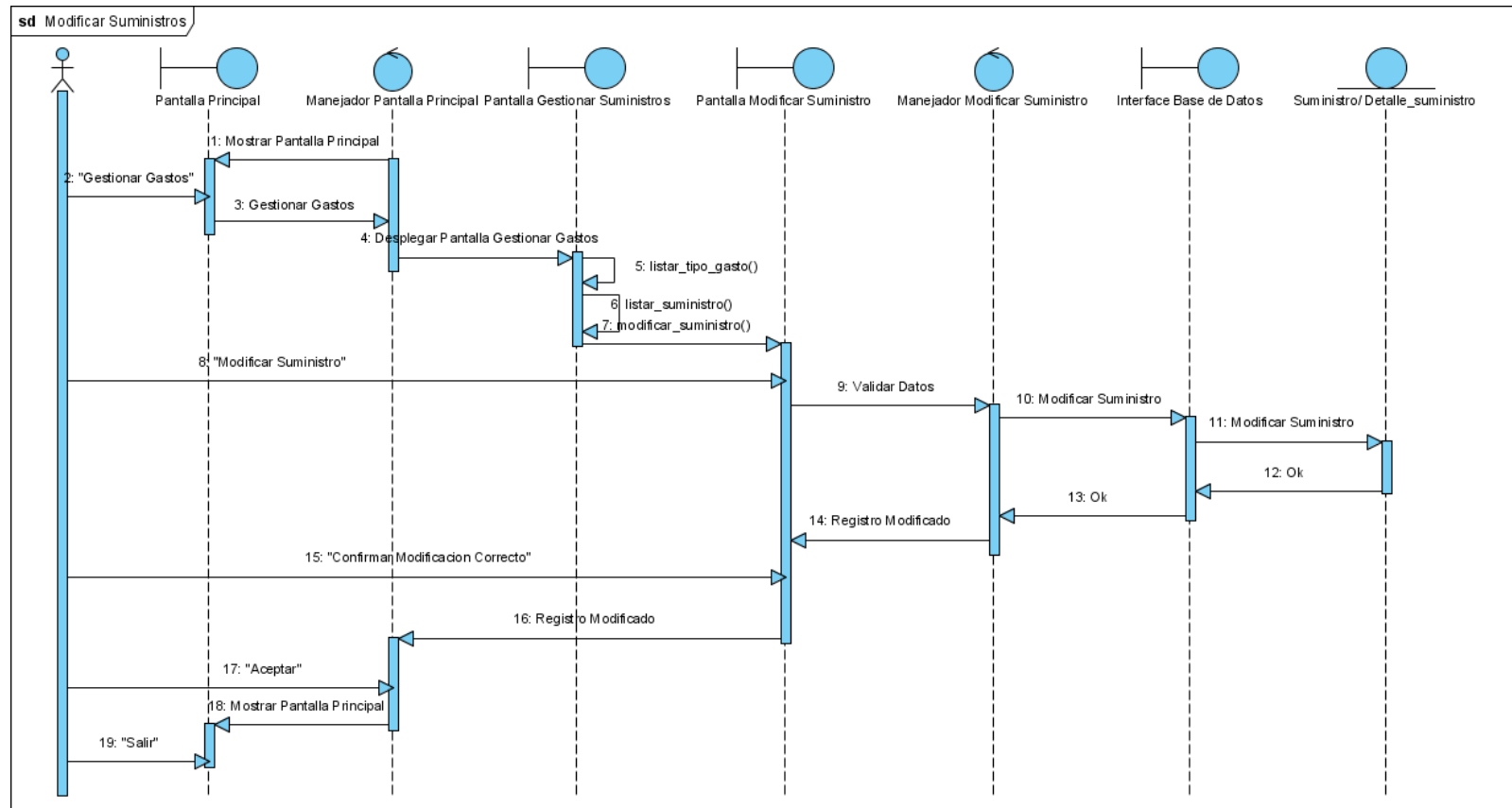


Figura 15 Diagrama Modificar Suministro

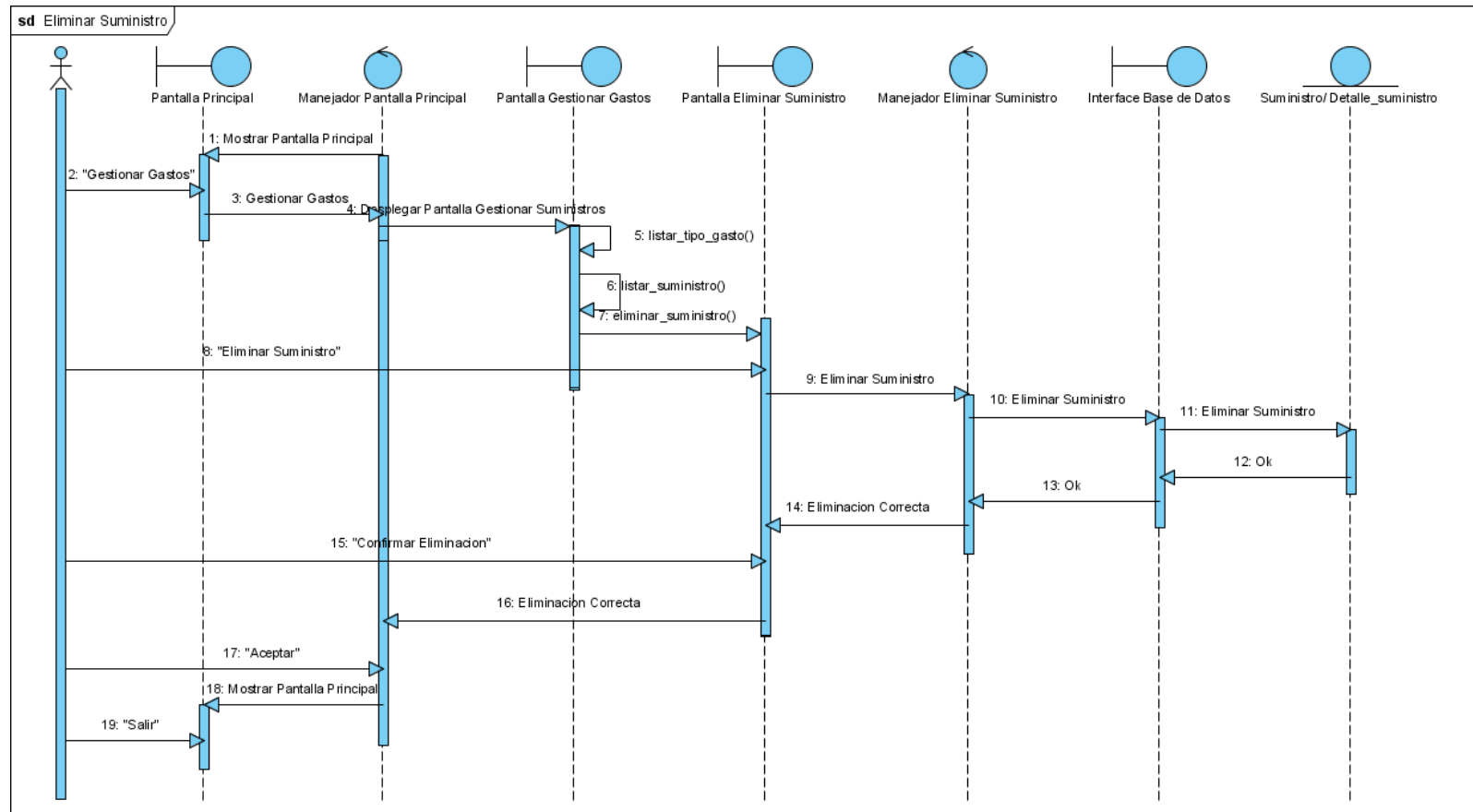


Figura 16 Diagrama Eliminar Suministro

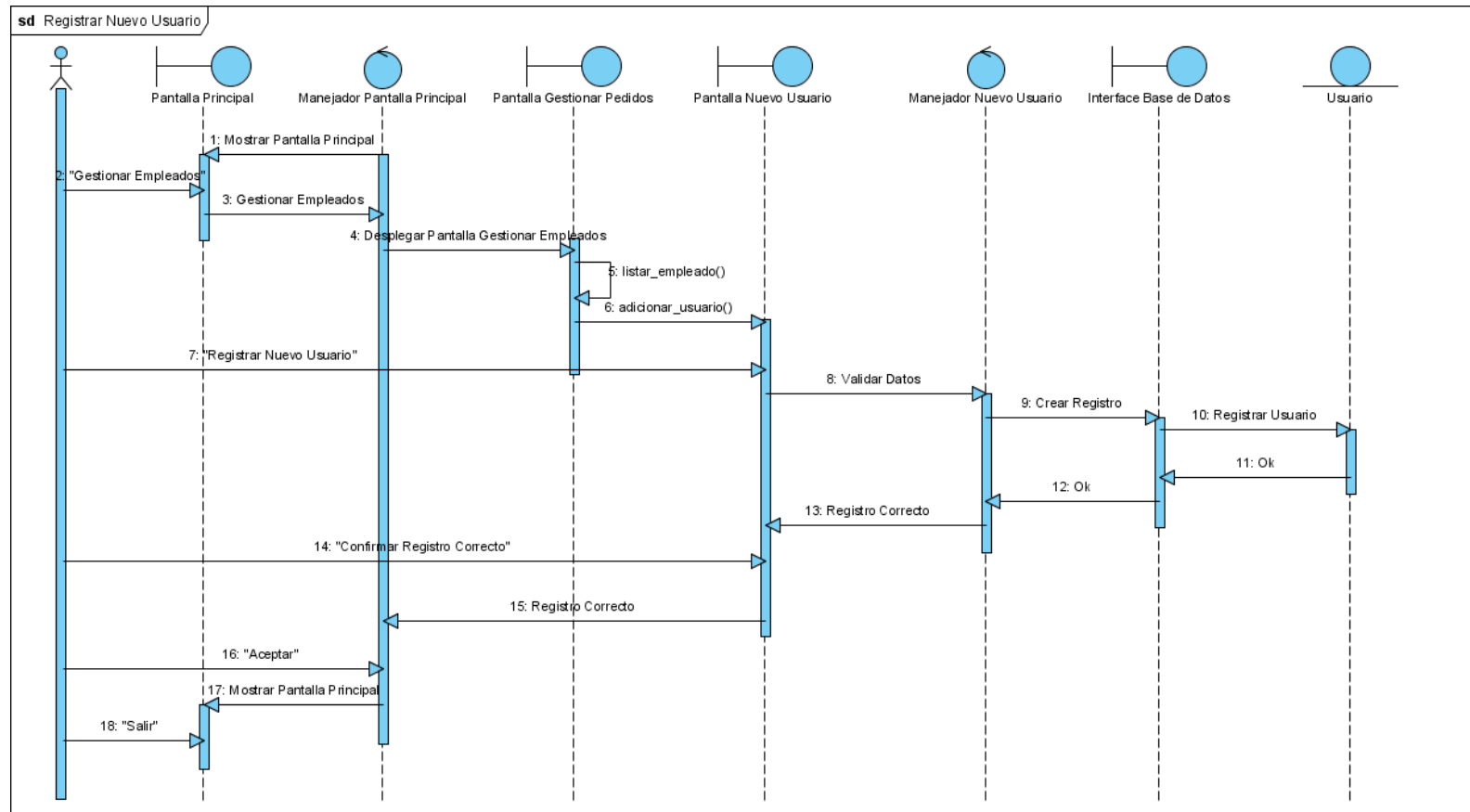


Figura 17Diagrama Registrar Nuevo Usuario

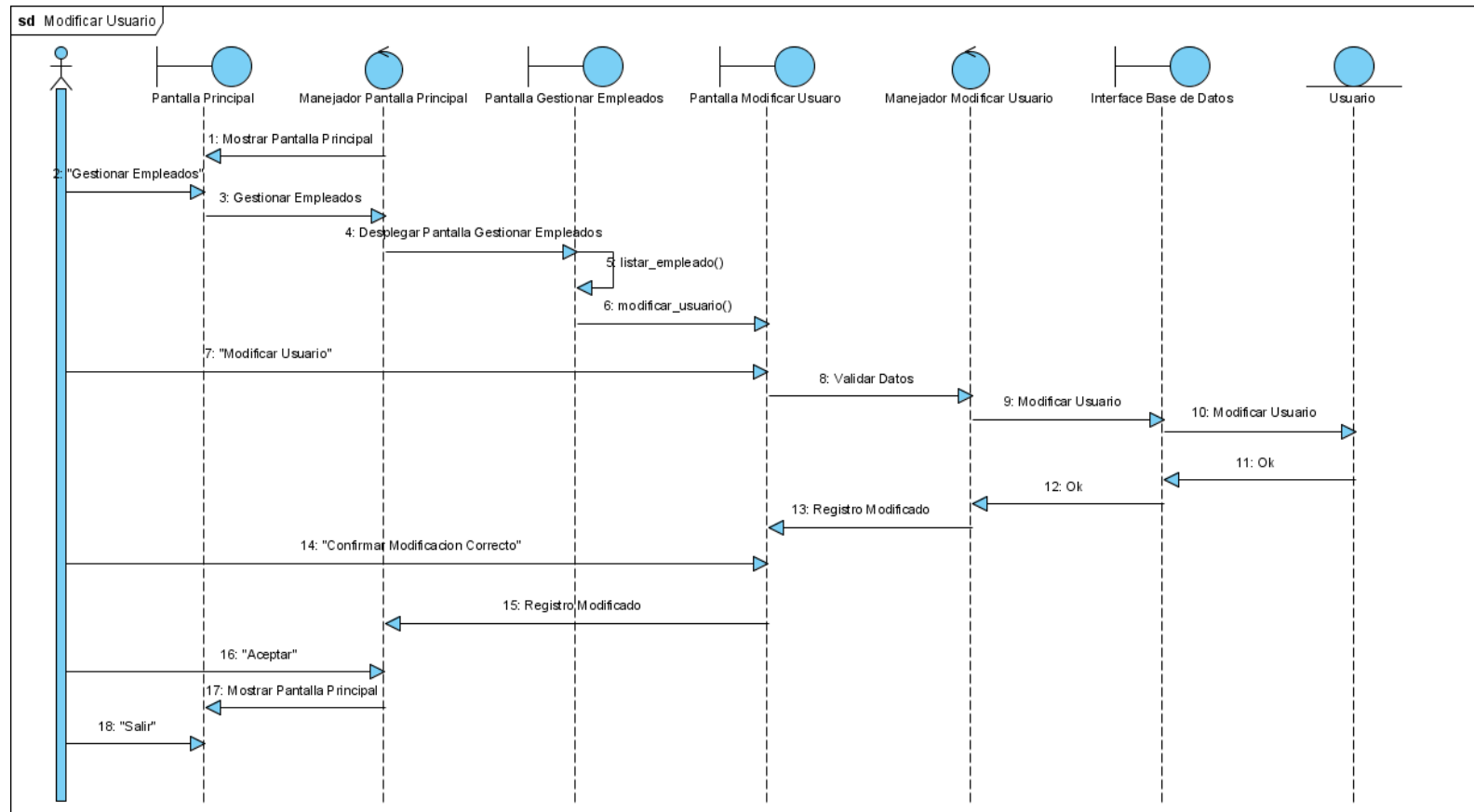


Figura 18 Diagrama Modificar Usuario

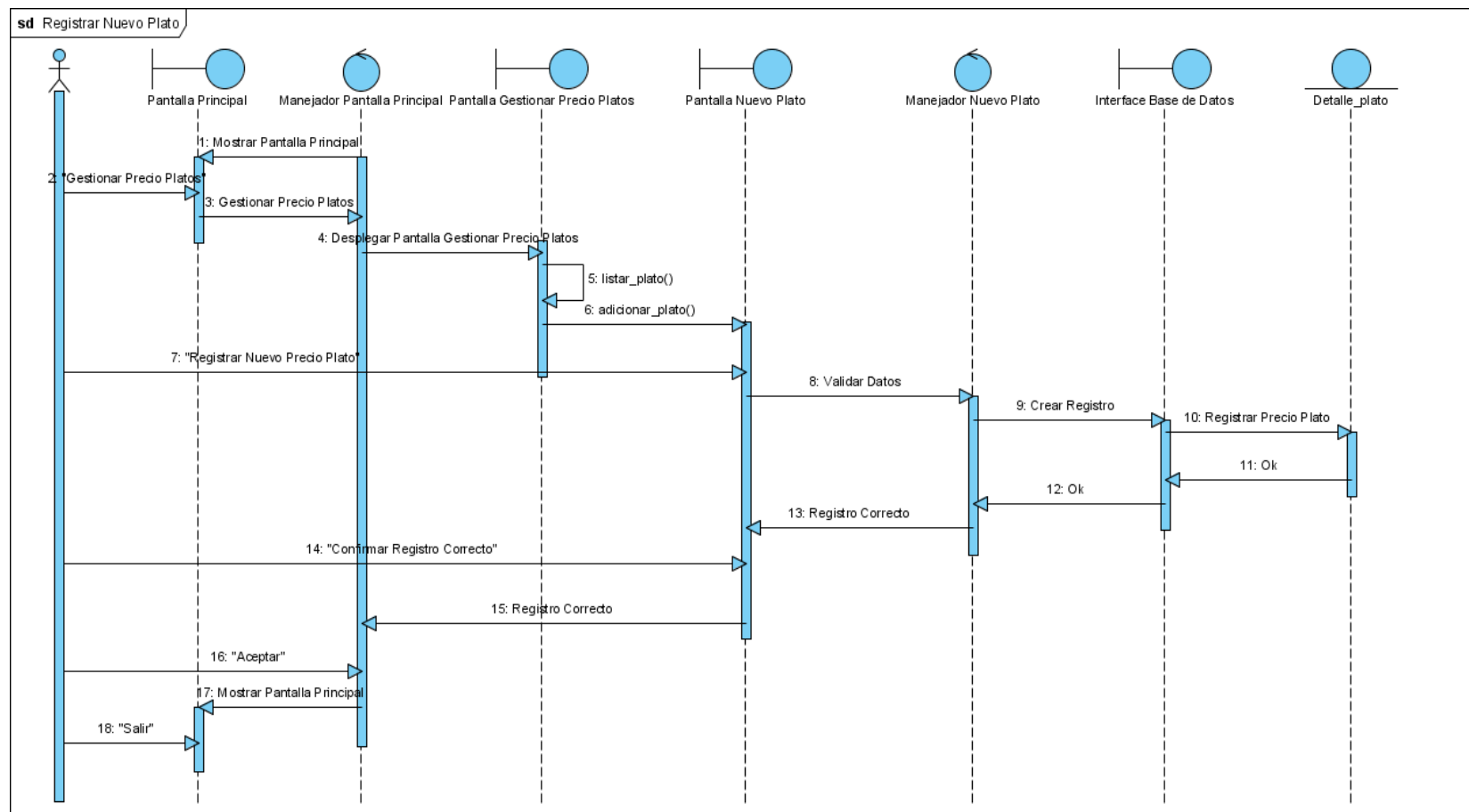


Figura 19Diagrama Registrar Nuevo Plato

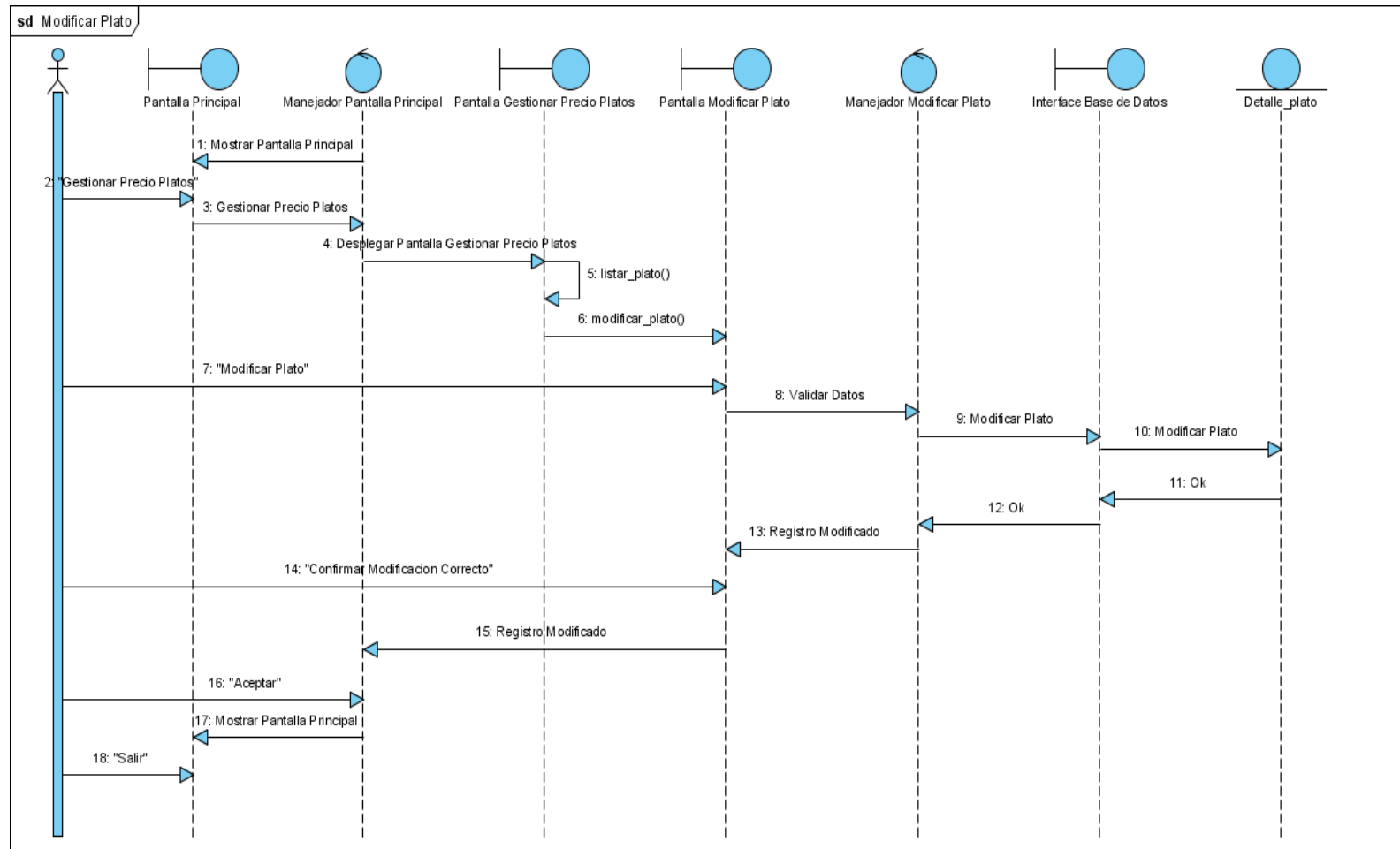


Figura 20 Diagrama Modificar Plato

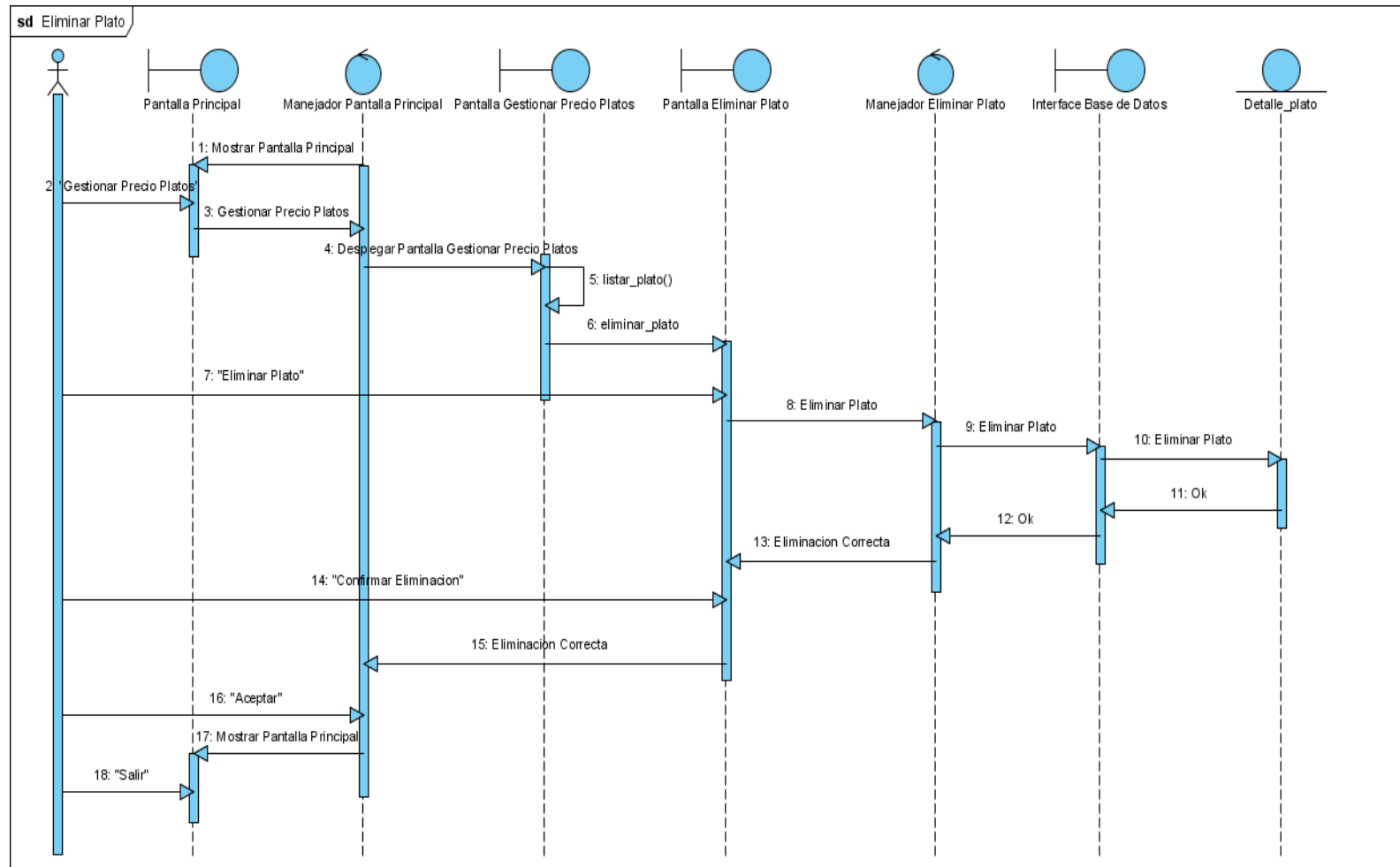


Figura 21 Diagrama Eliminar Plato

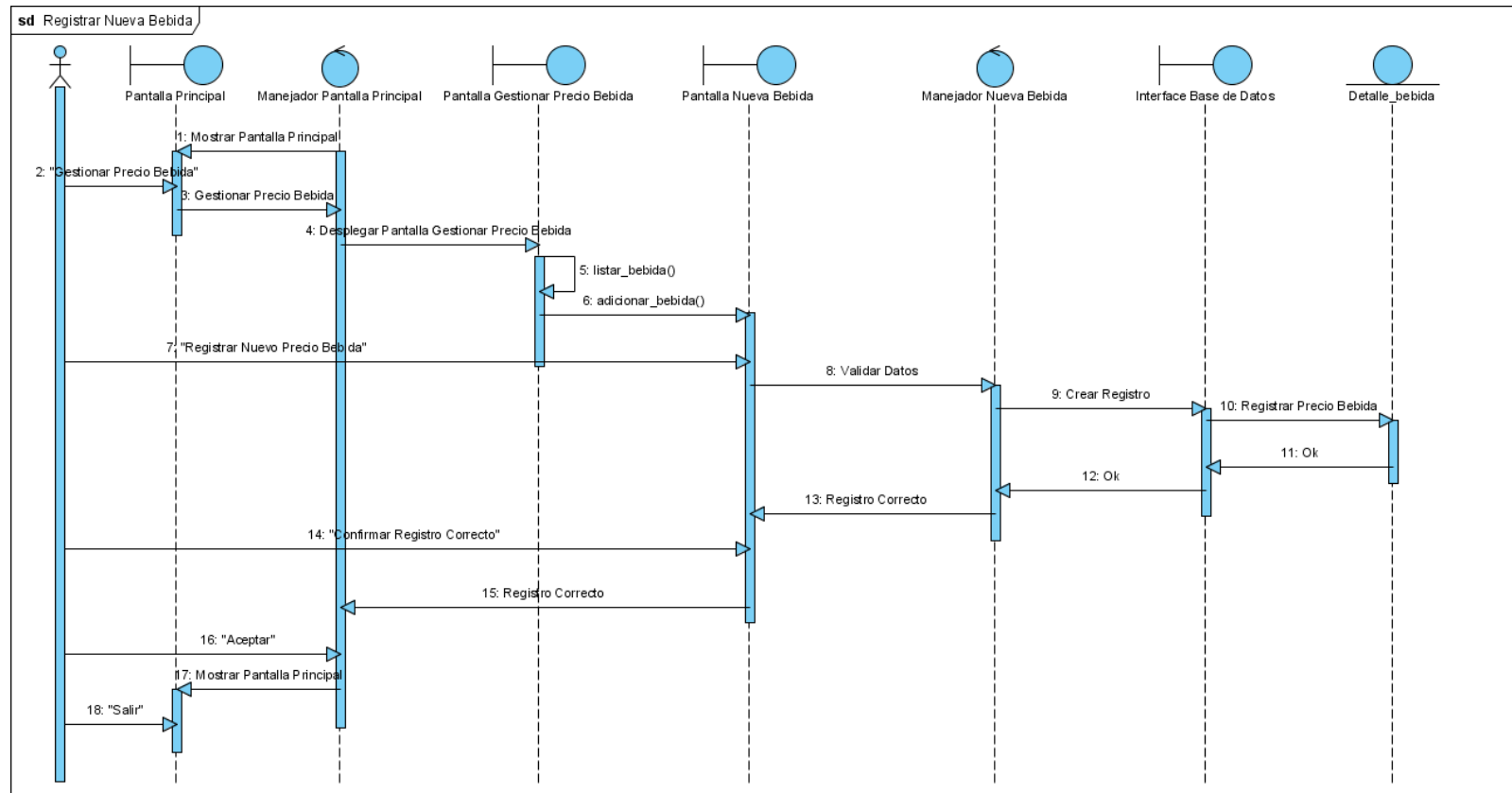


Figura 22 Diagrama Registrar Nueva Bebida

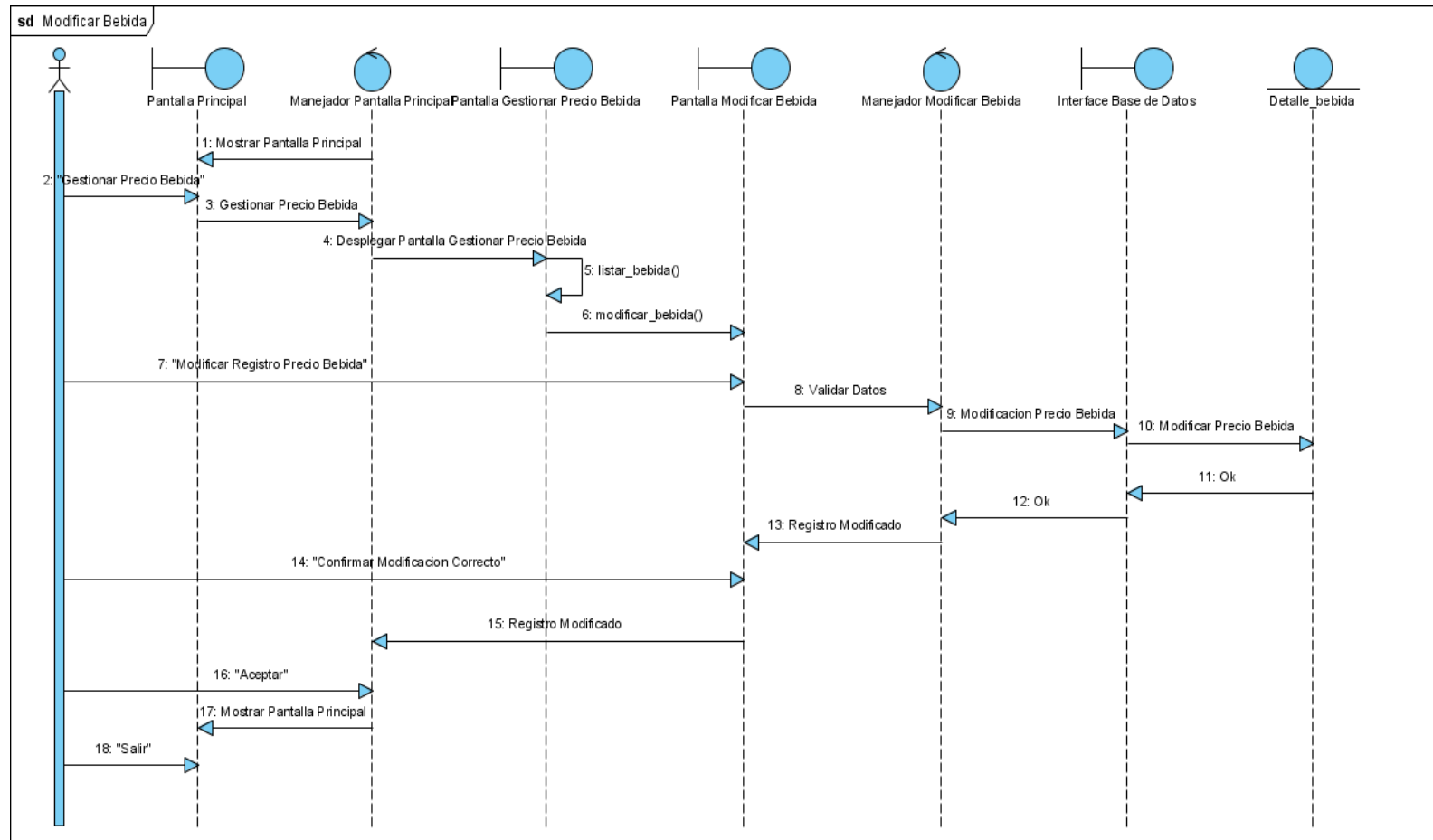


Figura 23 Diagrama Modificar Bebida

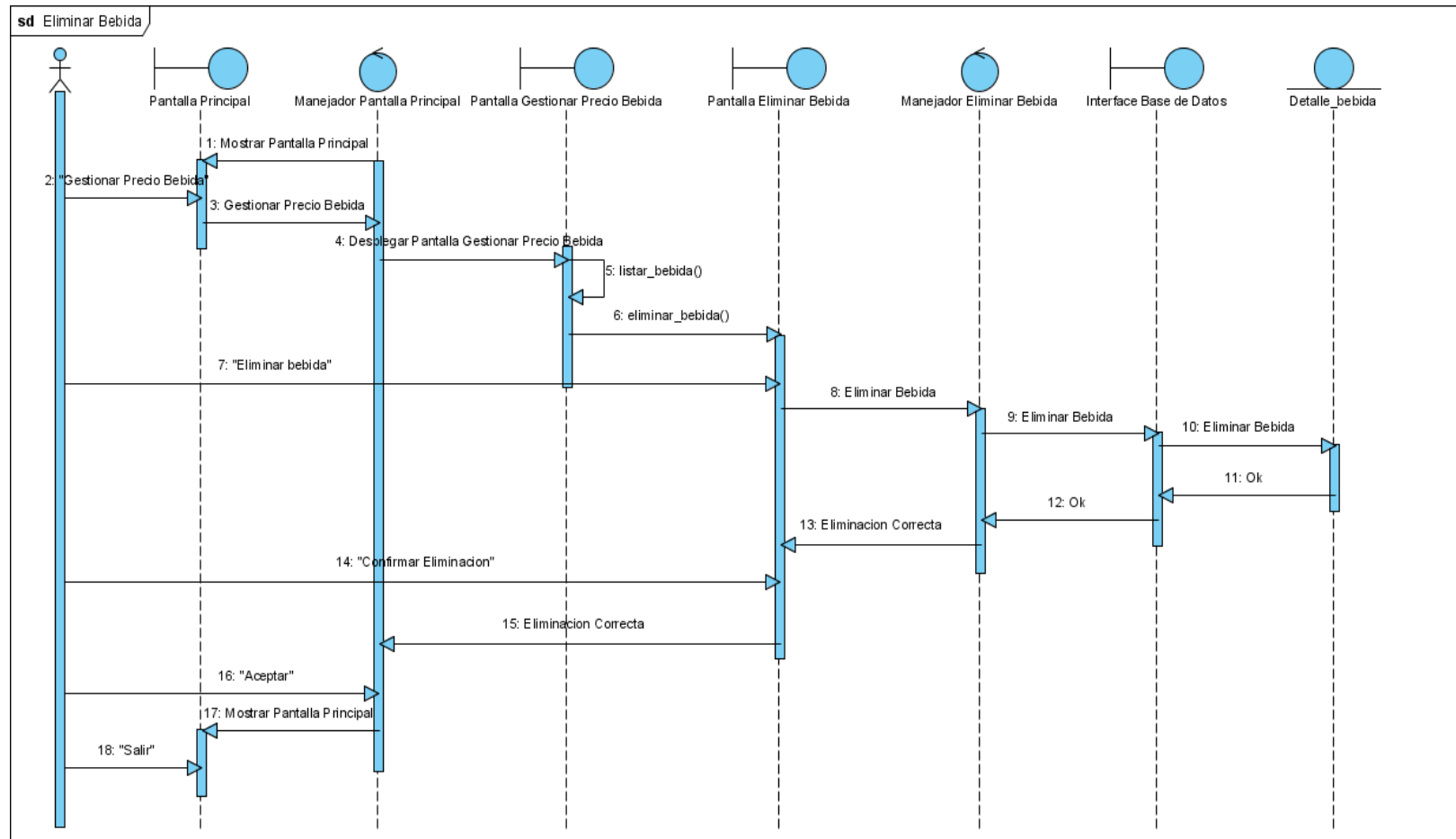


Figura 24 Diagrama Eliminar Bebida

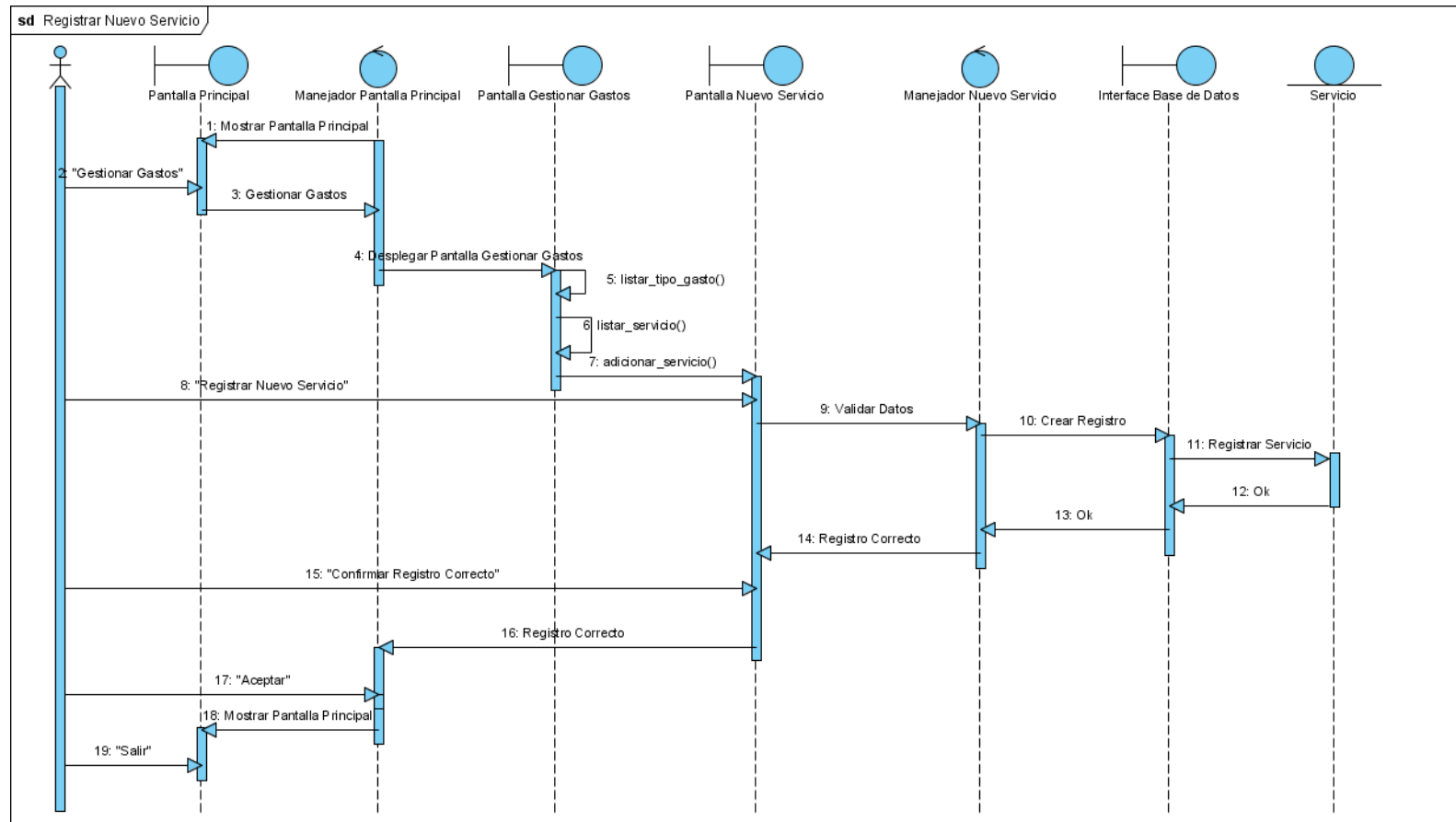


Figura 25 Diagrama Registrar Nuevo Servicio

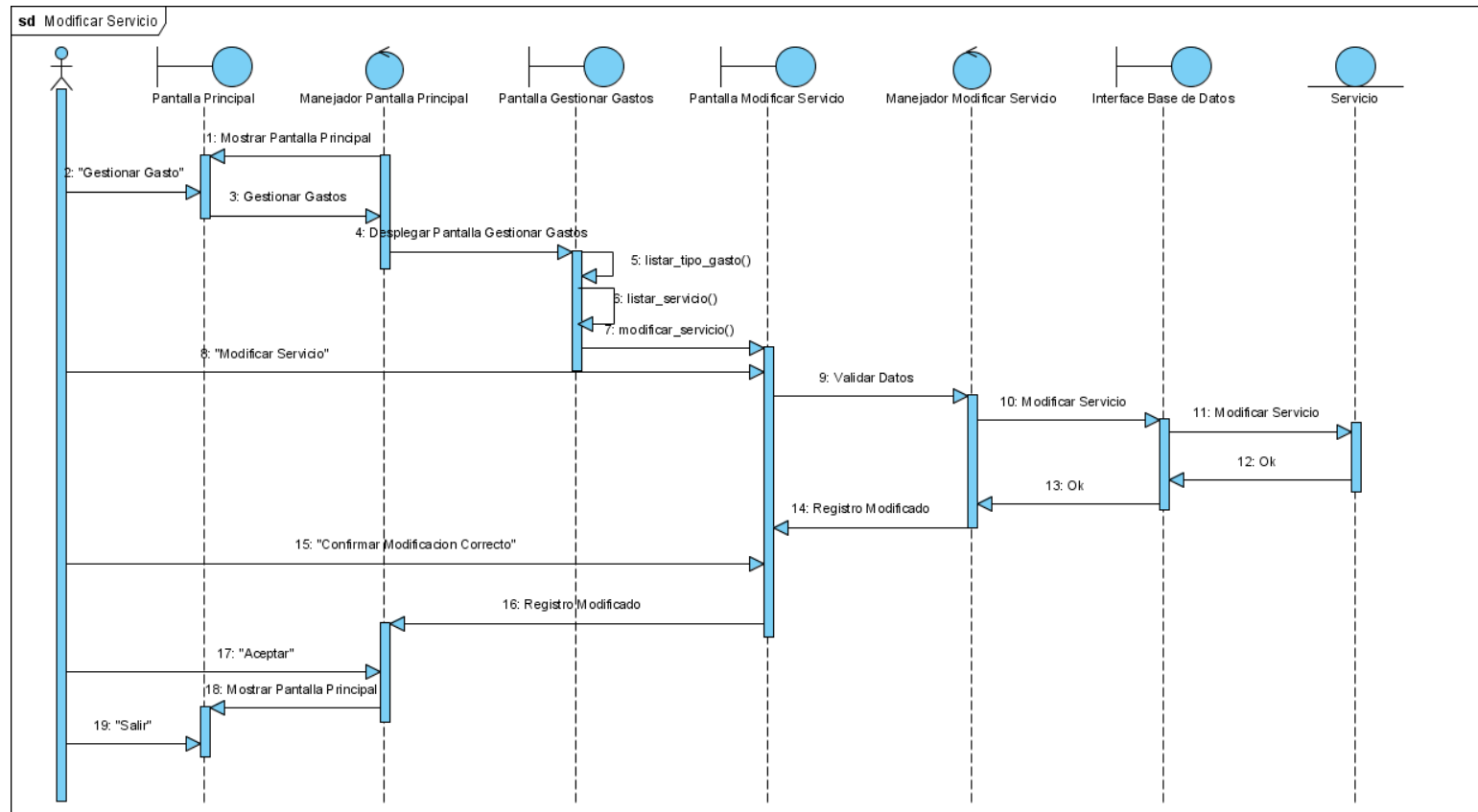


Figura 26 Diagrama Modificar Servicio

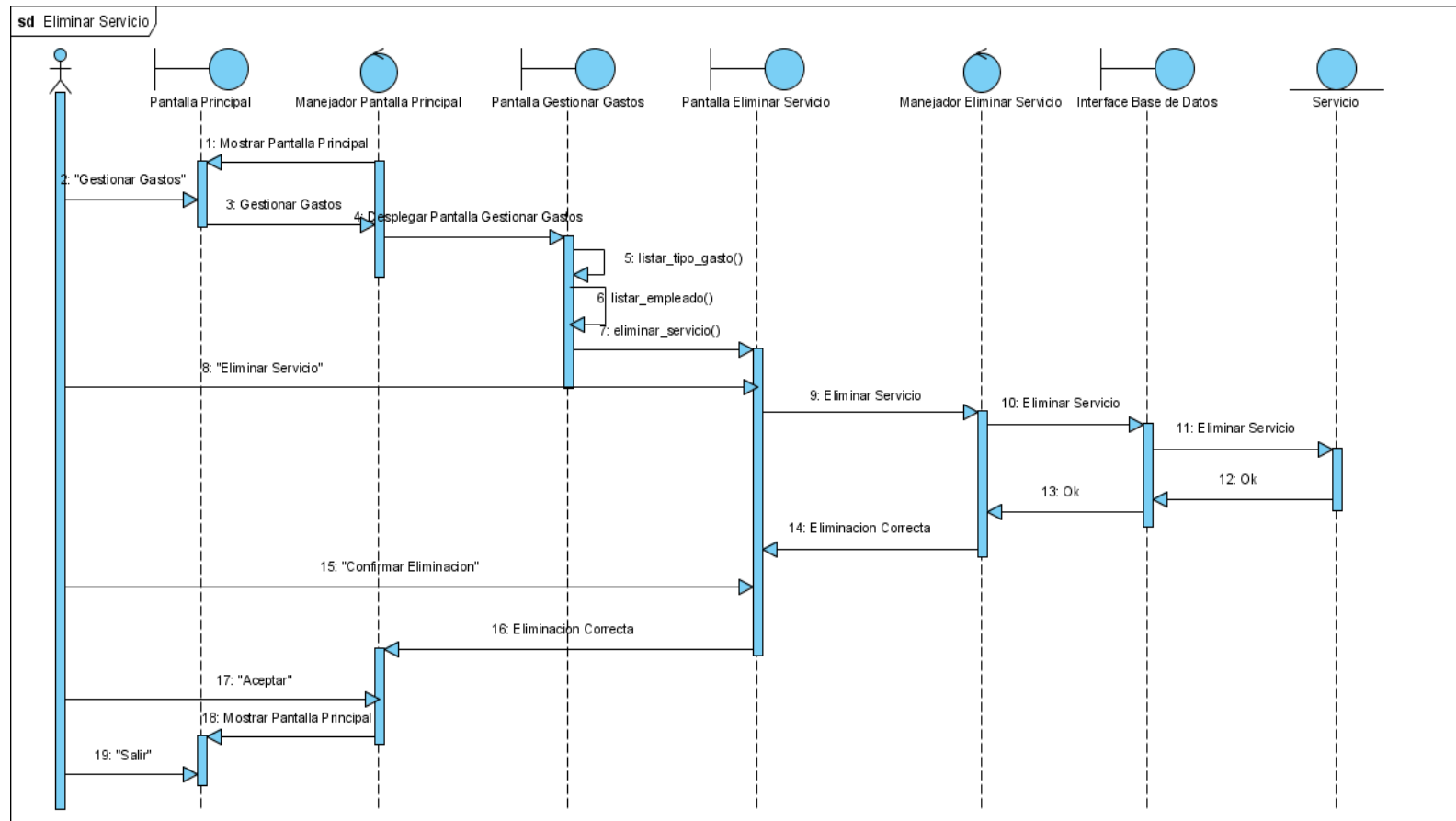


Figura 27 Diagrama Eliminar Servicio

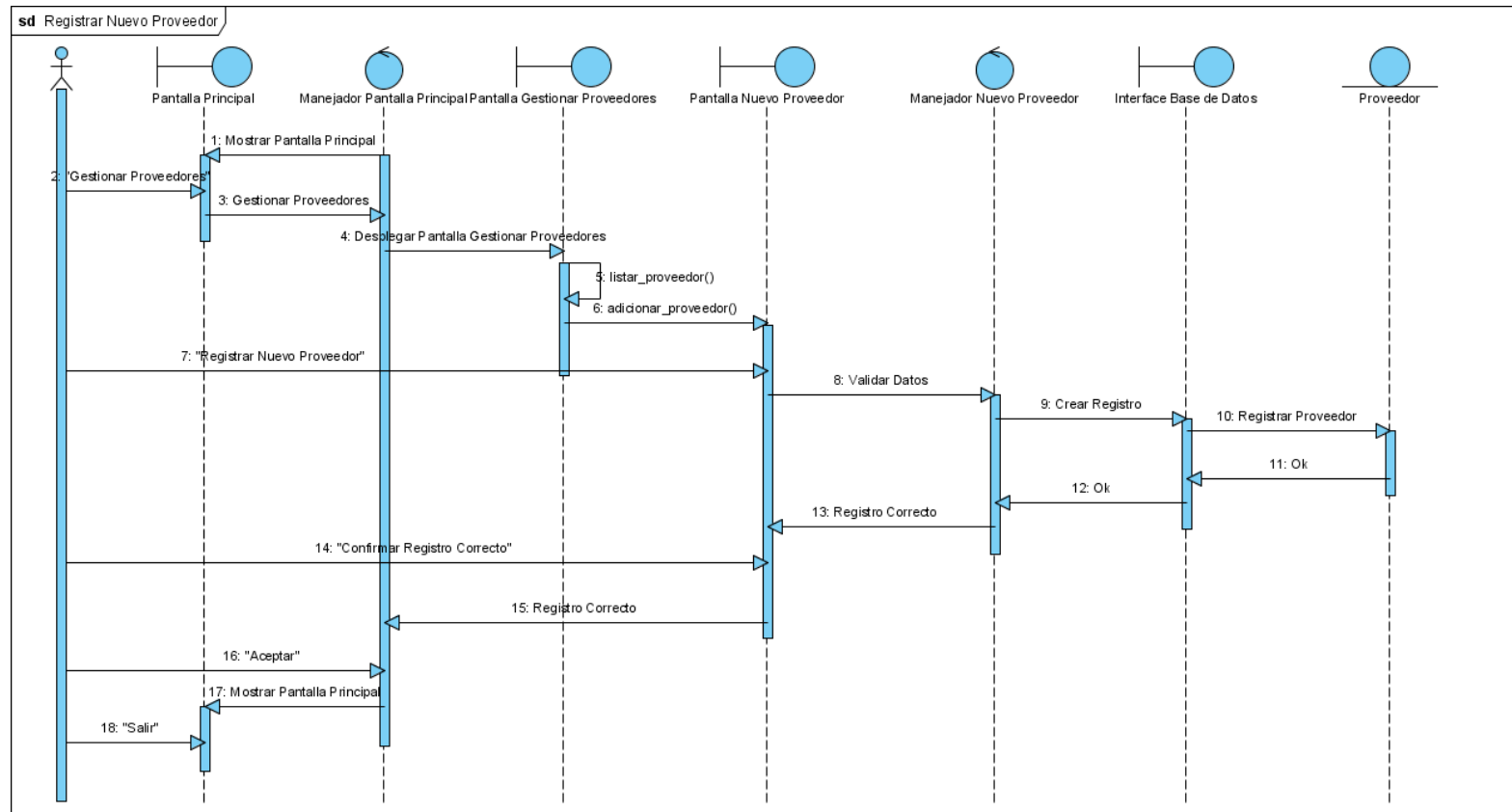


Figura 28Diagrama Registrar Nuevo Proveedor

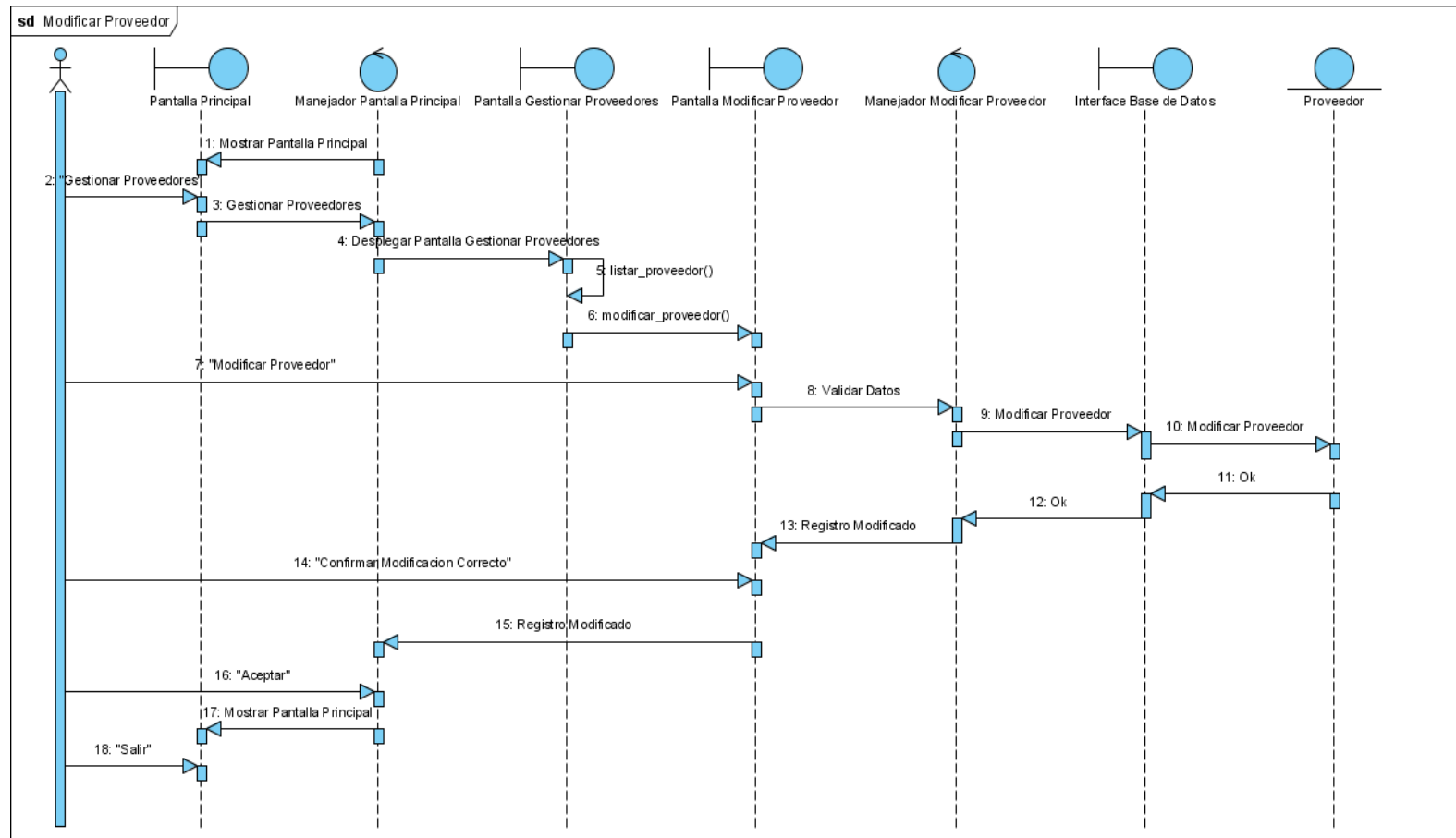


Figura 29 Diagrama Modificar Proveedor

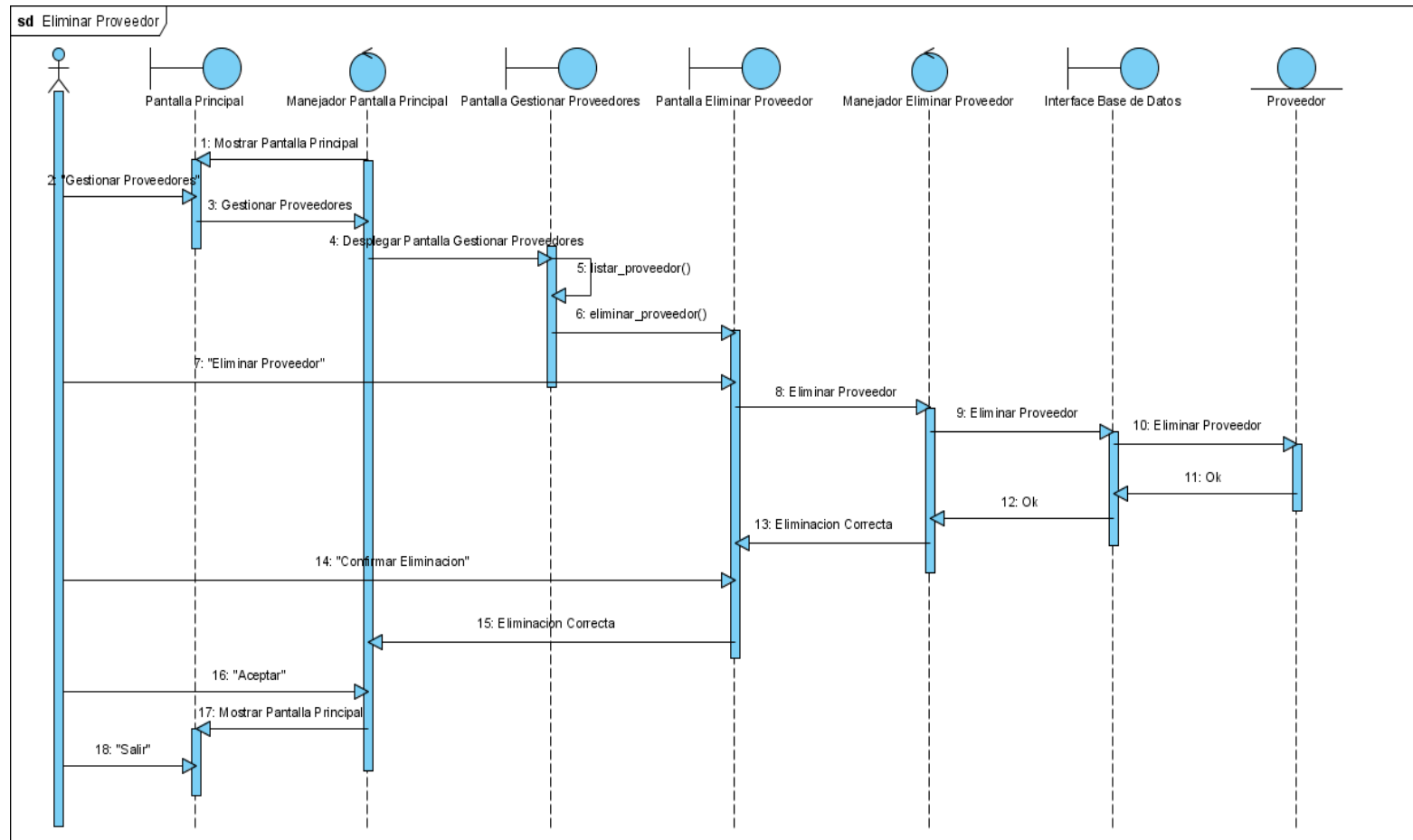


Figura 30 Diagrama Eliminar Proveedor

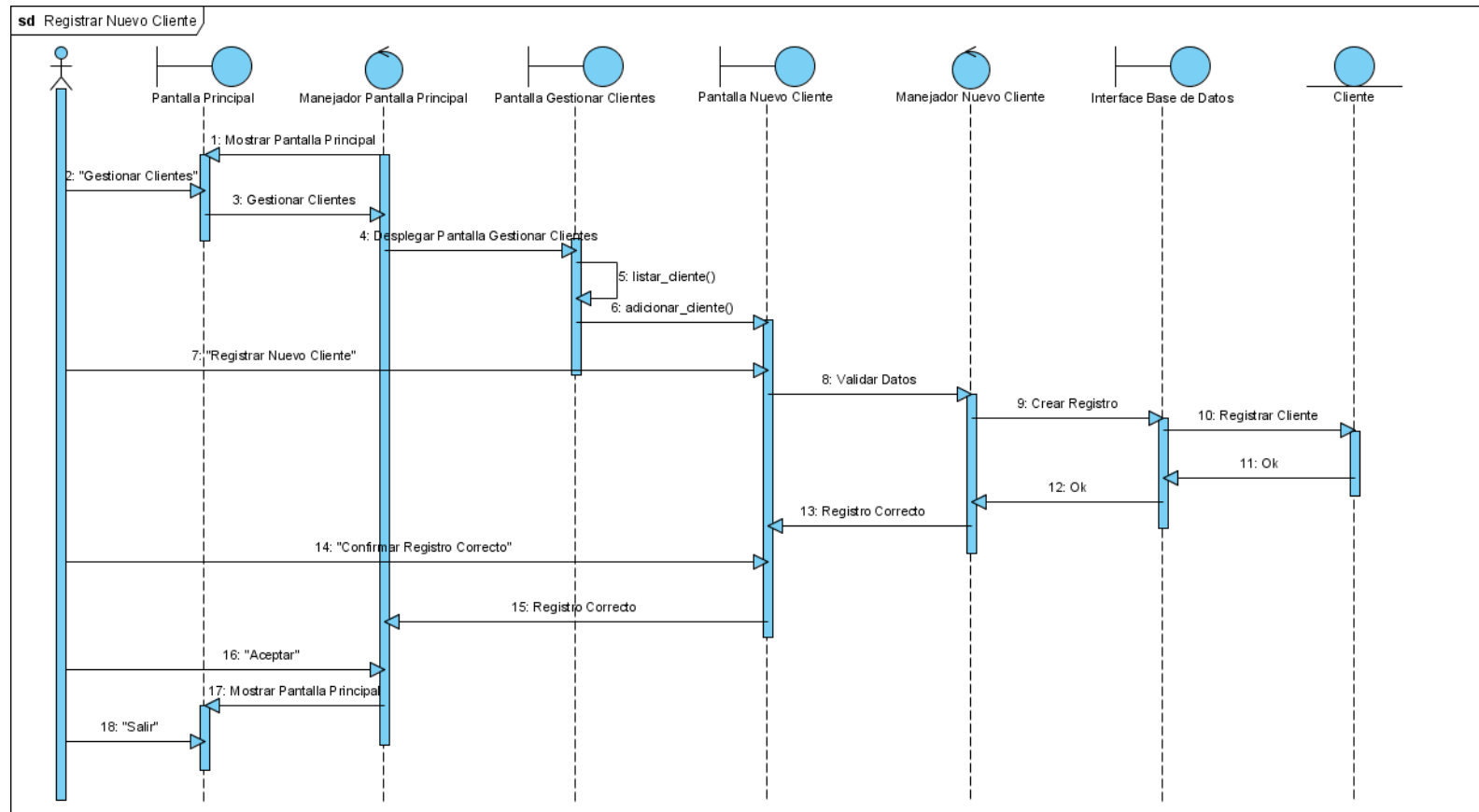


Figura 31 Diagrama Registrar Nuevo Cliente

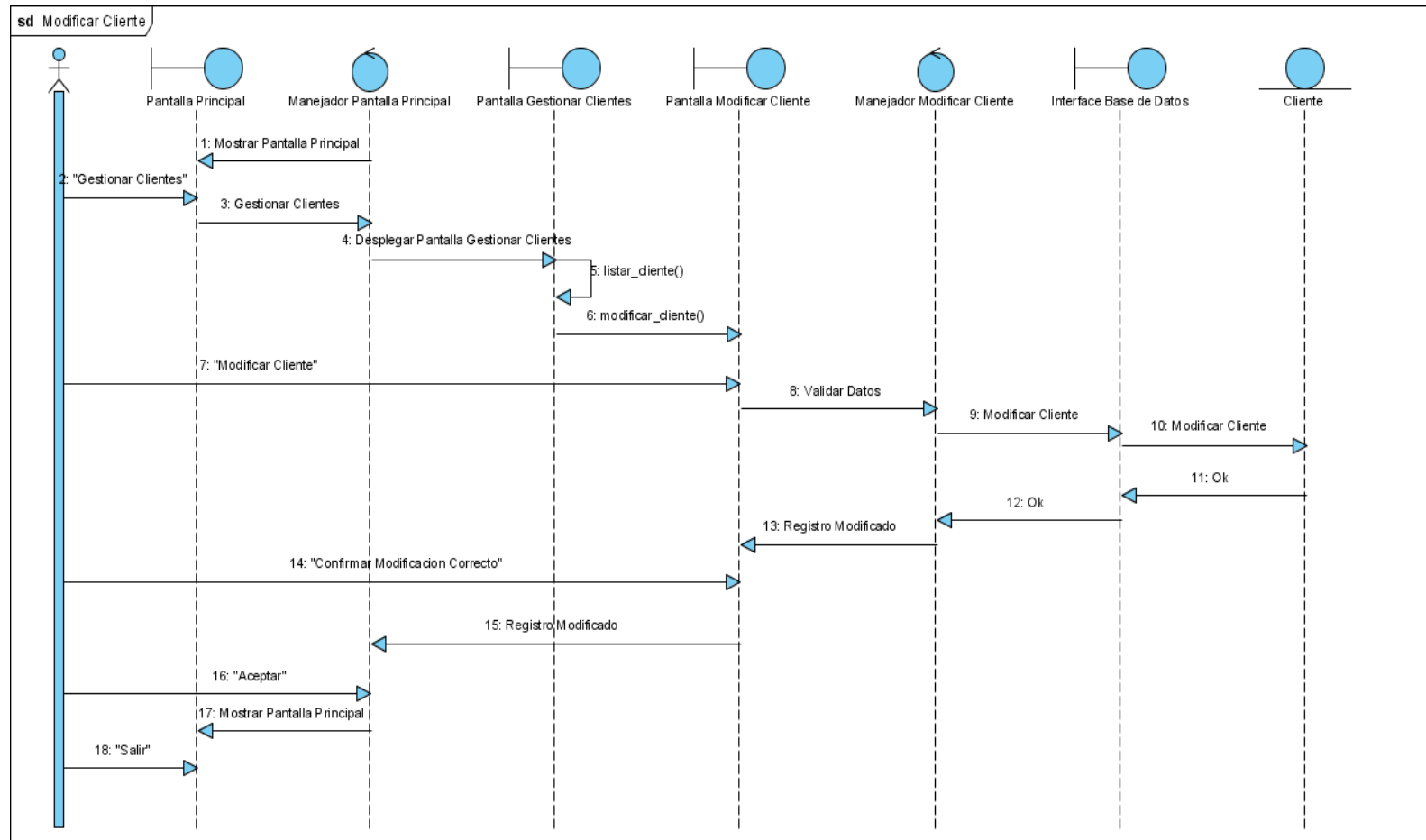


Figura 32 Diagrama Modificar Cliente

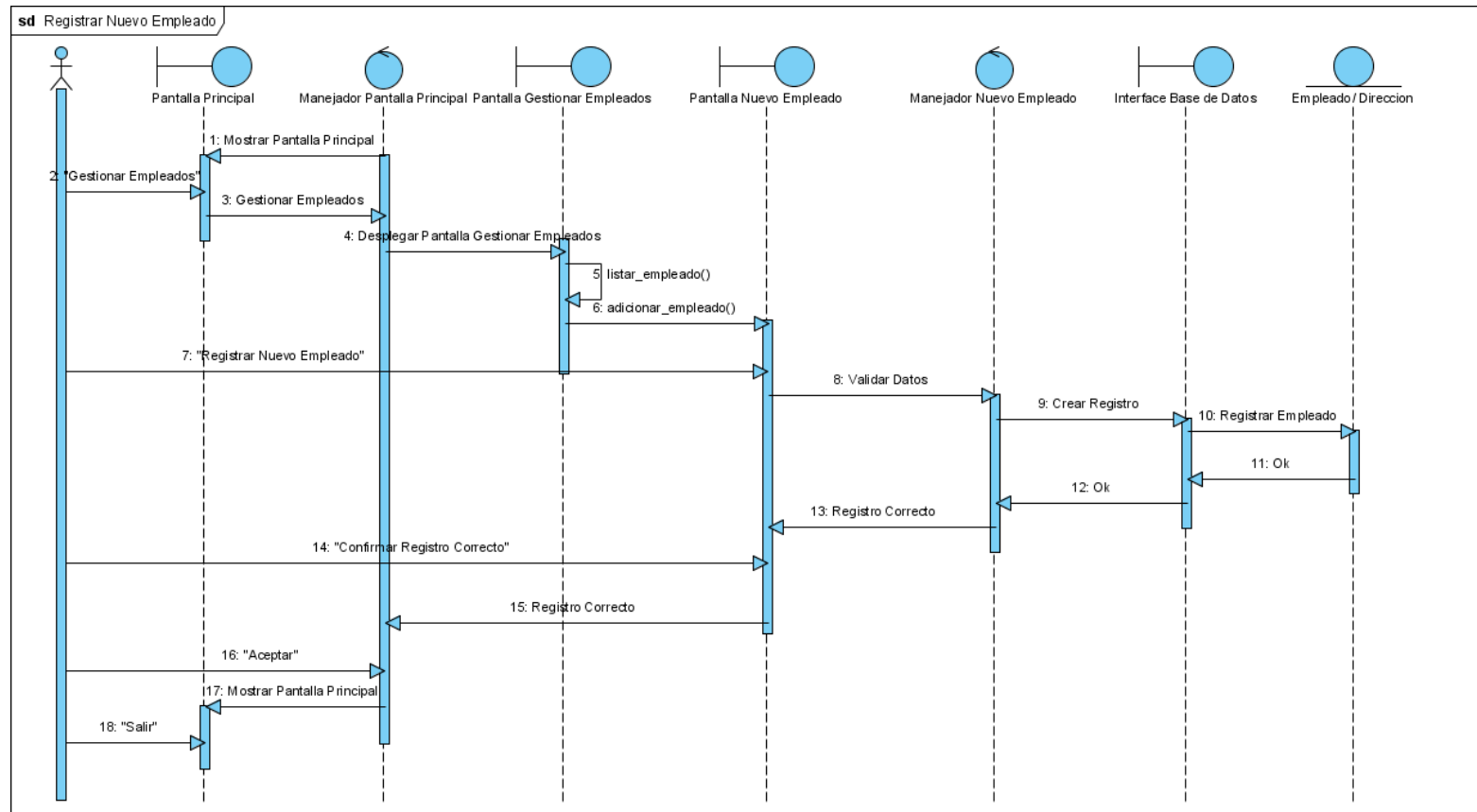


Figura 33 Diagrama Registrar Nuevo Empleado

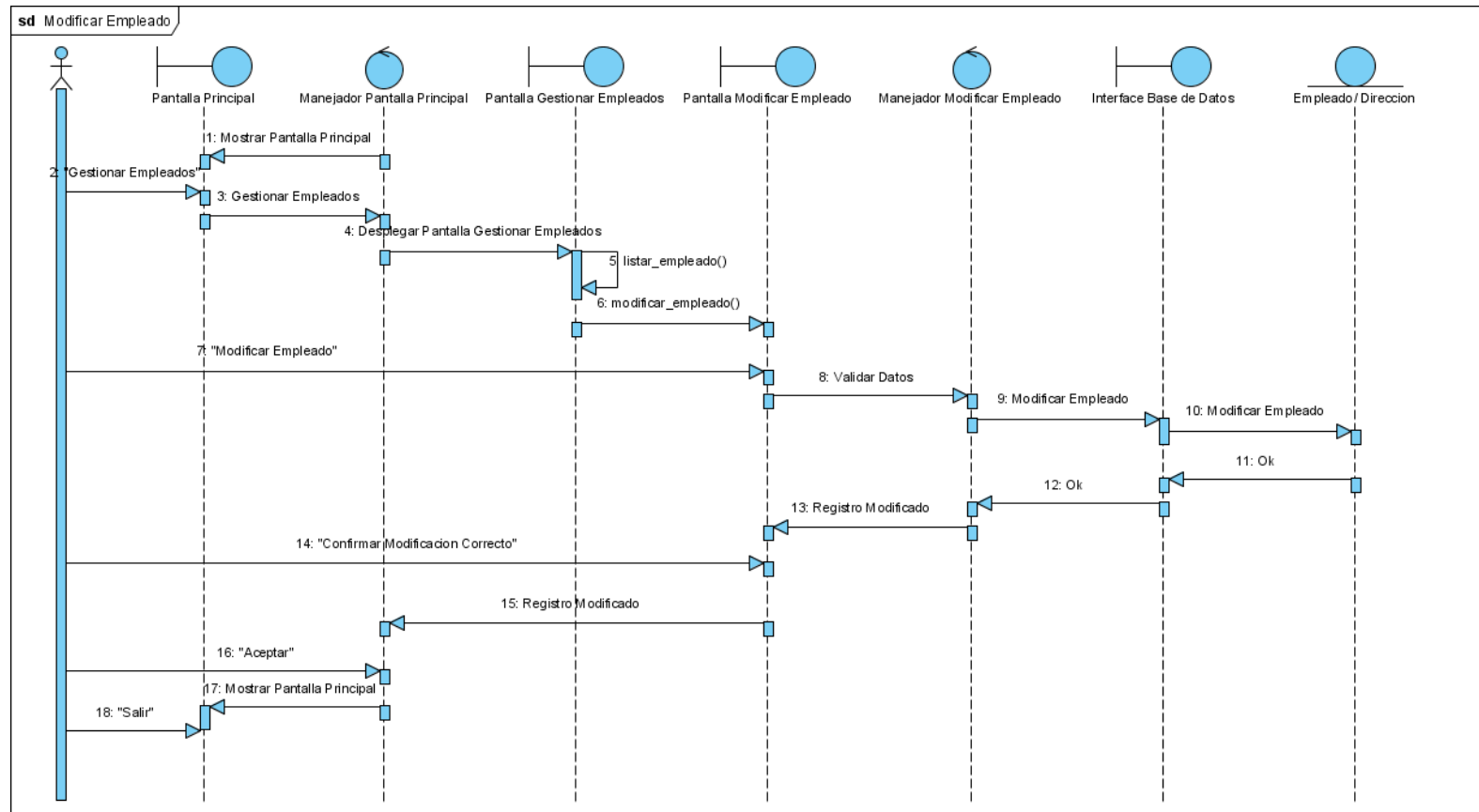


Figura 34 Diagrama Modificar Empleado

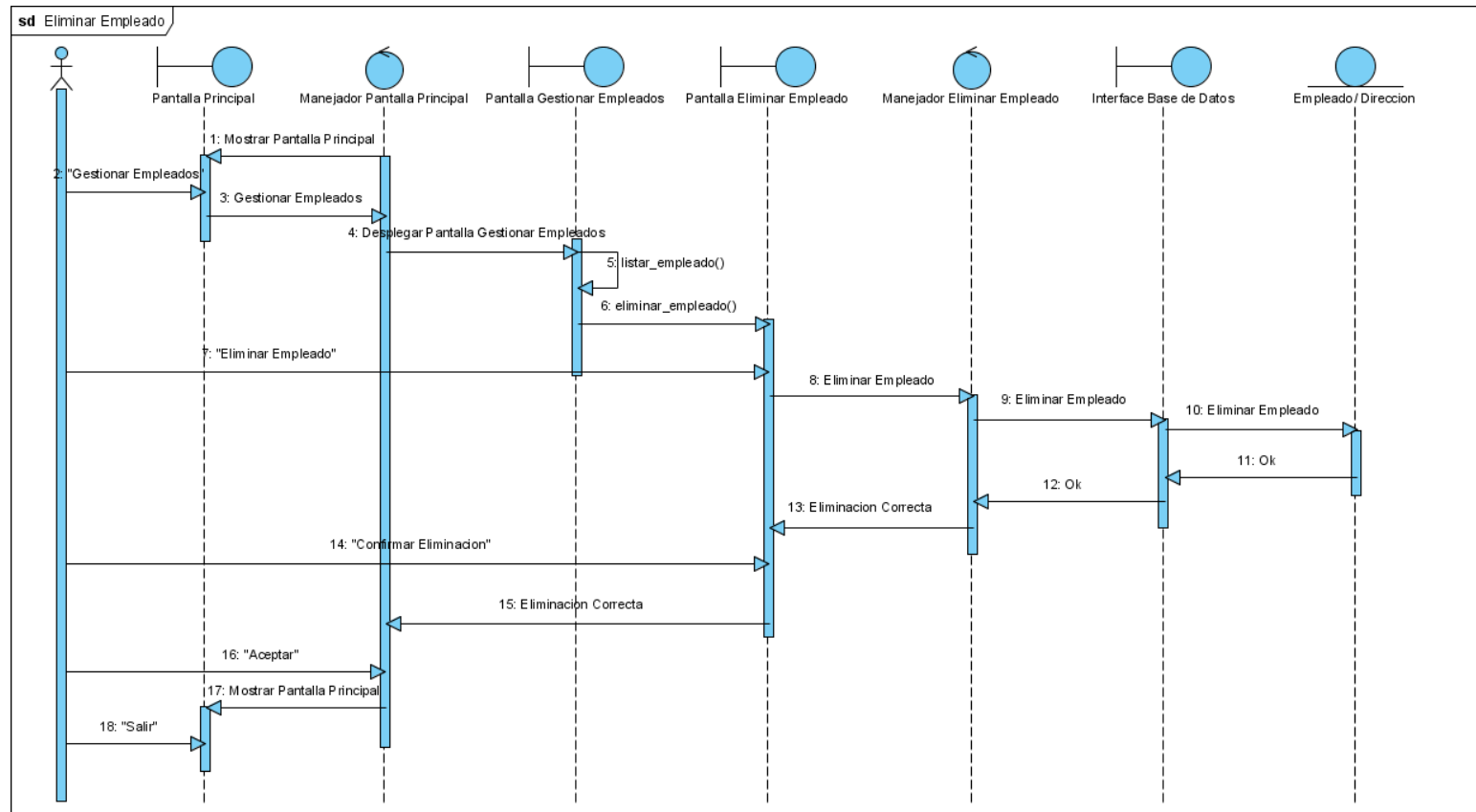


Figura 35Diagrama Eliminar Empleado

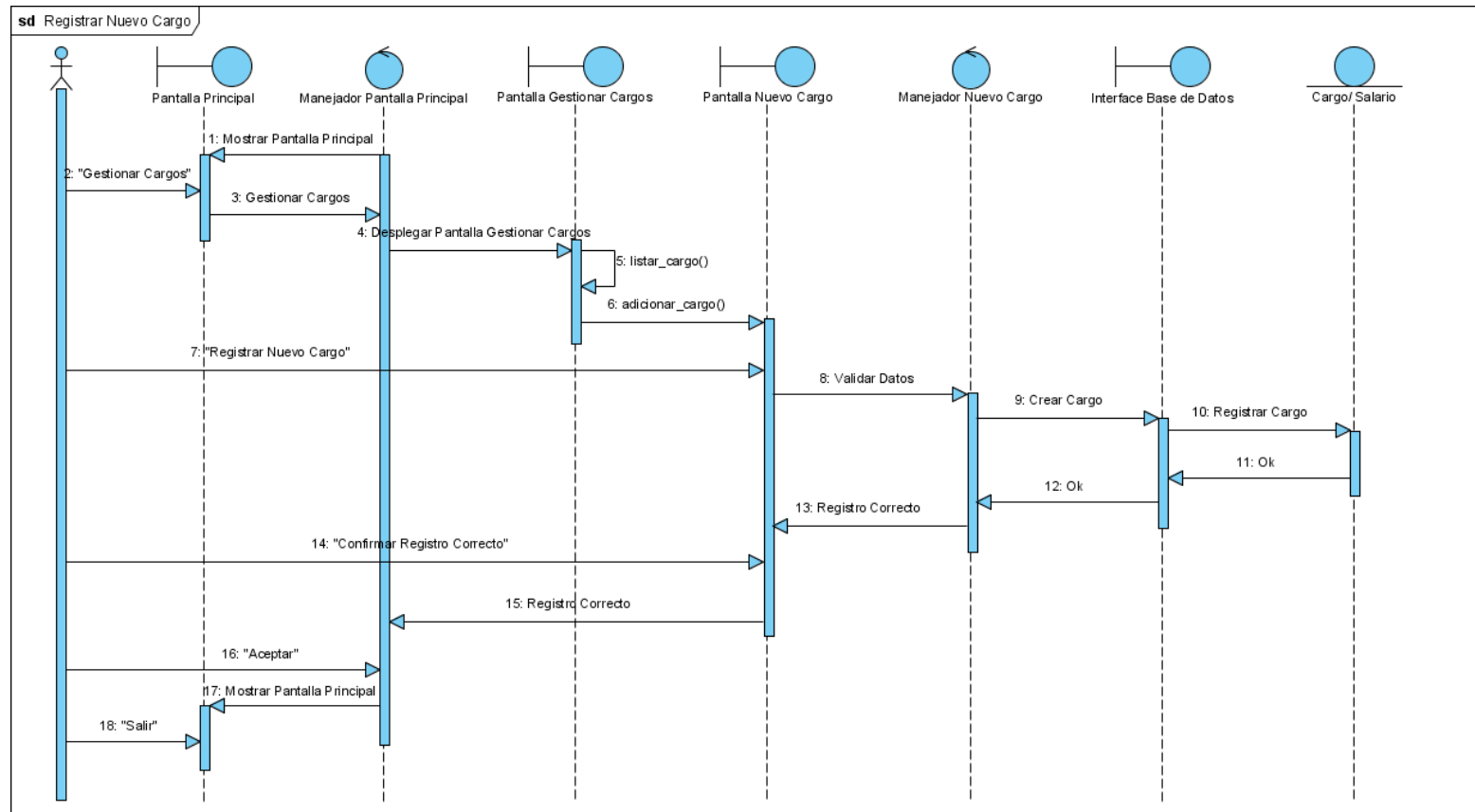


Figura 36 Diagrama Registrar Nuevo Cargo

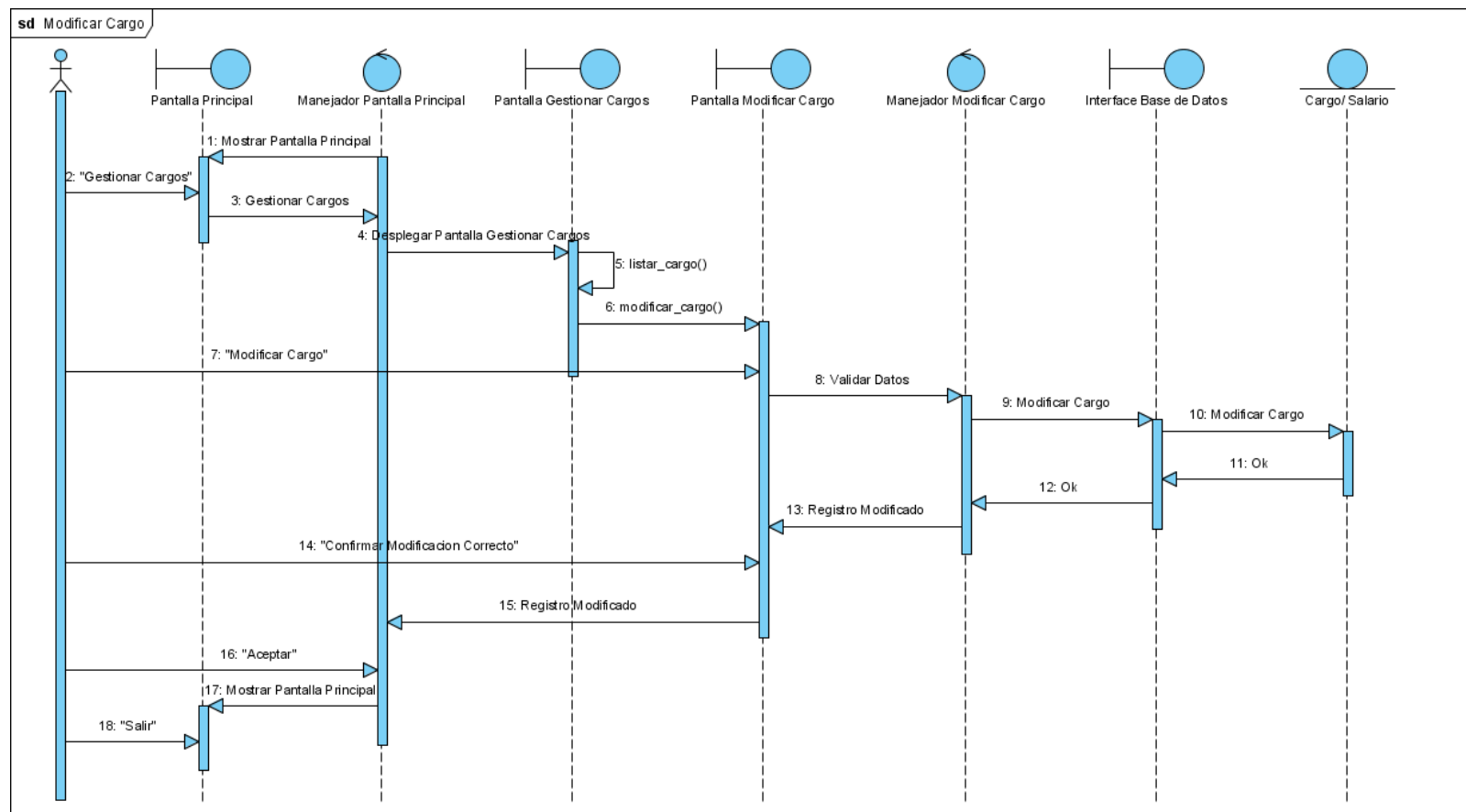


Figura 37 Diagrama Modificar Cargo

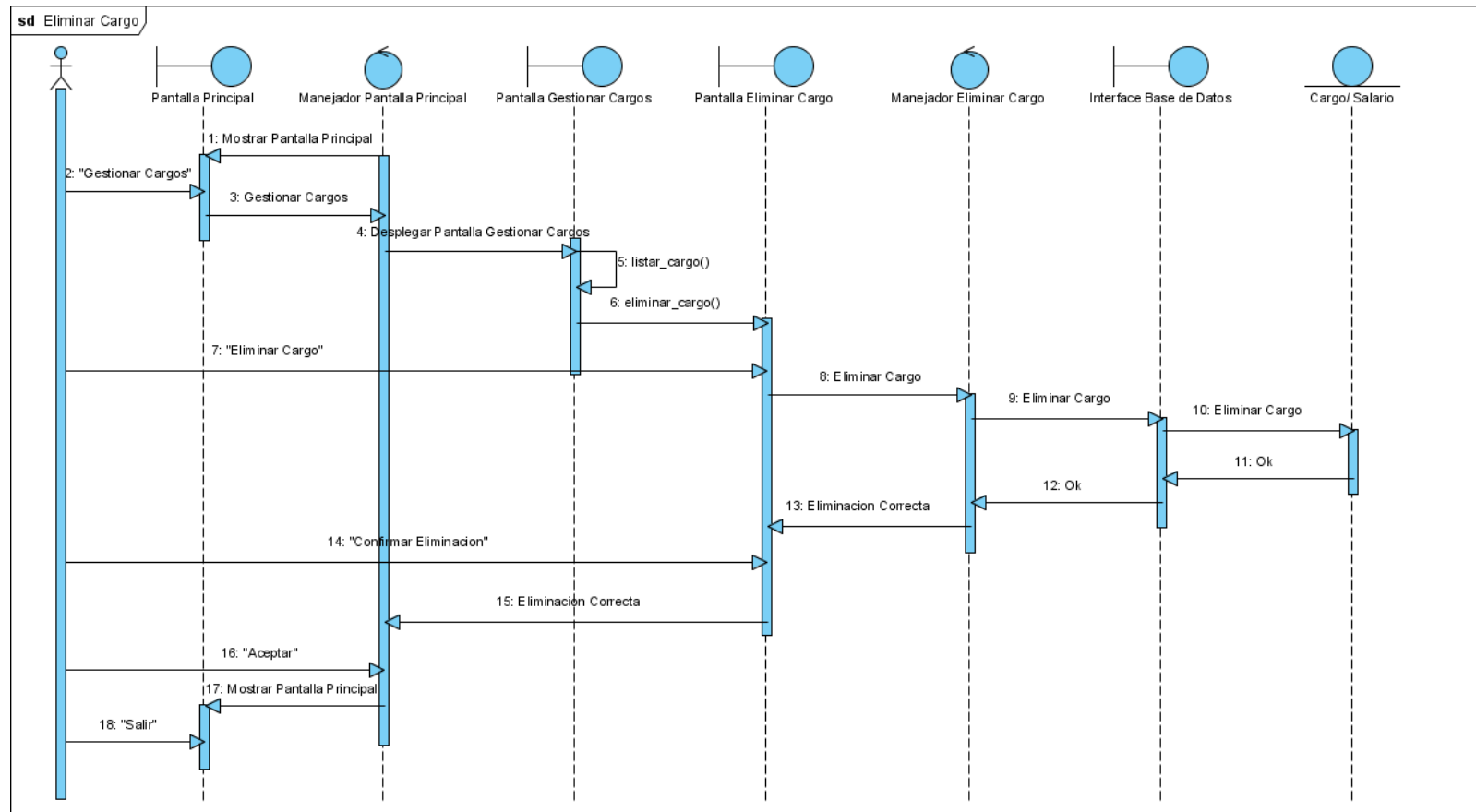


Figura 38 Diagrama Eliminar Cargo

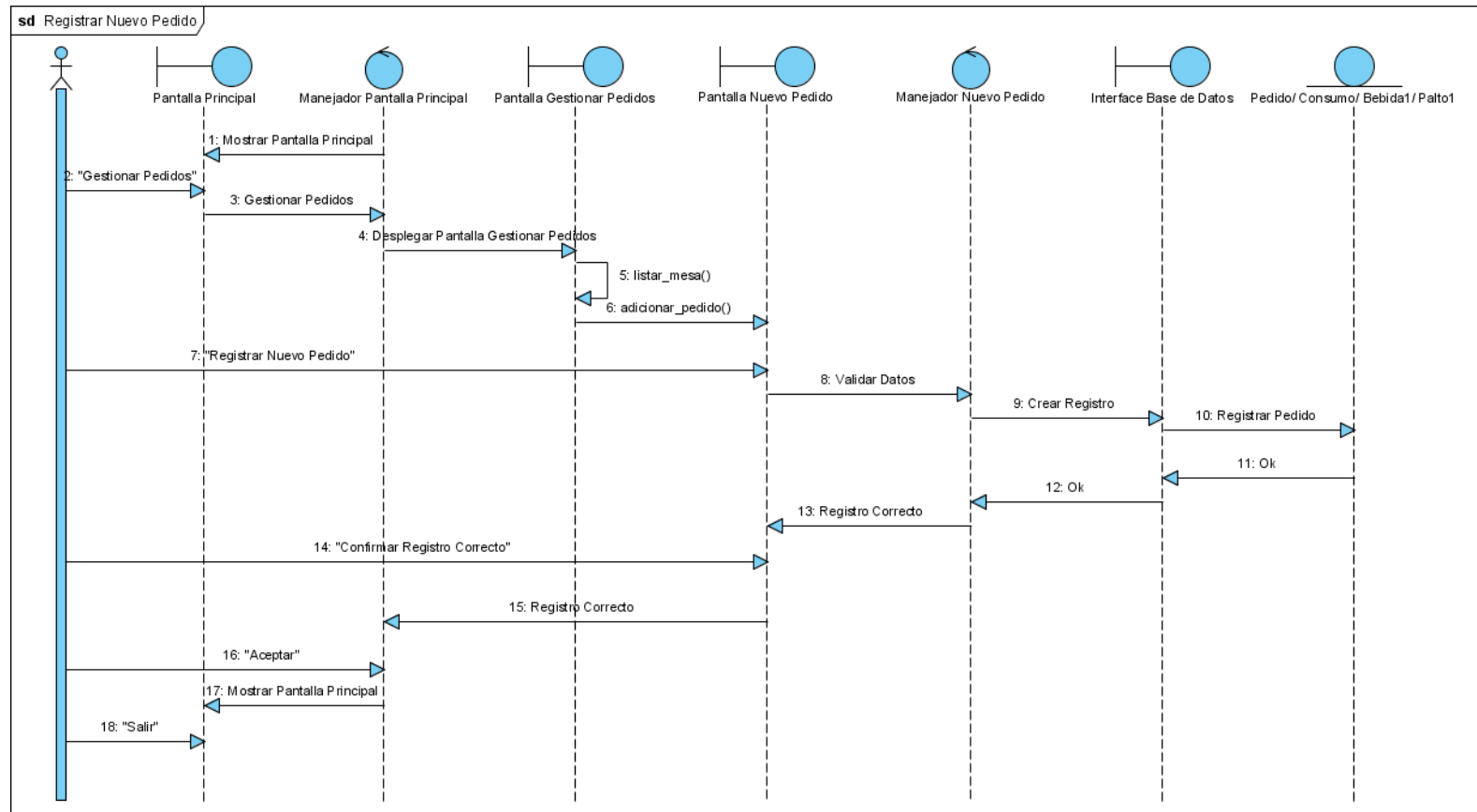


Figura 39 Diagrama Registrar Nuevo Pedido

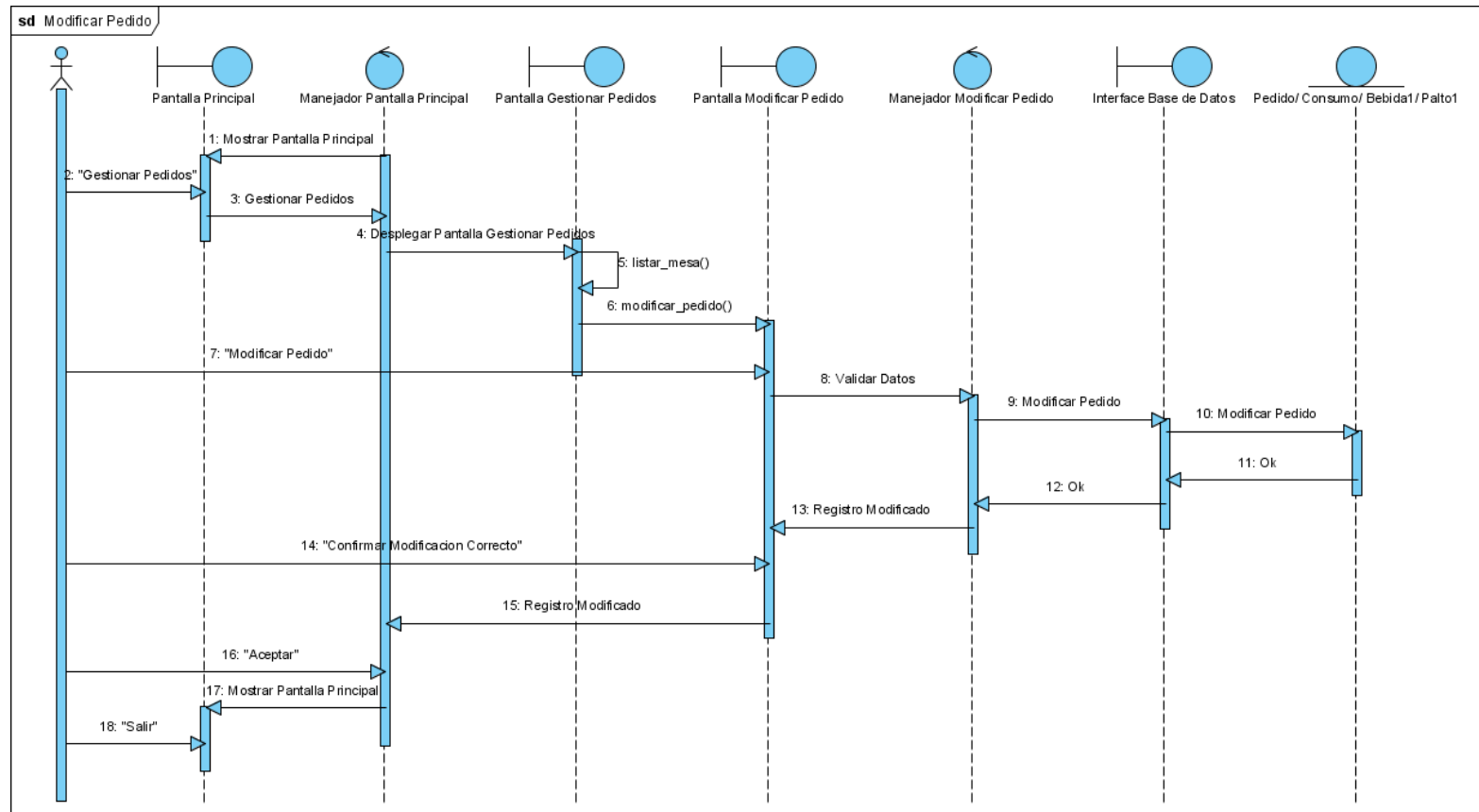


Figura 40 Diagrama Modificar Pedido

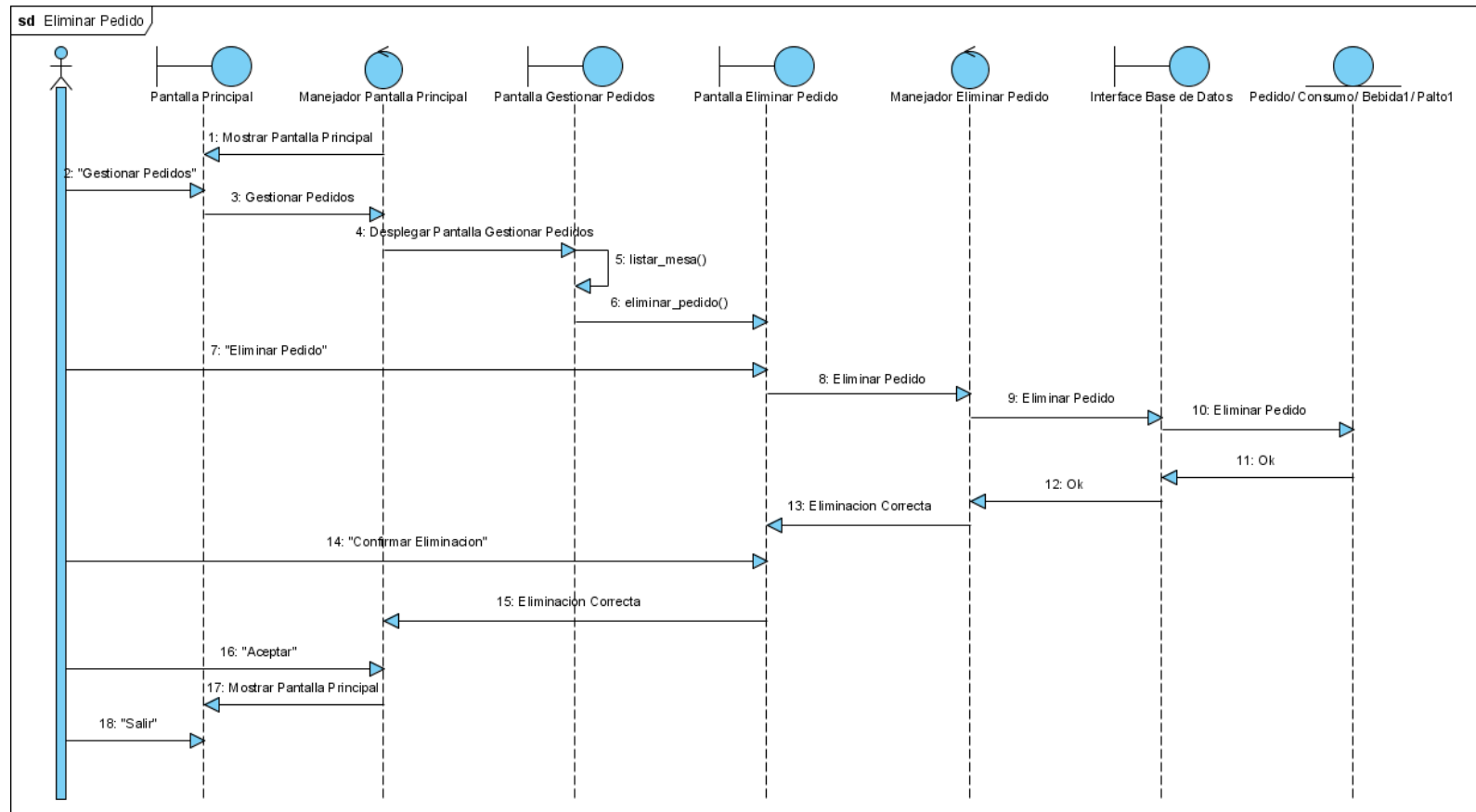


Figura 41 Diagrama Eliminar Pedido

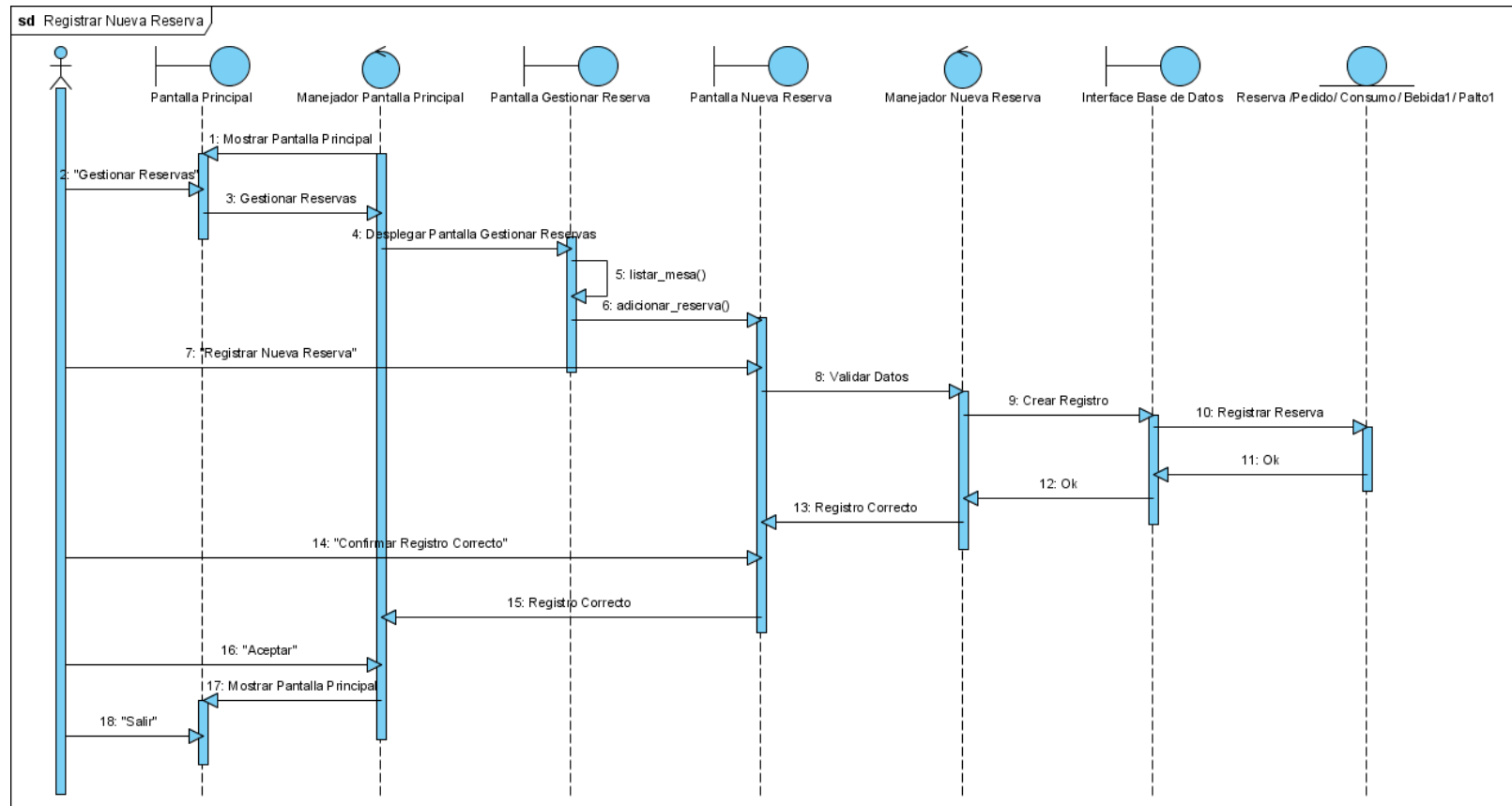


Figura 42 Diagrama Registrar Nueva Reserva

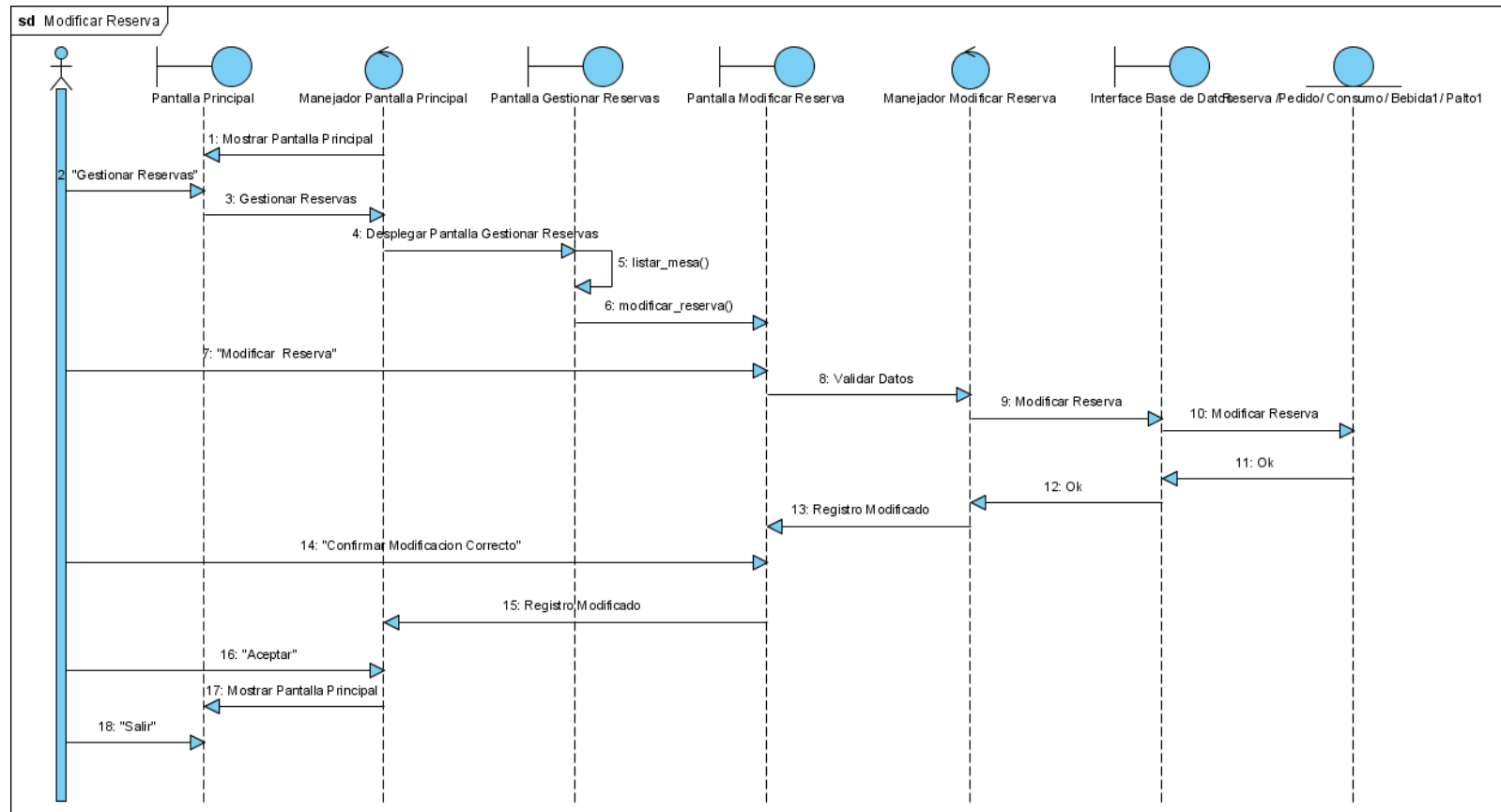


Figura 43 Diagrama Modificar Reserva

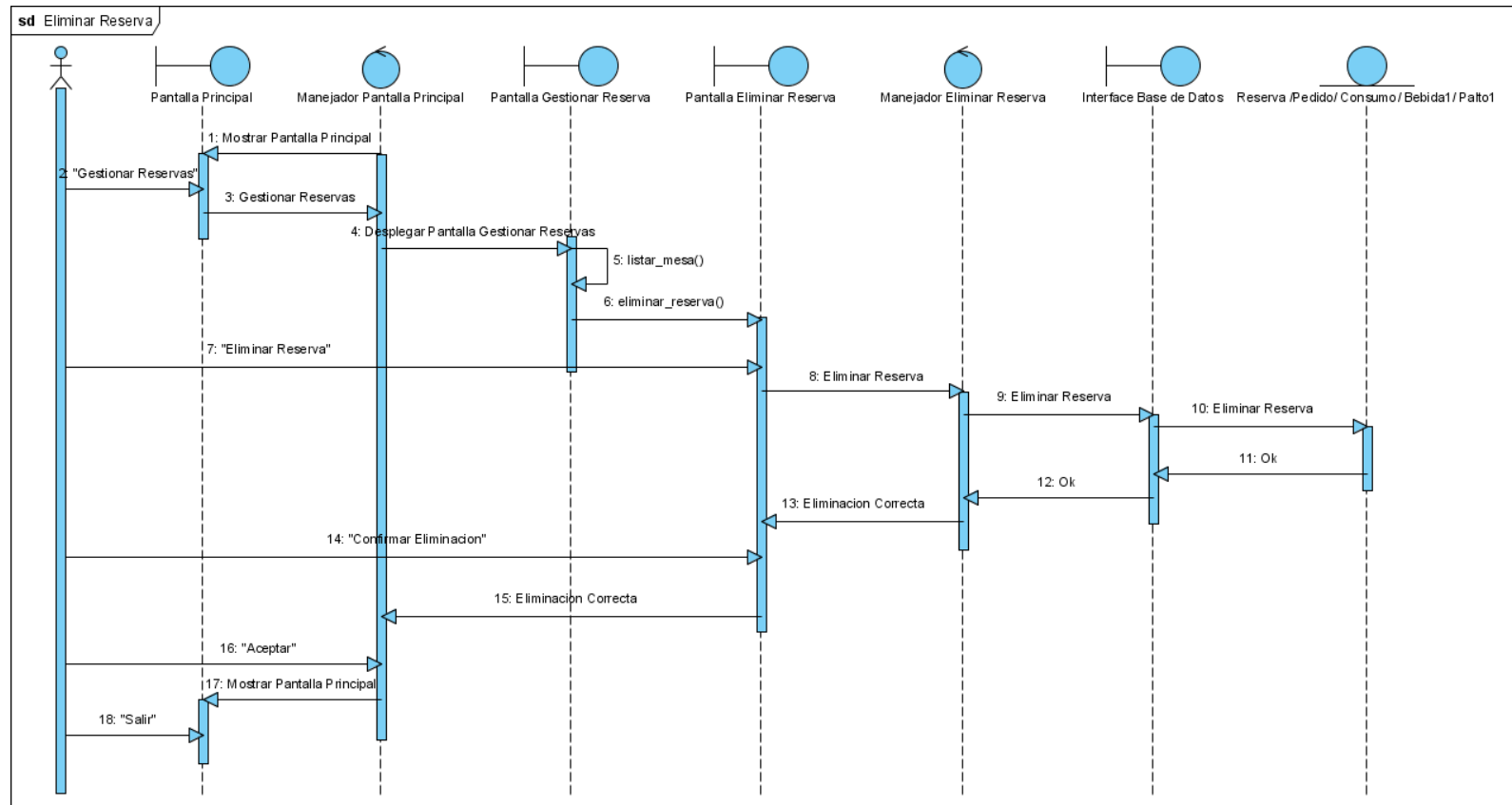


Figura 44 Diagrama Eliminar Reserva

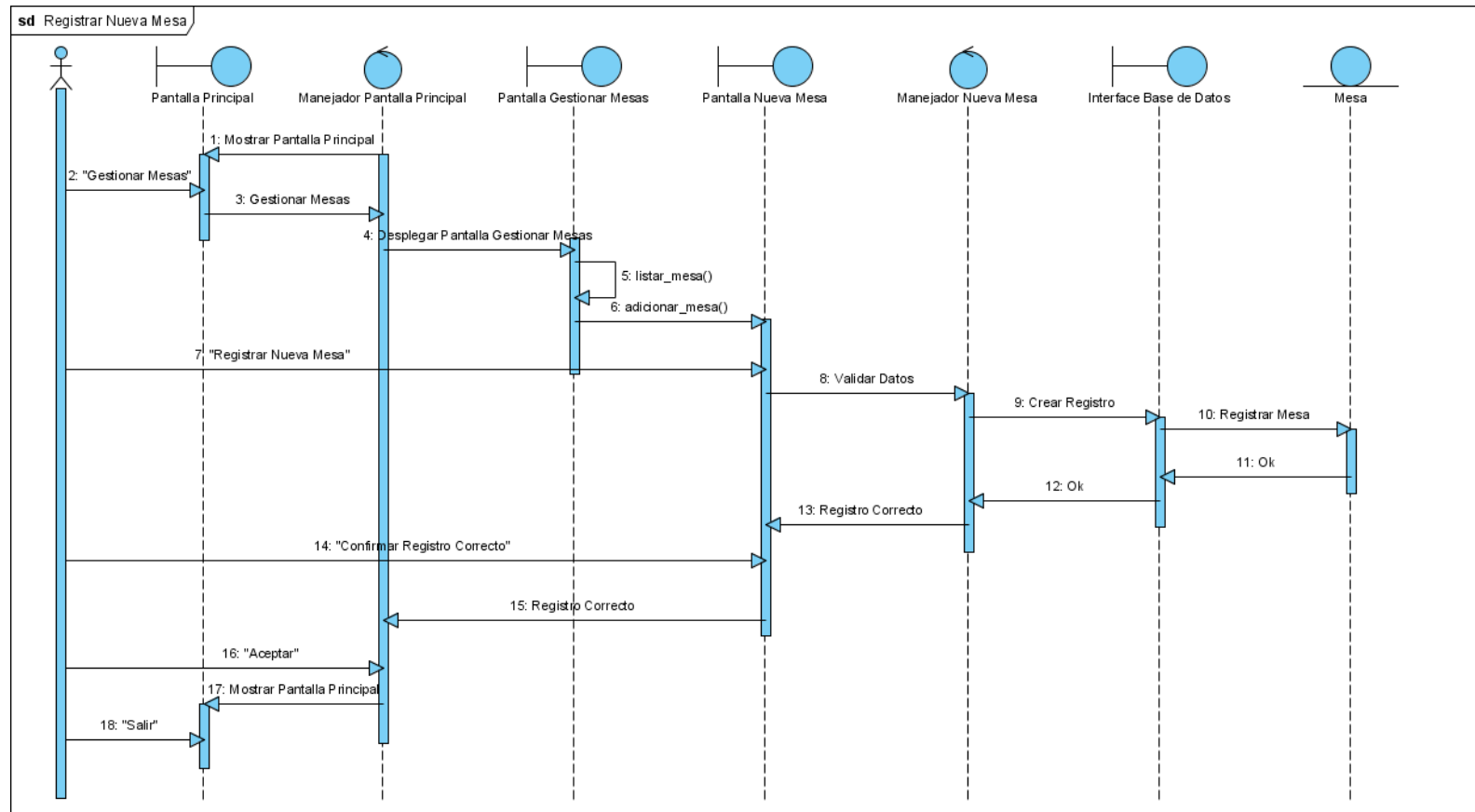


Figura 45 Diagrama Registrar Nueva Mesa

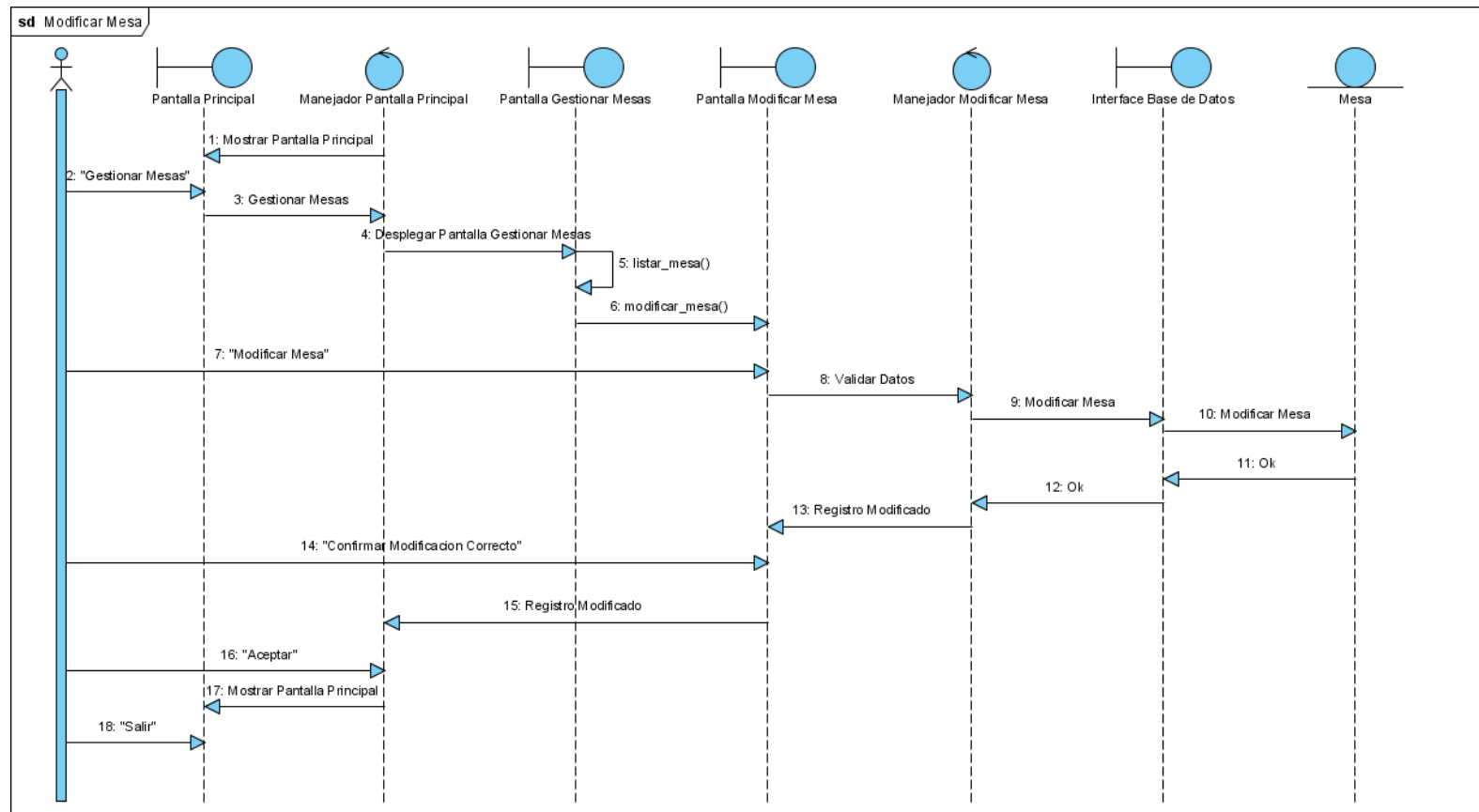


Figura 46 Diagrama Modificar Mesa

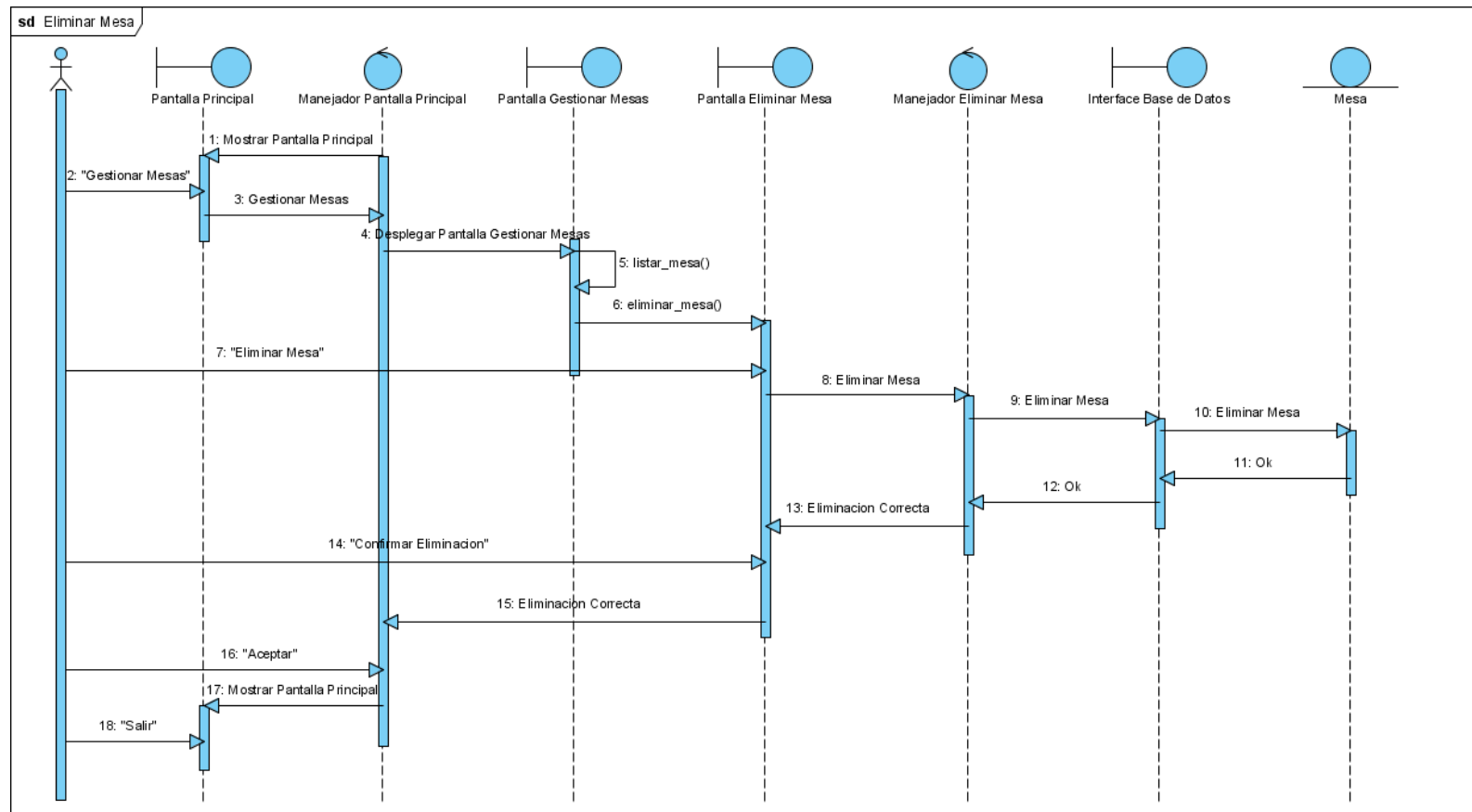


Figura 47 Diagrama Eliminar Mesa

II.1.5.1.3. Diagramas de Comunicación

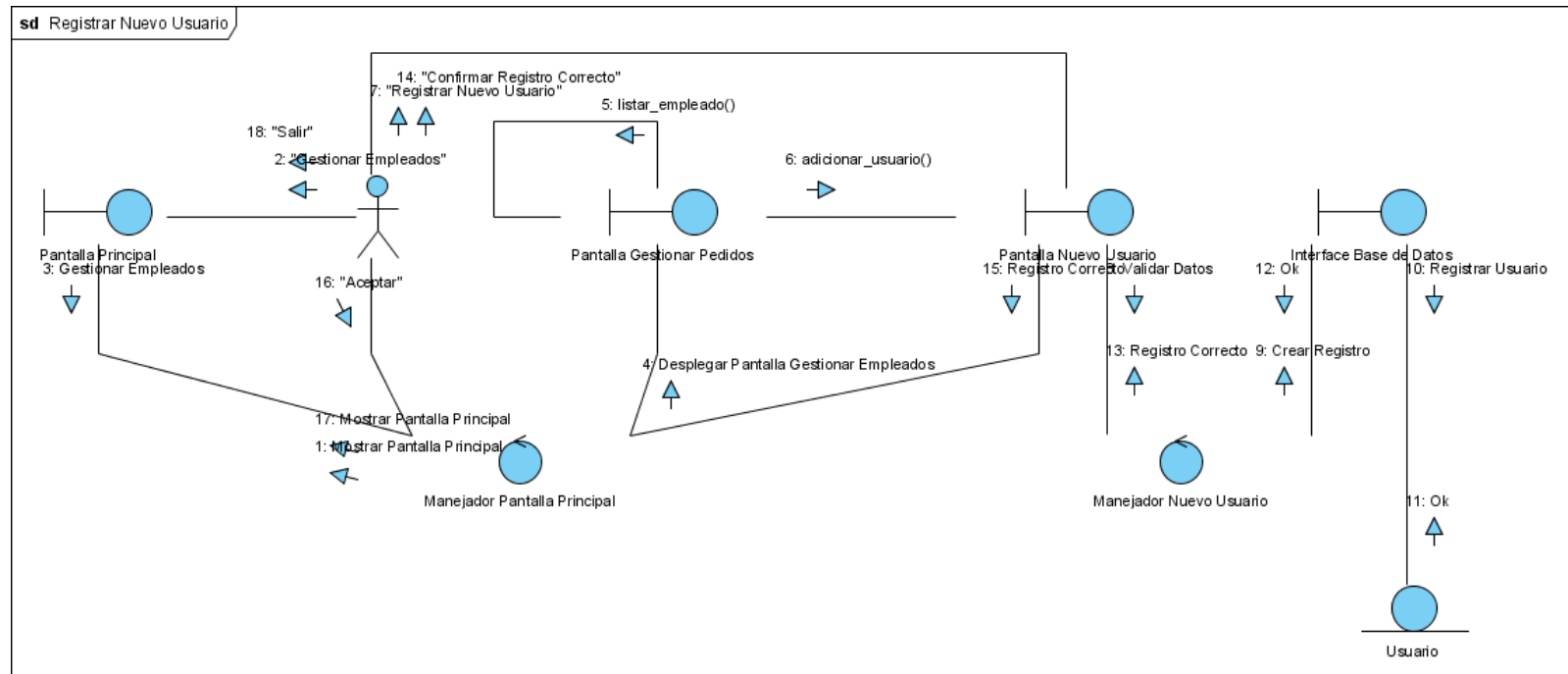


Figura 48 Diagrama Registrar Nuevo Usuario

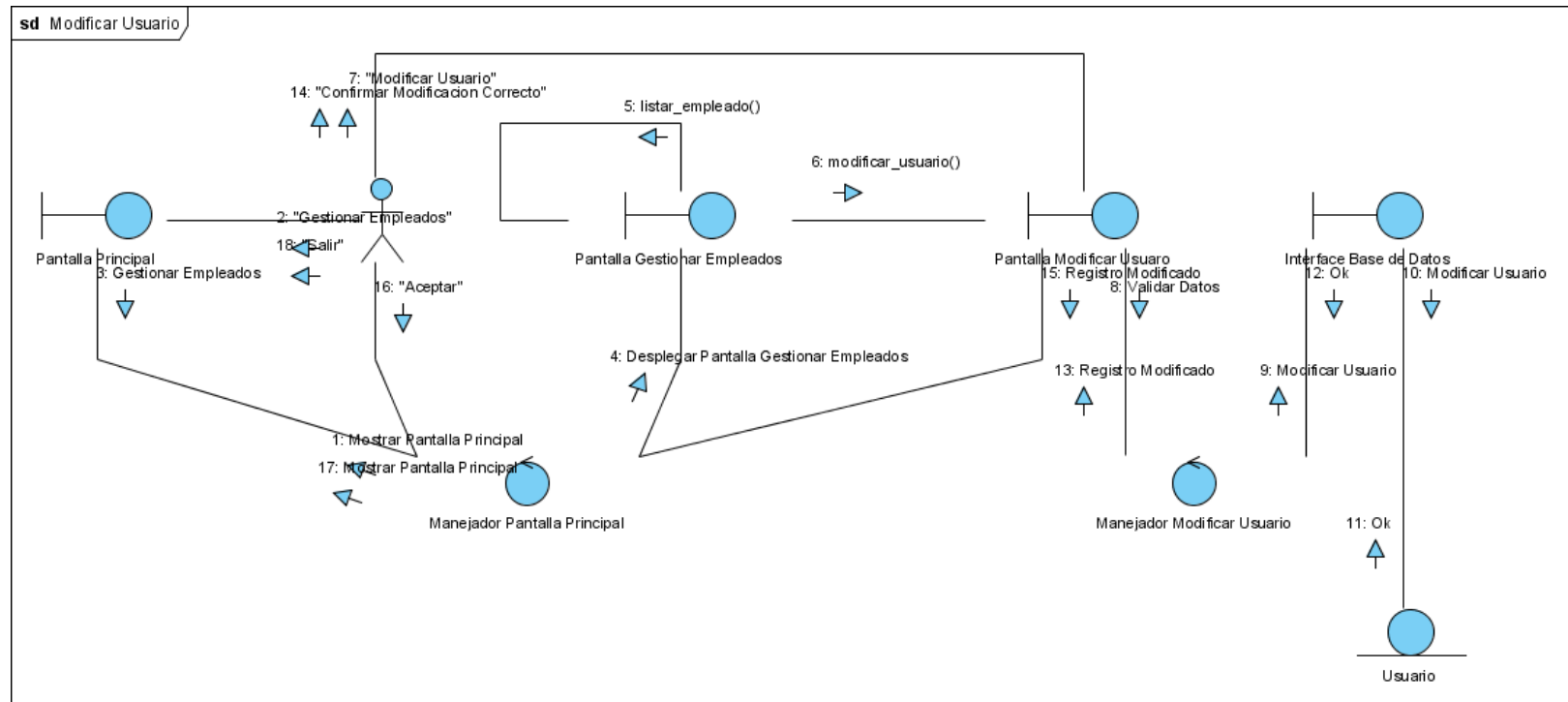


Figura 49 Diagrama Modificar Usuario

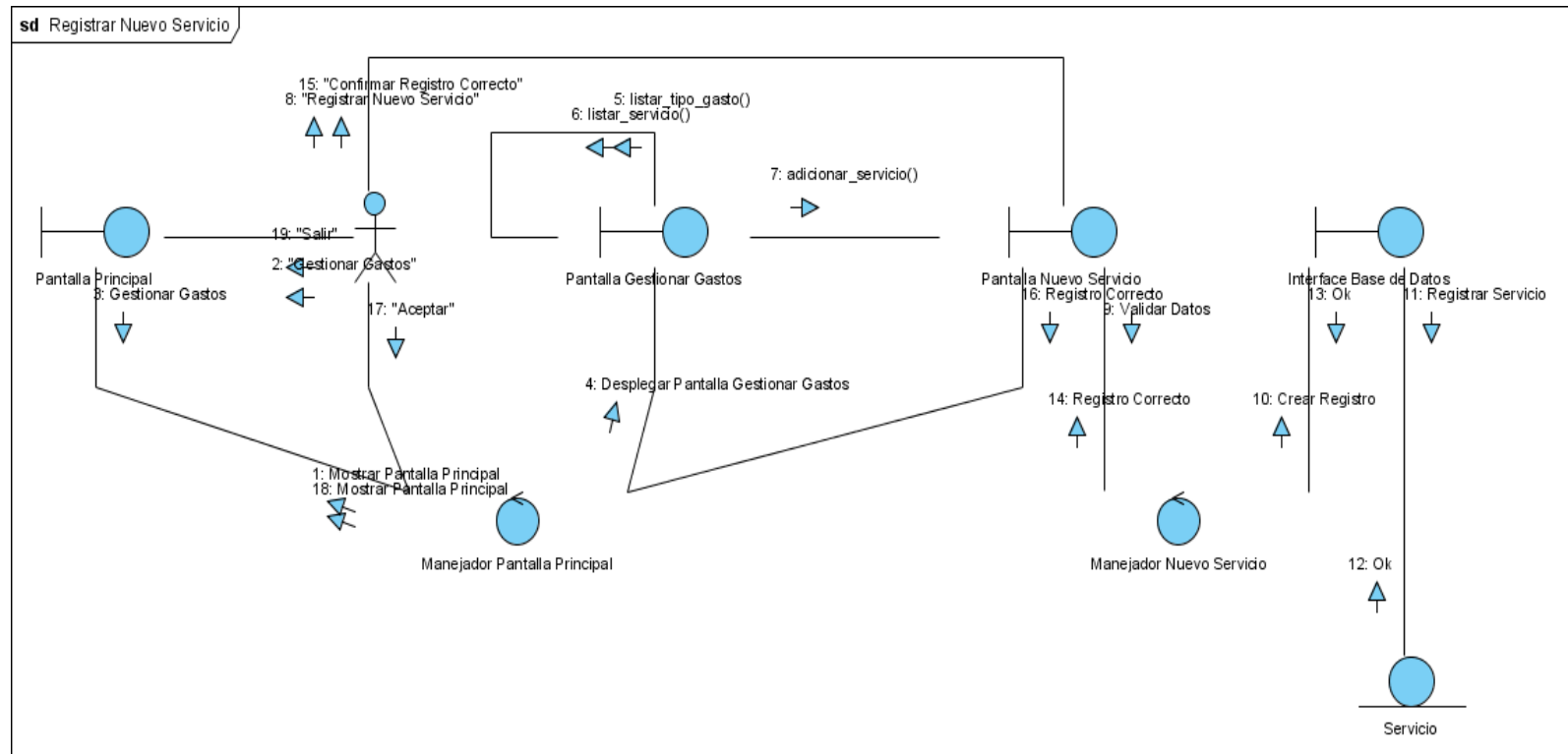


Figura 50 Diagrama Registrar Nuevo Servicio

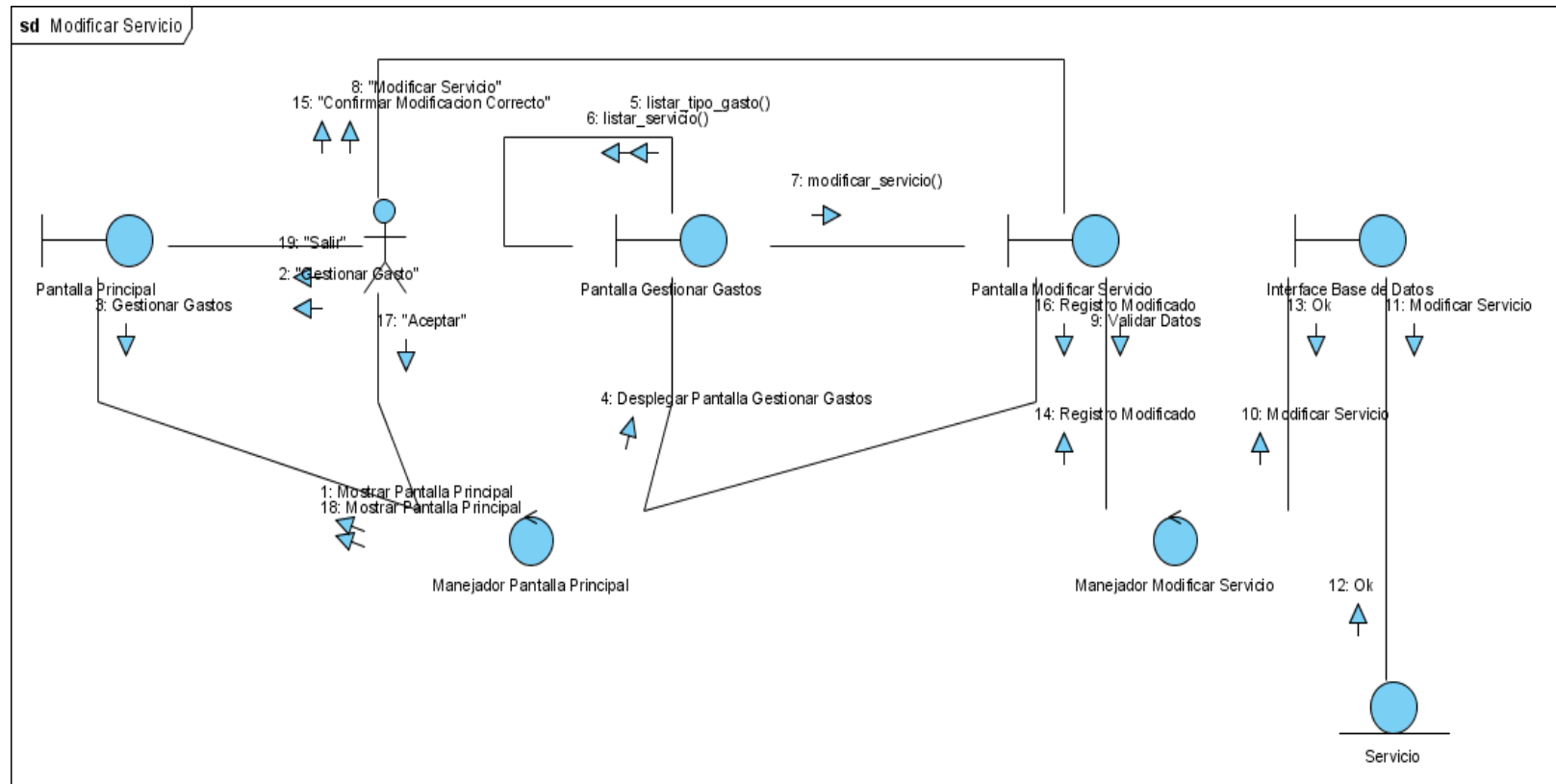


Figura 51 Diagrama Modificar Servicio

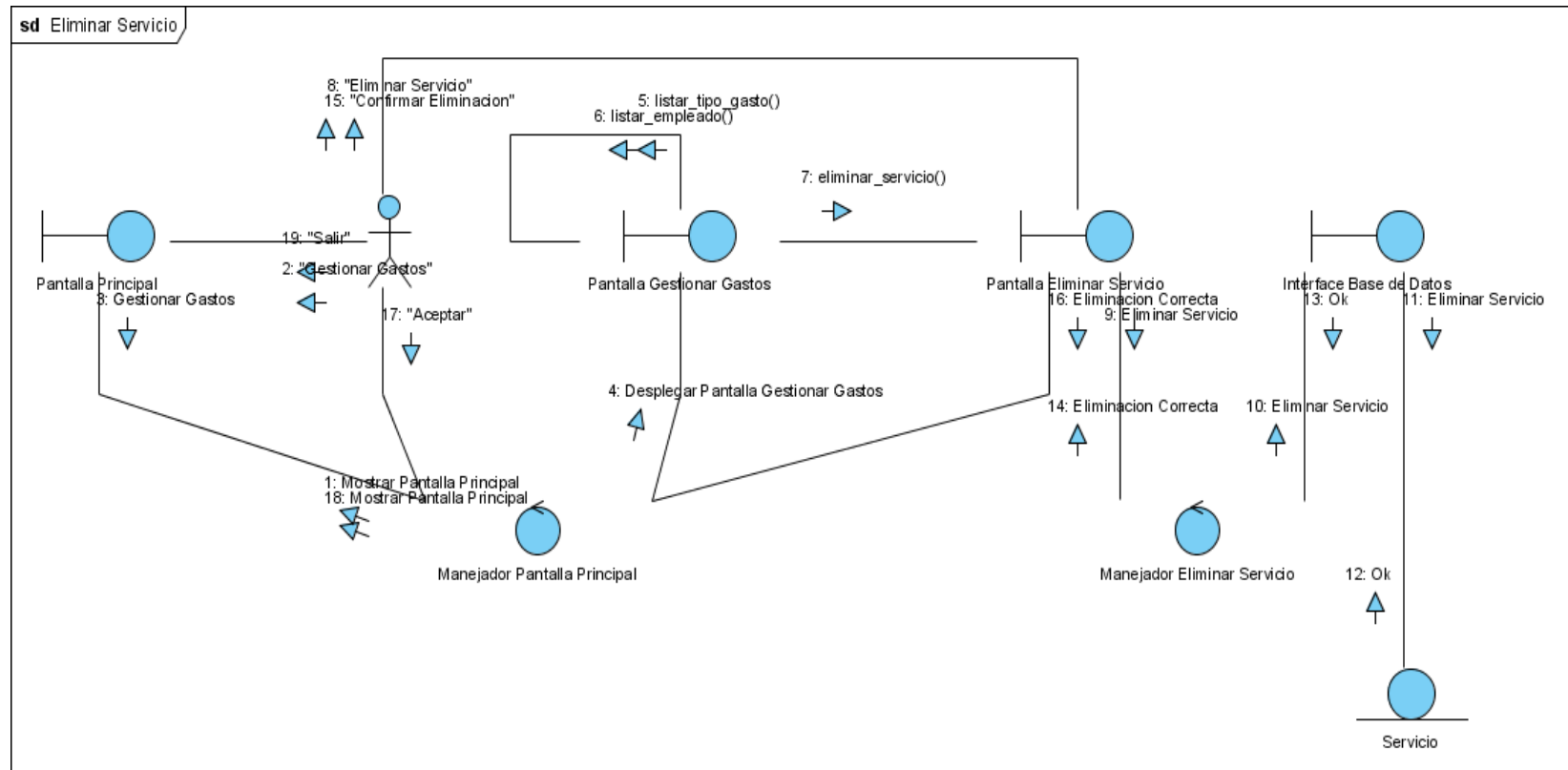


Figura 52 Diagrama Eliminar Servicio

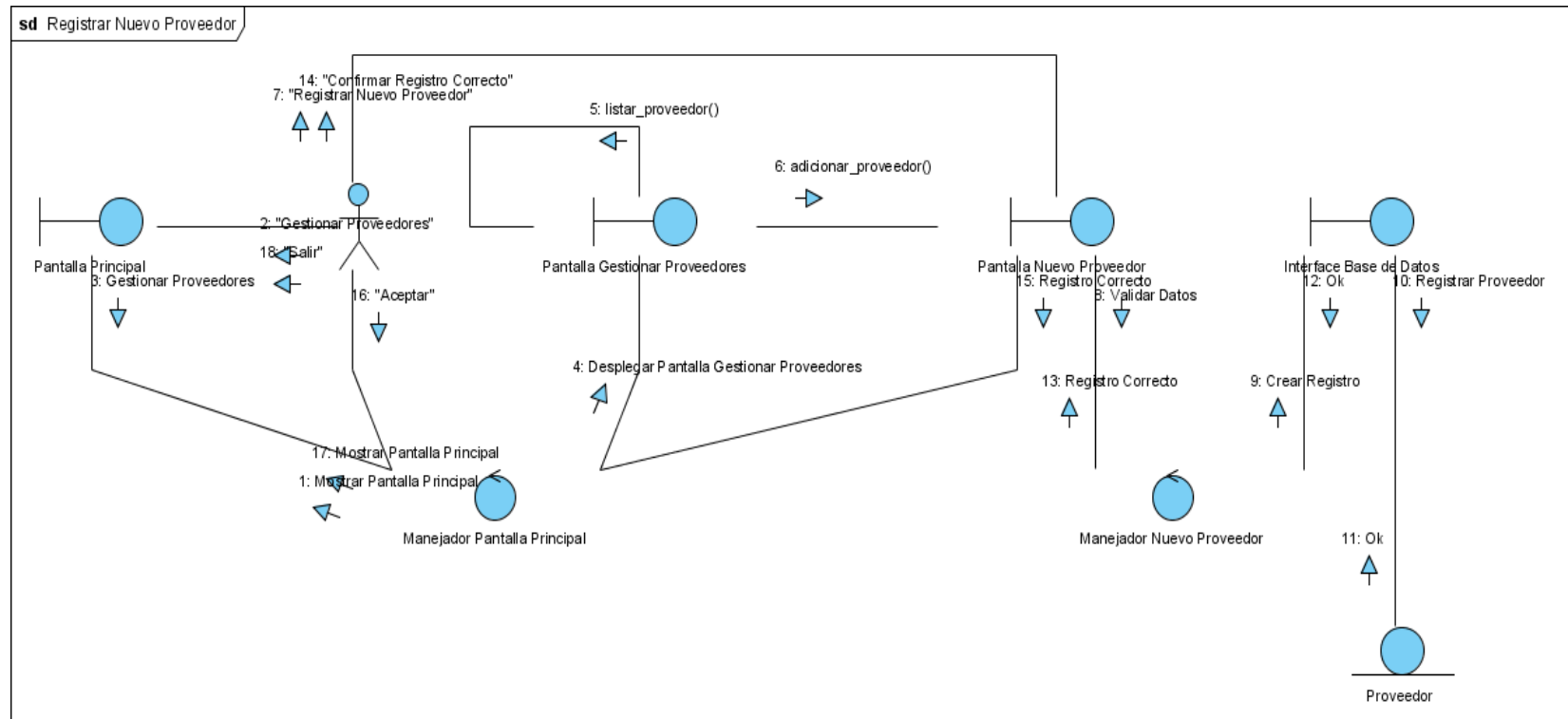


Figura 53 Diagrama Registrar Nuevo Proveedor

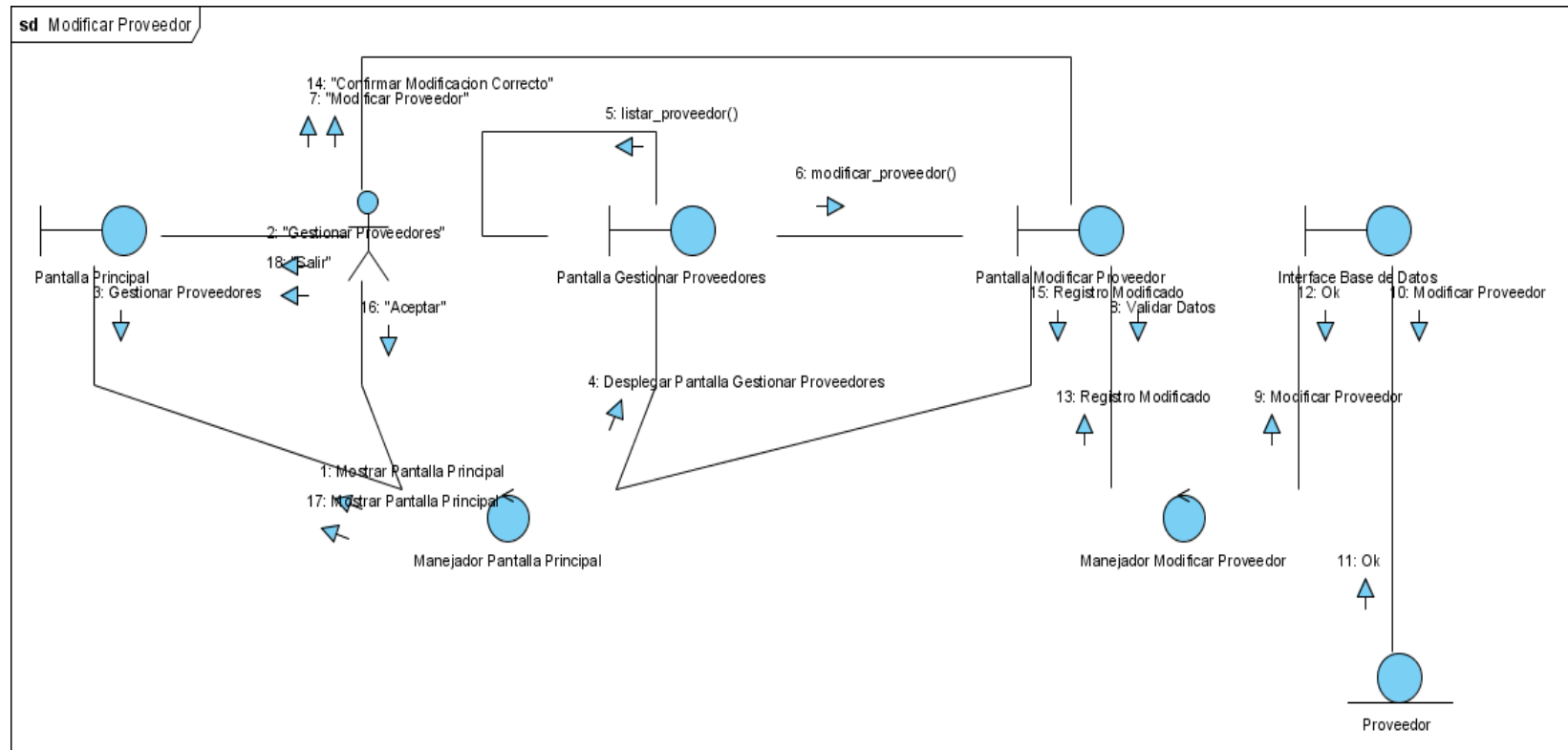


Figura 54 Diagrama Modificar Proveedor

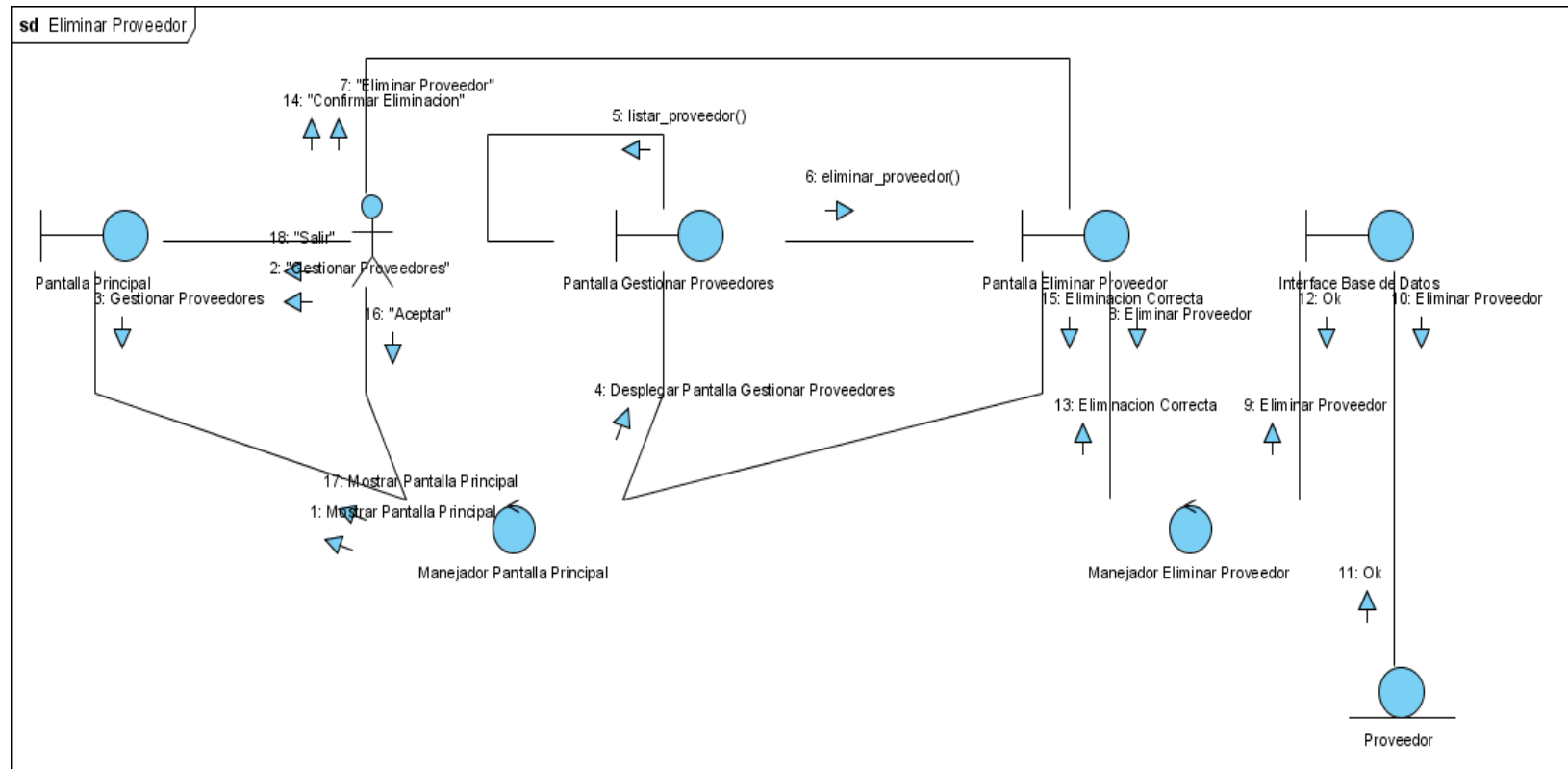


Figura 55 Diagrama Eliminar Proveedor

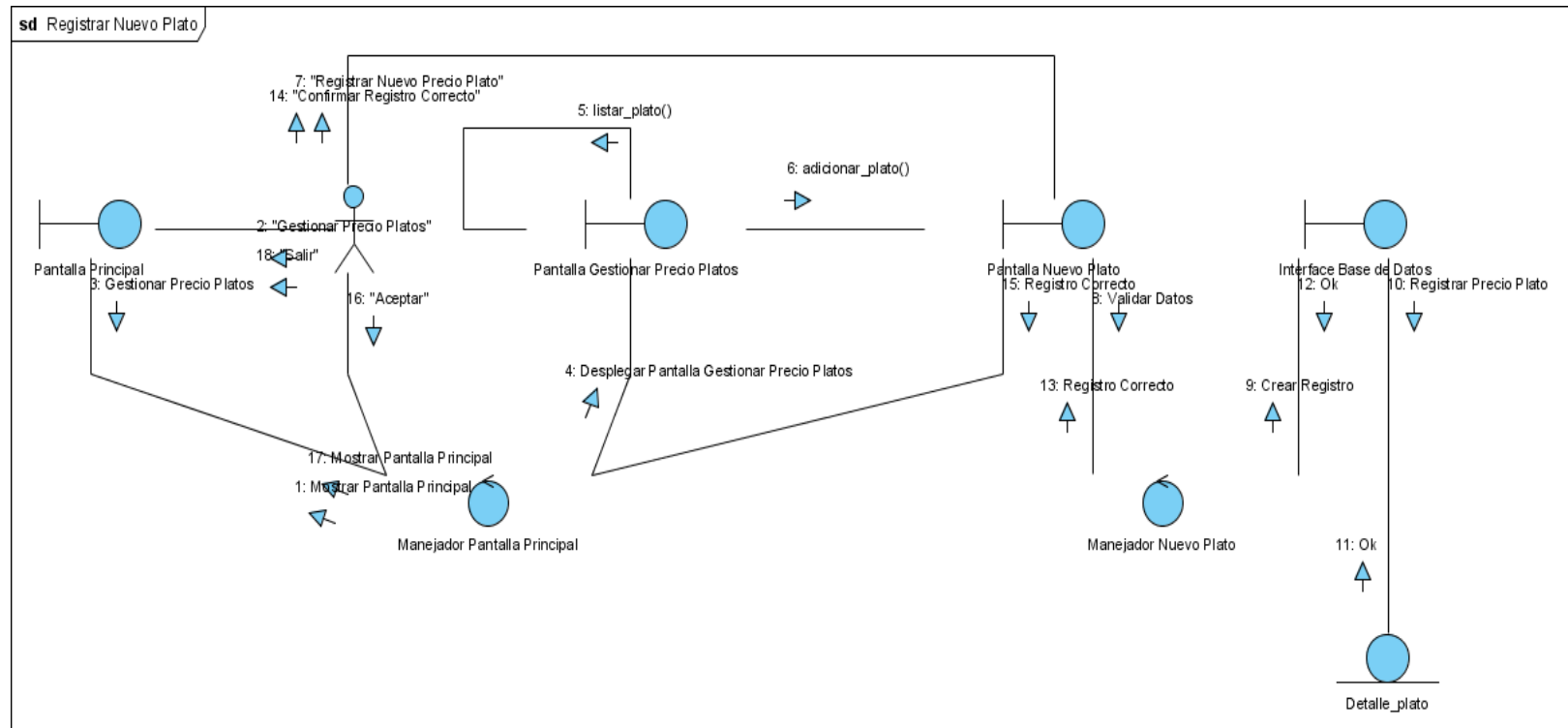


Figura 56 Diagrama Registrar Nuevo Plato

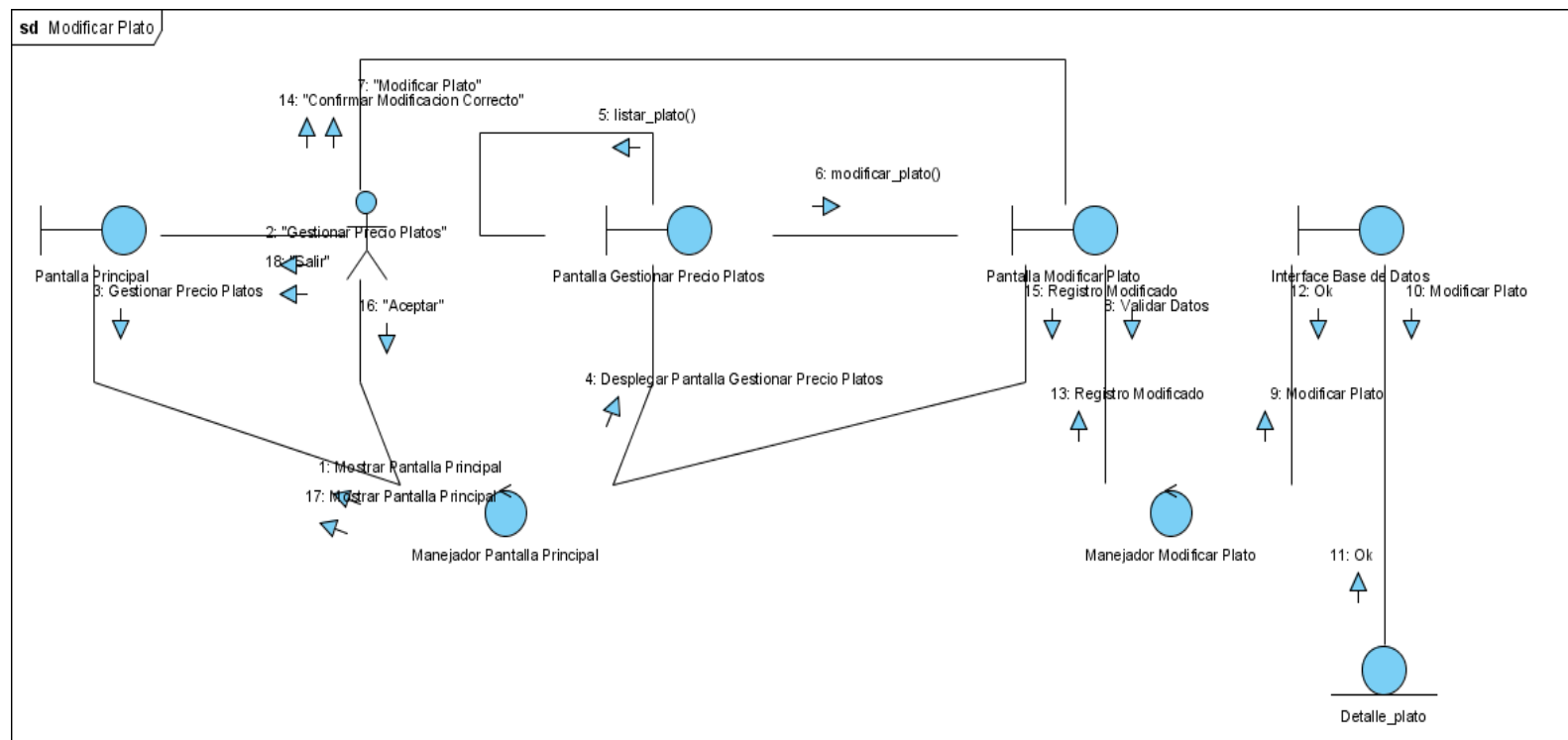


Figura 57 Diagrama Modificar Plato

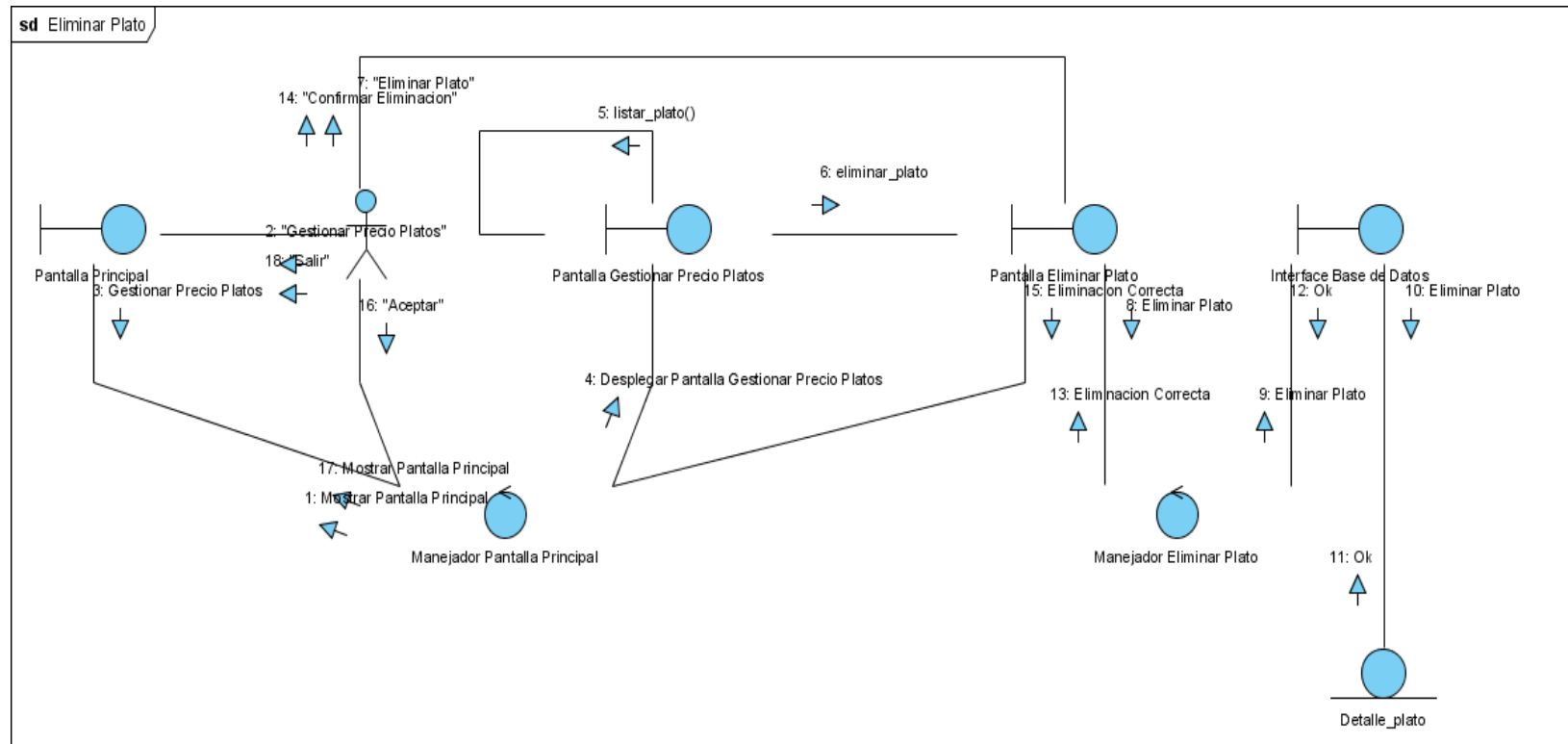


Figura 58 Diagrama Eliminar Plato

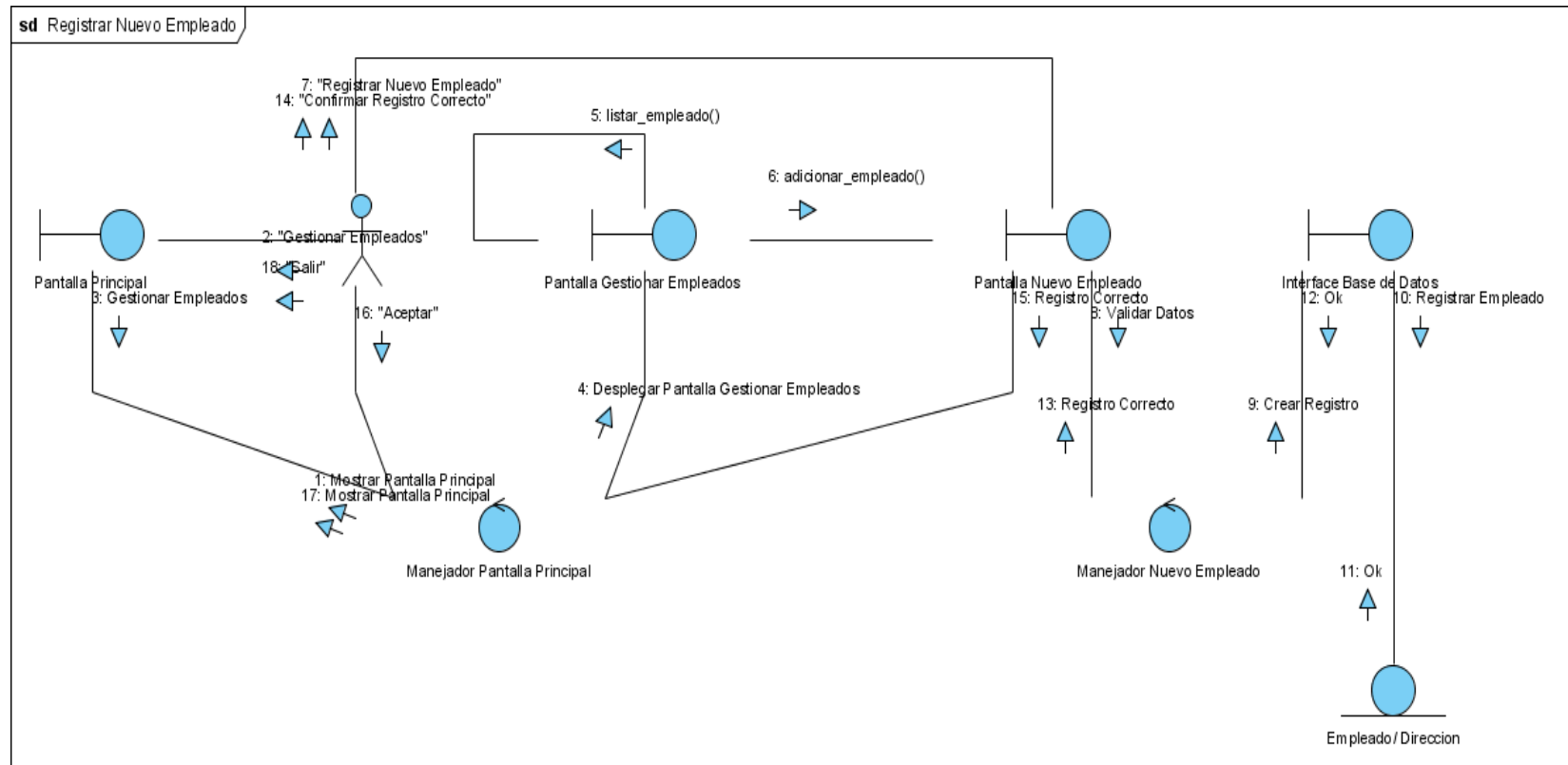


Figura 59 Diagrama Registrar Nuevo Empleado

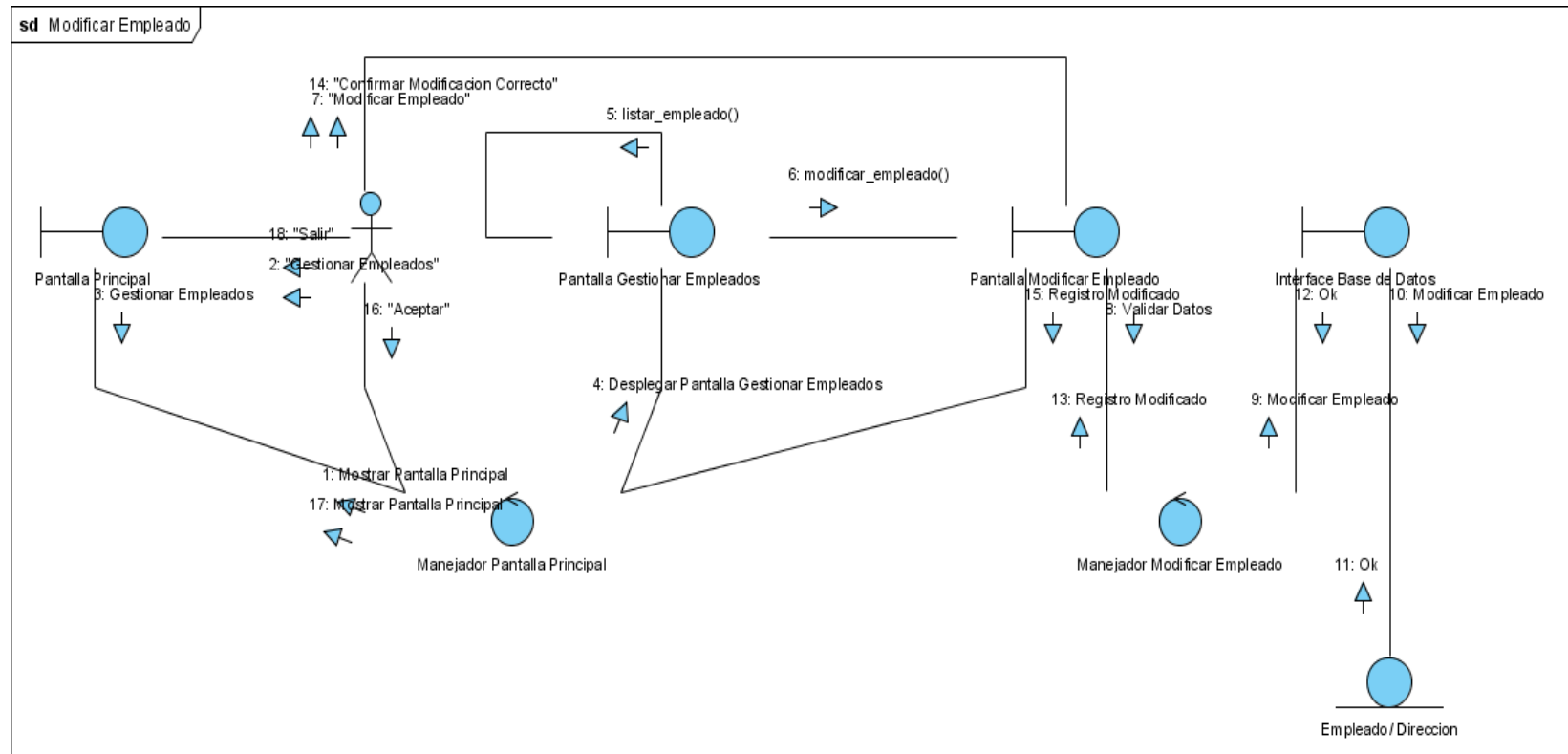


Figura 60 Diagrama Modificar Empleado

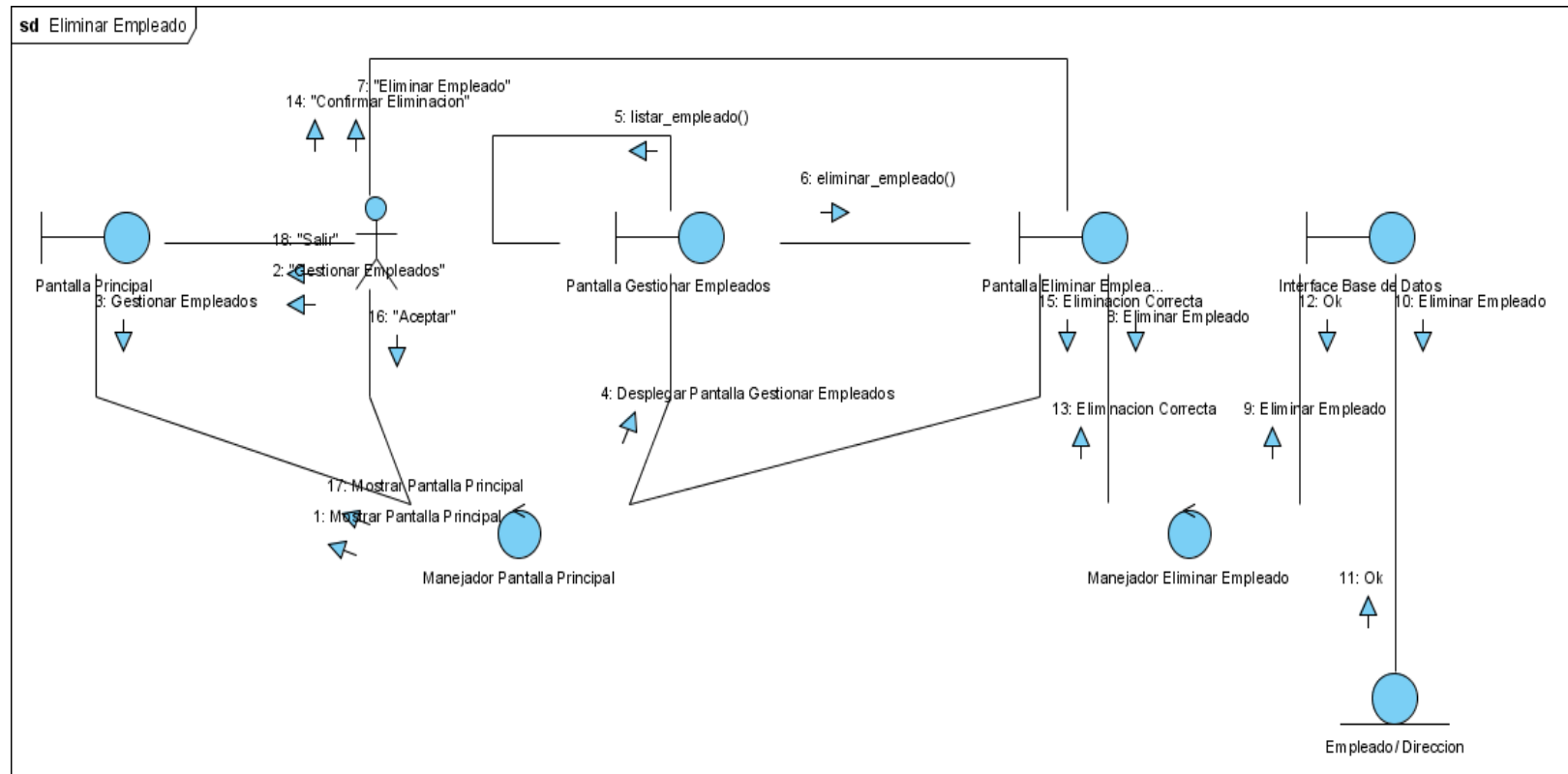


Figura 61 Diagrama Eliminar Empleado

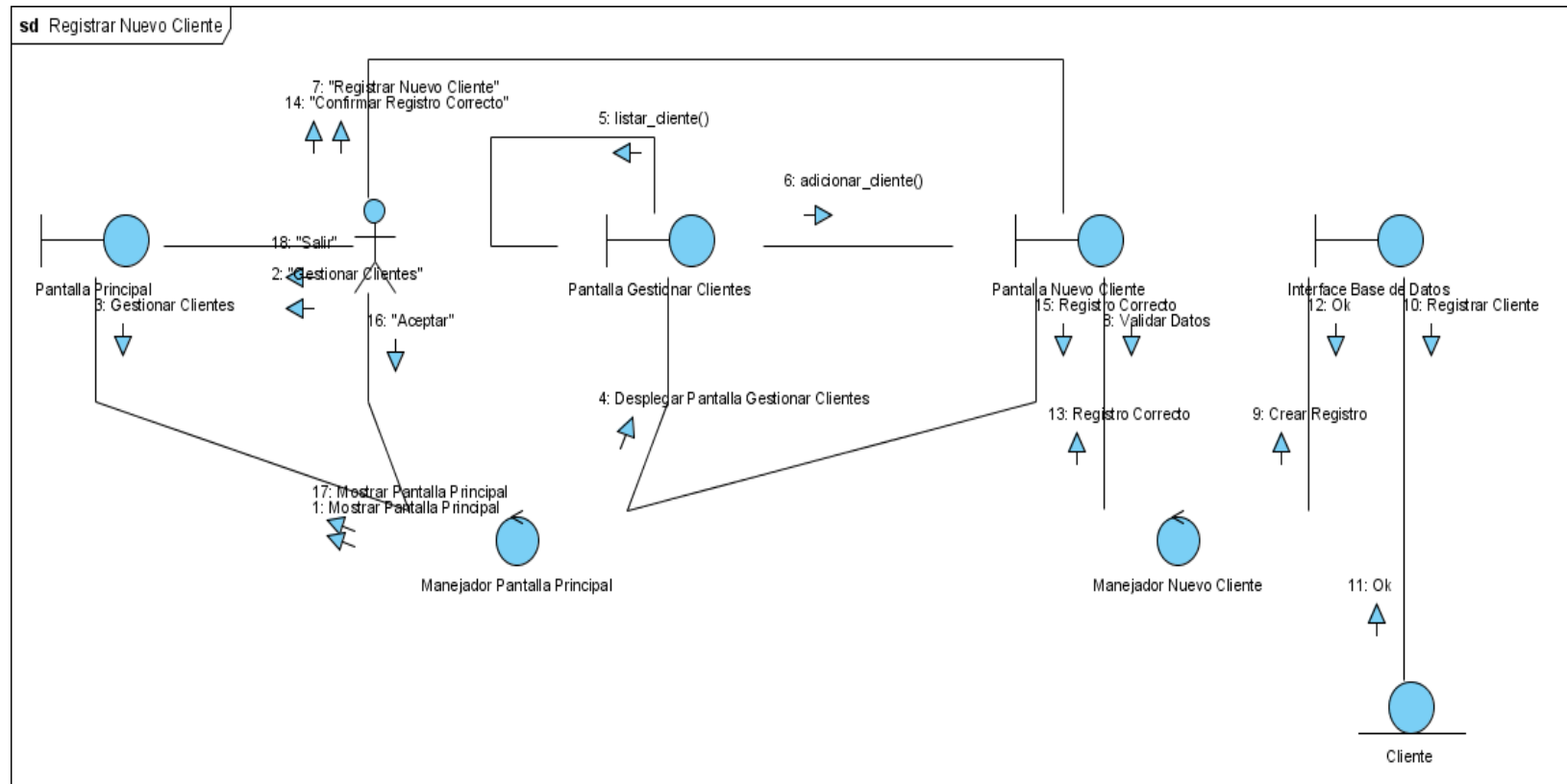


Figura 62 Diagrama Registrar Nuevo Cliente

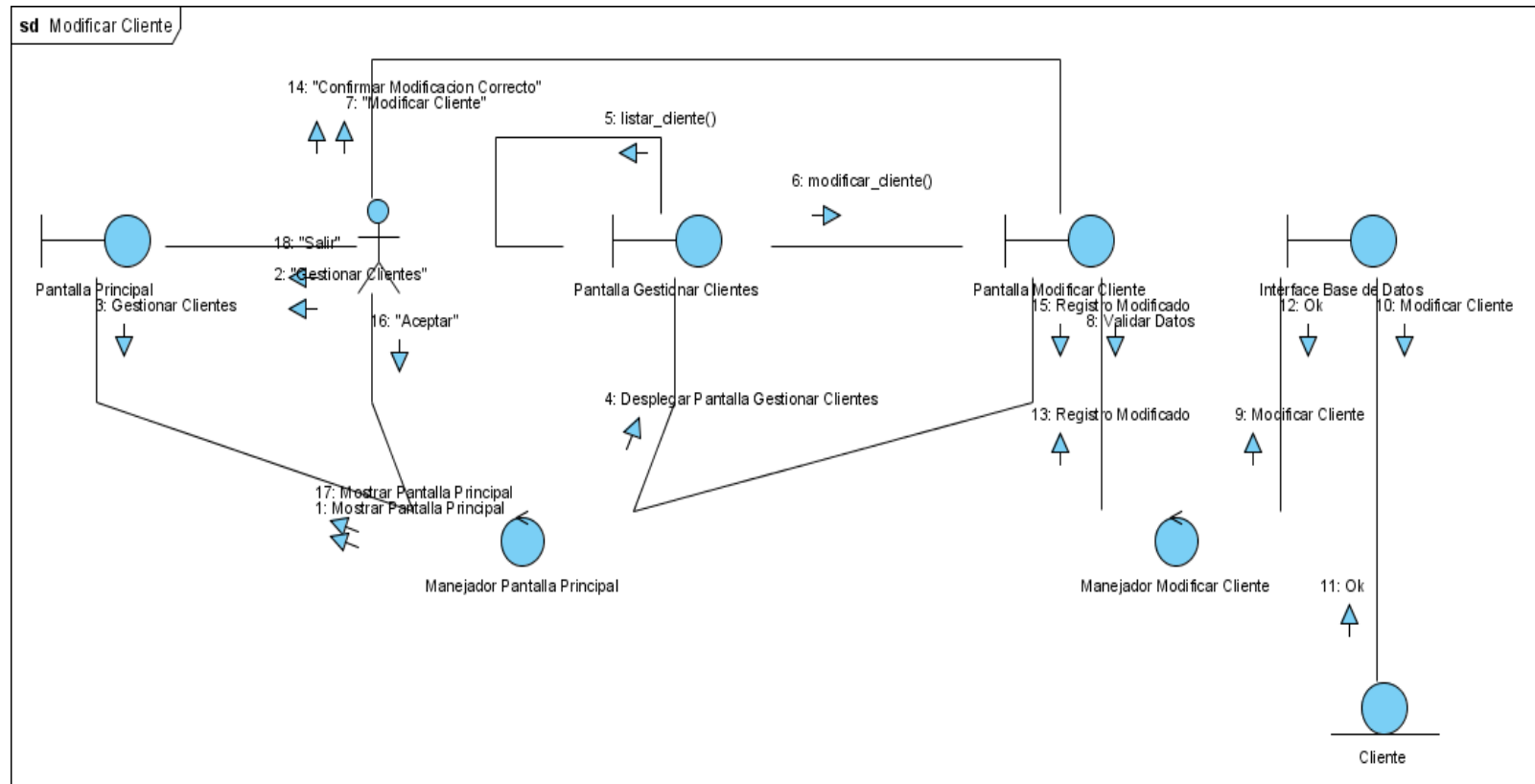


Figura 63 Diagrama Modificar Cliente

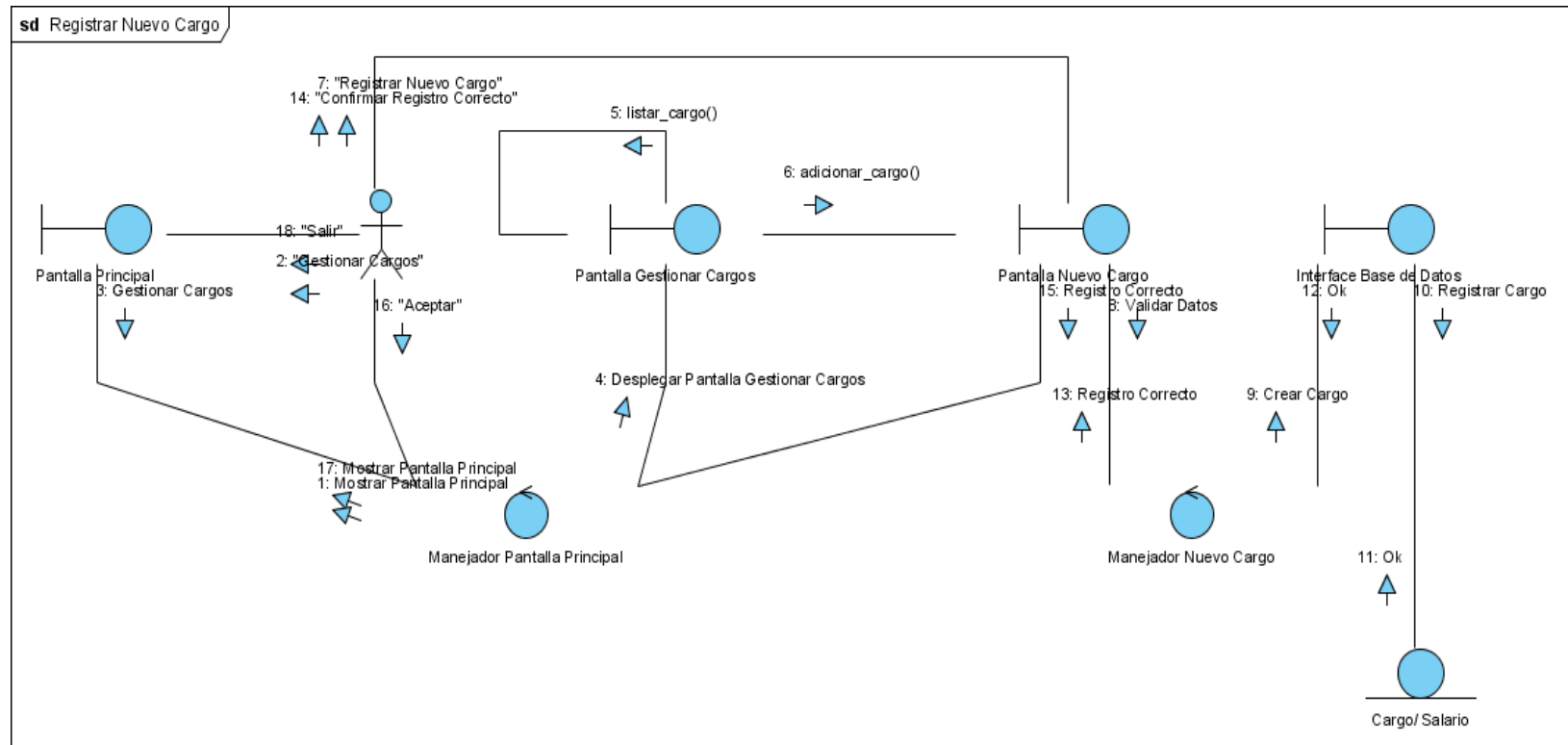


Figura 64 Diagrama Registrar Nuevo Cargo

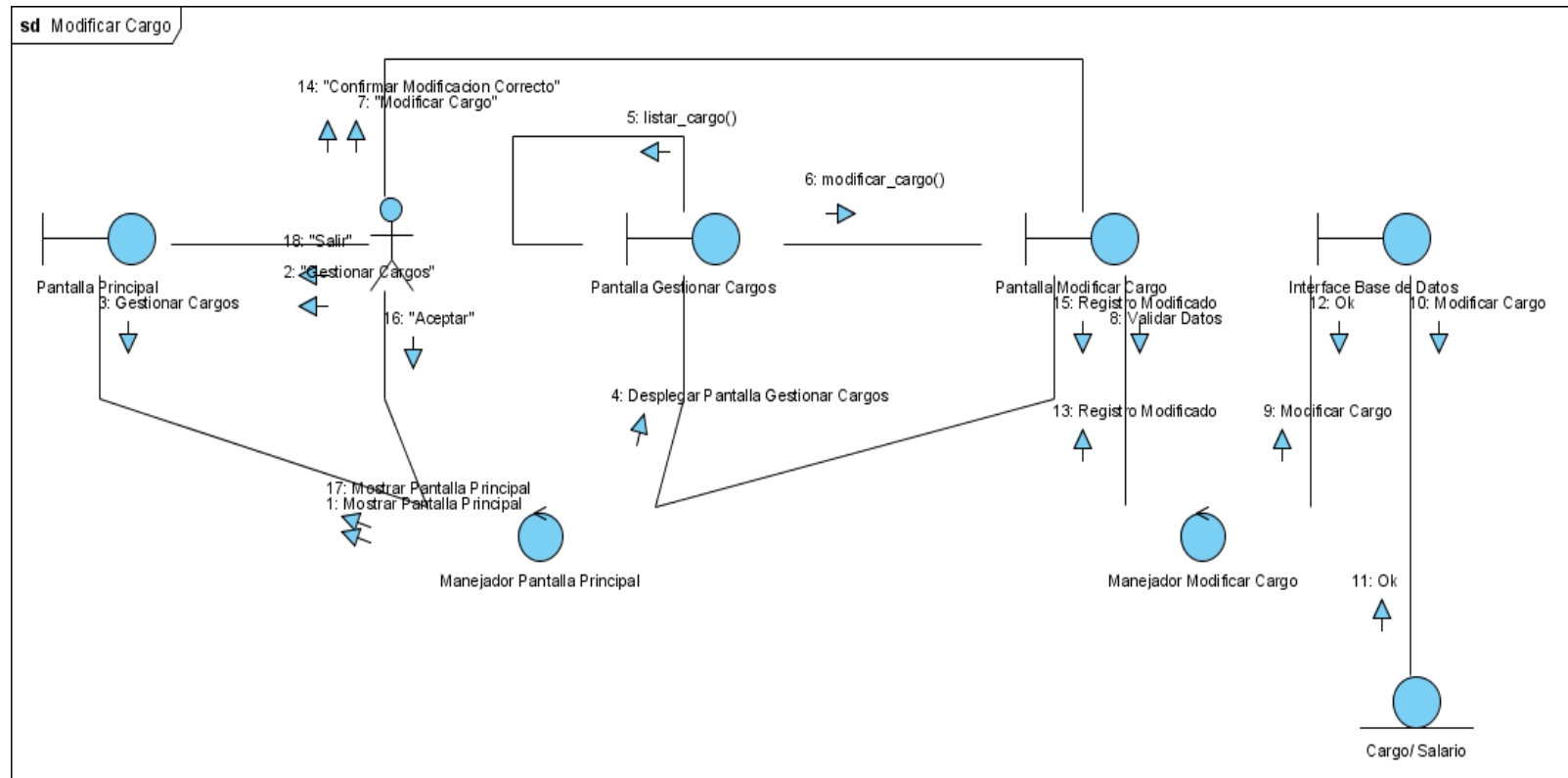


Figura 65 Diagrama Modificar Cargo

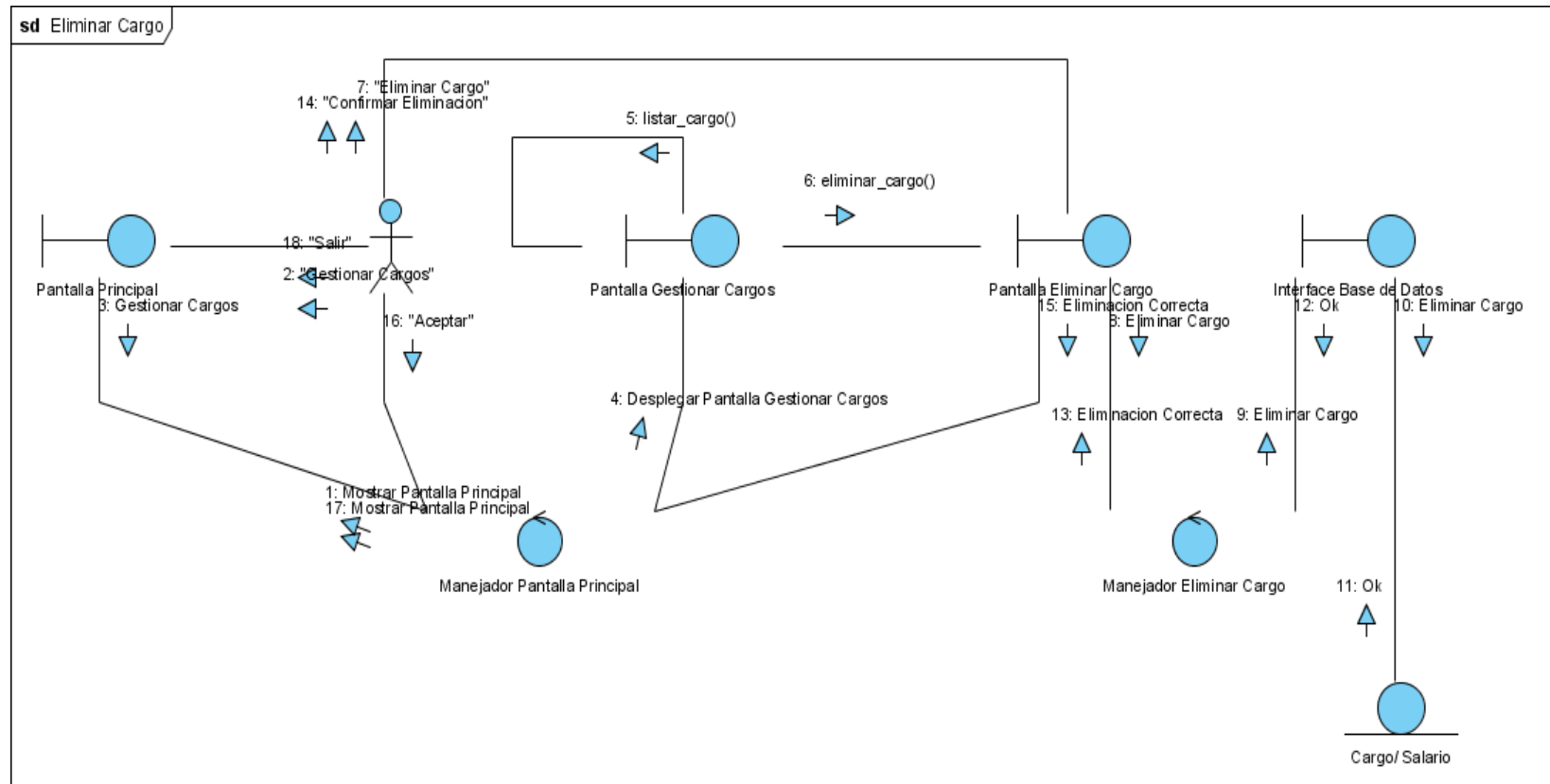


Figura 66 Diagrama Eliminar Cargo

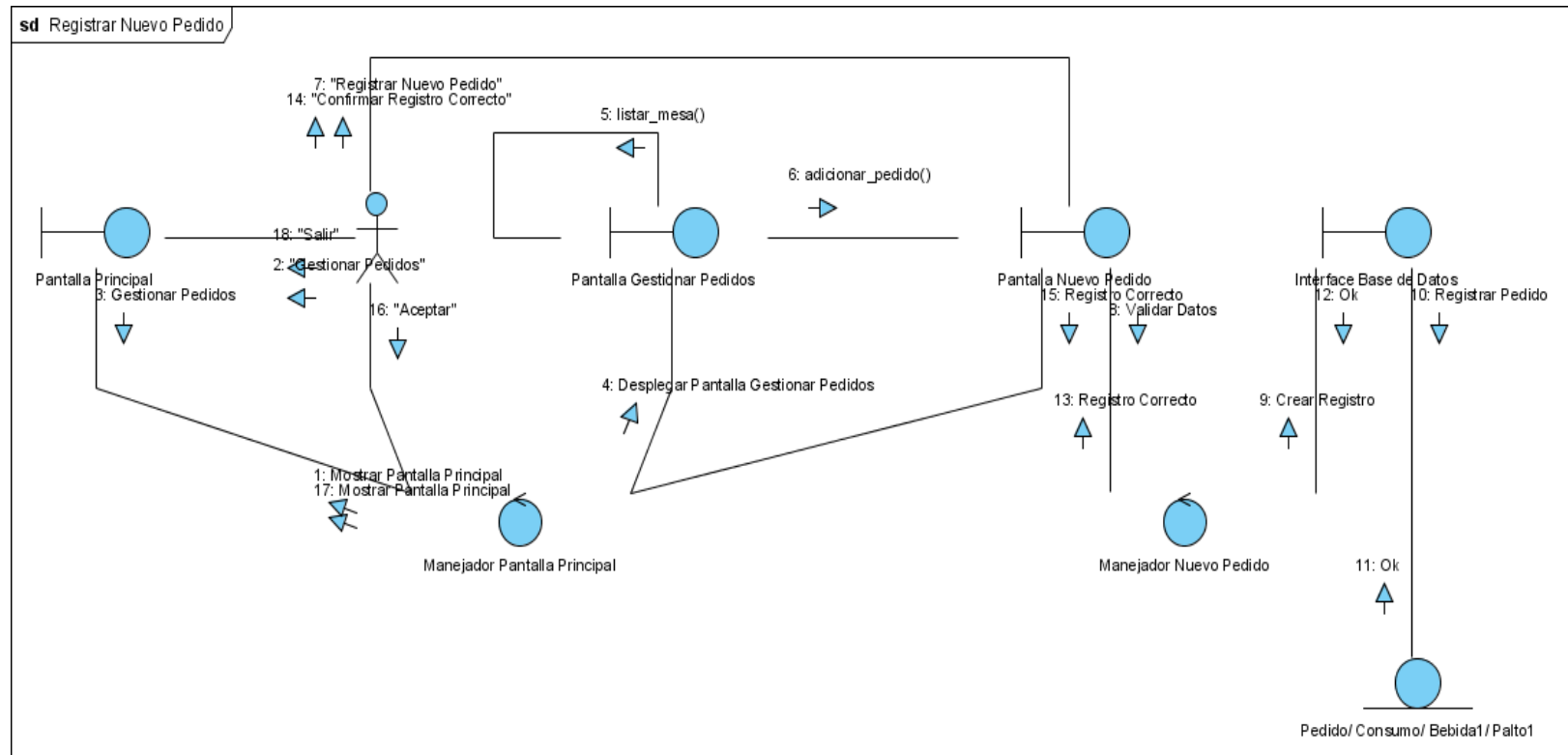


Figura 67 Diagrama Registrar Nuevo Pedido

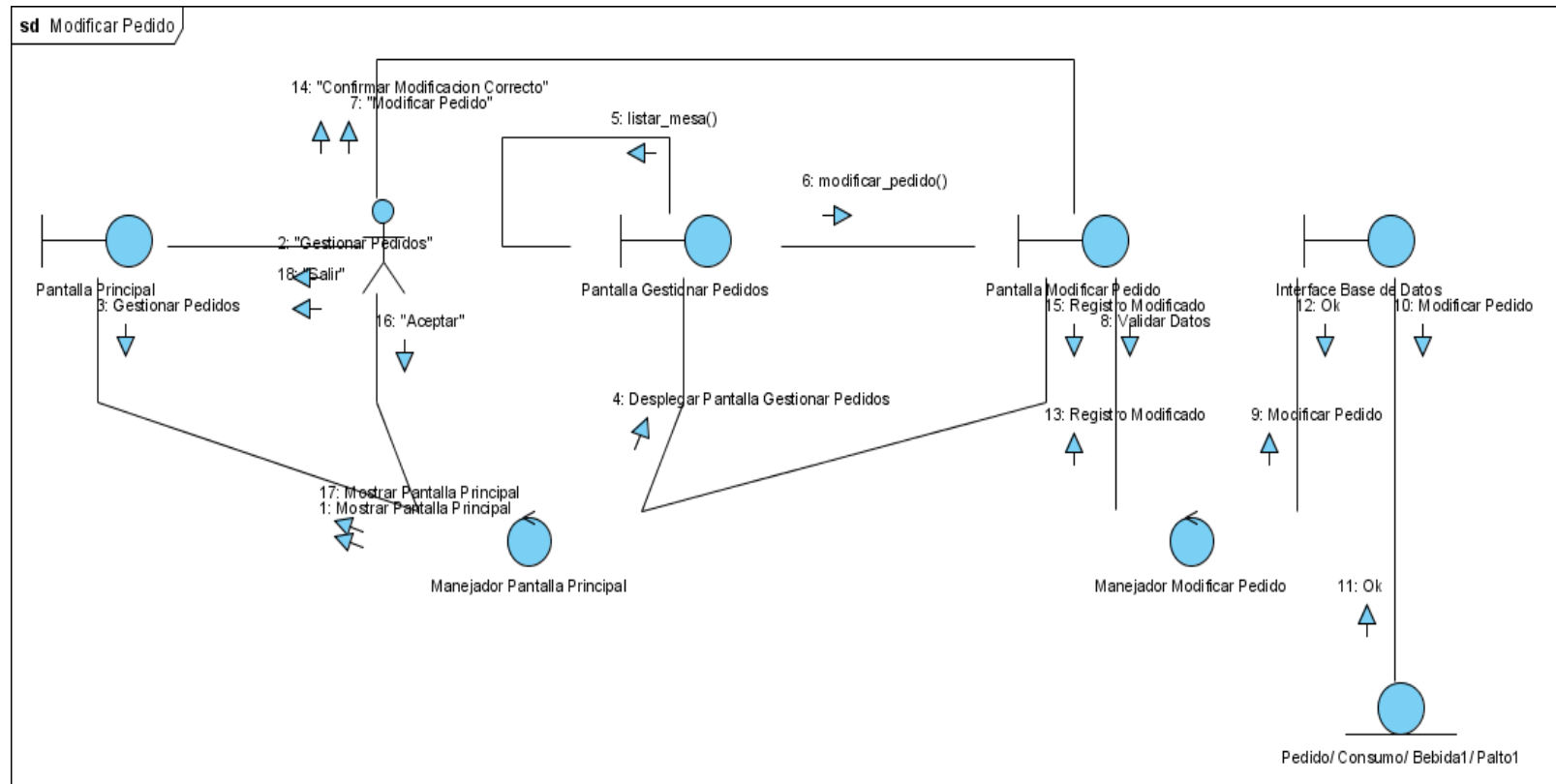


Figura 68 Diagrama Modificar Pedido

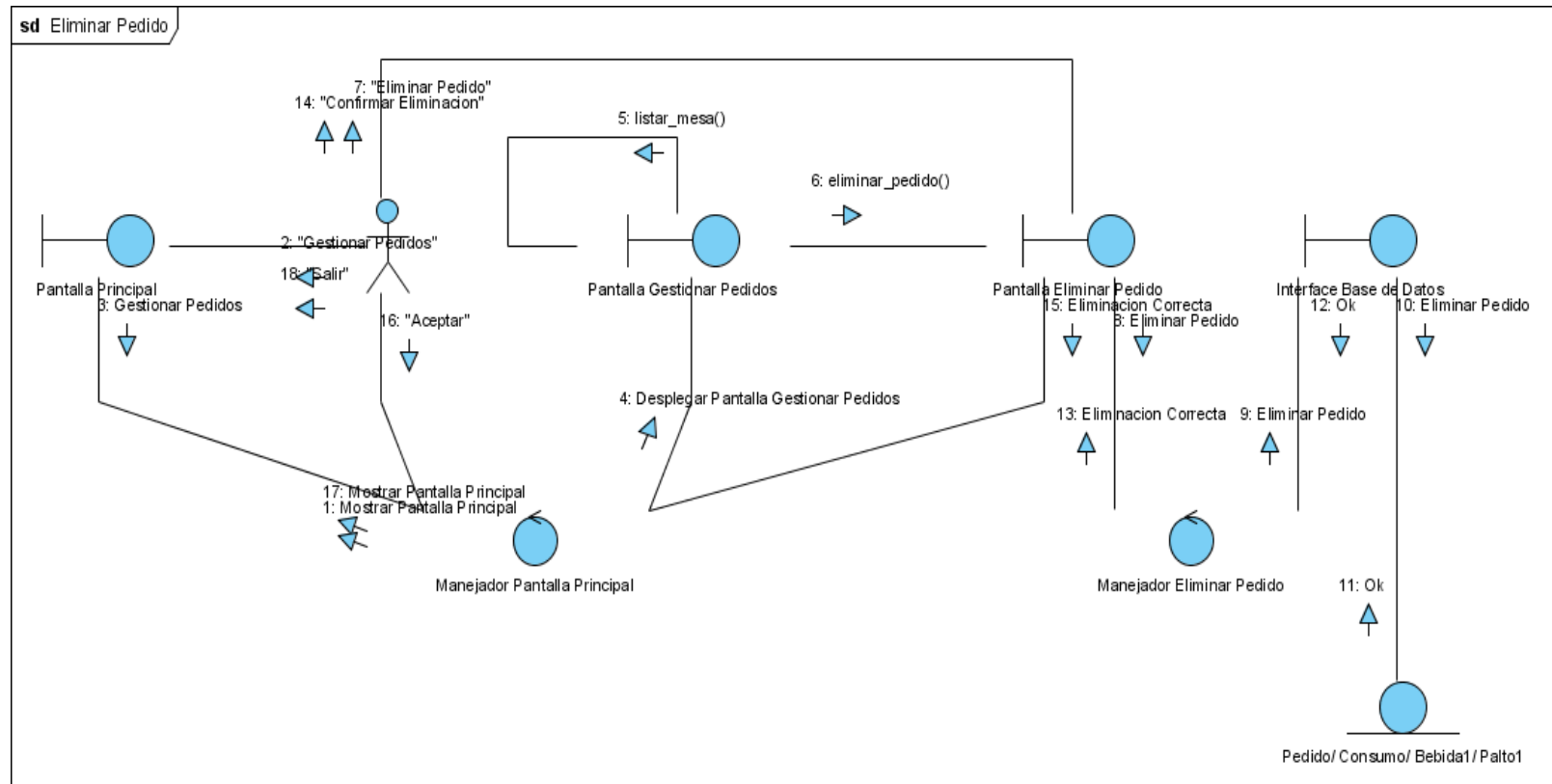


Figura 69 Diagrama Eliminar Pedido

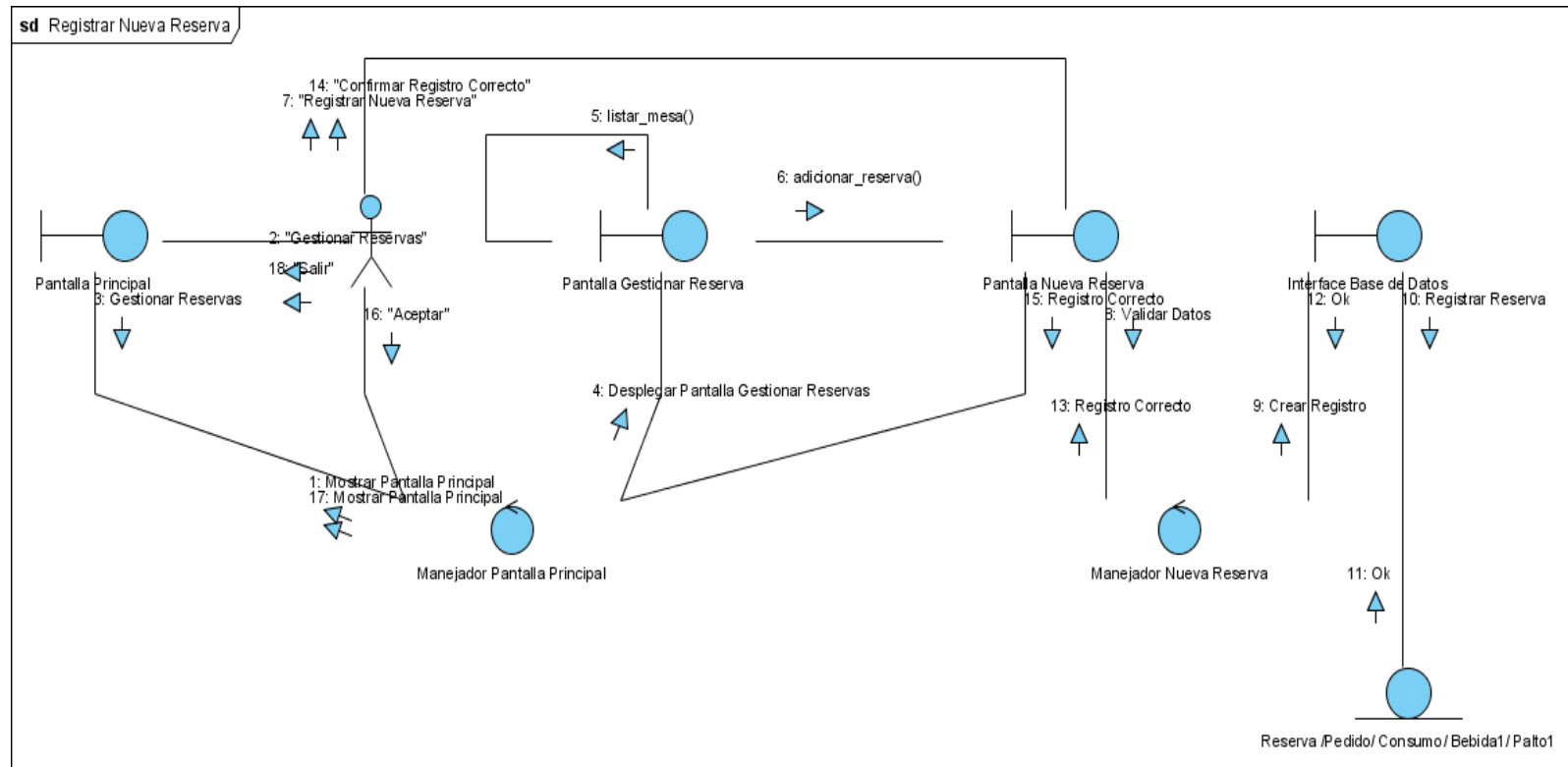


Figura 70 Diagrama Registrar Nueva Reserva

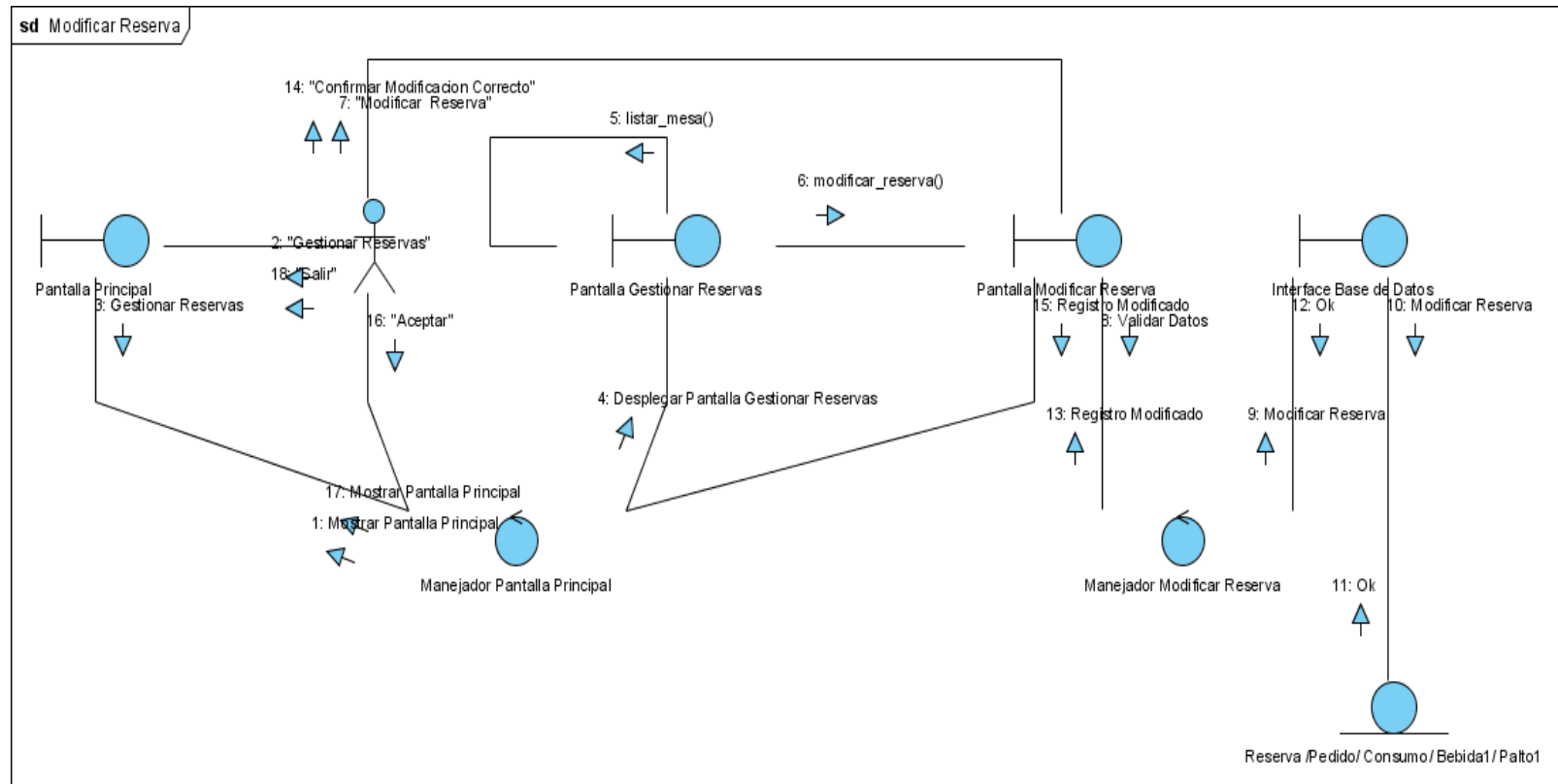


Figura 71 Diagrama Modificar Reserva

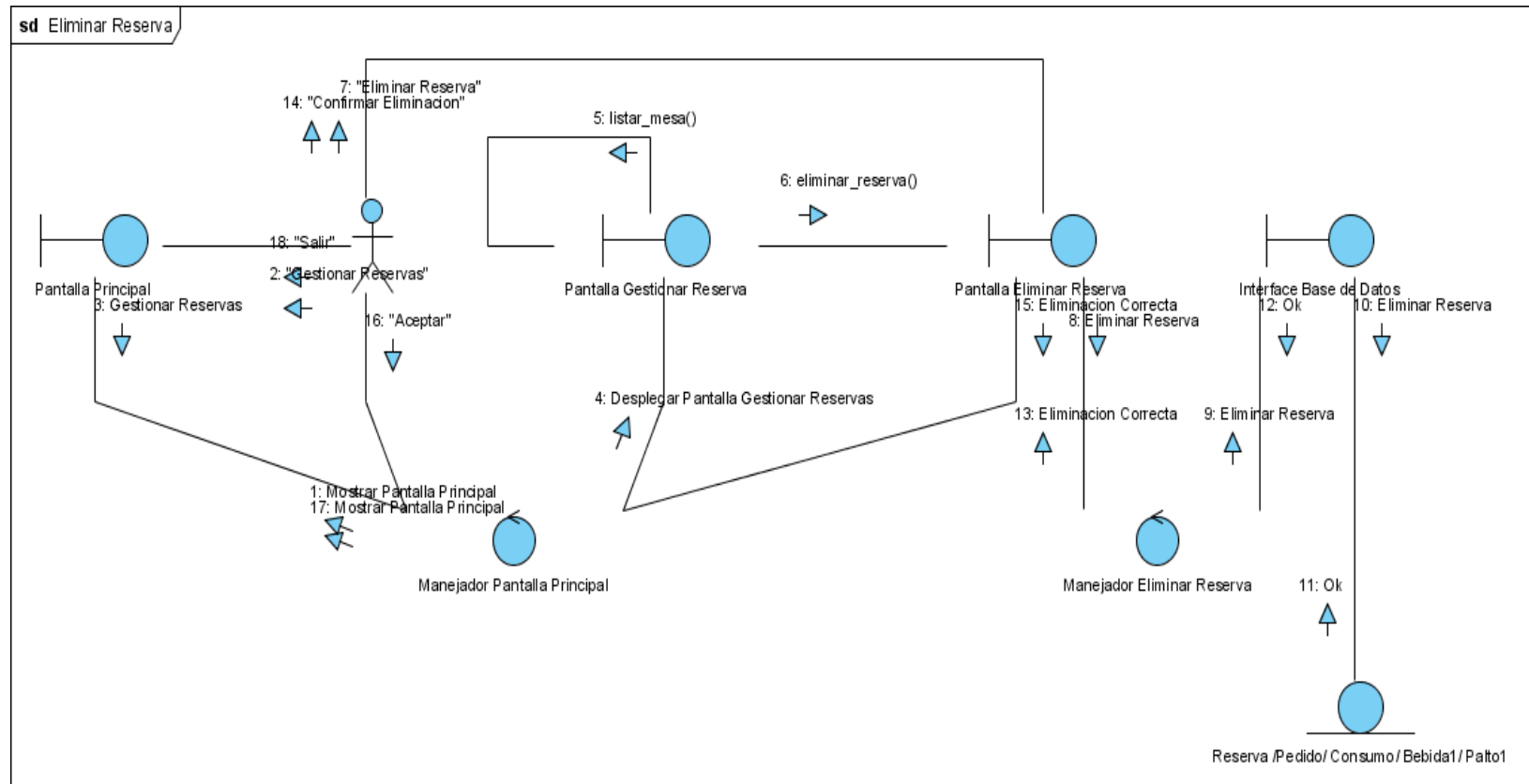


Figura 72 Diagrama Eliminar Reserva

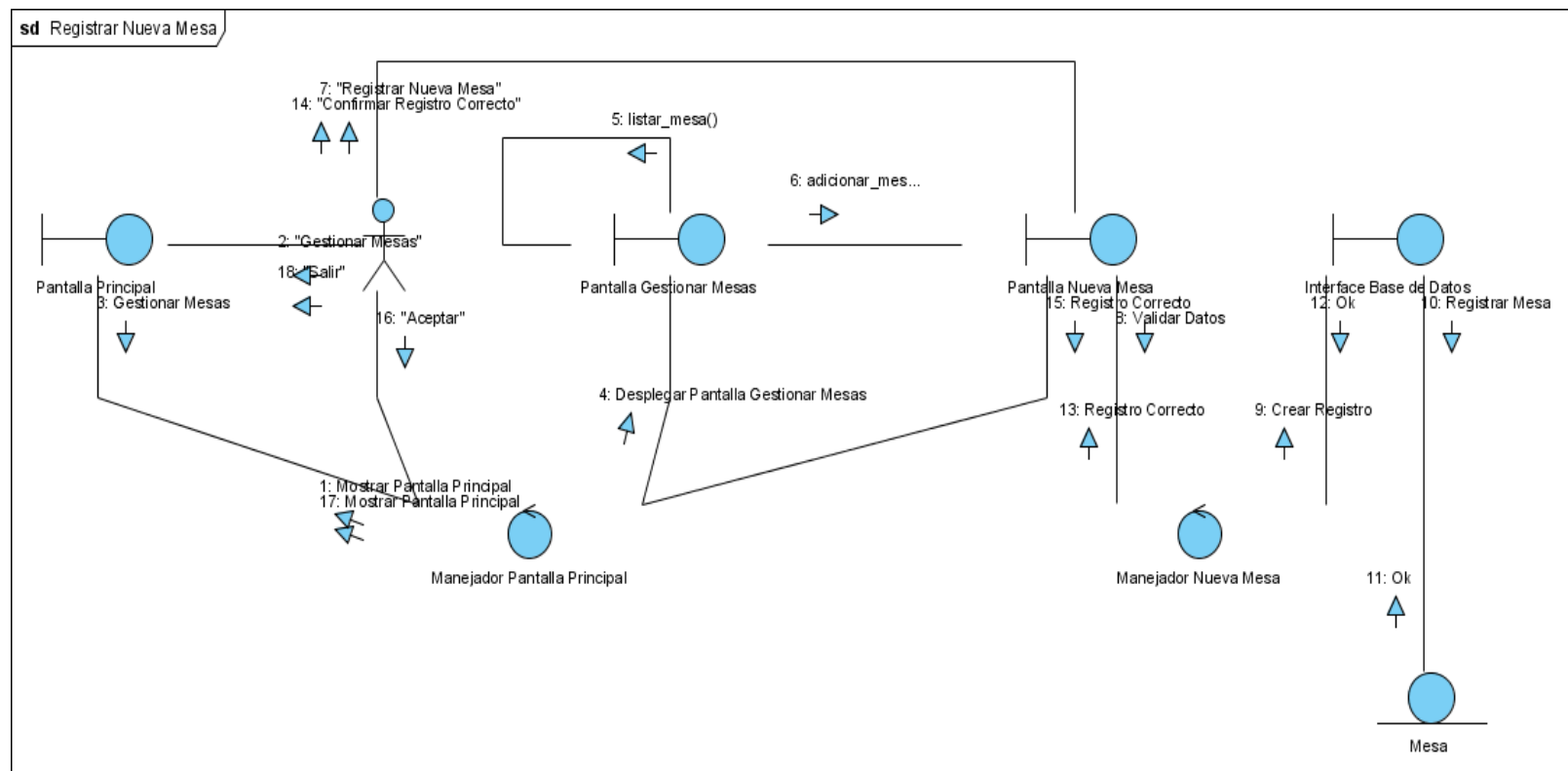


Figura 73 Diagrama Registrar Nueva Mesa

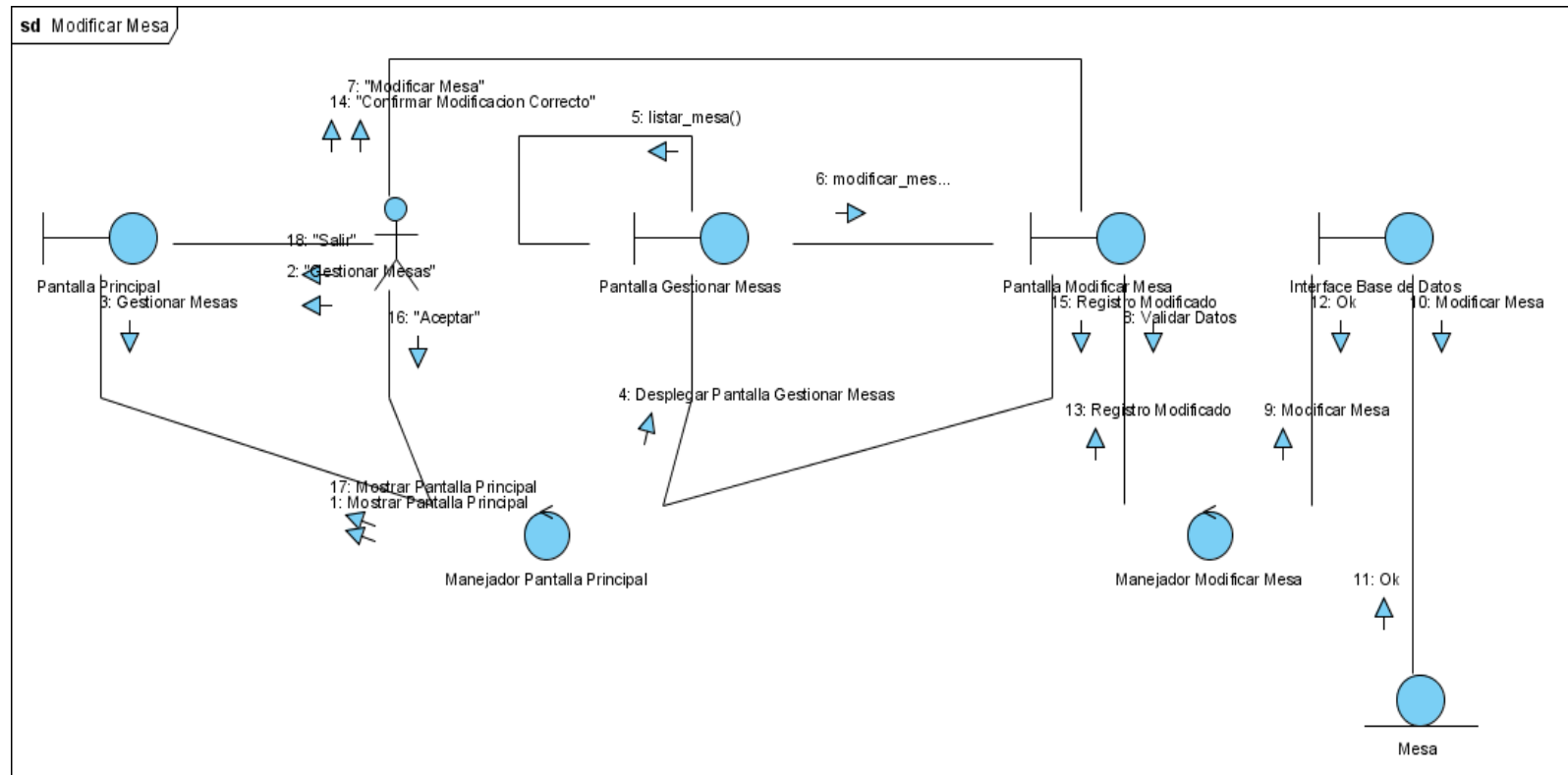


Figura 74 Diagrama Modificar Mesa

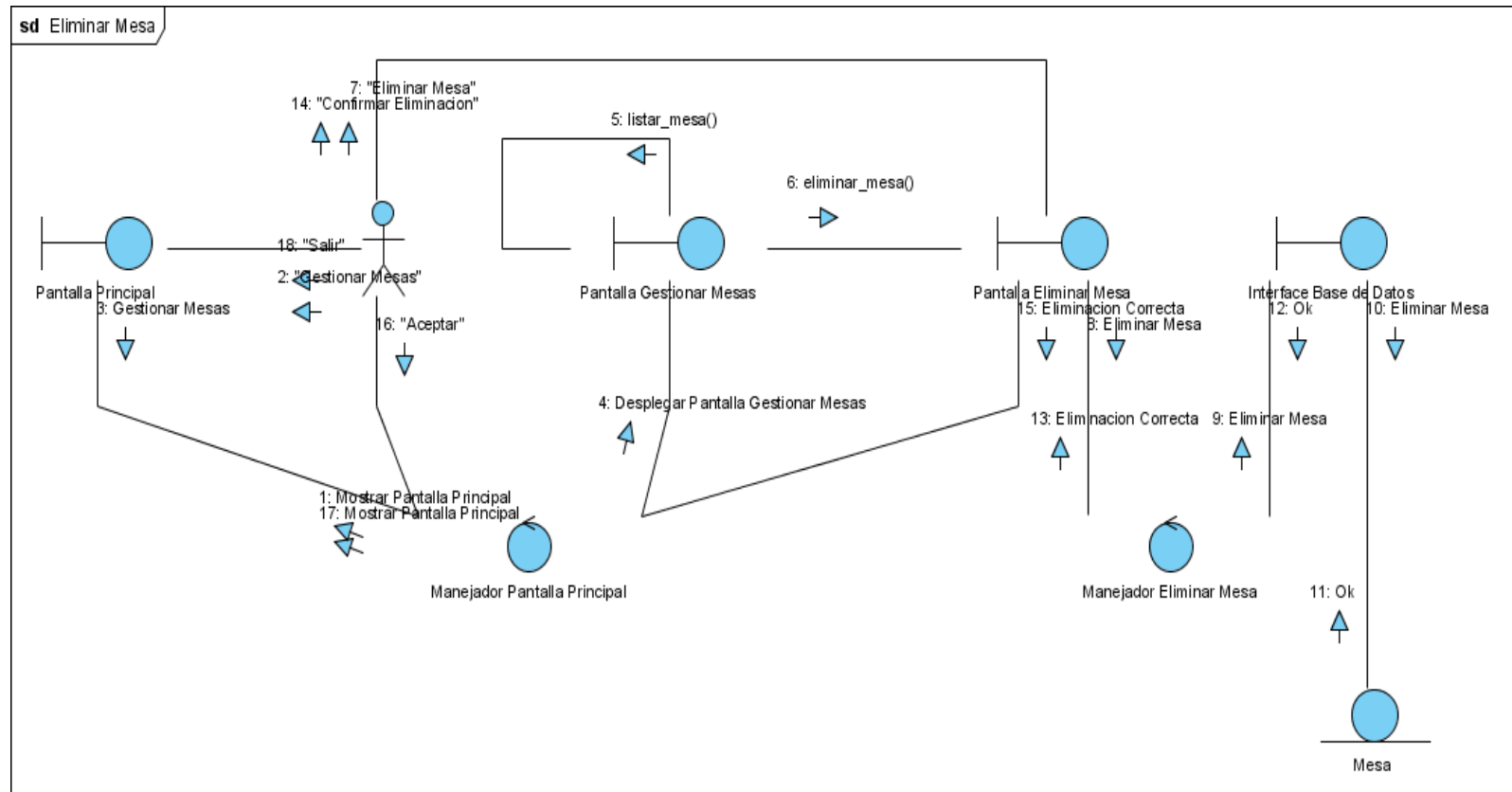


Figura 75 Diagrama Eliminar Mesa

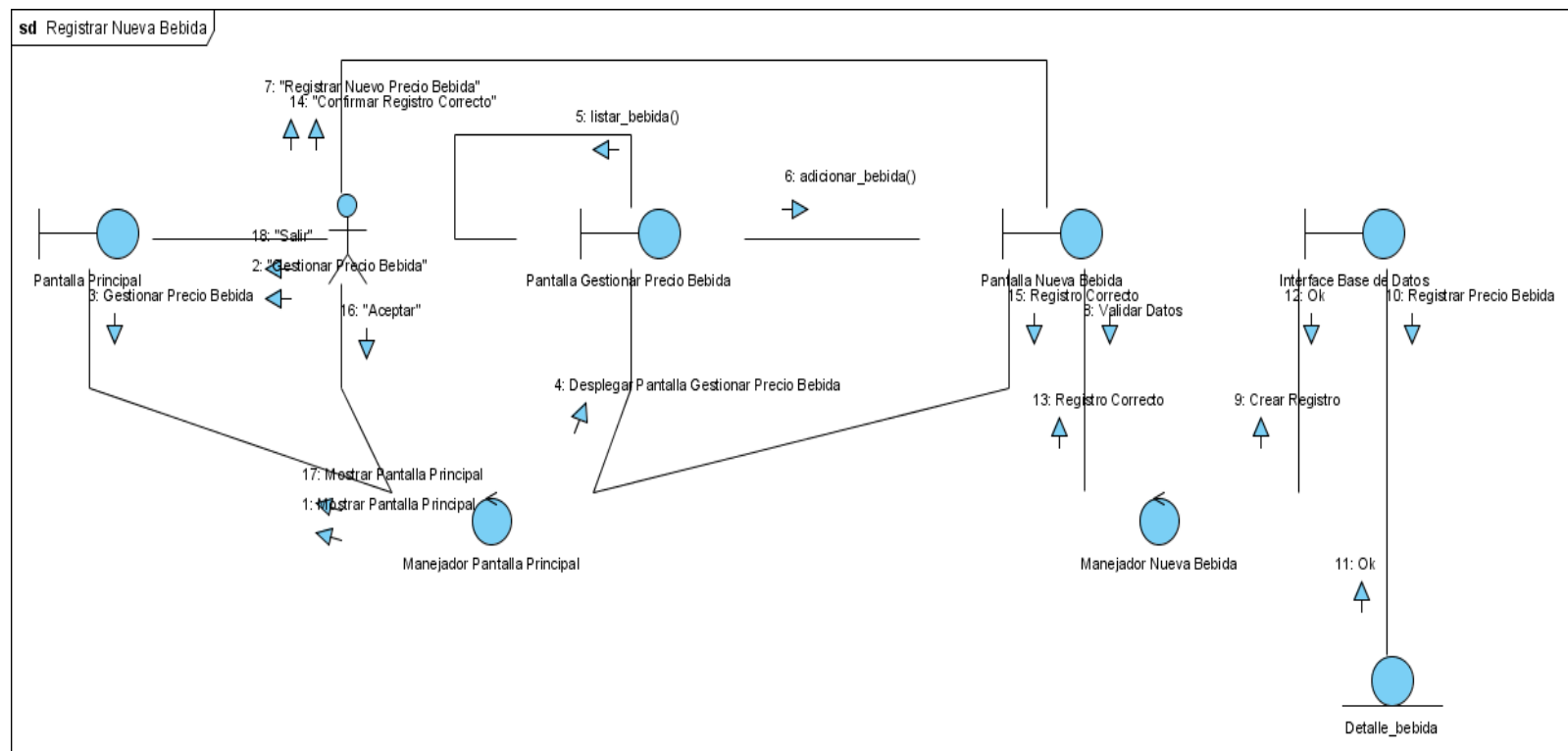


Figura 76 Diagrama Registrar Nueva Bebida

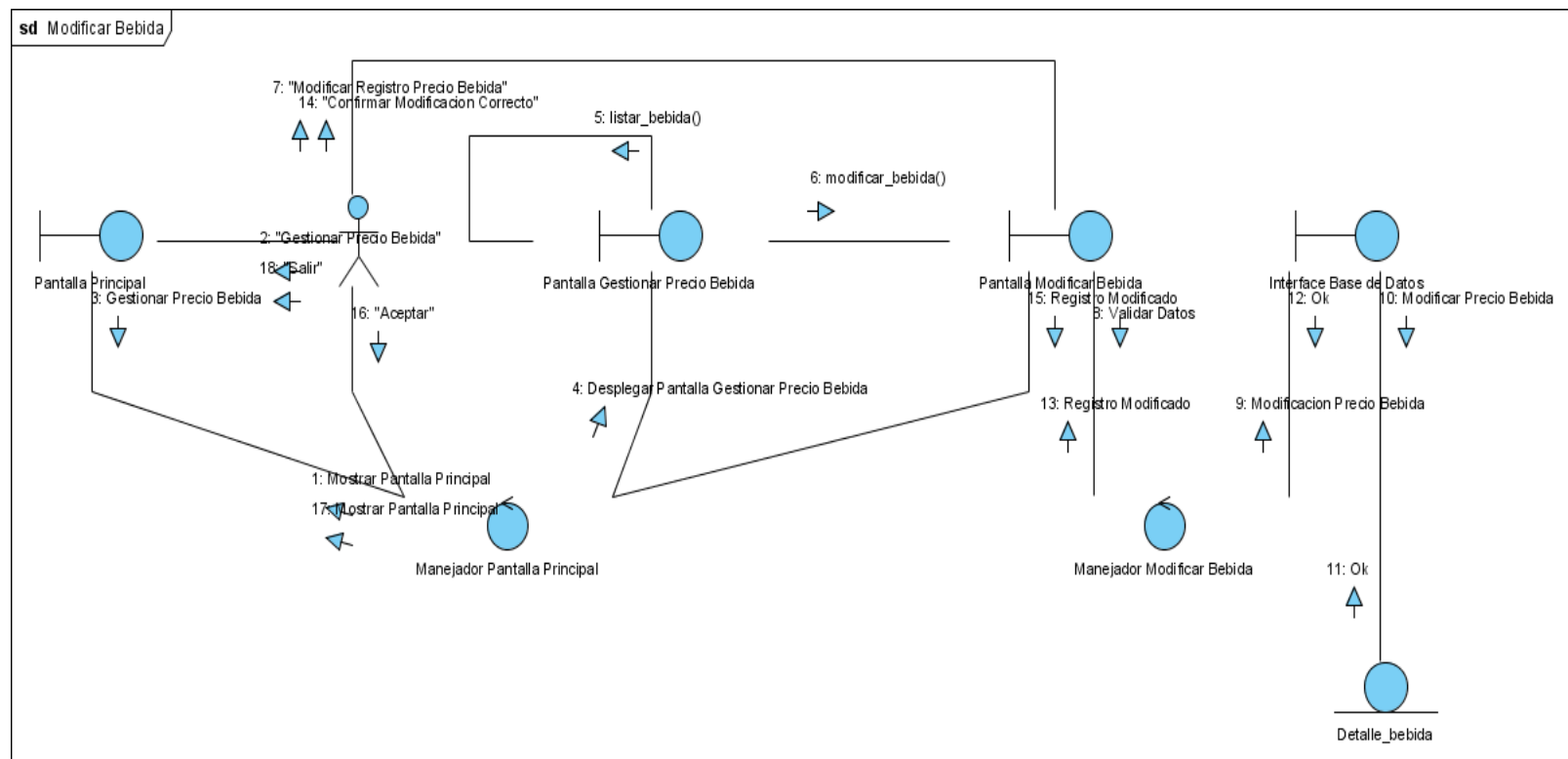


Figura 77 Diagrama Modificar Bebida

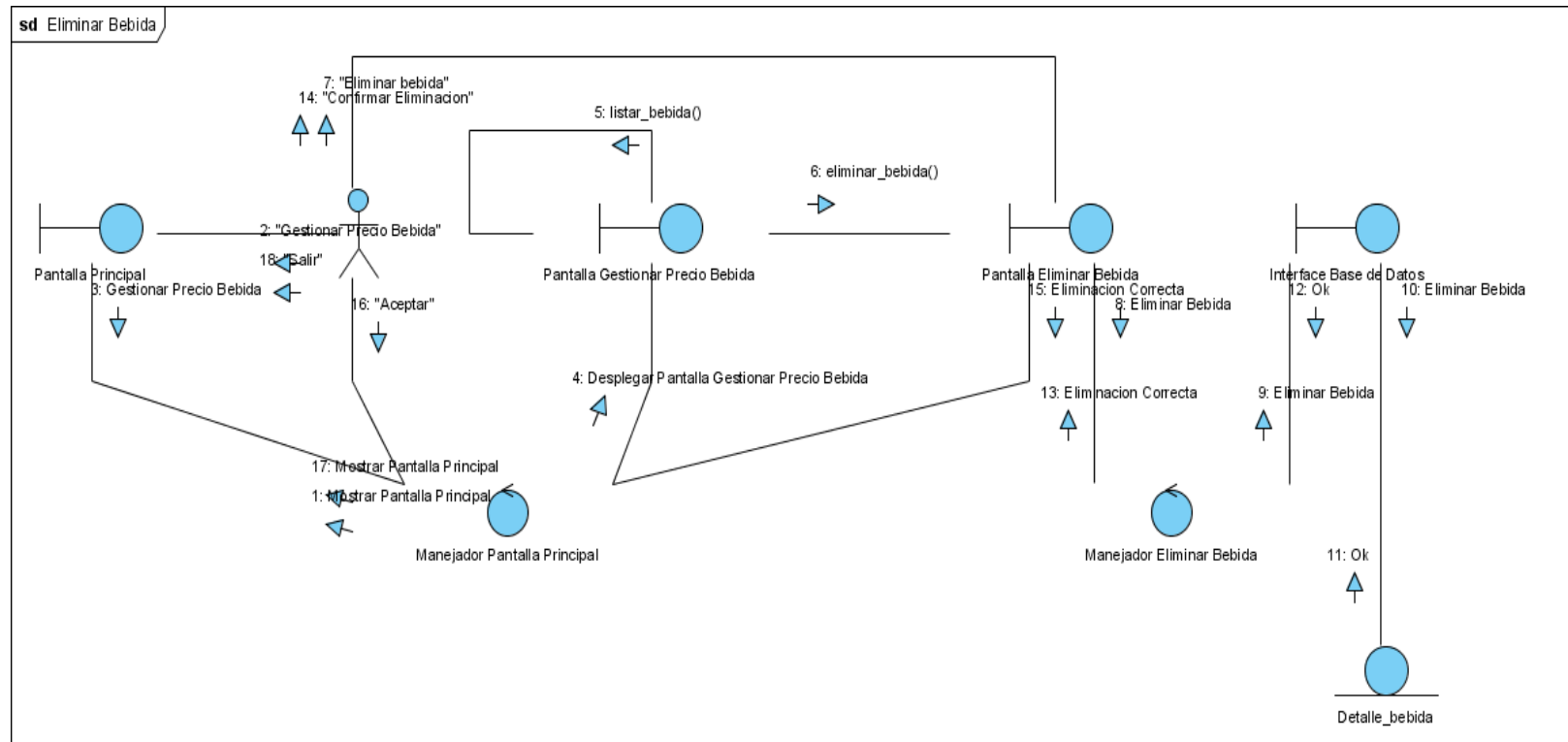


Figura 78 Diagrama Eliminar Bebida

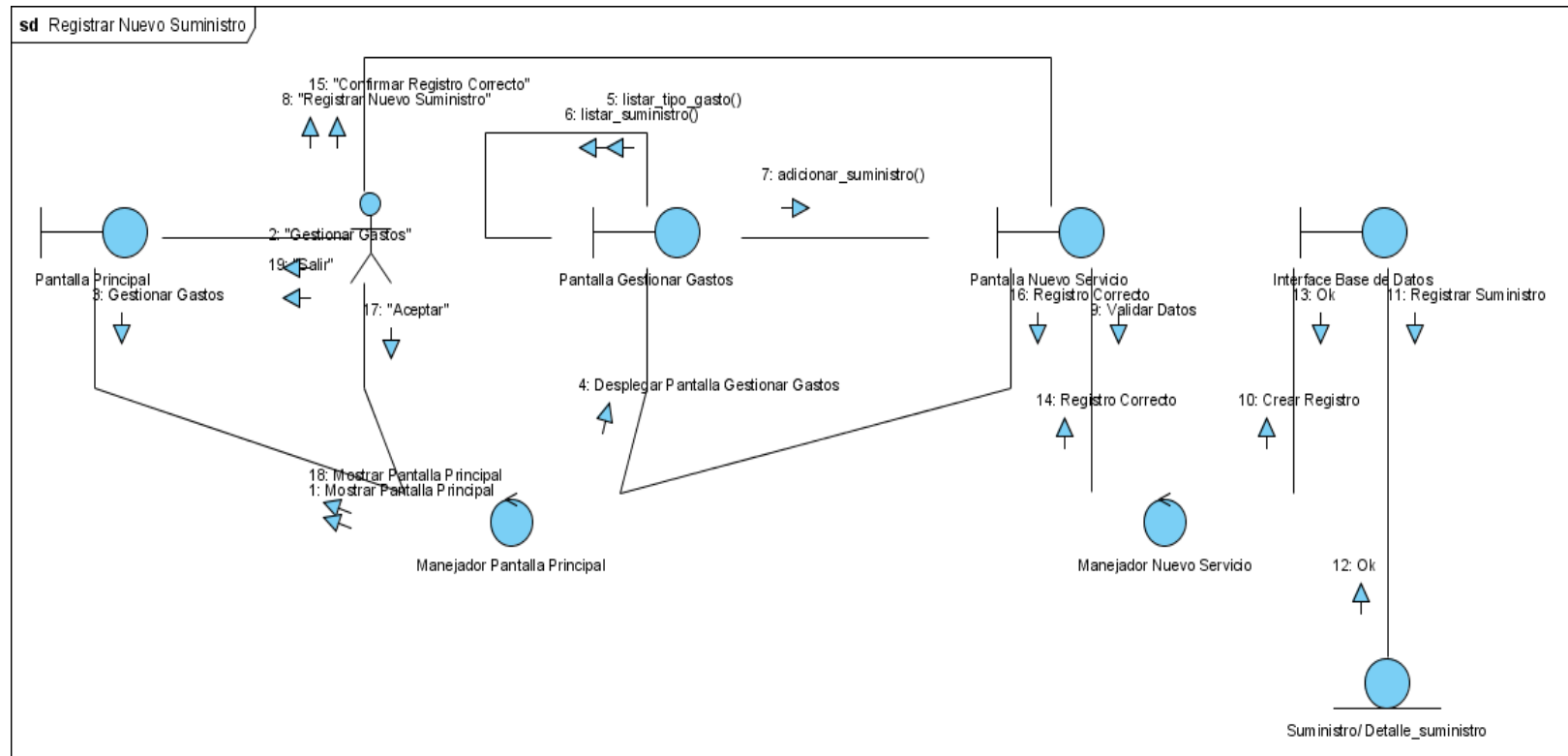


Figura 79 Diagrama Registrar Nuevo Suministro

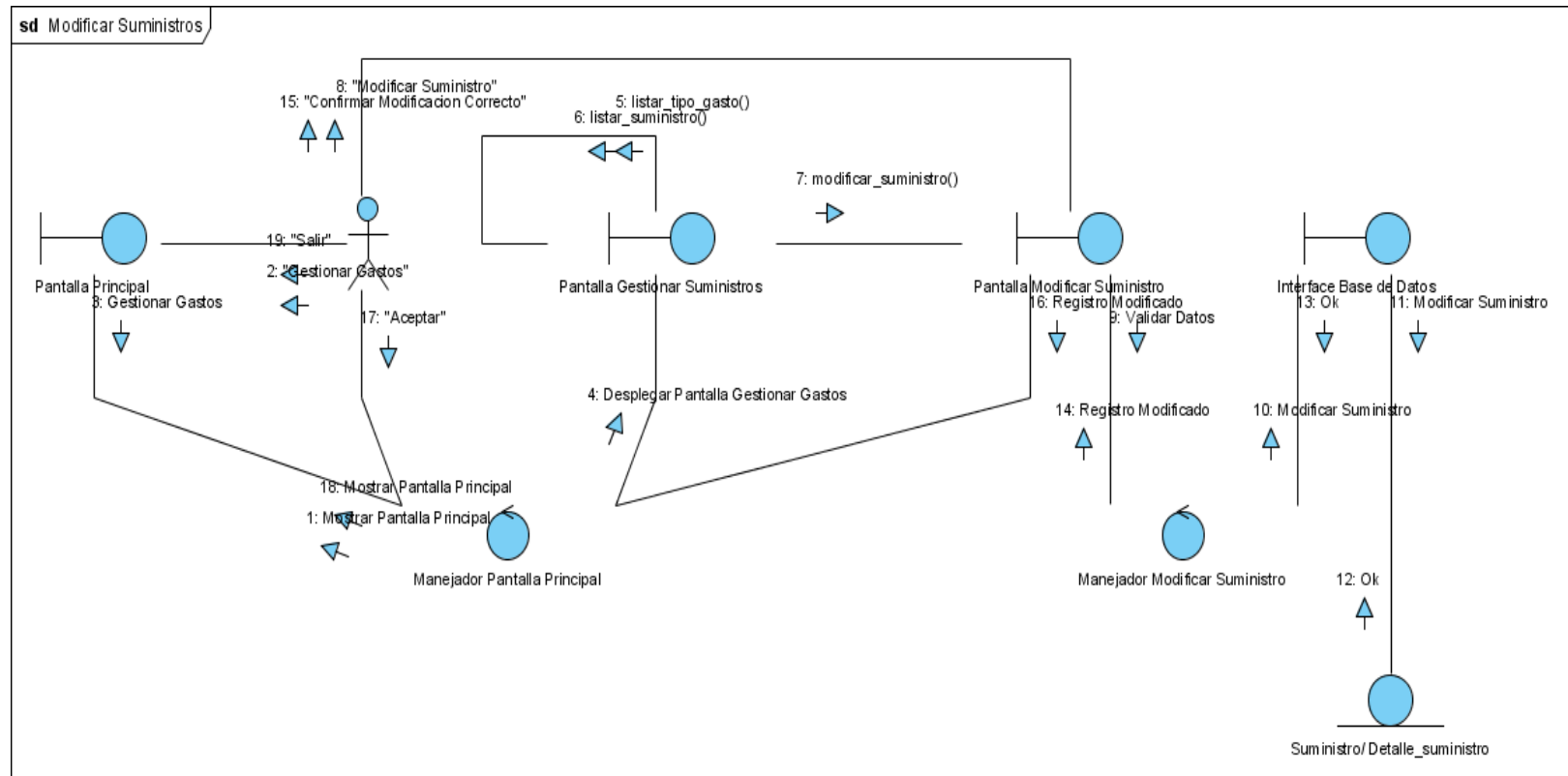


Figura 80 Diagrama Modificar Suministro

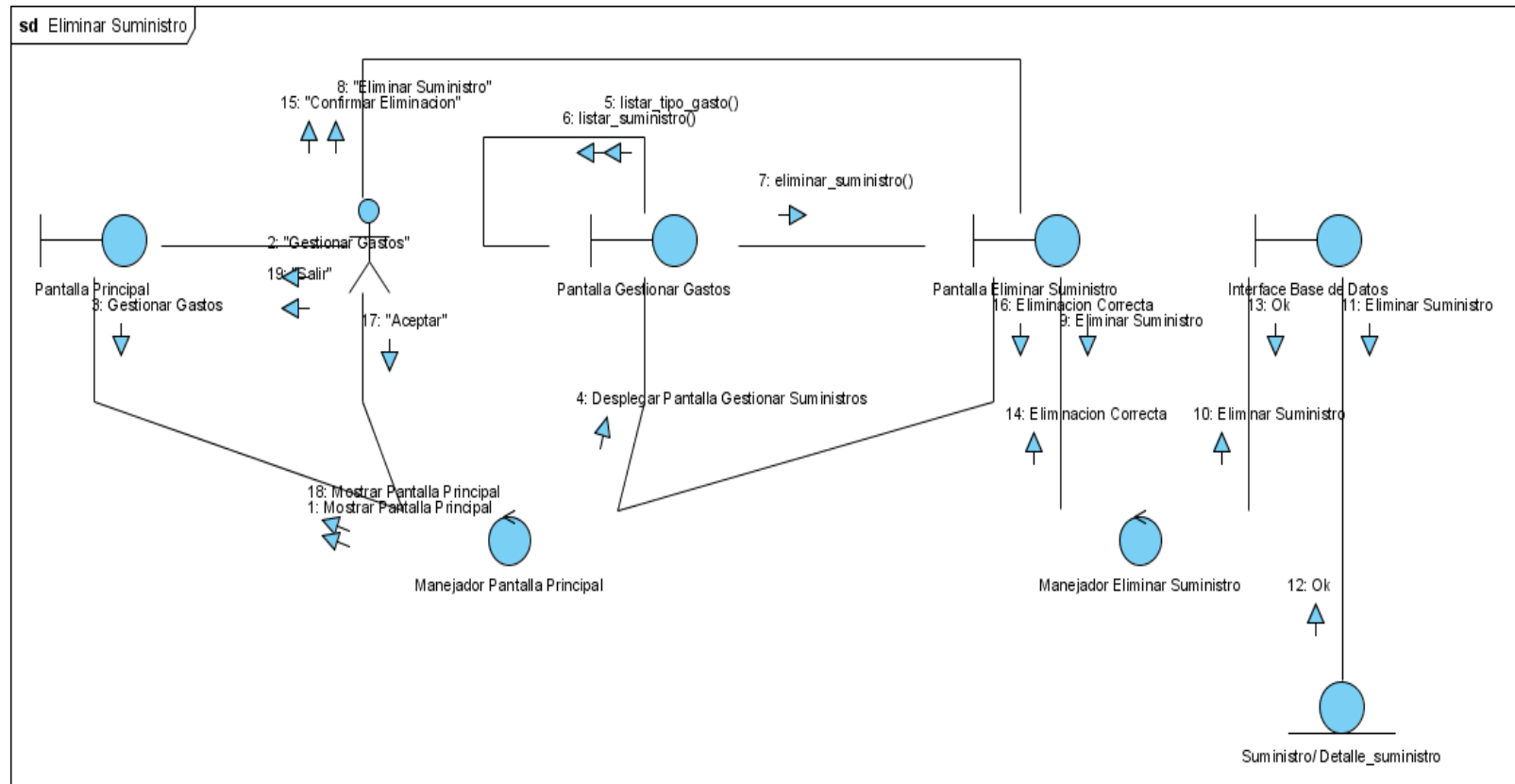


Figura 81 Diagrama Eliminar Suministro

II.1.5.1.4. Diagramas de Componentes

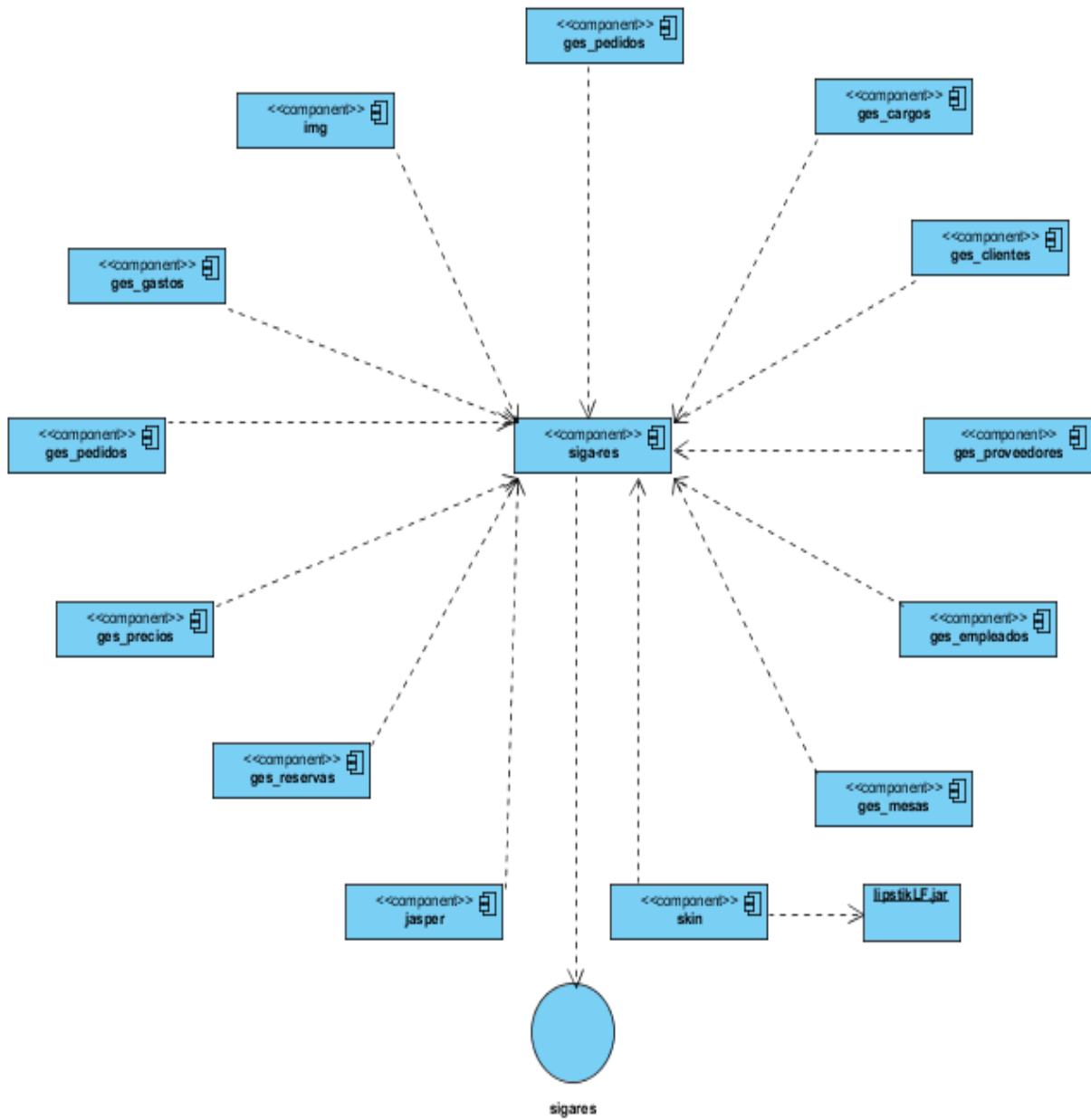


Figura 82 Diagrama de Componentes

II.1.5.1.5. Diagrama de Despliegue

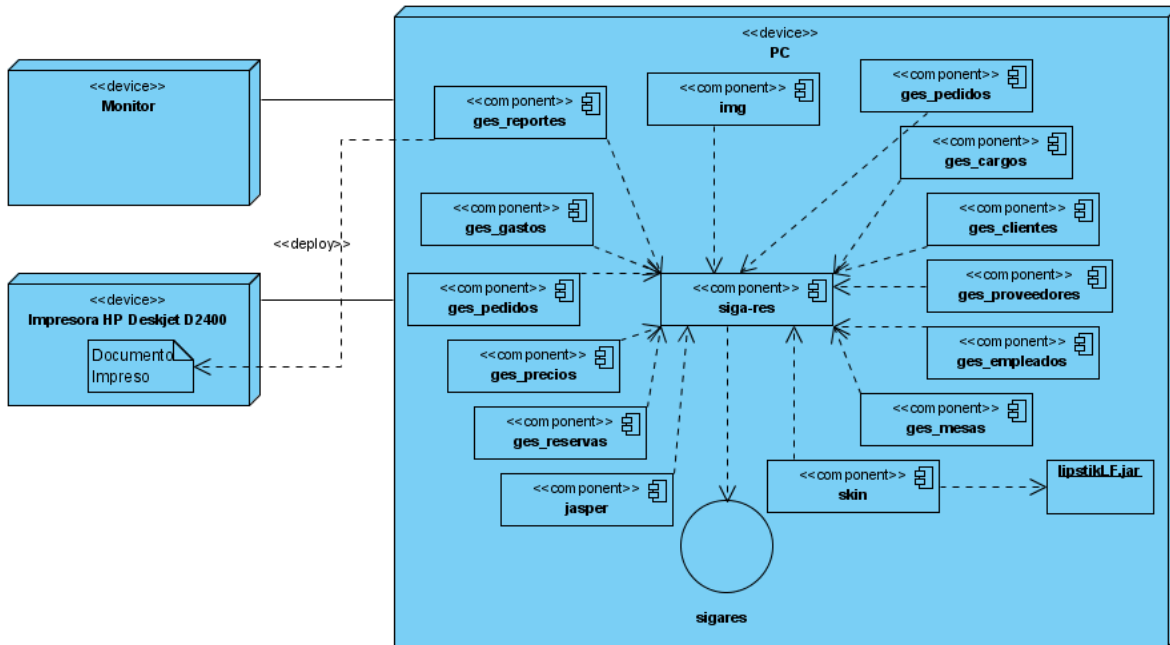


Figura 83 Diagrama de Componentes

II.1.5.1.6. Base de Datos

II.1.5.1.6.1. Diagrama de Clases

II.1.5.1.6.2. Conversión a Tablas Relacionales

| Tabla | Bebidas1 |
|-----------------|--|
| Clave Primaria | id_bebida |
| Atributos | botellas |
| Claves Foráneas | detalle_bebida id_det_bebida consumoid_consumo |

Tabla 1. Bebidas1

| Tabla | Cargo |
|----------------|--|
| Clave Primaria | id_cargo |
| Atributos | nom_cargo descripción tipo_permiso |

Tabla 5. Cargo

| Tabla | Cliente |
|----------------|--|
| Clave Primaria | id_cliente |
| Atributos | ci_cliente nombre_cli ap_cli am_cli sexo_cli |

Tabla 6. Cliente

| Tabla | Consumo |
|-----------------|----------------------------|
| Clave Primaria | id_consumo |
| Atributos | estado_consumo precio_t |
| Claves Foráneas | pedidoid_pedido |

Tabla 2. Consumo

| Tabla | Detalle_bebida |
|----------------|---|
| Clave Primaria | id_det_bebida |
| Atributos | nom_producto característica_bebida cc_bebida precio_unit_b |

Tabla 3. Detalle_bebida

| Tabla | Detalle_plato |
|----------------|---|
| Clave Primaria | id_det_bebida |
| Atributos | nom_plato característica_plato porción_x_persona precio_unit_p |

Tabla 4. Detalle_plato

| Tabla | Detalle_suministro |
|-----------------|---|
| Clave Primaria | id_det_sum |
| Atributos | nom_sum características_sum precio_unit_sum |
| Claves Foráneas | suministroid_suministro |

Tabla 5. Detalle_suministro

| Tabla | Direccion |
|----------------|---------------------------|
| Clave Primaria | id_direccion |
| Atributos | barrio calle numero |

Tabla 6. Direccion

| Tabla | Empleado |
|----------------|--|
| Clave Primaria | id_empleado |
| Atributos | id_empleado ci nombre ap_paterno ap_materno sexo teléfono fecha_contrato fecha_despido |

| | |
|-----------------|--|
| Claves Foráneas | direccionid_direccion cargoid_cargo |
|-----------------|--|

Tabla 7. Empleado

| | |
|----------------|------------|
| Tabla | Gastos |
| Clave Primaria | id_gasto |
| Atributos | tipo_gasto |

Tabla 8. Gastos

| | |
|-----------------|-----------------------------------|
| Tabla | Mesa |
| Clave Primaria | id_mesa |
| Atributos | num_mesa estado num_persona |
| Claves Foráneas | pedidoid_pedido |

Tabla 9. Mesa

| | |
|-----------------|--|
| Tabla | Pedido |
| Clave Primaria | id_pedido |
| Atributos | hora_pedido fecha_pedido |
| Claves Foráneas | usuarioid_usuario clienteid_cliente |

Tabla 10. Pedido

| Tabla | Plato1 |
|-----------------|--|
| Clave Primaria | id_plato |
| Atributos | cant_platos |
| Claves Foráneas | detalle_platoid_det_plato consumoid_consumo |

Tabla 11. Plato1

| Tabla | Proveedor |
|-----------------|---------------------------------------|
| Clave Primaria | id_proveedor |
| Atributos | nombre_p tipo_producto telefono |
| Claves Foráneas | direccionid_direccion |

Tabla 12. Proveedor

| Tabla | Reserva |
|-----------------|--|
| Clave Primaria | id_reserva |
| Atributos | fecha_ocupacion hora_ocupacion hora_lim_espera confirmado |
| Claves Foráneas | usuarioid_usuario clienteid_cliente |

Tabla 13. Reserva

| Tabla | Salario |
|-----------------|---------------|
| Clave Primaria | id_salario |
| Atributos | sueldo |
| Claves Foráneas | cargoid_cargo |

Tabla 14. Salario

| Tabla | Servicio |
|-----------------|--|
| Clave Primaria | id_servicio |
| Atributos | tipo_servicio fecha_pago_s importe_total_s |
| Claves Foráneas | empleadoid_empleado gastosid_gasto |

Tabla 15. Servicio

| Tabla | Suministro |
|-----------------|--|
| Clave Primaria | id_suministro |
| Atributos | tipo_suministro fecha_compra cantidad_producto importe_total_suministro |
| Claves Foráneas | empleadoid_empleado gastosid_gasto |

Tabla 16. Suministro

| Tabla | Usuario |
|-----------------|---------------------------------|
| Clave Primaria | id_usuario |
| Atributos | nombre_u clave_u estado_u |
| Claves Foráneas | empleadoid_empleado |

Tabla 17. Usuario

| Tabla | proveedor_suministro |
|-----------------|--|
| Claves Foráneas | proveedorid_proveedor suministroid_suministro |

Tabla 18. proveedor_suministro

| Tabla | reserva_reserva |
|-----------------|----------------------------------|
| Claves Foráneas | reservaid_reserva mesaid_mesa |

Tabla 19. reserva_reserva

II.1.5.1.6.3. Creación de tablas en Consultas SQL

```
-----
-- 1. Tabla Bebida1
-----
```

```
CREATE TABLE bebida1
(
  id_bebida serial NOT NULL,
  botellas integer NOT NULL,
  detalle_bebidaid_det_bebida serial NOT NULL,
  consumoid_consumo integer NOT NULL,
  CONSTRAINT bebida1_pkey PRIMARY KEY (id_bebida),
```

```
CONSTRAINT fkbebida1490228 FOREIGN KEY (consumoid_consumo)
```

```
REFERENCES consumo (id_consumo) MATCH SIMPLE
```

```
ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
CONSTRAINT fkbebida1758289 FOREIGN KEY (detalle_bebidaid_det_bebida)
```

```
REFERENCES detalle_bebida (id_det_bebida) MATCH SIMPLE
```

```
ON UPDATE NO ACTION ON DELETE NO ACTION
```

```
)
```

```
WITH (
```

```
  OIDS=FALSE
```

```
);
```

```
ALTER TABLE bebida1 OWNER TO postgres;
```

```
-----  
-- 2. Tabla Cargo  
-----
```

```
CREATE TABLE cargo
```

```
(
```

```
  id_cargo serial NOT NULL,
```

```
  nom_cargo character varying(255) NOT NULL,
```

```
  descripcion character varying(255) NOT NULL,
```

```
  tipo_permiso integer NOT NULL,
```

```
  CONSTRAINT cargo_pkey PRIMARY KEY (id_cargo)
```

```
)
```

```
WITH (
```

```
  OIDS=FALSE
```

```
);
```

```
ALTER TABLE cargo OWNER TO postgres;
```

```
-----  
-- 3. Tabla Cliente  
-----
```

```
CREATE TABLE cliente  
(  
  id_cliente serial NOT NULL,  
  ci_cliente integer NOT NULL,  
  nombre_cli character varying(255) NOT NULL,  
  ap_cli character varying(255) NOT NULL,  
  am_cli character varying(255) NOT NULL,  
  sexo_cli character varying(255) NOT NULL,  
  CONSTRAINT cliente_pkey PRIMARY KEY (id_cliente)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE cliente OWNER TO postgres;
```

```
-----  
-- 4. Tabla Consumo  
-----
```

```
CREATE TABLE consumo  
(  
  id_consumo serial NOT NULL,  
  estado_consumo character varying(255) NOT NULL,  
  precio_t double precision NOT NULL,  
  pedidoid_pedido integer NOT NULL,  
  CONSTRAINT consumo_pkey PRIMARY KEY (id_consumo),
```

```
CONSTRAINT fkconsumo811934 FOREIGN KEY (pedidoid_pedido)
  REFERENCES pedido (id_pedido) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE consumo OWNER TO postgres;

-----
-- 5. Tabla Detalle_bebida
-----

CREATE TABLE detalle_bebida
(
  id_det_bebida serial NOT NULL,
  nom_producto character varying(255) NOT NULL,
  caracteristica_bebida character varying(255) NOT NULL,
  cc_bebida integer NOT NULL,
  precio_unit_b integer NOT NULL,
  CONSTRAINT detalle_bebida_pkey PRIMARY KEY (id_det_bebida)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE detalle_bebida OWNER TO postgres;

-----
-- 6. Tabla Detalle_plato
```

```
-----  
CREATE TABLE detalle_plato  
(  
  id_det_plato serial NOT NULL,  
  nom_plato character varying(255) NOT NULL,  
  caracteristica_plato character varying(255) NOT NULL,  
  porcion_x_persona integer NOT NULL,  
  precio_unit_p integer NOT NULL,  
  CONSTRAINT detalle_plato_pkey PRIMARY KEY (id_det_plato)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE detalle_plato OWNER TO postgres;  
  
-----  
-- 7. Tabla Detalle_suministro  
-----  
CREATE TABLE detalle_suministro  
(  
  id_det_sum serial NOT NULL,  
  nom_sum character varying(255) NOT NULL,  
  caracteristas_sum character varying(255) NOT NULL,  
  precio_unit_sum double precision NOT NULL,  
  suministroid_suministro integer NOT NULL,  
  CONSTRAINT detalle_suministro_pkey PRIMARY KEY (id_det_sum),  
  CONSTRAINT fkdetalle_su89826 FOREIGN KEY (suministroid_suministro)  
    REFERENCES suministro (id_suministro) MATCH SIMPLE
```

```
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
    OIDS=FALSE
);
ALTER TABLE detalle_suministro OWNER TO postgres;
```

```
-----
-- 8. Tabla Direccion
-----
```

```
CREATE TABLE direccion
(
    id_direccion serial NOT NULL,
    barrio character varying(255) NOT NULL,
    calle character varying(255) NOT NULL,
    numero integer NOT NULL,
    CONSTRAINT direccion_pkey PRIMARY KEY (id_direccion)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE direccion OWNER TO postgres;
```

```
-----
-- 9. Tabla Empleado
```

```
-----  
CREATE TABLE empleado  
(  
  id_empleado serial NOT NULL,  
  ci integer NOT NULL,  
  nombre character varying(255) NOT NULL,  
  ap_paterno character varying(255) NOT NULL,  
  ap_materno character varying(255) NOT NULL,  
  sexo character varying(255) NOT NULL,  
  telefono character varying(255) NOT NULL,  
  fecha_contrato date NOT NULL,  
  fecha_despido date,  
  direccionid_direccion integer NOT NULL,  
  cargoid_cargo integer NOT NULL,  
  CONSTRAINT empleado_pkey PRIMARY KEY (id_empleado),  
  CONSTRAINT fkempleado230288 FOREIGN KEY (direccionid_direccion)  
    REFERENCES direccion (id_direccion) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION,  
  CONSTRAINT fkempleado94375 FOREIGN KEY (cargoid_cargo)  
    REFERENCES cargo (id_cargo) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE empleado OWNER TO postgres;
```

```
-----  
-- 10. Tabla Gastos  
-----
```

```
CREATE TABLE gastos  
(  
  id_gasto serial NOT NULL,  
  tipo_gasto character varying(255) NOT NULL,  
  CONSTRAINT gastos_pkey PRIMARY KEY (id_gasto)  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE gastos OWNER TO postgres;
```

```
-----  
-- 11. Tabla Mesa  
-----
```

```
CREATE TABLE mesa  
(  
  id_mesa serial NOT NULL,  
  num_mesa integer NOT NULL,  
  estado character varying(255) NOT NULL,  
  num_persona integer NOT NULL,  
  pedidoid_pedido integer,  
  CONSTRAINT mesa_pkey PRIMARY KEY (id_mesa),  
  CONSTRAINT fkmesa965494 FOREIGN KEY (pedidoid_pedido)  
    REFERENCES pedido (id_pedido) MATCH SIMPLE
```

```
        ON UPDATE NO ACTION ON DELETE NO ACTION
    )
WITH (
    OIDS=FALSE
);
ALTER TABLE mesa OWNER TO postgres;

-----

-- 12. Tabla Pedido
-----

CREATE TABLE pedido
(
    id_pedido serial NOT NULL,
    hora_pedido character varying(255) NOT NULL,
    fecha_pedido date NOT NULL,
    usuarioid_usuario integer NOT NULL,
    clienteid_cliente integer,
    CONSTRAINT pedido_pkey PRIMARY KEY (id_pedido),
    CONSTRAINT fkpedido329779 FOREIGN KEY (usuarioid_usuario)
        REFERENCES usuario (id_usuario) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkpedido819954 FOREIGN KEY (clienteid_cliente)
        REFERENCES cliente (id_cliente) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
    OIDS=FALSE
);
```

```
ALTER TABLE pedido OWNER TO postgres;

-----

-- 13. Tabla Plato1
-----

CREATE TABLE plato1
(
  id_plato serial NOT NULL,
  cantidad_platos integer NOT NULL,
  detalle_platoid_det_plato integer NOT NULL,
  consumoid_consumo integer NOT NULL,
  CONSTRAINT plato1_pkey PRIMARY KEY (id_plato),
  CONSTRAINT fkplato1389547 FOREIGN KEY (consumoid_consumo)
    REFERENCES consumo (id_consumo) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fkplato1434634 FOREIGN KEY (detalle_platoid_det_plato)
    REFERENCES detalle_plato (id_det_plato) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE plato1 OWNER TO postgres;

-----

-- 14. Tabla Proveedor
-----

CREATE TABLE proveedor
```

```
(
id_proveedor serial NOT NULL,
nombre_p character varying(255) NOT NULL,
tipo_producto character varying(255) NOT NULL,
telefono character varying(255) NOT NULL,
direccionid_direccion integer NOT NULL,
CONSTRAINT proveedor_pkey PRIMARY KEY (id_proveedor),
CONSTRAINT fkproveedor168806 FOREIGN KEY (direccionid_direccion)
REFERENCES direccion (id_direccion) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
OIDS=FALSE
);
ALTER TABLE proveedor OWNER TO postgres;

-----
-- 15. Tabla Proveedor_suministro
-----

CREATE TABLE proveedor_suministro
(
proveedorid_proveedor integer NOT NULL,
suministroid_suministro integer NOT NULL,
CONSTRAINT proveedor_suministro_pkey PRIMARY KEY (proveedorid_proveedor,
suministroid_suministro),
CONSTRAINT fkproveedor_43781 FOREIGN KEY (proveedorid_proveedor)
REFERENCES proveedor (id_proveedor) MATCH SIMPLE
ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
CONSTRAINT fkproveedor_551692 FOREIGN KEY (suministroid_suministro)
  REFERENCES suministro (id_suministro) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE proveedor_suministro OWNER TO postgres;

-----
-- 16. Tabla Reserva
-----

CREATE TABLE reserva
(
  id_reserva serial NOT NULL,
  fecha_ocupacion date NOT NULL,
  hora_ocupacion character varying(255) NOT NULL,
  hora_lim_espera character varying(255) NOT NULL,
  confirmado character varying(255) NOT NULL,
  usuarioid_usuario integer NOT NULL,
  clienteid_cliente integer NOT NULL,
  CONSTRAINT reserva_pkey PRIMARY KEY (id_reserva),
  CONSTRAINT fkreserva291145 FOREIGN KEY (usuarioid_usuario)
    REFERENCES usuario (id_usuario) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fkreserva781320 FOREIGN KEY (clienteid_cliente)
    REFERENCES cliente (id_cliente) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
```

```
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE reserva OWNER TO postgres;  
  
-----  
-- 17. Tabla Reserva_mesa  
-----  
  
CREATE TABLE reserva_mesa  
(  
  reservaid_reserva integer NOT NULL,  
  mesaid_mesa integer NOT NULL,  
  CONSTRAINT reserva_mesa_pkey PRIMARY KEY (reservaid_reserva, mesaid_mesa),  
  CONSTRAINT fkreserva_me269611 FOREIGN KEY (reservaid_reserva)  
    REFERENCES reserva (id_reserva) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION,  
  CONSTRAINT fkreserva_me462974 FOREIGN KEY (mesaid_mesa)  
    REFERENCES mesa (id_mesa) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION  
)  
WITH (  
  OIDS=FALSE  
);  
  
ALTER TABLE reserva_mesa OWNER TO postgres;  
  
-----  
-- 18. Tabla Salario
```

```
-----  
CREATE TABLE salario  
(  
  id_salario serial NOT NULL,  
  sueldo integer NOT NULL,  
  cargoid_cargo integer NOT NULL,  
  CONSTRAINT salario_pkey PRIMARY KEY (id_salario),  
  CONSTRAINT fksalario535639 FOREIGN KEY (cargoid_cargo)  
    REFERENCES cargo (id_cargo) MATCH SIMPLE  
    ON UPDATE NO ACTION ON DELETE NO ACTION  
)  
WITH (  
  OIDS=FALSE  
);  
ALTER TABLE salario OWNER TO postgres;
```

```
-----  
-- 19. Tabla Servicio  
-----
```

```
CREATE TABLE servicio  
(  
  id_servicio serial NOT NULL,  
  tipo_servicio character varying(255) NOT NULL,  
  fecha_pago_s date NOT NULL,  
  importe_total_s double precision NOT NULL,  
  empleadoid_empleado integer NOT NULL,  
  gastosid_gasto integer NOT NULL,  
  CONSTRAINT servicio_pkey PRIMARY KEY (id_servicio),
```

```
CONSTRAINT fkservicio462098 FOREIGN KEY (empleadoid_pleado)
  REFERENCES empleado (id_pleado) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT fkservicio974168 FOREIGN KEY (gastosid_gasto)
  REFERENCES gastos (id_gasto) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE servicio OWNER TO postgres;
```

-- 20. Tabla Suministro

```
CREATE TABLE suministro
(
  id_suministro serial NOT NULL,
  tipo_suministro character varying(255) NOT NULL,
  fecha_compra date NOT NULL,
  cantidad_producto double precision NOT NULL,
  importe_total_suministro double precision NOT NULL,
  empleadoid_pleado integer NOT NULL,
  gastosid_gasto integer NOT NULL,
  CONSTRAINT suministro_pkey PRIMARY KEY (id_suministro),
  CONSTRAINT fksuministro684440 FOREIGN KEY (gastosid_gasto)
    REFERENCES gastos (id_gasto) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
CONSTRAINT fksuministro799220 FOREIGN KEY (empleadoid_empleado)
  REFERENCES empleado (id_empleado) MATCH SIMPLE
  ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);
ALTER TABLE suministro OWNER TO postgres;

-----
-- 21. Tabla Usuario
-----

CREATE TABLE usuario
(
  id_usuario serial NOT NULL,
  nombre_u character varying(255) NOT NULL,
  clave_u character varying(255) NOT NULL,
  empleadoid_empleado integer NOT NULL,
  estado_u character varying(255) NOT NULL,
  CONSTRAINT usuario_pkey PRIMARY KEY (id_usuario),
  CONSTRAINT fkusuario417154 FOREIGN KEY (empleadoid_empleado)
    REFERENCES empleado (id_empleado) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (
  OIDS=FALSE
);ALTER TABLE usuario OWNER TO postgres;
```

II.1.5.1.7. Especificación de Métodos

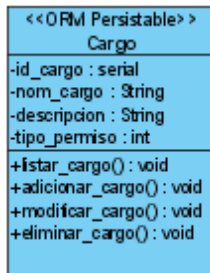


Figura 84 Clase Cargo

| | |
|--------------------|---|
| Objeto | Cargo |
| Método | listar_cargo |
| Detalle | obj=bd.select("select nom_cargo, tipo_permiso, id_cargo from cargo"); |
| Descripción | Por medio de este método, se puede listar los cargos. |

| | |
|---------------|-----------------|
| Objeto | Cargo |
| Método | adicionar_cargo |

| | |
|--------------------|--|
| Detalle | <pre>bd.update("insert into cargo values('"+t1.getText()+"', '"+t3.getText()+"', "+(Integer)sp.getValue()+")"); bd.update("insert into salario values('"+t2.getText()+"', "+id_cargo+"");</pre> |
| Descripción | Por medio de este método, se puede adicionar cargos. |

| | |
|--------------------|--|
| Objeto | Cargo |
| Método | modificar_cargo |
| Detalle | <pre>bd.update("update salario set sueldo='"+t2.getText()+" where cargoid_cargo="+id_c); bd.update("update cargo set nom_cargo='"+t1.getText()+"', descripcion='"+t3.getText()+"', tipo_permiso='"+(Integer)sp.getValue()+" where id_cargo="+id_c);</pre> |
| Descripción | Por medio de este método, se puede modificar cargos. |

| | |
|--|--|
| | |
|--|--|

| | |
|--------------------|--|
| Objeto | Cargo |
| Método | eliminar_cargo |
| Detalle | <pre>bd.update("delete from salario where cargoid_cargo="+id_cargo2+""); bd.update("delete from cargo where id_cargo="+id_cargo2+"");</pre> |
| Descripción | Por medio de este método, se puede eliminar cargos. |

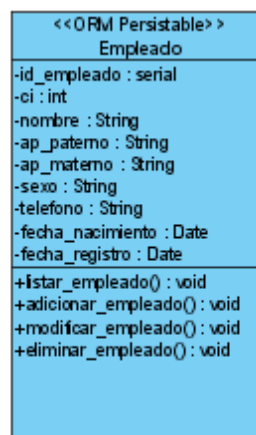


Figura 85 Clase Empleado

| | |
|---------------|----------|
| Objeto | Empleado |
|---------------|----------|

| | |
|--------------------|---|
| Método | listar_empleado |
| Detalle | <pre>bd.select("select e.nombre, e.ap_paterno, e.ap_materno, c.nom_cargo, e.id_empleado, e.direccionid_direccion from empleado e, cargo c where c.id_cargo=e.cargoid_cargo"); bd.select("select e.fecha_nacimiento, e.telefono, e.fecha_registro, d.calle, d.barrio, d.numero from empleado e, direccion d where e.direccionid_direccion=d.id_direccion and e.id_empleado="+id_emp); bd.select("Select * from usuario where empleadoid_empleado="+id_emp2);</pre> |
| Descripción | Por medio de este método, se puede listar los empleados. |

| | |
|----------------|---|
| Objeto | Empleado |
| Método | adicionar_empleado |
| Detalle | <pre>bd.update("insert into direccion values('"+t7.getText()+"", "+t8.getText()+"", "+t9.getText()+")"); bd.update("insert into empleado(ci, nombre, ap_paterno, ap_materno, sexo, telefono, fecha_nacimiento, direccionid_direccion, cargoid_cargo,e_estado) values("+t1.getText()+", "+t2.getText()+", "+t3.getText()+",</pre> |

| | |
|--------------------|--|
| | <pre> "+t4.getText()+",""+sexo+", "+t5.getText()+", "+fecha_c+", "+max_id_dir+", "+id_cargo+',ACTIVO' "); </pre> |
| Descripción | Por medio de este método, se puede adicionar empleados. |

| | |
|--------------------|--|
| Objeto | Empleado |
| Método | modificar_empleado |
| Detalle | <pre> bd.update("update direccion set barrio="+t7.getText()+", calle="+t8.getText()+", numero="+t9.getText()+" where id_direccion in(select direccionid_direccion from empleado where id_empleado="+id_emp+")"); bd.update("update empleado set nombre="+t2.getText()+", ap_paterno="+t3.getText()+", ap_materno="+t4.getText()+", sexo="+sexo+", telefono="+t5.getText()+", fecha_contrato="+fecha_c+", cargoid_cargo="+id_cargo+", e_estado='ACTIVO' where id_empleado="+id_emp); </pre> |
| Descripción | Por medio de este método, se puede modificar empleados. |

| | |
|--|--|
| | |
|--|--|

| | |
|--------------------|---|
| Objeto | Empleado |
| Método | eliminar_empleado |
| Detalle | <pre>bd.update("delete from empleado where id_empleado="+id_emp+""); bd.update("delete from direccion where id_direccion="+id_direc+"");</pre> |
| Descripción | Por medio de este método, se puede eliminar empleados. |

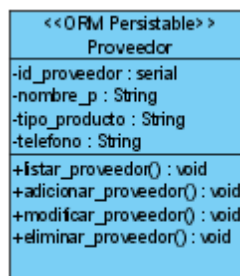


Figura 86 Clase Proveedor

| | |
|---------------|-----------|
| Objeto | Proveedor |
| | |

| | |
|--------------------|--|
| Método | listar_proveedor |
| Detalle | <pre>bd.select("select nombre_p, tipo_producto, id_proveedor, direccionid_direccion from proveedor"); bd.select("select p.telefono, d.calle, d.barrio, d.numero from proveedor p, direccion d" where p.direccionid_direccion=d.id_direccion and p.id_proveedor="+id_pro);</pre> |
| Descripción | Por medio de este método, se puede listar los proveedores. |

| | |
|----------------|---|
| Objeto | Proveedor |
| Método | adicionar_proveedor |
| Detalle | <pre>bd.update("insert into direccion values('"+t4.getText()+"', '"+t5.getText()+"', '"+t6.getText()+""); bd.update("insert into proveedor(nombre_p, tipo_producto, telefono, direccionid_direccion)" values('"+t1.getText()+"', '"+t2.getText()+"', '"+t3.getText()+"',"+max_id_dir+" ");</pre> |
| | |

| | |
|--------------------|---|
| Descripción | Por medio de este método, se puede adicionar proveedores. |
|--------------------|---|

| | |
|--------------------|---|
| Objeto | Proveedores |
| Método | modificar_proveedor |
| Detalle | <pre> bd.update("update direccion set barrio="+t4.getText()+", calle="+t5.getText()+", numero="+t6.getText()+" where id_direccion in(select direccionid_direccion from proveedor where id_proveedor="+id_pro+") "); bd.update("update proveedor set nombre_p="+t1.getText()+", tipo_producto="+t2.getText()+", telefono="+t3.getText()+"" where id_proveedor="+id_pro); </pre> |
| Descripción | Por medio de este método, se puede modificar proveedores. |

| | |
|---------------|-------------|
| Objeto | Proveedores |
| | |

| | |
|--------------------|---|
| Método | eliminar_proveedor |
| Detalle | <pre>bd.update("delete from proveedor where id_proveedor="+id_pro+""); bd.update("delete from direccion where id_direccion="+id_direc+"");</pre> |
| Descripción | Por medio de este método, se puede eliminar proveedores. |

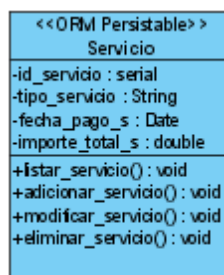


Figura 87 Clase Servicio

| | |
|---------------|---|
| Objeto | Servicio |
| Método | listar_servicio |
| | bd.select("select tipo_servicio, fecha_pago_s, importe_total_s, |

| | |
|--------------------|---|
| Detalle | id_servicio from servicio where (fecha_pago_s>="" + fechadesde.getDate() + "")and(fecha_pago_s<="" + fechahasta.getDate() + ""); |
| Descripción | Por medio de este método, se puede listar los servicios. |

| | |
|--------------------|---|
| Objeto | Servicio |
| Método | adicionar_servicio |
| Detalle | bd.update("insert into servicio(tipo_servicio, fecha_pago_s, importe_total_s, empleadoid_empleado,gastosid_gasto)" + " values("" + t1.getText() + "", "" + fecharegistro.getDate() + "", "" + Double.parseDouble(t2.getText()) + "", "" + id_e + "", "" + id_g + ""); |
| Descripción | Por medio de este método, se puede adicionar servicios. |

| | |
|--|--|
| | |
|--|--|

| | |
|--------------------|---|
| Objeto | Servicio |
| Método | modificar_servicio |
| Detalle | |
| Descripción | Por medio de este método, se puede modificar servicios. |

| | |
|--------------------|---|
| Objeto | Servicio |
| Método | eliminar_servicio |
| Detalle | bd.update("delete from servicio where id_servicio="+id_servv+""); |
| Descripción | Por medio de este método, se puede eliminar servicio. |

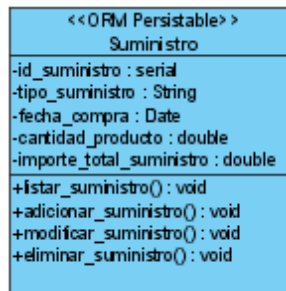


Figura 88 Clase Suministro

| | |
|--------------------|--|
| Objeto | Suministro |
| Método | listar_suministro |
| Detalle | <pre> bd.select("select s.tipo_suministro, ds.nom_sum, s.fecha_compra, s.cantidad_producto, ds.precio_unit_sum, s.importe_total_suministro, s.id_suministro, ds.id_det_sum from suministro s,detalle_suministro ds where (s.id_suministro=ds.suministroid_suministro)and (s.fecha_compra>=""+fechadesde.getDate()+")and(s.fecha_compra<=""+fechahasta.getDate()+""); </pre> |
| Descripción | Por medio de este método, se puede listar los suministros. |

| | |
|---------------|------------|
| Objeto | Suministro |
|---------------|------------|

| | |
|--------------------|---|
| Método | adicionar_suministro |
| Detalle | <pre> bd.update("insert into suministro(tipo_suministro, fecha_compra, cantidad_producto, importe_total_suministro,empleadoid_empleado,gastosid_gasto) values("+t1.getText()+",""+fecharegistro.getDate()+", "+Double.parseDouble(t4.getText()+",""+Double.parseDouble(t6.getText())+", "+id_e+", "+id_g+)"); bd.update("insert into detalle_suministro(nom_sum, caracteristica_sum, precio_unit_sum, suministroid_suministro) values("+t2.getText()+", "+t5.getText()+", "+Double.parseDouble(t3.getText()+", "+maxidsum[0][0]+)"); </pre> |
| Descripción | Por medio de este método, se puede adicionar suministros. |

| | |
|--------------------|---|
| Objeto | Suministros |
| Método | modificar_suministro |
| Detalle | |
| Descripción | Por medio de este método, se puede modificar suministros. |

| | |
|--|--|
| | |
|--|--|

| | |
|--------------------|--|
| Objeto | Suministros |
| Método | eliminar_suministro |
| Detalle | <pre>bd.update("delete from detalle_suministro where id_det_sum="+id_det_sum+""); bd.update("delete from suministro where id_suministro="+id_sum+"");</pre> |
| Descripción | Por medio de este método, se puede eliminar suministro. |

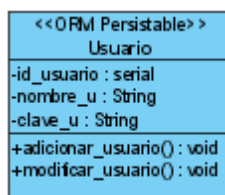


Figura 89 Clase Usuario

| | |
|---------------|---------|
| Objeto | Usuario |
|---------------|---------|

| | |
|--------------------|--|
| Método | adicionar_usuario |
| Detalle | <code>bd.update("insert into usuario values("+t1.getText()+", "+t2.getText()+", "+ide+", 'ACTIVO')");</code> |
| Descripción | Por medio de este método, se puede adicionar usuarios. |

| | |
|--------------------|---|
| Objeto | Usuario |
| Método | modificar_usuario |
| Detalle | <code>bd.update("update usuario set nombre_u="+t1.getText()+", estado_u="+cb.getSelectedItem()+" where id_usuario="+id_user+"");</code> |
| Descripción | Por medio de este método, se puede modificar usuarios. |

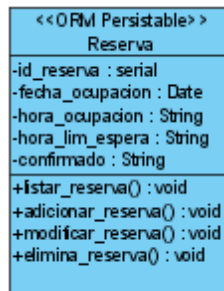


Figura 90 Clase Reserva

| | |
|--------------------|--|
| Objeto | Reserva |
| Método | listar_reserva |
| Detalle | bd.select("select r.id_reserva, m.num_mesa, r.fecha_ocupacion, r.hora_ocupacion, r.hora_lim_espera,r.clienteid_cliente ,r.test from reserva r,reserva_mesa rs,mesa m where (r.fecha_ocupacion='"+fecha+"')and(r.confirmado='NO')and(r.id_reserva=r s.reservaid_reserva)and(rs.mesaid_mesa=m.id_mesa) order by fecha_ocupacion"); |
| Descripción | Por medio de este método, se puede listar las reservas. |
| Objeto | Reserva |
| Método | adicionar_reserva |

| | |
|--------------------|--|
| Detalle | <pre> bd.update("insert into reserva values("+fechareserva.getDate()+",""+t1.getText()+",""+t2.getText()+",'N O'," +id_u+", "+id_cli+", 'HOY')"); bd.update("insert into reserva_mesa values("+maxidres[0][0]+", "+id_m+""); bd.update("insert into pedido values("+t1.getText()+", "" +fechareserva.getDate()+", "+id_u+", "+id_cli+""); bd.update("insert into consumo values("+t3.getText()+", "+ftotal+", "+maxidped[0][0]+")"); bd.update("insert into plato1 values("+Temp2[i][0]+", "+Temp[i][0]+", "+maxidconsu[0][0]+")"); bd.update("insert into bebida1 values("+Temp3[i][0]+", "+Temp1[i][0]+", "+maxidconsu[0][0]+")"); </pre> |
| Descripción | Por medio de este método, se puede adicionar reservas. |

| | |
|----------------|---|
| Objeto | Reserva |
| Método | modificar_reserva |
| Detalle | <pre> bd.update("update reserva set fecha_ocupacion =""+fechareserva.getDate()+", hora_ocupacion=""+t1.getText()+", hora_lim_esp </pre> |

| | |
|--------------------|--|
| | <pre> era="+t2.getText()+" where hora_ocupacion="+hr+""; bd.update("update pedido set fecha_pedido ="+fechareserva.getDate()+" ,hora_pedido="+t1.getText()+" where hora_pedido="+hr+""); bd.update("update reserva_mesa set mesaid_mesa="+id_m+" where reservaid_reserva="+id_res); bd.update("insert into consumo values("+t3.getText()+" , "+ftotal+", "+max_id_ped+""); bd.update("insert into plato1 values("+Temp2[i][0]+", "+Temp[i][0]+", "+max_id_consu+""); bd.update("insert into bebida1 values("+Temp3[i][0]+", "+Temp1[i][0]+", "+max_id_consu+""); </pre> |
| Descripción | Por medio de este método, se puede modificar reservas. |

| | |
|----------------|--|
| Objeto | Reserva |
| Método | eliminar_reserva |
| Detalle | <pre> bd.update("delete from bebida1 where consumoid_consumo="+consumoid); bd.update("delete from plato1 where consumoid_consumo="+consumoid); </pre> |

| | |
|--------------------|--|
| | <pre>bd.update("delete from consumo where pedidoid_pedido="+max_id_ped); bd.update("delete from pedido where id_pedido="+max_id_ped);</pre> |
| Descripción | Por medio de este método, se puede eliminar reservas. |

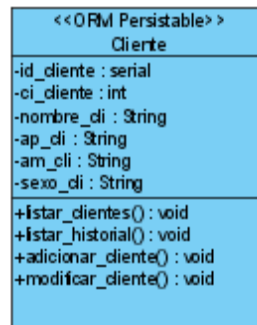


Figura 91 Clase Cliente

| | |
|----------------|--|
| Objeto | Cliente |
| Método | listar_cliente |
| Detalle | bd.select("select nombre_cli, ap_cli, am_cli, ci_cliente, id_cliente from cliente"); |
| | |

| | |
|--------------------|---|
| Descripción | Por medio de este método, se puede listar los clientes. |
|--------------------|---|

| | |
|--------------------|---|
| Objeto | Cliente |
| Método | adicionar_cliente |
| Detalle | bd.update("insert into cliente (ci_cliente, nombre_cli, ap_cli, am_cli, sexo_cli)values("+t5.getText()+", '"+t1.getText()+", '"+t2.getText()+", '"+t3.getText()+", '"+sexo+""); |
| Descripción | Por medio de este método, se puede adicionar clientes. |

| | |
|----------------|---|
| Objeto | Cliente |
| Método | modificar_cliente |
| Detalle | bd.update("update cliente set ci_cliente="+t5.getText()+", nombre_cli="+t1.getText()+", ap_cli="+t2.getText()+", am_cli="+t3.getText()+", sexo_cli="+sexo+" where |

| | |
|--------------------|--|
| | id_cliente="+id); |
| Descripción | Por medio de este método, se puede modificar clientes. |

| | |
|--------------------|--|
| Objeto | Cliente |
| Método | listar_historial |
| Detalle | bd.select("select p.id_pedido, p.hora_pedido, p.fecha_pedido, c.estado_consumo, c.precio_t from consumo c, pedido p where (p.id_pedido=c.pedidoid_pedido)and(p.clienteid_cliente="+id_c+)"); |
| Descripción | Por medio de este método, se puede listar el historial de los clientes. |

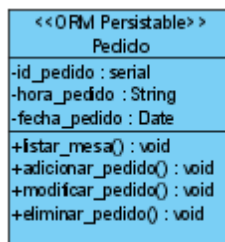


Figura 92 Clase Pedido

| | |
|--------------------|---|
| Objeto | Pedido |
| Método | listar_pedido |
| Detalle | <pre>obj=bd.select("select num_mesa, num_persona, estado, id_mesa, pedidoid_pedido from mesa order by num_mesa"); obj1=bd.select("select d.nom_plato, pla.cantidad_platos, d.precio_unit_p,c.precio_t from mesa m,pedido p,plato1 pla,detalle_plato d,consumo c where (m.pedidoid_pedido=p.id_pedido) and (p.id_pedido=c.pedidoid_pedido) and (c.id_consumo=pla.consumoid_consumo) and (pla.detalle_platoid_det_plato=d.id_det_plato)and(m.id_mesa="+id_mesa+"")"); ; obj2=bd.select("select d.nom_producto, be.botellas, d.precio_unit_b,c.precio_t from mesa m,pedido p,bebida1 be,detalle_bebida d,consumo c where (m.pedidoid_pedido=p.id_pedido) and (p.id_pedido=c.pedidoid_pedido) and (c.id_consumo=be.consumoid_consumo) and (be.detalle_bebidaid_det_bebida=d.id_det_bebida)and(m.id_mesa="+id_mesa+ "")");</pre> |
| Descripción | Por medio de este método, se puede listar los pedidos. |

| | |
|---------------|--------|
| Objeto | Pedido |
|---------------|--------|

| | |
|--------------------|---|
| Método | adicionar_pedido |
| Detalle | <pre> bd.update("insert into pedido values('"+t1.getText()+"', '"+t2.getText()+"', '"+id_us+"',0)"); bd.update("update mesa set estado='"+estado+"',pedidoid_pedido='"+maxidped[0][0]+" where id_mesa='"+id_m); bd.update("insert into consumo values('"+t3.getText()+"', '"+ftotal+"', '"+maxidped[0][0]+""); bd.update("insert into plato1 values('"+Temp2[i][0]+"', '"+Temp[i][0]+"', '"+maxidconsu[0][0]+""); bd.update("insert into bebida1 values('"+Temp3[i][0]+"', '"+Temp1[i][0]+"', '"+maxidconsu[0][0]+""); </pre> |
| Descripción | Por medio de este método, se puede adicionar pedidos. |

| | |
|----------------|--|
| Objeto | Pedido |
| Método | modificar_pedido |
| Detalle | <pre> bd.update("insert into consumo values('"+t3.getText()+"', '"+ftotal+"', '"+max_id_ped+""); bd.update("insert into plato1 values('"+Temp2[i][0]+"', </pre> |

| | |
|--------------------|--|
| | <pre> "+Temp[i][0]+", "+max_id_consu+"); bd.update("insert into bebida1 values("+Temp3[i][0]+", "+Temp1[i][0]+", "+max_id_consu+"); </pre> |
| Descripción | Por medio de este método, se puede modificar pedidos. |

| | |
|--------------------|--|
| Objeto | Pedido |
| Método | eliminar_pedido |
| Detalle | <pre> bd.update("delete from bebida1 where consumoid_consumo="+consumoid); bd.update("delete from plato1 where consumoid_consumo="+consumoid); bd.update("delete from consumo where pedidoid_pedido="+max_id_ped); bd.update("delete from pedido where id_pedido="+max_id_ped); </pre> |
| Descripción | Por medio de este método, se puede eliminar los pedidos. |

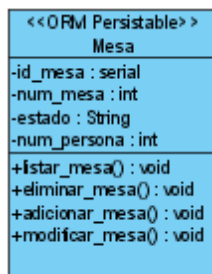


Figura 93 Clase Mesa

| | |
|--------------------|---|
| Objeto | Mesa |
| Método | listar_mesa |
| Detalle | obj=bd.select("select num_mesa, num_persona, estado, id_mesa from mesa"); |
| Descripción | Por medio de este método, se puede listar las mesas. |

| | |
|---------------|----------------|
| Objeto | Mesa |
| Método | adicionar_mesa |
| | |

| | |
|--------------------|---|
| Detalle | <code>bd.update("insert into mesa values("+t1.getText()+", '"+t2.getText()+", "+(Integer)sp.getValue()+")");</code> |
| Descripción | Por medio de este método, se puede adicionar mesas. |

| | |
|--------------------|---|
| Objeto | Mesa |
| Método | modificar_mesa |
| Detalle | <code>bd.update("update mesa set num_mesa="+t1.getText()+", num_persona="+((Integer)sp.getValue())+", estado='"+t2.getText()+"' where id_cargo="+id_me);</code> |
| Descripción | Por medio de este método, se puede modificar mesas. |

| | |
|---------------|---------------|
| Objeto | Mesa |
| Método | eliminar_mesa |

| | |
|--------------------|---|
| Detalle | <code>bd.update("delete from mesa where id_mesa="+id_mesa+"");</code> |
| Descripción | Por medio de este método, se puede eliminar las mesas. |

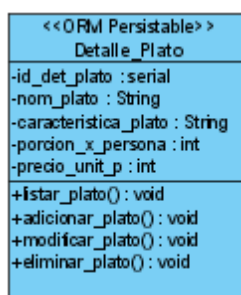


Figura 94 Clase Detalle Plato

| | |
|--------------------|--|
| Objeto | Detalle_Plato |
| Método | listar_plato |
| Detalle | <code>bd.select("select nom_plato, porcion_x_persona, precio_unit_p, caracteristica_plato, id_det_plato from detalle_plato");</code> |
| Descripción | Por medio de este método, se puede listar los platos. |

| | |
|--------------------|--|
| Objeto | Detalle_Plato |
| Método | adicionar_plato |
| Detalle | bd.update("insert into detalle_plato(nom_plato, caracteristica_plato, porcion_x_persona, precio_unit_p) values('"+t1.getText()+"', '"+t1.getText()+"', '"+t2.getText()+"', '"+t3.getText()+"");"); |
| Descripción | Por medio de este método, se puede adicionar platos. |

| | |
|--------------------|--|
| Objeto | Detalle_Plato |
| Método | modificar_plato |
| Detalle | bd.update("update detalle_plato set nom_plato='"+t1.getText()+"', caracteristica_plato='"+t1.getText()+"', porcion_x_persona='"+t2.getText()+"', precio_unit_p='"+t3.getText()+" where id_det_plato='"+id_pla);"); |
| Descripción | Por medio de este método, se puede modificar platos. |

| | |
|--|--|
| | |
|--|--|

| | |
|--------------------|---|
| Objeto | Detalle_Plato |
| Método | eliminar_plato |
| Detalle | bd.update("delete from detalle_plato where id_det_plato="+id_p+""); |
| Descripción | Por medio de este método, se puede eliminar las mesas. |

```

<<ORMI Persistable>>
Detalle_Bebida
-id_det_bebida : serial
-nom_producto : String
-caracteristica_bebida : String
-cc_bebida : int
-precio_unit_b : int
+listar_bebida() : void
+adicionar_bebida() : void
+modificar_bebida() : void
+eliminar_bebida() : void
    
```

Figura 95 Clase Detalle Bebida

| | |
|---------------|----------------|
| Objeto | Detalle_Bebida |
| Método | listar_bebida |

| | |
|--------------------|---|
| | |
| Detalle | bd.select("select nom_producto, cc_bebida, precio_unit_b, característica_bebida, id_det_bebida from detalle_bebida"); |
| Descripción | Por medio de este método, se puede listar las bebidas. |

| | |
|--------------------|---|
| Objeto | Detalle_Bebida |
| Método | adicionar_bebida |
| Detalle | bd.update("insert into detalle_bebida(nom_producto, característica_bebida, cc_bebida, precio_unit_b)values('"+t1.getText()+"', '"+t1.getText()+"', '"+t2.getText()+"', '"+t3.getText()+"')"); |
| Descripción | Por medio de este método, se puede adicionar bebidas. |

| | |
|---------------|------------------|
| Objeto | Detalle_Bebida |
| Método | modificar_bebida |

| | |
|--------------------|---|
| | |
| Detalle | <pre>bd.update("update detalle_bebida set nom_producto="+t1.getText()+", caracteristica_bebida="+te1.getText()+", cc_bebida="+t2.getText()+", precio_unit_b="+t3.getText()+ where id_det_bebida="+id_be);</pre> |
| Descripción | Por medio de este método, se puede modificar bebidas. |

| | |
|--------------------|--|
| Objeto | Detalle_Bebida |
| Método | eliminar_bebida |
| Detalle | <pre>bd.update("delete from detalle_bebida where id_det_bebida="+id_b+"");</pre> |
| Descripción | Por medio de este método, se puede eliminar las bebidas. |

II.1.5.1.8. Interfaz Grafica de Usuario (GUI)

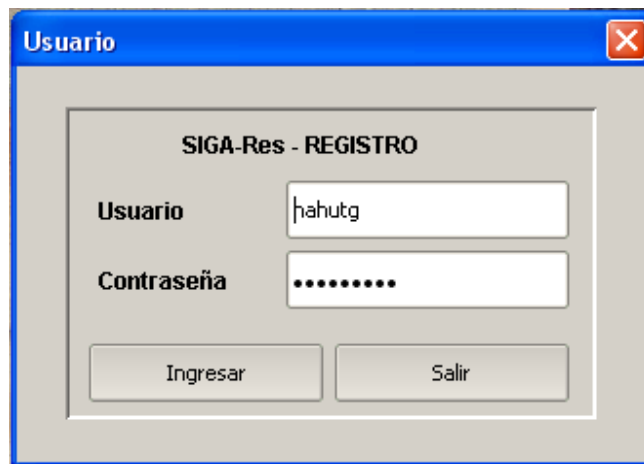


Figura 96 Ingreso de Usuario

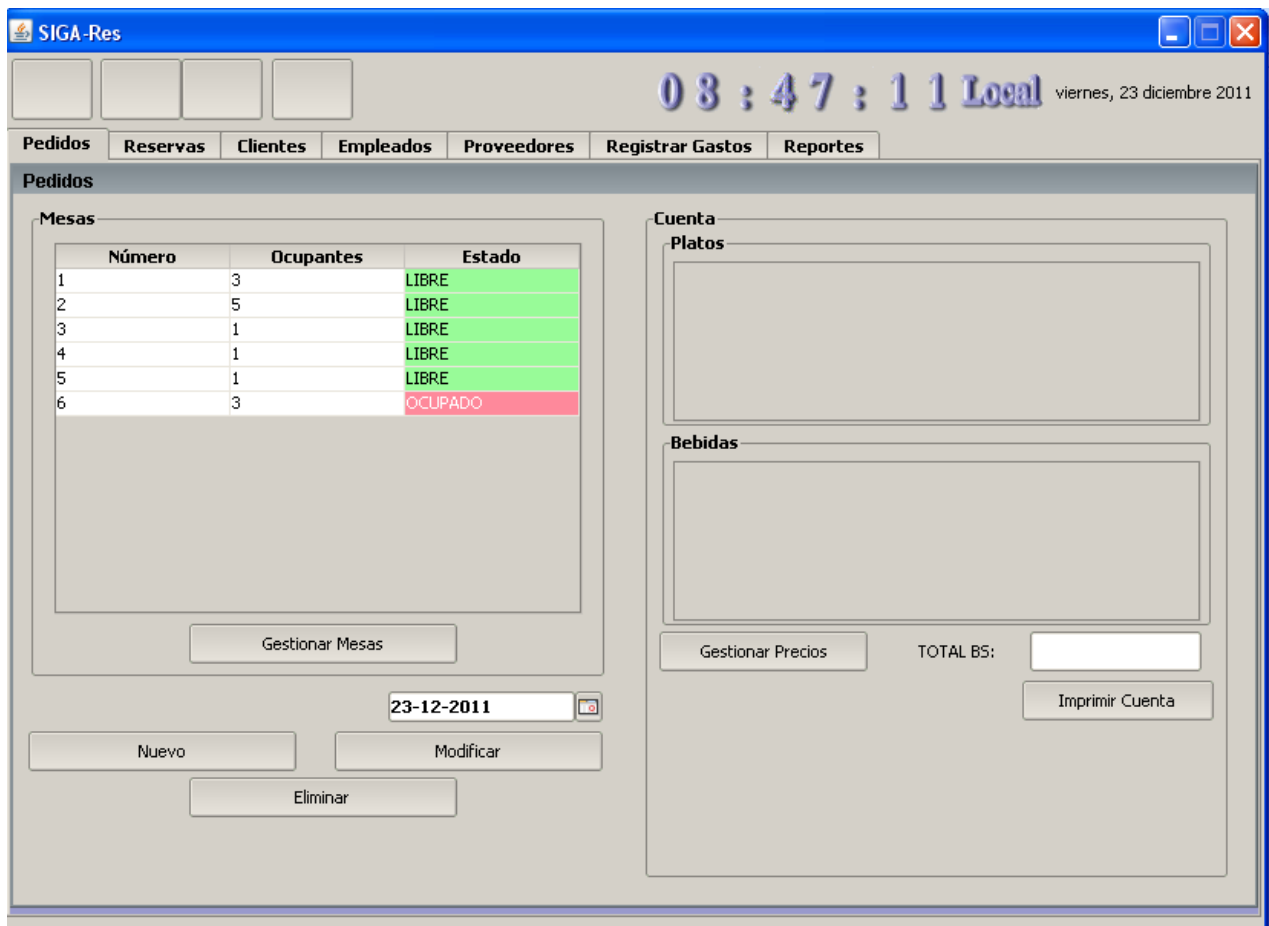


Figura 97 Gestionar Pedidos

Datos
 Hora Pedido: 08:47:57 Fecha de Pedido: 2011-12-23 Estado: SIN CANCELAR

Generar Pedido

Tipos de platos disponibles

| NOMBRE | PRECIO |
|----------------------|--------|
| CHICHARRON DE SURUBI | 35 |
| SABALO A LA PARRILLA | 28 |
| SABALO FRITO C/L | 25 |
| SABALO FRITO S/L | 22 |
| SURUBI A LA PARRILLA | 37 |

TROZOS PEQUEÑOS DE CHICHARRON FRITOS ACOMPAÑADOS DE PAPA, MOTE Y LIMON

Tipos de bebidas gaseosas y vinos disponibles

| NOMBRE | PRECIO |
|----------------|--------|
| AGUA PERSONAL | 4 |
| CERVEZA PACEÑA | 12 |
| COCA COLA | 5 |
| COCA COLA | 8 |
| COCA COLA | 10 |
| COCA COLA | 0 |

AGUA CHICA PERSONAL

Cuenta

Platos

| NOMBRE | PRECIO | CANTIDAD | ID |
|-------------------|--------|----------|----|
| CHICHARRON ... 35 | | 1 | 1 |

Bebidas

| NOMBRE | PRECIO | CANTIDAD | ID |
|-----------------|--------|----------|----|
| AGUA PERSONAL 4 | | 1 | 6 |

TOTAL BS: 39.0

Guardar Cancelar

Figura 98 Registrar Nuevo Pedido

Datos
 Hora Pedido: 03:59:16 Fecha de Pedido: 2010-11-29 Estado: SIN CANCELAR

Generar Pedido

Tipos de platos disponibles

| NOMBRE | PRECIO |
|----------------------|--------|
| CHICHARRON DE SURUBI | 35 |
| SABALO A LA PARRILLA | 28 |
| SABALO FRITO C/L | 25 |
| SABALO FRITO S/L | 22 |
| SURUBI A LA PARRILLA | 37 |

Tipos de bebidas gaseosas y vinos disponibles

| NOMBRE | PRECIO |
|----------------|--------|
| AGUA PERSONAL | 4 |
| CERVEZA PACEÑA | 12 |
| COCA COLA | 5 |
| COCA COLA | 8 |
| COCA COLA | 10 |
| COCA COLA | 0 |

Cuenta

Platos

| NOMBRE | PRECIO | CANTIDAD | ID |
|-------------------|--------|----------|----|
| CHICHARRON ... 35 | | 1 | 1 |

Bebidas

| NOMBRE | PRECIO | CANTIDAD | ID |
|-----------------|--------|----------|----|
| AGUA PERSONAL 4 | | 1 | 6 |
| CERVEZA PAC... | 12 | 1 | 8 |

TOTAL BS: 51.0

Registrar Pedido Cancelar Pedido

Figura 99 Modificar Pedido

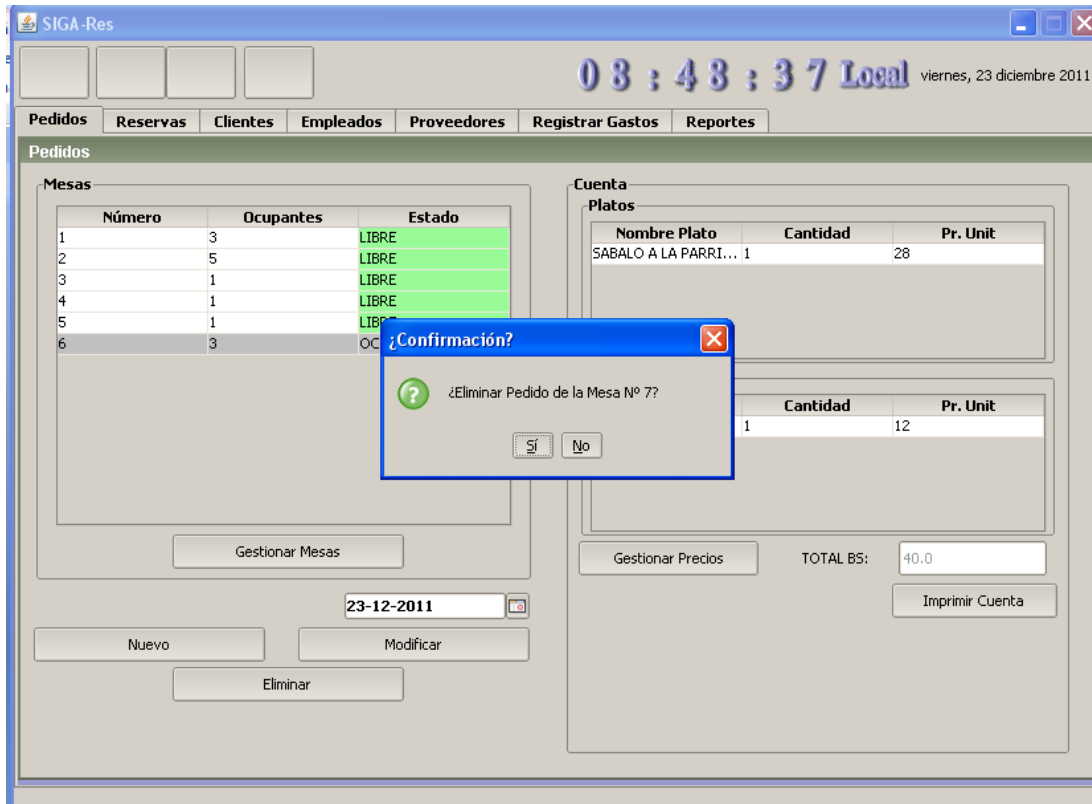
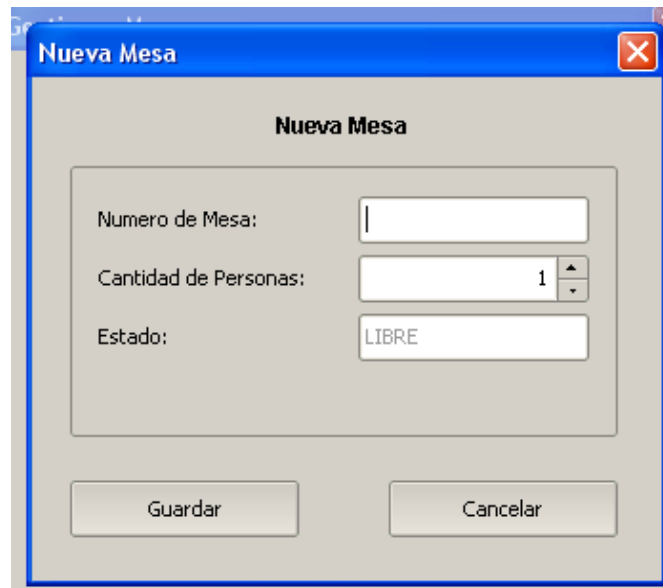


Figura 100 Eliminar Pedido




Figura 101 Gestionar Mesas



The image shows a dialog box titled "Nueva Mesa" with a blue header and a close button (X) in the top right corner. The main content area is titled "Nueva Mesa" and contains three input fields: "Numero de Mesa:" with an empty text box, "Cantidad de Personas:" with a spinner box set to "1", and "Estado:" with a text box containing "LIBRE". At the bottom, there are two buttons: "Guardar" and "Cancelar".

Figura 102 Registrar Nueva Mesa



The image shows a dialog box titled "Modificar Mesa" with a blue header and a close button (X) in the top right corner. The main content area is titled "Modificar Mesa" and contains three input fields: "Numero de Mesa:" with a text box containing "4", "Cantidad de Personas:" with a spinner box set to "1", and "Estado:" with a text box containing "LIBRE". At the bottom, there are two buttons: "Modificar" and "Cancelar".

Figura 103 Modificar Mesa

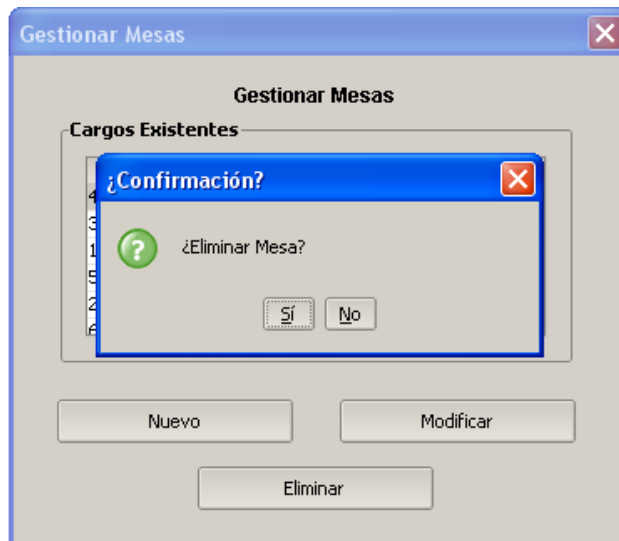


Figura 104 Eliminar Mesa

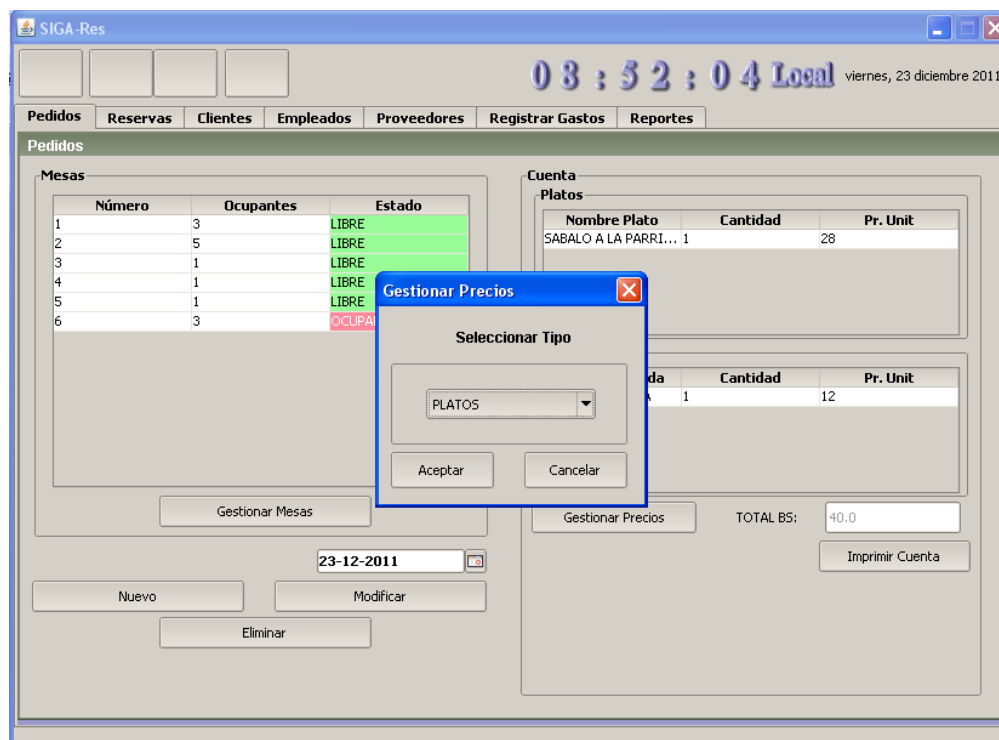


Figura 105 Seleccionar Tipo

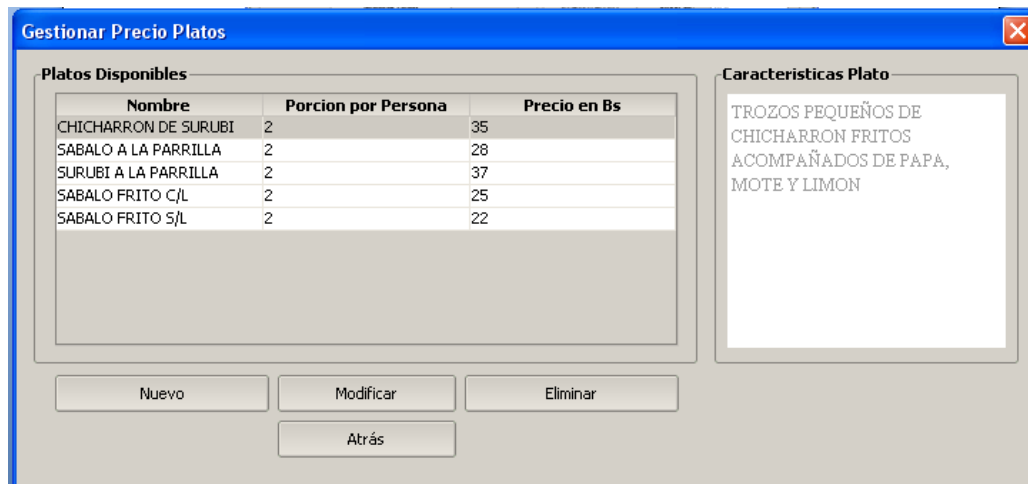


Figura 106 Gestionar Precios Platos



Figura 107 Registrar Nuevo Plato



Figura 108 Modificar Plato

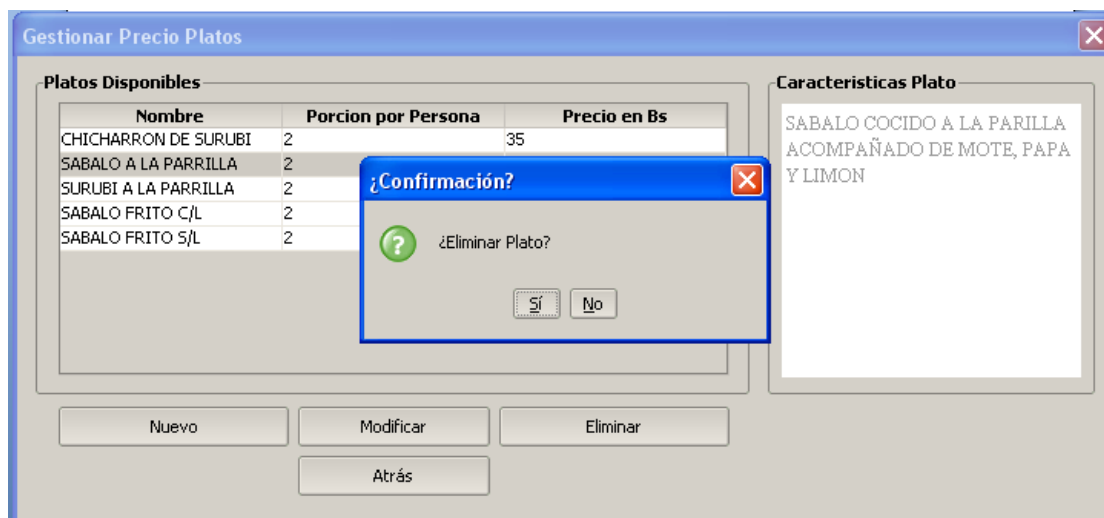


Figura 109 Eliminar Plato

Gestionar Precio Bebidas

Bebidas Disponibles

| Nombre | Cantidad de CC | Precio en Bs |
|----------------------|----------------|--------------|
| COCA COLA | 750 | 5 |
| VINO BLANCO ARANJUEZ | 800 | 20 |
| VINO TINTO KHOLBERG | 800 | 22 |
| COCA COLA | 1000 | 8 |
| SODA SIFON | 450 | 8 |
| AGUA PERSONAL | 600 | 4 |
| COCA COLA | 1500 | 10 |
| CERVEZA PACEÑA | 750 | 12 |

Características Bebida

BEBIDA GASEOSA NEGRA
PERSONAL

Nueva Modificar Eliminar

Atrás

Figura 110 Gestionar Precio Bebidas

Nueva Bebida

Datos de Bebida

Nombre Bebida:

Centimetros Cubicos:

Precio Unitario:

Características de la Bebida

Guardar Cancelar

Figura 111 Registrar Nueva Bebida

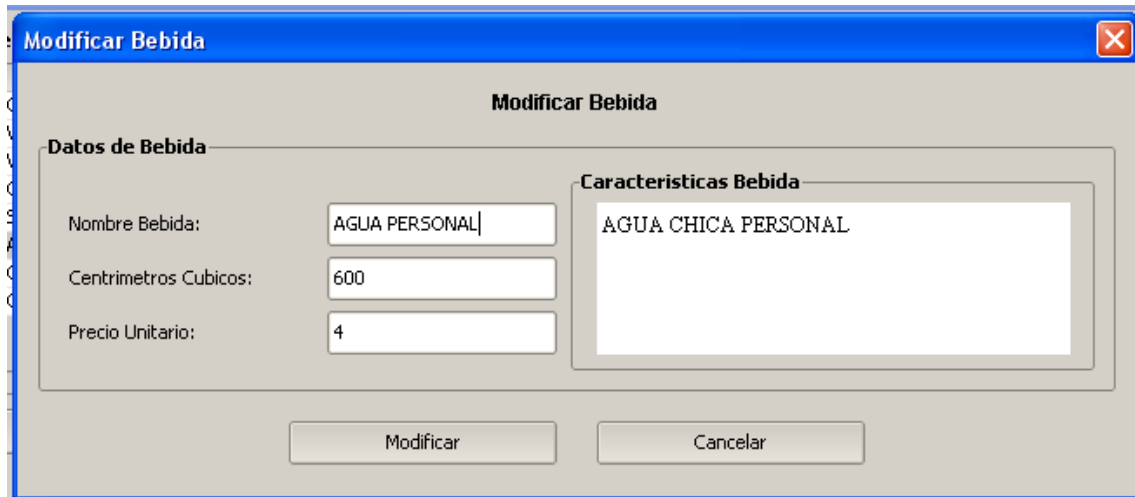


Figura 112 Modificar Bebida

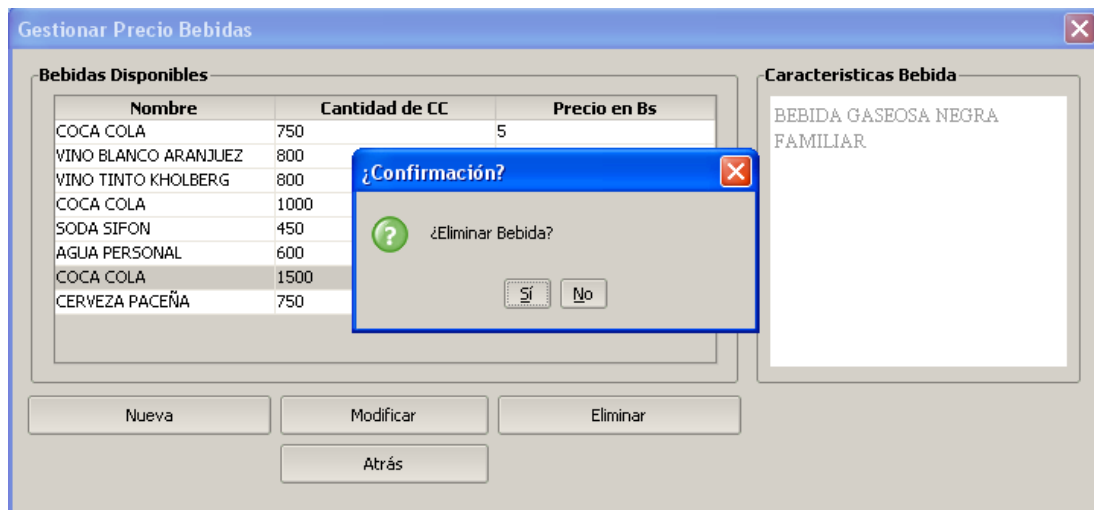


Figura 113 Eliminar Bebida

Verificar Cliente

Verificar Cliente

Seleccionar Cliente

Cliente:

| AP | AM | NOMBRE | CI |
|--------|----------|--------------|---------|
| AÑAZGO | AÑAZGO | MARISOL | 145745 |
| MAXI | MAXI | GERARDO | 123548 |
| PATIÑO | ARELLANO | MARIA ISABEL | 23132 |
| PEREZ | RIZOTI | JORGE | 5658965 |
| PRIETO | JAUREGUI | MARTA | 1455847 |
| TOLAY | GEREZ | ARACELI | 1234567 |

Figura 114 Verificar Cliente

SIGA-Res 08 : 59 : 53 Local viernes, 23 diciembre 2011

Reservas

Reservaciones Hoy Todas

| Nro | Mesa | Fecha R... | Hora Res | Hora Lim | Estado |
|-----|------|-------------|----------|----------|--------|
| 21 | 1 | 2011-12-... | 01:42:29 | 01:42:29 | PASADO |
| 20 | 1 | 2011-12-... | 00:42:58 | 00:42:58 | PASADO |
| 24 | 1 | 2011-12-... | 23:28:01 | 23:28:01 | PASADO |
| 23 | 2 | 2011-12-... | 01:51:26 | 01:51:26 | PASADO |

Reservacion hecha por

Nro de CI:

Nombre del Cliente:

Cuenta

Platos

| Nombre Plato | Cantidad | Pr. Unit |
|--------------------|----------|----------|
| CHICHARRON DE S... | 1 | 35 |

Bebidas

| Nombre Bebida | Cantidad | Pr. Unit |
|----------------|----------|----------|
| CERVEZA PACEÑA | 1 | 12 |

TOTAL BS:

Figura 115 Gestionar Reservas

Nueva Reserva

Datos
 Hora Reserva: 09:00:39 Fecha de Reserva: 23-12-2011 Estado: SIN CANCELAR
 Número Mesa: 1 Ocupantes: 3

Generar Reserva

Tipos de platos disponibles

| NOMBRE | PRECIO |
|----------------------|--------|
| CHICHARRON DE SURUBI | 35 |
| SABALO A LA PARRILLA | 28 |
| SABALO FRITO C/L | 25 |
| SABALO FRITO S/L | 22 |
| SURUBI A LA PARRILLA | 37 |

Tipos de bebidas gaseosas y vinos disponibles

| NOMBRE | PRECIO |
|----------------|--------|
| AGUA PERSONAL | 4 |
| CERVEZA PACEÑA | 12 |
| COCA COLA | 5 |
| COCA COLA | 8 |
| COCA COLA | 10 |

Cuenta

Platos

Bebidas

TOTAL BS:

Guardar Cancelar

Figura 116 Registrar Nueva Reserva

Modificar Reserva

Datos
 Hora Pedido: 01:51:26 Fecha de Pedido: 22-12-2011 Estado: SIN CANCELAR
 Número Mesa: 1 Ocupantes: 3

Generar Reserva

Tipos de platos disponibles

| NOMBRE | PRECIO |
|----------------------|--------|
| CHICHARRON DE SURUBI | 35 |
| SABALO A LA PARRILLA | 28 |
| SABALO FRITO C/L | 25 |
| SABALO FRITO S/L | 22 |
| SURUBI A LA PARRILLA | 37 |

Tipos de bebidas gaseosas y vinos disponibles

| NOMBRE | PRECIO |
|----------------|--------|
| AGUA PERSONAL | 4 |
| CERVEZA PACEÑA | 12 |
| COCA COLA | 5 |
| COCA COLA | 8 |
| COCA COLA | 10 |

Cuenta

Platos

| NOMBRE | PRECIO | CANTIDAD | ID |
|----------------|--------|----------|----|
| CHICHARRON ... | 35 | 1 | 1 |

Bebidas

| NOMBRE | PRECIO | CANTIDAD | ID |
|----------------|--------|----------|----|
| CERVEZA PAC... | 12 | 1 | 8 |

TOTAL BS: 47.0

Modificar Cancelar

Figura 117 Modificar Reserva

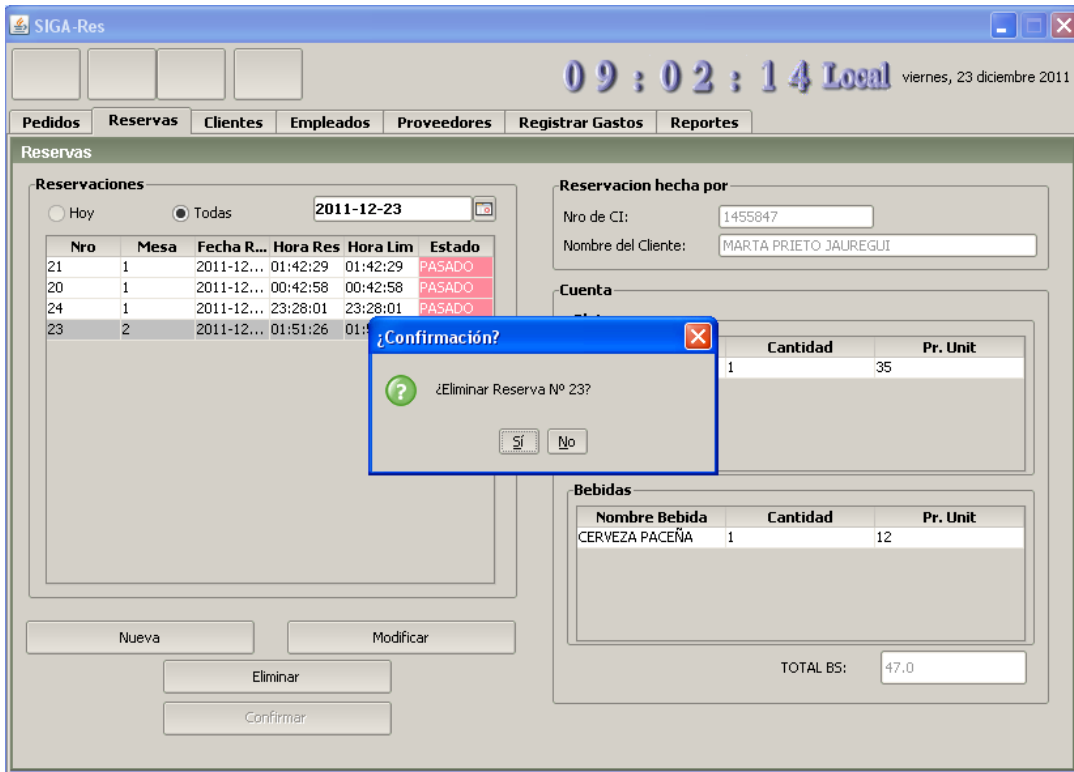


Figura 118 Eliminar Reserva

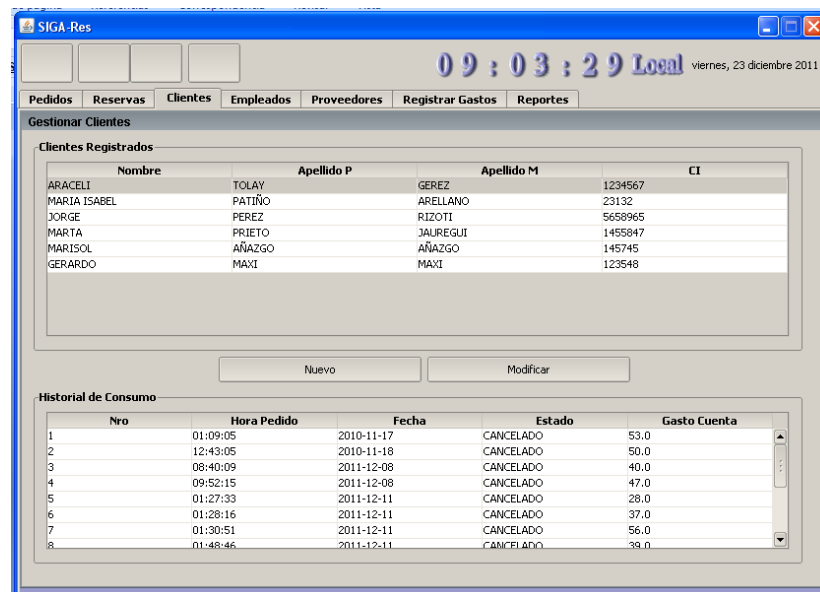


Figura 119 Gestionar Clientes

Nuevo Cliente

Nº Documento:

Nombres:

Ap. Paterno:

Ap. Materno:

Sexo

MASCULINO FEMENINO

Guardar Cancelar

Figura 120 Registrar Nuevo Cliente

Modificar Cliente

Nº Documento:

Nombres:

Ap. Paterno:

Ap. Materno:

Sexo

MASCULINO FEMENINO

Modificar Cancelar

Figura 121 Modificar Cliente

SIGA-Res 09:05:27 Local viernes, 23 diciembre 2011

Pedidos Reservas Clientes **Empleados** Proveedores Registrar Gastos Reportes

Gestionar Empleados

Empleados Registrados

| Nombre | Apellido P. | Apellido M. | Cargo |
|-----------------|-------------|-------------|-----------|
| NAHUEL NELSON | TOLAY | GEREZ | SUPERUSER |
| ROBERTO CLAUDIO | TOLAY | TOLAY | SUPERUSER |

Usuario

Crear Usuario

Modificar Usuario

Gestionar Cargos

Nuevo Modificar Eliminar

Datos Personales de Empleado

Fecha de Nacimiento: 1987-09-29

Teléfono: 72995311

Fecha de Registro:

Dirección

Calle: ALTO DE LA ALIANZA

Barrio: SAN GERONIMO

Número: 279

Figura 122 Gestionar Empleados

Nuevo Empleado

Nuevo Empleado

Datos del Empleado

C.I.:

Nombres:

Ap. Paterno:

Ap. Materno:

Teléfono:

Carga: SUPERUSER

Sexo

MASCULINO FEMENINO

Dirección

Zona:

Calle:

Número:

Fecha de Nacimiento

Mes Día Año

12 23 2011

Guardar Cancelar

Figura 123 Registrar Nuevo Empleado

The image shows a software window titled "Modificar Empleado" with a close button in the top right corner. The window contains a form with the following sections and fields:

- Datos del Empleado:**
 - C.I.: 7136544
 - Nombres: NAHUEL NELSON
 - Ap. Paterno: TOLAY
 - Ap. Materno: GEREZ
 - Teléfono: 72995311
 - Cargo: SUPERUSER (dropdown menu)
 - Sexo:
 - MASCULINO
 - FEMENINO
- Dirección:**
 - Zona: 1
 - Calle: SAN GERONIMO
 - Número: ALTO DE LA ALIANZA
- Fecha de Nacimiento:**
 - Mes: 09
 - Día: 29
 - Año: 1987

At the bottom of the form, there are two buttons: "Modificar" and "Cancelar".

Figura 124 Modificar Empleado

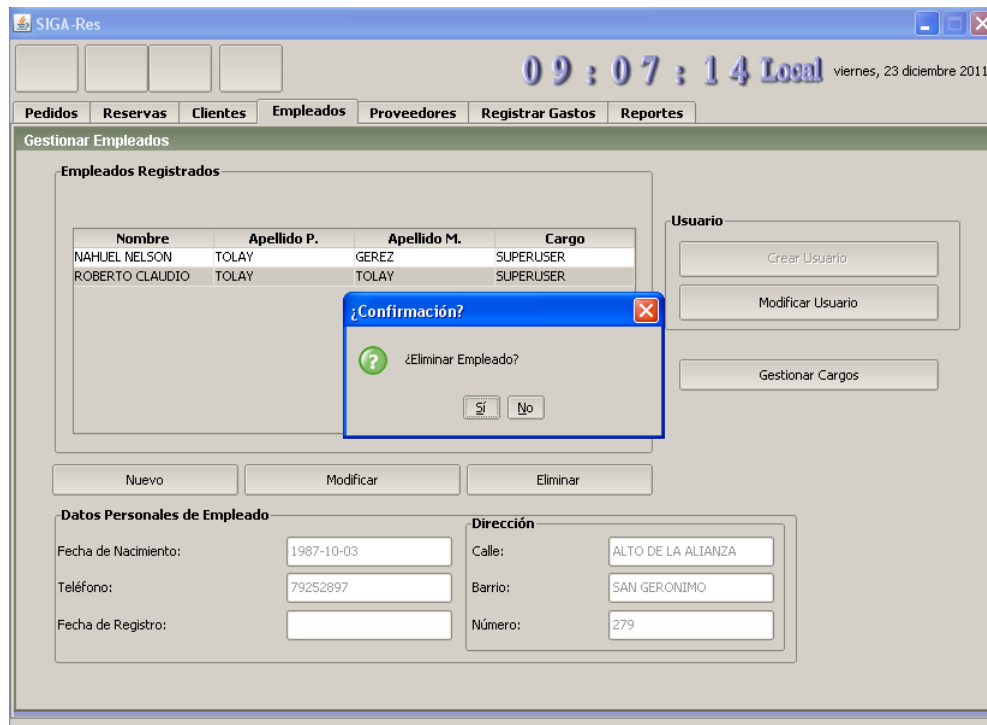


Figura 125 Eliminar Empleado

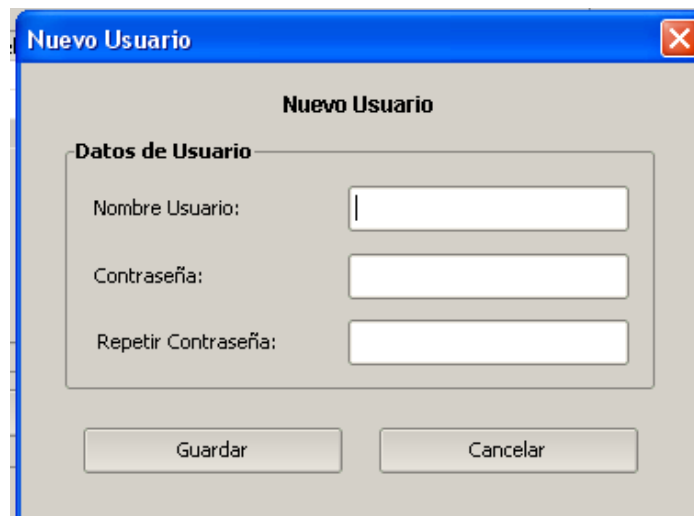


Figura 126 Crear Nuevo Usuario

Modificar Usuario

Datos de Usuario

Nombre de Usuario:

Contraseña Actual:

Nueva Contraseña:

Repetir Contraseña:

Estado:

Figura 127 Modificar Usuario

Gestionar Cargos

Cargos Existentes

| Cargo | Permiso |
|---------------|---------|
| SUPERUSER | 1 |
| ADMINISTRADOR | 1 |

Figura 128 Gestionar Cargos

Nuevo Cargo

Nombre de Cargo:

Descripcion:

Permisos:

Salario:

Guardar Cancelar

Figura 129 Registrar Nuevo Cargo

Modificar Cargo

Nombre de Cargo:

Descripcion:

Permisos:

Salario:

Modificar Cancelar

Figura 130 Modificar Cargo

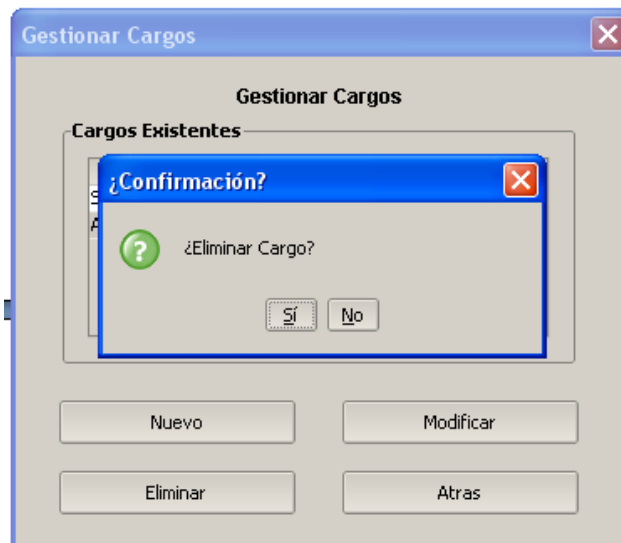


Figura 131 Eliminar Cargo

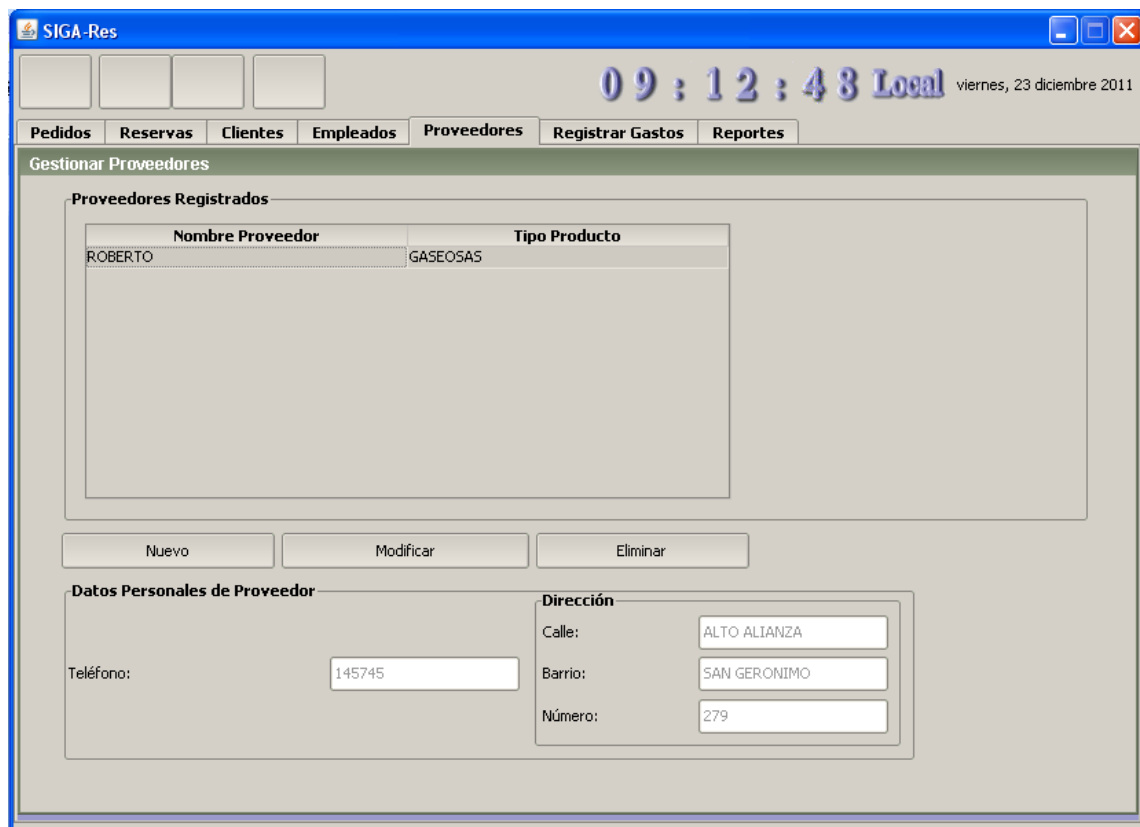


Figura 132 Gestionar Proveedores

Nuevo Proveedor

Datos del Proveedor

Nombre Proveedor:

Tipo Productos:

Teléfono:

Dirección de Proveedor

| Zona | Calle | Número |
|----------------------|----------------------|----------------------|
| <input type="text"/> | <input type="text"/> | <input type="text"/> |

Guardar Cancelar

Figura 133 Registrar Nuevo Proveedor

Modificar Proveedor

Datos del Proveedor

Nombre Proveedor:

Tipo Productos:

Teléfono:

Dirección de Proveedor

| Zona | Calle | Número |
|---|---|----------------------------------|
| <input type="text" value="SAN GERONIMO"/> | <input type="text" value="ALTO ALIANZA"/> | <input type="text" value="279"/> |

Modificar Cancelar

Figura 134 Modificar Proveedor

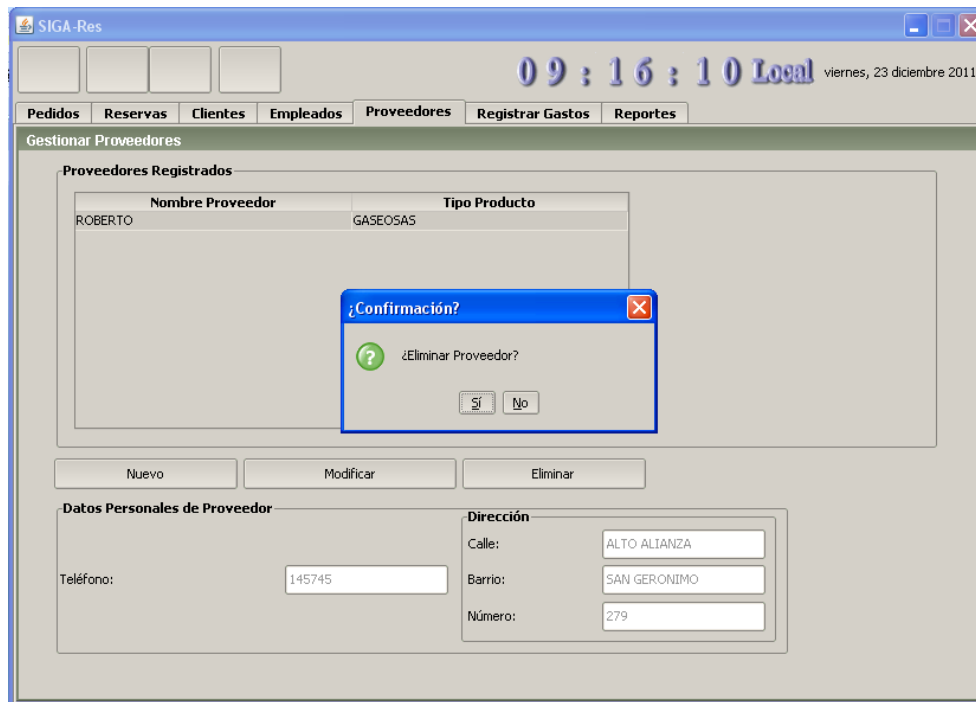


Figura 135 Eliminar Proveedor

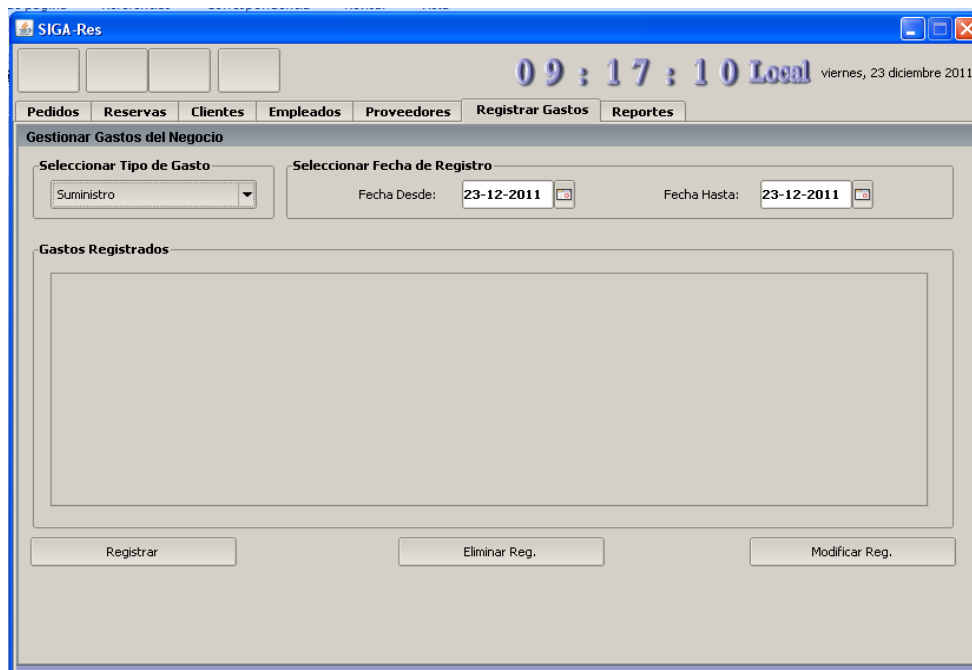


Figura 136 Gestionar Gastos del Negocio

Figura 137 Registrar Nuevo Suministro

| Suministro | Nombre | Precio U. | Importe T. |
|---------------|-----------|-----------|------------|
| MATERIA PRIMA | ACEITE | 0 | 50.0 |
| BEBIDAS | COCA COLA | 0 | 100.0 |
| MATERIA PRIMA | PESCADO | 0 | 400.0 |

Figura 138 Eliminar Suministro

Nuevo Servicio

Tipo Servicio:

Importe Total:

Seleccionar Fecha de Pago

Guardar Cancelar

Figura 139 Registrar Nuevo Servicio

SIGA-Res

09:22:16 Local viernes, 23 diciembre 2011

Pedidos Reservas Clientes Empleados Proveedores Registrar Gastos Reportes

Gestionar Gastos del Negocio

Seleccionar Tipo de Gasto: Servicio

Seleccionar Fecha de Registro: Fecha Desde: 01-12-2011 Fecha Hasta: 31-12-2011

| Servicio | Importe Total |
|----------|---------------|
| LUZ | 1,7 |
| GAS | 5 |
| TELEFONO | 0 |
| AGUA | 1,7 |
| LUZ | 0 |

¿Confirmación? ¿Eliminar Servicio?

Si No

Registrar Eliminar Reg. Modificar Reg.

Figura 140 Eliminar Servicio

II.1.6. Medios de Verificación

Tarija, 10 de Abril 2011

Señor

Nahuel Nelson Tolay Gerez

Presente.-

Con toda la Atención:

A tiempo de Saludarle y desearle éxito en la carrera emprendida por usted en el campo de la informática, quiero agradecerle por la implementación del Sistema de Gestión Administrativa para Restaurantes denominado Siga-Res en forma exitosa.

Con este motivo tan importante quiero reiterar a usted mi agradecimiento y felicitarlo por el arduo trabajo realizado.

Muy atentamente.

Sr. Roberto Claudio Tolay

Gerente propietario del Restaurante "La Carreta"

Indicador de nivel de eficiencia

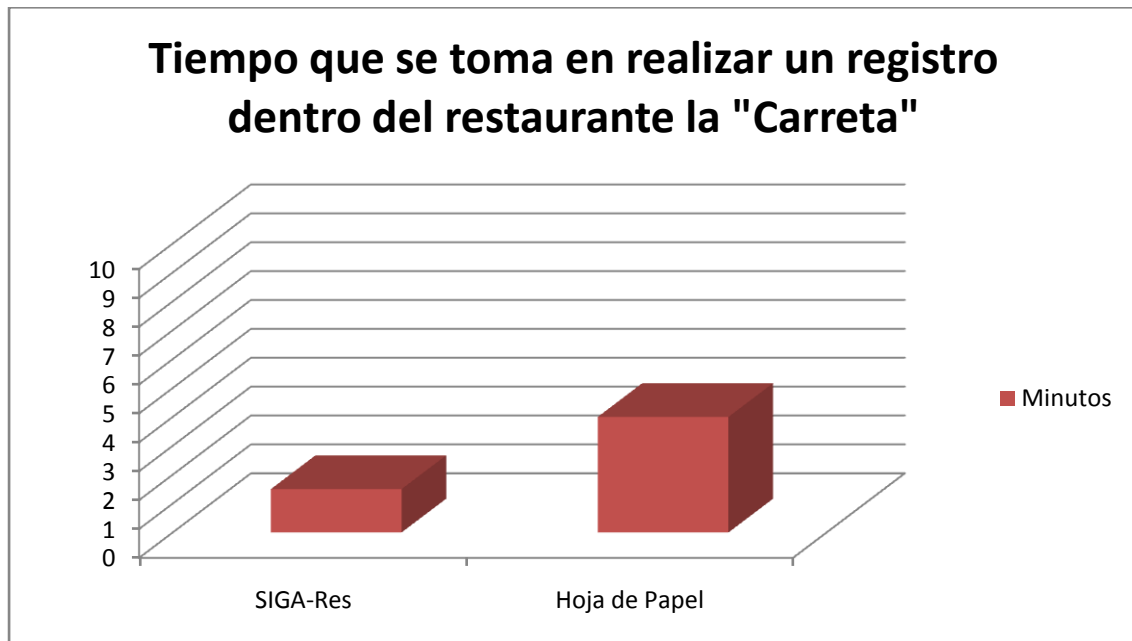


Figura 141. Indicador de eficiencia respecto al tiempo

Porcentaje eficiencia-tiempo

$$\text{eficiencia-tiempo} = 100 - ((1,5 / 4) * 100)$$

$$\text{eficiencia-tiempo} = 100 - 37,5$$

$$\text{eficiencia-tiempo} = \mathbf{62,5 \%}$$

Una vez obtenido el porcentaje de eficiencia-tiempo al realizar el cálculo de la realización de registros podemos ver que el nivel de eficiencia con respecto al tiempo dio como resultado aproximadamente en un 62,5 %.

Este indicador permite verificar que tras la implementación del Sistema SIGA-Res, se comprueba de acuerdo a declaraciones de los usuarios del Sistema que el porcentaje erróneo de registros de información se reduce gracias a la facilidad de uso del Sistema en el tratamiento de los datos.

II.2. Capacitación a Usuarios del Sistema

II.2.1 Lineamientos Generales

II.2.1. Marco Teórico

II.2.1.1. Capacitación a Usuarios del Sistema

Es enseñar a los usuarios que se relacionan u operan en un proceso de implantación. La Responsabilidad de esta capacitación de los Usuarios primarios y secundarios es del Analista, desde el personal de captura de datos hasta aquellos que toman las decisiones sin usar una Computadora.

No se debe incluir a personas de diferentes niveles de habilidad e intereses de trabajo; debido a que si en una Empresa existen trabajadores inexpertos no se pueden incluir en la misma sección de los expertos ya que ambos grupos quedarán perdidos.

II.2.1.2. TIC's (Tecnologías de Información y Comunicación)

Las nuevas tecnologías pueden emplearse en el sistema educativo de tres maneras distintas: como objeto de aprendizaje, como medio para aprender y como apoyo al aprendizaje.

En el estado actual de cosas es normal considerar las nuevas tecnologías como objeto de aprendizaje en sí mismo. Permite que las personas se familiaricen con el ordenador y adquieran las competencias necesarias para hacer del mismo un instrumento útil a lo largo de los estudios, en el mundo del trabajo o en la formación continua cuando sean adultos.

Se consideran que las tecnologías son utilizadas como un medio de aprendizaje cuando es una herramienta al servicio de la formación a distancia, no presencial y del auto aprendizaje o son ejercicios de repetición, cursos en línea a través de Internet, de videoconferencia, cdroms, programas de simulación o de ejercicios, etc. Este procedimiento se enmarca dentro de la enseñanza tradicional como complemento o enriquecimiento de los contenidos presentados.

Pero donde las nuevas tecnologías encuentran su verdadero sitio en la enseñanza es como apoyo al aprendizaje. Las tecnologías así entendidas se hayan pedagógicamente integradas en el proceso de aprendizaje, tienen su sitio en el aula, responden a unas necesidades de formación más proactivas y son empleadas de forma cotidiana. La integración pedagógica de las tecnologías difiere de la formación en las tecnologías y se enmarca en una perspectiva de formación continua y de evolución personal y profesional como un saber aprender.

La búsqueda y el tratamiento de la información inherente a estos objetivos de formación constituyen la piedra angular de tales estrategias y representan actualmente uno de los componentes de base para una utilización eficaz y clara de Internet ya sea en el medio escolar como en la vida privada. Para cada uno de estos elementos mencionados, las nuevas tecnologías, sobre todos las situadas en red, constituyen una fuente que permite variar las formas de hacer para atender a los resultados deseados. Entre los instrumentos más utilizados en el contexto escolar destacamos: tratamiento de textos, hojas de cálculo, bases de datos o de información, programas didácticos, de simulación y de ejercicios, cdroms, presentaciones electrónicas, editores de páginas HTML, programas de autoría, foros de debate, la cámara digital, la videoconferencia, etc. Entre las actividades a desarrollar mencionamos: correspondencia escolar, búsqueda de documentación, producción de un periódico de clase o de centro, realización de proyectos como web-quest u otros, intercambios con clases de otras ciudades o países, etc.

Podrán utilizarse las nuevas tecnologías, pero se seguirá inmerso en la pedagogía tradicional si no se ha variado la postura de que el profesor tiene la respuesta y se pide al alumno que la reproduzca. En una sociedad en la que la información ocupa un lugar tan importante es preciso cambiar de pedagogía y considerar que el alumno inteligente es el que sabe hacer preguntas y es capaz de decir cómo se responde a esas cuestiones.

La integración de las tecnologías así entendidas sabe pasar de estrategias de enseñanza a estrategias de aprendizaje.

II.2.2 Propuesta

Una vez concluido el diseño, desarrollo e implementación del Sistema se deberá proceder a la capacitación a los usuarios del mismo, ya que estos no contarán con la suficiente experiencia para realizar un manejo apropiado del mismo, esta capacitación es el segundo componente del Proyecto dando culminación al mismo, es fundamental que este componente se realice con éxito asegurando que el usuario pueda sacar el máximo provecho del Sistema y comprobar que el Sistema entregado esta acorde a los requerimientos especificados.

II.2.2.1 Objetivos de la Propuesta

II.2.2.1.1 Objetivo General

Realizar la capacitación de usuarios del Sistema para que los mismo logren conseguir los objetivos del Proyecto sacando el máximo provecho a las funcionalidades del Sistema.

II.2.2.1.2 Objetivos Específicos

- Emplear una metodología adecuada para la capacitación.
- Preparar el material que se utilizará en la capacitación.
- Elaborar un plan de clases para distribuir de forma correcta el tiempo que será empleado para la capacitación.
- Hacer uso de las TIC's para complementar la capacitación.

II.2.2.3.2. Metodología para realizar la Capacitación

Para llevar a cabo la capacitación a usuarios finales una vez que la implementación del Sistema termina se procede a tomar en cuenta lo siguiente:

1. Establecimiento de Objetivos Mensurables.- Los objetivos del entrenamiento para todo el personal deben ser indicados claramente, para permitir que los capacitados sepan lo que se espera de ellos. Además, los objetivos permiten la evaluación de la capacitación cuando ha terminado. Los operadores deben saber cosas básicas, tales como el encendido de la máquina, qué hacer cuando suceden errores comunes, búsqueda de fallas básicas y poder solucionar el problema.
2. Uso de Métodos de capacitación adecuados.- Cada usuario y operador necesitará una capacitación ligeramente diferente. Hasta cierto punto, sus trabajos determinan lo que necesitan saber, y su personalidad, experiencia y conocimientos de fondo determinan cómo aprender mejor. Algunos usuarios aprenden mejor viendo, otros oyendo y otros haciendo. Debido a que, por lo general, no es posible personalizar la capacitación para un individuo, frecuentemente la mejor manera de proceder es con una combinación de los métodos. De esta forma se llega a la mayoría de los usuarios por medio de un método u otro.
3. Empleo de materiales de capacitación comprensibles.- Al planear la capacitación de los usuarios, los analistas de sistemas deben darse cuenta de la importancia de materiales de capacitación bien preparados. Estos incluyen manuales de capacitación, costos de capacitación, en donde a los usuarios les es asignado trabajo por medio de un caso que incorpora la mayoría de las interacciones comúnmente encontradas con el sistema. La mayoría del software en paquete proporciona tutoriales en línea para ilustrar las funciones básicas.

Los materiales de capacitación deben estar escritos con claridad, deben tener buenos índices, estar escritos para la audiencia adecuada con un mínimo de vocabulario especial y disponible para cualquiera que los necesite.

II.2.2.3.3. Beneficios para el Cliente

Los principales beneficios serán que el usuario estará capacitado y en condiciones de hacer el uso correcto del Sistema, por consiguiente al hacer un buen uso de este se podrá mejorar la administración del Restaurante la Carreta optimizando su economía previniendo pérdidas significativas.

II.2.2.3.4. Beneficios Directos

Los beneficios directos recaen sobre el Gerente y el personal que será capacitado:

- Gerente Propietario
- Empleados del Restaurante

II.2.2.3.5. Beneficios Indirectos

Los beneficios indirectos recaen sobre terceros que no estén involucrados de forma directa con el Gerente Propietario.

- Empleados que no son usuarios del sistema
- Proveedores
- Clientes y entidades que requieran de los servicios de catering a futuro

II.2.3.3. Organización

| Nro. Clase | Contenido | Duración | Medios de Enseñanza - aprendizaje |
|--------------|---------------------------------------|----------|-----------------------------------|
| • Lección 1: | Instalación de la plataforma Java 6.0 | 20 min | Diapositivas |

| | | | |
|---------------------|---|--------|--------------|
| • Lección 2: | Instalación del Sistema Gestor de la Base de Datos PostgreSQL 8.3 | 15 min | Diapositivas |
| • Lección 3: | Instalación del Sistema de gestión para el registro de transacciones de compra y venta, “SIGA-Res”. | 15 min | Diapositivas |
| • Lección 4: | Creación y restauración de la Base de Datos del Sistema | 30 min | Diapositivas |
| • Lección 5: | Acceso al sistema “SIGA-Res” | 20 min | Diapositivas |
| • Lección 6: | Creación de una cuenta de usuario | 15 min | Diapositivas |
| • Lección 7: | Manejo de los módulos del sistema | 80 min | Diapositivas |
| • Lección 8: | Generación de Reportes | 25 min | Diapositivas |

Materiales Usados en la Capacitación

| | |
|---------------------------------------|--|
| Proyectora Epson PowerLite 77C |  |
| PC Escritorio |  |

II.2.4. Medios de Verificación

Tarija, 10 de Abril 2011

Señor

Nahuel Nelson Tolay Gerez

Presente.-

Con toda la Atención:

A tiempo de Saludarle muy cordialmente me dirijo a usted para agradecerle por la capacitación sobre el Sistema de Gestión para Restaurantes SIGA-Res, realizado en nuestras instalaciones el 10 de marzo del 2011, de manera exitosa.

Muy atentamente.

Sr. Roberto Claudio Tolay

Gerente propietario del Restaurante "La Carreta"

CAPÍTULO III – CONCLUSIONES Y RECOMENDACIONES

III.1. Conclusiones

Por medio de la realización del presente documento se cumplió con los objetivos y componentes planificados del proyecto, orientados a solucionar problemas que generaba complicaciones dentro de la administración del restaurante la "Carreta", problemas referidos a las tareas cotidianas de almacenamiento, empleados, proveedores, en especial de pedidos de cuentas, compras y ventas realizadas; todas las actividades que dependen de estas acciones, se tornaron en un trabajo que exigía un gran agotamiento físico, de tiempo, y con alto riesgo de pérdida de información primordial.

- Gracias al proyecto, el restaurante la "Carreta" cuenta con un moderno Sistema de Gestión dirigido a mejorar dificultades en el ámbito transaccional, que gracias a la notable colaboración de los usuarios, expresando necesidades y exigencias que requerían, fueron tomadas muy en cuenta a fin de solucionar sus problemas en la etapa de análisis y diseño.
- Tras la implementación del sistema se expuso temas sobre nuevas tecnologías de información y comunicación, que ayudaron al entendimiento de conceptos actuales, tips sobre mantenimiento de Pcs, familiarizando al usuario con el software y hardware del computador.
- El modelo lógico del proyecto se basó en la estructura y arquitectura de la información manejada en el restaurante la "Carreta", reflejando a través de sus entidades y relaciones, la estructura de la información en los registros. Para las fases de análisis y diseño del sistema se utilizó el lenguaje unificado del modelado (UML).
- Los temas brindados durante la jornada de capacitación coadyuvaron a mejorar y actualizar los conocimientos de los usuarios del sistema, que participaron de la capacitación brindada.

- Cumplidas las expectativas y el desarrollo e implantación de este proyecto se demostró la eficiencia obtenida con la ejecución del proyecto denotada por los usuarios, cumpliendo así a cabalidad con el propósito planteado.

III.2. Recomendaciones

- Con respecto a la parte administrativa, las funciones del coordinador del restaurante (el gerente) confirman que no tiene los conocimientos adecuados para manejarlo; ya que no organiza, ni controla, ni dirige la operación, lo que corrobora que para que un negocio sea rentable no es suficiente con cocinar bien y dar un buen servicio.
- Para que se obtenga mayor rentabilidad es necesario mejorar la administración desde el punto de vista de calidad, cantidad, precio, atención, limpieza, control transaccional y mantenimiento, por consecuencia habrá mejor ganancia. El costo que pueda representar será mínimo con la gran cantidad de beneficios que se pueden obtener.
- Y con una mejor rentabilidad el restaurante podrá destinar recursos para mantener y mejorar los servicios e instalaciones.

Con respecto a los componentes se sugiere:

- Poblar la base de datos del sistema lo más antes posible para que de esta manera puedan observarse los beneficios que ofrece el sistema en cuanto a seguridad y a velocidad de acceso a la información.
- Se recomienda usarlo en un equipo de cómputo que cumpla con los requisitos de hardware y software especificados en los manuales de instalación y de usuario.
- Se recomienda llevar a cabo las políticas de seguridad comunicadas en las jornadas de capacitación para que se eviten problemas en el normal

funcionamiento del equipo de cómputo, del sistema y la información que maneja, y demás aplicaciones.

- Ante alguna falla en la funcionalidad del sistema, se recomienda recurrir a la solución de problemas contemplado en el manual de usuario. Si el problema no es subsanado contactar a los desarrolladores del proyecto.
- Para futuros usuarios del sistema se recomienda a la dirigencia exigir conocimientos básicos de computación y otorgar los manuales correspondientes.