

18  
2 ej.



UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

FACULTAD DE INGENIERIA

DESARROLLO DE UN SISTEMA  
PARA EL CONTROL DE ARTICULOS  
DE ACTIVO FIJO

**T E S I S**

QUE PARA OBTENER EL TITULO DE:

**INGENIERO EN COMPUTACION**

**P R E S E N T A :**

**CANELA MANZO PATRICIA**

DIRECTOR DE TESIS: ING. SEBASTIAN POBLANO ORDOÑEZ



MEXICO, D. F.

JUNIO 1992

**TESIS CON  
FALLA DE ORIGEN**



Universidad Nacional  
Autónoma de México



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# I N D I C E

INTRODUCCION .....	i
--------------------	---

## CAPITULO I

### ANTECEDENTES

1.1 ANTECEDENTES .....	1
1.2 Definición del Problema y sus Necesidades .....	2
1.3 Estudio de la Situación Actual .....	3
1.4 Critica de la Situación Actual .....	9
1.5 Propuesta de Solución .....	9
1.6 Recursos .....	11
1.6.1 Recursos de Hardware .....	11
1.6.2 Recursos de Software .....	11
1.6.3 Recursos Humanos .....	12
1.7 Viabilidad del Proyecto .....	12

## CAPITULO II

### INGENIERIA DE SOFTWARE

2.1 Crisis del Software .....	13
2.2 Ingenieria de Software .....	14
2.3 Ciclo de Vida de un Sistema .....	16
2.4 Entropia .....	19

### CAPITULO III

#### ANALISIS DE REQUERIMIENTOS

3.1	Requerimientos .....	23
3.2	Herramientas del Análisis Estructurado .....	26
3.2.1	Diagrama de Flujo de Datos .....	47
3.3	Diccionario de Datos .....	50
3.4	Miniespecificaciones .....	54

### CAPITULO IV

#### DISENO DEL SISTEMA

4.1	Diseño Estructurado .....	84
4.2.	Cualidades de un Buen Diseño .....	89
4.2.1	Acoplamiento .....	89
4.2.2	Cohesión .....	90
4.3	Metodología del Diseño Estructurado .....	92
4.4	Diagramas de Estructura del Sistema .....	96
4.5	Características del Lenguaje de Programación ....	103
4.6	Diseño de la Base de Datos .....	105

### CAPITULO V

#### IMPLEMENTACION.

5.1	Codificación .....	117
5.2	Documentación .....	119

## **CAPITULO VI**

### **INTEGRACION Y PRUEBAS.**

<b>6.1 Integración .....</b>	<b>123</b>
<b>6.2 Pruebas .....</b>	<b>127</b>

## **CAPITULO VII**

### **MANTENIMIENTO.**

<b>7.1 Técnicas de Tipo Administrativo .....</b>	<b>132</b>
<b>7.2 Técnicas de Tipo Técnico .....</b>	<b>132</b>
<b>7.3.1 Mantenimiento Correctivo .....</b>	<b>133</b>
<b>7.3.2 Mantenimiento Adaptivo .....</b>	<b>133</b>
<b>7.3.3 Mantenimiento Adaptivo .....</b>	<b>133</b>
<b>Conclusiones .....</b>	<b>135</b>
<b>Bibliografía .....</b>	<b>137</b>

## I N T R O D U C C I O N

Para que la Universidad Nacional Autónoma de México pueda cumplir la importante tarea de impartir educación media y superior, y de desarrollar investigación y extensión universitaria en nuestro país, ha de conocer los recursos con que cuenta, para así poder administrarlos racionalmente y lograr su mejor rendimiento.

Por esto es imperativo registrar, controlar, preservar e incrementar el Activo Fijo de que dispone cada una de las diferentes dependencias universitarias.

Por este motivo la Dirección General de Servicios de Computo para la Administración (DGSCAd), se dio a la tarea de crear un sistema que permita a las autoridades de las dependencias de la U.N.A.M., contar con información precisa y oportuna, relativa al Control del Activo Fijo.

El sistema al ser de tipo general y contar con las bases fundamentales del manejo del Control del Activo Fijo que se maneja en la U.N.A.M., puede ser instalado en cualquier dependencia de la U.N.A.M.

La presentación de esta tesis esta dividida en diferentes fases las cuales permiten llevar a cabo un control efectivo del sistema durante toda su etapa de diseño e implementación, cada una de estas se desarrolla en los diferentes capitulos que a continuación se detallan.

En el capítulo primero se da un panorama general sobre los antecedentes o conocimientos previos que deben conocerse para analizar posteriormente y de una manera estructurada la problemática en estudio.

Dentro del capítulo segundo hablaremos en forma general sobre lo que es la Ingeniería de Software y la razón de porque usar técnicas especializadas para la realización de sistemas, además se realiza una explicación de la técnica utilizada para el desarrollo de nuestro problema.

Dentro del tercer capítulo se presenta la fase del Análisis Estructurado, es la encargada de estudiar dentro de la problemática, la organización existente y el medio ambiente que lo rodea, se plantean los objetivos, se proponen y evalúan una o diversas alternativas de solución indentificando sus entradas y salidas, para finalmente elaborar Diagramas Lógicos de Flujo de Datos, con su respectivo Diccionario de Datos y un Pseudocódigo de la solución más viable, antes de codificarse en un lenguaje de alto nivel.

Dentro del cuarto capítulo se realiza la fase correspondiente al Diseño del Sistema, empezando por Definir todas las herramientas necesarias a considerar para obtener un buen diseño.

Después de haber definido las bases teóricas, se aplican éstas al problema real. Como parte complementaria al diseño se especifica la selección del lenguaje de programación a utilizar, y como en este caso se optó por el uso del administrador de bases de datos DBASEIII, se especifican los conceptos más importantes de Bases de datos para proceder al diseño de ésta.

Dentro del capítulo cinco es el correspondiente a la fase de Implementación del sistema donde se especifica la codificación realizada para el sistema de Control del Activo Fijo (RESGUARDOS).

Dentro del capítulo seis se presenta la fase del sistema correspondiente a la Integración y las Pruebas, definiéndose las técnicas elegidas al problema real, así también se especifican los tipos de prueba que existen y técnicas para la preparación de éstas.

Dentro del capítulo siete donde se encuentran la actividad de mantenimiento, se indica la forma en que debe de ser operado el sistema mediante el manual de usuario. Mientras que la actividad de mantenimiento consiste en poder satisfacer nuevos requerimientos de otros usuarios, mediante posibles cambios en la programación ya

existente.

Para finalizar se presentan las conclusiones así como la bibliografía que fue requerida a lo largo de la elaboración de esta tesis.

## CAPITULO I

### **1.1 ANTECEDENTES.**

La Universidad Nacional Autónoma de México cuenta actualmente con aproximadamente 150 dependencias incluyendo facultades, institutos, escuelas y dependencias administrativas entre otras, por lo tanto surge la necesidad de manejar su información en forma rápida y concisa. Es por ello que opta por la automatización de todas sus actividades con el fin de cumplir satisfactoriamente sus objetivos, dejando atrás el proceso manual.

La Dirección General de Servicios de Cómputo para la Administración ( DGSCAd ), entre otras funciones es la encargada de proveer a las diferentes dependencias de todo el apoyo de los procesos automatizado a nivel administrativo. Es por ello que dentro de esta dependencia se desarrollan sistemas con el fin de hacer más accesible y rápido los diferentes procesos administrativos dentro de las dependencias de nuestra Universidad.

La DGSCAd. a través del Departamento de Asesoría e Implantación de Sistema decidió la realización del proyecto de

tesis DESARROLLO DE UN SISTEMA, PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO, EN DEPENDENCIAS UNIVERSITARIAS (RESGUARDOS), que permita manejar y controlar su información referente al control del activo fijo (bienes inventariables) de una manera ágil, rápida y confiable tomando en cuenta las normas establecidas por la U.N.A.M. para su control.

## **1.2 DEFINICION DEL PROBLEMA Y SUS NECESIDADES.**

Para que cada dependencia pueda cumplir cabalmente la importante tarea para la que fue creada, debe de conocer los artículos con que cuenta, para así poder administrarlo racionalmente y lograr su mejor rendimiento.

El problema en estudio está referido al proceso del control del Activo Fijo en las diferentes dependencias de la U.N.A.M., por lo que nos referiremos al medio ambiente de dicho problema.

El medio ambiente en el cual se desenvuelve el problema, es un ambiente administrativo, donde las personas que se hacen cargo del proceso de Control del Activo Fijo, no están familiarizados con

el uso de la computadora. Además el personal que se dedica al Control del Activo Fijo, por lo general realizan otras funciones por lo que a veces la premura de tiempo ocasiona errores durante el proceso.

Por otra parte, es común en dependencias de la U.N.A.M., que la gente cambie muy frecuente de actividades, por lo que se necesita una capacitación continua para la realización del proceso de Control del Activo Fijo tal y como se realiza en nuestra Universidad. Por lo anterior se considera que sería muy conveniente que este proceso se realiza de manera rápida, pero que además no sea muy complicado de aprender.

### **1.3 ESTUDIO DE LA SITUACION ACTUAL.**

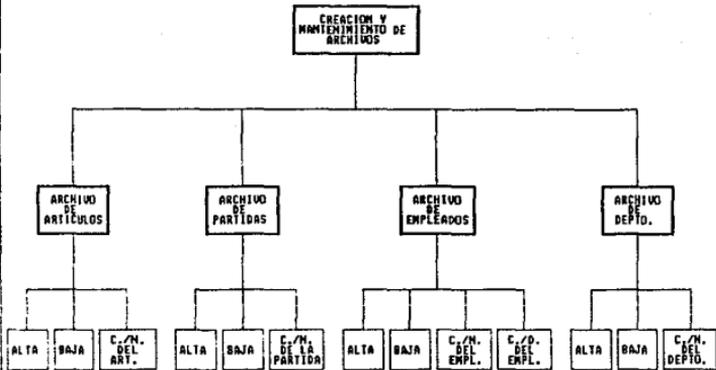
Actualmente el proceso de Control del Activo Fijo se realiza con ayuda de un sistema desarrollado en lenguaje de programación "COBOL", cuya finalidad es facilitar el control de los bienes inventariables que se tienen en las diferentes dependencias universitarias.

El sistema está integrado por cuatro módulos, cada uno de los cuales tiene su menú principal.

- 1.- MÓDULO DE CREACION Y MANTENIMIENTO DE ARCHIVO
- 2.- MÓDULO DE REGISTRO DE TRANSACCIONES
- 3.- MÓDULO DE REPORTES
- 4.- MÓDULO DE CONSULTA POR TERMINAL

A continuación se presentan los diagramas de cada uno de los módulos que integran el sistema.

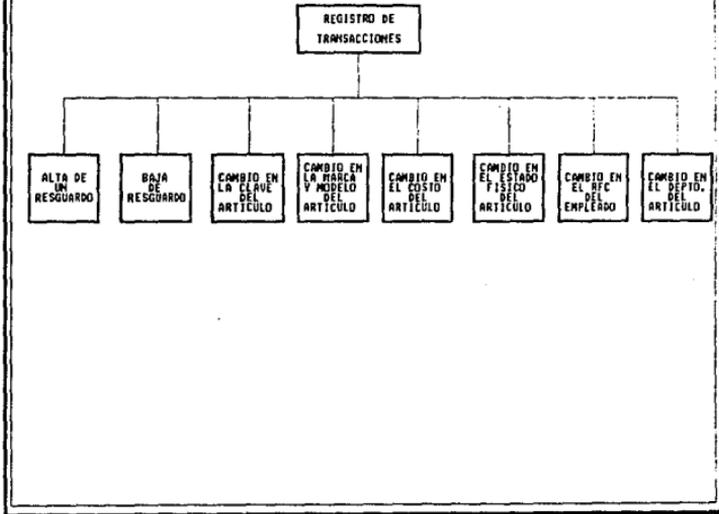
## MODULO DE CREACION Y MANTENIMIENTO DE ARCHIVOS



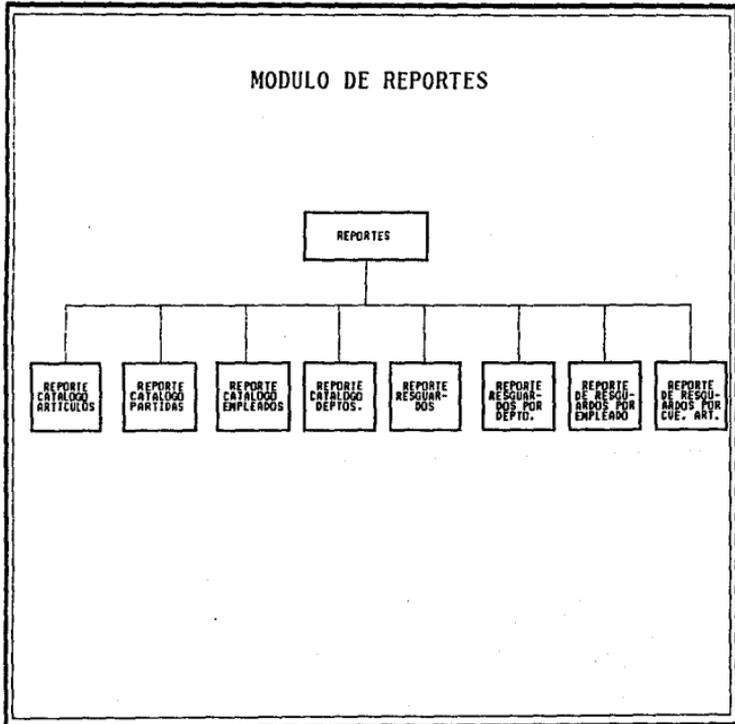
C./N. = CAMBIO DEL NOMBRE

C./D. = CAMBIO EN EL NOMBRE DEL DEPARTAMENTO

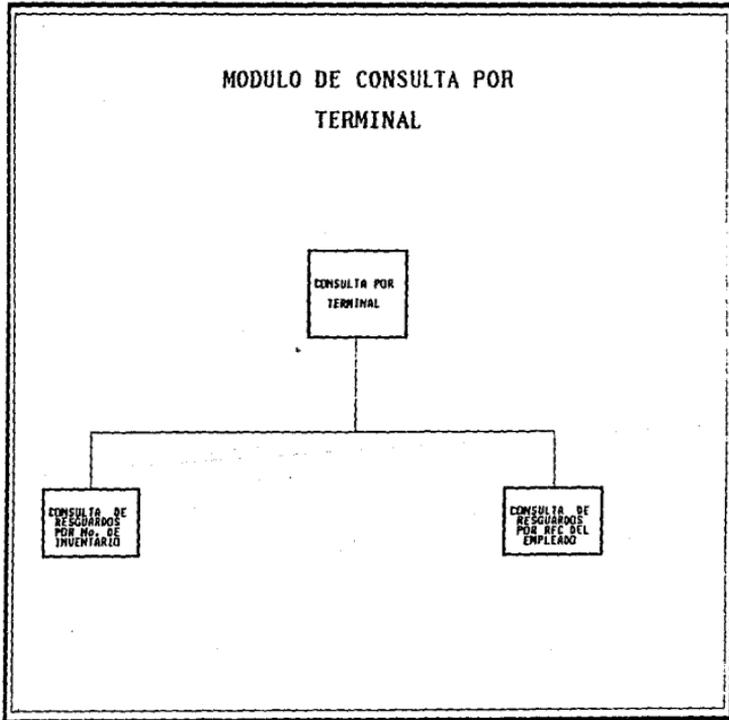
## MODULO DE REGISTRO DE TRANSACCIONES



## MODULO DE REPORTES



# MODULO DE CONSULTA POR TERMINAL



#### **1.4 CRITICA DE LA SITUACION ACTUAL.**

Como se menciona anteriormente el proceso se realiza actualmente con ayuda de un sistema el cual se desarrollo en una de las primeras versiones del lenguaje de programación "COBOL", y en microcomputadora "CROMENCO"; después se transporto a microcomputadora PC compatibles con IBM, la cual presenta el problema de dañar la información en un determinado número de registro(1500 aprox.) de los archivos relativos. Su mantenimiento actualmente es complicado.

Es conveniente enfatizar que algunas de las funciones del proceso del Control del Activo Fijo se realizan en forma manual.

Su transportación a otra versión del Lenguaje de Programación COBOL, equivaldría casi a la realización de una nueva versión del sistema, ya que de una versión a otra son muy notables las diferencias.

#### **1.5 PROPUESTA DE SOLUCION**

De acuerdo a la situación actual se plantea la necesidad de elaborar un nuevo sistema que proporcione mayores ventajas al ya

existente y que vaya de acorde a los recursos de software, hardware y humanos con que se cuentan actualmente. A demás de que automatice las funciones que se realizan en forma manual.

El hecho de un nuevo desarrollo trae ciertas desventajas como sería la inversión de tiempo, pero a este le aunamos los diferentes beneficios con que contará el sistema como serían:

- Capturar, validar y almacenar la información referente al Control del Activo Fijo.
- Mejorar el tiempo de respuesta.
- Implementar innovaciones al sistema:
- Autorespaldo.
- Automatizar la recuperación.
- Empaquetamiento de los archivos de datos, para una mayor protección de la información.
- El usuario podrá consultar información via pantalla de video y a través de reportes.
- Alcanzar un alto grado de funcionalidad.

## 1.6 RECURSOS

Mencionando los recursos con que se cuenta para la realización del sistema no existen ningún impedimento puesto que la Dirección General de Servicios de Cómputo para la Administración cuenta con los siguientes recursos:

### 1.6.1 RECURSOS DE HARDWARE

En cuanto a éstos recursos se cuenta con microcomputadoras PC AT compatibles con IBM con disco duro, impresoras, papel para impresoras, discos flexibles de 5 $\frac{1}{4}$  pulg. con sus respectivos Kbytes de memoria.

### 1.6.2 RECURSOS DE SOFTWARE

Existen diferentes paquetes de desarrollo como lo son manejadores de bases de datos, compiladores, lenguajes de tercera y cuarta generación, utilerías, procesadores de datos, etc.

Es conveniente enfatizar que no existe ninguna restricción legal para usar el software para la solución del problema.

### 1.6.3 RECURSOS HUMANOS

Con respecto a los recursos humanos la institución cuenta con personal calificado para cubrir en forma absoluta las tareas que el sistema requiera para su funcionalidad.

### 1.7 VIABILIDAD DEL PROYECTO

Del problema en estudio y de los beneficios planteados, la decisión a tomar es que lo más conveniente para la solución del problema, es la elaboración de un nuevo sistema para el Control de Artículos de Activo Fijo en microcomputadora PC compatible con IBM para ser usado por las diferentes dependencias de la U.N.A.M.

## CAPITULO II

### INGENIERIA DE SOFTWARE

La finalidad de este capítulo es mostrar un panorama general de la Ingeniería de Software.

Primero trataremos las causas que dieron origen a ella, así como sus consecuencias.

#### 2.1 CRISIS DEL SOFTWARE

El término Ingeniería de Software se introdujo por primera vez a finales de la década de 1960 en una conferencia celebrada para analizar la llamada " Crisis del Software ". Esta crisis fue el resultado directo de que mientras el costo del hardware disminuía el del software aumentaba.

Las primeras experiencias en la construcción de grandes sistemas de software, eran producto de deficientes realizaciones, las cuales son difíciles de mantener.

Los problemas detectados en la Crisis del Software son:

- Los costos se incrementan en forma exponencial.
- Los proyectos no se terminan a tiempo, ni con el

presupuesto programado.

- El mantenimiento del sistema absorbe la mayor parte de recursos de la gente de desarrollo.

Estos a su vez generan otros problemas:

- Insatisfacción del usuario con el sistema.
- Dudosa calidad del software.
- Dificultad en el mantenimiento.

Habia una urgente necesidad de crear nuevas técnicas y metodologías que permitieran controlar la complejidad inherente a los grandes sistemas de software.

Debido a estos problemas surge la necesidad de crear herramientas para solucionarlos, es entonces cuando surge la Ingeniería de Software, que es un intento para formalizar las etapas en el desarrollo de programas y generar software de alta calidad y alto rendimiento.

## 2.2 INGENIERIA DE SOFTWARE

Existen una gran diversidad de definiciones de la Ingeniería de Software y una de las más acertadas es la siguiente:

La Ingeniería de Software es el establecimiento y uso de

métodos de ingeniería con el fin de obtener software rentable y funcional, y a su vez generar sistemas de cómputo eficaces, confiables, transportables y rentables. Esta definición surge por Fritz L. Bauer en la Universidad de Munich Alemania en 1972.

El tener un software rentable y funcional implica mucho más que un software de alta calidad y sus características son las siguientes:

### SOFTWARE

- Confiable** : La confiabilidad esta definida como la capacidad para desempeñar una función requerida bajo ciertas condiciones durante un tiempo específico.
  
- Eficiente** : Grado con el que un producto de programación efectúa sus funciones, mediante un mínimo de recursos computacionales.
  
- Eficaz** : Que satisfaga las necesidades para lo que fue realizado.
  
- Solidez** : Grado con el que un producto de

programación puede continuar operando correctamente, a pesar de introducir datos inválidos.

**Transportable:** Realizar productos con un ambiente de cómputo común.

Con el fin de obtener las características antes mencionadas, las cuales se han ido desarrollando en las numerosas metodologías para el desarrollo de software, las cuales se basan en los principios de análisis y diseño estructurado. De todas estas metodologías se ha elegido para el desarrollo del sistema de Control de Activo Fijo ( RESGUARDOS ) en microcomputadora PC, es el que esta basado en el " Ciclo de Vida del Sistema " y presenta las diferentes fases las cuales forman parte durante la vida de un sistema, desde su nacimiento hasta después de su liberación ya que se encuentra incluido su mantenimiento.

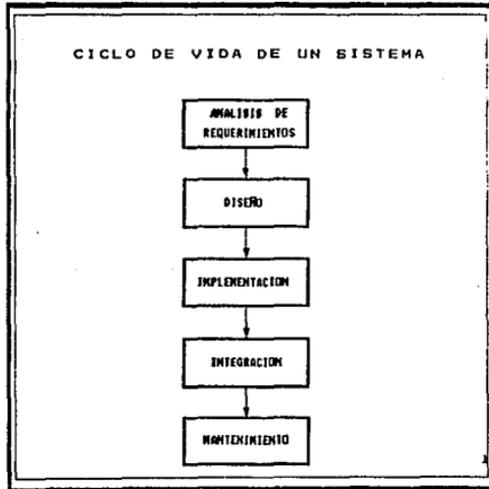
### **2.3 CICLO DE VIDA DE UN SISTEMA**

El ciclo de vida del software surge alrededor de 1975-76 y fue el resultado de la necesidad de caracterizar de algún modo el sistema.

Por ciclo de vida de un sistema se entiende el conjunto de fases por las que pasa a lo largo del tiempo, desde la fase de estudio y concepción hasta la de realización, explotación y mantenimiento.

En la actualidad existen diversas representaciones de lo que es el "Ciclo de Vida", ya que cada empresa tiene su propio prototipo específico, modificando tan solo alguna característica para su proyecto.

Las fases del "Ciclo de Vida de un Sistema" son representados en el siguiente diagrama.



Es conveniente enfatizar que todas las etapas son importantes y que dependiendo de su buen desarrollo se podrá obtener un producto de alta calidad.

**Análisis de Requerimientos:** En esta etapa los requerimientos de todos los elementos del sistema se centran e intensifican especialmente en el software. La evaluación del problema y la síntesis de la solución, evaluar el flujo y estructura de la información, refinar en detalle todas las funciones del programa en detalle, establecer las características de la interface del sistema y descubrir las ligaduras del diseño.

**Diseño:** En esta etapa el proceso de Diseño traduce los requerimientos en una representación del software que puede ser establecida de forma que obtenga la calidad requerida antes de que comience la codificación. Esto se logrará a través de técnicas de Diseño Estructurado.

**Implementación:** En esta etapa el Diseño debe traducirse en forma legible para la máquina. El paso de la codificación ejecuta esta tarea. En esta etapa también se encuentra la documentación del sistema.

**Integración:** En la parte de integración y pruebas se deberán de integrar los módulos implementados para que queden todos integrados en un solo sistema. La prueba se enfoca sobre la lógica interna del software, asegurando que todas las sentencias se han probado, y sobre las funciones externas, esto es, realizando pruebas que aseguren que la entrada definida producirá los resultados que realmente se requieren.

**Mantenimiento:** Esta fase suele ser (aunque no necesariamente) la más larga del ciclo de vida. Se instala el sistema y se pone en uso práctico. La actividad de mantenimiento implica mejorar la aplicación de las unidades del sistema y aumentar los beneficios de éste a medida que se perciben nuevas necesidades.

## **2.4 ENTROPIA EN EL CICLO DE VIDA DE UN SISTEMA**

La Entropía es un fenómeno que juega un papel central en el desarrollo de un proceso.

El concepto de entropía es una de las contribuciones más elegantes a la ingeniería y se presenta en todos los procesos naturales y también es un fenómeno que sufre un sistema durante su ciclo de vida. Este fenómeno se define como la pérdida de

estructuración que sufre un sistema durante su ciclo de vida, es por eso que se debe procurar que el porcentaje de entropía existente sea mínimo. En general un sistema se mantiene hasta un límite de un 60 % de entropía por lo que se debe poner mucha atención a la etapa de mantenimiento ya que los sistemas siguen las siguientes leyes de evolución dinámica de los sistemas:

- *Ley del cambio continuo:* Los sistemas que sufren muchos cambios son inmantenibles y tienen que rehacerse.
- *Ley del incremento de entropía:* La entropía aumenta con el tiempo.
- *Ley del crecimiento estáticamente plano:* En un periodo de tiempo disminuye la cantidad de errores.

Con lo descrito anteriormente nuestro objetivo consistirá en minimizar el total de entropía integral durante el ciclo de vida, lo cual requerirá que cada fase este bien planeada y realizada.

Causas que hacen que la entropía exista:

- Una comunicación inadecuada entre el usuario y las personas que desarrollan el sistema.

- Una metodología inadecuada para el desarrollo de cada fase.
- Documentación incompleta.
- Mala planeación de tareas.

De esta manera se ha establecido un panorama muy general de lo que es la Ingeniería de Software, lo que pretende esta rama de la ingeniería, la metodología en el DESARROLLO DE UN SISTEMA, PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO, EN DEPENDENCIAS UNIVERSITARIAS con el fin de obtener un sistema de alta calidad y alto rendimiento.

### CAPITULO III

#### ANÁLISIS DE REQUERIMIENTOS

La fase que a continuación se desarrolla para la solución de nuestro problema es la del análisis de requerimientos, en la cual se hará un estudio exhaustivo del problema para así poder encontrar la solución óptima, usando lo que se conoce como análisis estructurado.

El análisis estructurado se puede definir como una metodología para especificar la funcionalidad del sistema.

El problema a resolver ya está claramente establecido en la fase anterior, el cual cumple con las siguientes características:

- Debe ser conciso
- Debe ser mantenible
- Debe ser particionable
- Debe ser entendible
- Debe ser lógico

Para la realización del Análisis Estructurado en este documento hará uso de las herramientas:

- Diagramas de Flujo de Datos.
- Diccionario de Datos.

A continuación aplicaremos la metodología correspondiente a

esta fase.

### 3.1 REQUERIMIENTOS

1. Se necesita poder crear y mantener un archivo para el:
  - Catálogo de Partidas: es donde se reflejan todos los movimientos de las partidas a utilizar.
  - Catálogo de Artículos: es donde se realiza el mantenimiento de los artículos.
  - Catálogo de Departamentos: es donde se realizan todos los movimientos de los departamentos a utilizar.
  - Catálogo de Empleados: es donde se almacena toda la información de los empleados a utilizar.
  - Resguardo: es donde se reflejan todos los movimientos al Control del Activo Fijo.
  - Baresg: es donde se almacena la información referente a la baja de los resguardos.

2. Es necesario obtener las entradas necesarias al sistema y además poder validar estas entradas para evitar errores en el proceso, estas entradas que son los catálogos de partida, artículos, departamentos, empleados y a los resguardos.

3. Toda la información deberá quedar almacenada, aun sin estar correcta, con el fin de evitar la doble captura de la información.

4. Si algún dato cumple con los requerimientos de entrada pero no se encuentra correcto, debe de existir la manera de corregirlo.

5. Se requiere que el sistema permita al usuario dar de baja cualquier movimiento ya sea de los catálogos o de alguna alta de resguardo.

6. Se debe permitir al usuario consultar la información vía terminal.

7. Contando con la información correcta se deberá emitir los siguientes reportes:

- a) Reporte General de Resguardos.
- b) Reportes por R.F.C. del empleado(s) responsable.
- c) Reportes por Departamento del empleado(S) responsable.
- d) Reportes por Clave del Artículo(s).
- e) Reportes de los respectivos catálogos.
- f) Reporte Asignado de Resguardos por Deptos. y por Dependencia.

g) Reporte Asignado de Resguardos por empleados y por Dependencia.

h) Reporte de Conciliación de inventarios con la Dirección General de Patrimonio.

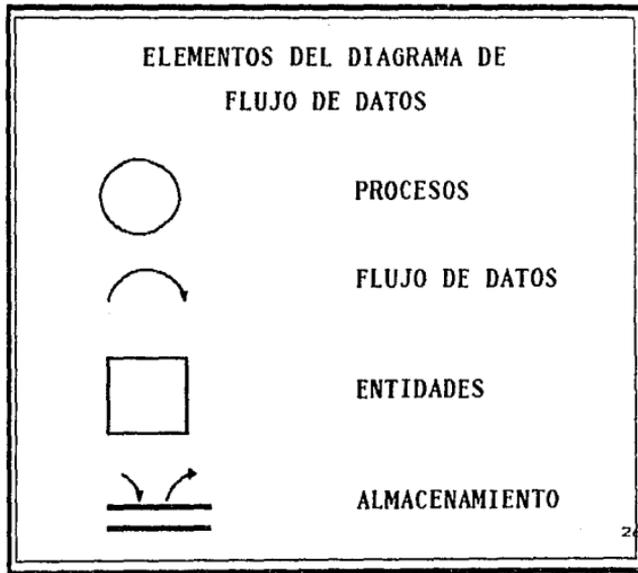
B. Se necesita que se pueda cancelar cualquier alta de resguardos.

### 3.2 HERRAMIENTAS DEL ANÁLISIS ESTRUCTURADO.

#### 3.2.1 DIAGRAMA DE FLUJO DE DATOS

El Diagrama de Flujo de Datos (DFD) , es una forma de representación gráfica del flujo de datos.

En los DFD, se usan los siguientes elementos:



### **PROCESOS**

Se representa por medio de círculos poniendo en el interior el nombre de proceso.

Representan las operaciones, manuales o mecanizadas, que se realizan con los datos.

Deben tener , al menos, un flujo de entrada de datos y uno de salida.

Para indicar la secuencia de los procesos, en el interior del círculo, además del nombre del proceso, se podrá un número de orden de ejecución del mismo.

En una primera fase deben ser muy general con la idea de dar una visión global del sistema y de los objetivos a cumplir. Posteriormente, cada círculo o proceso dará lugar en un DFD de mayor nivel de detalle a una representación gráfica del mismo. Al DFD más general también se le llama Diagrama de Contexto. Dará una imagen de las relaciones del sistema con otros subsistemas y determinará los límites del mismo.

### **FLUJO DE DATOS**

Representan el movimiento de información u objeto entre las personas o departamentos contemplados.

Gráficamente se dibujan como líneas que unen al emisor con el receptor de la información u objeto, indicando el sentido del movimiento por medio de una punta de flecha.

### **ENTIDADES**

Son personas o servicios que reciben o emiten algún flujo de información.

Se representan como rectángulos en cuyo interior figura el nombre de la entidad.

### **ALMACENAMIENTO**

Su representación gráfica se corresponde con dos líneas paralelas en cuyo interior se pone el nombre del archivo.

Corresponden a los conjuntos básicos de información de la empresa y podrán ser manuales o informatizados y serán origen o destino de un flujo de datos. A su vez proporcionarán entradas de información a los procesos o serán salida de los mismos.

### **PASOS PARA REALIZAR UN DFD**

- 1.- Identificación de fuentes externas.
- 2.- Identificación de los diferentes entradas y salidas que se pueden esperar del proceso agrupándolas lógicamente y separando las entradas y salidas que manejan errores y excepciones.
- 3.- Realizar un diagrama tentativo el cual comience con las fuentes primarias de entrada y que contengan todas las transformaciones lógicas de los datos excepto aquellos que manejan errores y excepciones.

4.- Checar este diagrama con el proceso real asegurándose que se incluyeron todas las entradas y salidas solicitadas así como las transformaciones lógicas de los datos.

5.- Dibujar un segundo diagrama en el que se incluya los errores y se minimice el número de cruces de flujo de datos.

6.- Realizar un recorrido con una persona involucrada en el proceso para su revisión.

7.- Mandarlo a dibujo.

Ahora bien como ya se establecieron los pasos a seguir para la realización de DFD lo aplicaremos a nuestro problema.

#### FUENTES EXTERNAS.

Solo existe una fuente externa ya que el sistema es de tipo conversacional y será el usuario.

#### ENTRADAS.

La primera entrada al sistema de acuerdo a lo descrito dentro de los requerimientos, son el catálogo de partidas, artículos, departamentos y empleados, así como la alta de un resguardo.

**PARTIDAS**

Las partidas, para mayor simplicidad serán indentificadas por un número de 3 dígitos. La descripción de estas partidas son las que se manejan en la Dirección General de Proveduría de la U.N.A.M.

**PARTIDA:** Es el número que se le asigna a la Partida Presupuestal.

**NOMBRE DE LA PARTIDA:** Descripción de la Partida.

Para obtener los datos antes descritos se sugiere la siguiente forma de captura:

CREACION DEL CATALOGO DE PARTIDAS	
<b>PARTIDA:</b>	<input type="text"/> <input type="text"/> <input type="text"/>
<b>NOMBRE PARTIDA:</b>	<input type="text"/>

**ARTICULOS.**

A los artículos se hará referencia mediante una clave de 8 caracteres de los cuales los 5 primeros forman el folio del artículo y los dos últimos representan el tipo, existiendo un carácter en blanco entre el folio y el tipo. Después se hace referencia al grupo que pertenece así como a la partida presupuestal del artículo.

**ARTICULO:** Es la clave que se le asigna al artículo, y es con la cual se hace referencia al artículo, estas claves proporcionadas por la Dirección General de Patrimonio.

**GRUPO:** Es la clave a la cual pertenece el artículo.

**PARTIDA:** Es un número de 3 posiciones mediante el cual se hace referencia a las diferentes partidas presupuestales, dependiendo a cual de ellas pertenezca el artículo.

**DESCRIPCION:** Nombre del artículo.

Para obtener la información descrita se sugiere la siguiente captura.

CREACION DEL CATALOGO DE ARTICULOS												
ARTICULO:								GRUPO:		PARTIDA:		
DESCRIPCION:												

**DEPARTAMENTOS.**

Esta tercera entrada al sistema es la que nos permite realizar los movimientos al catálogo de departamentos.

**DEPARTAMENTO:** Consta de un clave de 3 digito mediante la cual se hace referencia al departamento, estas claves son propias de cada dependencia.

**NOMBRE DEPARTAMENTO :** Aquí se hace referencia a la descripción del departamento.

La sugerencia de captura es la siguiente:

CREACION DEL CATALOGO DE DEPARTAMENTOS	
DEPARTAMENTO:	<input type="text"/> <input type="text"/> <input type="text"/>
NOMBRE DEPARTAMENTO:	

## EMPLEADOS.

Esta entrada al sistema nos pertine realizar todos los movimientos al catálogo de empleados, se sugiere que en la captura se consideren los siguientes datos.

R.F.C.: Es el registro Federal de Contribuyentes del empleado

DEPARTAMENTO: Es la clave del departamento al que pertenece el empleado.

NOMBRE DEL EMPLEADO: Aquí se hace referencia al nombre del empleado, con el siguiente formato apellidos: paterno, materno y nombre.

Y su captura se sugiere de la siguiente forma:

CREACION DEL CATALOGO DE EMPLEADOS										
R.F.C.:										
NOMBRE:										
DEPTO:										

### **ALTA DE UN RESGUARDO.**

Está entrada al sistema es la que nos permite realizar la asignación de un bien al empleado, es decir aquí es cuando se responsabiliza al personal de los diferentes artículos que tendrá su cargo, mediante el número de inventario el cual es único para cada artículo o bien. La información que debe contener es la siguiente:

**NUMERO DE INVENTARIO:** Número asignado al bien, es único. compuesto por 7 dígitos.

**NUMERO DE SERIE:** Número de serie asignado al bien, es único y no todos los bienes lo tienen.

**ARTICULO:** Indica la clave del artículo al cual fue asignado el número de inventario.

**GRUPO:** Es el que corresponde de acuerdo al catálogo de artículos.

**PARTIDA:** Es el número de partida presupuestal a la que corresponde el artículo.

**NOMBRE ARTICULO:** Es el nombre del artículo que corresponde de acuerdo al catálogo de artículos.

**TIPO DE ADQUISICION:** Es el tipo de la adquisición del artículo.

**FECHA DE ADQUISICION:** Es la fecha en que fue adquirido el artículo.

**R.F.C.:** Registro Federal de Contribuyente del empleado que tendrá bajo su responsabilidad el bien o artículo.

**NOMBRE DEL EMPLEADO:** Es el nombre del empleado al cual corresponda el R.F.C. de acuerdo al catálogo de empleados.

**DEPARTAMENTO:** Número del departamento al cual pertenece el empleado, de acuerdo al catálogo de empleados.

**CANTIDAD:** En caso de que los bienes se hayan considerado globalmente anotar la

cantidad correspondiente.

**MARCA DEL ARTICULO:** Es la marca comercial del articulo.

**MODELO DEL ARTICULO:** Es el que se indica en la placa del fabricante, si existe.

**COSTO UNITARIO:** El precio de adquisición del bien en moneda nacional.

**ESTADO FISICO:** Estado fisico actual en que se encuentra el articulo.

**LUGAR DEL INVENTARIO:** Es el lugar en donde se encuentra asignado el inventario en caso de que no sea el mismo al del empleado.

Para obtener los datos antes descritos se sugiere la siguiente forma de captura:

NOBRE DE LA DEPENDENCIA  
SISTEMA DE RESGUARDOS

## ALTA DE RESGUARDOS

No. INVENTARIO: 9999999 SERIE: 99999999999 CVE. ART: 99999 AA GRUPO: 1 PART: 414  
DESCRIPCION ART.: DESARPAADOR PUNTA CUADRADA FECHA DE ADD: 31/12/99  
TIPO ADD.: 1 COMPRA DIRECTA  
RFC.: CAMP-AS0224 NOMBRE: CANELA RANZO PATRICIA DEPTO: 308  
CANTIDAD: 1 MARCA: HELVE MODELO: X24 E.F.: B  
PRECIO: 9.999 UBICACION: C FOTOCOPIADORA

**SALIDAS DEL SISTEMA.**

Como salidas se tiene los siguientes reportes:

**- REPORTE NUMERICO DE ARTICULOS.**

Este reporte emite la información referente al catálogo de artículos que existen hasta este momento.

12/12/79:		UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO		13:40
NOMBRE DE LA DEPENDENCIA				
CATALOGO DE BIENES				
*****	****	*****	*****	
POLIO	TIPO	GRUPO	DESCRIPCION	
*****	****	*****	*****	
07455	F	07	PONTA BAFILIN	
47077		03	MAGUINA CORTADORA DE PAPEL	
47142	AA	03	MAGUINA PUNZADORA	

**- REPORTE NUMERICO DE PARTIDAS.**

El cual es un reporte donde aparecen las partidas presupuestal dadas de alta con su respectiva descripción.

12/12/79:		UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO		13:40
NOMBRE DE LA DEPENDENCIA				
CATALOGO DE PARTIDAS				
*****	*****	*****		
PARTIDAS	DESCRIPCION			
*****	*****			
411	UTILES, MATERIALES Y ACCESORIOS			
514	EQUIPO DE COMPUTO			

**- REPORTE NUMERICO DE DEPARTAMENTOS.**

En este reporte aparece el número del departamento así como su nombre.

12/12/91 UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO 1345	
NOMBRE DE LA DEPENDENCIA	
CATALOGO DE DEPARTAMENTOS	
***** DEPTO. *****	***** NOMBRE DEL DEPARTAMENTO *****
100	DIRECCION GENERAL
305	ASESORIA

**- REPORTE ALFABETICO DE EMPLEADOS.**

Este reporte emite la información referente a los datos del empleado, rfc, nombre, departamento al que pertenece.

12/12/91 UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO 1345			
NOMBRE DE LA DEPENDENCIA			
CATALOGO DE EMPLEADOS			
**** R.F.C. *****	***** NOMBRE DEL EMPLEADO *****	***** DEPTO. *****	***** NOMBRE DEPTO. *****
CAHF-450224	CANELA MANZED PATRICIA	305	ASESORIA
CANN-5V1223	CASTILLO NISHIMURA RAFAEL	100	DIRECCION GENERAL

FECHA: 10/06/92

PAGINA 1

## UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

## DIRECCION GENERAL DE SERVICIO DE COMPUTO PARA LA ADMINISTRACION

## REPORTE NUMERICO DE RESGUARDOS

NUMERO INVENTARIO	FOLIO	TIPO GRUPO	DESCRIPCION	MARCA	MODELO	R.F.C.	NOMBRE RESPONSABLE	DEPTO.
1	79657	06	P I N Z A S	URREA-PROI	DE MECANIC	MASS-560805	MARTINEZ SIBAJA SIXTO	308
2	40459	02	06 DESARMADOR PLANO MEDIANO	PROTO 6*	# 9606	FD MASS-560805	MARTINEZ SIBAJA SIXTO	308
3	47175	06	ESCUADRA UNIVERSA; METALICA	STANLEY	500D--23"	AERS-520322	ACEVES ROSAS SERGIO	301
4	47092	06	ESCUADRA DE PRECISION CON NIVEL	METROMEI		AERS-520322	ACEVES ROSAS SERGIO	301
5	78993	06	P E R J E C T O	PROTO-PROF	MUM.12	AERS-520322	ACEVES ROSAS SERGIO	301
6	79657	06	P I N Z A S	VISE-GRIP	DE PRECISION	AERS-520322	ACEVES ROSAS SERGIO	301
7	69961	06	MARTILLO DE BOLA	TRUFER	MB-24	AERS-520322	ACEVES ROSAS SERGIO	301
8	40059	06	E U C H I L L A	STANLEY	99-A	AERS-520322	ACEVES ROSAS SERGIO	301
9	79657	06	P I N Z A S	KLEIN MUN	DE ELECTRI	AERS-520322	ACEVES ROSAS SERGIO	301
-1	89279	06	SECADOR ELECTRICO DE AIRE PARA MANOS.			MENC-591030	MEJIA MONTIEL MA.ELENA	308
-2	51051	06	EXTINGUIDOR DE POLVO QUIMICO SECO			SER. EU 83 MENC-591030	MEJIA MONTIEL MA.ELENA	308
-3	89279	06	SECADOR ELECTRICO DE AIRE PARA MANOS.	BAZO	ARRIBA NUJ	MENC-591030	MEJIA MONTIEL MA.ELENA	308

FECHA: 10/06/92

## UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PÁGINA 3

DIRECCION GENERAL DE SERVICIO DE COMPUTO PARA LA ADMINISTRACION

## RESGUARDOS POR EMPLEADO

NOMBRE: POBLANO ORDÓÑEZ SEBASTIAN

R.F.C.: P005-460225

DEPARTAMENTO: 306

NUMERO INVENTARIO	FOLIO	TIPO	GRUPO	DESCRIPCION	MARCA	MODELO	PRECIO UNITARIO
34064	93011	V1	07	TARJETERO DE DOS CAJONES PARA TARJETA CHICA	D.N.NACIONAL	11 CAJONE P/27.	0
240092	78753	03		P E R C H E R O		4 GANCHOS	0
240167	60906	AC	07	SILLON GIRATORIO CON ASIENYO Y RESPALDO DE TELA.	D.N. NACIONAL	BEIGE Y CAFE	0
240169	47080	A1	07	ESCRITORIO EJECUTIVO	D.N. NACIONAL	5 CAJONES	0
240278	90810	BA	07	SILLA FIJA CON ASIENYO Y RESPALDO DE PLASTICO.	D.N.NACIONAL	TAPIZ NEGRO	0
278010	90810	AA	07	SILLA SECRETARIAL GIRATORIA CON ASIENYO Y RESPALDO EN PLASTICO	D.N.NACIONAL	TAPIZ MIEL	0
278243	13706	B7	07	ARCHIVERO TAMAJO OFICIO CON 7 GAVETAS.	D.N.NACIONAL	FINDENTE	0
327199	70811	V2	07	MESA REDONDA	D.N. NACIONAL	CUBIERTA MADERA	0
330866	90810	BA	07	SILLA FIJA CON ASIENYO Y RESPALDO DE PLASTICO.	D.N.NACIONAL	TAPIZ NEGRO	0

Me hago responsable de los bienes relacionados y notificarle a la sección de inventarios de cualquier cambio, sustitución, robo o pérdida etc.

---

 POBLANO ORDÓÑEZ SEBASTIAN

## UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

## DIRECCION GENERAL DE SERVICIO DE COMPUTO PARA LA ADMINISTRACION

## REPORTE DE RESGUARDOS POR DEPARTAMENTOS

DEPARTAMENTO: 306

ASESORIA

NUMERO INVENTARIO	FOLIO	TIPO	GRUPO	DESCRIPCION	MARCA	MODELO	R.F.C.	
							RESPONSABLE	NOMBRE UNITARIO
54884	93011	VI	07	TARJETERO DE DOS CAJONES PARA TARJETA CHICA	D.M.NACIONAL	11 CAJONE P/27.	PO05-460225	POBLANO DRDGEZ SEBASTIAN
240092	70753	07		P E R C H E R D		4 GANCHOS	PO05-460225	POBLANO DRDGEZ SEBASTIAN
240167	90906	AC	07	SILLON GIRATORIO CON ASIENTO Y RESPALDO DE TELA.	D.M. NACIONAL	BEIGE Y CAFE	PO05-460225	POBLANO DRDGEZ SEBASTIAN
240169	47080	A1	07	ESCRITORIO EJECUTIVO	D.M. NACIONAL	5 CAJONES	PO05-460225	POBLANO DRDGEZ SEBASTIAN
240278	90810	BA	07	SILLA FIJA CON ASIENTO Y RESPALDO DE PLASTICO.	D.M.NACIONAL	TAPIZ NEGRO	PO05-460225	POBLANO DRDGEZ SEBASTIAN
278010	90810	AA	07	SILLA SECRETARIAL GIRATORIA CON ASIENTO Y RESPALDO EN PLAS	D.M.NACIONAL	TAPIZ NIEI.	PO05-460225	POBLANO DRDGEZ SEBASTIAN
278243	13206	B2	07	ARCHIVERO TANDARD OFICIO CON 2 GAVETAS.	D.M.NACIONAL	PENDIENTE	PO05-460225	POBLANO DRDGEZ SEBASTIAN
327199	70811	V2	07	MESA REDONDA	D.M. NACIONAL	CUBIERTA MADERA	PO05-460225	POBLANO DRDGEZ SEBASTIAN
330866	90810	BA	07	SILLA FIJA CON ASIENTO Y RESPALDO DE PLASTICO.	D.M.NACIONAL	TAPIZ NEGRO	PO05-460225	POBLANO DRDGEZ SEBASTIAN
333873	21292	07		BOTE PARA BASURA		LOPEZ MORTON DE MADERA	PO05-460225	POBLANO DRDGEZ SEBASTIAN
521962	68888	B	06	MADUJNA CALCULADORA	HEWLETT PACKARD		PO05-460225	POBLANO DRDGEZ SEBASTIAN
531786	88753	06		SACAPUNTAS ELECTRICO	PARASONIC		PO05-460225	POBLANO DRDGEZ SEBASTIAN
538716	90810	BA	07	SILLA FIJA CON ASIENTO Y RESPALDO DE PLASTICO.	D.M.NACIONAL	TAPIZ NEGRO	PO05-460225	POBLANO DRDGEZ SEBASTIAN

## UNIVERSIDAD NACIONAL AUTÓNOMA DE MEXICO

## DIRECCION GENERAL DE SERVICIO DE COMPUTO PARA LA ADMINISTRACION

## REPORTE DE RESGUARDOS POR CLAVE DEL ARTICULO

FOLIO TIPO GRUPO: 9096 AC 07 DESCRIPCION: SILLON GIRATORIO CON ASIENTO Y RESPALDO DE TELA.

NUMERO INVENTARIO	MARCA	MODELO	P.F.C. RESPONSABLE	NOMBRE	DEPTO.	EDO. FIS.
240167	D.M. NACIONAL	BEIGE Y CAFE	PODS-660225	POBLANO GADGUEZ SEBASTIAN	306	N
1015667	RIVIERA	AE 1 A	HEBL-530520	HERNANDEZ BANDA JOSE LUIS	308	B
1040949	RIVIERA	RECLINABLE	CADA-510930	CAMPOS ORTIZ JERONIMO	304	N
1040951	RIVIERA	RECLINABLE	CAHE-520823	CASTILLO HUERTA ERNESTO	307	N
1040952	RIVIERA	RECLINABLE	CAHE-520823	CASTILLO HUERTA ERNESTO	307	N
1040953	RIVIERA	RECLINABLE	CAEH-560725	CAMPOS ESTRADA HECTOR (ANEJO)	305	N
1040954	RIVIERA	RECLINABLE	CAEH-560725	CAMPOS ESTRADA HECTOR (ANEJO)	305	N
1040955	RIVIERA	RECLINABLE	CAEH-560725	CAMPOS ESTRADA HECTOR (ANEJO)	305	N
1170006	D.M. NACIONAL		VOMZ-520724	VOUSSAS MAPUEZ JUAN	101	N
1170038	RIVIERA	BILL CAFE OBSCUR	CAEH-560725	CAMPOS ESTRADA HECTOR (ANEJO)	305	N
1178909	P.M. STEELE	1104-AT	VALV-310829	VALDESPINO VAZQUEZ JOVY	201	N
1184900	RIVIERA	CA-20	ACRS-520322	ACEVES ROSAS SERGIO	301	N
1184901	RIVIERA	AE1A	SECA-431130	SERRATO CRUI ANDRES	302	B

Una vez descritas las entradas y salidas al sistema se analizarán los procesos globales para cumplir con los requerimientos ya especificados:

Podemos establecer la necesidad de contar con siete funciones específicas englobadas en un módulo cada una de ellas, y son las siguientes:

**MODULO 1 CATALOGOS.**

Permite obtener la información de los diferentes catálogos (partidas, artículos, departamentos y empleados) a utilizar durante el proceso de Control del Activo Fijo de la Dependencia, y se debe de poder mantener la información dando altas, bajas, cambios en los respectivos catálogos.

**MODULO 2 RESGUARDOS.**

Este módulo es el que se encarga de la captura de la asignación de los resguardos a los diferentes empleados, se debe validar el número de inventario que no se encuentre repetido, así como su número de serie en caso de que el artículo lo tenga y la información referente a los diferentes catálogos que se encuentre dada de alta, en caso contrario no permitirá que la información queda almacenada.

También se podrá cancelar un resguardo, es decir si se dio

de alta un resguardo que no existe, o que fue capturado por un error en el proceso, aquí se borraría.

**MODULO 3 CAMBIOS.**

Dentro de este módulo se realizan las diferentes correcciones o cambios a los resguardos; ya sea para asignar cambios de uno o todos los resguardos de un empleado a otro, así como modificar el departamento de un bien, la clave del artículo que no corresponda a ese número de inventario y cambios en cualquiera de los campos de cantidad, marca, modelo, precio, estado físico, tipo y fecha de adquisición, así como la ubicación del bien.

**MODULO 4 BAJAS.**

La función de este módulo es proporcionar las bajas de los diferentes resguardos y pueden ser :

- Por reemplazo del bien.
- Por que el estado físico del bien no es el adecuado.
- Por pérdida o robo del bien.

**MODULO 5 CONSULTAS.**

Este proceso permite realizar consultas de los resguardos, ya sean por número de inventario, por número de serie, por R.F.C. del empleado, por departamento y de la baja de un resguardo.

**MODULO 6 REPORTES.**

La función de este módulo es emitir los diferentes reportes, ya sean de los respectivos catálogos , resguardos clasificados por número de inventario, R.F.C. del empleado, por departamento, clave del artículo, el general de inventario, el reporte de bajas de resguardos, asignado de resguardos por departamento, por empleado y total de la dependencia.

**MODULO 7 UTILERIAS.**

Este proceso permite al usuario hacer sus respectivas protección y restauración de la información, además el usuario podrá indexar sus archivos, así como podrá dar de alta el nombre de su respectiva dependencia.

Respecto a la validación de la información del sistema se encuentran las siguientes:

- No se debe dar de alta un registro ya existente.
- No se debe dar de alta resguardo que no contenga el número de inventario, responsable del resguardo o clave del artículo.

Una vez descritas las entradas, salidas y procesos se presentan los diagramas de flujo de datos que corresponden a los procesos del sistema de Control del Activo Fijo.

**DIAGRAMAS DE FLUJO DE DATOS DEL  
SISTEMA PARA EL CONTROL DE ARTICULOS  
DE ACTIVO FIJO (RESGUARDOS)**

**0. D.F.D. DEL SISTEMA PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO**

**1. D.F.D. DE CATALOGOS DEL SISTEMA**

**1.1 D.F.D. DE PARTIDAS**

**1.1.1 D.F.D. DE ALTA DE PARTIDA**

**1.1.2 D.F.D. DE BAJA DE PARTIDA**

**1.1.3 D.F.D. DE CAMBIO EN LA PARTIDA**

**1.1.4. D.F.D. CONSULTA DE LA PARTIDA**

**1.2 D.F.D. DE ARTICULOS**

**1.2.1 D.F.D. DE ALTA DE ARTICULO**

**1.2.2 D.F.D. DE BAJA DE ARTICULO**

**1.2.3 D.F.D. DE CAMBIO EN EL ARTICULO**

**1.2.4. D.F.D. CONSULTA EN EL ARTICULO**

**1.3 D.F.D. DE DEPARTAMENTOS**

**1.3.1 D.F.D. DE ALTA DE DEPARTAMENTO**

**1.3.2 D.F.D. DE BAJA DEL DEPARTAMENTO**

**1.3.3 D.F.D. DE CAMBIO DEL DEPARTAMENTO**

**1.3.4 D.F.D. CONSULTA DEL DEPARTAMENTO**

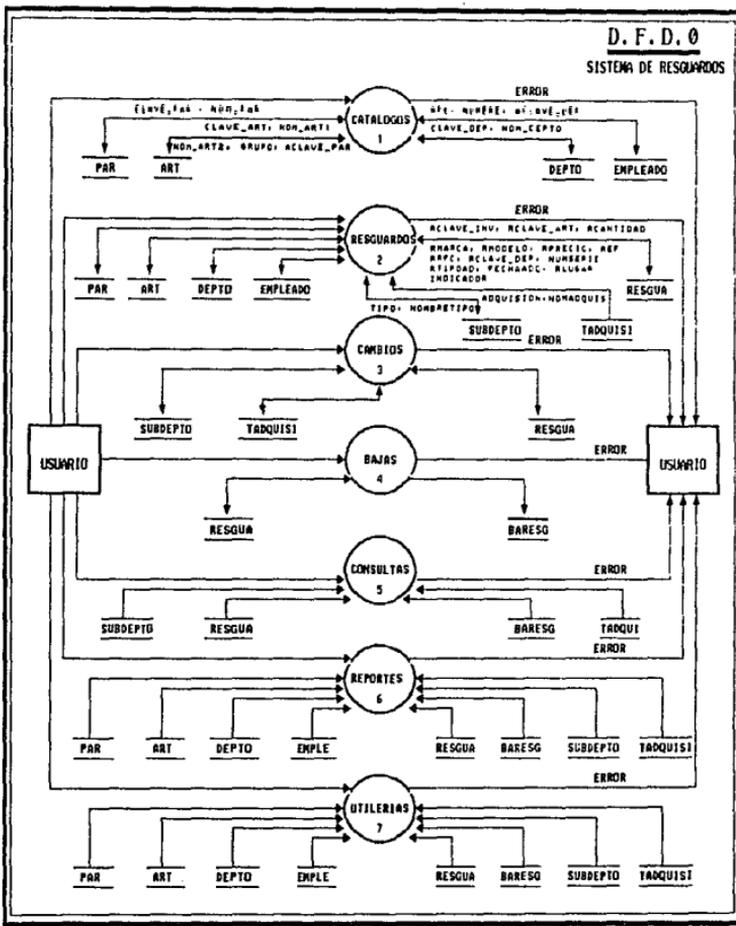
DIAGRAMAS DE FLUJO DE DATOS DEL  
SISTEMA PARA EL CONTROL DE ARTICULOS  
DE ACTIVO FIJO (RESGUARDOS)

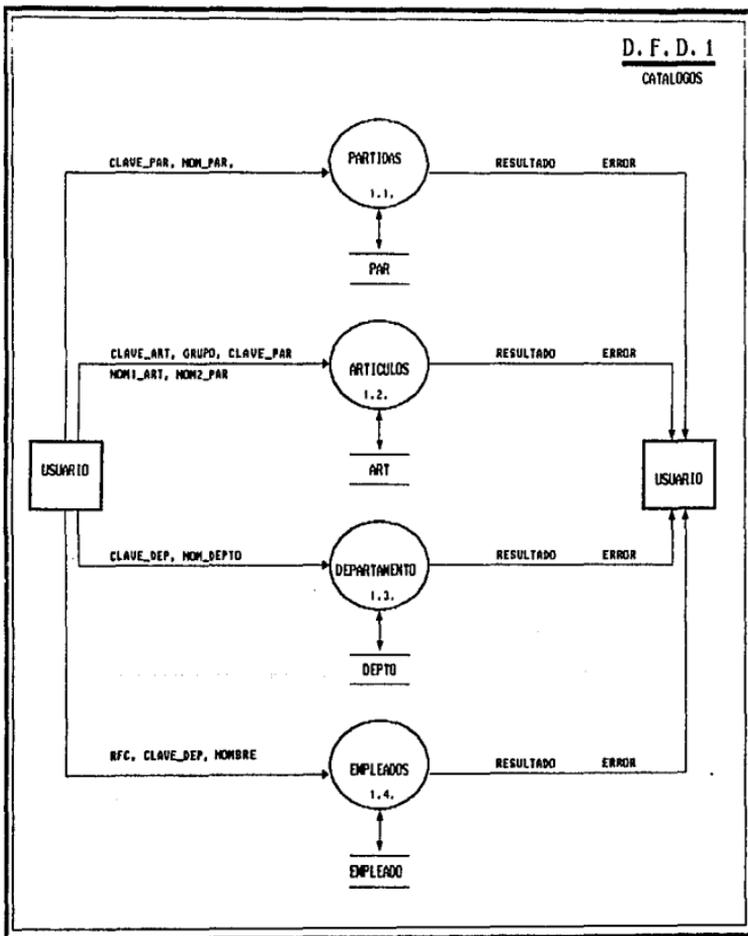
- 1.4 D.F.D. DE EMPLEADOS
  - 1.4.1 D.F.D. DE ALTA DE EMPLEADOS
  - 1.4.2 D.F.D. DE BAJA DE EMPLEADOS
  - 1.4.3 D.F.D. DE CAMBIO DE EMPLEADO
  - 1.4.4 D.F.D. CONSULTA DE EMPLEADO
  
- 2. D.F.D. DE RESGUARDOS
  - 2.1. D.F.D. DE ALTA DE RESGUARDOS
  - 2.2. D.F.D. DE CANCELACION DE RESGUARDOS
  
- 3. D.F.D. DE CAMBIOS
  - 3.1. D.F.D. DE CAMBIOS AL RESGUARDO POR NUMERO DE INVENTARIO
  - 3.2. D.F.D. DE CAMBIO EN EL RESPONSABLE DEL RESGUARDO
  - 3.3. D.F.D. DE CAMBIOS DE RESGUARDOS DE UN EMPLEADO A OTRO
  - 3.4. D.F.D. DE CAMBIOS EN EL DEPARTAMENTO DEL RESGUARDO

DIAGRAMAS DE FLUJO DE DATOS DEL  
SISTEMA PARA EL CONTROL DE ARTICULOS  
DE ACTIVO FIJO (RESGUARDOS)

- 4. D.F.D. DE BAJA DE UN RESGUARDO
  
- 5. D.F.D. DE CONSULTA DE RESGUARDOS
- 5.1. D.F.D. DE CONSULTA DE RESGUARDO POR NUMERO DE INVENTARIO
- 5.2. D.F.D. DE CONSULTA DE RESGUARDOS POR NUMERO DE SERIE
- 5.3. D.F.D. DE CONSULTA DE RESGUARDOS POR RFC DEL EMPLEADO
- 5.4. D.F.D. DE CONSULTA DE BAJA DE UN RESGUARDO
  
- 6. D.F.D. DE REPORTE
- 6.1. D.F.D. DE REPORTE DE LOS CATALOGOS
- 6.1.1. D.F.D. DE REPORTE DE CATALOGO DE ARTICULOS
- 6.1.2. D.F.D. DE REPORTE DE CATALOGO DE PARTIDAS
- 6.1.3. D.F.D. DE REPORTE DE CATALOGO DE DEPARTAMENTOS
- 6.1.4. D.F.D. DE REPORTE DE CATALOGO DE EMPLEADOS
- 6.2. D.F.D. DE REPORTE DE RESGUARDOS
- 6.2.1. D.F.D. DE REPORTE GENERAL DE RESGUARDOS
- 6.2.2. D.F.D. DE REPORTE GENERAL DE EMPLEADO (S)
- 6.2.3. D.F.D. DE REPORTE GENERAL DE DEPARTAMENTO (S)
- 6.2.4. D.F.D. DE REPORTE GENERAL DE ARTICULO (S)
- 6.3. D.F.D. DE REPORTE GENERAL DE BAJA DE RESGUARDOS
- 6.4. D.F.D. DE REPORTE DE CONCILIACION DE RESGUARDOS
- 6.5. D.F.D. DE REPORTE GENERAL DE ASIGNADO POR DEPARTAMENTO
- 6.6. D.F.D. DE REPORTE GENERAL DE ASIGNADO POR EMPLEADO

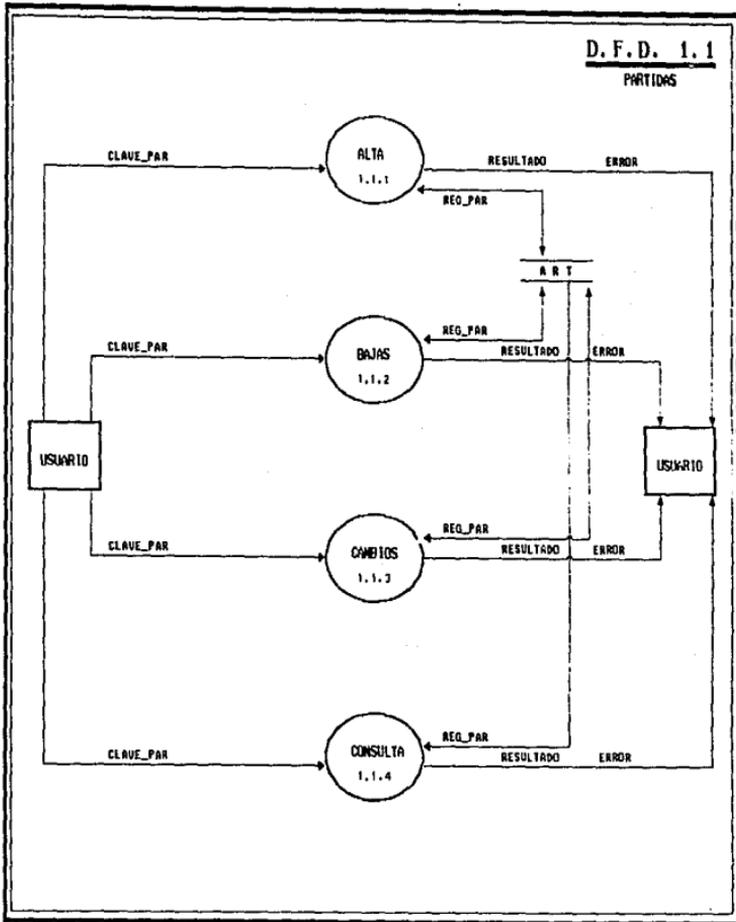
D. F. D. 0  
 SISTEMA DE RESGUARDOS





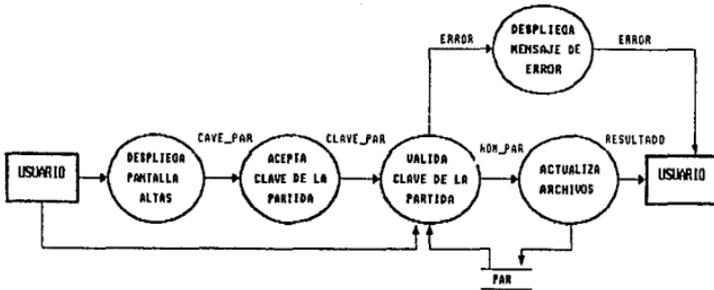
D. F. D. 1.1

PARTIDAS



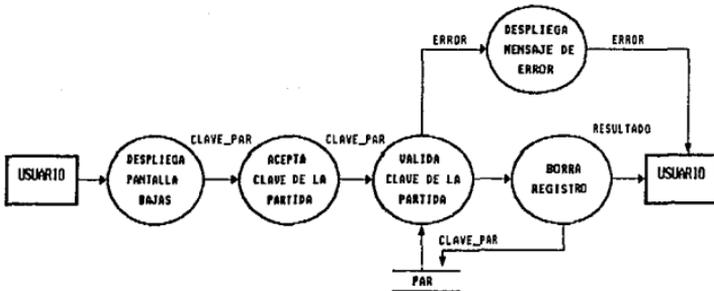
D. F. D. 1. 1. 1

ALTA DE PARTIDAS



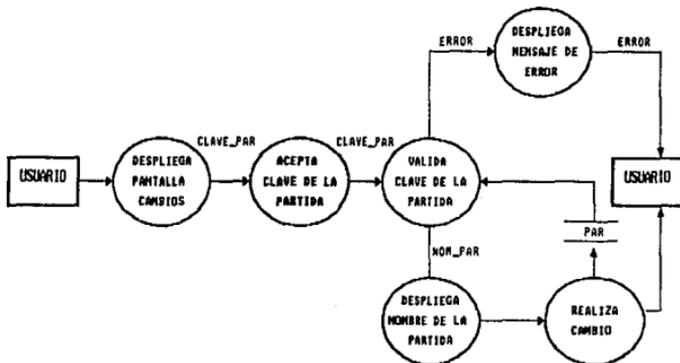
D. F. D. 1. 1. 2

BAJA DE PARTIDAS



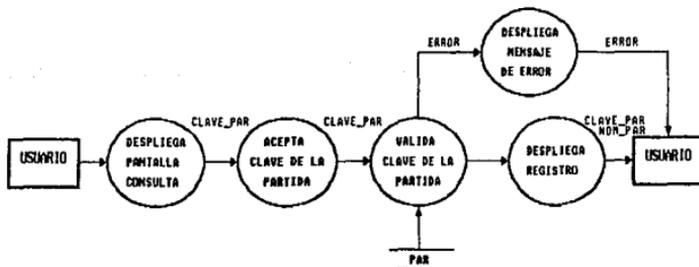
D.F.D. 1.1.3

CAMBIO EN LA PARTIDA



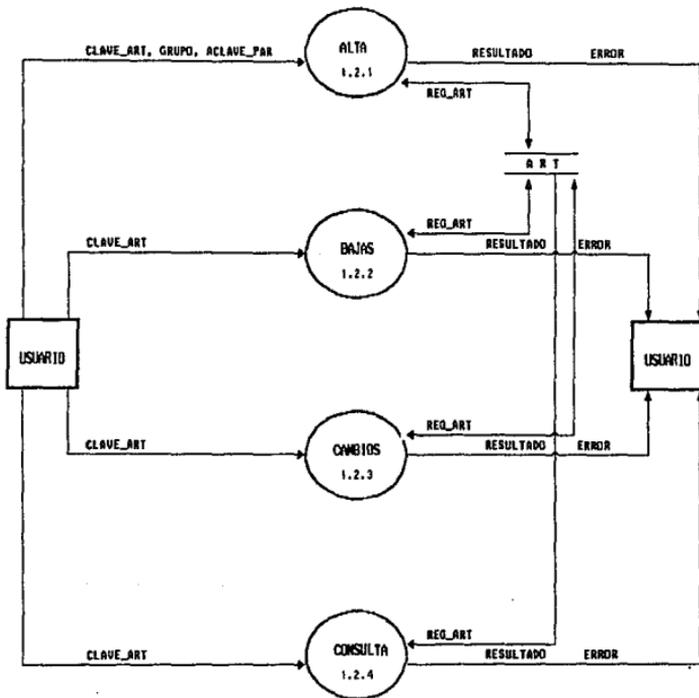
D.F.D. 1.1.4

CONSULTA EN LA PARTIDA



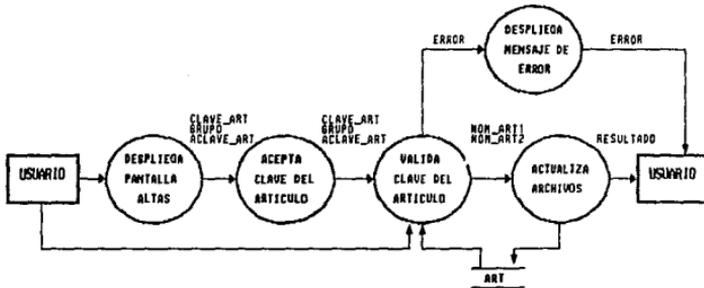
D. F. D. 1.2

ARTICULOS



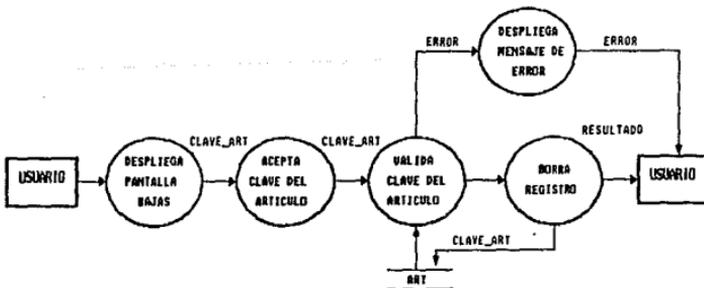
D. F. D. 1. 2. 1

ALTA DE ARTICULOS



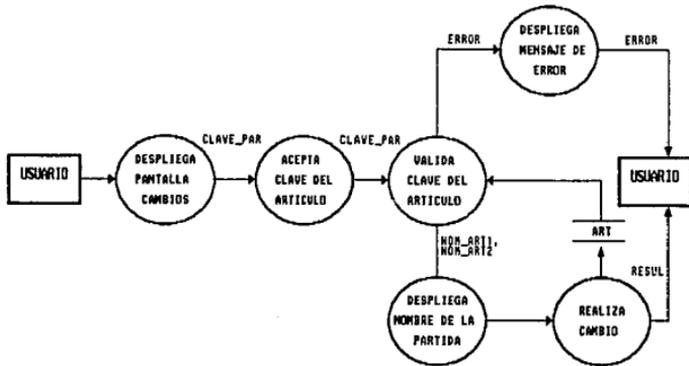
D. F. D. 1. 2. 2

BAJA DE ARTICULOS



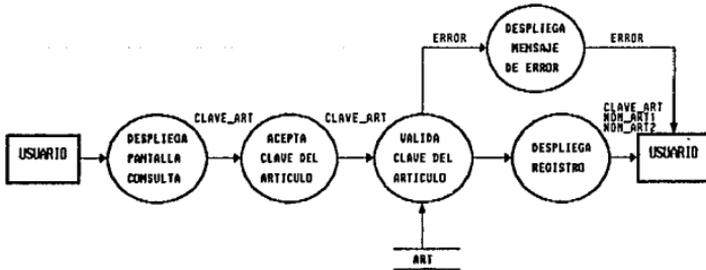
D. F. D. 1.2.3

CMDBO EN EL ARTICULO



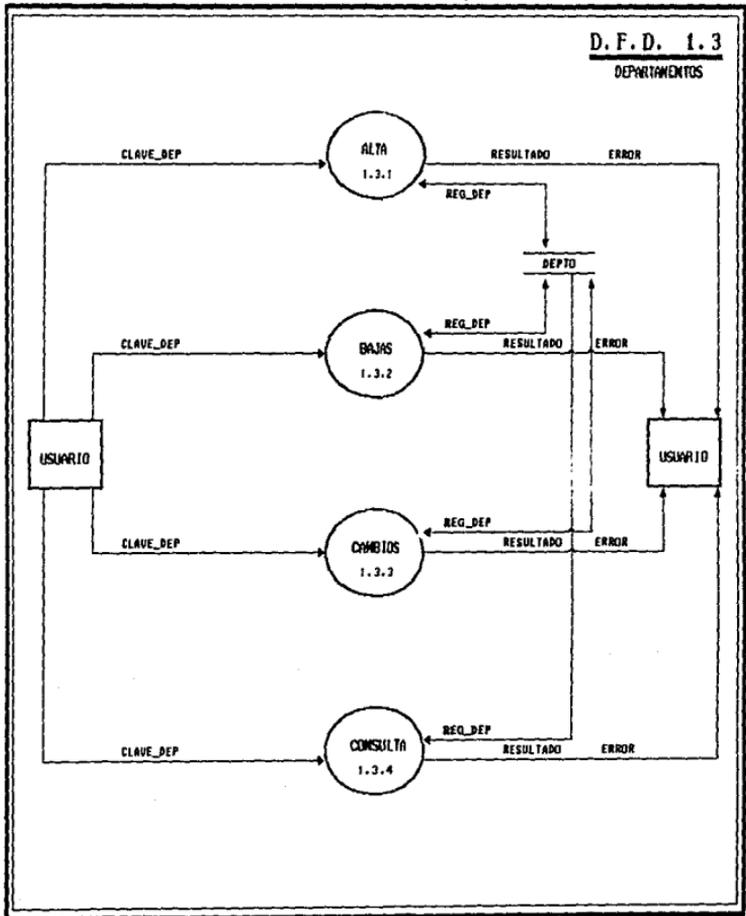
D. F. D. 1.2.4

CONSULTA DEL ARTICULO



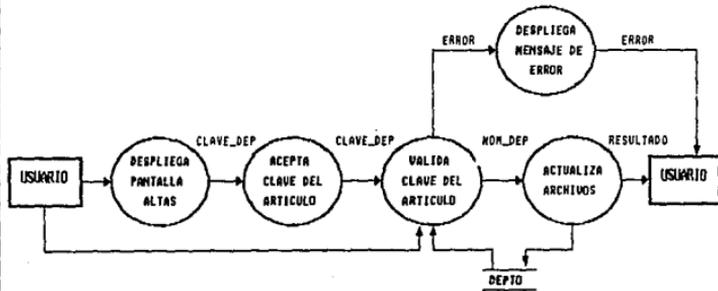
D. F. D. 1.3

DEPARTAMENTOS



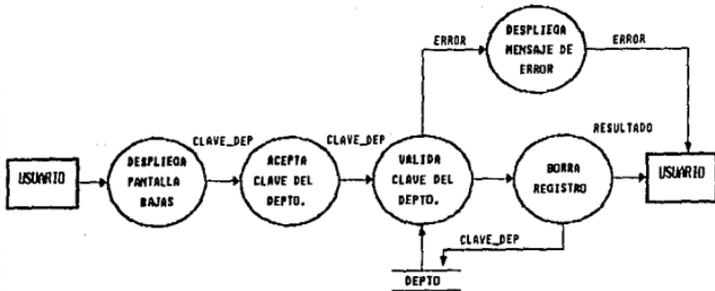
D. F. D. 1.3.1

ALTA DE DEPARTAMENTOS



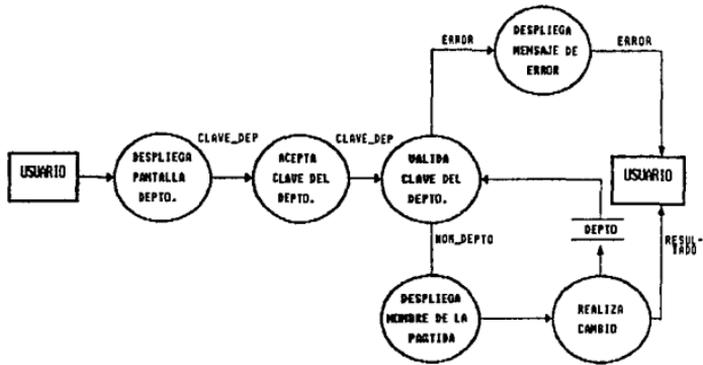
D. F. D. 1.3.2

BAJA DE DEPARTAMENTO



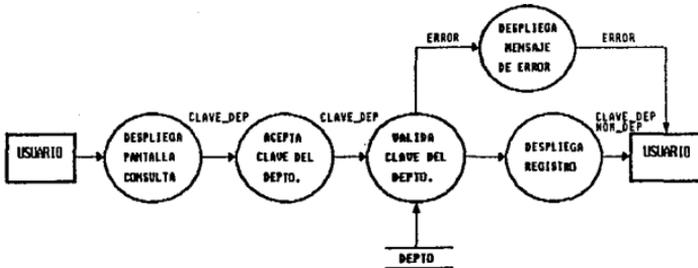
D. F. D. 1.3.3

CAMBIO EN EL DEPARTAMENTO



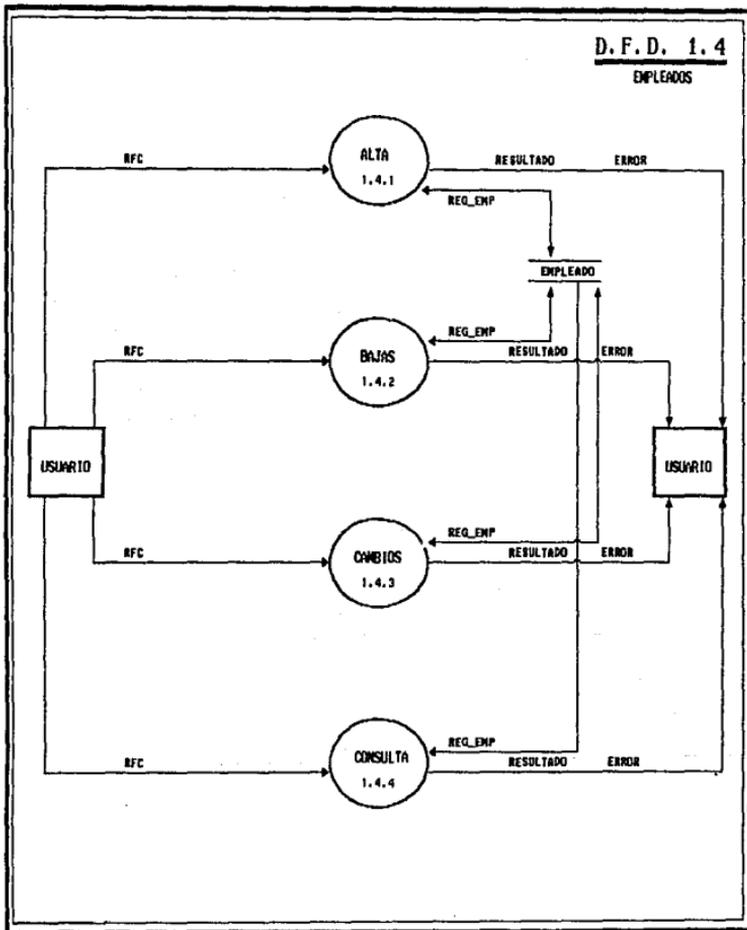
D. F. D. 1.3.4

CONSULTA EN EL DEPARTAMENTO



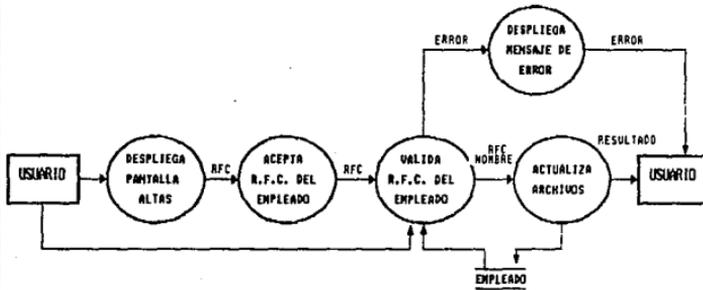
D. F. D. 1.4

EMPLEADOS



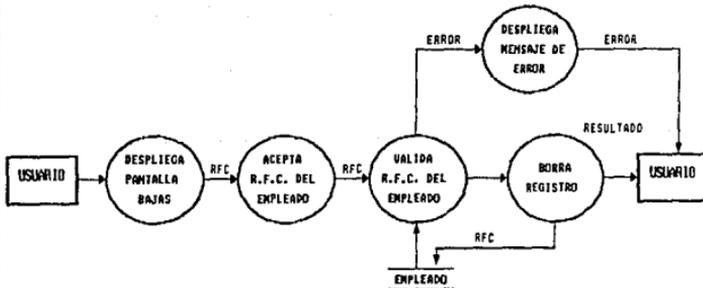
D. F. D. 1.4.1

ALTA DE EMPLEADOS



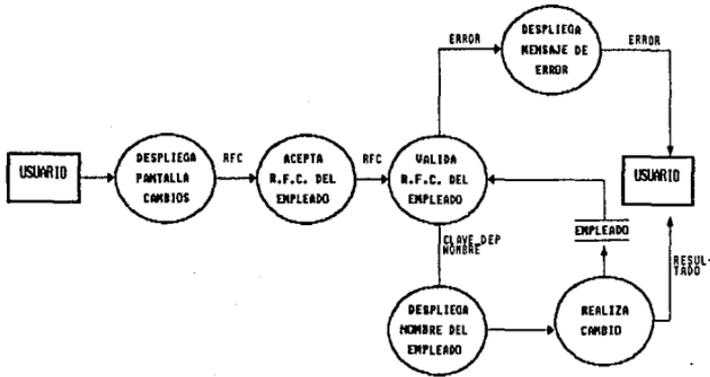
D. F. D. 1.4.2

BAJA DE EMPLEADO



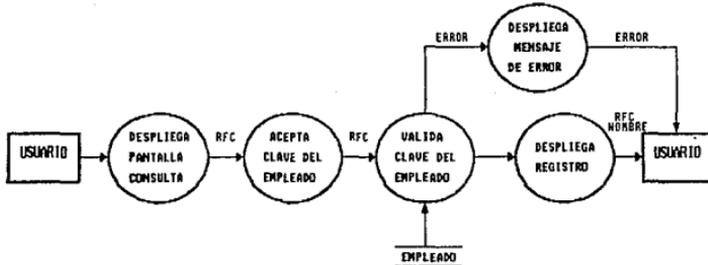
D. F. D. 1.4.3

CAMBIO EN EL EMPLEADO



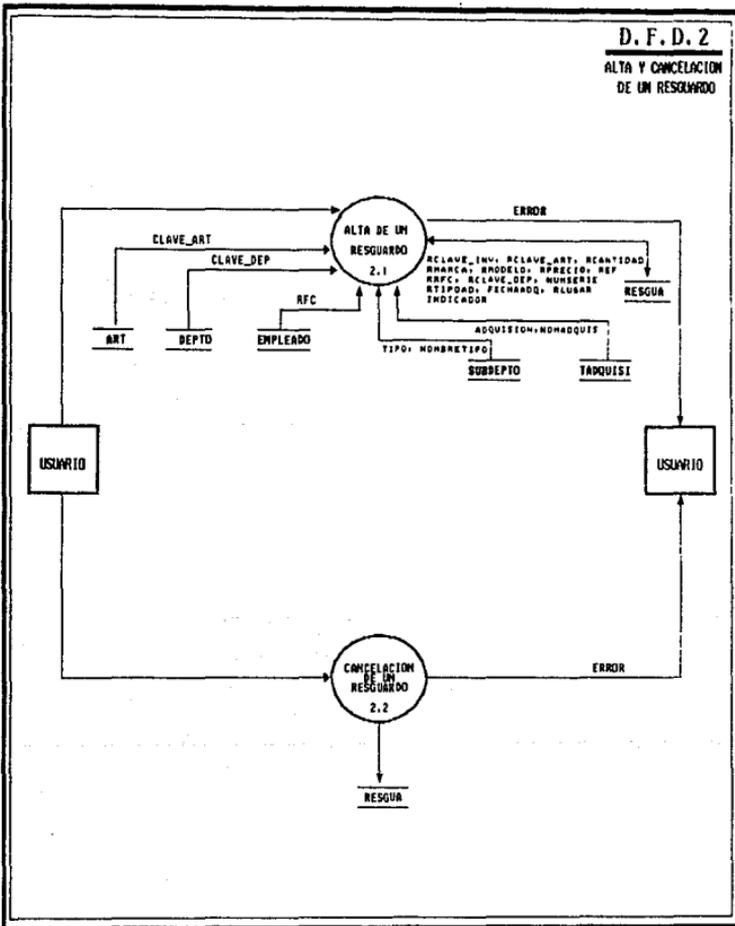
D. F. D. 1.4.4

CONSULTA EN EL DEPARTAMENTO



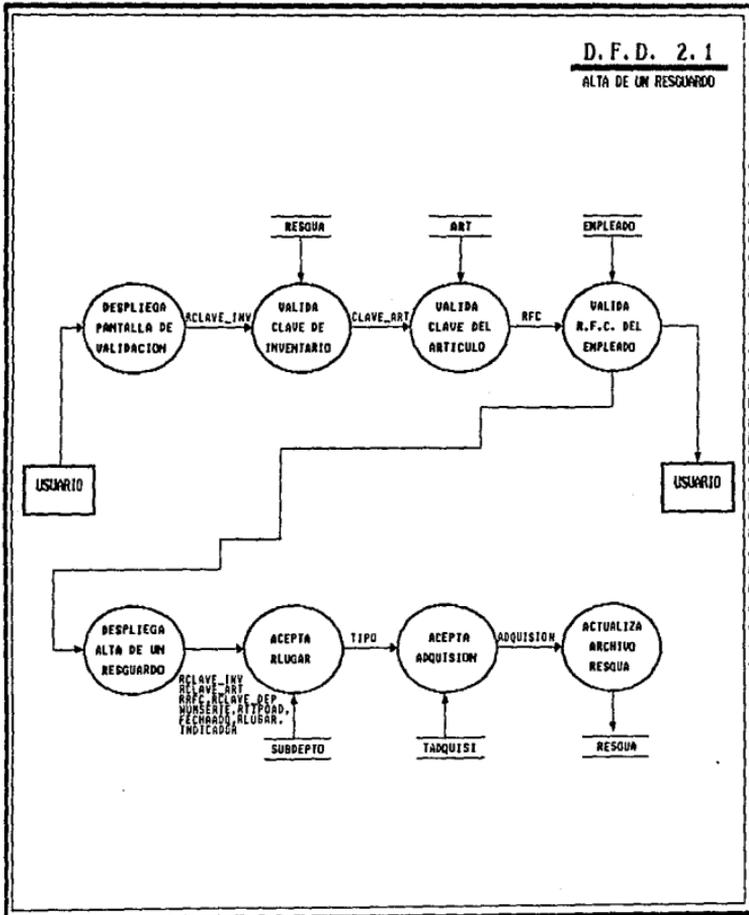
D. F. D. 2

ALTA Y CANCELACION  
DE UN RESGUARDO

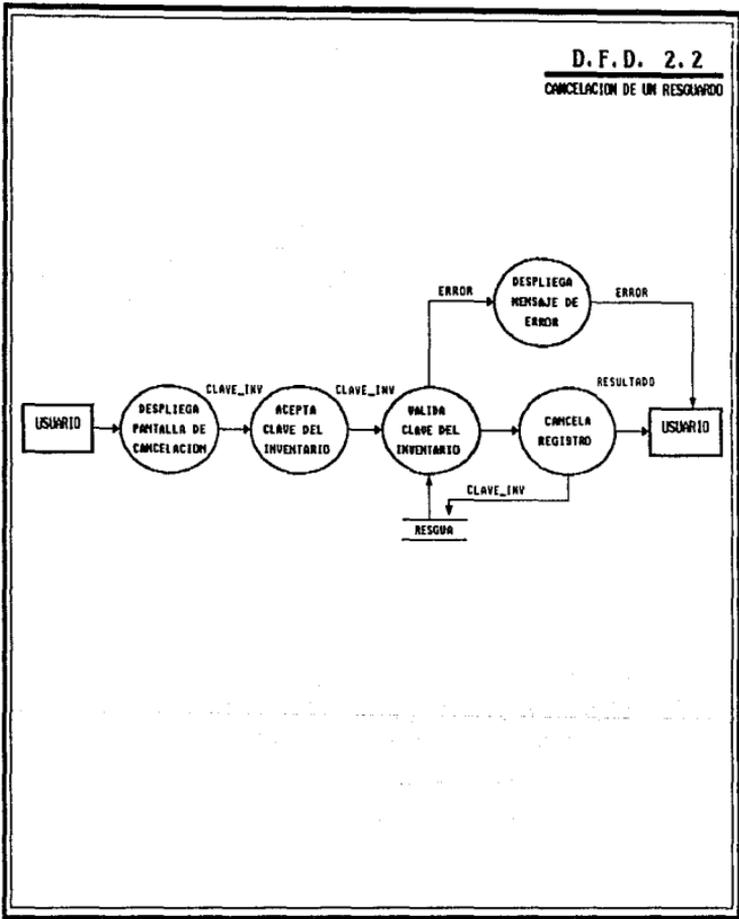


D. F. D. 2.1

ALTA DE UN RESGUARDO

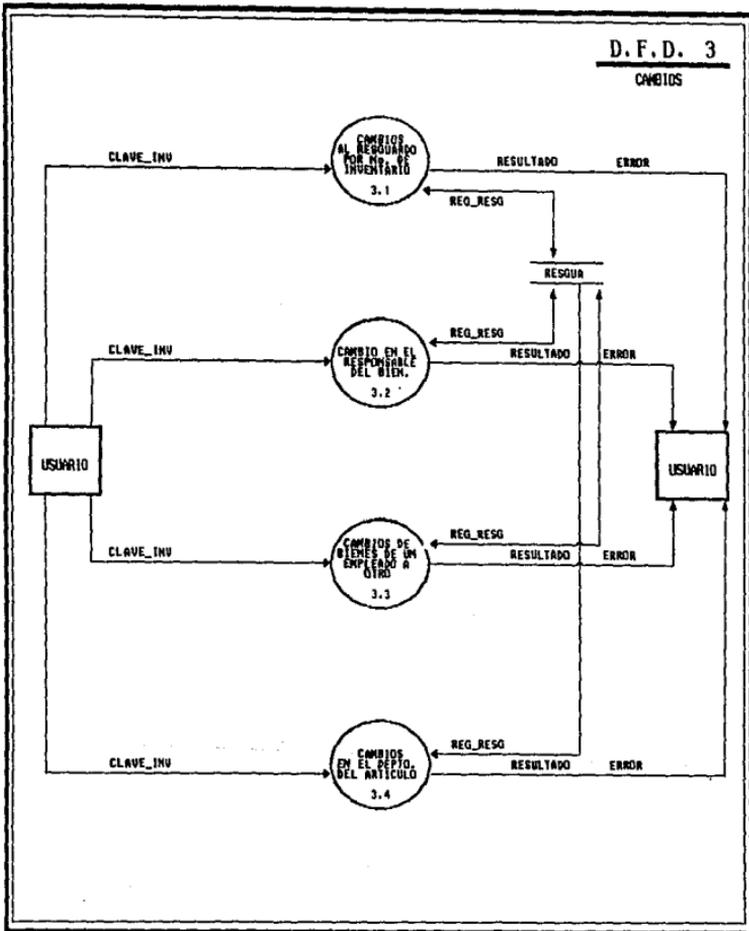


D. F. D. 2. 2  
CANCELACION DE UN RESERVO



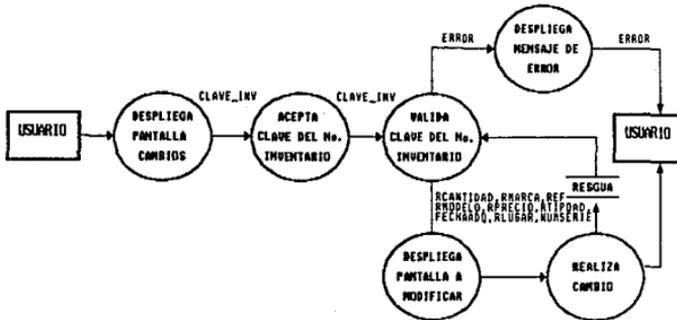
D. F. D. 3

CAMBIOS



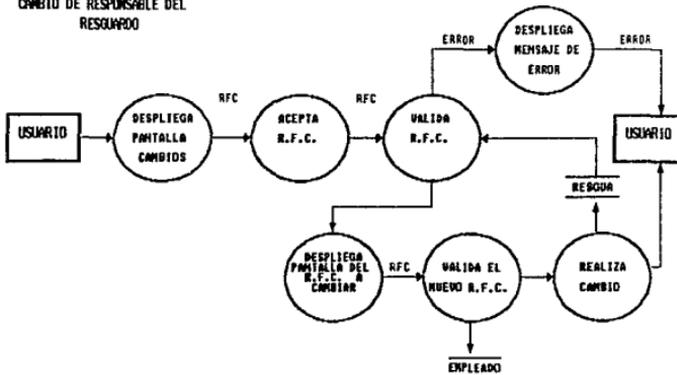
D. F. D. 3.1

CAMBIO AL RESGUARDO



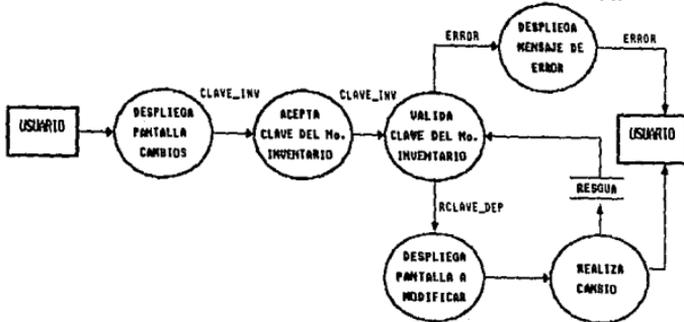
D. F. D. 3.2

CAMBIO DE RESPONSABLE DEL RESGUARDO



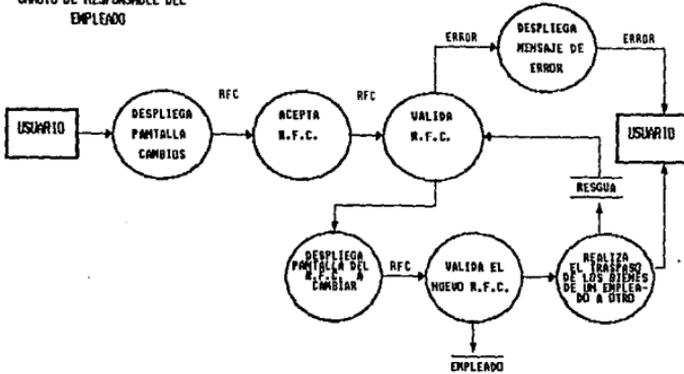
D. F. D. 3.3

CAMBIO AL DEPARTAMENTO  
DEL EMPLEADO



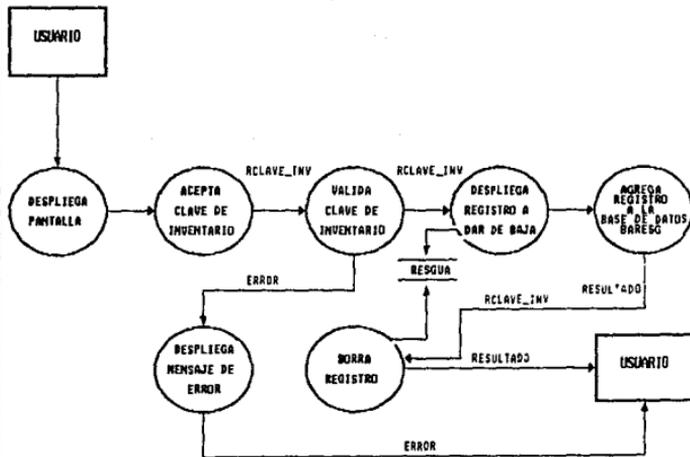
D. F. D. 3.4

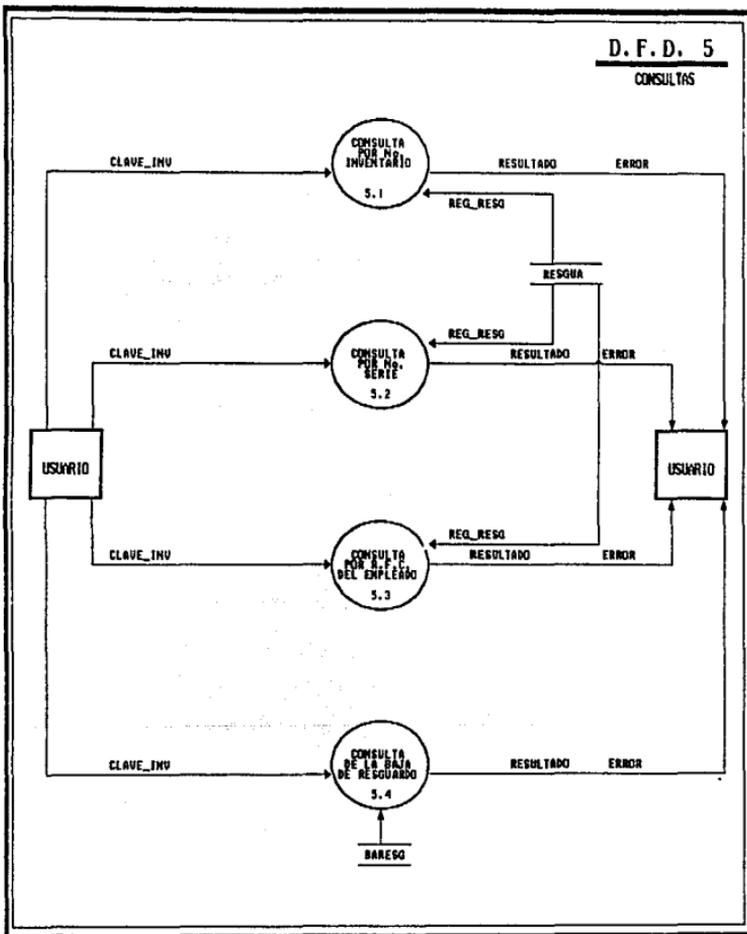
CAMBIO DE RESPONSABLE DEL  
EMPLEADO



D. F. D. 4.

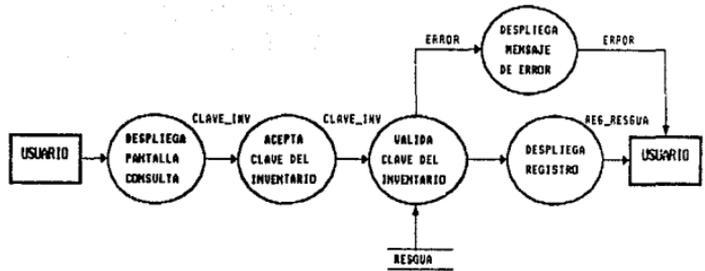
BAJA DE RESERVO  
POR No. DE INVENTARIO





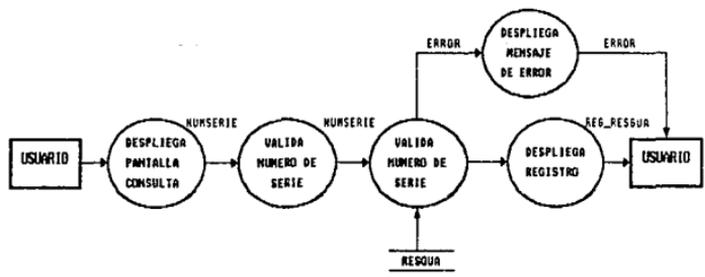
D. F. D. 5.1

CONSULTA EN EL RESGUARDO  
POR No. DE INVENTARIO



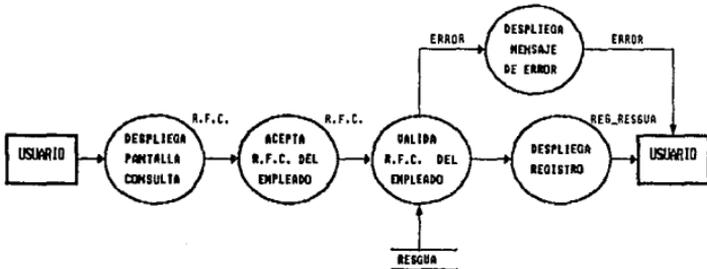
D. F. D. 5.2

CONSULTA DEL RESGUARDO  
POR No. DE SERIE



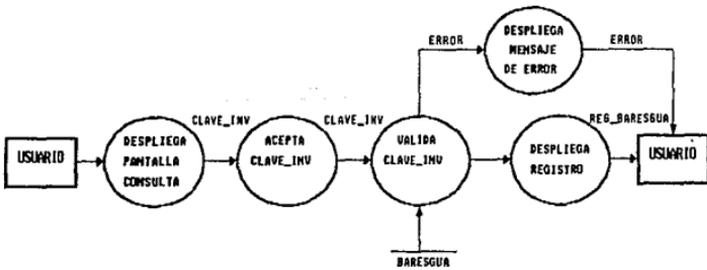
D. F. D. 5.3

CONSULTA DE RESGUARDOS  
POR R.F.C DEL EMPLEADO



D. F. D. 5.4

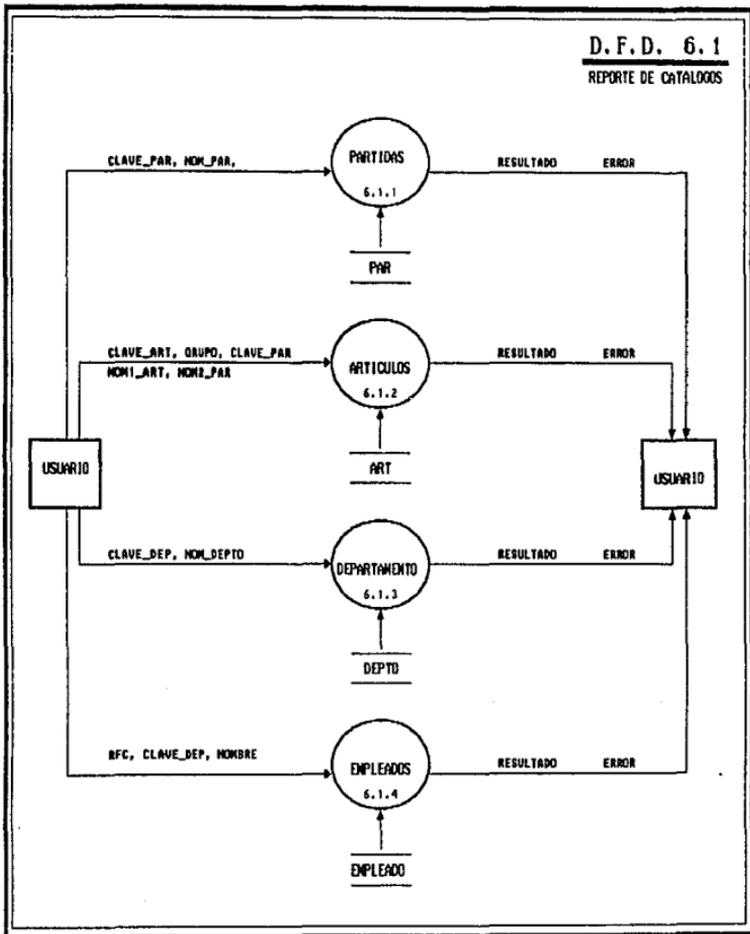
CONSULTA DE BAJA DE RESGUARDO  
POR No. DE INVENTARIO





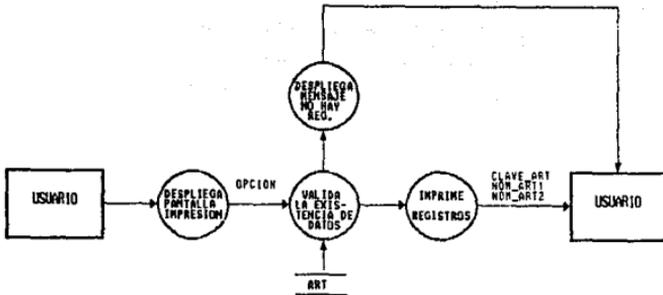
D. F. D. 6.1

REPORTE DE CATALOGOS



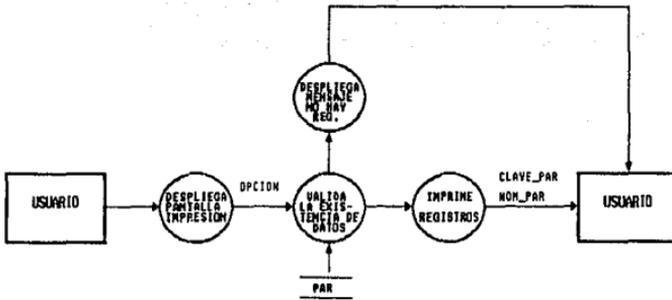
D. F. D. 6. 1. 1

REPORTE DE CATALOGO  
DE ARTICULOS



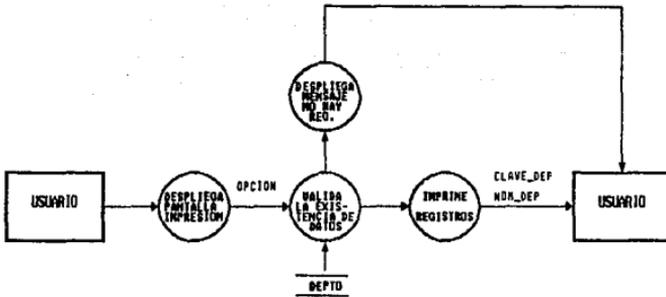
D. F. D. 6. 1. 2

REPORTE DE CATALOGO  
DE PARTIDAS



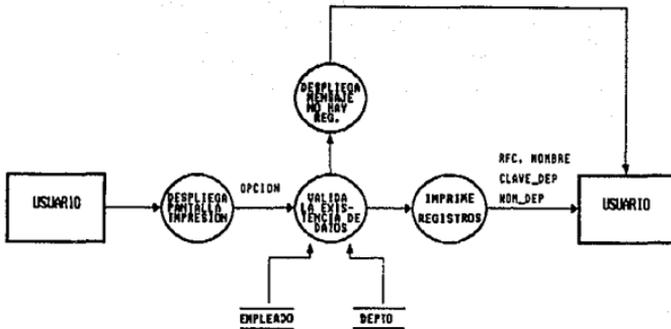
D. F. D. 6.1.3

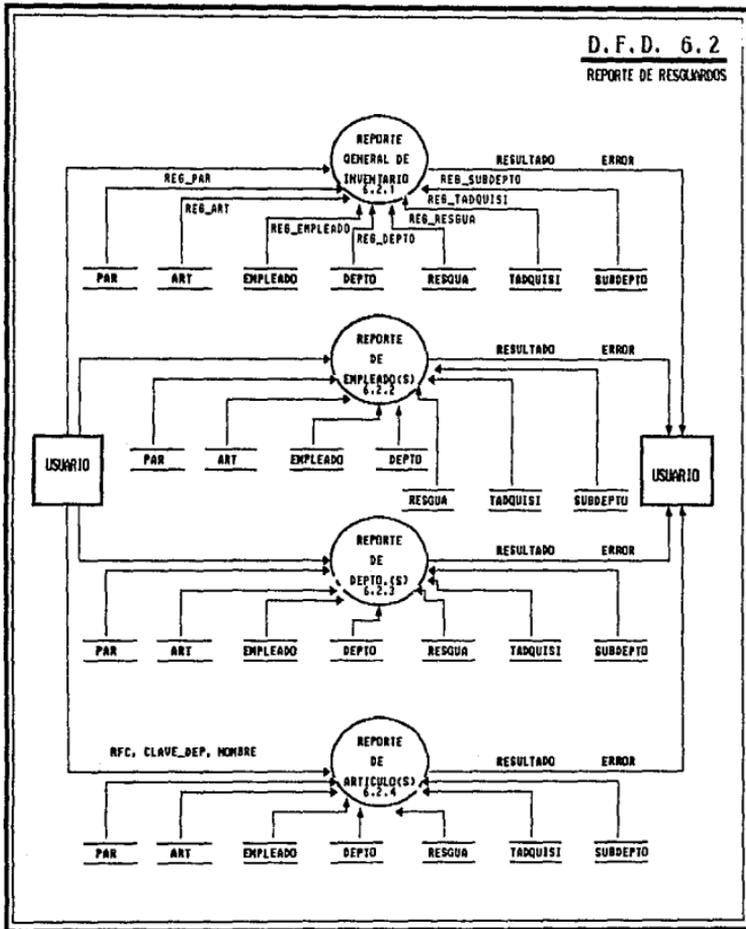
REPORTE DE CATALOGO  
DE DEPARTAMENTOS



D.F.D. 6.1.4

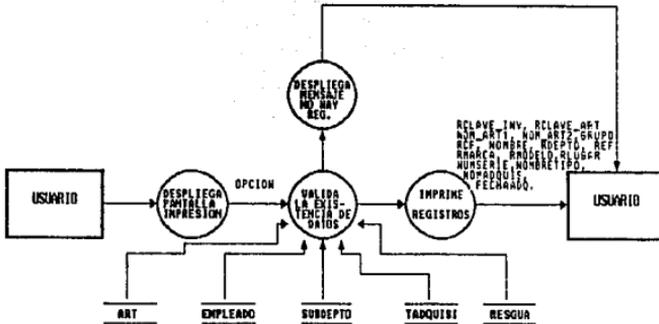
REPORTE DE CATALOGO  
DE EMPLEADOS





D. F. D. 6.2.1

REPORTE GENERAL DE  
RECURSOS

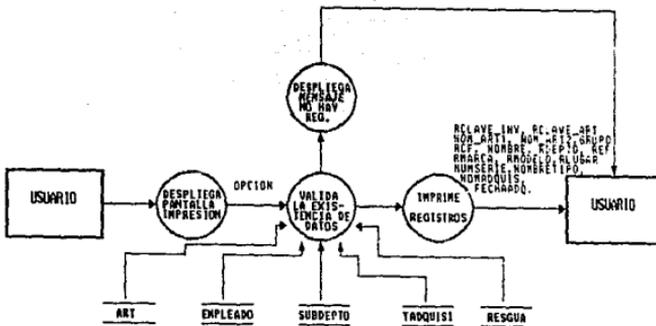






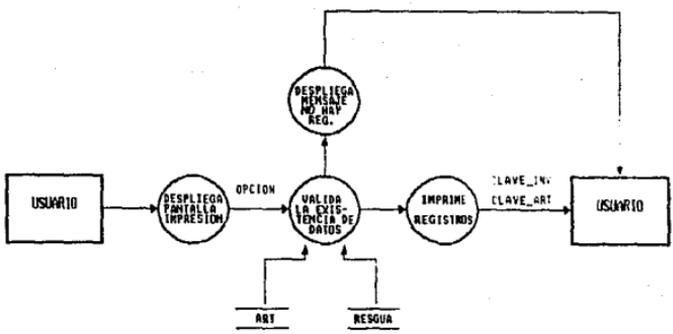
D. F. D. 6. 2. 4

REPORTE GENERAL DE  
RESUMARIOS DE ARTICULOS



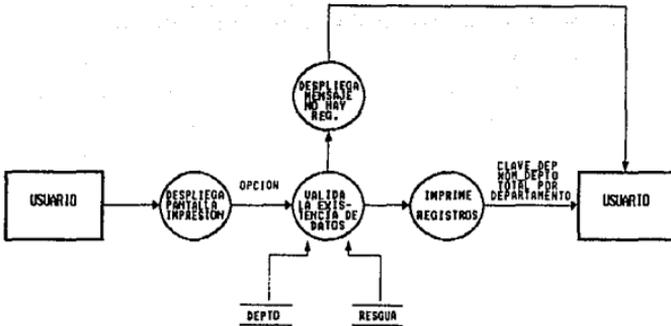


D.F.D. 6,4  
REPORTE DE CONCILIACION  
DE RESGUARDOS



D. F. D. 6.5

REPORTE DE ASIGNADO POR  
DEPARTAMENTO



### 3.2.2 DICCIONARIO DE DATOS

El diccionario de datos (DD) contiene las definiciones de todos los datos mencionados en el DFD, en una especificación del proceso y en el propio diccionario de datos. Los datos compuestos (datos que pueden ser divididos) se definen en términos de sus componentes; los datos elementales (datos que no pueden ser divididos) se definen en términos del significado de cada uno de los valores que puede asumir. Por tanto, el *diccionario de datos* está compuesto de definiciones de flujo de datos, archivos (datos almacenados) y datos usados en los procesos.

Las características que debe cumplir un buen *diccionario de datos* es que debe particionar la información, debe ser sencillo, no redundante y debe contar con una sección de sinónimos.

La notación utilizada en el diccionario de datos es la siguiente:

NOTACION	SIGNIFICADO
=	SE COMPONE DE
+	Y
[1...1]	SELECCIONA UNA DE
{ ... }	ITERACIONES
( ... )	OPCIONAL
* ... *	COMENTARIO

**DICCIONARIO DE DATOS DE LOS DFD's**

**SECCION DE DATOS COMPUESTOS:**

Partida = clave\_par + nombre\_par

Articulo = clave\_art + grupo + aclave\_par + nom\_art1 + nom\_art2

depto. = clave\_dep + nom\_depto

empleado = rfc + eclave\_dep + nombre

resguardo = rclave\_inv + numserie + rclave\_art + rcantida + rmarca + rmodelo + rrfc + rclave\_dep + rlugar + rtipoad

adquisición= adquisicion + nomadquis

lugar = tipo + nombretipo

Baresgua = rclave\_inv + rclave\_art + rcantida + rmarca + rmodelo + rrfc + rclave\_dep + rlugar + rtipoad + fechabaja + folio

**SECCION DE DATOS ELEMENTALES**

.clave\_par = { digito }  
 nombre\_par = { caracter }

clave\_art = { digito }  
 grupo = { digito }  
 aclave\_par = { digito }  
 nom\_art1 = { caracter }  
 nom\_art2 = { caracter }

```
clave_dep = ( digito )
nom_depto = ( caracter )
rfc       = ( caracter )
clave_dep = ( digito )
nombre    = ( caracter )
```

```
rclave_inv= ( digito )
numserie  = ( caracter )
rclave_art= ( digito )
rcantidad = ( digito )
rmarca    = ( caracter )
rmodelo   = ( caracter )
rprecio   = ( digito )
ref       = ( rrfc )
rrfc      = ( caracter )
rclave_dep= ( digito )
rtipond   = ( digito )
fechaadq  = dd + mm + aa
rlugar    = ( caracter )
folio     = ( digito )
dd        = [01/..../31]
mm        = [01/..../12]
aa        = [01/..../99]
```

```
adquisicion = ( digito )
nomadquis   = ( caracter )
```

```
tipo        = ( caracter )
nombretipo  = ( caracter )
```

## SECCION ARCHIVOS

PAR = clave\_par + nombre\_par  
ART = clave\_art + grupo + aclave\_par + nom\_art1 +  
nom\_art2  
DEPTO = clave\_dep + nom\_depto  
EMPLEADO = rfc + eclave\_dep + nombre  
REGUA = rclave\_inv + rcantiad + rmarca + rmodelo + rprecio  
+ ref + rrfc + rclave\_dep  
TADQUISI = adquisición + nomadquis  
SUBDEPTO = tipo + nombretipo  
BARESG = rclave\_inv + rcantiad + rmarca + rmodelo + rprecio  
+ ref + rrfc + rclave\_dep + fechabaja + folio

## SECCION DE SINONIMOS

clave\_art = aclave\_art = rclave\_art  
clave\_dep = eclave\_dep = rclave\_dep  
rfc = rrfc

### 3.3 MINIESPECIFICACIONES

Una miniespecificación de las primitivas funcionales con el objetivo de tener una información concisa de lo que está desarrollado, no ambigua, lógica y no física, completa que indique qué hace el proceso y no cómo lo hace. Para el desarrollo de estas miniespecificaciones se usa el Español Estructurado el cual es un lenguaje de especificación considerado como un subconjunto del idioma español con un conjunto limitado de maneras de combinar oraciones.

Es de vocabulario limitado porque solo se utilizan verbos en modo imperativo, existen palabras reservadas y se relaciona los datos (DFD y DD).

Una miniespecificación debe ser entendible para toda la gente y no tener expresiones algebraicas matemáticas sino que éstas deben ser narradas.

A continuación se presentan las miniespecificaciones de los procesos descritos en los DFD's.

**DEFINICION DE MINIESPECIFICACIONES DEL SISTEMA  
PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO**

**PROCESO : CONTROLA EL MENU PRINCIPAL**

**OBJETIVO: SELECCIONAR EL MOVIMIENTO A REALIZAR DEPENDIENDO  
DE LA OPCION SELECCIONADA POR EL USUARIO.**

**Proceso : SISTEMA DE CONTROL DE ACTIVO FIJO.**

**REPITE**

**Desplegar el menú principal**

**Leer opción**

**CASO de opción igual a:**

**ACTUALIZAR Y MANTENER LOS CATALOGOS**

**RESGUARDOS**

**REALIZAR CAMBIOS A RESGUARDOS**

**REALIZAR BAJAS**

**CONSULTAR RESGUARDOS**

**CREAR REPORTES**

**REALIZAR MANTENIMIENTO**

**SALIR DEL SISTEMA**

**De otra manera:**

**Desplegar mensaje de error**

**HASTA opción = fin**

**FIN DEL SISTEMA.**

**PROCESO: ACTUALIZAR Y MANTENER CATALOGOS**

**OBJETIVO: SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.**

**Proceso : ACTUALIZA Y MANTIENE CATALOGOS**

**REPITE**

**Desplegar el menú de catálogos**

**Leer opción**

**CASO de opción igual a:**

**PARTIDAS**

**ARTICULOS**

**DEPARTAMENTOS**

**EMPLEADOS**

**REGRESAR AL MENU PRINCIPAL.**

**De otra manera:**

**Desplegar mensaje de error**

**HASTA opción = fin**

**FIN DE ACTUALIZA CATALOGOS.**

**PROCESO: ACTUALIZAR Y MANTENER CATALOGO DE PARTIDAS**

**OBJETIVO: SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.**

**Proceso : ACTUALIZA Y MANTIENE CATALOGO DE PARTIDAS**

**REPITE**

Desplegar el menú de partidas

Leer opción

CASO de opción igual a:

REALIZA ALTAS EN EL CATALOGO DE PARTIDAS

REALIZA BAJAS EN EL CATALOGO DE PARTIDAS

REALIZA CAMBIOS EN EL CATALOGO DE PARTIDAS

REALIZA CONSULTAS EN EL CATALOGO DE PARTIDAS

REGRESAR AL MENU DE CATALOGOS.

De otra manera:

Desplegar mensaje de error

HASTA opción = fin

**FIN DE ACTUALIZA CATALOGO DE PARTIDAS.**

**PROCESO: REALIZA ALTAS DE PARTIDAS**

**OBJETIVO: DAR DEL ALTA REGISTROS DE PARTIDAS EN LA BASE DE DATOS.**

**Proceso : REALIZA ALTA DE PARTIDAS.**

**REPITE**

Desplegar pantalla de altas

Leer clave de la partida y nombre de la partida

Valida clave de la partida

Actualiza base de datos

HASTA otra alta = 'no'

**FIN DE ALTAS DE PARTIDAS.**

**PROCESO: REALIZA BAJAS DE PARTIDAS**

**OBJETIVO: DAR DE BAJA REGISTROS DE PARTIDAS EN LA BASE DE DATOS.**

**Proceso : REALIZA BAJAS DE PARTIDAS.**

**REPITE**

Desplegar pantalla de bajas

Leer clave de la partida

Valida clave de la partida

Borra registro de la base de datos

HASTA otra baja = 'no'

**FIN DE BAJAS DE PARTIDAS.**

**PROCESO:** REALIZA CAMBIOS A LAS PARTIDAS

**OBJETIVO:** MODIFICAR REGISTROS DE PARTIDAS EN LA BASE DE DATOS.

**Proceso :** REALIZA CAMBIOS A LAS PARTIDAS.

REPITE

Desplegar pantalla de cambios

Leer clave de la partida

Valida clave de la partida

Sustituye los datos anteriores por los nuevos en la base de datos

HASTA otra cambio = 'no'

FIN DE CAMBIOS DE PARTIDAS.

**PROCESO:** REALIZA CONSULTA DE PARTIDAS

**OBJETIVO:** REALIZA CONSULTA DE REGISTROS DE PARTIDAS EN LA BASE DE DATOS.

**Proceso :** REALIZA CONSULTA DE PARTIDAS.

REPITE

Desplegar pantalla de consulta

Leer clave de la partida

Valida clave de la partida

Despliega la información

HASTA otra baja = 'no'

FIN DE CONSULTAS DE PARTIDAS.

**PROCESO:** ACTUALIZAR Y MANTENER CATALOGO DE ARTICULOS

**OBJETIVO:** SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.

**Proceso :** ACTUALIZA Y MANTIENE CATALOGO DE ARTICULOS

**REPITE**

Desplegar el menú de catálogo de artículos

Leer opción

CASO de opción igual a:

REALIZA ALTAS EN EL CATALOGO DE ARTICULOS

REALIZA BAJAS EN EL CATALOGO DE ARTICULOS

REALIZA CAMBIOS EN EL CATALOGO DE ARTICULOS

REALIZA CONSULTAS EN EL CATALOGO DE ARTICULOS

REGRESAR AL MENU DE CATALOGOS.

De otra manera:

Desplegar mensaje de error

HASTA opción = fin

**FIN DE ACTUALIZA CATALOGO DE ARTICULOS.**

**PROCESO: REALIZA ALTAS DE ARTICULOS**

**OBJETIVO: DAR DE ALTA REGISTROS DE ARTICULOS EN LA BASE DE DATOS.**

**Proceso : REALIZA ALTA DE ARTICULOS.**

**REPITE**

Desplegar pantalla de altas

Leer clave, partida y nombre del articulo

Valida clave y partida del articulo

Actualiza base de datos

HASTA otra alta = 'no'

**FIN DE ALTAS DE ARTICULOS.**

**PROCESO: REALIZA BAJAS DE ARTICULOS**

**OBJETIVO: DAR DE BAJA REGISTROS DE ARTICULOS DE LA BASE DE DATOS.**

**Proceso : REALIZA BAJAS DE ARTICULOS.**

**REPITE**

Desplegar pantalla de bajas

Leer clave del articulo

Valida clave del articulo

Borra registro de la base de datos

HASTA otra baja = 'no'

**FIN DE BAJAS DE ARTICULOS.**

**PROCESO:** REALIZA CAMBIOS A LOS ARTICULOS

**OBJETIVO:** MODIFICAR REGISTROS DE ARTICULOS EN LA BASE DE DATOS.

**Proceso :** REALIZA CAMBIOS A LOS ARTICULOS.

REPITE

Desplegar pantalla de cambios

Leer clave del artículo

Valida clave del artículo

Sustituye los datos anteriores por los nuevos en la base de datos

HASTA otra cambio = 'no'

FIN DE CAMBIOS DE ARTICULOS.

**PROCESO:** REALIZA CONSULTA DE ARTICULOS.

**OBJETIVO:** REALIZAR CONSULTAS EN LOS REGISTROS DE ARTICULOS DE LA BASE DE DATOS.

**Proceso :** REALIZA CONSULTA DE ARTICULOS

REPITE

Desplegar pantalla de consulta

Leer clave del artículo

Valida clave del artículo

Despliega la información

HASTA otra baja = 'no'

FIN DE CONSULTAS DE ARTICULOS.

**PROCESO:** ACTUALIZAR Y MANTENER CATALOGO DE DEPARTAMENTOS

**OBJETIVO:** SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.

Proceso : ACTUALIZA Y MANTIENE CATALOGO DE DEPARTAMENTOS

REPITE

Desplegar el menú de catálogo de departamento

Leer opción

CASO de opción igual a:

REALIZA ALTAS EN EL CATALOGO DE DEPARTAMENTOS

REALIZA BAJAS EN EL CATALOGO DE DEPARTAMENTOS

REALIZA CAMBIOS EN EL CATALOGO DE DEPARTAMENTOS

REALIZA CONSULTAS EN EL CATALOGO DEPARTAMENTOS

REGRESAR AL MENU DE CATALOGOS.

De otra manera:

Desplegar mensaje de error

HASTA opción = fin

FIN DE ACTUALIZA CATALOGO DE DEPARTAMENTOS.

**PROCESO:** REALIZA ALTAS DE DEPARTAMENTOS

**OBJETIVO:** DAR DE ALTA REGISTROS DE DEPARTAMENTOS EN LA BASE DE DATOS.

**Proceso :** REALIZA ALTA DE DEPARTAMENTOS.

REPITE

Desplegar pantalla de altas  
Leer clave y nombre del departamento  
Valida clave del departamento  
Actualiza base de datos

HASTA otra alta = 'no'

FIN DE ALTAS DE DEPARTAMENTOS.

**PROCESO:** REALIZA BAJAS DE DEPARTAMENTOS

**OBJETIVO:** DAR DE BAJA REGISTROS DE DEPARTAMENTOS EN LA BASE DE DATOS.

**Proceso :** REALIZA BAJAS DE DEPARTAMENTOS.

REPITE

Desplegar pantalla de bajas  
Leer clave del departamento  
Valida clave del departamento  
Borra registro de la base de datos

HASTA otra baja = 'no'

FIN DE BAJAS DE DEPARTAMENTOS.

**PROCESO:** REALIZA CAMBIOS A LOS DEPARTAMENTOS

**OBJETIVO:** MODIFICAR REGISTROS DE DEPARTAMENTOS EN LA BASE DE DATOS.

**Proceso :** REALIZA CAMBIOS A LOS DEPARTAMENTOS.

REPITE

Desplegar pantalla de cambios

Leer clave del departamento

Valida clave del departamento

Sustituye los datos anteriores por los nuevos en la base de datos

HASTA otra cambio = 'no'

FIN DE CAMBIOS DE DEPARTAMENTOS.

**PROCESO:** REALIZA CONSULTA DE DEPARTAMENTOS.

**OBJETIVO:** CONSULTAR REGISTROS DE DEPARTAMENTOS EN LA BASE DE DATOS.

**Proceso :** REALIZA CONSULTA DE DEPARTAMENTOS

REPITE

Desplegar pantalla de consulta

Leer clave del departamento

Valida clave del departamento

Despliega la información

HASTA otra baja = 'no'

FIN DE CONSULTAS DE DEPARTAMENTOS.

**PROCESO:** ACTUALIZAR Y MANTENER CATALOGO DE EMPLEADOS

**OBJETIVO:** SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.

**Proceso :** ACTUALIZA Y MANTIENE CATALOGO DE EMPLEADOS

**REPITE**

Desplegar el menú de catálogo de empleados

Leer opción

CASO de opción igual a:

REALIZA ALTAS EN EL CATALOGO DE EMPLEADOS

REALIZA BAJAS EN EL CATALOGO DE EMPLEADOS

REALIZA CAMBIOS EN EL CATALOGO DE EMPLEADOS

REALIZA CONSULTAS EN EL CATALOGO EMPLEADOS

REGRESAR AL MENU DE CATALOGOS.

De otra manera:

Desplegar mensaje de error

HASTA opción = fin

FIN DE ACTUALIZA CATALOGO DE EMPLEADOS.

**PROCESO: REALIZA ALTAS DE EMPLEADOS**

**OBJETIVO:** DAR DE ALTA REGISTROS DE EMPLEADOS EN LA BASE DE DATOS.

**Proceso :** REALIZA ALTA DE EMPLEADOS.

**REPITE**

Desplegar pantalla de altas

Leer rfc, clave depto. y nombre del empleado

Valida rfc del empleado

Actualiza base de datos

HASTA otra alta = 'no'

FIN DE ALTAS DE EMPLEADOS.

**PROCESO: REALIZA BAJAS DE EMPLEADOS**

**OBJETIVO:** DAR DE BAJA REGISTROS DE EMPLEADOS EN LA BASE DE DATOS.

**Proceso :** REALIZA BAJAS DE EMPLEADOS.

**REPITE**

Desplegar pantalla de bajas

Leer rfc del empleado

Valida rfc del empleado

Borra registro de la base de datos

HASTA otra baja = 'no'

FIN DE BAJAS DE EMPLEADOS.

**PROCESO:** REALIZA CAMBIOS A LOS EMPLEADOS

**OBJETIVO:** MODIFICAR REGISTROS DE EMPLEADOS EN LA BASE DE DATOS.

**Proceso :** REALIZA CAMBIOS A LOS EMPLEADOS.

REPITE

Desplegar pantalla de cambios

Leer rfc del empleado

Valida rfc del empleado

Sustituye los datos anteriores por los nuevos en la base de datos

HASTA otra cambio = 'no'

FIN DE CAMBIOS DE EMPLEADOS.

**PROCESO:** REALIZA CONSULTA DE EMPLEADOS.

**OBJETIVO:** CONSULTAR REGISTROS DE EMPLEADOS EN LA BASE DE DATOS.

**Proceso :** REALIZA CONSULTA DE EMPLEADOS

REPITE

Desplegar pantalla de consulta

Leer rfc del empleado

Valida rfc del empleado

Despliega la información

HASTA otra baja = 'no'

FIN DE CONSULTAS DE EMPLEADOS.

**PROCESO: MENU DE RESGUARDOS**

**OBJETIVO: SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.**

**Proceso : MENU DE RESGUARDOS**

**REPITE**

**Desplegar el menú de resguardos**

**Leer opción**

**CASO de opción igual a:**

**REALIZA ALTAS DE RESGUARDOS**

**REALIZA BAJAS DE RESGUARDOS**

**REGRESAR AL MENU DE RESGUARDO.**

**De otra manera:**

**Desplegar mensaje de error**

**HASTA opción = fin**

**FIN DE MENU DE RESGUARDOS.**

**PROCESO: CAPTURA DE RESGUARDOS**

**OBJETIVO: DAR DE ALTA RESGUARDOS.**

**Proceso : REALIZA CAPTURA.**

**REPITE**

Desplegar pantalla de altas

Leer número de inventario, clave del artículo y rfc del empleado responsable

Valida número de inventario, clave del artículo y rfc del empleado responsable.

Actualiza base de datos

HASTA otra alta =

FIN DE ALTA DE RESGUARDOS.

**PROCESO: REALIZA CANCELACION DE UN RESGUARDO.**

**OBJETIVO: DAR UNA CANCELACION DE UN REGISTRO DE LA BASE DE DATOS.**

**Proceso : REALIZA CANCELACIONES.**

**REPITE**

Desplegar pantalla de cancelaciones

Leer número de inventario

Valida número de inventario

Despliega la información

Cancela registro de la base de datos

HASTA otra cancelación =

FIN DE CANCELACIONES DE INVENTARIOS.

**PROCESO:** MENU DE CAMBIOS DE RESGUARDOS.

**OBJETIVO:** SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.

Proceso : MENU DE CAMBIOS DE RESGUARDOS

REPITE

Desplegar el menú de cambios de resguardos

Leer opción

CASO de opción igual a:

CAMBIO DE RESGUARDOS POR No. DE INVENTARIO

CAMBIO DE RESGUARDO DE EMPLEADOS

CAMBIO EN EL DEPARTAMENTO DEL RESGUARDO

CAMBIO EN EL LA CLAVE DEL ARTICULO DEL RESGUARDO

CAMBIO EN LA PARTIDA DEL RESGUARDO

De otra manera:

Desplegar mensaje de error

HASTA opción = fin

FIN DE MENU DE CAMBIOS DE RESGUARDOS.

**PROCESO: REALIZA CAMBIOS POR No. DE INVENTARIO**

**OBJETIVO:** PODER MODIFICAR LA CANTIDAD, MARCA, MODELO, COSTO UNITARIO, EDO. FISICO, FECHA Y TIPO DE ADQUISICION Y SUBDEPTO DEL RESGUARDO.

**Proceso :** REALIZA CAMBIO POR No. DE INVENTARIO

**REPITE**

Desplegar pantalla de cambios

Leer número de inventario

Valida número de inventario

Leer nuevos datos

Sustituir los datos anteriores por los nuevos en la base de datos

HASTA otra cambio =

**FIN DE CAMBIOS DE RESGUARDOS.**

**PROCESO: REALIZA CAMBIO DE RESGUARDOS DE EMPLEADO(S)**

**OBJETIVO:** PODER CAMBIAR AL RESPONSABLE DEL RESGUARDO.

**Proceso :** REALIZA CAMBIO DE RESGUARDOS DE UN EMPLEADO A OTRO

**REPITE**

Desplegar pantalla de cambios

Leer número de inventario

Valida número de inventario

Leer nuevos datos

Sustituir los datos anteriores por los nuevos en la base de datos

HASTA otra cambio =

**FIN DE CAMBIOS DE RESGUARDOS.**

**PROCESO: REALIZA CAMBIOS POR No. DE DEPARTAMENTO**

**OBJETIVO: PODER MODIFICAR EL DEPARTAMENTO DEL RESGUARDO.**

**Proceso : REALIZA CAMBIO POR No. DE DEPARTAMENTO**

**REPITE**

Desplegar pantalla de cambios

Leer número de inventario

Valida número de inventario

Leer nuevos datos

Sustituir los datos anteriores por los nuevos en la base de datos

HASTA otra cambio =

FIN DE CAMBIOS DE RESGUARDOS.

**PROCESO: REALIZA CAMBIO POR CLAVE DEL ARTICULO**

**OBJETIVO: PODER CAMBIAR AL RESPONSABLE DE UN RESGUARDO.**

**Proceso : REALIZA CAMBIO EN LA CLAVE DEL ARTICULO DEL RESGUARDO.**

**REPITE**

Desplegar pantalla de cambios

Leer número de inventario

Valida número de inventario

Leer nuevos datos

Sustituir los datos anteriores por los nuevos en la base de datos

HASTA otra cambio =

FIN DE CAMBIOS DE RESGUARDOS.

**PROCESO: REALIZA BAJA DEL RESGUARDO.**

**OBJETIVO: PODER DAR DE BAJA LOS RESGUARDOS.**

**Proceso : REALIZA CAMBIO EN LA PARTIDA DEL ARTICULO DEL RESGUARDO.**

**REPITE**

**Desplegar pantalla de bajas**

**Leer número del resguardo**

**Valida número del resguardo**

**Borra registro de la base de datos.**

**HASTA otra baja = ' '**

**FIN DE BAJA DE RESGUARDOS.**

**PROCESO: REALIZA CONSULTAS DE RESGUARDOS**

**OBJETIVO: SELECCIONAR MOVIMIENTO A REALIZAR DEPENDIENDO DE LA OPCION SELECCIONADA POR EL USUARIO.**

**Proceso : REALIZA CONSULTAS**

**REPITE**

**Desplegar el menú de consultas de resguardos**

**Leer opción**

**CASO de opción igual a:**

**REALIZA CONSULTA POR No. DE INVENTARIO**

**REALIZA CONSULTA POR R.F.C. DEL EMPLEADO RESPONSABLE.**

**REALIZA CONSULTA POR No. DE SERIE**

**REALIZA CONSULTA POR CLAVE DEL ARTICULO**

**REALIZA CONSULTA DE BAJAS DE RESGUARDO**

**FIN DE CONSULTA DE RESGUARDO**

**De otra manera:**

**Desplegar mensaje de error**

**HASTA opción = fin**

**FIN DE ACTUALIZA CATALOGO DE CONSULTAS.**

**PROCESO: REALIZA CONSULTA POR No. DE INVENTARIO**

**OBJETIVO: CONSULTAR UN REGISTRO DE RESGUARDOS POR N MERO DE INVENTARIO.**

**Proceso : REALIZA CONSULTA POR No. DE INVENTARIO  
REPITE**

Desplegar pantalla de consulta

Leer el número de inventario

Valida el número de inventario

Despliega información referente al resguardo

De otra manera :

Desplegar mensaje de error

HASTA otra consulta =

FIN DE CONSULTA POR No. DE INVENTARIO.

**PROCESO: REALIZA CONSULTA POR R.F.C. DEL EMPLEADO.**

**OBJETIVO: CONSULTAR RESGUARDOS DE EMPLEADOS EN LA BASE DE DATOS.**

**Proceso : REALIZA CONSULTA POR R.F.C. DEL EMPLEADO**

**REPITE**

Desplegar pantalla de consulta

Leer rfc del empleado

Valida rfc del empleado

Despliega la información referente al R.F.C.

De otra manera:

Desplegar mensaje de error.

HASTA otra consulta =

FIN DE CONSULTAS DE RESGUARDOS POR EMPLEADO.

**PROCESO:** REALIZA CONSULTA POR CLAVE DEL DEPARTAMENTO

**OBJETIVO:** CONSULTAR REGISTROS DE RESGUARDOS POR CLAVE DEL DEPARTAMENTO.

**Proceso :** REALIZA CONSULTA POR CLAVE DEL DEPARTAMENTO

REPITE

Desplegar pantalla de consulta

Leer el la clave del departamento

Valida la clave del departamento

Despliega información referente a los resguardos

De otra manera :

Desplegar mensaje de error

HASTA otra consulta =

FIN DE CONSULTA POR CLAVE DEL DEPARTAMENTO.

**PROCESO:** REALIZA CONSULTA POR CLAVE DEL ARTICULO

**OBJETIVO:** CONSULTAR RESGUARDOS POR CLAVE DEL ARTICULO DE LA BASE DE DATOS.

**Proceso :** REALIZA CONSULTA POR CLAVE DEL ARTICULO

REPITE

Desplegar pantalla de consulta

Leer clave del articulo

Valida clave del articulo

Despliega la información a los resguardos que contengan está clave del articulo.

De otra manera:

Desplegar mensaje de error.

HASTA otra consulta =

FIN DE CONSULTAS DE RESGUARDOS POR EMPLEADO.

**PROCESO: REALIZA CONSULTA DE BAJAS DE RESGUARDOS**

**OBJETIVO: CONSULTAR UN REGISTRO DE RESGUARDOS POR N MERO DE INVENTARIO.**

**Proceso : REALIZA CONSULTA DE BAJAS PO No. DE INVENTARIO**

**REPITE**

Desplegar pantalla de consulta

Leer el número de inventario

Valida el número de inventario

Despliega información referente a la baja del resguardo.

De otra manera :

Desplegar mensaje de error

HASTA otra consulta =

**FIN DE CONSULTA DE BAJAS DE RESGUARDOS.**

ESTA TESIS NO DEBE  
SALIR DE LA BIBLIOTECA

ANALISIS DE REQUERIMIENTOS

---

**PROCESO : CREAR REPORTES**

**OBJETIVO: EMITIR REPORTES DE LOS RESPECTIVOS CAT LOGOS.**

**Proceso: CREAR REPORTES**

**REPITE**

Desplegar menú de reportes

Leer opción

Caso de opción igual a:

- 1: Genera reporte de catálogos
- 2: Genera reporte general de resguardos
- 3: Genera reporte de departamento (s)
- 4: Genera reporte de empleado (s)
- 5: Genera reporte articulo (s)
- 6: Genera reporte general de inventarios y artículos
- 7: Genera reporte general de bajas de resguardos
- 8: Regresar al menú principal.

De otra manera:

Desplegar mensaje de error

Hasta opción=0

**FIN DE REPORTES.**

**PROCESO : CREAR REPORTES DE LOS CATALOGOS**

**OBJETIVO:** EMITIR REPORTES DE LOS RESPECTIVOS CAT LOGOS.

**Proceso:** CREAR REPORTES DE LOS CATALOGOS

REPITE

Desplegar menú de reportes

Leer opción

Caso de opción igual a:

- 1: Genera reporte numérico de partida
- 2: Genera reporte ordenado por articulo
- 3: Genera reporte numérico por departamento
- 4: Genera reporte ordenado por R.F.C.
- 5: Salir del proceso

regresar al menú principal.

De otra manera:

Desplegar mensaje de error

Hasta opción=0

**FIN DE REPORTES.**

**PROCESO : GENERA REPORTE NUMERICO DE PARTIDAS**

OBJETIVO: EMITIR REPORTE NUMERICO DE PARTIDAS.

Proceso: GENERA REPORTE NUMERICO

Ordena PAR por número de Partida

Imprime

FIN GENERA REPORTE NUMERICO DE PARTIDAS.

**PROCESO : GENERA REPORTE ORDENADO POR ARTICULO**

OBJETIVO: EMITIR REPORTE POR CLAVE DEL ARTICULO.

Proceso: GENERA REPORTE ORDENADO POR ARTICULO

Ordena ART por clave del articulo

Imprime

FIN GENERA REPORTE DE ARTICULOS.

**PROCESO : GENERA REPORTE NUMERICO POR DEPARTAMENTO**

OBJETIVO: EMITIR REPORTE POR CLAVE DEL ARTICULO.

Proceso: GENERA REPORTE ORDENADO POR ARTICULO

Ordena ART por clave del articulo

Imprime

FIN GENERA REPORTE DE ARTICULOS.

**PROCESO : GENERA REPORTE ORDENADO POR RFC DEL EMPLEADO**

OBJETIVO: EMITIR REPORTE ORDENADO POR R.F.C. DEL EMPLEADO

Proceso: GENERA REPORTE ORDENADO POR R.F.C. del empleado

Ordena EMPLEADO por R.F.C.

Imprime

FIN GENERA REPORTE DE EMPLEADOS.

**PROCESO : GENERA REPORTE DE RESGUARDOS**

**OBJETIVO:** EMITIR REPORTE DE TODOS LOS RESGUARDOS ORDENADO POR NO. DE INVENTARIO.

**Proceso:** GENERA REPORTE NUMERICO

Ordena RESGUA por número de inventario.

Imprime

**FIN GENERA REPORTE DE RESGUARDOS.**

**PROCESO : GENERA REPORTE DE RESGUARDOS ORDENADO POR EMPLEADO (S)**

**OBJETIVO:** EMITIR REPORTE ALFABETICO DE RFC DE LOS RESGUARDOS DEL EMPLEADO.

**Proceso:** GENERA REPORTE DE RESGUARDOS ORDENADO POR RFC DEL EMPLEADO

Ordena RESGUA por RFC y Número de inventario

Imprime

**FIN GENERA REPORTE DE EMPLEADOS.**

**PROCESO : GENERA REPORTE DE RESGUARDOS ORDENADO POR DEPARTAMENTO**

**OBJETIVO:** EMITIR REPORTE ORDENADO POR N MERO DEL DEPARTAMENTO

**Proceso:** GENERA REPORTE ORDENADO POR NUMERO DEL DEPARTAMENTO.

Ordena RESGUA por Número de Departamento y Número de inventario.

Imprime

**FIN GENERA REPORTE DE DEPARTAMENTOS.**

**PROCESO : GENERA REPORTE DE RESGUARDOS ORDENADO POR CLAVE DEL ARTICULO.**

**OBJETIVO: EMITIR REPORTE ORDENADO POR R.F.C. DEL EMPLEADO**

**Proceso: GENERA REPORTE ORDENADO POR R.F.C. del empleado**

Ordena RESGUA por R.F.C. del empleado y por Número de Inventario.

Imprime

**FIN GENERA REPORTE DE EMPLEADOS.**

## CAPITULO IV

### DISEÑO DEL SISTEMA

#### 4.1 DISEÑO ESTRUCTURADO

El diseño estructurado es la elaboración de un sistema jerárquico con sus componentes y partes interrelacionadas entre sí de la mejor manera posible.

El diseño estructurado no crea una actividad en el proceso de desarrollo de los sistemas, sino solamente consolida, formaliza y hace visibles actividades y decisiones que ocurren inevitablemente durante la creación del sistema y las aborda como alternativas evaluables.

Resumiendo el diseño estructurado determina cuántas partes tendrá el sistema, que hace cada una, como están organizadas y cuales son las interrelaciones entre ellas para resolver la problemática en cuestión.

Como entrada al diseño del sistema se tienen principalmente los DFD's y DD, obtenidos en la fase anterior, y la salida que se produce es un documento de especificación estructurada.

El objetivo del diseño estructurado es el de desarrollar software a un costo mínimo y con mayor durabilidad, esto implica

crear sistemas flexibles, de alta calidad, eficientes, eficaces, y como una consecuencia natural de la fase de análisis.

La filosofía del diseño estructurado, se basa en el principio de dividir un problema, en pequeños problemas con el fin de facilitar el desarrollo, el mantenimiento y las modificaciones sobre el mismo diseño. Se puede afirmar que el diseño estructurado busca atacar la complejidad de un sistema muy grande mediante el uso de los procedimientos, particionando el sistema en " caja negras " y organizando jerárquicamente el problema.

Una caja negra cumple con las siguientes características:

- Se conocen las entradas
- Las salidas que producirá
- Se conoce su función ( que hace con las entradas para generar las salidas esperadas) y no es necesario conocer el cómo realizar dicha función.

Las ventajas de el uso de " cajas negras ", es que éstas son muy fáciles de construir, su prueba se hace muy sencilla, se pueden corregir fácilmente y son muy entendibles.

Por otra parte la idea de jerarquizar es aún mas antigua que la de particionar ya que todos los sistemas complejos estan organizados en jerarquias de partes a su vez formadas por otras pequeñas partes.

Con las dos ideas anteriores, la de particionar y la de jerarquizar, el diseño estructurado conforma como su principal

herramienta al Diagrama Estructurado, este diagrama ilustra la partición del sistema en módulos ( Cajas negras ) mostrando su jerarquía, organización y comunicación.

Las ventajas de usar un diagrama estructurado es tener una herramienta para el diseño de tipo gráfico, particionable, riguroso pero a la vez flexible, que además sirve como documentación del sistema, y es una gran ayuda al mantenimiento y modificaciones del sistema.

Antes de la elaboración del diagrama es necesario entender los conceptos esenciales que se expondrán a continuación.

## DIAGRAMA ESTRUCTURADO

Un diagrama de estructura está formado por módulos o cajas negras, entendiéndose como módulo a una colección de instrucciones de programa caracterizadas por cuatro atributos:

- Entrada y Salida
- Función
- Mecanismo
- Datos Internos

1. Entrada	Que requiere el módulo
Salida	Que es lo que regresa el módulo

---

2. Función	Que es lo que realiza con su entrada para producir la salida
3. Mecanismo	Como se realiza la función
4. Datos Internos	Datos a los que hace referencia de manera solitaria.

El módulo tiene también otros atributos, como el nombre mediante el cual es referenciado dentro de toda la unidad.

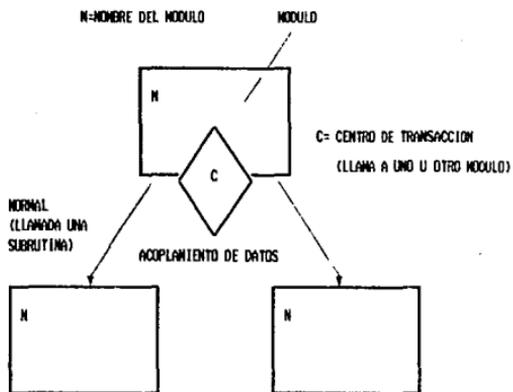
Un módulo se representa gráficamente como una caja rectangular con su nombre en el interior, siendo el nombre, un enunciado breve que describe la función que realiza el módulo.

### CONEXION ENTRE MODULOS

El símbolo que nos permite conectar módulos con módulo es una flecha, esta flecha muestra una llamada a una subrutina y mediante el sentido de la flecha se puede saber cual es el módulo que invoca al otro módulo subordinado. Los datos estan representados por flechas mas pequeñas en las cuales el sentido de éstas indica de donde se realiza la transferencia, si la flechita tiene en un extremo un circulo relleno, indica que se trata de la transferencia de una bandera.

En el caso que la invocación de los módulos hijos sea opcional, esto se representa con un rombo en el módulo padre.

## SIMBOLOGIA DE DIAGRAMAS DE ESTRUCTURA



## 4.2 CUALIDADES DE UN BUEN DISEÑO

Uno de los principios fundamentales del diseño estructurado es el hecho de particionar un gran sistema en módulos manejables; de esta forma es vital que este particionamiento se realice de tal forma que se obtengan módulos que sean lo más independiente posible entre sí, a este criterio se le conoce como " *ACOPLAMIENTO* ", y además se debe perseguir que cada módulo tenga una función bien definida y produzca una salida simple, a este criterio se le conoce como " *COHESION* ". Juntos estos elementos son los factores que sirven para distinguir de un buen diseño a uno malo. En un diseño bueno lo que se pretende es que exista el mínimo de acoplamiento entre módulos y se busca una máxima cohesión entre los elementos de cada módulo.

### 4.2.1 ACOPLAMIENTO

Este término se define como la medida de interdependencia entre módulos, entonces los módulos altamente acoplados están unidos por interconexiones rígidas.

El diagrama de estructura persigue el desacoplamiento ya que al aumentar el acoplamiento el mantenimiento se hace complicado.

Podemos distinguir los siguientes tipos de acoplamiento de menor acoplamiento a mayor son los siguientes:

- Acoplamiento por datos
- Acoplamiento por estampilla
- Acoplamiento por control
- Acoplamiento por área común
- Acoplamiento por contenido

#### 4.2.2 COHESION

La cohesión se define como una medida de fuerza de asociación de la consistencia de los elementos de un módulo, entendiéndose como elemento una instrucción o un grupo de instrucciones, una llamada a otro módulo, es decir un pedazo de código que realiza algún trabajo.

Además la cohesión junto con el concepto de acoplamiento nos permite medir la forma de realización de las particiones.

Un buen diseño se debe caracterizar por módulos que mantengan un alto grado de cohesión entre sus elementos, es decir que estén genuinamente relacionados unos con otros y por otra parte se debe evitar que los elementos de un módulo se relacionen con los de otro para evitar un alto grado de acoplamiento.

Los diferentes niveles de cohesión de mayor a menor son los siguientes:

- Cohesión Funcional
- Cohesión Secuencial
- Cohesión Comunicacional
- Cohesión de Procedimiento
- Cohesión Temporal
- Cohesión Lógica
- Cohesión Coincidental

Los primeros tres niveles son los más aceptables, los demás niveles producen diseños caros de desarrollar y mantener.

### 4.3 METODOLOGIA DEL DISEÑO ESTRUCTURADO.

La metodología del diseño estructurado sirve para transformar la salida del análisis estructurado en la entrada a la implementación, es decir, transforma los DFD's en Diagramas de Estructura. El proceso del diseño es iterativo y descendente (TOP-DOWN). Esto es, los pasos se hacen repetidamente trabajando de arriba hacia abajo hasta definir completamente la estructura modular del sistema, especificando en el proceso las funciones de los módulos, las interrelaciones entre ellos y sus interfases. Al ir definiendo la estructura y las especificaciones de los módulos se logra tener una documentación concurrente que además de haber servido de herramienta del diseño, queda como documentación del sistema.

Primer paso.- Definir la función del módulo más alto nivel, ésta será la función del sistema.

Segundo Paso.- Seleccionar la técnica de modularización, que puede ser

- a) Diseño Centrado en Transformaciones.
- b) Diseño Centrado en Transacciones.

Tercer Paso.- Aplicar la técnica o combinación de técnicas seleccionadas iterativamente hasta llegar a los módulos de más bajo nivel.

a) **DISEÑO CENTRADO EN TRANSFORMACIONES.** Es una de las estrategias principales para diseñar sistemas altamente balanceados, también se le conoce como *análisis de transformaciones*. Para el diseño centrado de transformación se requiere del diagrama de flujo de datos obtenido en la etapa de análisis del sistema, obteniendo como resultado un diagrama de estructura, en el que el módulo superior trabaja con datos altamente procesados, datos lógicos.

La estrategia se puede resumir de la siguiente forma:

- 1.- Obtener el DFD del problema.
- 2.- Identificar la transformación central, esto puede realizarse siguiendo las ramas aferentes y deferentes del DFD, hasta encontrar el punto en que los datos son independientes de los dispositivos de entrada-salida, los procesos entre estos puntos forman la transformación central.
- 3.- Identificar las ramas principales de datos de entrada y salida. Determinar los puntos de máxima abstracción.
- 4.- Diseñar la estructura a partir de la información previa con un módulo para cada rama principal de entrada y un módulo para cada rama principal de salida.
- 5.- Para cada uno de los módulos de entrada, identificar la última transformación necesaria para producir los datos en la forma en que los regresa el módulo. Después identifique la forma de entrada justo antes de la última transformación.

Para módulos de salida, identifique el primer proceso necesario para acercarse a la salida deseada con el formato deseado de resultados.

El Diseño Centrado de Transformaciones es la mejor estrategia para convertir un DFD en un diagrama de estructura, de cualquier forma existe una técnica suplementaria llamada *Diseño Centrado en Transacciones* el cual es conveniente para realizar un diagrama de estructura de un sistema que procesa transacciones.

b) **DISEÑO CENTRADO EN TRANSACCION.** Esta estrategia tiene dos usos principales, primero: permite simplificar un DFD muy complicado en pequeños DFD's, uno por cada transacción que el sistema procese, estos pequeños DFD's son suficientes para ser convertidos en el diagrama de estructura usando el Diseño Centrado por Transformación, segundo: el Diseño Centrado por Transacciones puede ser usado para combinar diagramas de estructura flexible a cambios.

El Diseño Centrado en Transacción tiene el siguiente principio fundamental: separar las múltiples transacciones por tipo y no como un proceso común, en otras palabras se debe encontrar que tan similar o diferentes son los tipos de módulos, donde debe existir un módulo separado el cual es el responsable del proceso de cada tipo de transacción, a este módulo se le conoce como módulo de transacción, existen tantos módulos de transacción como tipos

de transacciones haya en el sistema. Existen otro tipo de módulo el cual determina el tipo de cada transacción del sistema total, el cual es llamado módulo de transacción central.

Un tipo de centro de transacción se encuentra frecuentemente en sistemas en línea, en los cuales se da un menú al usuario, para que seleccione una de varias opciones.

Normalmente es muy fácil reconocer transacciones, centro de transacciones, y burbujas de procesos de transacciones en un DFD simplemente con ver el diagrama, de cualquier forma, el flujo de datos entre burbujas determina su tipo y siguiendo su ruta en varias direcciones dependiendo de su tipo, se puede localizar una transacción encontrando el centro de transacción.

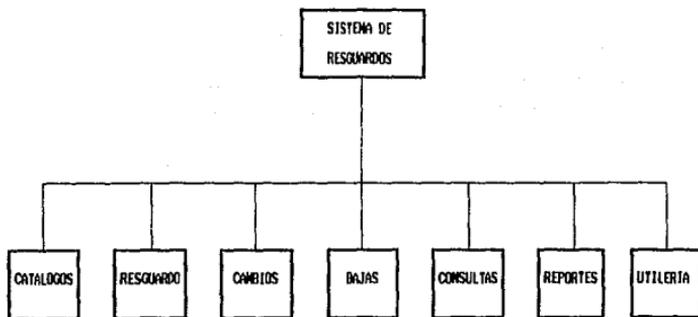
Para nuestro problema en particular, de acuerdo a sus DFD's realizados, en la etapa de análisis estructurado, podemos determinar, que en este caso, es necesario usar métodos de Diseño Centrado por Transacción para realizar el diagrama de estructura.

Para un buen diseño de estas metodologías es necesario aplicar los criterios de cohesión y acoplamiento.

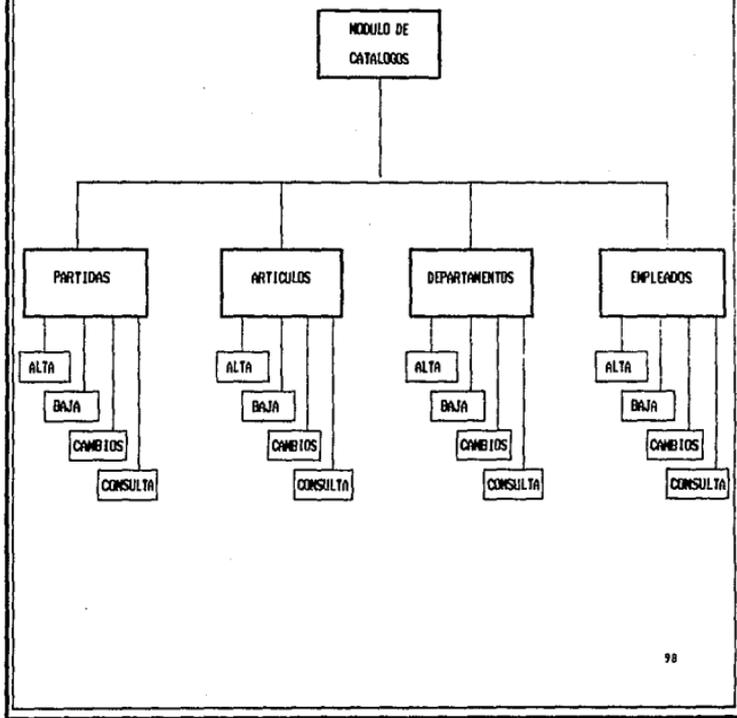
#### **4.4 DIAGRAMAS DE ESTRUCTURAS DEL SISTEMA PROPUESTO**

En las siguientes páginas se muestran algunos Diagramas de Estructura que se obtuvieron del sistema.

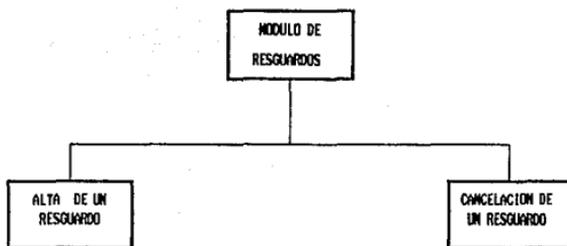
CARTA ESTRUCTURADA DEL SISTEMA DE  
CONTROL DE ACTIVO FIJO  
(RESGUARDOS)



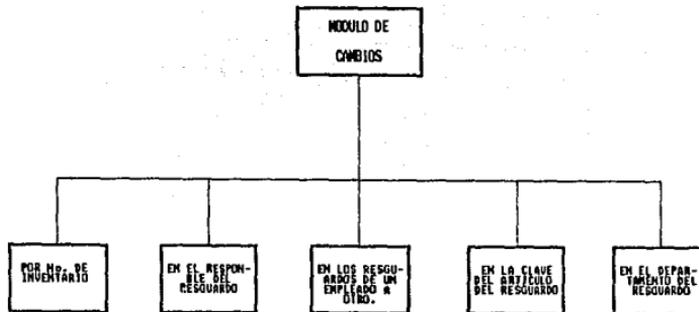
## CARTA ESTRUCTURADA DE LOS CATALOGOS



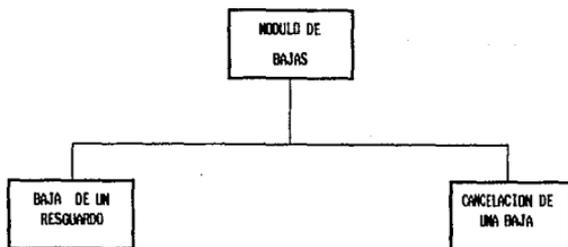
## CARTA ESTRUCTURADA DE RESGUARDOS



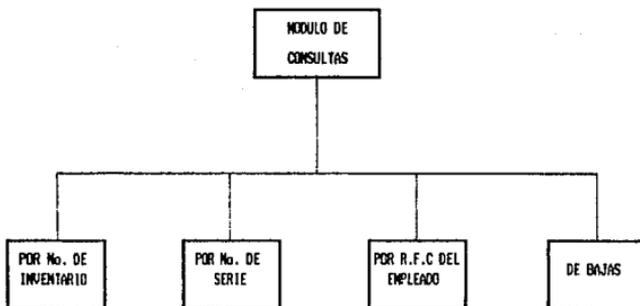
CARTA ESTRUCTURADA DEL MODULO  
DE CAMBIOS A LOS RESGUARDOS



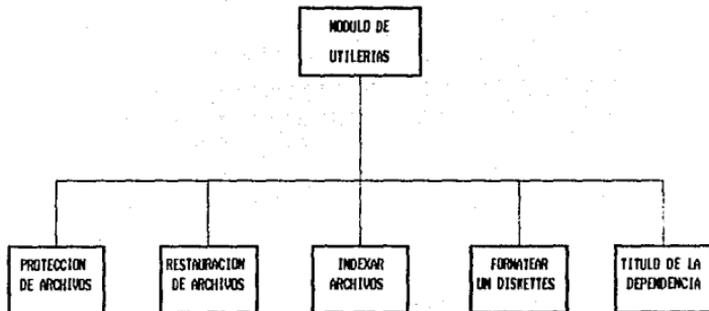
CARTA ESTRUCTURADA DE BAJAS  
DE RESGUARDOS



CARTA ESTRUCTURADA DEL MODULO  
DE CONSULTA DE RESGUARDOS



CARTA ESTRUCTURADA DEL MODULO  
DE UTILERIAS



#### 4.5 CARACTERISTICAS DEL LENGUAJE DE PROGRAMACION.

Una vez definido el problema y haber diseñado los procedimientos que forman el sistema, se procederá a revisar las características de DBase III Plus y de Clipper.

##### VENTAJAS:

- Administra adecuadamente las funciones interrelacionadas de la organización.

- La base de datos contiene la descripción de la información en forma independiente de las funciones de programación lo cual trae como ventaja el liberar a los programadores del manejo de los datos.

- Ampliación sin necesidad de ajustar el sistema.

- Puede identificarse la información repetida mediante la integración de los datos de diferentes aplicaciones en una misma base de datos eliminando redundancia

- Mayor disponibilidad de la información aprovechando la base de datos en el desarrollo de nuevas aplicaciones.

##### DESVENTAJAS:

- DBASE III PLUS es un intérprete por lo que no genera código ejecutable.

Para solucionar este problema se tiene la opción de utilizar CLIPPER.

CLIPPER cuenta con un manejador de base de datos compatible con DBASE III PLUS, además de ser un compilador y ligador genera programas ejecutables.

#### CARACTERISTICAS DE DBASE III PLUS

- El manejador de bases de datos DBASE III, consta de 124 órdenes que permiten manipular la información y fue diseñado para obtener pleno rendimiento de las microcomputadoras de 16 bits y está descrito en lenguaje C.

- DBASE III PLUS puede manejar hasta 1000 millones de registros y hasta 128 campos por registro, puede ordenar varios campos a la vez y trabajar hasta con 10 ficheros simultáneamente.

#### CARACTERISTICAS DE CLIPPER

- Es mucho menor el tiempo de ejecución del sistema que al usar un intérprete.

- Mayor seguridad del código fuente, además que éste es convertidor a lenguaje máquina durante la compilación y ligado, este código de máquina el cual es el menor nivel de código posible en una computadora y este no se puede convertir a código fuente.

- Permite usar hasta 64000 variables activas en memoria, y un máximo de 1024 campos por base de datos.

#### 4.4 DISEÑO DE LA BASE DE DATOS.

Una de las actividades que se deben de desarrollar paralela al diseño estructurado es el diseño de la base de datos.

Antes de proceder al diseño es importante señalar las características generales de la base de datos.

Los primeros esfuerzos en manejo de bases de datos, se iniciaron a fines de los años sesenta, por las necesidades que se tenían de proporcionar acceso efectivo y eficiente a los usuarios y/o programas de aplicaciones, a grandes archivos, los cuales requerían además integrarse en una base. Las estructuras de archivos convencionales, con sus limitaciones características de llaves únicas de acceso secuencial, o un solo tipo de registros, resultaban inadecuadas para ámbitos tan exigentes y ocasionaban tiempo de procesamiento y requerimientos de almacenamiento externo.

Los primeros diseños de sistemas de manejo de bases denominados jerárquicos, invertidas y de red aparecieron a fines de los años sesenta. Surgieron también los sistemas de bases/telecomunicaciones de datos, en los cuales las facilidades de la base y del proceso de datos están muy desarrollados.

Es de primordial importancia, el surgimiento inevitable de aplicaciones de la computadora y de sistemas de información centrados en bases de datos.

Más recientemente se ha iniciado la integración de diversos sistemas computarizados y sus archivos asociados, dentro de sistemas de información más capaces y provechosos, centrados en una base de datos con vías a satisfacer a las comunidades de usuarios.

Han surgido muchos retos debidos a la necesidad de integrar dentro de bases de datos, archivos y datos diversos con mayor efectividad de costos para usuarios con diferentes prioridades y requerimientos. Entre estos retos se tiene la necesidad de estructuras de datos y posibilidades de organización de archivos, independencia de datos entre programas de aplicación y estructuras particulares dadas a la base, habilidad para que los usuarios compartan datos, irredundancia de la información, alto nivel de rendimiento, eficiencia.

#### CONCEPTOS FUNDAMENTALES.

La unidad más pequeña de datos es el campo. Un conjunto de datos elementales constituye un registro lógico o entidad. Un tipo de registro lógico es un registro con una constitución particular de datos elementales; un archivo es una colección de ocurrencias de un mismo tipo de registros finalmente una base de datos es un conjunto de ocurrencias de varios tipos de registros como sus ocurrencias están interrelacionadas mediante relaciones específicas.

Una base de datos reside típicamente en dispositivos externos de almacén de acceso directo.

Un sistema de manejo de bases de datos DBMS, es un sistema de programas o software capaz de dar soporte y manejar un banco de datos integrados.

#### NORMALIZACION.

El proceso de normalización identificalos datos redundantes que pueden existir en la estructura lógica, determina claves únicas necesarias para el acceso a los elementos de datos y ayuda a establecer las relaciones necesarias entre los elementos de datos y ayuda a establecer las relaciones necesarias entre elementos de datos.

La normalización no comprende aspectos de la organización física de la base de datos, únicamente trata con el ámbito del diseño lógico.

Para una base de datos puede definirse tres tipos de las llamadas formas normales, estas formas proporcionan una mejoría en las propiedades de la base de datos respecto a cambios en la misma.

Se dice que una relación se encuentra en primera forma normal si todos sus dominios son simples, es decir si es una tabla bidimensional o plana.

Toda relación normalizada esta automáticamente en primera forma normal. Un proceso adicional de normalización puede tomar una

relación en primera forma normal y separarla en relaciones mas simples de las que se dicen estan en segunda forma normal. Otro proceso mas de normalización puede tomar una relación en segunda forma normal y separarlas en relaciones aún mas simples de las que se dicen que estan en tercera forma normal.

De una manera mas formal se define:

- Una relación esta en primera forma normal, si todos sus atributos son atómicos, definiendo como atributo atómico aquel que se da cuando existente una solo instancia.

- Una relación se encuentra en segunda forma normal si esta en primera forma normal y todos sus atributos no llave dependen de la llave completa.

- Una relación se encuentra en tercera forma normal si esta en segunda forma normal y no presenta transitividades bajo un conjunto de dependencias funcionales.

El proceso de normalización se puede resumir de la siguiente manera:

1.- Tomar la relación en primera forma normal y remover todas las dependencias funcionales incompletas mediante la aplicación de proyecciones apropiadas. Esto produce un conjunto de relaciones de segunda forma normal si cada llave candidata consiste de un solo atributo, entonces la relación en primera forma normal se encuentra automáticamente en segunda forma normal también.

2.- Aplicar proyecciones a las relaciones en segunda forma normal, de tal manera que las relaciones que resulten cada determinante sea una llave candidata o un atributo primo (un atributo primo es uno que sea miembro del conjunto de atributos que constituyen a una llave candidata de atributos múltiples) o en otras palabras, tal que los atributos no primos sean mutuamente independientes.

De acuerdo a toda la teoría de base de datos resumida en las líneas anteriores se ha podido definir un proceso de diseño de la base de datos para que estas cumplan con las características óptimas de un buen diseño; este proceso se puede desglosar en tres etapas:

- Diseño Conceptual
- Diseño Lógico
- Diseño Físico

#### DISEÑO CONCEPTUAL.

El objetivo del diseño conceptual de la base de datos es obtener un modelo de las necesidades de información, independientemente de las características del sistema de cómputo.

El modelo conceptual de una base de datos es un enunciado estructurado de las entidades que deben almacenarse y de las relaciones que existen entre ellas.

**DISEÑO LÓGICO.**

En el diseño lógico de la base de datos se presentan las entidades y las relaciones del modelo conceptual, utilizando las estructuras de algún sistema de administración de datos o las estructuras de un sistema de archivos.

Las relaciones se pueden expresar en un modelo E-R (entidad-relacionamiento) el cual es un modelo de datos no implementado en alguna máquina y sirve para el diseño de sistemas. Este modelo esta compuesto por tablas y gráficas que utilizan la siguiente nomenclatura:

- a) Un rectángulo representa a una entidad.
- b) Los círculos representan a los atributos, los cuales estan ligados a sus entidades mediante líneas.
- c) Los rombos representan a las relaciones entre entidades.

**DISEÑO FÍSICO.**

El diseño físico de la base de datos se hacen estudios de los tiempos de acceso para ver si corresponden a los requerimientos de tiempo de respuesta especificados por el usuario. Para mejorar los tiempos de acceso se pueden utilizar estructuras de mas bajo nivel que las que se utilizan a nivel lógico, o bien estructuras lógicas alternativas con redundancia.

Las dos primeras etapas en el diseño de una base de datos son siempre necesarias. En ocasiones, no se precisa del diseño físico,

dependiendo del tipo de sistema de administración de datos o del tipo de sistema de archivos con que se cuente.

Para nuestro problema particular se desarrolló el siguiente diseño:

**DISEÑO CONCEPTUAL.**

De acuerdo a lo especificado en la fase de análisis del sistema se necesita almacenar la siguiente información:

- Es necesario almacenar la información referente a los catálogos

1) PARTIDAS. Es necesario almacenar la clave de la partida y su nombre.

2) ARTICULOS. Es necesario almacenar la clave del artículo el grupo y partida a la que pertenece, así como su nombre.

3) DEPARTAMENTOS. Es necesario almacenar la clave del departamento y su nombre.

4) EMPLEADOS. Es indispensable almacenar el Registro Federal de Contribuyentes (R.F.C.) el nombre, así como el departamento al cual está asignado el empleado.

5) RESGUARDOS. Es necesario almacenar toda la información referente a la alta de un resguardo, la relación que guarda con los diferentes catálogos es:

- Para el de artículo es su clave
- Para el de empleados es el R.F.C.

6) BARESG. Es necesario almacenar la información referente a las bajas de los resguardos, ya que es necesario tener la historia del resguardo para cualquier duda o aclaración con la Dirección General de Patrimonio. La relación que guarda con los diferentes catálogos es de acuerdo a sus respectiva clave.

#### DISEÑO LOGICO.

Para el sistema de Control de Artículos de Activo Fijo Pc (RESGUARDOS) se utiliza el administrador de DBASE III PLUS el cual nos permite realizar selecciones, join, totalizar registros, actualizar registros de manera automática entre otras cosas.

La definición lógica de la base de datos para el sistema de RESGUARDOS requiere de los siguientes archivos:

#### ARCHIVO DE PARTIDAS.

Contiene la información de las partidas.

Estructura:

PAR(CLAVE\_PAR, NOM\_PAR)

#### ARCHIVO DE ARTICULOS.

Contiene la información de los artículos.

Estructura:

ART(CLAVE\_ART, NOM\_ART1, NOM\_ART2, GRUPO, ACLAVE\_PAR)

**ARCHIVO DE DEPARTAMENTOS.**

Contiene la información de los departamentos.

Estructura:

DEPTO(CLAVE\_DEP, NOM\_DEPTO)

**ARCHIVO DE EMPLEADOS.**

Contiene la información de los empleados.

Estructura:

EMPLEADO(RFC, NOMBRE, ACLAVE\_DEP)

**ARCHIVO DE RESGUARDOS.**

Contiene la información referente a los resguardos asignados a los diferentes empleados.

Estructura:

RESGUA(RCLAVE\_INV, RCLAVE\_ART, RCANTIDAD, RMARCA, RMODELO,  
RPRECIO, RRFC, REF, RCLAVE\_DEP, NUM\_SERIE, RTIPOADO,  
FECHAADO, RLUGAR)

**ARCHIVO DE BAJA DE RESGUARDOS**

Contiene la información referente a los resguardos asignados a los diferentes empleados.

**Estructura:**

BARESG(RCLAVE\_INV, RCLAVE\_ART, RCANTIDAD, RMARCA, RMODELO, RPRECIO, RRFC, REF, RCLAVE\_DEP, NUM\_SERIE, RTIPOADO, FECHAADO, RLUGAR, FOLIO1, FECHABAJA)

**ARCHIVO DE SUBDEPARTAMENTOS.**

Contiene la información referente a los sub-departamentos.

**Estructura:**

SUBDEPTO(TIPO, NOMBRETIPO)

**ARCHIVO DE TIPO DE ADQUISICIONES.**

Contiene la información referente a la tabla de los tipos de adquisiciones de nuestra Universidad.

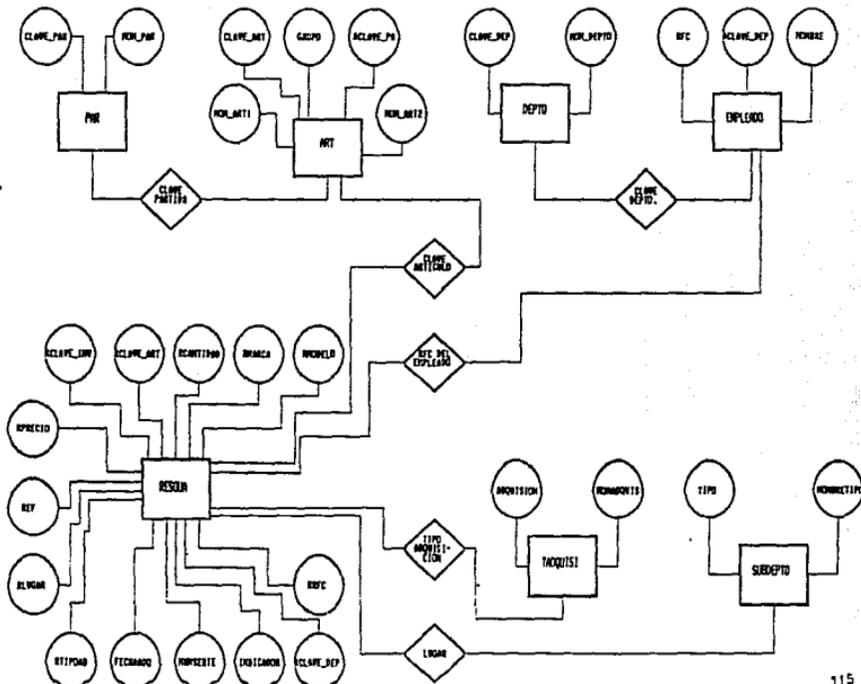
**Estructura:**

TADQUISI(ADQUISICION, NOMADQUIS)

A continuación se presenta el diagrama del modelo ENTER-RELACION DEL SISTEMA DE CONTROL DE ARTICULOS DE ACTIVO FIJO.

MODELO DE ENTER - RELACION

SISTEMA DE CONTROL DE ARTICULOS DE ACTIVO FIJO



**DISEÑO FISICO.**

En este problema no se tiene requerimientos que demanden un acceso a la base de datos de determinado tiempo, como sucede en sistemas de tiempo real.

## CAPITULO V

### IMPLEMENTACION

Durante la fase de Implementación se realizará la codificación y la documentación de los programas especificados por el diseño del sistema.

#### 5.1 CODIFICACION.

Consiste en la traducción de las especificaciones de proceso de cada módulo en instrucciones ejecutables por un lenguaje de programación.

Se pueden definir dos tipos de codificación:

- *BOTTOM-UP O ASCENDENTE*
- *TOP-DOWN O DESCENDENTE*

**CODIFICACION ASCENDENTE.** Esta codificación permite atender inicialmente los módulos de los niveles jerárquicos mas bajos, ya que estos contienen las operaciones mas primitivas del programa y permite establecer la base para de ahí construir las operaciones mas complejas.

**CODIFICACION DESCENDENTE.** De manera contraria opera la codificación descendente, es decir que primero se codificarán los niveles más altos de la jerarquía y así sucesivamente hasta llegar a los niveles inferiores.

El Sistema para el Control de Artículos de Activo Fijo desarrollado se codificará usando una técnica de codificación TOP-DOWN de tal forma que primero se codificará el menú principal, después los menús siguientes en la jerarquía y después los procesos que son llamados por estos.

Por otra parte la codificación se utilizaron las miniespecificaciones obtenidas en la fase de análisis y el diseño realizado del sistema.

En cuanto a lo referente a la implementación de la base de datos utilizada (DBASEIII PLUS), posee varias instrucciones para el manejo de *base de datos relacional* como son la creación y modificación de estructura de la base de datos, sin tener que modificar los programas de aplicación.

## 5.2 DOCUMENTACION.

Es necesario que dentro del código fuente se presente cierta documentación explicando que es lo que se esta realizando a manera de comentario, de tal forma que se facilite el proceso de mantenimiento.

El SISTEMA PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO para PC como parte de su documentación presenta al principio de cada uno de los programas fuentes, datos como son:

- Nombre del Programa
- Función que realiza
- Fecha de elaboración
- Nombre de la persona que lo elaboro.

Por otra parte, a lo largo del código se encuentran algunos comentarios que indican la función de alguna sección con el fin de establecer la mayor claridad posible, además de que el código se presenta sangrado para indicar principio y fin de bloques.

A continuación se nombran los programas fuentes que integran el SISTEMA PARA EL CONTROL DE ARTICULOS DE ACTIVO FIJO, y se anexa carpeta con el contenido de dichos programas.

PROGRAMAS FUENTES DEL  
SISTEMA DE CONTROL DE ARTICULOS  
DE ACTIVO FIJO

INVENTAR.PRG

CATALOGO.PRG

PARTIDAS.PRG

ALTAPAR.PRG  
BAJAPAR.PRG  
CONSAPR.PRG  
CAMBPAR.PRG

ARTICULO.PRG

ALTARTI.PRG  
BAJARTI.PRG  
CONSARTI.PRG  
CAMBARTI.PRG

DEPTO.PRG

ALTADP.PRG  
BAJADP.PRG  
CONSADP.PRG  
CAMSDP.PRG

EMPLEADO.PRG

ALTADMP.PRG  
BAJADMP.PRG  
CONSADMP.PRG  
CAMDMP.PRG

RESQUAR.PRG

ALTARESQ.PRG

CANCELAR.PRG

PROGRAMAS FUENTES DEL  
SISTEMA DE CONTROL DE ARTICULOS  
DE ACTIVO FIJO

INVENTAR. PRG

CAMBIO5. PRG

CAMBIO51. PRG  
CAMBIO52. PRG  
CAMBIO53. PRG  
CAMBIO54. PRG  
CAMBIO55. PRG

BAJAS. PRG

BAJARES0. PRG  
BAJACHNC. PRG

CONSULTA. PRG

CONSINV. PRG  
CONSERIE. PRG  
CONSRFC. PRG  
CONSRJA. PRG

PROGRAMAS FUENTES DEL  
SISTEMA DE CONTROL DE ARTICULOS  
DE ACTIVO FIJO

INVENTAR. PRG

REPORTES. PRG

RECART. PRG  
RECART. PRG  
RECART. PRG  
RECART. PRG

REDETA. PRG  
REDETA. PRG  
REDETA. PRG  
REDETA. PRG  
REDETA. PRG  
REDETA. PRG  
REDETA. PRG

REBAJAS. PRG  
REBAJAS. PRG  
REBAJAS. PRG

UTILERIA. PRG

BACK\_UP. PRG  
REST\_ORE. PRG  
INDEXAR. PRG  
FORMAT. PRG  
TITULO. PRG

## CAPITULO VI

### INTEGRACION Y PRUEBAS

#### 6.1 INTEGRACION.

La finalidad de esta etapa, es la integración funcional de los módulos de un programa de computadora, ajustándolos a las particularidades de cada sistema de cómputo.

La fase de integración esta constituida por dos aspectos:

- La manera de como se van a combinar los módulos para formar programas.
- El diseño de pruebas, la cual me permite identificar errores de codificación.

Podemos definir los siguientes tipos de integración:

##### 6.1.1 INTEGRACION NO INCREMENTAL.

Primero, se debe escribir y probar cada módulo por separado, que es a lo que se llama "UNIT-TEST".

Inmediatamente después reunir y probar todas las unidades agrupadas en subsistemas (subsystem-test).

Después se combinan todos los subsistemas de tal forma que se integre todo al sistema.

Finalmente se realiza la prueba de todo el sistema.

El problema que presenta esta forma de integración es que las interfases mas cruciales de sistemas son probadas mas tarde en el proyecto, cuando hay muy poco tiempo de corregir los errores que se pudieran encontrar, ya que los errores emergen hasta el término del proceso mismo.

#### 6.1.2 INTEGRACION INCREMENTAL.

Esta forma de integración permite detectar con relativa facilidad los errores de programación relacionados con la intercomunicación entre módulos, ya que al integrarse un nuevo módulo ya probado individualmente a un conjunto que ya también fue validado, la presencia de algún error podrá atribuirse en primera instancia a la interconexión modular, además de permitir la identificación de errores en los módulos del sistema adicionados más recientemente.

La implementación incremental se puede describir a través del siguiente algoritmo:

REPETIR HASTA que el sistema este completo

    Probar un módulo.

    Adicionar ese módulo a la combinación existente.

    Probar la nueva combinación.

FIN REPITE.

Liberar el sistema

Este tipo de integración da mas flexibilidad en cuanto al uso de recursos ya que puede llevar a cabo muchos caminos.

Podemos definir tres formas de integración incremental:

- a) INTEGRACION TOP-DOWN INCREMENTAL
- b) INTEGRACION BOTTOM-UP INCREMENTAL
- c) INTEGRACION SANDWICH INCREMENTAL

a) INTEGRACION TOP-DOWN INCREMENTAL.

Este tipo de integración consiste en probar el módulo que se encuentra en el nivel superior del diagrama de estructura de cada módulo subordinado, será simulado por un módulo falso cuya función es rudimentaria (no hace nada, solo despliega letreros), y se usa en lugar del módulo subordinado real que aún no ha sido integrado. Cada módulo subordinado real que aún no ha sido integrado. Cada módulo falso será mas tarde reemplazado por un módulo real hasta llegar a la parte inferior del sistema.

b) INTEGRACION BOTTOM-UP INCREMENTAL.

La integración bottom-up se realiza de manera inversa a la top-down, se integra un módulo de los inferiores y se van probando los módulos en forma piramidal hasta llegar al de nivel superior.

La ventaja de este tipo de integración es que por la dirección en que se va probando los módulos, primero se prueban las entradas y salidas físicas.

c) INTEGRACION SANDWICH INCREMENTAL.

Esta forma de integración se caracteriza porque combina las ventajas de los métodos de integración bottom-up y top-down. Primero se integra la mitad del sistema partiendo de la parte superior usando el método top-down y después se integra la otra mitad a través del método bottom-up. Esta técnica presenta la desventaja de que no es posible traslapar las etapas de diseño e integración y pruebas ya que no se puede realizar la integración hasta haber finalizado el diseño total del sistema.

El sistema de Control Artículos de Activo Fijo se integró mediante la técnica de integración incremental bottom-up, pero considerando como módulos de mas alto nivel los de nivel uno, es decir para el módulo de catálogos, primero se probaron por separado los módulos inferiores correspondientes al módulo de catálogo.

De la misma manera como se integró el módulo de catálogo se integraron al sistema los seis módulos restantes agrupando al final todos para ser llamados por "INVENTAR", esto se pudo realizar de esta manera dado que la comunicación entre los modulos del sistema se da de izquierda a derecha, por lo que si el módulo cinco necesita datos o archivos creados por el módulo dos, esto ya existían sin tener la necesidad de generarlos de manera externa.

La única desventaja que se presentó es que como se probaron los módulos de abajo hacia arriba, en el caso de existir parámetros

o variables globales requeridas por el módulo en prueba, estos se debían de generar para la prueba y cuando se realizaba la integración se debía de eliminar las declaraciones o instrucciones que no habían servido para realizar esa generación de datos externos, pero esto se presenta contra la ventaja de que cuando se integra el módulo superior inmediato los datos generados por el módulo inferior pueden ser usados tal cual por el módulo prueba.

## 6.2 PRUEBAS.

La etapa de pruebas del sistema consiste en la realización de un procedimiento que nos permita conocer si el sistema cumple con su objetivo, en situaciones diferentes, de acuerdo con lo especificado por el usuario.

Las pruebas se definen como un ejercicio del sistema con el fin de encontrar los errores que pudieran existir en éste. Esta fase del sistema va muy ligada a la integración, por lo que las dos etapas constituyen una sola. Primero se realizan pruebas de los módulos por separado, y después de realizarse la integración se deberá probar el sistema como tal.

La fase de pruebas al sistema es crucial y debe de ser realista y tratar de cubrir muy bien todas las condiciones que se puedan presentar durante la operación del sistema.

- PRUEBAS DE IMPLANTACION. Este tipo de pruebas deben de

aplicarse a cada versión del sistema y tienen como objetivo encontrar errores de especificación, concepto y diseño en el ambiente real (equipo, sistema operativo, etc) en donde los programas serán finalmente instalados.

- PRUEBAS DEL SISTEMA. Las pruebas del sistema son realizadas cuando se ha concluido con la integración del sistema. Este tipo de pruebas deben de cubrir cada una de las facetas del sistema en producciones internas, y tiene como objetivo probar la compatibilidad del sistema con la documentación del usuario.

- PRUEBAS DE ACEPTACION. Este tipo de pruebas, requiere de la presencia del usuario final por lo que la preparación de estas pruebas quedarán completamente en manos del usuario.

Para la realización de pruebas de definen dos técnicas:

- Pruebas de Caja Negra. Esta técnica indica la realización de pruebas para determinadas entradas al sistema analizando si las salidas correspondientes a dichas entradas son las adecuadas.

Como se podrá observar, en la aplicación de esta técnica no interviene de ninguna manera con el código interno del sistema, es decir que los casos de prueba se realizan sin importar la forma en que fue desarrollado el sistema.

Dado que las posibles entradas a un sistema por lo general son infinitas, el hecho de observar el comportamiento del sistema a todas las entradas repercutiría en un aumento en el costo del producto además de realizarse una gran inversión de tiempo, es por esto que se recomienda para la realización de los casos de prueba, agrupar las entradas que cumplan con características similares. Estos grupos deberán cubrir entradas típicas, casos frontera, datos válidos e inválidos.

La selección de casos de prueba es un proceso muy minucioso y muchas veces depende de la habilidad del analista para que la agrupación de las entradas sea la más acertada posible, ya que se deberá partir de que si una de las entrada del grupo produce una salida errónea todas las entradas de dicho grupo serán, también procesadas incorrectamente por el programa.

**Pruebas de Caja Blanca.** Esta técnica de pruebas, considera el código interno del sistema. Los casos de pruebas realizados a partir del diseño y código de los programas del sistema.

Los casos de prueba diseñados deben permitir constatar que el código satisface con las especificaciones del diseño.

Para elaborar los casos de prueba es importante la conceptualización de un programa de cómputo como una secuencia de rutas, donde la ruta seleccionada en un caso dado dependa de los datos de entrada.

Es claro que el número de rutas en un programa es infinito, por lo que no es factible el desarrollo de cada caso de prueba para cada ruta, es por esto que no se recomienda la realización de casos de prueba donde se hagan llamadas a todas las posibles combinaciones de estado en las condiciones de cada punto de decisión del programa dado que una porción significativa de los errores en el código sucede en los puntos de decisión.

## **CAPITULO VII**

### **MANTENIMIENTO**

La última etapa de la vida de un sistema es conocida como mantenimiento. El principal objetivo del mantenimiento es que la vida útil del sistema se alargue, sin embargo, este objetivo generalmente nunca se cumple o sólo parcialmente. Algunos de los factores que influyen en este problema son los siguientes:

- Insuficiencia o falta de documentación.
- Incompatibilidad existente entre la documentación y el diseño.
- Dificulta al tratar de entender y sobre todo de modificar el sistema.

En realidad se puede afirmar que la realización de las fases de la vida del sistema anteriores a la fase de mantenimiento son de suma importancia, ya que el hecho de tratar de realizar modificaciones razonables al sistema después de liberado es mas frustrante y costoso que si estas se realizaran a través del desarrollo de éste.

El mantenimiento de un sistema es el proceso más caro realizado durante el ciclo de vida de un sistema.

Dado el gran costo de esta fase, se considera muy importante la realización de sistemas funcionalmente modulares, como se definió en el capítulo de diseño, y el uso de técnicas que nos permitan alargar la vida del sistema al realizar un mejor mantenimiento, estas técnicas se dividen en dos grupos:

- *TECNICAS DE TIPO ADMINISTRATIVO.* Aquí es donde se involucra el concepto de Administración de la Configuración, la cual consiste en la realización de un seguimiento íntegro de un sistema para acumular toda la historia del sistema en una base de datos (Bitácora del Sistema). Esta base de datos deberá proporcionar los cambios efectuados en cierto periodo, los recursos de los que hace uso, y una biblioteca de programas.

- *TECNICAS DE TIPO TECNICO.* Estas técnicas están enfocadas al uso de gente experta para realizar el mantenimiento y el desarrollo de los sistemas considerando hasta donde sea posible el futuro mantenimiento.

Dentro del mantenimiento de un sistema podemos definir la existencia de los siguientes tipos de mantenimiento:

- **MANTENIMIENTO CORRECTIVO.** Consiste simplemente en la corrección de errores que se presentan en el sistema.
- **MANTENIMIENTO PERFECTIVO.** Consiste en enriquecer al sistema con las necesidades del usuario.
- **MANTENIMIENTO ADAPTIVO.** Es el que se refiere a cambios en el medio ambiente del sistema como son el Sistema Operativo, Hardware, Bases de datos, Clipper, etc.

En cuanto a los tipos de mantenimiento definido se tienen los siguientes grados de incidencia durante la vida de un sistema:

- <b>MANTENIMIENTO CORRECTIVO</b>	<b>20%</b>
- <b>MANTENIMIENTO ADAPTIVO</b>	<b>25%</b>
- <b>MANTENIMIENTO PERFECTIVO</b>	<b>55%</b>

Durante todo el desarrollo del sistema de Control de Activo Fijo para microcomputadora, se consideraron los posibles problemas que pudieran surgir relacionados con el mantenimiento del sistema, es por esto que se enfatizó en la realización de un sistema modular.

En lo referente al mantenimiento adaptivo del sistema, éste fué desarrollado para ser utilizado bajo sistema operativo MSDOS o PCDOS versión 3.0 o mayor, esto es dado por las características de DbaseIII y Clipper, por otro lado en el caso de existir cambios en cuanto al número de versión del sistema operativo, los cambios

**MANTENIMIENTO**

---

a realizar en el sistema serian minimos, en el remoto caso de que existieran, dado que las instrucciones utilizadas en los programas de aplicaciones son muy generales.

## CONCLUSIONES

Una vez implementado el sistema se requiere evaluarlo para determinar si se alcanzaron los objetivos planteados.

Se puede establecer que se cumplieron dichos objetivos satisfactoriamente, ya que el " Desarrollo de un Sistema para el Control de Artículos de Activo Fijo (Resguardos), permite llevar un control efectivo de todos los posibles bienes inventariables en dependencias de la U.N.A.M.

El sistema esta implementado en Dbase III PLUS compilado con Clipper, con sus programas realizados en forma modular, para que puedan ser modificado en el momento que sea requerido. Además, se proporciona un Manual de Usuario para que se pueda instalar y utilizar eficientemente el sistema de Control de Artículos de Activo Fijo.

Por lo tanto, podemos evaluar nuestro sistema en los siguiente puntos:

UTIL. Satisface plenamente las necesidades de los usuarios, ya que les permite tener un control eficiente de los bienes inventariables, además que esta desarrollado para que lo operen personas sin experiencia computacional.

**BAJO COSTO.** Solo requiere de una microcomputadora PC compatible y su impresora, aunado que todas las dependencias administrativas de la U.N.A.M. tienen al menos una microcomputadora PC compatible para desarrollar sus respectivas funciones administrativas, por lo tanto podemos decir que tiene un muy bajo costo.

**MODULAR.** El sistema se puede modificar fácilmente para poder satisfacer nuevos requerimientos, pudiéndose realizar un efectivo mantenimiento.

**COMPATIBLE.** El sistema es compatible, puede utilizarse en diferentes marcas de microcomputadoras PC y por lo tanto el sistema a su vez es transportable.

Actualmente el Desarrollo de un Sistema para el Control de Artículos de Activo Fijo (Resguardos) se implanto en aproximadamente 70 dependencias universitarias.

Por lo mencionado anteriormente se puede concluir que los resultados son satisfactorios, ya que se alcanzaron los objetivos propuestos para esta tesis.

## BIBLIOGRAFIA

INGENIERIA DE SOFTWARE

UN ENFOQUE PRACTICO

ROGER S. PRESSMAN

EDIT. MC GRAW HILL.

INGENIERIA DE SOFTWARE

RICHARD FAIRLEY

EDIT. MC GRAW HILL.

APLIQUE EL DBASE III PLUS

EDWARD JONES

EDIT. OSBORNE / MC GRAW HILL.

NOTAS DE LAS MATERIAS:

INGENIERIA DE PROGRAMACION

BASES DE DATOS

INGENIERIA DE SOFTWARE

IAN SOMMERVILLE

EDIT. ADISON-WENSLEY IBEROAMERICANA

**METODOLOGIAS DE DESARROLLO**

**ANTONIO LOPEZ FUENSALIDA**

**EDIT. MACROBIT.**

**MANUAL DEL SISTEMA DE**

**CONTROL DE ACTIVO FIJO UNAM**

**DIRECCION GENERAL DE PATRIMONIO**

**EDITADO POR: DIRECCION GENERAL DE PUBLICACIONES**

**AN INTRODUCTION TO DATABASE SYSTEMS**

**C. J. DATE.**

**EDIT. ADISSON WENSLEY**