

**UNIVERSIDAD MAYOR DE SAN ANDRÉS  
FACULTAD DE CIENCIAS PURAS Y NATURALES  
CARRERA DE INFORMÁTICA**



**PROYECTO DE GRADO  
“SISTEMA DE CONTROL Y SEGUIMIENTO PARA EL TRATAMIENTO DE  
ACEITES”  
CASO: RAMSOIL**

**PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA  
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS**

**AUTOR : LEVÍ LAURA RAMOS  
TUTOR : LIC . NANCY ORIHUELA SEQUEIROS  
REVISOR : LIC . MENFY MORALES RIOS**

**La Paz – Bolivia  
2009**

## DEDICATORIA

---

*A mi padre Leónidas, a quien amo, quien es mi ejemplo de vida, y vela para que nunca me falte nada.*

*A mi madre Mercedes, la mujer que más amo, quien me apoya y me da fuerzas para seguir adelante.*

*A mi hermano Edson a quien quiero mucho, por su cariño, y por aguantarme.*

---

## **AGRADECIMIENTOS**

A DIOS porque gracias a él tengo familia, amigos, mi música, que junto a él forman mis cuatro pilares en esta vida.

A mi familia que sin su apoyo en momentos críticos quizás no hubiese salido adelante, y que gracias a los consejos de mi padre y de mi madre a ellos principalmente les doy las gracias.

Agradecer a mi tío, José Luis Ramos Gutiérrez a quien quiero mucho, y me brindo su apoyo hasta el día de hoy, tanto en mis estudios, como en mi vida personal.

Quiero de gran manera agradecer a la Lic. Nancy Orihuela Sequeiros por su paciencia y su colaboración en el transcurso de la elaboración del Proyecto de Grado, de igual manera agradezco a mi revisora Lic. Menfy Morales Rios por su apoyo, sugerencias y observaciones que me ayudaron a superar y alcanzar los objetivos trazados.

Gracias a mis amigos que me apoyaron a lo largo de mi vida universitaria, y que de mi parte siempre tendrán a alguien incondicional.

## RESUMEN

La empresa RAMSOIL es una empresa dedicada al refinado de aceites para automóviles, lleva en la industria siete años en dicha labor, teniendo un gran crecimiento en el ámbito empresarial.

El presente proyecto de grado “Sistema de Control y Seguimiento para el Tratamiento de Aceites, Caso: RAMSOIL” , tiene como propósito el de coadyuvar con el control y la información de la producción de refinado de aceites, así también el de poder registrar las compras de insumos y materiales, ventas de aceites, y poder mostrar gracias a la programación lineal una sugerencia con respecto a la mejora de la producción diaria.

Gracias a la metodología RUP (Proceso Racional Unificado), y al lenguaje UML para el modelado, se puede observar paso a paso el proceso de análisis y desarrollo realizado para el presente proyecto de grado, y que por ello se pueda cumplir con la exigencias requeridas por la empresa, también se da conformidad a las métricas de calidad realizadas, y que gracias a estas se pueden medir la funcionalidad, confiabilidad, portabilidad, y mantenibilidad del sistema.

Para el desarrollo del sistema se utilizaron las herramientas de software libre como ser PHP como lenguaje de programación, MYSQL como gestor de base de datos, y servidor web Apache.

## ÍNDICE GENERAL

### CAPÍTULO I

#### INTRODUCCIÓN

Pág.

1.1 INTRODUCCIÓN .....	1
1.2 ANTECEDENTES .....	3
1.3 PROBLEMA .....	3
1.3.1 IDENTIFICACIÓN DEL PROBLEMA .....	3
1.3.2 LISTA DE PROBLEMAS .....	4
1.4 OBJETIVO GENERAL .....	4
1.5 OBJETIVOS ESPECÍFICOS .....	5
1.6 LÍMITES Y ALCANCES .....	5
1.7 APORTE.....	6
1.8 JUSTIFICACIÓN .....	7
1.8.1 JUSTIFICACIÓN TÉCNICA .....	7
1.8.2 JUSTIFICACIÓN OPERATIVA .....	7
1.8.3 JUSTIFICACIÓN ECONÓMICA.....	7
1.9 MÉTODOS DE INVESTIGACIÓN Y METODOLOGÍA.....	7
1.10 ANÁLISIS COSTO BENEFICIO .....	8
1.10 ANÁLISIS COSTO BENEFICIO .....	8
1.10.1 ANÁLISIS DE COSTOS .....	8
1.10.2 ANÁLISIS DE BENEFICIOS .....	9

## CAPÍTULO II

### MARCO TEÓRICO

2.1 INTRODUCCIÓN .....	11
2.2 PROCESO UNIFICADO RACIONAL (RUP) .....	11
2.2.1 DIRIGIDO POR CASOS DE USO .....	13
2.2.2 CENTRADO EN LA ARQUITECTURA .....	13
2.2.3 ITERATIVO E INCREMENTAL .....	14
2.3 LENGUAJE UNIFICADO DEL MODELO .....	16
2.3.1 CASOS DE USO .....	16
2.3.1.1 DIAGRAMAS DE CASOS DE USO DE ALTO NIVEL .....	18
2.3.1.2 DIAGRAMAS DE CASOS DE USO EXPANDIDO .....	18
2.3.2 DIAGRAMA DE SECUENCIA .....	19
2.3.3 DIAGRAMA DE COLABORACIÓN .....	20
2.3.4 DIAGRAMA DE CLASES .....	21
2.4 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS .....	22
2.5 PROGRAMACIÓN LINEAL .....	24
2.5.1 MÉTODO SIMPLEX .....	25
2.6 MÉTRICAS DE CALIDAD .....	28
2.6.1 FUNCIONALIDAD .....	29
2.6.2 CONFIABILIDAD .....	31
2.6.3 PORTABILIDAD .....	32
2.6.4 MANTENIBILIDAD .....	33
2.7 SEGURIDAD DEL SISTEMA .....	33

## CAPÍTULO III

### ANÁLISIS Y DISEÑO

3.1 INTRODUCCIÓN .....	35
3.2 DESCRIPCIÓN DE LOS ACTORES .....	35
3.3 CATÁLOGO DE REQUERIMIENTOS DEL SISTEMA .....	36
3.4 ANÁLISIS .....	39
3.4.1 DISEÑO DE CASOS DE USO .....	39
3.4.1.1 DIAGRAMA DE CASOS DE USO DE ALTO NIVEL.....	39
3.4.1.2 DESCRIPCIÓN CASOS DE USO DE ALTO NIVEL.....	40
3.4.1.3 CASOS DE USO ESENCIAL O EXPANDIDO .....	43
3.4.2 DIAGRAMA DE SECUENCIA .....	52
3.5 DISEÑO .....	55
3.5.1 DIAGRAMA DE CLASES .....	55
3.5.2 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS .....	56
3.5.3 DIAGRAMA DE COLABORACIÓN .....	57
3.5.4 DISEÑO LÓGICO DEL SISTEMA .....	60
3.6 APLICACIÓN DE LA PROGRAMACIÓN LINEAL – MÉTODO DEL SIMPLEX .....	62
3.7 ARQUITECTURA DEL SISTEMA .....	66
3.8 MÉTRICAS DE CALIDAD.....	68
3.8.1 FUNCIONALIDAD .....	68
3.8.2 CONFIABILIDAD .....	71
3.8.3 PORTABILIDAD.....	73
3.8.4 MANTENIBILIDAD .....	74
3.8.4.1 MANTENIMIENTO ADAPTIVO.....	74

3.8.4.2 MANTENIMIENTO PERFECTIVO.....	74
3.8.5 FACILIDAD DE USO .....	75
3.9 SEGURIDAD DEL SISTEMA .....	75
3.10 DISEÑO DE INTERFAZ.....	76

## CAPÍTULO IV

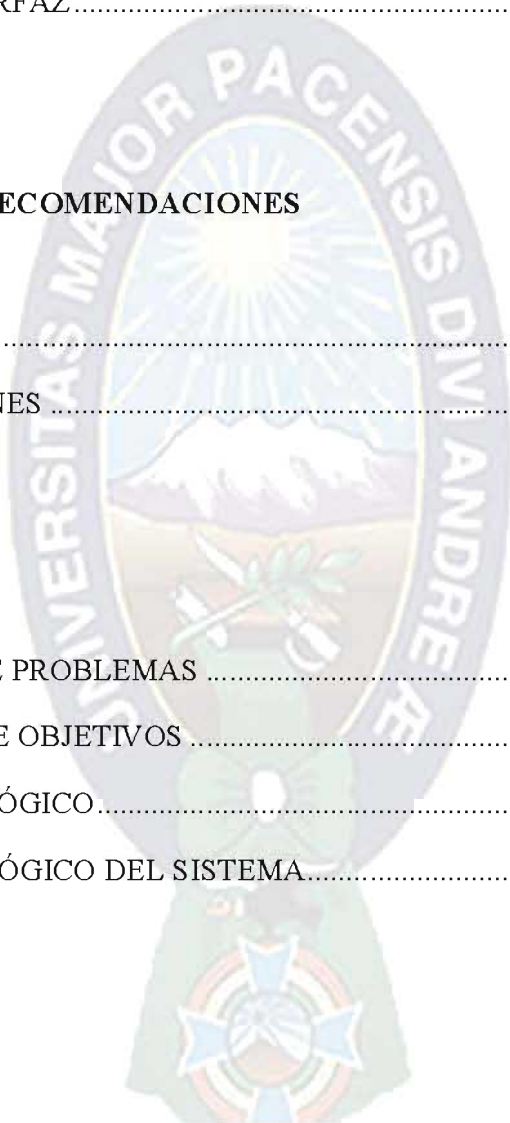
### CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES.....	80
4.2 RECOMENDACIONES .....	82

### BIBLIOGRAFIA

ANEXO A ÁRBOL DE PROBLEMAS .....	A-1
ANEXO B ÁRBOL DE OBJETIVOS .....	B-1
ANEXO C MARCO LÓGICO.....	C-1
ANEXO D DISEÑO LÓGICO DEL SISTEMA.....	D-1

### DOCUMENTACIÓN







---

**CAPÍTULO I**  
**INTRODUCCIÓN**

## CAPITULO I

### INTRODUCCIÓN

#### 1.1 INTRODUCCIÓN

Actualmente existen empresas dedicadas a la fabricación de bienes que son de uso diario, y que sin estos no podríamos satisfacer nuestras necesidades primordiales.

El aceite como lubricante de automóviles se ha convertido en un bien imprescindible hoy en día, puesto que este es necesariamente adquirible por aquellos que dependen de usar un automóvil como medio de transporte para poder trasladarse de un lugar a otro, y que utilizan este medio de transporte de forma particular o pública (uso de minibús, autobús, trufi, taxi), y es por eso la importancia que tienen las empresas fabricantes de aceites, es necesario que las mismas tengan un proceso automatizado en la fabricación de estos productos.

El aceite para automóvil es un líquido vital que se ubica entre las piezas que trabajan de manera solidaria, formando una fina película que evita la fricción en seco, que rayaría las partes móviles, causando su destrucción.

Para el funcionamiento de un automóvil existen distintos tipos de aceite, y estos son específicos para distintos modelos de motor. La utilización de un tipo de aceite está determinada por las características mismas del motor, y por la altura sobre el nivel del mar.

En la actualidad existen tres tipos de aceites para motores modernos:

- SAE 5W-50: Clima caluroso, con invierno no muy frío.
- SAE 15W-40: Clima templado; veranos cálidos e inviernos con frío de regular intensidad.
- SAE 20W-50: Clima extremoso, con cambios de temperatura del calor frío al calor. Éste tipo se puede usar todo el año con veranos cálidos e invierno gélido.

No todos los motores pueden utilizar cualquier aceite. Además del tipo de aceite, su duración ofreciendo buenas condiciones de protección, está directamente relacionada con el estado del motor y con su sincronización. Además de la pérdida de viscosidad producida por el uso normal durante muchos kilómetros, el aceite se degrada por acción de la temperatura de funcionamiento del motor, y existe una deficiencia en las cualidades de sus aditivos por la recolección de impurezas como: carbono, combustible, ácidos, residuos alquitranados, partículas metálicas.

El refinado de aceite es el proceso de purificación del aceite desde que éste es fabricado en su etapa inicial, hasta que sale como producto final, por ende se entiende que el re-refinado del aceite es la re-purificación del aceite, pero que esta re-purificación se lo realiza a un producto final que puede estar ya listo para la venta, en otras palabras, al aceite se le realiza una mejora, para que ésta tenga un nivel de viscosidad menor que la que tuvo antes del proceso de refinado inicial, consiguiendo así un producto de alta calidad, que supera al aceite para automóviles común.

La empresa RAMSOIL es una empresa dedicada principalmente al re-refinamiento, fabricación y distribución al por mayor de aceite para automóviles, cuyo objetivo principal es el de lograr la preferencia en el consumidor del producto realizado.

La empresa RAMSOIL actualmente no cuenta con un sistema informático que coadyuve en el control de todo el proceso de re-refinamiento del aceite, es así que se ve la necesidad de contar con un software para el control y seguimiento en dicho proceso que ayude al área de gerencia para así optimizar la toma de decisiones que está estrechamente relacionado con los procesos de planificación y control, logrando así sus objetivos.

El Sistema de Control y Seguimiento del proceso de re-refinado de aceites, presentado en éste documento toma en cuenta la complejidad que representa el tema, y servirá de apoyo al área de producción y gerencia, ya que ayudará a las decisiones oportunas y precisas, realizando un control y seguimiento del rendimiento del área de producción, identificando el tiempo y materiales utilizado en el proceso y desarrollo de los dos tipos de aceite que recicla la empresa, los cuales son : SAE 15W-40 y SAE 5W-50.

La propuesta del presente proyecto constituye el estudio y el análisis que sirve de apoyo fundamentalmente a la gerencia que es la que carece de información precisa y oportuna.

## **1.2. ANTECEDENTES**

El control y seguimiento de los procesos de producción para una empresa dedicada a la fabricación de un bien, es lo esencial para que este pueda tener una estabilidad en el manejo de las etapas de producción.

Dada la importancia que exige el control y seguimiento de la producción, existen proyectos de grado realizados para ayudar con este fin, y que se tomaron en cuenta como antecedentes para la realización del presente proyecto, los cuales son:

- Sistema de Información Integrado CEPPDM [Pauca,1999]
- Sistema Automatizado de Registro y Seguimiento para la Correspondencia y/o Procesos Judiciales Departamento de Asesoría Jurídica.[Ticon,2005].
- Sistema de Información para el Seguimiento y Control del Registro de Buques [Merca,2006].
- Sistema de Información y Control de Acopio de Cacao El Ceibo LTDA[Ergue,2007]

## **1.3. PROBLEMA**

### **1.3.1 IDENTIFICACIÓN DEL PROBLEMA**

El proceso de re-refinamiento tiene un control inestable, puesto que los obreros son los únicos que controlan el avance de producción, y la gerencia administra solamente el ingreso de insumos y producto final, no existe informes con respecto a la productividad por lo cual la gerencia también se ve afectada en este aspecto, entonces el proceso de refinado en sus etapas intermedias solamente queda a cargo por parte de los obreros que son asignados arbitrariamente, y los insumos junto con el material utilizado quedan expuestos a pérdidas no controladas, ya sea por que no se haya calculado bien la cantidad de material a utilizar, no se sepa que cantidad de insumos en el proceso mismo es el que se esta utilizando, el de stock. La

existencia del material e insumos registrados no sea la que se esperaba, y estos están archivados en archivos de forma manual, entonces no existe un control exacto de los insumos de entrada y el producto terminado.

### 1.3.2 LISTA DE PROBLEMAS

En base a estudios realizados que se manejan en la gerencia de RAMSOIL se identifican los siguientes problemas:

- No se maneja información sobre el rendimiento en las etapas de decantación, deshidratación, acidez y decoloración.

En las etapas de decantación que purifica el aceite de sustancias impuras, deshidratación que trata de filtrado de agua, acidez que trata de colocación de componentes químicos (ácidos), y decoloración que es el proceso de coloración y prensado del aceite, están encargados los obreros.

- Pérdidas de insumos no controlados en las etapas de deshidratación y acidez
- Los reportes de producto final son entregados de forma verbal por los obreros a cargo en la etapa final.
- Los obreros son asignados arbitrariamente para cada proceso.
- La información de los insumos utilizados en proceso y de stock está almacenada en files sin un estándar de clasificación.
- Fallas en cálculo de volúmenes de insumos para la producción.
- Toma de decisiones arbitrarias con respecto a procesos posteriores, basadas en evaluaciones al tanteo en función de: Insumos vs. Producto final.

Ver Anexo A (Árbol de problemas)

### 1.4. OBJETIVO GENERAL

Implementar el Sistema de Control del Proceso de re-refinado de Aceite para la empresa RAMSOIL y que éste coadyuve a mejorar el control y seguimiento en el proceso de re-refinado.

## 1.5. OBJETIVOS ESPECÍFICOS

- Implementar un control que refleje la asociación de insumos intermedios con sus respectivas salidas.
- Realizar módulos de seguimiento y control para cada etapa en el proceso de re-refinado de aceite.
- Implementar la Programación Lineal para el cálculo de volúmenes de producción, y costos.
- Generar informes exactos con respecto a la utilización de insumos y que ayuden a la toma de decisiones.
- Generar reportes sobre la productividad y resultados de las distintas etapas de producción.
- Establecer políticas de control de insumos en la productividad.
- Establecer patrones y realizar el control de asignación de obreros a procesos.
- Implementación del Sistema mediante intranet.

Ver Anexo B (Árbol de objetivos)

## 1.6. LÍMITES Y ALCANCES

En el presente proyecto se desarrolla el análisis, diseño e implementación del Sistema de control y Seguimiento para el proceso de re-refinado de aceite, y que éste contempla la unidad de producción de la fábrica de aceites RAMSOIL.

Se contemplan los siguientes puntos:

- Se tomará en cuenta principalmente el control en el volumen e insumos, de la producción en general constituidas en sus cuatro etapas : decantación, deshidratación, acidez, decoloración
- El control de los obreros se lo realizará en el ámbito laboral, es decir se tomará en cuenta el control y función de cada obrero, dentro de cada etapa del proceso de re-refinado.

- Se controlará las salidas de stock que se utiliza en almacenes para el control de los insumos.

En cuanto a los alcances el sistema debe proporcionar las siguientes salidas de información:

- Generar automáticamente el cálculo de volúmenes en los insumos, materia prima, para cada etapa de producción, y en sus respectivas salidas.
- Generar automáticamente la cantidad de volumen aproximado de aceite como producto final, tomando en cuenta parámetros de entrada en la etapa de inicio.
- Imprimir reportes sobre los insumos utilizados en producción y en stock.
- Imprimir reportes sobre la productividad para cada etapa.
- Generar reportes sobre los obreros y sus respectivas labores dentro de cada proceso.

### **1.7. APORTE**

El aporte que se desea brindar con el presente proyecto es el de automatizar las etapas del proceso de re-refinamiento, maximizando la productividad y generando información que coadyuve a la fácil y correcta toma de decisiones de la empresa.

Todo esto a través de la aplicación de Programación Lineal que nos servirá de herramienta para el cálculo de costos mínimos, costos máximos, flujos, para cada etapa en el proceso de refinado.

El usuario contará con una herramienta de automatización a la medida de sus requerimientos, para un óptimo control de los procesos en el área operativa y de producción.

Sistematización del proceso de refinado de aceite de acuerdo a los requerimientos, generando mayor control y efectividad en la productividad .

## **1.8. JUSTIFICACIÓN**

### **1.8.1 JUSTIFICACIÓN TÉCNICA**

El presente proyecto modifica los tradicionales flujos de información que se maneja dentro de la empresa , imponiendo así el uso de la tecnología informática y comunicacional, también tiene la ventaja que las herramientas de hardware y software a utilizarse están siendo tomadas en cuenta según las exigencias de la empresa.

### **1.8.2 JUSTIFICACIÓN OPERATIVA**

La facilidad de consulta sobre el avance en la producción, ayudará en la toma de decisiones gracias a los informes estandarizados y oportunos, donde la gerencia técnica podrá ver la productividad en función de los insumos, recursos de materia prima, tiempo, producto final, control de procesos.

### **1.8.3 JUSTIFICACIÓN ECONÓMICA**

El desempeño en la productividad influye en la pérdida o ganancia de dinero. Un sistema de control y seguimiento, permitirá que la empresa optimice todo el proceso de refinamiento, puesto que habrá un mejor control en cada etapa de la producción haciendo una administración y uso adecuado de insumos, recursos humanos, productos intermedios, producto final.

## **1.9 MÉTODOS DE INVESTIGACIÓN Y METODOLOGÍA**

Para el desarrollo del proyecto se utilizarán las siguientes herramientas de análisis:

- Árboles analíticos, en el que se describieron los problemas y objetivos
- Marco lógico, que resume en su totalidad la esencia del proyecto, y éste siendo un enfoque metodológico de mayor uso en diseño, ejecución y evaluación de proyectos de desarrollo, por ello aplicamos ésta perspectiva en el presente proyecto (Ver Anexo C).
- El método científico como procedimiento o conjunto de procedimientos que se utiliza para obtener conocimientos científicos, el modelo de trabajo o pauta general que



orienta la investigación, utilizando marco teórico como parte referencial y didáctica, árbol de problemas y objetivos.[Rosen,1998]

La metodología a usarse en el desarrollo del presente proyecto es el siguiente:

- Proceso Racional Unificado(RUP), siendo éste un proceso iterativo incremental basado en el análisis y diseño Orientado a Objetos, utiliza la notación del Modelo de Lenguaje Unificado(UML).[Jacob,2000]
- Lenguaje UML utilizado para el diseño del sistema. [Larma,1999]

Dentro de las herramientas a utilizar se tiene:

Lenguaje de programación PHP, lenguaje que se adecua a la programación de aplicaciones Web. Como gestor de Base de datos Mysql, de distribución libre y Apache como servidor de páginas web. Todo bajo el sistema operativo Windows.

## 1.10. ANÁLISIS COSTO BENEFICIO

### 1.10.1 ANÁLISIS DE COSTOS

En la siguiente tabla se mostrará el monto aproximado de cuanto se tendría que gastar en la realización del sistema, entonces el total reflejaría el costo del sistema en general.

COSTO	MONTO \$us	TOTAL
<b>Costo de Construir el Sistema</b>		
Analista de Sistemas	70	70
<b>Costo de Implementar el Sistema</b>		
Capacitación	50	
Realización de Base de Datos	50	
Gastos de Equipo	10	
Gastos de Instalación		110
<b>Costos Operacionales</b>		
Hardware y Software	800	
Mantenimiento	20	
<b>Total de Costos</b>		\$1000

**Tabla 1.1** Costos de la empresa con respecto al sistema a realizar.

## 1.10.2 ANÁLISIS DE BENEFICIOS

### Beneficios Tácticos

El sistema que se implementará podrá dar mas información con respecto al avance de producción, insumos y obreros, esto no quiere decir que se dependa en su totalidad del sistema de información para poder realizar la producción de aceite de un día en la empresa, mas bien sirve de apoyo a la hora de tomar de decisiones por parte de la gerencia.

### Beneficios estratégicos del nuevo sistema

Se tendría la capacidad de proporcionar información que anteriormente no se tenía, permitiendo la realización de consultas según se requiera, así como el cálculo de la cantidad de material e insumos a utilizar en cada proceso.

### Expresión de costos y beneficios del sistema

El estudio de costos y beneficios , orientado a cuatro años pudiera producir una tabla como esta.

	<b>Año1</b>	<b>Año2</b>	<b>Año 4</b>	<b>Total</b>
<b>Ganancia</b>	0	10000	20000	40000
<b>Gastos</b>	20000	15000	10000	5000
<b>Efectivo Neto</b>	-20000	-5000	10000	35000

**Tabla 1.2.** Ganancias proyectadas de la empresa a consecuencia de la implementación del sistema

## Gráfica Costo/Beneficio

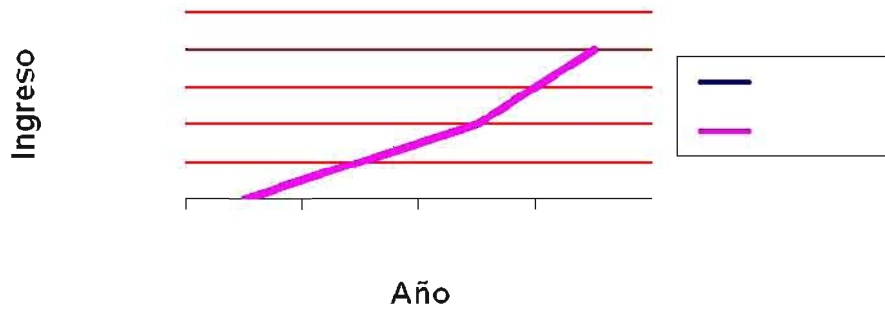


Figura. 1.1 Costo / Beneficio, proyecta beneficio a mediano y largo plazo

Fuente: Elaboración propia





---

**CAPÍTULO II**  
**MARCO TEÓRICO**

## CAPITULO II

### MÁRCO TEÓRICO

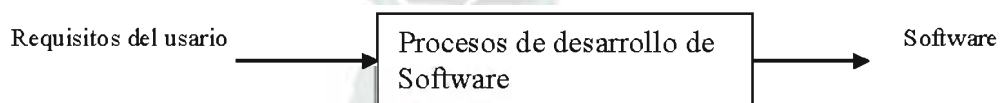
#### 2.1 INTRODUCCIÓN

En este capítulo se introducen los conceptos más relevantes sobre las metodologías, métodos y herramientas utilizadas para el desarrollo del presente proyecto de grado, pero no se puede dar una teoría completa es así que se trata de presentar una base para su fácil comprensión

#### 2.2 PROCESO UNIFICADO DE RATIONAL (RUP)

Un proceso define quien está haciendo que, y cuando, además dice como alcanzar un determinado objetivo. En la ingeniería de software el objetivo es construir un producto de software [Jacob,2000], vale decir, que todos los proyectos necesitan de un proceso que guíe sus actividades.

Según Jacobson en sus libros “El procesos Unificado de desarrollo de Software”, unos procesos efectivos proporcionan normas para el desarrollo eficiente de Software de calidad, captura y presenta las mejores practicas que la tecnología permite. Por tanto reduce el riesgo y hace el proyecto más predecible(ver Figura. 2.1).



**Figura 2.1** Proceso de Desarrollo de Software

**Fuente:** [Jacob, 2000]

Entre muchos investigadores de la orientación a objetos hay tres autores que se han destacado por sus contribuciones al uso del paradigma en todo el proceso de desarrollo: Ivar Jacobson, Grady Booch y James Rumbaugh. Luego de muchos años de trabajo individual desarrollado y difundido sus propios métodos, han unido sus teorías y su experiencia, y se han puesto a la

cabeza de un formidable grupo de investigadores para contribuir dos herramientas con las cuales buscan estandarizar y por ende facilitar el uso de los objetos en la programación: El lenguaje Unificado de Modelo (UML Unified Modeling Language) y el proceso unificado rotacional para el desarrollo d programas(RUP, Rational Unified Process )mientras que UML, es ya un lenguaje maduro que ha logrado la aceptación de amplios sectores de las industria y la academia, RUP sigue siendo aún una propuesta que deberá depurarse y templarse al calor de la experiencia de su aplicación en el campo y los portes de los casos de estudio[Jacob,2000] (ver Figura. 2.2).

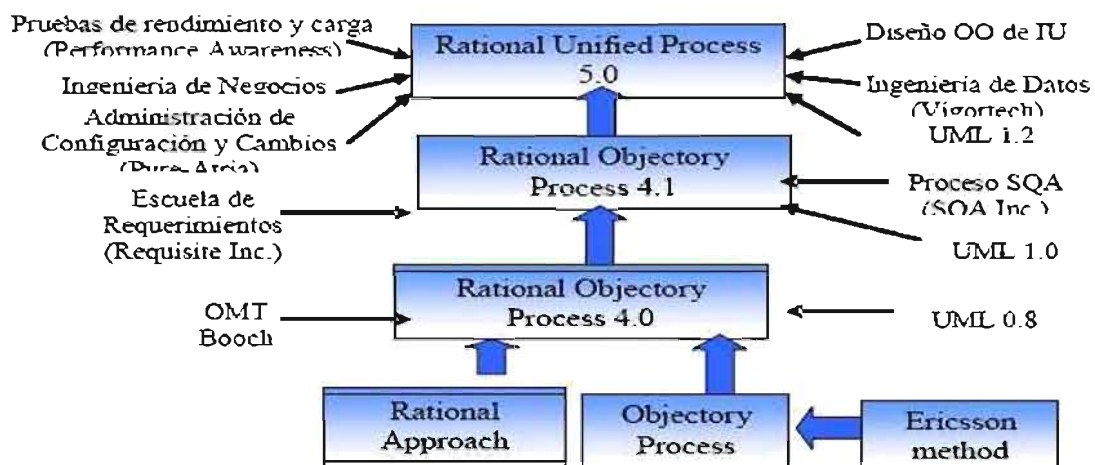


Figura 2.2 Historial de Procesos Unificados

Fuente: [Jacob, 2000]

RUP y UML están estrechamente relacionados entre si, pues mientras el primero establece las actividades y los criterios para conducir un sistema desde su máximo nivel de abstracción, el segundo ofrece la notación gráfica necesaria para representar los sucesos, modelos, que se obtienen de procesos de refinamiento.

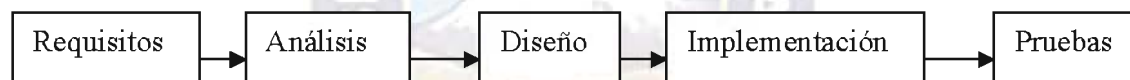
RUP se define como un procesos dirigido por:

- Casos de Uso
- Centrado en la Arquitectura
- Iterativo e Incremental

### 2.2.1 DIRIGIDO POR CASOS DE USO

Procesos de desarrollo de software utiliza los casos de uso como una herramienta para la obtención de requisitos de usuario. Donde los casos de uso son para definir la funcionalidad del sistema, y guían al desarrollador en la construcción del arquitectura del sistema.

La descripción obtenida de los requerimientos debe ser comprendida por casos de uso que nos ayudan a recopilar la información acerca de la interacción que tiene los usuarios en este caso actores con el sistema. Un caso de uso es una secuencia, reacciones que el sistema lleva a cabo para ofrecer un resultado de valor a algún actor, que sirven para realizar pruebas sobre los componentes desarrollados(ver Figura. 2.3). Los casos de uso enlazan los flujos de trabajo fundamentales. El proyecto progresa a través de estos flujos de trabajo, que inician en los casos de uso[Jacob 2000].



**Figura 2.3:** Casos de uso que enlaza los flujos de trabajo fundamental

**Fuente:** [Larma,1999]

### 2.2.2 CENTRADO EN LA ARQUITECTURA

En el caso de software la arquitectura se refiere a un conjunto de decisiones significativas acerca de la organización de un sistema, la elección de los elementos acerca de la organización de un sistema software, la selección de los elementos estructurales a partir de las cuales se componen el sistema con su respectivo comportamiento y las interacciones entre esos elementos y la composición de esos elementos estructurales.

La necesidad de una arquitectura radica en poder comprender el sistema, es decir que todos los que están involucrados con su desarrollo deben entender el problema al cual va enfocado el

sistema de software para satisfacer las demandas individuales y de la organización mediante la utilización de los diagramas definidos por UML.

La organización es un punto muy importante ya que cuanto mayor sea la organización del proyecto software mayor será la comunicación entre los desarrolladores para coordinar sus esfuerzos dividiendo el sistema en subsistemas definiendo las interfaces correctas de diseño.

Al conocer el dominio de problema y con que componentes se piensa en como conectar esos componentes para cumplir con los requisitos del sistema y realizar los modelos de casos de uso reutilizando dichos componentes.

En la arquitectura de la construcción, antes de construir un edificio, este se completa desde varios puntos de vista: estructura, condiciones eléctricas, fontanería, etc.

### 2.2.3 ITERATIVO E INCREMENTAL

Jacobson en su libro "El Proceso Unificado de Desarrollo de Software", explica que en esta fase proporciona la estrategia para desarrollar un producto de software en pasos pequeños manejables:

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

"Un ciclo de vida iterativo se basa en el agrandamiento y perfeccionamiento secuencial de un sistema a través de múltiples ciclos de desarrollo de análisis, diseño implementación y pruebas".

El modelo incremental entrega el software en partes pequeñas pero utilizables, llamadas "incrementos" [Press,1998]. En general, cada incremento se construye sobre aquel que ha sido entregado.



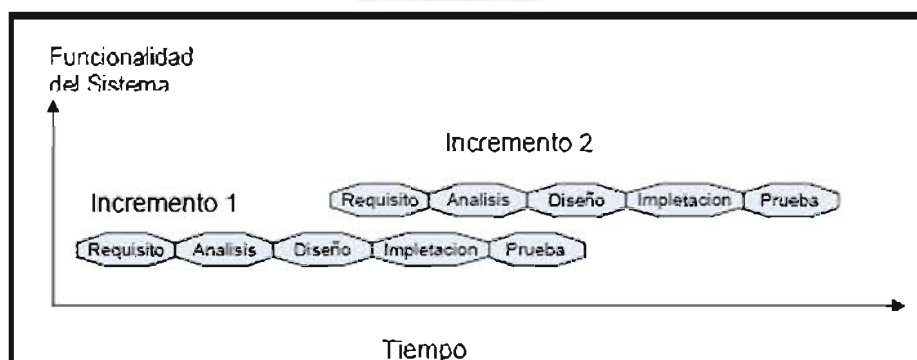
Las ventajas de un desarrollo de software con un ciclo de vida iterativo se dan gracias a la retroalimentación en cada ciclo por lo cual se crea un sistema más robusto. En cada incremento que y tiene el sistema se va perfeccionando aún mas, lo cual permite al usuario realizar las modificaciones requeridas en el transcurso del tiempo [Press,1998].

Todo sistema informático complejo supone un gran esfuerzo que puede durar desde varios meses hasta años, por lo tanto, lo mas practico es decidir en varias fases. Actualmente se puede hablar de ciclos de vida en los que se realiza varios recorridos por todas las fases.

Las ventajas de un desarrollo de software con un ciclo de vida iterativo se da gracias a la retroalimentación en cada ciclo por lo cual se crea un sistema mas robusto. En cada incremento que tiene el sistema se va perfeccionando aún mas, lo cual permite al usuario realzar las modificaciones requeridas en el transcurso del tiempo.

Todos los sistemas informáticos complejos suponen de un gran esfuerzo que puede durar desde varios meses hasta años; por lo tanto, lo mas practico es dividirla en varias fases.

Actualmente se puede hablar de ciclos de vida en los que se realizan varios recorridos por todas las fases. Cada recorrido por las fases se denomina iterativo del proyecto en la que se realizan varios tipos de trabajo(denominados flujos). Además cada iteración parte de la anterior incrementado o revisando la funcionalidad implementada. Se suele denominar proceso(ver Figura 2.4).



**Figura 2.4** Proceso Iterativo e Incremental

**Fuente:** [Ferrer,2005]

## 2.3 LENGUAJE UNIFICADO DEL MODELO

El lenguaje unificado de modelado (UML-Unifiet Modeling lenguaje), es una consolidación de muchas de las notaciones y conceptos más usadas en el desarrollo de análisis y diseño orientado a objetos. UML es la unificación y la consolidación del trabajo de Grade Booch, James Rumbaugh e Ivar Jacobson, es un lenguaje de modelado que tiene una notación gráfica para visualizar, especificar, construir y documentar cada una de las partes que comprende el desarrollo de Software.

Las técnicas en el UML, fueron diseñadas para la mejor comprensión del desarrollo orientado a objetos que se describen a continuación.

### 2.3.1 CASOS DE USO

Los casos de uso no son propiamente un caso de análisis, se limitan a describir procesos de dominio que pueden expresarse en forma narrativa en un formato estructurado de prosa y pueden se eficaces en un proyecto de tecnología no orientada a objetos. No obstante, constituyen un paso preliminar muy útil porque describen las especificaciones de un sistema [Jacob, 2000]

- a) **Actores.-** El actor es una entidad externa del sistema que de alguna manera participa en la historia del cado de uso. Por lo regular, estimula el sistema con eventos de entrada o recibe algo de él. Los actores están representados por el papel que desempeñan en el caso: Cliente, técnico u otro. Conviene escribir su nombre con mayúscula en la narrativa del caso para facilitar la identificación(ver Figura 2.5).

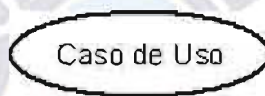


Actor

**Figura 2.5** Actor

**Fuente:** [Larma,1999]

b) **Caso de Uso.**- El caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso [Jacob, 2000]. Los casos de uso son historias o casos de utilización de un sistema, no son exactamente los requerimientos, ni las especificaciones funcionales, sino que ejemplifican e incluyen tácticamente los requerimientos en las historias que narran (ver Figura. 2.6).

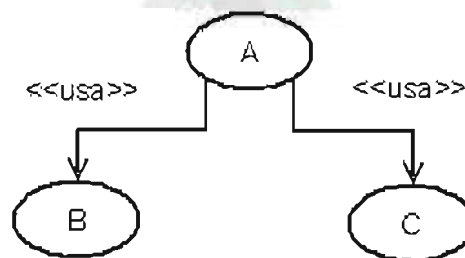


**Figura 2.6** Caso de Uso

**Fuente:** [Larma,1999]

c) **Relación.**- Si un caso de uso inicia o contiene el comportamiento de otro se dice que usa el segundo caso, eso es una relación unidireccional [Jacob 92]. Esta relación puede presentar uno de los siguientes tipos:

- La relación “usa”, se utiliza cuando se quiere reflejar un comportamiento común en varios casos de uso (ver Figura. 2.7).
- La relación “extiende”, se utiliza cuando se requiere reflejar un comportamiento opcional de un caso de uso, es decir, es cuando tiene un caso similar a otro, cuyo contexto tiene mucho más detalle.



**Figura 2.7** Relación de usos

**Fuente:** [Larma,1999]

### 2.3.1.1 DIAGRAMAS DE CASOS DE USO DE ALTO NIVEL

En un caso de uso descrito a alto nivel la descripción es muy general, normalmente se condensa en dos o tres frases. Es útil para comprender el ámbito y el grado de complejidad del sistema[Larma,1999] . De entre los varios formatos que existen para describir casos de uso se toma en cuenta el de la tabla 2.1

Caso de Uso	Nombre del caso de uso
Actores	Lista de actores participantes en el Caso de Uso
Tipo	1. Primario, secundario u opcional 2. Esencial o real.
Descripción	Breve explicación del Caso de Uso.

**Tabla 2.1** Formato de un caso de uso de alto nivel.

**Fuente**[Larma,1999]

### 2.3.1.2 DIAGRAMAS DE CASOS DE USO EXPANDIDO

Son casos expandidos que se expresan en una forma teórica que contiene poca tecnología y pocos detalles de implementación, las decisiones de diseño se posponen y se abstraen de la realidad, especialmente las concernientes a la interfaz para el usuario. Un caso de este tipo describe el proceso a partir de sus actividades y motivos esenciales. El grado de abstracción con que se describe existe en un continuo: “la descripción puede ser más o menos esencial” [Larma,1999].

Son esquemas de interacción entre el usuario y el sistema que ayudan a la comprensión del funcionamiento de este último, por lo que se ha visto que es útil aplicarlos. En cuanto a la notación se tiene el siguiente formato en la tabla 2.2.

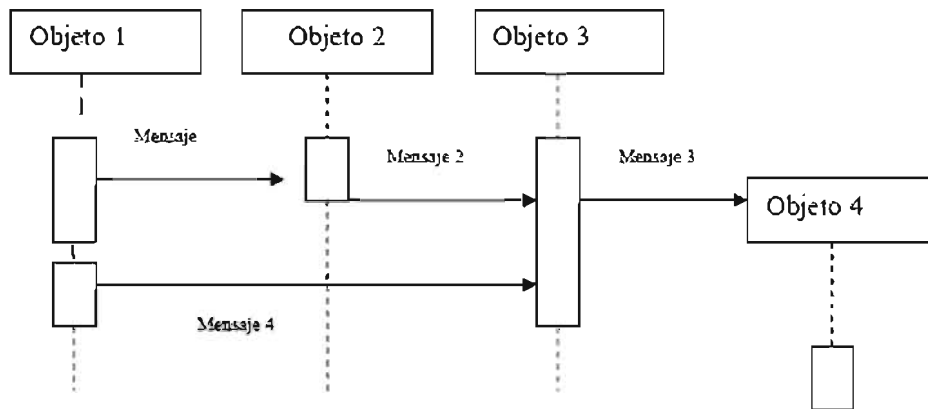
<b>Caso de Uso</b>	Nombre del caso de uso	
<b>Actores</b>	Lista de actores participantes en el caso de uso.	
<b>Propósito</b>	Intención del caso de uso	
<b>Descripción</b>	Repetición de caso de uso de alto nivel o alguna síntesis similar	
<b>Tipo</b>	1. primario, secundario u opcional. 2. esencial o real	
<b>Referencia Cruzada</b>	Casos de uso relacionados y funciones también relacionadas al sistema	
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	Acciones numeradas de los actores	Descripciones numeradas de las respuestas del sistema.

**Tabla 2.2** Formato de un caso de uso expandido.

**Fuente**[Larma,1999]

### 2.3.2 DIAGRAMA DE SECUENCIA

El diagrama de secuencia es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico del sistema de información haciendo énfasis en la secuencia de los mensajes intercambiados por los objetos [Jacob, 2000]. El diagrama de secuencia tiene dos dimensiones, el eje vertical representa el tiempo y el eje horizontal los diferentes objetos. El tiempo avanza desde la parte superior hacia el interior, cada objeto tiene asociado una línea de vida y focos de control. La línea de vida indica el intervalo de tiempo durante el que existe ese objeto. Un foco de control o activación muestra el periodo de tiempo (ver Figura. 2.8).

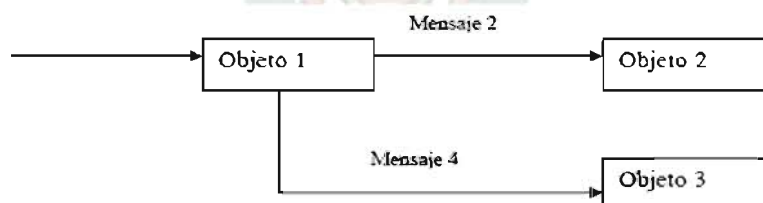


**Figura 2.8** Diagrama de Secuencia

Fuente: [Jacob, 2000]

### 2.3.3 DIAGRAMA DE COLABORACIÓN

El diseño orientado a objetos tiene por objeto las especificaciones lógicas del software que cumplan con los requisitos funcionales, un paso esencial de esta fase es la asignación de responsabilidades entre los objetos y mostrar como interactúan a través de mensajes, expresados en diagramas de colaboración que son un tipo de diagrama, cuyo objetivo es describir el comportamiento dinámico del sistema de información, cómo interactúan los objetos entre sí, es decir con que otros objetos tiene vínculos o intercambian mensajes a un determinado objeto [Jacob, 2000]. Un diagrama de colaboración muestra la misma información que un diagrama de secuencia, pero de forma diferente. En los diagramas de colaboración coexiste una secuencia temporal en el eje vertical, es decir la colocación de los mensajes en el diagrama no indica cual es el orden en el que se sucede. Además la colocación de los objetos es más flexible y permite mostrar de forma más clara cuáles son la colaboraciones entre ellos(ver Figura. 2.9).



**Figura 2.9** Diagrama de Colaboración

Fuente: [Jacob, 2000]

### 2.3.4 DIAGRAMA DE CLASES

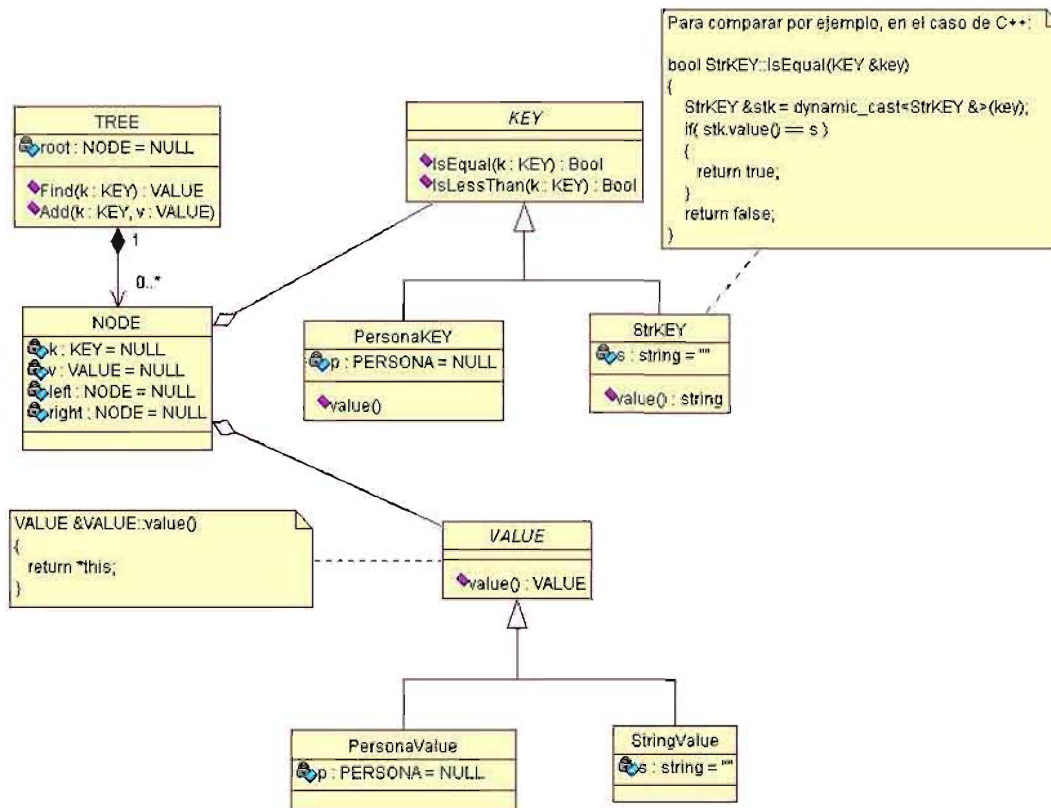
Es la representación de los aspectos estáticos del sistema, utilizando diversos mecanismos de abstracción (clasificación, generalización, agregación). El diagrama de clases recoge las clases de objetos y sus asociaciones. En este diagrama se representa la estructura y el comportamiento de cada uno de los objetos del sistema y sus relaciones con los demás objetos, pero no muestran información temporal [Ferre, 2005].

Se advierten los siguientes tipos de relación:

1. La Asociación que representan un conjunto de enlaces entre objetos o instancias de clases.
2. Herencia que indica que una subclase hereda los métodos y atributos especificados por una superclase hereda los métodos por ende la subclase, además de poseer sus propios métodos y atributos poseerá las características y atributos visibles de la superclase.
3. La agregación que es un tipo de relación jerárquica entre un objeto que representa la totalidad de ese objeto y las partes que la componen. Permite el agrupamiento físico de estructuras relacionadas lógicamente.

En el siguiente ejemplo se muestran las distintas clases relacionadas entre si:

- Se muestra la asociación entre la clase TREE (árbol) y la clase NODE (nodo), es decir la asociación que existe entre ambas, y la cardinalidad de uno a muchos que significa que un árbol tiene n nodos.
- La herencia que se da entre las clases Personavalue, StringValue, cuyo padre sería la clase VALUE, dando a entender que existe un valor (value) en común que puede heredarse de la clase padre. Y de la misma forma se da con las clases KEY, PersonaKey, StrKey
- Cabe la importancia de notar que en la clase PersonaKey y StrKey están denotados los métodos y atributos que debería tener toda clase.



**Figura 2.10** Diagrama de Clases

Fuente: [Ferre, 2005]

## 2.4 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS

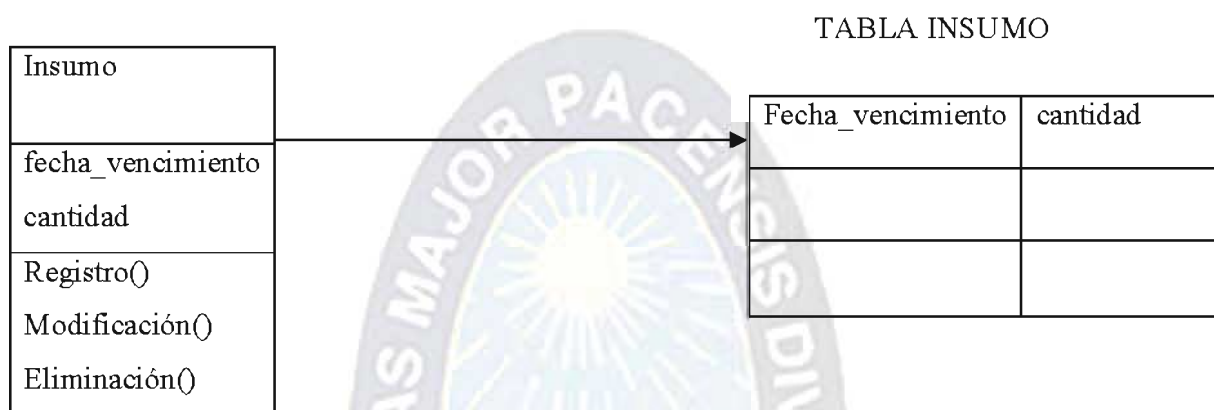
Consiste en determinar el esquema de base de datos que se utilizará, en este caso se vio conveniente el estudio de bases de datos objeto-relacionales.

En la generalidad de las aplicaciones es necesario guardar y recuperar la información en un mecanismo de almacenamiento persistente, una base de datos relacional por ejemplo. Dado el predominio de estas últimas, a menudo se requiere su uso en vez de otras bases más manejables orientadas a objetos. De ser así, surgen varios problemas a causa de la desigualdad entre representaciones de datos orientadas a registros, y las que se orientan a objetos, se requieren servicios especiales de ambos tipos en las bases de datos relacionales [Larma, 1999].



¿Cómo mapear in objeto a un archivo o a un esquema de base de datos relacional?

El patrón de representación de objetos a tablas propone definir una tabla para cada clase de objeto persistente. Los atributos de objetos que contienen tipos primitivos de datos (número, booleano, cadena y otros) se mapean en las columnas.[Larma,1999]. La figura 2.11 ilustra este caso.



**Figura 2.11** Representación de objetos como tablas.

**Fuente:** [Ferre, 2005]

Conviene contar con un medio que relacione los objetos con los registros y de asegurarse de que la repetición de la materialización de un registro no culmine en la duplicación de objetos. El patrón identificador de objetos (IDO) se propone asignar un IDO a cada registro y objeto (o al agente de un objeto). Un identificador de objetos suele ser un valor alfanumérico, es único para un objeto en específico, toda tabla de base de datos relacional tiene un IDO como clave primaria, y los objetos también contarán (directa o indirectamente) con un identificador[Larma,1999].

Si todos los objetos se asocian a un IDO y si todas las tablas poseen una clave primaria IDO los objetos podrán mapearse de modo singular en una región de alguna tabla.

¿Cómo representar las relaciones de objetos en una tabla de una base de datos relacional?. La respuesta se da en el patrón de representación de las relaciones de objetos como tablas que propone lo siguiente:

- Asociación uno a uno
  - o Colocar una clave foránea de identificador de objetos en una o en las dos tablas que representan los objetos en la relación.
  - o Crear una tabla asociativa que registre los identificadores de objetos de cada relación.
- Asociaciones de uno a muchos, una colección por ejemplo
  - o Crear una tabla asociativa que registre los identificadores de cada objeto en la relación.
- Asociaciones de muchos a muchos
  - o Crear una tabla asociativa que registre todos los identificadores de objetos en la relación

## 2.5 PROGRAMACIÓN LINEAL

Mucha gente sitúa el desarrollo de la programación lineal entre los avances científicos más importantes de la mitad del siglo XX, y debemos estar de acuerdo con esta afirmación si tenemos en cuenta que su impacto desde 1950 ha sido extraordinario. Se han escrito decenas de libros de texto sobre la materia y los artículos publicados que describen aplicaciones importantes se cuentan ahora por cientos. De hecho, una proporción importante de todo el cálculo científico que se lleva a cabo en computadoras se dedica al uso de la programación lineal y a técnicas íntimamente relacionadas. (Esta proporción se estimó en un 25%, en un estudio de la IBM) [Thale,2007]..

Un modelo de programación lineal proporciona un método eficiente para determinar una decisión óptima, (o una estrategia óptima o un plan óptimo) escogida de un gran número de decisiones posibles.

En todos los problemas de Programación Lineal, el objetivo es la maximización o minimización de alguna cantidad.

## 2.5.1 MÉTODO SIMPLEX

De forma obligatoria se deben cumplir los siguientes requerimientos para construir un modelo de Programación Lineal basándonos en el método del simplex específicamente [Thale,2007].

**Requerimiento 1.** Función objetivo. (F.O).

Debe haber un objetivo (o meta o blanco) que la optimización desea alcanzar.

**Requerimiento 2.** Restricciones y decisiones.

Debe haber cursos o alternativas de acción o decisiones, uno de los cuáles permite alcanzar el objetivo.

**Requerimiento 3.** La F.O y las restricciones son lineales.

Deben utilizarse solamente ecuaciones lineales o desigualdades lineales.

Modelo standard de Programación Lineal

Optimizar  $Z = C_1X_1 + C_2X_2 + \dots + C_n X_n$ . Función objetivo.

Sujeta a  $a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n \leq b_1$

$a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n \leq b_2$

Restricciones

$a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n \leq b_m$

Debiendo ser

$X_1 \geq 0, X_2 \geq 0, \dots, X_n \geq 0$

Donde :

$X_j$  : variables de decisión,  $j = 1, 2, \dots, n$ .

$n$  : número de variables.

$m$  : número de restricciones.

$a_{ij}, b_i, c_j$  constantes,  $i = 1, 2, \dots, m$ .

Pasos para la construcción del modelo

- Definir las variables de decisión.
- Definir el objetivo o meta en términos de las variables de decisión.
- Definir las restricciones.
- Restringir todas las variables para que sean no negativas.

En infinidad de aplicaciones de la industria, la economía, la estrategia militar, y en todos donde existan actividades que requieran modelos. se presentan situaciones en las que se exige maximizar o minimizar algunas funciones que se encuentran sujetas a determinadas limitaciones, que llamaremos restricciones.

Para hacernos una idea más clara de estos supuestos, veamos dos ejemplos[Isfti,2000] :

Ejemplo 1: Problema de máximos.

En una granja se preparan dos clases de piensos, P y Q, mezclando dos productos A y B. Un saco de P contiene 8 kg de A y 2 de B, y un saco de Q contiene 10 kg de A y 5 de B. Cada saco de P se vende a 300 ptas. y cada saco de Q a 800 ptas. Si en la granja hay almacenados 80 kg de A y 25 de B, ¿cuántos sacos de cada tipo de pienso deben preparar para obtener los máximos ingresos?

Ejemplo 2: Problema de mínimos.

Una campaña para promocionar una marca de productos lácteos se basa en el reparto gratuito de yogures con sabor a limón o a fresa. Se decide repartir al menos 30000 yogures. Cada yogur de limón necesita para su elaboración 0.5 gramos de un producto de fermentación y cada yogur de fresa necesita 0.2 gramos de este mismo producto. Se dispone de 9 kilogramos.

El coste de producción de un yogur de limón es de 30 pesetas y 20 pesetas uno de fresa.

En los dos ejemplos descritos está claro que tanto la cantidad que deseamos maximizar como la cantidad que deseamos minimizar podemos expresarlas en forma de ecuaciones lineales.

Por otra parte, las restricciones que imponen las condiciones de ambos problemas se pueden expresar en forma de inecuaciones lineales.

Tratemos de plantear en términos matemáticos los dos ejemplos anteriores:

1) Si designamos por  $x$  al número de sacos de pienso de clase P y por  $y$  el número de sacos de pienso de clase Q que se han de vender, la función :  $Z = 300x + 800y$  representará la cantidad de pesetas obtenidas por la venta de los sacos, y por tanto es la que debemos maximizar.

Podemos hacer un pequeño cuadro que nos ayude a obtener las restricciones:

	Nº	kg de A	kg de B
P	$x$	$8x$	$2x$
Q	$y$	$10y$	$5y$
		$\leq 80$	$\leq 25$

Por otra parte, las variables  $x$  e  $y$ , lógicamente, han de ser no negativas, por tanto :  $x \geq 0$ ,  $y \geq 0$

Conjunto de restricciones:

$$8x + 10y \leq 80$$

$$2x + 5y \leq 25$$

$$x \geq 0, y \geq 0$$

2) Si representamos por  $x$  el número de yogures de limón e  $y$  al número de yogures de fresa, se tiene que la función de coste es  $Z = 30x + 20y$ .

Por otra parte, las condiciones del problema imponen las siguientes restricciones:

- De número :  $x + y \leq 80$
- De fermentación:  $0.5x + 0.2y \leq 9000$
- Las variables  $x$  e  $y$  han de ser, lógicamente, no negativas; es decir:  $x \geq 0$ ,  $y \geq 0$

Conjunto de restricciones:

$$x + y \leq 80$$

$$0.5x + 0.2y \leq 9000$$

$$x \geq 0, y \geq 0$$

Un problema de programación lineal en dos variables, tiene la siguiente formulación estándar:

$$\left\{ \begin{array}{l} \text{Maximizar } z = f(x, y) = ax + by + c \\ \text{sujeto a: } \quad a_1x + b_1y \leq c_1 \\ \quad \quad \quad a_2x + b_2y \leq c_2 \\ \quad \quad \quad \vdots \\ \quad \quad \quad a_nx + b_ny \leq c_n \end{array} \right.$$

pudiendo cambiarse maximizar por minimizar, y el sentido de las desigualdades.

En un problema de programación lineal intervienen [Isfti,2000]:

- La función  $f(x,y) = ax + by + c$  llamada función objetivo y que es necesario optimizar. En esa expresión  $x$  e  $y$  son las variables de decisión, mientras que  $a$ ,  $b$  y  $c$  son constantes.
- Las restricciones que deben ser inecuaciones lineales. Su número depende del problema en cuestión. El carácter de desigualdad viene impuesto por las limitaciones, disponibilidades o necesidades, que son: inferiores a ... ( menores:  $<$  o  $\leq$ ); como mínimo de ... ( mayores:  $>$  o  $\geq$  ). Tanto si se trata de maximizar como de minimizar, las desigualdades pueden darse en cualquiera de los dos sentidos.
- Al conjunto de valores de  $x$  e  $y$  que verifican todas y cada una de las restricciones se lo denomina conjunto (o región ) factible. Todo punto de ese conjunto puede ser solución del problema; todo punto no perteneciente a ese conjunto no puede ser solución. En el apartado siguiente veremos como se determina la región factible.
- La solución óptima del problema será un par de valores  $(x_0, y_0)$  del conjunto factible que haga que  $f(x,y)$  tome el valor máximo o mínimo.

## 2.6 MÉTRICAS DE CALIDAD

Los desarrolladores de software más hastiados estarán de acuerdo que el software de alta calidad es una de las metas más importantes. La calidad del software es una compleja mezcla de factores que varían a través de diferentes aplicaciones y según el cliente que las pida [Press,1998].

La calidad del software se define también, en términos de ausencia de errores en el funcionamiento del sistema. El ajuste a las necesidades del usuario, el sistema debe ser flexible u susceptible a modificaciones que se puedan realizar de manera rápida y oportuna. El sistema debe alcanzar un desempeño apropiado en términos de tiempo, volumen y espacio. Un sistema debe cumplir de la mejor forma los estándares internacionales establecidos, en lo que a la calidad de software se refiere.

### 2.6.1 FUNCIONALIDAD

El grado en que el software satisface las necesidades. Los puntos de función (PF) se describen como medidas básicas desde donde se calculan métricas de productividad. Los datos de PF se utilizan de dos formas durante la estimación del proyecto software:

- Como una variable de estimación que se utiliza para dimensionar cada elemento del software.
- Como métricas de línea base recopilada de proyectos anteriores, y utilizados junto con variables de estimación para desarrollar proyecciones de costo y esfuerzo.

Para estimaciones de PF, la descomposición funciona de la siguiente manera (ver Tabla 2.3):

Dominio de Información	Descripción
Número de entradas de usuario	Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.
Número de salidas de usuario	Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, y demás. Los elementos de datos particulares

	dentro de un informe no se cuentan de forma separada.
Número de archivos	Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).
Número de interfaces externas	Se cuenta todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de disco) que se utilizan para transmitir información a otros sistemas.

**Tabla 2.3** Dominios de información de Punto Función.

**Fuente:** [Press,1998]

Los puntos de función se calculan completando la Tabla 2.4

Parámetros de medición	Cuenta	Factor de Ponderación			Resultado
		Simple	Medio	Complejo	
Número de entradas de usuario	N1	3	4	6	N1 *factor
Número de salidas de usuario.	N1	4	5	7	N2 *factor
Número de peticiones de usuario.	N1	3	4	6	N3 *factor
Número de archivos.	N1	7	10	15	N4 *factor
Número de interfaces externas.	N1	5	7	10	N5 *factor
<b>CUENTA TOTAL</b>					$\Sigma(Ni*factor)$

**Tabla 2.4** Cálculos de los Punto Función.

**Fuente:** [Press,1998]

Para calcular los PF, se utiliza la relación siguiente:

$$PF = CUENTA TOTAL * (0.65 + 0.01 * \Sigma(Fi))$$

CUENTA TOTAL = Sumatoria de todas las entradas de la Tabla .2.4.



Fi = Son valores de ajuste a la complejidad según las respuestas a las preguntas siguientes.

1. ¿Requiere el sistema copias de seguridad y recuperación flexible?
2. ¿Se requiere comunicación de datos?
3. ¿Existen funciones del procedimiento distribuido?
4. ¿Es crítico el rendimiento?
5. ¿Se ejecutara el sistema en un entorno operativo existente y fuertemente utilizado?
6. ¿Requiere el sistema entrada de datos interactiva?
7. ¿Requiere la entrada de datos interactiva que las transiciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?
8. ¿Se actualiza los archivos maestros de forma interactiva?
9. ¿Son complejas las entradas, las salidas, los archivos y las peticiones?
10. ¿Es complejo el procesamiento interno?
11. ¿Se ha diseñado el código para ser reutilizable?
12. ¿Están concluidas en el diseño la conversión y la instalación?
13. ¿Se ha desarrollado el sistema para soportar múltiples instalaciones en diferentes organizaciones?
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizado por el usuario?

Cada una de las preguntas, es respondida usando una escala con rangos desde 0 (no importante), hasta 5 (absolutamente esencial).

## **2.6.2 CONFIABILIDAD**

Cantidad de tiempo que el software está disponible para su uso. La confiabilidad está definida como la probabilidad del sistema libre de fallos en un contexto determinado y durante un periodo de tiempo [Plati,2003] .

La confiabilidad se expresa como una probabilidad, esta se encuentra en una escala de 0 a 1. Si la probabilidad se acerca a uno, mas confiabilidad brindará el sistema. Para el usuario, el sistema será mas confiable mientras presenta menos errores.

La confiabilidad de un sistema se calcula mediante la siguiente función:

$$R(t) = e^{-\lambda P(T)} \quad \text{donde } T > 1$$

$R(t)$ : Confiabilidad del sistema

$\lambda$  : Probabilidad de que se presente una perturbación

$P(T)$ : Probabilidad de fallo en caso de perturbación.

Para medir la confiabilidad, se necesita elaborar un diagrama de transferencia, el cual se detalla todos los módulos que la conforman. Una vez elaborado el diagrama, se procede el cálculo de  $R(t)$ , teniendo en cuenta los siguientes teoremas:

**T1.**  $n$  componentes que están conectados en serie y el  $i$ -ésimo tiene confiabilidad  $R_i(t)$ , entonces la confiabilidad del sistema completo,  $R_i(t)$  está dado por:

$$R_i(t) = R_1(t) * R_2(t) * \dots * R_n(t)$$

**T2.**  $n$  componentes que trabajen en paralelo y el  $i$ -ésimo tiene  $R_i(t)$ , entonces la confiabilidad del sistema en su totalidad, está dado por:

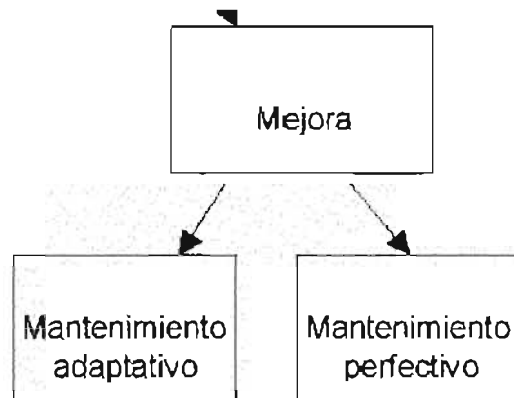
$$R_i(t) = 1 - [1 - R_1(t)] * [1 - R_2(t)] * \dots * [1 - R_n(t)]$$

### 2.6.3 PORTABILIDAD

La portabilidad de un sistema de información, se define como la factibilidad de transferir un producto a diferentes entornos de hardware/software, sin necesidad de aplicar acciones o mecanismos distintos. También es considerado como la capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito dentro del mismo entorno. Las características más importantes que se consideran para este factor son: la facilidad de instalación, facilidad de ajuste y facilidad de adaptación al cambio [Plati, 2003].

#### 2.6.4 MANTENIBILIDAD

Dentro de los tipos de mantenimiento para la mejora, se muestra dos: el mantenimiento adaptativo y el mantenimiento perfectivo [Juan C,1995], como se muestra en la figura 2.10



**Figura 2.12** Clasificación de tipos de mantenimiento

**Fuente:** [Juan C,1995]

- **Adaptivo.** Modificación de un producto software, después de su entrega, para conseguir que sea utilizable en un nuevo entorno.
- **Perfectivo.** Modificación de un producto software después de su entrega, para mejorar su rendimiento o su mantenibilidad.

#### 2.7 SEGURIDAD DEL SISTEMA

Intranet es una infraestructura basada en los estándares y tecnologías de Internet que soporta el compartir información dentro de un grupo bien definido y limitado [Geoci,2000].

Aunque una Intranet sea una red privada en la que se tengan grupos bien definidos y limitados ésta no se encuentra exenta de ataques que pudiesen poner en riesgo la información que maneja, ya que la mayoría de éstos son provocados por sus mismos usuarios.

La mayoría de las estadísticas de seguridad en cómputo indican que cerca del 80% de los fraudes relacionados con las computadoras provienen de los usuarios internos, por esto las intranets son las más vulnerables a ataques de ésta índole.

Por tal es importante establecer normas y políticas de seguridad a fin de tener un sistema confiable, y seguro. Un punto muy importante dentro de las políticas es el que tienen que ir acompañadas de sanciones, las cuales deberán también ser redactadas, revisadas, autorizadas, aplicadas y actualizadas[Geoci,2000] .

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





---

**CAPÍTULO III**  
**ANÁLISIS Y DISEÑO**

## CAPITULO III

### ANÁLISIS Y DISEÑO

#### 3.1 INTRODUCCIÓN

El análisis del " Sistema de Control y Seguimiento para el Tratamiento de Aceites ", hace uso de los instrumentos métodos y técnicas descritas en el capítulo anterior "Marco teórico" brindando a la empresa el soporte administrativo para la toma de decisiones que se aduce a sus necesidades tales como el control en los distintos procesos de refinado de aceite.

El análisis sigue los lineamientos del Proceso Unificado de Desarrollo de Software (RUP), utilizando el Lenguaje de Modelado Unificado (UML) representándose todos los esquemas de un sistemas de software de acuerdo los modelos descritos por RUP.

#### 3.2 DESCRIPCIÓN DE LOS ACTORES

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

A continuación se da una lista de los actores o usuarios identificados.

**Gerente General** Es la persona que necesita estar mas informado teniendo un control y seguimiento de las actividades de la empresa en especial en la producción de aceites tomando en cuenta los productos intermedios que salen de esta. Sus funciones son:

- Aprueba y da viabilidad a las compra de insumos.
- Aprueba y realiza la venta del aceite a los clientes establecidos.
- Evalúa resultados intermedios sacados en el proceso de refinado.

**Secretaria** Se encarga del trato con proveedores para las compras de insumos, y también con los clientes para las ventas, además:

- Recibe solicitudes de compra, y tiene potestad de realizar las mismas, en coordinación con Gerencia General.
- Controla en almacenes el stock de insumos.
- Realiza la venta a clientes en coordinación con Gerencia general.
- Emite reportes generales.

**Jefe de Producción** Se encarga del control de la producción en general y también tiene potestad sobre la asignación de funciones a obreros, y su rendimiento, también:

- Realiza el control de insumos en cada etapa de la producción del refinado.
- Controla el volumen de los productos intermedios.
- Emite reportes sobre la producción diaria.

**Encargado de Almacenes** Se encarga de proporcionar materiales para el proceso de refinado, en todas sus etapas, y:

- Da viabilidad a la solicitud de pedidos dentro de la empresa.
- Recepciona y controla entradas y salidas de insumos para la empresa.
- Emite notas de entrega de los materiales entregados.

### 3.3 CATÁLOGO DE REQUERIMIENTOS DEL SISTEMA

Un proyecto no puede ser exitoso sin una especificación correcta y exhaustiva de los requerimientos, donde describe las necesidades o deseos de un producto.

En las siguientes tablas se reflejan las funciones del sistema, donde la primera columna hace referencia a la cantidad de funciones para una tarea o módulo específico, la segunda columna describe las funciones en si que engloba un módulo, la tercera columna muestra las clasificaciones que pueden tener cada función, y entre ellas están:

- **Evidente** : Función que debe realizarse, y el usuario debería saber que se ha realizado.
- **Oculto**: Debe realizarse, aunque no es visible para los usuarios.

- **Superflua:** Opcionales, su inclusión no repercute significativamente en el costo ni en otras funciones.

En la tabla 3.1 se especifica la funcionalidad que tiene el sistema para el ámbito de registro y salida de insumos , además de un control sobre el stock , días de estadía para cada insumo y material

Ref:#	FUNCIÓN	CATEGORÍA
R1.1	Registro de insumos entrantes y salientes .	Evidente
R1.2	Registro de materiales entrantes a almacén.	Evidente
R1.3	Verificación del mínimo en Stock, tanto para insumos como para materiales	Oculto
R1.4	Control de estadía en almacén durante 30 días para insumos clasificados.	Oculto
R1.5	Control de estadía en almacén durante 1 semana para el ácido sulfúrico.	Oculto
R1.6	Realizar Informe	Evidente

**Tabla 3.1.** Requisitos funcionales para el control de almacenes

La tabla 3.2 muestra los requisitos para el control de insumos que son utilizados en el proceso de refinado .

REF:#	FUNCIÓN	CATEGORÍA
R2.1	Registro de Insumos utilizados para los procesos de refinado.	Evidente
R2.2	Control de asignación de insumos , con alerta de stock mínimo o vacío	Evidente
R2.3	Registro de actividad cumplida.	Evidente

**Tabla 3.2.** Requisitos funcionales para el control de insumos utilizados en el proceso de refinado.



La tabla 3.3 muestra todos los requisitos para realizar el control de los obreros, desde su registro en la base de datos, hasta la asignación de cada uno de ellos a los distintos procesos que intervienen en el refinado de aceite.

REF:#	FUNCIÓN	CATEGORÍA
R3.1	Registro de obreros .	Evidente
R3.2	Registro de asignación de tareas a cada obrero	Evidente
R3.3	Control de asignación de horarios sin temor a duplicidad de obreros a distintas tareas en un mismo horario.	Oculto
R3.4	Realizar informe.	Evidente

**Tabla 3.3.** Requisitos funcionales para el control de asignación de obreros

En la tabla 3.4 se refleja los requisitos para la realización de la venta de aceites, y esto comprende, el registro de los clientes, así como la generación de reportes según especificaciones.

REF:#	FUNCIÓN	CATEGORÍA
R4.1	Registro de clientes.	Evidente
R4.2	Registro de las ventas realizadas diariamente, teniendo en cuenta a quien se vendió y a que fechas.	Evidente
R4.3	Realizar Informe.	Evidente

**Tabla 3.4.** Requisitos funcionales para la venta de aceites.

En la tabla 3.5 se muestra la especificación sobre las funciones con respecto al producto final, y sobre el cálculo aproximado para la toma de decisiones de ingresos monetarios , basado en un modelo estadístico.

REF:#	FUNCIÓN	CATEGORÍA
R5.1	Registro del producto final.	Evidente
R5.2	Cálculo de la cantidad de producto final a prepararse para una siguiente etapa de producción, para que este de como resultado un aproximado de ingresos monetarios máximos.	Evidente
R5.3	Realizar informe.	Evidente

**Tabla 3.5** Requisitos funcionales para el cálculo de siguientes etapas.

### 3.4 ANÁLISIS

#### 3.4.1 DISEÑO DE CASOS DE USO

##### 3.4.1.1 DIAGRAMA DE CASOS DE USO DE ALTO NIVEL

Los casos de uso describen la secuencia de eventos de un actor (Agente externo) es decir es un documento narrativo de los actores del sistema para completar un proceso (ver acápite 2.3.1.1). Los casos de uso de alto nivel que a continuación se detallan son escenarios que facilitan una descripción de cómo se utilizará el sistema (ver Figura. 3.7).

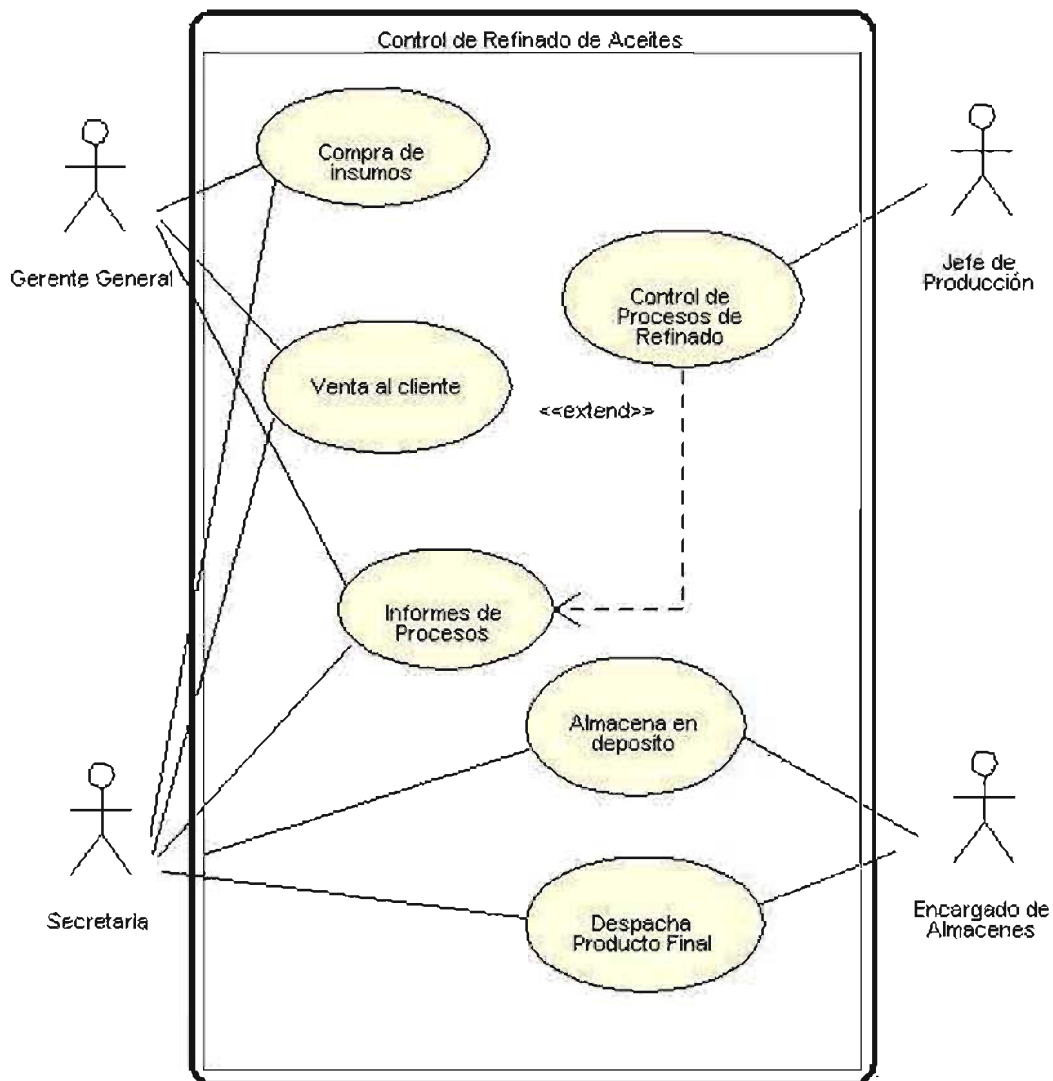


Figura 3.1 Caso de uso (alto nivel) para el control de refinado de aceites

### 3.4.1.2 DESCRIPCIÓN CASOS DE USO DE ALTO NIVEL

A continuación se muestran las tablas referentes a la descripción del caso de uso, cada tabla muestra la descripción por cada caso de uso a fin de entender rápidamente el grado de complejidad y funcionalidad del sistema (ver Tablas 3.6, ..., 3.11).

Caso de uso	Compra de Insumos
Actores	Gerente General, secretaria

<b>Tipo</b>	Primario
<b>Descripción</b>	Se realiza la adquisición de la materia prima e insumos de los distintos proveedores registrados o en su defecto por registrar en sistema, dicha adquisición lo pueden realizar el gerente general y también la secretaria, se registra dicha compra.

**Tabla 3.6.** Descripción del uso Compra de Insumos

<b>Caso de uso</b>	<b>Venta al cliente</b>
<b>Actores</b>	Gerente General, secretaria
<b>Tipo</b>	Primario
<b>Descripción</b>	Se registra toda venta que se haga. La venta del aceite como producto final se lo vende a clientes registrados en sistema, o en su defecto por registrar en el sistema , y dicha venta lo puede realizar tanto el gerente general como también la secretaria

**Tabla 3.7.** Descripción del uso Venta al Cliente

<b>Caso de uso</b>	<b>Consultas e informes de los procesos</b>
<b>Actores</b>	Gerente general, secretaria, jefe de producción
<b>Tipo</b>	Primario
<b>Descripción</b>	Realizar reportes para las distintas consultas que puedan tener el gerente general, secretaria y el jefe de producción sobre la producción, como el control de los insumos en cada etapa, cantidad de productos intermedios, y cálculos de costos realizados por el sistema.

**Tabla 3.8.** Descripción del uso Consultas e informes de los procesos.

<b>Caso de uso</b>	<b>Control del proceso de refinado de aceites</b>
<b>Actores</b>	Gerente general, secretaria, encargado de almacenes
<b>Tipo</b>	Primario

<b>Descripción</b>	El sistema calcula el volumen del producto final a realizar para que los ingresos monetarios sean máximos, para cada etapa del proceso se registra los insumos utilizados, los obreros que fueron asignados para cada tarea, se registra el volumen real del producto intermedio, donde toda esta información es de interés para el gerente general , secretaria, encargado de almacenes y dicha información se ve reflejado en los reportes.
--------------------	---

**Tabla 3.9.** Descripción del uso Control del proceso de refinado de aceites.

<b>Caso de uso</b>	<b>Almacenamiento en deposito</b>
<b>Actores</b>	Encargado de almacenes, secretaria
<b>Tipo</b>	Primario
<b>Descripción</b>	El encargado de almacenes registra todas las entradas de insumos, aceites, y da viabilidad para cada movimiento de estos, además debe verificar el stock mínimo, control de almacén de 30 días para insumos clasificados, y control de 1 semana para el ácido sulfúrico, todo esto mediante el sistema. Los reportes diarios son inspeccionados por la secretaria.

**Tabla 3.10** Descripción del uso Almacerramiento en deposito

<b>Caso de uso</b>	<b>Despacho del producto final</b>
<b>Actores</b>	Encargado de almacenes, secretaria
<b>Tipo</b>	Primario
<b>Descripción</b>	El encargado de almacenes debe de registrar todas las salidas de aceite por ventas, y la secretaria debe de encargarse que la cantidad especificada sea exacta y llegue a destino.

**Tabla 3.11.** Descripción del uso Despacho del producto final.

### 3.4.1.3 CASOS DE USO ESENCIAL O EXPANDIDO

Los casos de uso expandido muestran a detalle los procesos antes mencionados, tienen información breve que describe el proceso, el curso normal de los eventos que detalla la interacción de los actores y el sistema, además de cursos alternativos que pueden presentarse en el curso de los eventos (ver acápite 2.3.1.2).

A continuación se mostrarán los casos de uso expandidos, de los procesos reflejados en el diagrama de casos de alto nivel (ver Figura. 3.2,....., Figura. 3.7), consecutivamente a cada diagrama se muestra una tabla describiendo a detalle en que consiste cada una de ellas (ver Tablas 3.12,.....,3.17).

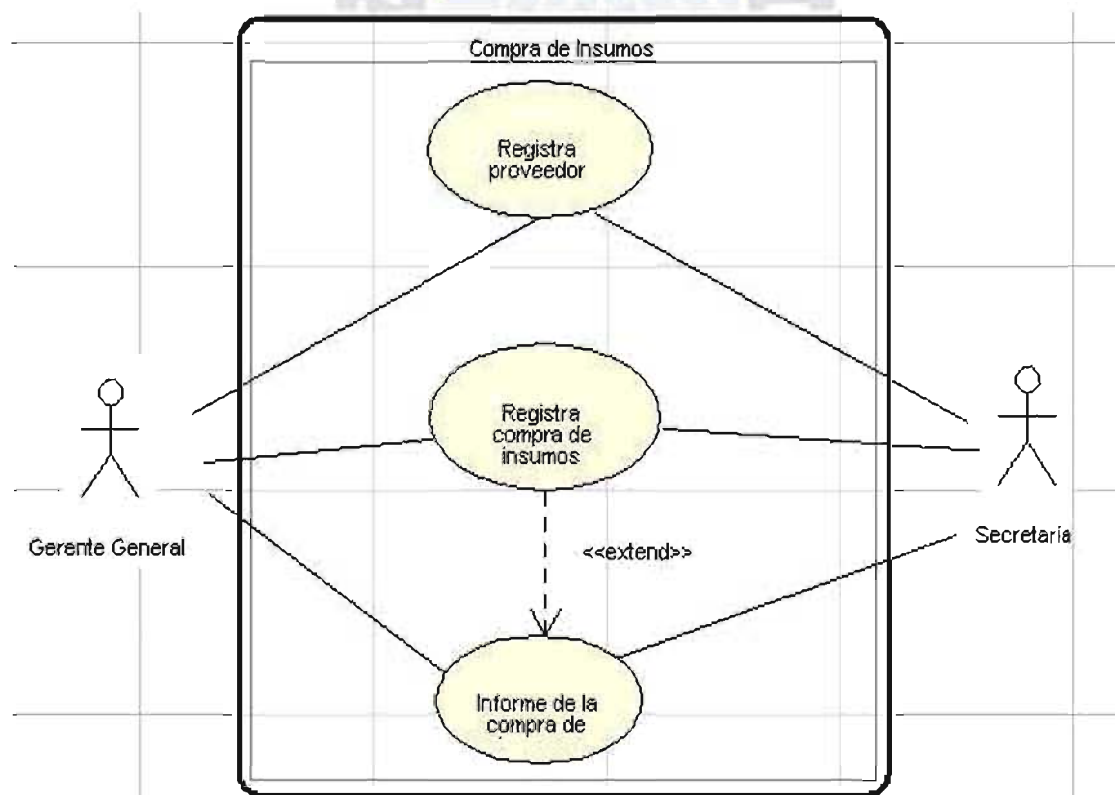


Figura 3.2 Diagrama de caso de uso para la compra de insumos.

<b>Caso de Uso</b>	<b>Compra de Insumos</b>	
<b>Actores</b>	Gerente General, Secretaria	
<b>Propósito</b>	Utilización de insumos en proceso de refinado.	
<b>Descripción</b>	El gerente general y la secretaria tienen potestad para evaluar y realizar la compra de insumos, pero la secretaria debe consultar primeramente con el gerente general antes de realizar la compra. Y si el proveedor no existe en sistema este también debe de ser registrado, luego de realizar la compra se debe sacar un informe de la compra realizada.	
<b>Tipo</b>	Primario	
<b>Referencia Cruzada</b>	R1.1	
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.- Realiza el ingreso de datos necesarios para la compra de insumos  3.- Registra la información del proveedor en caso de ser necesario.  4.- Inserta datos de compra al sistema.  6.- Selecciona imprimir reporte de compra.	2. Imprime en pantalla los datos que están registrados en la base de datos para que estos sean asignados en el registro de la compra.  5.- Registra datos insertados sobre la compra de los insumos.  7.- Imprime reporte de compra realizada.

**Tabla 3.12.** Descripción del diagrama compra de insumos.

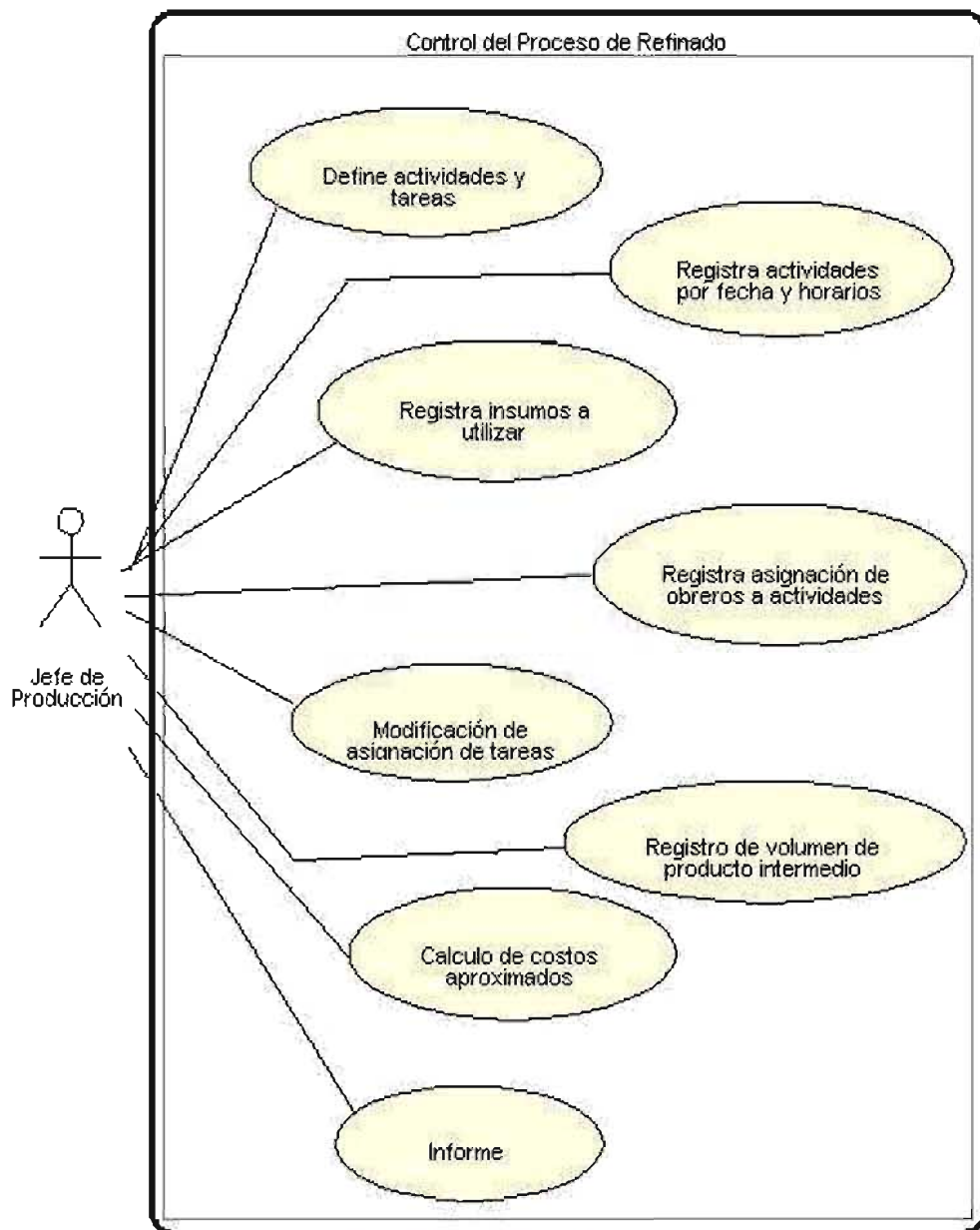


Figura 3.3 Diagrama de caso de uso para el control del proceso de refinado.

<b>Caso de Uso</b>	<b>Control de proceso de refinado de aceite</b>
<b>Actores</b>	Jefe de Producción
<b>Propósito</b>	Control de Tareas Específicas sin tiempos de ocio e inactividad.
<b>Descripción</b>	El jefe de producción realiza un análisis de actividades según el plan del día, o de la semana, luego inserta en el sistema las tareas a realizar dados



	<p>ciertos tiempos, y a cada obrero se le asigna un trabajo en específico, entonces el sistema va registrando las designaciones hechas, en caso de que exista un choque de horarios el sistema avisara el error, y si es así entonces el jefe de producción debe modificar dicho horario, también se realiza el registro de los insumos para cada etapa del proceso de refinado.</p> <p>Existe también la posibilidad de realizar un cálculo de costos en el sistema tomando en cuenta los insumos utilizados.</p> <p>El sistema imprime un informe sobre las asignaciones realizadas.</p>	
<b>Tipo</b>	Primario.	
<b>Referencia Cruzada</b>	R.6.1, R.6.2, R.6.3, R.6.4	
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.- Registra datos sobre las tareas a realizar y sus respectivos horarios.</p> <p>3.- Registra a los obreros que ejecutarán cada tarea específica, e insumos a ser utilizados.</p> <p>5.- Modifica el registra la asignación.</p> <p>6.- Selecciona realizar un reporte sobre las asignaciones de los obreros según fecha o según actividad.</p> <p>8.- Selecciona realizar un reporte sobre los insumos utilizados para la presente actividad.</p>	<p>2.- Muestra selección de obreros e insumos con posibilidad de ser usados .</p> <p>4.- Muestra mensaje para la designación bien hecha, o en caso contrario muestra mensaje de error para la designación mal establecida.</p> <p>7.- Imprime informe.</p> <p>9.- Imprime reporte</p>

**Tabla 3.13.** Descripción del diagrama control de proceso de refinado de aceite.

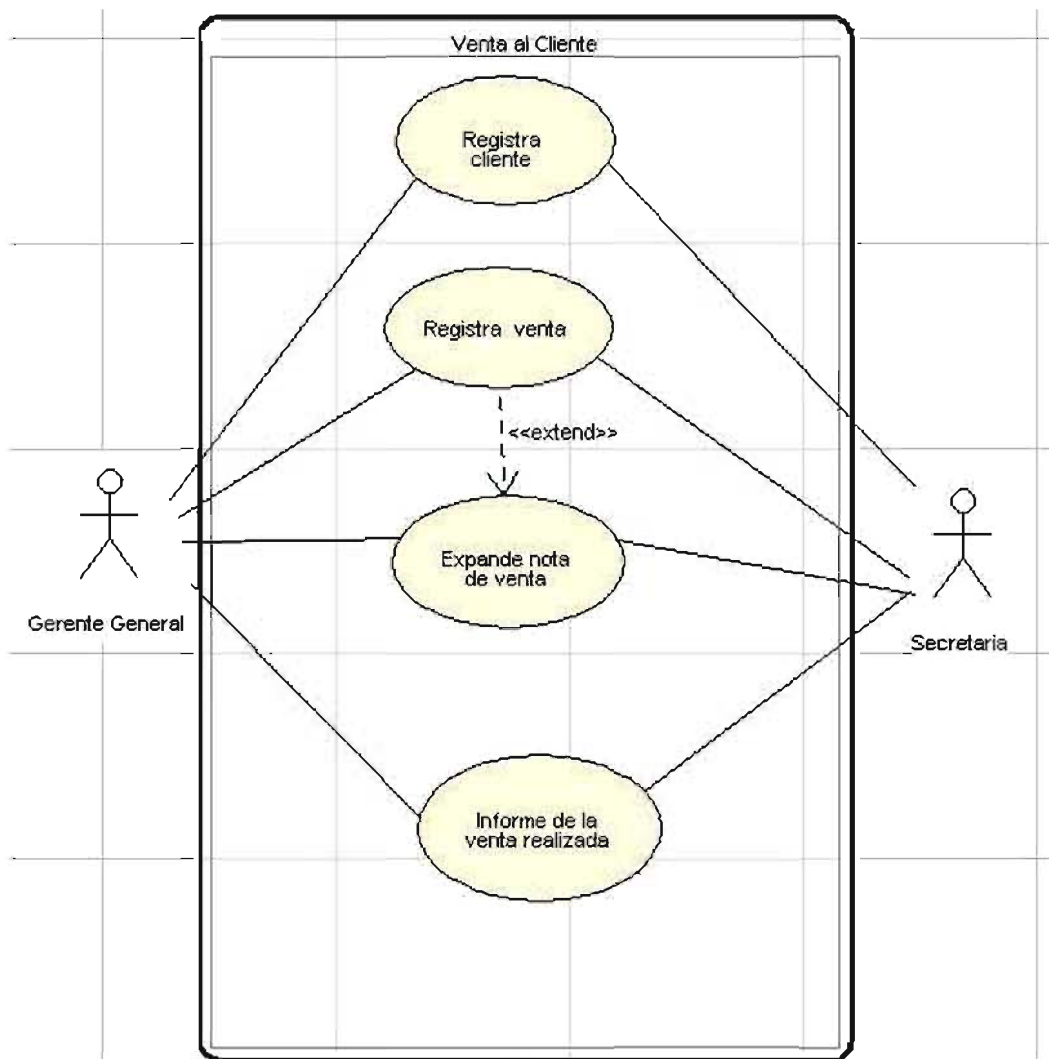
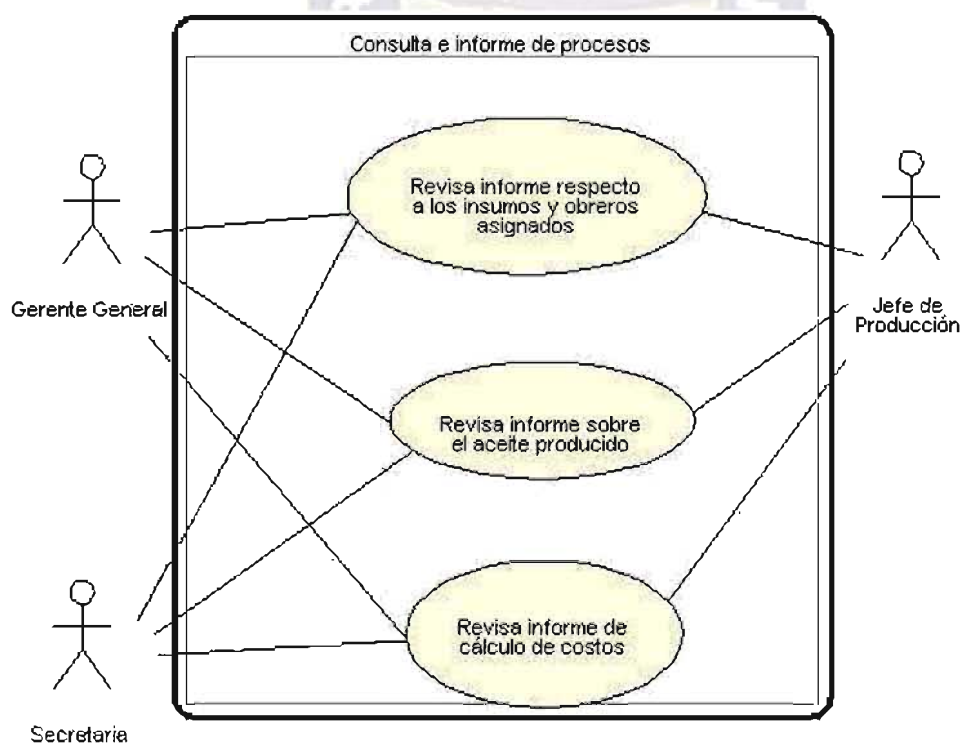


Figura 3.4 Diagrama de caso de uso para la venta al cliente

<b>Caso de Uso</b>	<b>Venta al Cliente</b>
<b>Actores</b>	Gerente General, Secretaria
<b>Propósito</b>	Control de ventas realizadas.
<b>Descripción</b>	El gerente general y la secretaria tienen la potestad para evaluar la venta, pero la secretaria debe consultar con el gerente para realizar una venta, siendo así se realiza el registro del cliente si no existiese en sistema, luego se registra la venta realizada y se entrega una nota de venta al cliente. Se imprime un informe sobre la venta realizada.
<b>Tipo</b>	Primario.

<b>Referencia</b>	R.7.1, R.7.2, R.7.3	
<b>Cruzada</b>		
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.- Realiza el ingreso de datos necesarios para la venta de aceites	2. Imprime en pantalla los datos que están registrados en la base de datos para que estos sean asignados en el registro de la venta.
	3.- Registra la información del cliente en caso de ser necesario.	
	4.- Inserta datos de venta al sistema.	5.- Registra datos insertados sobre la venta de aceite.
	6.- Selecciona imprimir reporte de compra.	7.- Imprime reporte de compra realizada.

**Tabla 3.14.** Descripción del diagrama venta al cliente.



**Figura 3.5** Diagrama de caso de uso para consultas e informes de los procesos.

<b>Caso de Uso</b>	<b>Consulta e Informes de los Procesos</b>	
<b>Actores</b>	Gerente General, Secretaria, Jefe de Producción	
<b>Propósito</b>	Realizar un conteo físico de los insumos, y control de volumen en productos intermedios	
<b>Descripción</b>	Tanto el gerente general , secretaria,, y jefe de producción mediante reuniones analizan los informes y reportes sacados de los procesos de refinado para los tipos de aceite que realiza la empresa, y estos incluyen: insumos utilizados , volúmenes de productos intermedios, obreros que participaron, calculo de costos aproximados.	
<b>Tipo</b>	Primario.	
<b>Referencia</b>	R.2.2, R.3.2, R.3.2, R.4.2, R.5.2, R.8.3	
<b>Cruzada</b>		
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.- El gerente general solicita reunión e información de los procesos realizados.</p> <p>2.- Ingres a al sistema el gerente general y consulta sobre, que insumos se utilizaron, que cantidad, cuanto de volumen se produjo de productos intermedios, el costo aproximado.</p> <p>3.- Se analiza y se compara con la existencia física.</p> <p>4.- Se debate y se toma decisiones en la reunión</p>	<p>3.- Se muestra el tipo de reporte que se desea realizar . Genera un reporte</p>

**Tabla 3.15** Descripción del diagrama consulta e informe de procesos.

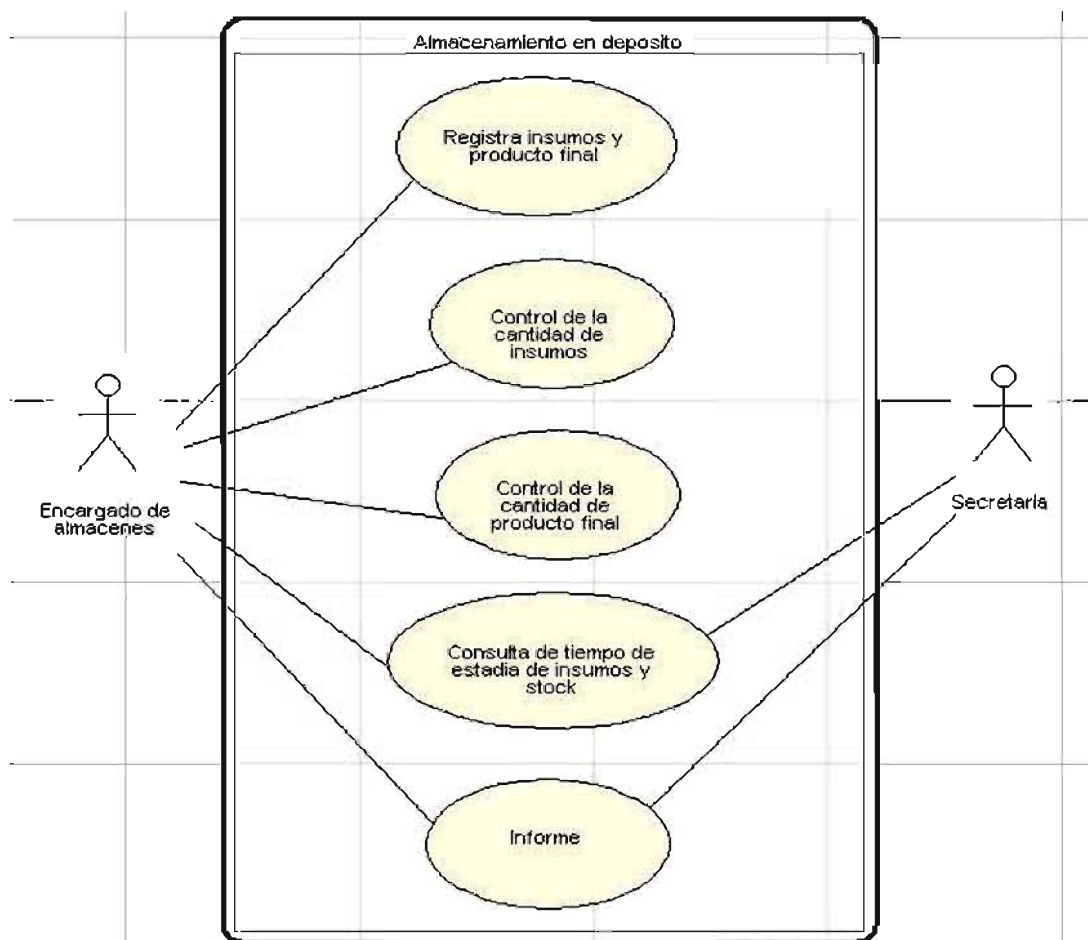


Figura 3.6 Diagrama de casos de uso para el almacenamiento en deposito

Caso de Uso	Almacenamiento en deposito
Actores	Encargado de almacenes, secretaria
Propósito	Control de inventario.
Descripción	El encargado de almacenes registra los ingresos de los insumos y producto final, para luego realizar el control de cantidad de los mismos. También realiza la consulta a sistema sobre el tiempo de estadía de insumos. Genera informe sobre almacenes, que secretaria recopila .
Tipo	Primario.
Referencia Cruzada	R.1.2, R.1.3, R.1.4

Acción del Actor	Respuesta del Sistema
1.- El encargado de almacenes registra datos sobre los insumos y producto final entrantes, y la cantidad en total.	2.- El sistema registra datos ingresados., y calcula el tiempo de estadía de cada insumo.
3.- Selecciona la opción de impresión de tiempo de estadía en insumos.	4.- Imprime reporte

Tabla 3.16. Descripción del diagrama almacenamiento en depósito

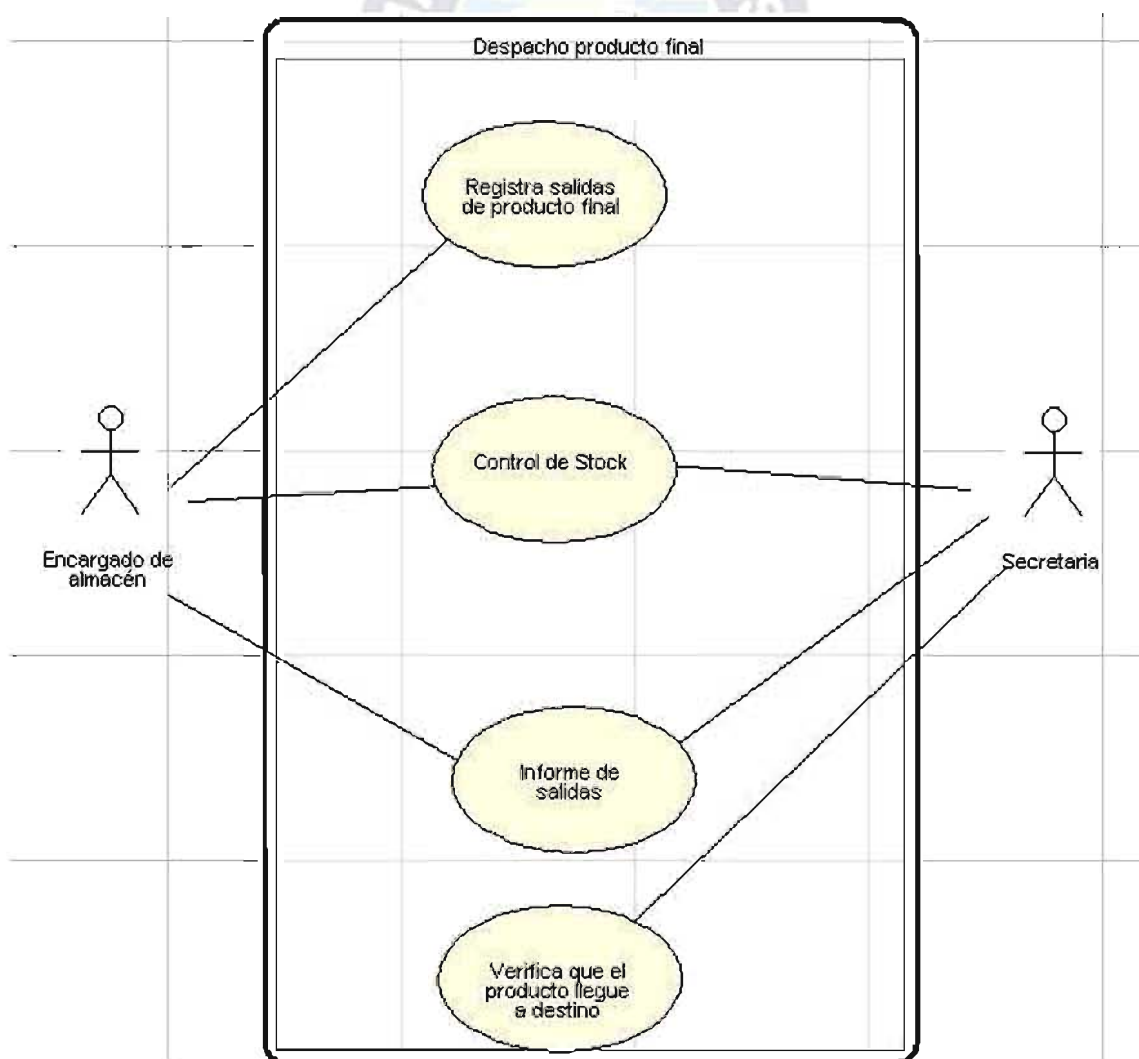


Figura 3.7 Diagrama de caso de uso despacho de producto final

<b>Caso de Uso</b>	<b>Despacho Producto Final</b>	
<b>Actores</b>	Encargado de almacenes, secretaria	
<b>Propósito</b>	Controlar la cantidad que sale de almacén del producto final	
<b>Descripción</b>	<p>El encargado de almacenes registra las salidas de aceite de ambos tipos, y realiza el control de stock que igualmente es calculado por el sistema automáticamente, y dicho control también lo realiza periódicamente la secretaria.</p> <p>El encargado de almacén realiza el informe de las salidas de almacén, que es recopilado por secretaria .</p>	
<b>Tipo</b>	Primario.	
<b>Referencia</b>	R.1.2, R.1.5	
<b>Cruzada</b>		
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.- El encargado de almacenes registra las salidas de almacén del producto final.</p> <p>3.- Consulta sobre el stock actual de producto final, y sobre las salidas .</p>	<p>2.-Registra datos de salidas.</p> <p>4.- Genera reporte .</p>

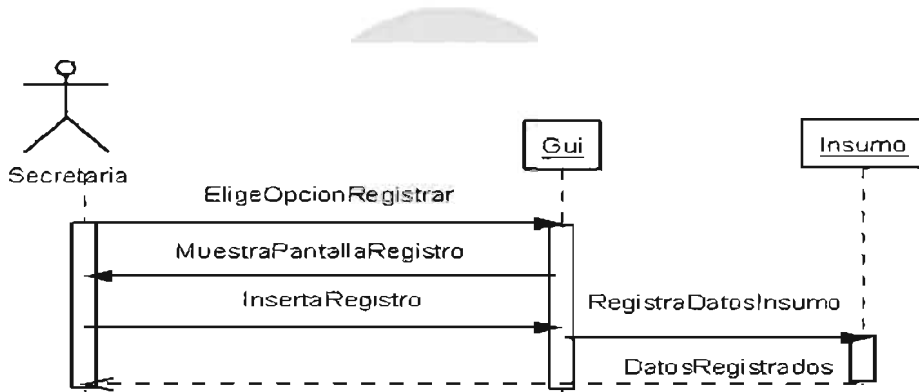
**Tabla 3.17.** Descripción del diagrama despacho de producto final

### 3.4.2 DIAGRAMA DE SECUENCIA

El diagrama de secuencia le muestra la forma en que se comunican los objetos al transcurrir el tiempo en el orden de las llamadas/eventos del sistema(ver acápite 2.3.2). El evento del sistema es una entrada externa que origina una operación del sistema como respuesta al evento, representados en secuencias, el detalle del diagrama depende de la fase en la que estemos, lo que pretendamos contar con el diagrama y a quién.

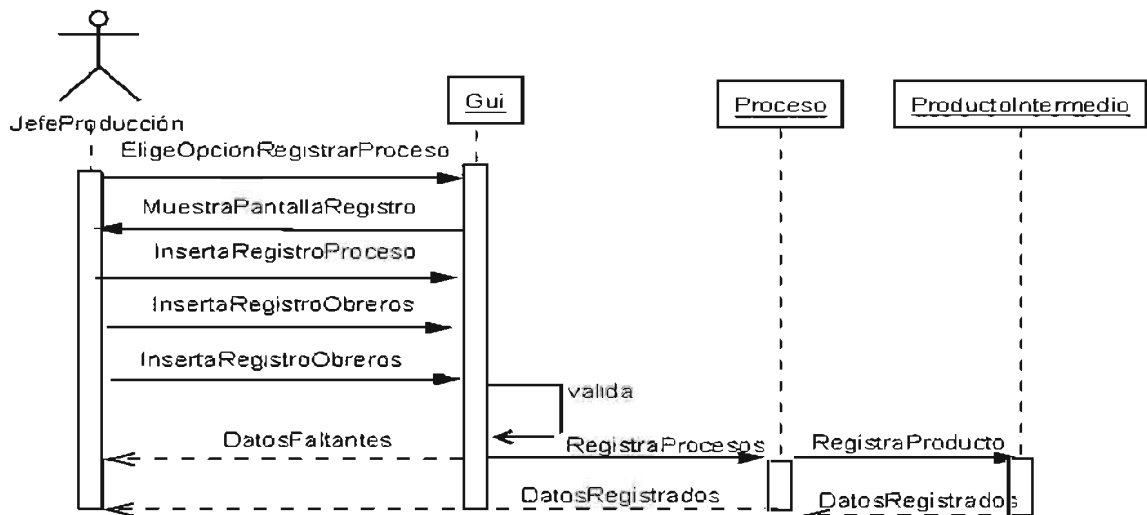
A continuación se muestran los diagramas de secuencia correspondientes al sistema:

El diagrama de la figura 3.8 describe la comunicación entre el usuario que en este caso es la secretaria y los objetos que participan en el proceso del registro de insumos , donde dicho proceso es necesario para el seguimiento del proceso de refinado.



**Figura 3.8** Diagrama de secuencia : Registra insumos y materiales

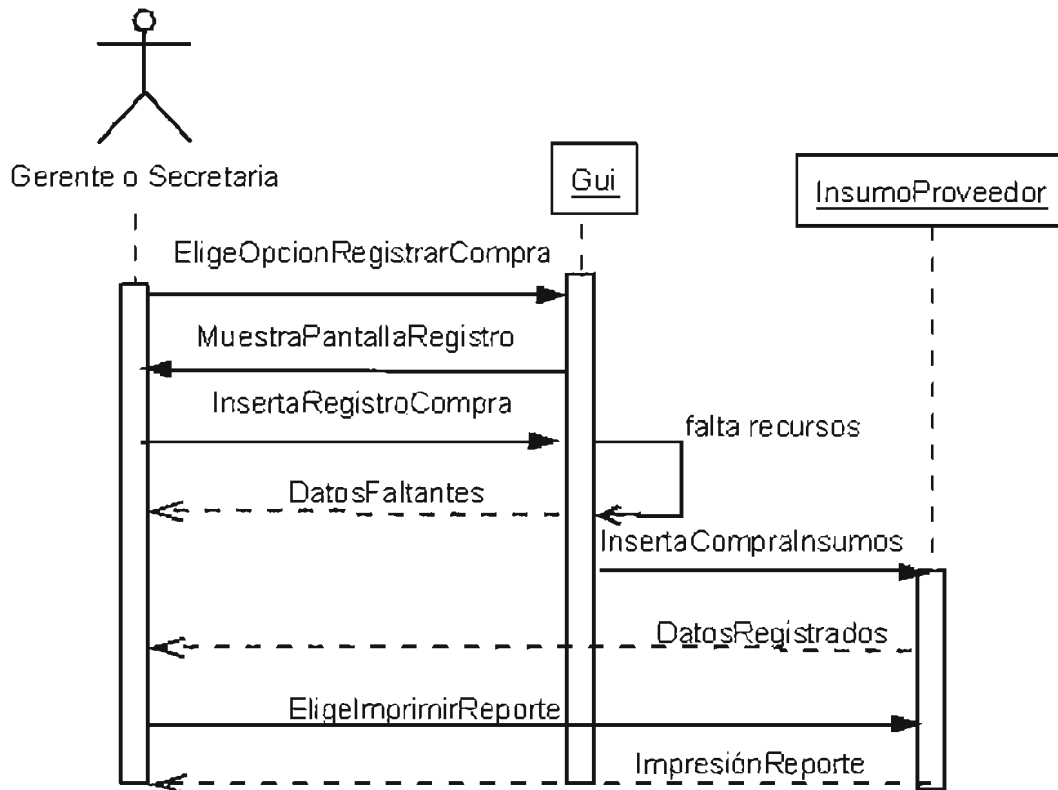
El diagrama de la figura 3.9 muestra las transiciones realizadas para el control de procesos dentro del sistema , el cual tiene que ver con el registro de los insumos a ser utilizados , y los obreros que serán asignados a cada actividad.



**Figura 3.9** Diagrama de secuencia : Control de Actividades



El diagrama de la figura 3.10 muestra la secuencia para la compra de insumos o materiales que se realiza , dando a entender la relación entre el objeto InsumoProveedor y el usuario ya sea Gerente o Secretaria que participan en dicho proceso.



**Figura 3.10** Diagrama de secuencia : Compra de Insumos o Materiales

El diagrama de la figura 3.11 muestra las transiciones realizadas para la venta de aceites dentro del sistema ,dando a entender la relación entre el usuario ya sea Gerenta o Secretaria ,y el objeto compra.

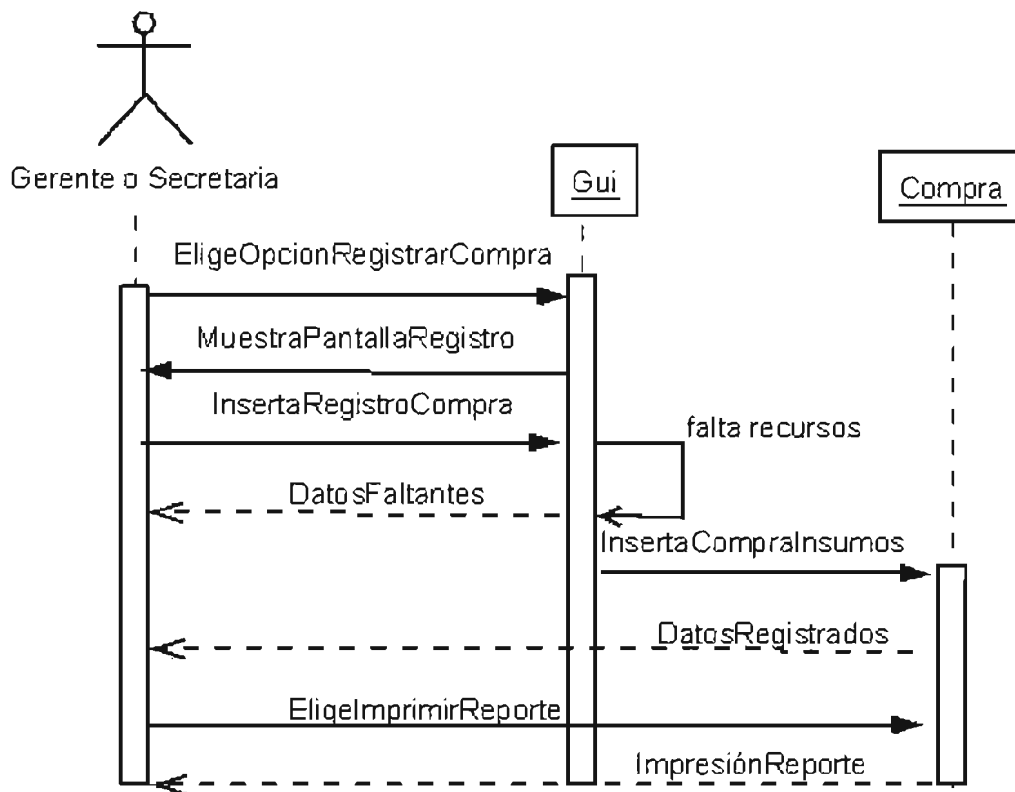


Figura 3.11 Diagrama de secuencia : Venta de Aceite

### 3.5 DISEÑO

#### 3.5.1 DIAGRAMA DE CLASES

Este diagrama describe graficamente la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Se define a una clase como categoria o grupo de cosas que tiene atributos o acciones similares(ver acápite 2.3.4 ). A continuación en la figura 3.12 se muestra el diagrama de clases del sistema.

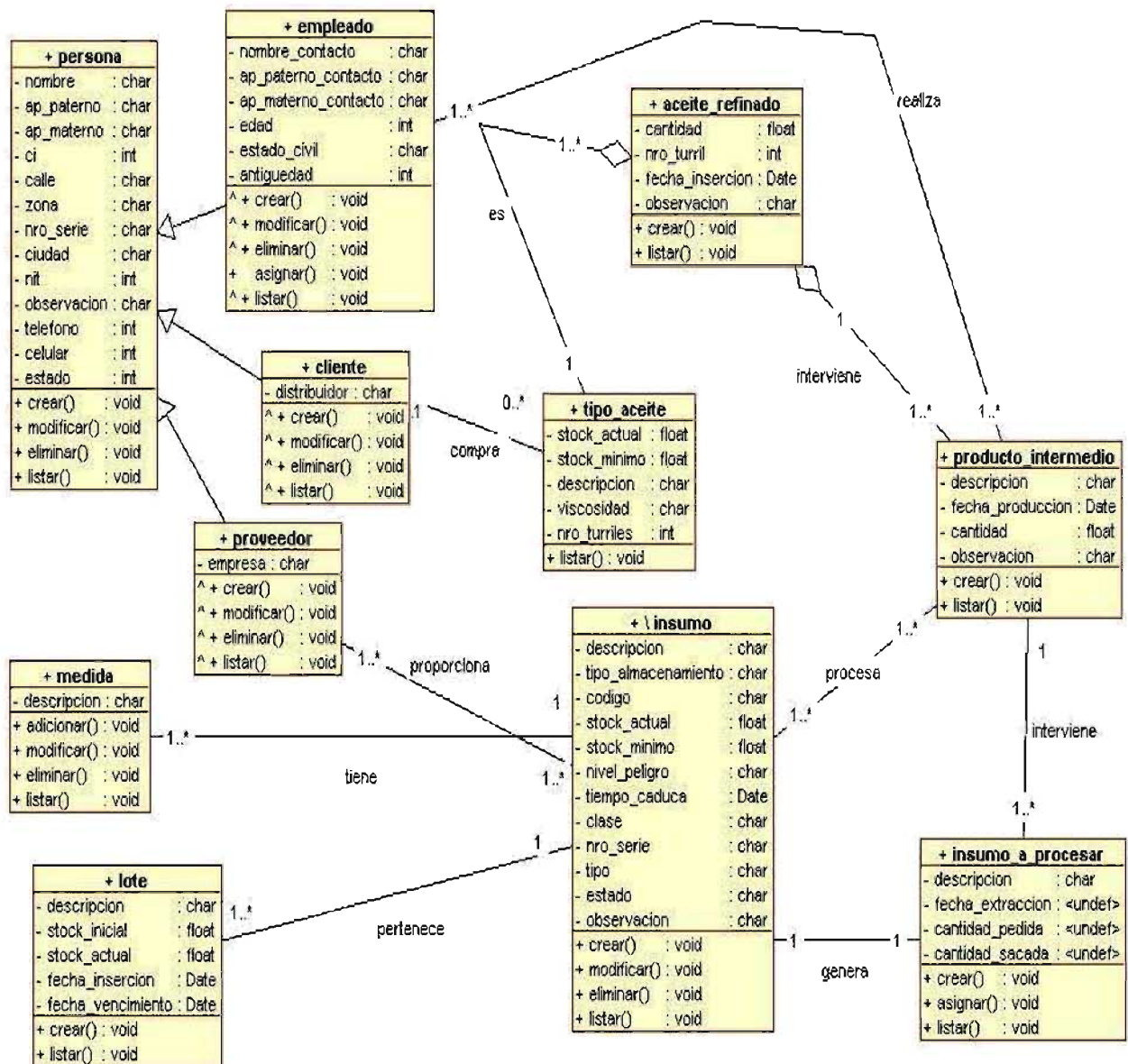


Figura 3.12 Diagrama de clases

### 3.5.2 DEFINICIÓN DEL ESQUEMA DE BASE DE DATOS

A partir de las especificaciones que se tienen para transformar un diagrama de clases a un modelo entidad relación(ver acápite 2.4), es que a continuación se presenta el modelo que representa de manera gráfica en la figura 3.13 el modelo entidad-relación del sistema

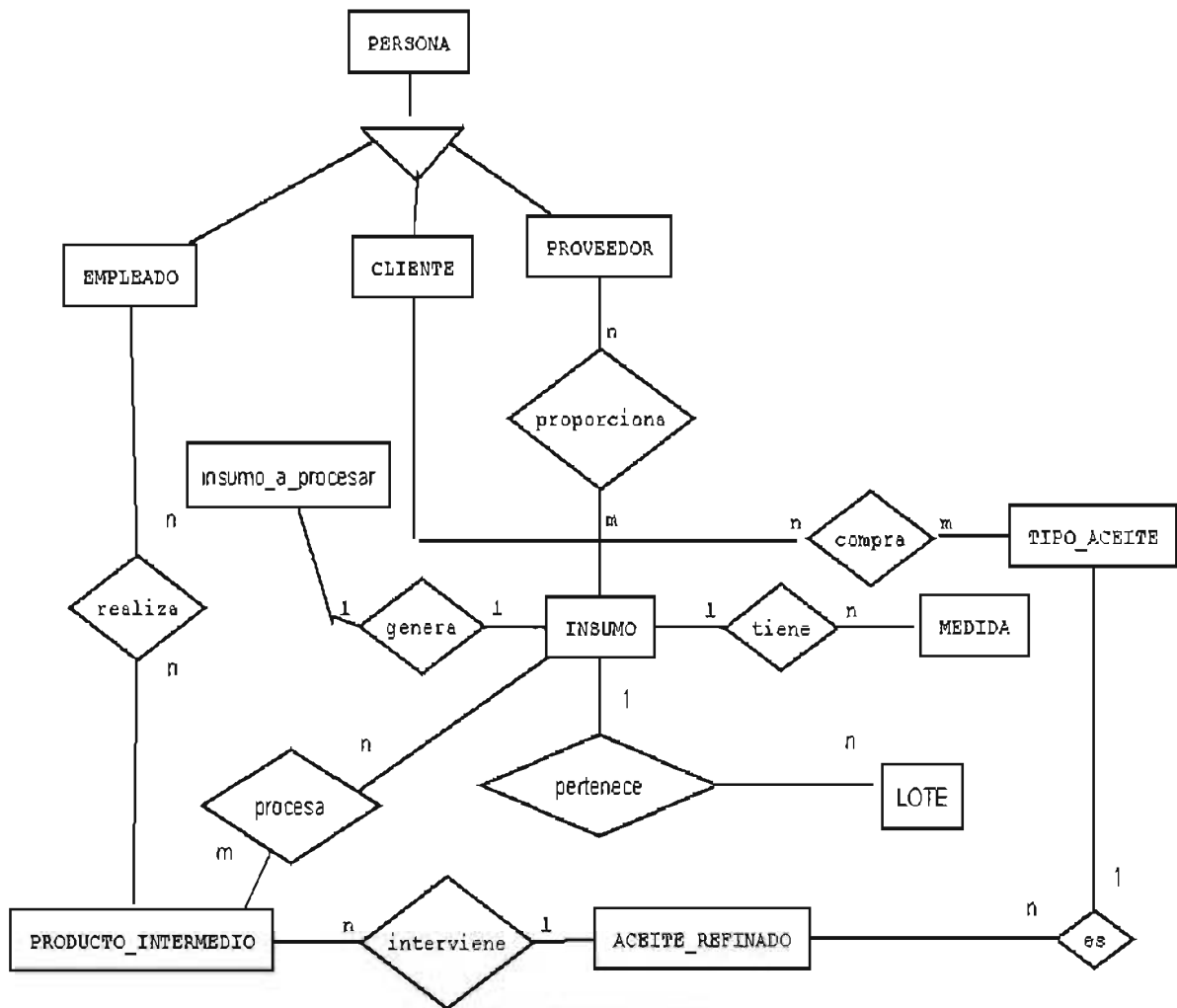


Figura 3.13 Diagrama entidad-relación

### 3.5.3 DIAGRAMA DE COLABORACION

Los diagramas de colaboración muestran la forma en que los objetos colaboran entre sí, mostrando los mensajes que envían entre ellos, destacando el contexto y organización en general de los objetos que interactúan (ver acápite 2.3.3). Dicho de otra manera es la asignación de responsabilidades entre los objetos y mostrar su interacción.

A continuación se muestra los diagramas de colaboración respecto al sistema.

El diagrama de colaboración de la figura 3.14 muestra las operaciones para la compra de insumos , que están definidos en el espacio, y que estos mismos se van a convertir en métodos para ciertas clases

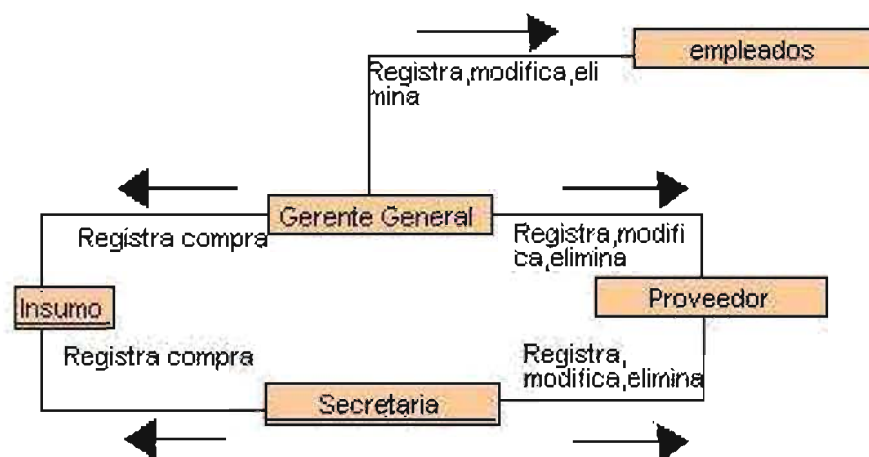


Figura 3.14 Diagrama de colaboración : Compra de insumos

En la figura 3.15 se muestra el diagrama de colaboración para la asignación de actividades y el cálculo de costos , que se reflejan en operaciones asignadas a los objetos.

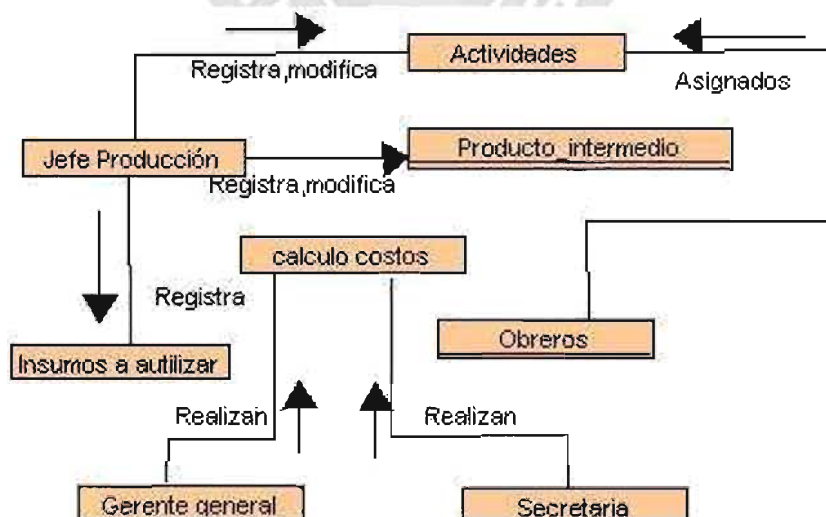


Figura 3.15 Diagrama de colaboración : Asignación de actividades y cálculo de costos

El diagrama de colaboración de la figura 3.16 muestra la venta de aceite como producto final, donde se refleja las operaciones necesarias para dicho proceso.

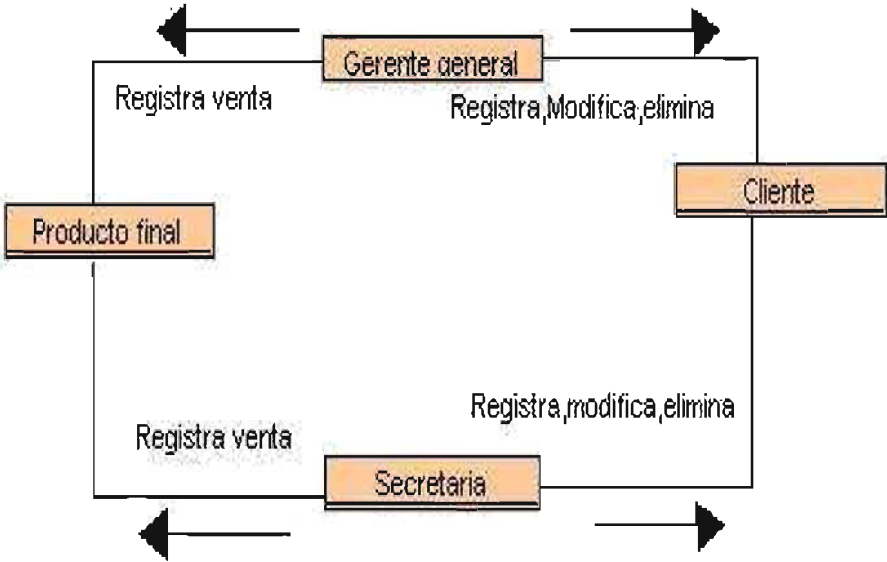


Figura 3.16 Diagrama de colaboración : Venta de producto final

La figura 3.17 refleja como el sistema genera los reportes necesarios de forma genérica, es decir todos los reportes necesarios y solicitados para los distintos módulos.

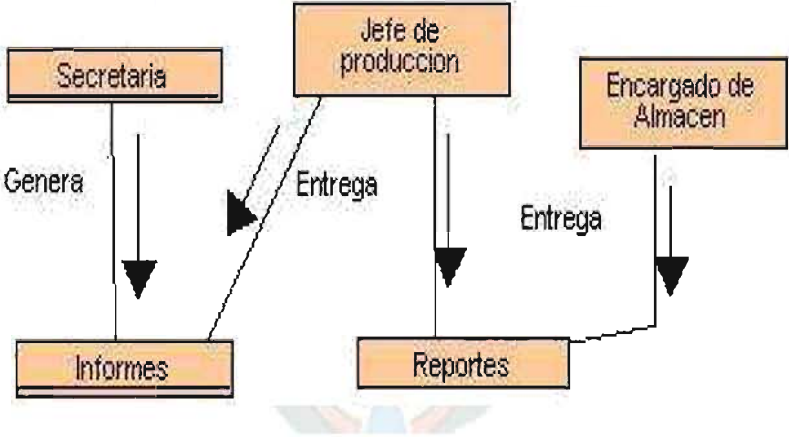


Figura 3.17 Diagrama de colaboración : Genera reportes

### 3.5.4 DISEÑO LÓGICO DEL SISTEMA

A continuación se presenta el esquema lógico de las tablas más importantes de la base de datos (ver tablas 3.18,3.19), y el resto se encuentra en el Anexo D.

<pre>CREATE TABLE `proceso` (   `id_proceso` int(11) NOT NULL,   `id_desc_pro` int(11) NOT NULL,   `id_empleado` int(11) NOT NULL,   `fecha_ini` date default NULL,   `fecha_fin` date default NULL,   `tanque` varchar(20) default NULL,   `id_tipo_aceite` int(11) NOT NULL,   PRIMARY KEY (id_proceso),   FOREIGN KEY (id_desc_pro) references descripcion(id_desc_pro) );</pre>	
<b>Descripción de los campos</b>	
id_proceso	Id de uno de los procesos del refinado de aceite.
id_empleado	Id del empleado que esta asociado al proceso
id_desc_pro	Id foranea , de la tabla de descripcion del proceso.
fecha_ini	Fecha de inicio del proceso
fecha_fin	Fecha de culminación del proceso.
tanque	Tanque donde se realiza cada proceso del refinado.
id_tipo_aceite	Id de la tabla donde hace referencia al tipo de aceite que se esta procesando.

**Tabla 3.18.** Definición lógica de la tabla proceso

<pre>CREATE TABLE `producto_intermedio` (   `id_pro_int` int(11) NOT NULL,   `id_proceso` int(11) default NULL,   `id_aceite` int(11) default NULL,   `descripcion` varchar(20) default NULL,   `cantidad` float default NULL,   `fecha_produccion` date default NULL,   `observacion` varchar(70) default NULL );</pre>	
<b>Descripción de los campos</b>	
id_pro_int	Id que hace referencia al producto intermedio que se controla para cada etapa.
id_proceso	Id que hace referencia a la tabla proceso cuyo valor correspondiente a que proceso pertenece el producto intermedio.
id_aceite	Id del aceite al que como producto final pertenecerá.
descripcion	Descripción del producto intermedio.
cantidad	Cantidad reflejado en litros del producto intermedio producido.
fecha_produccion	Fecha en que se realizo el producto intermedio.
observacion	Observación pertinente del producto realizado.

**Tabla 3.19.** Definición lógica de la tabla producto intermedio

### 3.6 APLICACIÓN DE LA PROGRAMACIÓN LINEAL – MÉTODO DEL SIMPLEX

En el presente proyecto, para el aporte a la toma de decisiones de la empresa, se emplea un modelo estadístico aplicando a la programación lineal, el cual ayuda al área de gerencia y producción a opinar y tomar decisiones con respecto a la cantidad de aceite a producir para que este tenga un mayor beneficio monetario(ver Acápite 2.5.1).

El problema específico que se identifico, es de un tratamiento de maximización, es decir que se tiene que maximizar los ingresos en función de la cantidad de insumos en stock, al precio



fijado de venta del aceite, a las restricción del uso de insumos. Se debe considerar las siguientes observaciones.

- Las variables a encontrar son dos, ya que únicamente son dos tipos de aceite que se vende.
- El aceite de tipo SAE 15W-40 por cada 10 kg se vende a 40 Bs( Bolivianos), al igual que el aceite de tipo SAE 5W-50 por cada 10 kg se vende a 42 Bs(Bolivianos).
- Las cantidades usadas de los insumos son fijas para cada etapa del proceso de refinado.
- Los insumos indicados se miden según la tabla 3.20 que a continuación se muestra:

	Ácido sulfúrico	Aceite Vegetal	Bentoina	Kal	Gas licuado	Hidroxido de Sodio	Carburo	Electrodos	Aditivo
SW-40	2 kg	2 kg	3 kg	0.3kg	4 kg	1 kg	8 kg	5.7 kg	2 kg
SW-50	1 kg	3 kg	1 kg	0.3Kg	3 kg	1 kg	8 kg	5.7 kg	3 kg

Tabla 3.20. Insumos utilizados con valor fijo para cada tipo de aceite.

El modelo planteado para el problema de la empresa en particular es el siguiente:

- Elección de incognitas

X = Cantidad de aceite SAE 15W-40 a producir.

Y = Cantidad de aceite SAE 5W-50 a producir .

- Función objetivo que esta determinado por:

$$F(x,y) = 30 x + 20 y$$

- Restricciones:

$$2x + 1 y \leq \text{stock actual de ácido sulfúrico}$$

$$2 x + 3y \leq \text{stock actual de aceite vegetal}$$

$$3 x + 1y \leq \text{stock actual de bentoina}$$

Solo se toma en cuenta los tres primeros insumos, puesto que para los demás la cantidad requerida para ambos tipos de aceite es la misma.

Los pasos a seguir para resolver el problema son los siguientes:

Vamos a resolver mediante el método del simplex:

$$\begin{aligned} \text{Maximizar } Z &= f(x,y) = 30x + 20y \\ \text{sujeto a: } & 2x + y \leq 18 \\ & 2x + 3y \leq 42 \\ & 3x + y \leq 24 \\ & x \geq 0, y \geq 0 \end{aligned}$$

Se consideran las siguientes fases:

1. Convertir las desigualdades en igualdades

Se introduce una variable de holgura por cada una de las restricciones, para convertirlas en igualdades, resultando el sistema de ecuaciones lineales:

$$\begin{aligned} 2x + y + h &= 18 \\ 2x + 3y + s &= 42 \\ 3x + y + d &= 24 \end{aligned}$$

2. Igualar la función objetivo a cero

$$-30x - 20y + Z = 0$$

3. Escribir la tabla inicial simplex

En las columnas aparecerán todas las variables del problema y, en las filas, los coeficientes de las igualdades obtenidas, una fila para cada restricción y la última fila con los coeficientes de la función objetivo (ver Figura 3.18):

Iteración n° 1

Base	Variable de decisión		Variable de holgura			Valores solución
	x	y	h	s	d	
h	2	1	1	0	0	18
s	2	3	0	1	0	42
d	3	1	0	0	1	24
Z	-3	-2	0	0	0	0

**Figura 3.18** Primera Iteración del Método Simplex.

4. Encontrar la variable de decisión que entra en la base y la variable de holgura que sale de la base.

Para escoger la variable de decisión que entra en la base, nos fijamos en la última fila, la de los coeficientes de la función objetivo y escogemos la variable con el coeficiente negativo mayor.

En nuestro caso, la variable x de coeficiente - 3.

Si existiesen dos o más coeficientes iguales que cumplan la condición anterior, entonces se elige uno cualquiera de ellos.

Si en la última fila no existiese ningún coeficiente negativo, significa que se ha alcanzado la solución óptima. Por tanto, lo que va a determinar el final del proceso de aplicación del método del simplex, es que en la última fila no haya elementos negativos.

Para encontrar la variable de holgura que tiene que salir de la base, se divide cada término de la última columna (valores solución) por el término correspondiente de la columna pivote, siempre que estos últimos sean mayores que cero. En nuestro caso:

$$18/2 [=9], 42/2 [=21] \text{ y } 24/3 [=8]$$

El término de la columna pivote que en la división anterior dé lugar al menor cociente positivo, el 3, ya 8 es el menor, indica la fila de la variable de holgura que sale de la base.

En la intersección de la fila pivote y columna pivote tenemos el elemento pivote operacional 3.

5. Encontrar los coeficientes de la nueva tabla.

Los nuevos coeficientes de  $x$  se obtienen dividiendo todos los coeficientes de la fila  $d$  por el pivote operacional, 3, que es el que hay que convertir en 1.

Mediante la reducción gaussiana hacemos ceros los restantes términos de su columna, con lo que obtenemos los nuevos coeficientes de las otras filas incluyendo los de la función objetivo  $Z$  (ver Figura 3.19).

Iteración nº 2

Base	Variable de decisión		Variable de holgura			Valores solución
	$x$	$y$	$h$	$s$	$d$	
$h$	0	$1/3$	1	0	$-2/3$	2
$s$	0	$7/3$	0	1	$-2/3$	26
$x$	1	$1/3$	0	0	$1/3$	8
$Z$	0	-1	0	0	1	24

Figura 3.19 Segunda Iteración del Método Simplex

Como en los elementos de la última fila hay uno negativo, -1, significa que no hemos llegado todavía a la solución óptima. Hay que repetir el proceso:

- La variable que entra en la base es “ $y$ ”, por ser la variable que corresponde al coeficiente -1.
- Para calcular la variable que sale, dividimos los términos de la última columna entre los términos correspondientes de la nueva columna pivote: y como el menor cociente positivo es “6”, tenemos que la variable de holgura que sale es “ $h$ ”.

El elemento pivote, que ahora hay que hacer 1, es  $1/3$ .

Operando de forma análoga a la anterior obtenemos la figura 3.20:

Tabla III . Iteración nº 3

Base	Variable de decisión		Variable de holgura			Valores solución
	x	y	h	s	d	
y	0	1	3	0	-2	6
s	0	0	-7	0	4	12
x	1	0	-1	0	1	6
Z	0	0	3	0	-1	30

Figura 3.20 Segunda Iteración del Método Simplex

Como en los elementos de la última fila hay uno negativo, -1, significa que no hemos llegado todavía a la solución óptima. Hay que repetir el proceso desde el paso 3 hasta el paso último 5(ver Figura 3.21):

Tabla IV . Final del proceso

Base	Variable de decisión		Variable de holgura			Valores solución
	x	y	h	s	d	
y	0	1	-1/2	0	0	12
d	0	0	-7/4	0	1	3
x	1	0	-3/4	0	0	3
Z	0	0	5/4	0	0	33

Figura 3.21 Última Iteración del Método Simplex

Como todos los coeficientes de la fila de la función objetivo son positivos, se ha llegado a la solución óptima.

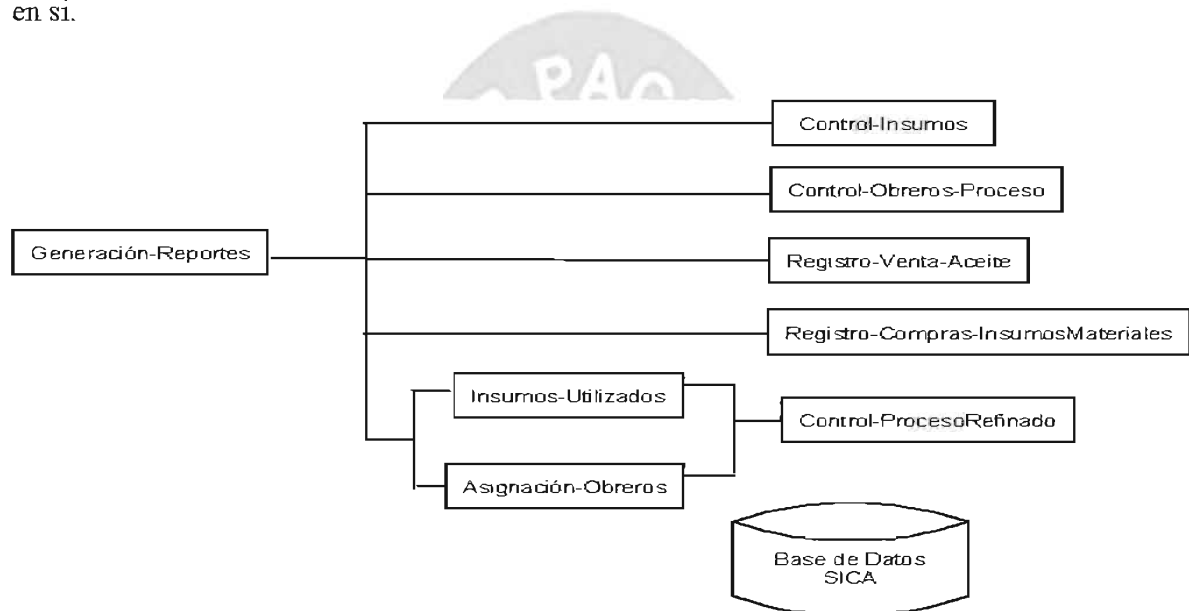
Los solución óptima viene dada por el valor de Z en la columna de los valores solución, en este caso: 33. En la misma columna se puede observar el vértice donde se alcanza, observando las filas correspondientes a las variables de decisión que han entrado en la base: D(3,12)

El resultado sobre la cantidad a producir, y de cuanto se tendrá de ganancias , depende de la cantidad en stock de cada insumo mencionado.

### 3.7 ARQUITECTURA DEL SISTEMA

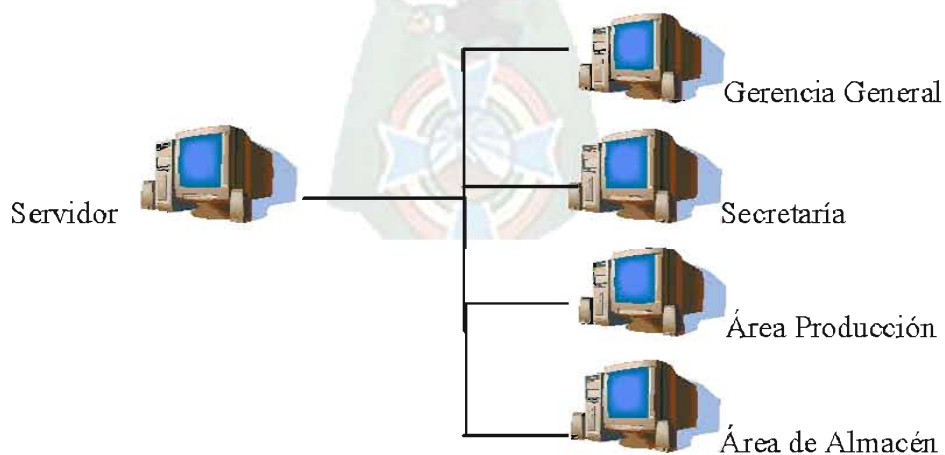
La arquitectura del sistema esta comprendido tanto en la descripción del diseño y contenido, incluye información sobre la organización fundamental de el sistema, que incluye sus componentes, las relaciones entre sí y el ambiente, y los principios que gobiernan su diseño y evolución.

En La figura 3.22 se muestra toda la lógica de aplicación, es decir la arquitectura del software en si.



**Figura 3.22** Diseño de la arquitectura de software

A continuación se presenta el esquema de la arquitectura del sistema en la figura 3.23, donde en cada área se encuentra un computador, y todas ellas se conectan a un servidor común en el cual comparten la información centralizada.



**Figura 3.23** Conformación del reparto de equipos.

### 3.8 METRICAS DE CALIDAD

#### 3.8.1 FUNCIONALIDAD

El punto función es una métrica orientada a la función del software y del proceso por el cual se desarrolla. Se centra en la funcionalidad o utilidad del programa, los puntos de función se calculan realizando una serie de actividades comenzando por determinar los siguientes números:(ver acápite 2.6.1)

- **Número de entradas de usuarios.** Se cuenta cada entrada de usuario que proporciona al software diferentes datos orientados a la aplicación.
- **Número de salidas de usuarios.** Estas se refieren a informes, mensajes de error, es decir salidas que proporcionen al usuario información orientada a la aplicación.
- **Número de peticiones de usuario.** Una petición está definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida.
- **Número de archivos.** Se cuenta cada archivo maestro lógico
- **Número de interfaces externas** Se cuenta todas las interfaces legibles por el ordenador que son solicitados para transmitir información a otro sistema.

De acuerdo a lo mencionado es que se tiene los resultados en la tabla 3.21:

Entradas de usuario	11
Salidas de usuario	14
Consultas de usuario	9
Número de archivos	12
Interfaces externas	0

**Tabla 3.21.** Entradas para el cálculo de funcionalidad

Los puntos de función se calculan rellenando la tabla 3.22 con los datos obtenidos, considerando un factor de ponderación medio.

Parámetros de medición	Cuenta		Facotor de ponderación MEDIO		Totales
Número de entradas de usuario.	11	x	4	=	44
Número de salidas de usuario	14	x	5	=	70
Número de consultas de usuario	9	x	4	=	36
Número de archivos	12	x	10	=	120
Número de interfaces externas	0	x	7	=	0
<b>CUENTA TOTAL</b>					<b>270</b>

**Tabla 3.22.** Calculo de puntos de función sin ajustar.

La relación que permite calcular los puntos de función es la siguiente:

$$PF = CUENTA\_TOTAL * (\text{Grado\_de\_Confiabilidad} + \text{Tasa\_de\_error} * \sum fi)$$

Donde:

- PF = Medida de funcionalidad
- CUENTA\_TOTAL = Es la suma del valor de las entradas, salidas, peticiones, interfaces externas y archivos
- Grado\_de\_confiabilidad = Es la confiabilidad estimada del sistema.
- Tasa\_de\_error = Probabilidad subjetiva estimada del dominio de la información, este error estimado es del 1%.
- Fi = Son valores de ajuste de complejidad que toman los valores de la tabla 3.22 y que dan respuesta a las preguntas de la tabla 3.23

<b>Sin importancia</b>	<b>0</b>
<b>Incidental</b>	<b>1</b>
<b>Moderado</b>	<b>2</b>



Medio	3
Significativo	4
Esencial	5

Tabla 3.23. Valores de ajuste de complejidad.

ESCALA	Sin Importancia	Incremental	Moderado	Medio	Significativo	Esencial
Factor	0	1	2	3	4	5
1. ¿requiere el sistema copias de seguridad y de recuperación fiables?						X
2. ¿Se requiere comunicación de datos?					X	
3. ¿Existen funciones de proceso distribuidos?		X				
4. ¿Es crítico el rendimiento?		X				
5. ¿Será ejecutado el sistema en S:O existente?						X
6. ¿Requiere el sistema de entrada interactiva?				X		
7. ¿Requiere el sistema de entrada de datos interactiva sobre múltiples ventanas?				X		
8. ¿Se actualizan los archivos maestros de manera interactiva?				X		
9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones?		X				
10. ¿Es complejo el procesamiento interno?			X			
11. ¿Se ha diseñado el código para ser reutilizable?				X		
12. ¿Están incluidas en el diseño la conversión y la instalación?					X	
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones?					X	

14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?						X
Total( $\sum F_i$ )	44					

**Tabla 3.24.** Ajuste de complejidad del punto función.

Con la obtención de los anteriores datos y considerando un grado de confiabilidad del 65% es que a continuación calculamos el valor de PF:

$$PF = \text{Cuenta\_Total} * (\text{Grado\_de\_Confiabilidad} + \text{Tasa\_de\_error} * \sum F_i)$$

$$PF = 270 * (0.65 + 0.01 * 44)$$

$$PF = 294.3$$

Si consideramos el máximo valor de ajuste de complejidad como  $\sum F_i = 70$ , se tiene:

$$PF = 270 * (0.65 + 0.01 * 70)$$

$$PF = 364.5$$

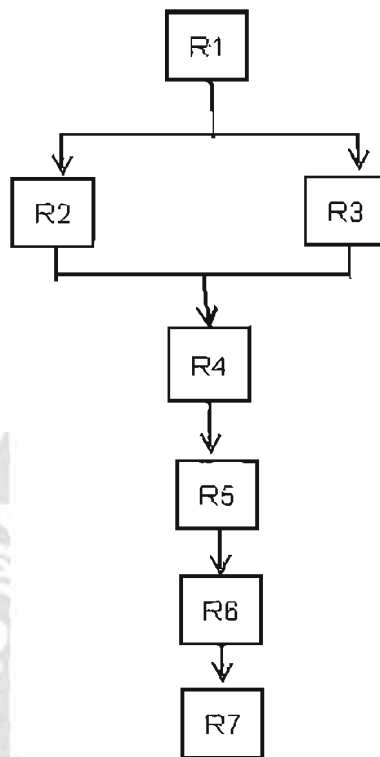
Entonces si  $\sum F_i$  es considerada como el 100%, la relación obtenida entre los puntos será:

$$PF / Pf \text{ máximo} = 294.3 / 364.5 = 0.80$$

Por lo tanto la funcionalidad del sistema es del 80% tomando en cuenta el punto de función máximo.

### 3.8.2 CONFIABILIDAD

En la cantidad de tiempo que el software está disponible para su uso, es posible medir la confiabilidad tomando en cuenta la probabilidad del sistema que este libre de fallos en un contexto determinado y durante un periodo de tiempo (ver acápite 2.6.2). En la figura 3.24 se muestra el diagrama de transferencia detallando sus componentes.



**Figura 3.24** Diagrama de transferencia de módulos

**Fuente:** Elaboración propia

Donde:

- R1: Registro compra de insumos y/o materiales.
- R2: Registro de insumos utilizados en el procesos de refinado.
- R3: Registro de obreros asignados a los diferentes procesos.
- R4: Control del proceso de Refinado en sus cuatro etapas.
- R5: Informe del control del proceso.
- R6: Venta de Aceites a consumidor final.
- R7: Reportes en general.

La confiabilidad de cada componente del diagrama, está especificado en la tabla 3.25.

$\lambda$ : Probabilidad de que se presente una perturbación.

P(T): Probabilidad de fallo en caso de perturbación.

Ri(t)	$\Lambda$	P(T)	$e^{-\lambda P(T)}$
R1(t)	0.2	0.1	0.98
R2(t)	0.15	0.1	0.99
R3(t)	0.2	0.1	0.98
R4(t)	0.2	0.15	0.97
R5(t)	0.15	0.1	0.99
R6(t)	0.15	0.1	0.98
R7(t)	0.15	0.1	0.98

**Tabla 3.25** Cálculo de la confiabilidad de cada instancia de los módulos.

Ahora se debe calcular la confiabilidad del sistema en su totalidad:

Para R2,R3

$$R_A(t) = 1 - [1 - \{R_2(t) * R_3(t)\}]$$

$$R_A(t) = 1 - [1 - \{0.99 * 0.98\}]$$

$$R_A(t) = 0.97$$

Entonces la confiabilidad del sistema esta dado por:

$$R(t) = R_A(t) * R_1(t) * R_4(t) * R_5(t) * R_6(t) * R_7(t)$$

$$R(t) = 0.97 * 0.98 * 0.97 * 0.99 * 0.98 * 0.98$$

$$R(t) = 0.89$$

Por lo tanto, el sistema presenta una confiabilidad del 0.89, lo que quiere decir que el 89% de las ocasiones, el sistema funciona sin presentar fallos y el resto (11%), presenta fallos que no afectan de sobremanera el desempeño global del sistema.

### 3.8.3 PORTABILIDAD

Para el presente sistema se tiene que el hardware en el que funciona de manera estable del lado del servidor, esta dado por un equipo Pentium IV, el acceso a este servidor es atreves de una

intranet donde solo pueden acceder los usuarios autorizados del sistema. Las terminales de donde se accede al servidor tienen características de ser equipos Pentium IV. Por otro lado el Sistema Operativo del lado del servidor es Windows XP, al igual que las terminales de acceso al sistema(ver acápite 2.6.3).

El software es apto para funcionar bajo distintas plataformas, específicamente en Linux debido a que es una aplicación hecha en lenguaje de programación php, Gestor de Base de Datos MySql y servidor Apache que tienen esta característica.

Con respecto al tamaño físico de la aplicación y la Base de Datos, estos ocupan un espacio de 5 Mb aproximadamente. Por lo tanto, podemos concluir que el sistema no requiere de un gran esfuerzo para su traslado de un entorno de Hardware y Software a otro.

### **3.8.4 MANTENIBILIDAD**

#### **3.8.4.1 MANTENIMIENTO ADAPTIVO**

El mantenimiento adaptivo ocurrirá cuando se cambien las políticas o cuando se cambie la estructura organizacional, o cambie el personal de la empresa, modificaciones que se harán que el sistema cambie en poca o gran medida, cambios para los cuales el sistema está preparado en adaptarse a algunos de estos casos, pero para otros más complejos se deberá hacer revisión de los procesos y su adaptación con los nuevos cambios que se generen(ver acápite 2.6.4).

#### **3.8.4.2 MANTENIMIENTO PERFECTIVO**

El sistema está completamente abierto a añadir o adicionar nuevas funcionalidades de acuerdo a los nuevos requerimientos del cliente, siempre y cuando sean relacionados con el servicio e información que brinda el sistema(ver acápite 2.6.4).

### 3.8.5 FACILIDAD DE USO

La medición de la facilidad de uso se puede entender como la facilidad que el usuario tiene para conocer al sistema, tanto como para comprenderlo, aprenderlo y operarlo. A continuación presentamos en la tabla 3.26 los resultados obtenidos en la enseñanza de la manipulación del software a los cuatro usuarios.

USUARIOS	FACILIDAD DE COMPRENSIÓN	FACILIDAD DE APRENDIZAJE	FACILIDAD DE OPERACIÓN
Usuario1	91%	90%	90%
Usuario1	95%	90%	92%
Usuario1	98%	95%	95%
Usuario1	95%	90%	90%
<b>Promedio</b>	<b>94.5%</b>	<b>91.2%</b>	<b>91.7%</b>

**Tabla 3.26** Resultados para el cálculo de facilidad de uso.

Por lo tanto de acuerdo a los resultados de la tabla 3.26 se obtuvo que la facilidad de uso del sistema es de un 94%.

### 3.9 SEGURIDAD DEL SISTEMA

Dado que el sistema corre bajo una intranet se debe considerar por lo menos algunas reglas básicas a cumplirse para que la seguridad del sistema no sea una preocupación para la empresa (ver acápite 2.7). En la implementación del sistema se considera las siguientes políticas de seguridad:

- **Política de contraseñas.** El sistema es capaz de realizar la comprobación de contraseñas para los 4 tipos de usuarios que se tiene actualmente, también es capaz de asignar nuevos usuarios con niveles de acceso.

- **Política de uso adecuado.** En concreto se especifica que está terminantemente prohibido ejecutar programas que intenten adivinar las contraseñas alojadas en las tablas de usuarios de máquinas locales o remotas, o instalar programas maliciosos, insertar dispositivos de almacenamiento secundarios infectados con virus sin previo análisis.
- **Políticas de respaldo.** El administrador del sistema es el responsable de realizar respaldos de la información periódicamente. Cada treinta días deberá efectuarse un respaldo completo del sistema y también deberán ser respaldados todos los archivos que fueron modificados o creados.

### 3.10 DISEÑO DE INTERFAZ

A continuación se muestra las principales interfaces del sistema

- El "Sistema de Control para el Tratamiento de Refinado de Aceites , Caso: RAMSOIL", presenta como pantalla de ingreso la figura 3.25, donde el usuario debe registrar su código y contraseña, donde estos serán verificados según corresponda:



The image shows a login interface for the 'SISTEMA DE CONTROL PARA EL REFINADO DE ACEITES RAMSOIL'. At the top, there is a banner with the word 'SICONSOIL' in large, stylized letters. Below the banner, the title 'SISTEMA DE CONTROL PARA EL REFINADO DE ACEITES RAMSOIL' is displayed in blue. The login form consists of two input fields: 'Usuario' and 'Contraseña', followed by 'Aceptar' and 'Limpiar' buttons.

**Figura 3.25** Pantalla de ingreso al sistema.

- En la figura 3.26 se muestra la pantalla de registro, modificación, eliminación, tanto para insumos, empleados, proveedores, donde dicho formato es genérico para todos los casos.

Nº	Código	Descripción	Tipo Almacenamiento	Stock Actual	Stock Mínimo	Nivel Peligro	Tiempo de Vigencia	Tipo	Clase	Nro Serie	Unidades	Observación
1	001	ACIDO SULFURICO	BODEGA	60050	2	ALTO	7	MATERIA PRIMA	ABC	1254	GR, KR,	DE ALTA PELIGROSIDAD
2	002	ACEITE VEGETAL	BODEGA	70060	10	BAJO	30	MATERIA PRIMA	ABC	1254	GR, KR,	
3	003	BENTONINA	BODEGA	90650	2	ALTO	18	MATERIA PRIMA	ABC	1248	GR, KR,	DE MEDIA PELIGROSIDAD
4	004	ADITIVO(KELTAN 43)	BODEGA	80050	3	MEDIO	9	MATERIA PRIMA	ACB	1248	GR, KR,	
5	005	KAL	BODEGA	70080	34	BAJO	30	MATERIA PRIMA	ABC	1389	GR, KG,	
6	006	SEBO ANIMAL	BODEGA SECUNDAREA	80060	3	BAJO	50	MATERIA PRIMA			GR, KG,	USADO CON GASAS
7	007	HIDROXIDO DE SODIO	BODEGA	12050	4	MEDIO	58	MATERIA PRIMA	ABC	1254	GR, KR,	
8	008	CARBURO	BODEGA SECUNDAREA	14079	38	BAJO	40	MATERIA PRIMA			GR, KG,	USAR CON GASAS
9	009	ELECTRODOS	BODEGA	15078	34	MEDIO	12	MATERIA PRIMA	ABC	12458	GR, KR,	
10	010	CASCOS	BODEGA 3	318	100	MINIMO	365	MATERIAL	SEGURIDAD	123-wt	UNIDAD, CAJA,	
11	011	PALAS	BODEGA 3	312	50	MINIMO	365	MATERIAL	HERRAMIENT	124-WT	UNIDAD, MEDIA DOCENA,	
12	012	MANDILES	BODEGA 3	363	200	BAJO	365	MATERIAL	ROPA DE TRABAJO	125-WT	UNIDAD,	

**Figura 3.26** Pantalla de Registro, Modificación y Eliminación genérico para los datos de Insumo, Empleado, Proveedor

- En la figura 3.27 se muestra la pantalla donde se registra las compras de los materiales e insumos, pudiendo escoger el proveedor, y el material o insumo a comprar .



COMPRAS DE INSUMOS Y MATERIALES

Número de compra: C-4  
 Empresa Proveedor: PERNOS BUSTAMANTE  
 Representante Proveedor: HUMBERTO PEÑALOZA BUSTAMANTE  
 Fecha: 08/06/2009  
 Número de ítems a insertar: 3

DETALLE DE COMPRA

Código	Insumo	Cantidad Real	Unidad de Medida Real	Equivalente Unidades	Unidad de Medida Mínima	Observación	Precio Unitario	Precio Total
002	ACEITE VEGETAL	30.5	GR	30.5	GR		45.2	1378.600
004	ADITIVO(KELTAN 43)	40.2	KR	40002	GR		63	2520126
008	CARBUR0	50	GR	50	GR		45.3	2265
							<input type="button" value="Calculo"/>	<input type="button" value="Total"/>
								2523709

**Figura 3.27** Pantalla de Registro de Compras de Insumos y/o Materiales

- En la figura 3.28 se muestra la pantalla donde se registra las ventas de los dos tipos de aceite que fabrica la empresa, pudiendo escoger al cliente y que tipo de aceite es el que se está vendiendo.

REGISTRO DE VENTA DE ACEITES

Número de venta: V-10  
 Cliente: HERRERA LUNARIO HUGO  
 Distribuidor: HUGO'S  
 Fecha Entrega: 08/06/2009  
 Fecha del Pedido: 08/06/2009 08:00   
 Número de ítems a insertar: 2

DETALLE DE VENTA DE ACEITES

Tipo Aceite	Cantidad Solicitada(Lts.)	Cantidad Entregada(Lts.)	Nº Tariff	Observación	Precio Litro	Precio Total
SAE 15W-40	50	50	30		40.5	2025
SAE 5W-50	60	60	40		42	2520
					<input type="button" value="Calculo"/>	<input type="button" value="Total"/>
						4545

**Figura 3.28** Pantalla de Registro de Ventas de Aceites

- La pantalla que muestra la figura 3.29 refleja el control del proceso de refinado en sí, pudiéndose observar el dinamismo para la elección de distintos insumos que se requiera de un lote en específico, también la asignación de los obreros para ese proceso, además que el proceso es controlado según fecha.

The screenshot shows a software interface for refining process control. At the top, there is a navigation bar with the following options: 'Registrar...', 'Compras', 'Ventas', 'Procesos', and 'Reportes/ Informes'. Below this, there is a central form for process details:

- Proceso: DECANTACION
- Fecha Inicial: [ ] Fecha Inicial
- Fecha Final: [ ] Fecha Final
- Tanque: [ ]
- Aceite: SAE 15W-40

Below the process details, there are two main sections:

#### INSUMOS A UTILIZAR

INSERTA ITEM :: Inserta Item

Código	Insumo	Stock Actual	Stock Mínimo	Nivel Peligro	Clave
001	ACIDO SULFURICO	60050	2	ALTO	ABC

Observación: DE ALTA PELIGROSIDAD

Lote	Elige	Cantidad Pedida
L-1	[ ]	125

Fecha de extracción: 26/06/2009 08:00 Fecha Cantidad Extraída: 125

#### OBREROS A INTERVENIR

INSERTA ITEM :: Inserta Item

Nombre Completo	Antigüedad
QUINONES BLANCO RAFAEL	2

Observación: [ ]

Proceso Utilizado	Fecha Inicio	Fecha Final
ACIDEZ	10/06/2009 08:00:00	10/06/2009 08:00:00

**Figura 3.29** Pantalla Genérica para el Control de los Procesos de Refinado



---

**CAPÍTULO IV**  
**CONCLUSIONES Y RECOMENDACIONES**

## CAPITULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1 CONCLUSIONES

Como conclusión se puede rescatar que el uso de la programación lineal es de gran ayuda no solo para la toma de decisiones, sino también para la aplicabilidad hacia un mejor desempeño, puesto que se construyen modelos específicos según al requerimiento que se tenga en distintas áreas donde esta pueda ser aplicada.

El sistema de control y seguimiento para el tratamiento de aceites, ha sido concluido satisfactoriamente, y a la fecha se realiza su implantación de manera paralela al trabajo que se realiza en el registro de insumos, obreros, sin presentar mayores observaciones por parte de los usuarios.

Con la puesta en marcha del sistema, se ha logrado conseguir que el usuario ahora tenga un mayor control, y pueda tener una visión más amplia de todo el proceso que se realiza para el refinado de aceites, además entre lo destacable está que se puede tener informes detallados de los distintos módulos planteados.

De forma más explícita, y dando conformidad a los objetivos específicos planteados (ver Acápite 1.5), se ha logrado lo siguiente:

- Se tiene un control sobre cada salida de los insumos, en cada etapa en la producción, y que este da como consecuencia la disminución de stock general y por lotes.
- Por cada una de las cuatro etapas en el ciclo de la producción de refinado se tiene un seguimiento secuencial controlado por el sistema.
- Se efectuó la aplicación de la programación lineal usando el método simplex, en el cual se comprueba mediante una comparación de datos que usando este modelo estadístico se logra obtener los datos más óptimos para la producción de aceites, y que

estos obtengan un beneficio mayor en ganancias para la empresa. Esta aplicación se muestra en la siguiente interfaz(ver Figura. 4.1):

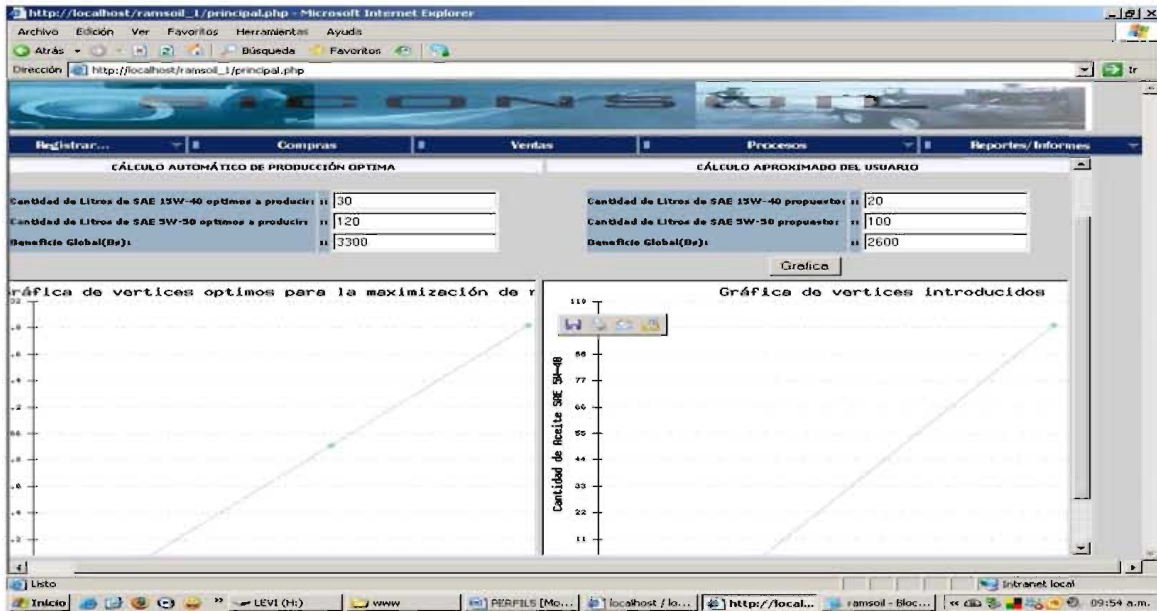


Figura 4.1 Página de comparación de datos factible

Y de la misma manera se muestra el caso en que los datos son rechazados según las restricciones expuestas en el capítulo de análisis y diseño, según se muestra en el siguiente interfaz(ver Figura. 4.2):

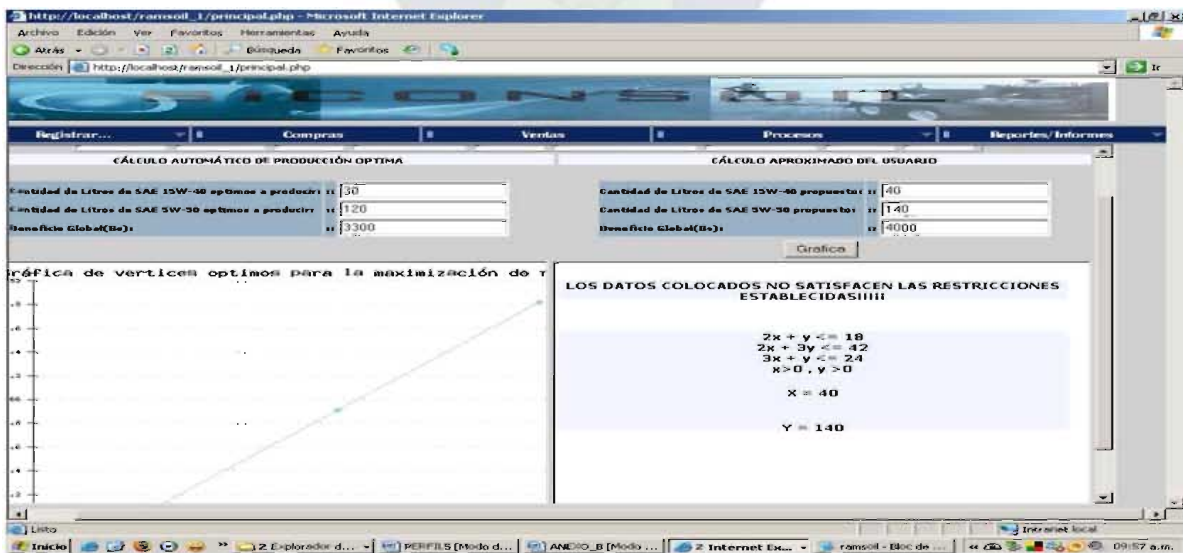


Figura 4.2 Página de comparación de datos rechazada.

- El sistema genera informes sobre la compra de insumos, utilización para producción, y la cantidad de estos en stock general y por lotes.
- El sistema genera reportes de las cantidades con respecto al producto intermedio, es decir al volumen del aceite en proceso de refinado por cada etapa del ciclo de producción.
- Gracias al sistema, se logra ver el comportamiento de uso de insumos para la producción, por lo que estos datos ayudan a la toma de decisiones, y a establecer políticas por parte de la gerencia, y área de producción.
- Se logra realizar un control de asignación de obreros a procesos, lo cual ayuda a establecer patrones laborales con respecto a la función de cada obrero, y así poder realizar un mayor control de estos.
- El sistema puesto en producción trabaja en un entorno web mediante intranet.

Además de cumplir con los siguientes puntos:

- Se tiene un control más destacable para las ventas y compras que realiza la empresa.
- Los reportes son personalizados, es decir que el usuario tiene la ventaja de poder elegir los campos que tendrá el reporte según conveniencia.

Por todo lo expuesto anteriormente se concluye que el Sistema de Control y seguimiento para el tratamiento de aceites, cumple de manera satisfactoria con los objetivos planteados en la fase inicial.

## **4.2 RECOMENDACIONES**

Con la puesta en marcha del presente sistema y analizando los logros obtenidos con el mismo se recomienda:

- Desarrollar sistemas que puedan integrarse a este, dentro de la institución, ya que se ha visto de que gran parte de los procesos de las demás unidades, dependen de gran manera al control del proceso de refinado en sus cuatro etapas.

- Implementar un módulo de control de almacenes y activos fijos, específicos para la empresa, ya que este control debe de realizarse mas detalladamente.



## REFERENCIAS BIBLIOGRÁFICAS

- [Jacob, 2000] Jacobson/Booch/Rumbauch  
El lenguaje Unificado de Modelado Manual de Referencia.
- [Larma,1999] Larman Carig –“UML y patrones” - México 1999 1ra edición
- [Merca,2006] Osvaldo Mercado Torrico  
Sistema de Información para el Seguimiento y Control del Registro de Buques
- [Pauca,1999] Felipe Paucara Condori  
Sistema de información integrado CEPPDM
- [Press,1998] Roger S. Pressman 2003, Ingeniería del Software un Enfoque Práctico, Quinta Edición
- [Plati,2003] Platitini Velthuis M. & Garcia Rubio F.J., Calidad en el Desarrollo y Mantenimiento del Software, 2003
- [Rosen,1998] “Ciencia para todos”, Colegio Nacional y fondo de cultura, México
- [Ticon,2005] María Susana Ticonipa Condori  
Sistema automatizado de registro u seguimiento para la Correspondencia y/o procesos judiciales departamento de asesoría Jurídica

## REFERENCIAS ELECTRÓNICAS

- [Ferrer,2005] Xavier Ferre Grau, María Isabel Sánchez Segura  
Desarrollo Orientado a Objetos con UML
- [Geoci,2000] <http://www.geocities.com/SiliconValley/2208/Insituacion.html>.



[Isfti,2000] <http://www.isftic.mepsyd.es/w3/eos/MaterialesEducativos/mem2003/prog/gramacion/actividades/tecnicas.htm>

[Juan C,1995] Juan Cesar Martínez  
<http://www.ci.ulsa.mx/~elinos/docencia/ctrldesa/ISO14764.pdf>

[Thale,2007] <http://thales.cica.es/rd/Recursos/rd98/Matematicas/29/mixto.html>





**ANEXO A**

---

**ÁRBOL DE PROBLEMAS**

## ÁRBOL DE PROBLEMAS



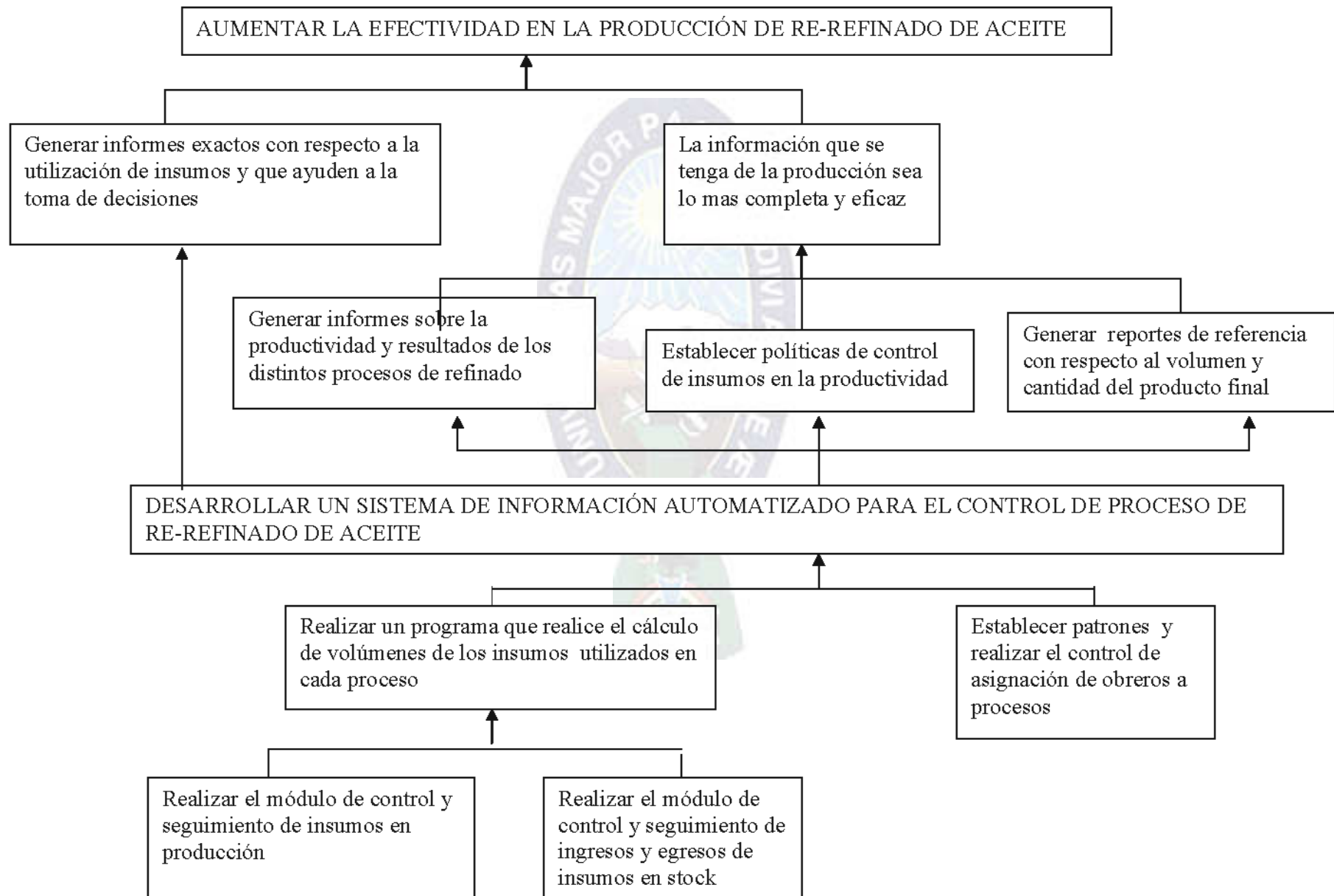


**ANEXO B**

---

**ÁRBOL DE OBJETIVOS**

## ÁRBOL DE OBJETIVOS





---


**ANEXO C**  
**MARCO LÓGICO**

MARCO LÓGICO

RESUMEN NARRATIVO Y OBJETIVOS	INDICADORES VERIFICABLES	OBJ.	FUENTES Y MEDIOS DE VERIFICACIÓN	<u>SUPUESTOS</u>
<p><u>FIN</u></p> <p>Aumentar la efectividad en la producción de re-refinado de aceite</p>	<p>Lograr que gracias al sistema de información se perciba un aumento del 50% o mas en la productividad de la empresa</p>		<p>Informes de evaluación otorgadas por la empresa</p>	<p>Contar con los recursos y materiales necesarios en un tiempo específico para el desarrollo de las actividades</p>
<p><u>PROPÓSITO</u></p> <p>Diseñar, desarrollar e implementar un sistema de Información automatizado que proporcione confiabilidad a la hora de controlar, verificar insumos, y resultados finales en el proceso de el re-refinado de aceite</p>	<p>Tener a mano un informe completo sobre los insumos utilizados y los que están en stock</p> <p>Control sobre obreros y sus funciones</p> <p>El proceso de re-refinado tendrá como resultado salidas controlables</p>		<p>Historial sobre las entradas y salidas de los insumos.</p> <p>Informes de las actividades hechas por los obreros</p> <p>Informes estadísticos de comparación : Insumos de entrada vs. producto final</p>	<p>Contar con que la institución no altere las reglas de producción</p> <p>Cambio de personal de gerencia</p> <p>Disponibilidad de tiempo del gerente limitada</p>

<u>RESUMEN NARRATIVO Y OBJETIVOS</u>	<u>INDICADORES OBJ. VERIFICABLES</u>	<u>FUENTES Y MEDIOS DE VERIFICACIÓN</u>	<u>SUPUESTOS</u>
<u>PRODUCTOS</u>			
Módulo de generación de consultas en insumos de	Módulo que consulta las salidas y entradas de insumos en stock y producción. Desarrollado partir del 1 de febrero del 2008	Dispositivos de almacenamiento: CD's, u otros.	Virus, cambio de personal de Gerencia
Módulo de control de asignación de obreros	Un módulo que permite el control de obreros por cada proceso. Desarrollado a partir del 1 de marzo	Dispositivos de almacenamiento: CD's, u otros.	Virus, cambio de personal de Gerencia
Módulo de seguimiento de etapas de producción de refinado	Módulo que permite evaluar cada etapa de producción. Desarrollado a partir de 16 de marzo.	Dispositivos de almacenamiento: CD's, u otros.	Virus, cambio de personal de Gerencia
Módulo de elaboración de reportes de producción	Emisión de reportes sobre el seguimiento de la producción.	Dispositivos de almacenamiento: CD's, u otros.	Virus, cambio de personal de Gerencia



<u>RESUMEN NARRATIVO Y OBJETIVOS</u>	<u>INDICADORES OBJ. VERIFICABLES</u>	<u>FUENTES Y MEDIOS DE VERIFICACIÓN</u>	<u>SUPUESTOS</u>
<u>ACTIVIDADES</u>			
a)Recopilación de información	a)Del 11 al 18 de octubre	 <p>Informes y formularios de entrevistas.</p>	Disponibilidad de información.
b)Realizar entrevistas al personal de la empresa	b) Del 21 de octubre al 1 de noviembre		
c) Realizar la determinación de requerimientos	c) Del 3 de noviembre al 30.		
d)Revisión y seguimiento de anotaciones sobre el desarrollo de la producción	d)2 semanas		
e)Familiarización y revisión de cada etapa del proceso de refinado	e)3 semanas		
f) Realizar el análisis del sistema	f)3 semanas		
		<p>Documentación del análisis y diseño del sistema que se desarrollará.</p>	<p>Accidentes en la revisión de las etapas de acidez y decoloración</p>

<u>RESUMEN NARRATIVO Y OBJETIVOS</u>	<u>INDICADORES OBJ. VERIFICABLES</u>	<u>FUENTES Y MEDIOS DE VERIFICACIÓN</u>	<u>SUPUESTOS</u>
g) Realizar el diseño del sistema	g)2 semanas	Control de avance del sistema efectuado por el gerente de la empresa.  Aprobación de las pruebas de funcionamiento	
h) Diseñar la Base de Datos	h)2 semanas		
i) Desarrollo del Sistema	i)3 meses Costo de 20\$us.		
j) Implementación del Sistema	j)1 mes Csoto de 50\$us		
Total tiempo = 9 meses Total costo=70\$us.			



**ANEXO D**

---

**DISEÑO LÓGICO DEL SISTEMA**

## DESCRIPCIÓN DE TABLAS DE LA BASE DE DATOS

A continuación se muestran las tablas correspondientes a la base de datos:

```
CREATE TABLE `persona` (
  `id_persona` int(11) NOT NULL,
  `nombre` varchar(30) NOT NULL,
  `ap_paterno` varchar(30) NOT NULL,
  `ap_materno` varchar(30) default NULL,
  `ci` int(10) NOT NULL,
  `calle` varchar(20) default NULL,
  `zona` varchar(20) default NULL,
  `nro_recide` int(11) default NULL,
  `ciudad` varchar(15) default NULL,
  `nit` varchar(15) default NULL,
  `observacion` varchar(70) default NULL,
  `estado` int(11) NOT NULL,
  `telefono` int(11) default NULL,
  `celular` varchar(15) default NULL ,
  PRIMARY KEY (id_persona),
);
```

Descripción de los campos	
id_persona	Id de cada una de las personas.
nombre	Primer nombre de la persona.
ap_paterno	Apellido paterno de la persona.
ap_materno	Apellido materno de la persona.
ci	Carnet de identidad.
calle	Calle donde reside la persona.
zona	Zona donde reside la persona.
nro_recide	Número de casa o oficina.
ciudad	Departamento donde reside
nit	NIT de la persona

observación	Observación de la persona.
estado	Si es 1 esta vigente, si es 0 no esta vigente.
teléfono	Teléfono particular de la persona.

**Tabla 1.** Definición lógica de la tabla persona.

<pre>CREATE TABLE `cliente` (   `id_persona` int(11) NOT NULL,   `distribuidor` varchar(30) default NULL,   `id_cliente` int(11) NOT NULL,   PRIMARY KEY (id_cliente),   FOREIGN KEY (id_persona) references persona(id_persona) );</pre>	
<b>Descripción de los campos</b>	
id_persona	Id de cada una de las personas.
distribuidor	Distribuidor al que pueda pertenecer el cliente
id_cliente	Id del cliente asociado a la venta de aceites.

**Tabla 2.** Definición lógica de la tabla cliente.

<pre>CREATE TABLE `proveedor` (   `id_persona` int(11) NOT NULL,   `empresa` varchar(40) default NULL,   `id_proveedor` int(11) NOT NULL,   PRIMARY KEY (id_proveedor),   FOREIGN KEY (id_persona) references persona(id_persona) );</pre>	
<b>Descripción de los campos</b>	
id_persona	Id de cada una de las personas.
empresa	Empresa al que pueda pertenecer el proveedor.

id_proveedor	Id del proveedor asociado a la co,para de insumos.
--------------	--

**Tabla 3.** Definición lógica de la tabla proveedor.

<pre>CREATE TABLE `aceite_refinado` (   `id_aceite` int(11) NOT NULL,   `cantidad` float NOT NULL,   `nro_turril` int(11) NOT NULL,   `fecha_insercion` datetime NOT NULL,   `observacion` varchar(70) NOT NULL,   `id_proceso` int(11) NOT NULL,   `descripcion` varchar(30) default NULL,   PRIMARY KEY (id_aceite,id_proceso), );</pre>	
<b>Descripción de los campos</b>	
id_aceite	Id del aceite asociado en el proceso de refinado.
cantidad	Cantidad de aceit refinado producido.
nro_turril	Número de turriles usados para la producción
fecha_insercion	Fecha en el que se realizo la insercion del aceite
observacion	Observación del aceite producido.
id_proceso	Id del proceso al que pertenece.
descripcion	Descripcion dela aceite refinado.

**Tabla 4.** Definición lógica de la tabla aceite\_refinado.

<pre>CREATE TABLE `tipo_aceite` (   `id_tipo_aceite` int(11) NOT NULL,   `stock_actual` float NOT NULL,   `stock_minimo` float NOT NULL,   `descripcion` varchar(20) NOT NULL,   `viscosidad` varchar(15) NOT NULL,   `nro_turriles` int(11) NOT NULL,</pre>	
--	--

PRIMARY KEY (id_tipo_aceite), );	
<b>Descripción de los campos</b>	
id_tipo_aceite	Id del tipo de aceite.
stock_actual	Stock actual de uno de los dos tipos de aceite
stock_minimo	Stock mínimo de uno de los dos tipos de aceite
descripcion	Descripción del aceite que se maneja en la empresa
viscosidad	Nivel de viscosidad
nro_turiles	Número de turiles almacenados para cada tipo de aceite.

**Tabla 5.** Definición lógica de la tabla tipo\_Aceite.

CREATE TABLE `medida` ( `id_medida` int(11) NOT NULL, `descripcion` varchar(20) NOT NULL, `id_insumo` int(11) NOT NULL, PRIMARY KEY (id_medida), );	
<b>Descripción de los campos</b>	
id_medida	Id de la medida asignada para los insumos o materiales.
descripcion	Descripción de la medida a utilizar.
Id_insumo	Id del insumo asignado a la medida.

**Tabla 6** Definición lógica de la tabla medida.

CREATE TABLE `lote` ( `id_lote` int(11) NOT NULL, `id_insumo` int(11) NOT NULL, `stock_inicial` float NOT NULL,	
--	--

<pre> `descripcion` varchar(30) NOT NULL, `fecha_insercion` datetime NOT NULL, `stock_actual` float NOT NULL, `fecha_vencimiento` datetime NOT NULL, `id_proveedor` int(11) NOT NULL, PRIMARY KEY (id_lote), ); </pre>	
<b>Descripción de los campos</b>	
id_lote	Id del lote en donde se almacenan los insumos y materiales
id_insumo	Id del insumo al cual pertenece en el lote
stock_inicial	Stock inicial con el cual comenzo el lote.
descripcion	Descripcion del lote.
fecha_insercion	Fecha en el cual se comenzo a llenar el lote.
stock_actual	Stock actual del lote.
fecha_vencimiento	Fecha de vencimiento que se tiene por cada lote.
id_proveedor	Id del proveedor que suministra a cierto lote.

**Tabla 7.** Definición lógica de la tabla lote.

<pre> CREATE TABLE `insumo` (   `id_insumo` int(11) NOT NULL,   `descripcion` varchar(30) NOT NULL,   `tipo_almacenamiento` varchar(30) NOT NULL,   `codigo` varchar(30) NOT NULL,   `stock_actual` float default NULL,   `stock_minimo` float default NULL,   `nivel_peligro` varchar(10) default NULL,   `tiempo_caduca` int(11) default NULL,   `clase` varchar(30) default NULL,   `nro_serie` varchar(20) default NULL, </pre>
---



<pre> `observacion` varchar(70) default NULL, `tipo` varchar(20) default NULL, `estado` int(11) NOT NULL, `id_tipo_insumo` int(11) NOT NULL PRIMARY KEY (id_insumo), ); </pre>	
<b>Descripción de los campos</b>	
id_insumo	Id del insumo o material.
descripcion	Descripcion que hace referencia al insumo o material
tipo_almacenamiento	Tipo de almacenamiento en el cual se guardan los insumos.
codigo	Código por cada insumo registrado.
stock_actual	Stock actual por cada insumo regsitado.
stock_minimo	Stock mínimo permitido ppor cada insumo
nivel_peligro	Nivel de peligro por cada insumo.
tiempo_caduca	Tiempo de caducidad de cada insumo
clase	Número de clasificación
nro_serie	Número de serie.
observacion	Observacion de cada insumo o material.
tipo	Tipo de insumo clasificado según la empresa.
estado	Hace referencia a la vigencia que tiene cada registro.
id_tipo_insumo	Id de que tipo de insumo se refiere.

**Tabla 8.** Definición lógica de la tabla insumo.

<pre> CREATE TABLE `insumo_proceso` (   `id_insumo` int(11) NOT NULL,   `id_proceso` int(11) NOT NULL,   `fecha_sacada` date NOT NULL,   `id_lote` int(11) NOT NULL,   `cantidad_sacada` float NOT NULL, </pre>
---

<pre>`cantidad_pedida` float NOT NULL , PRIMARY KEY (id_insumo,id_proceso), );</pre>	
Descripción de los campos	
id_insumo	Id del insumo o material.
id_proceso	Id del proceso en el cual el insumo interviene.
fecha_sacada	Fecha en el cual el insumoes sacado de almacén.
id_lote	Id del lote de donde se saca el insumo.
cantidad_sacada	Cantiad real sacada de almacenes.
cantidad_pedida	Cantidad pedida para producción.

**Tabla 9.** Definición lógica de la tabla insumo\_proceso.

