

ESCUELA POLITECNICA DEL EJERCITO

FACULTAD DE INGENIERIA EN SISTEMAS E INFORMATICA

PLAN DE TESIS

TITULO DE LA TESIS:

“Sistema de Operaciones Comerciales para la Compañía
Repsol YPF Ecuador”

NOMBRE ALUMNO DENUNCIANTE:

Eric Fernando Narváez Galarza

Sangolquí, 23 de Febrero del 2005

I. Índice de Contenidos.

| | |
|--|----|
| RESUMEN | 6 |
| CAPÍTULO I | 7 |
| MARCO TEORICO..... | 7 |
| 1.1 Metodología OMT..... | 7 |
| 1.1.1 Generalidades | 7 |
| 1.1.2 Metodología OMT aplicada al sistema de Operaciones Comerciales | 9 |
| 1.1.2.1 Requerimientos..... | 9 |
| 1.1.2.2 Análisis de Objetos..... | 10 |
| 1.1.2.2.1 Modelo de Objetos..... | 11 |
| 1.1.2.2.2 Modelo Dinámico | 12 |
| 1.1.2.2.3 Modelo Funcional..... | 13 |
| 1.1.2.3 Diseño del Sistema | 14 |
| 1.1.2.4 Diseño de Objetos..... | 15 |
| 1.1.2.5 Implementación | 16 |
| 1.2 Introducción a Visual Studio .Net | 17 |
| 1.2.1 Generalidades. | 17 |
| 1.2.2 .Net Framework..... | 17 |
| 1.2.2.1 Biblioteca de clases de .Net Framework..... | 20 |
| 1.2.2.2 ASP.NET | 20 |
| 1.2.3 Arquitectura Cliente | 21 |
| 1.2.4 Arquitectura Servidor | 21 |
| 1.3 Introducción a la tecnología ASP .NET y Web Services XML | 22 |
| 1.3.1 ASP .NET | 22 |
| 1.3.1.1 Seguridad en ASP . NET | 23 |
| 1.3.1.2 Arquitectura de Asp .NET | 23 |
| 1.3.2 Web Services XML | 24 |
| 1.3.2.1 Escenarios de Servicios Web XML..... | 25 |
| 1.3.2.1.1 Servicios Simples | 25 |
| 1.3.2.1.2 Integración de aplicaciones | 25 |
| 1.3.2.1.3 Soluciones de flujo de trabajo | 26 |
| 1.3.2.2 Infraestructura de Servicios Web XML..... | 26 |
| 1.3.2.3 Periodo de duración de un Servicio Web XML. | 28 |
| 1.3.3 Servicios Web XML creados mediante ASP . NET..... | 29 |
| 1.4 Introducción a Oracle DataBase..... | 30 |
| 1.4.1 Oracle 8i..... | 30 |
| 1.4.2 Arquitectura de Oracle DataBase 8i | 31 |
| 1.4.3 Procesos en Oracle. | 34 |
| 1.4.3.1 Procesos Servidores..... | 34 |
| 1.4.3.2 Procesos Demonios | 34 |
| 1.4.3.3 Control de Recursos..... | 35 |
| 1.5 Introducción a Oracle Reports Developer | 35 |
| 1.5.1 Oracle Reports 6i..... | 35 |
| 1.5.2 Características y Beneficios | 35 |
| 1.5.3 Desarrollo rápido | 37 |
| 1.6 Introducción a Oracle Discoverer | 37 |

| | | |
|---------------------|---|----|
| 1.6.1 | Oracle Discoverer 4i..... | 37 |
| 1.6.2 | Características y Beneficios | 37 |
| 1.6.3 | Arquitectura de Oracle Discoverer | 38 |
| CAPÍTULO II..... | | 40 |
| REQUERIMIENTOS..... | | 40 |
| 2.1 | Recolección de la Información | 40 |
| 2.1.1 | Entrevistas..... | 40 |
| 2.1.2 | Resultado de las entrevistas..... | 40 |
| 2.2 | Especificación de Requerimientos según el Estándar IEEE 830 | 41 |
| 2.2.1 | Introducción | 41 |
| 2.2.1.1 | Propósito | 41 |
| 2.2.1.2 | Ámbito del sistema | 41 |
| 2.2.1.3 | Definiciones, acrónimos y abreviaturas..... | 43 |
| 2.2.1.3.1 | Definiciones | 43 |
| 2.2.1.3.2 | Acrónimos y Abreviaturas | 44 |
| 2.2.1.4 | Referencias..... | 45 |
| 2.2.1.5 | Visión general del documento..... | 46 |
| 2.2.2 | Descripción General | 46 |
| 2.2.2.1 | Perspectiva del producto..... | 46 |
| 2.2.2.2 | Funciones del producto..... | 49 |
| 2.2.2.3 | Características de los Usuarios..... | 49 |
| 2.2.2.4 | Restricciones | 50 |
| 2.2.2.4.1 | Políticas de la empresa..... | 50 |
| 2.2.2.4.2 | Interfaces con otras aplicaciones | 50 |
| 2.2.2.4.3 | Protocolos de comunicación..... | 51 |
| 2.2.2.4.4 | Requisitos de fiabilidad..... | 51 |
| 2.2.2.4.5 | Consideraciones acerca de la seguridad..... | 51 |
| 2.2.2.4.6 | Requisitos futuros..... | 51 |
| 2.2.3 | Requisitos Específicos | 52 |
| 2.2.3.1 | Interfaces externas | 52 |
| 2.2.3.2 | Interfaces de Usuario | 52 |
| 2.2.3.3 | Interfaces de Software | 52 |
| 2.2.3.4 | Interfaces de comunicaciones | 52 |
| 2.2.3.5 | Funciones | 52 |
| 2.2.3.6 | Requisitos de rendimiento..... | 57 |
| 2.2.3.7 | Atributos del sistema | 58 |
| 2.2.3.8 | Atributos del Software..... | 58 |
| 2.2.3.9 | Estándares del software..... | 60 |
| 2.3 | Construcción de Casos de Uso..... | 61 |
| 2.3.1 | Seguridad y Reporte..... | 61 |
| 2.3.2 | Ventas..... | 62 |
| 2.3.3 | Precios..... | 62 |
| 2.3.4 | Ajuste °Api..... | 63 |
| 2.3.5 | Ocp..... | 63 |
| 2.3.6 | Facturación Sote..... | 64 |
| 2.3.7 | Pagos Tivacuno..... | 64 |
| 2.3.8 | Ineficiencia Petrolera..... | 64 |

| | | |
|--|---|-----|
| 2.3.9 | Balance de Entrega | 65 |
| 2.3.10 | Modelo de Producción..... | 65 |
| 2.4 | Especificación de casos de uso | 65 |
| 2.4.1 | Identificación de caso de uso y descripción..... | 65 |
| CAPÍTULO III | | 66 |
| ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS..... | | 66 |
| 3.1 | Análisis Orientado a Objetos..... | 66 |
| 3.1.1 | Modelo de Objetos | 66 |
| 3.1.2 | Identificación de Clases y Objetos | 66 |
| 3.1.3 | Diccionario de Datos de las Clases Identificadas..... | 68 |
| 3.1.4 | Identificación de las asociaciones entre clases..... | 88 |
| 3.1.5 | Construcción del Modelo de Clases | 90 |
| 3.2.1 | Modelo Dinámico | 91 |
| 3.2.1.1 | Construcción Diagramas de Secuencia..... | 91 |
| 3.2.1.1.1 | Ventas | 91 |
| 3.2.1.1.2 | Precios..... | 105 |
| 3.2.1.1.3 | Factores API..... | 109 |
| 3.2.1.1.4 | OCP..... | 110 |
| 3.2.1.1.5 | Descuentos | 116 |
| 3.2.1.1.6 | Facturación SOTE | 118 |
| 3.2.1.1.7 | Pagos Tivacuno | 124 |
| 3.2.1.1.8 | Ineficiencia Petrolera | 126 |
| 3.2.1.1.9 | Producciones Fiscalizadas..... | 127 |
| 3.2.1.1.10 | Balance de Entregas..... | 130 |
| 3.3.1 | Modelo Funcional..... | 132 |
| 3.3.1.1 | Construcción de Diagramas de Flujo de Datos | 132 |
| 3.3.1.1.1 | Ventas | 132 |
| 3.3.1.1.2 | Precios..... | 142 |
| 3.3.1.1.3 | OCP..... | 146 |
| 3.3.1.1.4 | Descuentos | 151 |
| 3.3.1.1.5 | Facturación SOTE | 154 |
| 3.3.1.1.6 | Gestionar Pagos Tivacuno | 158 |
| 3.3.1.1.7 | Gestionar Ineficiencia Petrolera | 159 |
| 3.3.1.1.8 | Gestionar Producciones Fiscalizadas..... | 160 |
| 3.3.1.1.9 | Gestionar Balance de Entregas..... | 164 |
| 3.4.1 | Otros Modelos..... | 166 |
| 3.4.1.1 | Construcción Modelo N-tier | 166 |
| 3.4.1.2 | Construcción Modelo Visual | 167 |
| 3.4.1.3 | Construcción Modelo de Componentes..... | 168 |
| 3.2 | Diseño del Sistema | 169 |
| 3.2.1 | Organización del Sistema en Subsistemas..... | 169 |
| 3.2.2 | Identificación de recursos globales, control y mantenimiento..... | 169 |
| 3.2.3 | Estructura básica del sistema | 170 |
| 3.3 | Diseño de Objetos | 173 |
| 3.3.1 | Modelo de Objetos Detallado..... | 173 |
| 3.3.2 | Modelo Entidad Relación | 174 |
| CAPÍTULO IV | | 176 |

| | |
|--|-----|
| PRUEBAS E IMPLEMENTACION | 176 |
| 4.1 Introducción..... | 176 |
| 4.2 Problemas encontrados..... | 176 |
| 4.2.1 Verificación de la Interfaz del SOC..... | 176 |
| 4.2.2 Verificación de las operaciones del SOC..... | 182 |
| 4.3 Validación del Sistema SOC..... | 184 |
| 4.3.1 Validación de las Salidas del Sistema | 184 |
| 4.3.2 Tiempo de ejecución del sistema..... | 184 |
| 4.4 Manual del Usuario | 184 |
| CAPÍTULO V..... | 185 |
| CONCLUSIONES Y RECOMENDACIONES..... | 185 |
| 5.1 Conclusiones | 185 |
| 5.2 Recomendaciones..... | 186 |
| BIBLIOGRAFÍA | 189 |

II. Listado de Tablas

| | |
|---|----|
| Tabla 1.1 (Cuadro de documentos arrojados por la OMT)..... | 19 |
| Tabla 1.2: (Estándares Servicios Web XML)..... | 32 |
| Tabla 2.1: (Definiciones de la Especificación de Requerimientos)..... | 46 |
| Tabla 2.2: (Acrónimos de la especificación de requerimientos)..... | 47 |

III. Listado de Figuras

| | |
|--|-----|
| Fig 1.1: (Etapas de la Metodología OMT)..... | 12 |
| Fig 1.2: (Código administrado en el servidor)..... | 22 |
| Fig 1.3: (Arquitectura ASP.NET) | 24 |
| Fig 1.4: (Infraestructura Servicios Web XML)..... | 30 |
| Fig 1.5: (Período de duración Servicio Web XML)..... | 31 |
| Fig 1.6: (Arquitectura de Oracle) | 32 |
| Fig 1.7: (Estructura Básica del System Global Area) | 33 |
| Fig 1.8: (Arquitectura de Oracle Discoverer 4.1.) | 41 |
| Fig 1.9: (Esquematación Cliente/Servidor)..... | 42 |
| Fig 2.1: (Planificación entrevistas) | 43 |
| Fig 2.2: (Diagrama de Bloques del sistema SOC) | 46 |
| Fig. 3.1: (Flujo de Datos Básico del SOC) | 171 |
| Fig 3.2: (Arquitectura Básica del SOC) | 172 |

IV. Listado de Anexos

| | |
|--------------|-----|
| ANEXO A..... | 192 |
| ANEXO B..... | 193 |

RESUMEN

Repsol-YPF Ecuador S.A. es una empresa de Exploración y Producción de Hidrocarburos, comprometida con la conservación del medio ambiente, el respeto y colaboración hacia las comunidades indígenas ubicadas en el área de su influencia, la seguridad de las personas y la calidad de sus procesos.

Para realizar los procesos que conlleva la comercialización del Crudo el departamento de Operaciones Comerciales esta dedicado a:

- ✓ Coordinar las exportaciones del crudo producido en Ecuador por Repsol YPF, con RYTTSA, OCP y Petroecuador,
- ✓ Administrar los contratos que opera Repsol YPF, esto es: Bloque 16, Bogi Capirón y Tivacuno,
- ✓ Cálculo del Ingreso mensual de la compañía y
- ✓ Administración de Operaciones con el OCP.

Toda esta información se encuentra en formatos Excel lo que hace difícil su acceso y la seguridad del mismo, por esta razón se automatizara todos los procesos comerciales y además se proporcionara reportes estáticos y dinámicos, es decir, los reportes estáticos serán desarrollados para que el usuario final los pueda ver como formatos establecidos y de acuerdo a los requerimientos que plantearon, en cambio, los reportes dinámicos serán ajustados por el usuario final según sus necesidades, mediante el uso de una herramienta flexible y amistosa.

base conceptos del mundo real. La unidad básica es el objeto que combina las estructuras de datos con los comportamientos en una entidad única.

La metodología OMT es una técnica que se hace cargo de todo el ciclo de vida del software a la cual la divide en cuatro fases consecutivas: análisis de objetos, diseño del sistema, diseño de objetos e implementación.

Además de esto la metodología emplea tres clases de modelos para describir al sistema:

- Modelo de Objetos
- Modelo Dinámico
- Modelo Funcional

El modelo de objetos que describe los objetos del sistema y sus interrelaciones; el modelo funcional, que describe las transformaciones de los datos del sistema; y el modelo dinámico, que describe las interacciones entre los objetos del sistema. Cada modelo es aplicable durante todas las etapas del desarrollo y adquiere los detalles de la construcción a medida que progresa el desarrollo. Una descripción completa de un sistema requiere los tres modelos.

El ciclo de vida OMT esta conformado por las siguientes etapas:

Análisis de Objetos

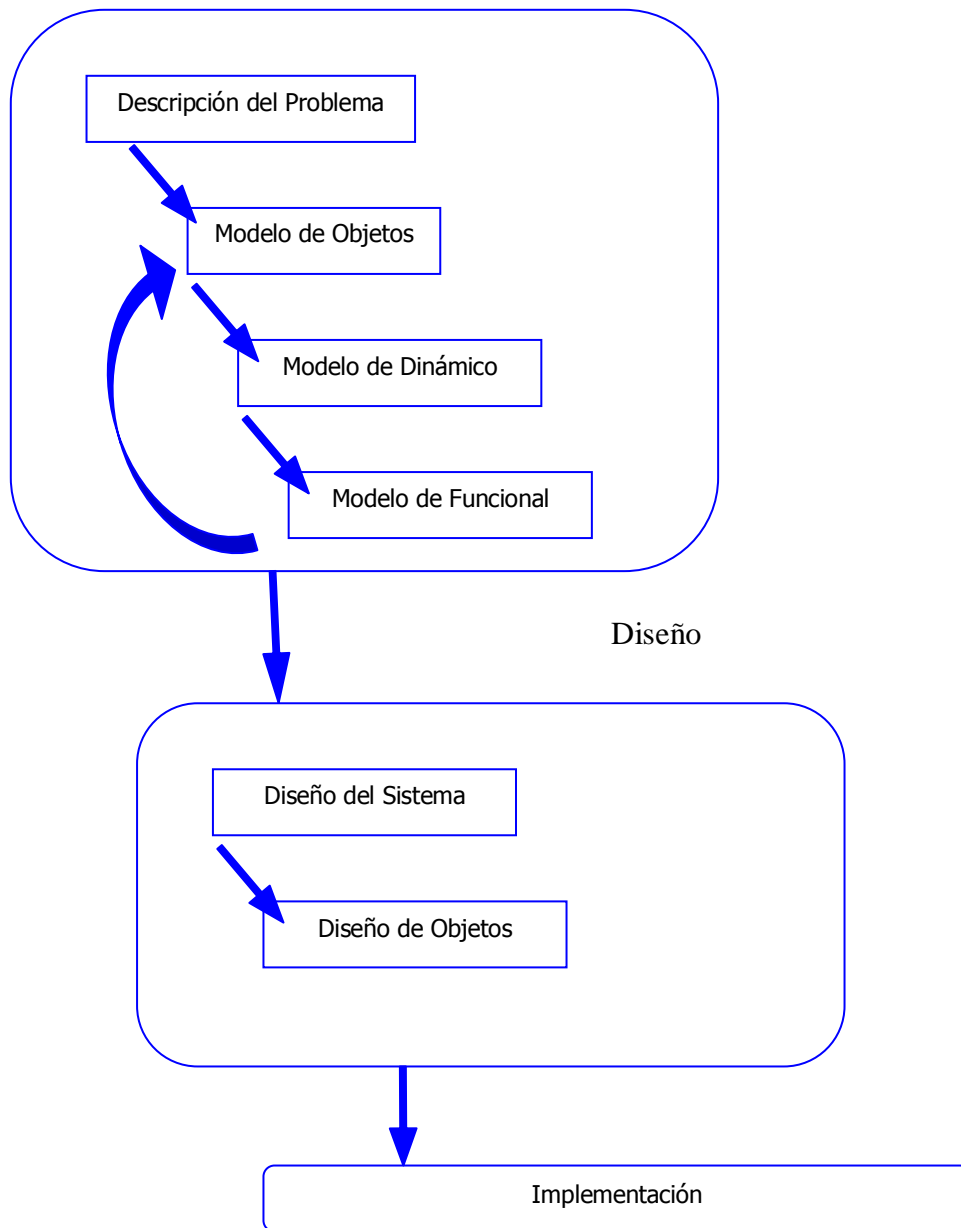


Fig 1.1: (Etapas de la Metodología OMT)

1.1.2 Metodología OMT aplicada al sistema de Operaciones Comerciales

1.1.2.1 Requerimientos

La etapa de obtención de requerimientos es en la cual se obtendrá toda la información necesaria sobre el proyecto que se desea construir, siguiendo normas establecidas como la IEEE 830 que es un estándar de Especificación de Requerimientos ó utilizando cualquier método de

recolección de información, el objetivo principal de esta fase es obtener la información detallada y precisa de lo que se desea hacer.

El desarrollo de la aplicación empieza con el levantamiento de los requerimientos o de la información necesaria por parte de la empresa o de los usuarios, es decir, como conciben el sistema y la manera en que formulan sus requerimientos.

Toda la información que se obtenga deberá ser suficiente y precisa para que no queden dudas sobre el rendimiento, funcionalidad, contexto y comportamiento de la aplicación y así evitar problemas de funcionalidad en el transcurso del proyecto.

La información que se obtenga de la fase de requerimientos será ordenada y tabulada para luego en la fase siguiente ser analizada.

1.1.2.2 Análisis de Objetos

En la fase de análisis de objetos toda la información que se obtuvo de la fase de requerimientos, es analizada, para luego construir un modelo del dominio del problema el cual muestra sus propiedades más importantes. El modelo de análisis es una abstracción resumida y precisa de lo que debe hacer el sistema deseado y no de la forma en que se hará. Los objetos del modelo deberán ser conceptos del dominio de la aplicación y no conceptos de implementación de la computadora tales como estructuras de datos.

El modelo de análisis no deberá contener ninguna decisión de implementación, los objetos se describirán en términos de atributos y operaciones que son visibles para el usuario.

Aquí es donde se determina el modelo de objetos, se hace una tentativa de clases (eliminar las clases irrelevantes), las posibles asociaciones entre las clases, luego se hace la refinación de asociaciones eliminando las redundantes o las que no tienen relevancia, posteriormente se hace una tentativa de atributos de objetos y enlaces.

Una vez obtenido los objetos del sistema se hace un refinamiento del modelo, posteriormente se busca un nivel de abstracción para modelar sub-sistemas, para buscar un sistema tangible y sólido.

Una vez desarrollado el modelo se introduce la noción de transacción, es una forma de modelar procesos o describir cambio de datos, movimiento de datos, una vez definido el movimiento de datos se define un diccionario de datos de todas las entidades modeladas.

Los modelos que se utilizan para la descripción del sistema y para el análisis orientado a objetos facilitan la comprensión y el desempeño

futuro de la aplicación, Una descripción completa del sistema requiere los tres modelos.

Un procedimiento típico de software contiene estos tres aspectos:

1. utiliza estructuras de datos (modelo de objetos),
2. secuencia las operaciones en el tiempo (modelo dinámico) y
3. transforma valores (modelo funcional).

Cada modelo referencia a entidades de los otros modelos, los tres modelos están relacionados entre sí. Las interconexiones entre los distintos modelos son limitadas y explícitas. Los buenos diseños aíslan los distintos aspectos del sistema y limitan el acoplamiento entre ellos.

El más importante es el modelo de objetos porque es necesario para describir “qué” está cambiando o transformándose, antes de describir “cuándo” y “cómo” cambia.

El enfoque orientado a objetos se centra primordialmente en identificar objetos procedentes del dominio de la aplicación ajustándoles después los procedimientos.

1.1.2.2.1 Modelo de Objetos

Describe la estructura estática (de datos), de los objetos del sistema (identidad, atributos y operaciones) y también sus relaciones. El modelo de objetos contiene diagramas de objetos. Un diagrama de objetos es un grafo cuyos nodos son clases de objetos y cuyos arcos son relaciones entre las clases. El diagrama contiene clases de objetos organizados en jerarquías que comparten una estructura y comportamiento comunes y que están asociadas a otras clases. Estas clases definen los atributos que lleva cada instancia de objeto y las operaciones que efectúa o sufre cada uno. En cada instancia de la clase se guardan los valores de esos atributos.

Los diagramas de objetos permiten representar gráficamente los objetos, las clases y sus relaciones mediante dos tipos de diagramas: los diagramas de clases y los diagramas de casos concretos (instancias).

Los diagramas de clases describen las clases que componen el sistema y que permitirán la creación de casos concretos, los diagramas de casos concretos describen la manera en que los objetos del sistema se relacionan y los casos concretos que existen en el sistema de cada clase. En los diagramas que componen este modelo se pueden representar los siguientes

elementos del sistema: objetos y clases, atributos, operaciones, y relaciones o asociaciones.

Los pasos que se deben seguir para construir un modelo de objetos correcto son:

- Identificar los Objetos y Clases
- Preparar un Diccionario de Datos
- Identificar asociaciones entre objetos
- Organizar y simplificar las clases usando herencia
- Agregar las clases en módulos.
- Iterar y refinar el modelo.

1.1.2.2.2 Modelo Dinámico

Describe los aspectos de comportamiento (de control) de un sistema que cambian con el tiempo. El modelo dinámico se utiliza para especificar e implementar los aspectos del control del sistema. Los modelos dinámicos contienen diagramas de estados. Un diagrama de estados es un grafo cuyos nodos son estados y cuyos arcos son transiciones entre estados causadas por sucesos o eventos.

Se especifican en este modelo la temporización y secuencia de operaciones (sucesos que marcan los cambios, secuencias de sucesos, estados que definen el contexto para los sucesos), y la organización de sucesos y de estados. El modelo dinámico captura el control, aquel aspecto de un sistema que describe las secuencias de operaciones que se producen sin tener en cuenta lo que hagan las operaciones, aquello a lo que afecten o la forma en la que estén implementadas.

Las acciones de los diagramas de estado se corresponden con funciones procedentes del modelo funcional; los sucesos de un diagrama de estado pasan a ser operaciones que se aplican a objetos dentro del modelo de objetos.

El diagrama de estados especifica la secuencia de estados que causa una cierta secuencia de sucesos. Si un objeto se encuentra en un cierto estado y se produce un suceso cuyo nombre corresponda al de una de sus transiciones, entonces el objeto pasa al estado que se encuentra en el extremo de destino de la transición. Se dice que la transición se dispara. Si hay más de una transición que sale de un estado, entonces el primer suceso que se produzca dará lugar a que se dispare la transición correspondiente

Para desarrollar un modelo dinámico se debe seguir los siguientes pasos.

1. Preparar escenarios para la secuencias de interacción típicas.
2. Se identifican sucesos que actúen entre objetos.
3. Se prepara un seguimiento de sucesos para cada escenario
4. Se construye un diagrama de estados.
5. Verificar, iterar y refinar el modelo dinámico.

1.1.2.2.3 Modelo Funcional

Describe las transformaciones (de función), de valores de datos que ocurren dentro del sistema, captura lo que hace el sistema, independientemente de cuando se haga o de la forma en que se haga. El modelo funcional contiene diagramas de flujo de datos. Un diagrama de flujo de datos es un grafo cuyos nodos son procesos y cuyos arcos son flujos de datos, se muestra las dependencias entre los valores y el cálculo de valores de salida a partir de los de entrada y de funciones, sin considerar cuando se ejecutan las funciones, ni siquiera si llegan a ejecutarse. Las funciones se invocan como acciones en el modelo dinámico y se muestran como operaciones que afectan a objetos en el modelo de objetos.

Consta de múltiples diagramas de flujo de datos, que muestran el flujo de valores desde las entradas externas, a través de las operaciones y almacenes internos de datos hasta las salidas externas. También incluyen restricciones entre valores dentro del modelo de objetos. Los diagramas de flujo de datos no muestran el control ni tampoco información acerca de la estructura de los objetos; todo esto pertenece a los modelos dinámico y de objetos.

Un diagrama de flujo de datos contiene procesos que transforman datos, flujos de datos que los trasladan, objetos actores que producen y consumen datos, y de almacenes de datos que los almacenan de forma pasiva.

Especificación de Operaciones y Restricciones.

Los procesos de los diagramas de flujo deben ser implementados eventualmente como operaciones que se aplican a objetos. Todo proceso atómico del más bajo nivel es una operación. Los procesos de nivel superior también se pueden considerar

operaciones, aún cuando una implementación pueda estar organizada de forma distinta del diagrama de datos que representa como consecuencia de la optimización.

La especificación de una operación se compone de dos partes. La primera de ellas es la que indica la interfaz de la operación: los argumentos que requiere y los valores que proporciona. La segunda, es la que explica la transformación de los valores de entrada para producir los valores de salida.

En cambio una restricción muestra la relación entre dos objetos al mismo tiempo o bien entre distintos valores del mismo objeto en instantes diferentes. Las restricciones se pueden expresar como una función total (un valor que es especificado completamente por otro) o como una función parcial (un valor que está restringido, pero no completamente especificado por otro valor). Las restricciones pueden aparecer en todas las clases del modelo.

Las restricciones de objetos especifican que algunos objetos dependen entera o parcialmente de otros objetos. Las restricciones dinámicas especifican relaciones entre los estados o sucesos de distintos objetos. Las restricciones funcionales especifican limitaciones aplicables a operaciones.

Una restricción entre valores de un objeto a lo largo del tiempo es lo que suele denominarse un invariante.

Para desarrollar un modelo funcional se debe seguir los siguientes pasos:

1. Identificar los valores de entrada y salida.
2. Construir diagramas de flujo de datos que muestren las dependencias funcionales.
3. Describir funciones.
4. Identificar las restricciones
5. Especificar los criterios de optimización
6. Verificar, iterar y refinar el modelo funcional.

1.1.2.3 Diseño del Sistema

Durante la fase de diseño de sistemas se toman decisiones de alto nivel acerca de la arquitectura global del sistema. El sistema de destino se organiza en subsistemas basados tanto en la estructura del análisis como en la arquitectura propuesta. El diseñador de sistemas deberá decidir qué características de rendimiento hay que optimizar. Seleccionando

una estrategia para atacar el problema y efectuando las reservas de recursos tentativas.

Los pasos a seguir para del diseño del sistema se describen a continuación:

1. Organizar el sistema en subsistemas,
2. Identificar la concurrencia inherente al problema,
3. Asignar los subsistemas a sus procesos y tareas,
4. Elegir la estrategia básica para almacenar los datos
5. Identificar los recursos globales y determinar mecanismo de control de acceso a ellos,
6. Elegir un método de implementación de control de software,
7. Considerar las condiciones limite,
8. Establecer las prioridades de decisión sobre las características deseables de producto de software.

1.1.2.4 Diseño de Objetos

Se construye un modelo de diseño basándose en el modelo de análisis que lleven incorporados detalles de implementación. El diseñador añade detalles al modelo de acuerdo con la estrategia establecida durante el diseño del sistema. El foco de atención del diseño de objetos son las estructuras de datos y los algoritmos necesarios para implementar cada una de las clases. Las clases de objetos procedentes del análisis siguen siendo significativas pero se aumentan con estructuras de datos y algoritmos del dominio de la computadora seleccionados para optimizar medidas importantes de rendimiento. Tanto los objetos del dominio de la aplicación como los objetos del dominio de la computadora se describen utilizando unos mismos conceptos y una misma notación orientados a objetos aún cuando existan en planos conceptuales diferentes.

Los pasos a seguir para el diseño de objetos son los siguientes:

1. Obtener las operaciones para el modelo de objetos a partir de los otros modelos:
 - Encontrar una operación para cada proceso en el modelo funcional.
 - Definir una operación para cada evento en el modelo dinámico, dependiendo de la implantación del control.
2. Diseñar los algoritmos para implantar las operaciones:
 - Escoger los algoritmos que minimicen el costo de implementación de las operaciones.
 - Seleccionar las estructuras de datos apropiadas para los algoritmos.

- Definir clases internas y operaciones nuevas según sea necesario.
 - Asignar las responsabilidades para las operaciones que no están asociadas claramente con una sola clase.
3. Optimizar las rutas de acceso a los datos:
 - Agregar asociaciones redundantes para minimizar los costos de acceso y maximizar la conveniencia.
 - Reacomodar los cálculos para una mayor eficiencia.
 - Guardar los valores derivados para evitar recalcular expresiones complicadas.
 4. Implantar el control del software introduciendo el esquema seleccionado durante el diseño de sistemas.
 5. Ajustar la estructura de clases para incrementar la herencia:
 - Reacomodar y ajustar las clases y las operaciones para incrementar la herencia.
 - Abstractar el comportamiento común de los grupos de clases.
 - Usar delegación para compartir comportamiento donde la herencia sea semánticamente inválida.
 6. Diseñar la implantación de las asociaciones:
 - Analizar las travesías de las asociaciones.
 - Implantar cada asociación como un objeto distinto o agregando atributos objeto-valor a una o ambas clases en la asociación.
 7. Determinar la representación de los atributos de los objetos.
 8. Empaquetar las clases y las asociaciones en módulos.

1.1.2.5 Implementación

Las clases de objetos y las relaciones desarrolladas durante su diseño se traducen finalmente a un lenguaje de programación concreto, a una base de datos o a una implementación en hardware. La programación debería ser una parte relativamente pequeña del ciclo de desarrollo y fundamentalmente mecánica porque todas las decisiones importantes deberán hacerse durante el diseño. El lenguaje de destino influye en cierta medida sobre las decisiones de diseño pero éste no debería depender de la estructura final de un lenguaje de programación.

Tabla 1.1 (Cuadro de Documentos Arrojadados por OMT)

| Fase | Etapas | Documento |
|--------------------|-------------------|---|
| Análisis | Modelo de Objetos | Diagramas de Objetos y diccionario de datos. |
| | Modelo Dinámico | Diagramas de Secuencia. |
| | Modelo Funcional | Diagrama de Flujo de Datos. |
| Diseño de Sistemas | | Estructura de la Arquitectura básica del sistema y decisiones estratégicas de alto nivel. |
| Diseño de Objetos. | | Modelo de Objetos detallado. |

1.2 Introducción a Visual Studio .Net

1.2.1 Generalidades.

Visual Studio .Net 2003 es una herramienta poderosa para la rápida generación de aplicaciones Web Asp .NET a escala empresarial y aplicaciones de escritorio de alto rendimiento. Visual Studio incluye herramientas de desarrollo basadas en componentes, como Visual C#, Visual J#, Visual Basic y Visual C++, así como diversas tecnologías suplementarias para simplificar el diseño, desarrollo e implementación en equipo de las soluciones.

Visual Studio admite el entorno de Microsoft .NET Framework, que ofrece Common Language Runtime y las clases de programación unificadas; ASP.NET utiliza estos componentes para crear aplicaciones Web ASP.NET y servicios Web XML.

Visual Studio .NET 2003 incluye una completa gama de funciones, desde modeladores que ayudan a componer visualmente las aplicaciones empresariales más complejas hasta la implementación de una aplicación en el más pequeño de los dispositivos. Utilizados por compañías de todos los tamaños en el mundo entero, Visual Studio .NET y la plataforma .NET Framework de Microsoft Windows proporcionan una completa herramienta, eficaz y sofisticada, para diseñar, desarrollar, depurar e implementar aplicaciones seguras para Microsoft Windows® y Web, a la vez sólidas y fáciles de utilizar.

Visual Basic .NET, Visual C++ .NET, Visual C# .NET y Visual J# .NET utilizan el mismo entorno de desarrollo integrado (IDE), que les permite compartir herramientas y facilita la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes aprovechan las funciones de .NET Framework.

1.2.2 .Net Framework

.NET Framework es un componente integral de Windows que admite la creación y la ejecución de la siguiente generación de aplicaciones y servicios Web XML. El diseño de .NET Framework está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Ofrecer un entorno de ejecución de código que fomente la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.

- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan secuencias de comandos o intérpretes de comandos.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en la Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.

.NET Framework contiene tres componentes principales: Common Language Runtime y la biblioteca de clases de .NET Framework. Y ASP .NET. Common Language Runtime es el fundamento de la tecnología. El motor de tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la interacción remota, al tiempo que aplica una seguridad estricta a los tipos y otras formas de especificación del código que fomentan su seguridad y solidez. De hecho, el concepto de administración de código es un principio básico del motor de tiempo de ejecución. El código destinado al motor de tiempo de ejecución se denomina código administrado, a diferencia del resto de código, que se conoce como código no administrado.

La biblioteca de clases, el otro componente principal de .NET Framework, es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios Web Forms y los servicios Web XML básico del motor de tiempo de ejecución. Además consta de una biblioteca de clases, interfaces y tipos de valor que se incluye en Microsoft .NET Framework SDK. Esta biblioteca brinda acceso a la funcionalidad del sistema y es la base sobre la que se crean las aplicaciones, los componentes y los controles de .NET Framework.

Esta constituida por espacios de nombres. Cada espacio de nombres contiene tipos que se pueden utilizar en el programa: clases, estructuras, enumeraciones, delegados e interfaces. Por ejemplo, cuando se crea un proyecto de Visual Basic o Visual C# en Visual Studio, se sigue haciendo referencia a las DLL más comunes de la clase base (ensamblados). Sin embargo si se necesita de alguna funcionalidad especial podrá agregar los espacios de nombres o DLL correspondientes a las funciones empleadas.

Common Language Runtime

Common Language Runtime administra la memoria, ejecución de subprocesos, ejecución de código, comprobación de la seguridad del código, compilación y demás servicios del sistema. Estas características son intrínsecas del código administrado que se ejecuta en Common Language Runtime.

Los compiladores y las herramientas exponen la funcionalidad del tiempo de ejecución y permiten escribir código con las ventajas que proporciona este entorno de ejecución administrado. El código desarrollado con un compilador de lenguaje orientado al tiempo de ejecución se denomina código administrado. Este código se beneficia de características como: la integración entre lenguajes, el control de excepciones entre lenguajes, la seguridad mejorada, la compatibilidad con la implementación y las versiones, un modelo simplificado de interacción y servicios de creación de perfiles y depuración.

Para permitir al motor de tiempo de ejecución proporcionar servicios al código administrado, los compiladores de lenguajes deben emitir metadatos que describen los tipos, los miembros y las referencias del código. Los metadatos se almacenan con el código; cada archivo ejecutable portable (PE) de Common Language Runtime cargable contiene metadatos.

El motor de tiempo de ejecución utiliza los metadatos para localizar y cargar clases, colocar instancias en memoria, resolver invocaciones a métodos, generar código nativo, exigir mecanismos de seguridad y establecer los límites del contexto del tiempo de ejecución.

El tiempo de ejecución controla automáticamente la disposición de los objetos y administra las referencias a éstos, liberándolos cuando ya no se utilizan. Los objetos cuya duración se administra de esta forma se denominan datos administrados. La recolección de elementos no utilizados elimina pérdidas de memoria así como otros errores habituales de programación. Con un código administrado se pueden utilizar datos administrados, datos no administrados o estos dos tipos de datos en una aplicación .NET. Framework. Como los compiladores de lenguajes proporcionan sus propios tipos, como tipos primitivos, no siempre se sabe (o no es necesario saber) si los datos se están administrando.

Además, el motor de tiempo de ejecución impone la solidez del código mediante la implementación de una infraestructura estricta de comprobación de tipos y código denominada CTS (Common Type System, Sistema de tipos común). CTS garantiza que todo el código administrado es autodescriptivo. Los diferentes compiladores de lenguajes de Microsoft y de terceros generan código administrado que se ajusta a CTS. Esto significa que el código administrado puede usar otros tipos e instancias administrados, al tiempo que se aplica inflexiblemente la fidelidad y seguridad de los tipos.

Aunque el motor de tiempo de ejecución está diseñado para el software del futuro, también es compatible con el software actual y el software antiguo. La interoperabilidad entre el código administrado y no administrado permite que los programadores continúen utilizando los componentes COM y las DLL que necesiten.

El motor de tiempo de ejecución está diseñado para mejorar el rendimiento. Aunque Common Language Runtime proporciona muchos servicios estándar

de motor de tiempo de ejecución, el código administrado nunca se interpreta. Una característica denominada compilación JIT (Just-In-Time) permite ejecutar todo el código administrado en el lenguaje máquina nativo del sistema en el que se ejecuta. Mientras tanto, el administrador de memoria evita que la memoria se pueda fragmentar y aumenta la zona de referencia de la memoria para mejorar aún más el rendimiento.

1.2.2.1 Biblioteca de clases de .Net Framework

La biblioteca de clases de .NET Framework es una colección de tipos reutilizables que se integran estrechamente con Common Language Runtime. La biblioteca de clases está orientada a objetos, lo que proporciona tipos de los que su propio código administrado puede derivar funciones. Esto ocasiona que los tipos de .NET Framework sean sencillos de utilizar y reduce el tiempo asociado con el aprendizaje de las nuevas características de .NET Framework.

Como en cualquier biblioteca de clases orientada a objetos, los tipos de .NET Framework permiten realizar diversas tareas de programación comunes, como son la administración de cadenas, recopilación de datos, conectividad de bases de datos y acceso a archivos. Además de estas tareas habituales, la biblioteca de clases incluye tipos adecuados para diversos escenarios de desarrollo especializados.

1.2.2.2 ASP.NET

ASP.NET está construida sobre las clases de programación del entorno .NET Framework, y proporciona un modelo para aplicaciones Web con un conjunto de controles y una infraestructura que simplifican la creación de aplicaciones Web ASP. ASP.NET incluye un conjunto de controles que encapsulan los elementos comunes de la interfaz de usuario de HTML, como cuadros de texto y menús desplegables. Sin embargo, dichos controles se ejecutan en el servidor Web, y envían al explorador Web su interfaz de usuario en forma de HTML. En el servidor, los controles exponen un modelo de programación orientado a objetos que ofrece al programador de Web la riqueza de la programación orientada a objetos. ASP.NET proporciona también servicios de infraestructura, como administración de estado de sesión y reciclado de procesos, que reducen aún más la cantidad de código que debe escribir el programador, e incrementan la fiabilidad de las aplicaciones. Asimismo, ASP.NET utiliza estos mismos conceptos para permitir a los programadores la entrega de software en forma de servicio. Mediante las funciones de servicios Web XML, los programadores de ASP.NET pueden escribir su lógica de empresa y utilizar la infraestructura de ASP.NET para entregar el servicio a través de SOAP.

1.2.3 Arquitectura Cliente

Las aplicaciones cliente constituyen lo más parecido a una aplicación de estilo tradicional en la programación basada en Windows. En este tipo de aplicaciones se muestran ventanas o formularios en el escritorio, lo que permite al usuario realizar una tarea. Entre las aplicaciones cliente se incluyen los procesadores de texto y las hojas de cálculo, además de aplicaciones empresariales, como herramientas de entrada de datos, de informes, etcétera. En las aplicaciones cliente se suelen emplear ventanas, menús, botones y otros elementos de la interfaz gráfica de usuario, y suelen tener acceso a recursos locales como el sistema de archivos y a dispositivos periféricos como las impresoras.

Otro tipo de aplicación cliente es el tradicional control ActiveX (reemplazado ahora por el control de Windows Forms) implementado en Internet como una página Web. Esta aplicación es muy parecida a otras aplicaciones cliente: se ejecuta de forma nativa, tiene acceso a los recursos locales e incluye elementos gráficos.

.NET Framework proporciona propiedades simples para ajustar los atributos visuales asociados con los formularios. En determinadas circunstancias, el sistema operativo subyacente no permite cambiar estos atributos directamente y, entonces, .NET Framework vuelve a crear los formularios de forma automática. Ésta es una de las múltiples maneras en que .NET Framework integra la interfaz del programador, con lo que la creación de código resulta más sencilla y más coherente.

A diferencia de los controles ActiveX, los controles de Windows Forms tienen acceso con una confianza parcial al equipo de un usuario. Esto significa que el código binario o que se ejecuta de forma nativa puede tener acceso a algunos de los recursos del sistema del usuario (como elementos de la GUI y acceso limitado a los archivos) sin tener acceso ni comprometer los demás recursos. Debido a la seguridad de acceso a código, muchas aplicaciones que antes era necesario instalar en el sistema de un usuario, ahora se pueden implementar a través del Web. Las aplicaciones pueden implementar las características de una aplicación local a la vez que se implementan como una página Web.

1.2.4 Arquitectura Servidor

Las aplicaciones de servidor en entornos administrados se implementan mediante hosts de motor de tiempo de ejecución. Las aplicaciones no administradas alojan Common Language Runtime, que permite al código administrado personalizado controlar el comportamiento del servidor. Este modelo proporciona todas las características de Common Language Runtime y la biblioteca de clases, además de obtener el rendimiento y la escalabilidad del servidor host.

En la ilustración siguiente se muestra un esquema de red básico donde se ejecuta código administrado en diferentes entornos de servidor. Los servidores como IIS (Internet Information Server) y SQL Server pueden realizar operaciones estándar mientras la lógica de la aplicación se ejecuta en el código administrado.

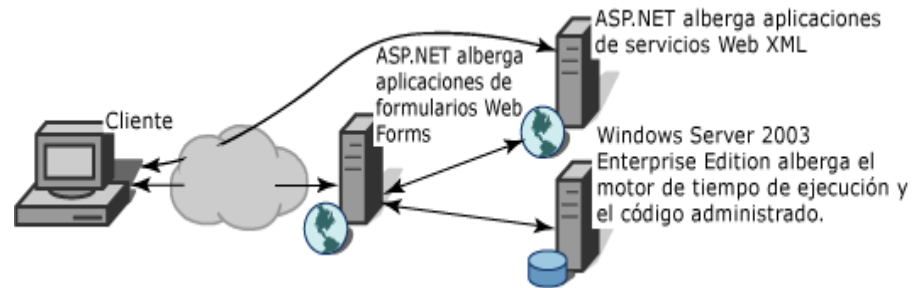


Fig 1.2: (Código administrado en el servidor)

1.3 Introducción a la tecnología ASP .NET y Web Services XML.

1.3.1 ASP .NET

ASP.NET es un marco de trabajo de programación generado en Common Language Runtime que puede utilizarse en un servidor para generar eficaces aplicaciones Web. Se pueden crear aplicaciones en cualquier lenguaje compatible con .NET, como Visual Basic .NET, C# y JScript .NET. Además, .NET Framework está disponible en su totalidad para cualquier aplicación ASP.NET.

ASP.NET está diseñado para funcionar sin problemas con editores HTML WYSIWYG (lo que se escribe es lo que se ve) y otras herramientas de programación como Microsoft Visual Studio .NET.

ASP.NET aprovecha las mejoras de rendimiento de .NET Framework y Common Language Runtime. Además, se ha diseñado para que ofrezca un rendimiento sensiblemente mejor que ASP y otras plataformas de programación Web. Todo el código de ASP.NET se compila, en lugar de interpretarse, lo que permite realizar enlaces en tiempo de diseño, establecer tipos inflexiblemente, compilar código nativo en modo Just-In-Time (JIT), entre otras muchas ventajas. ASP.NET ofrece unos completos servicios de almacenamiento en caché (tanto servicios integrados como API de almacenamiento en caché). ASP.NET se suministra con contadores de rendimiento que los programadores y los administradores del sistema pueden supervisar para probar nuevas aplicaciones y recopilar estadísticas de aplicaciones existentes.

Los formularios Web Forms permiten crear páginas Web basadas en formularios muy eficaces. Al crear estas páginas, se pueden usar controles de servidor ASP.NET para crear elementos comunes de la interfaz de usuario y programarlos para que realicen las tareas comunes. Estos controles permiten crear con rapidez un formulario Web Forms a partir de componentes integrados reutilizables o personalizados, con un código de página simplificado.

1.3.1.1 Seguridad en ASP . NET

La seguridad de los sitios Web es una cuestión de importancia fundamental, además de compleja. ASP.NET funciona junto con Microsoft .NET Framework y Servicios de Microsoft Internet Information Server (IIS) para ayudar a proporcionar aplicaciones Web seguras.

Para ayudar a proteger la seguridad de una aplicación ASP.NET, se deben realizar dos funciones principales :

Autenticación.- es la que ayuda a comprobar que el usuario es precisamente quien dice ser. La aplicación obtiene las credenciales (diversas formas de identificación, como nombre y contraseña) de un usuario, y las valida consultando a una autoridad determinada. Si las credenciales son válidas, se considera a la entidad que ha enviado las credenciales como una identidad autenticada.

Autorización.- Limita los derechos de acceso mediante la concesión o negación de permisos específicos a una identidad autenticada.

Además, Internet Information Services (IIS) puede conceder o negar el acceso en función de la dirección IP o del nombre de host del usuario. Cualquier autorización de acceso posterior se realiza mediante la autorización de la dirección URL del permiso de acceso al sistema de archivos NTFS.

Es importante entender cómo interactúan todos los diversos subsistemas de seguridad. Puesto que ASP.NET se basa en Microsoft .NET Framework, el programador de aplicaciones ASP.NET también tiene acceso a todas las características de seguridad integradas de .NET Framework, como la seguridad de acceso a código y la seguridad de acceso basada en funciones.

1.3.1.2 Arquitectura de Asp .NET

Todos los clientes Web se comunican con las aplicaciones ASP.NET a través de Servicios de Microsoft Internet Information Server (IIS). IIS descifra y, opcionalmente, realiza la autenticación de la solicitud. Si la opción Permitir anónimos está establecida en true, no se efectúa ninguna autenticación. IIS también busca el recurso solicitado (como una aplicación ASP.NET) y, si se autoriza al cliente, devuelve el recurso correspondiente.

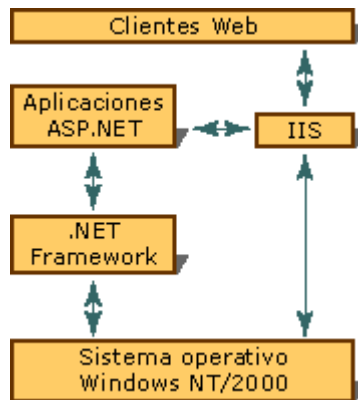


Fig 1.3: (Arquitectura ASP.NET)

1.3.2 Web Services XML

Un servicio Web XML es una entidad programable que proporciona un elemento de funcionalidad determinado, como lógica de aplicación, al que se puede tener acceso desde diversos sistemas potencialmente distintos mediante estándares de Internet muy extendidos, como XML y HTTP. Los servicios Web XML dependen en gran medida de la amplia aceptación de XML y otros estándares de Internet para crear una infraestructura que posibilite el funcionamiento conjunto de aplicaciones.

Un servicio Web XML puede ser utilizado internamente por una aplicación o bien ser expuesto de forma externa en Internet por varias aplicaciones. Dado que a través de una interfaz estándar es posible el acceso a un servicio Web XML, éste permite el funcionamiento de una serie de sistemas heterogéneos como un conjunto integrado.

En vez de centrarse en las posibilidades genéricas de portabilidad del código, los servicios Web XML proporcionan una solución viable para habilitar la interoperabilidad de datos y sistemas. Los servicios Web XML utilizan mensajería basada en XML como medio fundamental de comunicación de datos para contribuir a reducir las diferencias existentes entre entornos que utilizan distintos modelos de componentes, sistemas operativos y lenguajes de programación. Los programadores pueden crear aplicaciones que entrelacen servicios Web XML de una diversidad de orígenes, de modo similar a como utilizan tradicionalmente componentes en la creación de aplicaciones distribuidas.

Una de las características básicas de un servicio Web XML es el alto grado de abstracción existente entre la implementación y el consumo de un servicio. Al utilizar mensajería basada en XML como mecanismo de creación y acceso al servicio, el cliente del servicio Web XML y el proveedor de servicios Web XML no necesitan más conocimiento mutuo que el relativo a las entradas, las salidas y la ubicación.

1.3.2.1 Escenarios de Servicios Web XML.

Los escenarios de los Servicios Web XML sirven para comprender aun mejor la capacidad para desempeñar una función inestimable de utilidad, y son los siguientes:

- Servicios Simples
- Integración de aplicaciones
- Soluciones de flujo de trabajo

1.3.2.1.1 Servicios Simples

El motivo más elemental para el desarrollo de los servicios Web XML consiste en proporcionar alguna funcionalidad esencial para los clientes. Por ejemplo, un requisito de las aplicaciones de comercio electrónico suele ser la necesidad de calcular precios para distintas opciones de envío. Estas aplicaciones requerirían tablas actualizadas de costos de envío de cada compañía de envíos que se utilice para estos cálculos.

Como alternativa, una aplicación podría enviar un mensaje simple basado en XML a través de Internet, con un protocolo de transporte estándar como HTTP, al servicio Web XML de cálculo de costos del expedidor. El mensaje podría proporcionar el peso y las dimensiones del paquete, las ubicaciones de origen y de destino, además de otros parámetros, como la clase de servicio. El servicio Web XML del expedidor calcularía entonces el precio del envío a partir de la última tabla de costos, y devolvería el importe resultante a la aplicación que realiza la llamada, a través de un mensaje simple de respuesta basado en XML; esta cantidad se utilizaría para calcular el importe total que debe abonar el cliente.

1.3.2.1.2 Integración de aplicaciones

Puede utilizar servicios Web combinados para integrar una serie de aplicaciones aparentemente heterogéneas. La adopción generalizada de software personalizado en los departamentos de muchas empresas ha dado lugar a un gran número de datos y lógica empresarial muy útiles, pero aislados. Debido a la diversidad de circunstancias en las que se produce su desarrollo y al carácter siempre evolutivo de la tecnología, la creación de un ensamblado funcional a partir de estas aplicaciones es una tarea ingente.

Con los servicios Web XML es posible exponer la funcionalidad y los datos de cada aplicación como servicio Web XML. A continuación se puede crear una aplicación compuesta que utilice esta colección de servicios Web XML para habilitar la interoperabilidad entre las aplicaciones que la constituyen.

1.3.2.1.3 Soluciones de flujo de trabajo

Los servicios Web XML habilitan un eficaz mecanismo mediante el cual se pueden crear aplicaciones que constituyan soluciones de flujo de trabajo integrales. Estas soluciones son apropiadas para escenarios de larga ejecución, como los que amparan las transacciones entre empresas.

BizTalk Framework proporciona un nivel de protocolo adicional que define mecanismos para identificar y dirigir mensajes, definir su duración, empaquetarlos con archivos adjuntos, entregarlos de forma confiable a su destino, así como proteger su contenido para posibilitar y garantizar la autenticación, la integridad y la confidencialidad.

1.3.2.2 Infraestructura de Servicios Web XML

Los servicios Web XML deben ser independientes en lo que respecta a la selección del sistema operativo, modelo de objetos y lenguaje de programación con el fin de funcionar correctamente en la heterogeneidad del Web.

Para facilitar una adopción de los servicios Web XML tan generalizada como en el caso de otras tecnologías, estos servicios deben caracterizarse por lo siguiente:

- Correspondencia imprecisa: se considera que dos sistemas mantienen correspondencia imprecisa si la única condición impuesta a ambos consiste en comprender los mensajes de texto autodescriptivos mencionados con antelación.
- Comunicación ubicua: es poco probable que alguien cree, ahora o en el futuro, un sistema operativo que no incorpore la posibilidad de conexión a Internet, que por tanto equivale a un canal de comunicación ubicua. La posibilidad de conectar casi cualquier sistema o dispositivo a Internet garantizará que esos sistemas y dispositivos estén disponibles para cualquier otro sistema o dispositivo conectado a Internet.

- Formato de datos universal: mediante la adopción de estándares abiertos sobre métodos de comunicaciones patentados de bucle cerrado, cualquier sistema compatible con esos mismos estándares abiertos puede comprender los servicios Web XML. El uso de mensajes de texto autodestructivos que pueden compartir los servicios Web XML y sus clientes sin necesidad de conocer los sistemas subyacentes permite la comunicación entre sistemas autónomos y heterogéneos.

Los servicios Web XML emplean una infraestructura que proporciona lo siguiente: un mecanismo de descubrimiento para localizar servicios Web XML, una descripción de servicio para definir cómo se deben utilizar esos servicios y formatos de conexión estándar para la comunicación. El siguiente gráfico muestra la infraestructura de los Servicios Web XML:



Fig 1.4: (Infraestructura Servicios Web XML)

Los directorios de Servicios Web XML proporcionan una ubicación centralizada para localizar servicios Web XML proporcionados por otras organizaciones. Los directorios de servicios Web XML, como un registro UDDI, cumplen esta función. Los clientes de servicios Web XML pueden o no tener que referirse a un directorio de servicio Web XML.

El descubrimiento de Servicios Web XML es el proceso consistente en localizar, o descubrir, uno o varios documentos relacionados que describen un servicio Web XML determinado mediante Lenguaje de

descripción de servicios Web (WSDL). La especificación DISCO define un algoritmo para localizar descripciones de servicio. Si los clientes de un servicio Web conocen la ubicación de la descripción del servicio, pueden omitir el proceso de descubrimiento

La descripción de Servicios Web XML definen qué interacciones admite el servicio Web XML. Los clientes de servicios Web XML deben conocer cómo interactuar con un servicio Web XML para poder utilizarlo.

Los formatos de conexión de Servicios Web XML son protocolos que puede comprender cualquier sistema compatible con los estándares Web más utilizados. SOAP es el protocolo principal para la comunicación de servicios Web XML.

1.3.2.3 Periodo de duración de un Servicio Web XML.

El proceso que se produce cuando se realiza una llamada a un servicio Web XML es similar al que tiene lugar cuando se llama a un método regular. La diferencia principal es que, en vez de llamar a un método que se encuentra en la aplicación cliente, se genera un mensaje de solicitud sobre el transporte especificado, como HTTP. Dado que el método del servicio Web XML se puede encontrar en un equipo diferente, la información que tiene que procesar el servicio Web XML debe pasar por la red al servidor que aloja el servicio Web XML. El servicio Web XML procesa la información y devuelve el resultado, a través de la red, a la aplicación cliente.

En el diagrama siguiente se muestra el periodo de duración de un Servicio Web XML.

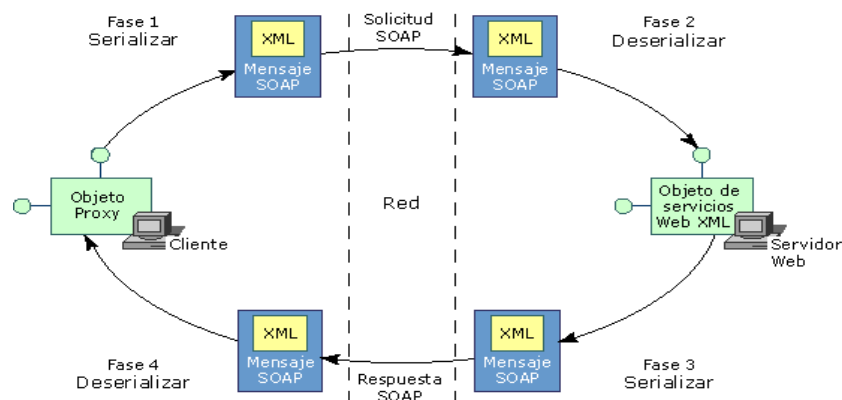


Fig 1.5: (Período de duración Servicio Web XML)

A continuación se describe la secuencia de eventos que se producen cuando se llama a un servicio Web XML:

1. El cliente crea una nueva instancia de una clase de proxy de servicio Web XML, este objeto reside en el mismo equipo que el cliente.
2. El cliente invoca un método en la clase de proxy.
3. La infraestructura del equipo cliente serializa los argumentos del método del servicio Web XML en un mensaje SOAP y lo envía a través de la red al servicio Web XML.
4. La infraestructura recibe el mensaje SOAP y deserializa el XML. Crea una instancia de la clase que implementa el servicio Web XML e invoca el método de servicio Web XML; el XML deserializado se pasa como argumentos.
5. El método de servicio Web XML ejecuta el código y establece finalmente el valor devuelto y los parámetros out.
6. La infraestructura del servidor Web serializa en un mensaje SOAP el valor devuelto y los parámetros de salida, y lo devuelve al cliente a través de la red.
7. La infraestructura de servicios Web XML, en el equipo cliente, recibe el mensaje SOAP, deserializa el XML en el valor devuelto y los parámetros out, y los pasa a la instancia de la clase de proxy.
8. El cliente recibe el valor devuelto y los parámetros out.

1.3.3 Servicios Web XML creados mediante ASP . NET

ASP.NET permite crear y publicar servicios Web XML mediante construcciones de programación conocidas, como métodos, tipos primitivos y tipos complejos definidos por el usuario. Además, .NET Framework contiene la infraestructura y las herramientas necesarias para crear clientes de servicios Web XML que pueden llamar a otros servicios Web XML compatibles con los estándares.

Con ASP.NET, puede crear servicios Web XML que utilizan estándares del sector para su implementación. Dado que los servicios Web XML creados se basan en estándares, se puede comunicar con ellos a través del Web mediante un cliente que se ejecute en cualquier plataforma que cumpla dichos estándares. En particular, los servicios Web XML emplean los estándares enumerados en la tabla siguiente.

Tabla 1.2: (Estándares Servicios Web XML)

| Estándar | Uso en servicios Web XML creados mediante ASP.NET |
|----------|---|
| XML | Formato de texto utilizado en la comunicación con los servicios Web XML mediante el protocolo SOAP. XML se usa para codificar las respuestas en la comunicación con servicios Web XML que utilizan los protocolos HTTP-GET y HTTP-POST. |

| | |
|---|---|
| SOAP | Protocolo de intercambio de mensajes basado en XML utilizado para llevar a cabo la comunicación entre los servicios Web XML y sus clientes. |
| Lenguaje de descripción de servicios Web (WSDL) | Describe el contrato de los mensajes que un servicio Web XML puede interpretar al comunicarse con un cliente de servicios Web XML. |
| XSD | Proporciona un sistema universal de tipos que permite la definición de tipos de datos y su transmisión entre plataformas. En un servicio Web XML, XSD define la estructura y los tipos de datos del código XML encapsulado en un mensaje SOAP enviado y recibido por un servicio Web XML. |
| application/x-www-form-urlencoded | Tipo MIME utilizado para codificar parámetros en una dirección URL. Dicha codificación se aplica a parámetros de solicitud para servicios Web XML que utilizan los protocolos HTTP-GET y HTTP-POST. |

La verdadera eficacia de los servicios Web XML se pone de manifiesto al considerar la infraestructura. Los servicios Web XML se construyen encima de .NET Framework y de Common Language Runtime. Un servicio Web XML puede aprovechar estas tecnologías. Por ejemplo, mediante la creación de servicios Web XML con ASP.NET, se puede aprovechar el rendimiento, la administración de estados y la compatibilidad con autenticación de ASP.NET.

La infraestructura que se genera para los servicios Web XML es compatible con estándares como SOAP, XML y WSDL, y permite a los clientes de otras plataformas interactuar con los servicios Web XML. Si un cliente puede enviar mensajes SOAP compatibles con los estándares, con un formato que se ajuste a una descripción de servicio, dicho cliente puede llamar a un servicio Web XML creado con ASP.NET (independientemente de la plataforma en la que resida el cliente).

1.4 Introducción a Oracle DataBase

1.4.1 Oracle 8i

La base de datos objeto-relacional Oracle es la primera creada para empresas dedicadas al manejo de datos empresariales, es la base de datos más flexible, y la que tiene una vía de costo beneficio, para manejar información empresarial. Esta base realiza recortes en los costos de manejo gerencial a la vez que proporciona la más alta calidad de servicio.

Algunas de las capacidades de esta base de datos son:

- √ Habilita el manejo complejo de código java y datos multimedia en el desarrollo de aplicaciones para el Internet.

- ✓ Mejora el soporte para gráficos o estructuras C++ y Java en datos encontrados en aplicaciones industriales, como son en telecomunicaciones, energía y finanzas.
- ✓ Facilita el código flexible y los conserva para los usos tradicionales de la base de datos.
- ✓ Habilita la base de datos para ser configurada en un dominio específico de una aplicación.

Combinando estas capacidades en un solo almacén de datos, la productividad de la industria, mayor robustez y desarrollo escalable, la base de datos objeto-relacional Oracle 8i definirá la siguiente generación de bases de datos.

Además de mejoras en la calidad y funcionamiento, la base de datos Oracle reduce notablemente los costes de manejo del entorno IT, con una simple instalación, los requisitos son altamente reducidos así como la configuración y manejo, y también proporciona un funcionamiento automático, estas y otras capacidades, ayudan a mejorar el DBA (Data Base Administration) y desarrollar la productividad y eficiencia.

Oracle es un producto integrado en tres importantes aspectos:

- ✓ Tiene una única arquitectura para definir ambos, objetos y datos relacionados
- ✓ El objeto tecnológico es combinar con los últimos avances en funcionalidad para data Warehousing y OLTP.
- ✓ Tiene una interfase consistente para el acceso de datos relacionados y objetos como SQL, Oracle Call Interface, JDBC, Oracle Objects for OLE y mucho más.

1.4.2 Arquitectura de Oracle DataBase 8i

Independientemente de la arquitectura computacional, o de su configuración, cada base datos dentro de oracle es asociada a una determinada instancia, y de igual una instancia puede abrir y utilizar solo una base de datos oracle en cualquier momento de su ejecución.

Cada vez que Oracle es ejecutado es inicializado, tanto el System Global Area (SGA) como los procesos demonios son levantados. El SGA junto con los procesos demonios son lo que se denomina una instancia.

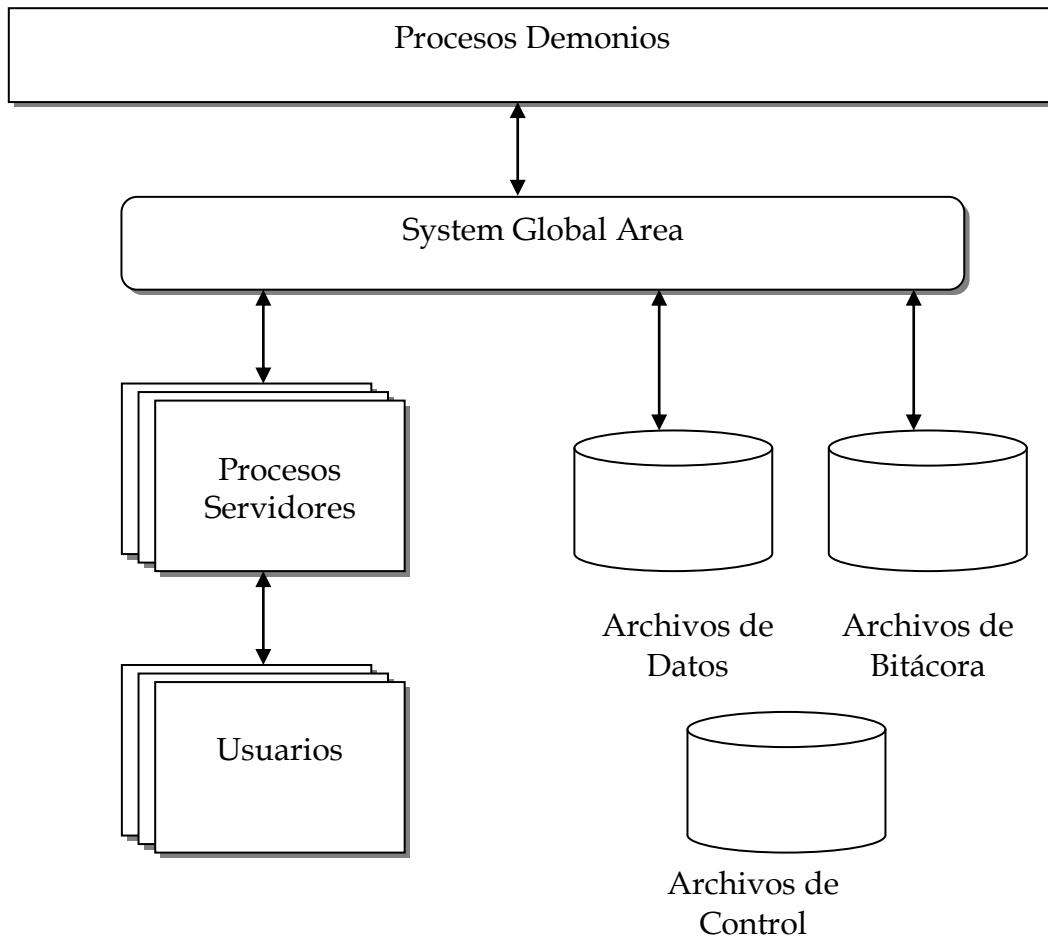


Fig 1.6: (Arquitectura de Oracle)

El System Global Area es un conjunto de estructuras de memoria compartida que contienen datos e información de control para una determinada instancia Oracle. El SGA se mantiene en la memoria virtual de computador en el que reside la instancia Oracle. Si dentro de la instancia existe la posibilidad de que más de un usuario se encuentre conectados simultáneamente, los datos dentro del SGA de la instancia son compartidos entre todos los usuarios.

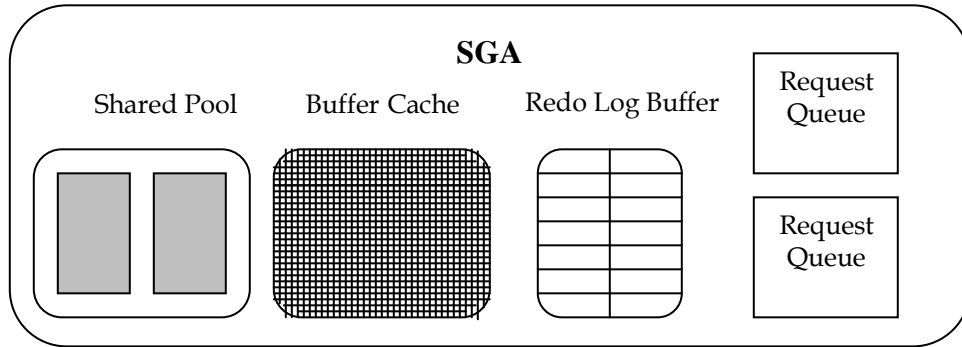


Fig 1.7: (Estructura Básica del System Global Area)

- Database Buffer Cache.- Su función es mantener bloques de datos leídos directamente de los archivos de datos. Cuando se procesa una consulta, el servidor busca los bloques de datos requeridos en esta estructura, Si el bloque no se encuentra en estructura, el proceso servidor lee el bloque de la memoria secundaria y coloca una copia en esta estructura. De esta forma, otras peticiones que requieran de este bloque de datos no requerirán de acceso a memoria secundaria.
- Redo Log Buffer.- Es un búfer circular que mantiene todos los cambios que han sido realizados sobre las bases de datos por operaciones de insert, update, delete, create, alter y drop. Las entradas de este búfer contienen toda la información necesaria para reconstruir todos los cambios realizados a la base de datos por medio de cualquier sentencia del DDL o del DML. El uso de Redo Buffer es estrictamente secuencial, en tal sentido pueden entrelazarse cambios en los bloques de datos producidos por transacciones diferentes.
- Shared Pool.- Esta estructura se utiliza durante el procesamiento de comandos y consta de la siguientes zonas:
 - Library Cache.- almacena la siguiente información relacionada con una instrucción de SQL:
 - √ Texto de la instrucción
 - √ Árbol de parsing, es decir la versión compilada de la instrucción
 - √ Plan de ejecución, es decir la secuencia de pasos a ser realizados para ejecutar la instrucción a bajo nivel de acuerdo a los resultados producidos por el optimizador de consultas.

Basando se en esta información, si una consulta es ejecutada nuevamente, y su información permanece todavía en el Library Cache, no será necesario compilar de nuevo la instrucción.

- Data Dictionary Cache.- también conocido como Dictionary Cache o Row Cache, almacena la información de uso más reciente sobre el

diccionario de datos. Esta información incluye definición de columnas, usuarios, passwords y privilegios. Durante la fase de compilación esta información es necesaria para resolver los nombres de los objetos utilizados en el comando SQL y para validar los privilegios de acceso.

- Request Queue y Response Queues.- Estas estructuras constituyen los elementos básicos para el manejo de conexiones de usuarios al manejador. Por cada instancia de Oracle existirá una Request Queue y varios Response Queues. Cuando un proceso usuario desea establecer conexión con el manejador, la solicitud de conexión será encolada en el Request Queue. Los procesos servidores tomarán las solicitudes de conexión y colocarán la petición en el Response Queues asociada al proceso que atendió la solicitud.

1.4.3 Procesos en Oracle.

Los procesos de Oracle realizan funciones para los usuarios del manejador, en general se puede establecer una partición de los procesos de Oracle.

- Procesos servidores o Shadow Proceses, los cuales ejecutan funciones de interacción con el servidor de datos basándose en las peticiones de los usuarios.
- Procesos Demonio o Background Proceses, quienes realizan funciones para el DBMS (Data Base Management System)
- Procesos de red, encargados de proveer la interconexión entre procesos de usuario y el DBMS o entre DBMS que establecen un sistema de bases de datos distribuidas.

1.4.3.1 Procesos Servidores

Los procesos servidores (Snnn) se comunican con los diferentes procesos de usuario e interactúan con Oracle para satisfacer las peticiones. Cuando un proceso de usuario solicita datos que no se encuentran en el SGA, el proceso servidor que atiende la petición será encargado de leer los bloques de datos y almacenarlos en el SGA. Puede existir una correspondencia de uno a uno entre procesos de usuario y procesos servidor; aunque un proceso servidor puede conectarse a múltiples procesos de usuario.

1.4.3.2 Procesos Demonios

Los procesos demonios, también conocidos como Background Proceses, constituyen programas que llevan a cabo funciones específicas de soporte y mantenimiento a la ejecución del servidor de base de datos.

1.4.3.3 Control de Recursos

Oracle administra el uso de los diferentes recursos que gestiona a través de “latches”. Un “latch” es un semáforo mantenido por la instancia de Oracle que permite administrar cierta actividad. Los “latches” limitan la cantidad de tiempo y espacio en los que un proceso puede mantener un recurso en un instante dado. El monitoreo de “latches” permite detectar situaciones en las que se crea contención por el acceso a un recurso. El número de “latches” existentes depende de la configuración y la información de los “latches” definidos y manejados por la instancia que puede ser extraída del diccionario de datos.

1.5 Introducción a Oracle Reports Developer

1.5.1 Oracle Reports 6i

Oracle Developer Reports es una de las herramientas más potentes y con mayores prestaciones del mercado orientada a la generación de informes.

La generación de los informes puede ser realizada desde procesos en lotes, en un entorno cliente-servidor o en entorno web.

La productividad que se puede alcanzar con Reports es muy alta, y puede llegar a realizarse un informe de producción sencillo en menos de un día.

Reports es una herramienta que puede ser utilizada tanto por usuarios novatos a través de sus asistentes como por usuarios experimentados que utilizan las características más avanzadas y complejas. Esta flexibilidad le convierte en una herramienta útil para todo el equipo de desarrollo y para toda la organización que ha de explotar la información almacenada en la base de datos.

Una de las ventajas principales de Oracle Reports es que está basado en un modelo page/frame, que prefiere la simplicidad, esto permite a los desarrolladores crear un reporte con múltiples consultas SQL sin tener un número limitado y además combinarlas con fuentes diferentes todo esto en un solo reporte.

Reports soporta SQL, PL/SQL, XML, JDBC, Oracle OLAP y archivos de texto en forma delimitada y en diferentes formatos, estas fuentes de datos son escritas en un API basado en Java que también permite crear y publicar otras fuentes de datos. Una clase en Java implementa una fuente de datos API que permite crear mecanismos de acceso a datos comunes e integrados dentro de los reportes.

1.5.2 Características y Beneficios

- Productividad. Oracle reports ofrece una alta productividad en el desarrollo de reportes para la web, ya que posee una poderosa herramienta en cuanto a tiempo

de creación y diseño combinada con un alto rendimiento en la ejecución de los mismos.

- ✓ Técnicas RAD (Rapid Application Design).
- ✓ Reutilización de componentes mediante clases y herencia.
- ✓ Integración con Designer 2000, para la generación automática.
- ✓ Interfaces con otros lenguajes (C, Java, ...).
- ✓ API abierta para la modificación de módulos.

- Escalabilidad. La creación de los reportes está basada en el concepto page-frame el cual posibilita la creación infinita de consultas en un solo reporte, permitiendo la escalabilidad constante así como también la compatibilidad entre sistemas operativos.

Además de:

- ✓ Funciona en una arquitectura multinivel.
- ✓ Optimización del tráfico de red (uso de arrays para sentencias DML).
- ✓ Posibilidad de distribuir el código entre el cliente, el servidor de aplicaciones y el servidor de base de datos.

- Sistema abierto. Reports es una herramienta que se la puede utilizar en un sin número de plataformas y con diferentes componentes, debido a que su estructura esta construida en Java y a los niveles de procesamiento que maneja. Algunos de los componentes que soporta son

- ✓ Controles OCX/ActiveX
- ✓ OLE (Object Link and Embedding)
- ✓ DDE (Dynamic Data Exchange)
- ✓ Acceso a cualquier base de datos (SQL Server, Informix, DB/2, y a cualquiera accesible vía ODBC como Access)

- Amigabilidad. Reports presenta una interfase para el desarrollo de reportes muy amigable, en la cual se puede utilizar la ayuda para generar un reporte Ejecutivo en pocos minutos así como también se puede utilizar las funciones más complejas (PL-SQL) para la creación de reportes gerenciales.

- ✓ Visualización dinámica
- ✓ Permite desarrollar aplicaciones con datos y gráficos interactivos.
- ✓ Informes a través de la web.
- ✓ Informes anidados (drill-down).
- ✓ Implantación en entornos web
- ✓ Permite implantar las aplicaciones en Internet o una intranet.
- ✓ Las aplicaciones cliente – servidor desarrolladas con versiones anteriores pueden ser llevadas a un interno multinivel.

√ La parte cliente se visualiza con un navegador estándar.

1.5.3 Desarrollo rápido

Usando un documento modelo de desarrollo céntrico y declarativo el desarrollador de Oracle Reports enfoca a un nuevo usuario en las tareas requeridas en la producción de un reporte, mientras permite a los desarrolladores experimentados tener el control total del desarrollo de su reporte. Cada tarea que requiere una mayor complejidad en la producción de un reporte es remitida para el uso de un Wizard apropiado, mediante el uso de un Report Templates y una pre-visión de la información existente que permite una rápida familiarización de la estructura del reporte una vez que las tareas desalentadoras, tales como modelos de información complejos y disposiciones de formato se vuelven simples acciones “drag and drop”.

Otra funcionalidad de Oracle Reports es el Query Wizard que permite la representación visual y construcción del complejo SQL en el cual se basan frecuentemente los reportes, en el cual el desarrollador de Report Wizard definirá una estructura del reporte, fuente de datos, grupos de quiebre y resúmenes.

1.6 Introducción a Oracle Discoverer

1.6.1 Oracle Discoverer 4i

Oracle Discoverer es una herramienta de acceso que permite a los profesionales del negocio visualizar la información de la base de su compañía. Discoverer es fácil de utilizar y proporciona una interfase lógica e intuitiva para la búsqueda de datos.

Oracle Discoverer maximiza las inversiones en la toma de decisiones, mientras minimiza el costo de entrenamiento de aplicaciones empresariales, redes y mantenimiento.

Discoverer es usado para construir consultas AD HOC usando datos de un área específica que son de interés para un grupo de usuarios, también provee información de datos sumariados hasta datos muy detallados.

Discoverer provee además portlets (componentes de java en el cliente) por estar integrado con Oracle 9iAS Portal, así como también mantiene soporte de bases de datos heterogéneas.

1.6.2 Características y Beneficios

Oracle Discoverer reúne estas características y permite que se logre los siguientes beneficios:

- Completo e Integrado.
- Visión para todos.
- Reduce los costos

Completo e Integrado.- Con la integración de Discoverer, Oracle DataBase, Oracle Application Server y Oracle E-Business Suite se simplifica en gran medida la seguridad, escalabilidad, acceso a datos y creación de metadatos además de reducir y mantener los costos e incrementar el desarrollo y la productividad de la empresa. La integración de Discoverer con Oracle Database reduce los costos de administración mientras incrementa la productividad, escalabilidad y el análisis de los usuarios en todos los niveles.

Visión para todos.- Fácil de usar es una de las más importantes consideraciones cuando se selecciona una herramienta Business Intelligent, por esta razón Discoverer puede ser usado por cualquier usuario de cualquier nivel con un mínimo de aprendizaje sobre la misma. Discoverer requiere de una única interfase con la cual los usuarios interactúan directamente con los datos usando una terminología de negocios que es familiar. Oracle Discoverer fue desarrollado en conjunción con Oracle Usability Laboratory y esta basado en pruebas de usabilidad intensiva.

Reduce los costos.- Los usuarios que accedan a la web necesitan ver la información en pocos segundos tan solo dando un click, sin embargo muchas veces son frustrados por un sistema de respuesta lento. Oracle Discoverer esta construido usando una estable arquitectura de componentes distribuidos n-tier, instalando los mismos alrededor de múltiples servidores mejorando una distribución virtual que maximiza el rendimiento.

Discoverer permite la creación de reportes ajustados a la necesidad del usuario final los cuales se pueden editar o dar un formato diferente, así como también permite crear gráficos estadísticos con múltiples estilos.

1.6.3 Arquitectura de Oracle Discoverer

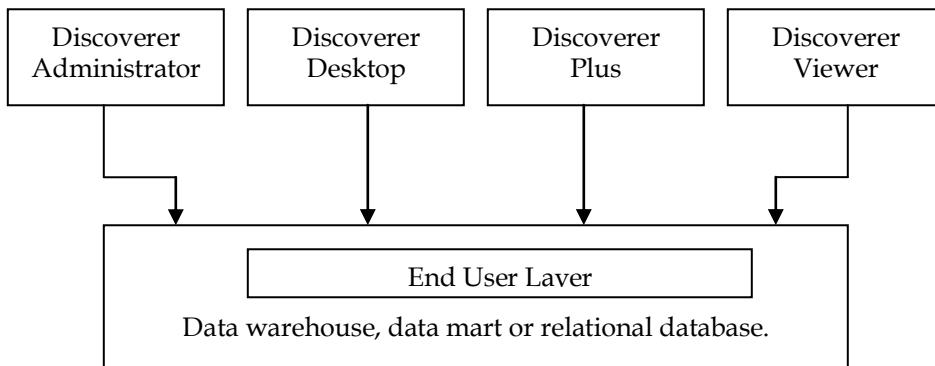


Fig 1.8: (Arquitectura de Oracle Discoverer 4.1.)

Discoverer Administrador.- este producto sirve para crear, mantener, administrar datos en la capa de usuario final (EUL), y para definir como los usuarios interactúan con los datos.

Discoverer Desktop.- como usuario final, usa este componente para realizar consultas AD HOC y generar reportes.

Discoverer Plus.- es una interfase basada en Web y puede ser accesada usando el browser preferido. Como usuario final, se puede usar este componente para crear, visualizar y editar los libros de trabajo sobre el Internet o intranet.

Discoverer Viewer.- este componente es útil para visualizar los workbooks y worksheets usando un browser. Los workbooks creados en Discoverer Desktop y los worksheets creado en Discoverer Plus pueden ser abiertos, pero no pueden ser editados usando viewer. Discoverer viewer hace la información disponible en el Internet o intranet a través de firewalls y permite que la información sea accesada desde portales corporativos.

End User Layer.- este componente es una capa de metadatos basada en el servidor que oculta la complejidad de la base de datos, permitiendo que los usuarios pueden interactuar con la base de datos sin tener que reconocer comandos de la misma. Esto permite a los usuarios usar el lenguaje de negocios con el que esta mas familiarizado

La herramienta discoverer se lo puede utilizar tanto para cliente/servidor como para Web esquematizando el siguiente diseño:

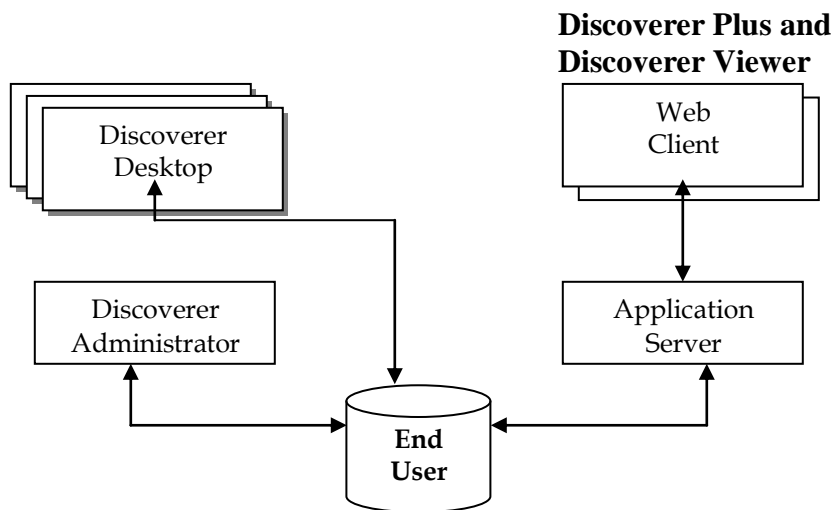
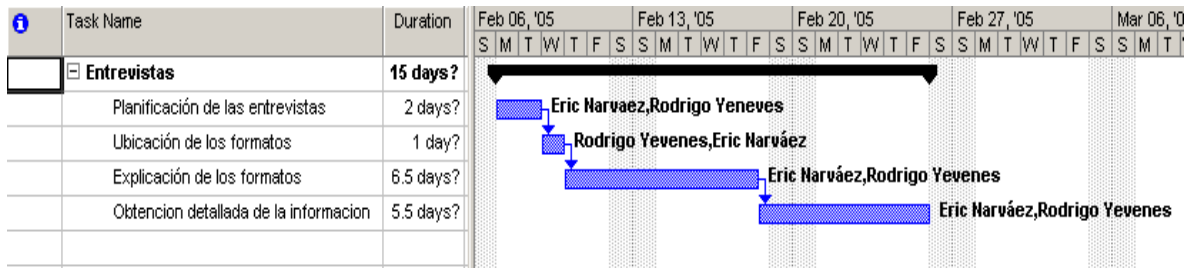


Fig 1.9: (Esquematización Cliente/Servidor)



2.2 Especificación de Requerimientos según el Estándar IEEE 830

2.2.1 Introducción

2.2.1.1 Propósito

El propósito es proporcionar una visión clara y objetiva de los requerimientos que deberán ser contemplados para la realización del Sistema de Operaciones Comerciales para Repsol YPF Ecuador.

La especificación de requisitos es definir de manera clara y precisa todas las funcionalidades y restricciones del sistema que se desea construir. El documento va dirigido al equipo de desarrollo y a los usuarios finales del sistema.

Una vez acabado este documento, servirá de base al equipo de desarrollo para la construcción del nuevo sistema.

2.2.1.2 Ámbito del sistema

Debido al continuo auge del uso de las redes (Internet) y a su facilidad de acceso, la empresa ha decidido crear un sistema que permita a sus usuarios ingresar, modificar y consultar información desde cualquier lugar que disponga de conexión a la intranet.

Con el nuevo sistema se espera aumentar los beneficios de la empresa, ya que será una herramienta fácil de manejar para el usuario, y se podrá mantener la información segura y disponible.

El software se denominará SOC (Sistema de Operaciones Comerciales), el mismo que estará encaminado a realizar las siguientes actividades:

√ Ventas.- Las ventas de la compañía y de los socios que las hacen diariamente, en donde consta: Gravedad del crudo, volúmen, precio, destino, la embarcación en la cual se carga, el tipo de crudo y el socio.

- Oriente
 - Bloque 16
 - Bloque 14
- Napo
 - Bloque 16
 - Shippers OCP
- Llenado de Línea OCP

√ Precios.-

- WTI Nymex.- Precio del crudo cotizado en la bolsa de valores de New York Mercantile Exchange.
 - de Referencia.- precios de referencial del crudo (Oriente, Napo).
 - Platts.- compañía global de información energética, y principal proveedor de los precios de cambio del sector energético.
- √ Producción Fiscalizada y Participación.- Son las participaciones que tiene tanto el estado como la compañía en los diferentes contratos.
 - Bloque 16
 - Bogi Capirón
 - Tivacuno
- √ Factores por °API.- Ajuste de los factores del calculo del API(American Petroleum Institute).
 - Factor K
 - API only coefficient
- √ Reporting mensual (imágenes), reportes de los ingresos de la compañía.
- √ Balance de entregas al OCP entre los socios del Bloque 16
- √ OCP
 - Transporte de crudo (Consortio y por socio del B16)
 - Programas de Embarque
 - Descuentos (Banco de Calidad, Consumo de Combustible, Pérdidas y Ganancias en la Línea)
 - Facturas por Transporte de crudo
 - Facturas por Compra de Combustible
- √ Facturación a Repsol-Ypf
 - Diferencial de Calidad OCP
 - Diferencia de Calidad SOTE
 - Diferencia de Precio
 - Tarifa de Transporte SOTE
 - Período de Comisión
 - Período de Pago
 - Reconciliación Según artículo 7.04 del ITSA
- √ Ineficiencia Petrolera.- Ineficiencia de la producción en miles de barriles y en la tasa de comercialización frente a años anteriores,

- ✓ Pagos Tivacuno.- Son los pagos que hace la compañía al banco central del contrato Tivacuno.

Uno de los beneficios más importantes que se obtendrán con el uso del sistema SOC será el de optimizar el manejo de la información, además de permitir acelerar el proceso de ingreso de datos y la emisión de reportes en relación a como se maneja actualmente.

Los principales objetivos que se esperan alcanzar con el sistema son:

- ✓ La interacción con el usuario,
- ✓ El manejo eficiente de los datos,
- ✓ El acceso fácil a la información,
- ✓ Operaciones y cálculos exactos,
- ✓ Acceso mediante permisos de usuario desde cualquier lugar de la Intranet,
- ✓ Creación de reportes estáticos y dinámicos,
- ✓ Minimizar el tiempo que con lleva el manejo de la información,
- ✓ Mantener la información consistente y disponible,
- ✓ Implementar niveles de seguridad, que permitan un mejor control del manejo y flujo de la información confidencial.

2.2.1.3 Definiciones, acrónimos y abreviaturas

2.2.1.3.1 Definiciones

Tabla 2.1 (Definiciones de la Especificación de Requerimientos)

| Término | DEFINICIÓN |
|----------------------|---|
| Equipo de desarrollo | Persona que se encarga del análisis, diseño, implementación y mantenimiento del sistema SOC. |
| Usuario del sistema | Persona registrada en el sistema SOC que tiene un perfil pre-designado. |
| Reportes Estáticos | Reportes que serán diseñados por el Usuario Administrador y que los demás usuarios no podrán modificar. |
| Reportes Dinámicos | Reportes que serán diseñados por el Usuario Operador de acuerdo a sus necesidades. |
| Niveles de Acceso | Permisos que tienen los usuarios para acceder al sistema. |
| Socio | Empresa que posee una parte las acciones del grupo. |
| Contrato | Áreas de influencia donde opera la compañía (Bloques) |
| Gravedad | Tipo de crudo (pesado, ligero) |
| Destino | País o región a donde se destina el crudo ó gas |
| Factores | Conjunto de formulas para hacer cálculos. |
| Buque | Embarcación en la cual se carga el crudo para su destino final. |

| | |
|----------------------|---|
| Volumen | Cantidad de crudo que se exporta (barriles por día). |
| Reportes del Platt's | Diario o revista donde se publican los precios del crudo. |
| | |

2.2.1.3.2 Acrónimos y Abreviaturas

Tabla 2.2 (Acrónimos de la especificación de requerimientos)

| Acrónimo & Abreviatura | Significado | |
|-----------------------------------|--|------------------|
| DBMS | Data Base Management System (Sistema administrador de base de datos) | |
| GUI | Graphic User Interface (Interfaz gráfica de usuario) | |
| SOC | Sistema de Operaciones Comerciales | |
| ST | Servidor de Transacciones | |
| TCP/IP | Transfer Control Protocol / Internet Protocol (Protocolo de Control de Transporte / Protocolo de Internet) | |
| IIS | Internet Information Services (Servicios de Internet) | |
| IAS | Internet Application Server (Sevidor de Aplicaciones.) | |
| WEB XML | Servicios XML Extensible Markup Language (Lenguaje de Marcado Extensible.) | |
| ASP .NET | Active Server Page .net (Paginas de Servidor Activas con tecnología .NET) | |
| API | American Petroleum Institute (Instituto de Petróleo Americano). | |
| SOTE | Sistema de Oleoductos Transecuatoriano. | |
| OCP | Oleoducto de Crudos Pesados | |
| YTD | Year to date (acumulado del año) | |
| ITD | Inception to date (acumulado desde el inicio de contrato) | |
| WTI Nymex | West Texas Itermediate (Crudo cotizado en la bolsa New York Mercantile Exchange) | Tipos de Crudos. |
| WTS | West Texas Sour | |
| OM | Oman | |
| BRT | Brent (Mezcla de petróleo crudo ligero que se cotiza en el International Petroleum Exchange) | |
| LLS | Light Louisiana Sweet | |
| BE | Balance de Entrega | |

| | |
|----------|------------------------------------|
| ER | Entrega Real |
| DE | Diferencias Excesivas |
| IT | Interés Trabajado |
| PP | Porcentaje de Participación. |
| DP | Diferencia de Precio |
| DCOCP | Diferencial de Calidad OCP |
| DCSOTE | Diferencial de Calidad SOTE |
| TTSOTE | Tarifa Transporte SOTE |
| CP | Commissioning Period |
| SoP | Ship or Pay |
| CCC | Consumo de Crudo Combustible |
| I(Kbls) | Ineficiencia en miles de barriles. |
| I(MUS) | Ineficiencia en Dolares. |
| Lldl | Llenado de Línea. |
| BT | Barriles Transportados |
| H | Hight |
| L | Low |
| DH | Descuento Hight |
| DL | Descuento Low |
| Bls/US\$ | Barriles por dolar |
| FOB | Valor total de Venta del Crudo. |
| BA | Barriles Ajustados |

2.2.1.4 Referencias

Referencias Internet:

- <http://www.platts.com>
- http://www.cali.gov.co/publico2/documentos/informatica/IEEE830_ext.pdf

Referencias Personales:

- Rodrigo Yevenes (Entrevistado), Asistente de Operaciones Comerciales.
- Patricia Valladares (Usuario), Asistente Financiera.

Referencias Técnicas:

- Hojas de calculo (Excel), utilizadas por los usuarios entrevistados.
- Modelos de clases.
- Modelos de Casos de Uso

- Diagramas varios.

2.2.1.5 Visión general del documento

A continuación se detallan los requisitos del software que serán contemplados en el sistema SOC, el cual está dividido en 2 partes.

- La primera que contempla la Descripción General de todos aquellos factores que tienen incidencia en los requerimientos del sistema.
- La segunda parte que se refiere a la descripción de los Requisitos Específicos en lo que tiene que ver con interfaces, funciones, requisitos de rendimiento, de diseño, atributos del sistema y otros.

2.2.2 Descripción General

2.2.2.1 Perspectiva del producto

El sistema SOC brindara entre los puntos principales la consistencia, seguridad, disponibilidad y acceso inmediato a los datos así como también la creación de reportes dinámicos que el usuario puede ajustar. La estructura del sistema SOC esta representado por el siguiente diagrama de bloques.

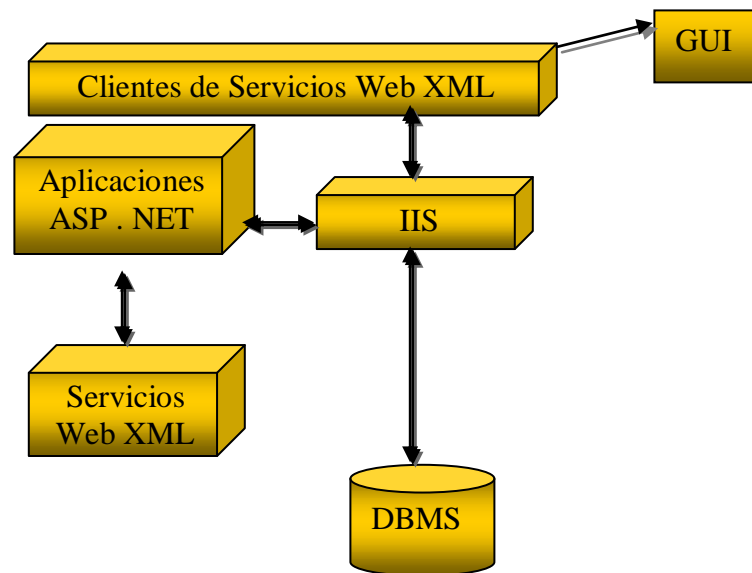


Fig 2.2: (Diagrama de Bloques del sistema SOC)

2.2.2.1.1 Interfaces del Sistema.

Las interfaces de la aplicación irán ajustadas de acuerdo a las normas corporativas de la compañía, a continuación se listan las interfaces generales con las que constará el sistema:

- ✓ Login: pagina de inicio de sesión.
- ✓ Principal: pagina que muestra el esquema general de aplicación, con menús y submenús.
- ✓ Buque: pagina con los datos de las embarcaciones que llevan el crudo.
- ✓ Partner: pagina con los datos de los socios que conforman Repsol-Ypf
- ✓ Destino: pagina con los datos de los destinos del crudo.
- ✓ TipoCrudo: pagina con los datos de los tipos de crudo.
- ✓ Contrato: pagina con los datos de los contratos que cuenta la compañía, (Bloques).
- ✓ Ventas Oriente: pagina con los datos de las ventas del crudo oriente.
- ✓ Ventas Napo: pagina con los datos de las ventas del crudo napo.
- ✓ Ventas Shippers: pagina con los datos de las ventas de las compañías al OCP.
- ✓ Llenado de Línea: pagina con los datos del llenado de línea de los contratos de la compañía entregados al OCP.
- ✓ FactorAPI: pagina con los datos de los factores de ajuste por °API (Factor K, Api Only Coefficient).
- ✓ Precio Referencia: pagina con los datos de los precios de referencia del crudo Oriente y Napo.
- ✓ Precios Platts: pagina con los datos de los precios reportados según del Platts.
- ✓ Comisión por período: pagina con los datos de la comisión por periodo de la compañía.
- ✓ Periodo Pagado: pagina con los datos de los pagos por período de la compañía.
- ✓ Reconciliación ISTA: pagina con los datos de la reconciliación según el artículo 7.04 del ISTA.
- ✓ Diferencial de Calidad SOTE: pagina con los datos del diferencial de calidad sote
- ✓ Diferencial de calidad OCP: pagina con los datos del diferencial de calidad OCP.
- ✓ Consumo de Combustible: pagina con los datos del Consumo de combustible.
- ✓ Ineficiencia Petrolera: pagina con los datos de la ineficiencia petrolera desde 1972.
- ✓ Pagos Tivacuno: pagina con los datos de los pagos tivacuno hechos por la compañía.

- ✓ Participación Bloque 16: pagina con los datos de las participaciones del estado y de la compañía tanto del bloque 16 como de Bogi-Capiron.
- ✓ Participación Tivacuno: pagina con los datos de las participaciones del estado y de la compañía del contrato tivacuno.
- ✓ ReporteMensual: pagina que muestra imágenes de los ingresos de la compañía.
- ✓ Descuentos del OCP: pagina con los datos de los descuentos que hace el OCP.
- ✓ Reportes: paginas por cada submenú que despliega los reportes de la aplicación.

2.2.2.1.2 Interfaces de Usuario

- ✓ Pantalla de inicio o ingreso de usuario. Esta constará de una imagen prediseñada y dos cajas de texto para el ingreso del usuario y el password correspondientes.
- ✓ Pantalla principal, en la cual se desplegara el menú principal de la aplicación ubicada en la parte superior de la ventana, para el acceso a todas las funciones. Esta estará conformada por imágenes, colores, botones, tipo de letra y diseño de acuerdo a las normas de la compañía.
- ✓ Por cada menú principal abra una pantalla con un submenú a la izquierda. Dependiendo del caso, por cada submenú habrá también otro submenú. El menú principal siempre se mantendrá visible a excepción que sé este manejando la interfase de seguridad del sistema.
- ✓ Pantallas independientes para cada proceso de la aplicación, se utilizarán hojas de estilos y controles de usuario, no frames. Los datos se presentarán en la parte central de la pantalla sin perderse el menú principal ni el submenú, además de botones para las operaciones básicas como; insertar, editar, eliminar y buscar datos.
- ✓ Para crear nuevos datos se creara o no nuevas pantallas dependiendo del número de datos a ingresar, si este número de datos es mínimo, los controles se presentarán en la misma pantalla.
- ✓ Pantalla única y diferente para la administración de la seguridad. Constara de un único menú principal en la parte izquierda de la pantalla. Los datos se presentarán en el centro de la misma. El menú principal desaparece.
- ✓ Pantallas individuales para la ejecución de reportes. Los link's a las consultas estarán ubicadas en los menús correspondientes, por cada submenú "consultar" abra otro submenú para especificar la consulta. Los parámetros para ejecutar la consulta se presentarán en el centro de la pantalla. Se escogerá los parámetros, el formato a presentar y se ejecutara el reporte.

- ✓ Cada pantalla cuenta con su propio método de navegación representado por una flecha que indica “regresar”.

2.2.2.1.3 Interfaces de Hardware.

- ✓ Los protocolos que se utilizan para la comunicación son el TCP/IP para la intranet y el SOAP para la comunicación entre el Web Services y la base de datos.
- ✓ Para los reportes se utilizará un despachador de reportes Oracle configurada en el puerto 9900.
- ✓ La configuración del software se la realizará e Internet Information Server 5.0.
- ✓ Se utilizará credenciales para el acceso a la aplicación.

2.2.2.2 Funciones del producto

El sistema brindará soporte para:

- ✓ Las ventas de crudo del Bloque 16 y Bloque 14 de cada uno de los socios,
- ✓ Las ventas de crudo del Bloque 16 y Bloque 14 del acumulado,
- ✓ Los precios de referencia del Crudo Napo,
- ✓ Los precios relacionados con los estándares internacionales (WTI),
- ✓ Las producciones fiscalizadas y participaciones del consorcio del Bloque 16, Bogi-Capiron y Tivacuno,
- ✓ Cáculo de los factores por °API,
- ✓ Los balances de entrega de crudo por los socios al OCP,
- ✓ El transporte de crudo, programación de embarques, descuentos y facturas por compra de combustibles,
- ✓ El banco de calidad del OCP,
- ✓ Llenado de línea al OCP en barriles por día y en dólares,
- ✓ Reportes del Platt's,
- ✓ Reportes mensuales.
- ✓ Calculo de la Ineficiencia Petrolera.
- ✓ Pagos tivacuno.

2.2.2.3 Características de los Usuarios

El sistema contemplara la existencia de tres tipos de usuarios:

- ✓ Usuario administrador, el mismo que deberá tener nivel de instrucción superior en el área de sistemas, ya que poseerá el control total sobre el sistema, es decir, la información, los reportes y de la Base de Datos.

- √ Usuario operador, el cual tendrá permisos suficientes para hacer operaciones sobre el sistema, deberá contar con conocimientos básicos de informática. Este tipo de usuario podrá manipular la información del departamento de Operaciones Comerciales y emitir los reportes.
- √ Usuario consultor, el cual tendrá permisos suficientes para consultar todos los reportes de la aplicación, deberá contar con conocimientos básicos de informática. Este usuario no podrá modificar los reportes ni crear nuevos.

2.2.2.4 Restricciones

2.2.2.4.1 Políticas de la empresa

Las operaciones y cálculos realizados por el personal del departamento de operaciones comerciales sobre los diversos formatos de calculo serán estrictamente definidos por ellos y el equipo de desarrollo no podrá hacer ninguna modificación.

Previo a la ejecución del proyecto se hizo un análisis tanto de las herramientas que se utilizaran para el desarrollo como de la infraestructura con la que la compañía dispone, por esta razón las herramientas se escogieron de acuerdo a los convenios que la compañía tiene con las empresas proveedoras de software (Microsoft, Oracle) y a la infraestructura con que cuenta la compañía actualmente.

Los datos que maneja el departamento de Operaciones Comerciales son altamente confidenciales, por esta razón los únicos que tendrán acceso a los mismos son: El equipo de desarrollo y los usuarios del sistema.

2.2.2.4.2 Interfaces con otras aplicaciones

La información del departamento de Operaciones Comerciales actualmente se encuentra expuesta en formatos Excel, por esta razón no se desarrollara ninguna interfase con estos formatos.

El sistema SOC tendrá una interfase con la aplicación denominada Oracle Discoverer la cual presentara los reportes diseñados por el usuario final.

El sistema SOC tendrá una interfase con la aplicación denominada Microsoft Word la cual presentara los documentos (.doc) relacionados con el departamento.

El sistema SOC tendrá una interfase con la aplicación denominada Adobe Acrobat la cual presentara los formatos (.pdf) relacionados con el departamento.

2.2.2.4.3 Protocolos de comunicación

El protocolo de comunicación que se utilizara para el funcionamiento del sistema SOC es el protocolo TCP/IP debido a que es un protocolo para Internet. Actualmente la compañía trabaja con este protocolo.

2.2.2.4.4 Requisitos de fiabilidad

Tanto el servidor de base de datos y el servidor de aplicaciones estarán ubicados en Quito se obtendrá siempre una respuesta inmediata a las operaciones que se le hagan al sistema y por ende no habrá cortes imprevistos en el funcionamiento del mismo.

La aplicación que se desarrollara tiene como fundamento principal ser multi-capa y multi-usuario, razón por la cual, tanto los equipos servidor como los equipos cliente deberán estar conectados en red y funcionando al menos a 128 Kbps.

2.2.2.4.5 Consideraciones acerca de la seguridad

Dado que el sistema permitirá el acceso de usuarios desde la Intranet de la Compañía, se manejarán perfiles de usuario que no tengan la capacidad de modificar la información.

En el grupo de usuarios que manejan la aplicación se definirán perfiles pre-establecidos para evitar el daño de la información

El Sistema SOC también constara de un nivel de Administración de Usuarios y Mantenimiento.

2.2.2.4.6 Requisitos futuros

En el futuro, el Sistema SOC podrá ser mejorado, de tal manera que abarque los procesos adicionales que surjan. En esta versión solo se abarcaran una parte de los archivos que se encuentran con formato Excel y que corresponden a los módulos citados, por esta razón el sistema será construido para agregar nuevos módulos.

2.2.3 Requisitos Específicos

2.2.3.1 Interfaces externas

A continuación se especificarán los requisitos que afectaran a la interfaz de usuario e interfaz de comunicaciones.

2.2.3.2 Interfaces de Usuario

La aplicación funcionará en un entorno navegador, la interfaz se mostrará al usuario en forma de página web.

Para cumplir con los requisitos del usuario, la GUI deberá ser sencilla de utilizar, con barras de herramientas y botones que permitan una fácil navegación.

El acceso de los usuarios al sistema desde la Intranet se lo realizara mediante el uso de un navegador, de preferencia Internet Explorer 5.0 o superior.

El usuario para poder utilizar la aplicación deberá ingresar un login y un password en la interfase que se presentara cuando se ejecute la aplicación, siempre y cuando este haya sido creado por el usuario administrador y disponga de los permisos suficientes.

Las interfaces de los reportes estáticos se presentarán de igual forma en un navegador, estos podrán ser accesados por todos los usuarios que utilicen la intranet.

2.2.3.3 Interfaces de Software

Se utilizará una arquitectura n-tier, Cliente/Servidor de transacciones/Servidor de base de datos /Despachador de reportes Oracle Report's/Despachador de reportes Oracle Discovery. Todas las terminales que utilizarán la aplicación estarán conectadas a la intranet de la compañía para comunicarse.

2.2.3.4 Interfaces de comunicaciones

Los usuarios que utilicen el sistema se comunicaran con los servidores de base de datos, transacciones, reportes y paginas Web mediante el protocolo TCP/IP que maneja la aplicación.

2.2.3.5 Funciones

√ Al iniciar la aplicación el usuario deberá ingresar el nombre de usuario y su contraseña correspondiente, dependiendo de esta el sistema mostrará los permisos asignados al usuario. Una vez el usuario ingresado en la aplicación deberá hacer uso del menú

principal para ingresar a las pantallas correspondientes e ingresar o manipular los datos.

- √ Las ventas de la compañía están divididas por el tipo de crudo que se extrae y por el contrato, además de las ventas que tienen las otras empresas dedicadas a la explotación de hidrocarburos, el formato en si de cada una de estas es la misma ya que se utilizan los mismos datos es decir: el destino a donde se envía el crudo, la embarcación en la que se va enviar, la compañía que lo envía, el tipo de crudo, el volumen de crudo exportado, la gravedad del crudo, el precio por barril y el precio total.

Estos datos serán ingresados día a día o según convenga al operador del sistema, además calculará el valor total de las ventas tanto del consorcio como de cada uno de los socios.

Los parámetros para generar los reportes de las ventas son: Año, mes, socio y el destino, por cada uno de estos habrá la opción de listar todos o el que escoja el usuario.

- √ Los precios, se manejará toda la información relacionada con los precios de referencia de Napo y Oriente, los precios del Nymex WTI y los precios del Platts. Estos datos serán ingresados por el usuario operador día a día y se calculará promedios mensuales y acumulados al año.

Cuando el usuario operador ingrese los datos del precio de referencia de oriente el sistema automáticamente calculará y guardara en la base de datos el Factor K.(Factor por °API). Estos datos se mostraran en la pantalla Ajuste por °API.

La formula para calcular el factor k:

Factor K = 1.3/100*Precio de Referencia Oriente.

Los parámetros de consulta para los reportes WTI y Platts serán: Año, mes y el día, dependiendo de que desea consultar el usuario, el sistema desplegara un resumen de los precios o a su vez el listado completo de los mismos. En el caso del reporte para Precios de referencia, los parámetros serán: Año, mes y el tipo de crudo (Oriente, Napo), dependiendo de lo que el usuario escoja el reporte desplegara solo uno de los dos tipos.

- √ Las producciones fiscalizadas y participaciones del consorcio, el usuario operador podrá ingresar el total de Producción y el Factor X Calculado mes a mes, a partir de estos datos se podrá obtener las participaciones de la contratista y las participaciones del estado. Además constará de dos tablas de mantenimiento en las cuales se podrá ingresar mes a mes las producciones del Bloque 16 y de Bogi-Capiron

Producción Bloque = Producción total / No. Días

Factor x1, Factor x2, Factor x3, Limite L1, Limite L2, para este caso son constantes.

Producción q1 = Si la producción por bloque es menor que el Limite L1, la producción q1 es igual a la producción por bloque, en caso contrario, la producción q1 es igual al limite L1 menos 1.

Producción q2 = Si la producción por bloque es mayor o igual al limite L1 y la producción por bloque es menor o igual al limite L2, la producción q2 es igual a la producción por bloque menos la producción q1, caso contrario, si la producción por bloque es mayor que el limite L2, la producción q2 es igual al limite L2 menos el limite L1 mas 1, en caso contrario la producción q2 será 0.

Producción q3 = Si la producción por bloque es mayor que el limite L2, la producción q3 es igual la producción por bloque menos la producción q1 menos la producción q2, en caso contrario la producción q3 será 0.

Factor x Calculado = Si la producción total es igual a cero, el factor x calculado será igual a cero, caso contrario, el factor x será igual a la sumatoria del factor x1 por la producción q1, el factor x2 por la producción q2 y el factor x3 por la producción q3 dividido para el total de la producción q.

Participación Contratista = la producción total por el factor x calculado.

Participación del Estado = la producción total menos la participación de la contratista.

De la misma manera se procederá con las producciones y participaciones de Tivacuno, en este caso, adicionalmente constará de dos tablas informativas de los Factores X (Contratos antes de agosto, Extensión de contratos.), de los cuales se obtendrá porcentajes dependiendo del precio Wti, como se muestra en las siguientes formulas:

Producción Tivacuno = producción total dividido para el numero de días.

WTI Nymex = este dato se traerá desde la clase WTI mediante la fecha.

Factor x1= Si el precio WTI es igual a cero, entonces el factor x1 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.75%, entonces el factor x1 es igual a x1 correspondiente de la clase Contratos antes de Agosto menos 0.01.

Factor x2= Si el precio WTI es igual a cero, entonces el factor x2 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.70%, entonces el factor x1 es igual a x2 correspondiente de las clases Contratos antes de Agosto menos 0.01.

Factor x3= Si el precio WTI es igual a cero, entonces el factor x3 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.61%, entonces el factor x3 es igual a x3 correspondiente de la clase Contratos antes de Agosto menos 0.01.

Limite L1 y Limite L2= son valores constantes.

Producción q1= si la producción tivacono es menor que el limite L1, entonces la producción q1 es igual a la producción tivacono, caso contrario, la producción q1 será igual al limite L1 menos uno.

Producción q2= si la producción tivacono es mayor o igual que el limite L1 y la producción tivacono es menor o igual que el limite L2, la producción q2 es igual a la producción tivacono menos el limite L1, caso contrario, si la producción tivacono es mayor al limite L2, la producción q2 es igual al limite L1 mas uno, caso contrario, la producción q2 es igual a 0.

Producción q3 = Si la producción tivacono es mayor o igual que el limite L2, la producción q3 es igual la producción tivacono menos la producción q1 menos la producción q2, en caso contrario la producción q3 será 0.

Factor x Calculado = el factor x es igual a la sumatoria del factor x1 por la producción q1, el factor x2 por la producción q2 y el factor x3 por la producción q3 dividido para el total de la producción q.

Participación Contratista = la producción total por el factor x calculado.

Participación del Estado = la producción total menos la participación de la contratista.

En los reportes se mostrará las participaciones del estado y del consorcio de cada uno de los contratos, así como también las producciones totales en una fecha preestablecida.

- √ Cálculo de los factores por °API, el cálculo del factor K se lo realizará con los datos de los Precios de Referencia de Oriente, y el Api Only Coefficient deberá ser ingresado por el usuario. Los parámetros de consulta para generar los reportes de los Factores por °API serán: Año, mes y el factor, entendiendo como factor, (Factor K y Api Only Coefficient). Dependiendo de lo que es usuario haya escogido el reporte desplegara solo uno de los dos factores.

- √ Los balances de entrega de crudo por los socios al OCP, el usuario operador podrá ingresar los datos de los barriles entregados al OCP día a día de Repsol YPF y Petrobras, a partir de estos datos se podrá calcular las entregas por interés trabajado, las entregas reales, las diferencias excesivas y el balance ITD de cada uno de los socios, como se muestra en las siguientes formula:

Total Entrega: barriles entregados de Repsol YPF más barriles entregados de Petrobras.

IT: Total de entrega por el porcentaje de participación de cada socio.

$$IT=TE*PP$$

ER: Total de entrega por el porcentaje de participación dividido para el 80 por ciento de cada uno de los socios.

$$ER=TE*(PP/80\%)$$

DE= Entrega real por el interés trabajado de cada uno de los socios.

$$DE=ER*IT$$

ITD= ITD del la fecha anterior más la entrega real menos el interés trabajado.

Con los datos completos se podrá generar reportes del balance por mes y el acumulado.

- √ El transporte de crudo, programación de embarques, descuentos y facturas por compra de combustibles, el usuario podrá ingresar los datos relacionados con el Banco de Calidad del OCP, Pérdidas y Ganancias en la Línea y el Consumo de Combustible, de cada una las empresas externas petroleras, en un periodo determinado o mes a mes. Se calculará un total general de cada empresa y los reportes será listados por mes y año. Estos datos serán calculados con sumatorias totales de la facturas.

En cuanto a la programación de embarques son documentos que se encuentran en un formato específico y que el operador podrá consultarlos desde la aplicación, así como también cargar los documentos en la aplicación.

- √ Llenado de línea al OCP en barriles y en dólares, en el llenado de línea en bbls, se ingresarán datos del volumen vendido del campo sur y de Bogi-Capirón así como la gravedad API correspondiente, con estos datos y el precio de venta se calculará el monto vendido de cada campo que corresponde al llenado de línea en dólares.

Los parámetros para generar este reporte serán: Año y el mes, con la posibilidad de escoger todos en cada uno de estos.

- √ OCP, la facturación OCP se llevan los datos de: Período de Comisión, Período de Pago, Reconciliación según el artículo 7.04 del ISTA y reembolso de combustible, los datos que se ingresan

son de las facturas que se obtiene del OCP y que contienen, el volumen transportado por campo y el total. Se calculará un total general y los reportes serán consultados por año y mes.

- √ Facturación a Repsol-Ypf, en donde se llevarán los datos del Diferencial de Calidad OCP, el Diferencial de Calidad SOTE, Tarifa de Transporte por SOTE y Diferencia de Precio, en estas clases se ingresarán los datos que se muestran en las facturas que recibe Repsol-Ypf los cuales serán creados día a día por el usuario operador se calculará un total general por cada fecha ingresada y los reportes serán consultados por año y mes.
- √ En la ineficiencia petrolera se ingresarán los datos de la producción y el precio, a partir de estos se calculará la ineficiencia frente a años anteriores tanto de ineficiencia en miles de barriles como la ineficiencia en la tasa de comercialización, los reportes serán consultados por año.
Para realizar los cálculos correspondientes se seleccionara un año base y partir del siguiente se calculará las ineficiencias

IP(Kbls)≡ Producción actual menos la producción del año anterior.

IP(MUS\$)≡ Ineficiencia de miles de barriles por el precio actual.

- √ En los pagos tivacuno, se ingresarán datos de la retención, producción, impuestos, ley 122, precio contratado, el pago que se hace y los períodos correspondientes, con estos datos se calculará el total del precio contratado así como también la diferencia entre el precio contratado total y el pago.

Total precio contratado: Precio contratado menos la retención menos la ley 122.

Diferencia: pago total menos el total precio contratado.

Los parámetros para generar el reporte serán: Año y mes con la posibilidad de consultar todos.

- √ La seguridad en el sistema será manejada únicamente por el usuario administrador de la aplicación y aquí se podrá definir a los usuarios, los roles y los permisos que estos tienen sobre el sistema.

2.2.3.6 Requisitos de rendimiento

El sistema SOC es un software diseñado para la web por lo que puede correr en cualquier plataforma y ser accesado por muchos usuarios a la vez.

Además se pretenderá los siguientes puntos:

- ✓ Dispondrá de una base de datos consistente y fiable,
- ✓ Trabaje a tiempo real garantizando la rapidez en la comunicación.
- ✓ Garantice la seguridad de la información.
- ✓ La información que manejará la base es totalmente numérica. Los datos calculados se los generará mediante código o consultas a la base, ningún dato de estas características va ser guardado.
- ✓ Las consultas, ingreso y mantenimiento de los datos se los realizará día a día.
- ✓ Los respaldos se los realizarán cada semana.
- ✓ Las limitaciones de capacidad de acceso no están definidas, los usuarios podrán acceder en cualquier momento y simultáneamente.

2.2.3.7 Atributos del sistema

SOC es un sistema orientado a la Web que maneja usuarios y niveles de acceso, además de guardar la información en una base de datos que solamente personal autorizado puede utilizar.

El sistema también constara con la emisión de reportes que estarán orientados tanto a la web como al escritorio mediante el uso de herramientas Oracle.

Los reportes que están orientados al escritorio, podrán ser creados por el usuario operador y el administrador según las necesidades que se presenten.

2.2.3.8 Atributos del Software

a) Seguridad

Es importante que tengamos los mecanismos de seguridad bien definidos. Por eso cuando un usuario intente conectarse al sistema, deberá introducir un nombre de usuario y una contraseña, si un usuario del sistema intenta acceder, y no introduce bien el identificador o la contraseña, el programa dará un mensaje de error advirtiendo que el usuario y la contraseña no coinciden con ningún identificador registrado.

Es también importante que las zonas a las que sólo pueda acceder el administrador estén seguras, si un usuario no autorizado intenta entrar, se le mostrará un mensaje informándole de que no está autorizado a acceder en ese dominio.

b) Disponibilidad

En cuanto a la disponibilidad y confiabilidad de la información el sistema siempre estará disponible y consistente, no habrá cortes imprevistos en la ejecución del mismo, ya que el sistema correrá desde la intranet de la compañía.

c) Mantenimiento

El mantenimiento de la aplicación se la podrá realizar remotamente y no se detendrá la ejecución del mismo en caso de demora. Lo fuentes estarán ubicados en la intranet de la compañía y solo podrán ser accedidos por el administrador del sistema, aquí se harán los cambios necesarios y luego se los subirá a producción sin detener la ejecución del mismo.

d) Portabilidad

La portabilidad del sistema estará ajustada a los requerimientos de la herramienta en la cual se desarrollo. La aplicación correrá en el cliente desde cualquier explorador de Internet, mientras que el servidor de las páginas Web tendrá que ser Windows no así de los Servicios Web.

Los diferentes usuarios y roles que se utilizarán en la aplicación para la base de datos, reportes de Oracle Reports y Oracle Discoverer, se listan a continuación:

Usuarios:

ADMSOC, administrador de la base de datos del Sistema de Operaciones Comerciales

REPORTING_SOC, usuario para acceder a los reportes de Oracle Report's

SOC_DISC, que es el dueño del EUL.

DISC_ADMIN., que es el que sirve para crear los reportes estándar de Discoverer y guardarlos en la base de datos.

DISC_PROA, que es el que ejecuta los reportes creados por el DISC_ADMIN

ADHOC_SOC, es un usuario con permisos de consulta sobre los objetos de ADMSOC, y con posibilidad de crear sus propios objetos. A este usuario se conectan los de Discoverer y no directamente al ADMSOC.

Roles:

ROLE_ON_ADMSOC, el cual posee privilegios de SELECT sobre las tablas del usuario ADMSOC.

ROLE_ON_ADHOC_SOC, el cual posee privilegios de SELECT sobre las tablas del usuario ADHOC_SOC

2.2.3.9 Estándares del software.

Está generalmente aceptado que la existencia de estándares para cualquier tipo de producto o actividad es siempre beneficiosa, aunque casi siempre supone un aumento de la rigidez y un inconveniente, al aumentar los requisitos que debe cumplir cualquier cosa para ajustarse a dichos estándares.

Pero por otra parte, entre los beneficios que puede aportar el disponer de un estándar de interfase de usuario es claro el hecho de que al estar establecido un modelo reduce el tiempo que se emplearía en evaluar posibles alternativas, al venir estas ya fijadas por el correspondiente estándar.

Con ello se dispone de más tiempo para el diseño de funciones específicas del negocio, más fácil portabilidad de módulos de sistema, más fácil mantenimiento del sistema y menos tiempo utilizado en el soporte de las consultas de usuarios sobre la utilización de la interfase, reduciéndose también con ello el coste de formación y entrenamiento de los usuarios finales en el uso de las aplicaciones.

Los criterios tomados en cuenta a la hora de diseñar nuestra pagina web son: Presentación, Interacción, Navegación y Contenidos.

Dichas funciones constituyen la ecuación de equilibrio fundamental, que interaccionando entre ellas y debido a sus interdependencias, nos darán las características de la interfase y su nivel de calidad.

En la Presentación intervienen las cualidades de: estética, metáforas y aprendizaje, en tanto que para la Interacción son requeridas las cualidades de: usabilidad, prevención y comprensibilidad.

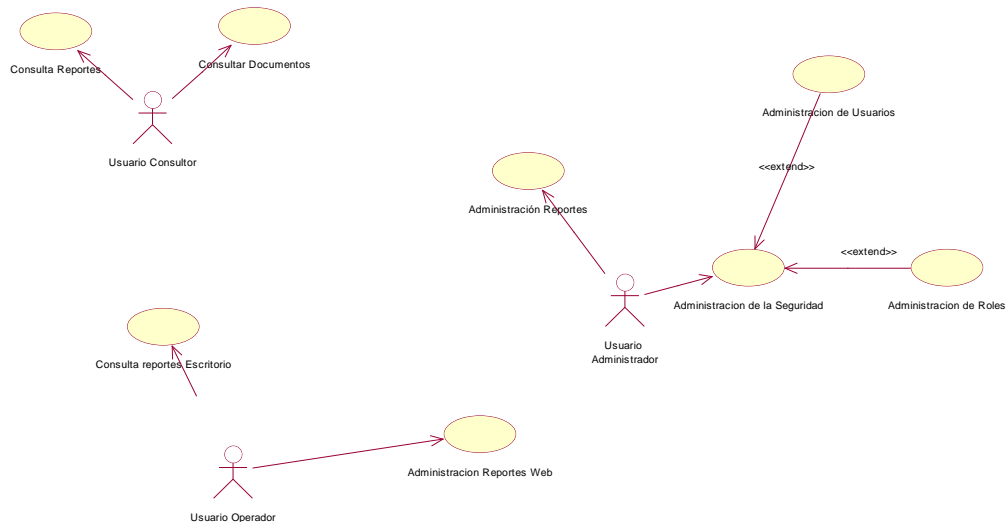
Los Contenidos constituyen el objetivo final de la aplicación y por ello son uno de los elementos fundamentales de la aplicación que supeditan claramente la función de Navegación, que se encuentra relacionada con las cualidades de: consistencia, intuitividad y simplicidad.

Algunos de los conceptos aplicados son:

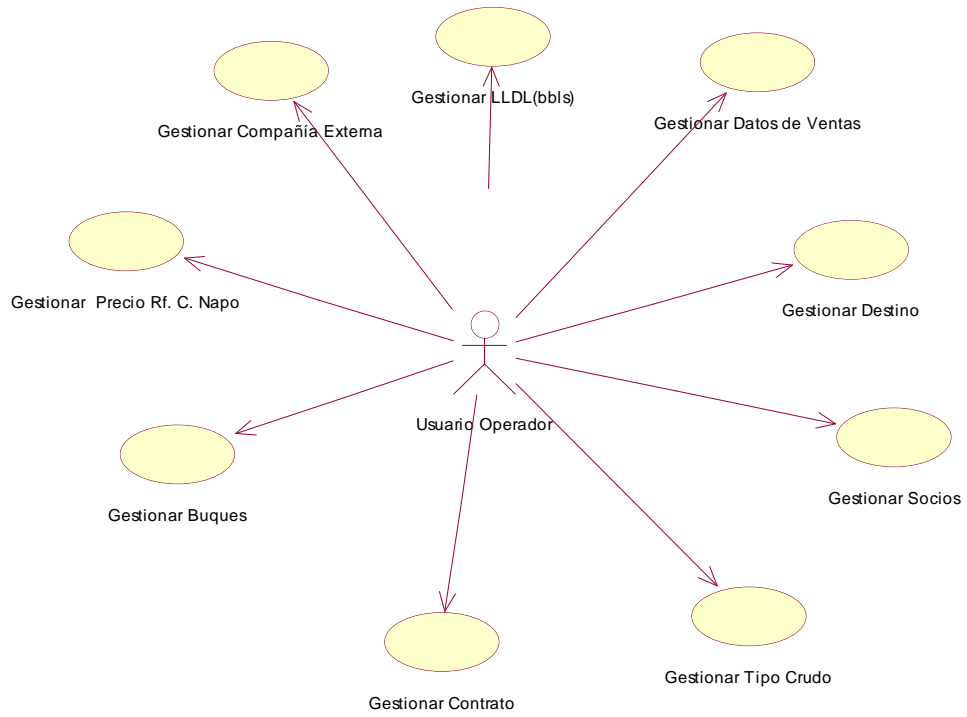
- Toda la información disponible se encuentra visible por el usuario.
- Muestra el flujo de trabajo para que el usuario se familiarice con él.
- Agrupa elementos por funciones.
- Botones de comando con etiqueta para una fácil comprensión de su función.
- Mostrar los datos no editables mediante el uso de etiquetas en lugar de campos de edición desactivados.
- Descripciones complementarias mediante tooltips
- Facilitar la lectura de los registros utilizando distintos fondos de líneas.
- Facilitar el acceso a las funciones disponibles, mediante el uso de menús de funciones, mostrando el flujo de trabajo.
- Alinear correctamente los campos según el tipo.
- Clara diferenciación del los registros en la tabla.

2.3 Construcción de Casos de Uso.

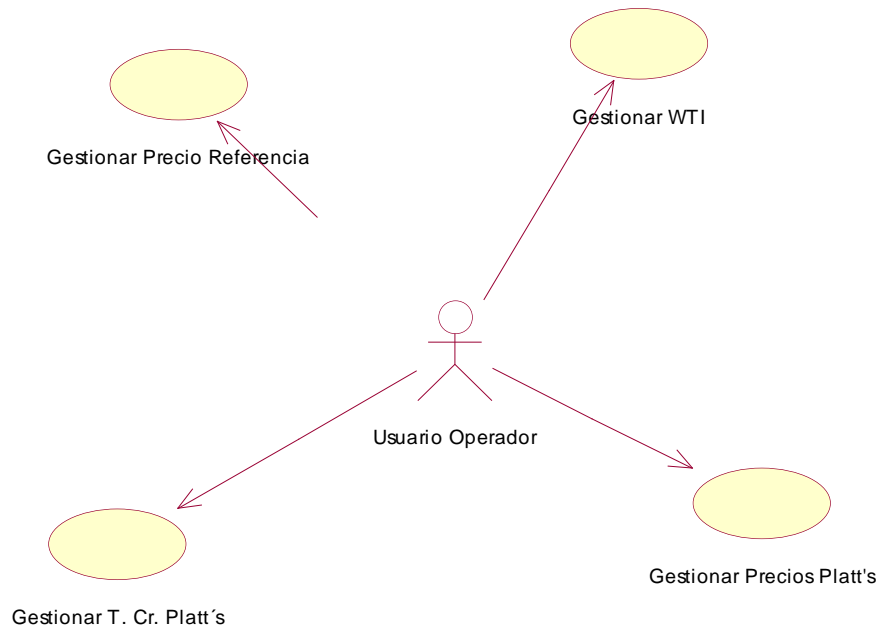
2.3.1 Seguridad y Reporte.



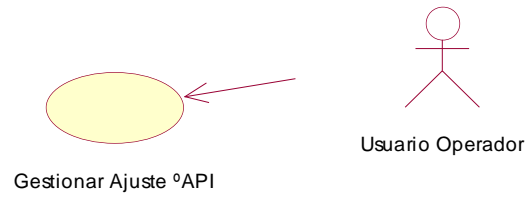
2.3.2 Ventas.



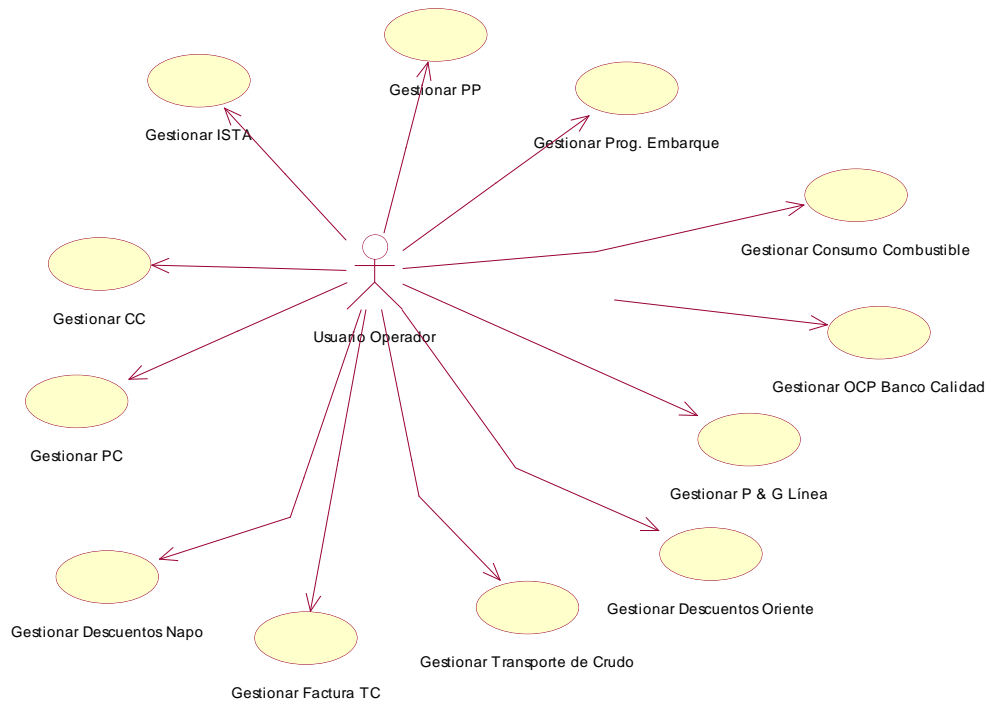
2.3.3 Precios.



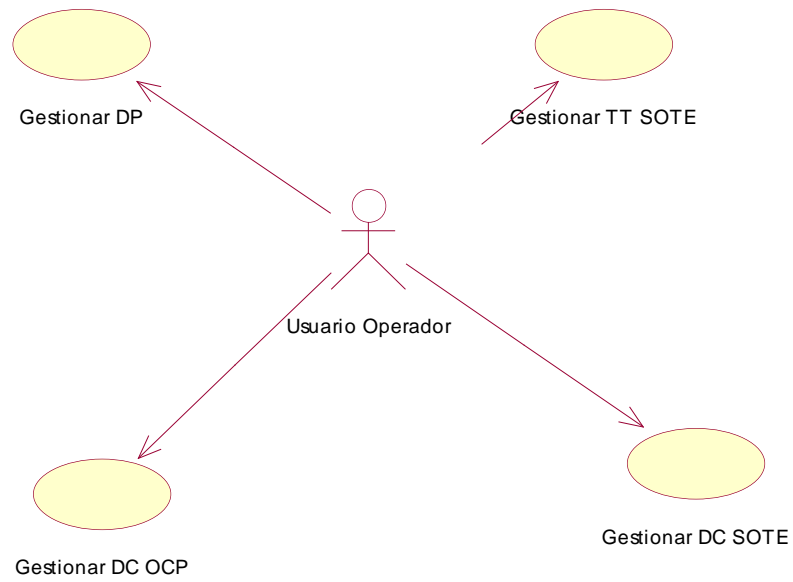
2.3.4 Ajuste °Api.



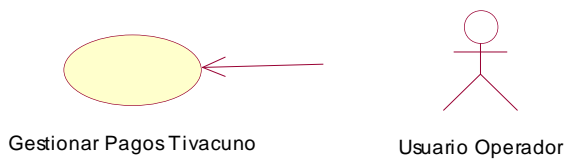
2.3.5 Ocp.



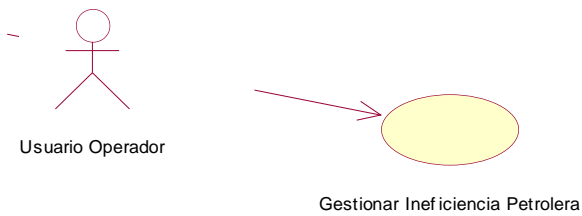
2.3.6 Facturación Sote



2.3.7 Pagos Tivacuno.



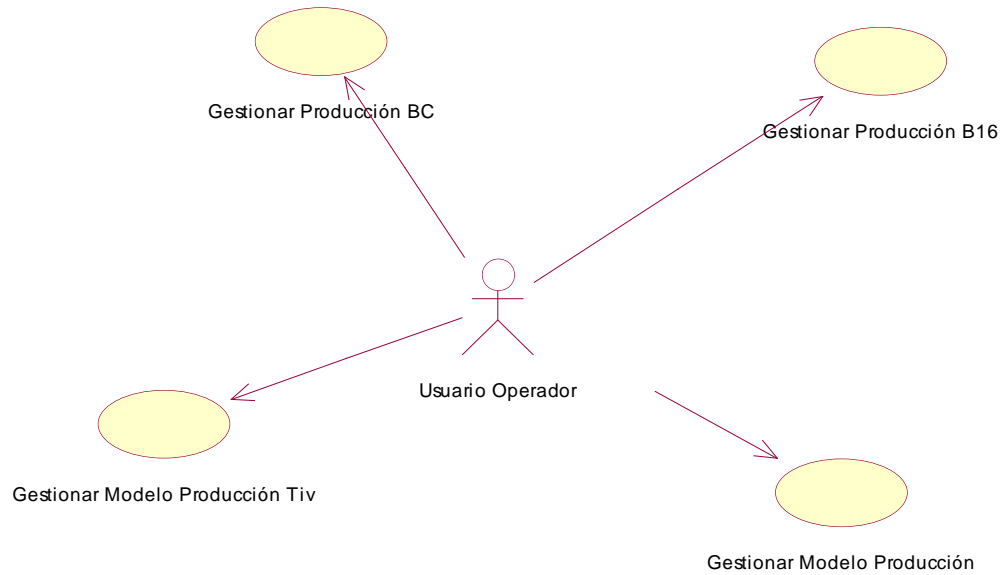
2.3.8 Ineficiencia Petrolera.



2.3.9 Balance de Entrega.



2.3.10 Modelo de Producción.



2.4 Especificación de casos de uso

2.4.1 Identificación de caso de uso y descripción.

Ver Anexo B

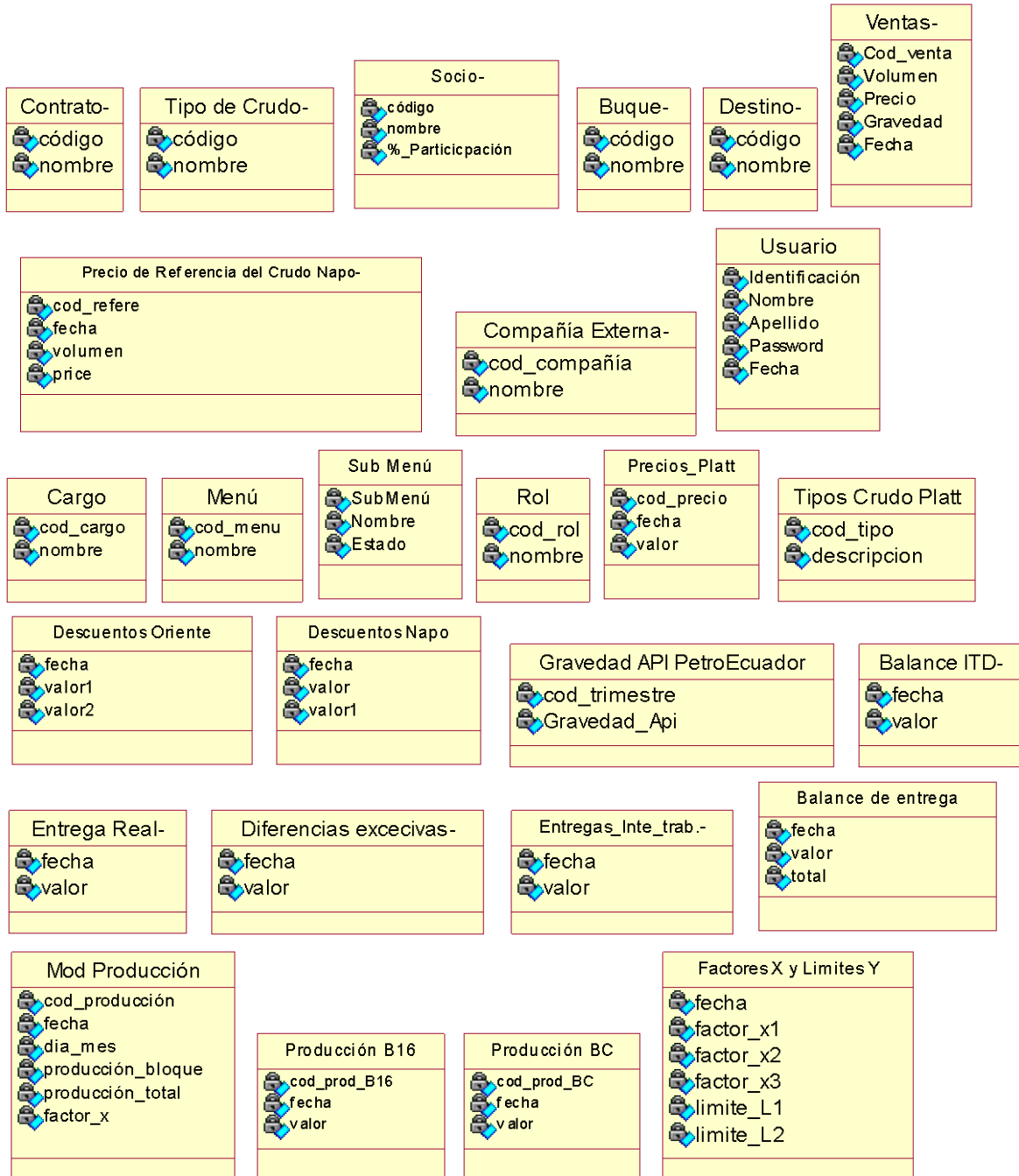
CAPÍTULO III

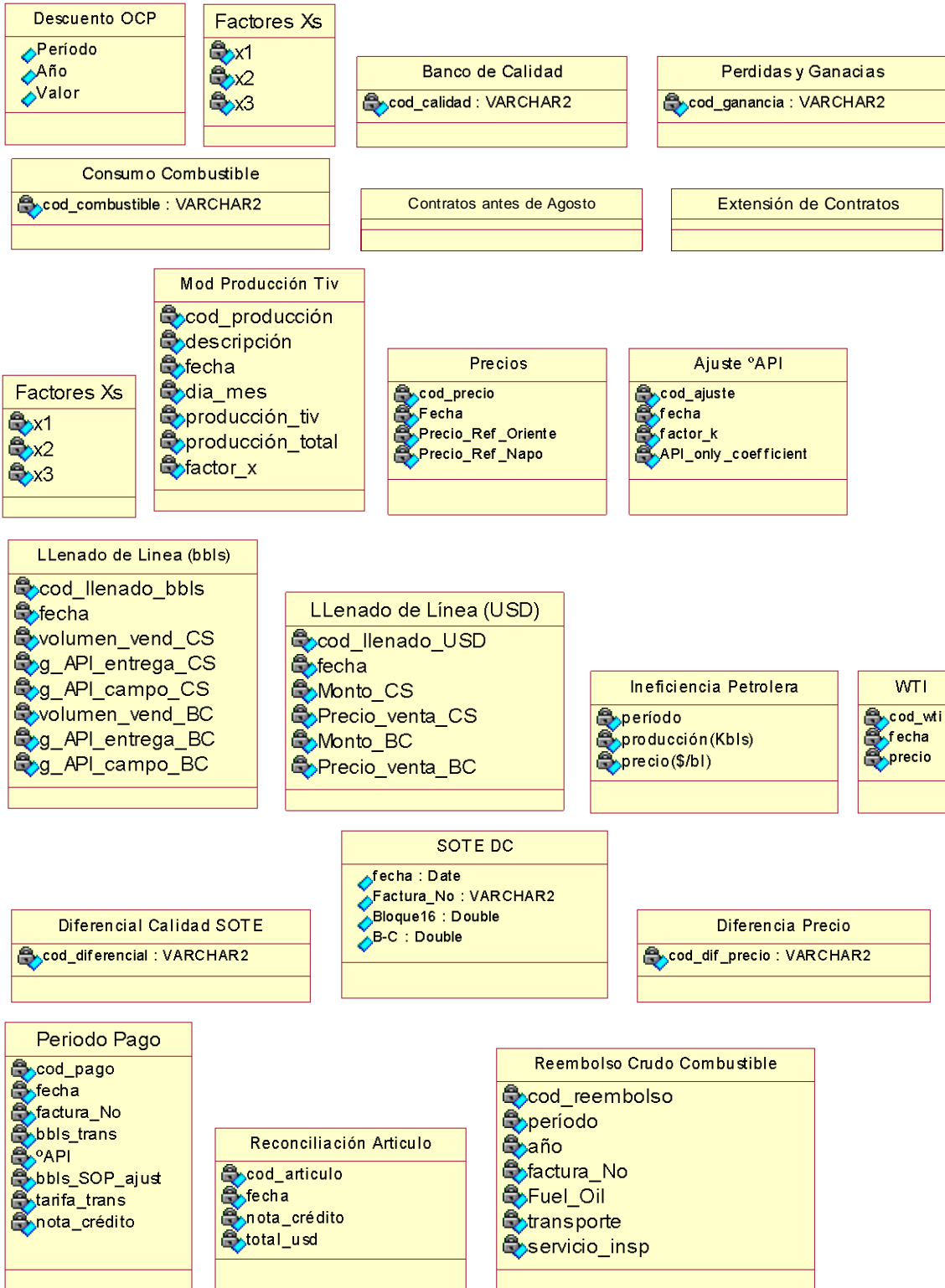
ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS.

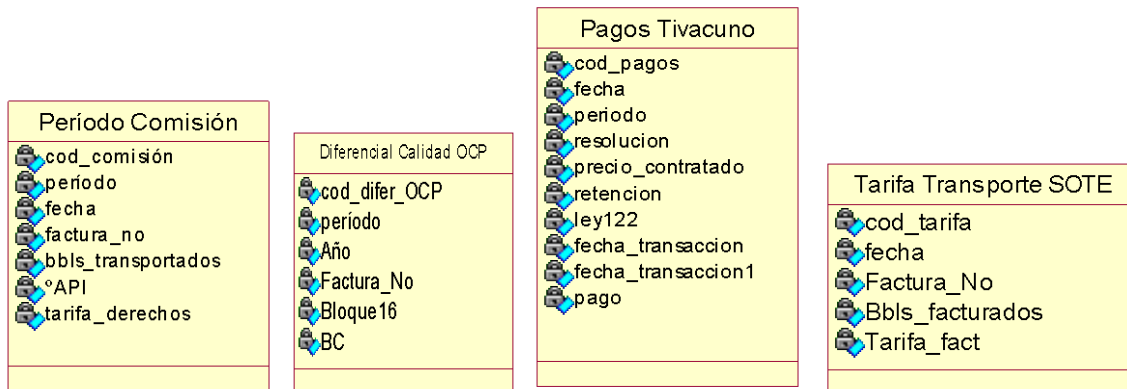
3.1 Análisis Orientado a Objetos.

3.1.1 Modelo de Objetos

3.1.2 Identificación de Clases y Objetos







3.1.3 Diccionario de Datos de las Clases Identificadas

Ajuste °API

Ajuste de los factores de cálculo del API (American Petroleum Institute).

Esta clase depende de la clase "Precios", la clase Ajuste °API toma el valor del precio de referencia del crudo oriente según corresponda a la fecha y mediante la siguiente formula realiza el cálculo correspondiente.

Factor K = $1.3/100 * \text{Precio de Referencia Oriente}$.

Api Only Coefficient = valor tomado de los datos del OCP.

Atributos Privados:

cod_ajuste : VARCHAR2

fecha : Date

factor_k : Double

API_only_coefficient : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Balance ITD

Es el balance acumulado desde el inicio del contrato por día.

Esta clase depende de las clases "Entregas_Inte_trab", "Entrega Real" y de la propia. La clase Balance ITD toma los datos de las clases Entregas_Inte_trab, Entrega Real y propia mediante la siguiente formula se hacen todos los cálculos y se genera en Balance ITD.

Balance ITD = ITD Anterior + ER Actual – IT Actual

Los cálculos del ITD se repiten para todos los usuarios.

Atributos Privados:

fecha : Date

valor : Double

Operaciones Privadas:

Insertar ()

Balance de entrega (BE)

El Balance de entrega son los datos que se ingresan de cada compañía para el cálculo de las entregas al OCP.

Esta clase es independiente, a partir de los datos que se ingresen en esta, se generara los datos de las siguientes clases: Balance ITD, Entrega Real, Diferencias Excesivas e Interés Trabajado.

Esta clase genera una suma total de los valores ingresados .

Atributos Privados:

fecha: Date

valor: Double

total: Double

Operaciones Privadas:

Insertar ():

Banco de Calidad

Banco de Calidad del OCP que tiene los datos de producción de las empresas petroleras externas en barriles en un período determinado.

Derivado desde Descuento OCP

Con los datos ingresados en esta clase se realizarán cálculos que determinan el total de cada una de las empresas externas.

Atributos Privados:

cod_calidad : VARCHAR2

Buque

Embarcación en el cual se carga el crudo para ser exportado a otros países.

Atributos Privados:

código: VARCHAR2
nombre: VARCHAR2

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Cargo

Actividad que realiza un usuario en la compañía

Atributos Privados:

cod_cargo : VARCHAR2
nombre : VARCHAR2

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Compañía Externa.

Empresa ajena al consorcio que explota un tipo de crudo y que sirve como precio de referencia.

Atributos Privados:

cod_compañía : VARCHAR2
nombre : VARCHAR2

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Consumo Combustible

Consumo de Combustible que lleva los datos de las empresas petroleras externas del consumo de combustible por un período determinado

Derivado desde Descuento OCP

Con los datos ingresados en esta clase se realizarán cálculos totales de cada una de las empresas externas.

Atributos Privados:

cod_combustible :

Contrato

Son las áreas de influencia en donde el consorcio explota el petróleo

Atributos Privados:

código : VARCHAR2

nombre : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Contratos antes de Agosto

Contratos antes de Agosto que lleva los datos de los factores de cálculo del WTI.

Derivado desde Factores Xs

Los datos de esta clase van ser constantes, utilizando una regla de comparación se extraerán los valores de X1, X2 o X3 dependiendo de lo que se necesite.

Descuento Napo y Oriente

Descuento del crudo Napo y Oriente relacionado con WTI reportado por el Platt's Latin American Wire Diariamente.

Los datos ingresados en esta clase son día a día, una vez ingresados los datos se calcula el promedio por mes y el promedio total.

Atributos Privados:

cod_desc : VARCHAR2

fecha : Date

D-L : Double

D-H : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Descuento OCP

Clase padre de los descuentos del OCP.

Esta clase heredará todos los atributos y operaciones de las clases hijas, Banco de Calidad, Consumo de Combustible, Pérdidas y Ganancias.

Public Attributes:

Período : VARCHAR2

Año : Date

Valor : Double

Public Operations:

Insertar () :

Modificar () :

Eliminar () :

Destino

Lugar o ubicación a donde se exporta el crudo.

Atributos Privados:

código : VARCHAR2

nombre : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Diferencia Precio

Diferencia de Precio 1999 clase hija de SOTE DC

La clase Diferencia Precio hereda todas las operaciones y atributos de la clase SOTE DC.

En esta clase se ingresarán los datos de las facturas de 1999 de los contratos Bloque 16 y Bogi Capirón. Con estos se calculará los totales que sean necesarios y mediante las siguientes formulas.

$DP = \text{Bloque 16} + \text{Bogi Capirón}$

* total por cada factura ingresada.

Atributos Privados:

cod_dif_precio : VARCHAR2

Diferencial Calidad OCP

Diferencial de calidad desde el inicio del OCP.

Esta clase constará de datos de las facturas del Diferencial de Calidad desde el inicio del OCP tanto del Bloque16 como de Bogi Capirón, con estos se calculara el total a pagar con la siguiente formula

* $DCOCP = \text{Bloque 16} + \text{Bogi Capirón}$.

* total por cada factura ingresada.

Atributos Privados:

cod_difer_OCP : VARCHAR2

período : VARCHAR2

Año : VARCHAR2

Factura_No : VARCHAR2

Bloque16 : Double

BC : Double

Operaciones Públicas:

Insertar () :

Modificar () :

Eliminar () :

Calcular () :

Diferencial Calidad SOTE

Diferencial de Calidad SOTE clase hija de SOTE DC

La clase Diferencial Calidad SOTE hereda todas las operaciones y atributos de la clase SOTE DC.

En esta clase se ingresarán los datos de las facturas desde 1999 tanto del contrato Bloque 16 como del Bogi Capirón, con estos se calculará el total con la siguiente formula.

* $DCSOTE = \text{Bloque 16} + \text{Bogi Capirón}$.

* total por cada factura ingresada.

Atributos Privados:

cod_diferencial : VARCHAR2

Diferencias excesivas(DE)

Son las diferencias excesivas entre la entrega real y la entrega por interés trabajado.

Esta clase depende de la clase Balance de Entrega, los datos ingresados en el Balance de Entrega generan Entrega por Interés Trabajado y Entrega Real, a partir de estas dos clases se generará los datos de la clase Diferencias Excesivas mediante la siguiente fórmula.

$$DE=ER-IT$$

Esta fórmula se aplica para el calculo de cada uno de los socios.

Atributos Privados:

fecha : Date

valor : Double

Operaciones Privadas:

Insertar () :

Entrega Real (ER)

Es la entrega real que hace cada uno de los socios por día.

Esta clase depende de la clase Balance de Entrega, los datos ingresados en esta última servirán para generar los de la clase Entrega Real mediante la siguiente fórmula.

$$ER=BE \text{ Repsol} * PP/80\%$$

La fórmula se repetirá para cada uno de los socios.

Atributos Privados:

fecha : Date

valor : Double

Operaciones Privadas:

Insertar () :

Entregas_Inte_trab. (IT)

Son las entregas de cada uno de los socios por interés trabajado por día.

Esta clase depende de la clase Balance de Entrega, los datos ingresados en esta última servirán para generar los de la clase Entregas_Inte_trab mediante la siguiente fórmula.

$$IT=-\text{Total BE} * PP$$

Fórmula que se aplica por cada socio.

Atributos Privados:

fecha : Date
valor : Double

Operaciones Privadas:

Insertar () :

Extensión de Contratos

Extensión de Contratos que lleva los datos de los factores X de cálculo del WTI.

Derivado desde Factores Xs.

Los datos de esta clase son constantes y no cambiarán en el tiempo, utilizando una regla de comparación se extraerán los datos de x1, x2 ó x3 dependiendo de lo que se requiera.

Factores X y Límites Y

Clase de los factores y límites que se utilizan para calcular las participaciones.

Estos datos serán ingresados según la fecha que se establezca pero no cambiarán en el tiempo, por esta razón cada vez que se cree un nuevo dato se agregarán automáticamente a la nueva fecha.

Atributos Privados:

fecha : Date
factor_x1 : Double
factor_x2 : Double
factor_x3 : Double
limite_L1 : Double
limite_L2 : Double

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Factores Xs

Clase padre de los Factores Xs.

Esta clase heredará todos sus atributos y operaciones a las clases: Contratos antes de Agosto, Extensión de Contratos.

Atributos Públicos:

WTI (US/DL) : NUMBER

x1 : Double

x2 : Double

x3 : Double

Operaciones Públicas:

Consultar () :

Gravedad API PetroEcuador

API American Petroleum Institute de acuerdo a PetroEcuador.

Atributos Privados:

cod_trimestre : VARCHAR2

Gravedad_Api : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Ineficiencia Petrolera

Datos de la ineficiencia petrolera desde 1972.

En esta clase se ingresarán los datos de la producción en miles de barriles (Kbbls) y el precio por barril (\$/bbl), se calculará la ineficiencia en miles de barriles y en a partir del año siguiente al año consultado con las formulas:

$I(\text{Kbbls}) = \text{Ineficiencia año siguiente al año base} - \text{Ineficiencia del año base.}$

$\text{Ineficiencia}(\text{MUS } \$) = I(\text{Kbbls}) * \text{Precio (formula 3.2.10)}$

Se calculará a partir del año base hasta el ultimo año que exista.

Atributos Privados:

período : Date

producción(Kbbls) : Double

precio(\$/bbl) : Double

Operaciones Publicas:

Insertar () :

Modificar () :

Calcular () :

Llenado de Línea (bbls)

Llenado de línea al OCP en barriles, que consta del volumen vendido de cada uno de los contratos así como de la entrega de la gravedad API. Se calcula el monto total vendido de cada contrato.

Atributos Privados:

cod_llenado_bbls : String
fecha : Date
volumen_vend_CS : Double
g_API_entrega_CS : Double
g_API_campo_CS : Double
volumen_vend_BC : Double
g_API_entrega_BC : Double
g_API_campo_BC : Double

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Llenado de Línea (USD)

Llenado de línea al OCP en dólares, que consta del monto de cada uno de los contratos y del precio de venta, esta clase depende de la clase "Llenado de Línea (bbls)".

La clase Llenado de Línea (USD) tomara los datos de la clase Llenado de Línea (bbls) y mediante la siguiente formula realizará los cálculos de acuerdo a la fecha de ingreso.

* Monto= Precio de Venta * Volumen Vendido

* Cálculo para cada contrato.

Atributos Privados:

cod_llenado_USD :
fecha : Date
Monto_CS : Double
Precio_venta_CS : Double
Monto_BC : Double
Precio_venta_BC : Double

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Menú

Son las funciones principales que el sistema presentará en la pantalla principal.

Atributos Privados:

cod_menu : VARCHAR2

nombre : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Mod Producción

Modelo de Producción de los contratos BLOQUE 16 y Bogi-Capirón.

Esta clase tomará los datos de las clases Producción B16 y Producción BC independientemente para realizar los cálculos de: la producción por bloque, las participaciones q, los factores x, los límites L, el factor x calculado, las Participaciones del Estado y las Participaciones Contratista, de cada uno de los contratos mediante el uso de las siguientes formulas.

Producción Bloque = Producción total / No. Días

Factor x1, Factor x2, Factor x3, Limite L1, Limite L2, para este caso son constantes.

Producción q1 = Si la producción por bloque es menor que el Limite L1, la producción q1 es igual a la producción por bloque, en caso contrario, la producción q1 es igual al limite L1 menos 1.

Producción q2 = Si la producción por bloque es mayor o igual al limite L1 y la producción por bloque es menor o igual al limite L2, la producción q2 es igual a la producción por bloque menos la producción q1, caso contrario, si la producción por bloque es mayor que el limite L2, la producción q2 es igual al limite L2 menos el limite L1 mas 1, en caso contrario la producción q2 será 0.

Producción q3 = Si la producción por bloque es mayor que el limite L2, la producción q3 es igual a la producción por bloque menos la producción q1 menos la producción q2, en caso contrario la producción q3 será 0.

Factor x Calculado = Si la producción total es igual a cero, el factor x calculado será igual a cero, caso contrario, el factor x será igual a la sumatoria del factor x1 por la producción q1, el

factor x2 por la producción q2 y el factor x3 por la producción q3 dividido para el total de la producción q.

Participación Contratista = la producción total por el factor x calculado.

Participación del Estado = la producción total menos la participación de la contratista.

Atributos Privados:

cod_producción : VARCHAR2

descripción : VARCHAR2

fecha : Date

producción : Double

factor_x : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Mod Producción Tiv

Modelo de Producción del contrato Tivacuno.

Esta clase se aplicara una regla de comparación para extraer los datos de: la producción por bloque, las participaciones q, el precio wti, los factores x, los límites L, el factor x calculado, las Participaciones del Estado y las Participaciones Contratista, mediante el uso de las siguientes formulas.

Producción Tivacuno= producción total dividido para el numero de días.

WTI Nymex= este dato se traerá desde la clase WTI mediante la fecha.

Factor x1= Si el precio WTI es igual a cero, entonces el factor x1 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.75%, entonces el factor x1 es igual a x1 correspondiente de la clase Contratos antes de Agosto menos 0.01.

Factor x2= Si el precio WTI es igual a cero, entonces el factor x2 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.70%, entonces el factor x1 es igual a x2 correspondiente de las clases Contratos antes de Agosto menos 0.01.

Factor x3= Si el precio WTI es igual a cero, entonces el factor x3 será igual a cero, caso contrario, si el precio WTI es menor o igual a 30.61%, entonces el factor x3 es igual a x3 correspondiente de la clase Contratos antes de Agosto menos 0.01.

Limite L1 y Limite L2= son valores constantes.

Producción q1= si la producción tivacuno es menor que el limite L1, entonces la producción q1 es igual a la producción tivacuno, caso contrario, la producción q1 será igual al limite L1 menos uno.

Producción q2= si la producción tivacuno es mayor o igual que el limite L1 y la producción tivacuno es menor o igual que el limite L2, la producción q2 es igual a la producción tivacuno menos el limite L1, caso contrario, si la producción tivacuno es mayor al limite L2, la producción q2 es igual al limite L1 mas uno, caso contrario, la producción q2 es igual a 0.

Producción q3 = Si la producción tivacuno es mayor o igual que el limite L2, la producción q3 es igual la producción tivacuno menos la producción q1 menos la producción q2, en caso contrario la producción q3 será 0.

Factor x Calculado = el factor x es igual a la sumatoria del factor x1 por la producción q1, el factor x2 por la producción q2 y el factor x3 por la producción q3 dividido para el total de la producción q.

Participación Contratista = la producción total por el factor x calculado.

Participación del Estado= la producción total menos la participación de la contratista.

Atributos Privados:

cod_producción : VARCHAR2

descripción : VARCHAR2

fecha : Date

dia_mes : Date

producción_tiv : Double

producción_total : Double

factor_x : Double

Operaciones Públicas:

Insertar () :

Modificar () :

Pérdidas y Ganancias

Pérdidas y Ganancias que llevan los datos de las empresas petroleras externas de las Pérdidas y Ganancias en la Línea en un período determinado.

Derivado desde Descuento OCP

Sé calculá una suma total de cada una de las empresas.

Atributos Privados:

cod_ganancia :

Periodo Pago

Período de pago de las facturas por transporte de crudo.

Los datos de esta clase son las facturas por período de pago, que constará de: los barriles transportados, barriles ajustados, la tarifa de transporte, las notas de crédito y los cálculos para el total de la tarifa de transporte que se lo realizará con la siguiente formula:

Total Tarifa Transporte = Barriles SOP Ajustados * Tarifa Transporte * numero de días.

Se calcula una sumatoria total de los barriles transportados y el total de la tarifa transportada.

Atributos Privados:

cod_pago : VARCHAR2

fecha : Date

factura_No : VARCHAR2

bbls_trans : Double

°API : Double

bbls_SOP_ajust : Double

tarifa_trans : Double

nota_crédito : VARCHAR2

Operaciones Públicas:

Insertar () :

Modificar () :

Eliminar () :

Calcular () :

Período Comisión

Período de Comisión

Los datos de esta clase son las facturas por período de comisión, que constará de: los barriles transportados, el api, el número de factura, los barriles ajustados y la tarifa de transporte los cálculos para los barriles ajustados y el total de se las realiza con las siguientes formulas:

$$BA=BT*(1.4239072-(0.0382353*API)+(0.0008158*API^2))$$
$$\text{Tarifa Transporte} = 1.6989 / (1 - \text{Royalty Rate})$$

$$\text{Total} = BA * \text{Tarifa Transporte}$$

Atributos Privados:

cod_comisión : VARCHAR2
período : VARCHAR2
fecha : Date
factura_no : VARCHAR2
bbls_transportados : Double
°API : Double
tarifa_derechos : Double

Operaciones Públicas:

Insertar () :
Modificar () :
Eliminar () :
Calcular () :

Precios

Precios de referencia de los tipos de crudo que explota el consorcio.

Atributos Privados:

cod_precio : String
Fecha : Date
Precio_Ref_Oriente : Double
Precio_Ref_Napo : Double

Operaciones Privadas:

Insertar () :
Modificar () :
Eliminar () :

Precio de Referencia Crudo Napo

Precio de referencia que tiene el crudo de Napo por compañía y que hereda algunos atributos de la clase ventas, como el

destino, el transporte, el contrato. Consta de atributos propios como el volumen, precio y una asociación de compañía.

Derivado desde Ventas

Con los datos ingresados en esta clase se podrá calcular totales de ventas como se indica en la siguiente formula:

Total = Volumen * Precio

Atributos Privados:

cod_refere : VARCHAR2

fecha : Date

volumen : Double

price : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Precios_Platt

Precios de cada uno de los crudos que son clasificados según el Platt's.

Con los datos ingresados en esta clase se podrá calcular los promedios anuales o mensuales.

Atributos Privados:

cod_precio : String

fecha : Date

valor : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Modificar () :

Producción B16

Producción del crudo del contrato 'BLOQUE 16' por mes.

Atributos Privados:

cod-prod_B16 : String

fecha : Date

valor : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Producción BC

Producción de crudo del contrato 'Bogi-Capiron' por mes.

Atributos Privados:

cod_prod_BC : String

fecha : Date

valor : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Reconciliación Artículo

Reconciliación de acuerdo a artículo 7.04 del ISTA.

Atributos Privados:

cod_articulo: VARCHAR2

fecha : Date

nota_crédito : VARCHAR2

total_usd : Double

Operaciones Públicas:

Insertar () :

Modificar () :

Eliminar () :

Calcular () :

Reembolso Crudo Combustible

Atributos Privados:

cod_reembolso : VARCHAR2

período : VARCHAR2

año : VARCHAR2

factura_No : VARCHAR2

Fuel_Oil : Double

transporte : Double

servicio_insp : Double

Operaciones Públicas:

Insertar () :
Modificar () :
Eliminar () :

Rol

Son los roles o los permisos que los usuarios tienen sobre la aplicación (Administrador, Operador, etc)

Atributos Privados:

cod_rol : VARCHAR2
nombre : VARCHAR2

Operaciones Privadas:

Insertar () :
Modificar () :

SOTE DC

Clase padre SOTE DC que contiene los datos de las facturas de la Diferencia de Precio 1999 y el Diferencial de Calidad SOTE

La clase SOTE DC derivara todos sus atributos y operaciones a las clases Diferencia de Precio 1999 y el Diferencial de Calidad SOTE.

Atributos Privados:

fecha : Date
Factura_No : VARCHAR2
Bloque16 : Double
B-C : Double

Operaciones Públicas:

Insertar () :
Modificar () :
Eliminar () :
Calcular () :

Socio

Empresa o compañía que esta asociada y que opera en las áreas de influencia de todo el consorcio.

Derivado desde Entregas_Inte_trab.

Atributos Privados:

código : VARCHAR2
nombre : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Sub Menú

Son los sub menú u opciones que cada uno de los menús tendrá.

Atributos Privados:

SubMenú : VARCHAR2

Nombre : VARCHAR2

Estado : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Tarifa Transporte SOTE

Tarifa del Transporte de crudo por el SOTE las facturas por el envío de los barriles de crudo.

Esta clase constará de datos como los barriles facturados y la tarifa por transporte de crudo facturada, con estos se calculara los respectivos totales, mediante formulas pre-establecidas.

Atributos Privados:

cod_tarifa : VARCHAR2

fecha : Date

Factura_No : VARCHAR2

Bbls_facturados : Double

Tarifa_fact : Double

Operaciones Públicas:

Insertar () :

Modificar () :

Eliminar () :

Calcular () :

Tipo de Crudo

Tipo de crudo que pueden ser categorizados de varios tipos de acuerdo a sus características. Pueden existir un sin número de tipos de crudo, cada venta tendrá un tipo de crudo.

Atributos Privados:

código : VARCHAR2

nombre : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Tipos Crudo Platt

Son los tipos de crudo según el Platt's.

Atributos Privados:

cod_tipo : String

descripcion : String

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Transporte Crudo Oleoductos

Datos del transporte de crudo por oleoductos.

Atributos Privados:

período: Date

transporte_total: Double

transporte_dia: Double

SOTE: Double

OTA: Double

OCP-2: Double

Operaciones Públicas:

Insertar ():

Modificar ():

Usuario

Clase usuario que lleva todos los datos de los usuarios que pueden ingresar al sistema, en esta clase solo podrá hacer operaciones el usuario administrador.

Derivado desde Usuario Consultor, Usuario Administrador, Usuario Operador

Atributos Privados:

Identificación : VARCHAR2

Nombre : VARCHAR2

Apellido : VARCHAR2

Password : VARCHAR2

Fecha : VARCHAR2

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

Ventas

Son las ventas que tiene cada uno de los socios y el acumulado del consorcio, posee atributos como el destino a donde se envía el crudo, el buque, el tipo de crudo, la compañía que lo produce, el volumen exportado y el precio.

Con los datos ingresados en esta clase se podrá hacer cálculos de los totales vendidos de cada uno de los socios y el acumulado del consorcio.

Atributos Privados:

Cod_venta : VARCHAR2

Volumen : Double

Precio : Double

Gravedad : Double

Operaciones Privadas:

Insertar () :

Modificar () :

Eliminar () :

WTI

Precio del crudo según el WTI (West Texas Intermediate), desde 1983.

Atributos Privados:

cod_wti : String

fecha : Date

precio : Double

Operaciones Privadas:

Insertar () :

Eliminar () :

Modificar () :

3.1.4 Identificación de las asociaciones entre clases.

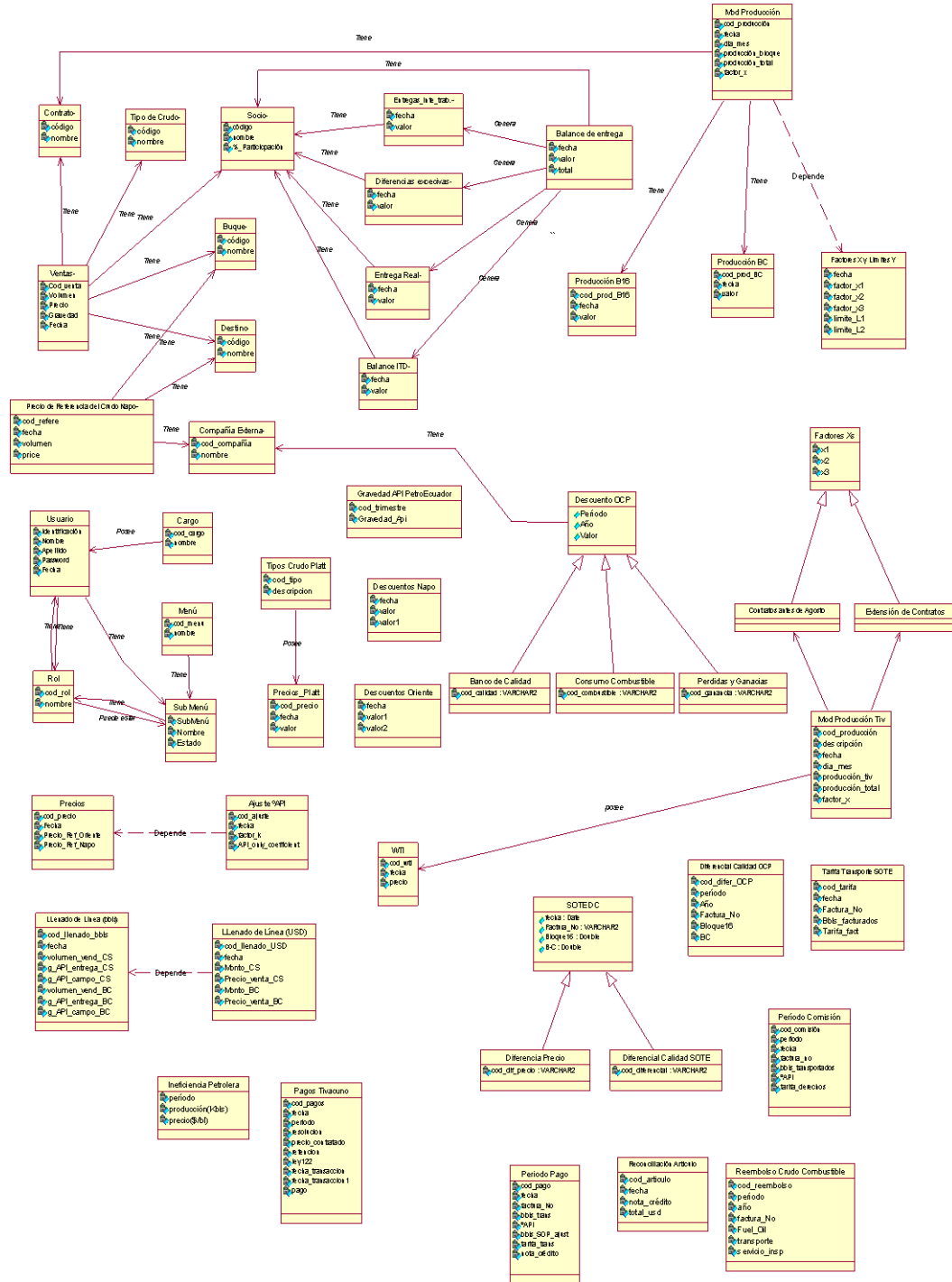
Locuciones Verbales

- ✓ Ventas incluye los contratos, los tipos de crudo, los socios, los buques y el destino.
- ✓ Napo Crude Oil Reference se deriva de Ventas.
- ✓ Balance de entrega proporciona las entregas por interés de trabajo, las diferencias excesivas, la entrega real y el balance ITD.
- ✓ Balance de entrega incluye los socios y sus porcentajes de participación.
- ✓ Entregas por interés trabajado incluye socio y porcentaje de participación.
- ✓ Diferencias excesivas incluye socio y porcentaje de participación.
- ✓ Entrega Real incluye socio y porcentaje de participación.
- ✓ Balance ITD incluye socio y porcentaje de participación.
- ✓ El usuario incluye cargos.
- ✓ El usuario incluye roles.
- ✓ El menú tiene varios submenús.
- ✓ El rol incluye submenús y menús.
- ✓ Los Tipos de crudo Platt incluye Precios Platt.
- ✓ Modelo de Producción incluye la Producción B16 y la Producción BC.
- ✓ Modelo de Producción depende de los Factores X y Limites Y.
- ✓ Descuentos OCP incluye compañía externa.
- ✓ Banco de Calidad se deriva de Descuento OCP.
- ✓ Consumo de Combustible se deriva de Descuento OCP.
- ✓ Pérdidas y Ganancias se deriva de Descuento OCP.
- ✓ Contratos antes de Agosto se deriva de Factores Xs.
- ✓ Extensión de Contratos se deriva de Factores Xs.
- ✓ Ajuste °API depende de Precios.
- ✓ Llenado de Línea (USD) depende de Llenado de Línea (bbls).
- ✓ Modelo de producción Tivacuno incluye Factores Xs.
- ✓ Modelo de producción Tivacuno Incluye WTI.
- ✓ Contratos antes de agosto se deriva de Factores Xs
- ✓ Extensión de Contratos se deriva de Factores Xs.

Locuciones Verbales implícitas.

- ✓ El consorcio esta formado por socios.
- ✓ Los socios tienen acciones.
- ✓ El sistema se encarga de la seguridad.
- ✓ El sistema se encarga del registro.
- ✓ El usuario maneja la aplicación.
- ✓ El usuario maneja los reportes.

3.1.5 Construcción del Modelo de Clases

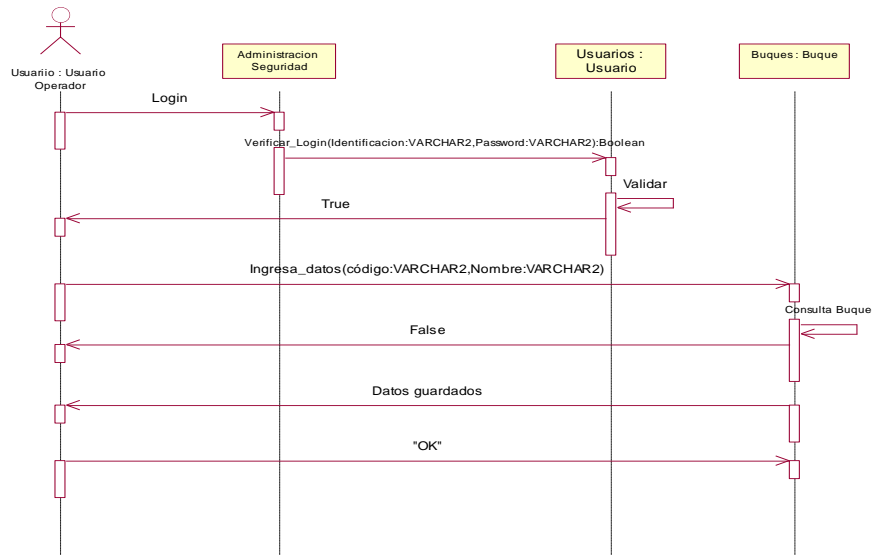


3.2.1 Modelo Dinámico

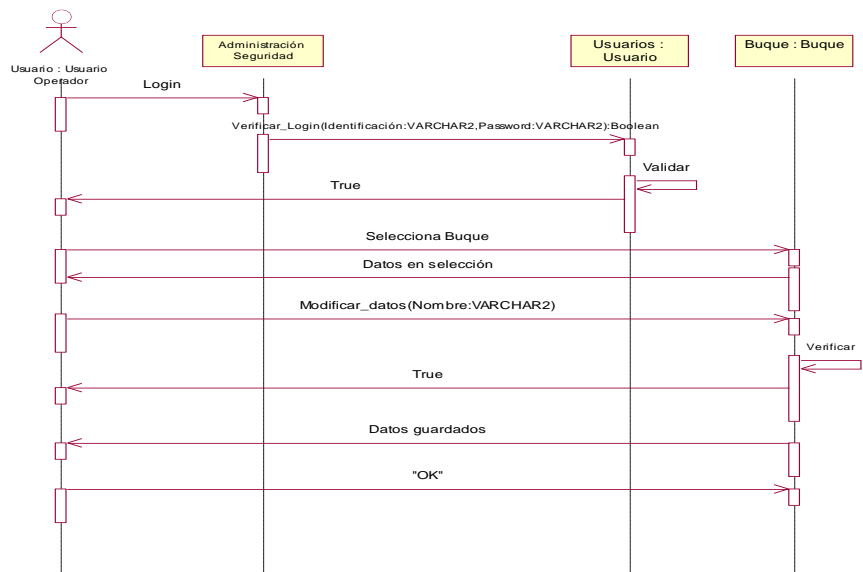
3.2.1.1 Construcción Diagramas de Secuencia.

3.2.1.1.1 Ventas

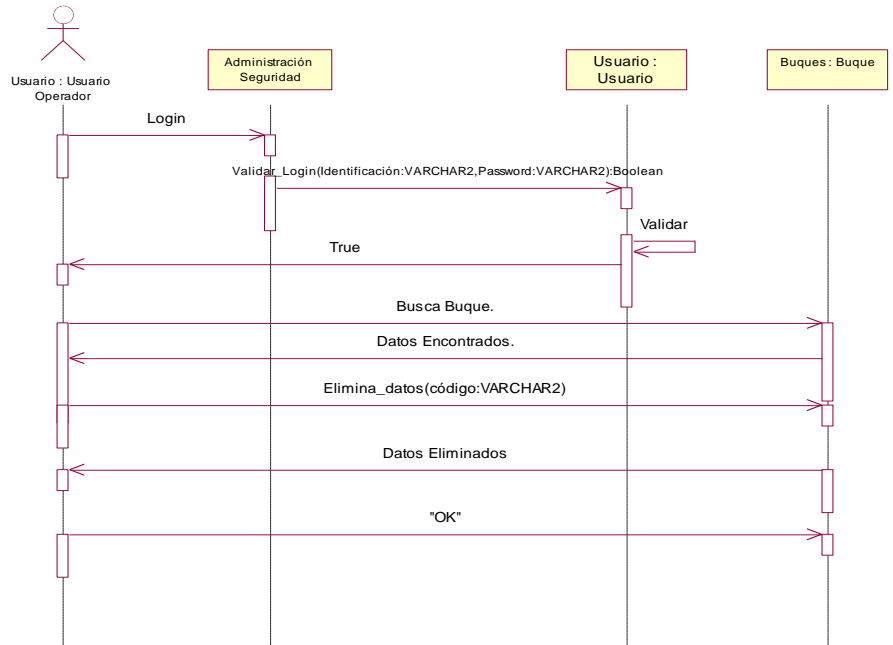
Buques Ingreso



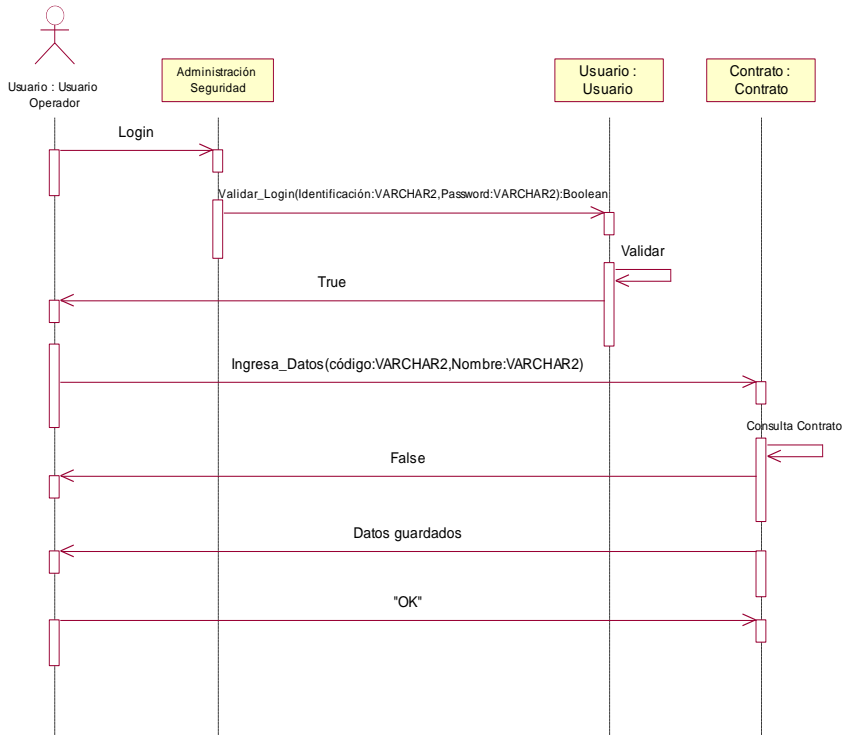
Modificación



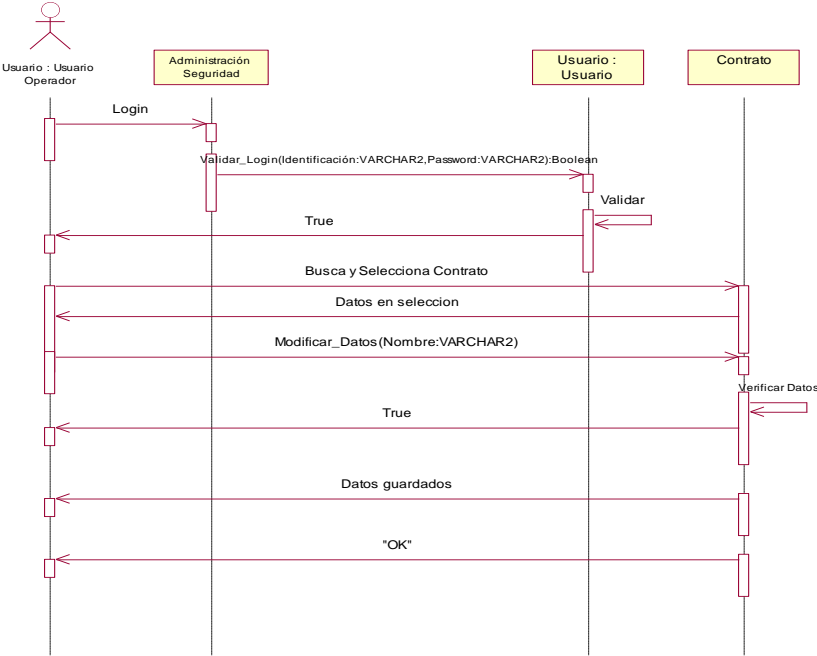
Eliminación



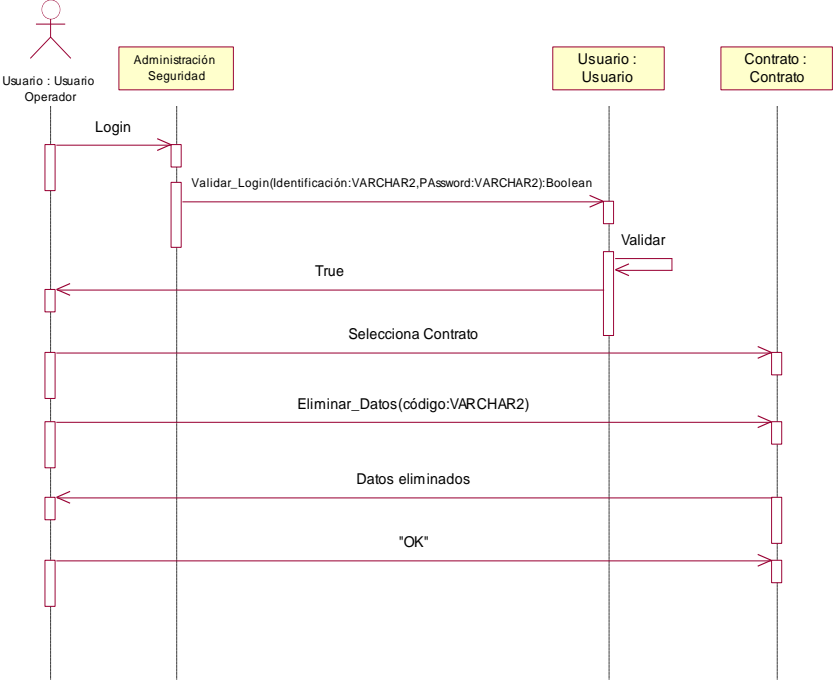
Contratos Ingreso



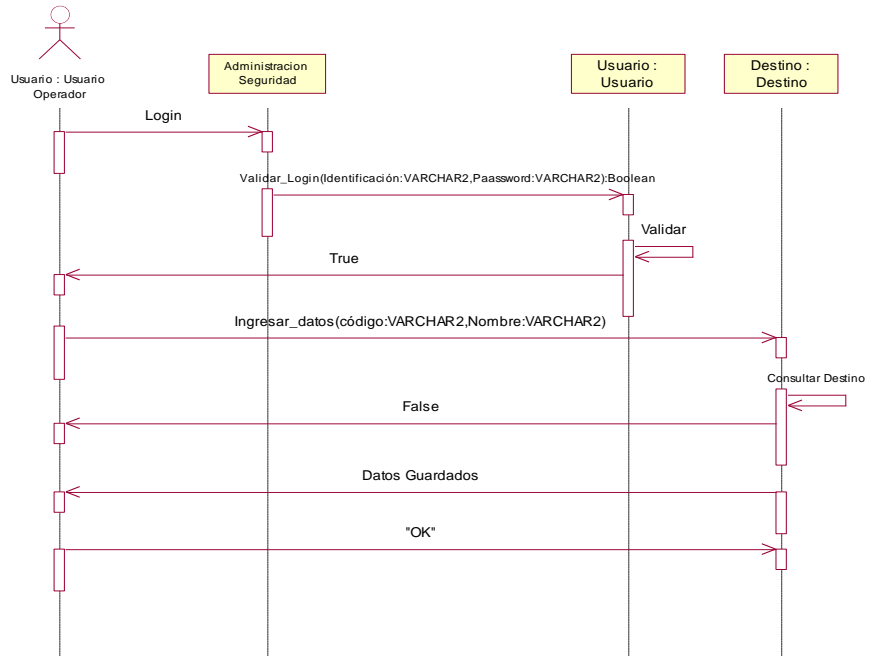
Modificación.



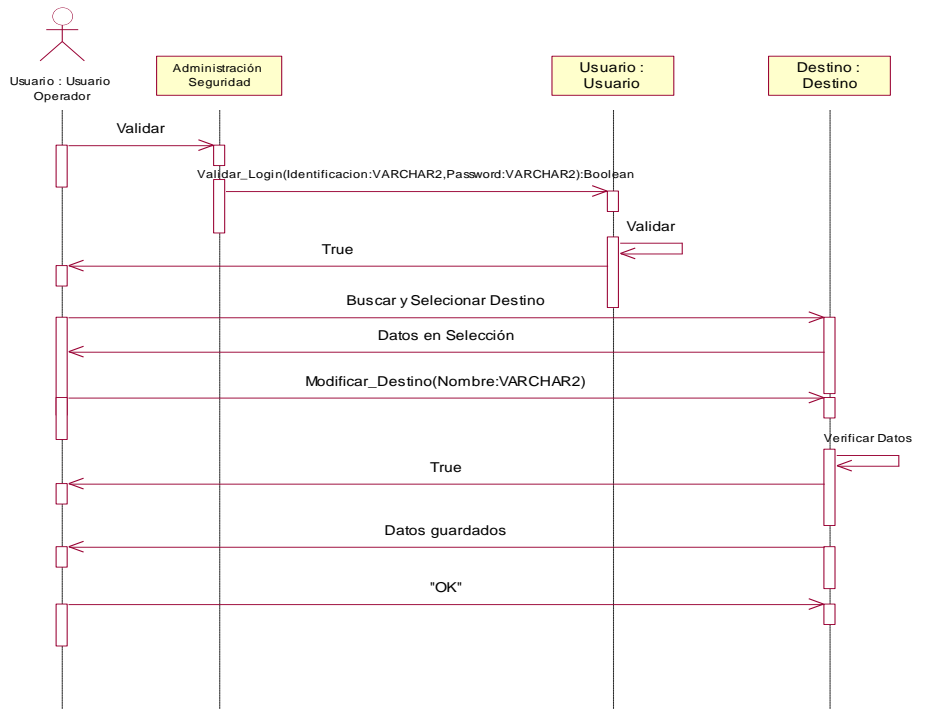
Eliminación



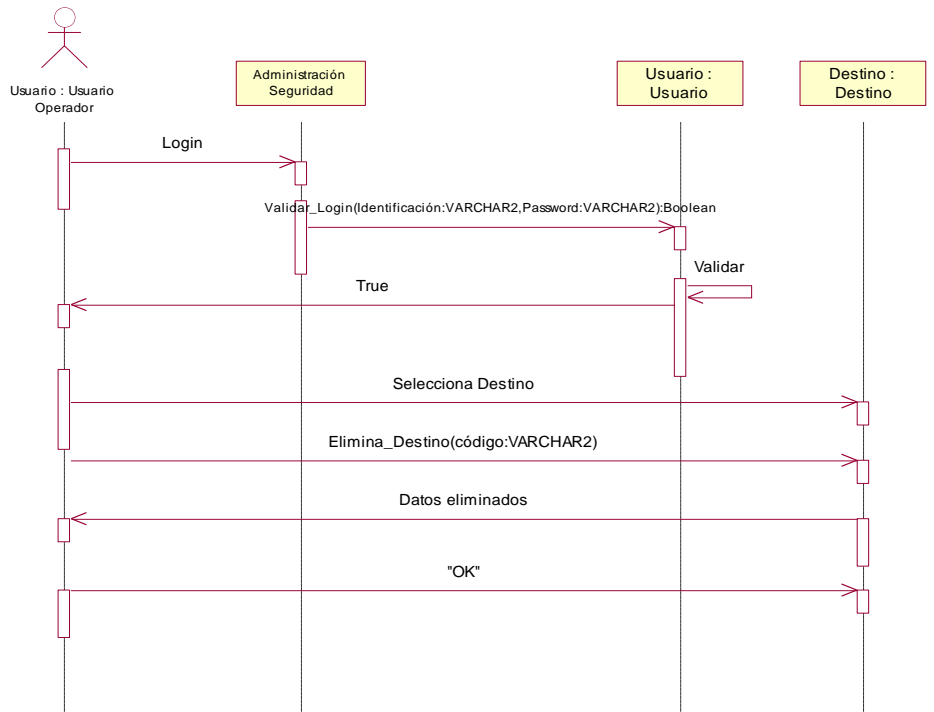
Destino Ingreso



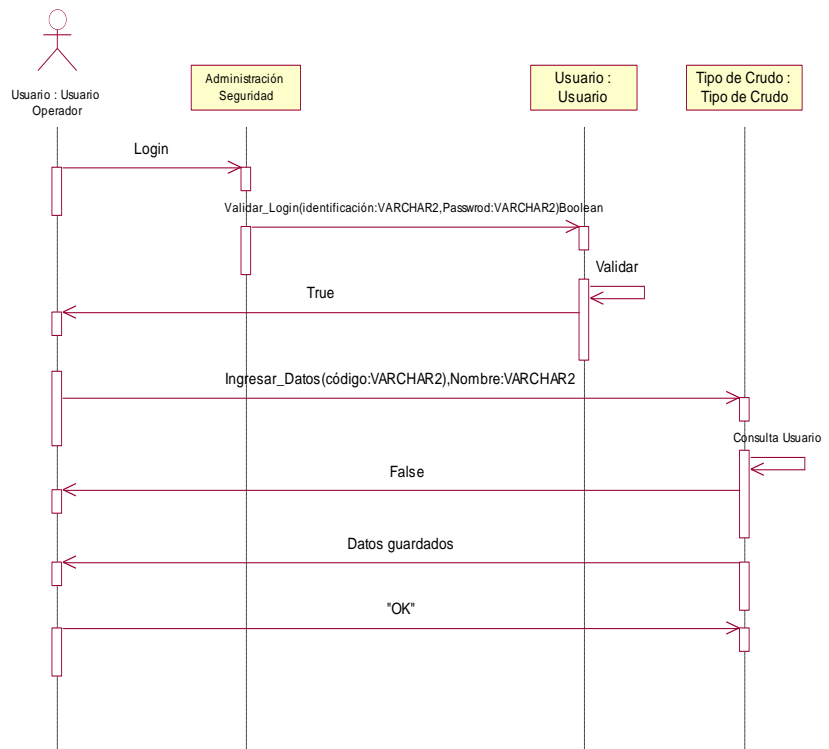
Modificación.



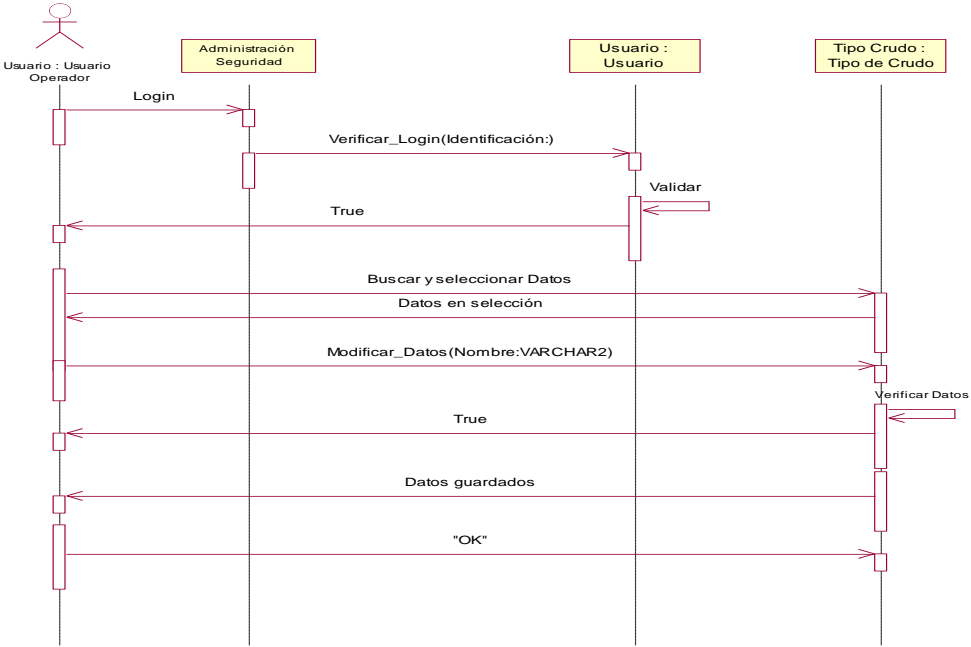
Eliminación



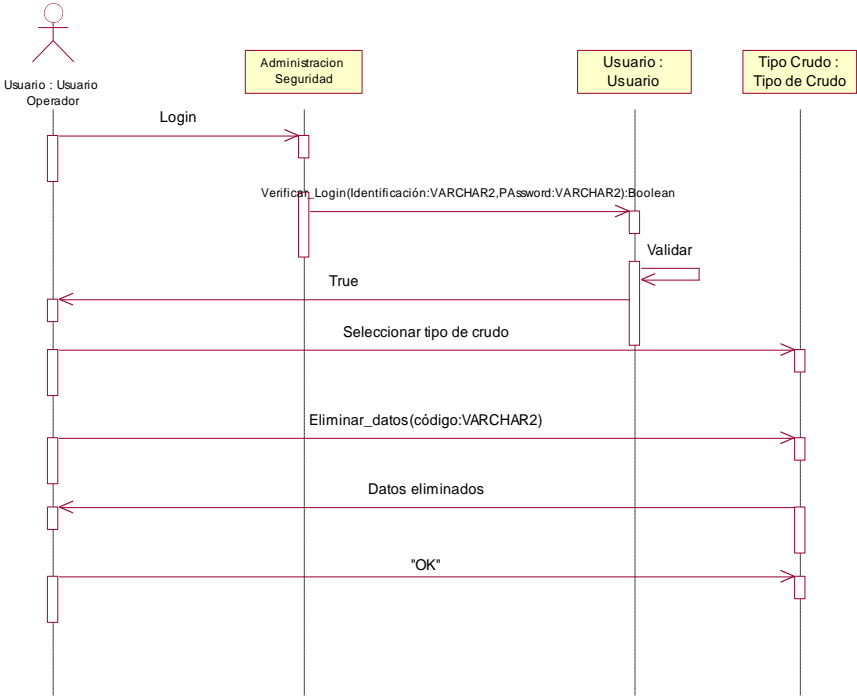
Tipo de Crudo Ingreso



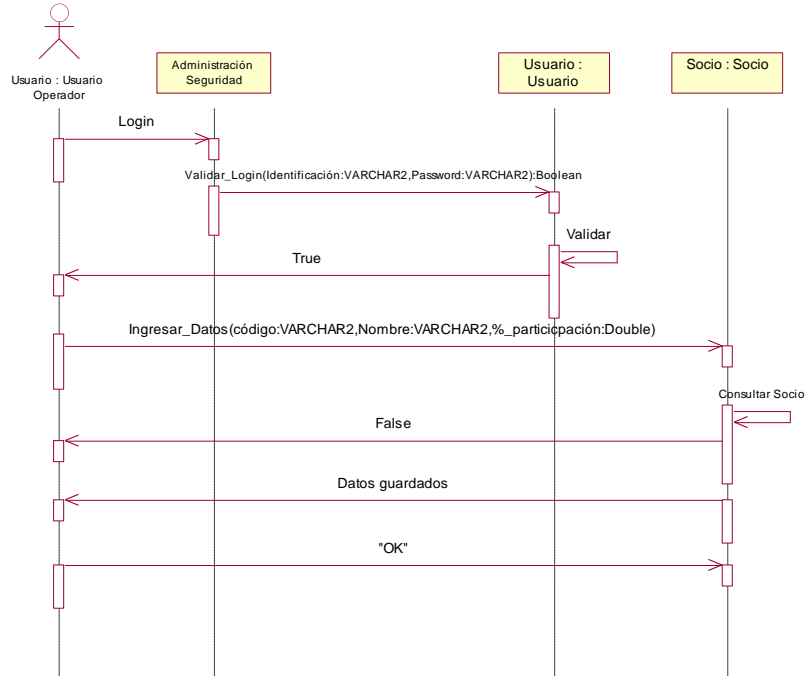
Modificación.



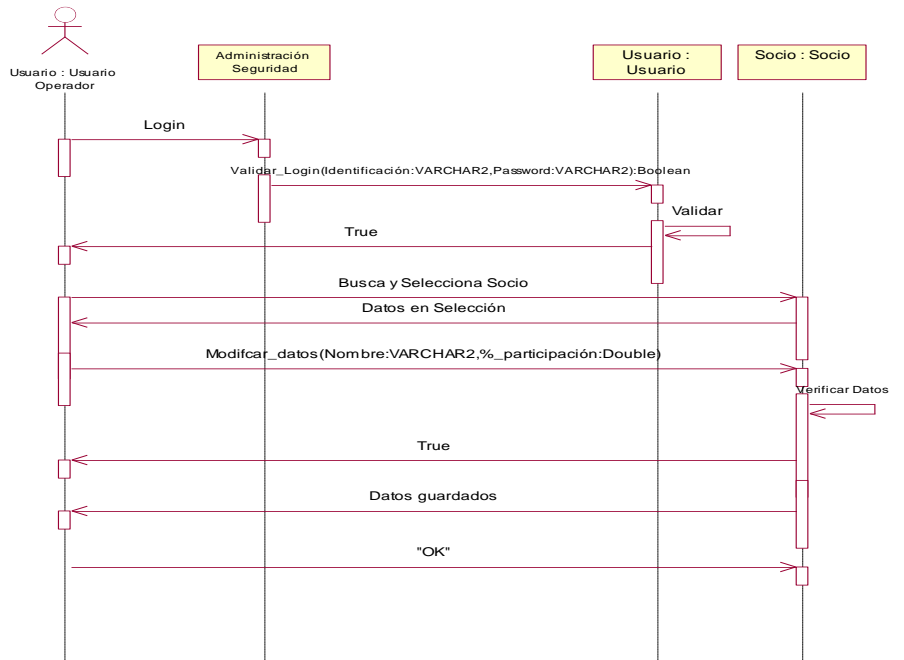
Eliminación



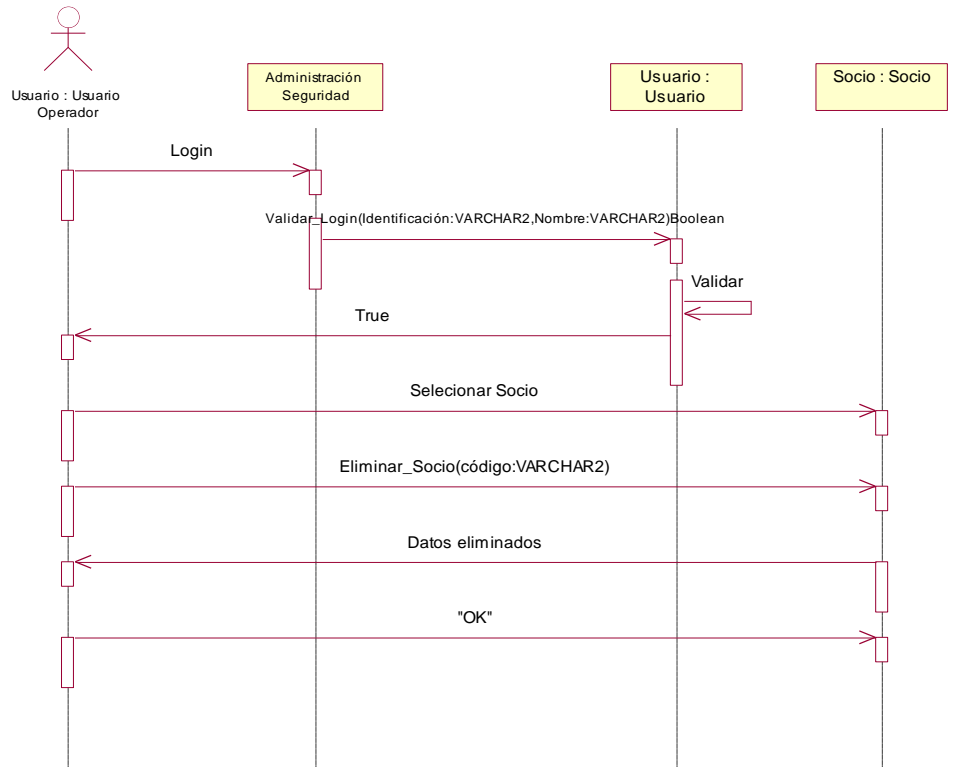
Socios Ingreso



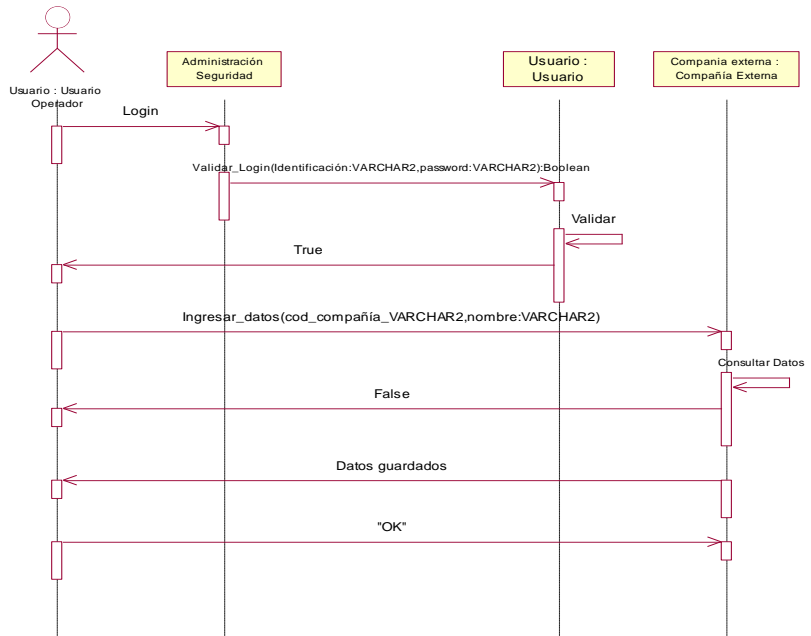
Modificación.



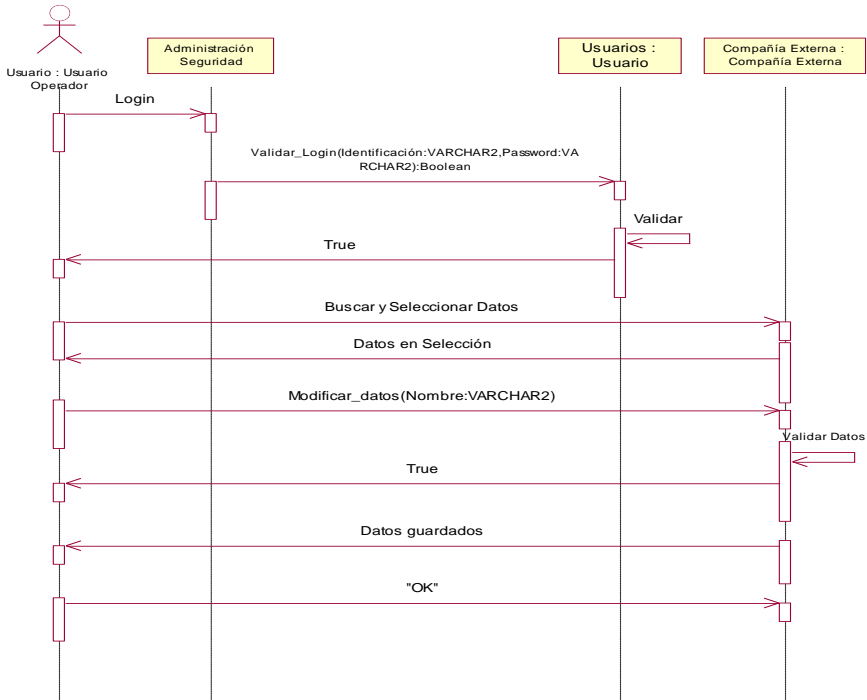
Eliminación



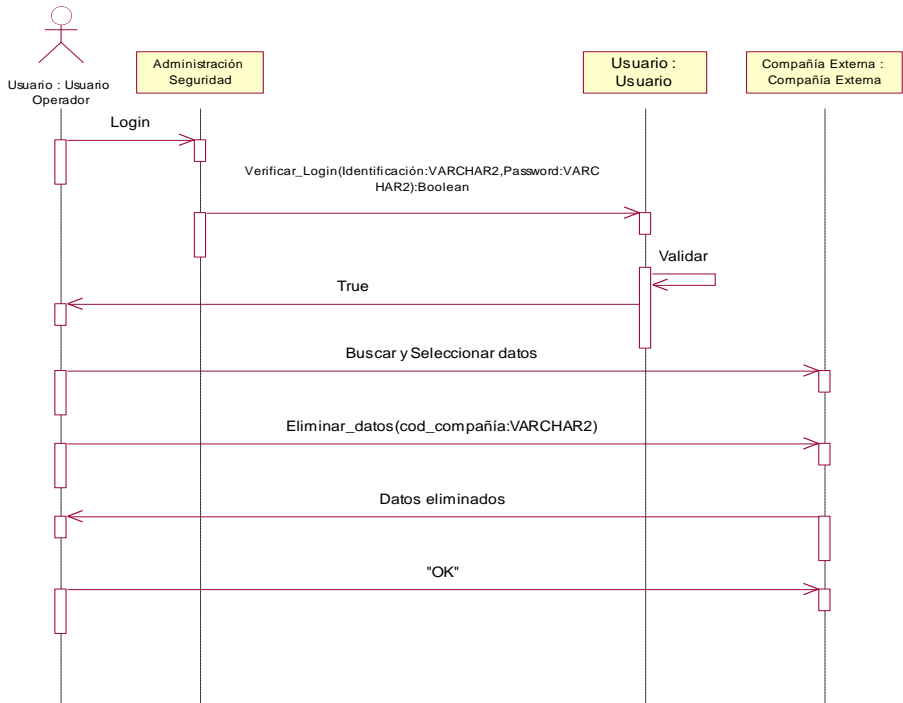
Compañías Ingreso



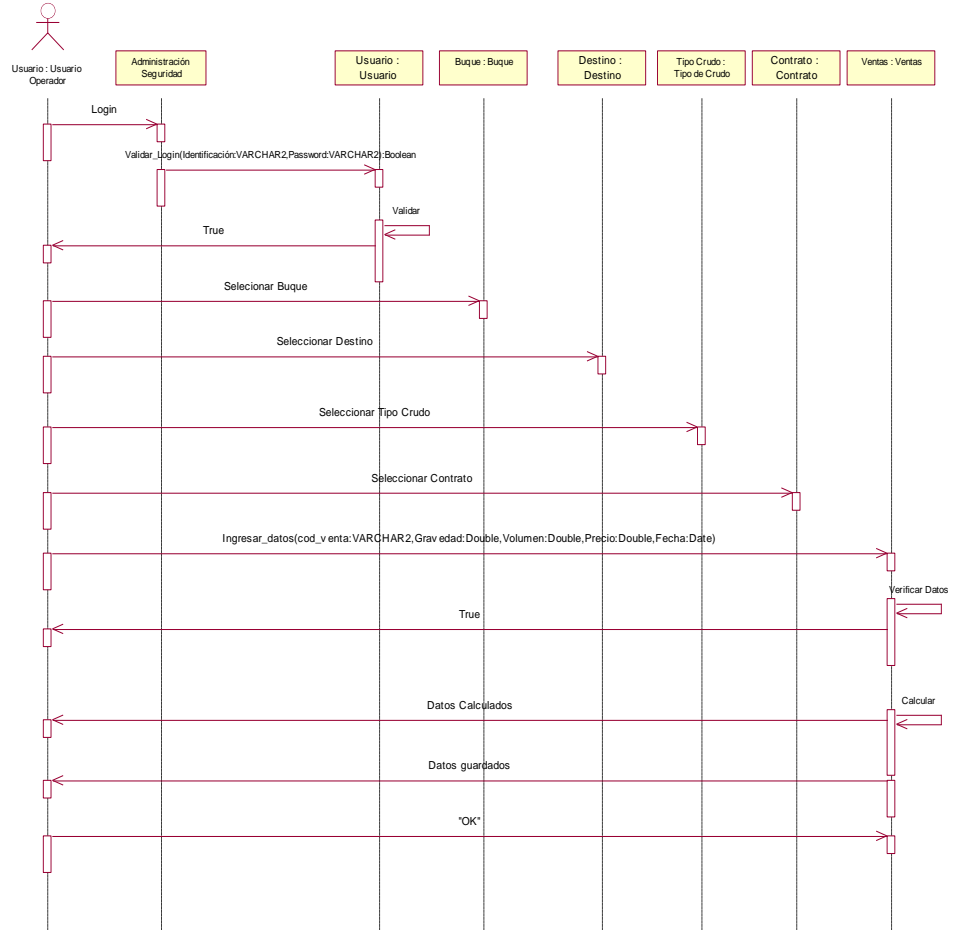
Modificación.



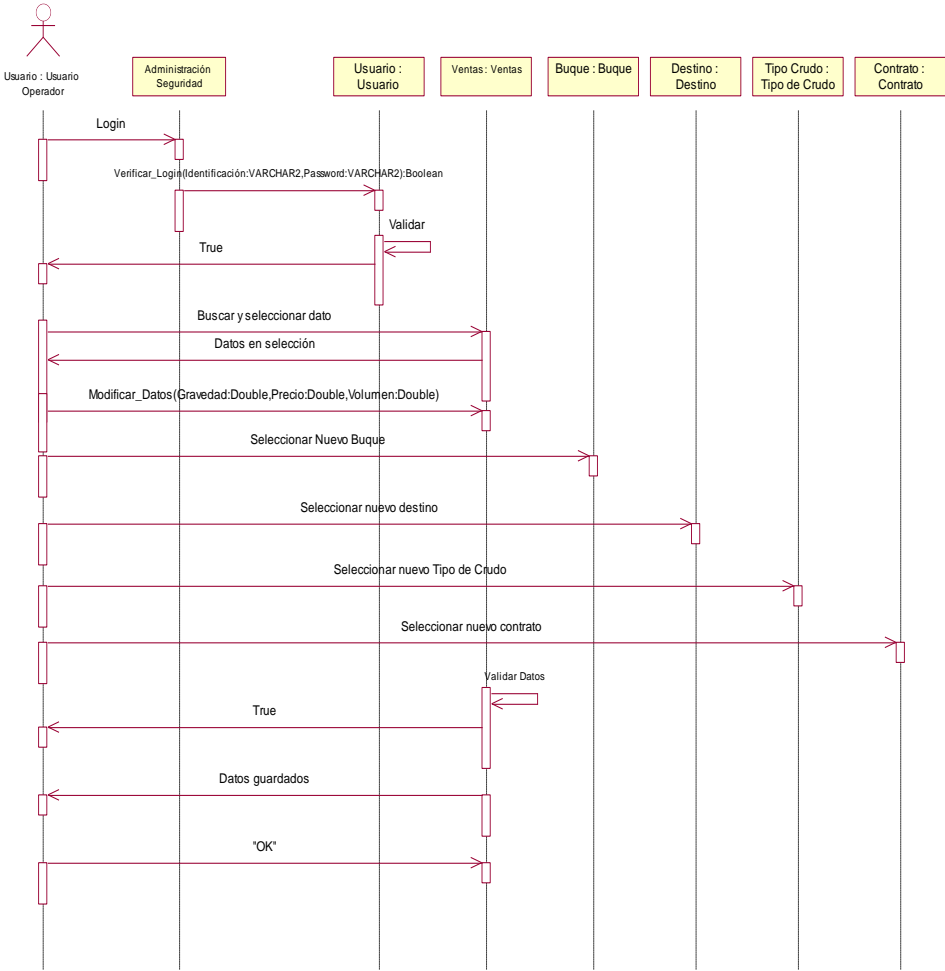
Eliminación



Ventas Ingreso

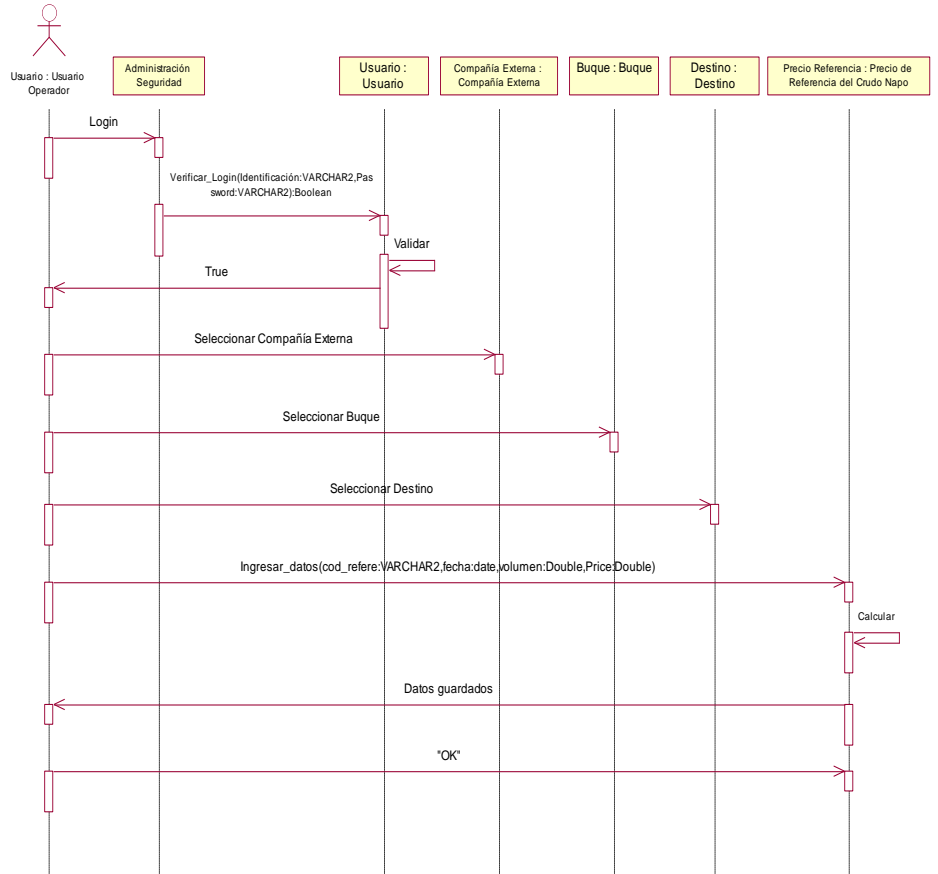


Modificación

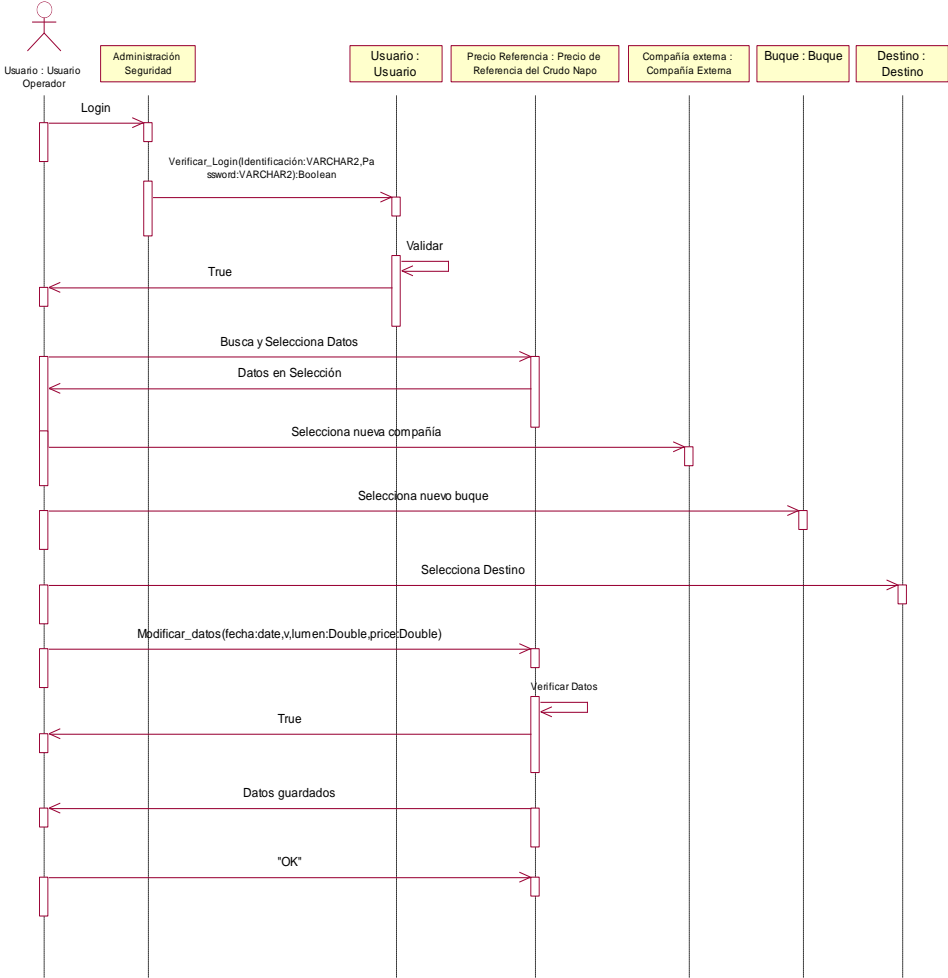


Shippers OCP

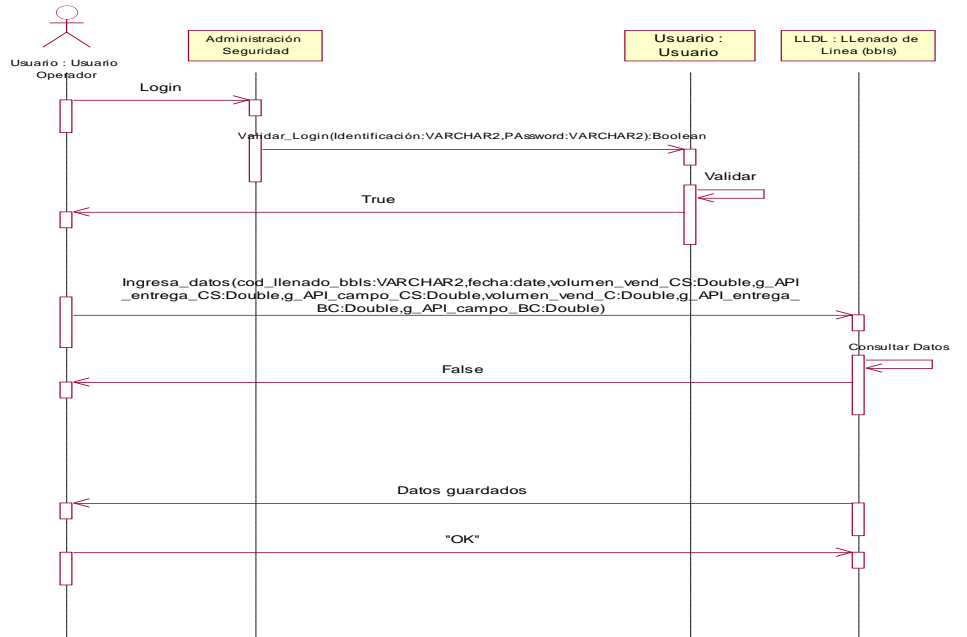
Ingreso



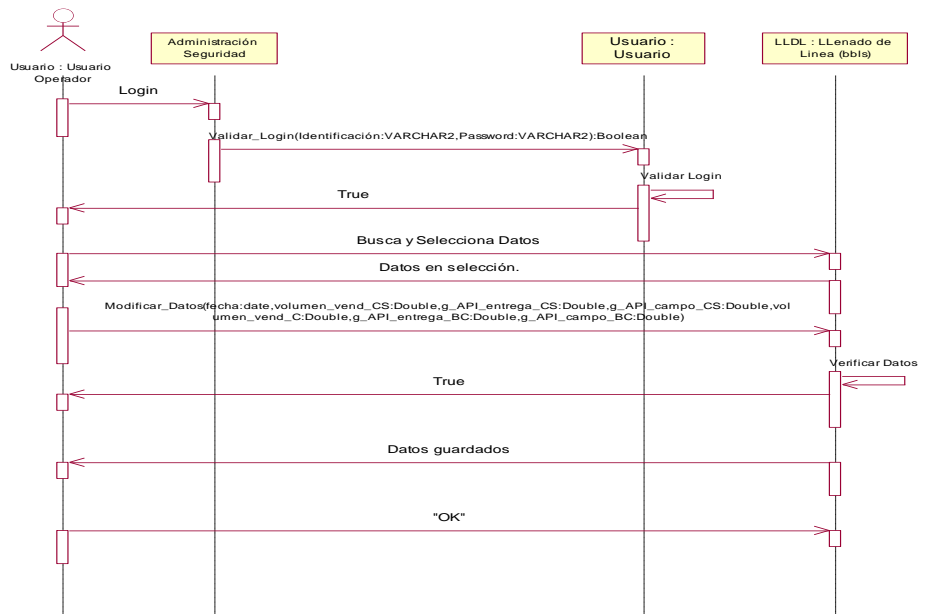
Modificación



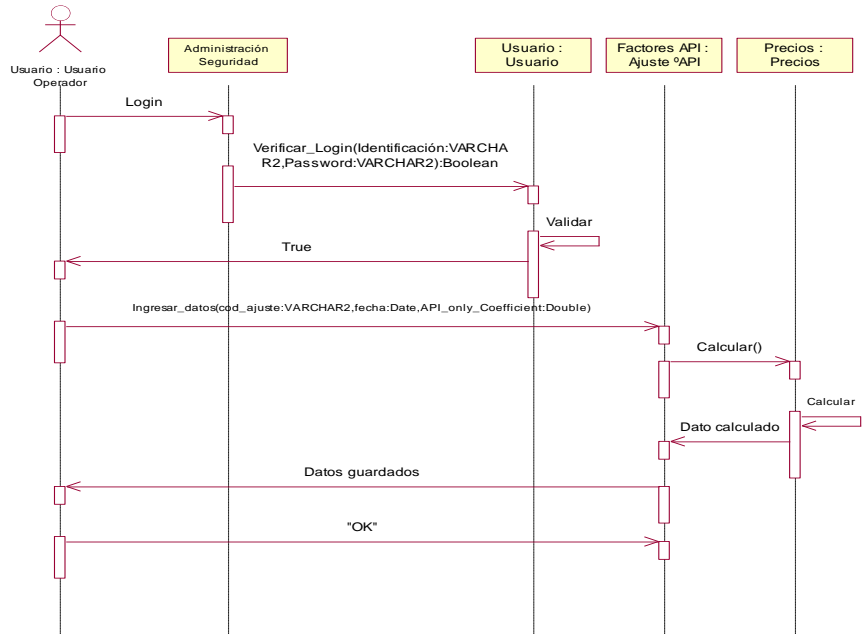
Llenado de Línea Ingreso



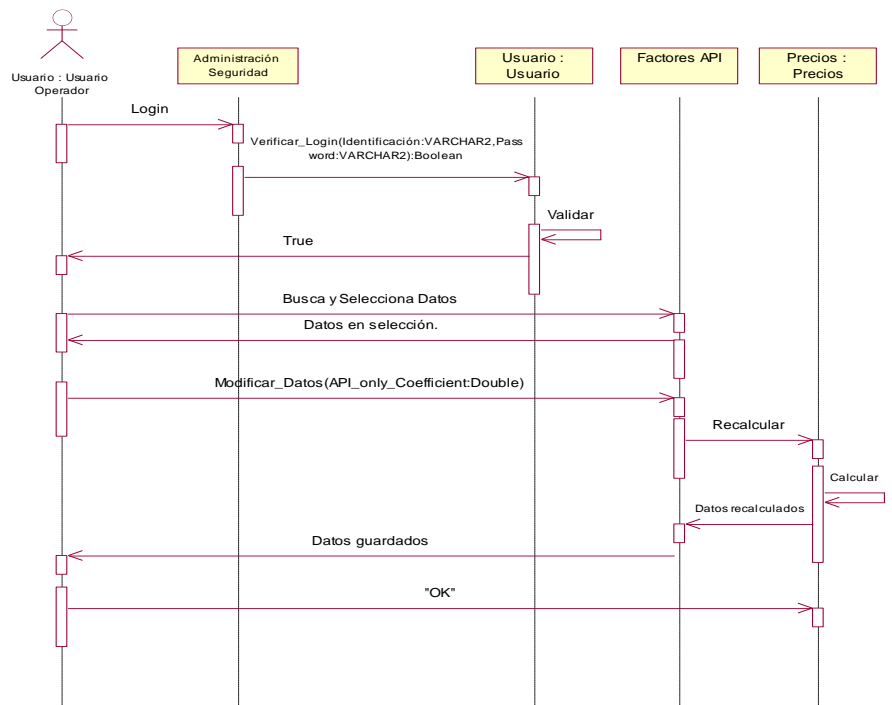
Modificación



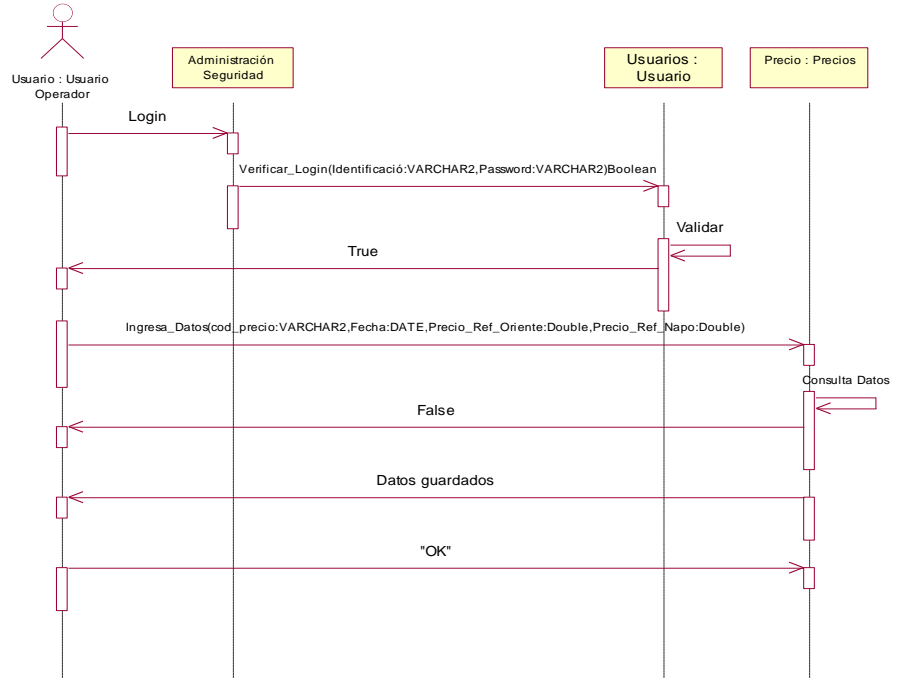
3.2.1.1.2 Precios Ajuste API Ingreso



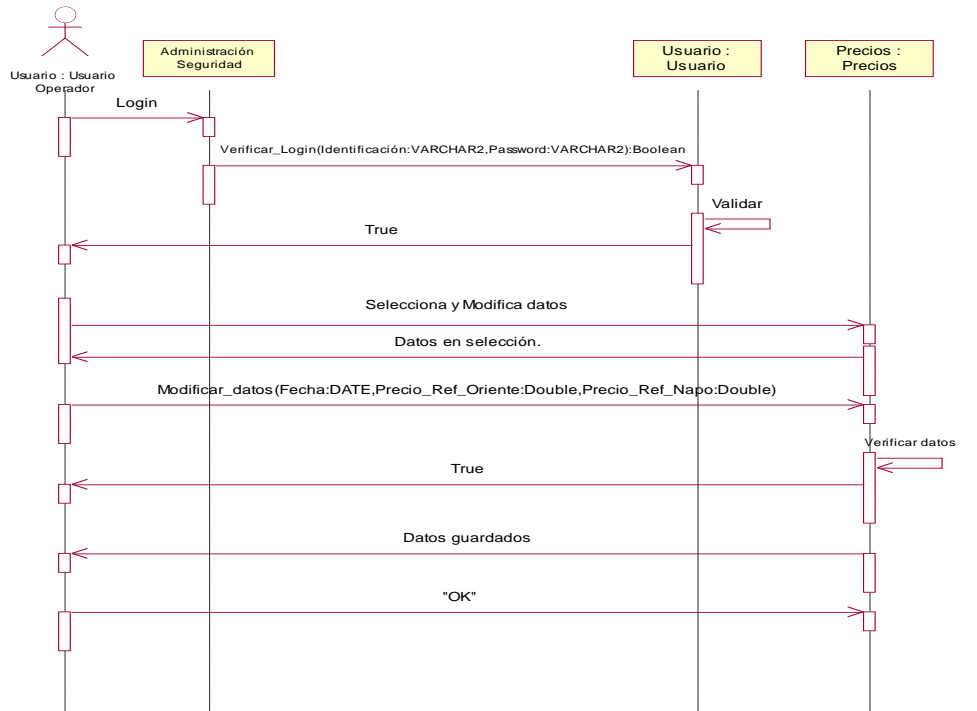
Modificación



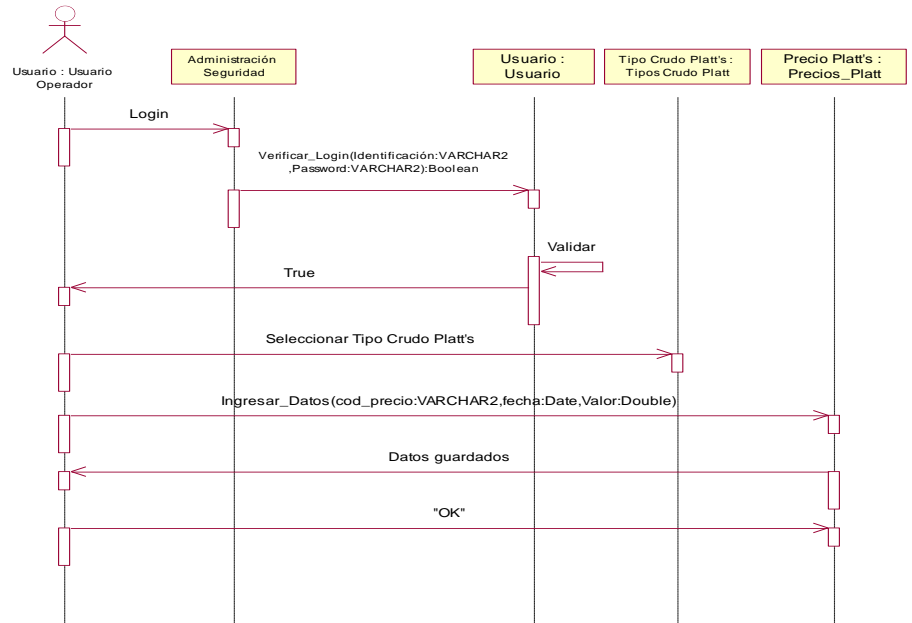
Precio de Referencia Ingreso



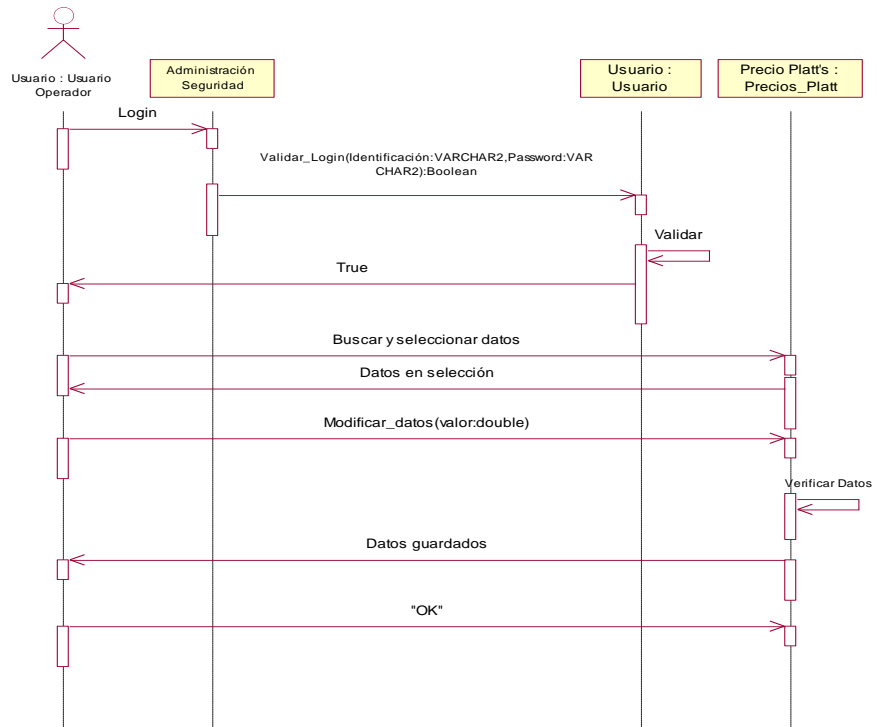
Modificación



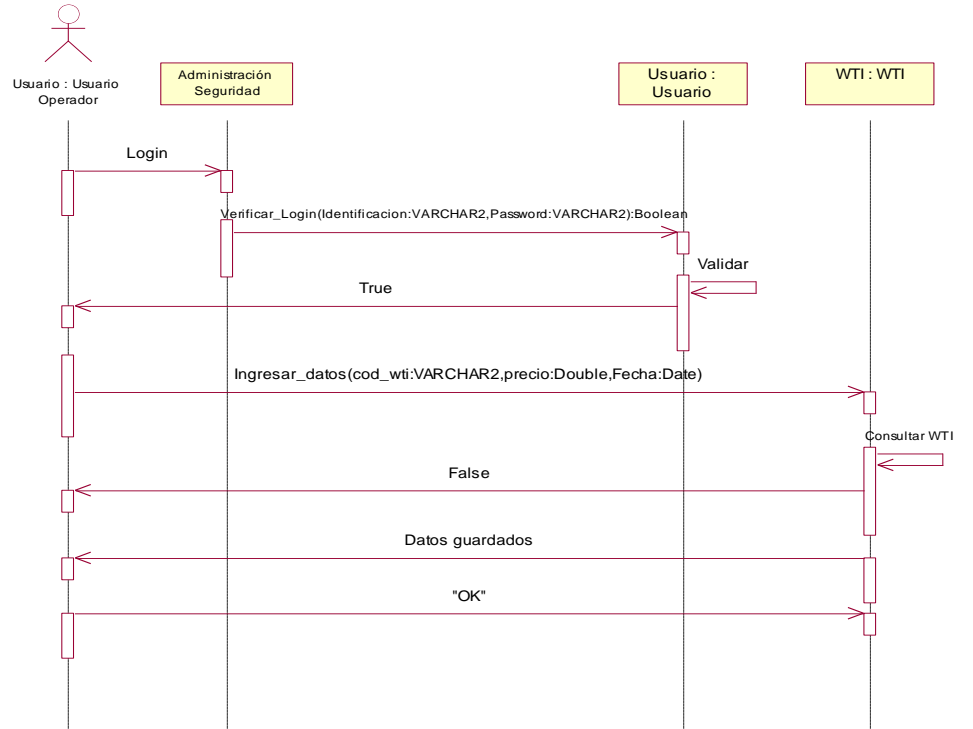
Precios Platt's Ingreso



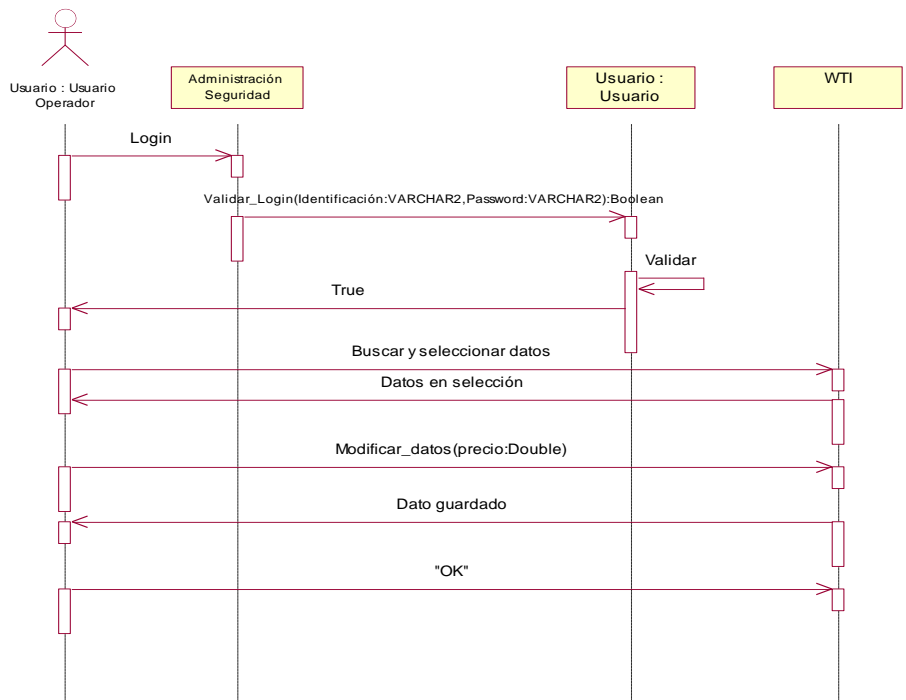
Modificación



Wti Ingreso

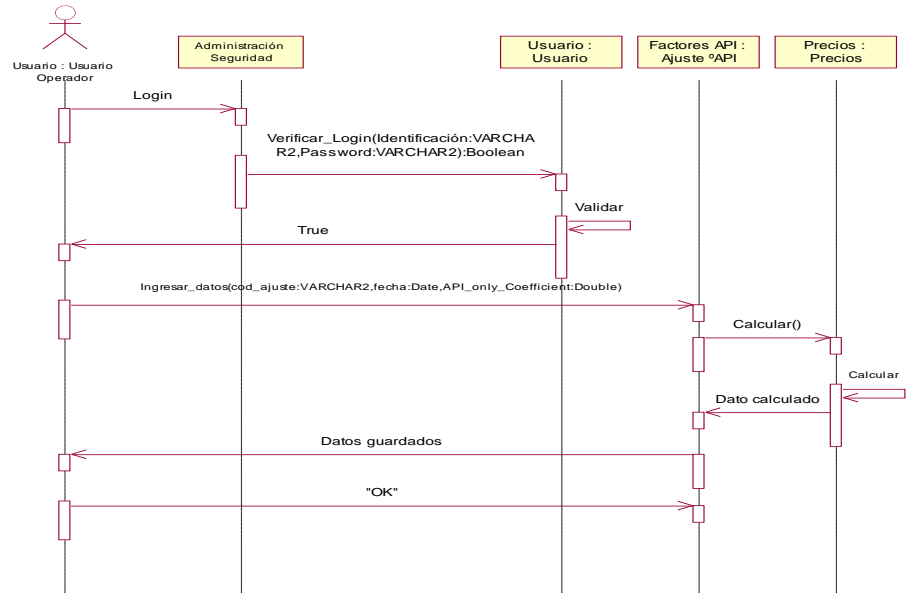


Modificación

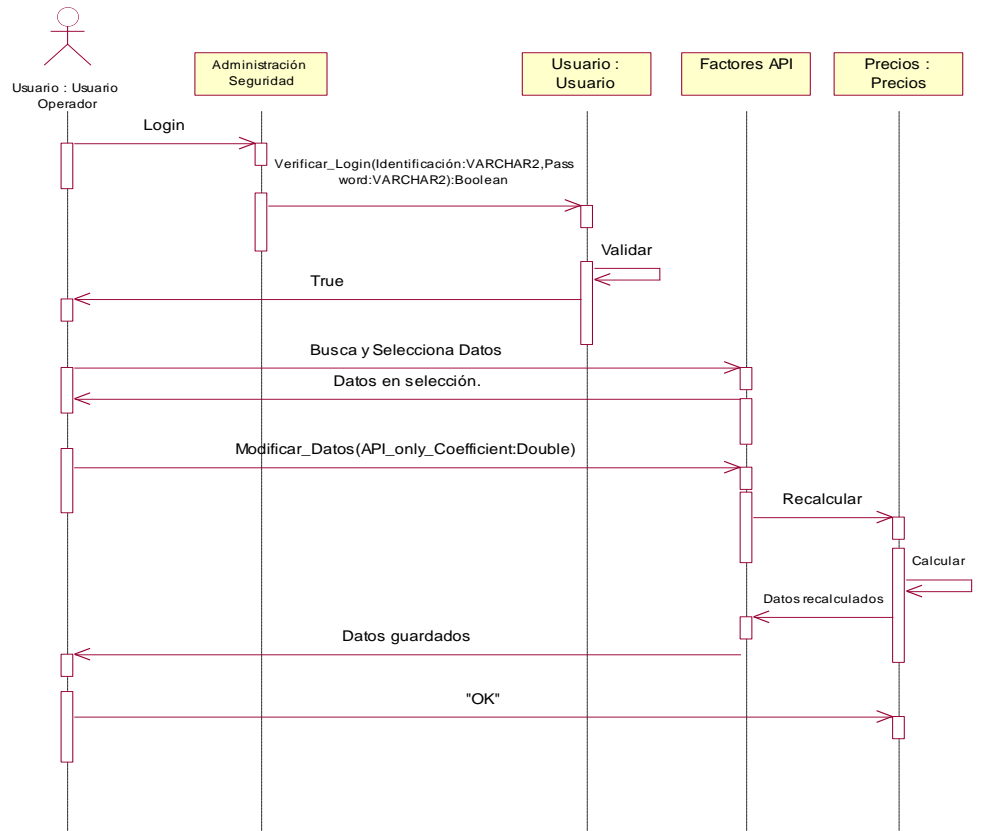


3.2.1.1.3 Factores API.

Ingreso

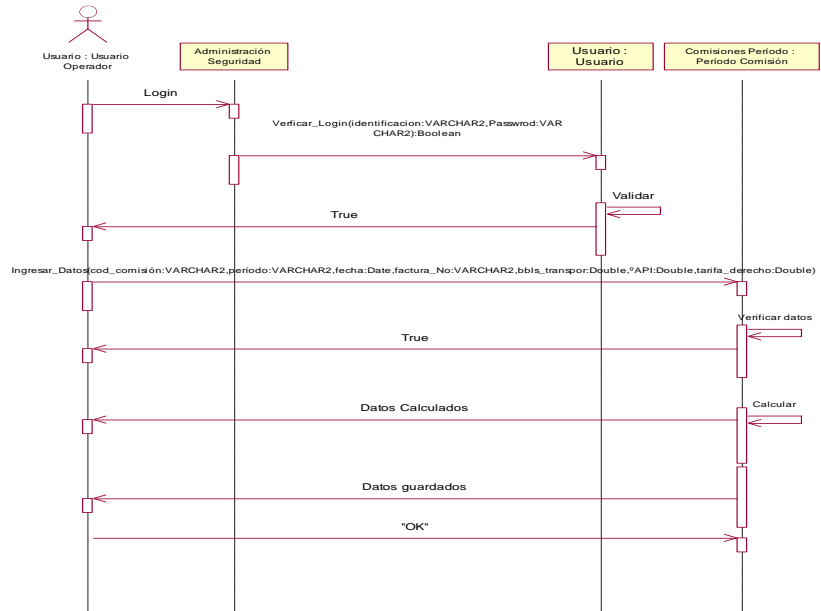


Modificación

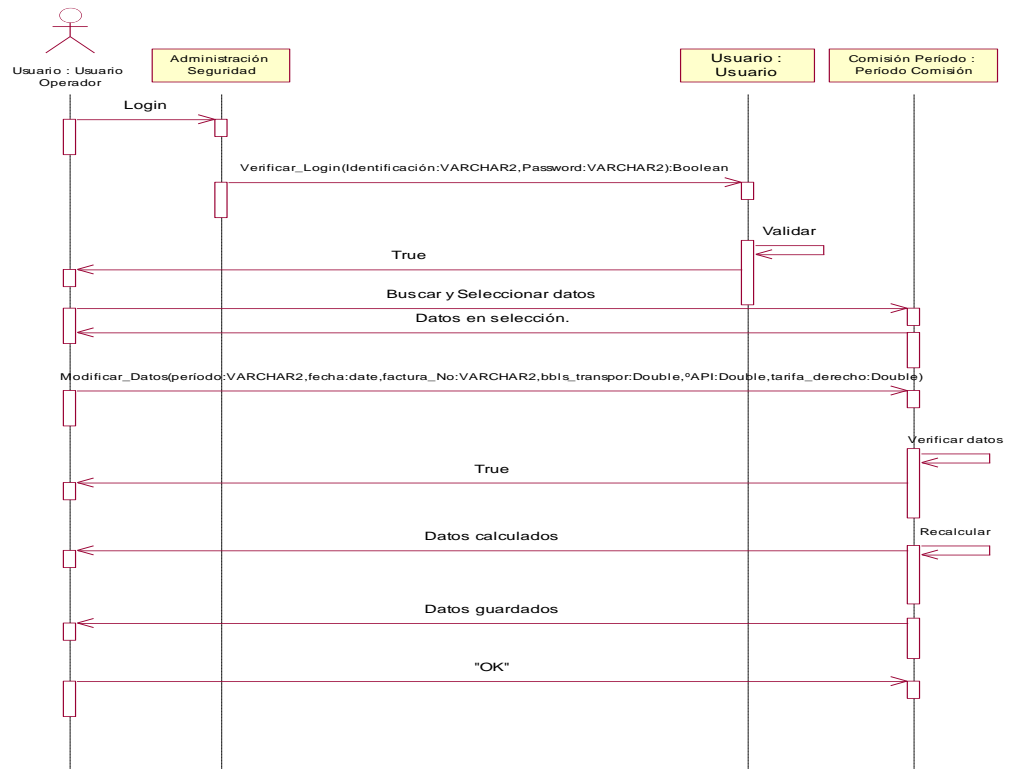


3.2.1.1.4 OCP

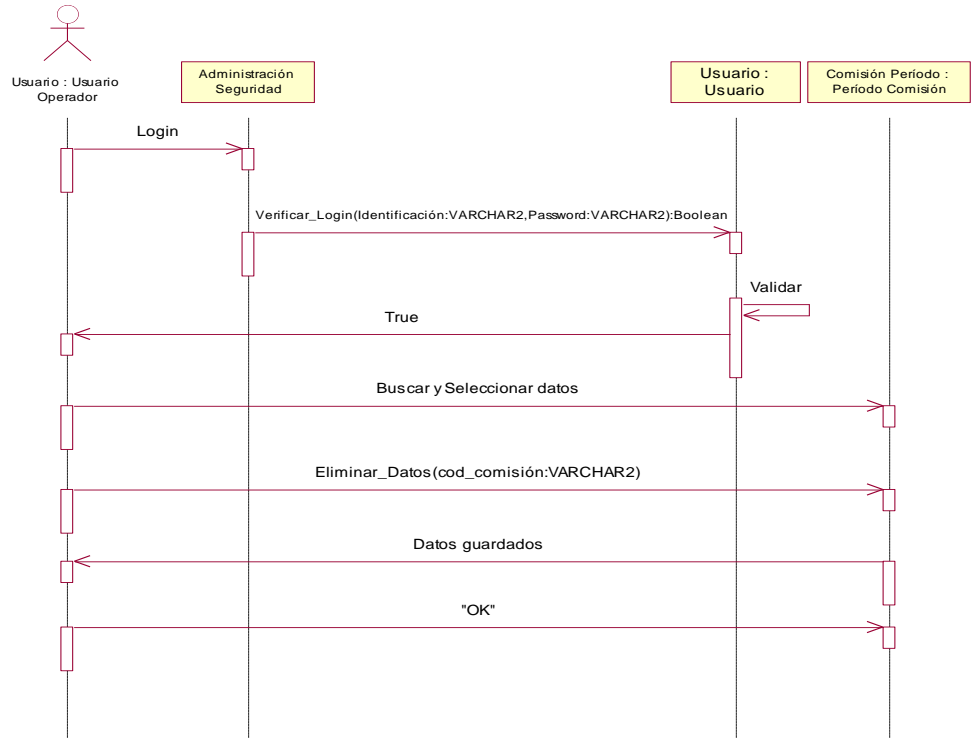
Factura Transporte de Crudo Pago por Comisión Ingresar



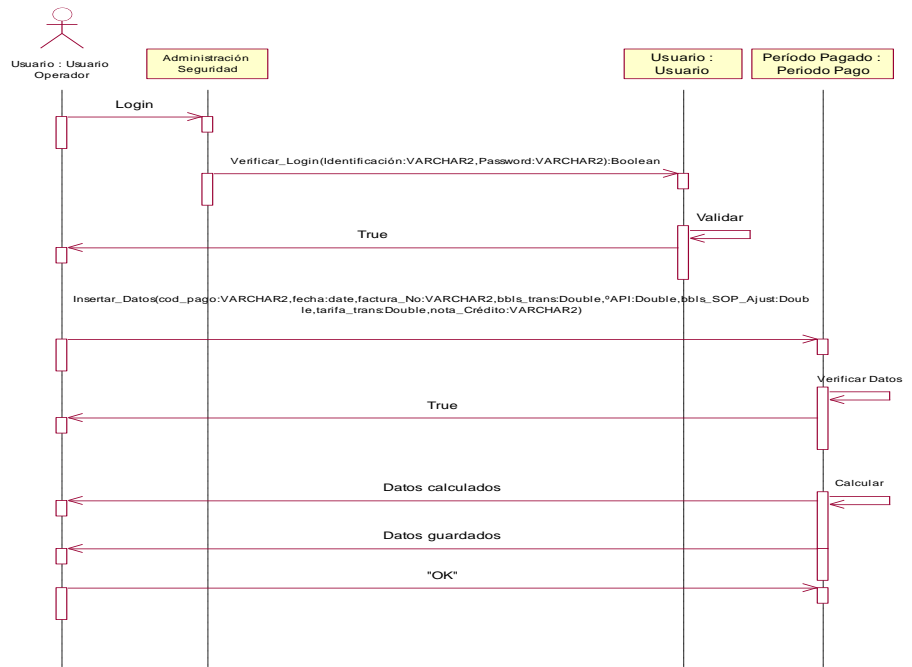
Modificar



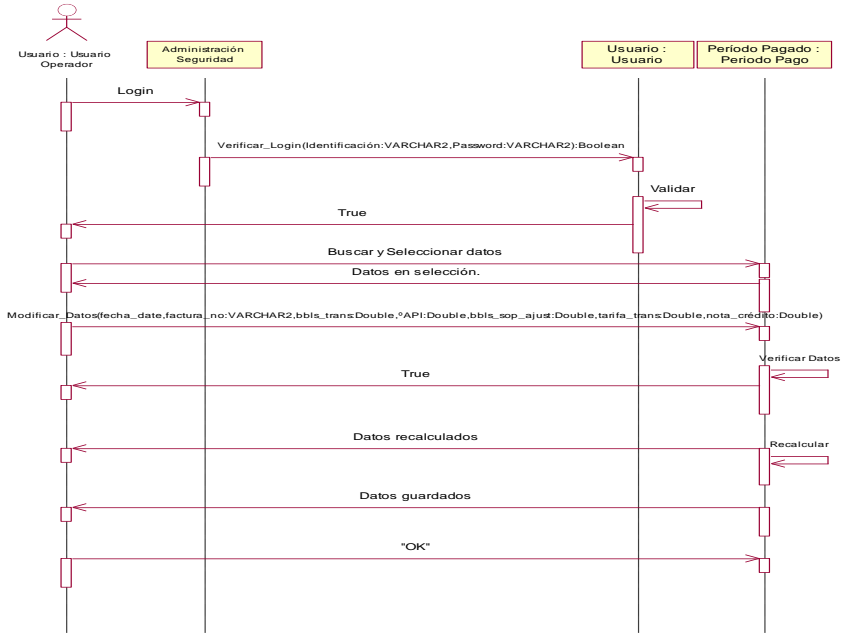
Eliminar



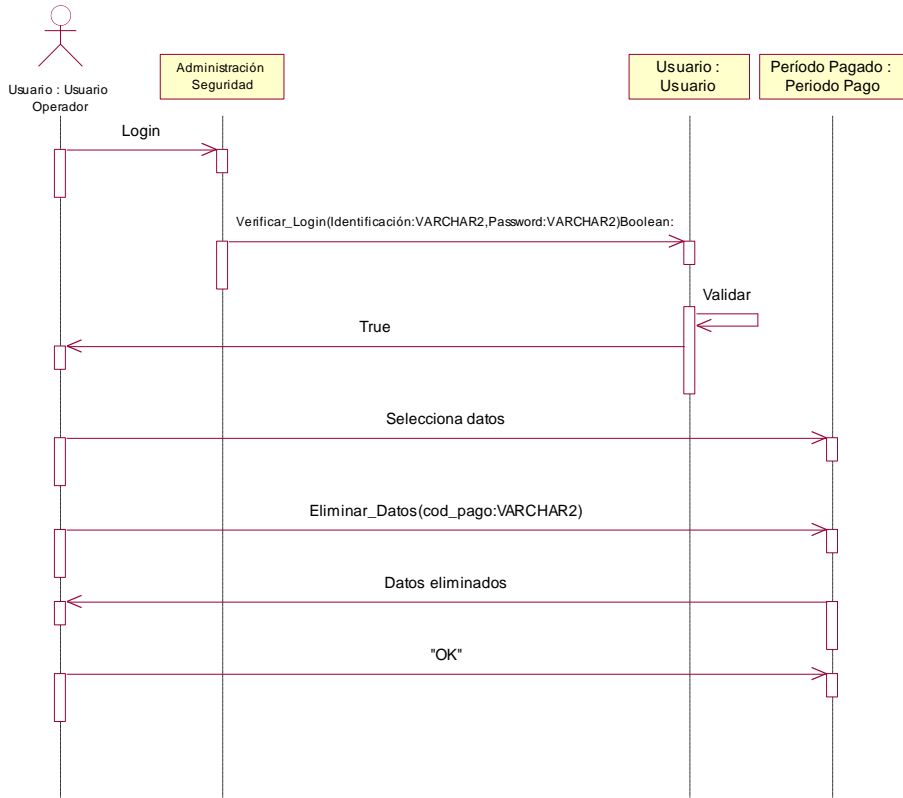
Período de Pago Ingresar



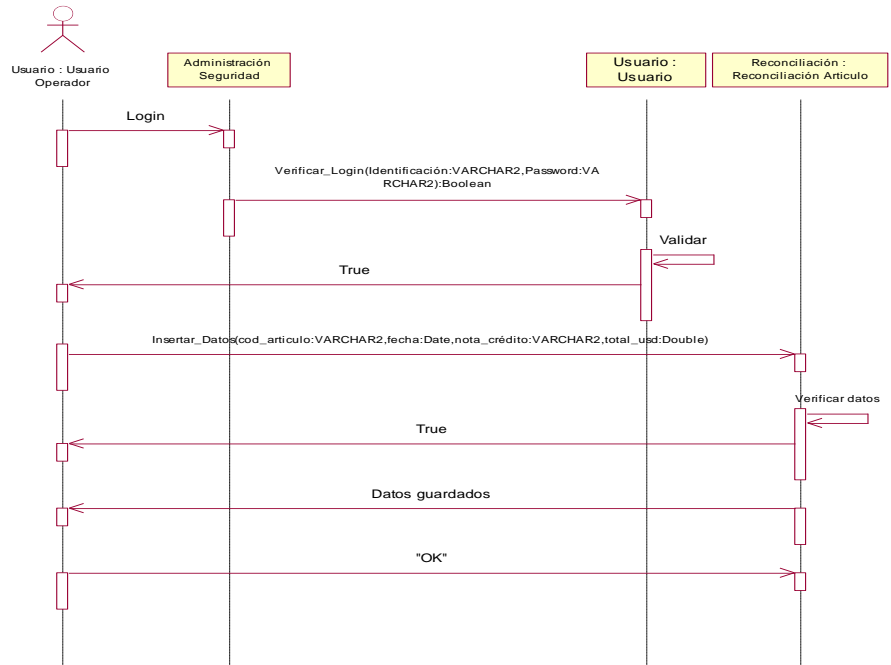
Modificar



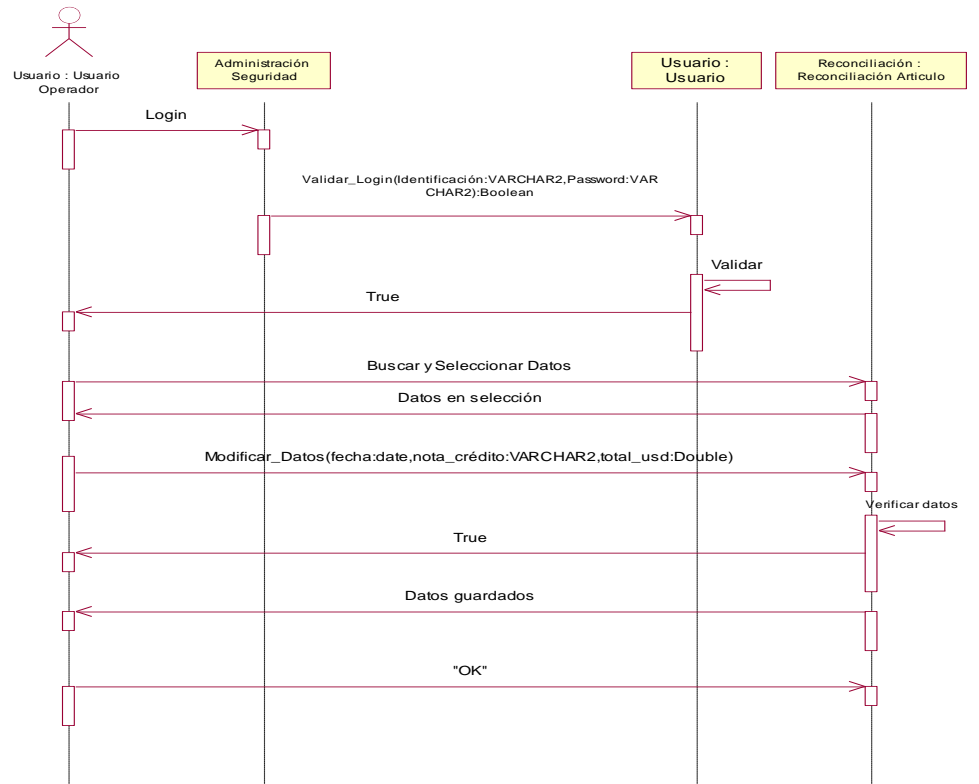
Eliminar



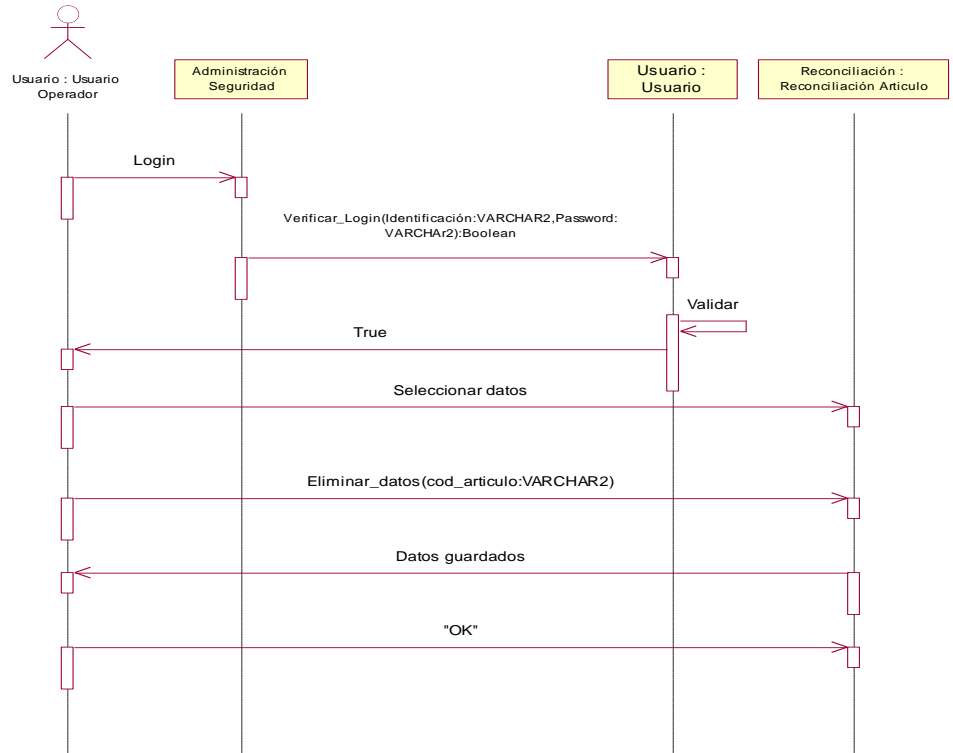
Reconciliación ISTA Ingresar



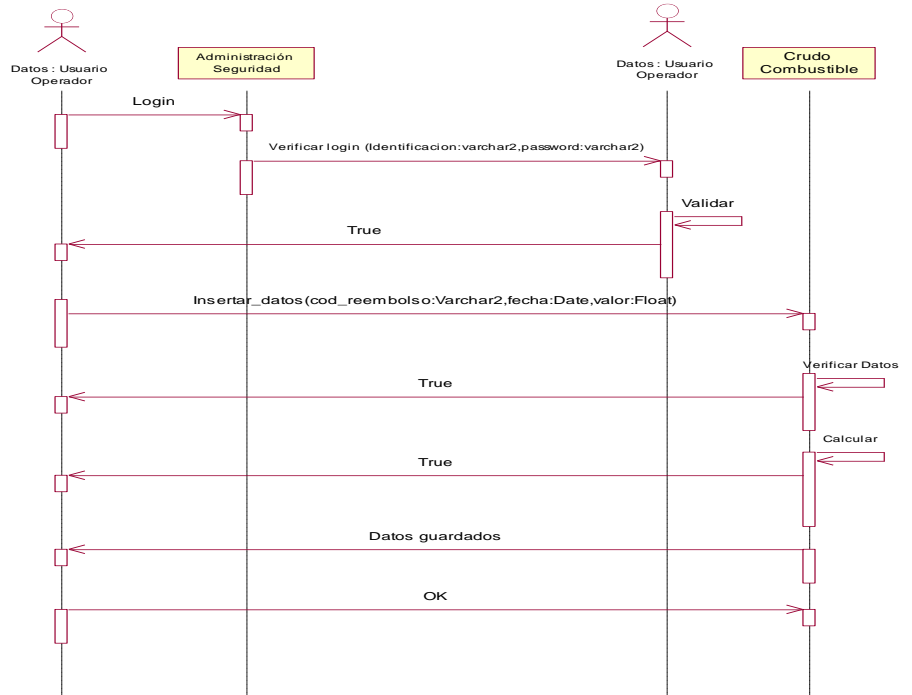
Modificar



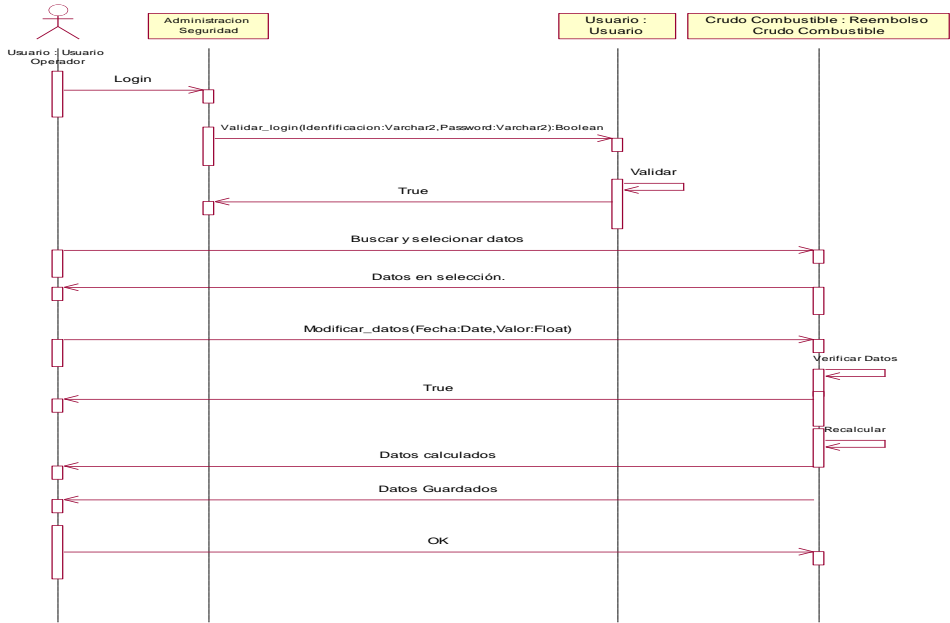
Eliminar



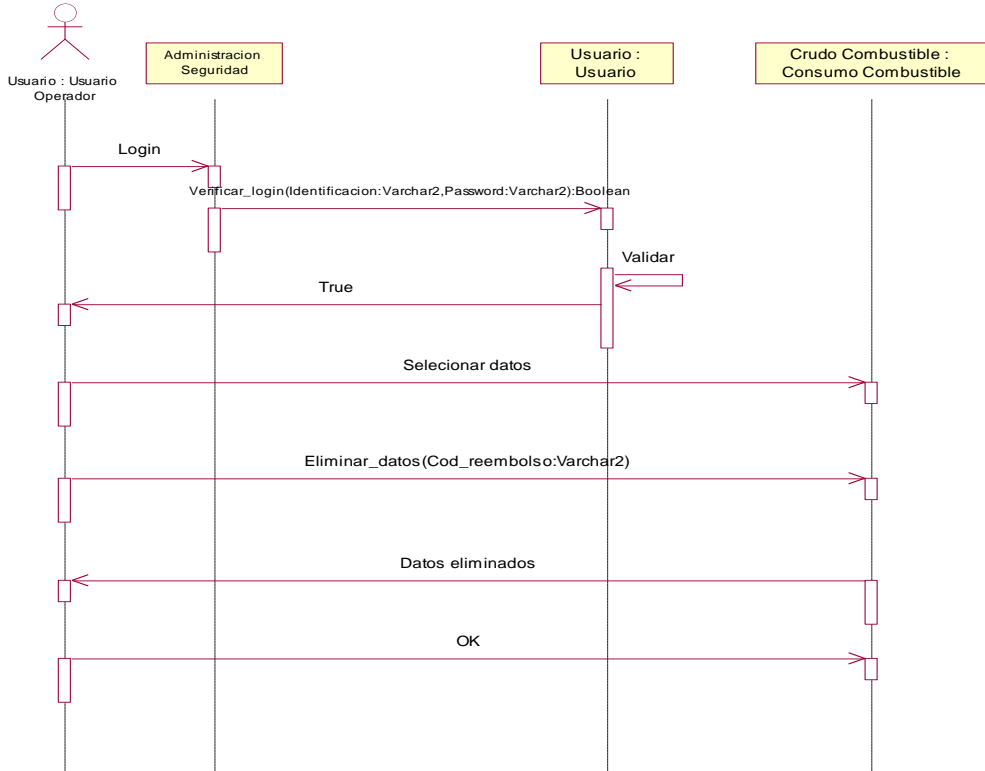
Reembolso Combustible Ingreso



Modificación



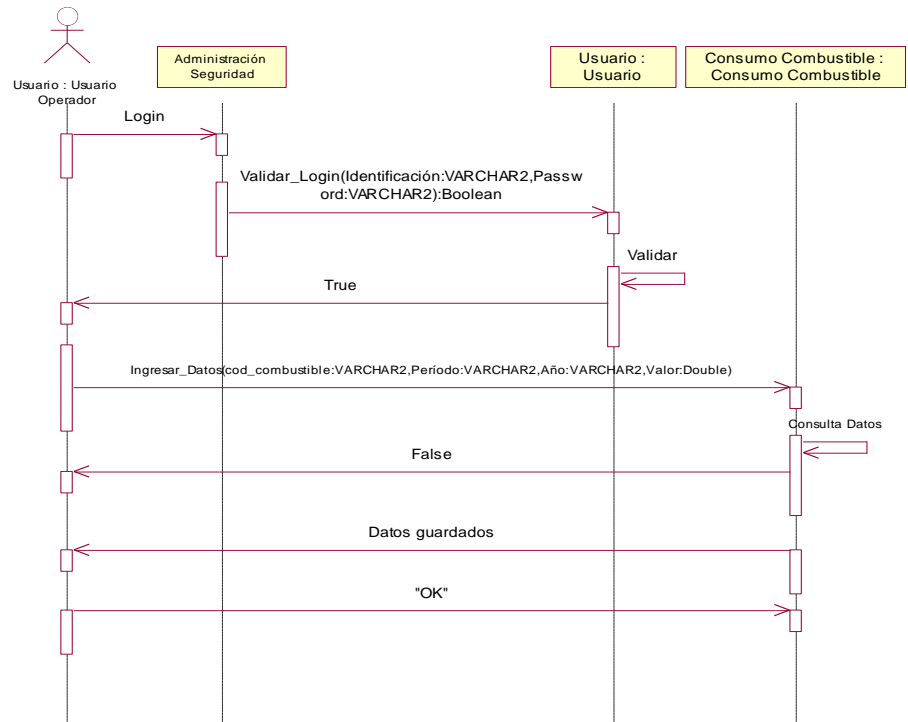
Eliminar



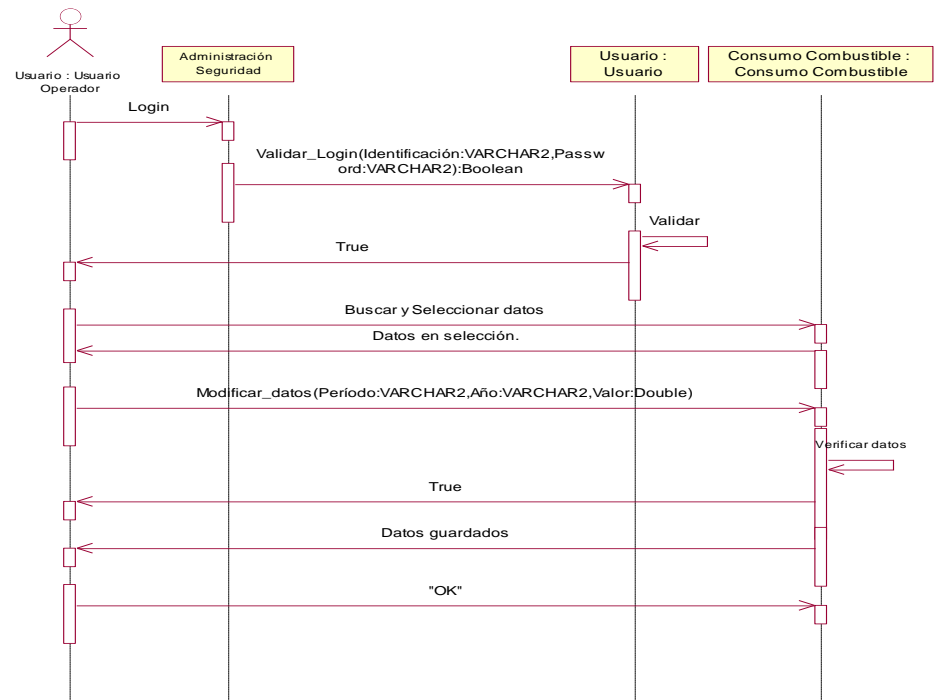
3.2.1.1.5 Descuentos

Cosumo de Combustible

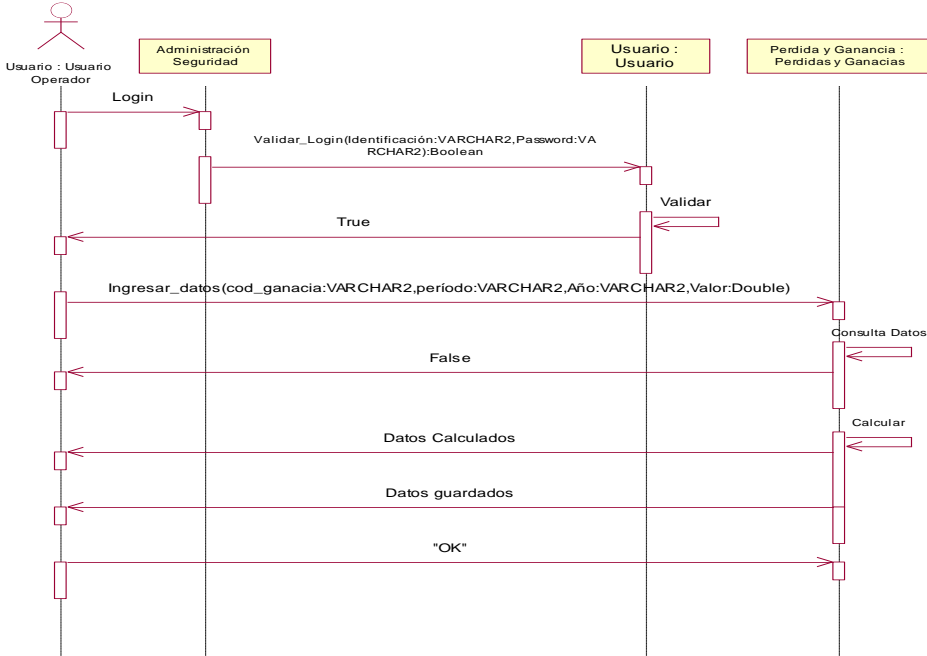
Ingreso



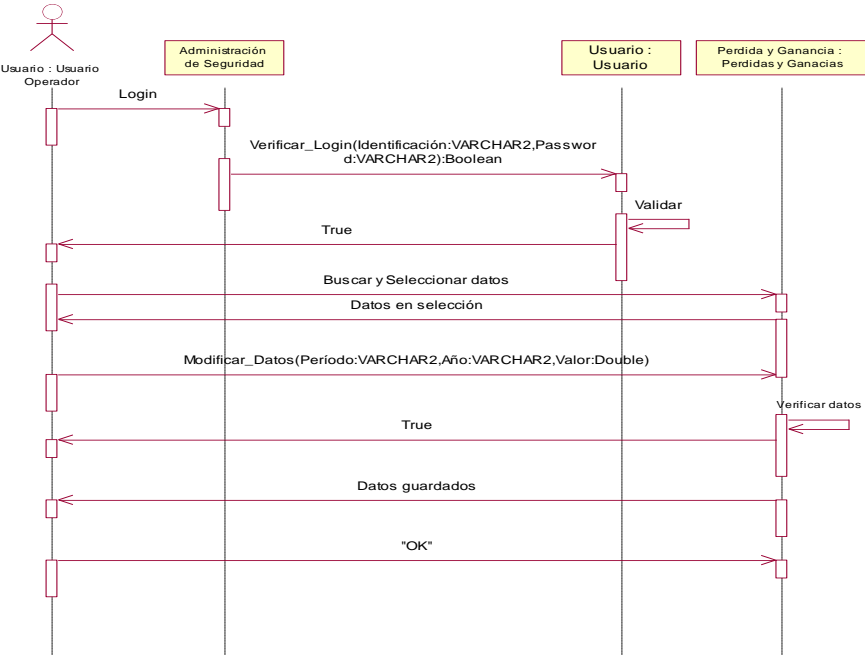
Modificación



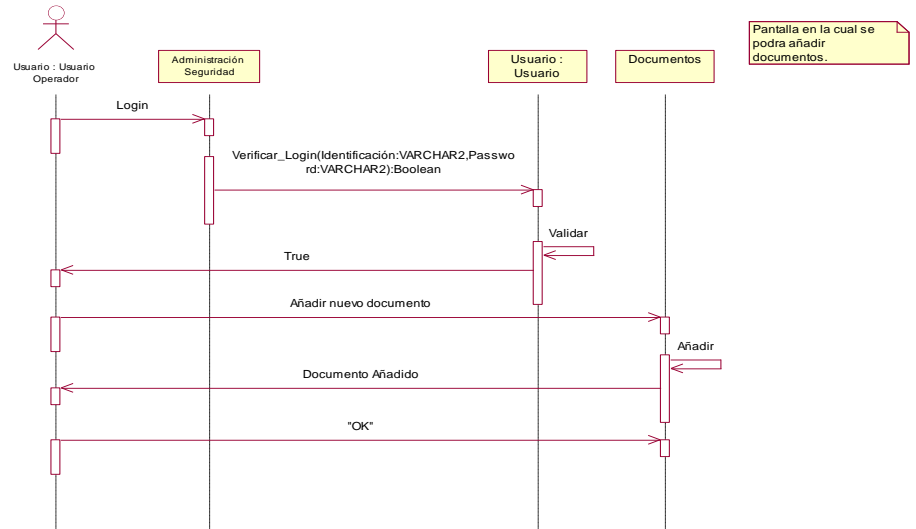
Perdidas y Ganancias Ingreso



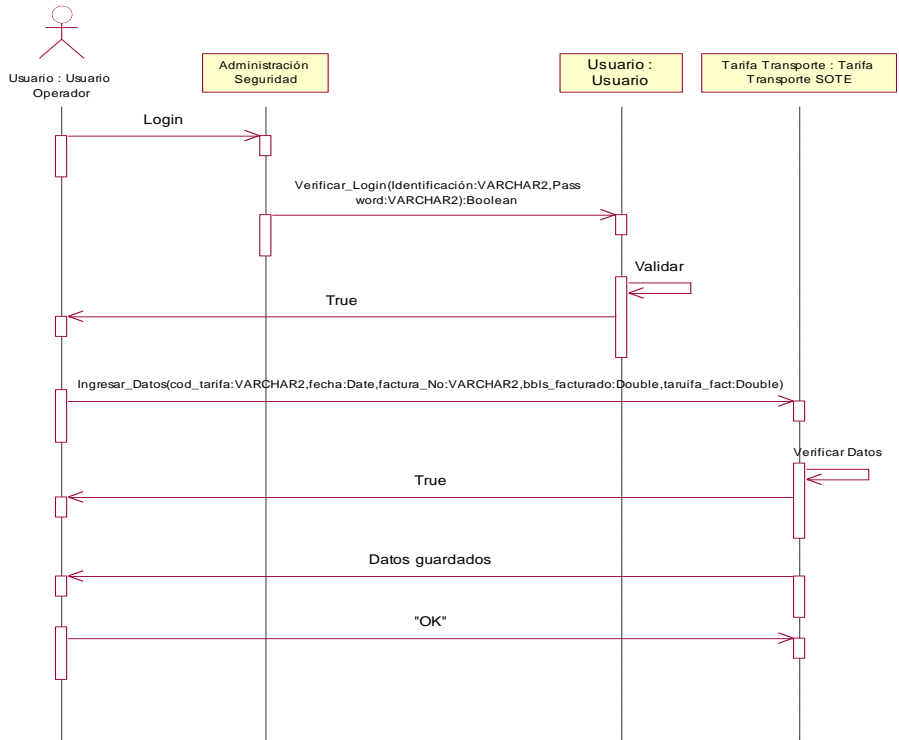
Modificación



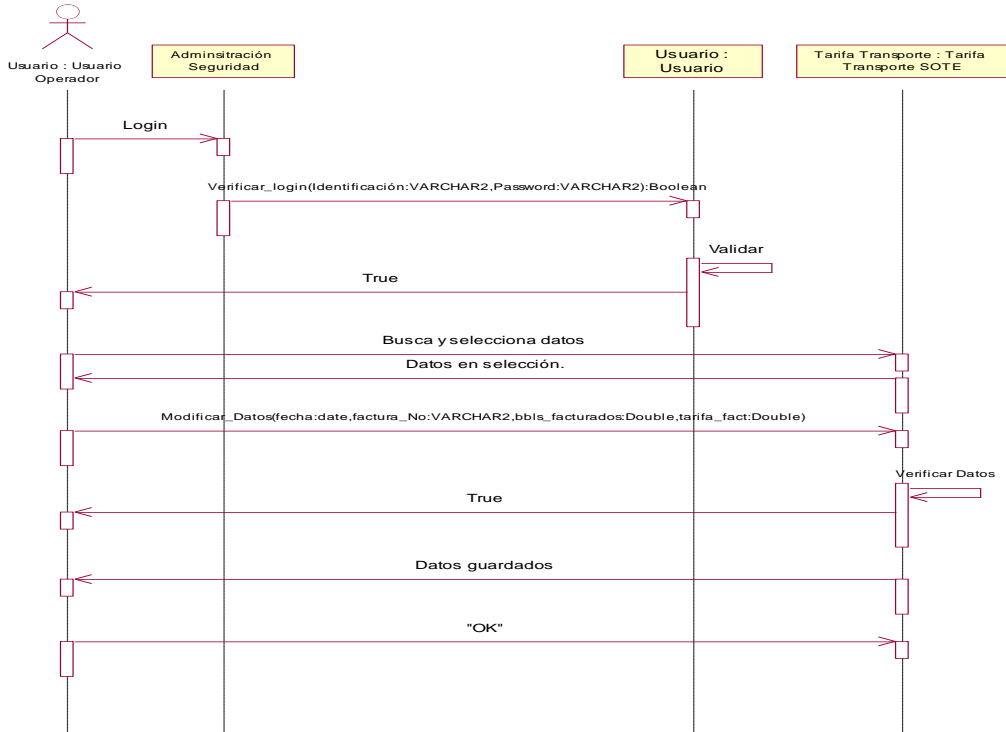
Programas de Embarque. Añadir



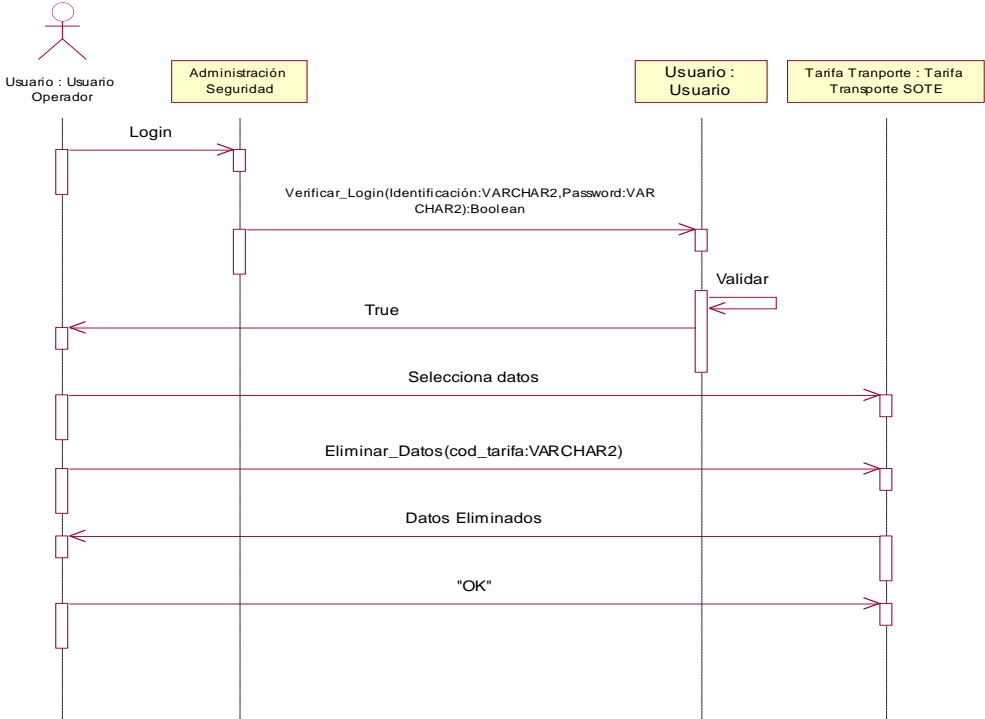
3.2.1.1.6 Facturación SOTE Tarifa de Transporte Ingreso



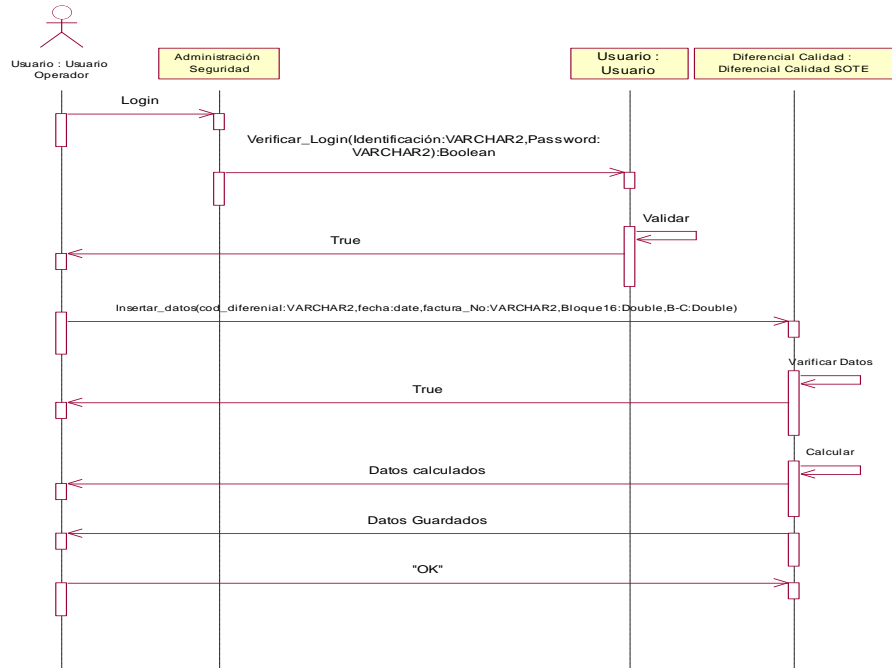
Modificación



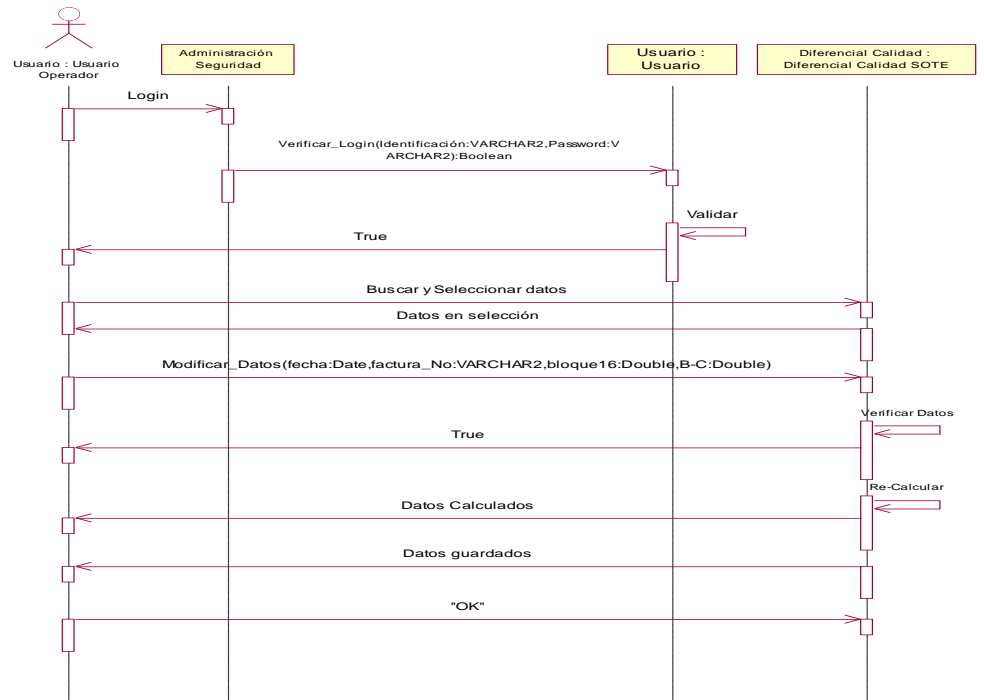
Eliminación



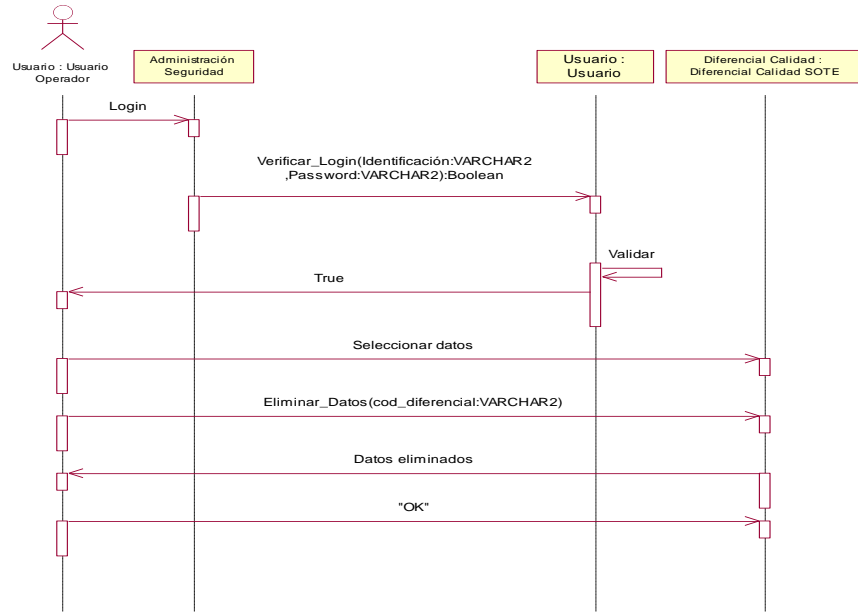
Diferencial de Calidad SOTE Ingreso



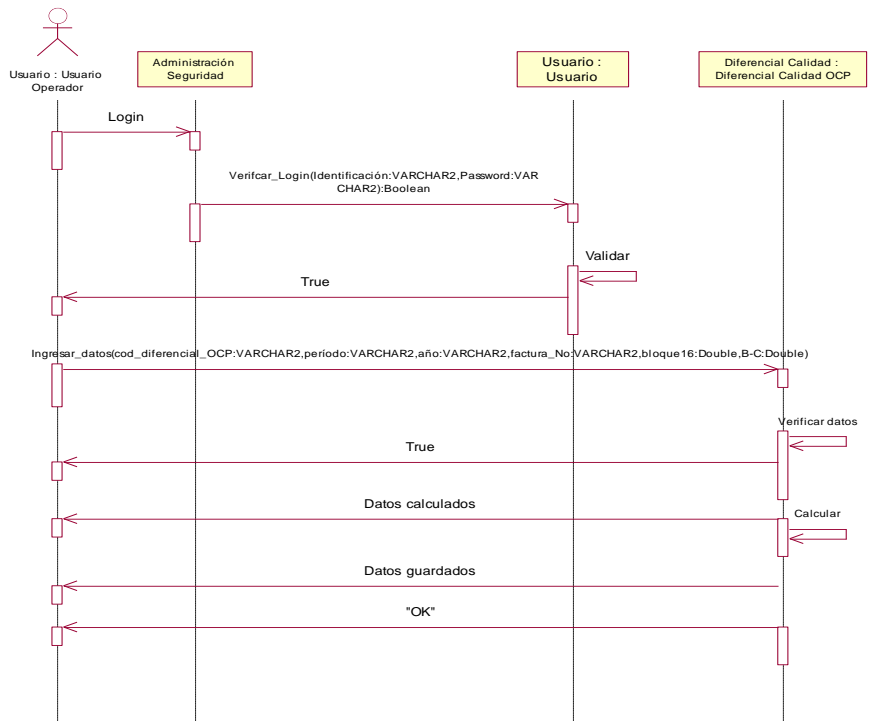
Modificación



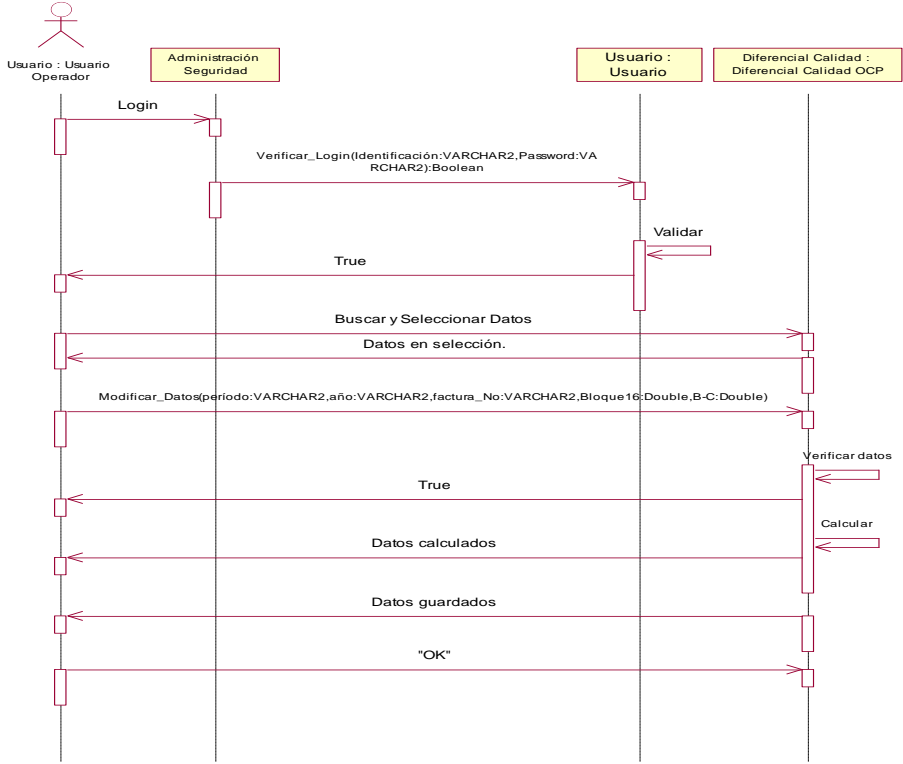
Eliminación



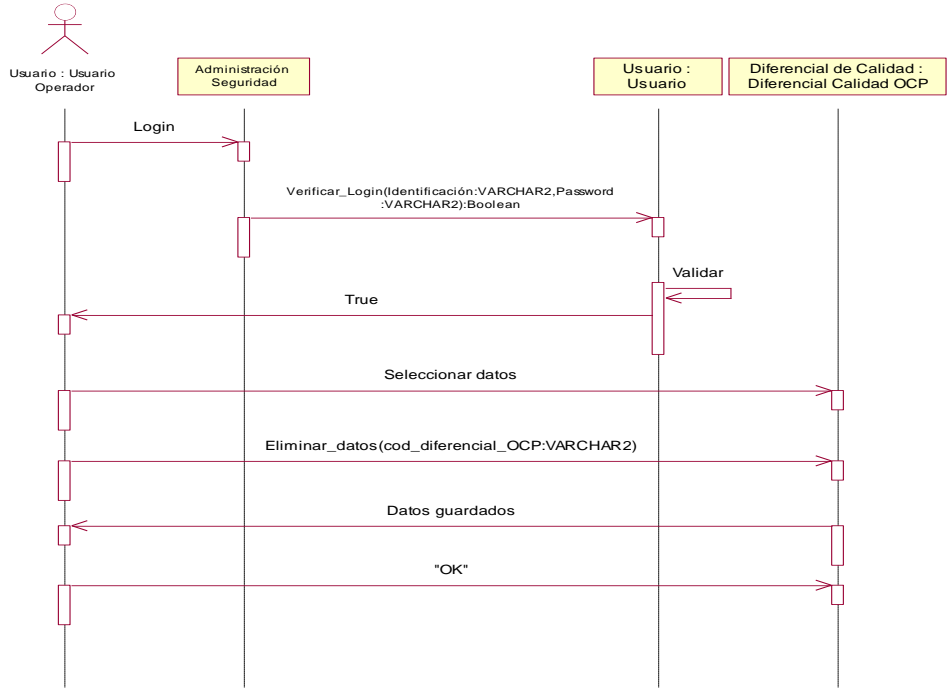
Diferencial de Calidad OCP Ingreso



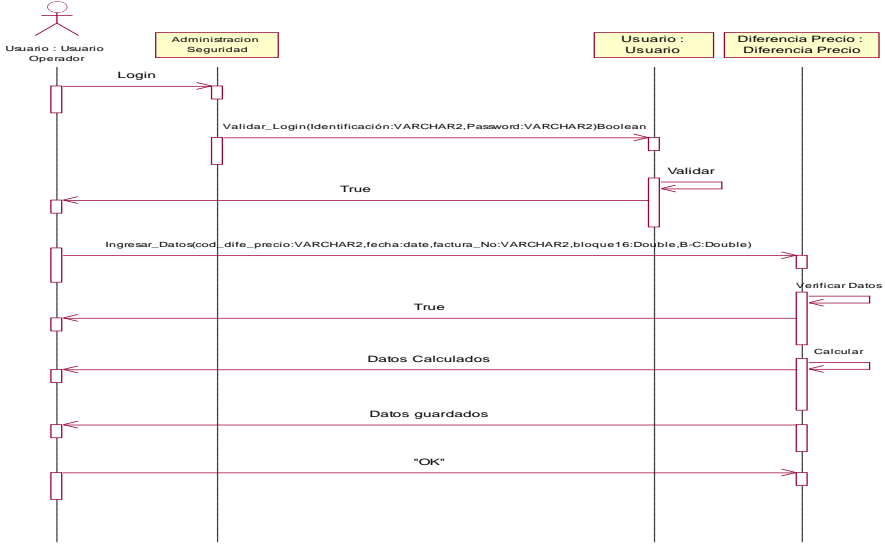
Modificación



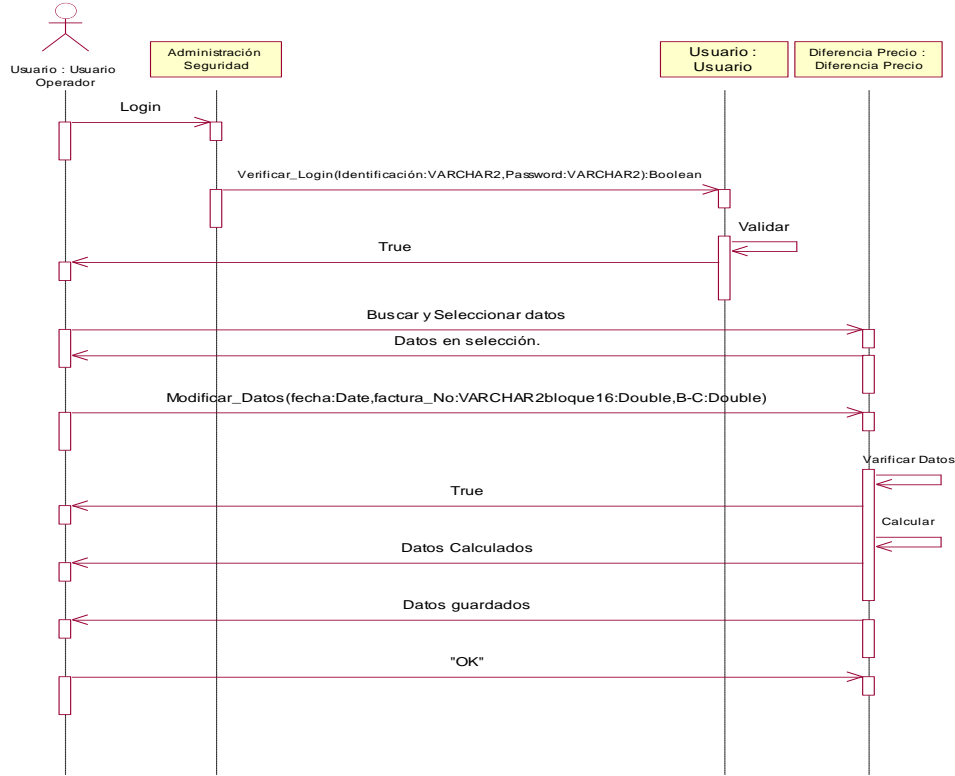
Eliminación



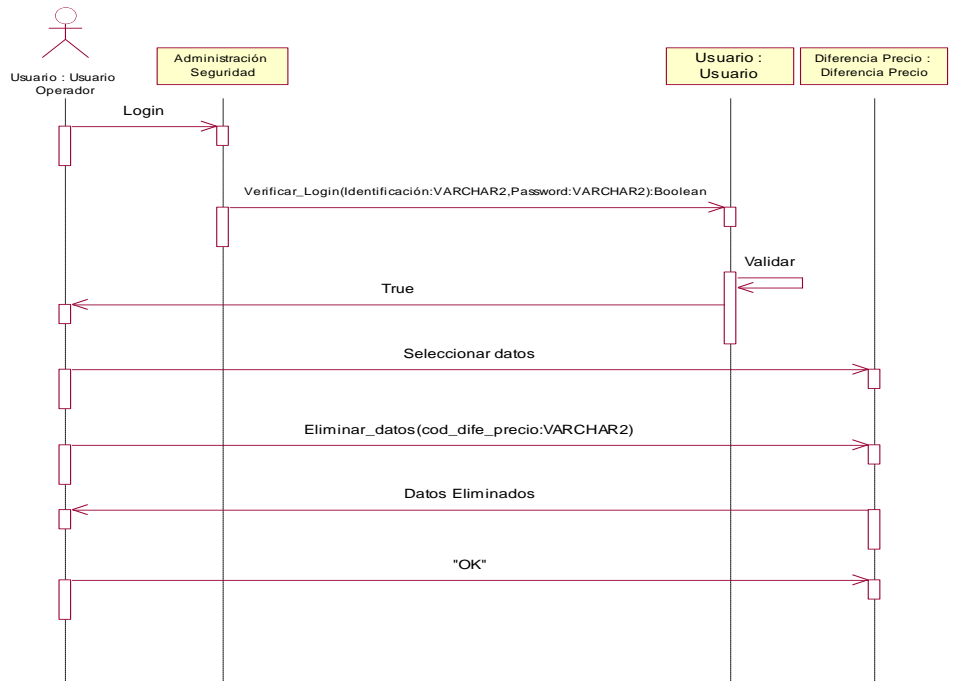
Diferencia de Precio Ingreso



Modificación

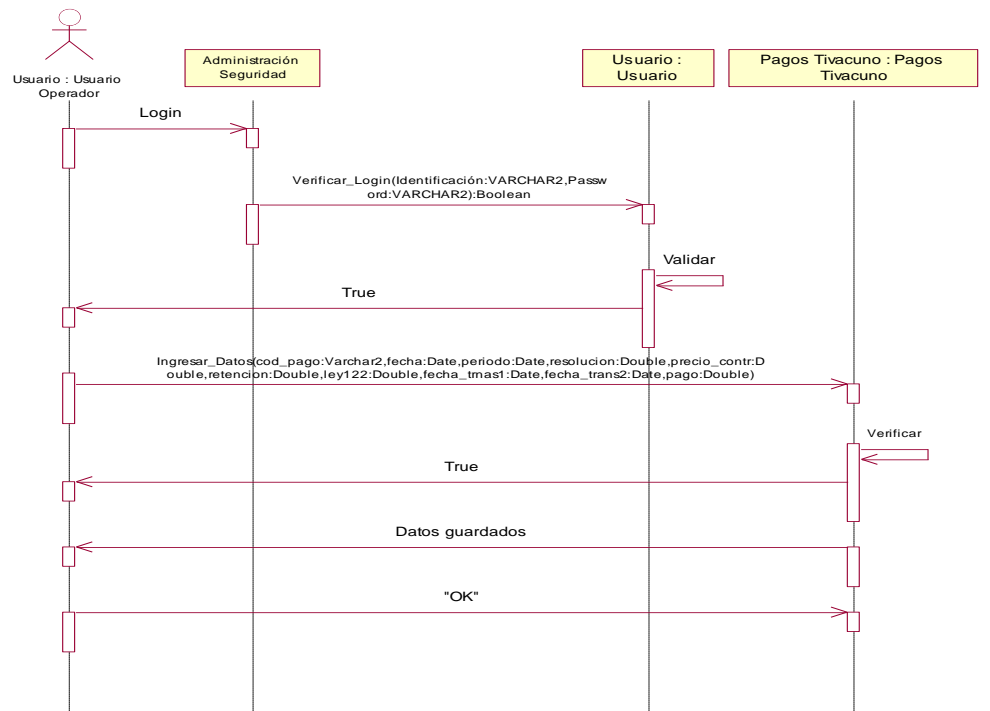


Eliminación

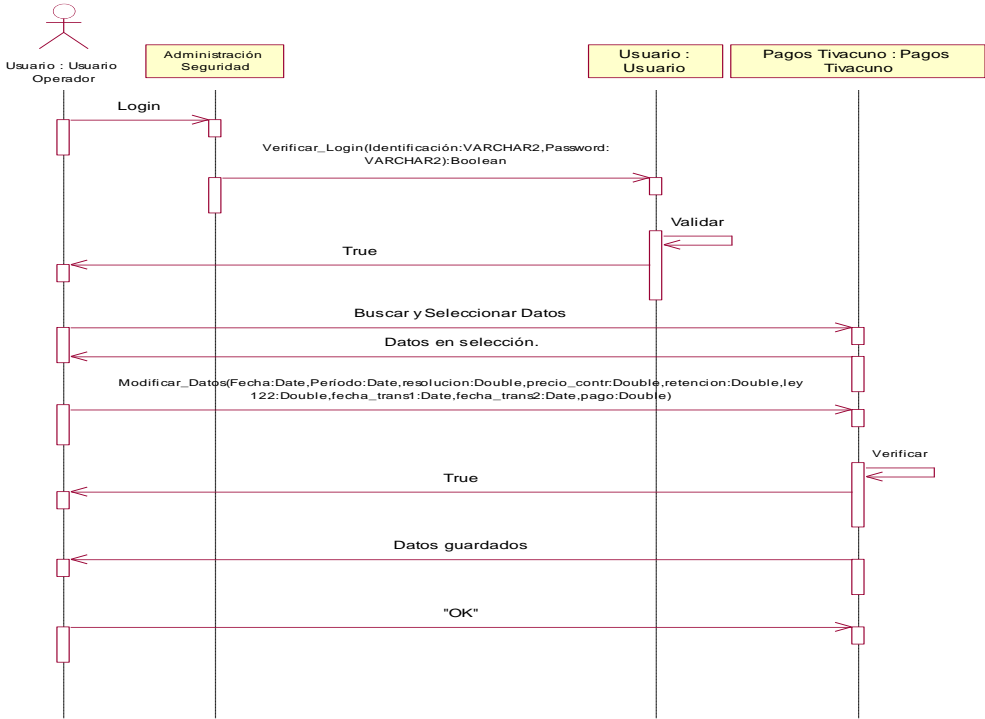


3.2.1.1.7 Pagos Tivacuno

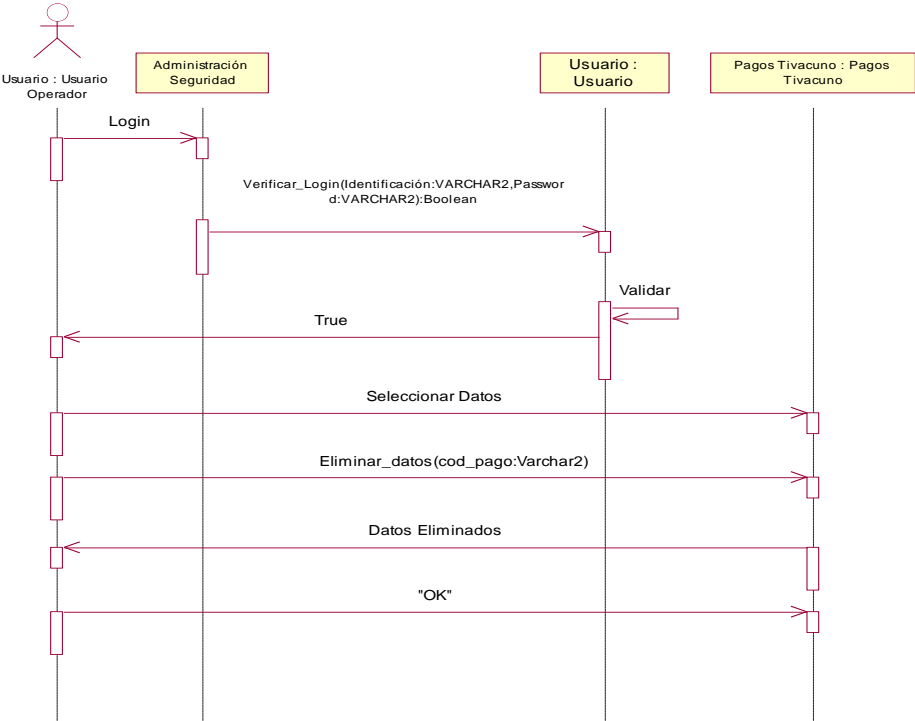
Ingreso



Modificación

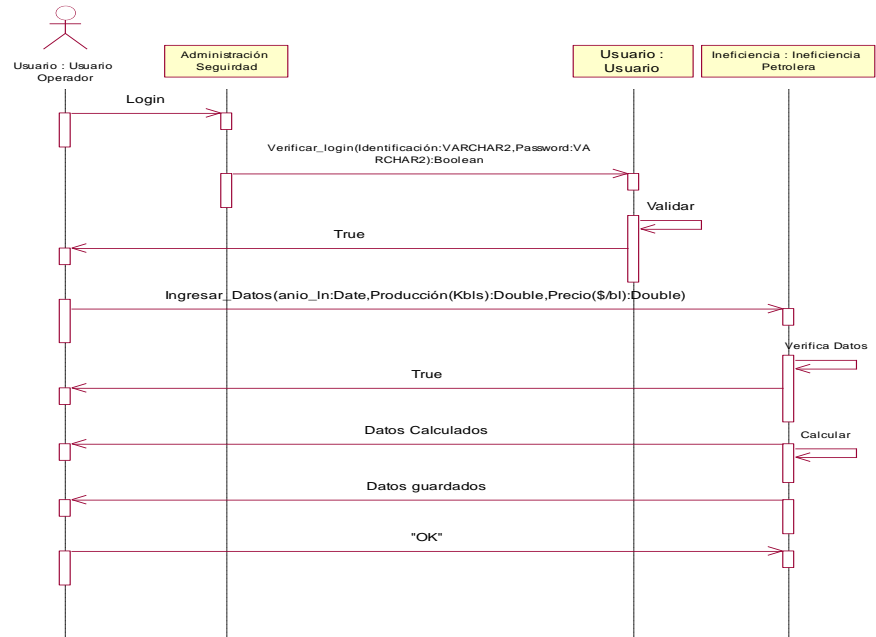


Eliminación

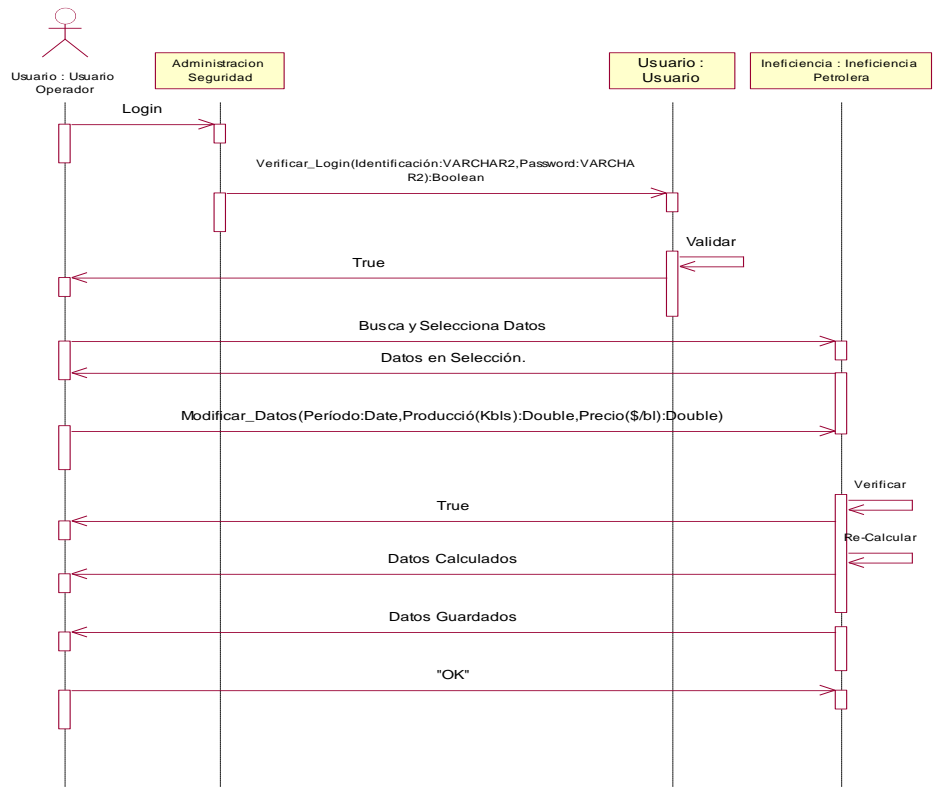


3.2.1.1.8 Ineficiencia Petrolera

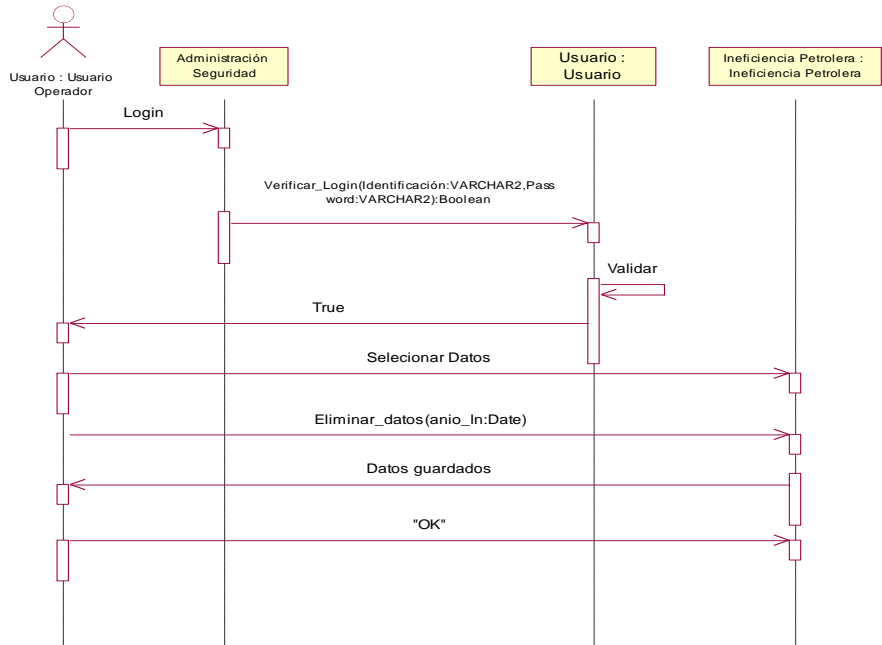
Ingreso



Modificación

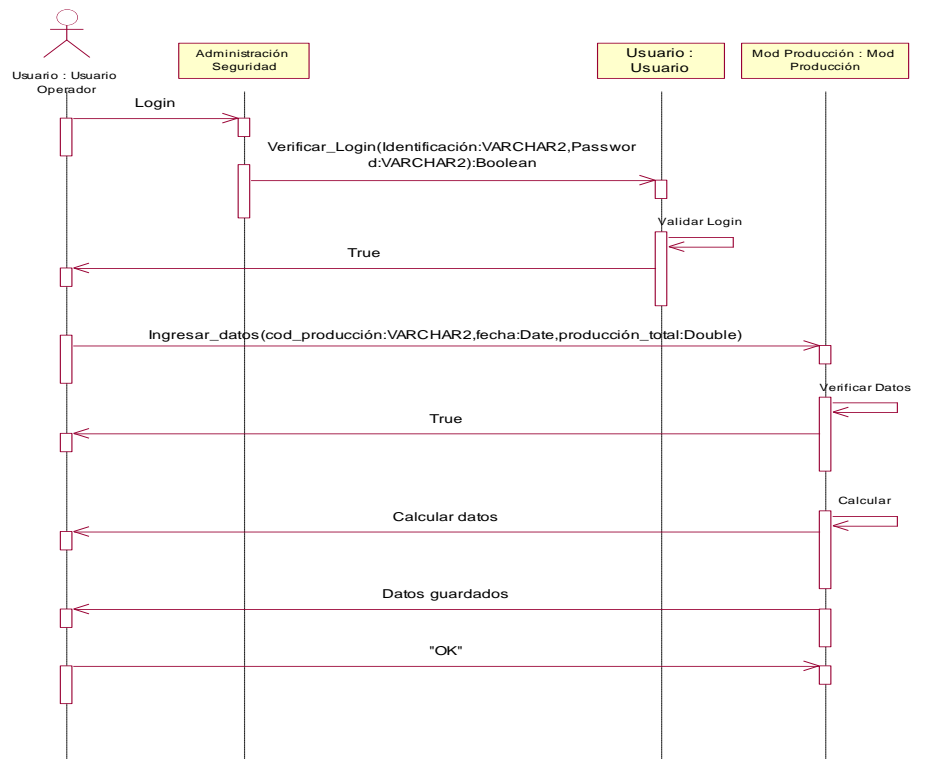


Eliminación

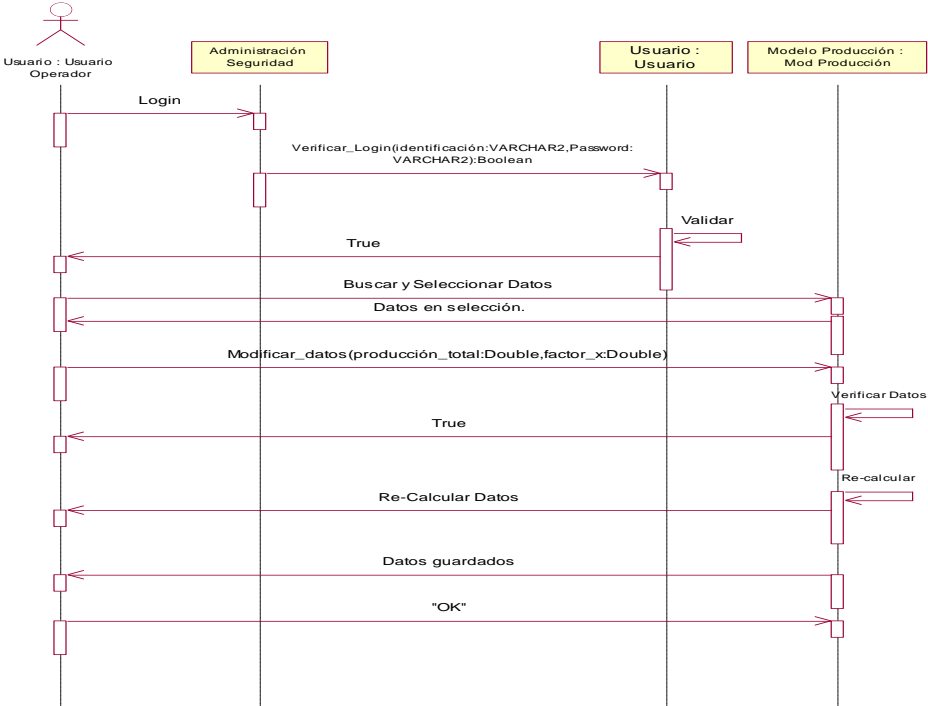


3.2.1.1.9 Producciones Fiscalizadas

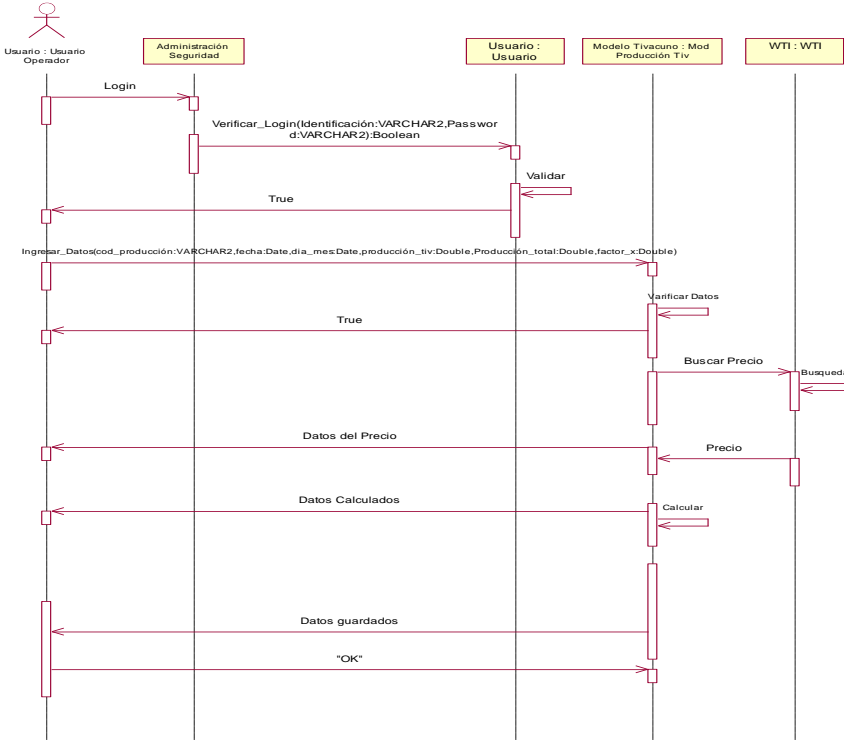
Bloque 16 Ingreso



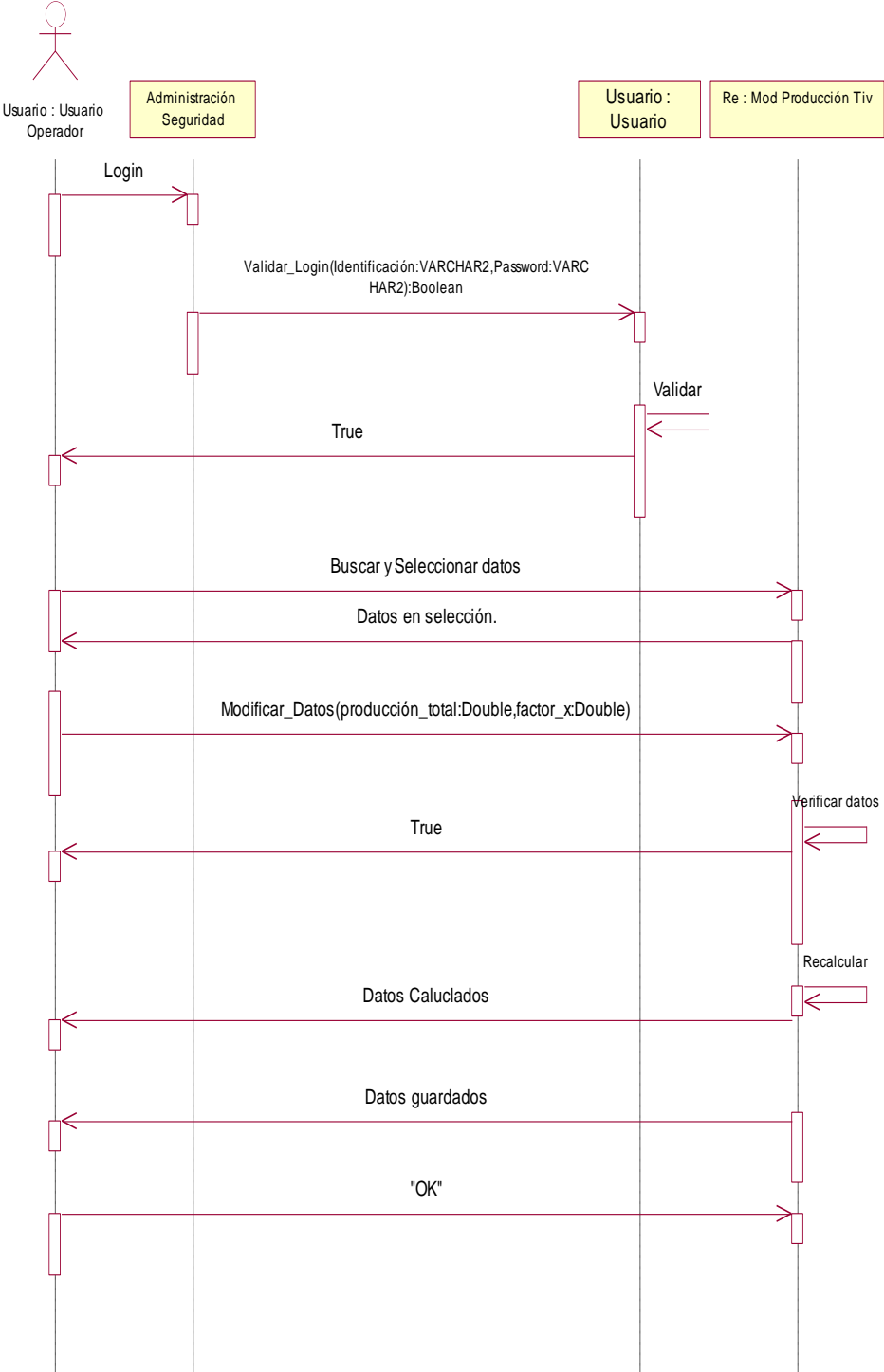
Modificación



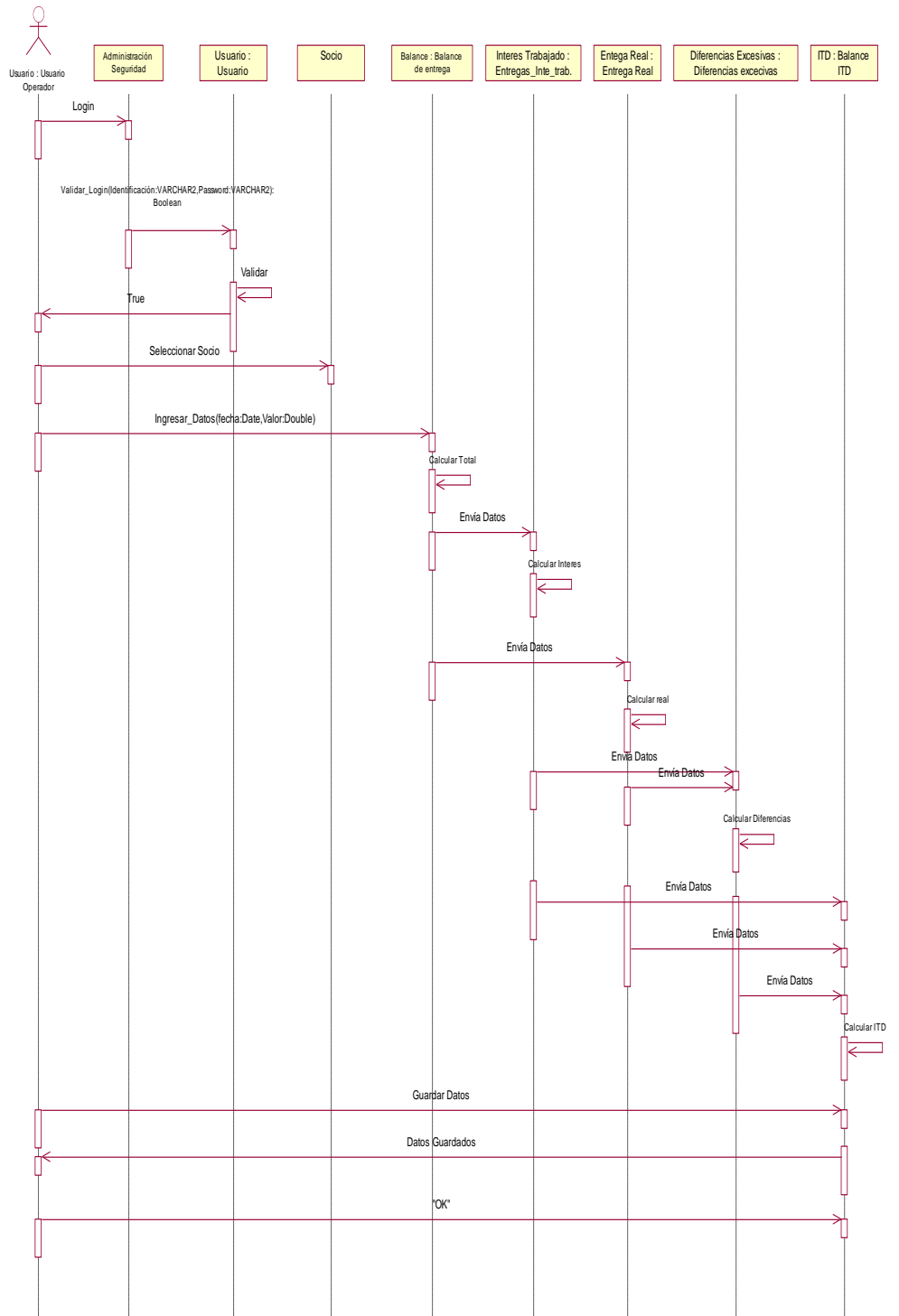
Tivacuno Ingreso



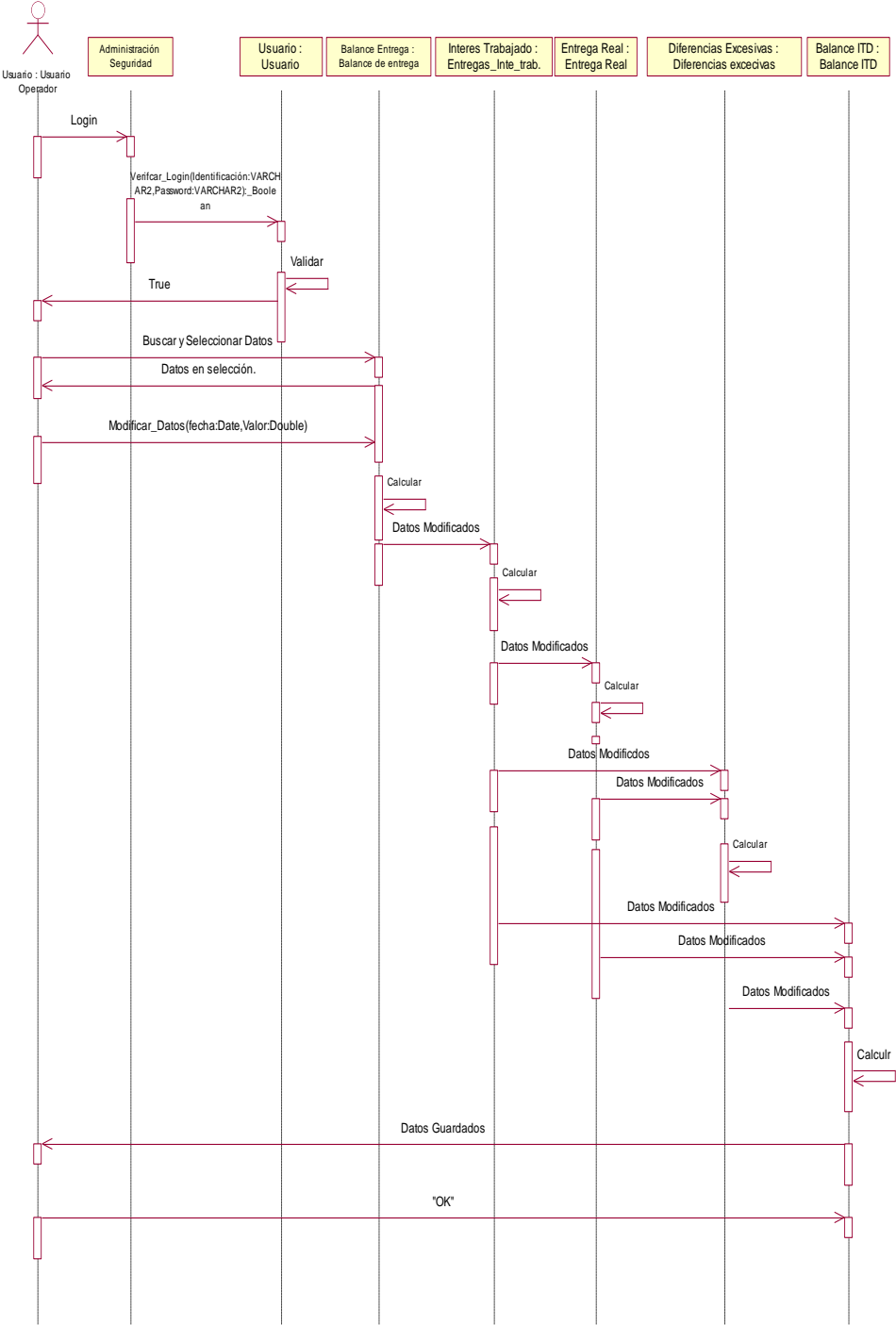
Modificación



3.2.1.1.10 Balance de Entregas Ingreso



Modificación

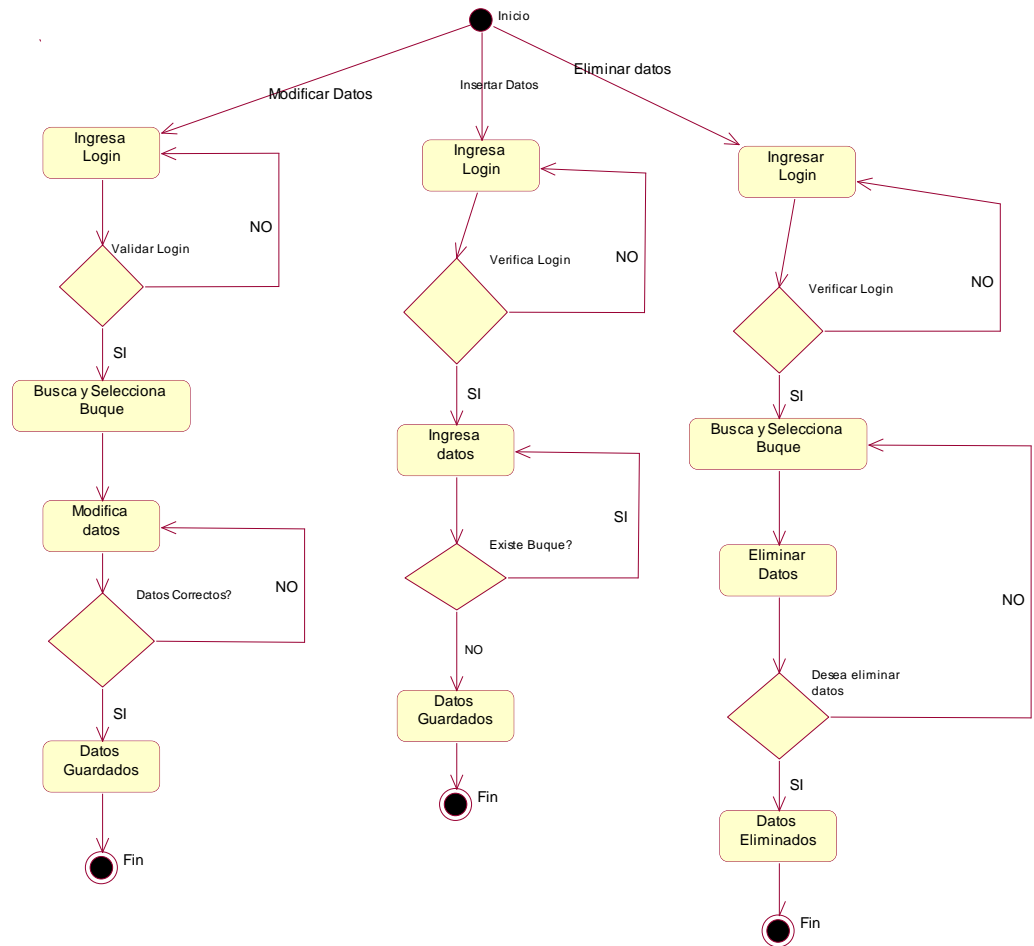


3.3.1 Modelo Funcional

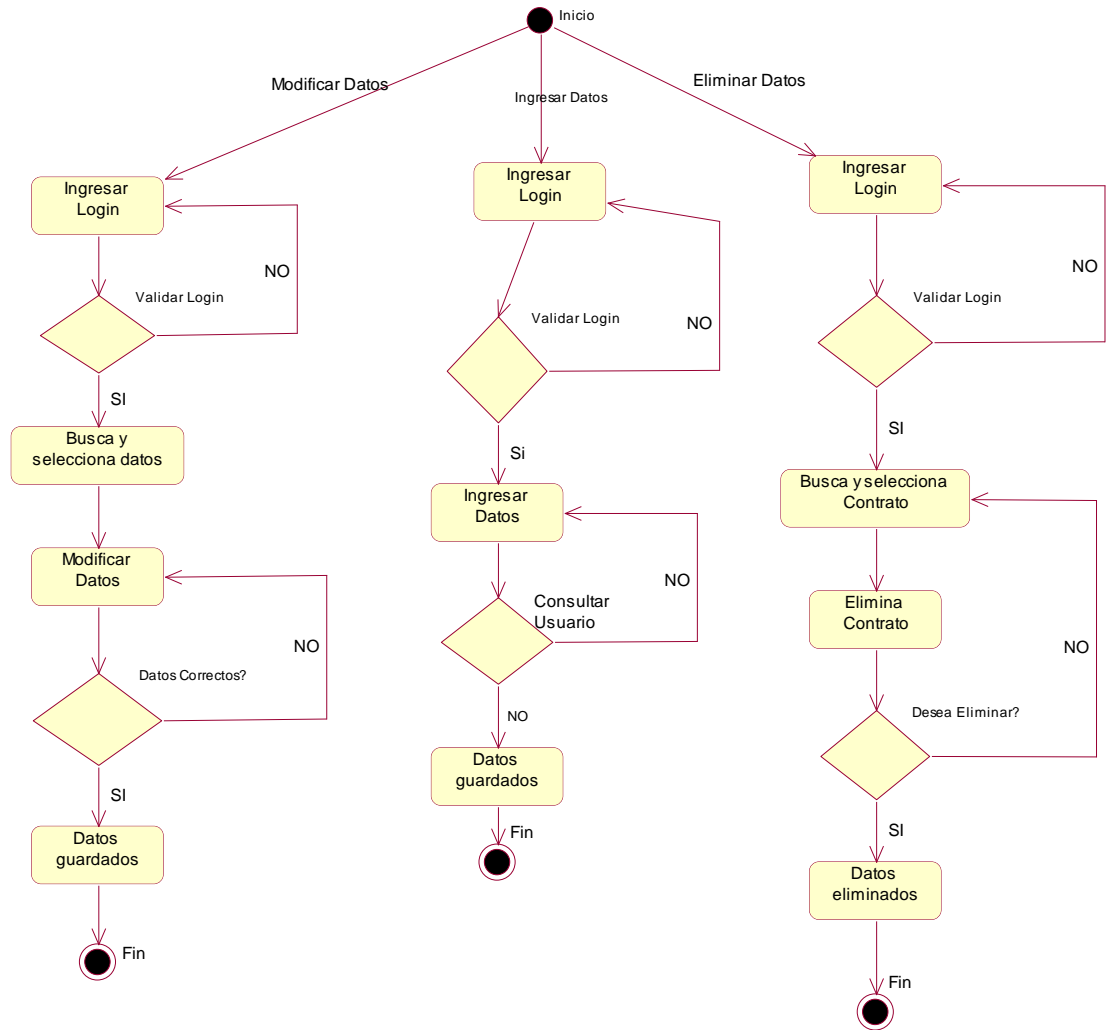
3.3.1.1 Construcción de Diagramas de Flujo de Datos

3.3.1.1.1 Ventas

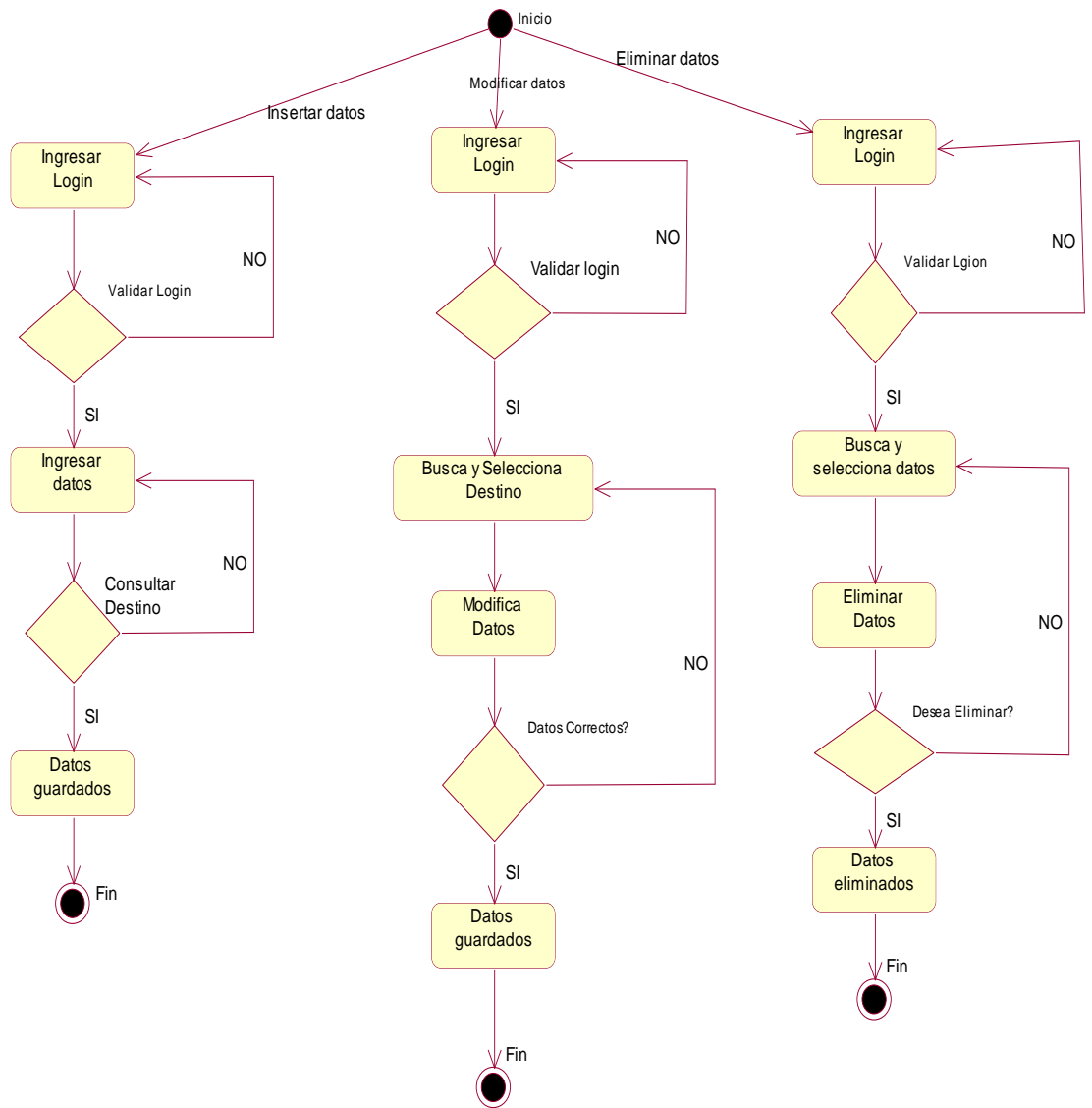
Gestionar Buques



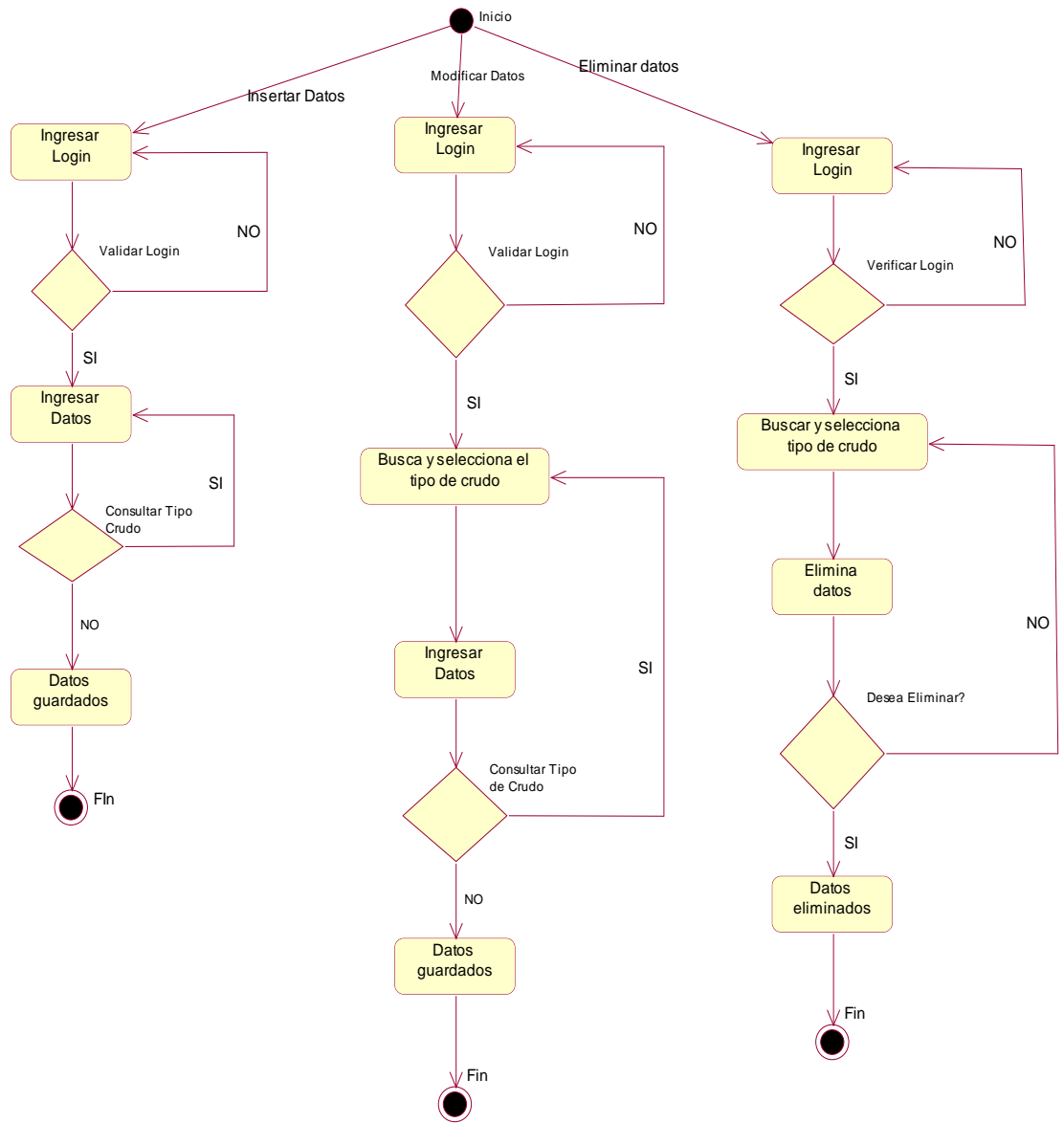
Gestionar Contratos



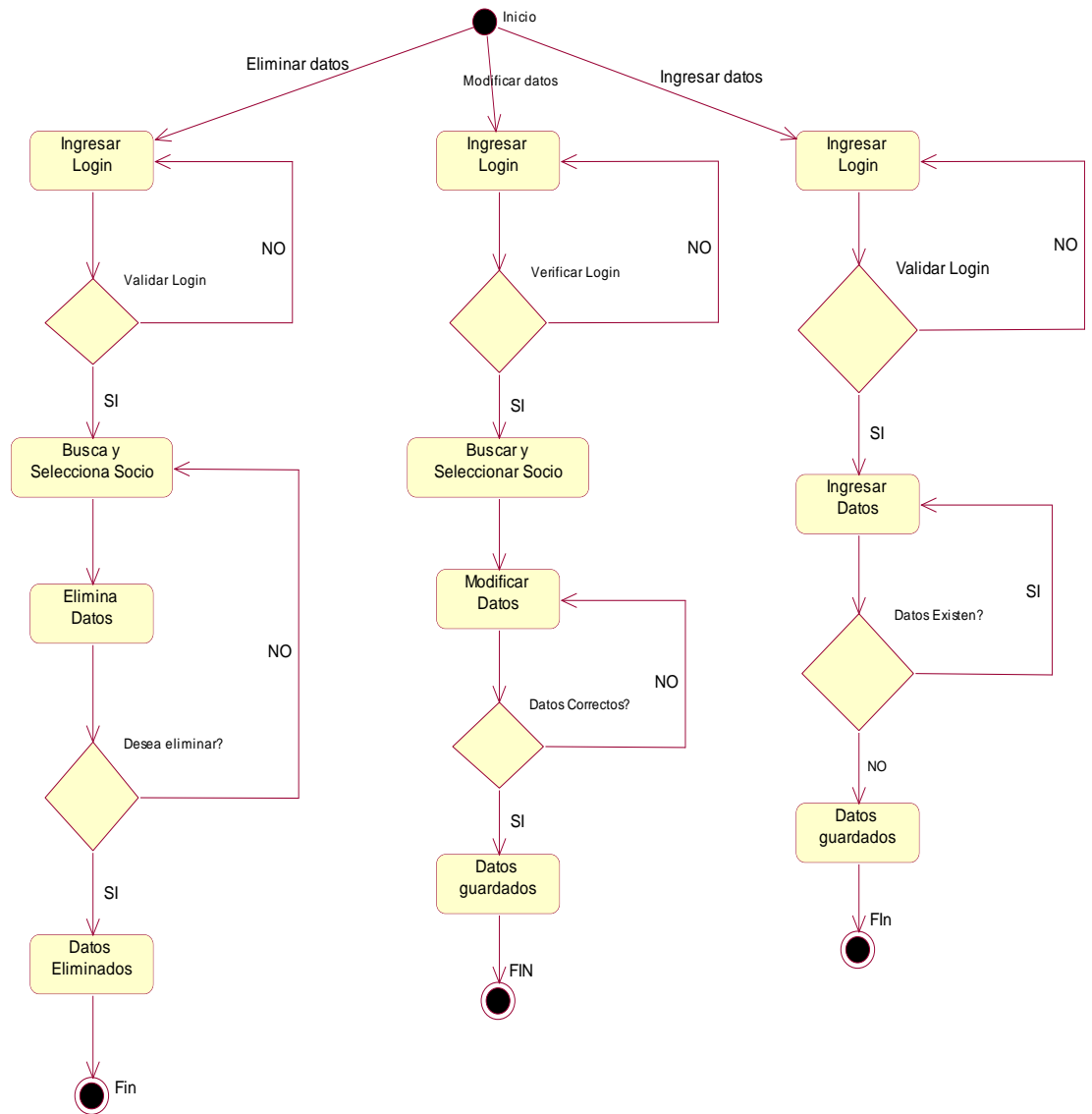
Gestionar Destino



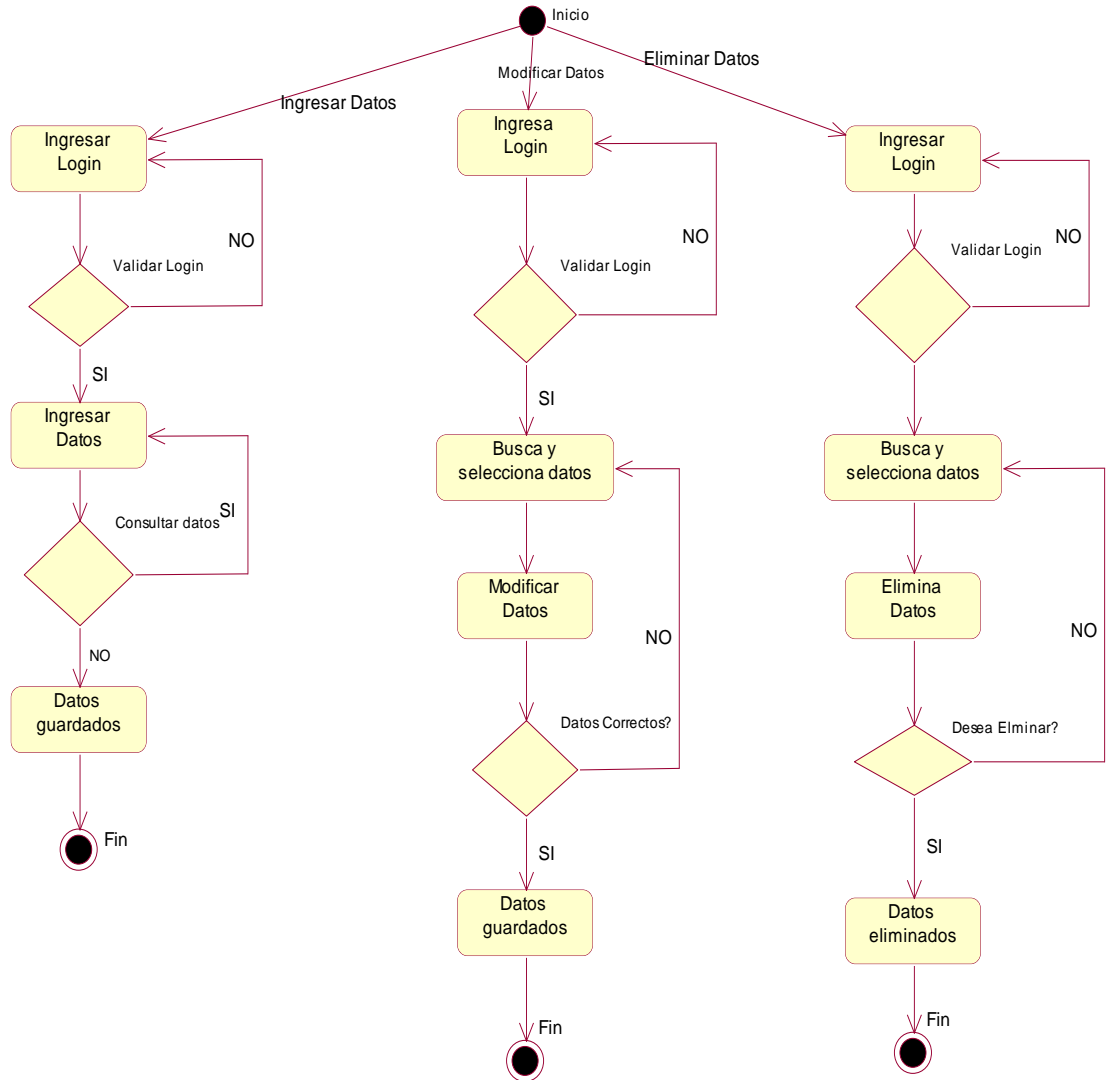
Gestionar Tipo de Crudo



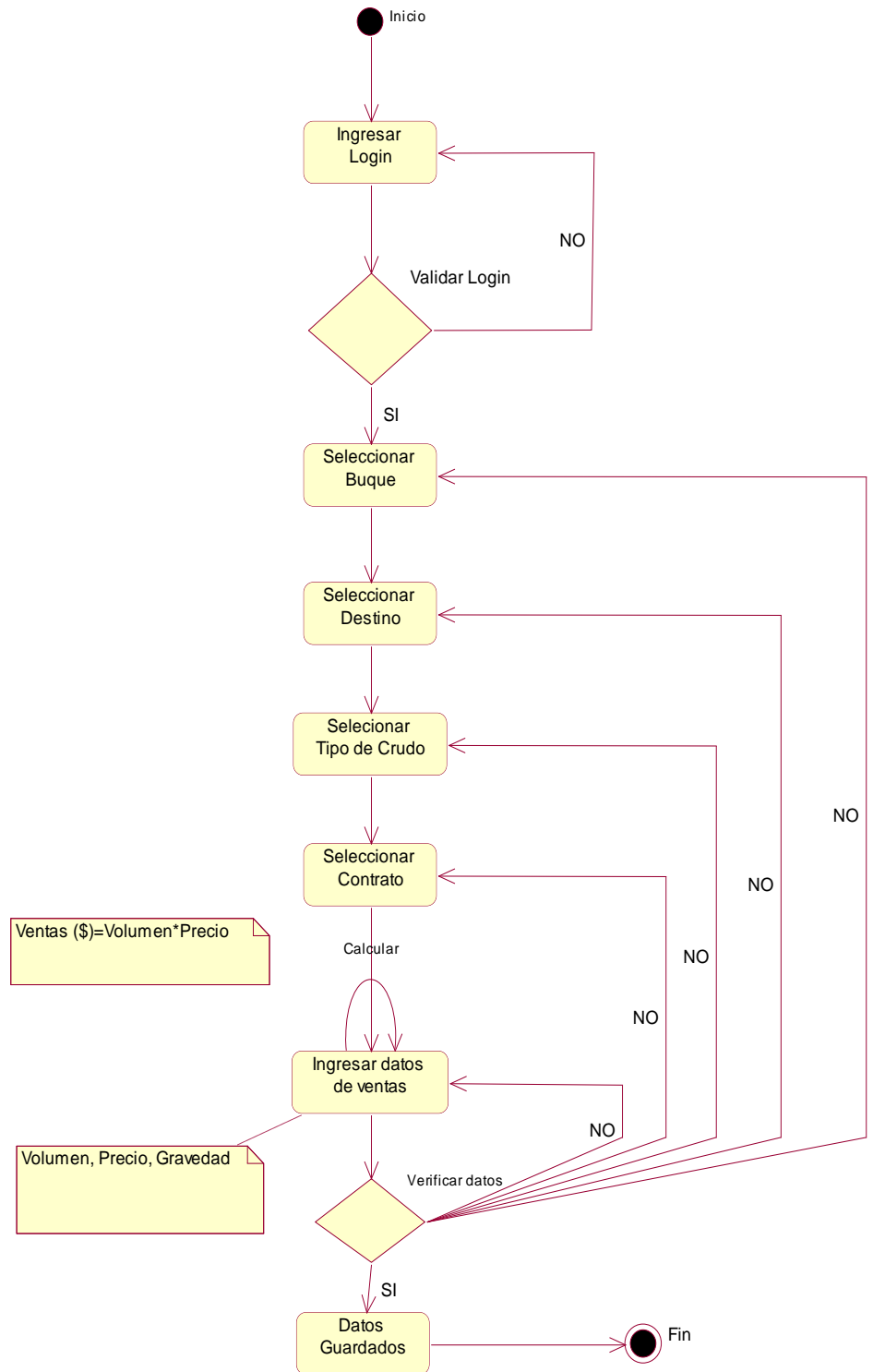
Gestionar Socios



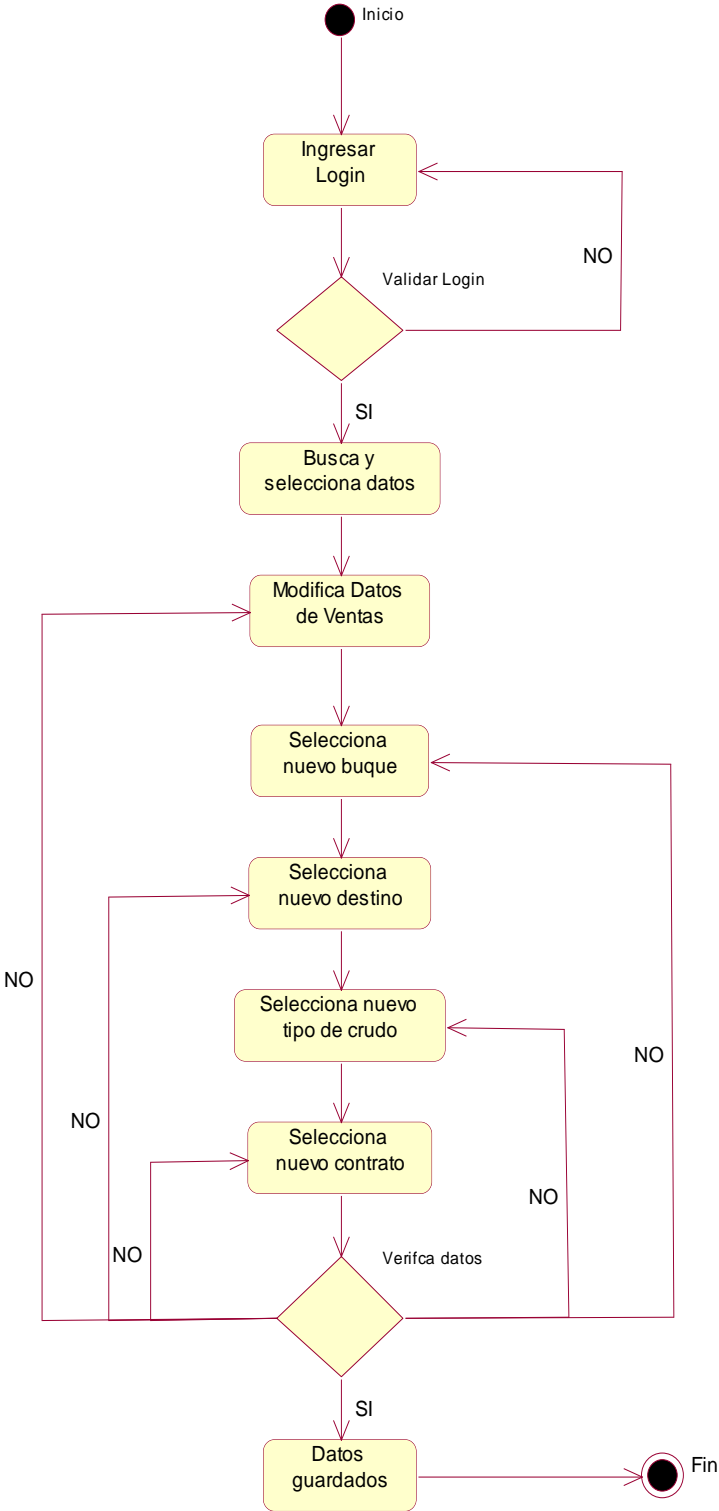
Gestionar Compañías



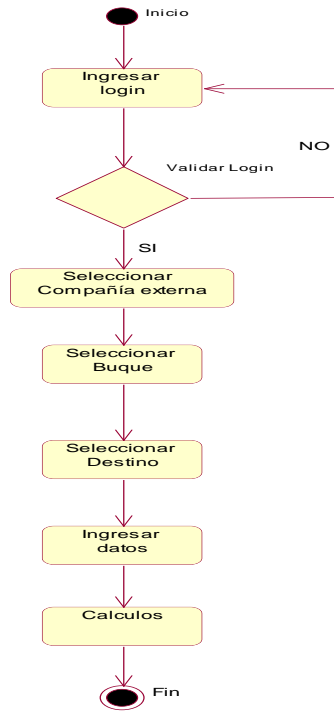
Gestionar Ventas Ingreso



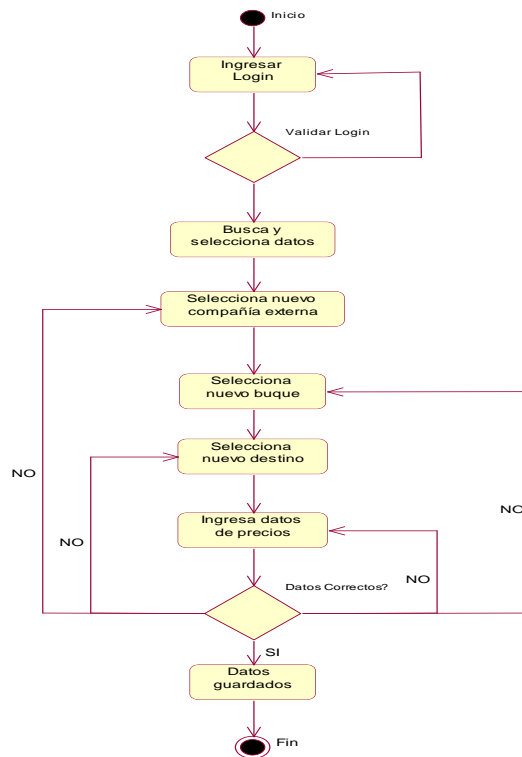
Modificación



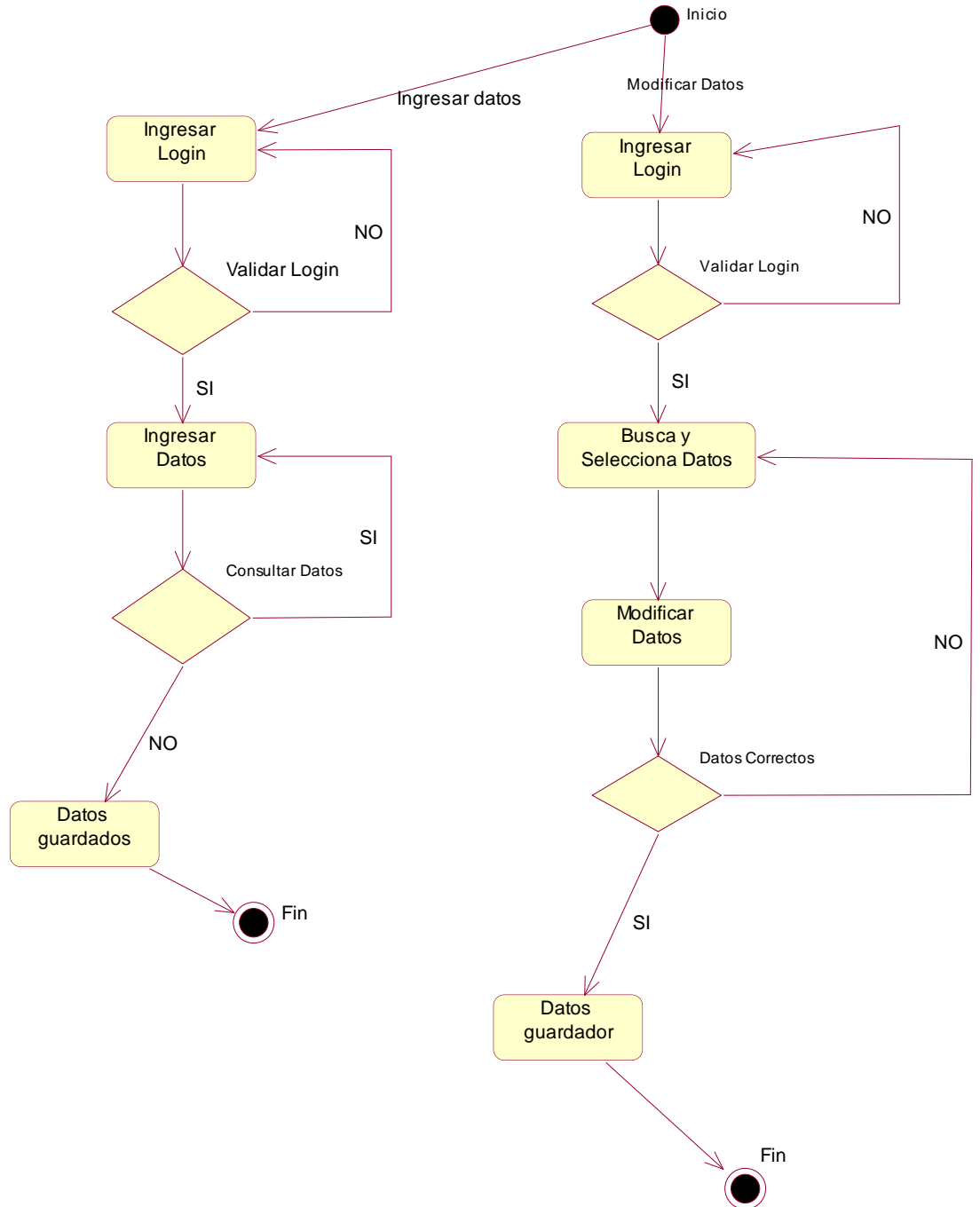
Shippers OCP Ingreso



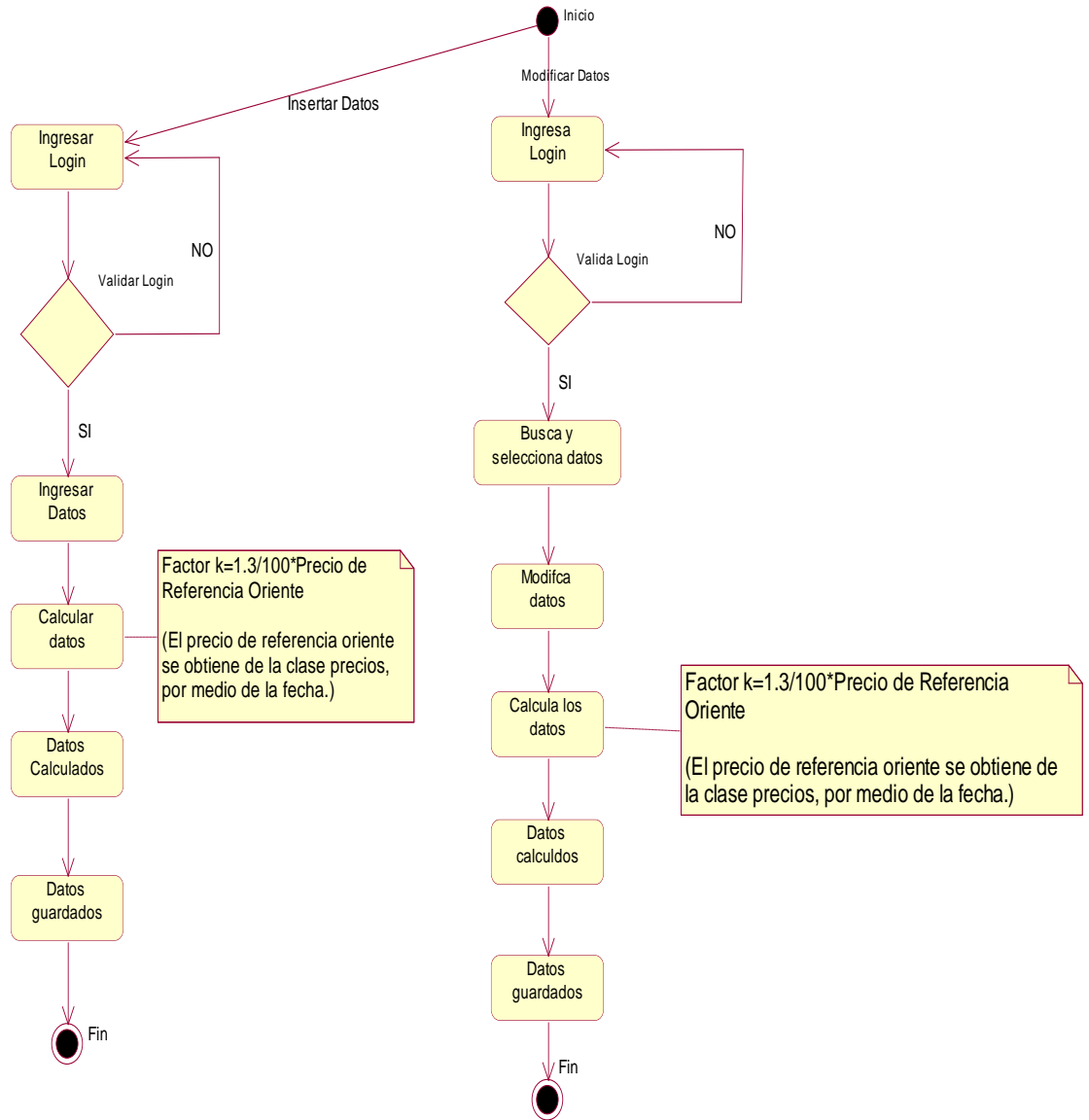
Modificación



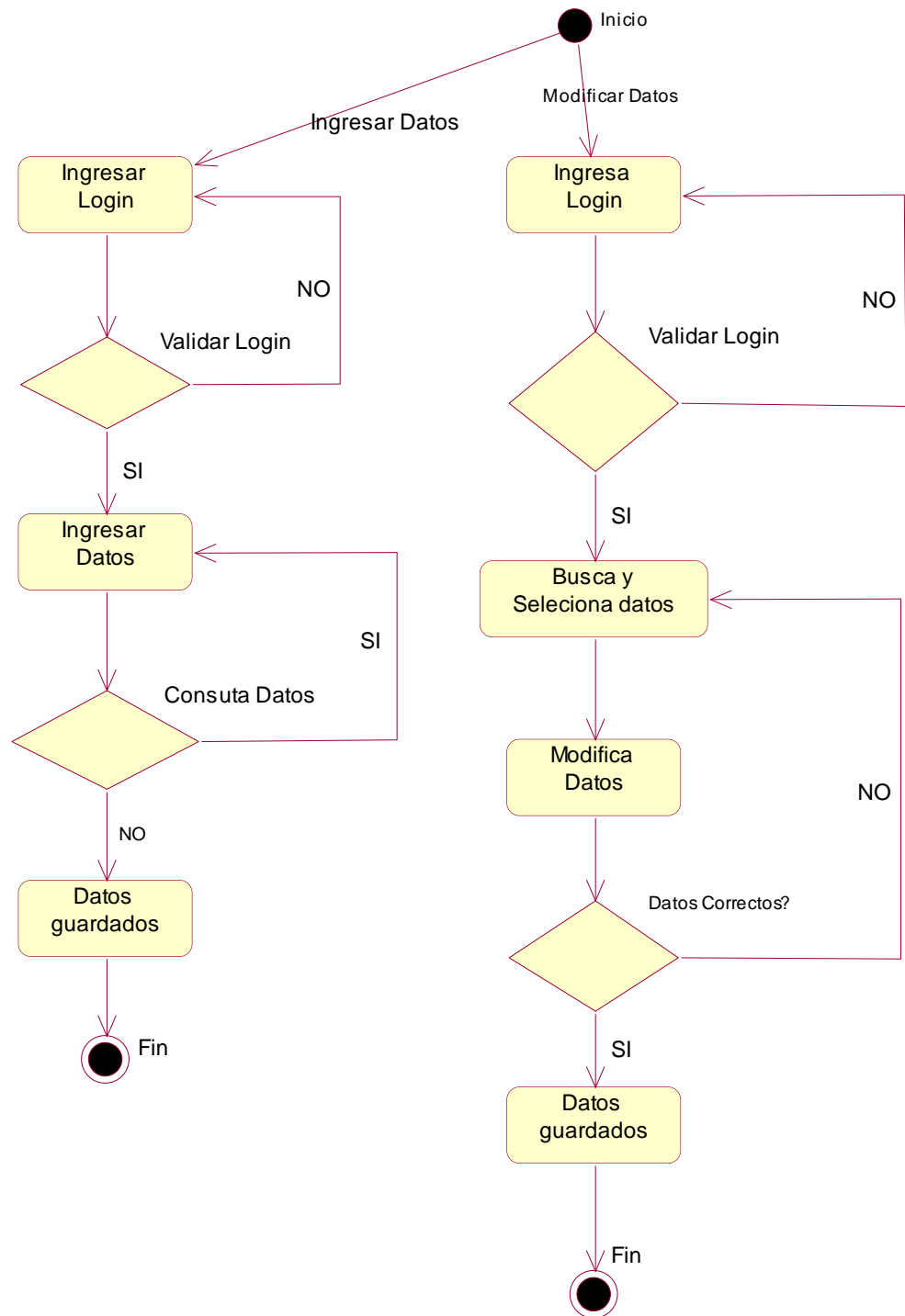
Gestionar Llenado de Línea



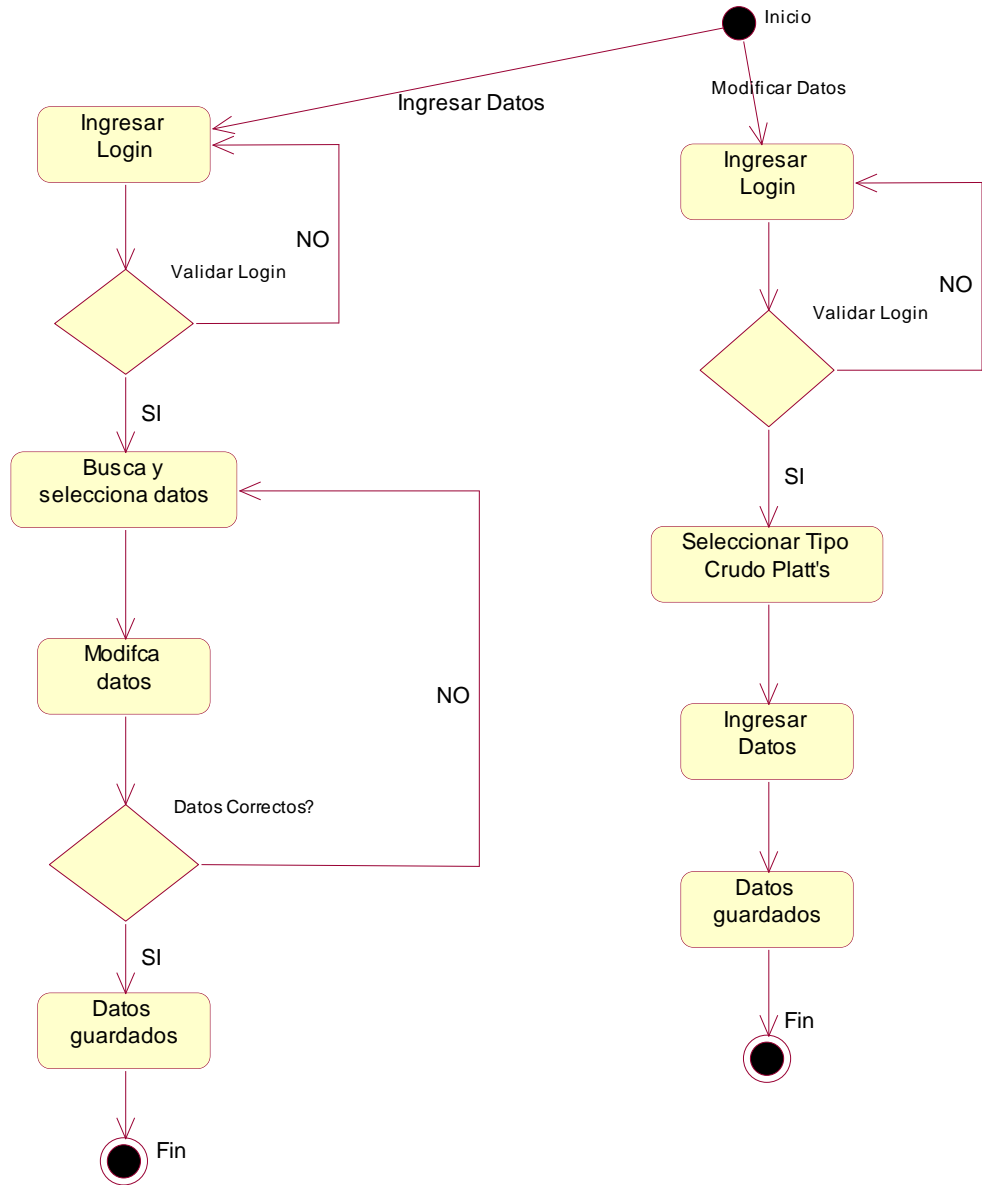
3.3.1.1.2 Precios Gestionar Ajuste API



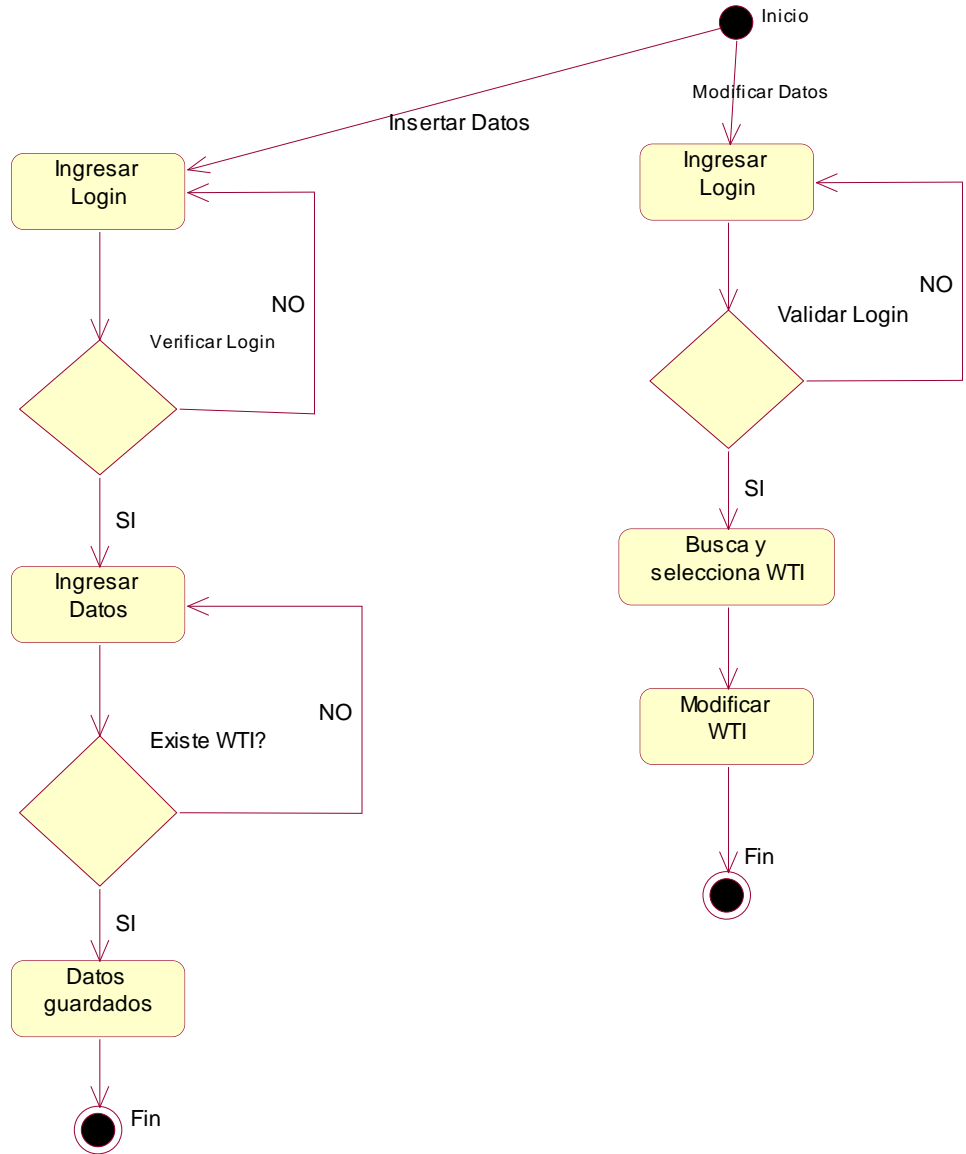
Gestionar Precio de Referencia



Gestionar Precios Platt's

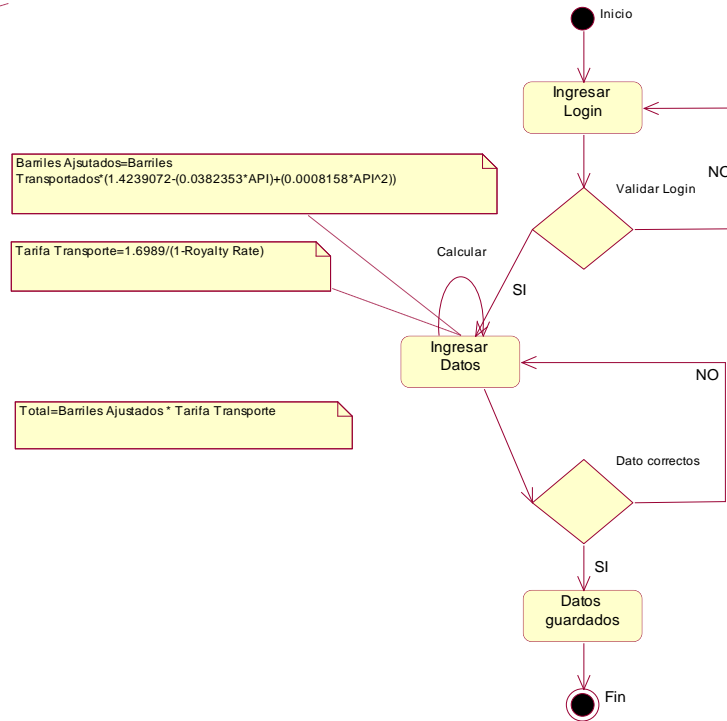


Gestionar Wti

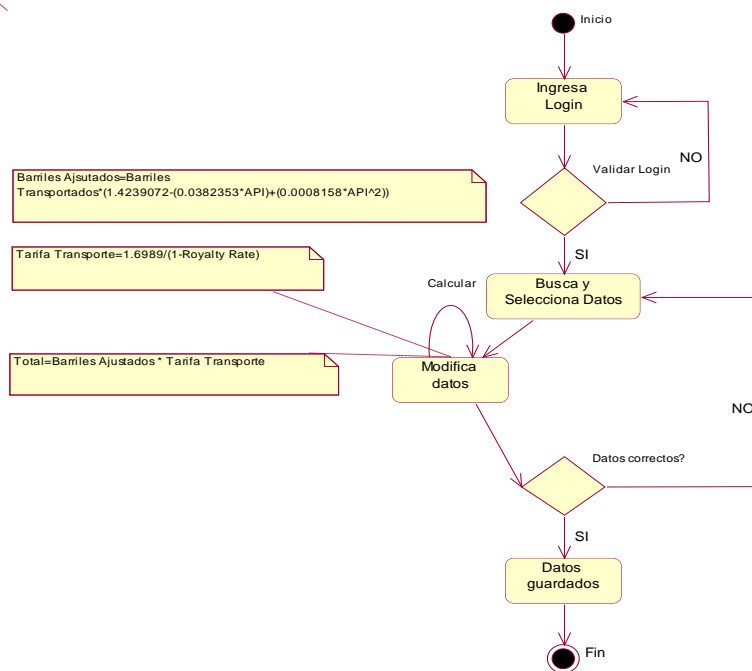


3.3.1.1.3 OCP

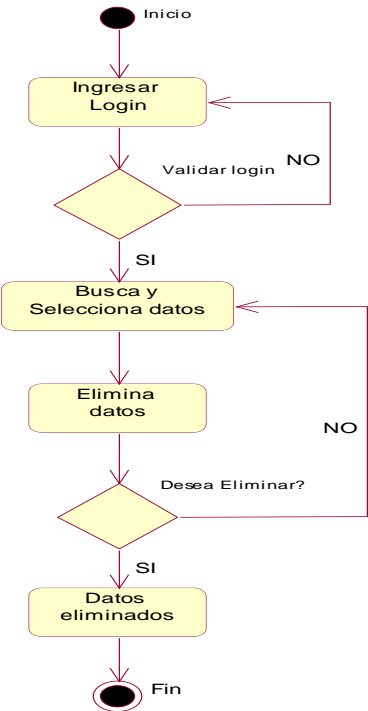
Gestionar Factura Transporte de Crudo Pago por Comisión Ingresar



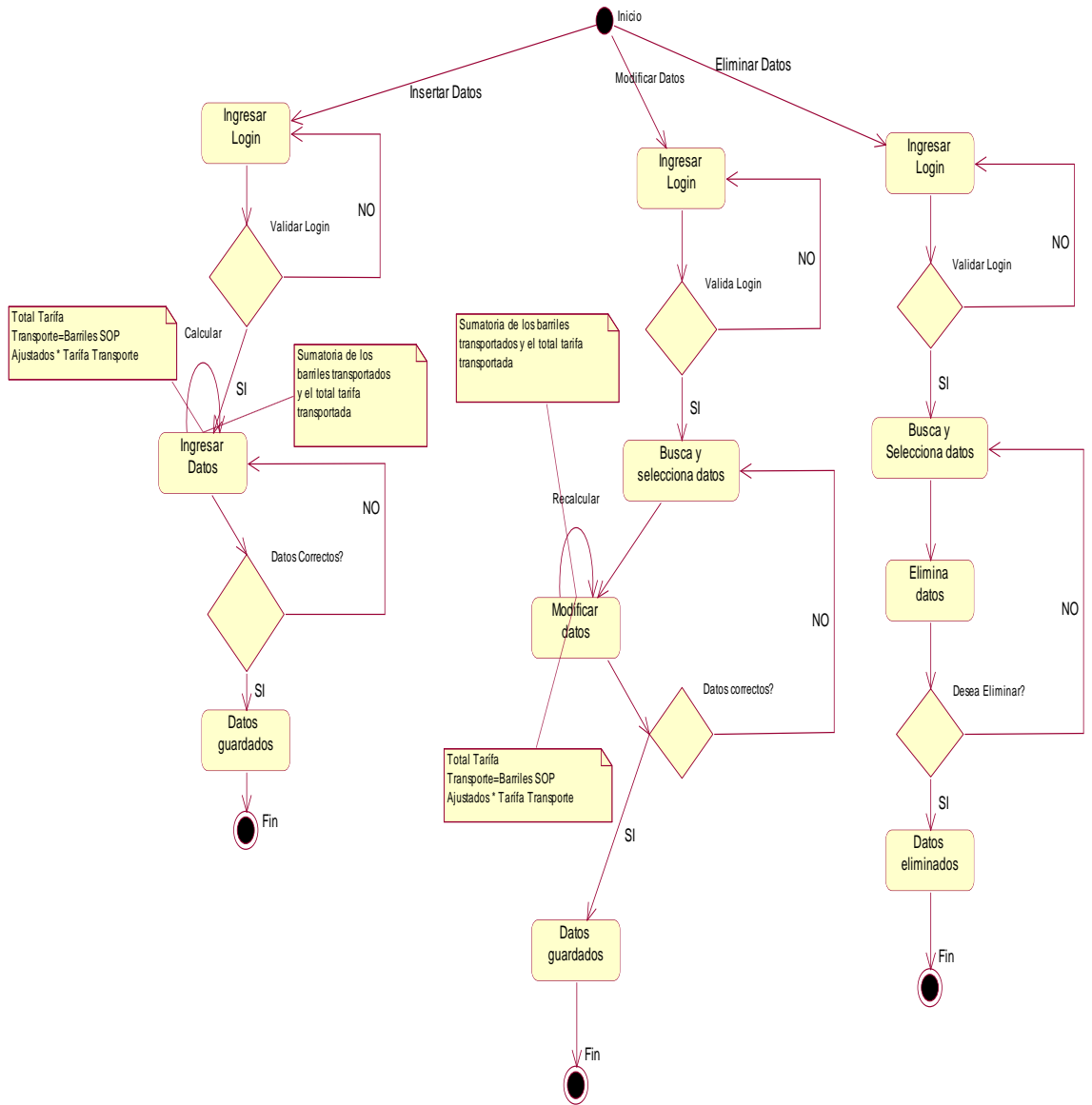
Modificar



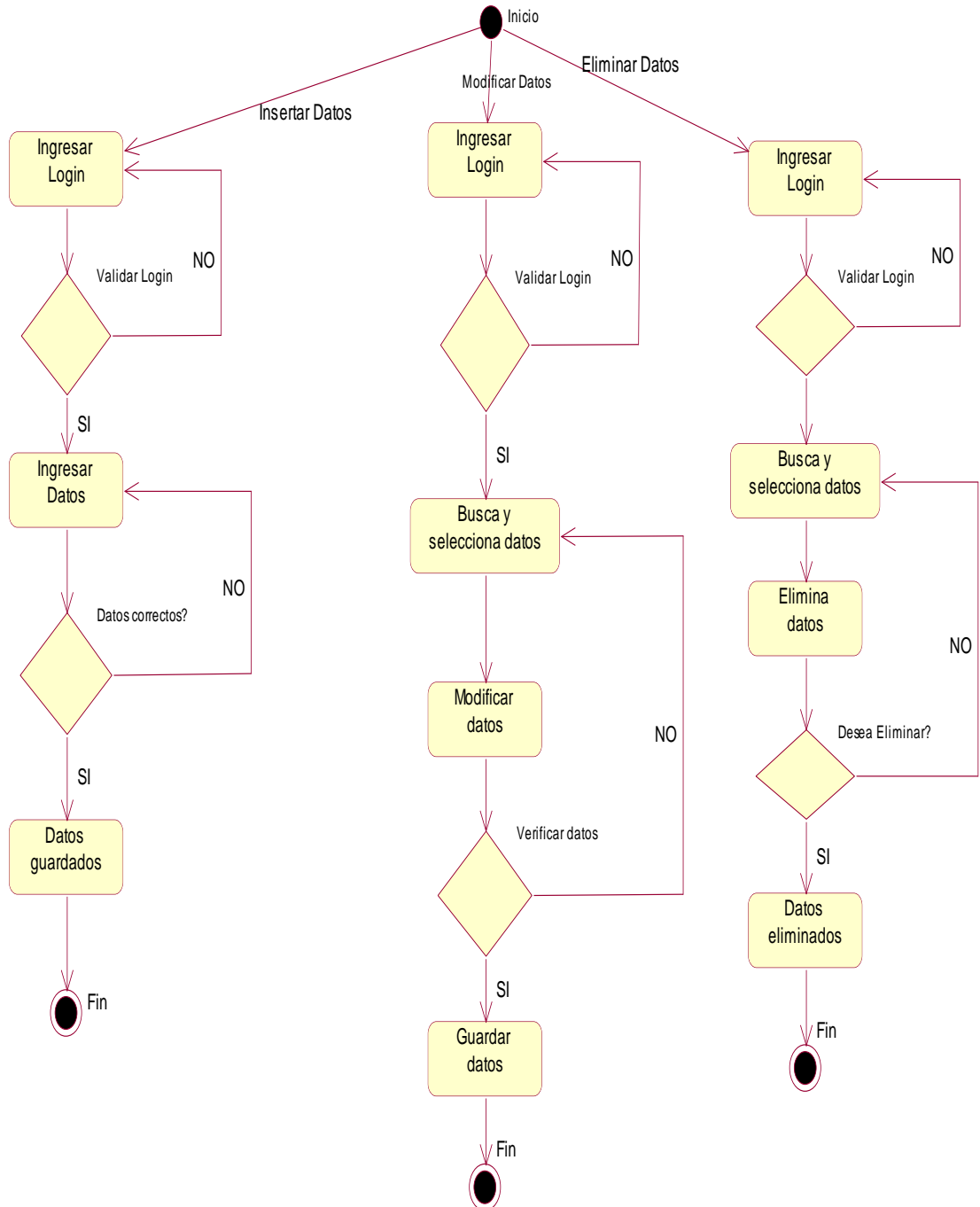
Eliminar



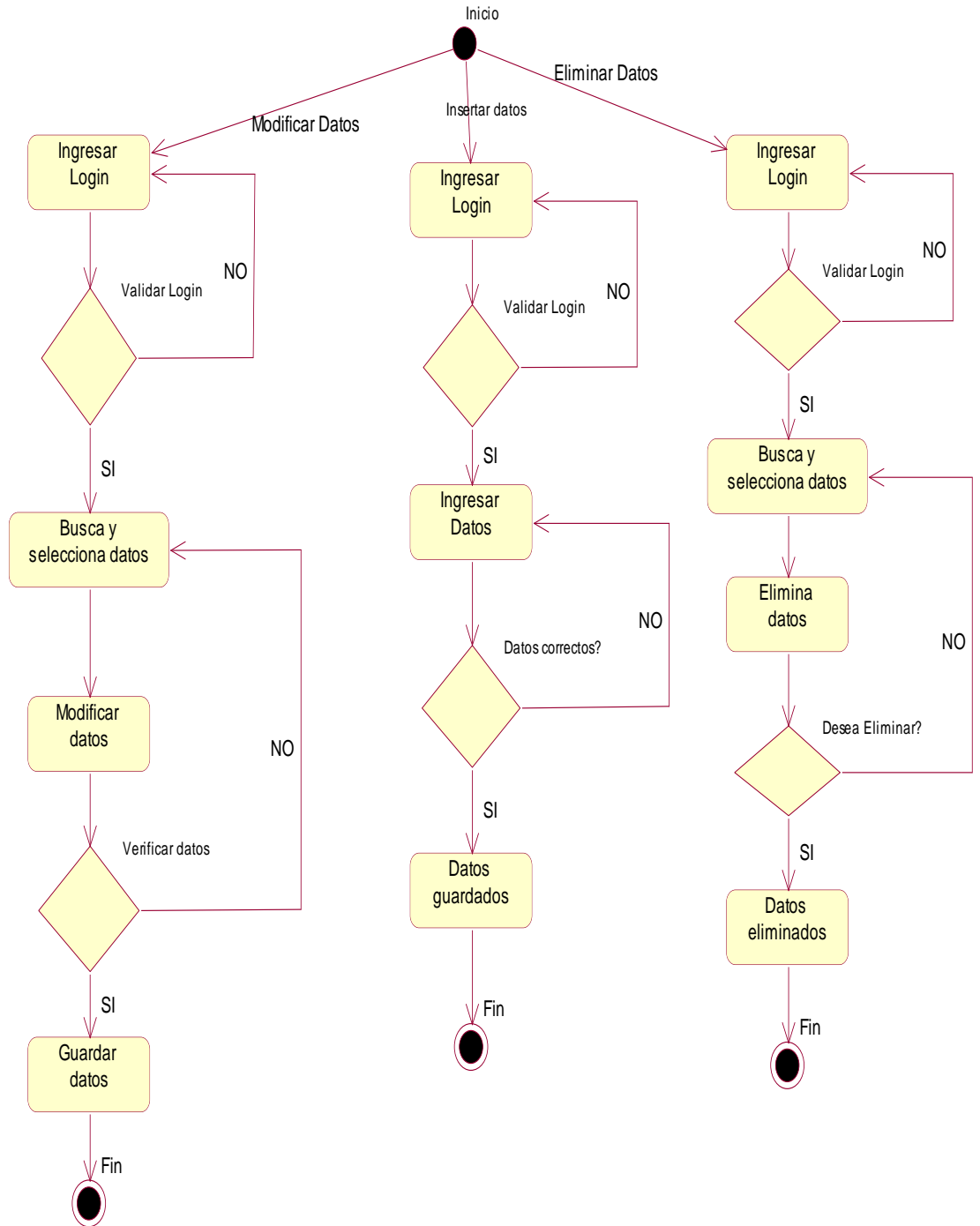
Gestionar Período de Pago



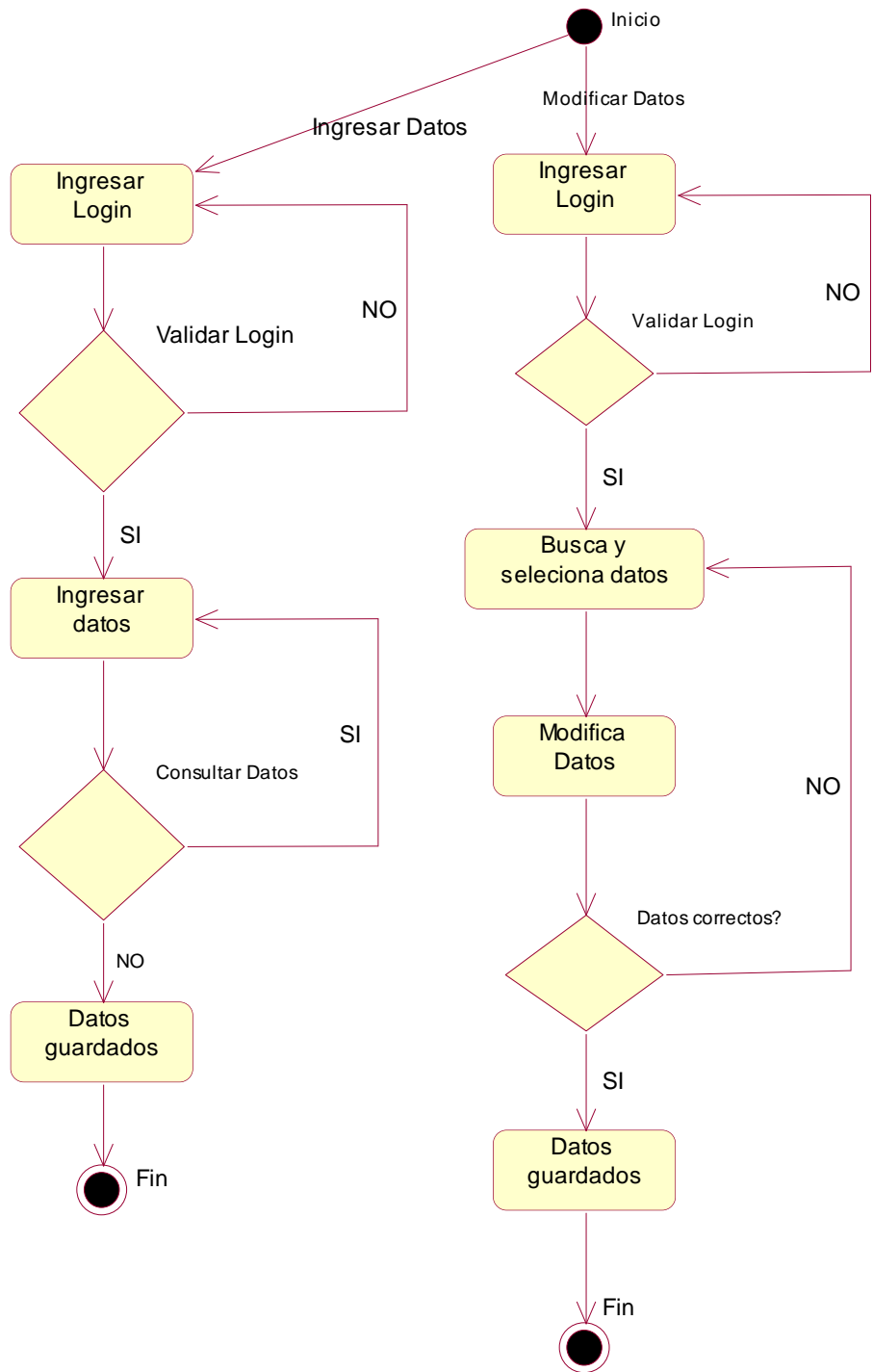
Gestionar Reconciliación ISTA



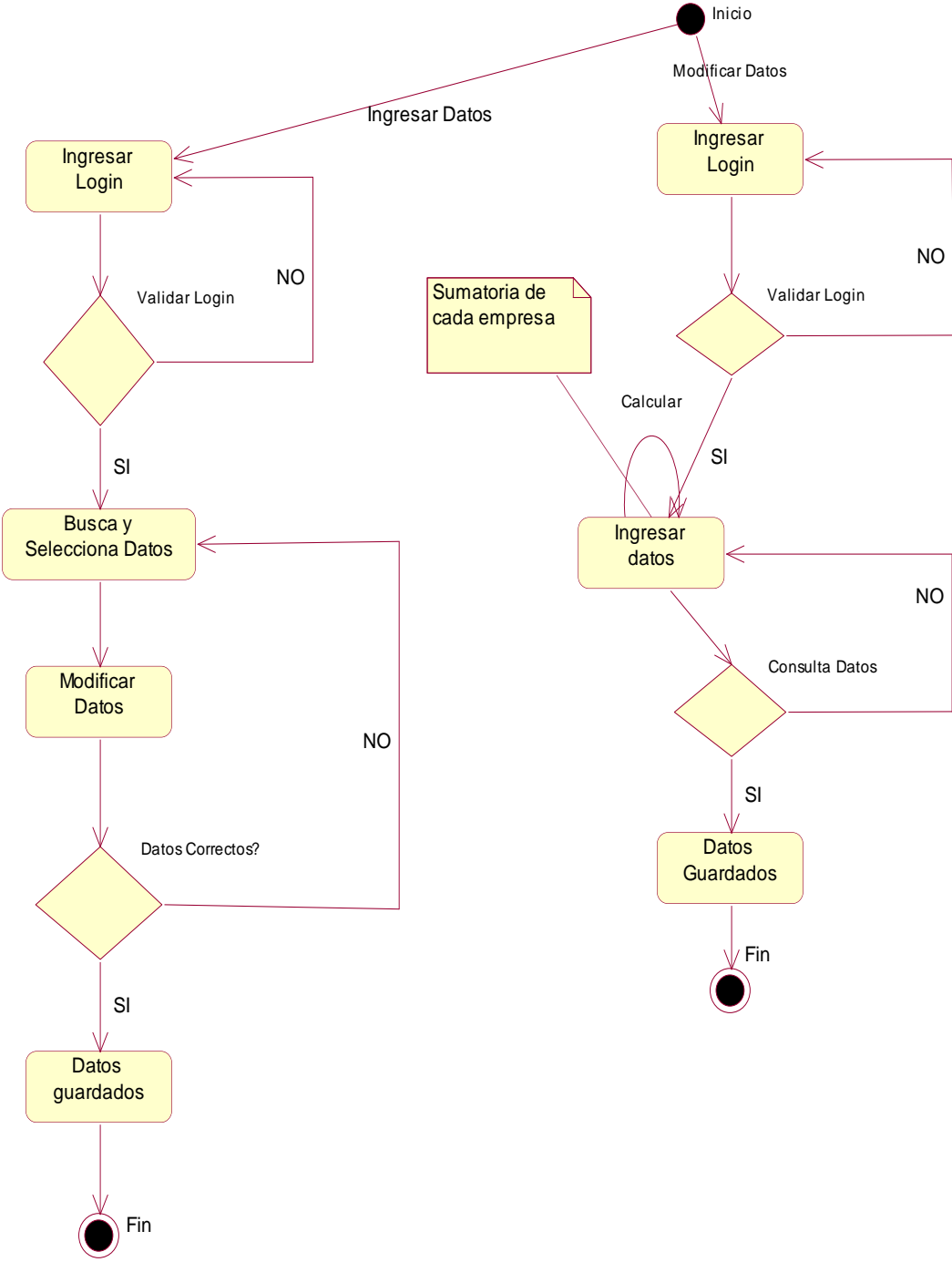
Gestionar Reembolso Combustible



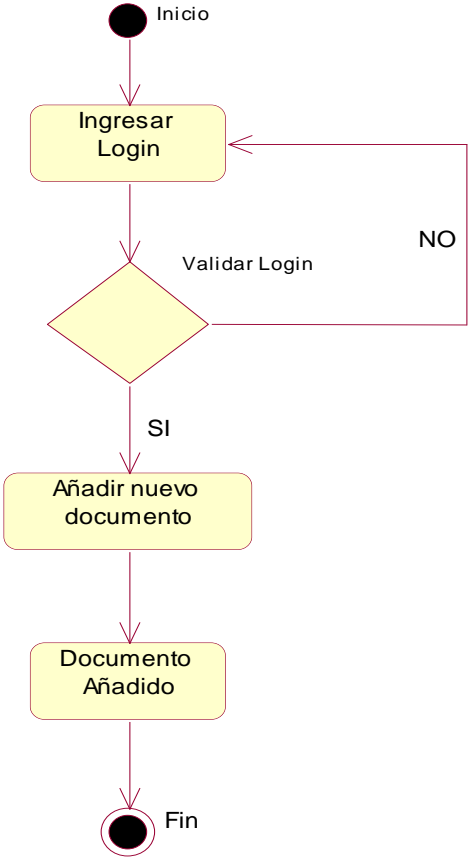
3.3.1.1.4 Descuentos Gestionar Consumo de Combustible



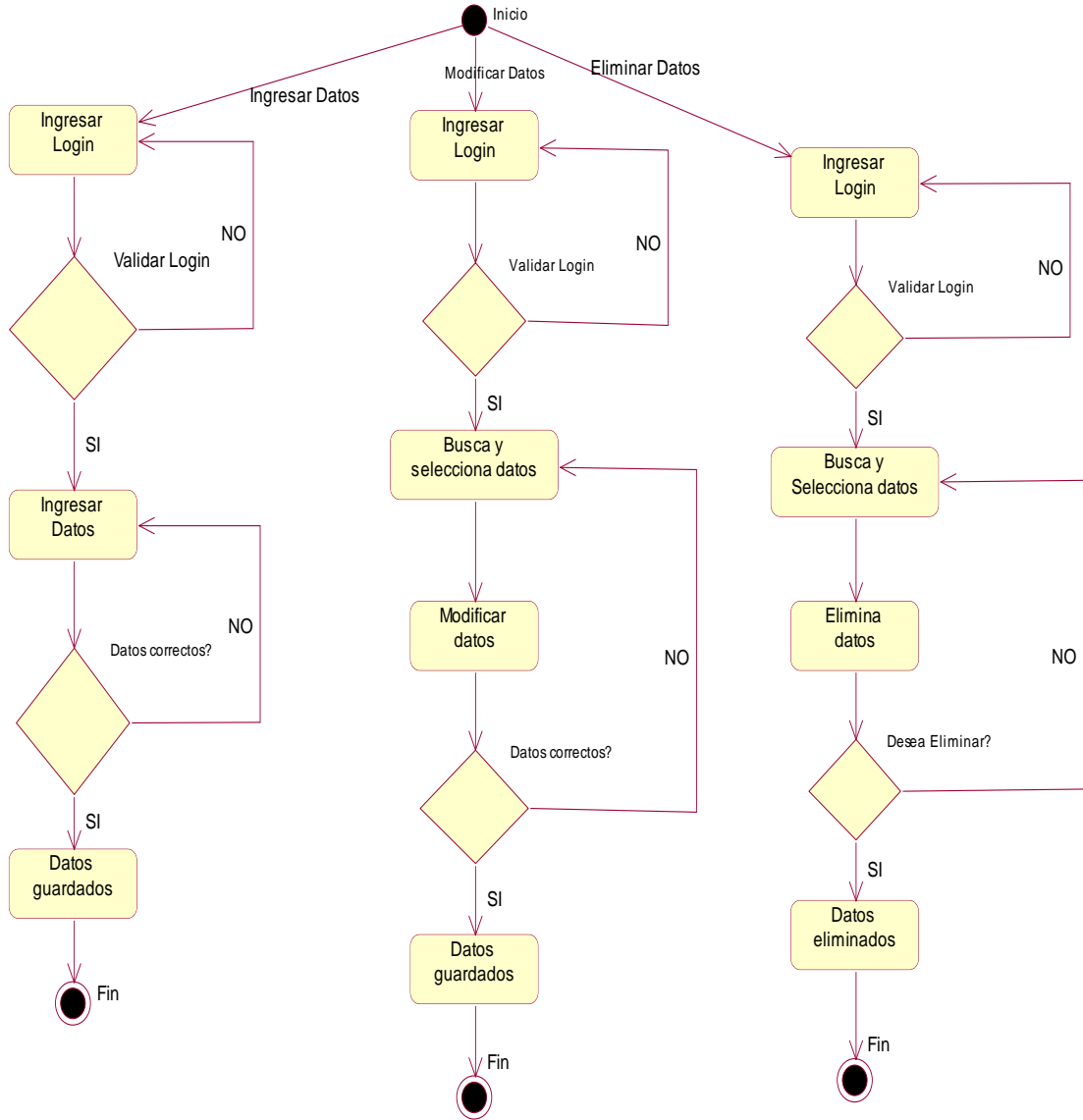
Gestionar Pérdidas y Ganancias



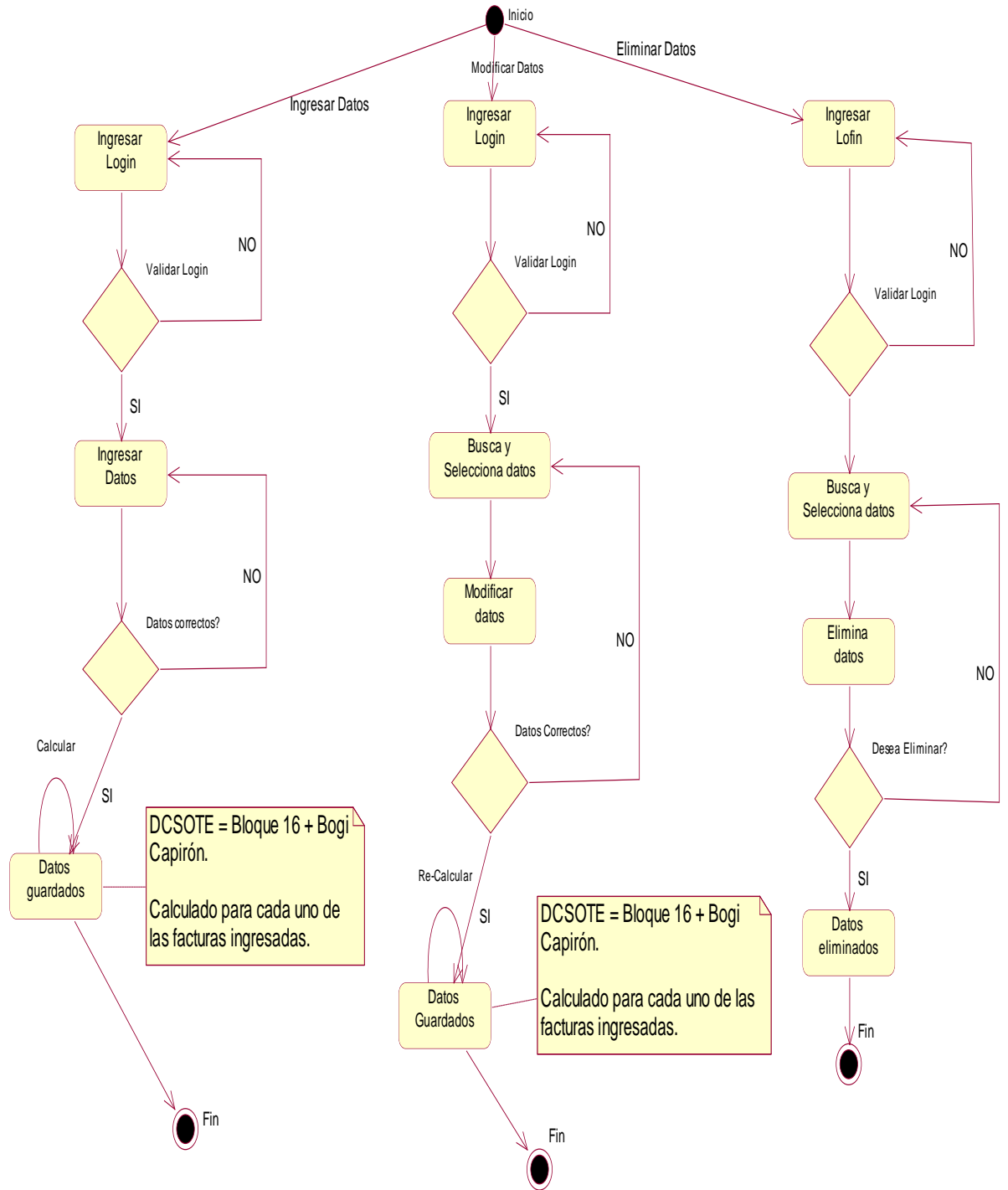
**Programas de Embarque.
Añadir**



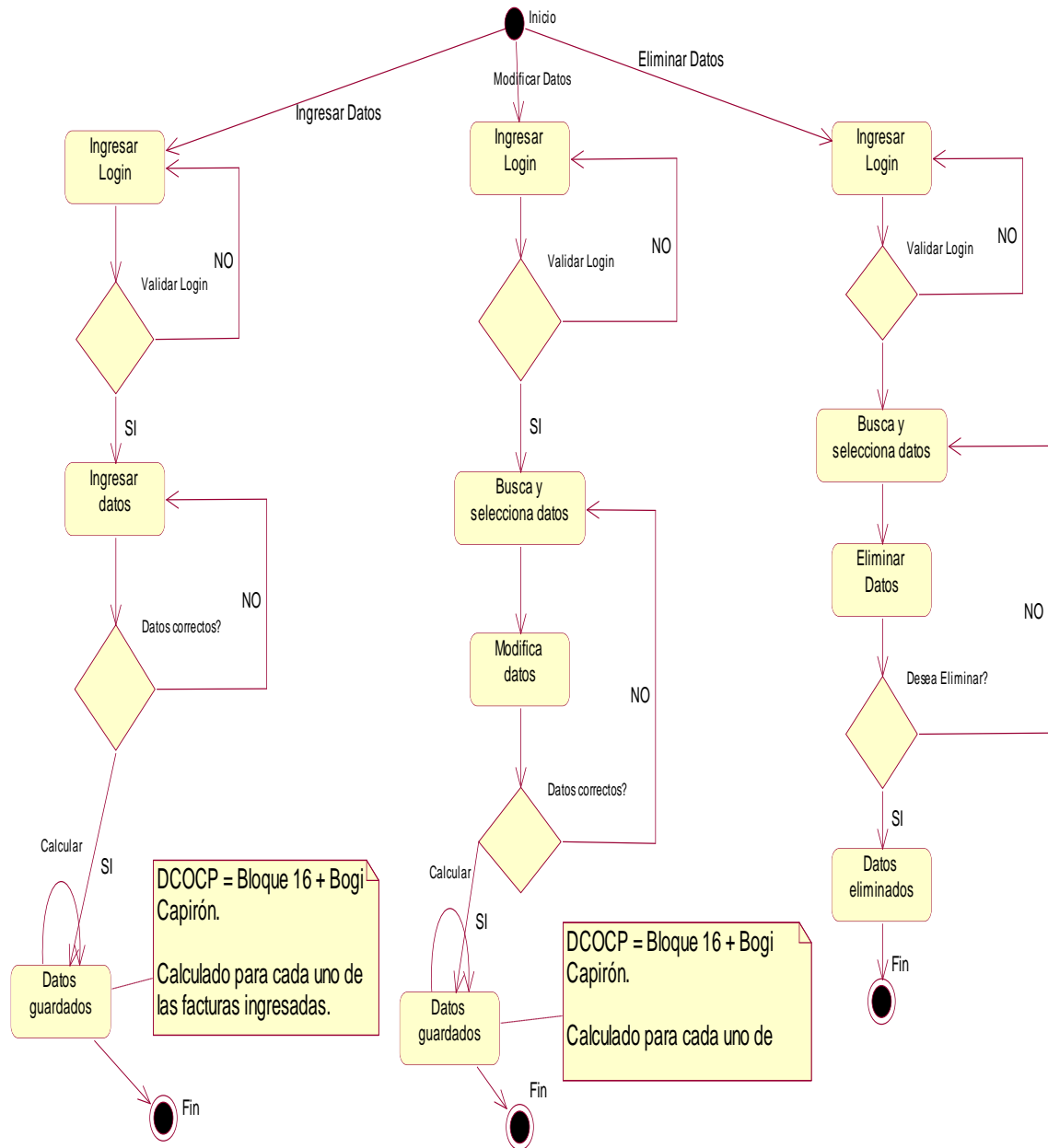
3.3.1.1.5 Facturación SOTE Gestionar Tarifa de Transporte



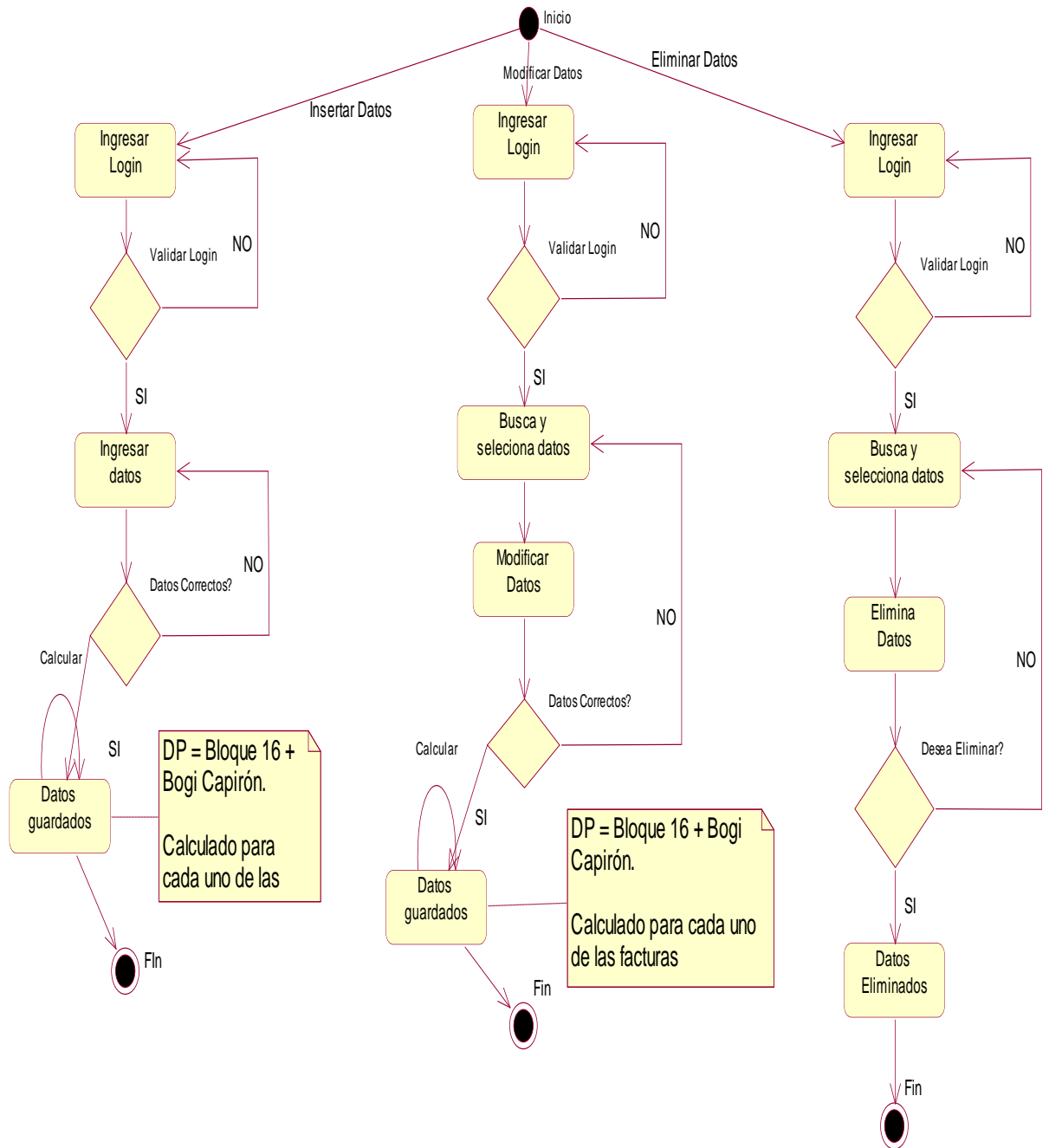
Gestionar Diferencial de Calidad SOTE



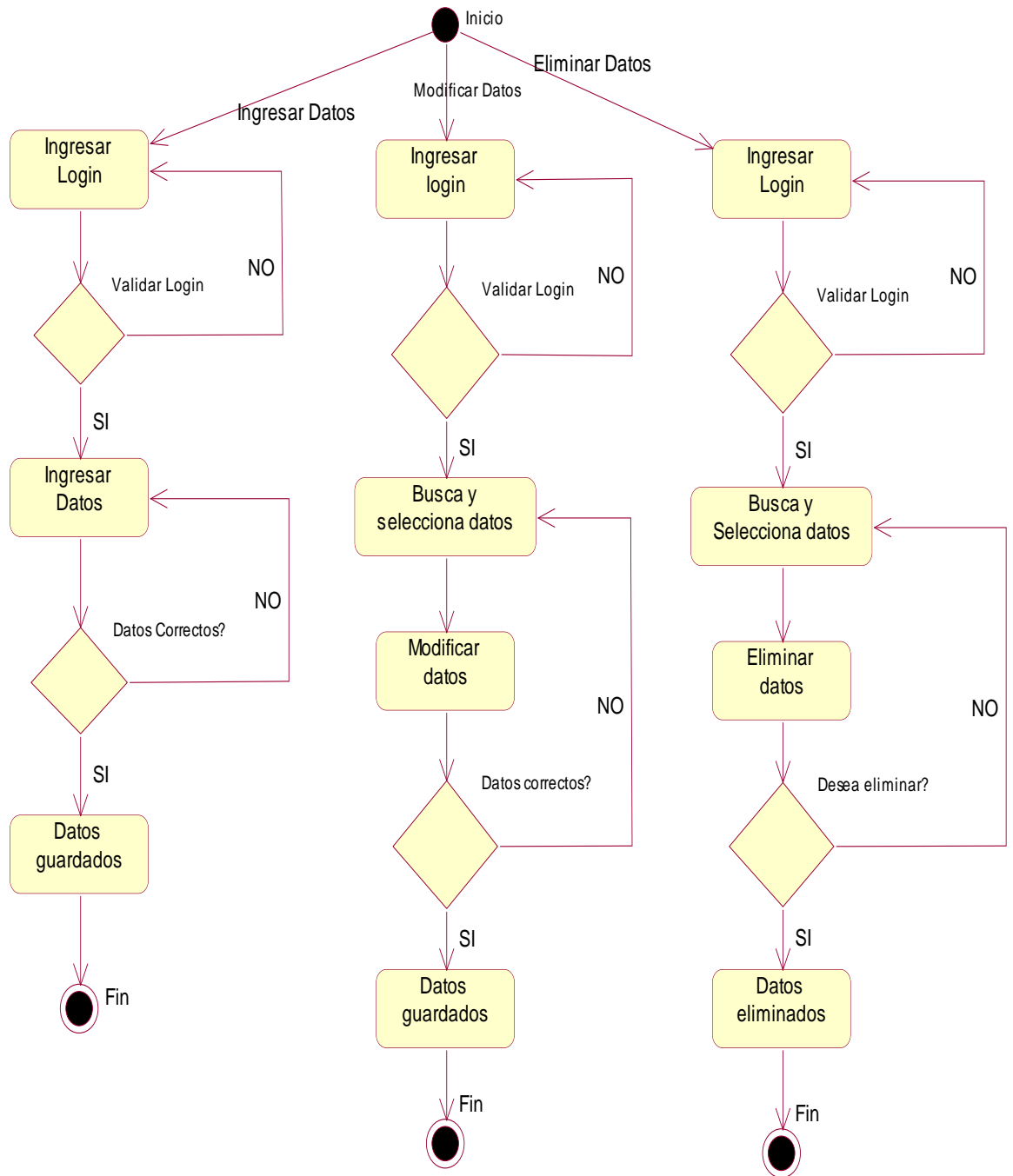
Gestionar Diferencial de Calidad OCP



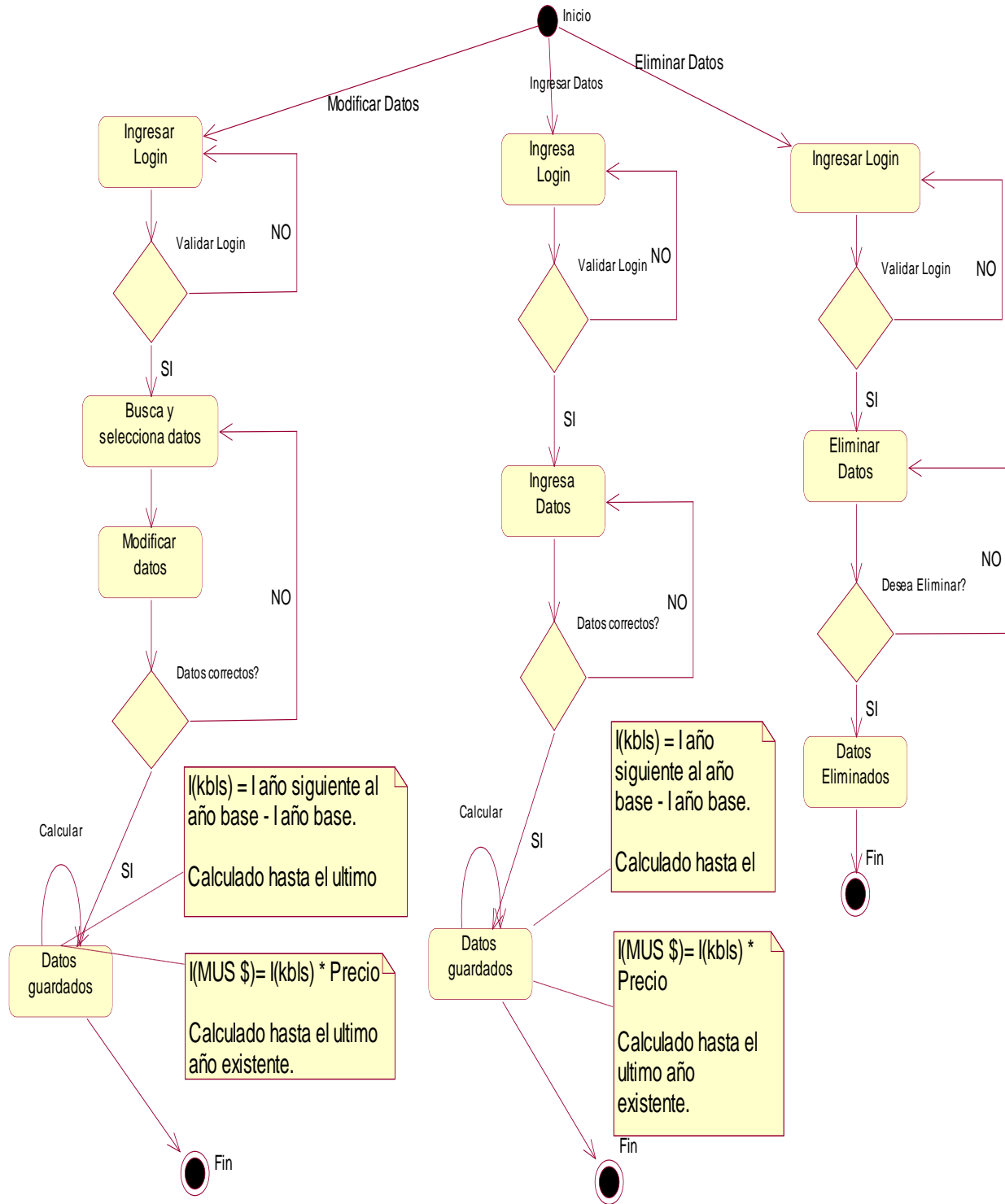
Gestionar Diferencia de Precio



3.3.1.1.6 Gestionar Pagos Tivacuno



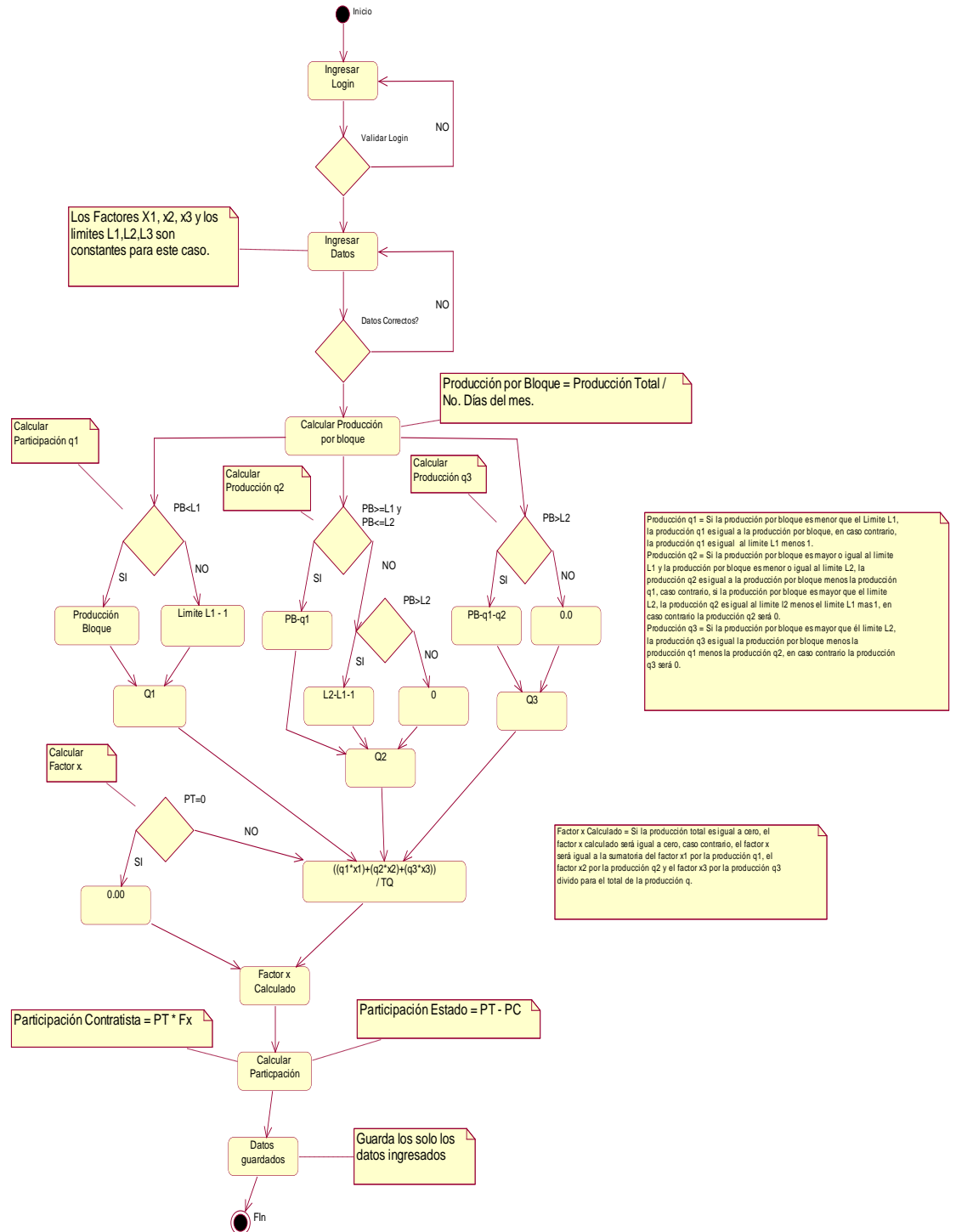
3.3.1.1.7 Gestionar Ineficiencia Petrolera



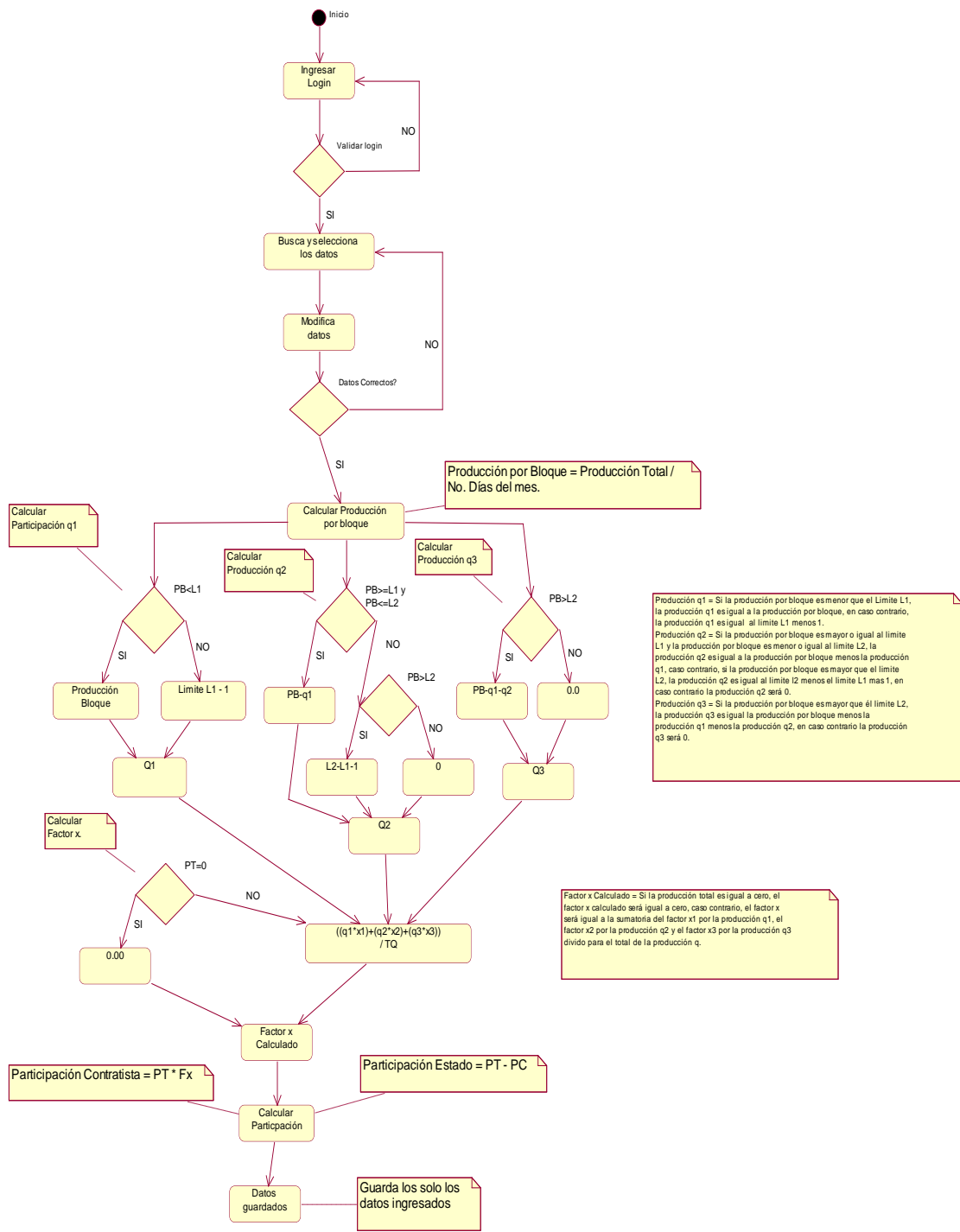
3.3.1.1.8 Gestionar Producciones Fiscalizadas

Bloque 16

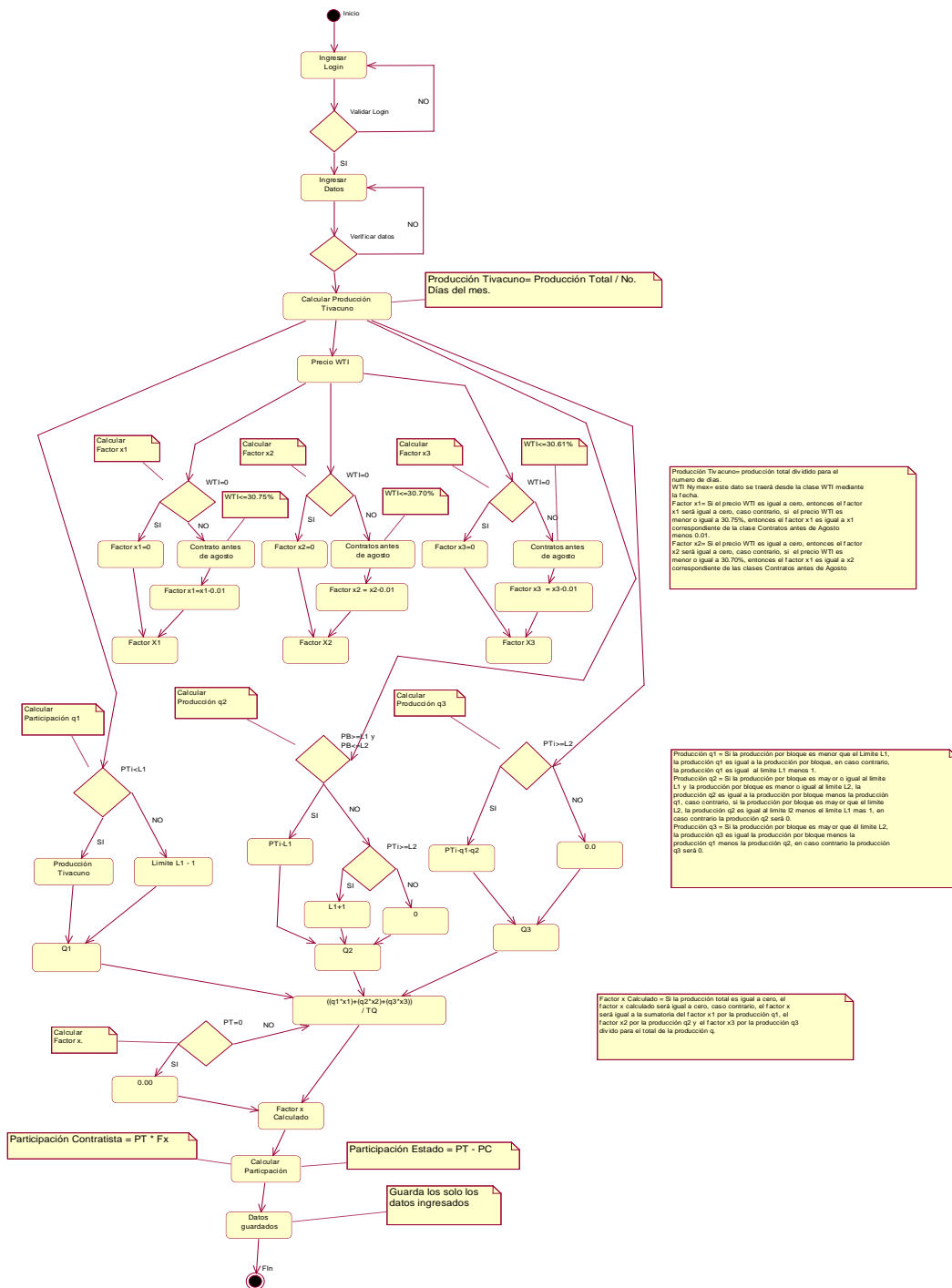
Ingreso



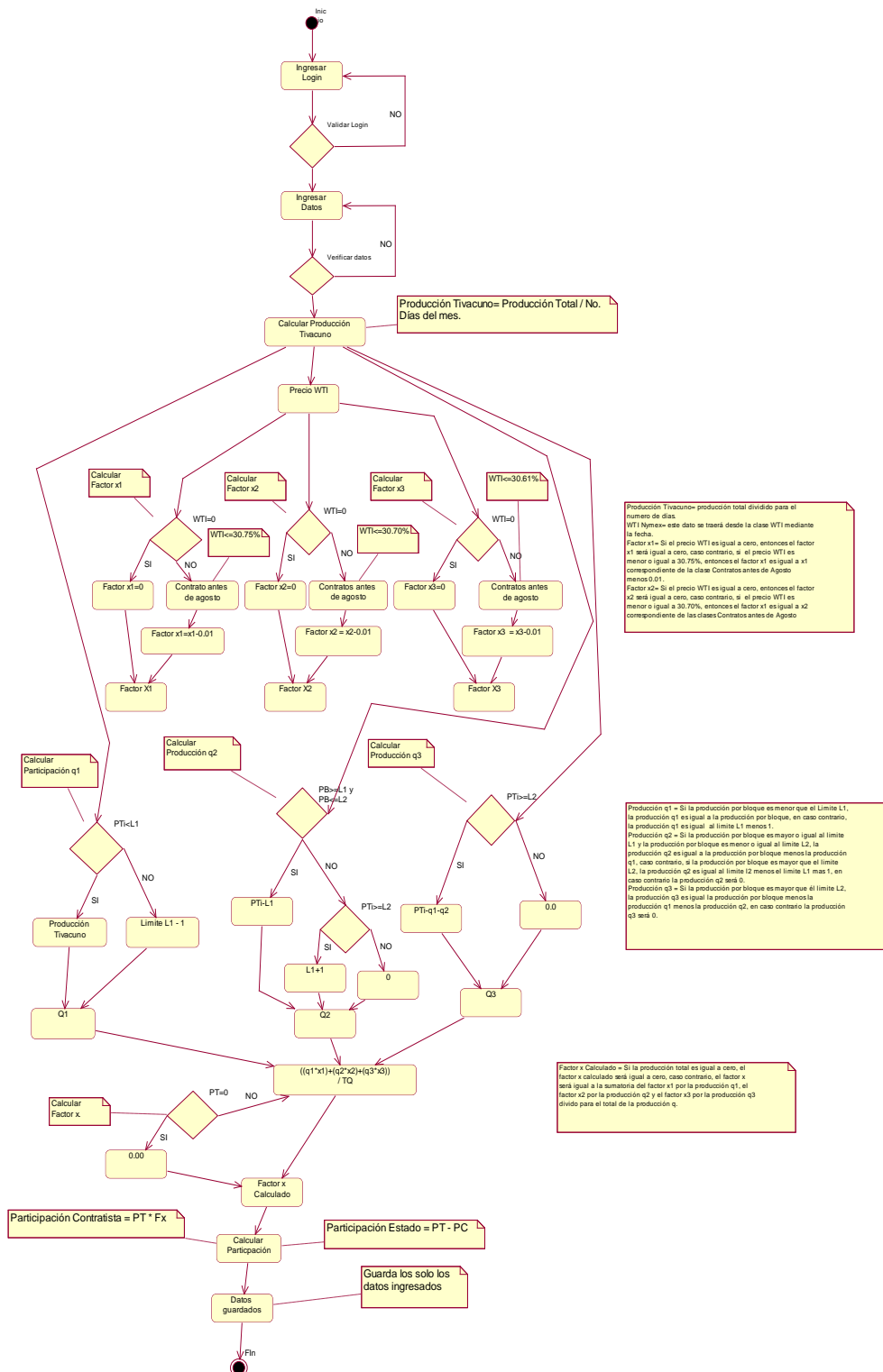
Modificación



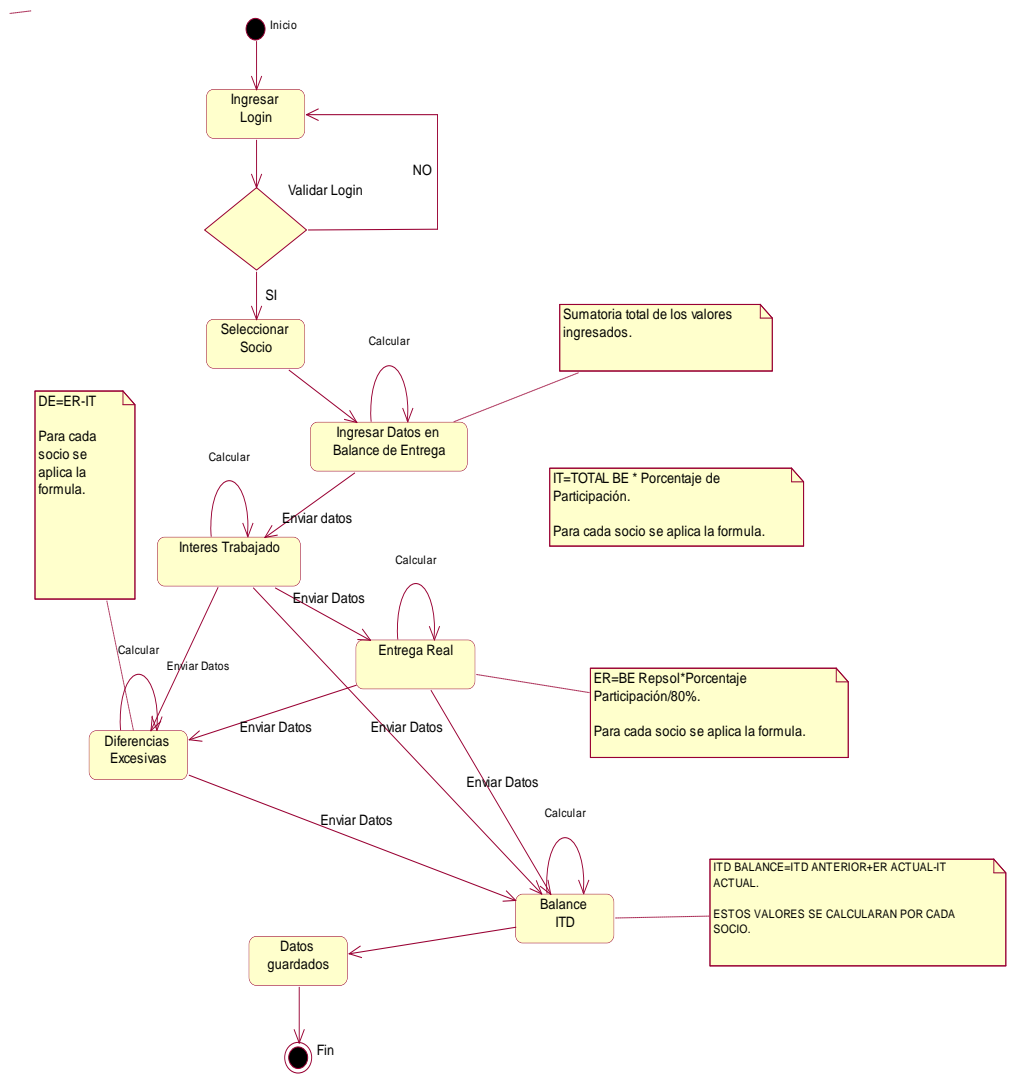
Tivacuno Ingreso



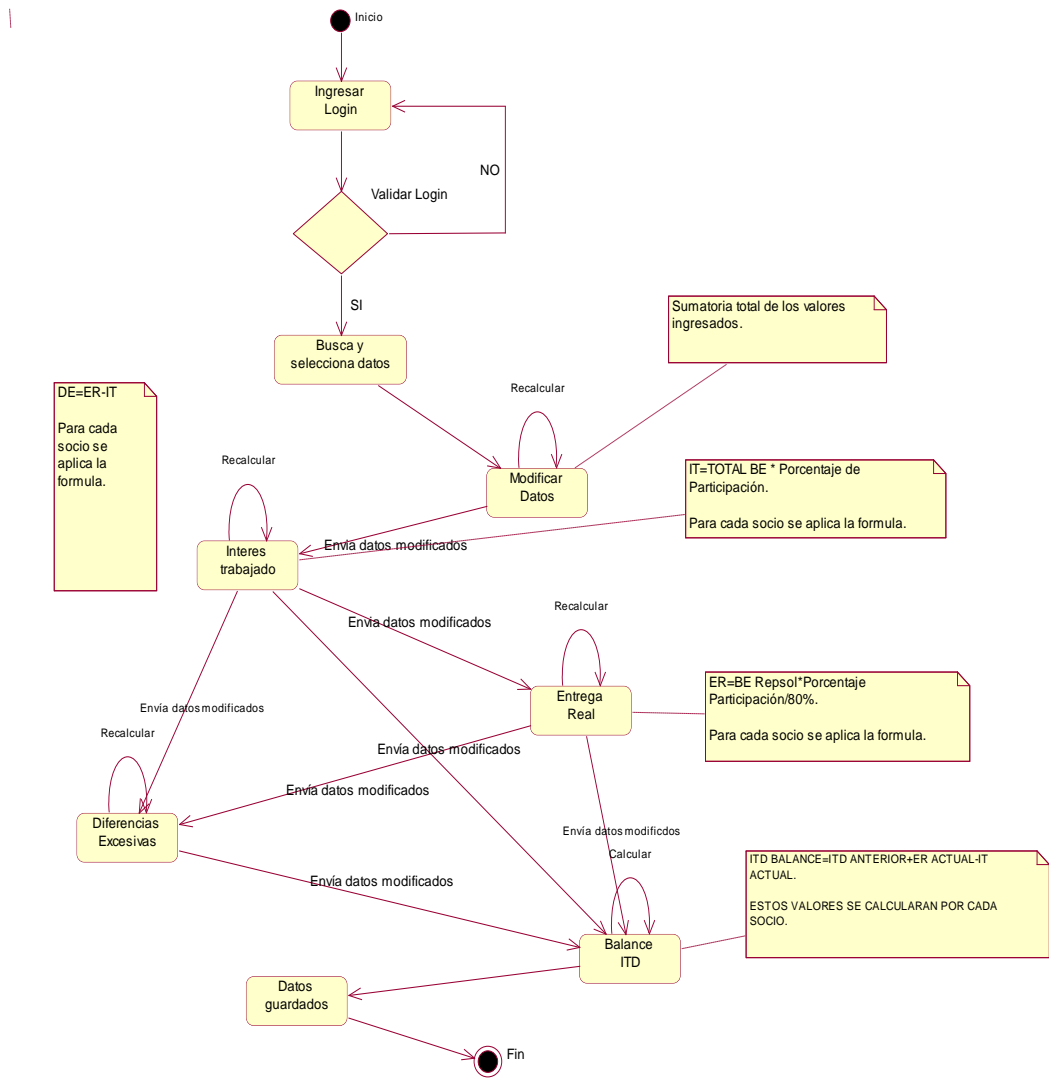
Modificación



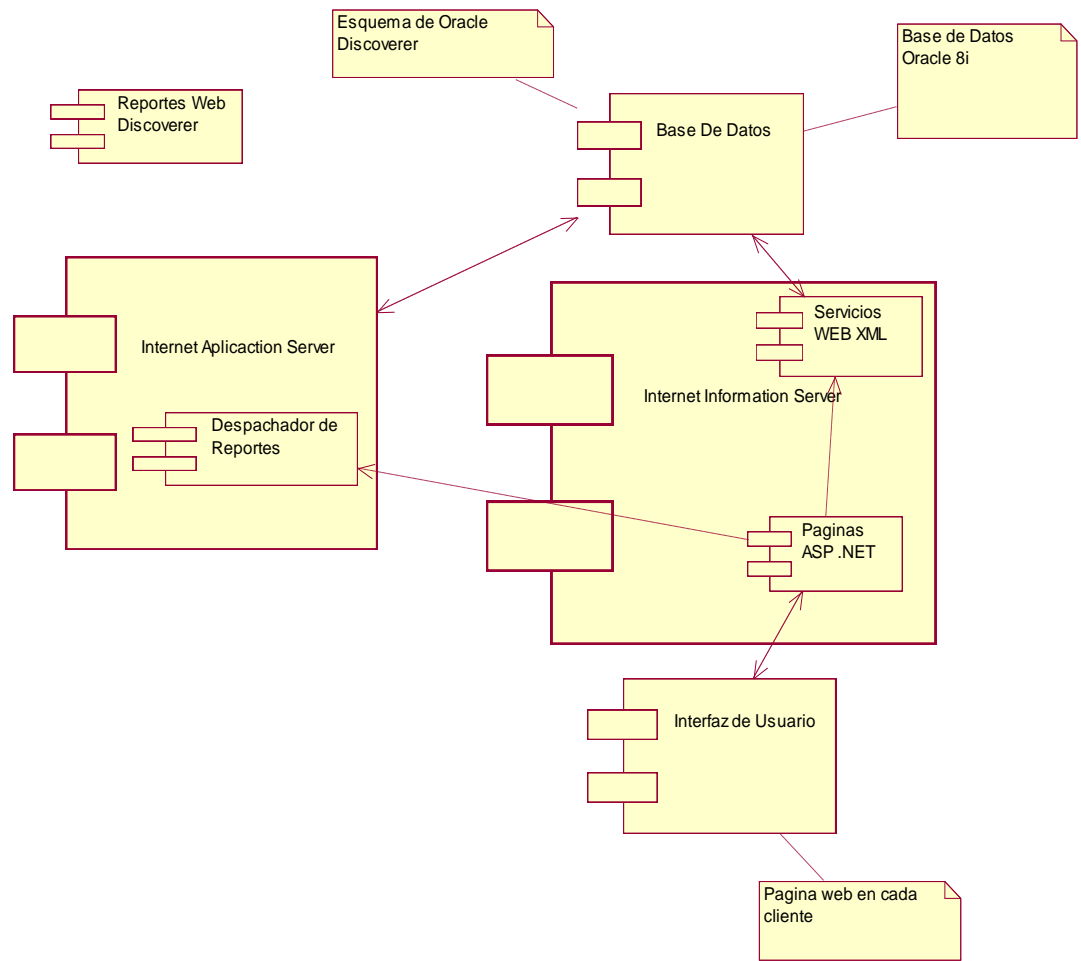
3.3.1.1.9 Gestionar Balance de Entregas Ingreso



Modificación



3.4.1.3 Construcción Modelo de Componentes



3.2 Diseño del Sistema

3.2.1 Organización del Sistema en Subsistemas.

El sistema de Operaciones Comerciales esta organizado en varios subsistemas los cuales se presentan a continuación:

- ✓ Ventas.- que abarca todas las ventas del consorcio, de los socios y de las compañías externas.
- ✓ Precios.- abarca todo lo referente a los precios del crudo internacional y nacional.
- ✓ Facturación.- abarca la facturación tanto de OCP como de SOTE.
- ✓ Producciones Fiscalizadas.- que abarca las producciones y participaciones de cada uno de los contratos de influencia de la compañía.
- ✓ Balance de entrega.- que abarca las entregas diarias de crudo al OCP entre los socios.

3.2.2 Identificación de recursos globales, control y mantenimiento.

El sistema de operaciones comerciales estará enfocado a una arquitectura n-capas y orientado a la web, razón por la cual los recursos utilizados en el diseño, desarrollo, implementación y pruebas fueron tomados en cuenta realizando un análisis previo a la realización del proyecto.

La identificación de estos recursos se listan a continuación:

Ambiente de desarrollo.- tanto para el desarrollo del sistema como para cada uno de los diferentes reportes que la aplicación presentará, se utilizara un ambiente de pruebas que tiene todas las características necesarias para su correcto funcionamiento.

Ambiente de Producción (Sistema).- el ambiente de producción que tiene la compañía esta adecuado a soportar las exigencias del sistema, tanto en arquitectura, velocidad y comunicación.

Ambiente de Producción(Reportes).- el ambiente de producción de reportes, de igual manera esta adecuado para soportar la continua lectura, petición y ejecución de los mismos.

Intranet.- la compañía cuenta con una infraestructura de red muy avanzada, que se la aprovechara para la comunicación entre las pc's de cada usuario y el ambiente de producción.

Lenguajes de Desarrollo, modelamiento y análisis.- de igual manera la compañía al tener convenio con varias empresas dedicadas al área de sistemas, ha obtenido las licencias necesarias de cada uno los productos utilizados para la realización del proyecto.

Para el control y mantenimiento del sistema se cuenta con un ambiente de pruebas el cual estará en permanente funcionamiento para la corrección de errores y adición de nuevos módulos o subsistemas.

3.2.3 Estructura básica del sistema

El sistema de Operaciones Comerciales utiliza tecnología web y arquitectura n capas, por esta razón la estructura básica estará definida de la siguiente manera:

Para cada una de las interfaces que se desarrollen, habrá un espacio determinado a ocupar en la pantalla, así, se creará una pantalla principal en la cual consten todos los menús principales del sistema y es aquí donde se cargarán el resto de páginas del sistema.

Los submenús para cada menú principal irán ubicados en la parte inferior izquierda de la pantalla de tal manera que al hacer un click en cualquiera de ellos la página solicitada se muestre en el centro. Dentro de cada submenú (siendo el caso) habrá también más submenús que se desplegarán en la misma posición que su anterior de tal manera que habrá una jerarquía de submenús muy amplia. Para el retorno al menú anterior se colocarán flechas de retorno que harán el trabajo. Cuando se opera con los submenús y las pantallas secundarias el menú principal no desaparecerá a menos que se esté en la consola de administración de seguridad.

Para la operación de los datos del sistema se desarrollarán pantallas similares las cuales constarán de una fuente para mostrar datos, botones para modificar, eliminar, crear datos nuevos, flechas de navegación, información de lo que se está haciendo y ayudas tipo tool-tip.

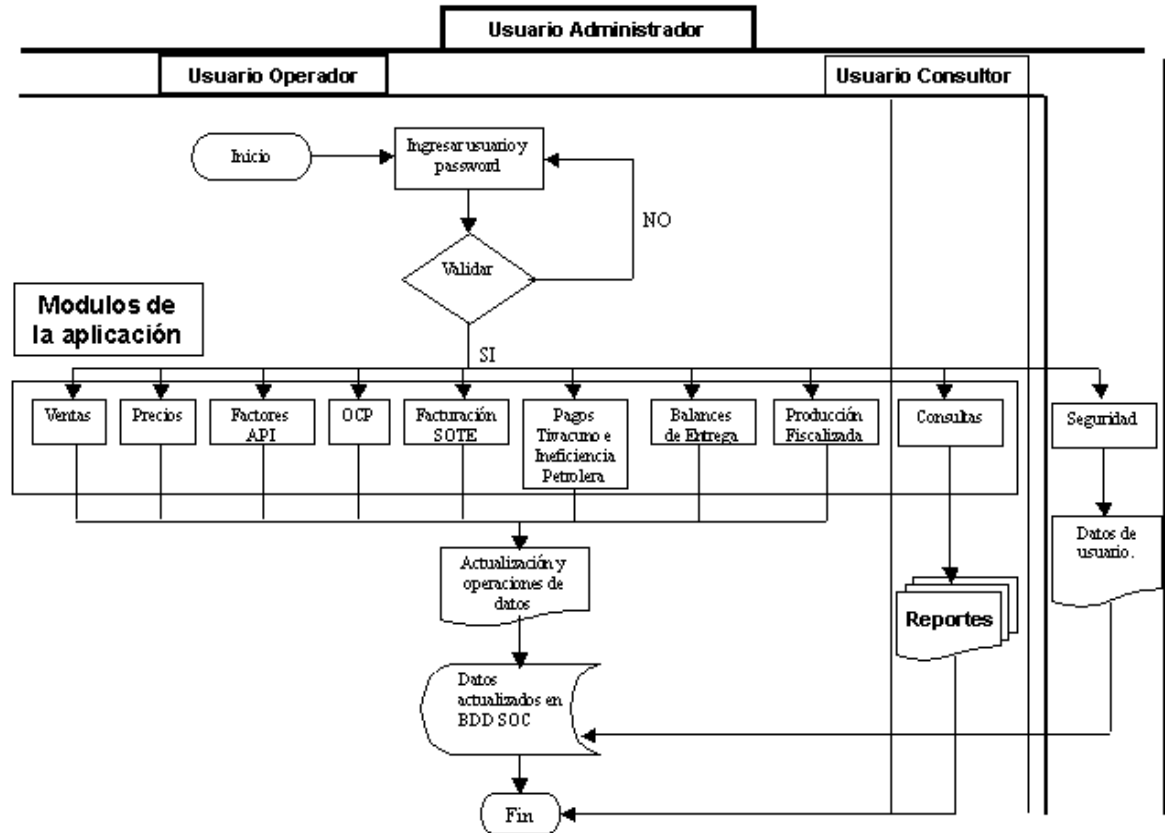
Cuando el usuario haga click en el botón nuevo se desplegará en la parte inferior de la fuente de datos las opciones para crear datos, siendo el caso de ser el número de ingresos demasiado grande como para abarcar en la misma página, se desplegará una página nueva en la posición de su anterior para proceder con el ingreso. De la misma manera esta página constará de las flechas de navegación, botón para guardar y si es necesario un calendario para escoger fechas.

Para la operación de las consultas del sistema habrá pantallas similares en las cuales se presentarán los parámetros a escoger y un botón para ejecutar el reporte, así como también un acceso directo a otra herramienta de reportes y las respectivas flechas de navegación.

Para todas las operaciones que maneja el sistema se definirán roles para los usuarios, los cuales son: Administrador, Operador, Consultor, el primero de ellos es aquel que tiene acceso total al sistema así como a las opciones de administración y seguridad, de la misma manera el operador tendrá acceso total excepto a las opciones de administración y seguridad y el consultor solo tendrá acceso a las opciones de consultas. La definición de operaciones sobre el sistema ya sea para el usuario operador y consultor serán únicamente permitidas por el administrador del sistema.

Por motivos de seguridad cada usuario que ingrese al sistema tiene un tiempo de inactividad de 30 minutos, después de finalizado el tiempo de espera, el sistema informara que la sesión a expirado y no se podrá hacer ninguna operación sobre el mismo

A continuación se muestra el flujo de datos básico del sistema:



Todos lo módulos que están a la izquierda de tiene acceso el Usuario Administrador
 Todos lo módulos que están a la izquierda de tiene acceso el Usuario Operador
 Todos lo módulos que están entre tiene acceso el Usuario Consultor

Fig. 3.1: (Flujo de Datos Básico del SOC)

El modo de acceso al sistema será mediante una pantalla de acceso en donde cada cliente tendrá que ingresar su usuario y password respectivos, asignados por el administrador.

Cada cliente que utiliza el sistema accederá al mismo mediante un browser, siempre y cuando este esté conectado a la intranet de la compañía, ya que el sistema se encontrara alojado en un solo sitio no es necesario hacer ninguna instalación previa en el cliente y este podrá ingresar a la aplicación las 24 horas del día.

El sistema utilizará una arquitectura n-tier para facilitar el acceso al mismo y evitar el congestionamiento de la información.

A continuación se presenta un diagrama de la arquitectura del sistema

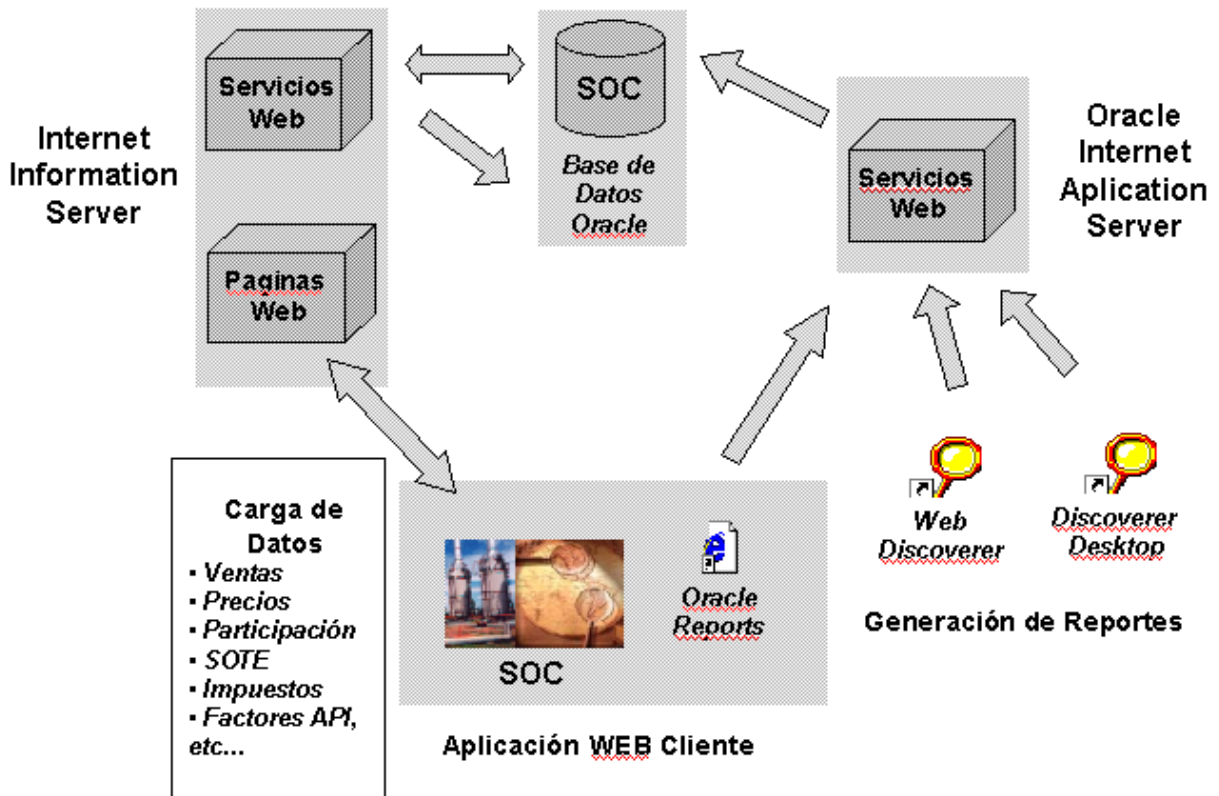
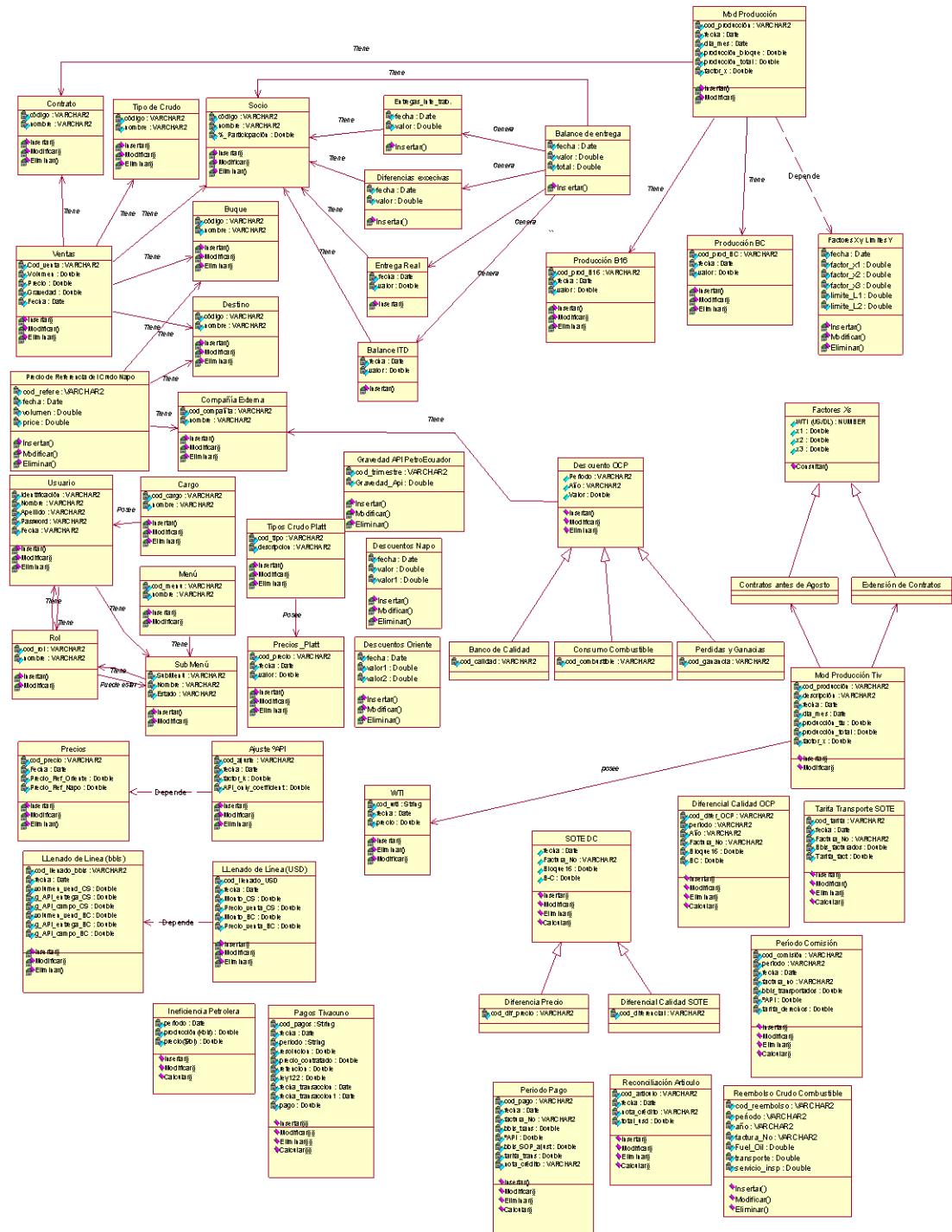


Fig 3.2: (Arquitectura Básica del SOC)

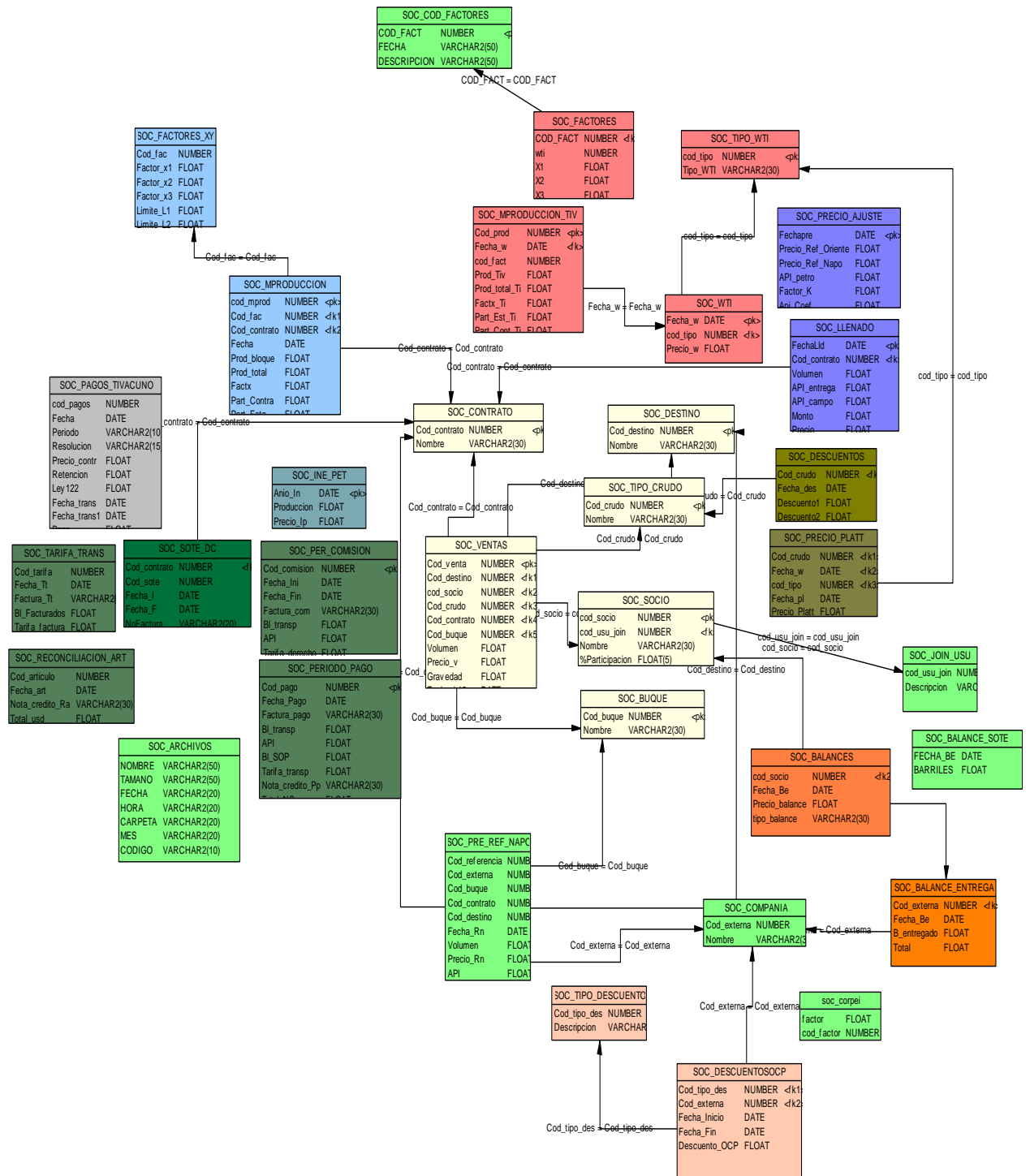
3.3 Diseño de Objetos

3.3.1 Modelo de Objetos Detallado

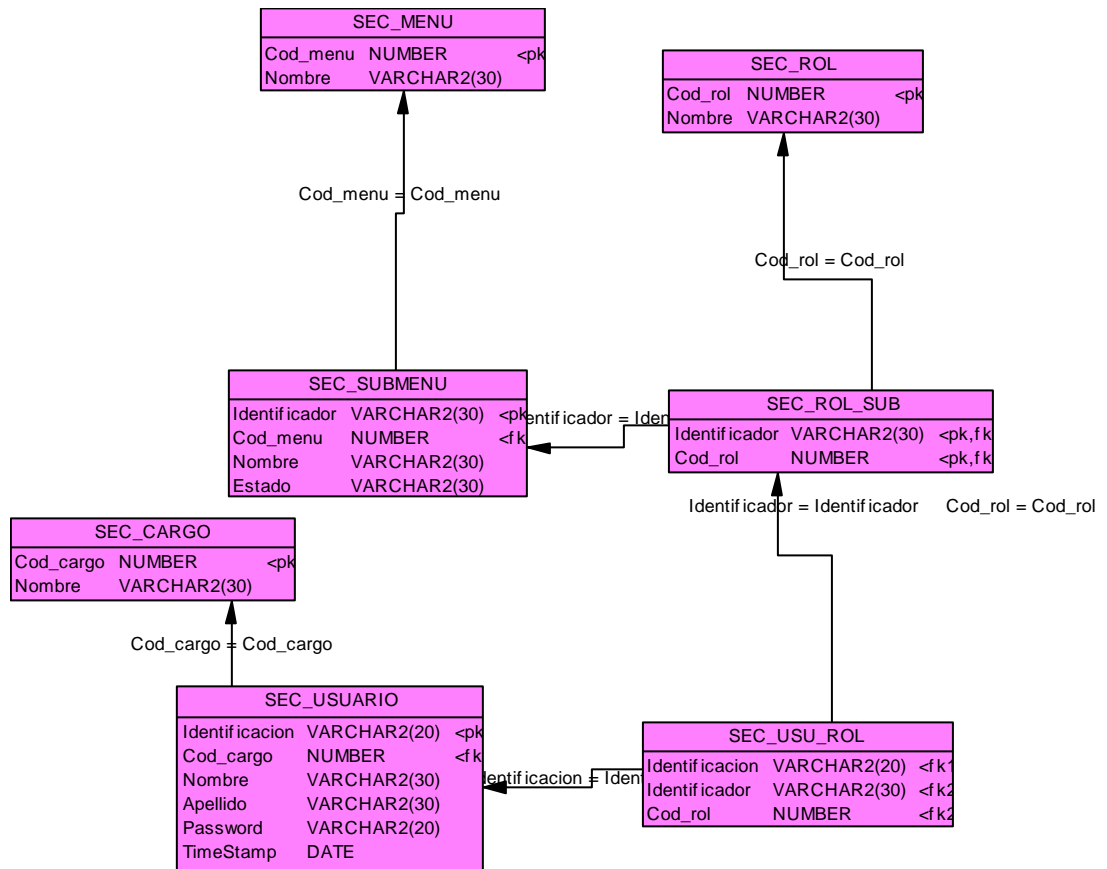


3.3.2 Modelo Entidad Relación

Soc



Seguridad



CAPÍTULO IV

PRUEBAS E IMPLEMENTACION

4.1 Introducción.

La información presentada en el actual capítulo es el resultado de un gran número de pruebas con el sistema desarrollado. Esta fase de pruebas se puede considerar una de las tareas que requieren de más tiempo ya que dadas las características del trabajo, una prueba consiste en poner a trabajar el sistema y observar cuidadosamente el comportamiento del mismo para tomar nota de los resultados.

Cada conjunto de pruebas aporta un nuevo dato de utilidad para el sistema obteniéndose así resultados con mejoras graduales. Este proceso de pruebas y la evolución en cuanto a los resultados obtenidos conforman el presente capítulo, en el cual se explican algunas de las pruebas aplicadas, los parámetros empleados, los problemas encontrados y las soluciones propuestas. La siguiente sección se ocupa sólo de dar una vista rápida de los problemas encontrados al momento de trabajar con el sistema y los efectos respectivos.

4.2 Problemas encontrados

Los problemas encontrados durante las pruebas del sistema son referentes a la interfaz de acceso, ingreso de datos, ejecución y formato de los reportes y configuración del ambiente. Durante este proceso se optó por hacer una depuración de cada módulo, identificando y corrigiendo el error potencial que estos tenían.

4.2.1 Verificación de la Interfaz del SOC.

Puesto que el sistema es accesado mediante un browser, se procedió primeramente con las pruebas de ésta.

1. Comprobar que el sistema sea ejecutable desde la Intranet en diferentes Plataformas. (E-1)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|--|------------------------------|--|------------|
| E-1.1 | Internet Explorer 6.0 en PC dentro de la compañía. SP1 | Acceder a la interfaz | Todas las páginas son accesibles y conservan su formato. | SI |
| E-1.2 | Internet Explorer 6.0 en PC dentro | Acceder a la interfaz | Todas las páginas son accesibles y conservan su | SI |

| | | | | |
|--|---------------------------|--|----------|--|
| | de la Compañía SP2. | | formato. | |
|--|---------------------------|--|----------|--|

Los resultados obtenidos en esta prueba están ajustados a los estándares corporativos de la compañía y no se podrá usar ningún reemplazo.

2. Verificar la implementación del acceso al sistema.(E-2)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|---|--|--|------------|
| E-2.1 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario correcto y password correcto. | El sistema permite el acceso identificando correctamente el tipo de usuario y ofreciéndole las opciones correspondientes a su tipo de cuenta | SI |
| E-2.2 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario correcto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.3 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario correcto y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.4 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario incorrecto y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.5 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario incorrecto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.6 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario nulo y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.7 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario nulo y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |

| | | | | |
|--------|---|--|--|----|
| E-2.8 | Acceder al sistema con la cuenta del administrador. | Nombre de usuario nulo y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.9 | Acceder al sistema con la cuenta del Operador | Nombre de usuario correcto y password correcto. | El sistema permite el acceso identificando correctamente el tipo de usuario y ofreciéndole las opciones correspondientes a su tipo de cuenta | SI |
| E-2.10 | Acceder al sistema con la cuenta del operador. | Nombre de usuario correcto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.11 | Acceder al sistema con la cuenta del operador. | Nombre de usuario correcto y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.12 | Acceder al sistema con la cuenta del operador. | Nombre de usuario incorrecto y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.13 | Acceder al sistema con la cuenta del operador. | Nombre de usuario incorrecto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.14 | Acceder al sistema con la cuenta del operador. | Nombre de usuario nulo y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.15 | Acceder al sistema con la cuenta del operador. | Nombre de usuario nulo y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.16 | Acceder al sistema con la cuenta del operador. | Nombre de usuario nulo y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.17 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario correcto y password correcto. | El sistema permite el acceso identificando correctamente el tipo de usuario y | SI |

| | | | | |
|--------|---|--|--|----|
| | | | ofreciéndole las opciones correspondientes a su tipo de cuenta | |
| E-2.18 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario correcto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.19 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario correcto y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.20 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario incorrecto y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.21 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario incorrecto y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.22 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario nulo y password correcto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.23 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario nulo y password incorrecto | El sistema marca error y vuelve a pedir datos de entrada | SI |
| E-2.24 | Acceder al sistema con la cuenta del consultor. | Nombre de usuario nulo y password nulo | El sistema marca error y vuelve a pedir datos de entrada | SI |

3. Verificar la implementación de la salida del sistema (E-3)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|---|---|--|------------|
| E-3.1 | Una sesión abierta con la cuenta de administrador | Cerrar una sesión con la cuenta de administrador. | El sistema pide una confirmación y si esta es dada cierra la sesión y redirige al usuario a la página de entrada al sistema. | SI |

| | | | | |
|-------|--|---|---|----|
| E-3.2 | Dos o más sesiones abiertas con la cuenta de operador | Cerrar una sesión con la cuenta de operador | El sistema pide una confirmación y si esta es dada cierra la sesión y redirige al usuario correspondiente a la página de entrada al sistema | SI |
| E-3.3 | Dos o más sesiones abiertas con la cuenta de consultor | Cerrar una sesión con la cuenta de operador | El sistema pide una confirmación y si esta es dada cierra la sesión y redirige al usuario correspondiente a la página de entrada al sistema | SI |

4. Verificar implementación del menú de opciones. (E-4)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|--|---|---|------------|
| E-4.1 | Probar el menú principal ubicado en la parte central -arriba de la pagina | Seleccionar del menú la opción. "Ventas" | El sistema despliega la página que contiene a los submenús para esta opción en el marco derecho. | SI |
| E-4.2 | Probar el menú principal ubicado en la parte central -arriba de la pagina. | Seleccionar del menú la opción "Precios" | El sistema despliega la página que contiene a los submenús para esta opción en el marco derecho. | SI |
| E-4.3 | Probar el menú principal ubicado en la parte central -arriba de la pagina. | Seleccionar del menú la opción "Ajuste API" | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.4 | Probar el menú principal | Seleccionar del menú la opción "OCP" | El sistema despliega la página que contiene a los | SI |

| | | | | |
|--------|---|--|---|----|
| | ubicado en la parte central - arriba de la pagina. | | submenús para esta opción en la el marco derecho. | |
| E-4.5 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Facturación SOTE” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.6 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Pagos Tivacuno” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.7 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Producción Petroecuador” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.8 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Producción Fiscalizada” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.9 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Balance de Entrega” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |
| E-4.10 | Probar el menú principal ubicado en la parte central - arriba de la pagina. | Seleccionar del menú la opción “Ingresos” | El sistema despliega la página que contiene a los submenús para esta opción en la el marco derecho. | SI |

Las opciones de cada uno de los submenús funcionan de idéntica manera que las de los menús, en los cuales no se encontró ninguna novedad.

5. Verificar la implementación de confirmación de salida. (E-5)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|---|---|--|------------|
| E-5.1 | Probar la opción salir. | Seleccionar la opción salir en el menú principal. | El sistema despliega una confirmación de salida. | SI |
| E-5.2 | El usuario se encuentra en la pagina de confirmación de salida. | Dar click en el botón "OK" | El sistema cierra la sesión y redirige al usuario a la pagina de entrada al sistema. | SI |
| E-5.3 | El usuario se encuentra en la pagina de confirmación de salida. | Dar click en el botón "Cancelar" | El sistema no cierra la sesión y el usuario puede seguir trabajando con la aplicación. | SI |

4.2.2 Verificación de las operaciones del SOC.

1. Verificación de la implementación de las operaciones. (D-1)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|----------------------------------|------------------------------------|--|-------------|
| D-1.1 | Probar el botón editar | Seleccionar opción "editar". | El sistema presenta en modo de edición los datos que el usuario selecciono | SI |
| D-1.2 | Probar el botón actualizar | Seleccionar la opción "Actualizar" | El sistema actualiza y guarda los datos en la base de datos y le confirma al usuario | SI |
| D-1.3 | Probar el botón cancelar | Seleccionar la opción "Cancelar" | El sistema devuelve al modo de vista los datos que el usuario selecciono | SI |
| D-1.4 | Probar el botón Eliminar | Seleccionar la opción "Eliminar" | El sistema borra los datos de la base de datos y le confirma al usuario | SI |
| D-1.5 | Probar el | Seleccionar la | El sistema | Depende del |

| | | | | |
|-------|--------------------------|----------------------------------|--|------|
| | botón nuevo | opción “Nuevo” | redirecciona al usuario a una pantalla para el ingreso de datos. | Caso |
| D-1.6 | Probar el botón guardar. | Seleccionar la opción “Guardar” | El sistema guarda los datos en la base de datos y le confirma al usuario | SI |
| D-1.7 | Probar el botón generar | Seleccionar la opción “Generar” | El sistema hace cálculos internos y genera los resultados a través de modelo económicos. | SI |
| D-1.8 | Probar el botón regresar | Seleccionar la opción “Regresar” | El sistema redirecciona al usuario a la pagina anteriormente visitada. | SI |
| D-1.9 | Probar el botón buscar | Seleccionar la opción “Buscar” | El sistema busca los datos ingresado por el usuario. | SI |

2. Verificar la lectura de la base de datos.(F-1)

| # | Descripción o contexto de Prueba | Entrada o acción del usuario | Reacción o resultado esperado del sistema | Confirmado |
|-------|---|--|---|------------|
| F-1.1 | El sistema intenta conectarse a la base de datos. | La base de datos no se encuentra activa o no permite el acceso | El sistema obtiene la excepción, para el proceso y despliega un mensaje de error. | SI |
| F-1.2 | Se bloqueo la base de datos, el usuario que inicio el proceso ha abandonado el Sistema. | Se perdió la conexión con la base de datos | El sistema obtiene la excepción, para el proceso y despliega un mensaje de error. | SI |
| F-1.3 | El sistema intenta leer una tabla de la base de datos. | La tabla requerida no se encuentra. | El sistema obtiene la excepción, para el proceso y despliega un mensaje de error. | SI |
| F-1.4 | El sistema intenta leer una tabla de la base de datos. | La tabla requerida no contiene datos. | El sistema obtiene la excepción, para el proceso y despliega un mensaje. | SI |

4.3 Validación del Sistema SOC.

4.3.1 Validación de las Salidas del Sistema

La finalidad del Sistema SOC es brindar al usuario final la información precisa acerca de las Operaciones Comerciales de la compañía, tomando en cuenta la cantidad de reportes y datos que existen, la única forma de comprobar la validez de los resultados del sistema es a mano. Con este fin, cada reporte de la aplicación fue ejecutado y comparado con los reportes existentes en ese momento (Datos Excel).

Debido a la gran cantidad de datos que se ingresaron a la base de datos desde formatos Excel y la forma como se los ingresaron, hubo muchos errores en la compatibilidad de los mismos, esto se pudo demostrar con la ejecución de los reportes y la comparación con las hojas Excel. Seguido a esto se corrigió de forma manual cada uno de los datos incorrectos.

Los problemas encontrados en la mayoría de reportes fueron la representación de los números con decimales, ya que cada decimal de más o menos afecta en sí al cálculo total de los valores, por otro lado, en cuanto a las operaciones y cálculos matemáticos aplicados no se encontraron anomalías.

Otro punto importante en la validación de los reportes fue la presentación del formato, en este aspecto se hizo considerables cambios a casi todos para mejorar la imagen y presentación final de los mismos.

4.3.2 Tiempo de ejecución del sistema

Las pruebas que se hicieron antes de lanzar el sistema a producción, se las hizo con una base de pruebas y en un ambiente de desarrollo en donde no se encontró ninguna novedad. Cuando se paso a producción todo el sistema, los resultados fueron mucho más satisfactorios debido a la gran infraestructura con que cuenta la compañía.

La gran cantidad de información que tiene la base de datos junto con el potencial de crecimiento no es un limitante para el desempeño del sistema, por otro lado, tanto la ejecución de reportes como las operaciones sobre la base de datos tienen un tiempo de respuesta aceptable.

4.4 Manual del Usuario

Ver Nexo A.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES.

Conclusiones

- √ La disponibilidad de la información fue un punto crucial en el desarrollo de este proyecto, ya que cada uno de los usuarios que llevaban la información antes de la construcción, tenían que acceder a formatos Excel ó pedir al administrador de estos documentos la información requerida, ahora, con el sistema en marcha cada persona que tiene acceso al mismo puede disponer de la información en formatos comunes, con datos confiables, actualizados, en forma inmediata y sin pérdida de tiempo.

- √ La seguridad del Sistema tiene un nivel corporativo, a pesar de que la compañía tiene sus métodos de protección contra intrusos, la aplicación cuenta con un sistema de seguridad basado en roles y niveles de usuarios, la razón por la cual se hizo esto, es por que la información mostrada en este sistema es confidencial tanto para personas externas como para algunas personas de la compañía, por esto, el número de usuarios que utiliza el sistema esta limitado a personas que ocupan cargos de administración de operaciones y de administración financiera.

- √ Otro de los puntos importantes en el desarrollo de este aplicativo fue el de los reportes dinámicos, cada usuario responsable de mantener la información actualizada esta en la capacidad de generar sus propios reportes, así como también los usuarios que utilizan el sistema para consultas, ya que cada uno puede acceder a la herramienta destinada para este fin, y así, la información esta siempre disponible y se la puede manipular de acuerdo a las necesidades que tenga cada usuario.

- √ Tanto la depuración de errores del sistema en sí y la de los reportes, se las hace en un ambiente de desarrollo creado para este fin, así el funcionamiento del sistema nunca se detiene y está disponible las 24 horas del día.

- √ En el presente trabajo se han obtenido los siguientes resultados:
 - Se propuso usar tecnología Asp .net y una arquitectura n-capas en el desarrollo de aplicaciones Web presentada en el plan del proyecto.
 - Se estudio y documento cada uno de los casos de las operaciones comerciales de la compañía presentada en el capítulo 3 de este documento.
 - Se definió el esquema de la base de datos del sistema de operaciones comerciales presentada en el capítulo 3 de este documento.
 - Se diseñó, optimizó el proceso, y se implementó en .NET el Sistema SOC presentados en los capítulos 3 y 4.
 - Se hicieron pruebas del Sistema SOC y los componentes de la interfaz, presentados en el capítulo 4 de este documento, logrando el desempeño esperado en esta fase de proyecto.

El sistema de Operaciones Comerciales se encuentra en producción al cual se le han hecho las correcciones necesarias para su óptimo funcionamiento, además el sistema constará de módulos que se irán desarrollando de acuerdo a los requerimientos que sigan apareciendo entre el equipo de desarrollo y el usuario final.

Por todas estas razones se puede decir que:

- √ El análisis, diseño, codificación e implementación del Sistema de Operaciones Comerciales a nivel de la compañía fue todo un éxito, ya que cada uno de los usuarios participó activamente a su debido momento en las etapas de desarrollo, aportando su conocimiento del negocio, aplicando los estándares de la compañía y haciendo mejoras substanciales en la etapa de pruebas, para la recolección de la información se aplicó la norma IEEE 830 (Especificación de Requerimientos) y para el análisis y diseño se aplicó la Metodología OMT con UML, por esta razón las personas que utilizan la aplicación pueden operar el sistema con facilidad y hacer uso de la información, mejorando así la toma de decisiones y los procesos consecuentes a las Operaciones Comerciales de REPSOL YPF Ecuador.

Recomendaciones

- √ El módulo de seguridad implantado en el sistema cuenta con las normas establecidas de seguridad en aplicaciones tipo Web, ya que como se ha dicho, la gran infraestructura de la compañía ayuda en gran medida a proteger la información, sin embargo, es recomendable sugerir que los usuarios que manejan la aplicación se reserven la clave solo para su uso personal y no deberá ser transferida o entregada a otro usuario sin previa autorización del administrador del sistema.

- √ De la misma manera el nivel, permisos y roles de cada uno de los usuarios fueron definidos de acuerdo a las políticas de confidencialidad de la información de la compañía, por esta razón se recomienda, antes de generar permisos a los usuarios de cualquier tipo, realizar un análisis de la información que se va a mostrar, el tipo de usuario que requiera de acceso y el nivel que se le pretende dar, esto, con el objetivo de disminuir el riesgo de fuga de la información.

- √ Se recomienda que todos los cambios que se apliquen al sistema tanto de fondo como de forma se los deberá desarrollar, implementar y probar en un entorno de pruebas, de tal manera que se depuren todos los errores que puedan arrojar los nuevos cambios. De la misma manera el ambiente de pruebas debe ser un reflejo total del ambiente de producción ya que solo así se podrá saber si el sistema va a responder (tiempo de respuesta, capacidad de operación) de la misma manera que cuando se lo implemente en producción.

BIBLIOGRAFÍA

Referencias Internet:

- ✓ <http://www.oracle.com/index.html>
Oracle Developer Suite
Oracle Reports Developer
OracleAS Discoverer
Oracle Data Base
Oracle Data Base 8i
- ✓ <http://www.microsoft.com/>
Visual Studio .NET
Visual Basic .NET
ASP .NET
Web Services
- ✓ http://www.oracle.com/technology/products/discoverer/pdf/OracleApplicationServer_Discoverer_10g_TechWP.pdf
Tutorial de OracleAS Discoverer
- ✓ http://www.oracle.com/technology/products/reports/10g/reports_technicalWP.pdf
Tutorial de Oracle Reports Developer
- ✓ <http://es.gotdotnet.com/quickstart/aspplus/doc/quickstart.aspx>
Tutorial de ASP .NET
- ✓ MSDN Visual Basic .NET 2003
Ayuda en línea de Visual Basic .Net 2003
- ✓ http://www.cgrsoftware.com/recursos_dir.htm?basicnet_1
- ✓ <http://www.eside.deusto.es/postgrado/puntonet/>
- ✓ <http://www.platts.com>

Referencias Personales:

- ✓ Rodrigo Yevenes (Entrevistado), Asistente de Operaciones Comerciales.
- ✓ Patricia Valladares (Usuario), Asistente Financiera.

Referencias Técnicas:

- ✓ Hojas de calculo (Excel), utilizadas por los usuarios entrevistados.
- ✓ Modelos de clases.
- ✓ Modelos de Casos de Uso
- ✓ Diagramas varios.

ANEXO A

valor : Double

ANEXO B