

1.- INTRODUCCIÓN

Con el paso del tiempo la tecnología avanza, las empresas se sienten en la necesidad de adquirirse tecnología para el mejoramiento de sus sistemas y a las vez sus procedimientos, con el fin de garantizar un eficaz funcionamiento y así obtener una adaptación paralela de condiciones con las empresas líderes del mercado. En tiempos pasados, se discutían las dificultades planteadas por los sistemas de información se refería a las personas, no a la tecnología de la información, luego se ha intentado despertar el interés ante las posibilidades que ella brinda y que se puede resolver la mayoría de los problemas técnicos, pero para quienes comprenden que la innovación basa en la tecnología es al fin un proceso social, cuando muchos años más tarde, se examinan las predicciones hechas con referencias al cambio tecnológico, resulta impresionante su penetración y la incorporación de la tecnología de la información a la vida cotidiana pues al fin, su influencia en el hombre no se ha hecho esperar y el ser humano a tomado amplia conciencia de la profunda alteración que la utilización no sólo de computador sino del conjunto de técnicas de tratamiento de la información derivada de su uso ha llevado a construir los retos del futuro.

El propósito de este documento se centra en recoger, analizar y definir las necesidades principales y características del sistema de control de ventas y stock de almacén. El documento se evoca principalmente funcionalidad requerida para su uso con respecto a sus usuarios finales, pues esto se basa en las diferentes interfaces que adoptaría el ya mencionado sistema, así como los detalles especificados para satisfacción de requerimientos del usuario final.

1.2.- ANTECEDENTES

La empresa que sirve de base para realizar el presente trabajo, es una empresa, dedicada a la compra y venta de equipos de tecnología; empresa en vías de desarrollo y expansión. Sus ventas son por medio de venta directa al cliente teniendo como producto líder el equipo de computación.

Tiene como inicio en su actividad el año 2007 ubicada en la ciudad de Bermejo cuenta con 6 empleados entre vendedores, técnicos y gerencia. Su principal actividad a la que se dedica es a la compra y venta de equipos de tecnología como ser:

- ✚ Computadoras y accesorios
- ✚ Equipos de Sonido
- ✚ Impresoras
- ✚ Celulares
- ✚ Televisores

Se ha podido observar que durante todos los procesos que se efectúan en la empresa SITEC, es deficiente puesto que, cuando un cliente solicita su mercancía el empleado debe ir a verificar si la mercancía existe o no, lo que ocasiona el descontento en los mismos al esperar mucho tiempo para saber su existencia, no existe un control de registro clientes, proveedores ni de empleados ya que su sistema es totalmente manual y para no acumular más documentos no lo hacen. Al momento de cotejarla mercancía existente para posterior efectúo de pedidos es muy tedioso, ya que tiene que contar una por una la mercancía y se pierde mucho tiempo valioso que se invertir en otras. De continuar funcionando de esta manera las operaciones y sus actividades no podrán ser aligeradas, al contrario se obstaculizaran más el trabajo debido al aumento de la clientela y la cantidad de información que será aún mayor. Debido a esto se propone sistematizar las actividades de compra de mercancía de dicha empresa y así mejorar su eficacia al trabajar de manera más rápida y segura

Realizando una investigación sobre sistemas que realicen el Control de Inventarios se pudo ver que existen Sistemas Contables que cuentan entre sus módulos control de inventarios:

1.2.1 Orión Stock

Es un Sistema de Inventarios y cuenta con múltiples características que la hacen vital para una organización que desee controlar sus activos para venta. Está desarrollada para el ambiente Windows que usted posea.

1.2.2 Sistema Contable Mónica

Es el programa de computador ideal para su negocio. Le permite realizar las facturas, controlar el inventario, realizar listas de precios, tener un archivo de clientes, proveedores, manejar las cuentas por cobrar, las cuentas corrientes, y la contabilidad básica de su empresa.

Módulos que Comprende Mónica

Facturas

Le permite crear facturas, modificarlas y eliminarlas. UD. puede especificar sus propios números de factura, ordenar las facturas por diferentes criterios (fechas, clientes, vendedor, etc.). Así mismo puede realizar devoluciones sobre el inventario. Las facturas actualizan el inventario apenas son creadas.

También puede realizar facturas en lotes (ideal por ejemplo cuando se tiene que facturar al mismo grupo de clientes al final del mes por un mismo concepto).

Puede obtener sus totales de ventas por fechas, clientes, reporte de impuestos, etc.

Inventarios

Controla los productos en el almacén: crea, modifica elimina productos (el código del producto puede ser hasta de 14 caracteres), puede hacer su lista de precios, precios-costo, para un grupo de productos, cambiar de precios de uno o varios productos a la vez.

Así mismo puede enviar cotizaciones a sus proveedores y luego convertirlos en órdenes de compra (está enlazado con los módulos de cuentas por pagar). Otro aspecto importante es el kardex que puede llevar de su inventario, es fácil de actualizar y manejar.

Cuentas por Cobrar

Cuando UD. vende al crédito, Mónica le permite controlar las facturas vencidas, abiertas, pagos parciales, etc.

Vd. puede tener un listado de clientes impagos, facturas canceladas, vencimientos de las próximas facturas. Asimismo obtener el estado de cuenta de cualquier cliente en cualquier momento.

Cuentas por Pagar

Si su empresa quiere llevar un registro de compras de sus proveedores, entonces este módulo es para UD. ya que le permite saber a qué proveedores les debe, cuándo se vencen las facturas, etc. Calendario de pagos para el proveedor. Estado de pagos para cada proveedor.

1.3.- PLANTEAMIENTO DEL PROBLEMA

De acuerdo al análisis de requerimientos, se ha podido observar que durante todos los procesos que se efectúan en la empresa SITEC, es deficiente puesto que, cuando un cliente solicita su mercancía el empleado debe ir a verificar si la mercancía existe o no, lo que ocasiona el descontento en los mismos al esperar mucho tiempo para saber su existencia, no existe un control de registro clientes, proveedores ni de empleados ya que su control es totalmente manual y para no acumular más documentos no lo hacen diariamente perdiendo el control adecuado de los datos de la empresa.

1.4.- OBJETIVOS

1.4.1.- OBJETIVO GENERAL

Realizar el Análisis y Diseño de un Sistema de Gestión que permita controlar las compras, ventas e inventarios de una manera rápida, organizada y sencilla beneficiando tanto al personal como a los clientes de la empresa, mediante el uso de una metodología adecuada.

1.4.2.- OBJETIVOS ESPECIFICOS

- ✚ Recolectar información sobre los procesos implicados en la gestión mediante entrevistas, cuestionarios y observación directa para hacer la ingeniería de requerimientos.
- ✚ Realizar el análisis y diseño utilizando la metodología RUP, apoyado con el lenguaje UML y herramientas CASE como Rational Rose.
- ✚ Diseñar una Base de Datos donde se almacenan la información de manera optima.
- ✚ Diseñar las entradas y salida de datos con pantallas intuitivas que permitan al usuario una rápida gestión en el Sistema.
- ✚ Realizar automáticamente copias de seguridad periódicas del Registro de la Base de Datos.
- ✚ Diseñar un sistema seguro que permita definir claramente los permisos de cada usuario.
- ✚ Diseñar un modulo de reportes de compras, ventas e inventarios para que el proceso organizativo sea más eficaz y de esta manera lograr un buen manejo de las transacciones que incurren en la empresa.

1.5.- JUSTIFICACIÓN

1.5.1.- JUSTIFICACION SOCIAL

El sistema de gestión a desarrollarse permitirá que una Empresa tenga una imagen proyectada hacia la sociedad en su conjunto y la atención de sus usuarios. Consiguiendo mejorar los servicios que esta presta e impulsará a la utilización de nuevas tecnologías, aumentando la competitividad con las empresas del mismo rubro.

1.5.2.- JUSTIFICACION TECNICA

El Sistema de gestión de compras, ventas e inventarios para su funcionamiento no precisa de un Equipo de Computación de última generación, haciendo que el sistema sea fácil de utilizar, de mantener y de actualizar.

1.5.3.- JUSTIFICACION ACADEMICA

El Desarrollo de presente Trabajo nos va a permitir Aplicar Conocimientos adquiridos durante los años de nuestra Educación Universitaria como ser (análisis, diseño), como también Aplicar la Investigación que iremos adquiriendo durante el Desarrollo del presente Trabajo, aplicando la Metodología RUP porque es una Metodología orientado a objetos y así lograr un Sistema de gestión que permita controlar las compras, ventas e inventarios , también nos servirá para poder desarrollarnos como futuros Profesionales. El presente Trabajo de Grado puede servir de apoyo y guía para los futuros Profesionales.

1.6.- ALCANCE

1. **Control de inventario:** permitirá el registro de información sobre los productos existentes y de los que se adquieran posteriormente. También se hará el registro de los proveedores y clientes.
2. **Control de usuarios y seguridad:** se realizara un registro de los usuarios que podrán acceder al sistema, para ello se asignara un nombre de usuario, contraseña y rol a cada usuario del mismo. Una vez que el usuario acceda al sistema solo podrá realizar aquellas operaciones que le son permitidas.
3. **Reportes:** el sistema permitirá realizar reportes de todos los productos, ventas, compras que se efectúen en la empresa.
4. **Consultas y búsquedas:** el usuario podrá realizar consultas y búsquedas de los productos de acuerdo a su descripción.

1.7.- LIMITACIONES

- ✚ No realizará la Contabilidad.
- ✚ No realizará Controles al Personal de la Empresa.
- ✚ El sistema no registrara ingresos de dinero.
- ✚ No incluye actualizaciones por efectos inflacionarios.
- ✚ No contempla el manejo de productos fraccionados, es decir, maneja el producto en su integridad.
- ✚ No se diseñara la elaboración de planes de pago de deudas con los proveedores.

2. MARCO TEORICO DE REFERENCIA

2.1 Sistemas de control de inventarios

Los inventarios son bienes tangibles (acumulación de bienes o materiales) que se tienen para la venta en el curso ordinario del negocio o para uso interno del mismo con el fin de que se utilicen en el proceso de operación de las diferentes unidades de trabajo.¹

Se utilizan para satisfacer la demanda sea esta interna o externa. Para tener una adecuada administración sobre los mismos, surge la necesidad de crear un sistema que permita llevar control de los productos que se almacenan para satisfacer la demanda ya sea interna o externa.

Un propósito fundamental de un sistema de control de inventarios, consiste en administrar de manera eficiente los recursos de la empresa, destinados a mantener un inventario, administrando eficientemente los flujos que el inventario presente, a fin de mantener un nivel de inventario adecuado, que se adapte a la demanda real que el producto o productos presente.

El principal reto al que se enfrentan los administradores de inventarios, no consiste en reducir al máximo los inventarios para abatir los costos que implica la existencia de los mismos, ni en mantener un exceso de inventario a fin de satisfacer la demanda, sino que se debe buscar el equilibrio entre estos dos puntos que permita operar con la mayor eficiencia posible.²

¹ Krajewski, Lee J. Administración de operaciones, estrategia y análisis. Quinta Edición. Editorial PEARSON Educación, México 2000. p. 555.

² Krajewski, Lee J. Administración de operaciones, estrategia y análisis. Quinta Edición. Editorial PEARSON Educación, México 2000. p. 544.

Equilibrio entre mantener inventarios altos o inventarios bajos.

Tener un nivel de existencia de mercaderías demasiado alto, significa que se tendrá un costoso inventario. Inventario que tendrá que ser vendido posteriormente con pérdidas, lo que afectaría la rentabilidad de la empresa. Por el contrario el no tener un nivel de existencia suficiente, significa la pérdida de potenciales ventas y conllevaría a tener clientes insatisfechos.

¿Por qué inventarios bajos?

Existen diversas razones para mantener un inventario en un nivel bajo, entre las cuales se puede mencionar, reducir: el costo de operación, intereses sobre capital, costos de almacenamiento y manejo, pago de seguros, mermas del inventario.

¿Por qué inventarios altos?

Entre las principales razones para mantener un nivel de existencia alto en el inventario se puede mencionar: mejorar el servicio al cliente, reducir costos de realizar pedidos, reducir el costo de transporte de la mercadería, comprar a mejores precios.

Es por ello que surge la pregunta ¿qué tan alto o qué tan bajo debe de ser un inventario? Las dos situaciones afectan la rentabilidad de la empresa y se debe buscar un equilibrio a este respecto. Los modelos económicos de inventarios tratados más adelante, dan una respuesta a este aspecto, pero la decisión final está a cargo de la administración de la empresa, la cual debe basarse en resultados científicos y la experiencia que se ha adquirido en el tiempo.

Con esto surge la necesidad de establecer por parte de la gerencia o administración de la empresa, un sistema que permita tener una administración eficiente de inventarios, con el principal objetivo de mantener una rentabilidad adecuada en operación.

2.1.1 Qué es un sistema de control de inventarios

Un sistema de control de inventarios, es aquel que nos permite llevar control de los productos que se manejan en una empresa u organización, considerando los elementos esenciales de un sistema; entradas, proceso, salidas y retroalimentación.

También en el proceso se valorizará la mercadería para determinar su costo y así establecer un precio de venta de acuerdo al margen de utilidad deseado por la empresa. La salida del sistema serán los productos para la venta con un precio establecido para su comercialización.

El objetivo del sistema de control de inventario es de establecer la mejor política de inventarios que permita tener los productos necesarios en la empresa que optimice los recursos de la empresa, permitiendo realizar las operaciones de manera eficiente.

2.1.1.1 Métodos de control de inventarios para costos

Los métodos de control de inventarios para costos permiten tener registro de las mercaderías que entran y salen y al mismo tiempo valorizarlos de acuerdo al sistema que se aplique.

Es de hacer notar que estos sistemas únicamente valorizan los productos de acuerdo al costo con que se adquirió los mismos, para determinar el precio de venta, hay que agregarle un porcentaje de utilidad, el cual lo determina la administración de la empresa.

2.1.1.1.1 Método PEPS

Primero en entrar, primero en salir; este método como su nombre lo indica, consiste en valorizar los productos que salen a los precios de las primeras entradas, hasta que los productos con ese precio se agoten. Dicho en términos simples, los productos se valorizan con los precios que entran a la bodega de almacenamiento.

2.1.1.1.2 Método de costos promedios

Este método de valuación permite valorizar la mercadería que se almacena, obteniendo el precio promedio de todos los productos que sean de un mismo tipo. Se determina el valor del costo total de los mismos dentro del número de existencia de ese producto.

2.1.1.1.3 Método estándar

Este método de valuación se aplica cuando las variaciones de los precios son mínimas. Consiste en fijar un precio a los productos, que se aproxime a las variaciones. Se aplica en forma constante a la valorización de las mercaderías que salen de la bodega.

Para la elaboración de este trabajo, se tomará en cuenta el método de valuación de Costo de Promedio Ponderados porque este es uno de los métodos que se adecua al tipo de movimiento de la empresa.

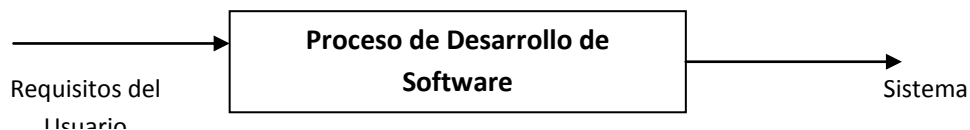
Los pronósticos de ventas se basan en la información que la empresa u organización tiene, de los resultados de ventas de períodos anteriores, a esos resultados se le denomina datos históricos, estos datos reflejan el comportamiento que ha tenido la demanda en un determinado marco de tiempo.

2.2 RUP (EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE)

El Proceso Unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los Requisitos de los Requisitos.

Requisitos de un usuario en un sistema software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitudes y diferentes tamaños de proyecto.

Figura 1: Representación gráfica del proceso de desarrollo de software



El Proceso Unificado está basado en componentes software interconectados a través de interfaces bien definidas.

El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. De hecho, UML es una parte esencial del Proceso Unificado, sus desarrollos fueron paralelos.

No obstante, los verdaderos aspectos definitorios del Proceso Unificado se resumen en tres frases claves, y esto es lo que hace único al Proceso Unificado:

- Dirigidos por casos de uso,
- Centrado en la arquitectura,
- E iterativo e incremental.

Dirigido por Casos de Uso: Los casos de uso han sido adoptados casi universalmente para la captura de requisitos de sistemas software en general y de sistemas basados en componentes en particular, pero los casos de uso son mucho más que una herramienta para capturar requisitos. Dirigen el proceso de desarrollo en su totalidad. Los casos de uso son la entrada fundamental cuando se identifican y

especifican clases, subsistemas e interfaces, cuando se identifican y especifican casos de prueba, y cuando se planifican las iteraciones del desarrollo y la integración del sistema. Para cada iteración, nos guían a través del conjunto completo de flujo de trabajo, desde la captura de requisitos, pasando por el análisis, diseño e implementación, hasta la prueba, enlazando estos diferentes flujos de trabajo.

Centrado en la Arquitectura: el desarrollo de sistemas software requiere el pensar y registrar las ideas de forma comprensible, no sólo para los siguientes desarrolladores, sino para otros tipos de usuarios. Los arquitectos la desarrollan iterando repetidas veces durante la fase de inicio y elaboración. De hecho el primer objetivo de la fase de elaboración es establecer una arquitectura sólida de forma que sea una arquitectura base ejecutable. Como resultado, se entra en la fase de construcción con unos fundamentos sólidos para construir el resto del sistema.

Iterativo e Incremental: la tercera clave proporciona la estrategia para desarrollar un producto software en pasos pequeños manejables:

- Planificar un poco.
- Especificar, diseñar e implementar un poco.
- Integrar, probar, y ejecutar un poco cada iteración.

Si estamos contentos con un paso, continuamos con el siguiente. Entre cada paso, obtenemos retroalimentación que nos permite ajustar nuestros objetivos para el siguiente paso, después se da el siguiente paso y después otro. Cuando se dan todos los pasos que se habían planificado, tenemos un producto desarrollado que podemos distribuir a los clientes y usuarios.

La Vida del Proceso Unificado

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo concluye con una versión del producto para los clientes.

Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición, cada fase se subdivide a su vez en iteraciones.

El Producto

Cada ciclo produce una nueva versión del sistema, y cada versión es un producto preparado para su entrega.

Para llevar a cabo un ciclo de manera eficiente, los desarrolladores necesitan todas las representaciones del producto software:

- Un modelo de casos de uso, con todos los casos de uso y su relación con los usuarios.
- Un modelo de análisis, con dos propósitos: refinar los casos de uso con más detalle y establecer la asignación inicial de funcionalidad del sistema a un conjunto de objetos que proporcionan el comportamiento.
- El modelo de diseño que define: a) la estructura estática del sistema en la forma de subsistemas, clases e interfaces y b) los casos de uso reflejados como colaboraciones entre los subsistemas, clases e interfaces.
- Un modelo de implementación, que incluye componentes (que representan el código fuente) y la correspondencia de las clases con los componentes.
- Un modelo de despliegue o distribución, que define los nodos físicos (ordenadores) y la correspondencia de los componentes con esos nodos.
- Un modelo de prueba, que describe los casos de prueba que verifican los casos de uso.
- Y, por supuesto, una representación de la arquitectura.

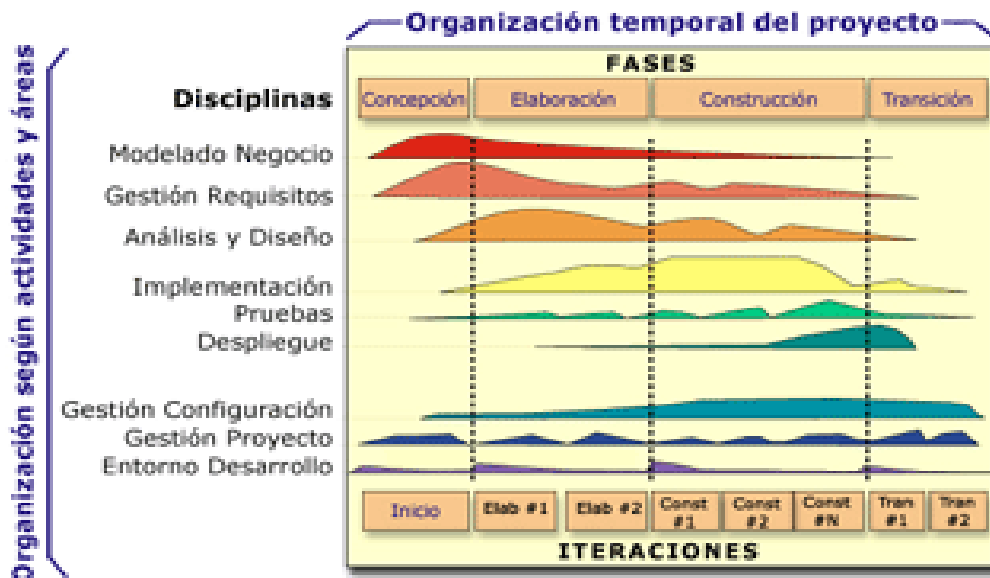
Todos estos modelos están relacionados. Junto, representan al sistema como un todo.³

Fases Dentro de un Ciclo

³ (BOOCH GRADY, JACOBSON IVAR, RUMBAUGH JAMES, EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE, Pág. 4 – 45).

Cada ciclo de desarrolla a lo largo del tiempo. Este tiempo, a su vez, se divide en cuatro fases, como se muestra en la figura.

Figura 2: Fases dentro de un ciclo del Proceso Unificado de Rational (RUP)



Proceso Unificado de Rational (RUP). Relación entre las fases del proyecto y las disciplinas técnicas que se abordan en cada una de las iteraciones.

Según la metodología RUP el ciclo de vida de un proyecto se divide en las siguientes fases:

- Fase de Concepción.
- Fase de elaboración.
- Fase de Construcción.
- Fase de Transmisión.

Fase de Concepción

- **Definición del Alcance del Sistema.** Se captura el contexto y los más importantes requisitos y restricciones hasta el nivel en el cual se puedan derivar criterios de aceptación para el producto final.

- **Planificación y Estimación de Costes** para el desarrollo del proyecto. Esto incluye la elaboración de un plan de gestión y de un plan de trabajos que contempla el uso de recursos.
- **Sintetizar una Arquitectura Candidata.** El plan de trabajos se basa en una arquitectura candidata, que es la mejor solución posible en base las restricciones de costes, plazo de desarrollo y tecnología disponibles. En algunos casos se puede requerir la construcción de un prototipo inicial, pero sólo al efecto de demostrar la viabilidad de la solución.
- **Selección de la Organización** del proyecto, incluyendo metodología y herramientas a utilizar para el control y ejecución del mismo.

Fase de Elaboración

- **Definir y Validar una Arquitectura Estable,** que será la base para el resto del desarrollo.
- **Refinamiento de la Visión del Sistema,** basándose en nueva información obtenida durante esta fase, se establece una sólida comprensión de los casos de uso más críticos que definen las decisiones arquitectónicas y de planificación.
- **Creación de los Planes de Desarrollo Detallados para las Iteraciones de la Fase de Construcción.**
- **Se Refina la Arquitectura y se Seleccionan los Componentes.** Se evalúan y seleccionan los componentes más apropiados para su integración en la arquitectura seleccionada.

Fase de Construcción

- **Gestión de los Recursos,** optimización y control de los procesos de construcción del software.
- **Se completa el Desarrollo de los Componentes** y/o subsistemas, probándolos contra un conjunto definido de criterios aprobados al inicio del proyecto.

Fase de Transición

- Ejecución de los planes de implantación.
- Se finalizan los manuales de usuario y mantenimiento.
- Pruebas del sistema en el entorno de explotación
- Creación de una reléase del sistema.
- Validación del sistema por los usuarios.
- Ajuste fino del sistema según la validación con el usuario.
- Se facilita la transición del sistema al personal de mantenimiento
- Se pone el producto a disposición del usuario final.⁴

2.2.1 INTRODUCCION A UML

Es un lenguaje gráfico para Visualizar, Especificar y Documentar cada una de las partes que comprende el Desarrollo de Software.

2.2.1.1 Modelo de Clases

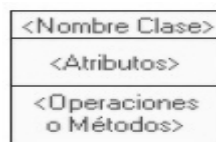
Sirve para visualizar las relaciones entre las clases que involucran el sistema las cuales pueden ser asociativas, de herencia, de uso y de contenimiento.

Elementos

- **Clase**

Es la unidad básica donde se encapsula toda la información de un Objeto. A través de ella podemos modelar.

Figura 3: Representación gráfica de una Clase



Atributos

⁴ (TORRES LETELIER PATRICIO, Desarrollo de Software Orientado a Objeto usando UML, Valencia – España, Pág. 20 –35).


Los atributos o características de una Clase pueden ser de tres tipos (Públicos, Privados y Protegidos), los que definen el grado de comunicación y visibilidad de ellos con el entorno.


Métodos

Los métodos u operaciones de una clase son la forma en cómo ésta interactúa con su entorno, También pueden ser de tres tipos (Públicos, Privados y Protegidos).

- **Relaciones entre Clases:** Es como se pueden interrelacionar dos o más clases.

La cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

Herencia (Especialización/Generalización):  Indica que una subclase hereda los métodos y atributos especificados por una Súper Clase, por ende la subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Súper Clase.

Agregación:  Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

- **Por Valor (Composición):** Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido está condicionado por el tiempo de vida del que lo incluye.
- **Por Referencia (Agregación):** Es de tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye.

Asociación:  Permite asociar objetos que se colaboran entre sí.

Es decir, que el tiempo de vida de un objeto no depende del otro.

Dependencia o Instanciación (uso):  Su instanciación es dependiente de otro objeto/clase.

Casos Particulares

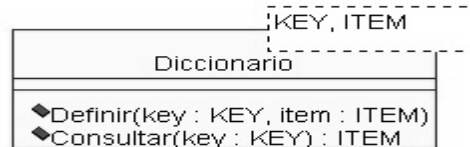
- **Clase Abstracta**

Se denota con el nombre de la Clase y de los métodos con letra "itálica". Esto indica que la clase definida no puede ser instanciada pues posee métodos abstractos (aún no

han sido definidos, es decir, sin implementación). La única forma de utilizarla es definiendo subclases, que implementan los métodos abstractos definidos.

- **Clase Parametrizada**

Figura 4: Representación gráfica de una Clase Parametrizada



Donde una llave o palabra tiene asociado un significado, pero en este caso las llaves y elementos pueden ser genéricos. La genericidad puede venir dada de un Témplate o bien de alguna estructura predefinida (especialización a través de clases).

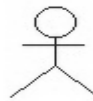
2.2.1.2 Casos de Uso

Representa la forma en cómo un Actor opera con el sistema de desarrollo, además de la forma, tipo y orden en como los elementos interactúan.

Elementos

- **Actor**

Figura 5: Representación gráfica de un actor



Un Actor es un rol que un usuario juega con respecto al Sistema. La palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.


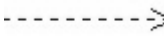

- **Caso de Uso**

Figura 6: Representación gráfica de un caso de uso



Es una Operación/Tarea específica que se realiza tras una orden de algún agente externo, que sea desde una petición de un actor o bien desde la invocación de otro caso de uso.

- **Relaciones**

- **Asociación**  Es la que indica la invocación desde un actor o caso de uso a otra operación.
- **Dependencia o Instanciación**  Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia se crea.
- **Generalización**  Es la que cumple una doble función de dependiendo de su estereotipo, que puede ser de **Uso** o de **Herencia**.

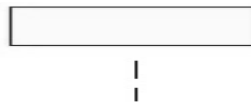
2.2.1.3 Diagrama de Interacción

Es la que representa como un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento.

Elementos

- **Objeto/Actor**

Figura 7: Representación gráfica de un objeto/actor



El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

- **Mensaje a Otro Objeto**

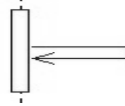
Figura 8: Representación gráfica de un mensaje a otro objeto



Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular.

- **Mensaje al Mismo Objeto**

Figura 9: Representación gráfica de un mensaje al mismo objeto



También es posible visualizar llamadas a métodos desde el mismo objeto.

2.2.1.4 Diagrama de Actividad

Un grafo de actividad es una variante de una maquina de estados, que muestran las actividades de computación implicadas en la ejecución de un cálculo. La vista de Actividad tiene que ver con la especificación del comportamiento como actividades conectadas por flujos de control. Estas vistas contienen grafos de actividades y se representa mediante diagramas de actividades. De forma un tanto libre se agrupa con otras vistas que tiene que ver con el comportamiento para formar las vistas dinámicas.⁵

2.2.1.5 Diagrama de colaboración

Muestra la interacción entre varios objetos y los enlaces que existen entre ellos.

Representa las interacciones entre objetos organizadas alrededor de los objetos y sus vinculaciones. A diferencia de un diagrama de secuencias, un diagrama de colaboraciones muestra las relaciones entre los objetos, no la secuencia en el tiempo en que se producen los mensajes. Los diagramas de secuencias y los diagramas de colaboraciones expresan información similar, pero en una forma diferente.

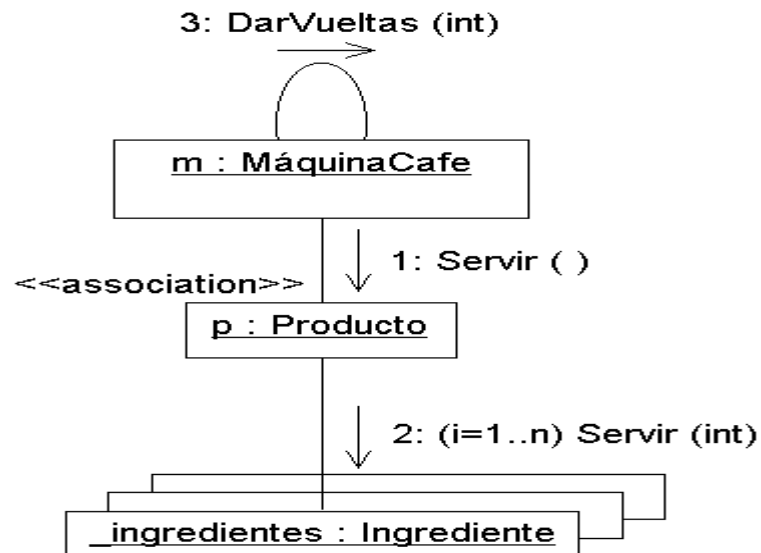
Formando parte de los diagramas de colaboración nos encontramos con objetos, enlaces y mensajes. Un objeto es una instancia de una clase que participa como una

⁵ (FOWLER MARTIN, SCOTT KENDALL, Uml Gota a Gota, 1999, México, Pág. 2 – 40).

interacción, existen objetos simples y complejos. Un objeto es activo si posee un thread o hilo de control y es capaz de iniciar la actividad de control, mientras que un objeto es pasivo si mantiene datos pero no inicia la actividad.

Un enlace es una instancia de una asociación que conecta dos objetos de un diagrama de colaboración. El enlace puede ser reflexivo si conecta a un elemento consigo mismo. La existencia de un enlace entre dos objetos indica que puede existir un intercambio de mensajes entre los objetos conectados.

Los diagramas de interacción indican el flujo de mensajes entre elementos del modelo, el flujo de mensajes representa el envío de un mensaje desde un objeto a otro si entre ellos existe un enlace. Los mensajes que se envían entre objetos pueden ser de distintos tipos, también según como se producen en el tiempo; existen mensajes simples, sincrónicos, balking, timeout y asíncronos. Durante la ejecución de un diagrama de colaboración se crean y destruyen objetos y enlaces.⁶



⁶ <http://www.desic.upv.es/uml>, 2008

2.3 Herramientas CASE (Computer Aided Software Engineering)

La ingeniería de Software asistida por computadora, proporciona un conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida de desarrollo de Sistemas de Información, completamente o en alguna de sus fases. El objetivo de las herramientas CASE es conseguir la generación automática de programas desde una especificación a nivel de diseño, utilizando Ingeniería Directa e Inversa para obtener el producto final que es el Software.

Las herramientas CASE para el análisis y diseño permiten crear modelos de sistemas que se piensan construir. La particularidad de estas herramientas es la evaluación de la calidad del modelo mediante la comprobación de la validez y la consistencia de dichos modelos. Esto permitirá cierto grado de confianza en la representación del análisis, ayudando a eliminar errores que se puedan cometer en esta etapa.

2.3.1 Enterprise Architect (EA)

Enterprise Architect es una herramienta para el plan y construcción de sistemas de software. EA apoya al UML describiendo un idioma visual por el que pueden definirse mapas o modelos de un proyecto.

EA es una herramienta progresiva que cubre todos los aspectos del ciclo de desarrollo de un sistema de software.

Enterprise Architect apoya todo el UML models/diagrams. Usted puede modelar los procesos comerciales, redes, las configuraciones del hardware, los mensajes y más. Estime el tamaño de su esfuerzo de trabajo de proyecto en horas. La captura y requisitos del rastro, recursos, planes de la prueba, defectos y demandas de cambio. Enterprise Architect tiene los rasgos que usted necesita diseñar y manejar para su desarrollo y aplicación.

Características generales de Enterprise Architect

Soporte para UML 2.0

Permite definir patrones personalizados para la reutilización de diseños

Permite generar listado de requisitos y matrices de trazabilidad

Permite generar listados de riesgos

Exporta e importa los diagramas en XML 2.1

Permite generar reportes personalizados en formato HTML y RTF

Permite generar código fuente en varios lenguajes (Java, C#, PHP)

Ingeniería inversa de código fuente en varios lenguajes (Java, C#, PHP)

Importa esquemas de bases de datos

Importa binarios en Net y Java

Soporta control de versiones⁷

2.3.2 Rational Rose

Rational Rose es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML 1.1.

Esta herramienta propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de Software.

Desarrollo Iterativo:

Rational Rose utiliza un proceso de desarrollo iterativo controlado (controlled iterative process development), donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Cada iteración comienza con una primera aproximación del análisis, diseño e implementación para identificar los riesgos del diseño, los cuales se utilizan para conducir la iteración, primero se identifican los riesgos y después se prueba la aplicación para que éstos se hagan mínimos.

Cuando la implementación pasa todas las pruebas que se determinan en el proceso, ésta se revisa y se añaden los elementos modificados al modelo de análisis y diseño.

⁷ Dra. María José Escalona Cuaresma “Características Generales de Enterprise Architect”

Una vez que la actualización del modelo se ha modificado, se realiza la siguiente iteración.

Trabajo en Grupo

También es posible descomponer el modelo en unidades controladas e integrarlas con un sistema para realizar el control de proyectos que permite mantener la integridad de dichas unidades.

Generador de Código

Se puede generar código en distintos lenguajes de programación a partir de un diseño en UML.

Ingeniería Inversa

Rational Rose proporciona mecanismos para realizar la denominada Ingeniería Inversa, es decir, a partir del código de un programa, se puede obtener información sobre su diseño.

2.4 Lenguaje de Programación Visual Basic

Visual Basic es un lenguaje de programación dirigido por eventos, desarrollado por Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y, en cierta medida, también la programación misma.

El lenguaje de programación Visual Basic es uno de los lenguajes de programación que utiliza una interfaz visual es decir que nos permite programar en un entorno gráfico, nos permite realizar un gran número de tareas sin escribir código, simplemente realizando operaciones con el ratón sobre la pantalla de la computadora.

Ventajas

- Posee una curva de aprendizaje muy rápida.
- Integra el diseño e implementación de formularios de Windows.

- Permite usar con facilidad la plataforma de los sistemas Windows, dado que tiene acceso prácticamente total a la API de Windows, incluidas librerías actuales.
- Es uno de los lenguajes de uso más extendido, por lo que resulta fácil encontrar información, documentación y fuentes para los proyectos.
- Fácilmente extensible mediante librerías DLL y componentes ActiveX de otros lenguajes.
- Posibilita añadir soporte para ejecución de scripts, VBScript o JScript, en las aplicaciones mediante Microsoft Script Control.
- Tiene acceso a la API multimedia de DirectX (versiones 7 y 8). También está disponible, de forma no oficial, un componente para trabajar con OpenGL 1.1.7
- Existe una versión, VBA, integrada en las aplicaciones de Microsoft Office, tanto Windows como Mac, que permite programar macros para extender y automatizar funcionalidades en documentos, hojas de cálculo, bases de datos (access).
- Si bien permite desarrollar grandes y complejas aplicaciones, también provee un entorno adecuado para realizar pequeños prototipos rápidos.

Inconvenientes

Las críticas hechas en las ediciones de Visual Basic anteriores a VB.NET son variadas,⁸ se citan entre ellas:

- Problema de versionado asociado con varias librerías runtime DLL, conocido como DLL Hell
- Pobre soporte para programación orientada a objetos⁸
- Incapacidad para crear aplicaciones multihilo, sin tener que recurrir a llamadas de la API de Windows.
- Dependencia de complejas y frágiles entradas de registro COM

⁸ Marc D'Aoust. [«Avoid Writing Tedious, Boring Code](#)

- La capacidad de utilizar controles en un sólo formulario es muy limitada en comparación a otras herramientas. DLL Hell DB, Libro Programando en Visual Basic,2002⁹

2.5 Microsoft Sql Server

Es un sistema de gestión de bases de datos relacionales (SGBD) basada en el lenguaje Transact-SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. Así de tener unas ventajas que a continuación se pueden describir.

Entre sus características figuran: Soporte de transacciones. Escalabilidad, estabilidad y seguridad. Soporta procedimientos almacenados. Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente. Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información. Además permite administrar información de otros servidores de datos Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en su versión 2005 pasa a ser el SQL Express Edition. Microsoft SQL Server constituye la alternativa de Microsoft a otros sistemas gestores de bases de datos como son Oracle, Sybase ASE o MySQL. Es común desarrollar completos proyectos complementando Microsoft SQL Server y Microsoft Access a través de los llamados ADP (Access Data Project). De esta forma se completa una potente base de datos (Microsoft SQL Server) con un entorno de desarrollo cómodo y de alto rendimiento (VBA Access) a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows. Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para varias plataformas de desarrollo, entre ellas .NET. Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas

⁹ http://es.wikipedia.org/wiki/Visual_Basic

Operativos de Microsoft.Paradox (base de datos)Base de datos relacional para entorno MS Windows, anteriormente disponible para MS-DOS y Linux, desarrollada actualmente por Corel e incluida en la suite ofimática WordPerfect Office. En los tiempos del MS-DOS, era una base de datos de bastante éxito, compitiendo con dBase, Clipper y FoxBase. Pasó al control de Borland después de la compra de Ansa Software en 1987.Aunque Borland la portó a Windows, su cuota de mercado es mucho menor que la de Microsoft Access, pero su lenguaje de programación (Objectpal) es Pascal lo que le hace más potente que Access que usa Visual Basic que limita bastante sus prestaciones si se compara con otras bases de datos que usan lenguajes más avanzados.

VENTAJAS

- a. El sistema es barato, y la tendencia a aceptar preferencialmente productos de Microsoft.
- b. Interfaz de acceso OLE DB Y ADO. Aunque se trata de una interfaz universal, SQL Server es una de las primeras bases de datos en soportarla.
- c. Mejor utilización de los recursos de la PC
- d. Menor necesidad de limpieza de las memorias intermedias durante el procesamiento de las transacciones

DESVENTAJAS

- a. Enorme cantidad de RAM que utiliza.
- b. Bloqueo a nivel de página.
- c. Dispositivos con crecimiento manual, un tamaño de página fijo y demasiado pequeño (2048KB).
- d. Una pésima implementación de los tipos de datos variables como varchar.

2.6 Microsoft Access

Es un sistema gestor de bases de datos relacionales (SGBD). Una base de datos suele definirse como un conjunto de información organizada sistemáticamente. En la terminología propia de las bases de datos hay tres conceptos claves dentro de las tablas: campo, registro y dato. Un campo es cada uno de los tipos de datos que se van a usar. Se hace referencia a los campos por su nombre. Un registro está formado por el conjunto de información en particular.

Un dato es la intersección entre un campo y un registro.

VENTAJAS

- a. Obtenerse mejores resultados con la interfaz de usuario de Office Fluent. Incluso sin ser experto se puede hacer seguimiento de la información
- b. Puede modificar un informe con una respuesta visual en tiempo real y guardar varias vistas para audiencias distintas.
- c. Gracias a la detección automática del tipo de datos, su manejo es tan sencillo como el Excel
- d. Crea tablas rápidamente sin preocuparse de la complejidad de las base de datos

DESVENTAJAS

- a. Para bases de datos de gran calibre (en cuanto a volumen de datos o de usuarios) es recomendable usar otros sistemas como MySQL o Microsoft SQL Server, y código VBA (Visual Basic para Aplicaciones).
- b. Entre sus mayores inconvenientes figuran que no es multiplataforma, pues sólo está disponible para sistemas operativos de Microsoft.¹⁰

¹⁰ <http://www.slideshare.net/HernanOkamura/sistemas-de-gestores-de-base-de-datos-13332504>

2.7 Seguridad del Sistema

- Políticas de contraseñas: Son una de las políticas más importantes, ya que por lo general, las contraseñas constituyen la primera y tal vez única manera de autenticación y, por tanto, la única línea de defensa contra ataques. Éstas establecen quién asignará la contraseña, qué longitud debe tener, a qué formato deberá apegarse, cómo será comunicada.
- Políticas de uso adecuado: Especifican lo que se considera un uso adecuado o inadecuado del sistema por parte de los usuarios, así como lo que está permitido y lo que está prohibido dentro del sistema de cómputo.
- Políticas de respaldos: Especifican qué información debe respaldarse, con qué periodicidad, qué medios de respaldo utilizar, cómo deberá ser restaurada la información, dónde deberán almacenarse los respaldos.¹¹

2.7.1 Copias de respaldo y recuperación

El Reglamento establece para los ficheros de datos de nivel alto que deberá conservarse una copia de respaldo y de los procedimientos de recuperación de los datos en un lugar diferente a aquél en que se encuentran los equipos informáticos que los tratan cumpliendo, en todo caso, las medidas de seguridad exigidas.

El almacenamiento y conservación de juegos de copias de seguridad y de los procedimientos de restauración de datos fuera de la ubicación principal (por ejemplo, en cámaras de seguridad de bancos, que nos ofrecen altas medidas de seguridad) nos garantiza la continuidad de la actividad y la disponibilidad de la información ante cualquier incidencia grave o muy grave, sea física o lógica, que afecte a los equipos y servidores centrales (incendios, inundaciones, etc.).¹²

¹¹ <http://www.bibliotecadigital.umsa.bo:8080/rddu/bitstream>

¹² Por: Andrés Veyrat Marqués de Manaca Consulting, S.L. “<http://www.microsoft.com/business/es-es/Content/Paginas/article.aspx?cbcid=326>”

2.7.2 Manejo de cuentas o roles de usuario

Una cuenta de usuario se trata de las credenciales únicas de un usuario en un dominio, ofreciéndole la posibilidad de iniciar sesión en el Dominio para tener acceso a los recursos de la red o de iniciar la sesión local en un equipo para tener acceso a los recursos locales. Cada persona que utilice la red regularmente debe tener una cuenta. Las cuentas de usuario se utilizan para controlar cómo un usuario tiene acceso al Dominio o a un equipo. Por ejemplo, puede limitar el número de horas en las que un usuario puede iniciar una sesión en el dominio, impresoras de red que puede utilizar... Es decir, gracias a las cuentas de usuario el Administrador puede controlar todo lo que un usuario puede hacer en un dominio, a través de las restricciones de su cuenta y la configuración de derechos de usuario.

Como administradores podemos definir el entorno de trabajo de un usuario para restringir o personalizar lo que un usuario ve y tiene disponible cuando inicia una sesión. Por ejemplo, podemos establecer los colores de la pantalla, la configuración del ratón, y que las conexiones de red y de impresoras sean siempre las mismas cuando un usuario inicie una sesión. Para administrar los entornos de trabajo de los usuarios se utilizan normas que regulan qué usuarios pueden utilizar un determinado recurso compartido del sistema, esto incluye archivos, carpetas, impresoras, etc.¹³

¹³ http://fmc.axar.net.es/winnt4svr/administracion/tema_04.htm

2.8 Herramientas para la etapa de prueba

Introducción.

La prueba de software es un conjunto de herramientas, técnicas y métodos que hacen a la excelencia del desempeño de un programa, así como también la mejor publicidad que una empresa dedicada a la producción de software pueda tener. Las técnicas para encontrar problemas en un programa son extensamente variadas y van desde el uso del ingenio por parte del personal de prueba hasta herramientas automatizadas que ayudan a aliviar el peso y el costo de tiempo de esta actividad. Pero de nada serviría conocer todas las técnicas de prueba de software, si un programa carece de documentación, el código es confuso, o no se han seguido pasos para la planificación y desarrollo del software, ya que sería como buscar una aguja en un pajar.

Es por eso que en este trabajo monográfico nos hemos volcado a definir no solo las herramientas, técnicas y métodos de prueba sino que también a todo el trabajo previo de control de calidad en el desarrollo de software, ya que sabemos que mucho mejor que encontrar y solucionar un problema es prevenir que no ocurra.

¿Qué es el control de calidad del software?

El control de calidad del software incluye desde el monitoreo de desarrollo de procesos haciendo respetar los estándares y procedimientos concordados asegurándose un buen seguimiento de programa y la detección y corrección de errores. El control de calidad del software está orientado a la prevención.

¿Qué es prueba de software?

La prueba de software involucra las operaciones del sistema bajo condiciones controladas y evaluando los resultados.

Las condiciones controladas pueden ser normales o anormales. La prueba puede intencionalmente esforzar al programa y producir errores en las respuestas para determinar si los sucesos ocurren cuando no tendrían que ocurrir o cuando los hechos no suceden cuando deberían suceder.

La prueba de software esta detectada a la detección.

La mayoría de las grandes organizaciones asumen la responsabilidad del control de calidad y prueba de software a tal medida que en la producción se incluyen desarrolladores de sistemas (analistas, programadores) y un grupo dedicado a la prueba de software para que estos grupos antes mencionados trabajen en conjunto cumpliendo el control de calidad (prevención) y la prueba de software (detección) logrando una tarea exitosa.

2.8.1 Prueba de caja negra

Esta prueba implica una variada selección de los datos de prueba así como una buena interpretación de los resultados para determinar el nivel de optimización de la funcionalidad del sistema.

Se ha determinado con diferentes estudios estadísticos, que el software no debe ser probado por el creador o grupo de creadores del sistema ya que el extenso conocimiento de la estructura interna del programa limita la variedad de datos probados o el encaminamiento de las pruebas es hacia ciertos rasgos del programa olvidando otras partes del software poco valoradas por su simpleza en la creación.

Según C. Kaner en su libro "Testing Computer Software" de 1993, el aspecto humano es esencial en la prueba de caja negra aplicando factibles sucesos de la vida real a la prueba, errores de tipeo, trabajar en aplicaciones equivocadas creyendo trabajar en la aplicación deseada, etc., pero sucede que los programadores han pasado tanto tiempo en la creación del sistema y al ser la prueba de caja negra una de las más tempranas sus hechos factibles de la vida real están entre el "begin" y el "end" de cada aplicación.

La prueba de caja negra ha hecho que cada organización dedicada al desarrollo de software contemple dentro de ella un departamento especial dedicado a las pruebas.

El principal objetivo es determinar la funcionalidad del software y parte de tratar al programa como si fuera una función matemática, estudiando si las respuestas o salidas son "codominio" de los datos entrantes "dominio". La prueba de caja negra

tiene otras metas, determinar la eficiencia del programa desde el desempeño en el equipo, el tiempo de retardo de las salidas hasta el nivel de recuperación del sistema luego de fallas o caídas sean estas producidas por manejo incorrecto de datos, equipo, o producidas externamente como cortes de energía.

La prueba de caja negra debe cubrir el espectro entero de sucesos factibles, de esto se debe ocupar el departamento de prueba, pero nunca se sabe si se han cubierto todos los casos o gran parte de ellos, no olvidemos que los testers pertenecen a la organización por lo que la prueba de caja negra no termina una vez que salió del laboratorio.

La prueba con intervención del usuario es un pequeño periodo de tiempo en el cual el usuario bajo el asesoramiento de testers, se desenvuelve en el software y examina la operabilidad de los datos que el maneja. El usuario dará la puntada final en la cuestión de datos de prueba. Esta parte de la prueba no podría hacerse sin que el usuario haya tenido previo contacto con los prototipos del sistema, y para los testers una efectiva interacción con herramientas CASE.

2.8.2 Prueba de caja blanca

Para esta prueba se consideran tres importantes puntos.

- I) Conocer el desarrollo interno del programa, determinante en el análisis de coherencia y consistencia del código.
- II) Considerar las reglas predefinidas por cada algoritmo.
- III) Comparar el desarrollo del programa en su código con la documentación pertinente.

La primera parte de esta prueba es el análisis estático.

- Análisis estático Manual

- Inspección: Determina si el código está completo y correcto, como también las especificaciones.
- Walkthrough: Interrelación informal entre testers, creadores y usuarios del sistema.
- Análisis estático Automático
- Verificación estática: Compara los valores generados por el programa con los rangos de valores predefinidos haciendo una descripción del funcionamiento de los procedimientos en términos booleanos determinando los puntos de falla.
- Ejecución simbólica: Hace un seguimiento de la comunicación entre funciones, módulos, aplicaciones, luego de que todas las partes hayan sido verificadas por separado.

La segunda parte de esta es el análisis dinámico. Para esto se cuenta con diferentes tipos de herramientas.

- Análisis de cobertura: Examina las extensiones del código, haciendo una caja blanca por módulo.
- Tráfico: Sigue todos los caminos de comunicación entre módulos guardando los valores de las variables en cada uno de ellos.
- Simulador: Simula partes del sistema para el cual el hardware no está habilitado.
- Sintonía: Testea los recursos utilizados durante la ejecución del programa.
- Prueba de certeza: Examina las construcciones lógicas del programa.¹⁴

Para este proyecto se recomienda la prueba de caja de negra porque es independiente del diseño del software. Además no necesita de conocimientos de programación.

Las pruebas se realizan desde un punto de vista de usuario y no como programador, tomando en cuenta todas las funciones de cómo se ve y su usabilidad que tenga el sistema.

¹⁴ M. Vázquez C. Falcato Editorial Prensa Técnica S. L. España

3.1 Especificaciones de Requerimientos

La especificación de requerimientos está dada a través del método entrevistas y cuestionarios con el gerente de la empresa SITEC (Ver anexo1).

3.2 Análisis de Sistema

En el presente trabajo se desarrollo la metodología RUP modelada en UML con el uso de los diagramas los cuales permitirán desarrollar los procesos de información del sistema. Dichos diagramas se listan a continuación:

- Diagramas de Casos de Uso.
- Diagramas de Secuencia.
- Diagramas de Colaboración
- Diagramas de Actividades.
- Diagramas de Clases.

3.2.1 Identificación de Actores

La identificación de actores en términos generales son usuarios del sistema los cuales interactúan, aportan y reciben información del sistema para coadyuvar a sus tareas cotidianas o necesidades demandadas.

A continuación se detallan los actores o usuarios identificados.

3.2.1.1 Gerente General: Es la persona que necesita estar más informado teniendo un control y seguimiento de las actividades de la empresa SITEC. Sus funciones son:

- ✓ Solicitar reportes de almacén para la toma de decisiones.
- ✓ Aprueba las solicitudes que llegan a la oficina de los clientes en caso de ser instituciones.
- ✓ Realiza el requerimiento a los proveedores de acuerdo al reporte de almacén.

3.2.1.2 Encargado de Almacén

- ✓ Realiza el Inventario de todos los productos dentro del Almacén.
- ✓ Elabora reportes de inventario.

3.2.1.3 Encargado de Ventas

- ✓ Elabora comprobantes de Salida y entrada de los productos requeridos.
- ✓ Realiza el registro de clientes
- ✓ Realiza las copias de seguridad periódicamente.
- ✓ Realiza los reportes de Existencia de productos.

3.2.1.4 Cliente

- ✓ Realiza la compra de los productos.

3.2.1.5 Proveedor

- ✓ Encargadas de proveer los productos al almacén de la empresa.
- ✓ Emite facturas o notas de venta.

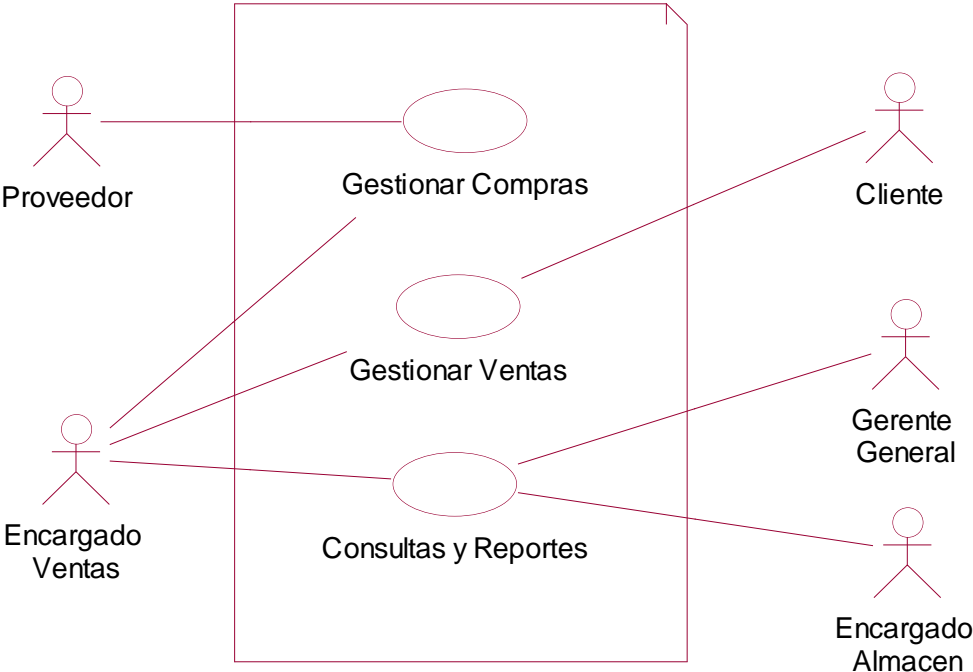
3.2.2 Identificación de Casos de Uso

El propósito de un caso de uso es definir una pieza de comportamiento coherente. La definición de un caso de uso incluye el comportamiento que implica: líneas principales, variaciones de comportamiento normal y todas las condiciones excepcionales, que pueden ocurrir con tal comportamiento, junto a la respuesta deseada.

Un caso de uso es una descripción lógica de una parte funcional del sistema que muestra las distintas operaciones y sus relaciones con los actores.

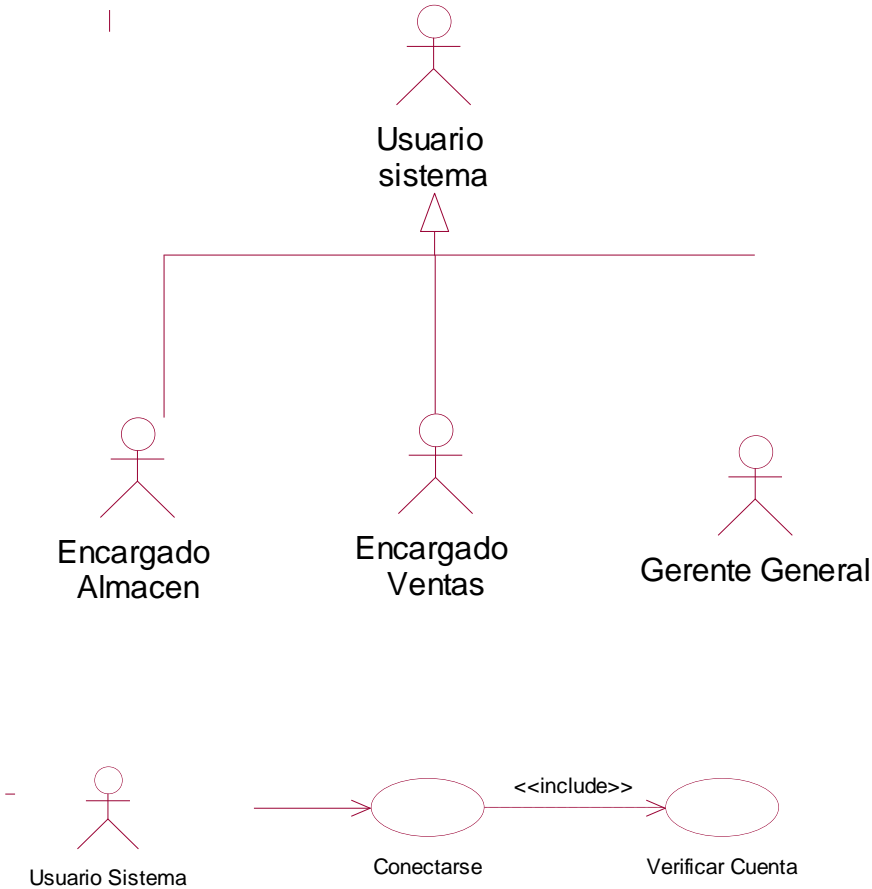
3.2.3 Diagramas de Casos de uso

3.2.3.1 Modelo de negocio



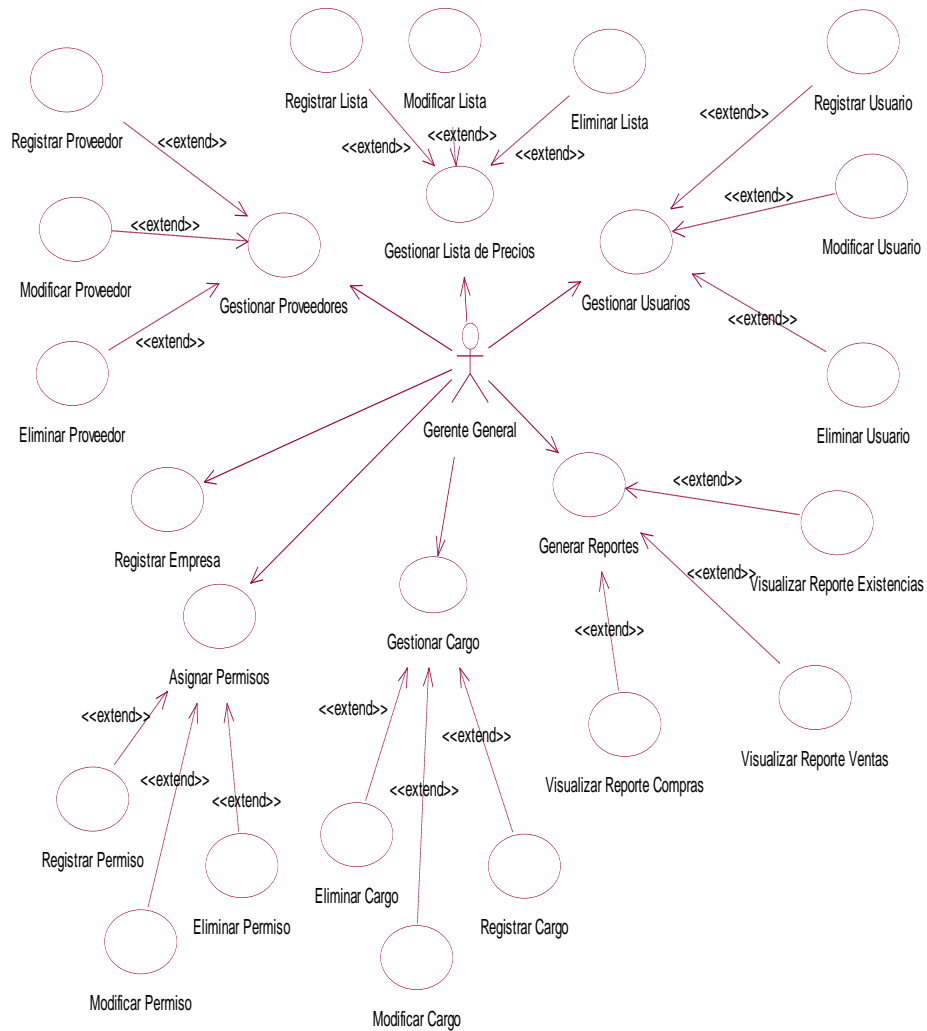
3.2.3.2 Casos de Uso para la Conexión al Sistema

El usuario del sistema utiliza el caso de uso conectarse e ingresar al sistema, mediante un nombre de usuario y contraseña



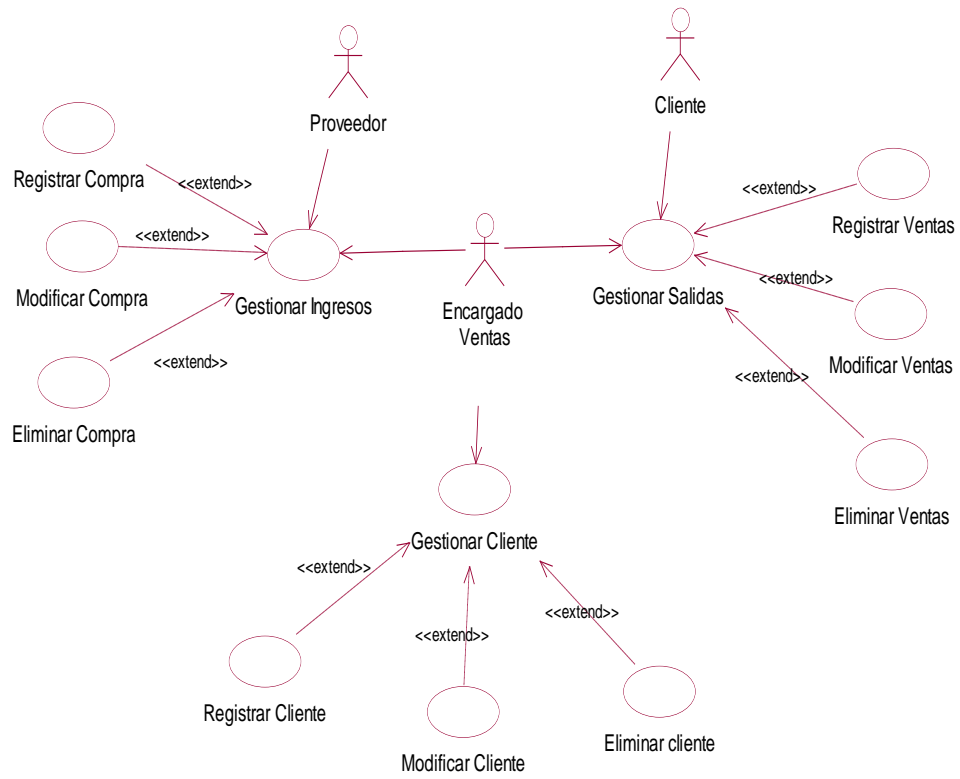
3.2.3.3 Casos de Uso para Gerente General

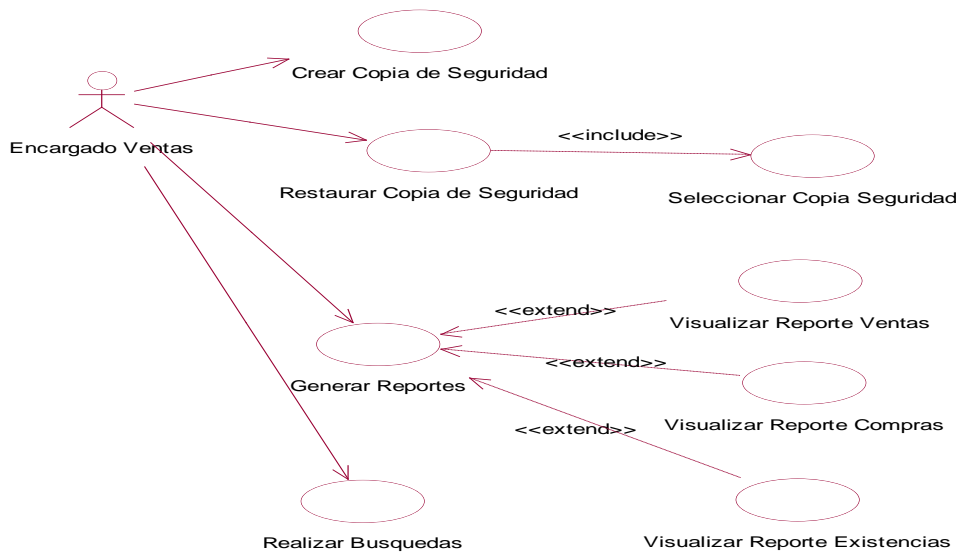
El caso de uso para gerente general son aquellos que realizan la gestión de proveedores, usuarios del sistema como vendedores y encargados de almacén, además de los permisos de estos usuarios.



3.2.3.4 Casos de Uso para Encargado de Ventas

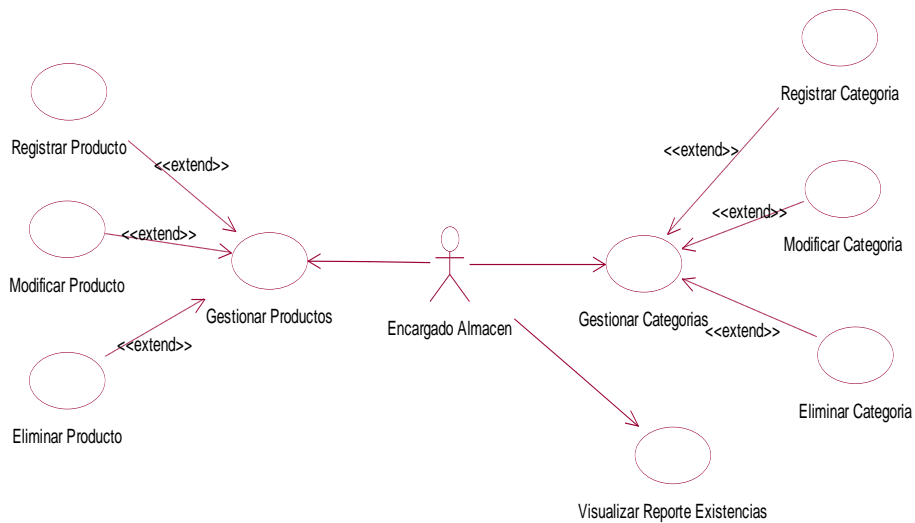
El caso de uso encargado de ventas es aquel que realiza la gestión de ingresos, entradas, clientes, reportes de ventas, compras y existencias.





3.2.3.5 Casos de Uso para Encargado Almacén.

El caso de uso de encargado de almacén es aquel que realiza gestión de los productos, categorías de productos y además realiza el inventario de los productos y de todas las existencias.



3.3 Descripción de Casos de Uso

Casos de Uso	Conectarse
Descripción	El usuario ingresa su usuario y clave
Actor	Usuario
Tipo	Primario
Precondición	<ul style="list-style-type: none">✓ Tener un usuario✓ Ingresado al sistema
Flujo Normal	<ol style="list-style-type: none">1. El sistema despliega el formulario para introducir el usuario y clave.2. El usuario ingresa el usuario y clave.3. El usuario hace click en el botón ingresar
Flujo Alternativo	<ol style="list-style-type: none">1. El usuario hace click en cerrar cancelando el ingreso al sistema
Pos condición	Validar Usuario

Casos de Uso	Registrar Empresa
Descripción	El gerente general registra los datos de la empresa.
Actor	Gerente General
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Registrar datos de empresa.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra el formulario para registrar. 2. El gerente general introduce los datos de la empresa. 3. El gerente general hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El gerente general hace click en el botón cancelar el registro. 2. El gerente general introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Usuarios
Descripción	El gerente general registra, modifica o elimina los datos de los usuarios en el sistema.
Actor	Gerente General
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar Usuarios.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los usuarios. 2. El gerente general selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El gerente general introduce los datos de los usuarios. 5. El gerente general selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El gerente general introduce los datos a modificar. 8. El gerente general selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los usuarios. 10. El gerente general hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El gerente general hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El gerente general introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Asignar permisos
Descripción	El Gerente General elige el usuario para asignarle un permiso.
Actor	Gerente General
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar asignar permisos
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un campo de texto para seleccionar el permiso. 2. El sistema le muestra un campo de texto para seleccionar el usuario. 3. El gerente general selecciona el permiso y el usuario. 4. El sistema asigna al usuario el permiso. 5. El gerente general hace click en guardar.
Flujo Alternativo	<ol style="list-style-type: none"> 6. El gerente general hace click en el botón cerrar.
Pos condición	Validar Datos

Casos de Uso	Gestionar Proveedores
Descripción	El gerente general registra, modifica o elimina los datos de los proveedores en el sistema.
Actor	Gerente General
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar proveedores.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los proveedores. 2. El gerente general selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El gerente general introduce los datos de los proveedores. 5. El gerente general selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El gerente general introduce los datos a modificar. 8. El gerente general selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los proveedores. 10. El gerente general hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El gerente general hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El gerente general introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Cargo
Descripción	El gerente general registra, modifica o elimina los datos de los cargos de todos los usuarios de la empresa.
Actor	Gerente General
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar cargos.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los cargos. 2. El gerente general selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El gerente general introduce los datos de los cargos. 5. El gerente general selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El gerente general introduce los datos a modificar. 8. El gerente general selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los cargos. 10. El gerente general hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El gerente general hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El gerente general introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Lista de Precios
Descripción	El encargado de ventas registra, modifica o elimina los precios de la lista de precios de la empresa.
Actor	Encargado Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar Lista de Precios.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los productos. 2. El encargado de ventas selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de ventas introduce los datos de los lista de precios y selecciona el producto al cual le afecta. 5. El encargado de ventas selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de ventas introduce los datos a modificar. 8. El encargado de ventas selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los productos. 10. El encargado de ventas hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El encargado de ventas introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Clientes
Descripción	El encargado de ventas registra, modifica o elimina los datos de los clientes en el sistema.
Actor	Encargado Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar clientes.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los clientes. 2. El encargado de ventas selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de ventas introduce los datos de los clientes. 5. El encargado de ventas selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de ventas introduce los datos a modificar. 8. El encargado de ventas selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los clientes. 10. El encargado de ventas hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El encargado de ventas introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Ingresos
Descripción	El Encargado de Ventas registra, modifica o elimina los datos de las compras de los distintos proveedores.
Actor	Encargado Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar Ingresos.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los productos. 2. El encargado de ventas selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de ventas introduce los datos de las compras. 5. El encargado de ventas selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de ventas introduce los datos a modificar. 8. El encargado de ventas selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los ingresos. 10. El encargado de ventas hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar cancelando la adición, modificación o eliminación. 2. El encargado de ventas introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Casos de Uso	Gestionar Salidas
Descripción	El Encargado de Ventas registra, modifica o elimina los datos de las ventas a los clientes de acuerdo a las existencias en el sistema.
Actor	Encargado Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar Salidas.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los productos. 2. El encargado de ventas selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de ventas introduce los datos de las ventas. 5. El encargado de ventas selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de ventas introduce los datos a modificar. 8. El encargado de ventas selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de la venta. 10. El encargado de ventas hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar cancelando la adición, modificación o eliminación. 2. El encargado de ventas introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

Caso de Uso	Visualizar Reporte de Existencias
Descripción	El sistema le muestra las existencias que existe en el almacén de la empresa.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Reporte Existencias.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra formulario para elegir los parámetros (Rango Fechas). 2. El encargado de ventas ingresa el rango de fechas. 3. El sistema visualiza reporte de las existencias de todos los productos dentro del almacén según el rango de fechas. 4. El encargado de ventas selecciona imprimir. 5. El sistema visualiza el reporte a imprimir.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar
Pos condición	Validar Datos

Caso de Uso	Visualizar Reporte de Compras
Descripción	El sistema le muestra las compras de acuerdo al rango de fechas.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Reporte Compras.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra formulario para elegir los parámetros (Rango Fechas). 2. El encargado de ventas ingresa el rango de fechas. 3. El sistema visualiza reporte de las compras de acuerdo a proveedores y según el rango de fechas. 4. El encargado de ventas selecciona imprimir. 5. El sistema visualiza el reporte a imprimir.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar
Pos condición	Validar Datos

Caso de Uso	Visualizar Reporte de Ventas
Descripción	El sistema le muestra las ventas de acuerdo al rango de fechas.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Reporte Ventas.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra formulario para elegir los parámetros (Rango Fechas). 2. El encargado de ventas ingresa el rango de fechas. 3. El sistema visualiza reporte de las ventas de acuerdo a clientes y según el rango de fechas. 4. El encargado de ventas selecciona imprimir. 5. El sistema visualiza el reporte a imprimir.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar
Pos condición	Validar Datos

Caso de Uso	Realizar Búsqueda
Descripción	El sistema realiza la búsqueda de algún producto en el almacén para ver su existencia.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Búsqueda.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra campo para introducir los parámetros de la búsqueda. 2. El encargado de ventas introduce los datos de los parámetros de búsqueda. 3. El sistema busca los productos dentro del almacén. 4. El sistema visualiza el producto y la cantidad existente.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar
Pos condición	Validar Datos

Casos de Uso	Crear Copia de Seguridad
Descripción	El encargado de ventas realiza copias de seguridad para poder luego restablecer si es necesario.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar administrar. ✓ Pulsado en copias de seguridad.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un campo de texto y la opción examinar para elegir la ubicación donde se realizará la copia de seguridad 2. El encargado de ventas ingresa la ubicación donde se realizara la nueva copia de seguridad. 3. El encargado de ventas hace click en el botón Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar.
Pos condición	Se guardan los datos.

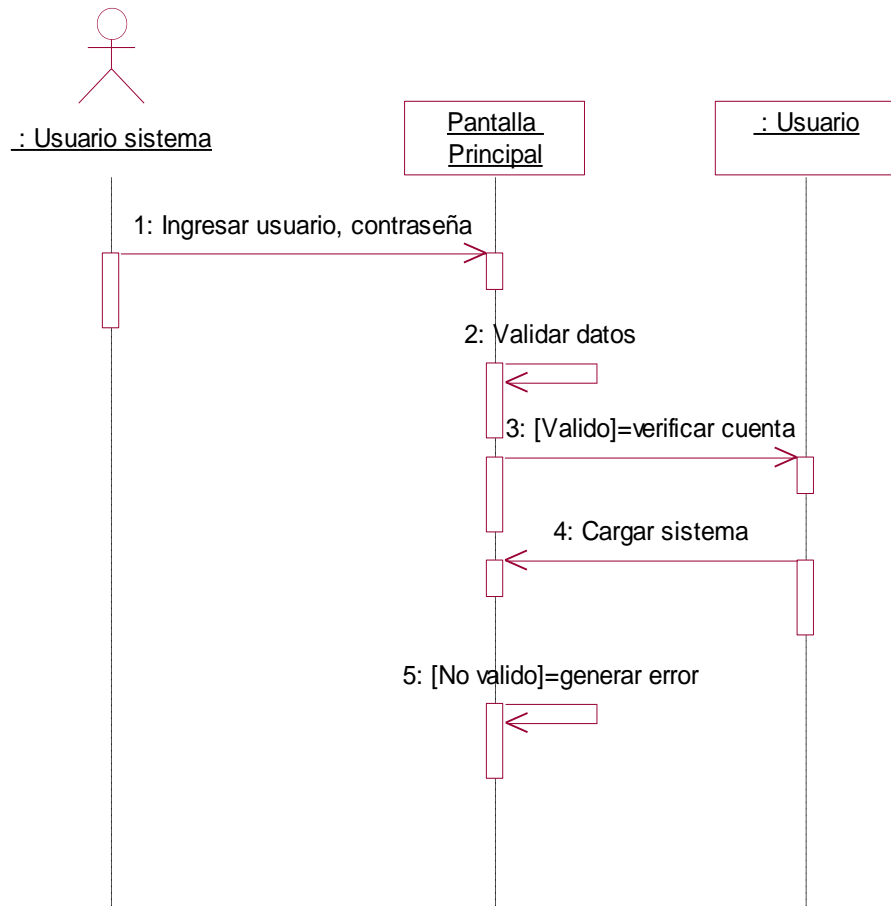
Casos de Uso	Restaurar Copias de Seguridad
Descripción	El encargado de ventas restablece una copia de seguridad que vea conveniente realizarlo.
Actor	Encargado de Ventas
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar administrar. ✓ Pulsado en restaurar copias de seguridad.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un campo de texto y la opción examinar para elegir la ubicación donde se encuentra la copia de seguridad a restablecer 2. El encargado de ventas ingresa la ubicación donde se encuentra la copia a restablecer. 3. El encargado de ventas hace click en el botón Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de ventas hace click en el botón cancelar.
Pos condición	Se restablecen los datos de la base de datos.

Casos de Uso	Gestionar Categorías
Descripción	El Encargado de almacén registra, modifica o elimina los datos de las categorías de los productos dentro del sistema.
Actor	Encargado de Almacén
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar categorías.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de las categorías. 2. El encargado de almacén selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de almacén introduce los datos de las categorías de los productos. 5. El encargado de almacén selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de almacén introduce los datos a modificar. 8. El encargado de almacén selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de las categorías. 10. El encargado de almacén hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de almacén hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El encargado de almacén introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

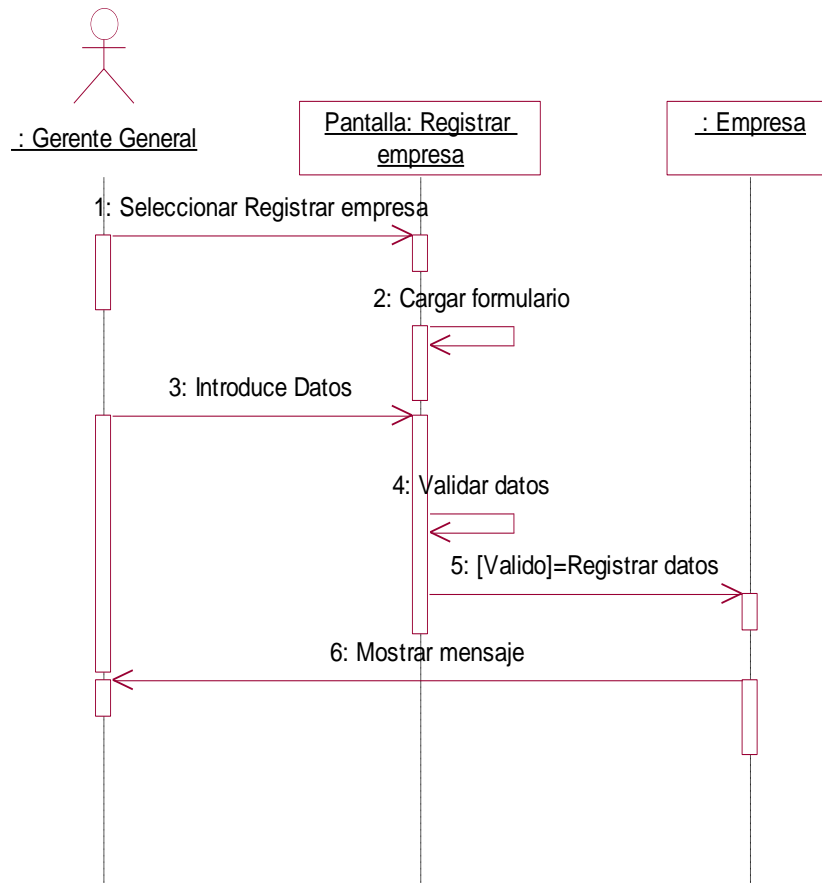
Casos de Uso	Gestionar Productos
Descripción	El Encargado de almacén registra, modifica o elimina los datos de los productos que cuenta el almacén dentro del sistema.
Actor	Encargado de Almacén
Tipo	Primario
Precondición	<ul style="list-style-type: none"> ✓ Conectarse ✓ Seleccionar Gestionar productos.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema le muestra un listado de los productos. 2. El encargado de almacén selecciona registrar. 3. El sistema le muestra el formulario para registrar. 4. El encargado de almacén introduce los datos de los productos selecciona la categoría y el almacén. 5. El encargado de almacén selecciona modificar. 6. El sistema le muestra el formulario con los datos a modificar. 7. El encargado de almacén introduce los datos a modificar. 8. El encargado de almacén selecciona eliminar. 9. El sistema le muestra un mensaje con la confirmación de la eliminación de los productos. 10. El encargado de almacén hace click en Aceptar.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El encargado de almacén hace click en el botón cancelar cancelando el registro, modificación o eliminación. 2. El encargado de almacén introduce datos inválidos. 3. El sistema muestra mensaje de error.
Pos condición	Validar Datos

3.4 Diagrama de Secuencia

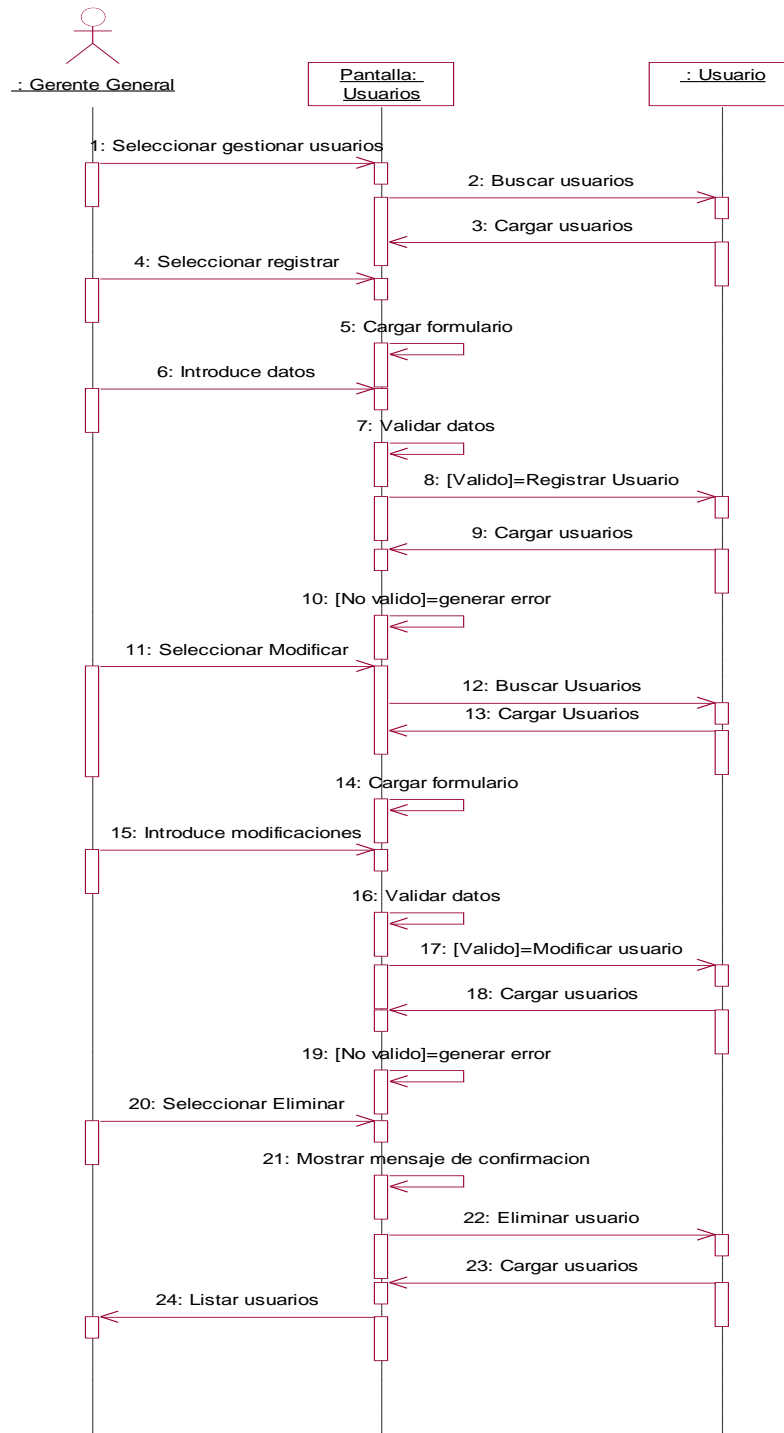
3.4.1 Diagrama Secuencia Conectarse



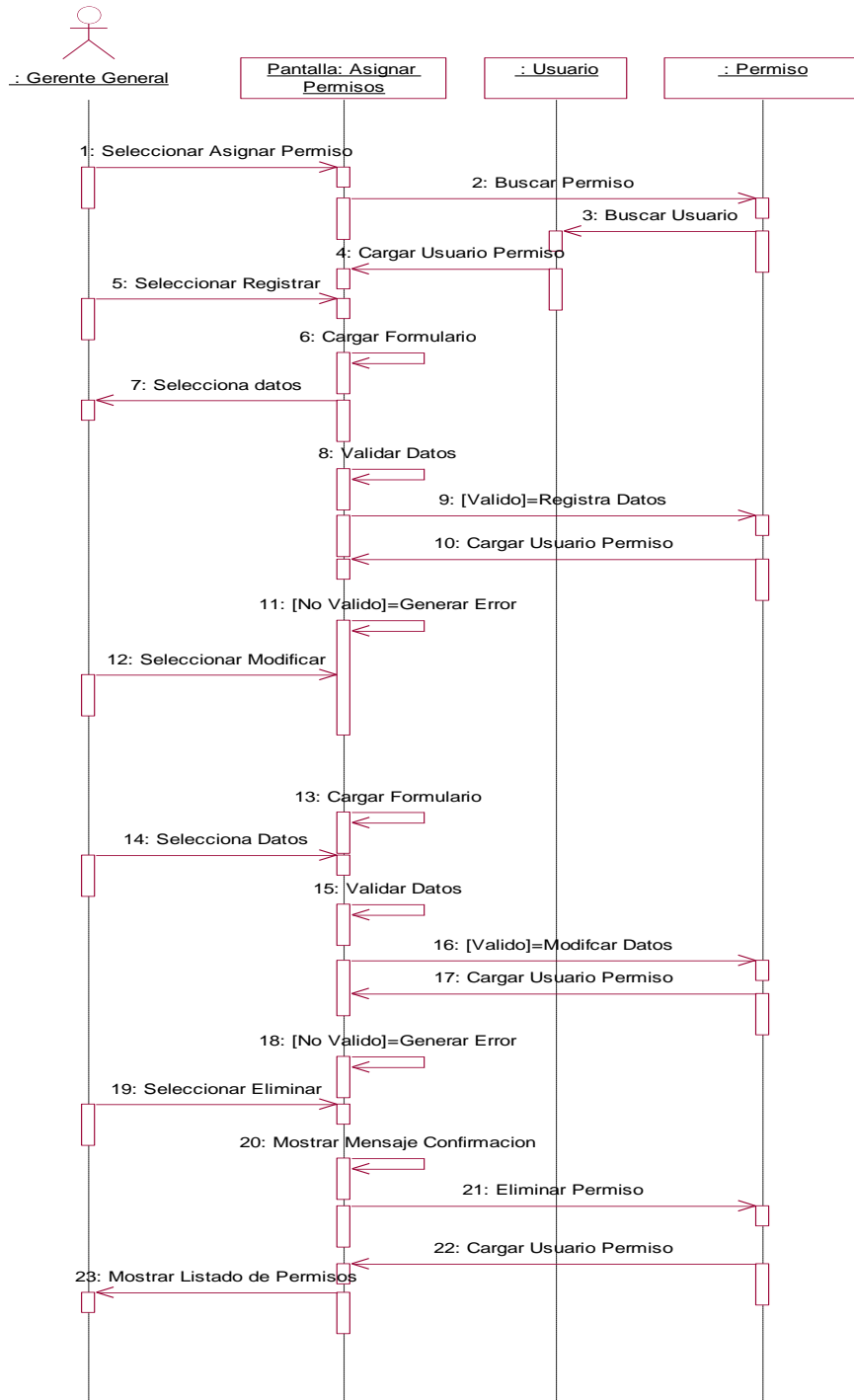
3.4.2 Diagrama Secuencia Registrar Empresa



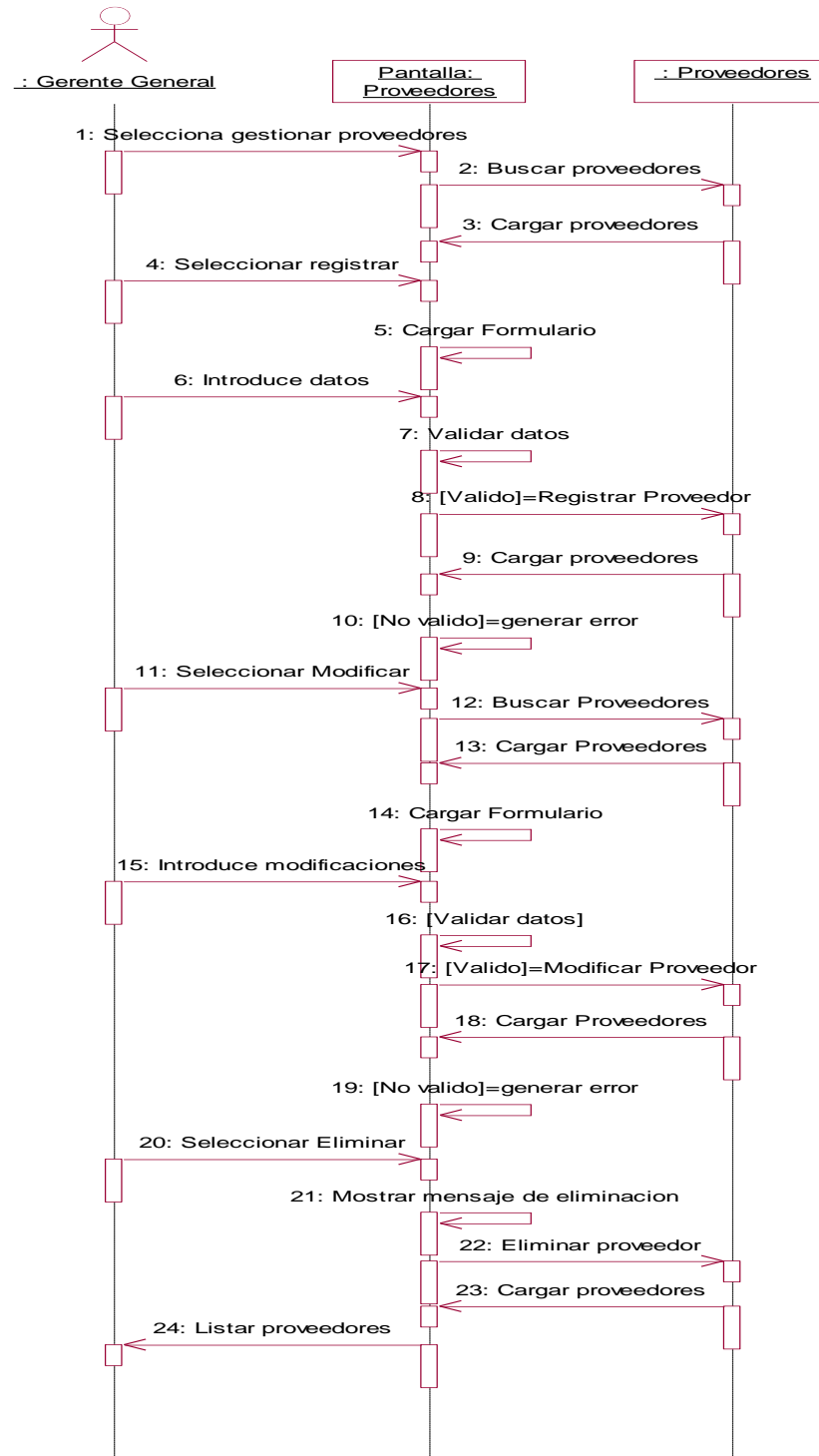
3.4.3 Diagrama Secuencia Gestionar Usuarios



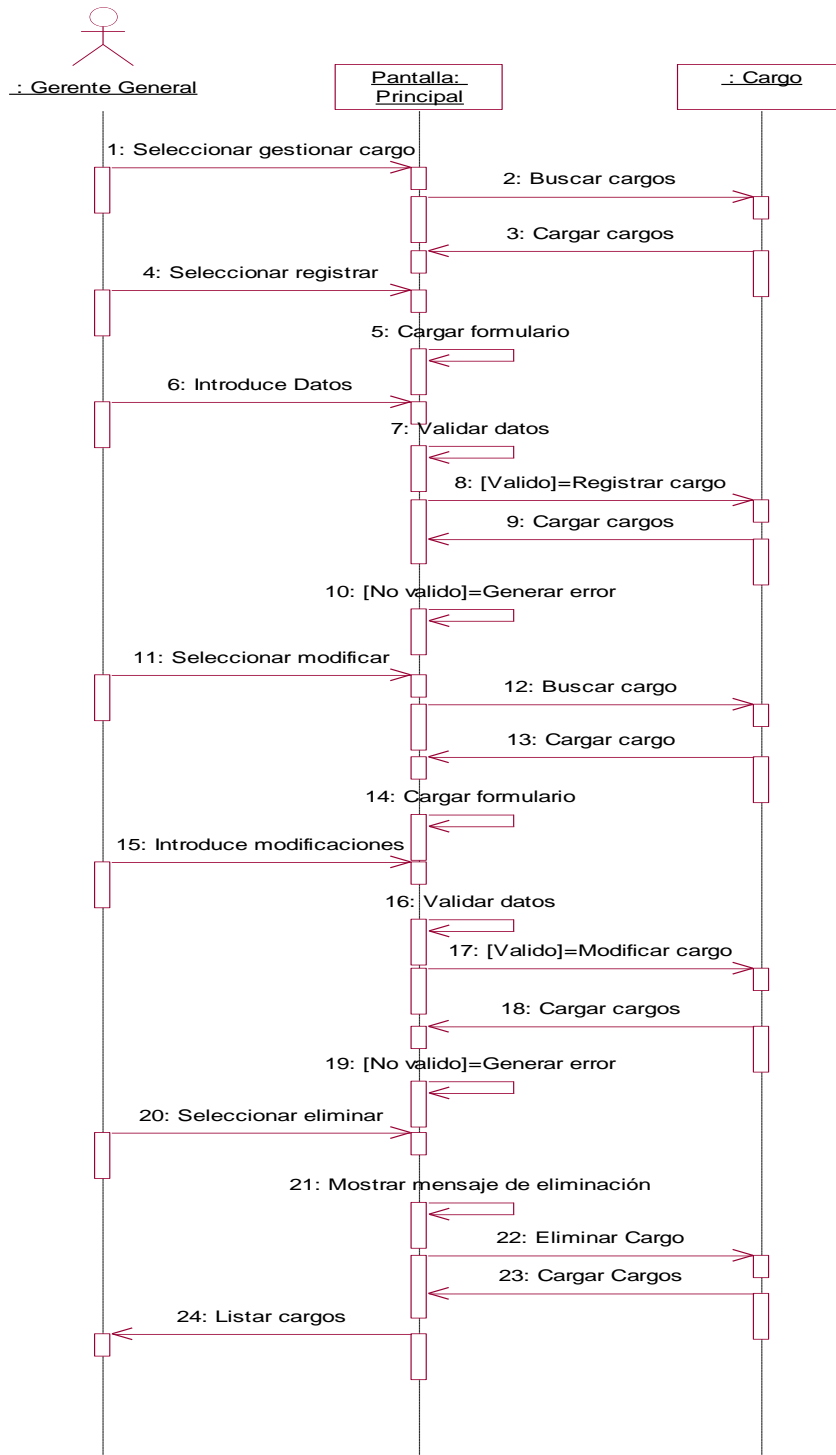
3.4.4 Diagrama Secuencia Asignar Permisos



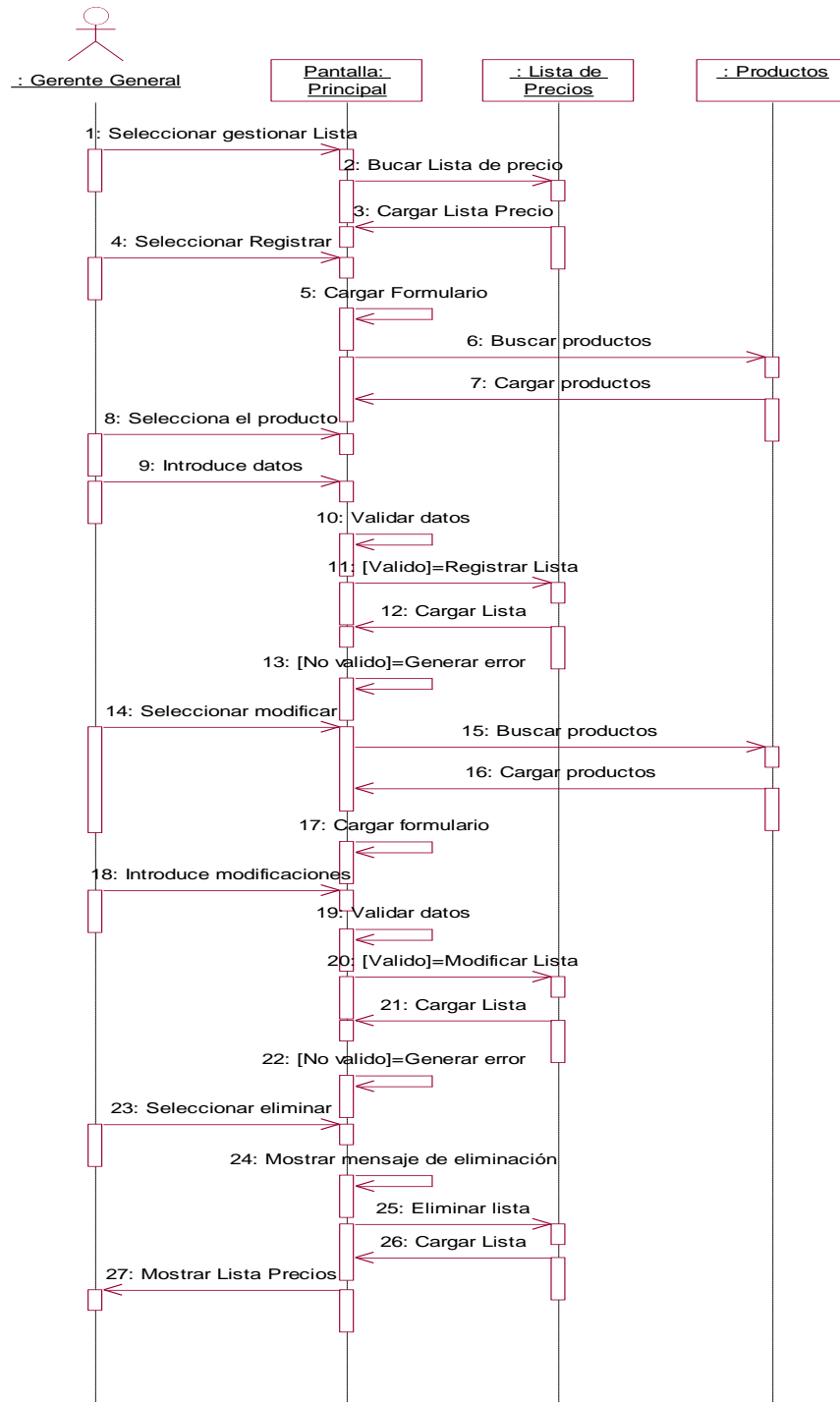
3.4.5 Diagrama Secuencia Gestionar Proveedores



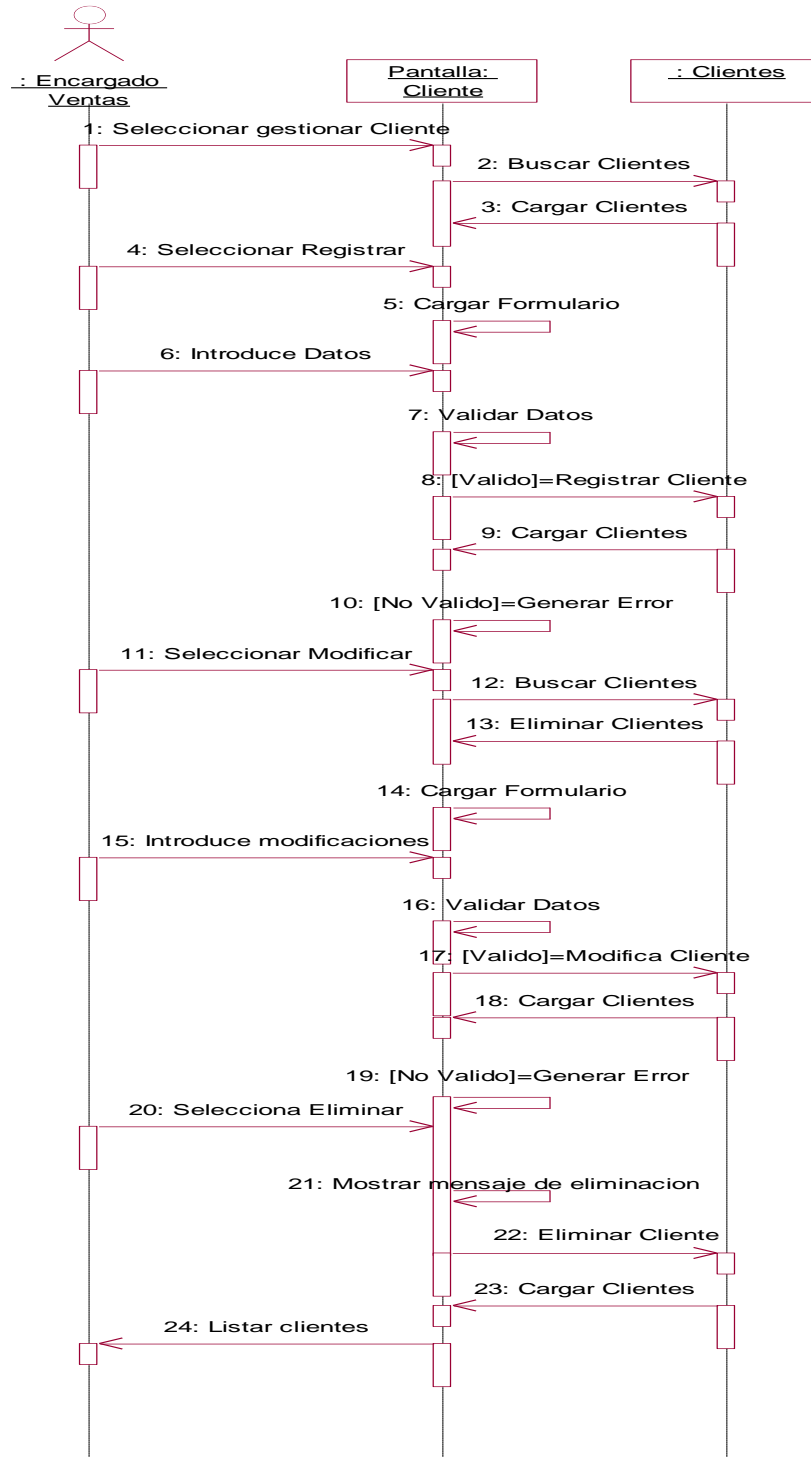
3.4.6 Diagrama Secuencia Gestionar Cargo



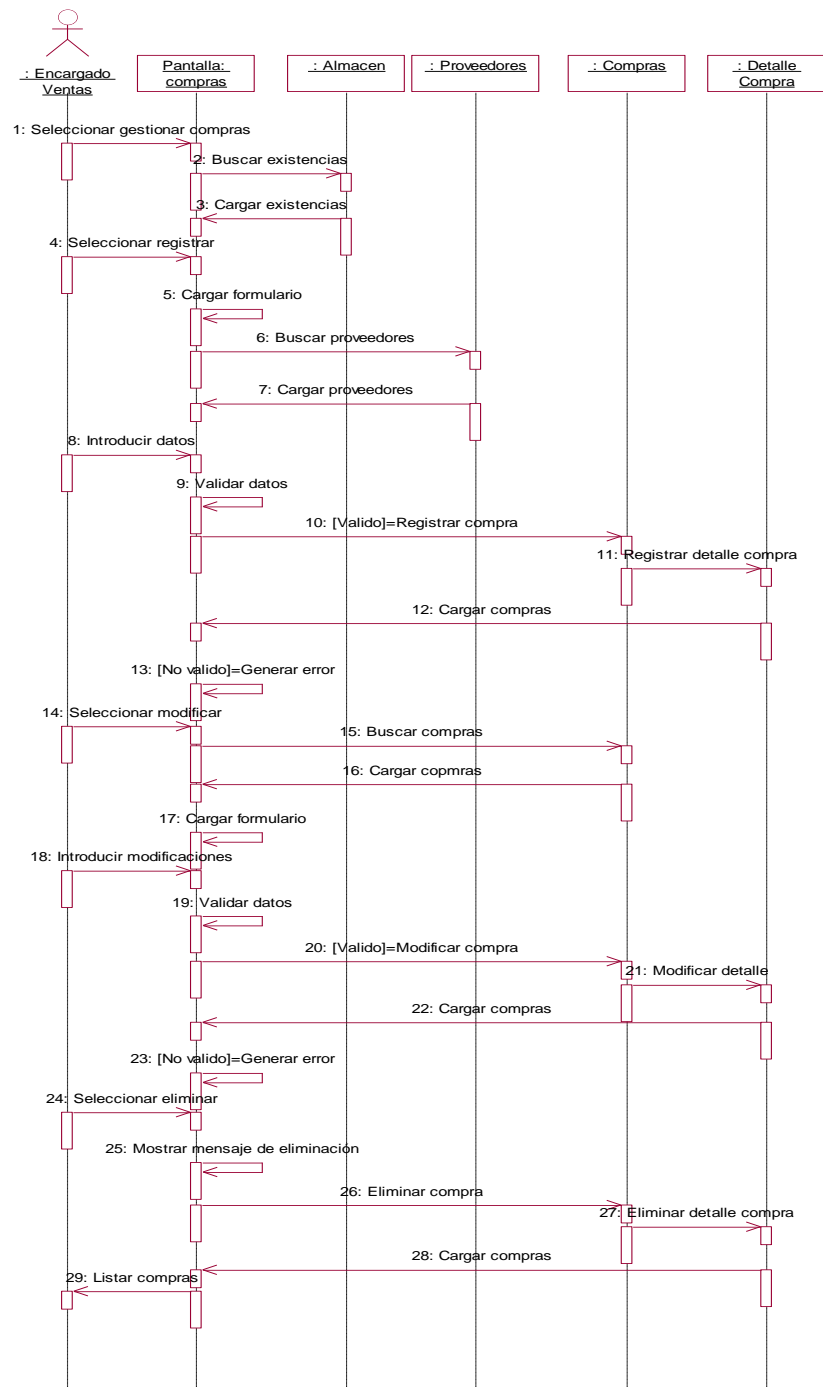
3.4.7 Diagrama Secuencia Gestionar Lista de Precios



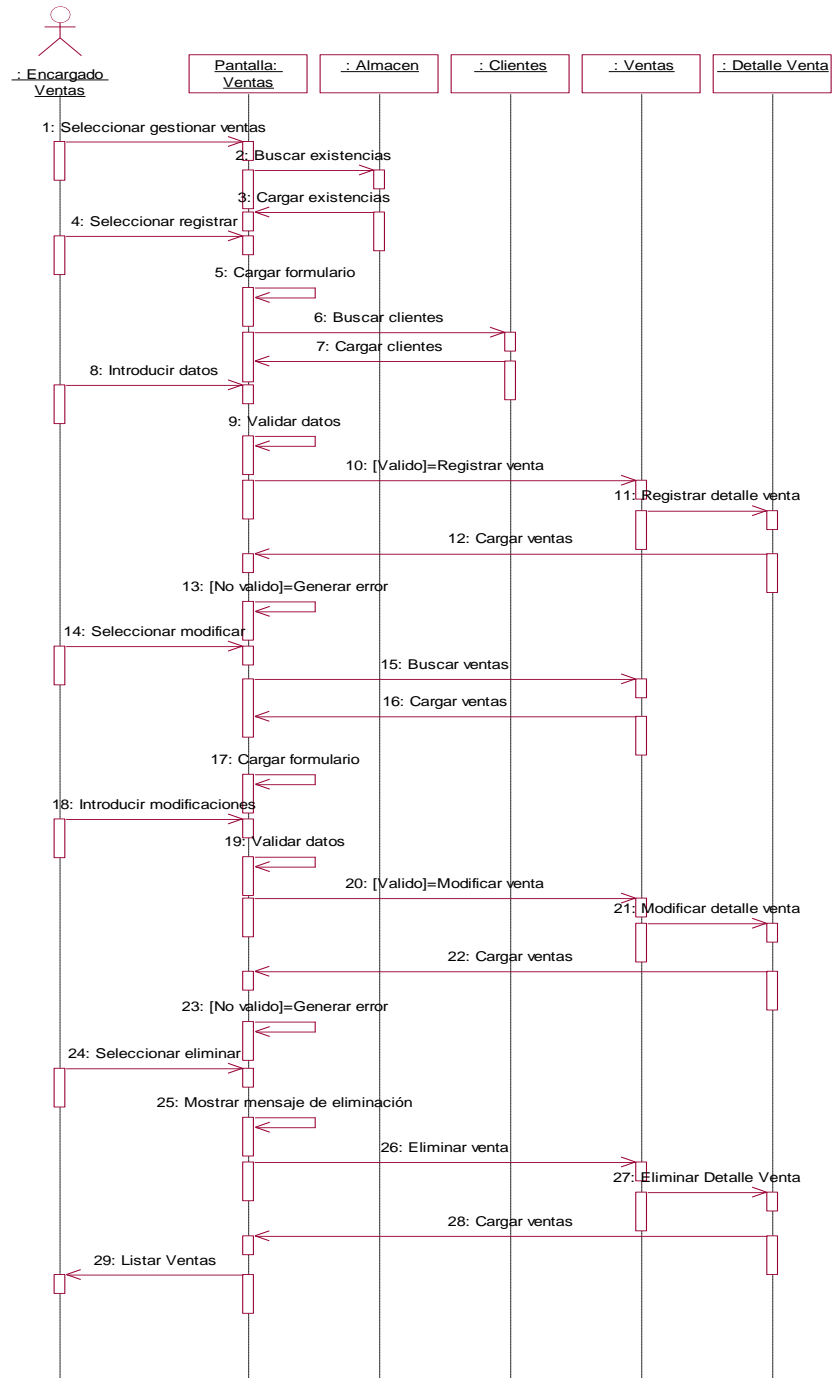
3.4.8 Diagrama Secuencia Gestionar Cliente



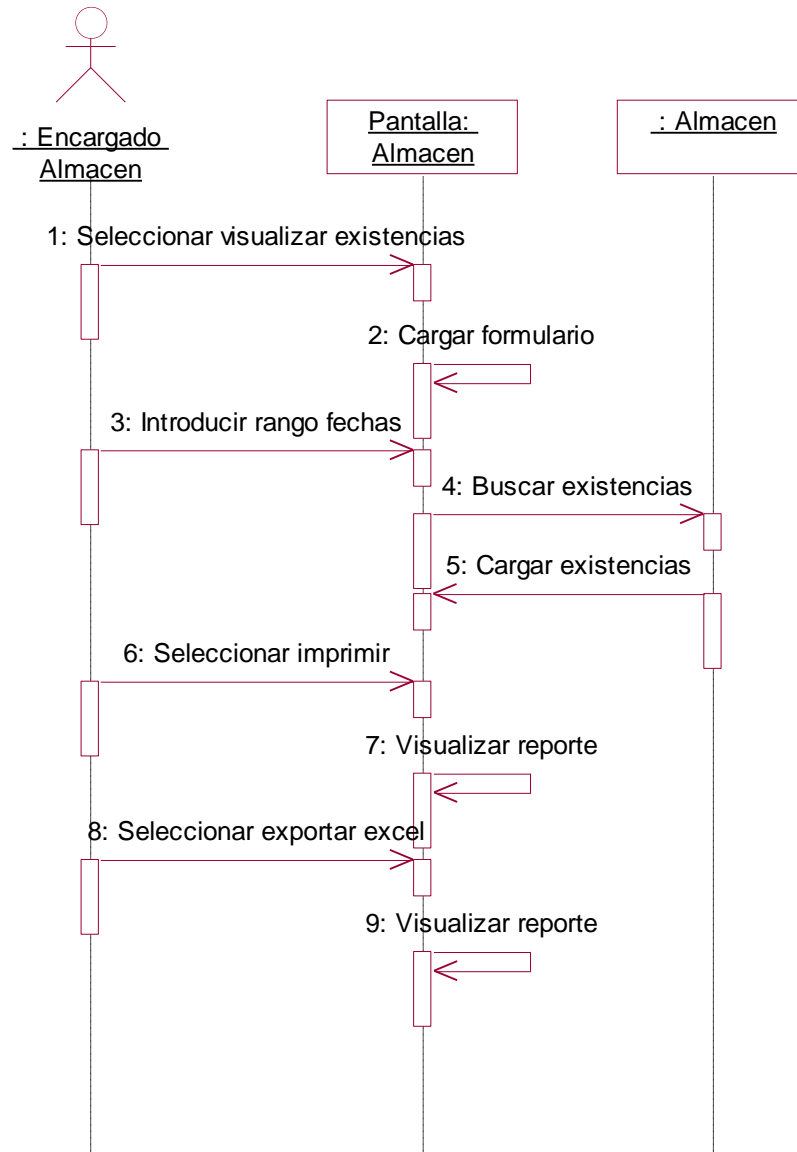
3.4.9 Diagrama Secuencia Gestionar Ingresos



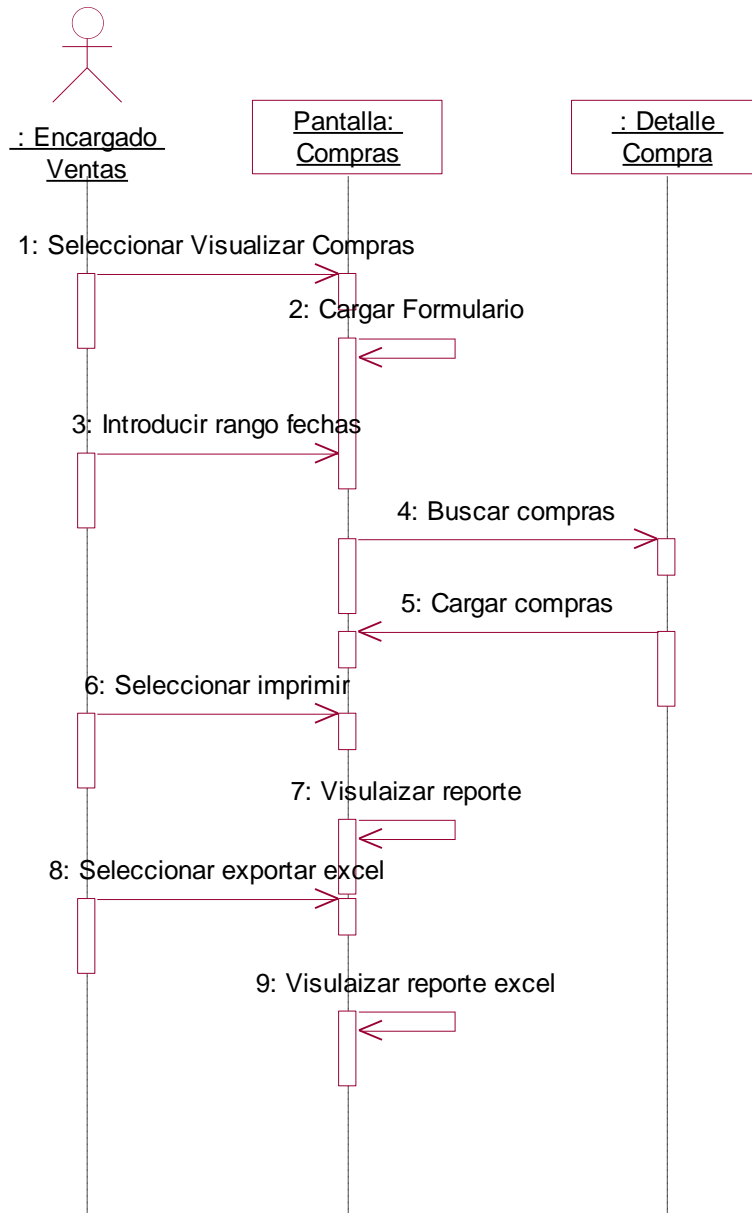
3.4.10 Diagrama Secuencia Gestionar Salidas



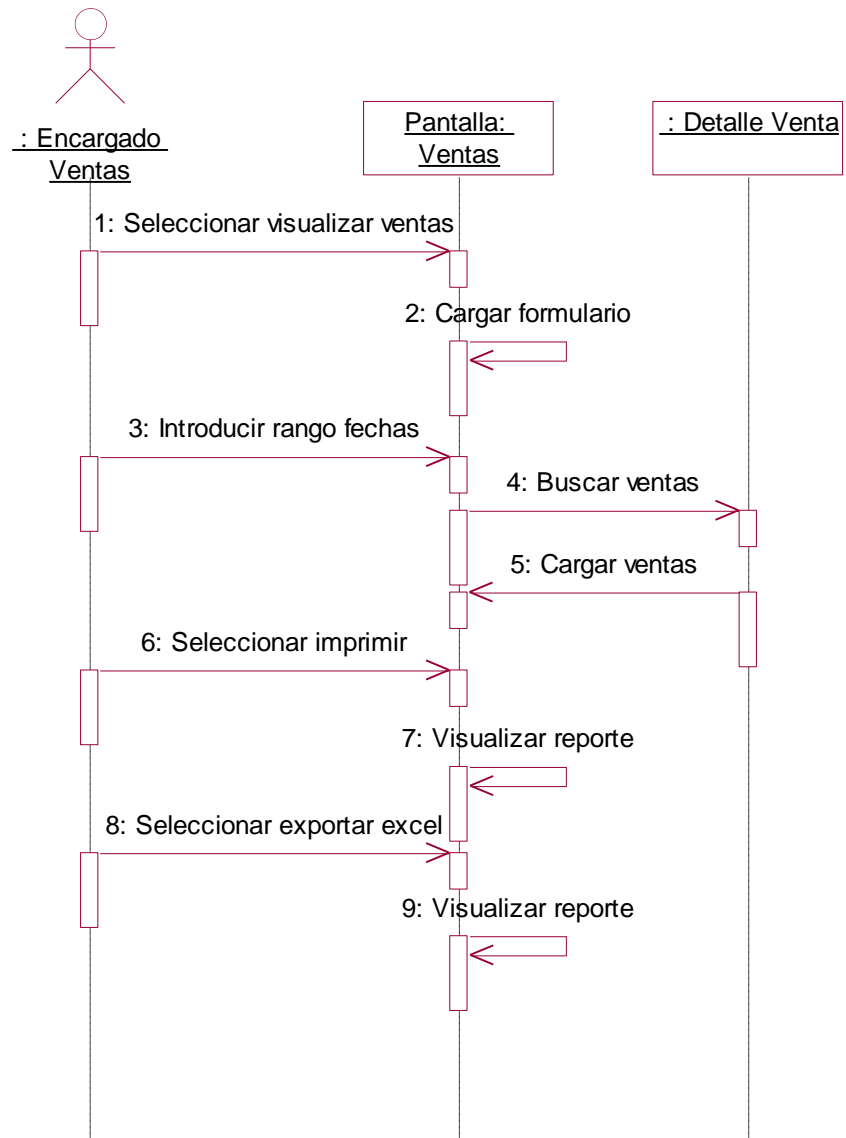
3.4.11 Diagrama Secuencia Visualizar Existencias



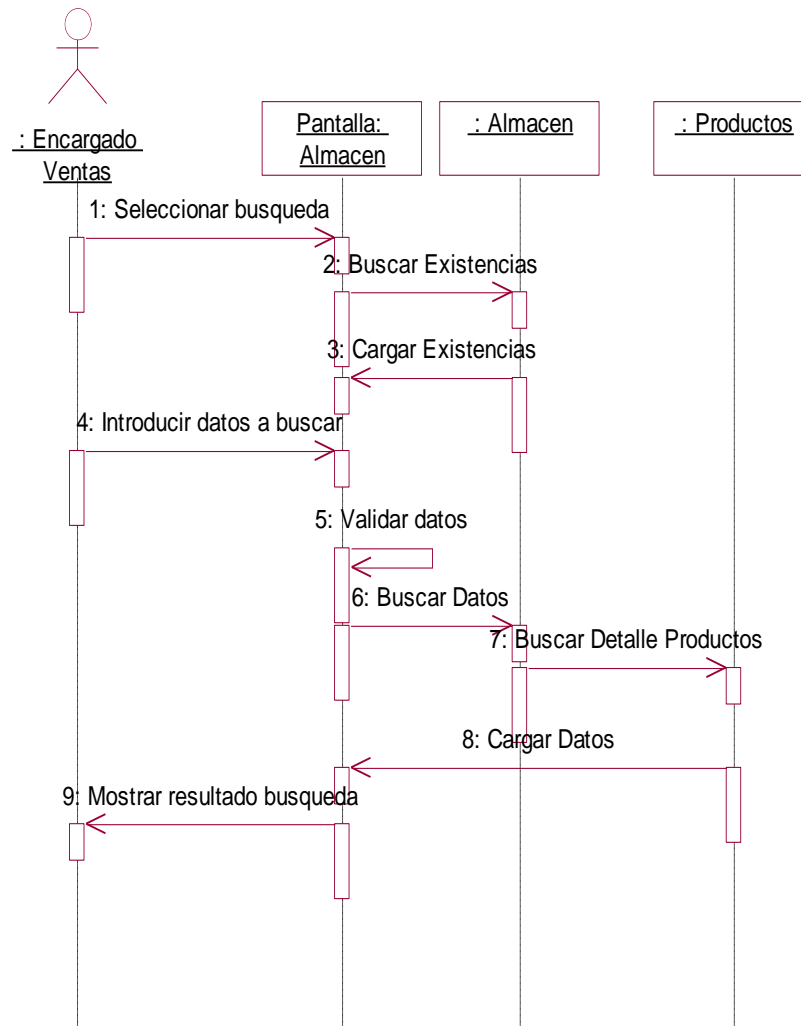
3.4.12 Diagrama Secuencia Visualizar Compras



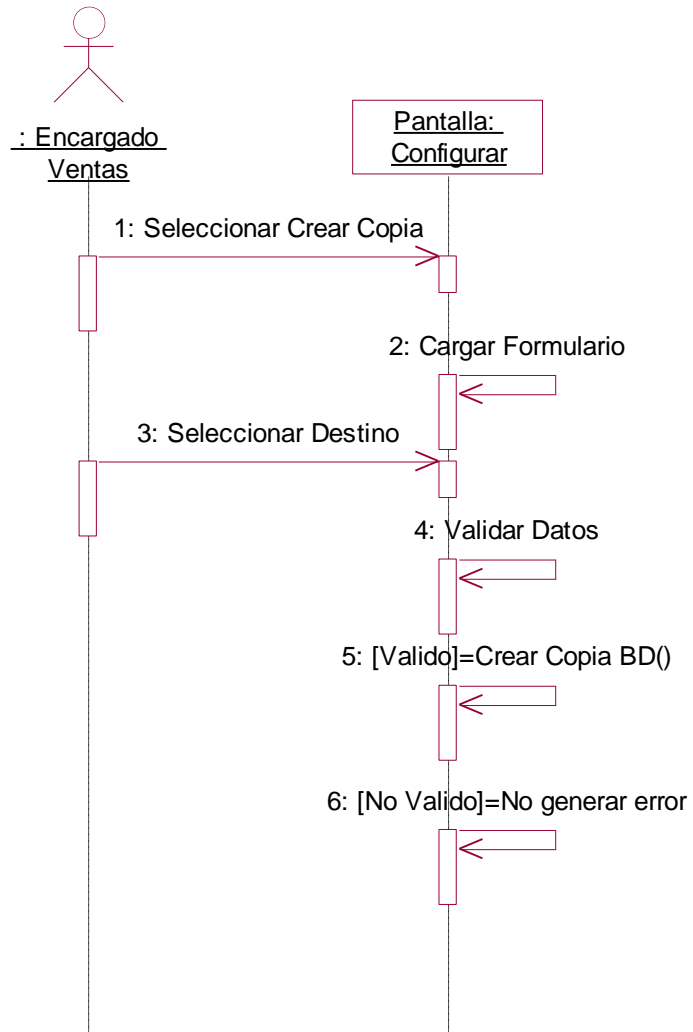
3.4.13 Diagrama Secuencia Visualizar Ventas



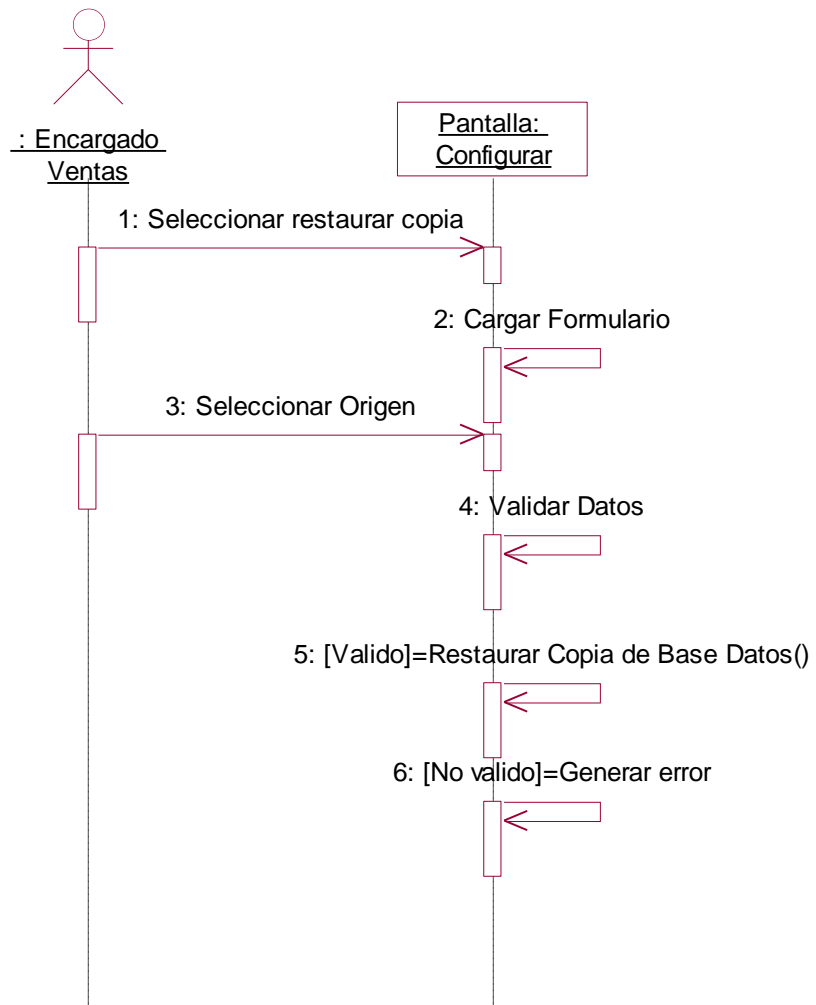
3.4.14 Diagrama Secuencia Realizar Búsquedas



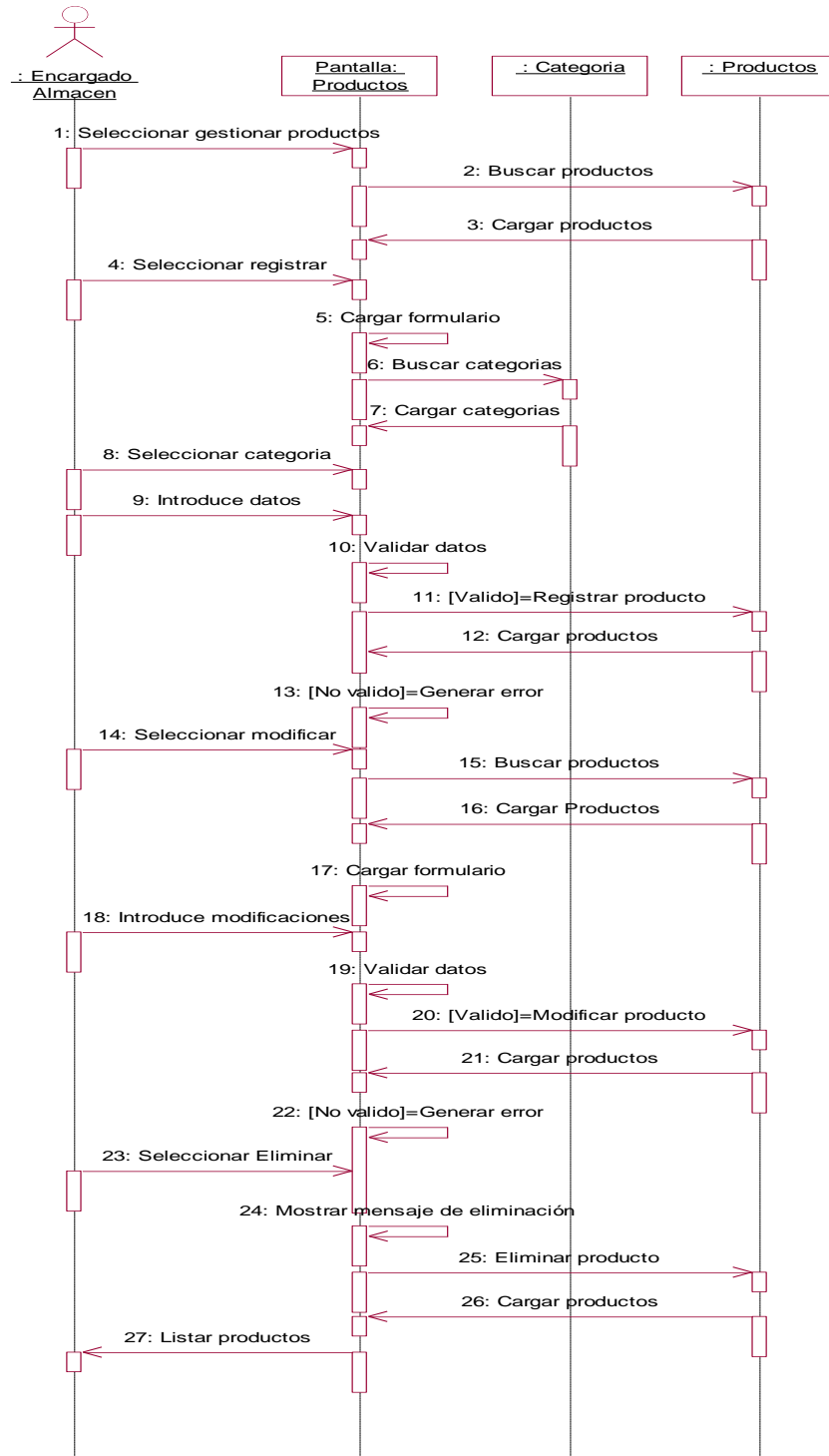
3.4.15 Diagrama Secuencia Crear Copia Seguridad



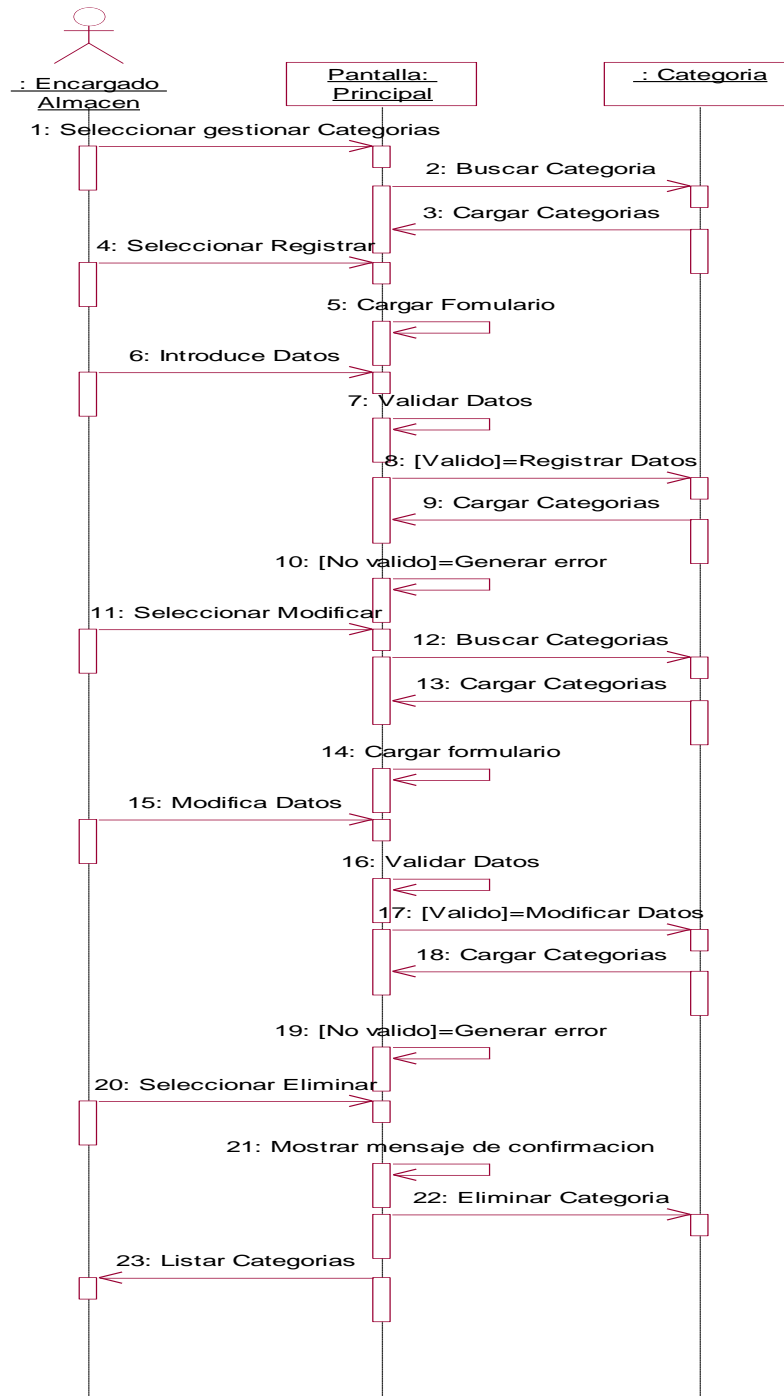
3.4.16 Diagrama Secuencia Restaurar Copia Seguridad



3.4.17 Diagrama Secuencia Gestionar Productos

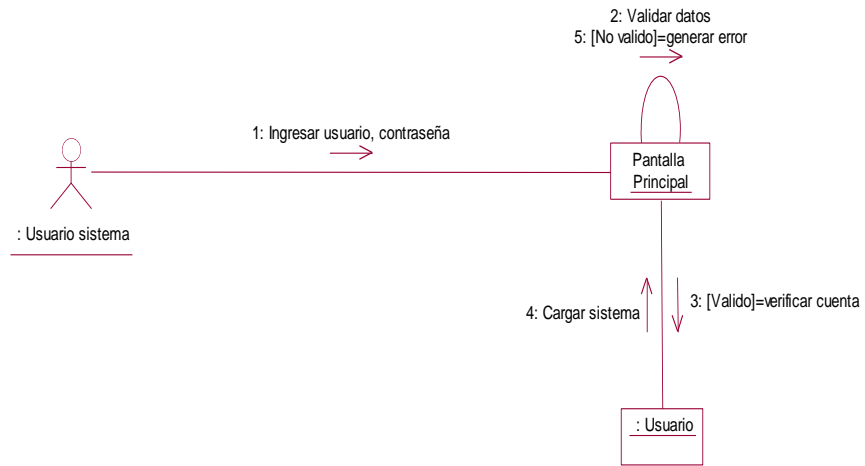


3.4.18 Diagrama Secuencia Gestionar Categorías

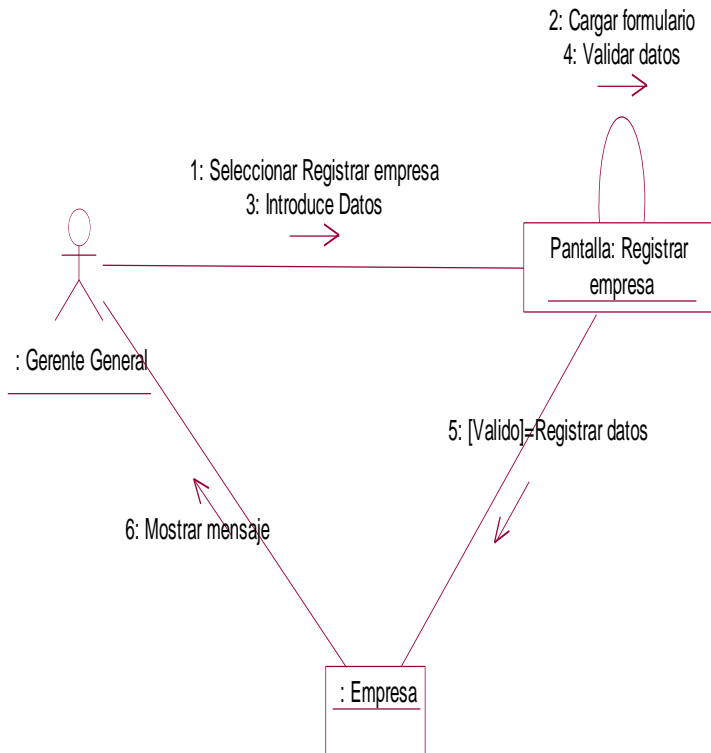


3.5 Diagrama Colaboración

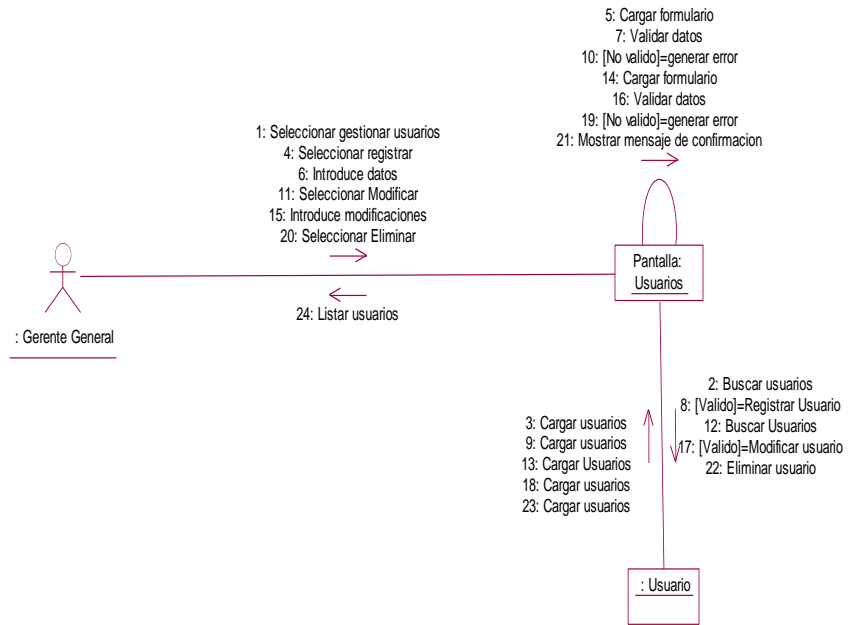
3.5.1 Diagrama Colaboración Conectarse



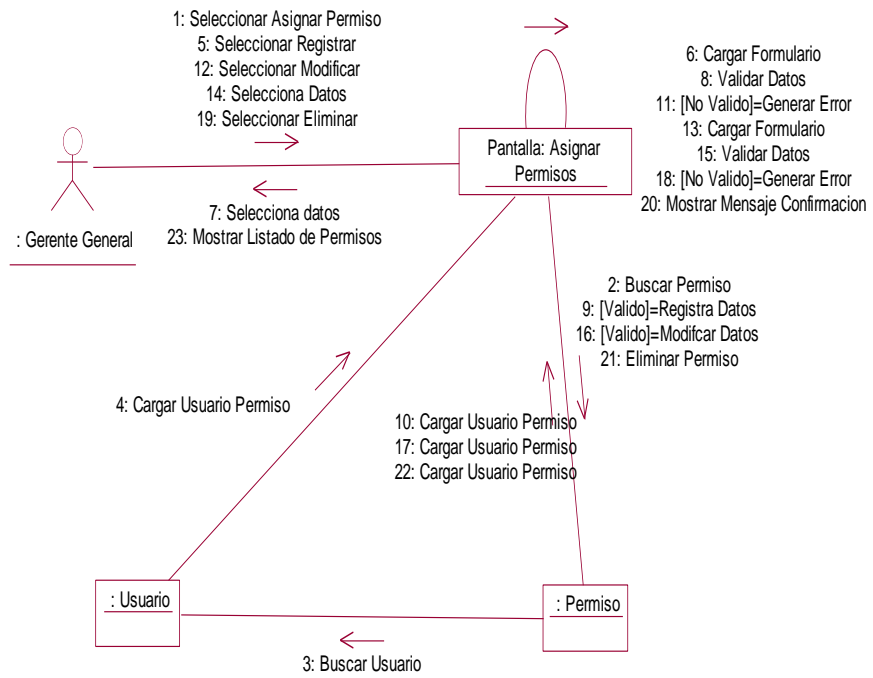
3.5.2 Diagrama Colaboración Registrar Empresa



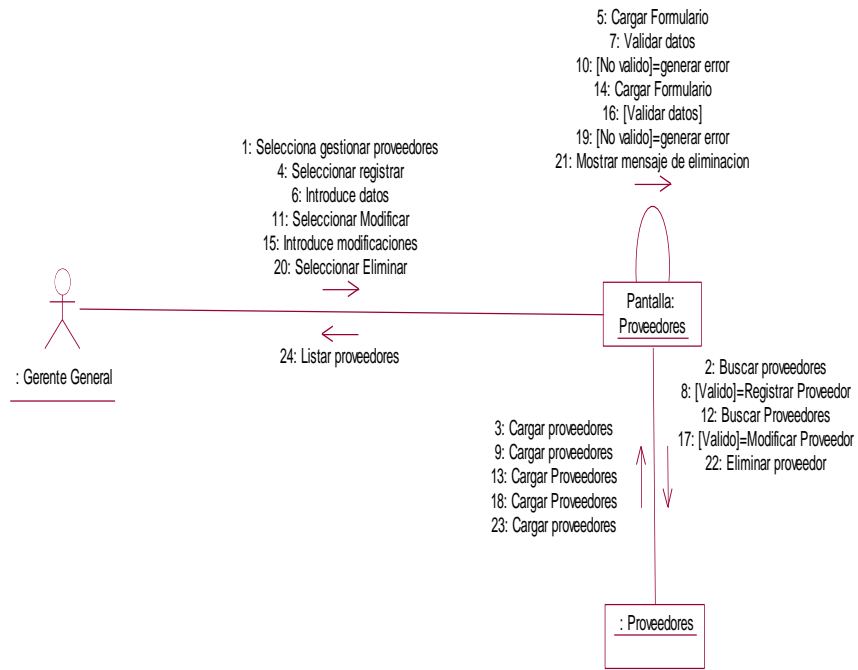
3.5.3 Diagrama Colaboración Gestionar Usuarios



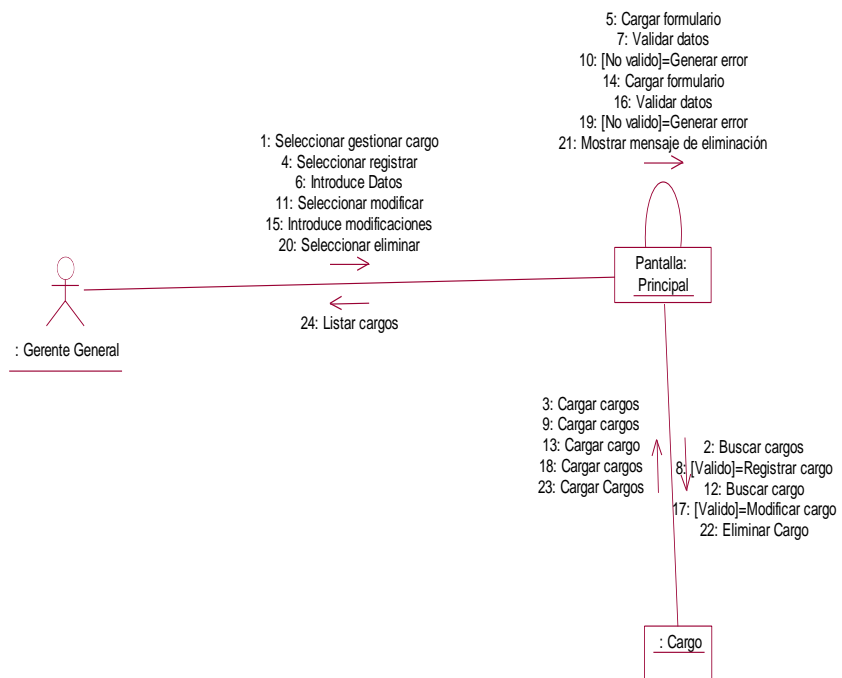
3.5.4 Diagrama Colaboración Asignar Permisos



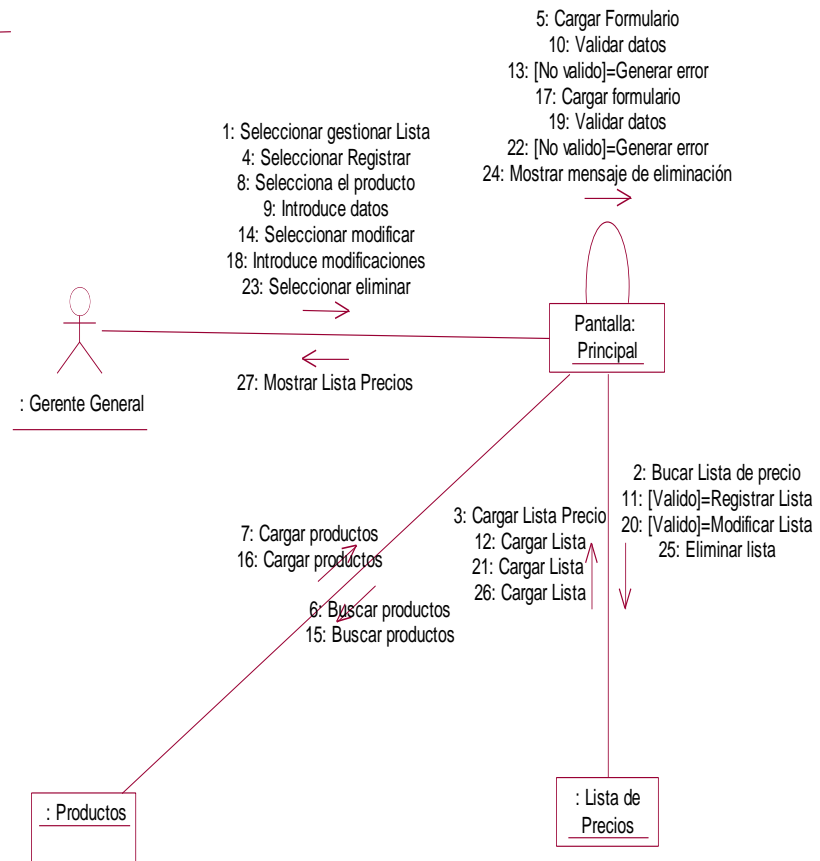
3.5.5 Diagrama Colaboración Gestionar Proveedores



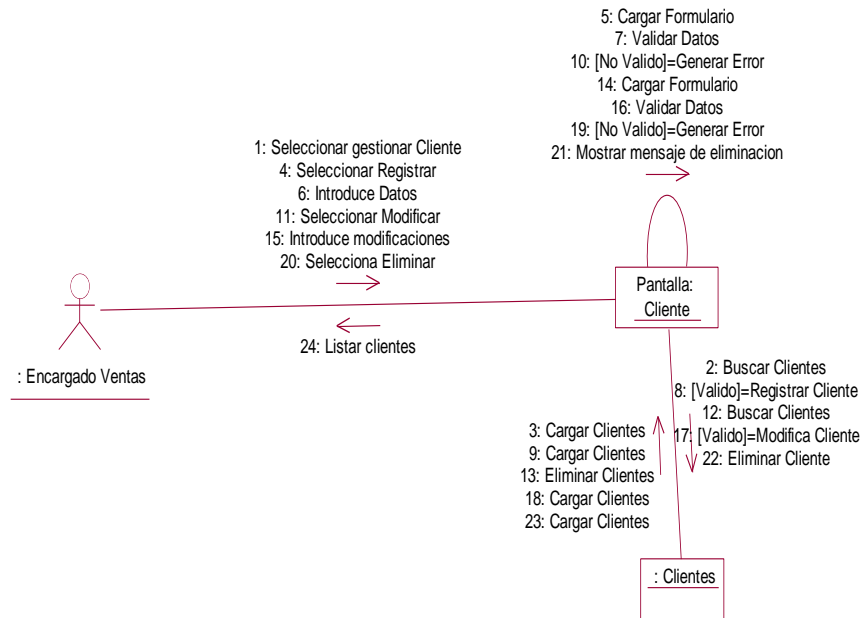
3.5.6 Diagrama Colaboración Gestionar Cargo



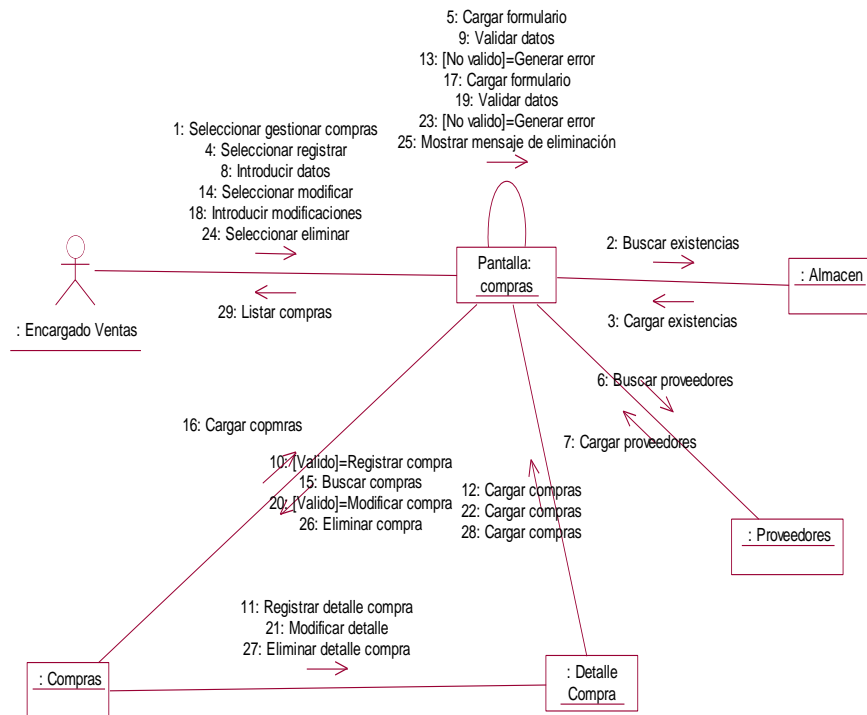
3.5.7 Diagrama Colaboración Gestionar Lista de Precios



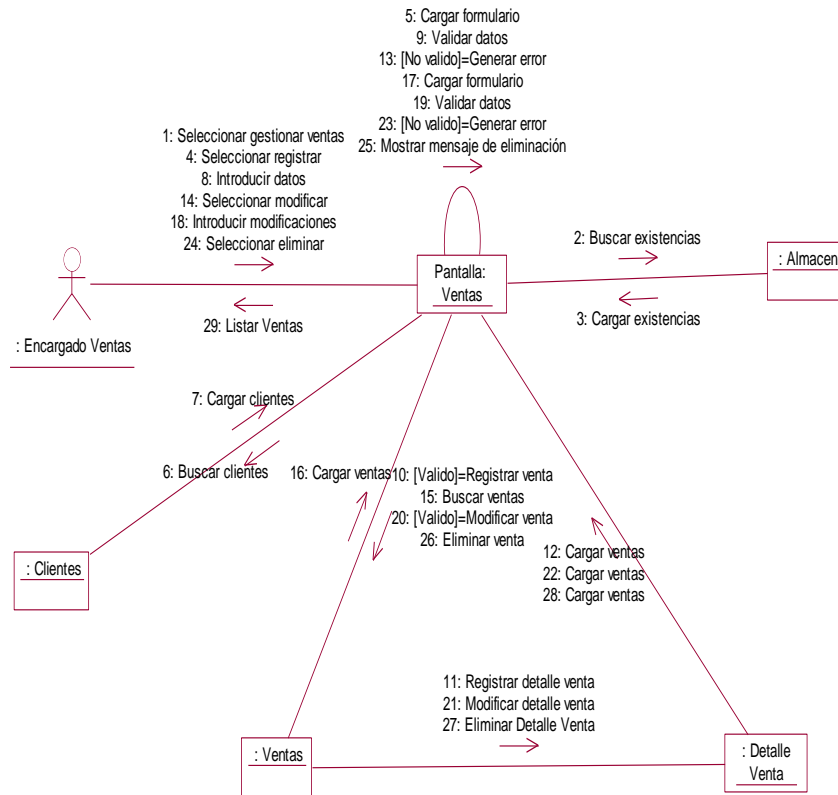
3.5.8 Diagrama Colaboración Gestionar Cliente



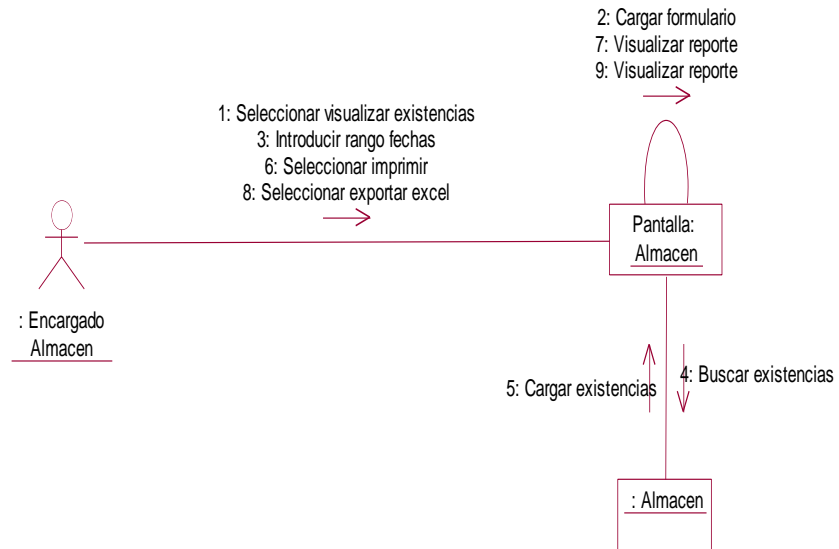
3.5.9 Diagrama Colaboración Gestionar Ingresos



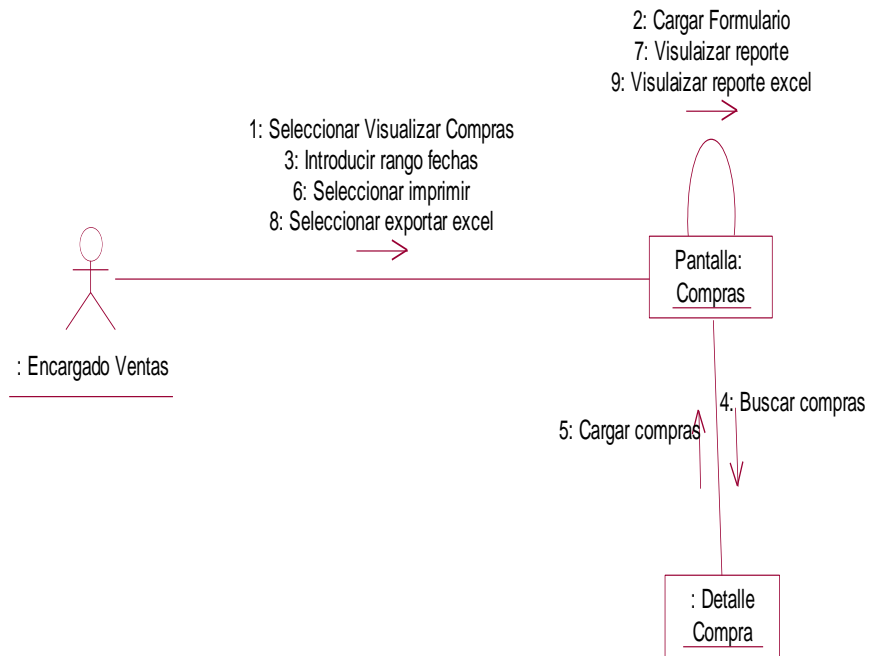
3.5.10 Diagrama Colaboración Gestionar Salidas



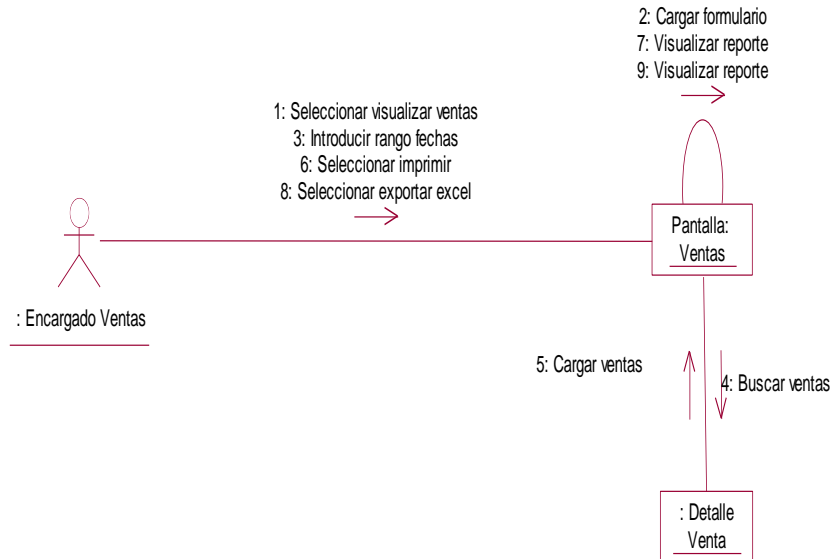
3.5.11 Diagrama Colaboración Visualizar Existencias



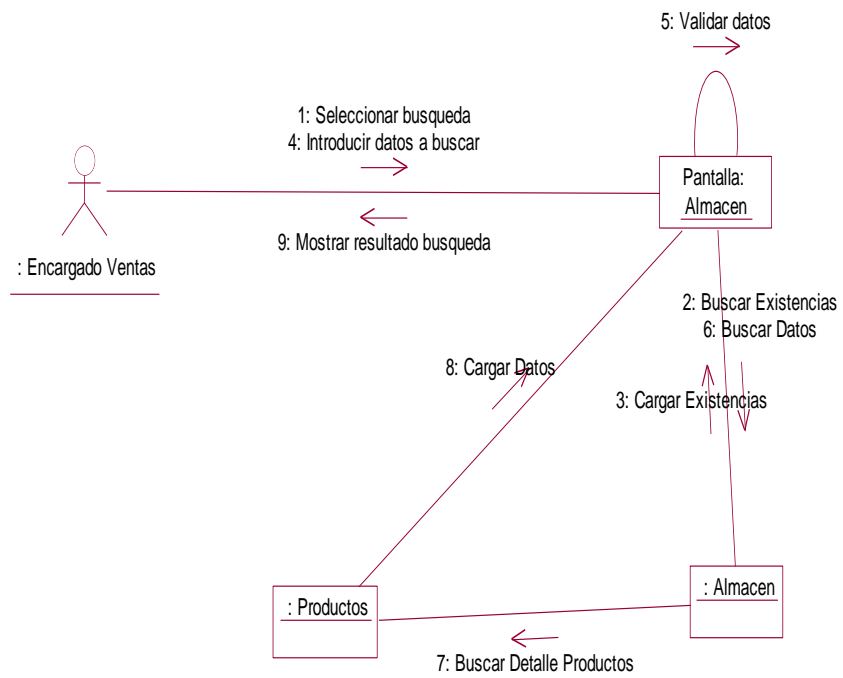
3.5.12 Diagrama Colaboración Visualizar Compras



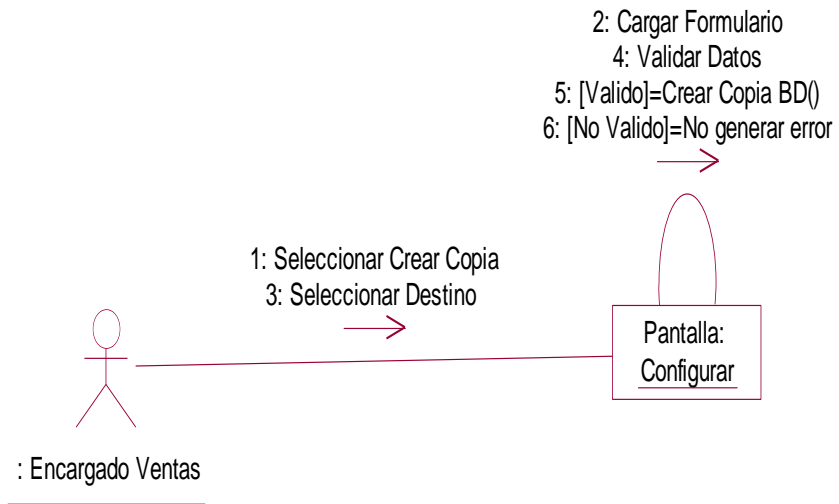
3.5.13 Diagrama Colaboración Visualizar Ventas



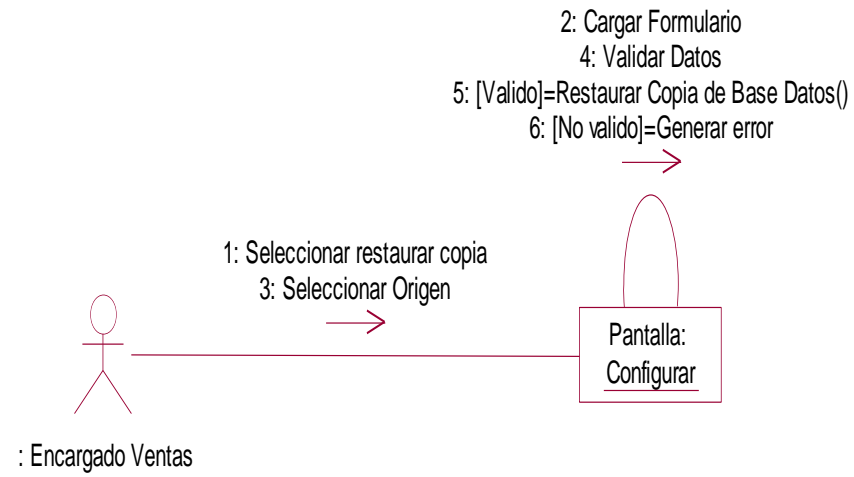
3.5.14 Diagrama Colaboración Realizar Búsquedas



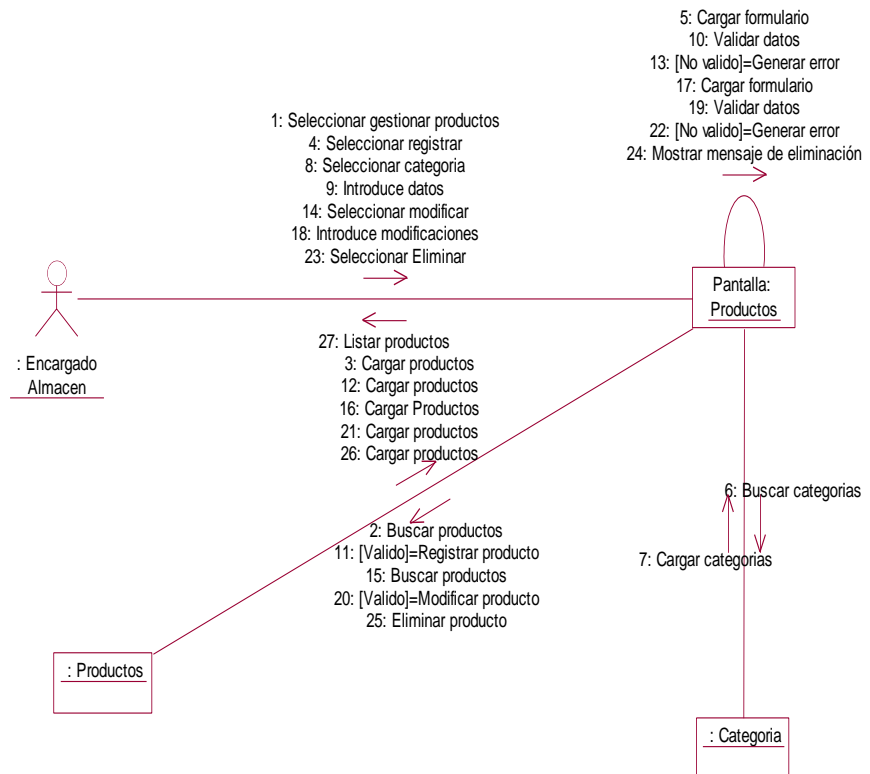
3.5.15 Diagrama Colaboración Crear Copia Seguridad



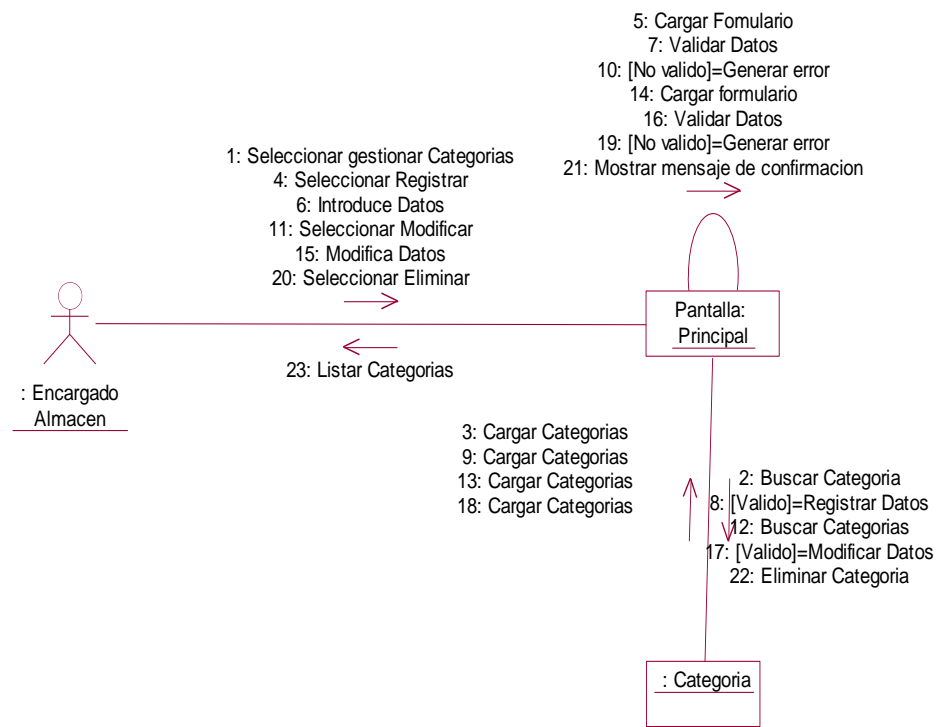
3.5.16 Diagrama Colaboración Restaurar Copia Seguridad



3.5.17 Diagrama Colaboración Gestionar Productos

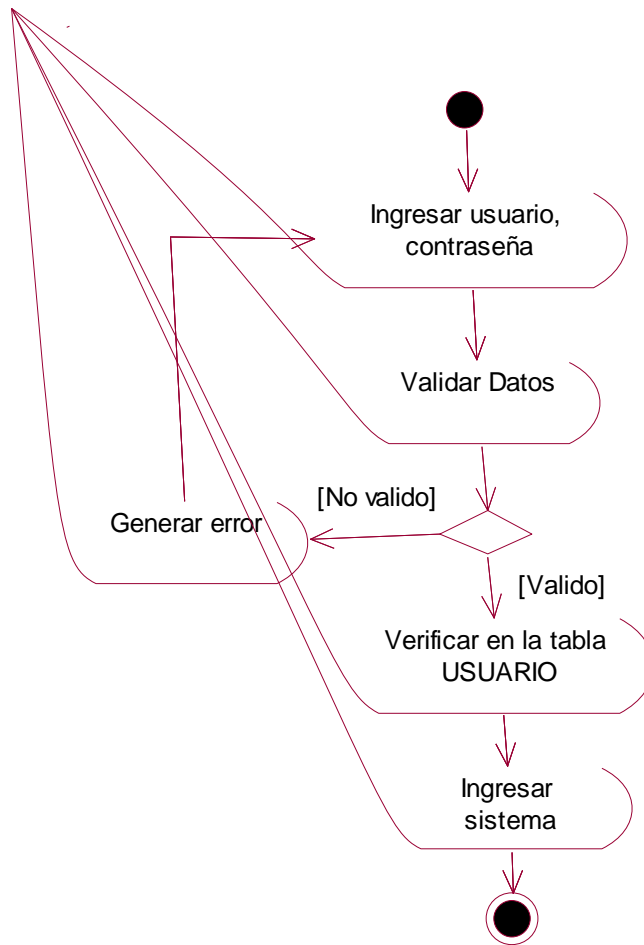


3.5.18 Diagrama Colaboración Gestionar Categorías

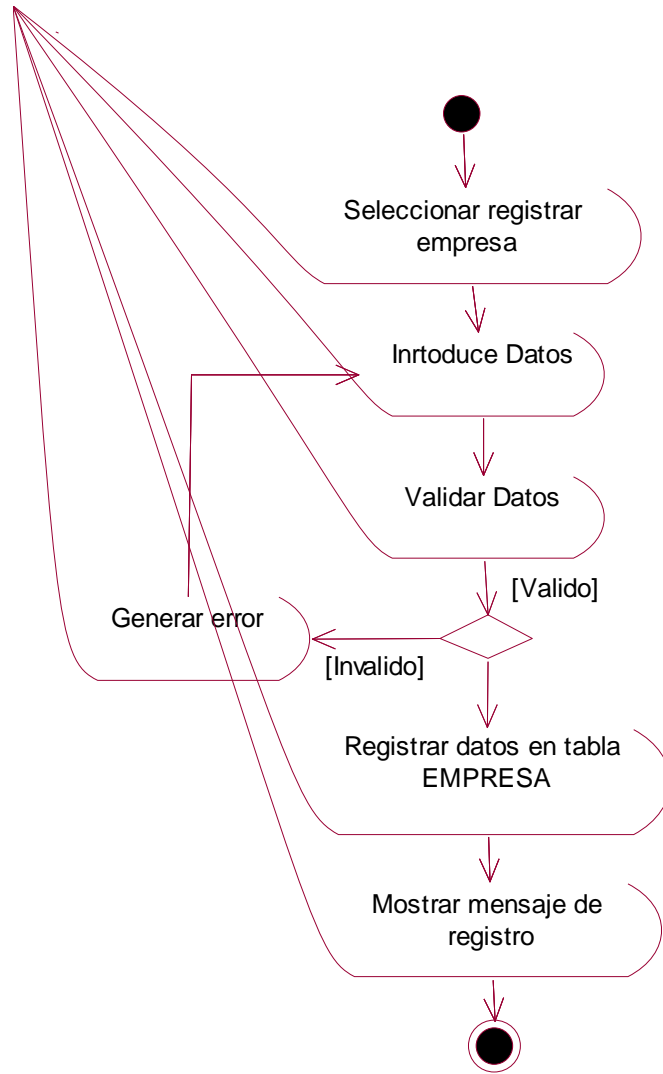


3.6 Diagrama Actividades

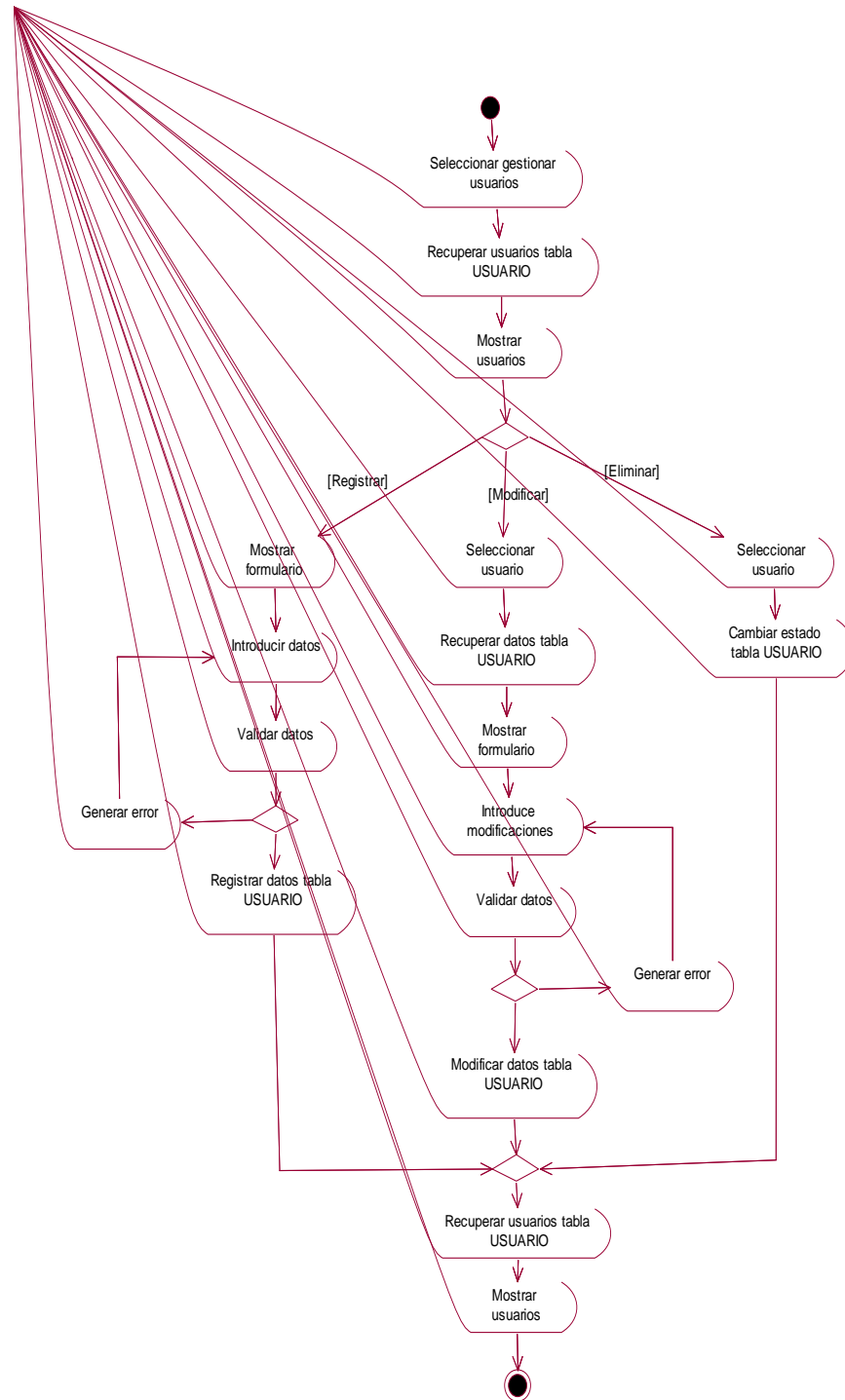
3.6.1 Diagrama Actividades Conectarse



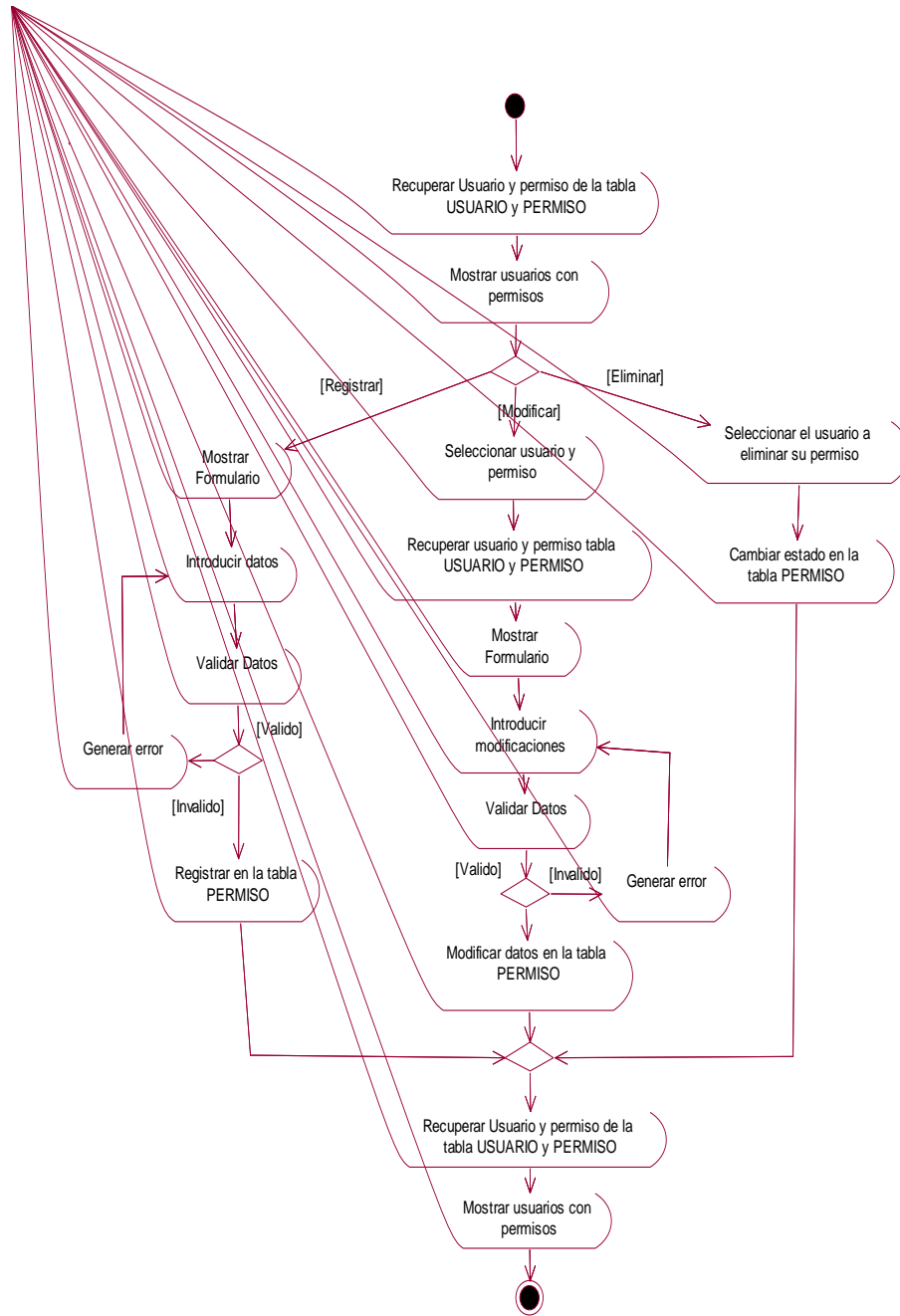
3.6.2 Diagrama Actividades Registrar Empresa



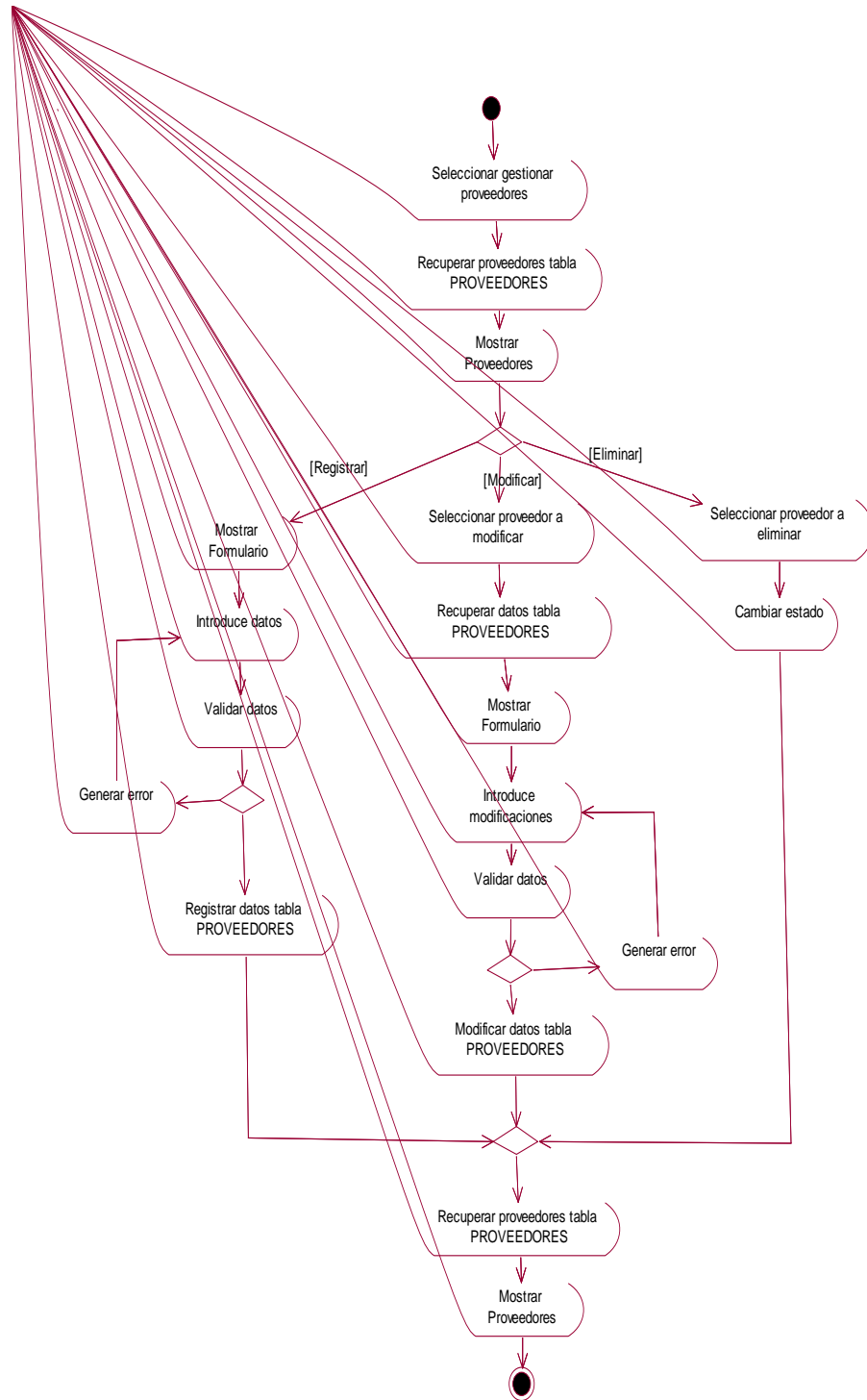
3.6.3 Diagrama Actividades Gestionar Usuarios



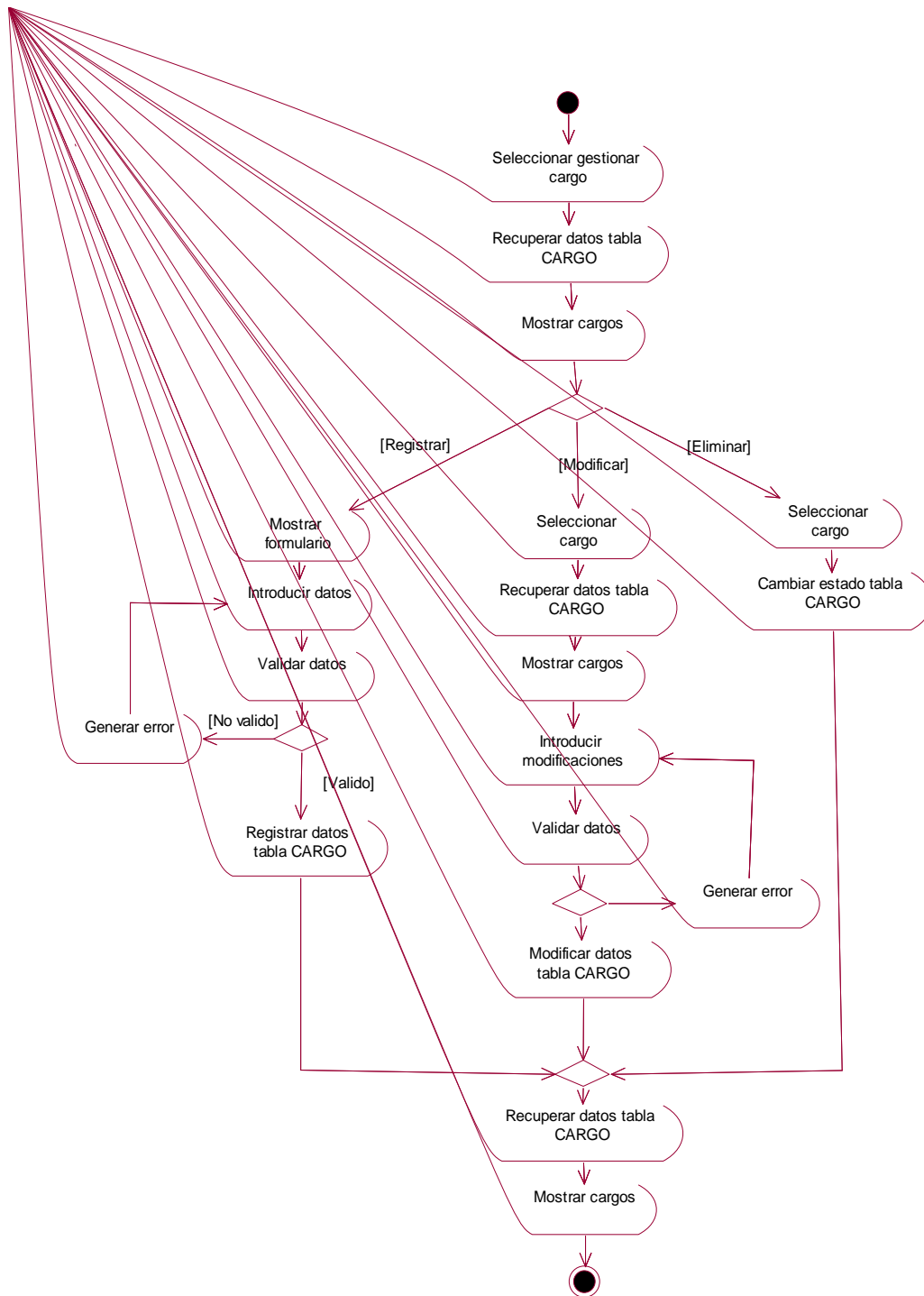
3.6.4 Diagrama Actividades Asignar Permisos



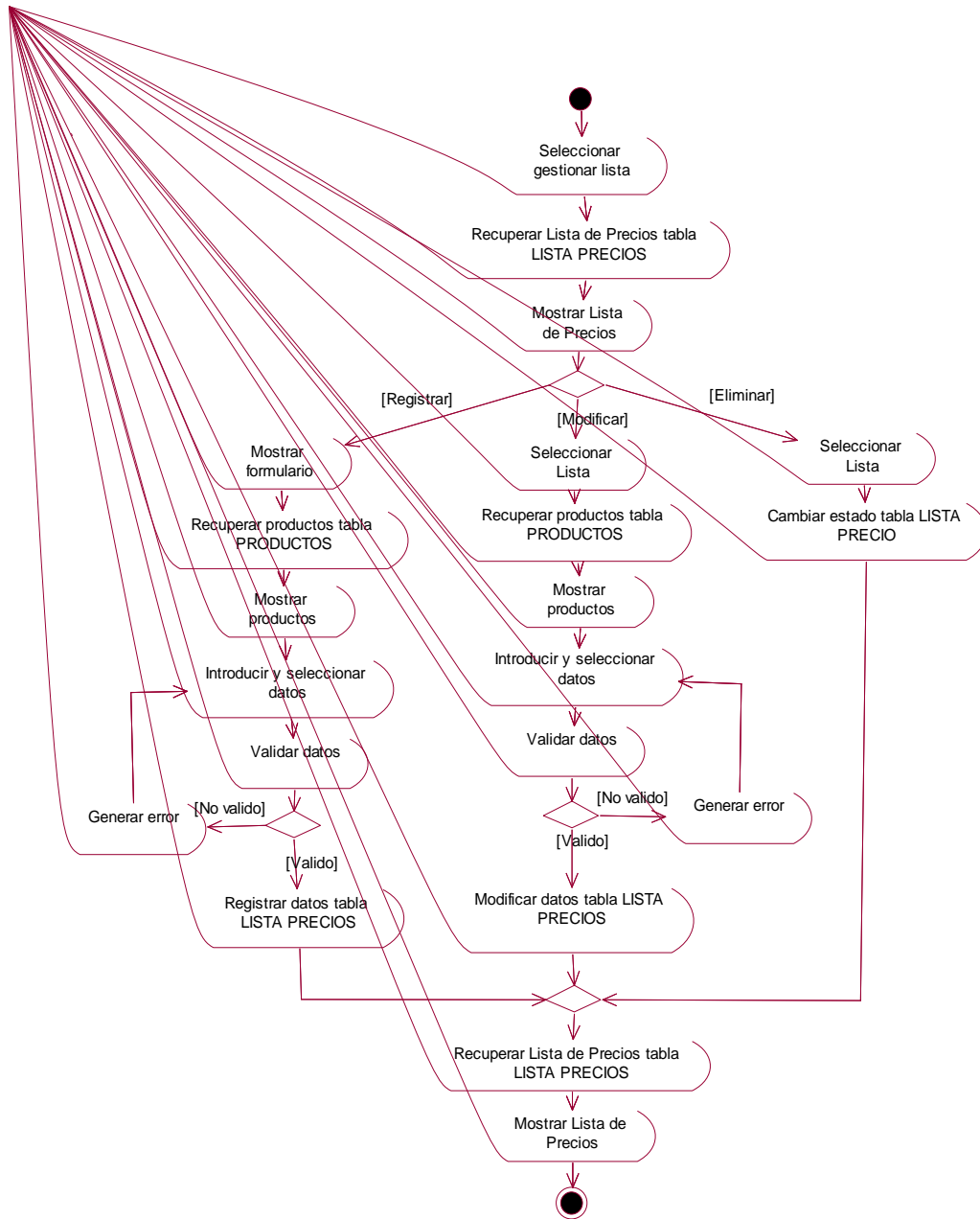
3.6.5 Diagrama Actividades Gestionar Proveedores



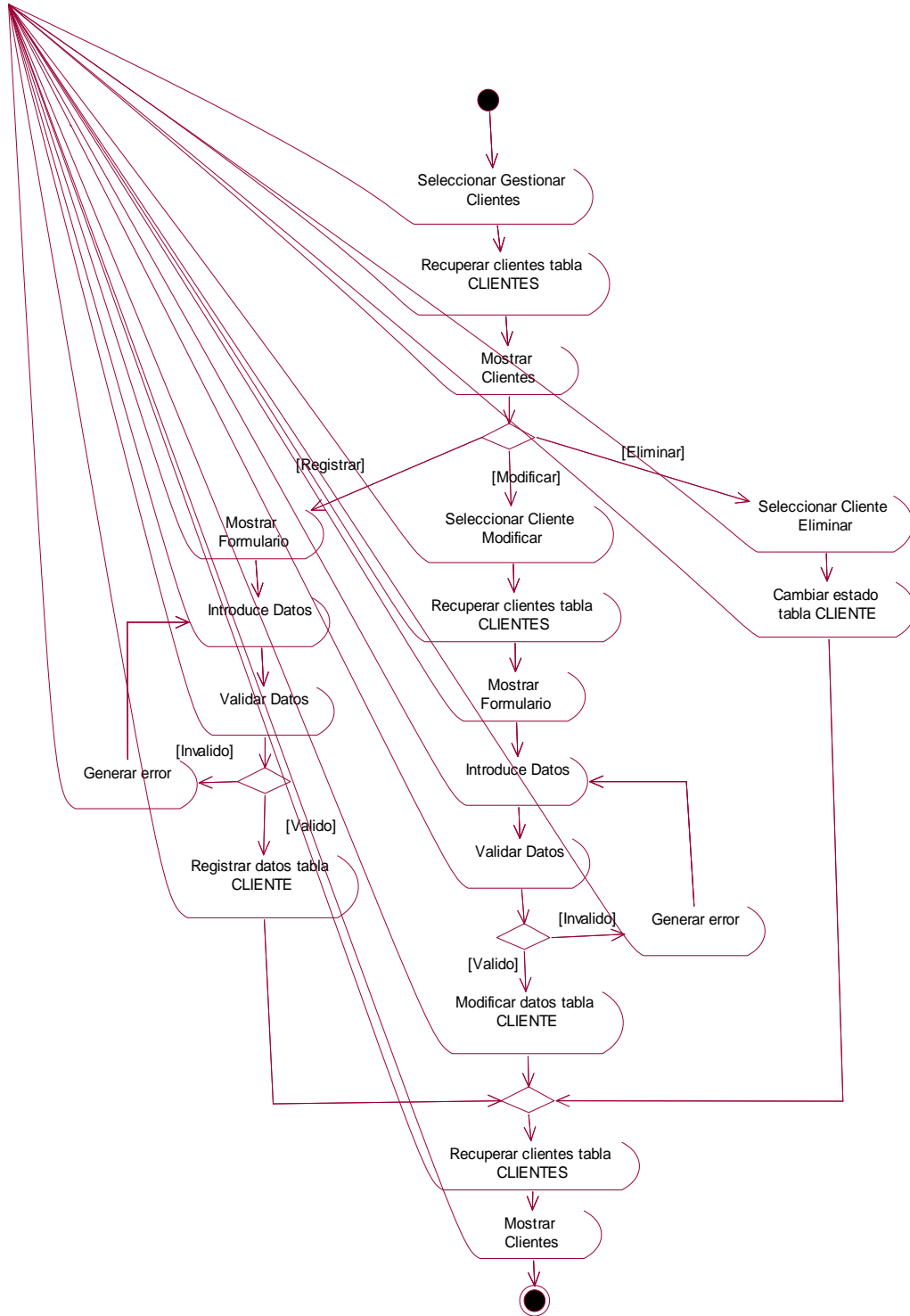
3.6.6 Diagrama Actividades Gestionar Cargo



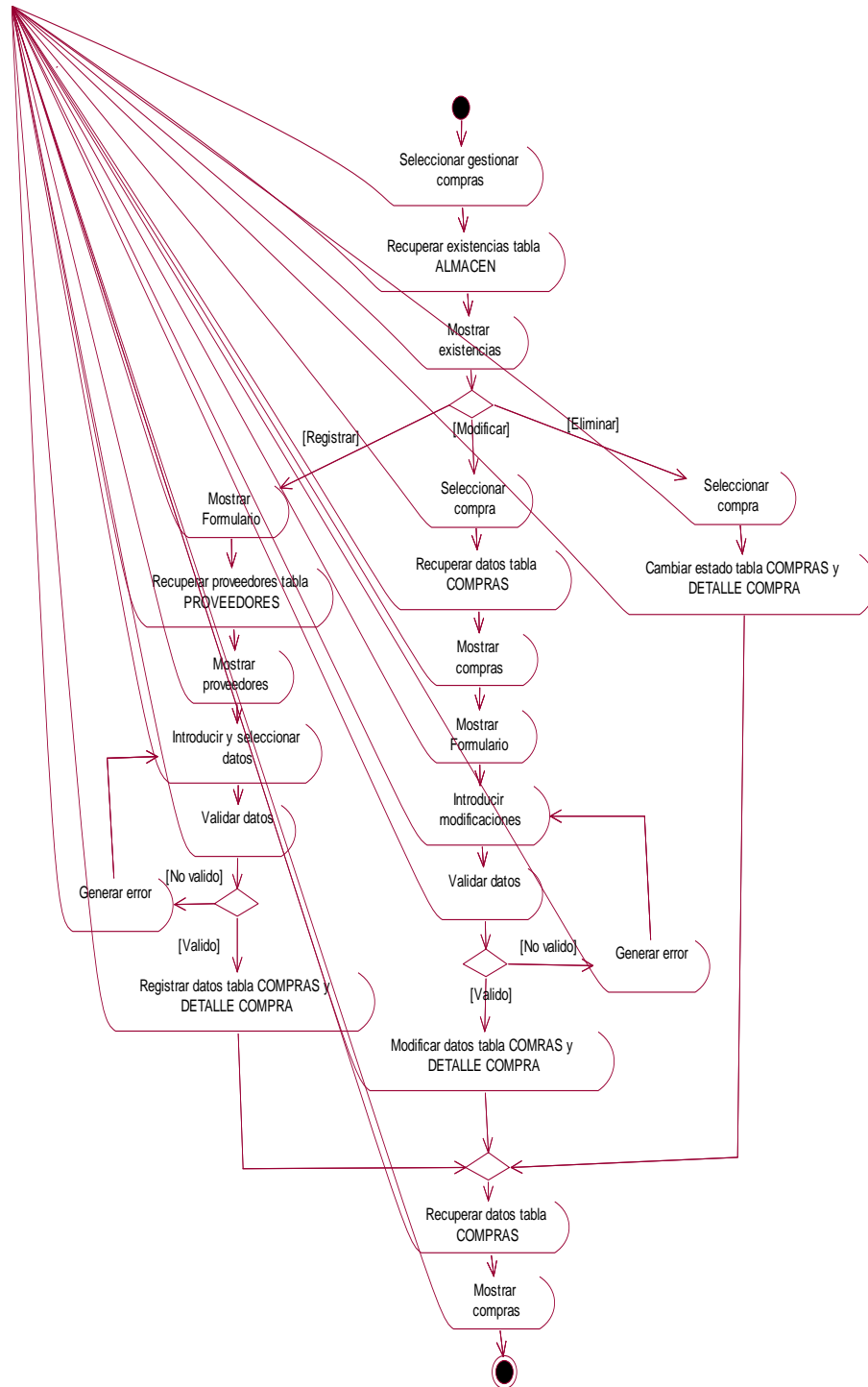
3.6.7 Diagrama Actividades Gestionar Lista de Precios



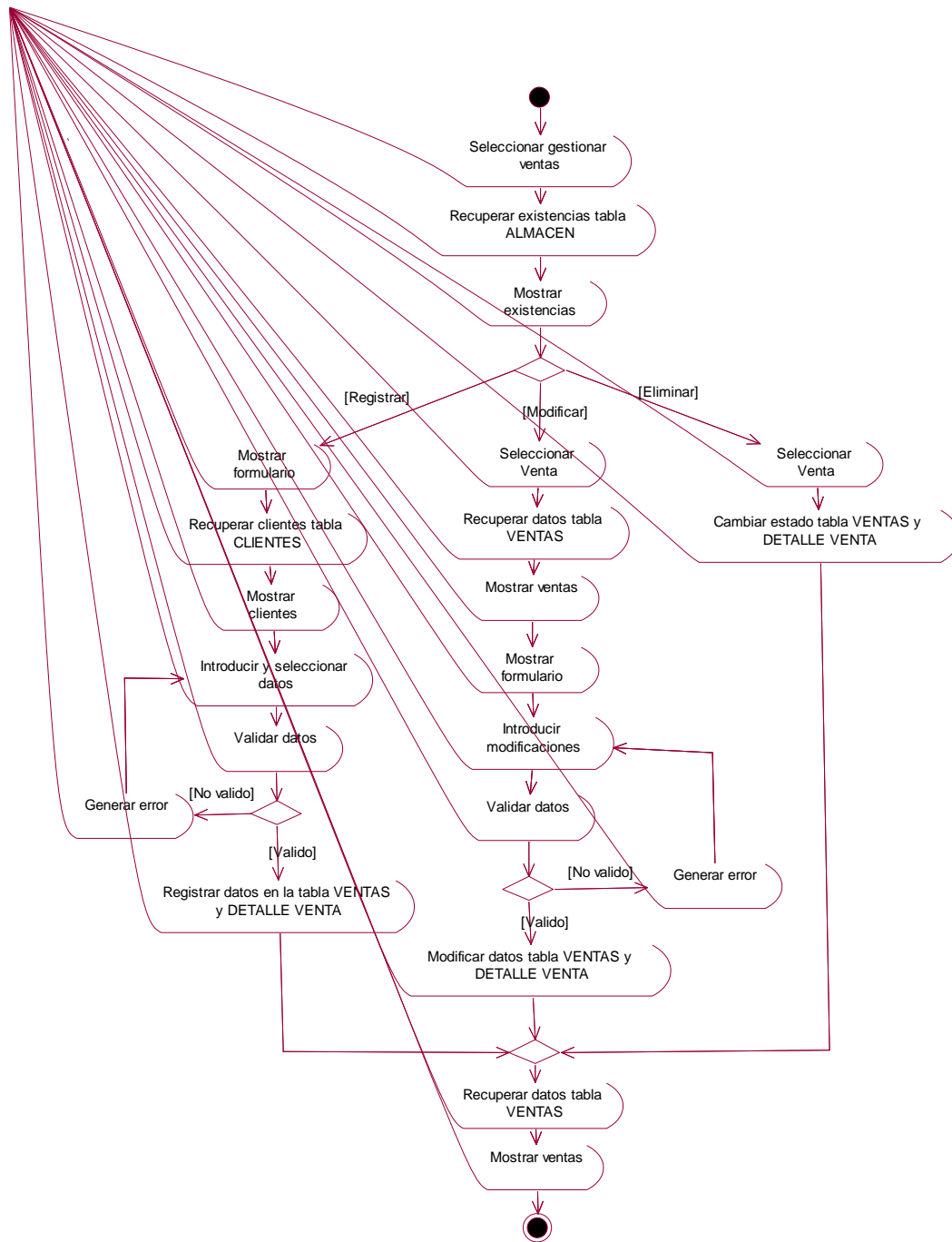
3.6.8 Diagrama Actividades Gestionar Cliente



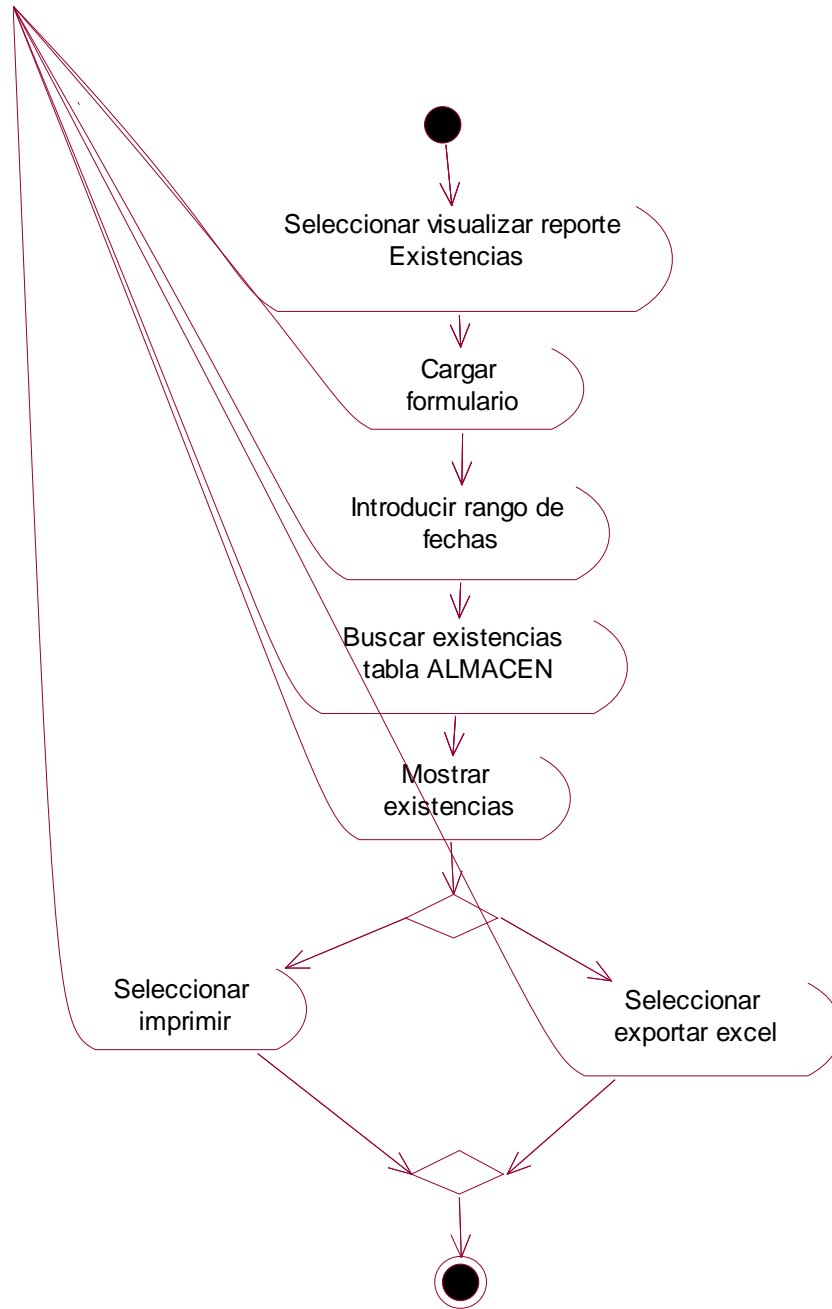
3.6.9 Diagrama Actividades Gestionar Ingresos



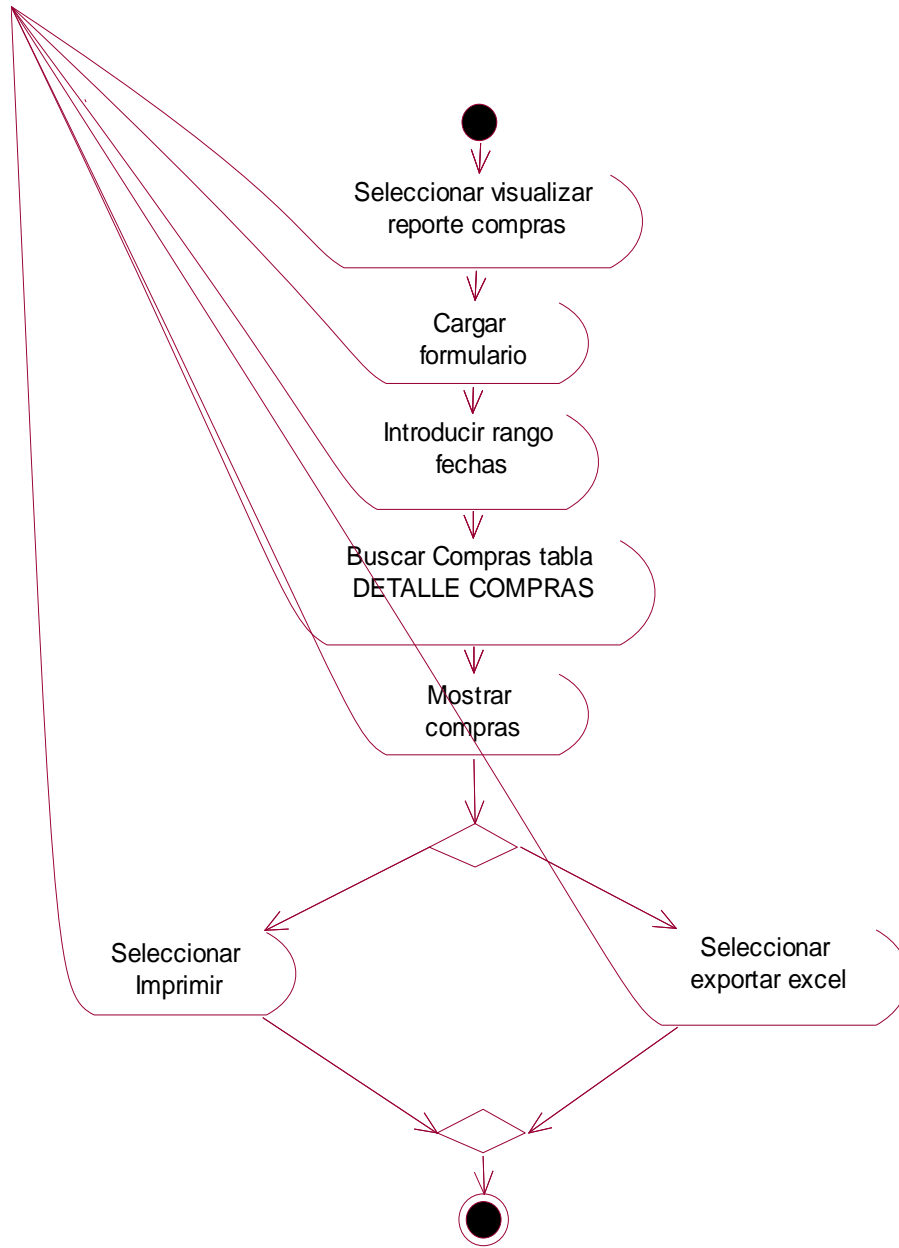
3.6.10 Diagrama Actividades Gestionar Salidas



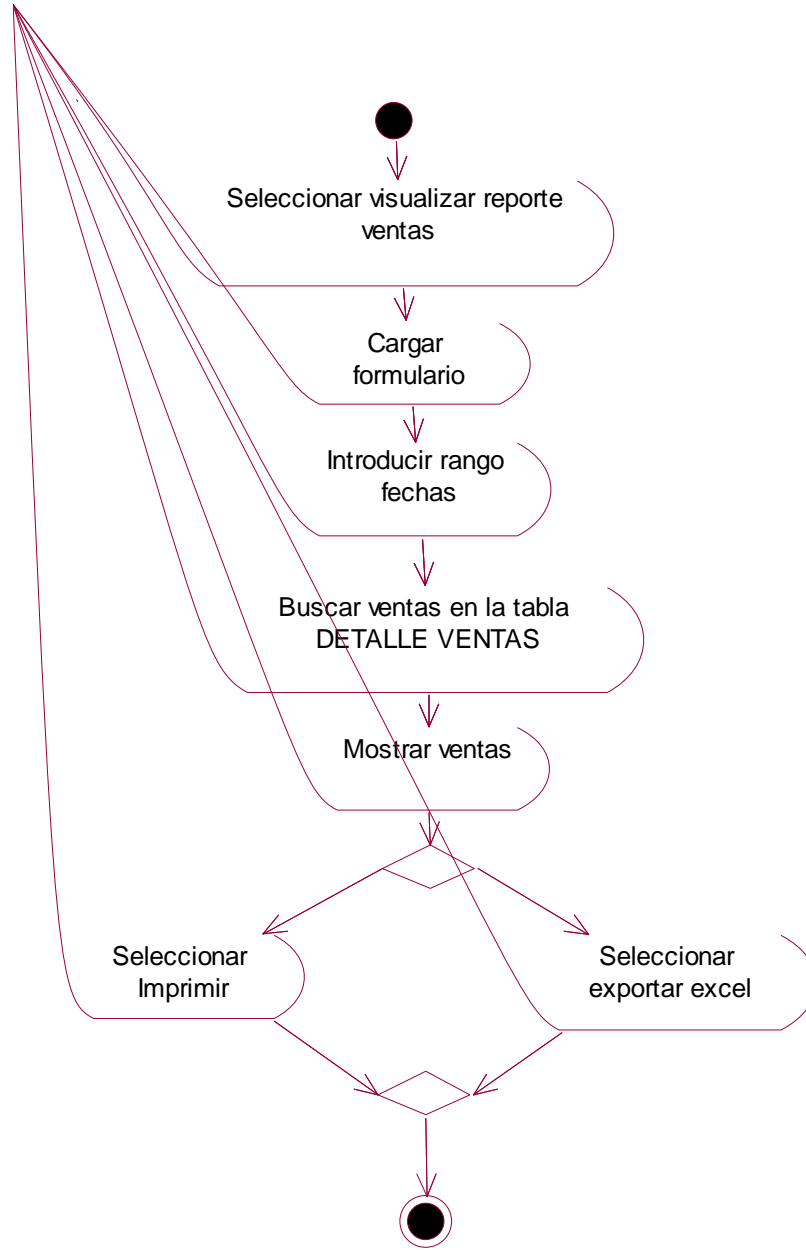
3.6.11 Diagrama Actividades Visualizar Existencias



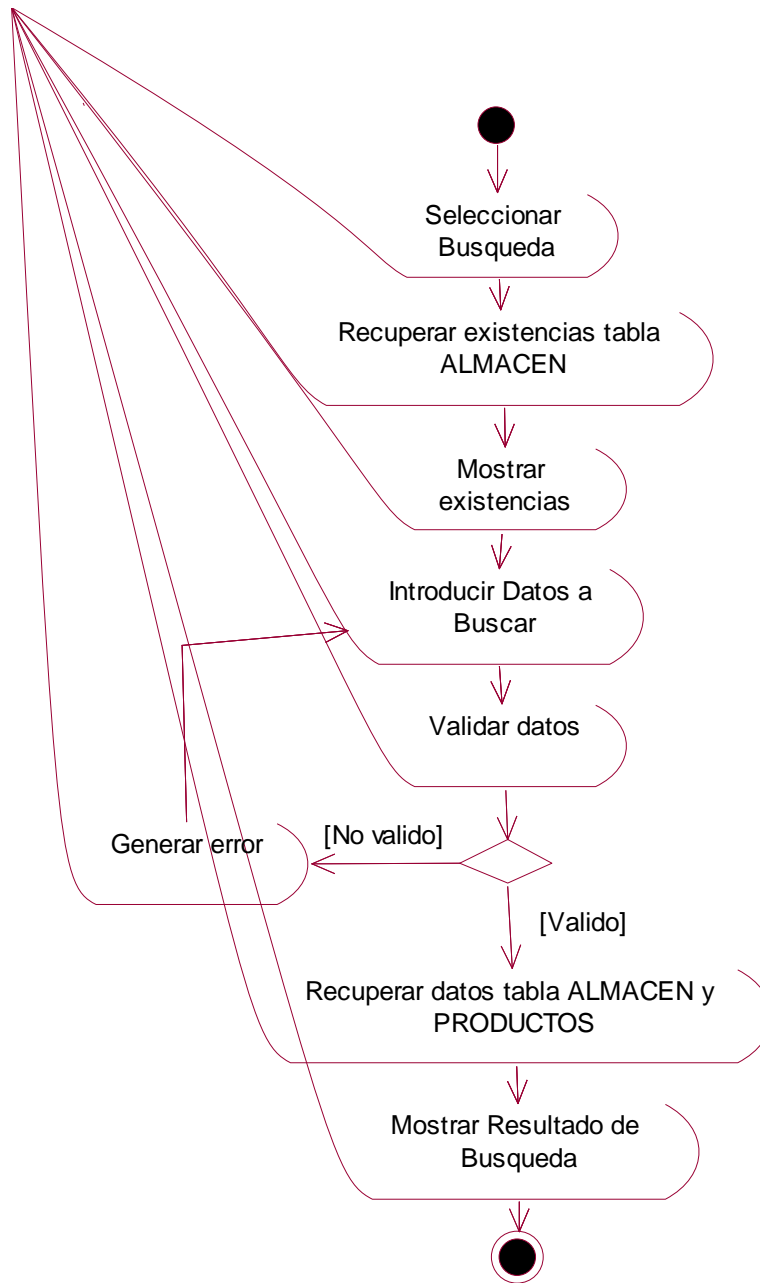
3.6.12 Diagrama Actividades Visualizar Compras



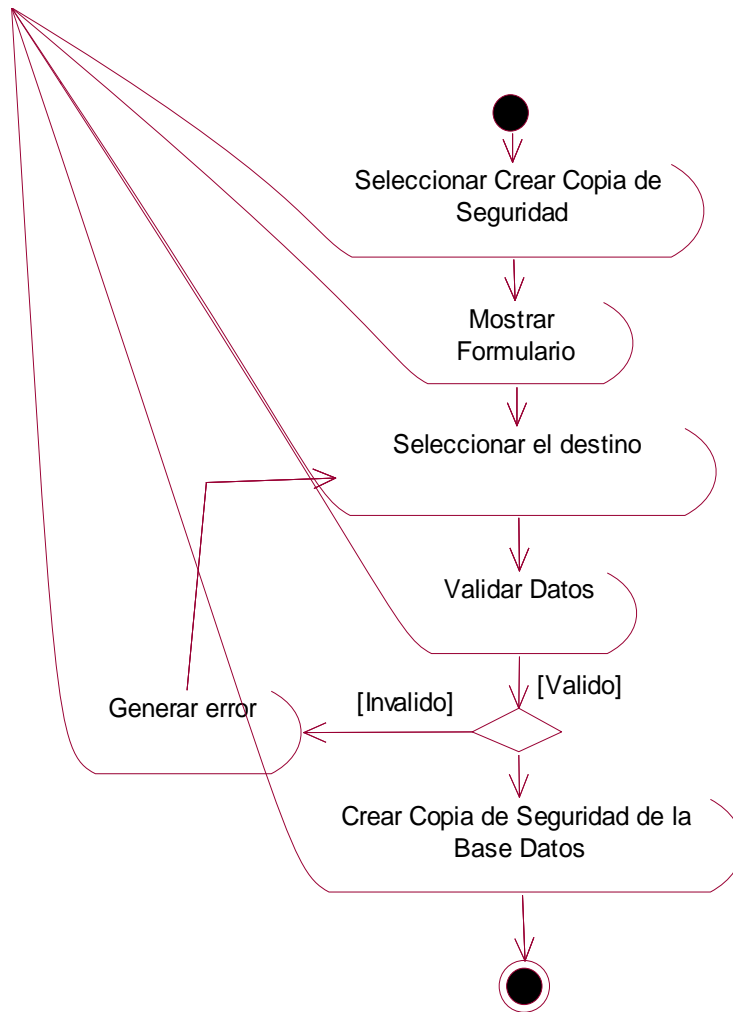
3.6.13 Diagrama Actividades Visualizar Ventas



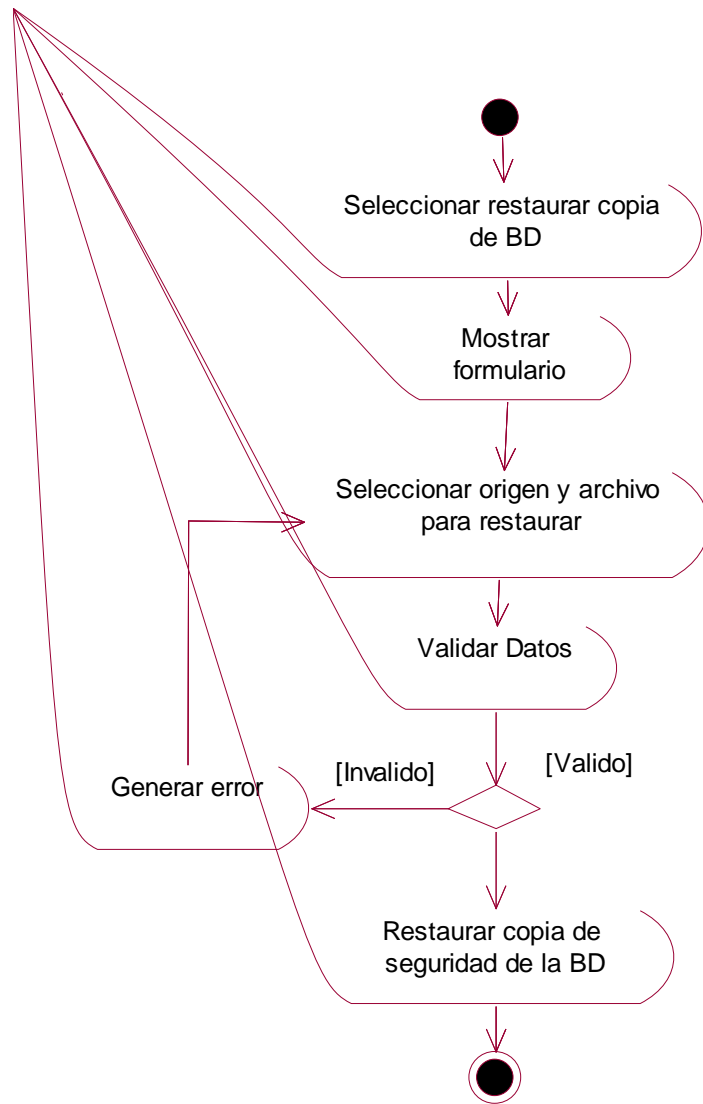
3.6.14 Diagrama Actividades Realizar Búsquedas



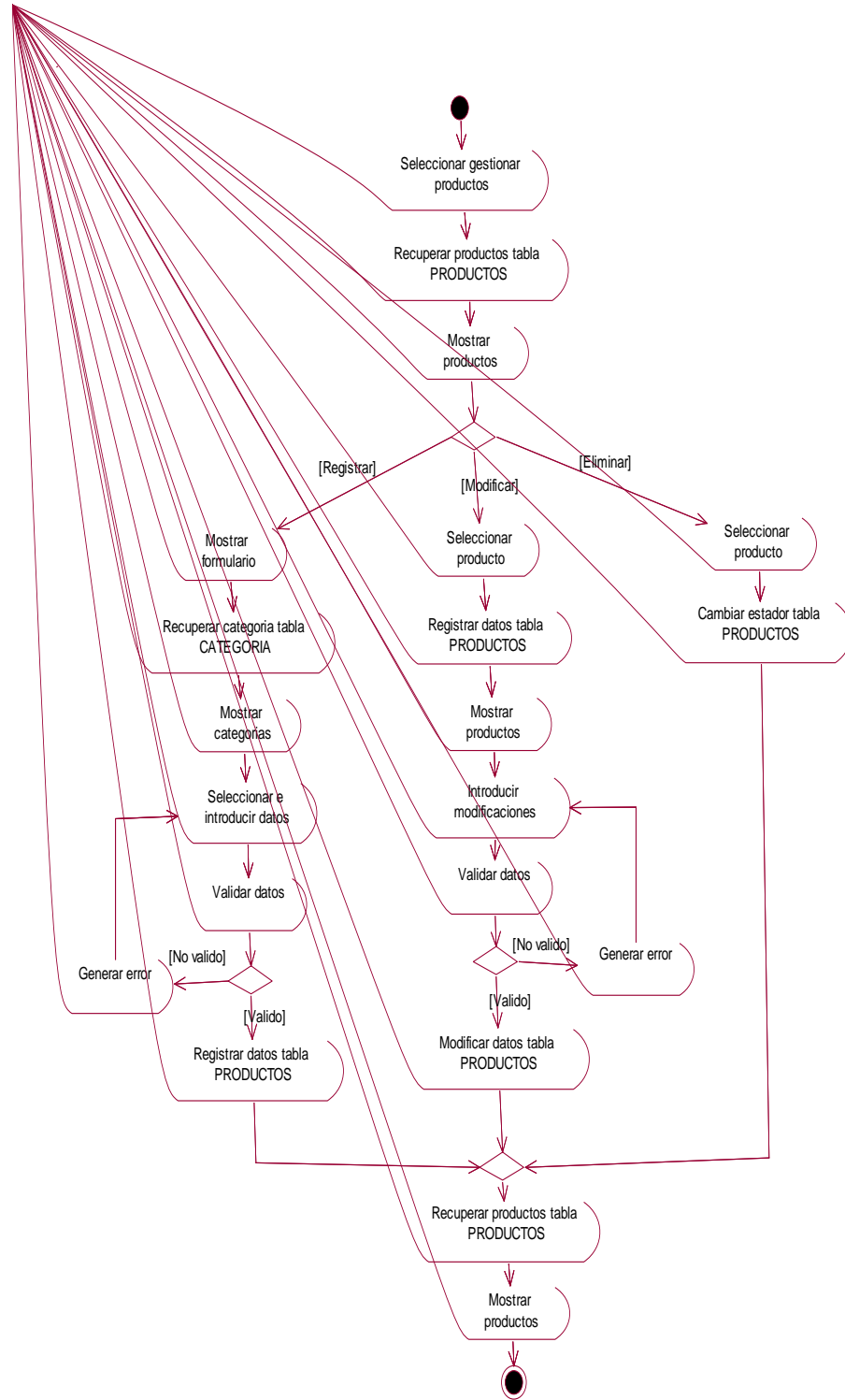
3.6.15 Diagrama Actividades Crear Copia Seguridad



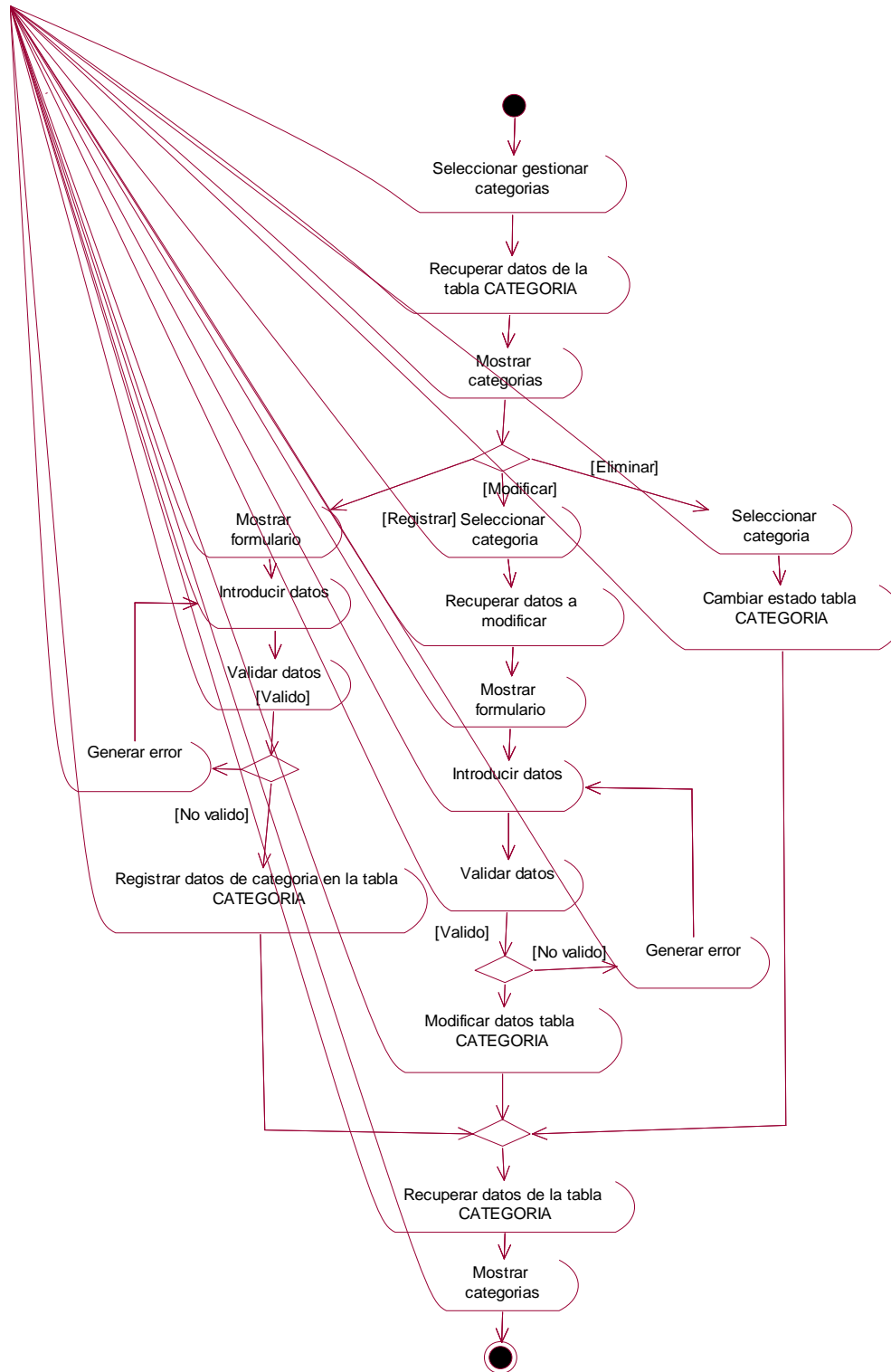
3.6.16 Diagrama Actividades Restaurar Copia Seguridad



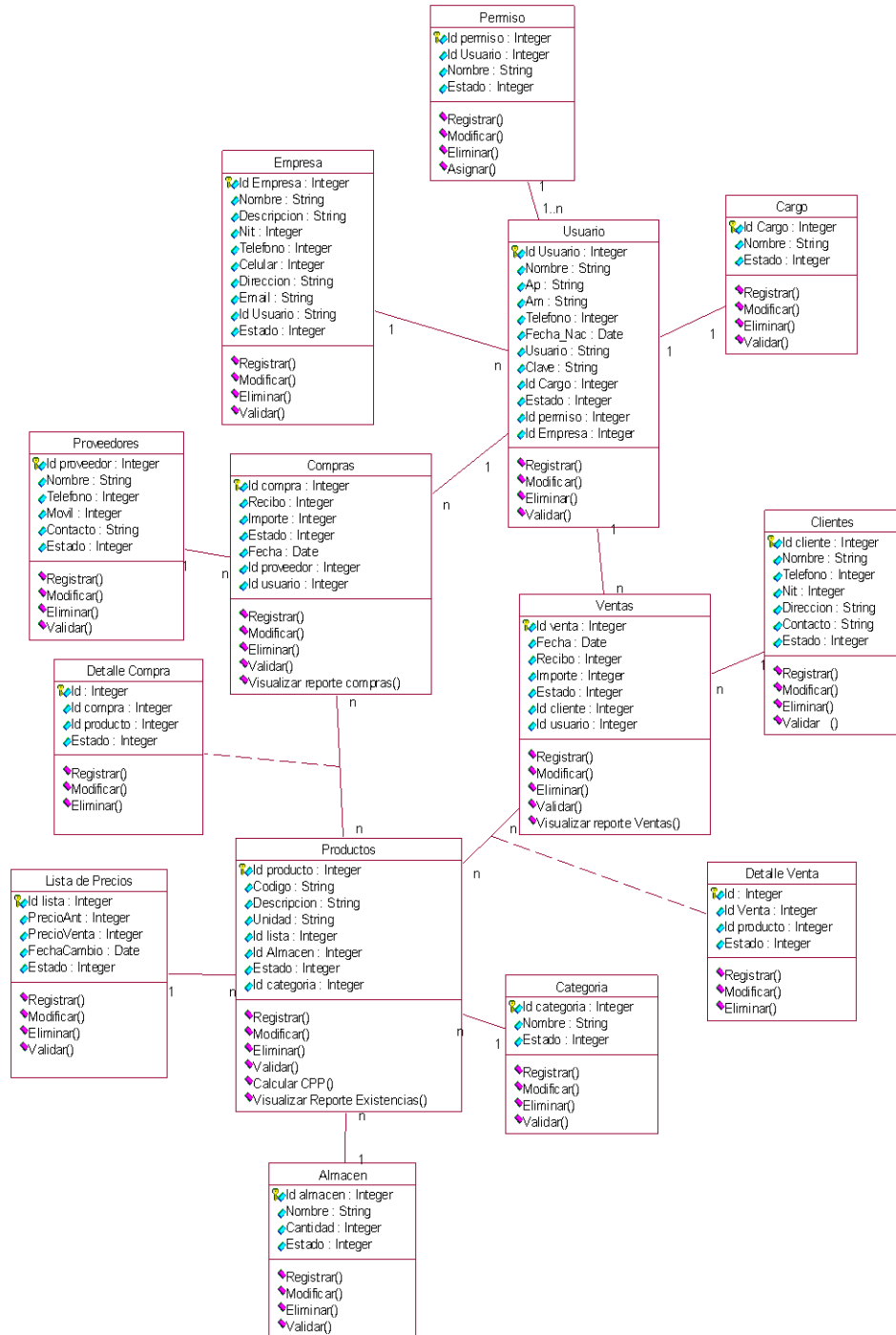
3.6.17 Diagrama Actividades Gestionar Productos



3.6.18 Diagrama Actividades Gestionar Categorías

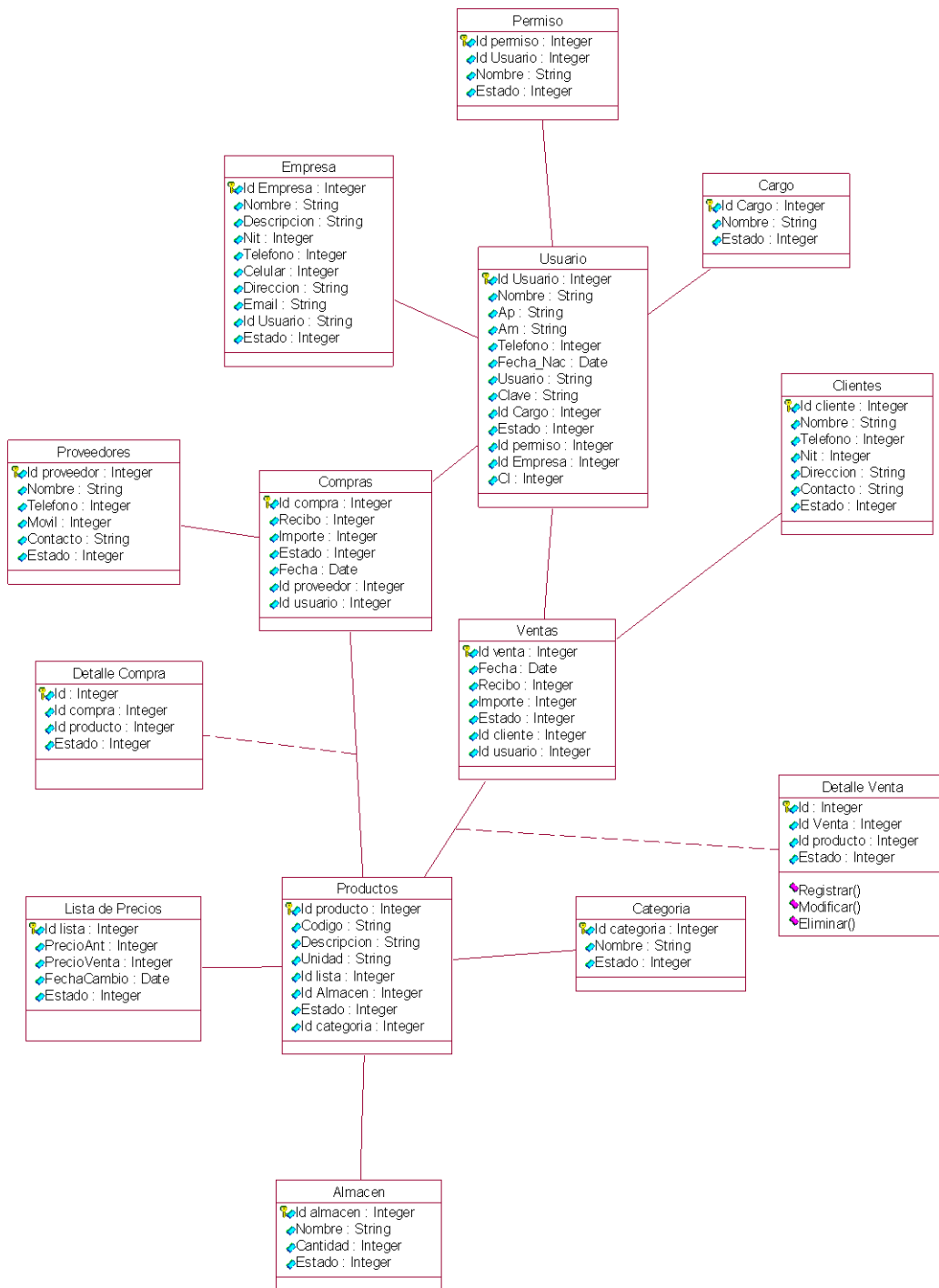


3.7 Diagrama Clases



4. Descripción de la Base de Datos

4.1 Diseño de la Base Datos



4.2 Descripción de las tablas

Nombre: Tabla Empresa			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_empresa	Int(11)	SI	NO
nombre	Varchar(100)	NO	NO
descripcion	Varchar(100)	NO	NO
nit	Int(10)	NO	NO
teléfono	Varchar(20)	NO	NO
direccion	varchar(100)	NO	NO
celular	int(20)	NO	NO
dirección	Varchar(20)	NO	NO
estado	char(1)	NO	NO

Nombre: Tabla Usuario			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_usuario	Int(11)	SI	NO
Id_cargo	int(11)	NO	SI
Id_permiso	int(11)	NO	SI
Id_empresa	Int(11)	NO	SI
ci	Varchar(15)	NO	NO
nombre	varchar(100)	NO	NO
ap	varchar(100)	NO	NO
am	Varchar(100)	NO	NO
Fecha_nac	date	NO	NO
telefono	Varchar(100)	NO	NO

usuario	text	NO	NO
clave	text	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Cargo			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_cargo	Int(11)	SI	NO
nombre	Varchar(100)	NO	NO
estado	char(1)	NO	NO

Nombre: Tabla Permiso			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
Id_permiso	int(11)	NO	SI
id_usuario	Int(11)	SI	NO
nombre	varchar(100)	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Proveedores			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_proveedor	Int(11)	SI	NO
nombre	varchar(100)	NO	NO
telefono	int(100)	NO	NO
celular	int(100)	NO	NO
contacto	Varchar(100)	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Clientes			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_cliente	Int(11)	SI	NO
nombre	varchar(100)	NO	NO
telefono	int(100)	NO	NO
celular	int(100)	NO	NO
nit	int(100)	NO	NO
direccion	varchar(100)	NO	NO
contacto	varchar(100)	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Categoría			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
Id_categoria	int(11)	NO	SI
nombre	varchar(100)	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Compras			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_compra	Int(11)	SI	NO
Id_usuario	int(11)	NO	SI
Id_proveedor	int(11)	NO	SI
recibo	Int(100)	NO	NO
importe	int(100)	NO	NO
fecha	date	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Ventas			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_ventas	Int(11)	SI	NO
Id_usuario	int(11)	NO	SI
Id_cliente	int(11)	NO	SI
recibo	Int(100)	NO	NO
importe	int(100)	NO	NO
fecha	date	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Almacén			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
Id_almacen	int(11)	NO	SI
nombre	varchar(100)	NO	NO
cantidad	int(100)	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Lista de Precios			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
Id_lista	int(11)	NO	SI
Precio_ant	int(100)	NO	NO
precioventa	int(100)	NO	NO
fechacambio	date	NO	NO
estado	Char(1)	NO	NO

Nombre: Tabla Detalle Compra			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id	Int(11)	SI	NO
Id_compra	int(11)	NO	SI
Id_producto	int(11)	NO	SI
estado	Char(1)	NO	NO

Nombre: Tabla Detalle venta			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id	Int(11)	SI	NO
Id_venta	int(11)	NO	SI
Id_producto	int(11)	NO	SI
estado	Char(1)	NO	NO

Nombre: Tabla Productos			
Campo	Tipo Dato	Llave Primaria	Llave Foránea
id_producto	Int(11)	SI	NO
Id_lista	int(11)	NO	SI
Id_almacen	int(11)	NO	SI
Id_categoria	Int(11)	NO	SI
codigo	Varchar(100)	NO	NO
descripcion	varchar(100)	NO	NO
unidad	varchar(100)	NO	NO
estado	Char(1)	NO	NO

4.3 Creación de la Base de Datos

Tabla Empresa

```
CREATE TABLE empresa (  
  id_empresa int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  descripción varchar(100),  
  nit int(10)  
  telefono int(20),  
  celular int(20),  
  direccion varchar(20),  
  email varchar(100),  
  _estado char(1) NOT NULL default 'A',  
  PRIMARY KEY (id_empresa)  
)TYPE=InnoDB;
```

Tabla Usuario

```
CREATE TABLE `usuario` (  
  `id_usuario` int(11) NOT NULL auto_increment,  
  id_cargo int(11) NOT NULL,  
  id_permiso int(11) NOT NULL,  
  id_empresa int(11) NOT NULL,  
  ci varchar(15) NOT NULL,  
  nombre varchar(100) NOT NULL,  
  ap varchar(100) NOT NULL,  
  am varchar(100) NOT NULL,  
  fecha_nac date,  
  telefono varchar(100) NOT NULL,  
  `usuario` text NOT NULL,  
  `clave` text NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_usuario`),  
  FOREIGN KEY `id_cargo` REFERENCES cargo (`id_cargo`)  
  FOREIGN KEY `id_permiso` REFERENCES permiso (`id_permiso`)  
  FOREIGN KEY `id_empresa` REFERENCES empresa (`id_empresa`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3;
```

Tabla Cargo

```
CREATE TABLE `cargo` (  
  id_cargo int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  _estado char(1) NOT NULL default 'A',  
  PRIMARY KEY (id_cargo)  
)TYPE=InnoDB;
```

Tabla Permiso

```
CREATE TABLE `permiso` (  
  `id_permiso` int(11) NOT NULL auto_increment,  
  `id_usuario` int(11) NOT NULL,  
  `nombre` varchar(100) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_permiso`),  
  FOREIGN KEY `id_usuario` REFERENCES usuario (`id_usuario`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

Tabla Proveedores

```
CREATE TABLE `proveedores` (  
  `id_proveedor` int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  telefono int(100) NOT NULL,  
  celular int(100) NOT NULL,  
  contacto varchar(100) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_proveedor`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3;
```

Tabla Clientes

```
CREATE TABLE `clientes` (  
  `id_cliente` int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  telefono int(100) NOT NULL,  
  celular int(100) NOT NULL,  
  nit int(100) NOT NULL,  
  direccion varchar(100) NOT NULL,  
  contacto varchar(100) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_cliente`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3;
```

Tabla Categoria

```
CREATE TABLE `categoria` (  
  id_categoria int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (id_categoria)  
)TYPE=InnoDB;
```

Tabla Compras

```
CREATE TABLE `compras` (  
  `id_compra` int(11) NOT NULL auto_increment,  
  `id_usuario` int(11) NOT NULL,  
  `id_proveedor` int(11) NOT NULL,  
  `recibo` int(100) NOT NULL,  
  `importe` int(100) NOT NULL,  
  `fecha` date,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_compra`),  
  FOREIGN KEY `id_usuario` REFERENCES usuario (`id_usuario`),  
  FOREIGN KEY `id_proveedor` REFERENCES proveedores (`id_proveedor`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

Tabla Ventas

```
CREATE TABLE `ventas` (  
  `id_venta` int(11) NOT NULL auto_increment,  
  `id_usuario` int(11) NOT NULL,  
  `id_cliente` int(11) NOT NULL,  
  `recibo` int(100) NOT NULL,  
  `importe` int(100) NOT NULL,  
  `fecha` date,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_venta`),  
  FOREIGN KEY `id_usuario` REFERENCES usuario (`id_usuario`),  
  FOREIGN KEY `id_cliente` REFERENCES clientes (`id_cliente`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

Tabla Almacen

```
CREATE TABLE `almacen` (  
  id_almacen int(11) NOT NULL auto_increment,  
  nombre varchar(100) NOT NULL,  
  cantidad int(100) NOT NULL,  
  _estado char(1) NOT NULL default 'A',  
  PRIMARY KEY (id_almacen)  
)TYPE=InnoDB;
```

Tabla Lista de Precios

```
CREATE TABLE `lista de precios` (  
  id_lista int(11) NOT NULL auto_increment,  
  precio_ant int(100) NOT NULL,  
  precioventa int(100) NOT NULL,  
  fechacambio date,  
  _estado char(1) NOT NULL default 'A',  
  PRIMARY KEY (id_lista)  
)TYPE=InnoDB;
```

Tabla Detalle Compra

```
CREATE TABLE `detalle compra` (  
  `id` int(11) NOT NULL auto_increment,  
  `id_compra` int(11) NOT NULL,  
  `id_producto` int(11) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id`),  
  FOREIGN KEY `id_compra` REFERENCES compras (`id_compra`),  
  FOREIGN KEY `id_producto` REFERENCES productos (`id_producto`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

Tabla Detalle Venta

```
CREATE TABLE `detalle venta` (  
  `id` int(11) NOT NULL auto_increment,  
  `id_venta` int(11) NOT NULL,  
  `id_producto` int(11) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id`),  
  FOREIGN KEY `id_venta` REFERENCES ventas (`id_compra`),  
  FOREIGN KEY `id_producto` REFERENCES productos (`id_producto`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

Tabla Productos

```
CREATE TABLE `productos` (  
  `id_producto` int(11) NOT NULL auto_increment,  
  `id_lista` int(11) NOT NULL,  
  `id_almacen` int(11) NOT NULL,  
  `id_categoria` int(11) NOT NULL,  
  `codigo` varchar(100) NOT NULL,  
  `descripcion` varchar(100) NOT NULL,  
  `unidad` varchar(100) NOT NULL,  
  `_estado` char(1) NOT NULL default 'A',  
  PRIMARY KEY (`id_producto`),  
  FOREIGN KEY `id_lista` REFERENCES lista de precios (`id_lista`),
```

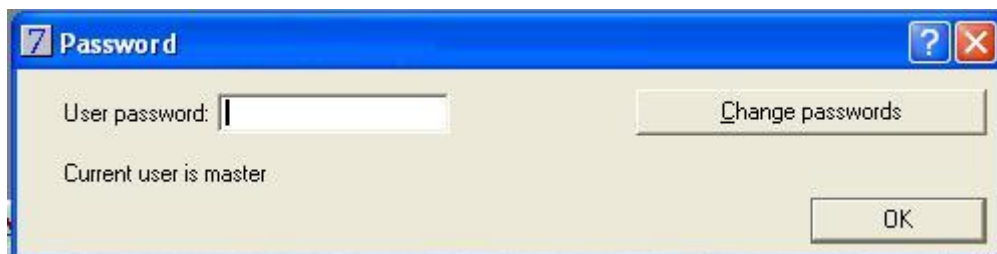
```
FOREIGN KEY `id_almacen` REFERENCES almacen (`id_almacen`),  
FOREIGN KEY `id_categoria` REFERENCES categoria (`id_categoria`),  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=2;
```

4.4 Diseño de Interfaces

Pantalla N°1: Principal



Pantalla N°2: Conectarse



En esta pantalla el usuario ingresa su nombre y clave para ingresar al sistema.

Pantalla N°3: Registrar Empresa

Modificar Especificaciones de la Empresa.

Nombre o Razón Social:	SITEC S.R.L	
Descripción:	Empresa Comercial	
Dirección:	Av Tarija	
Teléfono:	69614444	Fax: 69614398
e-mail:	SITEC@GMAIL.COM	
Web:	-	
Registro Tributario:	N.I.T. 1920702025	Sucursal: 0
# de Autorización:	[Yellow Box]	
Inicio de Gestión:	01-01-2009	dd-mm-aa

Usuario: Mi EmpresaPPPP

Acceptar / Cancelar

Pantalla donde se registra los datos de la empresa.

Pantalla N°4: Gestionar Usuarios

Usuarios ...

Opciones

	Código	Descripción
Regresar		
Insertar		
Modificar		
Eliminar		
Imprimir		

Pantalla para gestionar los usuarios registrar, modificar y eliminar los datos de los usuarios del sistema.

Pantalla N°5: Asignar Permisos

Configuración De Accesos De Los Usuarios

		Usuarios:				
		1	2	3	4	
Inventario	Acceso	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Aceptar <input type="checkbox"/> Cancelar
	Desbloquear Registros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Reportes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Compras	Acceso	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Facturación Manual	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Desbloquear Registros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Anular Registros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Reportes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Ventas	Acceso	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Facturación Manual	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Facturación Computarizada	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Desbloquear Registros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Anular Registros	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
	Reportes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Pantalla para asignar los permisos a los usuarios de acuerdo al rol que desempeña cada empleado.

Pantalla N°6: Gestionar Proveedores

NUEVA Definición de Proveedor...

Código : 001

Nombre o Razón Social :

Dirección :

Lugar : N.I.T. :

Teléfono/Fax :

Responsable :

Teléfono/Fax :

Observaciones:

NUEVA Definición de Proveedor...

Código: 001

Nombre o Razón Social: CREATIVO COMPUTACION Y SISTEMAS

Dirección: CALLE MEXICO Nº 1411

Lugar: LA PAZ N.I.T.: 0

Teléfono/Fax: 2311277

Responsable: VICTOR TRUJILLANO

Teléfono/Fax:

Observaciones:

Pantalla donde se registrar, modificar eliminar los datos de los proveedores de los productos que ofrece la empresa.

Pantalla N°7: Gestionar Lista de Precios

Cálculo Automático del Precio de Venta...

Descripción: 1 Monitor LCD SURE 17" Nuevo Modelo LCD HM1721

Tipo Cambio: 6.96 Bs./\$us.

	\$us.	Bs.	
P.U. Compra: a			
Utilidad (%): b	25.0		
Impuestos a la Venta "A" (%): c	13.0		
P.U. Venta "A":			$= a \times \frac{100 + b}{100 - c}$
Impuestos a la Venta "B" (%): d	5.0		
P.U. Venta "B":			$= a \times \frac{100 + b}{100 - d}$

P.U. Actuales		P.U. Nuevos	
P.U. Compra:		P.U. Compra:	
	\$us.		\$us.
P.U. Venta:		P.U. Venta:	
1:	Bs.	1:	0.00 Bs.
2:	Bs.	2:	0.00 Bs.
3:	\$us.	3:	0.00 \$us.
4:	\$us.	4:	\$us.

Cerrar

Guardar Y Continuar

NO Guardar Y Continuar

Pantalla donde se registrar, modificar o eliminar los datos de los precios de los productos.

Pantalla N°8: Gestionar Clientes

The screenshot shows a software window titled "Insertando Nuevo Registro De Cliente...". The window contains the following elements:

- Código :** A text box containing the value "464".
- Nombre :** A text box that is currently empty.
- Ciudad :** A text box that is currently empty.
- Dirección :** A text box that is currently empty.
- No. N.I.T. :** A text box that is currently empty.
- Telefono/Fax :** A text box that is currently empty.
- Observaciones :** A large text area for notes, currently empty.
- Buttons:** On the right side, there are two buttons: "Aceptar" (with a green checkmark icon) and "Cancelar" (with a red X icon).
- Radio Buttons:** At the bottom, there are two radio buttons: "Activo" (which is selected) and "Retirado".

Pantalla para registrar, modificar o eliminar los datos de un cliente.

Pantalla N°10: Gestionar Salidas

Detalle De Items Del Recibo De Salida...

		Descripción	Cantidad	
1	2	Mb Asrock 4coreDual-Sata2 Fsb1066/800/533ddrII66	1.00	Pza
2	3	Monitor Lcd Sure 16" (15.4" Viewable) Sr-L15wm Nue	1.00	Pza

Reporte en Pantalla Clic en "P" para Enviar a Impresora o Página Previa ----->

RECIBO DE SALIDA
SITEC S.R.L.

Centro de Costo: 001 Venta # Doc.: # Salida: **1**
Destinatario: Sitec Fecha: **20-11-2012**

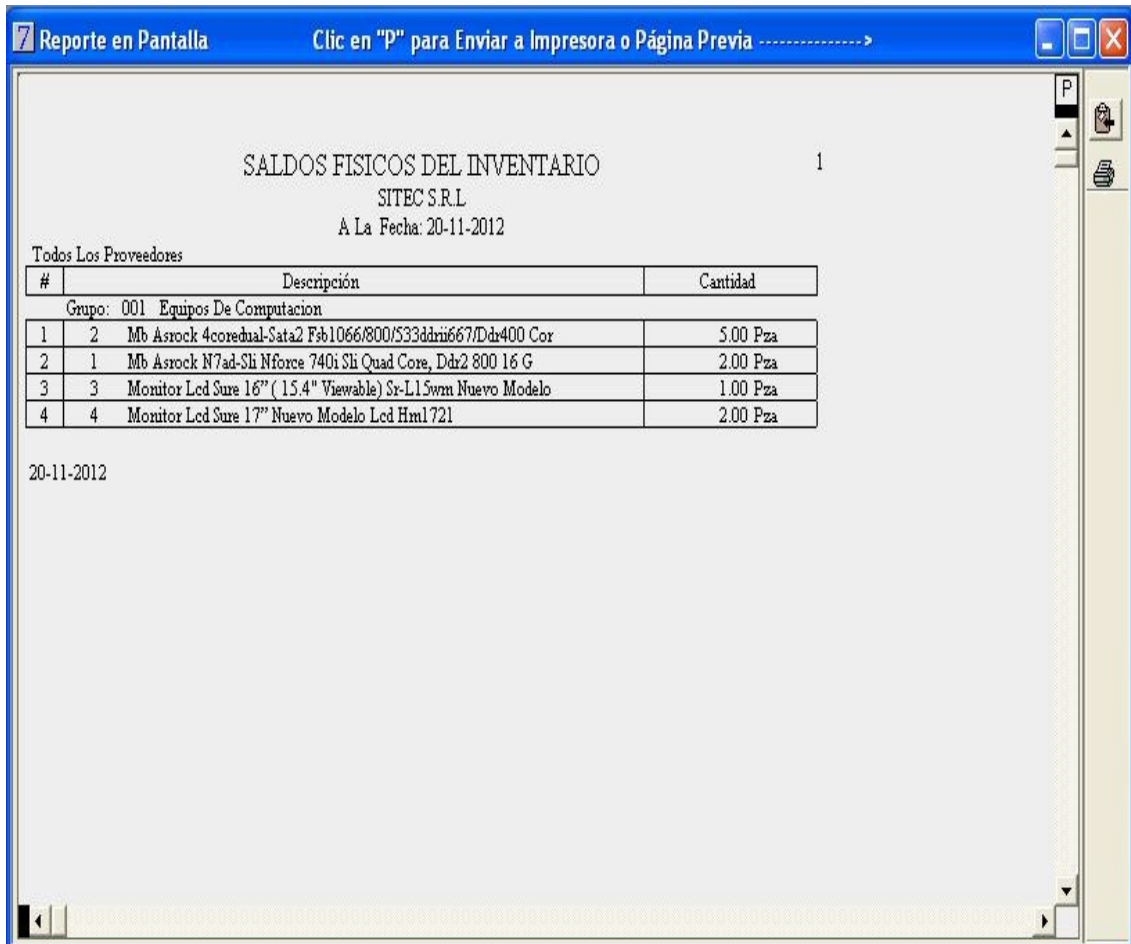
Formato Físico-Valorado (Bs.) D E T A L L E

	Descripción	P.U. Bs.	Cantidad	Importe Bs.
1	2 Mb Asrock 4coreDual-Sata2 Fsb1066/800/533ddrII667/D...	280.000	1.00 Pza	280.00
2	3 Monitor Lcd Sure 16" (15.4" Viewable) Sr-L15wm Nue...	200.000	1.00 Pza	200.00
3				
4				
22-11-2012			Total:	480.00

Son: Cuatrocientos Ochenta 00/100 Bolivianos

Pantalla donde se registrar, modificar y eliminar las ventas de los productos de los clientes que tiene la empresa.

Pantalla N°11: Visualizar Reporte Existencias



Reporte en Pantalla Clic en "P" para Enviar a Impresora o Página Previa ----->

SALDOS FISICOS DEL INVENTARIO 1
SITEC S.R.L.
A La Fecha: 20-11-2012

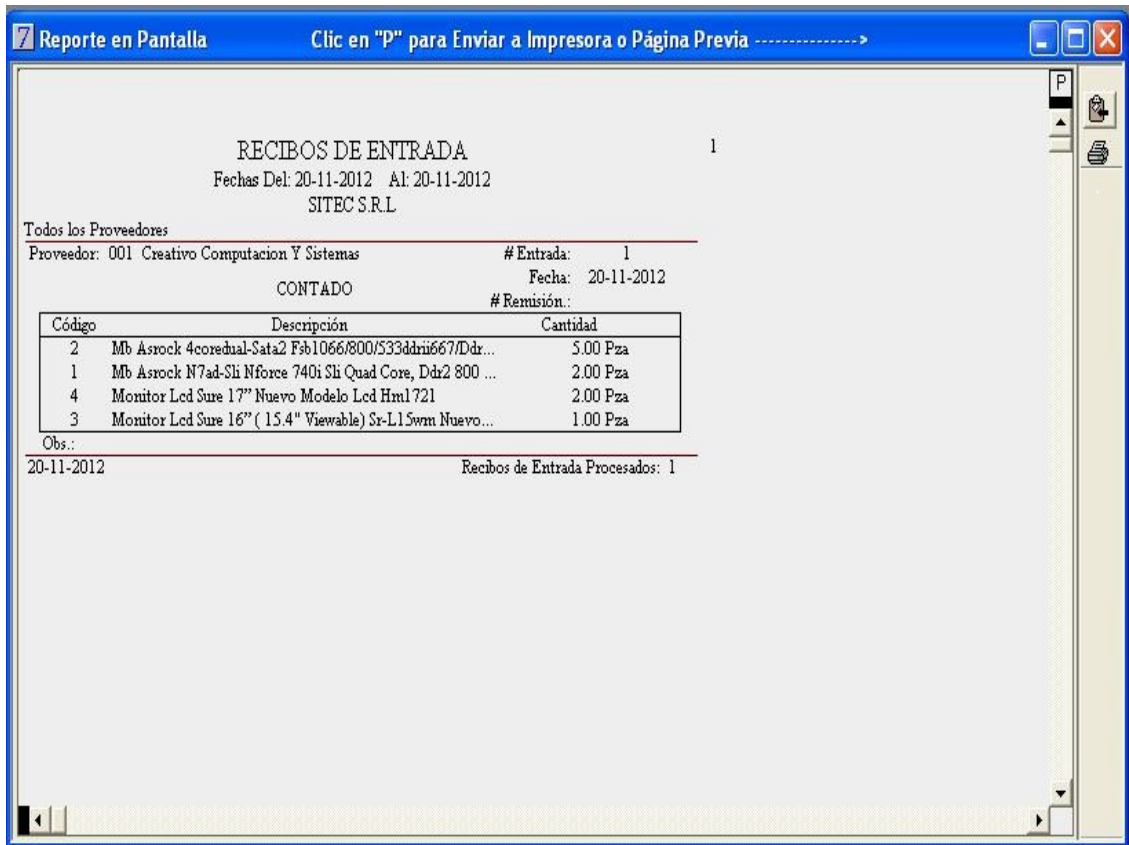
Todos Los Proveedores

#	Descripción	Cantidad
Grupo: 001 Equipos De Computacion		
1	2 Mb Asrock 4coresdual-Sata2 Fsb1066/800/533ddr1667/Ddr400 Cor	5.00 Pza
2	1 Mb Asrock N7ad-Sli Nforce 740i Sli Quad Core, Ddr2 800 16 G	2.00 Pza
3	3 Monitor Led Sure 16" (15.4" Viewable) Sr-L1.5wm Nuevo Modelo	1.00 Pza
4	4 Monitor Led Sure 17" Nuevo Modelo Led Hml 721	2.00 Pza

20-11-2012

Pantalla que muestra el reporte de las existencias de los productos el cual le permite imprimir o exportar a Excel.

Pantalla N°12: Visualizar Reporte Compras



Reporte en Pantalla Clic en "P" para Enviar a Impresora o Página Previa ----->

RECIBOS DE ENTRADA 1
Fechas Del: 20-11-2012 Al: 20-11-2012
SITEC S.R.L.

Todos los Proveedores
Proveedor: 001 Creativo Computacion Y Sistemas # Entrada: 1
Fecha: 20-11-2012
CONTADO # Remisión:

Código	Descripción	Cantidad
2	Mb Asrock 4coreDual-Sata2 Fb1066/800/S33ddri667/Ddr...	5.00 Pza
1	Mb Asrock N7ad-Sli Nforce 740i Sli Quad Core, Ddr2 800 ...	2.00 Pza
4	Monitor Lcd Sure 17" Nuevo Modelo Lcd Hml721	2.00 Pza
3	Monitor Lcd Sure 16" (15.4" Viewable) Sr-L15wm Nuevo...	1.00 Pza

Obs.:
20-11-2012 Recibos de Entrada Procesados: 1

Pantalla que muestra el reporte de las compras de los productos el cual le permite imprimir o exportar a Excel.

Pantalla N°13: Visualizar Reporte Ventas

Reporte en Pantalla Clic en "P" para Enviar a Impresora o Página Previa ----->

RECIBOS DE SALIDA 1

Fechas Del: 20-11-2012 Al: 20-11-2012
SITEC S.R.L

Todos los Centros de Costo Todos los Destinatarios

Centro de Costo: 001 Venta # Salida: 1
Destinatario: 001 Sitec Fecha: 20-11-2012
Doc.:

Código	Descripción	Cantidad
2	Mb Asrock 4coreDual-Sata2 Fsb1066/800/533drrii667/Ddr...	1.00 Pza
3	Monitor Led Sure 16" (15.4" Viewable) Sr-L15wm Nuevo...	1.00 Pza

Obs.:
20-11-2012 Recibos de Salida Procesados: 1

Pantalla que muestra el reporte de las compras de los productos el cual le permite imprimir o exportar a Excel.

Pantalla N°14: Crear Copia de Seguridad



Pantalla para crear una copia de seguridad del sistema para poder luego recuperar

Pantalla N°15: Restaurar Copia de Seguridad



Pantalla donde se restaura la copia de seguridad si el usuario cree necesario.

Pantalla N°16: Gestionar Productos

NUEVO Item...

Grupo:	001	EQUIPOS DE COMPUTACION	<input type="checkbox"/> DesBloqueado	✓
Proveedor:			<input checked="" type="checkbox"/> Activo	
Código		Descripción	Unidad	✗
Observaciones:				

NUEVO Item...






Grupo:	001	EQUIPOS DE COMPUTACION	<input type="checkbox"/> DesBloqueado	✓
Proveedor:	001	CREATIVO COMPUTACION Y SISTEMAS	<input checked="" type="checkbox"/> Activo	
Código	4	MONITOR LCD SURE 17 NUEVO MODELO LCD HM1721	Unidad	✗
			PZA	
Observaciones:				

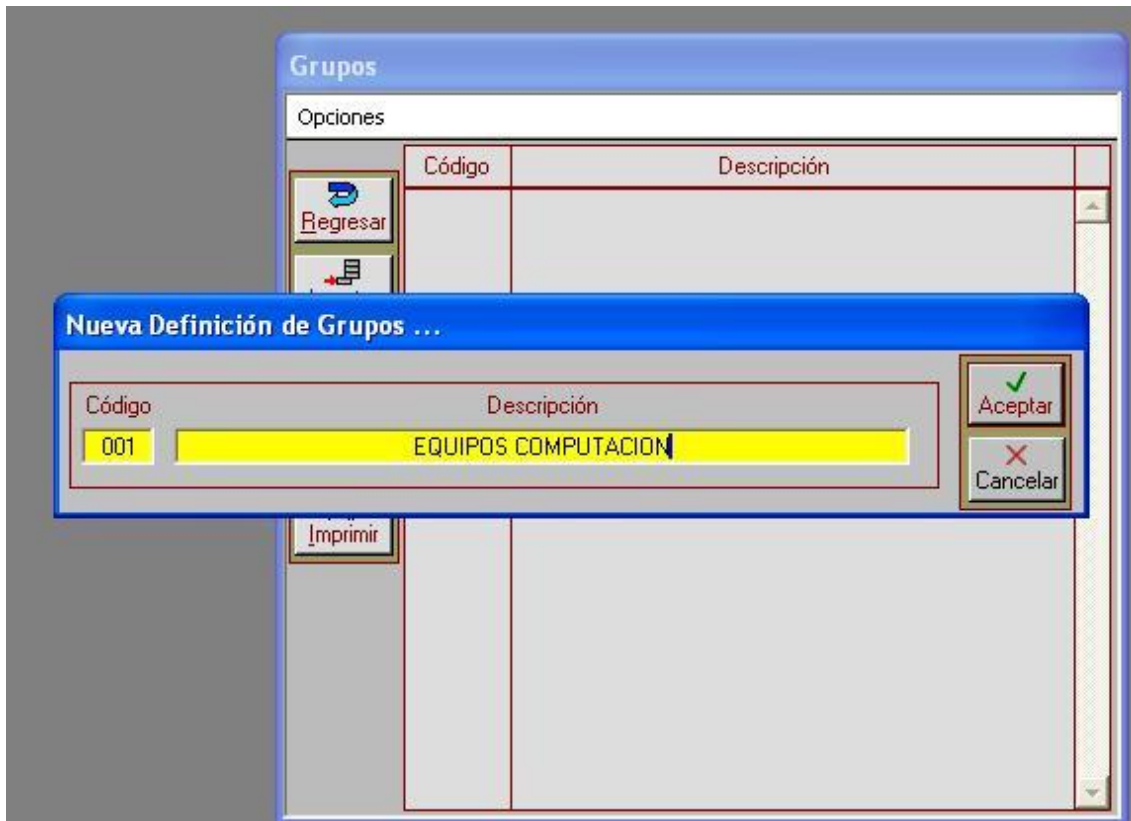
Pantalla que gestiona los productos registra, modifica y eliminar los datos de los productos que ofrece a los clientes.

Pantalla N°17: Gestionar Categorías

Grupos

Opciones

	Código	Descripción
 Regresar	001	Equipos De Computacion
 Insertar	002	Equipos De Sonido
 Modificar	003	Televisores
 Eliminar	004	Celulares
 Imprimir		



Pantalla para gestionar las categorías registrar, modificar y eliminar las distintas categorías de los productos.

5. Conclusiones y Recomendaciones

5.1 Conclusiones

Al finalizar el proyecto, y luego de haber atravesado los diferentes pasos necesarios para elaborar un sistema de información, se realizan una serie de conclusiones generales acerca de estos:

- ✚ Modelo de requerimientos: Es la primera etapa del proceso, y por lo tanto es muy relevante porque es la base sobre la cual se construyen las demás. Para la elaboración de la misma, se contó con el aporte del Gerente General de la empresa SITEC el cual fue de gran ayuda para la adecuada definición de los requerimientos, sobre los cuales se desarrollaron los primeros diagramas de UML, llamados Diagramas de Casos de Uso, debido a que estos diagramas permiten visualizar los actores, la funcionalidad que debe tener el sistema, y la relación que existe entre ambos.
- ✚ Modelo conceptual y análisis: Luego de haber completado la documentación correspondiente a cada caso de uso hallado en los requerimientos, se procedió a identificar en los escenarios, las primeras clases para el Diagrama de Clases, el cual funcionó como punto de partida para la siguiente etapa, esto debido a que en el análisis, la construcción de los diagramas de secuencia requieren tener esas clases previamente definidas, para emplearlas y mostrar la interacción que existe entre ellas en cada función del sistema.
- ✚ Modelo de diseño: Finalmente, para este momento se comienza a tener una vista general del sistema, para ver la funcionalidad y definición del mismo.
- ✚ Sin lugar a duda la elaboración de un sistema de información para una Empresa Comercial es un trabajo el cual conlleva muchos detalles relevantes, los cuales de no ser realmente tomados en cuenta en las etapas de análisis y diseño, pueden terminar siendo grandes fallas a la hora de implementar el sistema, y a pesar de que no es tarea fácil desarrollar software de alta calidad, flexible y sin errores, es posible proveer finalmente con la solución óptima y necesaria a las empresas comerciales para que haya un eficiente desarrollo en las actividades diarias mejorando el nivel de atención al Cliente haciendo uso para lograrlo de material magnético va realizar copias de seguridad.

5.2 Recomendaciones

La EMPRESA SITEC debe mantenerse informada de los cambios en las necesidades de sus clientes y a la vez formar un alto nivel de creatividad organizacional que permita el cambio y su gestión para que de manera constante, se vigile la calidad de los productos y la estrategia tecnológica, logrando así convertirse en una empresa proactiva e innovadora que este en constante búsqueda de su propia superación.

Para futuros de estudios de la empresa este sistema de gestión para el control de ventas, compras e inventarios sirve como inquietud para crear un subsistema para la impresión de facturas.

Por ello recomiendo, se Desarrolle e Implemente el Sistema de Gestión para el control de ventas, compras e inventarios para la empresa SITEC el cual ayudará a mejorar la atención de los clientes y además llevar un mejor control de los clientes que mas realizan compras para poder darles un mejor trato en descuentos, obteniendo mejores resultados para la empresa.

Además cabe recalcar que este estudio que abarca el análisis y diseño es adaptable para cualquier empresa comercial de cualquier rubro porque no restringe los tipos de productos se puede crear las categorías que desees y así mismo dentro de ellos relacionar todos los productos que se vea conveniente, y llevar un control adecuado de los empleados, clientes y proveedores de la empresa.

6. Glosario de Términos

TIC: Tecnologías de la Información y la Comunicación.

Sistema de Información: Es un conjunto de elementos orientados al tratamiento y administración de datos e información.

Automatizado: Tratar con un ordenador o computadora.

Métodos: que se refiere al medio utilizado para llegar a un fin.

Herramientas: Es un objeto elaborado a fin de facilitar la realización de una tarea.

Procedimientos: Es la acción de proceder o el **método** de ejecutar algunas cosas. Se trata de una **serie común de pasos definidos**, que permiten realizar un trabajo de forma correcta.

Sistema de Seguridad: Es el área de la informática que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con esta (incluyendo la información contenido).

RUP: Rational Unified Process.

UML: Lenguaje de Modelado Unificado.

Optimizar: **Buscar la mejor manera de realizar una actividad.**

Interfaz: La **interfaz de usuario** es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

Periodicidad: Calidad de lo que ocurre o se efectúa cada cierto espacio de tiempo, repetición regular: la periodicidad de las sesiones plenarias.

SGBD: Sistema gestor de bases de datos.

Empresa Comercial: Persona que se dedica a vender productos, generalmente a comisión, en representación de la empresa para la que trabaja.

Costo: Gasto que se hace para la obtención de una cosa o servicio.

Inventario: Relación detallada de bienes o pertenencias

Cliente: Persona que habitualmente compra en un establecimiento o requiere de sus servicios. A menudo se emplea cliente para el género femenino.

Proveedor: Persona o empresa que abastece de algunos artículos necesarios: si detecta alguna anomalía en el producto, acuda a su proveedor habitual y se lo cambiará.