

# CAPITULO V

## ANÁLISIS Y DISEÑO DEL SISTEMA INFORMATICO

### 5.1. Análisis orientado a objetos

El presente sistema esta enfocado a la orientación a objetos, en donde se determinan las clases y objetos junto a sus funciones y relaciones. En esta etapa se trata de modelar el comportamiento del sistema en base a los múltiples objetos que lo conforman.

Los procesos que se involucran en el desarrollo de sistemas necesitan ser representados de una forma gráfica, de tal forma que los procesos a manejar sean claros y entendibles. Es por esto, que se utiliza los diagramas UML (Lenguaje de Modelado Unificado), que nos permiten visualizar, construir y documentar los componentes de un sistema.

A continuación se lista los tres diagramas UML que se van a incluir en este proyecto:

- Diagrama de clases
- Diagrama casos de uso
- Diagrama de secuencia

UML(Unified Modeling Language).- Los diagramas UML son herramientas CASE(Ingeniería Asistida por Computadora) y por tanto son muy importantes en el análisis y diseño de un sistema. Es un lenguaje de modelado que permite representar gráficamente los elementos de un sistema de software, aunque actualmente también se utiliza para sistemas de hardware y organizaciones del mundo real.

Este lenguaje es muy completo ya que posee nueve tipos de diagramas para modelar los diferentes sistemas y son los siguientes:

- Diagramas de Casos de Uso: Modela el entorno del sistema y su funcionalidad.
- Diagramas de Secuencia: Modela las interacciones entre objetos.

- Diagramas de Colaboración: Es un modelo alternativo al diagrama de secuencia.
- Diagramas de Estado: Modela todo tipo de mensajes entre los objetos del sistema.
- Diagramas de Actividad: Modela las transiciones internas de los objetos.
- Diagramas de Clases: Modela la estructura estática de las clases.
- Diagramas de Objetos: Modela la estructura estática de los objetos
- Diagramas de Componentes: Modela el comportamiento entre componentes.
- Diagramas de Implementación: Modela la distribución del sistema.

Otra manera de clasificar los diagramas UML se presenta en la siguiente tabla:

Área	Vista	Diagramas	Conceptos Principales
Estructural	Vista Estática	Diagrama de Clases	Clase, asociación, generalización, dependencia.
	Vista de Casos de Uso	Diagramas de Casos de Uso	Caso de Uso, Actor, asociación, extensión,
	Vista de Implementación	Diagramas de Componentes	Componente, interfaz, dependencia, relación.
Dinámica	Vista de Estados de máquina	Diagramas de Estados	Estado, evento, transición, acción.
	Vista de actividad	Diagramas de Actividad	Estado, actividad, transición, determinación, división.
	Vista de interacción	Diagramas de Secuencia	Interacción, objeto, mensaje, activación.
		Diagramas de Colaboración	Colaboración, interacción, rol de colaboración, mensaje.
Administración o Gestión de modelo	Vista de Gestión de modelo	Diagramas de Clases	Paquete, subsistema, modelo.
Extensión de UML	Todas	Todos	Restricción, estereotipo, valores, etiquetados.

**Tabla 5.1.** Clasificación de los Diagramas UML

En la siguiente sección se estudia los tres tipos de diagramas que se implementan en el presente trabajo de investigación.

### **5.1.1. Diagrama de Clases**

Se define las clases y su interacción entre las mismas. Una clase es una descripción de un conjunto de objetos que tienen un comportamiento y estructura similar, y además comparte los mismos atributos, relaciones y semántica. La estructura está conformada por atributos y asociaciones. El comportamiento está descrito por su funcionamiento. Una clase normalmente lleva a cabo uno o más interfaces.

Las propiedades de una clase son las siguientes:

*Atributos.*- Describe las características que va a tener la clase.

*Identificadores.*- Son atributos de la clase, o las combinaciones de atributos de la clase cuyos valores identifican cada ocurrencia de la clase.

*Operaciones.*- Es una especificación de una consulta que se puede ejecutar.

*Asociaciones.*- Despliegan la lista de atributos relacionados que proceden de las asociaciones que están conectadas a la clase.

En la siguiente figura se observa el diagrama de clases para el proyecto de crudos.

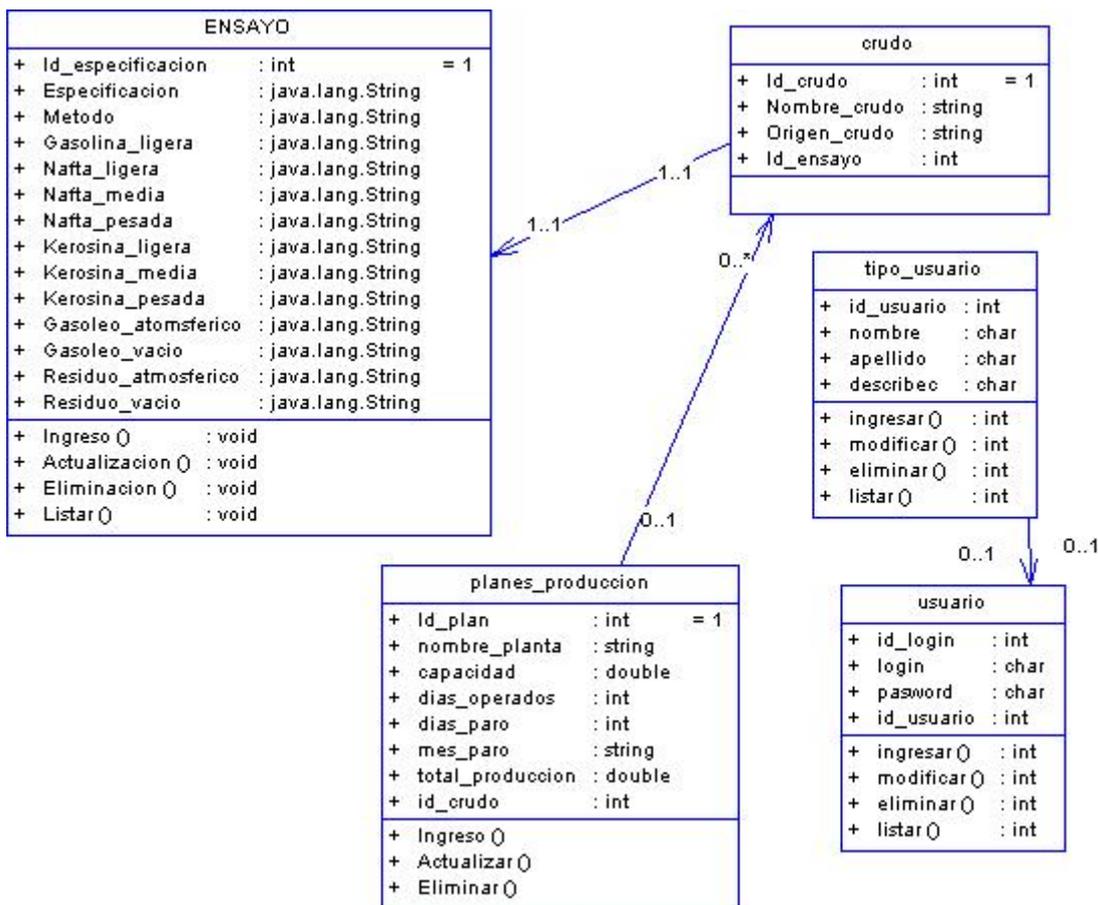


Figura 5.1. Diagrama de clases preliminar de ensayos de crudo

### 5.1.2. Diagrama de Casos de Uso

Este diagrama representa la forma de cómo un actor opera con el sistema en desarrollo, además de visualizar el orden en que los elementos estarán interactuando.

Este tipo de diagrama tiene sus componentes como son:

**Actor:** Cumple el rol de un usuario que interacciona con el sistema. En el diagrama el actor puede comunicarse únicamente con un caso de uso, o con otro actor.



Figura 5.2. Representación de un actor.

**Casos de uso:** Se trata de una operación específica que se ejecuta por una orden de un agente externo, sea ésta desde un actor o desde otro caso de uso. Los casos de uso tienen un nombre que indica la funcionalidad del mismo en el diagrama.



**Figura 5.3.** Representación de un caso de uso

**Relaciones:** Existen relaciones de:

**Asociación.**- Indica la invocación desde un actor o caso de uso a otro caso de uso. Representación:  $\longrightarrow$

**Dependencia o Instanciación.**- Indica la relación entre clases, donde una clase depende de otra (instancia). Representación:  $----->$

**Generalización.**- Tiene dos funciones dependiendo de su estereotipo que puede ser de uso ( $\ll\text{uses}\gg$ ) o de herencia ( $\ll\text{extends}\gg$ ). Esta relación está orientada para casos de uso y no para actor.  
Representación:  $\longrightarrow$

$\ll\text{uses}\gg$  se utiliza cuando se tiene un conjunto de características que son similares en más de un caso de uso y así no volver a copiar la descripción de la característica.

$\ll\text{extends}\gg$  Se utiliza cuando un caso de uso es similar a otro.

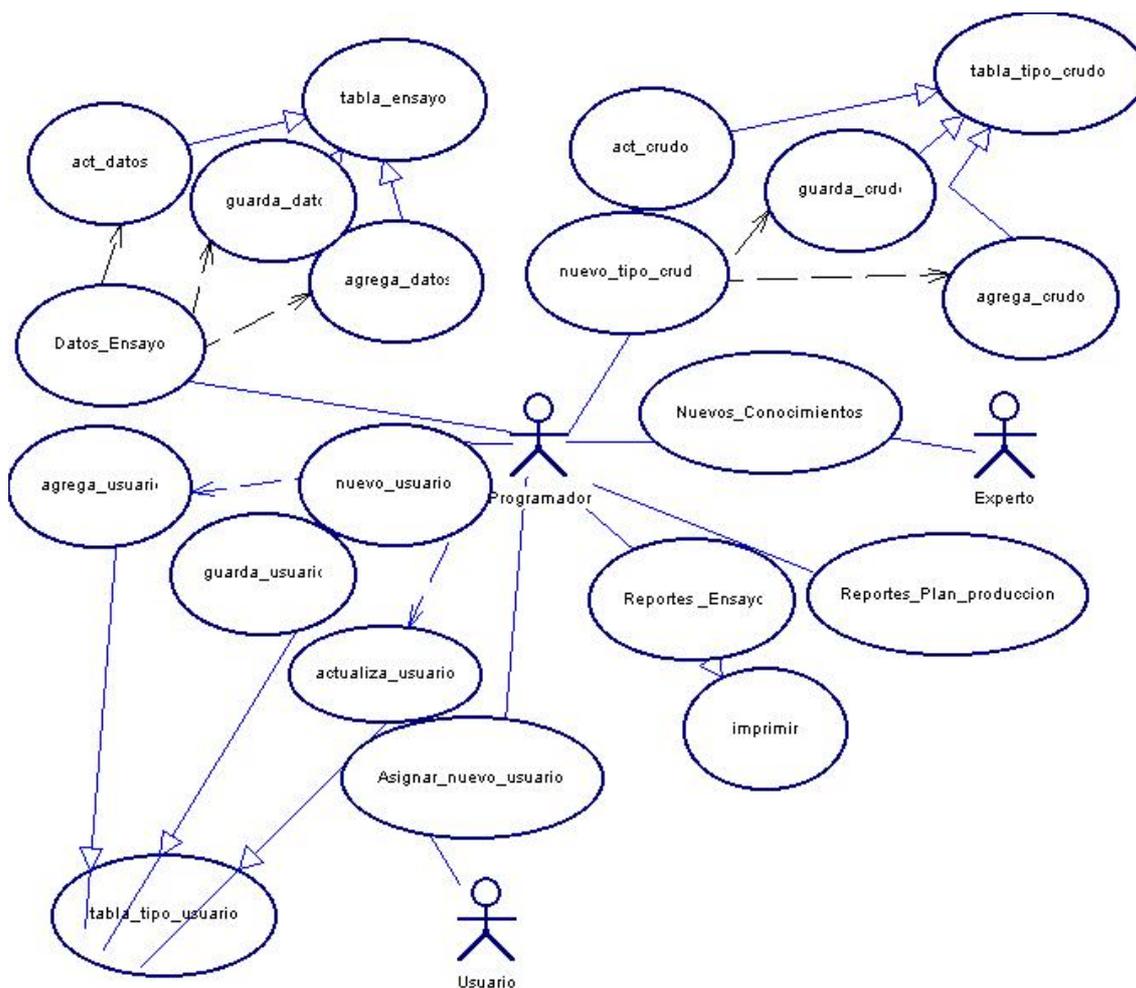


Figura 5.4. Diagrama de casos de uso preliminar.

### 5.1.3. Diagrama de Secuencia

El diagrama de secuencia es parte del Modelo Orientado a Objetos. Muestra los objetos (instancias de una clase), y los mensajes que existen en las relaciones. Aquí se puede incluir actores que interactúen con el sistema.

Su representación gráfica se asimila a un mapa bidimensional, donde la parte vertical es el eje de tiempo, y la parte horizontal muestra las funciones de los objetos.

Los elementos de este diagrama son:

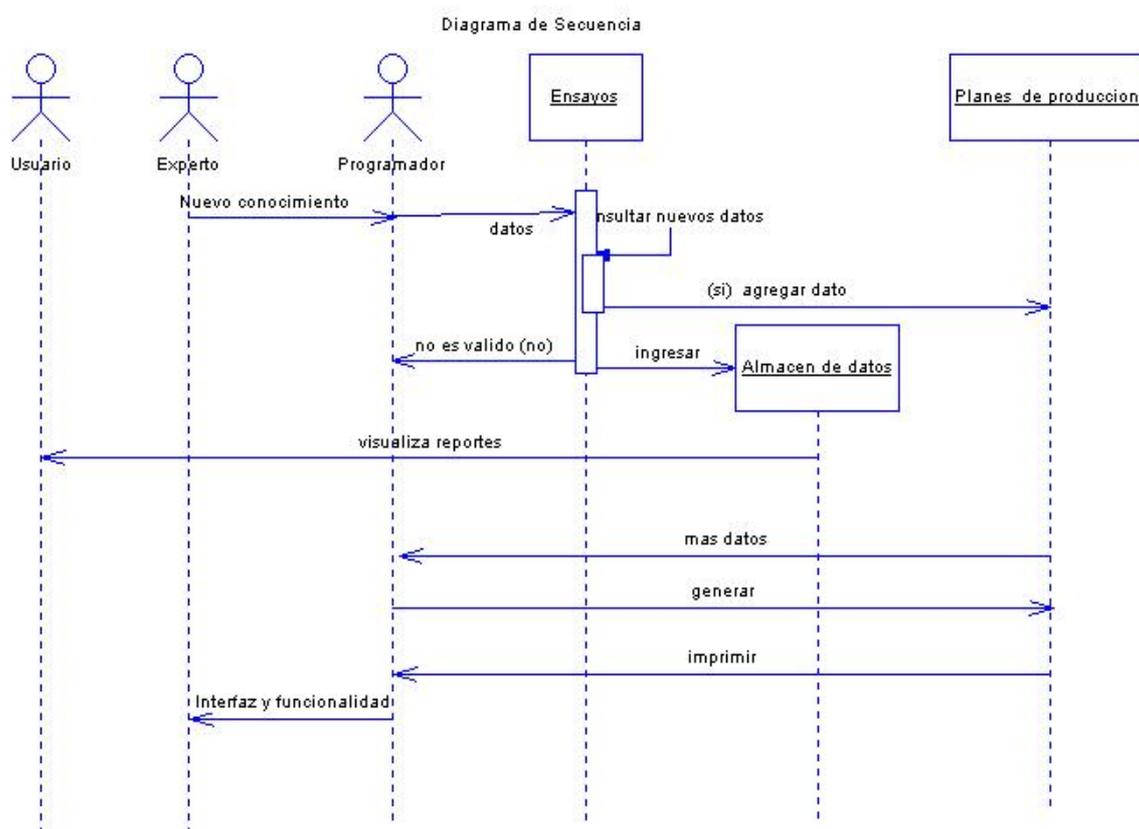
**Paquete.-** Es usado para organizar los objetos en grupos.

**Actor.-** Representa una agente externo, proceso o alguna interacción con el sistema.

**Objeto.**- Instancia de una clase

**Activación.**- Es la ejecución de un procedimiento.

**Mensaje.**- Comunicación entre objetos.

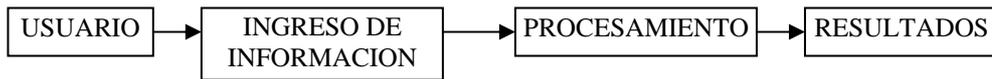


**Figura 5.5.** Diagrama de secuencia preliminar

## 5.2. Definición de la estructura de Objetos

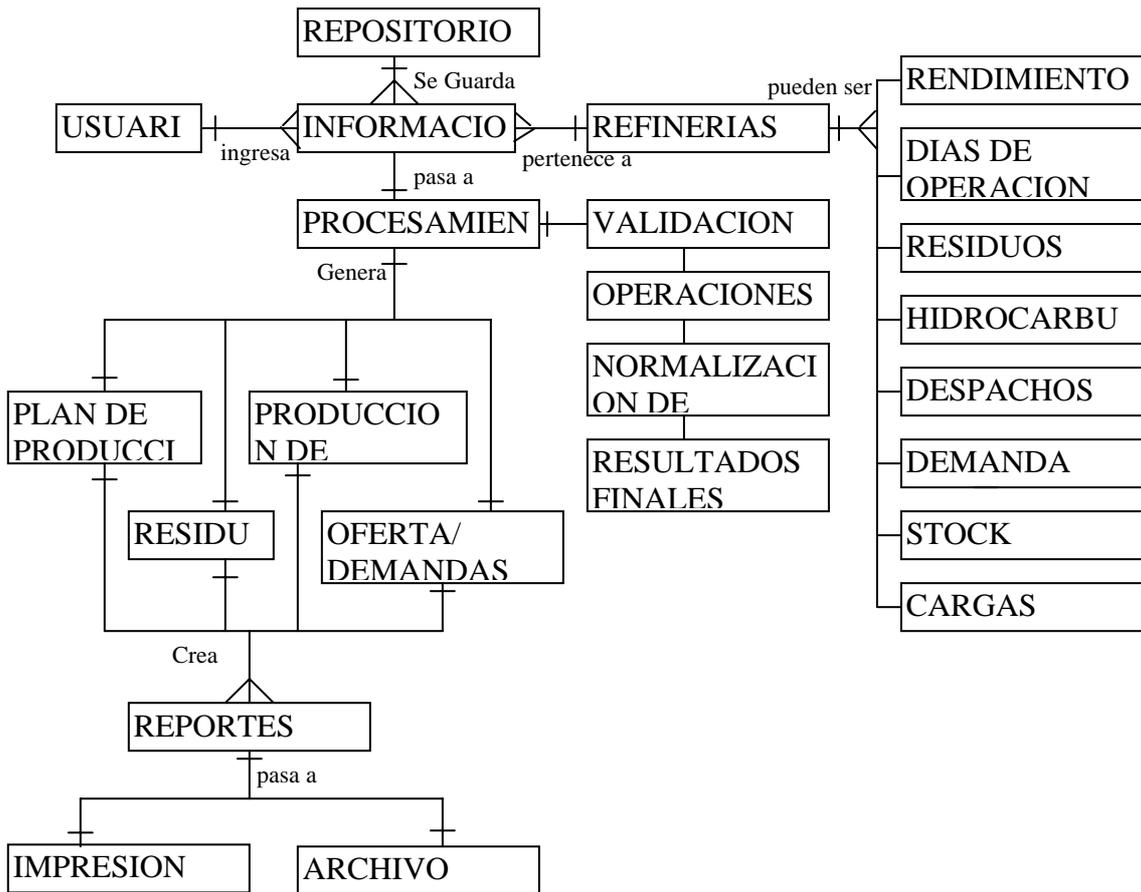
Además de los diagramas UML se especifica el diagrama de flujo de objetos donde se visualiza la secuencia de las operaciones y relaciones que existe entre objetos del sistema.

En esta parte se define la categoría de los objetos y la forma en que se asocian. Entonces, la secuencia general de los objetos y métodos esta basada en el siguiente diagrama.



**Figura 5.6.** Secuencia de los Objetos del Sistema.

Por tanto la estructura de los anteriores objetos con sus respectivas relaciones detalladas se muestran en el siguiente diagrama.



**Figura 5.7.** Diagrama de Objetos internos y externos al sistema.

**5.2.1. Diagrama de Flujo de Objetos**

Este tipo de diagrama de flujo de objetos está enfocado al comportamiento interno de los objetos del sistema, además podemos visualizar las relaciones entre los distintos objetos y el modelo total del sistema.

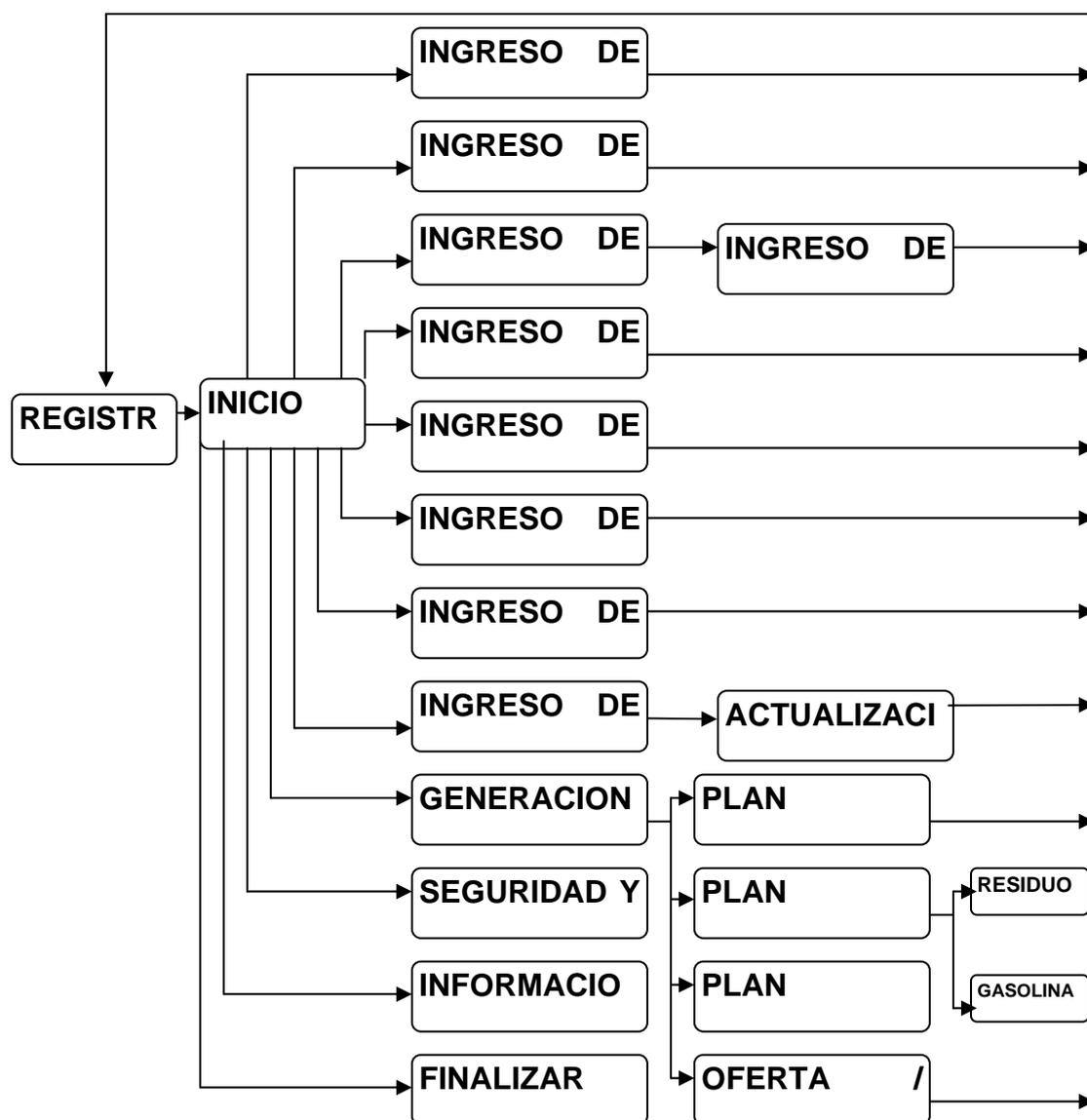


Figura 5.8. Diagrama de flujo de objetos del sistema

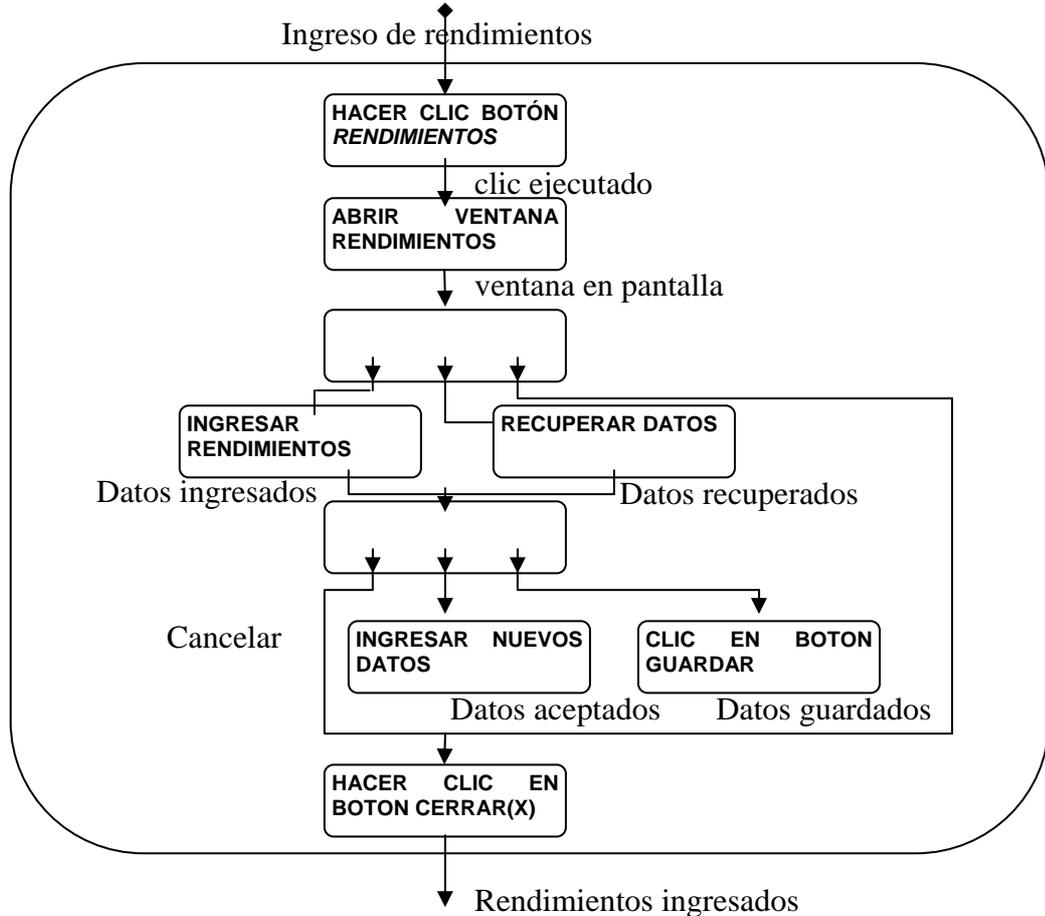
**Descripción.-** El sistema se compone de varios procesos con los cuales se ha diseñado el flujo de los objetos en el siguiente orden:

1. Ingreso de rendimientos
2. Ingreso de días de operación
3. Ingreso de residuos
4. Ingreso de Hidrocarburos
5. Ingreso de Despachos
6. Ingreso de Demandas

7. Ingreso de Stock
8. Ingreso de Carga
9. Generación de Reportes
10. Seguridad en acceso
11. Información técnica de refinerías
12. Finalizar sistema

Además se debe considerar lo siguiente:

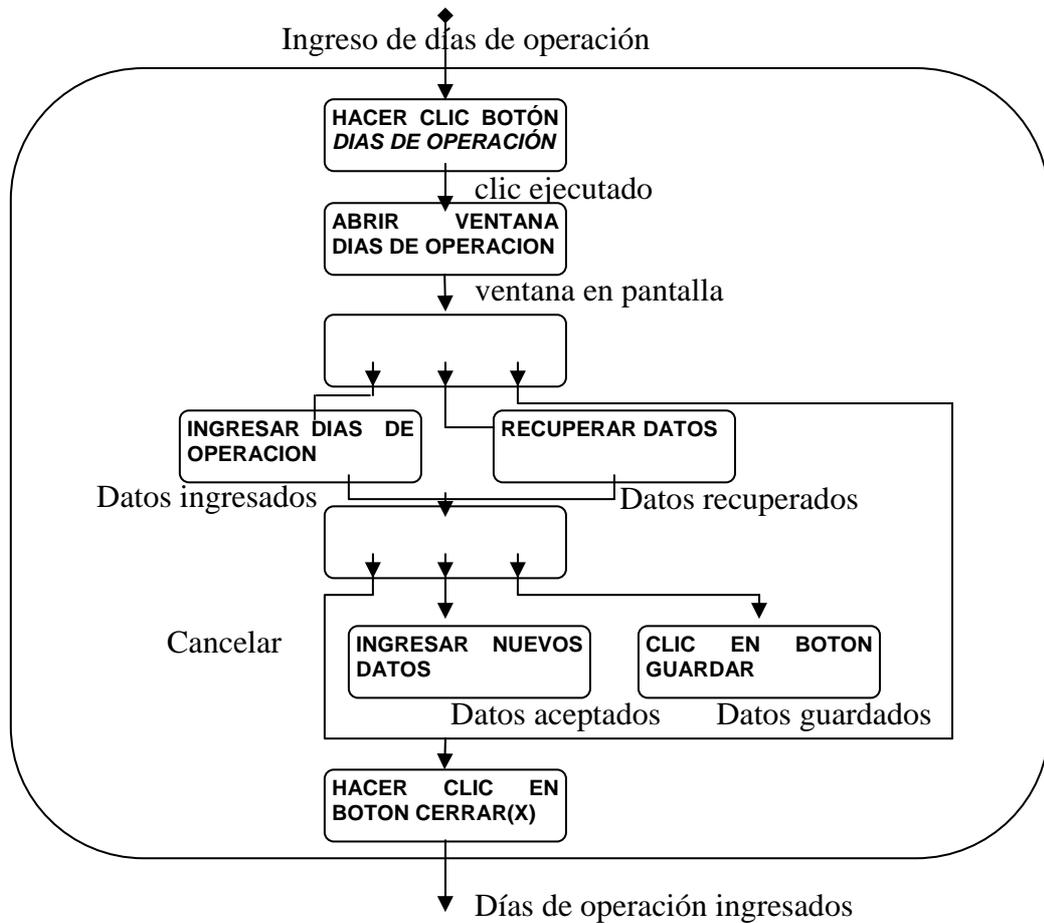
1. Los datos pueden ser ingresados o recuperados directamente desde la base de datos.
2. Cada vez que grabe la información de un módulo actual; automáticamente se guardan también los datos de los otros módulos (aunque estén vacíos), es decir no se puede guardar individualmente.
3. Los campos solicitados al momento de guardar son:
  - Nombre del proyecto (hasta 50 caracteres)
  - Descripción
  - Fecha (se recupera automáticamente del sistema)

**Esquema +1, nivel 1: INGRESO DE RENDIMIENTOS**

**Descripción:** Para el proceso de ingreso de datos sobre rendimientos de los productos de cada refinera, se debe tomar en cuenta:

1. Diferenciar datos de rendimientos de las tres refineras existentes.
2. Los datos de rendimientos son sólo valores porcentuales(%).
3. Únicamente se ingresa datos específicos de rendimientos sin códigos ni descripción alguna, por la razón de que salen numerosos productos de cada refinera.

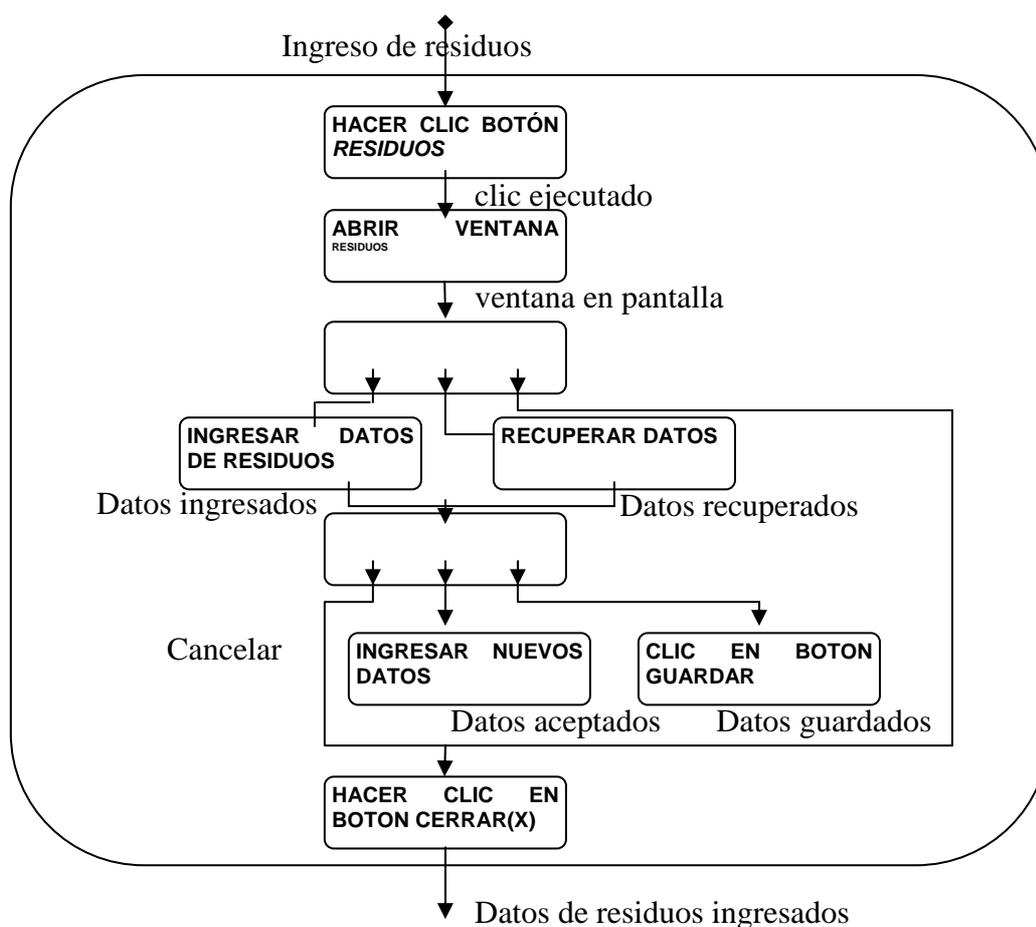
## Esquema +2, nivel 1: INGRESO DE DIAS DE OPERACION



**Descripción:** En este módulo, el ingreso de días de operación se debe hacerlo por cada planta o unidad de las refinerías, y se toma en cuenta lo siguiente:

1. Los datos de los días de operación son mensuales y representan el tiempo de funcionamiento de cada refinería.
2. Tomar en cuenta los años bisiestos, puesto que un día al no ser tomado en cuenta implica una gran baja en el rendimiento de la refinería.

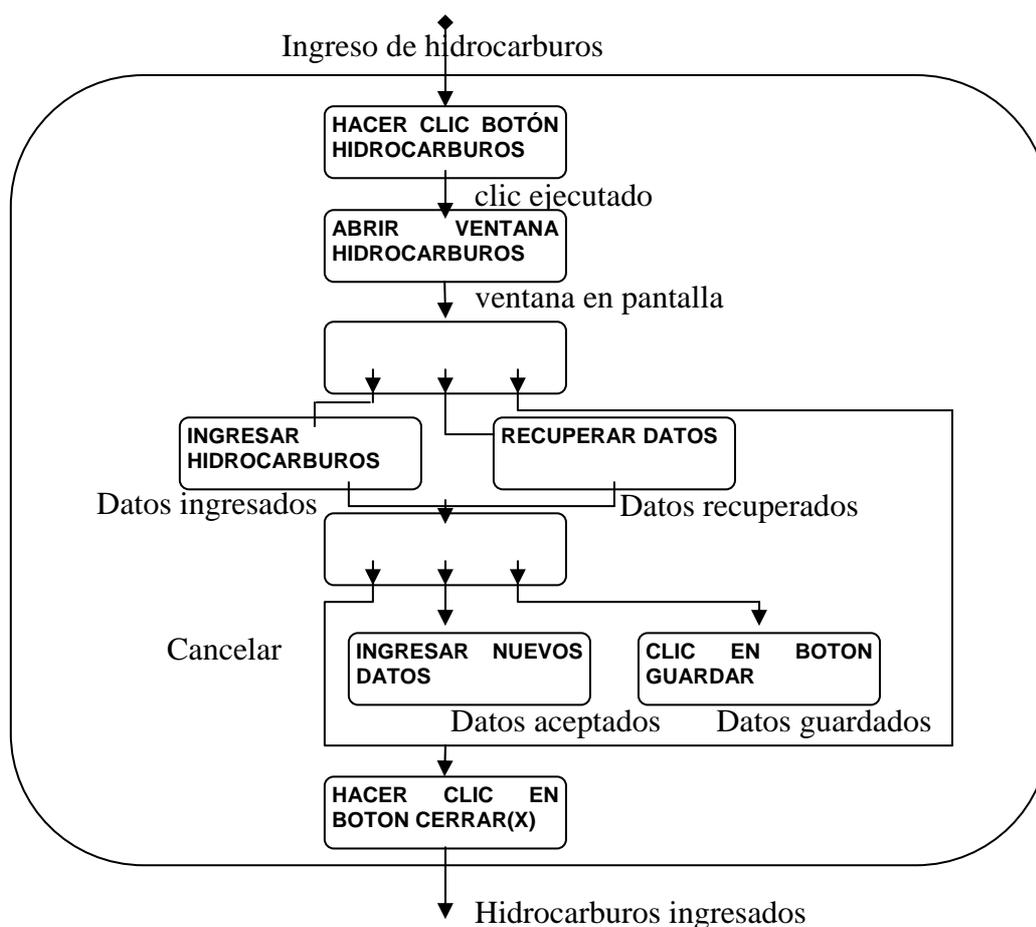
## Esquema +3, nivel 1: INGRESO DE RESIDUOS



**Descripción:** Los residuos son valores que pertenecen sólo al módulo de la Refinería Esmeraldas y se verifica:

1. Deben ser lo más aproximados, ya que son datos de productos que luego de ser procesados se obtienen nuevos productos.
2. Representan cargas de productos a cada refinería y en su mayoría se ingresa en barriles por día (BLS/DIA)

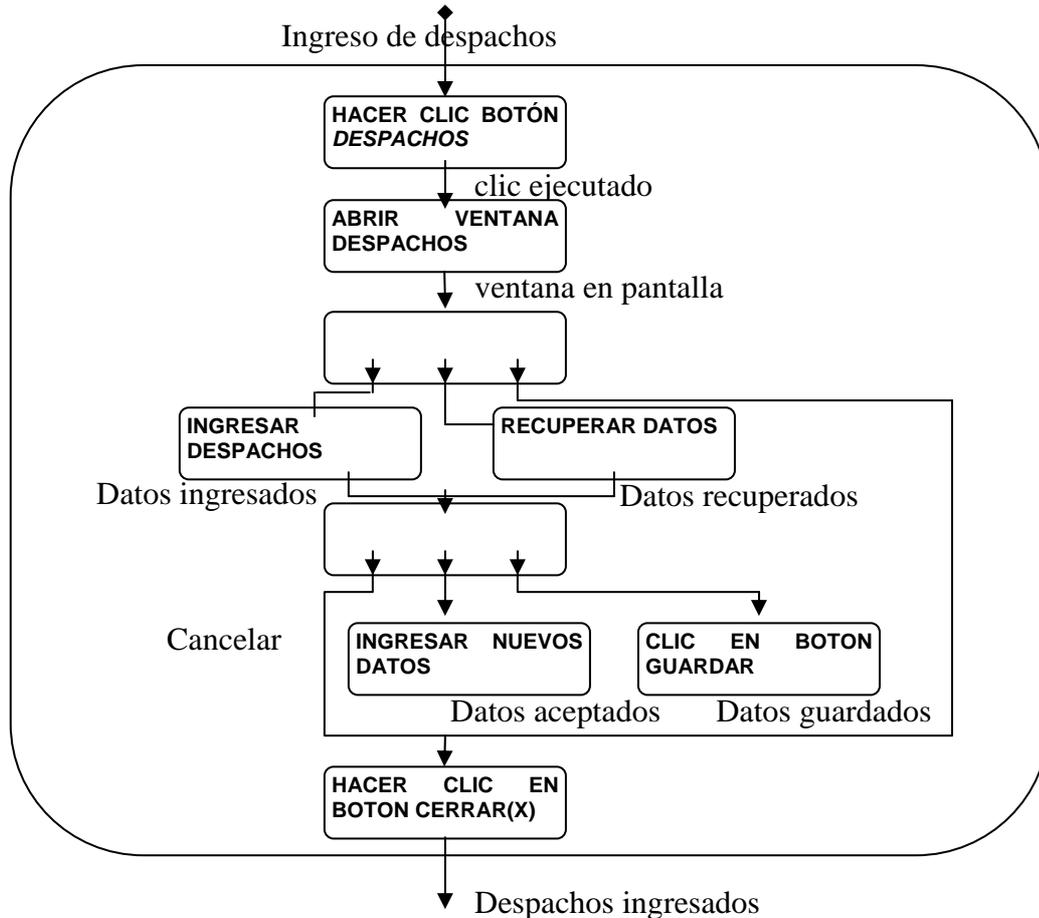
## Esquema +4, nivel 1: INGRESO DE HIDROCARBUROS



**Descripción:** Los datos de hidrocarburos son muy importantes ya que se rigen por las normas INEN. Se debe tomar en cuenta:

1. Se verifica e ingresa los valores de los distintos tipos de hidrocarburos.
2. Lo datos ingresados no deben salirse de las normas establecidas por el Instituto Ecuatoriano de Normalización (INEN).

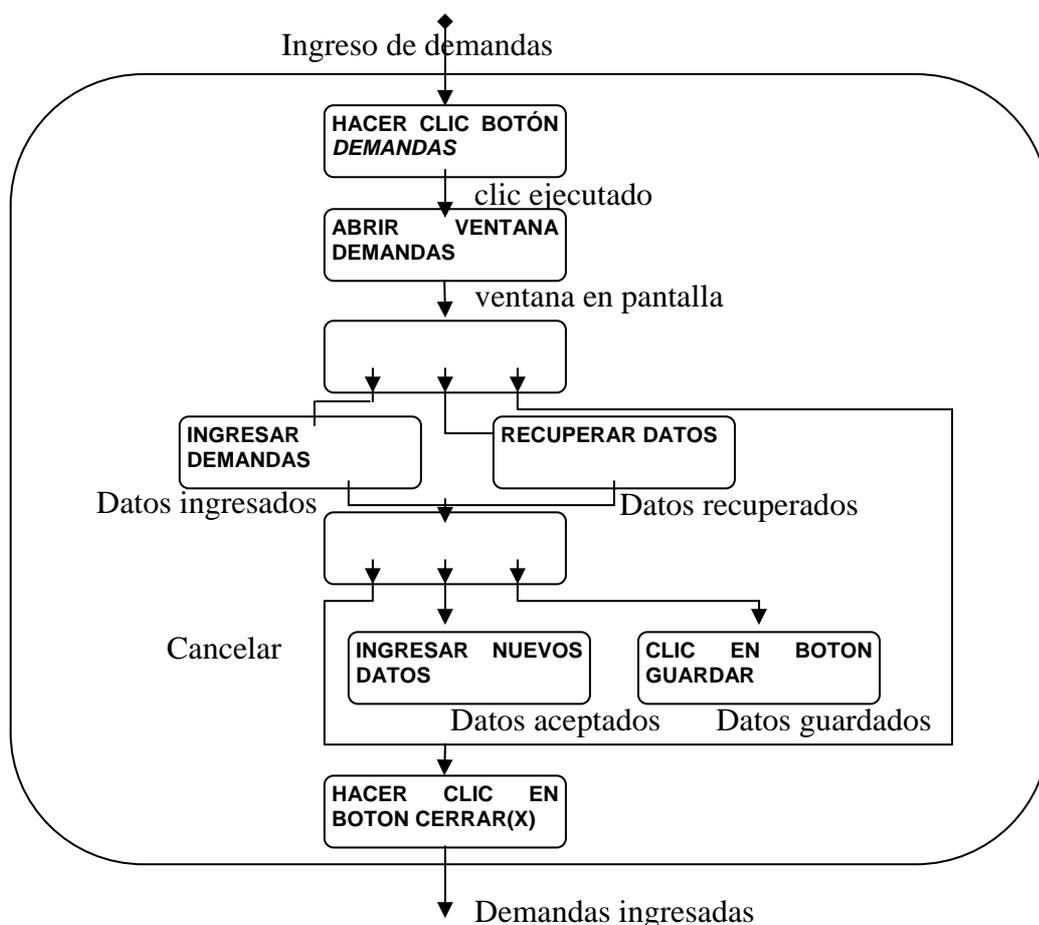
## Esquema +5, nivel 1: INGRESO DE DESPACHOS



**Descripción:** El proceso para ingresar los despachos, implica:

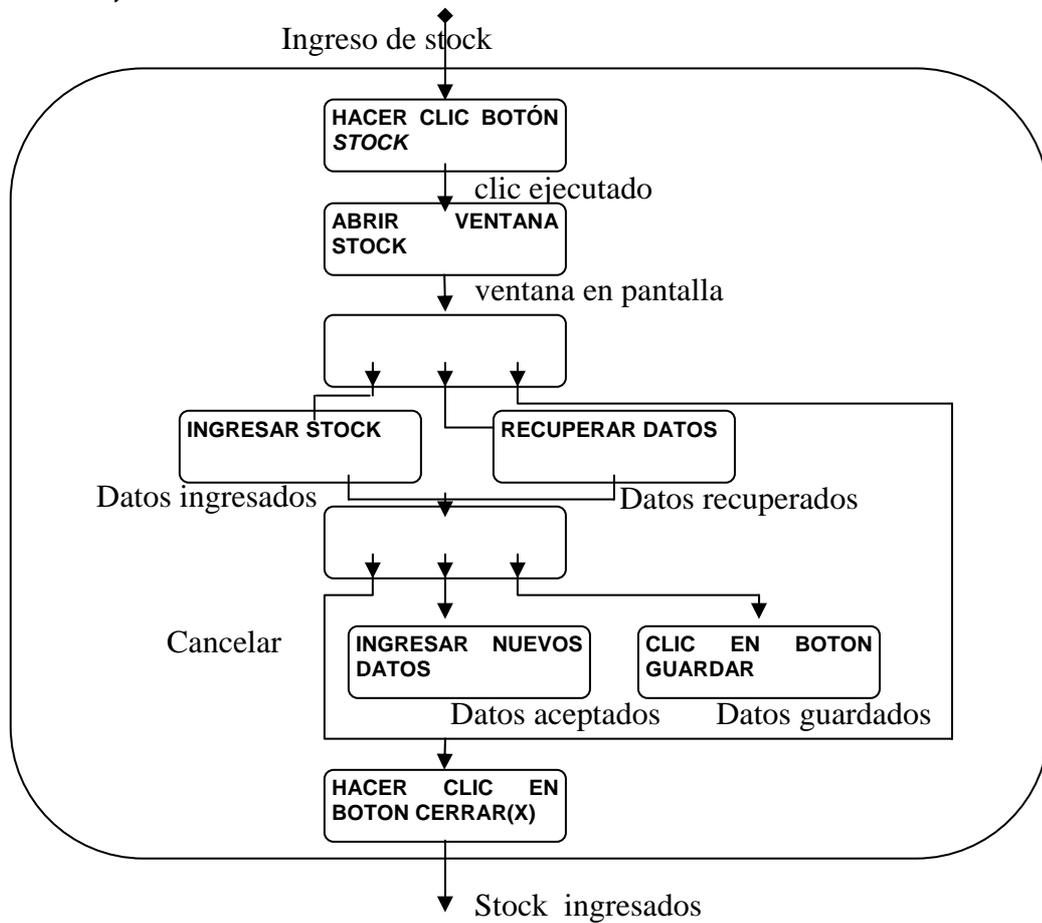
1. Reconocer los datos de tres productos importantes de refinería, como son: diesel 1, jet fuel y absorber oil.
2. Estos datos son porcentuales y entonces la suma de sus proporciones debe ser uno(1).

## Esquema +6, nivel 1: INGRESO DE DEMANDAS



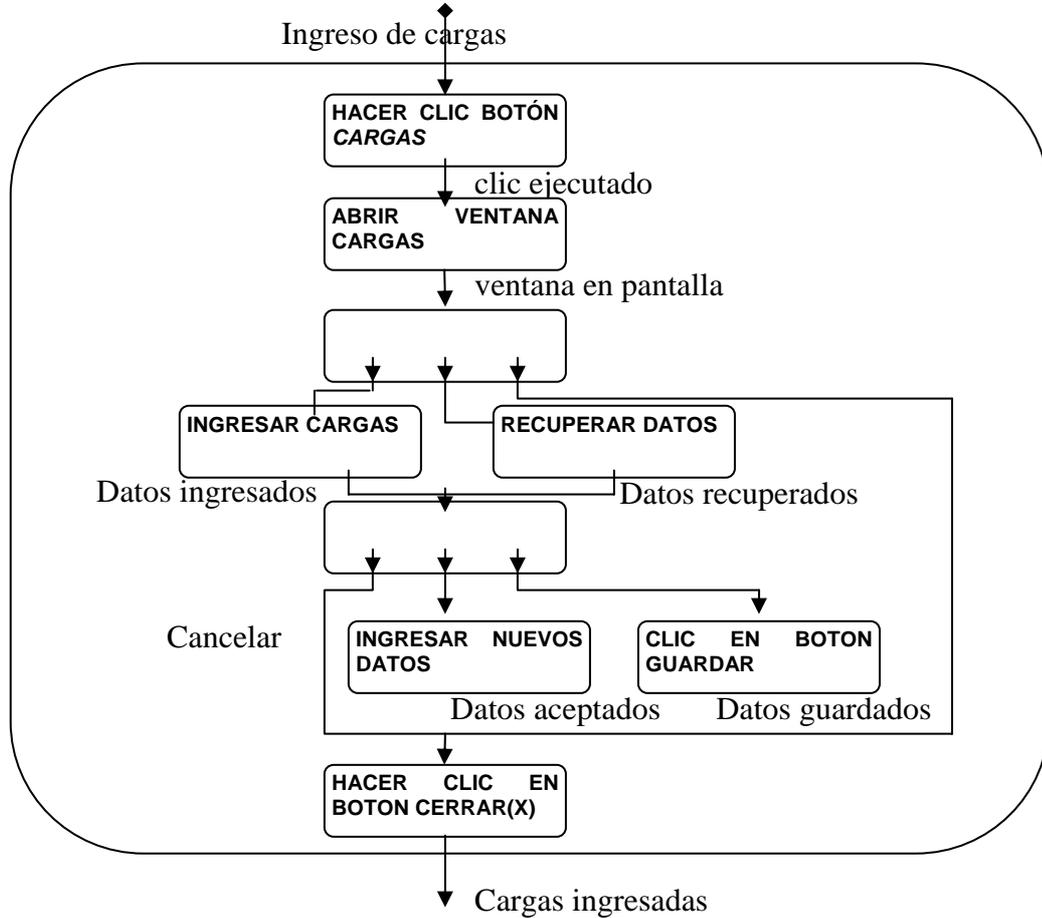
**Descripción:** La secuencia que se sigue para este proceso y sus consideraciones son:

1. Los valores representan a los productos que se venden o exportan.
2. Los datos de las demandas deben ser ingresados en dólares(USD)
3. Como son valores altos; en los decimales se aproxima al inmediato superior o inferior, quedando un dato entero.

**Esquema +7, nivel 1: INGRESO DE STOCK**

**Descripción:** La secuencia de pasos para cumplir con este módulo de ingreso es:

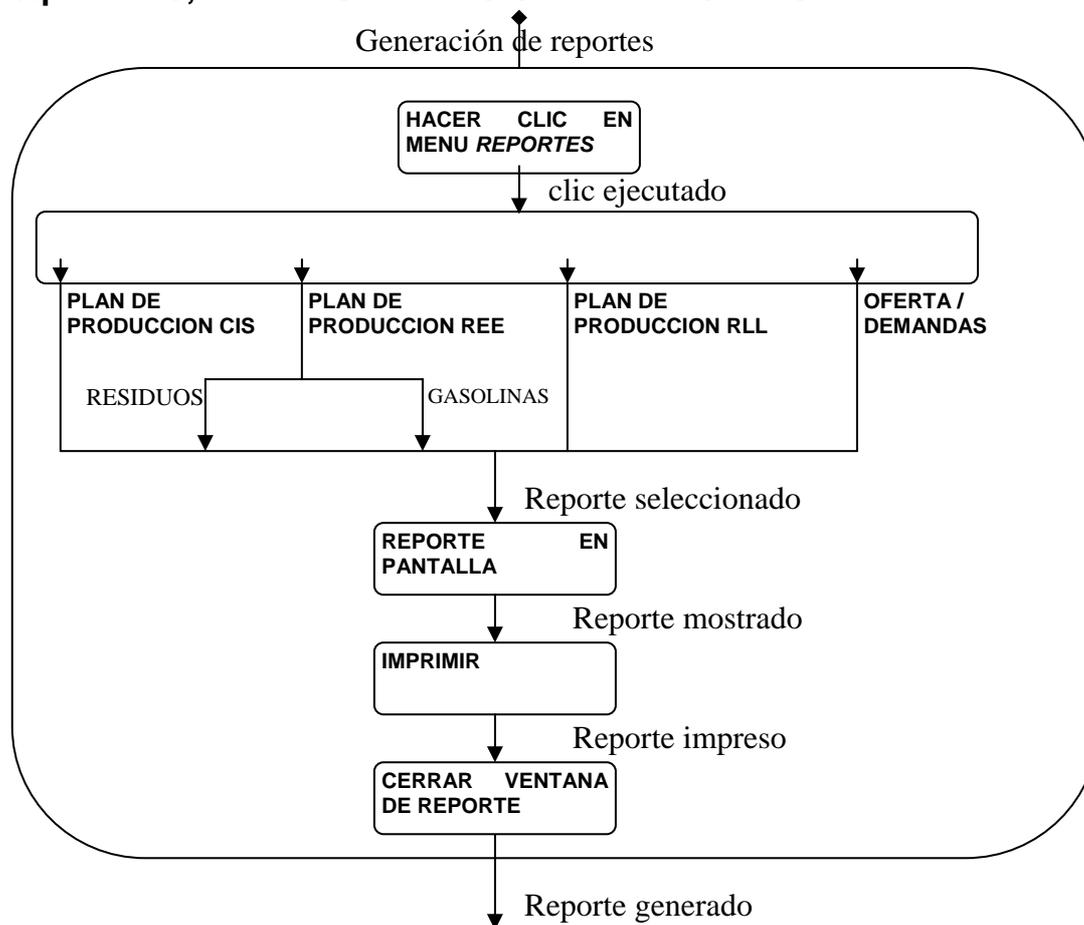
1. Verificar los datos que representan al Stock de los productos
2. Los datos de Stock hacen referencia a los productos de las tres refinerías

**Esquema +8, nivel 1: INGRESO DE CARGAS**

**Descripción:** El ingreso de cargas, implica conocer y/o consultar las nuevas especificaciones de carga que rigen para cada unidad de refinería.

Se toma en consideración:

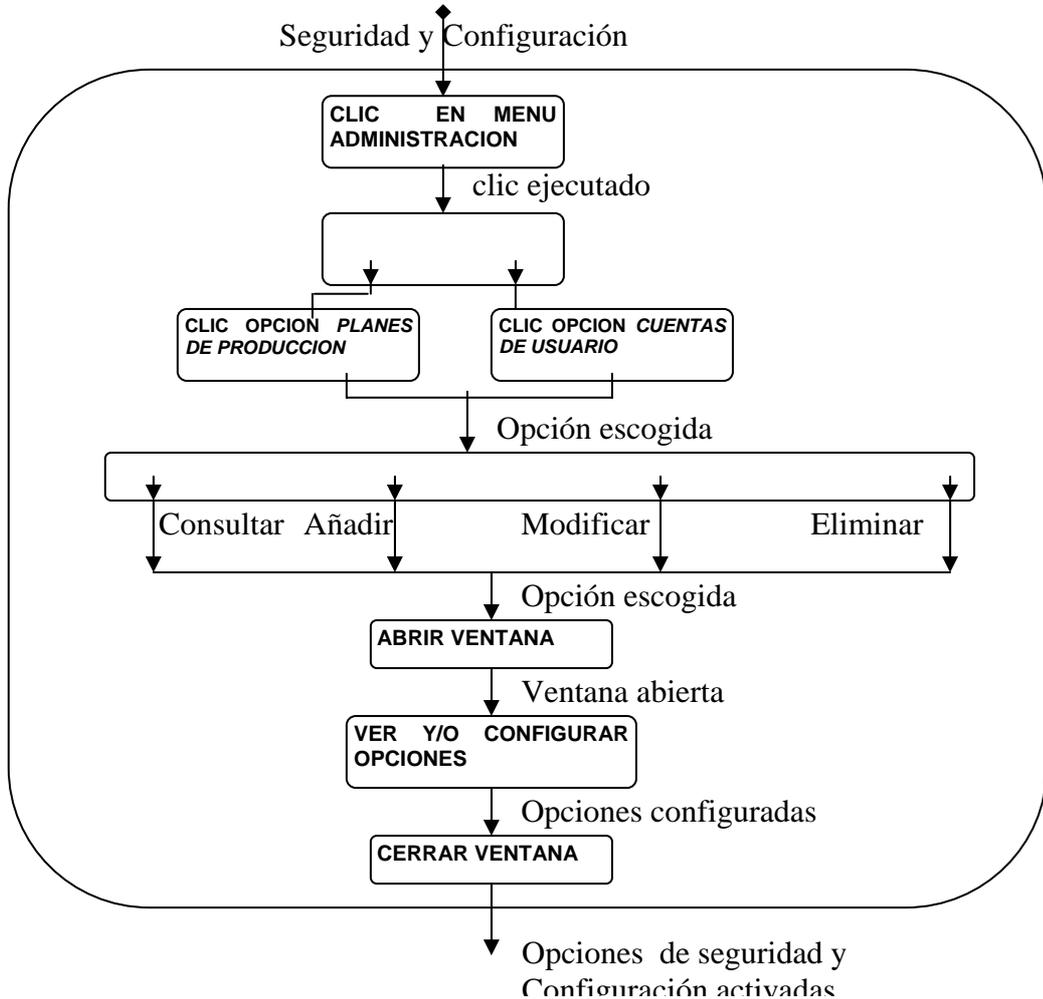
1. Verificar y/o modificar los datos de carga de crudo en la ventana de información técnica.
2. Los campos necesarios para cada refinería son:
  - Capacidad de carga (valor entero)
  - Días de operación (valor entero, cuantificados en un año)
  - Porcentaje de funcionamiento
3. El campo de cargas representa la capacidad que tiene cada unidad de proceso de las refinerías y por tanto casi no varía.

**Esquema +9, nivel 1: GENERACION DE REPORTE**

**Descripción:** Generar los diferentes reportes de datos por cada refinería implica:

1. Tener generados previamente los respectivos planes de producción, residuos, ofertas y demandas como también las gasolinas.
2. También se implementa como reporte toda la información ingresada a cada módulo, con el fin de tener datos fuentes y procesados.
3. Existen dos formatos para los reportes (vista como documento y el otro como tabla o cuadrícula).

## Esquema +10, nivel 1: SEGURIDAD Y CONFIGURACION

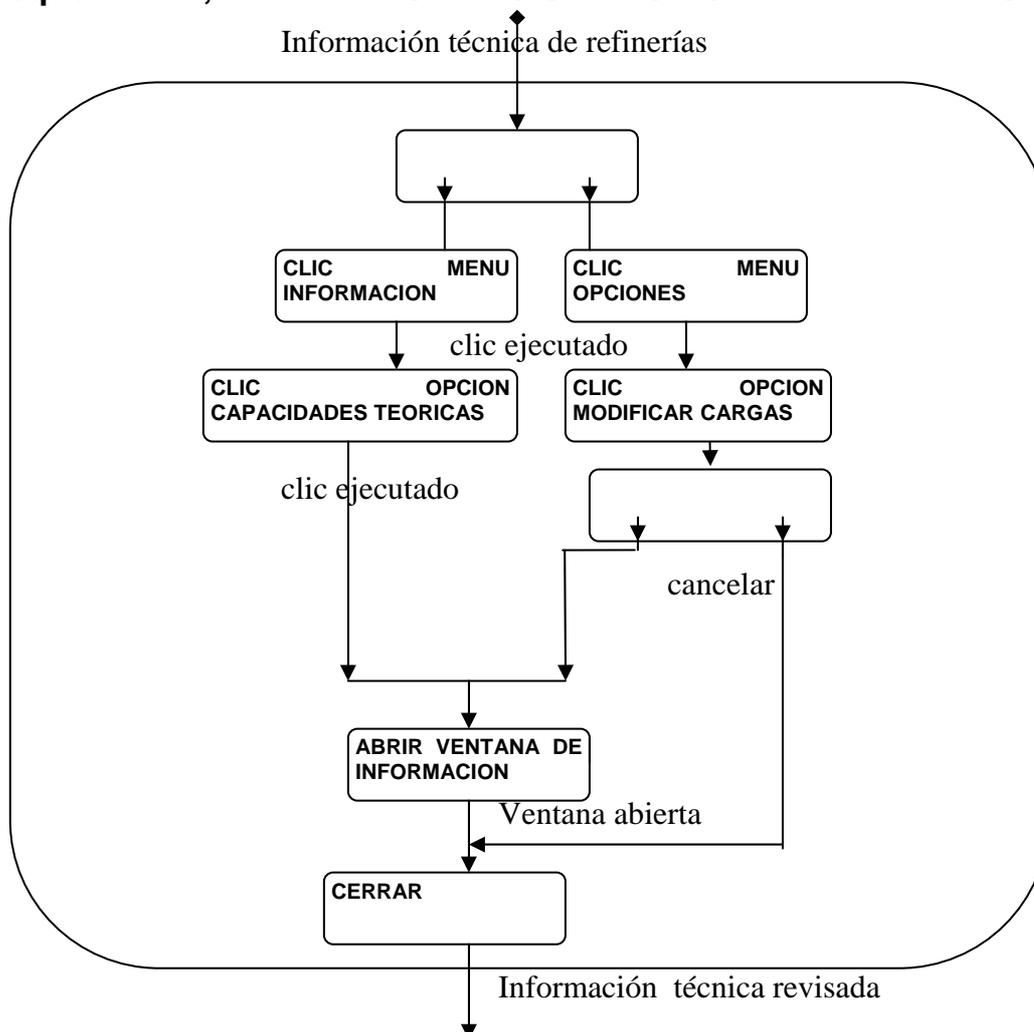


**Descripción:** Este proceso tiene dos enfoques de administración, a nivel de los mismos resultados que genera el sistema y a nivel de seguridad de la información.

Considerar los siguientes puntos:

1. Toda opción modificada afecta permanentemente desde el momento de activarla en adelante.
2. Puede tener acceso a todas las opciones de este módulo siempre y cuando ingrese al sistema como un usuario con privilegios de administrador.

## Esquema +11, nivel 1: INFORMACION TECNICA DE REFINERIAS



**Descripción:** La revisión de la información técnica es un proceso que toma en cuenta las siguientes notas:

1. Son datos recopilados directamente desde refinерías y en la mayoría de los casos no varía de un período a otro.
2. Todo cambio a estos valores afectará a todos los proyectos que se carguen en adelante, ya que se trata de información que se actualiza en la base de datos.

### 5.3. Definición de los elementos del plan de producción

Se revisa a continuación algunos procesos que permiten calcular por medio del computador el estado actual de las operaciones de las refinerías en cuanto a producción y comercialización (oferta y demanda)

**Rendimientos.-** Es el valor que indica la cantidad de producto o derivado que se ha obtenido a partir de un crudo ingresado a refinería; así tenemos como ejemplo: El rendimiento de Diesel es 0.23 (significa que del total de carga de crudo en barriles se a generado un rendimiento del 23%). Descrito como fórmula queda así:

$$\frac{\text{cantidad de producto extraído}}{\text{carga total}}$$

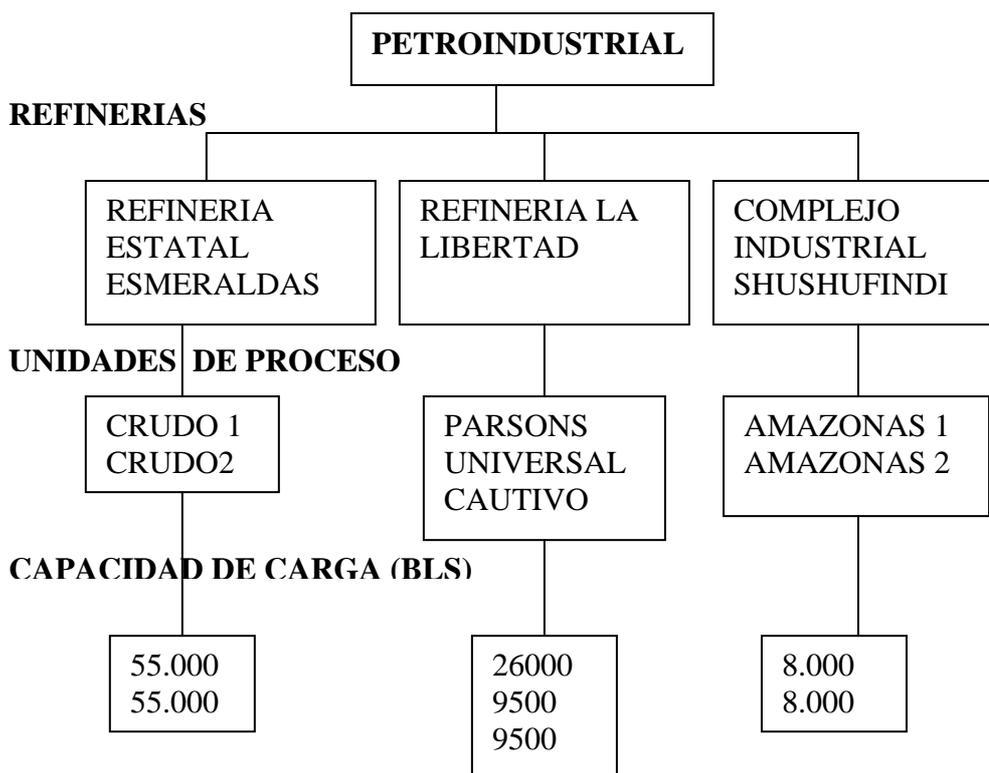
**Días de Operación.-** Este parámetro contiene los días de operación o funcionamiento en cada uno de los meses que ha tenido las diferentes unidades de proceso de cada refinería. Así, como ejemplo se anota: la planta cautivo en el mes de enero ha tenido 19 días de operación, en febrero 21 días, etc. Estos datos se registran en un formulario que se indica a continuación:

ENERO	FEBRERO	MARZO	ABRIL	MAYO	JUNIO	JULIO	AGOSTO	..	
19	21	31	30	30	30	31	30		

**Tabla 5.2.** Formulario para ingreso de días de operación.

Se lleva este registro, porque muchas veces las plantas de producción se paran por distintas situaciones como: daños en los equipos, mantenimiento de maquinaria, no funciona al cien por ciento de capacidad o por disposiciones gubernamentales.

En el siguiente gráfico se indica como están distribuidas las tres refinerías de Petroindustrial junto con sus unidades o plantas de proceso y sus respectivas capacidades de carga.



**Figura 5.9.** Distribución de las tres refinерías ecuatorianas.

**Residuos.-** El crudo que es ingresado a una refinерía produce diferentes derivados pero una vez procesado, siempre existe un sobrante o residuo que es nuevamente procesado para sacar nuevos derivados. Los valores de residuo pertenecen a cada unidad de proceso. Ejemplo: la unidad Crudo Reducido 1 ha tenido un residuo de 0,532, que equivale al 53% del crudo total. Aquí también se toma en cuenta las cantidades de producto que se ha comercializado por la empresa (Termoesmeraldas)

**Hidrocarburos.-** Son compuestos orgánicos que están presentes en su mayoría en todo crudo, y por tanto, luego del refinamiento del crudo, se encuentran presentes en los derivados, pero la composición de hidrocarburos en un derivado se rige por normas establecidas por el Instituto Ecuatoriano de Normalización (INEN), así como por ejemplo: la gasolina extra no debe tener en su composición más de 80 octanos, 20 de aromáticos, 0.75 de benceno y 16.6 de olefinas. Estos datos son ingresados al sistema para su procesamiento.

**Despachos.-** Se refiere al porcentaje de participación de las refinerías en los diferentes despachos de derivados. Por tanto sólo se considera el despacho de los principales productos, a saber: diesel 1, gasolina, lpg, entre otros.

**Demandas.-** Son valores que representan a las demandas de todos los derivados de petróleo producidos en las refinerías. Las demandas se registran mensualmente y son calculadas en dólares americanos (USD). Como ejemplo, se puede decir que la demanda de Naftas en el mes de enero es de 949.348 barriles.

**Stocks.-** Están determinados por la cantidad de derivados que queda en inventario. Se contabiliza en barriles y además se unifica los stocks de los productos de las tres refinerías. Ejemplo: Gasolina Super tiene en stock 35.773 BLS.

**Cargas.-** Son valores muy importantes para el cálculo de los distintos planes de producción. Aquí, se especifica la capacidad de carga de cada unidad de proceso, los días de operación durante el año o período y el porcentaje de funcionamiento de las refinerías.

Podemos decir que esta información nos permite generar y evaluar los diferentes planes de producción, basándonos en fórmulas y ecuaciones básicas para cálculos de ofertas, demandas, capacidades de carga, operatividad de las refinerías, entre otros. Como ejemplo, para el cálculo de la carga anual de las unidades de proceso se utiliza la siguiente fórmula:

$$\text{Capacidad\_unidad1} = \frac{(\text{carga\_nominal} * \text{días\_operación} * \text{porcentaje\_operación})}{100}$$

donde;

carga\_nominal es el valor que representa la capacidad en barriles de una unidad de proceso.

días\_operación representan al número de días en el año que la unidad de proceso ha permanecido operativa.

porcentaje\_operación valor porcentual promedio que refleja el tiempo de funcionamiento de la unidad de proceso.

## **5.4. Estudio del Lenguaje de Programación para el control de Crudos**

Es indispensable hacer uso de herramientas de programación que sean multiplataforma debido a que las empresas actualizan continuamente sus sistemas informáticos. Una de estas herramientas es el lenguaje de programación PowerBuilder, cuyas versiones ya vienen para Windows , Sun, Solaris, HP-UX, Macintosh e IBM AIX.

### **5.4.1. PowerBuilder como herramienta de desarrollo de aplicaciones**

PowerBuilder es una excelente herramienta para desarrollo empresarial orientada a objetos que permite construir diferentes tipos de aplicaciones, como cliente/servidor, aplicaciones distribuidas y aplicaciones para internet.

En PowerBuilder se realiza todos los procedimientos y funciones que involucran el tratamiento de datos de crudo, ya que ofrece más rendimiento y precisión en el cálculo de operaciones, además de la facilidad que presenta su entorno de programación.

Su fácil conectividad con otras herramientas de programación para el almacenamiento de datos como en este caso es SQL Server, hace que los datos puedan ser tratados o procesados de una forma segura y transparente. Es por estas razones que PowerBuilder ha llegado a ser un software muy confiable y competitivo frente a las demás herramientas de desarrollo.

### **5.4.2. Versiones del lenguaje de programación PowerBuilder**

Este software viene en tres versiones comerciales:

PowerBuilder Enterprise, Professional y Desktop; de las cuales se hará uso de la versión Enterprise(Empresarial); esta versión es muy completa debido a que posee todas las funcionalidades disponibles del producto.

A continuación se detalla cada versión comercial del producto.

*Enterprise.*- Está orientada al análisis y diseño, desarrollo de aplicaciones distribuidas y de cliente/servidor, así como también para la generación de componentes.

*Professional.*- Es útil para desarrollo a nivel departamental o área, donde se aplique el trabajo en grupo (ObjectCycle) y no sea necesario el acceso nativo a la base de datos.

*Desktop.*- Diseñada para desarrolladores independientes que realicen aplicaciones monousuario y que únicamente tenga acceso a base de datos ODBC.

Una de las versiones actuales que esta siendo distribuida por la casa Sybase es la 9.0. Al respecto, su entorno de programación ha cambiado significativamente en comparación con las versiones anteriores. Más adelante se detalla las principales características de este lenguaje.

#### **5.4.3. Características del Lenguaje de Programación PowerBuilder**

La razón de escoger este lenguaje, es por su alto rendimiento y confiabilidad de resultados, que lo han ubicado en uno de los preferidos en el mercado. Prueba de esto citaré algunas de sus características.

- Orientación a objetos
- Facilidad de acceso a datos almacenados (Data Window)
- Es un lenguaje PowerScript (módulos de código)
- Posee una herramienta de control de rendimiento
- Editor de código configurable
- Infomaker: Generador de informes
- Conectividad ODBC
- Tiene todos los drivers nativos de PowerBuilder
- Soporte al API de Microsoft para el control de versiones, entre otras.

#### 5.4.4. Entorno de programación de PowerBuilder 9.0

El entorno de programación permite realizar cualquier aplicación crítica de negocio, cuyos resultados también se pueden visualizar en entornos Web y otros formatos.

Como se mencionó anteriormente, esta versión viene con algunas utilidades adicionales como son: dos paneles o ventanas; la una que detalla el *árbol de sistema* en donde se muestra la ruta y los nombres de todos los objetos que se hayan creado y salvado, y la otra utilidad es la *ventana clip* que almacena las acciones de copiado y pegado para que después puedan ser utilizadas.

Además de los componentes mencionados anteriormente se han incorporado opciones de depuración, configuración y ejecución de proyectos, todo esto se lo puede visualizar en la barra principal (powerbar).

Cada programa creado en PowerBuilder posee un espacio de trabajo (workspace), en donde se guardan objetos propios del lenguaje que conforman el sistema, tales como aplicaciones, ventanas, menús, objetos de usuario, datawindows, componentes, etc.

#### 5.4.5. Principales funcionalidades de PowerBuilder 9.0

Se presenta a continuación una breve descripción de las actuales funcionalidades de powerbuilder:

- Web Services. Permite construir y publicar servicios web para Microsoft .NET y otras redes.
- RAD JavaServer Page (JSP) Targets. Permite crear componentes JSP.
- XML DataWindow. Facilita la importación de datos de un documento XML.
- PowerBuilder Native Interfaz. Admite la incorporación de clases del lenguaje C++, y así acceder a sus métodos en la aplicación.

- Servidores de Aplicación de terceras partes. Soporta BEA WebLogic ServerTM, IBM WebSphere y Servidores de Aplicaciones J2EE
- PowerBuilder Document Object Model. Permite manipular los documentos XML desde el código PowerScript.
- Creación de Informes PDF. A través del objeto DataWindow se puede generar informes en formato PDF..

#### **5.4.6. Tipos de Instalación de PowerBuilder**

En todas las versiones de PowerBuilder vienen tres tipos de instalación: compacta, avanzada y típica.

*Compacta.*- Esta instalación ahorra el espacio en disco, debido a que instala los requerimientos mínimos para el funcionamiento del paquete.

*Avanzada (custom).*- Debe ser guiada por usuarios que conocen de todas las funcionalidades del paquete, es el tipo de instalación que permite seleccionar los componentes que serán instalados.

*Típica.*- Es una de las instalaciones que la mayoría de los usuarios la utilizan debido a que en disco se graban las utilidades más comúnmente utilizadas por los desarrolladores.

#### **5.4.7. PowerBuilder como administrador de datos externos**

PowerBuilder permite trabajar con datos almacenados en una aplicación externa. Los datos son recuperados, manipulados y presentados por medio de una interfaz gráfica adecuada para el estudio de crudos, la misma que es creada utilizando el lenguaje de programación en mención.

Debido al gran potencial de sus utilidades; en especial la de acceso a datos como es el objeto y control DataWindow; PowerBuilder puede administrar datos externos de una forma fácil e interactiva, en donde el usuario únicamente realiza sus operaciones de ingreso, recuperación, modificación de datos navegando por la ventanas que están diseñadas para el efecto.

## **5.5. Determinación de Procedimientos y Métodos de programación**

Gran parte de este proyecto tiene acceso a datos externos, que en este caso están almacenados en la conocida base de datos SQL Server. La mayoría de estos datos son numéricos tanto enteros, flotantes y de valores porcentuales.

El aplicativo constará de varias funciones o métodos que en el capítulo anterior se las mencionó y que controlarán los resultados en base a los procedimientos seguidos para la evaluación de crudos y mezclas.

En esta sección se analiza los procedimientos que intervienen en la parte del aplicativo que se refiere a los ensayos de crudo y en la siguiente sección se estudia la parte del plan de producción.

Los métodos que se implementan en esta parte del sistema se resumen en el siguiente tema.

### **Métodos principales**

Los módulos o funciones principales que se han tomado en cuenta para el correcto tratamiento de crudos, son:

#### **5.5.1. Módulo de clasificación de crudos**

En esta parte se toman en cuenta el tratamiento de valores de grado de calidad de crudos, que permitirán la clasificación de estos crudos de una forma muy clara y precisa. Para luego si es necesario utilizarlos en mezclas de crudos.

### **5.5.2. Módulo de Conversión de datos de crudos**

La mayoría de datos de crudos necesitan ser transformados a otros tipos de datos para el manejo de fórmulas y un correcto seguimiento de un mismo sistema de medida. Así como por ejemplo valores de grados centígrados deben ser transformados en Fahrenheit, valores porcentuales en decimales, etc.

### **5.5.3. Módulo de validación de crudos**

Aquí se valida todos los datos de ensayos de crudos, es decir tiene que ver mucho la exactitud de cada dato debido a que los resultados finales se verían afectados en gran medida. Además se valida los procedimientos involucrados en el tratamiento de crudos.

### **5.5.4. Módulo de Evaluación de crudos**

Se definen todas y cada una de las fórmulas y funciones ya establecidas para manejo de crudos y a la vez se verificará la validez de datos resultantes con datos reales (datos reportados desde refinerías), para un control estricto de ensayos de crudos.

Cabe indicar que para una buena optimización del manejo de estas funciones, se harán presentes otro tipo de funciones para aligerar y presentar los datos de una forma segura, como pueden ser de verificación de ingreso de datos, depuración de algoritmos, tratamiento de errores, etc.

El sistema además consta de procedimientos que generan informes sobre los ensayos de los crudos que más se encuentran en el Ecuador. Este informe consta de todas las especificaciones dadas por los técnicos de las refinerías además de los datos resultantes que se han evaluado en el sistema.

Una vez generado los ensayos de crudos, se procede a presentar los respectivos programas de producción que son reportes de diferentes estudios de producción de derivados. Además, aquí se detallan valores sobre capacidad de barriles de crudo que se alimentan a las refinerías, así como también la disponibilidad de funcionamiento de cada refinería.

Todos los procedimientos y métodos están codificados con la ayuda de la herramienta PowerScript de PowerBuilder, la misma que contiene funciones e instrucciones que facilitan el desarrollo de los diferentes módulos y así se puede crear una interfaz y funcionalidad propia de una aplicación con metodologías de sistema experto.

### 5.6. Elementos de la programación orientada a objetos

Actualmente la mayoría de sistemas desarrollados se basan en la programación orientada a objetos. Este aplicativo también está encaminado a ese tipo de programación.

La programación orientada a objetos permite incorporar mejoras ya sea en el diseño, desarrollo y mantenimiento del software, esto se realiza gracias a que se fundamenta en el uso de dos elementos que son las *Clases* y los *Objetos*.

Una **clase** es la generalización del objeto, es decir cuando programamos un objeto y se define sus características y funcionalidades, lo que se está haciendo en realidad es dar forma a una clase.

Un **objeto** es la representación de una entidad, sea ésta real o conceptual y además tiene una estructura bien definida. Todo objeto cumple con las siguientes características:

**Abstracción.-** Significa que un objeto cumple con sus funciones sin importar en que ámbito ha sido creado.

**Encapsulación.-** Permite al objeto ocultar sus datos (código), y mostrar su funcionalidad por medio de sus métodos.

**Herencia.-** Indica que se puede crear objetos que tengan propiedades y métodos de otros objetos. En resumen, podemos crear objetos a partir de otros sin la necesidad de describir todo.

**Polimorfismo.-** Indica que el mismo método o propiedad puede realizar diferentes acciones.

Básicamente, un objeto esta conformado por tres partes que son: relaciones, propiedades y métodos.

La **relación** existe cuando un objeto esta vinculado a otro por medio de los punteros.

La **propiedad** permite diferenciar un objeto de otro, además las propiedades pueden ser heredadas a sus descendientes.

El **método** es la parte que permite dar operabilidad al objeto por medio de la ejecución de instrucciones específicas(código).

Se ha dividido el aplicativo en varios módulos o procedimientos, con el fin de abarcar toda su funcionalidad, y estos son: globales, planes de producción, panel de opciones, reportes, fórmulas y cálculos.

### 5.6.1. Módulos del sistema

Módulo Global
<b>Método</b> <ul style="list-style-type: none"><li>• Declaración de variables y constantes que son utilizadas para los diferentes cálculos y operaciones realizadas en los demás módulos</li><li>• Definición de variables que sirven como: inicialización, banderas y valores constantes (valor de Pi, días laborables, etc).</li></ul>

**Módulo Planes de Producción****Método**

- Identificar el o los datos de los productos que se relacionan con otros.
- Analizar la proporción de los productos con la que aportan en cantidad a la obtención del nuevo producto.
- Calcular los valores de oferta y demanda según parámetros ya establecidos (importaciones, exportaciones, consumo interno, transporte, etc).
- Permitir guardar los datos ingresados para la generación de nuevos planes de producción.

**Módulo Panel de opciones****Método**

- Acceder a los controles que permiten el ingreso de datos
- Ingresar la información a todos los módulos del sistema
- Generar los diferentes resultados mediante fórmulas ya establecidas.
- Esta dividido en la parte de ingresos y en la parte de generación de planes de producción de cada refinería.

**Módulo Reportes****Método**

- Presentación de la información como reporte en su propio formulario o ventana.
- Permitir generar reportes en dos tipos de formatos (tipo documento y tipo cuadrícula).
- Pueden ser impresos o grabados como archivos físicos en disco

<b>Módulo Fórmulas y Cálculos</b>
<p><b>Método</b></p> <ul style="list-style-type: none"> <li>• Aplicación de ecuaciones y fórmulas para la generación de planes de producción</li> <li>• Verificación de resultados, que estén dentro de las normas establecidas.</li> <li>• Mantener la consistencia de los datos de ingreso y los resultantes (proporcionalidad).</li> </ul>

### 5.7. Implementación de la metodología de sistemas expertos

La metodología de encadenamiento de reglas está aplicada al módulo de producción de naftas como se explicó brevemente en el Capítulo III (Estudio de Sistemas Expertos y su aplicación en el área del petróleo).

En esta parte se explica los pasos necesarios para emplear dicha metodología en el cálculo de naftas. La siguiente tabla muestra los datos de ingreso con los que se realiza las respectivas operaciones a fin de obtener los volúmenes de producción.

RON
AROMATICOS
BENCENO
OLEFINAS
PRODUCCION NAFTAS
CARGA
REFORMADORA

**Tabla 5.3.** Lista de datos de ingreso

La tabla 5.4. está conformada por los datos que se va a presentar como parte del plan de producción. Esta información también es calculada para las tres diferentes gasolinas que son : gasolina extra, gasolina super y excedente de naftas.

NAFTAS	VOL	RON	AROM	BENC	OLEF	RON MEZ	ARO MEZ	BEN MEZ	OLEF MEZ
LIVIANA									
PESADA									
REFORM1									
REFORM2									
TRATADA									
NAO.IMPOR									
BASE LIB.									

**Tabla 5.4.** Datos generales para la producción de gasolinas

### 5.7.1. Restricciones técnicas en los resultados

Los valores de Ron mezcla, Aromaticos mezcla, Benceno mezcla y Olefinas mezcla, están restringidos según las normas INEN, que especifican el rango de valores que pueden tener. Así, se menciona a continuación dichos intervalos:

RON: 80 a 80.5 on

Aromáticos: 20 a 20.5 %vol

Benceno: 0.75 a 0.80 %vol

Olefinas: 16.50 a 16.60 %vol

### 5.7.2. Proceso para el cálculo de gasolinas

A partir de los datos de ingreso se realiza los siguientes pasos.

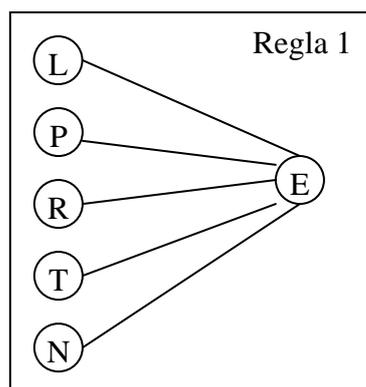
1. Aplicar el encadenamiento de reglas en la producción de gasolinas
2. Calcular las mezclas de los cuatro hidrocarburos.

3. Realizar el mismo procedimiento anterior para la gasolina super y excedente de naftas.
4. Verificar resultados finales.

#### *Aplicación del encadenamiento de reglas en la producción de gasolinas*

Como se revisó en el capítulo III, la red de reglas satisface a la solución del problema de producción de gasolinas. Ahora, se procede a pasar de la regla a una fórmula o procedimiento para así poder codificarla e implementarla en el aplicativo.

La figura 5.10 muestra una de las reglas que conforman el encadenamiento de reglas, y con la cuál se procede a analizarla para proceder a codificarla.



**Figura 5.10.** Regla 1, del encadenamiento de reglas

La regla 1 a su vez, puede separarse en sus respectivas reglas de inferencia, así:

$L \rightarrow E$ ;     $P \rightarrow E$     ;     $R \rightarrow E$     ;     $T \rightarrow E$     ;     $N \rightarrow E$

Partiendo de estas reglas de inferencia, se plantea el procedimiento a seguir para su codificación.

**Si** existe nafta liviana **entonces** se puede preparar gasolina extra

Si existe nafta pesada **entonces** se puede preparar gasolina extra

Ahora, para obtener el volumen de nafta liviana utilizada en gasolina extra, se procede aplicar la siguiente fórmula (con datos mensuales).

$$\text{Volumen nafta liviana en extra} = \left( \frac{\text{porcentaje de funcionamiento de la Refinería}}{100} \right) * \text{producción total de nafta liviana}$$

**Ejemplo:**  $V_{nle} = ( 97.2 / 100 ) * 108419 = 105384$  BPD (barriles por día)

Como sabemos, por la regla lógica 1 (Capítulo 3, figura 3.5), el total de nafta liviana se distribuye en gasolina extra y super, entonces se deduce de la anterior fórmula, que el restante de ese cálculo corresponde a la gasolina super. Indicado como fórmula, quedaría:

$$\text{Volumen nafta liviana en super} = \text{producción total de nafta} - \text{volumen nafta liviana en extra.}$$

**Ejemplo:**  $V_{nls} = 108419 - 105384 = 3036$  BPD (barriles por día)

Este mismo proceso que se realizó para calcular el volumen de nafta liviana se ejecuta también para calcular el volumen de las restantes naftas.

Sin más adentrarnos en las operaciones de sistemas expertos que se involucran en este proceso, cabe destacar que, también se utilizan conceptos de programación lineal como son la maximización y minimización de cargas a las refinerías y que tienen que ver mucho en los resultados finales.

*Calcular las mezclas de los cuatro hidrocarburos.*

Se realiza la operación de mezclas de hidrocarburos con el fin de conocer la proporción de cada hidrocarburo que se encuentran en cada nafta producida.

Esta parte se lo realiza mediante una fórmula que sirve para las mezclas de los cuatro tipos de hidrocarburos.

**Fórmula para mezcla:**

$$(\text{volumen de nafta} / \text{total de naftas}) * \text{hidrocarburo}$$

donde,

*volumen de nafta:* es el volumen de cada nafta producida

*total de naftas:* es la suma de toda la producción de naftas.

*Hidrocarburo:* es el valor de cada hidrocarburo

Así mismo esta fórmula se debe aplicar para las tablas de gasolina super y excedente de naftas.

Una vez realizados los cálculos de mezclas, se procede a sumarlas para comparar con las normas INEN para derivados.

Un ejemplo de este proceso se muestra a continuación:

NAFTAS	VOL	RON	AROM	BENC	OLEF	RON MEZ	ARO MEZ	BEN MEZ	OLEF MEZ
LIVIANA	1.180.638	72	0,77	0,22	2,53	16,20	0,17	0,05	0,57
PESADA	749.000	54	8,16	0,23	0,75	7,71	1,16	0,03	0,11
REFORM1	0	80	35,30	1,18	0,55	0,00	0,00	0,00	0,00
REFORM2	551.000	83	51,00	3,20	0,00	8,71	5,35	0,34	0,00
TRATADA	2428.237	90	26,20	0,64	30,17	41,64	12,12	0,30	13,96
NAO.IMPOR	340.000	91	25,00	0,60	30,00	5,89	1,62	0,04	1,94
BASE LIB.	0	63	8,70	0,00	0,00	0,00	0,00	0,00	0,00
<b>TOTAL</b>	<b>5.248.875</b>					<b>80,14</b>	<b>20,43</b>	<b>0,75</b>	<b>16,58</b>

**Tabla 5.5.** Ejemplo de los resultados finales para la gasolina extra

Como vemos los valores de los hidrocarburos se encuentran en el intervalo que exige las normas INEN (Ver Anexo D), por tanto son resultados válidos.

## 5.8. Implementación del módulo de ensayos de crudo

Un ensayo de crudo se conforma de una lista de especificaciones o propiedades, las mismas que permiten identificar la calidad de un crudo. Estas propiedades se las obtiene por medio de pruebas de laboratorio, realizando procesos de destilación y así revisando la composición de cada crudo. (Ver Anexo G).

Una vez realizadas las pruebas de laboratorio se obtiene información del crudo, la misma que viene hacer parte del ensayo, y contiene los siguientes aspectos:

1. Datos analíticos del crudo
2. Especificaciones del crudo
3. Curva TBP (True Boiling Point)

Los **datos analíticos del crudo** permiten saber directamente las características principales de dicho crudo, y por tanto dan una idea clara de la calidad del crudo, y estas son:

- Gravedad Específica a 60<sup>o</sup> F
- Gravedad API a 60<sup>o</sup> F
- Viscosidad a 100 (cSt)
- Viscosidad a 122 (cSt)
- Viscosidad a 140 (cSt)
- Azufre (% peso)
- Factor K UOP
- Punto de escurrimiento (°C)
- Punto de inflamación (°C)
- Contenido de sal (lb/MBIs)

En algunos ensayos se describen más características que en otros, pero las que acabamos de enlistar son las más importantes.

Dentro de las **especificaciones de crudo** están todas y cada una de las propiedades que pueda tener un crudo luego de pasar por un riguroso

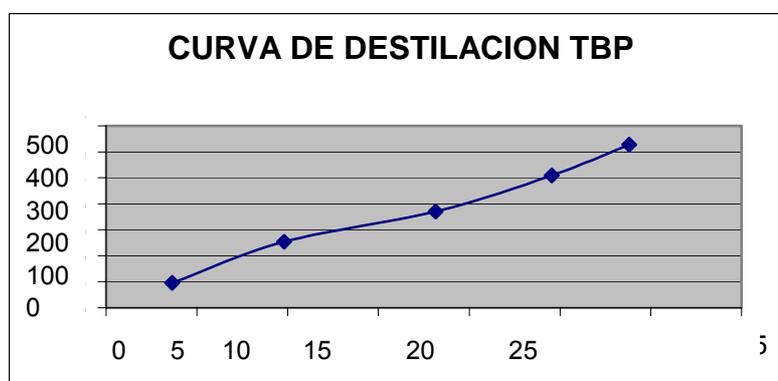
análisis de laboratorio. Sirven para revisar la composición del crudo a diferentes temperaturas en  $^{\circ}\text{C}$  y  $^{\circ}\text{F}$ .

*Especificaciones:*

- Rendimiento (%vol)
- Rendimiento (% peso)
- Gravedad específica
- Destilación ASTM inicial
- Destilación 5, 10,30,50,70,90 % de vol. recuperado
- Viscosidad a 100,122,140,180,210,260
- Azufre
- Punto de escurrimiento
- Punto de Inflamación
- Número de Octano
- Aromáticos
- Parafinas, etc.

Estas propiedades pueden llegar a ser más de setenta, con su respectivos valores para cada intervalo de ebullición del crudo.

La **curva de destilación** nos permite saber gráficamente el volumen recuperado de crudo en cada intervalo de temperatura al que ha sido expuesto, es una curva que tiende a incrementarse a medida que aumenta la temperatura.



**Figura 5.11.** Ejemplo de una curva de destilación TBP (temp vs vol)

Un ensayo de crudo, con sus respectivos cortes de ebullición y sus propiedades se puede observar en la siguiente tabla.

### Ensayo de crudo

#### Aceite crudo Lago Agrio

Propiedad	Gasolina Ligera	Nafta Ligera	Nafta Media	Nafta Pesada	Diesel	Gasóleo	Residuo
Intervalo Ebullición(°F)	C5-172	172-212	212-302	302-374	374-536	536-1000	1000 +
Rendimiento (%Vol)	0.858	4.842	3.30	8.20	13.60	16.80	52.4
Gravedad Específica (°F)	0.65	0.71	0.75	0.78	0.81	0.86	0.97
Gravedad API	83.28	65.08	55.79	47.73	41.27	31.67	14.29
Azufre (% peso)	0.008	0.01	0.01	0.01	0.06	0.4	1.26
RON	71.50	58.40	51.10	<45			
Presión de vapor Reid	16.70	3.67					
Factor K Uop		12.11	11.91	11.78	11.77	1174	
Parafinas (% vol)		68.55	57.51	46.31			
Aromáticos (%vol)		3.99	12.97	14.32	15.9		

**Tabla 5.6.** Especificaciones del crudo Lago Agrio

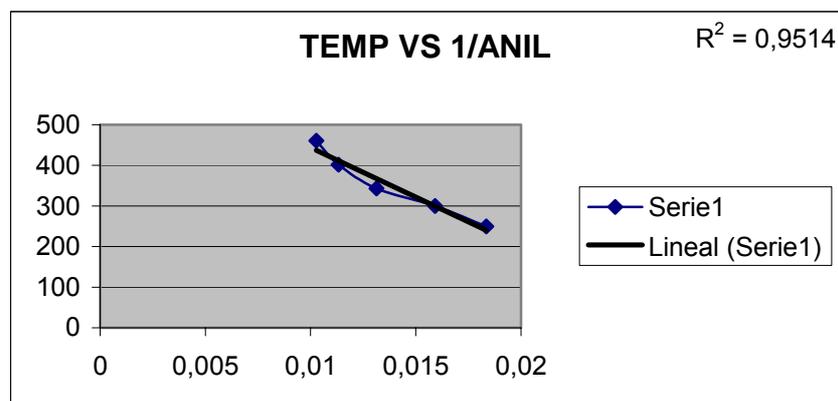
Esta información debe ser ingresada al software RPMS (ver Capítulo 2), pero el problema surge ya que este software acepta datos para cortes ya establecidos y muchas veces estos datos no se ajustan a los cortes de ebullición que vienen detallados en los reportes de cada crudo.

En este módulo del aplicativo, para resolver el anterior problema se hace mención a una técnica de las matemáticas que es el ajuste de curvas.

- El proceso empieza graficando los diferentes cortes de ebullición vs propiedad del crudo.
- Luego se aplica una de las ecuaciones de ajuste de curvas, para así proceder a graficarla, según como indique la ecuación.

Ecuaciones para ajuste de curvas:

$y = b + ax$	cuando	y vs x
$y = be^{ax}$	cuando	log y vs x
$y = be^{ax} + \alpha$	cuando	log(y - $\alpha$ ) vs x
$y = bx^a$	cuando	log y vs log x
$y = bx^a + \alpha$	cuando	log (y - $\alpha$ ) vs log x
$y = a + b / x$	cuando	y vs 1/x
$y = (x / (a+bx))$	cuando	x/y vs x
$y = a + bx + cx^2$	cuando	(y-y <sub>n</sub> )/(x - x <sub>n</sub> ) vs x



**Figura 5.12.** Gráfico Temp vs 1/ Anilina, aplicada el ajuste de curvas

Este tipo de gráficos se los realiza en la hoja de cálculo Microsoft Excel, y con la ayuda del factor  $R^2$  se puede determinar si el ajuste de curva es más aproximado a la original.

- c. Para cada propiedad del crudo se aplica cada una de las ecuaciones de ajuste de curvas, esto se hace con el fin de verificar cuál ecuación se ajusta a los datos de ebullición originales con los del programa RPMS.
- d. Una vez escogida la ecuación, ésta se la pasa a código de lenguaje de programación para así generar el nuevo ensayo de crudo con los nuevos cortes de ebullición y los nuevos valores de las propiedades del crudo.

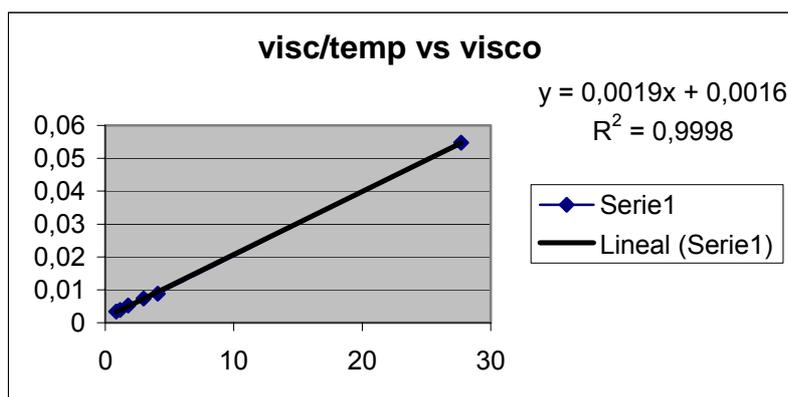
### 5.8.1. Ejemplo de cálculo del nuevo ensayo de crudo

A partir de los gráficos y datos analizados en MS Excel, se procede aplicar la ecuación elegida para pasar a código de Powerbuilder.

Así, una de las propiedades para este ejemplo es la Viscosidad del crudo, donde vemos en Excel que la ecuación que más se ajusta a la curva original (temp vs viscosidad) es:

$$y = ( x / (a+bx))$$

Por tanto se aplica la gráfica: viscos/temp vs visco. (Ver anexo H)



**Figura 5.13.** Gráfico viscos/temp vs visco, aplicada el ajuste de curvas

Donde podemos observar que los valores de a y b son despejados ( $a=0.0019$  y  $b=0.0016$ ) y así se los puede pasar a código de

powerbuilder. Luego se despeja la variable x o y, para luego reemplazar por los valores de temperatura o de propiedad.

Para pasar a código de powerbuilder, todas los intervalos de ebullición y de las propiedades, son almacenados en dos vectores y así se facilita el cálculo de los nuevos valores de ensayo de crudo.

### 5.9. Diseño de los objetos del sistema

El diseño de los controles y objetos es importante al momento de desarrollar el interfaz y su funcionalidad. Se menciona a continuación todos los componentes del sistema, además de sus propiedades, funciones y las formas en que interactúan con el entorno del sistema. Es importante también describir los objetos que conforman el interfaz como son: ventanas, objetos de acceso a datos, menús, cuadros de diálogo entre otros elementos.

Para la elaboración del interfaz del sistema se ha hecho uso de los siguientes controles que ofrece la herramienta de programación PowerBuilder.

- *Espacio de trabajo (Workspace)*.- Es el repositorio de todos los objetos y demás elementos que conforman el interfaz. Por tanto es el más importante elemento de interfaz.
- *Aplicación (Application)*.- Aquí se definen las instrucciones para la conexión con la base de datos, además define la ventana inicial del sistema.
- *Ventana (window)*.- Contiene a la mayoría de elementos del interfaz y en este proyecto es utilizada para procesos de abrir, guardar, ingresar, etc.
- *Texto estático (Static Text)*.- Usado sólo para mostrar información corta (títulos, mensajes de aviso, ayuda, detalle de opciones,etc) .
- *Control de edición de texto (Single Line Edit)*.- Fue usado para el ingreso de información, la misma que puede ser de cualquier tipo de dato (entero, decimal, cadena, etc).

- *Ventana de datos (Datawindow).*- Usado para la presentación de los datos recuperados desde la base de datos. Tiene sus propiedades, funciones y eventos para ser administrado de mejor manera.
- *Menús (Menu).*- Este objeto se encuentra presente con todas sus opciones durante todo el trayecto en que se está utilizando el sistema. Esto pasa por tratarse de ventanas tipo MDI las mismas que tienen un solo menú.
- *Botón de Comando (CommandButton).*- Utilizado para ejecutar acciones específicas como: Aceptar, Guardar, Salir, Generar, Recuperar, etc.

Entre otros elementos que no se describen arriba, están: los objetos de usuario, los cuadros de lista, las listas desplegables, botones con imagen, etc.

Estos son los elementos más importantes que formaron parte del interfaz del sistema,

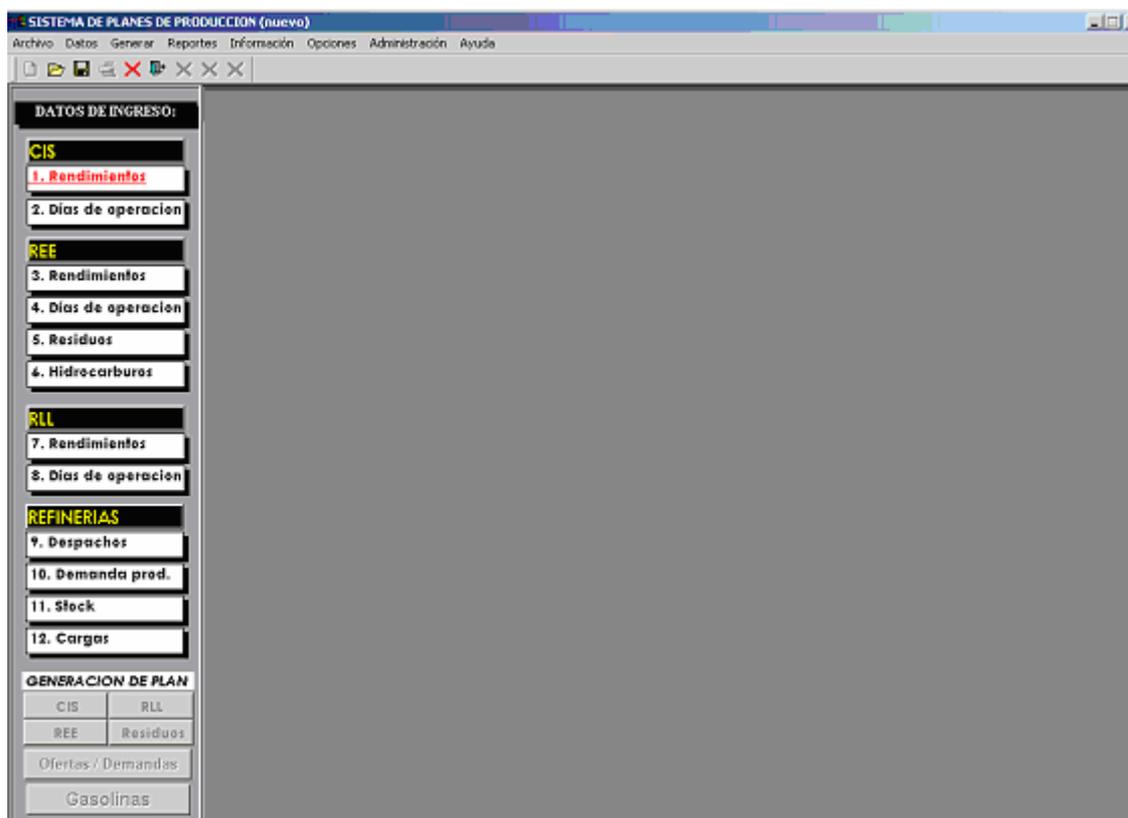
### 5.9.1. Ventana inicial: REGISTRO DE USUARIO



Figura 5.14. Ventana de registro de usuario

**Componentes:**

OBJETO	CANTIDAD	USO
Ventana	1	Contiene a los demás objetos.
Texto Estático	6	Muestra los títulos y mensajes.
Botón de comando	2	Para ejecutar una acción específica
Rectángulo	3	Como fondo de otros objetos.
Cuadro de Texto	2	Para ingresar datos del usuario

**5.9.2. Ventana: PRINCIPAL****Figura 5.15.** Ventana principal

Con esta ventana inicial se pretende indicar la mayoría de los objetos del interfaz del sistema.

**Componentes:**

COMPONENTE	CANTIDAD	USO
Ventana	2	Contienen opciones y demás objetos
Texto estático	18	Muestra títulos y mensajes.
Botón de comando	6	Ejecutan acciones específicas
Rectángulo	1	Encierran a otros objetos
Menú	1	Despliega las opciones del sistema
Barra de herramientas	1	Muestra accesos directos del menú.

A partir de la anterior ventana se generan todas las demás, que a su vez presentan o solicitan información, ya sea para reportes, ingresos, configuraciones y para otros usos.

**5.9.3. Ventana de ingreso: DATOS DE RENDIMIENTOS**

Las ventanas de ingreso tienen el siguiente formato.

Figura 5.16. Ventana de ingreso de datos

**Componentes:**

COMPONENTE	CANTIDAD	USO
Ventana	1	Contiene a los demás objetos
Texto estático	1	Título de la ventana
Ventana de datos	2	Despliega información recuperada de la base de datos
Cuadros de texto	29	Para ingresar los valores

En otras ventanas de ingreso, los cuadros de texto son reemplazados por ventanas de datos (Datawindow) para un mejor tratamiento de esa información.

Puesto que las restantes 12 ventanas de ingreso de datos son similares a la anterior en el aspecto del diseño, no se las detalla.