



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE AUTOABASTECIMIENTO FARMACEUTICO A HOSPITALES

T E S I S
QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION
PRESENTAN:
ELIZABETH CALIXTO CORONA
ALFONSO CRUZ CASTILLO
ELIA GUTIERREZ BOCANEGRA
JOSE SAUL MARTINEZ MENDEZ
ARTURO SANDOVAL ARRIAGA

DIRECTOR DE LA TESIS: M. EN I. JUAN CARLOS ROA BEIZA



MEXICO, D. F.

1998

TESIS CON FALLA DE ORIGEN

264083



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 esta 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.

*Agradecemos a la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería por la formación adquirida, a nuestros profesores por su enseñanza y dedicación, en especial a nuestro director de tesis Juan Carlos Roa Beiza por su dirección, apoyo y amistad.*

A Dios. Bendigo y alabo a Dios por permitirme terminar una etapa más en mi vida y poder disfrutarla con mis seres queridos y brindarles una pequeña cosecha de su amor.

A mis padres, por su gran apoyo durante toda mi vida. A mi padre, por enseñarme los grandes valores del éxito y mantener una visión ante la vida. A mi madre, por su fortaleza, ejemplo y motivación para lograr mis sueños. A los dos les dedico esta tesis, ¡Alegría mía!.

A mis hermanos, por el amor y amistad que les tengo espero que lo que hasta hoy haya realizado sea para ellos un ejemplo de vida y enseñanza, como lo han sido ellos en mi vida.

A mi esposo, por ser la persona a quien amo tanto y que vino a completar mi vida por su gran apoyo, amor y comprensión.

Elizabeth

--

*A Zoé y a mi mamá*

*gracias.*

*Alfonso*

*A mi madre, por su gran paciencia y apoyo, por sus consejos, y por ser mi fortaleza en todos los momentos de mi vida.*

*A mi padre (†), por sus sacrificios para que terminara mi carrera y por enseñarme que sólo con trabajo, constancia y cariño se pueden realizar nuestros sueños.*

*A mis hermanos: Octavio, Martha, Carolina y Olivia, por su ayuda, apoyo y comprensión para que concluyera este proyecto. Y sobre todo por ser un aliciente para seguir adelante.*

*A mis amigos, por sus consejos y palabras de aliento.*

*A mis compañeros de tesis: Elizabeth, Saúl, Arturo y Alfonso*

*"Gracias"*

*Elia*

*A mis padres, mis hermanos y toda mi familia. Muchas gracias por su apoyo y comprensión.*

*A mis amigos, por sus consejos y ayuda.*

*Saúl*

*A mis padres, por todo*

*A.....*

*Clo, por todo tu apoyo*

*La memoria de Lulú*

*Poncho, Dina, Kare, Roy I y Roy II, por un reencuentro*

*Bebito y cachetes, por su cariño*

*Tía Julieta, por ser alguien mas que una tía*

*Gonzalo, por tu amistad*

*Arturo*

## Capítulo I

**SISTEMA DE AUTOABASTECIMIENTO FARMACÉUTICO A HOSPITALES**

|  |    |
|--|----|
| 1.1. Antecedentes.....   | 1  |
| 1.1.1. Historia general.....   | 1  |
| 1.1.2. Historia en nuestro país.....   | 3  |
| 1.1.3. Perfil de la industria.....   | 7  |
| 1.1.4. Asistencia de pacientes y características de un Hospital.....   | 10 |
| 1.1.5. Características de un farmacéutico.....   | 14 |
| 1.2. Especificaciones para la elaboración de productos.....  | 16 |
| 1.2.1. Control de calidad.....   | 16 |
| 1.2.1.1. Organización del control y seguridad de la calidad.....   | 17 |
| 1.2.1.2. Control de calidad total.....   | 18 |
| 1.2.1.3. Fases de control de la calidad de productos farmacéuticos.....  | 19 |
| 1.2.2. Control ambiental.....  | 21 |
| 1.2.3. Práctica actual de buena fabricación en la elaboración, procesamiento,<br>envasado o almacenamiento de drogas humanas y veterinarias..... | 23 |
| 1.3. Lineamientos para la realización de servicios.....  | 30 |
| 1.3.1. Función de servicio.....  | 30 |
| 1.3.2. Requerimientos para un servicio.....  | 44 |
| 1.4. Requerimientos para la realización de un pedido.....  | 48 |
| 1.4.1. Procedimientos para la realización de un pedido.....  | 52 |
| 1.4.2. El efecto de la normalización y la catalogación para la elaboración de un<br>pedido.....  | 61 |
| 1.5. Características, ventajas y desventajas de Visual Basic.....  | 65 |
| 1.5.1. Introducción a Visual Basic.....  | 65 |
| 1.5.2. Requerimientos de hardware.....   | 66 |
| 1.5.3. Elaboración de programas orientados a eventos.....  | 67 |
| 1.5.4. Terminología de Visual Basic.....   | 68 |
| 1.5.5. Acceso a la barra de menús.....   | 69 |
| 1.5.6. Creación de controles con la caja de herramientas.....  | 75 |
| 1.5.7. Escritura del código funcional.....   | 79 |
| 1.5.8. Utilización del ambiente de trabajo Visual Basic.....   | 82 |

|        |  |     |
|--------|--|-----|
| 1.5.9. | Alcance de las variables locales y globales.....             | 85  |
| 1.6.   | Características, ventajas y desventajas de Access 7.0.....   | 88  |
| 1.6.1. | Introducción a Access.....                                   | 88  |
| 1.6.2. | Diseño de una base de datos.....                             | 89  |
| 1.6.3. | Objetos de Access.....                                       | 92  |
| 1.6.4. | Herramientas para compartir información al mismo tiempo..... | 104 |
| 1.6.5. | Especificaciones de uso de Access.....                       | 107 |

## Capítulo II

### PLANTEAMIENTO DEL PROBLEMA

|          |  |     |
|----------|--|-----|
| 2.1.     | Definición del problema.....   | 109 |
| 2.2.     | Descripción de objetivos.....  | 118 |
| 2.2.1.   | Objetivos a cubrir del sistema.....  | 118 |
| 2.3.     | Conceptos de bases de datos relacionales.....  | 121 |
| 2.3.1.   | Introducción.....  | 121 |
| 2.3.2.   | Componentes de un RDBMS.....   | 121 |
| 2.3.3.   | Objetivos de los sistemas de bases de datos.....   | 123 |
| 2.3.4.   | Importancia de la planeación de las bases de datos.....  | 125 |
| 2.3.5.   | <i>Normalización</i> .....   | 129 |
| 2.4.     | Metodologías del diseño.....   | 154 |
| 2.4.1.   | Las diferentes tendencias de las metodologías de desarrollo.....   | 158 |
| 2.4.2.   | Estadísticos y clasificación de metodologías.....  | 167 |
| 2.4.3.   | La Metodología de análisis y diseño estructurado de Yourdon.....   | 180 |
| 2.5.     | Análisis del sistema.....  | 180 |
| 2.5.1.   | <i>Situación actual y situación que se pretende alcanzar</i> .....   | 180 |
| 2.5.2.   | Requerimientos del usuario.....  | 184 |
| 2.5.3.   | Recopilación y análisis de la información.....   | 185 |
| 2.5.4.   | Propuestas de solución.....  | 190 |
| 2.5.4.1. | Sistemas administradores de bases de datos para PC's y elección de alguna (características, ventajas y desventajas)..... | 190 |
| 2.5.4.2. | Características de herramientas visuales para el front-end y elección de la óptima.....                                  | 208 |
| 2.5.5.   | Diagrama de funcionalidad.....   | 224 |

**Capítulo III**  
**DISEÑO E IMPLANTACIÓN DEL SISTEMA**

|          |  |     |
|----------|--|-----|
| 3.1.     | Diagrama de flujo de datos.....                                | 229 |
| 3.2.     | Modelo entidad-relación y normalización de datos.....          | 236 |
| 3.2.1.   | Limitantes de mapeo.....                                       | 241 |
| 3.2.2.   | Diagrama entidad-relación.....                                 | 242 |
| 3.2.3.   | Normalización de datos.....                                    | 245 |
| 3.2.3.1. | Primera forma normal.....                                      | 245 |
| 3.2.3.2. | Segunda forma normal.....                                      | 249 |
| 3.2.3.3. | Tercera forma normal.....                                      | 251 |
| 3.3.     | Diccionario de datos.....                                      | 253 |
| 3.4.     | Diseño y construcción de la base de datos.....                 | 263 |
| 3.5.     | Diseño y construcción del front-end.....                       | 276 |
| 3.5.1.   | Fases para el desarrollo de aplicaciones con Visual Basic..... | 276 |
| 3.6.     | Pruebas e integración del sistema.....                         | 312 |
| 3.6.1.   | Pruebas del software.....                                      | 312 |
| 3.6.1.1. | Pruebas por módulo o unidad.....                               | 315 |
| 3.6.1.2. | Pruebas de integración de módulos o unidades.....              | 320 |
| 3.6.2.   | Pruebas orientadas al usuario.....                             | 322 |
| 3.7.     | Implantación.....  | 324 |
| 3.7.1.   | Objetivos.....   | 324 |
| 3.7.2.   | Definición.....  | 325 |
| 3.7.3.   | Instalación.....   | 325 |
| 3.7.4.   | Capacitación.....  | 330 |

**Capítulo IV**  
**IMPACTO DEL SISTEMA**

|        |                                     |     |
|--------|-------------------------------------|-----|
| 4.1.   | Mantenimiento del Sistema.....      | 333 |
| 4.1.1. | Preventivo.....                     | 334 |
| 4.1.2. | Correctivo.....                     | 335 |
| 4.1.3. | Control de cambios.....             | 336 |
| 4.1.4. | Consistencia de la información..... | 340 |

Conclusiones

Bibliografía

Apéndice A. "Código del Programa"

Apéndice B. "Manual de Usuario"

Apéndice C. "Formatos"

Apéndice D. "Diccionario de Datos"

# INTRODUCCIÓN

## Objetivo General

Un servicio es la función social que desempeña toda organización pública o privada para satisfacer una necesidad. En el caso de la industria farmacéutica el servicio que proporciona tiene especial trascendencia por estar en juego la salud y la vida humana. El tránsito adecuado de medicamentos de prescripción, es una extensión vital del proceso de investigación y desarrollo de nuevos y mejores medios de prevención y tratamiento de las enfermedades. La promoción y la difusión de información con fines educativos aseguran que los plenos beneficios de años de trabajo y de enorme inversión de ingenio y de dinero pueden llegar con prontitud a los pacientes. La industria farmacéutica cree que deben definirse y respetarse niveles elevados de exigencia en todas sus actividades.

En la actualidad algunas empresas farmacéuticas no cuentan con sistemas computarizados para la elaboración de inventarios, ni de distribución de pedidos, ni mucho menos un control de sus clientes. Para ello se requiere tener una actualización constante de los mismos y tener bajo control los factores que afecten a cada uno

Muchos organismos públicos y privados se han incorporado al mundo de la computación y con ello han obtenido las ventajas de los sistemas computarizados para realizar sus objetivos con un mayor grado de eficiencia y rapidez en sus actividades cotidianas.

Los directivos de dichas empresas se han percatado de la necesidad de filtrar su información que se obtiene de los procesos diarios, para que en un futuro permita tomar decisiones acertadas, con exactitud y veracidad. Esto con el propósito de satisfacer al máximo a sus clientes.

Por otro lado, dentro del área de cómputo se han desarrollado potentes herramientas, como son los manejadores de bases de datos y las interfaces visuales de usuario, que permiten el diseño e implantación de sistemas robustos y veloces en muy poco tiempo.

El objetivo del presente trabajo, es desarrollar un sistema computarizado que lleve un control de los diferentes clientes de una empresa farmacéutica, así como el control de sus productos y servicios.

Cumpliendo con las características óptimas para el control de productos en la empresa y promover la eficiencia mediante el uso de sistemas computarizados, atendiendo en un mínimo de tiempo los pedidos de sus clientes, dar respuesta al momento de qué producto se encuentra en inventario y/o agotado y acortar los canales de comunicación entre el cliente y la empresa, teniendo así una respuesta inmediata a los requerimientos del cliente.

El sistema propuesto, pretende conformarse como un mecanismo conveniente para dotar a las empresas farmacéuticas en México de elementos que les permitan realizar una mejor administración de sus recursos al mismo tiempo de elevar la seguridad y eficiencia en sus operaciones.

# **CAPÍTULO I**

## **SISTEMA DE AUTOABASTECIMIENTO FARMACÉUTICO A HOSPITALES**

# 1. SISTEMA DE AUTOABASTECIMIENTO FARMACÉUTICO A HOSPITALES

## 1.1. ANTECEDENTES

### 1.1.1. Historia General

En el segundo milenio A. C. las civilizaciones de Babilonia y Egipto ya habían producido pequeñas tabletas de arcilla y los largos rollos de pergamino, que sobreviven como los más antiguos farmacéuticos; con ello se demuestra que estos pueblos de valles fluviales ya conocían, aunque en forma rudimentaria, muchas de las formas básicas de administración de drogas que se emplean hoy en día, tales como: gárgaras, supositorios, inhalaciones, cataplasmas, ungüentos y otros centenares de sustancias diferentes usadas como drogas, entre ellas tenemos: asafétida, dátiles, ajo, habas, eléboro o veratro, etc. Es posible que al aplicarlas nunca se haya dejado de lado el uso de ritos teológicos y mágicos para aumentar la potencia de los medicamentos.

Un milenio más tarde los griegos trataron de encontrar explicaciones a todo. Por ello en el campo médico se encuentran los seguidores de Hipócrates, el cual nació unos 460 años A. C.; sus escritos y prácticas muestran las bases de todo método científico observación y clasificación, rechazo de teorías y una cautelosa generalización e inducción siempre abiertas a la discusión crítica y a la revisión, esto provocó un avance al mundo farmacéutico.

Al llegar el Cristianismo los monasterios se convirtieron en los centros de la vida intelectual, incluyendo el estudio y la práctica farmacéutica, los monjes a menudo cultivaban plantas medicinales fuera del monasterio y guardaban las drogas en un cuarto en el cual se incluía un laboratorio. Algunos de estos laboratorios eran famosos por su destilación de aguas aromáticas y cordiales.

Durante la segunda mitad de la Edad Media Europea la farmacia se alejó gradualmente de los monasterios, se separó más de la medicina y empezó a desarrollar normas y responsabilidades independientes en los centros más urbanizados. Esta tendencia se observa en Italia, España y Francia, que fueron puntos de tránsito de la droga y para los conocimientos farmacéuticos que seguían las rutas comerciales del Mediterráneo provenientes de la civilización Islámica.

En la Europa Moderna, la farmacia conquistó un lugar definido entre las profesiones científicas, por ejemplo en Prusia en 1725 se instituyeron exámenes obligatorios basados en normas académicas. A fines del siglo XVIII aparecieron instituciones privadas de educación farmacéutica en territorio Germano. En Italia a principios del siglo XIX se requerían estudios académicos y exámenes de farmacia en toda la península. Más o menos en la misma época la legislación francesa creó seis escuelas superiores de farmacia en el país (1803).

El primer establecimiento de E. U. que incluyó la farmacia en las atribuciones de uno de sus profesores fue la Medical School of Philadelphia College, en 1789. En esta institución, médicos enseñaban farmacia a otros médicos. Antes de la fundación del Philadelphia College of Pharmacy en 1821, fueron pocos los intentos de enseñar farmacia a los farmacéuticos propiamente dicho.

La importancia y demanda de estudios en el área farmacéutica ha logrado hoy en día tener un número indefinido de colegios y universidades en todo el mundo, con lo cual se crea una nueva profesión. El profesionalista de esta área ya no es sólo un servidor de la medicina que prepara prescripciones secretas que contienen medicación de valor, que a menudo es dudosa y que requieren arte y técnica. Hoy es un integrante del equipo sanitario que maneja drogas de gran potencia y valor, que son fabricadas por una industria actualizada y avanzada, el profesionalista las maneja consultando abiertamente a médicos y pacientes seguro de que así lo exige su conocimiento como experto en drogas.

## 1.1.2. Historia en Nuestro País

La industria farmacéutica en México se ha transformado de una actividad raquítica, limitada a la fabricación de agua destilada y a la importación de productos totalmente manufacturados, en una industria que abastece el 97% de las necesidades de medicamentos de la población.

La industria farmacéutica es de gran importancia en México, además de contribuir con el 12% del PIB de la industria química, provee al país de medicamentos e insumos básicos para la atención de la salud a precios accesibles (comparados con los países desarrollados en donde su precio llega a multiplicarse en 6 u 8 veces más).

México ocupa el undécimo lugar mundial en términos del valor del mercado químico-farmacéutico. En la década pasada, la industria farmacéutica experimentó un tremendo crecimiento, más del 160% acumulado de 1980 a 1992, con un crecimiento moderado en los últimos 3 años por arriba del 3%, aún con la contracción en los niveles de inversión presentados en el sector.

Esta industria es altamente intensa en inversión de capital, es decir, se necesita un monto importante de recursos y de una constante adaptación de procesos que condicionan y rigen su desarrollo. Su exportación requiere de esfuerzos concertados en investigación y desarrollo para mantener la innovación de productos.

Este sector contribuye con el 0.48% del PIB nacional y con el 2.01% del PIB manufacturero. Aunque cuantitativamente es un segmento pequeño en la economía, la naturaleza altamente tecnológica así como su estrecha relación con el Sistema Nacional de Salud, sustentan y justifican su importancia estratégica para el país.

Actualmente la industria farmacéutica mexicana está integrada por empresas fabricantes de materias primas (fármaco-químicas), medicamentos (de uso humano y veterinario) y productos auxiliares para la salud (material de curación, quirúrgico y radiológico, reactivos de diagnóstico, productos dentales e higiénicos).

Completan la cadena productiva de esta rama el eslabón de la comercialización formado por: mayoristas, medio mayoristas y farmacias. La cadena de distribución, con datos estimados, se conforma por 2 distribuidores que a nivel nacional cubren aproximadamente el 50% del mercado en valores; 4 de cobertura seminacional, 14 distribuidores regionales y 14 cadenas importantes de farmacias y aproximadamente 15,900 farmacias ubicadas en todo el territorio mexicano.

Por Ley, las empresas farmacéuticas establecidas en México están agrupadas, en su mayor parte, en la Cámara Nacional de la Industria Farmacéutica (CANIFARMA), si bien algunas empresas fabricantes de productos auxiliares para la salud y fármaco-químicos también están agrupadas en la CANACINTRA.

### **Número de Empresas**

De acuerdo a los datos más recientes (1992) en CANACINTRA están agrupadas: 389 empresas de las cuales 225 fabrican medicamentos de uso humano (58%); 84 fabrican productos auxiliares para la salud (22%); 71 fabrican medicamentos de uso veterinario (18%) y 9 fármaco-químicos o materias primas (2%).

De acuerdo a la composición de la industria farmacéutica agrupada en CANACINTRA, por estratos, 290 empresas (el 75%), son micro, pequeñas y medianas y 99 se ubican en la escala industrial de grandes empresas.

### **Distribución Geográfica**

En cuanto a su distribución geográfica: 208 empresas están ubicadas en el Distrito Federal, 44 en el Estado de México, 41 en Jalisco, 17 en Puebla, 11 en Querétaro, 10

en Nuevo León e igual número en Morelos, el resto se distribuye en las siguientes entidades: Veracruz, Yucatán, Coahuila, Chihuahua, Michoacán y el menor número en Nayarit y Baja California.

### **Ventas**

En 1992 las ventas de la industria farmacéutica fueron del orden de los 3,057 millones de dólares. De este total 2,615 correspondieron a la línea humana, 249 a la industria **PAPS** (Productos Auxiliares para la Salud), 101 a la industria farmacéutica veterinaria y 92 a la fármaco-química.

Actualmente, la industria farmacéutica abastece un 97% las necesidades de medicamentos de la población y sólo se importa el 3%. Cabe señalar que ningún país produce para sí mismo el 100%, ya que existen especialistas cuya producción no sería costeable ya que no es necesario fabricar localmente, en función del perfil epidemiológico y las necesidades de cada país.

El 81.43% de las ventas de la industria farmacéutica (en volumen) se destinan al mercado privado y el 18.57% al mercado público (integrado por las instituciones del sector de salud: ISSSTE, IMSS, SSA); el 60% de las ventas totales de la industria farmacéutica agrupada en CANIFARMA (en unidades) corresponden al mercado privado y el 40% al mercado público.

Este sector presenta un importante grado de integración nacional: materias primas y excipientes (que son nacionales entre un 40% y 60%); materiales de envase y empaque (que son nacionales en un 100%), mano de obra (que es totalmente nacional). Por lo tanto, en promedio, la integración de la industria farmacéutica sobre el costo directo es mayor del 80%.

El total de bienes en el mercado es de 7,000 en 19,000 presentaciones. Una característica importante del mercado farmacéutico es que se rige por un sistema

concertado de control de precios de todas las medicinas, aspecto que las hace del 35% al 50% más baratas que las estadounidenses.

En México el precio promedio de un medicamento es de 2.41 USD, en Estados Unidos es de 12.90 USD, en Canadá es de 10.11 USD y en Argentina es de 5.34 USD.

La industria farmacéutica mantiene una constante producción de productos novedosos, por marcas, por usos y por efectividad de acción. En promedio cada dos meses se introducen al mercado de 5 a 10 productos nuevos (entre éticos y genéricos). Esto se realiza generalmente a través de alianzas con marcas diversas (diferentes laboratorios) para ofrecer mayor escala y mejor distribución.

### **Importaciones y Exportaciones**

Desde 1980 la importación de medicinas representan el 2% del consumo nacional y las exportaciones el 4% de las ventas totales. Actualmente hay muy pocas exportaciones de México a Estados Unidos, en cambio de dicho país, México importa ingredientes activos y no fármacos.

Otros mercados a los que se concurren son: Centro y Sudamérica, África, Asia y el Caribe. La industria fármaco-química importa materias primas (reactivos y solventes) para fabricar principios activos. La industria farmacéutica produce medicinas dosificadas, usando principios activos adquiridos localmente o importados.

Este sector ha presentado tradicionalmente un déficit en su balanza comercial, sin embargo, en los años recientes han prestado incrementos significativos en lo general y en casos específicos se ha presentado un superávit dado que se han diversificado los mercados de alcance para la fabricación de medicamentos como lo es a Sudamérica y algunos países de Europa (por ejemplo: cancerígenos, analgésicos y vitamínicos). Sin embargo, el crecimiento de la balanza agregada para el segmento de fármaco-químicos sigue siendo deficiente.

## **Empleos**

La industria farmacéutica genera aproximadamente más de 50 mil empleos directos (y aproximadamente unos 90,000 indirectos) y se caracteriza por tener uno de los mejores salarios entre las industrias manufactureras debido a la gran cantidad de técnicos y profesionales especializados que requiere.

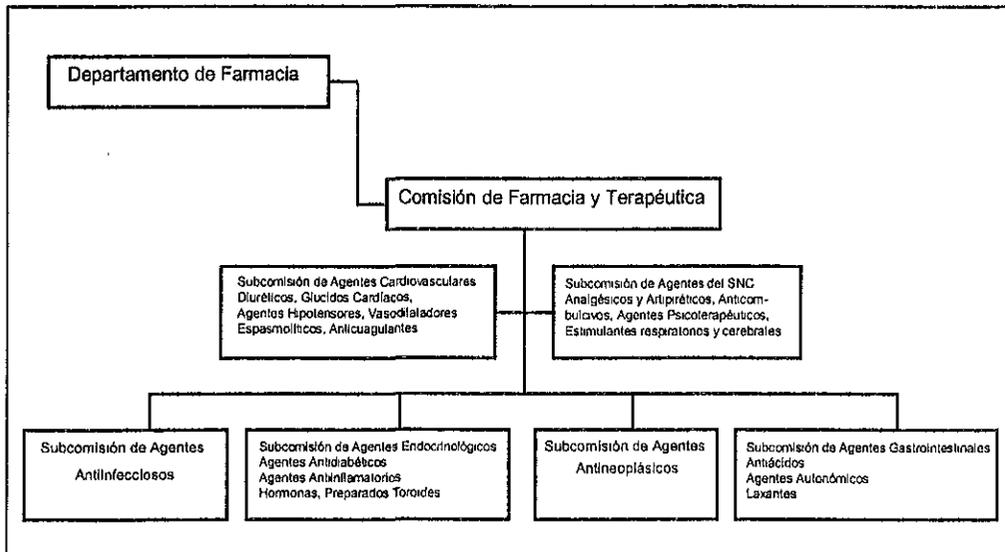
De 52,349 empleos directos: 41,142 son generados por las empresas grandes y aproximadamente 5,249 por empresas pequeñas, 5,694 por las empresas medianas y 354 por las microempresas.

Del total de empleos generados, el 80% se concentra en las empresas que fabrican medicamentos de uso humano; el 11% en la industria de productos auxiliares para la salud, el 4% en la industria veterinaria y el 3% en fármaco-químicos.

La industria farmacéutica tiene que renovarse constantemente, de lo contrario sería obsoleta, por lo que se busca innovaciones tecnológicas y un buen control de calidad.

### **1.1.3. Perfil de la Industria**

Una empresa que fabrica productos farmacéuticos, tiene una operación compleja, pues se reúnen con fines comunes un equipo de científicos y técnicos que representan prácticamente a todas las ciencias. Otras contribuciones al esfuerzo cooperativo son las de sus autoridades ejecutivas, abogados, contadores, ingenieros, analistas de sistemas, redactores, vendedores y muchos otros cuya habilidad y talento mantienen la viabilidad de esta empresa comercial con rasgos que le son propios. La figura 1.1.3.1 muestra los departamentos de una industria farmacéutica.



**Figura 1.1.3.1. Departamentos de una industria farmacéutica.**

La industria farmacéutica es única por las siguientes razones :

### **Categorías de Productos**

Los productos farmacéuticos más que los de cualquier otra industria importante, pueden afectar íntimamente la vida, la salud, el bienestar y la felicidad de la gente. El Physicians' Desk Reference clasifica más de 151 categorías terapéuticas diferentes, desde alérgenos y antibióticos hasta vitaminas y medios de contraste para rayos X. En el índice de nombres genéricos y químicos se enumeran más de 1,200 drogas diferentes.

### **Disciplinas Científicas**

La industria farmacéutica depende para poder existir de científicos, incluyendo farmacéuticos, que abarcan un inmenso compás de ciencia y técnica. Algunas de las disciplinas y funciones científicas que necesitan incluyen: química para el control de

calidad y síntesis, además para el procesamiento de componentes esenciales; farmacología y toxicología para la selección de drogas y las pruebas de seguridad; bioquímica para seguir el metabolismo y la absorción de drogas; la farmacia física para la investigación farmacéutica y desarrollo de procesos; las ciencias de la información para documentación y respaldo técnico; medicina para pruebas clínicas y por último la biostática para una evaluación científica exacta.

### **Legislación de Control de Calidad**

Las operaciones de la industria farmacéutica están bajo control constante de organismos de provincia y nacionales, cada uno con reglamentaciones propias que deben cumplirse. Esta legislación junto con el sentido de responsabilidad de la misma industria tiene como resultado la fabricación de productos al más alto nivel posible de calidad, esto a su vez abre oportunidades para los científicos encargados de mantener ese nivel de calidad.

### **Investigación y Desarrollo**

Estos aspectos contribuyen a dar un sello propio a la industria farmacéutica, aunque también existe naturalmente en otras industrias pero en ésta, trátase de productos vendidos con receta o sin ella (drogas, cosméticos, etc.), la investigación y desarrollo dan origen a todos los nuevos productos y a un buen número de innovaciones.

Los gastos destinados a la investigación y desarrollo son otro aspecto que caracteriza a la industria farmacéutica. Por la misma naturaleza de sus productos, los esfuerzos de investigación y desarrollo de productos farmacéuticos consumen generalmente una parte mucho mayor de los ingresos de la corporación que en otras industrias.

### **Datos Estadísticos**

El tamaño (magnitud) y la diversidad son otras características interesantes de esta industria. Existen literalmente miles de laboratorios y establecimientos farmacéuticos, todos registrados **FDA** (Food and Drug Administration), algunas desarrollan y fabrican

diversos productos que abarcan muchos campos de tratamiento y otras se especializan en campos terapéuticos limitados como los de preparaciones dermatológicas u oftálmicas.

#### **1.1.4. Asistencia Institucional de Pacientes y Características de un Hospital**

Se ha acuñado el término de asistencia "institucional" para reflejar este papel ampliado del hospital. Un hospital tiene divisiones departamentales, una de ellas es la farmacia institucional u hospitalaria, este departamento está a cargo de conseguir, almacenar, componer, elaborar, envasar, controlar, ensayar, expedir, distribuir y monitorear los medicamentos para los pacientes hospitalizados.

Cada departamento cumple con sus funciones de acuerdo a las normas y objetivos del hospital. Véase la figura 1.1.4.1, se muestra un organigrama de Farmacia Hospitalaria. He aquí algunos de los departamentos que se ocupan de la asistencia profesional de los pacientes (diagnóstica o terapéutica).

Asistencia Ambulatoria

Anestesia

Banco de Sangre

Esterilización Central

Laboratorios Clínicos

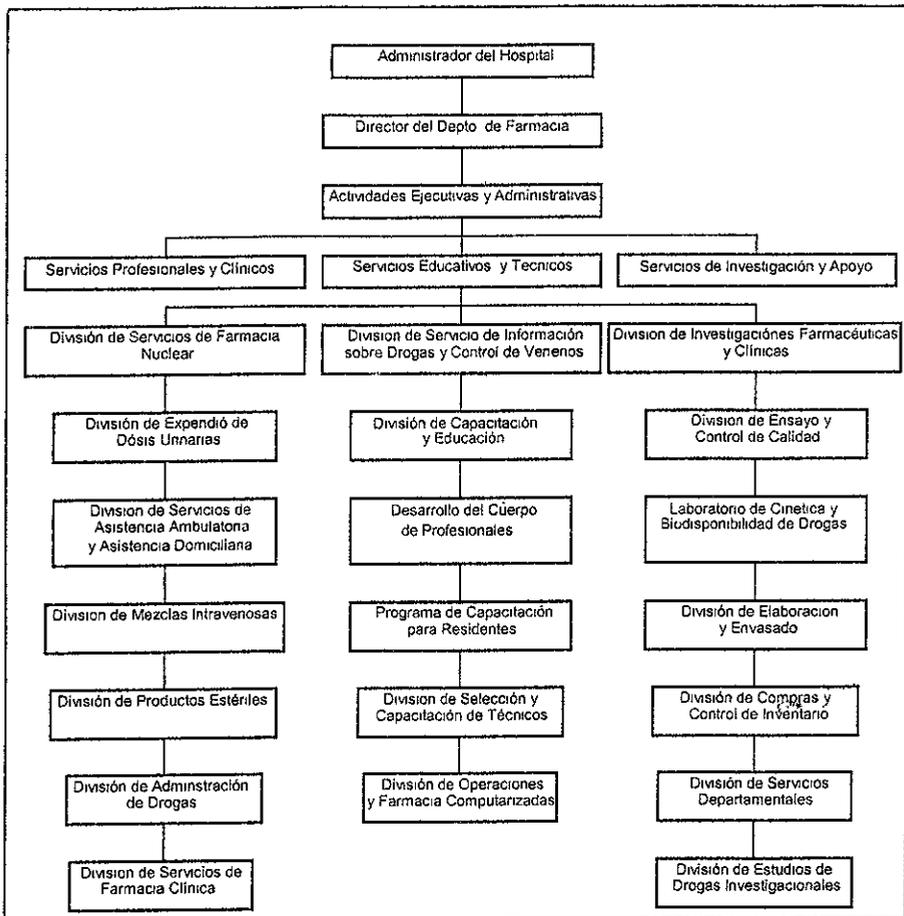
Servicio de Odontología

Servicio Dietético y Nutricional

Laboratorio Electrocardiográfico

Laboratorio Electroencefalográfico

Sala de Guardia



**Figura 1.1.4.1. Organización farmacia hospitalaria.**

Biblioteca Médica

Registros Médicos

Servicio Social Médico

Medicina Nuclear

Servicio de Enfermería

Terapia Ocupacional

Servicio de Farmacia

Medicina Física

Radiología y Radioterapia

Terapia Respiratoria

Los departamentos que se ocupan del manejo comercial o de la parte administrativa del hospital comprenden:

Contaduría

Internación

Ingeniería Biomédica

Oficina Comercial

Cantina y Cafetería

Transporte Central

Créditos y Cobranzas

Servicios Computarizados

Ingeniería y Mantenimiento

Limpieza y Lavandería

Servicio de Informaciones

Personal y Pagos de Salarios

Oficina de Correos

Compras y Depósito de Existencias

Conmutador Telefónico

Servicio de Voluntarios

## **Características de un Hospital**

Los hospitales se clasifican de acuerdo a:

- Tipo de Servicio.
- Tiempo de Internación.
- Propietarios.
- Capacidad de Camas.

Por el tipo de servicio tenemos a los Hospitales Generales y Hospitales Especiales, la diferencia entre estos son:

General: Presta su servicio a pacientes de cualquier tipo de enfermedad; médica, quirúrgica, pediátrica, psiquiátrica y maternidad.

Especial: Son los que se restringen a casos especiales como cáncer, psiquiatría o pediatría.

Por el tiempo de internación tenemos hospitales de corto plazo y hospitales de largo plazo, los primeros hacen referencia a un periodo de internación de los pacientes menor a los 30 días; los segundos hacen referencia a un periodo de más de 30 días de internación.

Por tipo de propietarios tenemos a Hospitales Gubernamentales y Hospitales no Gubernamentales.

De los gubernamentales son:

Federales (fuerzas armadas, servicio de salud pública)

Estatales

Municipales

De los no gubernamentales son:

Sin fines de lucro

Individuales

Sociedades

Por capacidad de camas se tiene una tabla:

|                   |
|-------------------|
| menos de 50 camas |
| 50 a 99 camas     |
| 100 a 199 camas   |
| 200 a 299 camas   |
| 300 a 399 camas   |
| 400 a 499 camas   |
| 500 camas o más   |

En los tiempos modernos los hospitales han adoptado una cuarta función, la de la salud pública (medicina preventiva). En consecuencia las cuatro funciones fundamentales de los hospitales son atender pacientes, docencia, investigación y salud pública.

### **1.1.5. Características de un Farmacéutico**

Los farmacéuticos se reúnen con médicos especialistas en un plano de igualdad en una comisión formal de farmacia y terapéutica, realizando rondas médicas de asistencia para los pacientes en todas las cuestiones relacionadas con la drogoterapia.

Responsabilidades del farmacéutico:

1. Asegura el cumplimiento de las políticas y procedimientos establecidos.
2. Verifica la exactitud de las dosis preparadas:
  - a) mezclas intravenosas.
  - b) dosis unitarias.

- c) farmacia para pacientes externos.
3. Mejora el debido control de la droga:
    - a) verifica que la droga se almacene y entregue correctamente.
    - b) verifica el cumplimiento de todas las leyes estatales y federales sobre droga.
  4. Asegura que se trabaje con buenas técnicas para componer las mezclas intravenosas y los preparados extemporáneos.
  5. Provee un buen registro de datos y facturación:
    - a) registro de medicamentos de los pacientes.
    - b) registro de composiciones extemporáneas.
    - c) registros de mezclas intravenosas y su facturación.
    - d) registro de droga investigable.
    - e) facturación para pacientes externos.
    - f) Informes.
  6. Mantiene su competencia profesional, en particular en el conocimiento de la estabilidad e incompatibilidad de la droga.
  7. Asegura la debida capacitación de personal nuevo en las políticas y procedimientos del área de expendio central.
  8. Coordina las actividades del área con el personal disponible para usar en la menor medida posible el personal y los recursos.
  9. Mantiene prolija y ordenada el área de expendio central.
  10. Se comunica con el personal del laboratorio acerca de los nuevos acontecimientos en el área y colabora en las evaluaciones de los empleados.
  11. Provee información sobre droga al personal del laboratorio, médicos y enfermeras
  12. Coordina las necesidades farmacéuticas globales de las áreas asistenciales de los pacientes con el área expedidora central.

## **1.2. ESPECIFICACIONES PARA LA ELABORACIÓN DE PRODUCTOS**

### **1.2.1. Control de Calidad**

La trayectoria de un nuevo producto desde su concepción hasta que llega al consumidor es ardua, desde el punto de vista de inversiones a menudo riesgosa. Por cada compuesto que se descubre es necesario elaborar un método muy seguro de síntesis y extracción, definir normas de pureza y crear técnicas para determinar el destino a corto y largo plazo de la sustancia y sus metabolitos en los sistemas biológicos.

Numerosas pruebas en animales van conformando el perfil de toxicidad de la droga y revelan, mediante técnicas avanzadas de selección, sus efectos farmacológicos más evidentes. Los científicos dedicados al desarrollo farmacéutico deben fijar formas de dosificación que sean compatibles con la química del compuesto y los sitios efectores biológicos, y que se liberen generalmente en un tiempo determinado, la dosis apropiada de la droga. Las pruebas de estabilidad determinan la "vida en el estante" (vida útil del producto) y detectan los cambios producidos en su fórmula durante su conservación.

Es necesario contar con métodos confiables y eficientes para fabricar la droga. Rigurosas medidas de control, que a su vez son el resultado de investigaciones especializadas, se aplican al proceso y al producto, para asegurar que las especificaciones de la droga terminada queden dentro de límites bien definidos.

Dentro de la sección de materiales clínicos se debe conocer en extremo el uso proyectado de la droga probada, controlar su distribución a los lugares de pruebas clínicas, llevar registros ordenados y controlar de cerca su inventario y conservación. Además de saber utilizar técnicas de mezclado, formulación y fabricación, envase correcto y principios científicos afines.

La industria farmacéutica continúa siendo un sector vital del ciclo de la asistencia de la salud en la conducción de investigaciones y en la elaboración de productos que mantienen y restituyen la vida. El último decenio ha presenciado una evolución en los conceptos relacionados con la seguridad y control de calidad de estos productos.

Los cambios conducentes para asegurar la inocuidad y eficacia terapéutica de los productos farmacéuticos surgieron de una cantidad de factores que han operado dentro y fuera de la industria. Por dentro están las normas que la industria se ha impuesto a sí misma y cuyo ejemplo es un documento preparado en 1967 por la Asociación de Fabricantes Farmacéuticos, titulado "Principios Generales del Control de la Calidad Total en la Industria de las Drogas". El documento de la AMF sirvió de base para las normas reguladoras preparadas por la Administración de Alimentos y Drogas, que lleva por título, "Buenas Prácticas Actuales en la Elaboración, Procesamiento, Envasado o Tenencia de Drogas Humanas o Veterinarias" conocidas también como **CGMP**, se han convertido en las directivas externas primarias que se utilizan en la Industria y la Administración de Alimentos y Drogas en el control e inspección de las instalaciones elaboradas.

### **1.2.1.1. Organización del Control y Seguridad de la Calidad**

Aunque a menudo los términos de control de calidad y seguridad de la calidad se usan en forma relativa según la estructura de la empresa, se definirá por ello a cada una de éstas a continuación.

#### **Control de Calidad**

Se le puede definir como el control cotidiano de la calidad dentro de una compañía, con un departamento a cargo de hombres de ciencia y técnicos responsables de:

1. Aceptar o Rechazar las materias primas y componentes de envasado que se reciben

2. Probar e Inspeccionar los procesos internos
3. Verificar que los sistemas se controlan y monitorean
4. Aprobar o Rechazar las formas posológicas completadas

En consecuencia, el control de calidad no sólo comprende las pruebas analíticas del producto terminado, sino también la vigilancia de todas las operaciones, comenzando con la recepción de materias primas seguidas por las operaciones de producción y envasado, ensayo con los productos finales, su documentación, vigilancia y distribución.

### **Seguridad de la Calidad**

Se le define como la responsabilidad que una organización tiene de determinar que los sistemas, recursos y procedimientos escritos sean adecuados y se cumplan, a los efectos de asegurar que los productos son controlados y que en sus formas posológicas finales habrán de cumplir todas las especificaciones a las que está sujeta.

#### **1.2.1.2. Control de Calidad Total**

La alta calidad de los productos farmacéuticos es el resultado del minucioso cumplimiento de procedimientos escritos para realizar todas las operaciones, empezando con la investigación.

Algunos de los procedimientos para la realización del control de la calidad total son los siguientes: Las materias primas deben adquirirse de abastecedores de buena reputación para obtener productos estables y uniformes al ser incorporados en la forma posológica terminada. Hay que diseñar instalaciones y elegir los equipos apropiados para eliminar el potencial de contaminación cruzada de un producto por otro, planificar la circulación y los movimientos del personal para reducir la posibilidad de mezclar productos y además de que el aire y agua que se proveen para la producción sean adecuados en cantidad y calidad.

Los departamentos de embarque tienen la responsabilidad de cuidar que los productos estén a salvo de manipulaciones y condiciones ambientales adversas mientras están en tránsito hacia los lugares de distribución y/o clientes.

### **1.2.1.3. Fases de Control de la Calidad de Productos Farmacéuticos**

#### **Control Analítico**

El laboratorio de control analítico es responsable de ensayar y aprobar materias primas, procesos internos y productos terminados. En él se realizan análisis complejos, oportunos y exactos; los análisis son requeridos para evaluar la aceptabilidad de un producto. Se debe contar en este laboratorio con equipo sofisticado y exacto. Se debe contar también con especificaciones detalladas que conduzcan a un nivel de exactitud y limitación de rangos de aceptabilidad o rechazo del producto, así como métodos de ensayo.

Las especificaciones antes mencionadas se hallarán en varias fuentes de consulta. El **USPNF** (United States Pharmacopeial National Formulary) es publicado en un programa quinquenal por la Convención Farmacopeica de Estados Unidos. Los patrones establecidos por la UPS han sido reconocidos como oficiales por la Ley Federal de Alimentos, Drogas y Cosméticos y por el Congreso de los Estados Unidos de América. Los patrones son preparados y revisados con regularidad en todo o en parte, de modo que siempre están en vigencia.

El objetivo principal de los patrones es establecer normas autorizadas para materiales, sustancias y sus preparaciones que se utilizan en el arte de curar, además con ellos se establecen títulos, definiciones, descripciones y patrones para la identidad, calidad, potencia, pureza y cuando sea práctico métodos para su examen y fórmulas para su fabricación.

## **Control de Inspección**

En esta fase se comprende el muestreo de las materias primas, componentes de los envases y etiquetas: *la inspección física del producto en diversos pasos intermedios; la inspección de la línea de envasado y control del inventario de expedición dentro del ciclo de distribución.*

## **Documentación**

En el curso de la producción de un producto farmacéutico se generan muchos documentos y registros. A cada lote se le asigna un código o número específico. Toda la documentación relacionada con un código específico se conoce como "registro de lote" y comprende datos sobre cada fase significativa de la producción, control y distribución. El registro de lote es un detalle histórico de todas las fases de su producción interna además de las gráficas y las tiras impresas de las computadoras sobre operaciones importantes como autoclavado, secado, monitoreo de partículas en el aire, liofilización, etc.; se convierten en parte de la historia del lote. Una vez completado el lote incluso las pruebas analíticas y físicas finales, se debe completar un paso más antes de aprobar la distribución del lote, esto es, hacer una revisión final de todos los documentos y registros relacionados con éste. Se debe anexar cada documento del registro del lote para verificar si está completo y es exacto. Toda discrepancia debe investigarse y aclararse sin pérdida de tiempo. Sólo una vez completada esta revisión se puede autorizar su distribución.

Una vez autorizado el lote, hay que llevar registros exactos de expedición para que se sepa su distribución. Con estos registros se puede seguir el lote hasta la plaza mercantil, de modo que, si se presenta la necesidad se facilita su retiro del mercado.

Estos procedimientos son realizados por gente responsable y capacitada para contribuir en la seguridad e integridad del producto, este personal se basa en "Los Procedimientos Operativos Estándar" (POE) y de acuerdo a las disposiciones de la CGMP, el laboratorio da un informe "interno" sobre su nivel de cumplimiento y con ello

permitirá que se introduzcan los cambios o correcciones necesarias antes de que un producto falle o sea señalado durante una inspección por un investigador de la FDA.

### **Nuevos Adelantos**

El empleo de estadísticas y análisis de tendencias son recursos que la industria farmacéutica ya utiliza para determinar el tamaño apropiado de las muestras que se requieren para realizar ensayos. Además, para medir la uniformidad de las formas posológicas sólidas y para hacer diagramas tendencias de factores significativos a los efectos de corregir situaciones sin control antes de que se tenga un producto inaceptable.

El procesamiento electrónico de datos se ha convertido en otro recurso útil para estimar procesos y parámetros de prueba y para analizar los datos recogidos durante la producción. El control de operaciones por computadora y microprocesadores provee capacidad para elaborar productos de mayor calidad. Estos sistemas han venido desplazando a los más antiguos y han dado lugar a nuevos enfoques en control de procesos internos, en recolección y análisis de datos y en todo el sistema de control de calidad.

### **1.2.2. Control Ambiental**

Existen muchos adelantos en el control total de la calidad, entre ellos está el creciente reconocimiento de que el ambiente y los sistemas para controlar a ésta ejercen una influencia importante sobre la calidad del producto terminado. Se reconoce bien que los productos parenterales u oftálmicos estériles deben producirse de una manera tal que asegure su esterilidad, de modo que es esencial el control de las áreas donde se elaboren.

En estas áreas es fundamental hacer un monitoreo microbiológico del aire y del agua para controlar el nivel de partículas y de materia microbiana en el aire. En el Estándar

209 B del Gobierno Federal sobre "Requisitos de limpieza en habitaciones y lugares de trabajo, ambiente controlado" se describen varios niveles de áreas "limpias". La industria suele utilizar las especificaciones descritas, que clasifican la limpieza del aire de acuerdo con la cantidad de partículas (de un tamaño dado) por pie cúbico de aire. En general, las condiciones designadas como "clase 100" se mantienen en las áreas donde se envasan productos parenterales en recipientes estériles. A la clase 100 se le define como un área que se puede mantener con menos de cien partículas de 0.5 micras o más por pie cúbico de aire.

Otro procedimiento de control esencial es el monitoreo microbiológico del ambiente donde se elaboran productos no estériles. El objetivo de este monitoreo es el de determinar primero los niveles de partículas y microbios en un área para cerciorarse de que son razonables. Si resultan ser excesivos, se toman medidas para reducirlos a límites aceptables. Una vez desarrollada esta base se hace un monitoreo regular para saber si las operaciones siguen desarrollándose dentro de estos límites aceptables, de lo contrario, se toman medidas correctivas inmediatas.

El monitoreo y control de las partículas y materias microbianas aseguran mejor la calidad y estabilidad final del producto porque el ambiente ha sido controlado y el producto no ha sido expuesto a un nivel inaceptable de partículas generado por una situación sin control.

### 1.2.3. Práctica Actual de Buena Fabricación en la Elaboración, Procesamiento, Envasado o Almacenamiento de Drogas Humanas y Veterinarias

#### Provisiones Generales

|                   |   |
|-------------------|---|
| Especificaciones. | Interpretación.   |
| Definiciones.     | Se explican los alcances de las disposiciones para productos de drogas de prescripción y venta libre, incluso productos biológicos. |

#### Organización y Personal

|                               |  |
|-------------------------------|--|
| Calificación de Personal.     | El personal, sea de supervisión u operacional debe estar calificado por capacitación y experiencia para realizar los objetivos que se le asignen.                |
| Responsabilidad del Personal. | Se definen las responsabilidades del personal dedicado a la fabricación de productos de drogas con respecto a la higiene personal, indumentaria y estado médico. |
| Consultores.                  | Las calificaciones de los consultores deben ser aprobadas por control de calidad.  |

#### Edificios e Instalaciones

|   |   |
|---|---|
| <ul style="list-style-type: none"><li>• Diseño y Construcción.</li><li>• Iluminación.</li></ul> | Los edificios e instalaciones sólo se consideran aceptables si se prestan para el fin a que están destinados y se |
|---|---|

|  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• Ventilación, Filtración, Calentamiento y Enfriamiento del aire.</li> <li>• Plomería.</li> <li>• Cloacas y Desperdicios.</li> <li>• Baños e Instalaciones para lavarse.</li> <li>• Sanidad.</li> <li>• Mantenimiento.</li> </ul> | <p>les pueda mantener. Se delinean conceptos de construcción como sistemas de acondicionamiento de aire, iluminación, comedores y plomería, incluso aguas corrientes, sistemas cloacales y baños.</p> |
|--|---|

### Equipos

|  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• Diseño, Tamaño y sitio de los Equipos.</li> <li>• Construcción de los Equipos.</li> <li>• Limpieza y Mantenimiento de los Equipos.</li> </ul> | <p>Los equipos deben diseñarse y construirse del tamaño adecuado, estar en sitios convenientes y poder mantenerse, para que se los pueda considerar apropiados para el uso que se proyecta darles.</p>  |
| <p>Equipos Automáticos, Mecánicos y Electrónicos.</p>  | <p>Se alude al uso de equipos automáticos, procesadores de datos y computadoras, destacando la necesidad de verificar el ingreso y egreso de información y de tener debidamente calibrados los registradores, contadores y otros dispositivos electrónicos o mecánicos.</p> |

### Control de Componentes, Recipientes y Cierres para Productos de Drogas

|  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• Requisitos Generales.</li> <li>• Recepción y almacenamiento de componentes no ensayados, recipientes para productos y cierres.</li> </ul> | <p>Se debe contar con procedimientos escritos que describan la recepción, identificación, almacenamiento, manipulación, muestreo, ensayo y aprobación o rechazo de componentes</p> |
|--|--|

|   |  |
|---|--|
|   | (materias primas) y productos de drogas.   |
| <ul style="list-style-type: none"> <li>• Ensayo y Aprobación o Rechazo de Componentes, Recipientes para Productos de Drogas y Cierres.</li> <li>• Uso de Componentes, Recipientes y Cierres aprobados.</li> <li>• Componentes, Recipientes y Cierres Rechazados.</li> </ul> | Una vez aprobados o rechazados, estos materiales deben identificarse como tales y almacenarse. Si son aprobados, deben inventariarse de manera de que se asegure que se usará primero el material en existencia que había sido aprobado primero. Los materiales sujetos a deterioro durante el almacenamiento, deben volver a ensayar en un plazo apropiado basado en los perfiles de estabilidad. |
| Recipientes y Cierres del Producto.   | Los recipientes y cierres (materiales en contacto con el producto) no deben ser reactivos ni aditivos con el producto.   |

### Controles de Producción y Procesos

|  |  |
|--|--|
| Procedimientos Escritos y Desviaciones.  | Se requieren procedimientos escritos de operación estándar para cada proceso de producción. Toda la desviación de cada procedimiento debe investigarse, registrarse y aprobarse antes de la aceptación final del producto. |
| <ul style="list-style-type: none"> <li>• Cargado de Componentes.</li> <li>• Cálculo de Rendimiento.</li> <li>• Identificación de Equipos.</li> </ul> | Todos los productos deben formularse para que provean no menos del 100% de la cantidad requerida del componente activo. Llévense registros   |

|   |  |
|---|--|
|   | de cada componente y de la cantidad que se incorpora en cada lote.   |
| Muestreo y Ensayo de Materiales y Productos durante el procesado.   | Se identificarán los pasos importantes del procesado y se obtendrán muestras, pruebas y aprobaciones antes de continuar el ciclo de producción. En caso contrario se aplicarán limitaciones de tiempo a los pasos del procesado. |
| <ul style="list-style-type: none"> <li>• Limitaciones de tiempo de la Producción.</li> <li>• Control de la Contaminación Microbiológica.</li> </ul> | Se prepararán procedimientos adecuados para ensayar componentes, productos y ambientes para establecer que un producto no está microbiológicamente contaminado.  |
| Reprocesado.  | Se permite el reprocesado de un producto si existen procedimientos escritos que cubran los métodos que se han de usar y si control de calidad los aprueba.   |

### Control de Envasado y Etiquetado

|  |   |
|--|---|
| Examen de Materiales y Criterios de Uso.   | Los materiales de etiquetado y envasado se recibirán, identificarán, almacenarán, muestrearán y ensayarán de acuerdo con procedimientos detallados por escrito. |
| <ul style="list-style-type: none"> <li>• Seguro de Etiquetado.</li> <li>• Operaciones de Etiquetado y Envasado.</li> <li>• Inspección del producto.</li> </ul> | Se ejercerán controles especiales sobre el etiquetado para asegurarse de que sólo se emitan las etiquetas correctas para el envasado de un                      |

|                       |   |
|-----------------------|---|
|                       | producto dado con la calidad emitida.   |
| Fecha de Vencimiento. | Luego de los estudios de estabilidad que corresponden en las condiciones de temperatura prescrita, los productos que salen al mercado deben tener una fecha de vencimiento para tener la seguridad de que se usarán dentro de su vida de almacenamiento prevista. |

### Tenencia y Distribución

|  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• Procedimientos de Depósito.</li> <li>• Procedimientos de Distribución.</li> </ul> | Se describen los requisitos para almacenamiento y distribución de los productos y su tenencia en condiciones apropiadas de luz, temperatura y humedad. |
|--|--|

### Controles de Laboratorio

|  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• Requisitos Generales.</li> <li>• Pruebas y Autorización para distribuir el producto.</li> </ul>                   | Conciernen a los procedimientos escritos, en forma de especificaciones, normas, planes de muestreo y procedimientos de ensayo que se utilizan en el laboratorio para controlar componentes y productos terminados.                    |
| <ul style="list-style-type: none"> <li>• Pruebas de Estabilidad.</li> <li>• Requisitos para Pruebas especiales.</li> <li>• Muestras de Reserva.</li> </ul> | Se deberá seguir con un programa de pruebas de estabilidad para establecer las características de estabilidad de los productos, de lo anterior dependerá la asignación de condiciones para el almacenamiento y fechas de vencimiento. |

|                               |   |
|-------------------------------|---|
| Animales de Laboratorio.      | Los animales que se usen en cualquier prueba deben mantenerse y controlarse de una manera apropiada para su uso.  |
| Contaminación por Penicilina. | Los productos farmacéuticos no se pueden comercializar si al ensayarlos con cualquier procedimiento prescrito resulta tener cualquier nivel detectable de penicilina. |

### Registros e Informes

|  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• Requisitos Generales.</li> <li>• Limpieza de Equipos y Registro de su uso.</li> <li>• Registros de componentes, recipientes, cierres y etiquetas de productos.</li> </ul> | Durante la elaboración de productos farmacéuticos se deben generar detalles de los diversos registros y documentos que deben estar disponibles para examinarlos.  |
| <ul style="list-style-type: none"> <li>• Registros maestros de producción y control.</li> <li>• Registros de producción y control de lotes.</li> <li>• Resumen del registro de producción.</li> </ul>              | Preparar un registro maestro de producción para cada producto, describiendo todos los aspectos de su elaboración, envasado y control. De este registro maestro se obtienen los registros de los lotes individuales. |
| <ul style="list-style-type: none"> <li>• Registros de Laboratorio.</li> <li>• Registros de Distribución.</li> </ul>  | Comprenden anotaciones de embarque, remisiones, notas de carga y todos los documentos relacionados con la distribución.   |
| Archivo de quejas.   | Mantener el registro de las quejas recibidas de los consumidores y profesionales, junto con el informe y su respuesta.  |

## Productos Devueltos y Rescatados

|                       |  |
|-----------------------|--|
| Productos Devueltos.  | Llevar registro de los productos devueltos por los canales de distribución y por los motivos de su devolución. |
| Rescate de Productos. | Los productos farmacéuticos que han sido almacenados incorrectamente no se pueden rescatar para volver a usar. |

## 1.3 LINEAMIENTOS PARA LA REALIZACIÓN DE SERVICIOS

### 1.3.1. Función del Servicio

#### Introducción

Se presenta el Código de La Federación Internacional de la Industria del Medicamento (FIIM) como una normatividad que deben seguir las empresas de la industria farmacéutica. La FIIM es una organización no gubernamental (ONG) cuyos miembros son, *directamente o a través de organizaciones regionales*, las Asociaciones de la Industria Farmacéutica de más de 50 países de todo el mundo. La federación fue fundada en 1968 y en 1971 fue aceptada para mantener relaciones oficiales con la Organización Mundial de la Salud. Desde entonces ha sido nombrada consultora del Fondo Internacional de las Naciones Unidas para el Socorro de la Infancia (ONUDI), la Organización Mundial de la Propiedad Intelectual (OMPI), la Organización de las Naciones Unidas para el Desarrollo Industrial (UNIDO), la Conferencia de las Naciones Unidas sobre Comercio y Desarrollo (UNCTAD), el Consejo Económico y Social de las Naciones Unidas (ECOSOC) y el Consejo de Europa. La FIIM representa a la industria farmacéutica basada en la investigación y a otros fabricantes de medicamentos que deben ser dispensados mediante prescripción o bajo la supervisión de profesionales de la salud.

La promoción de medicamentos de prescripción dirigida a los profesionales sanitarios es una extensión vital del proceso de investigación y desarrollo de nuevos y mejores medios de prevención y tratamiento de las enfermedades. La promoción y la difusión de información con fines educativos asegura que los plenos beneficios de años de trabajo y de enorme inversión de ingenio y de dinero puedan llegar con prontitud a los pacientes de todo el mundo. La industria farmacéutica cree que deben definirse y respetarse niveles elevados de exigencia en todas sus actividades y está convencida de que, por lo que respecta a sus actividades de comercialización, la autodisciplina es el proceso que mejor sirve a los intereses públicos.

La adopción voluntaria del Código de la FIIM de Normas de Comercialización de Productos Farmacéuticos en 1981 estuvo en consonancia con el objetivo clave de la Federación, promulgado por sus miembros fundadores en 1968 en los Estatutos (Artículo 3), que establece que uno de los objetivos de la Federación estriba “en promover y apoyar el desarrollo continuo en la industria farmacéutica de principios y prácticas voluntariamente acordadas” y “en coordinar los esfuerzos de sus miembros hacia la realización de los objetivos mencionados”.

Cuando los Estatutos de la Federación fueron revisados en 1988, se estableció como requisito para la afiliación a la FIIM que las Asociaciones nacionales aceptaran, en nombre de sus compañías miembros, las condiciones del Código. Esta revisión del Código de la FIIM de Normas de Comercialización de Productos Farmacéuticos constituye un importante desarrollo del Código original. La Federación asume la definición de los Criterios Éticos para la Promoción de Medicamentos establecidos por la Organización Mundial de la Salud en 1988. La presente versión del Código de la FIIM se halla en concordancia con estos criterios, en tanto éstos son aplicables a nuestros miembros. Al igual que la versión original del Código de la FIIM, esta versión pretende definir criterios básicos para las prácticas de comercialización aplicables universalmente. Una proporción importante de las 52 asociaciones miembros de la FIIM tiene sus propios Códigos de Práctica. Algunos de ellos están basados en el Código de la FIIM y elaborados para adaptarse a las circunstancias y condiciones nacionales, mientras que otros derivan de versiones existentes con anterioridad a la introducción del código FIIM. Todos ellos, no obstante, incorporan los principios establecidos en el Código de la FIIM. Sería de desear que las compañías pertenecientes a la FIIM elaboraran sus propios códigos que incluyeran requisitos más específicos y reglamentos adicionales que deberían ser observados por sus empleados al llevar a cabo sus actividades promocionales.

## **Historia**

La versión original del Código de la FIIM de Normas de Comercialización de Productos Farmacéuticos fue aprobada por las Asociaciones Miembros en marzo de 1981. Al año

siguiente, se formuló un procedimiento de denuncia, al mismo tiempo que se introducía en el documento publicado una declaración acerca de la obligación de hacer constar el nombre genérico del producto en todos los anuncios.

Como consecuencia de una decisión del consejo en 1983, se ha venido publicando, en diversas ocasiones, información detallada acerca de las denuncias recibidas.

En 1988, una revisión de los estatutos incorporó un requisito previamente obligatorio, aunque informal, de que la aceptación de los términos del Código de la FIIM por parte de todas las compañías había de ser una condición para el ingreso en la Federación. Una consecuencia obligada de este desarrollo fue que la adopción de todo cambio subsiguiente en el Código y/o en sus procedimientos operativos estaría sujeto a las mismas reglas de procedimiento de un cambio en los Estatutos de la Federación.

En 1989, la versión publicada del Código incorporó un número de notas explicativas para facilitar la interpretación de la aplicación de este. Al mismo tiempo, el Código fue publicado en tres versiones idiomáticas separadas (español, francés e inglés), en lugar de la previa versión única trilingüe.

El presente Código Revisado es el resultado de las deliberaciones de un grupo de trabajo especializado, establecido por el Consejo de la FIIM, aprobado formalmente por la Asamblea General de la FIIM reunida en Malmö, Suecia, el 31 de agosto de 1994.

### **Aplicación del Código**

Además de establecer principios universales para una conducta ética de comercialización, el Código de la FIIM pretende proporcionar un Código operativo que pueda ser también utilizado en países donde no exista un Código nacional vigente con mayores exigencias. Para el establecimiento de una autodisciplina, existe un mecanismo, descrito posteriormente en el punto 1.3.2. "Requerimientos para un servicio", mediante el cual la FIIM tramita las denuncias de presuntas violaciones del Código. Cuando se comprueba la existencia de una violación del Código, el objetivo

ser proporcionada con objetividad, veracidad y buen gusto y debe estar de acuerdo con todas las leyes y reglamentos importantes. Las afirmaciones en relación con indicaciones terapéuticas y condiciones de uso deben estar basadas en evidencia científica válida e incluir información clara con respecto a los efectos secundarios, contraindicaciones y precauciones.

En la comercialización de productos farmacéuticos, se aplicarán por igual los mismos altos niveles de conducta ética en todos los países, con independencia del nivel de desarrollo de sus sistemas económico y sanitario.

## **El Código**

Temáticamente el Código lo podemos agrupar de la siguiente manera:

Código de la FIIM:

- I. Principios Generales (figura 1.3.1.1).
- II. Visitadores Médicos (figura 1.3.1.2).
- III. Simposios, Congresos y otros Medios de Comunicación Promoción (figura 1.3.1.3).
- IV. Hospitalidad y Objetos de Promoción.
- V. Material de Promoción Impreso.
- VI. Material de Promoción Audiovisual y de Soporte Informático.
- VII. Muestras Gratuitas.

## **I. Principios Generales**

### **1.0 Ámbito de Aplicación del Código.**

La expresión "producto farmacéutico" en el contexto del Código se aplica a todo producto farmacéutico o biológico primariamente destinado a ser utilizado por consejo o bajo supervisión de un profesional sanitario, para el diagnóstico, el tratamiento o la prevención de enfermedad en seres humanos o para modificar la estructura o cualquier función del cuerpo humano. Las actividades de promoción en el ámbito del

consiste en llevar a cabo, con la mayor celeridad posible, una corrección de la infracción. Las infracciones del Código son objeto de publicidad, elaborándose informes periódicos sobre la aplicación del Código.

La Federación cree que la aceptación de denuncias presentadas de buena fe, procedentes de cualquier origen, sobre cualquier aspecto del Código, ha de conducir a una aplicación del Código más eficaz que la revisión, por ejemplo, de lo que sería una pequeña muestra de anuncios que constituyen, en sí mismos, más que una modesta proporción de las actividades de *promoción de la industria*.

Una característica importante del Código es su aplicabilidad universal. Es un requisito que las disposiciones del Código se apliquen a toda compañía perteneciente a, por lo menos, una Asociación Miembro de cualquiera de los países del mundo en los que opera esta compañía. Se espera que las compañías que hayan establecido acuerdos de licencia y de representación, exijan a sus licenciatarios y representantes el respeto a las disposiciones del Código de la FIIM.

### **Compromisos de la Industria**

La industria farmacéutica internacional está comprometida a la mejora de la salud de la humanidad mediante la investigación y el desarrollo de nuevos medicamentos y la producción y comercialización de productos de calidad fiable, de acuerdo con las normas de buena práctica definidas internacionalmente.

Como parte de su compromiso con la salud, la industria tiene la obligación y la responsabilidad de proporcionar información fidedigna y educación sobre sus productos a los profesionales sanitarios, con objeto de que éstos adquieran una comprensión clara del uso apropiado de los medicamentos de prescripción.

Las actividades de promoción (prácticas de comercialización) deben estar de acuerdo con las normas éticas más elevadas y la información debe estar preparada para ayudar a los profesionales sanitarios a mejorar su servicio a los pacientes. La información debe

Código incluyen también la publicidad directa al consumidor, en aquellos países en que esté permitida por las leyes locales.

En todas las cuestiones de aplicación, interpretación y entrada en vigor de cualquier sección del Código, se entiende que gozará de precedencia la observación de las leyes, reglamentos y decisiones reglamentarias y requisitos locales.

## **2. Normas de Promoción**

El material de promoción de los productos farmacéuticos deberá ser preciso, imparcial y objetivo, además presentado de tal forma que no sólo respete los requisitos legales, sino también las exigencias éticas más estrictas y no ofenda el buen gusto. Las afirmaciones no deberán exceder el límite de lo permitido por la evidencia científica y deberán hacerse los esfuerzos necesarios para evitar toda ambigüedad.

## **3. Evidencia Científica**

La información presentada en el material de promoción deberá estar basada en una evaluación actualizada de la evidencia que sea científicamente válida y no deberá dar una impresión incorrecta o engañosa. La información para la prescripción estará apoyada sobre esta evidencia científica y concordará con la autorización de registro local del producto o, en los casos en que ésta no exista, deberá estar de acuerdo con una autorización concebida por una autoridad de registro de un país industrializado. Los datos científicos que justifican las afirmaciones y recomendaciones para el uso deberán ser facilitados, a petición, a los profesionales sanitarios.

## **4. Datos de Seguridad**

Se deberá guardar especial cuidado para que la información esencial referente a la seguridad de los productos farmacéuticos, por ejemplo, contraindicaciones, precauciones y efectos secundarios, sea comunicada de forma apropiada y consistente, sujeta a las prácticas legales, administrativas y médicas de cada país. El término "seguro" no debe ser utilizado sin una calificación.

*Deberá ser comunicada con prioridad a la autoridad sanitaria nacional competente toda información, sustentada en datos, sobre reacciones adversas graves o inesperadas asociadas a productos farmacéuticos.*

### **5. Promoción Encubierta y Engañosa**

El material de promoción, como envíos por correo y anuncios en revistas médicas, no *deberá disimular su naturaleza real.*

Los estudios de poscomercialización y de vigilancia deberán ser llevados a cabo con *una intención fundamentalmente científica o educativa.* Estos estudios no deben emprenderse simplemente como un procedimiento para promocionar un producto o para influenciar a los médicos, desprovisto de finalidad científica o educativa.

### **6. Comunicaciones antes del Registro**

Ningún producto farmacéutico debe ser promocionado para su uso en un determinado país hasta que la aprobación preceptiva para su comercialización en determinado uso haya sido concedida en dicho país. Este precepto, sin embargo, no presupone una limitación al derecho de que la comunidad científica y el público estén plenamente informados acerca del proceso médico y científico. No pretende restringir el intercambio total y adecuado de información científica relacionada con un producto farmacéutico, entre la cual se cuenta la divulgación apropiada de los hallazgos de la investigación en los medios de comunicación científicos o generales y en congresos científicos. Como tampoco trata de impedir la revelación, a los accionistas y a otras personas interesadas, de hechos relativos a cualquier producto farmacéutico, que puede ser necesaria o deseable, de acuerdo con la ley o cualquier otra reglamentación.

### **7. Comunicaciones al Público**

En aquellos países en que la ley permite entrar en comunicación directa con los pacientes en relación con sus medicamentos de prescripción, toda información deberá ser precisa, imparcial y no engañosa.

Cuando las compañías presten su ayuda a la ejecución de programas de concientización sobre enfermedades dirigidos al público y a pacientes, destinados a satisfacer las demandas sociales de mayor información y a aumentar la comprensión pública de la prevención de la enfermedad, tales actividades deberán respetar los más elevados niveles de exactitud y apoyar el papel de los profesionales sanitarios.

## **8. Procedimientos en las Compañías**

Las comunicaciones de promoción deberán ser sometidas a visado médico y, cuando sea pertinente, contar con el visto bueno de una persona responsable antes de su difusión. Esta persona responsable deberá poseer las necesarias calificaciones científicas o sanitarias.

Las compañías deberán establecer y mantener procedimientos apropiados para asegurar la total adherencia a los pertinentes códigos nacionales e internacionales y revisar y controlar todas sus actividades y materiales de promoción.

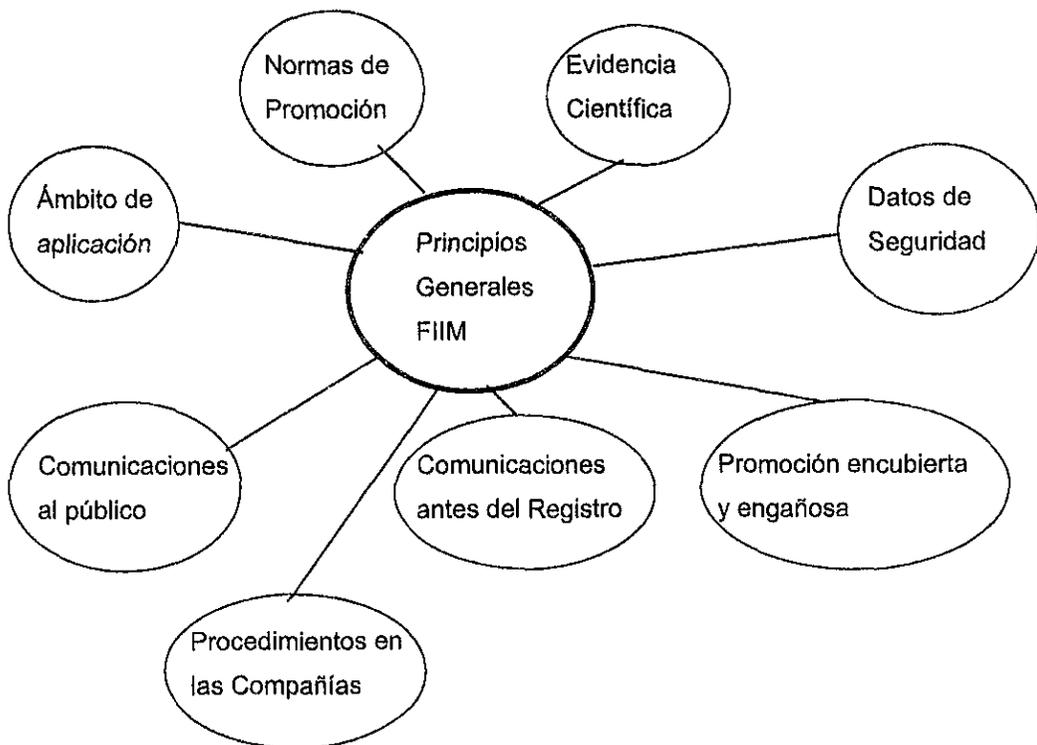
## **II. Visitadores Médicos**

### **1. Formación y Responsabilidades**

Los visitadores médicos deberán estar adecuadamente formados y poseer suficientes conocimientos médicos y técnicos para presentar la información sobre los productos de sus compañías de manera precisa, responsable y ética. Deberán informar a su compañía acerca de los contactos mantenidos con los profesionales sanitarios y transmitir los informes que reciban acerca del uso de productos y, en particular, las comunicaciones de efectos secundarios.

### **2. Responsabilidad de la Compañía**

Toda compañía asumirá la responsabilidad, de acuerdo con el Código, de corregir las infracciones que resulten del mal comportamiento o de la desfiguración de hechos de cualquier visitador médico.



**Figura 1.3.1.1. Principios Generales del Código de la Federación Internacional de la Industria del Medicamento.**

### **3. Remuneración**

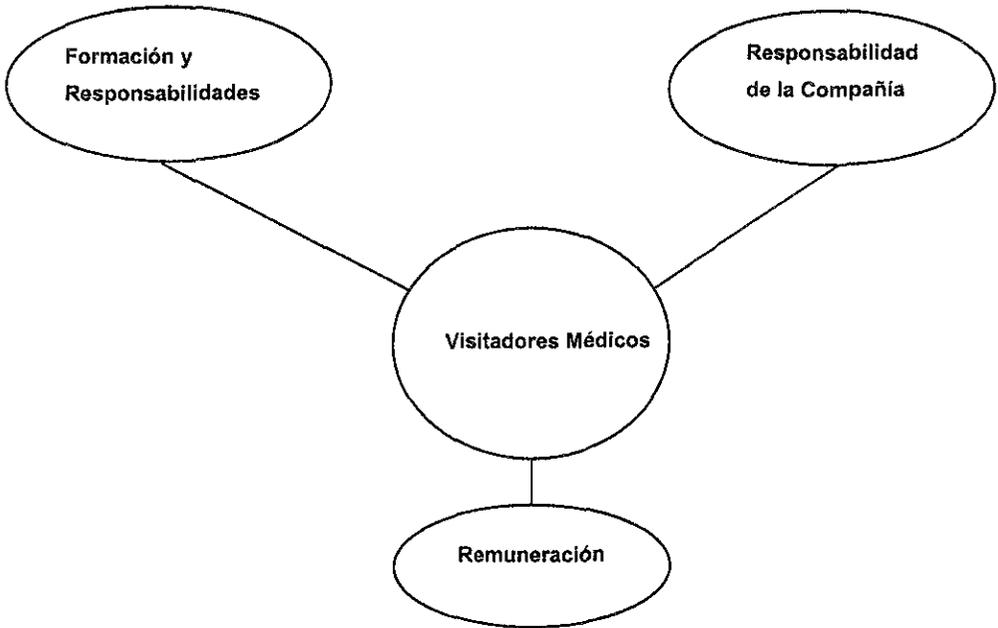
El sistema de remuneración de los visitantes médicos no debe estar concebido de modo que pueda influenciar adversamente la correcta prescripción de productos farmacéuticos por el médico.

### **Simposios, Congresos y Otros Medios de Comunicación Oral**

#### **1. Objetivos**

Los simposios, congresos y actividades similares son indispensables para la difusión de conocimientos y experiencia. Los *objetivos científicos* deberán constituir el *foco principal*

en la organización de tales reuniones y las invitaciones y los actos sociales no deberán ser incompatibles con tales objetivos.



**Figura 1.3.1.2. Políticas que deben seguir los visitadores médicos.**

## **2. Patrocinio**

Cuando una compañía farmacéutica o una asociación patrocina un simposio, congreso u otra actividad de carácter médico, sanitario o educativo:

- a) La existencia del patrocinio de la compañía o de la asociación deberá ser dada a conocer con claridad antes, durante y después de la reunión y también en sus actas. El material impreso, audiovisual o con soporte informático, resultante de tales reuniones, deberá reflejar con fidelidad las presentaciones y las discusiones;
- b) Las recepciones u otros actos sociales y los obsequios ofrecidos a los profesionales sanitarios deberán estar supeditados al objetivo principal de la reunión y su valor deberá ser mantenido a un nivel modesto;

- c) cualquier subvención a los profesionales sanitarios para participar en la reunión no deberá estar condicionada a ninguna obligación de promocionar ningún producto farmacéutico;
- d) si el programa está acreditado para educación médica posgraduada por una organización médica o profesional, la responsabilidad del contenido del programa recaerá en la organización responsable de obtener la acreditación de la reunión y la ayuda de la industria deberá ser puesta de manifiesto;
- e) el pago de honorarios razonables y el reembolso de gastos personales, incluyendo el viaje, a los presentadores y conferenciantes es una práctica normal y aceptable;
- f) las compañías no deberán pagar gastos de viaje a las personas que acompañan a los profesionales sanitarios invitados.

#### **IV. Hospitalidad y Objetos de Promoción**

##### **1. Hospitalidad**

No deberán ofrecerse beneficios financieros o materiales, incluidas las invitaciones inapropiadas, a los profesionales sanitarios, con objeto de influenciarlos en la prescripción de productos farmacéuticos.

##### **2. Objetos de Promoción**

La entrega de objetos de promoción de escaso valor, ofrecidos gratuitamente, podrá ser permitida siempre que estos objetos estén relacionados con el trabajo del profesional sanitario y/o aporten un beneficio para el paciente.

##### **3. Material Educativo**

Podrán entregarse a los profesionales sanitarios libros de texto o de referencia, publicaciones y otro material educativo, en tanto posean una función genuinamente educativa.

## **V. Material de Promoción Impreso**

### **1. General**

El material promocional impreso deberá ser presentado de forma que resulte legible. La base científica y la presentación de la información del producto deberá estar en conformidad con los principios generales expuestos en la Sección I del Código y, en su caso, con la información autorizada del producto.

### **2. Todos los Anuncios**

Todos los anuncios impresos deberán incluir:

- a) el nombre del producto (generalmente el nombre de la marca);
- b) los principios activos, utilizando el nombre aprobado, si éste existe;
- c) el nombre y la dirección de la compañía o de su representante responsable de la comercialización del producto.

#### **2.1. Anuncios Completos**

Anuncios completos son los que contienen afirmaciones de promoción relativas al uso de los productos. Además de los requisitos expresados en el punto V.2, los anuncios completos deberán también incluir información para la prescripción consistente, como mínimo, en:

- a) la indicación o indicaciones aprobadas junto con la dosificación y modo de uso;
- b) una sucinta declaración de las contraindicaciones, precauciones y efectos secundarios.

#### **2.2. Anuncios de Recuerdo**

En ausencia de reglamentación local, los anuncios abreviados o de recuerdo se definen como anuncios cortos que contienen tan sólo una simple declaración de indicaciones para designar la categoría terapéutica del producto.

Para los anuncios de recuerdo se aplicarán los requisitos del párrafo V.2, pero se podrá omitir la información para la prescripción, siempre que:

a) se incluya una fase que indique claramente que existe información disponible a petición de los profesionales sanitarios.

### **3. Referencias Bibliográficas**

Cuando el material de promoción impreso mencione estudios publicados, se deberán incluir las referencias bibliográficas correspondientes. Las citas de la literatura médica o de comunicaciones personales no deben alterar o distorsionar el sentido que quiso darles su autor o investigador clínico, ni el significado del trabajo o del estudio en que se basan.

### **4. Frecuencia y Volumen**

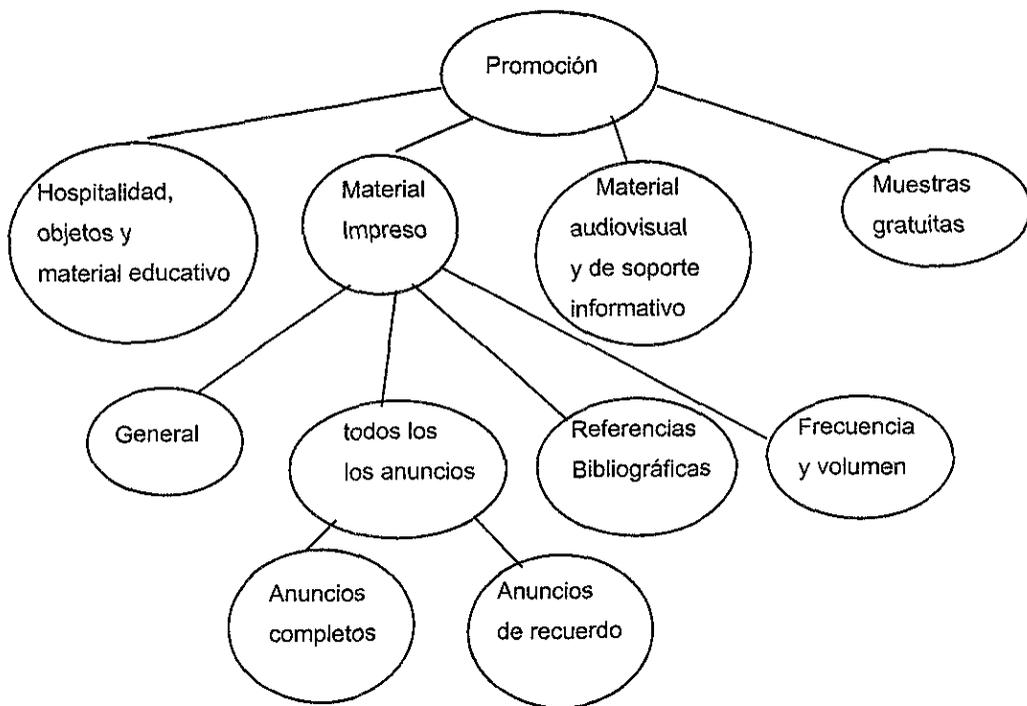
La frecuencia y el volumen de los envíos por correos de material impreso a los profesionales sanitarios deben ser razonables. Las peticiones de los médicos para que sus nombres se excluyan de las listas de envío deben ser respetadas, si bien deberán mantenerse las listas completas de direcciones para poder enviar información importante acerca de reacciones adversas, precauciones, advertencias, etc.

### **VI. Material de Promoción Audiovisual y de Soporte Informático**

El material de promoción presentado a través de estos medios deberá cumplir los requisitos fundamentales aplicables al material impreso, según se establece en la sección V. La información para la prescripción podrá ser omitida, siempre que se proporcione a la audiencia una información completa del producto.

### **VII. Muestras Gratuitas**

Se podrán entregar muestras gratuitas, siempre identificadas como tales, a las personas facultadas para prescribir o dispensar, con el objeto de que se familiaricen con los productos, que adquieran experiencia con el producto en su práctica, o a petición.



**Figura 1.3.1.3. Lineamientos generales de la promoción.**

**Adición:**

**Información para los Países en Vías de Desarrollo**

Teniendo en cuenta que muchos países en vías de desarrollo no se hallan plenamente informados acerca de las indicaciones, contraindicaciones, efectos secundarios, etc. de ciertos medicamentos que han sido aceptados en los países industrializados, la FIMM ha ofrecido ayuda para proporcionar gratuitamente a los Departamentos de Salud de los gobiernos de países en desarrollo, ejemplares de ediciones actualizadas de compendios.

## 1.3.2. Requerimientos para un servicio

El procedimiento de las denuncias está esquematizado en la figura 1.3.2.1 para observar el mecanismo que sigue en las denuncias.

### 1. Verificación y Tramitación de las Denuncias

Cuando se presente la denuncia de una presunta infracción del Código de la FIIM al Secretariado de la FIIM, en primer lugar será validada para asegurar que:

- a) *parece referirse a un asunto real y ha sido presentada de buena fe;*
- b) *existe información suficiente para permitir que la denuncia siga su curso.*

Una misma denuncia podrá abarcar más de un "caso", es decir, la denuncia podrá referirse, por ejemplo, a varios anuncios de diferentes compañías y/o de diversos productos. Cada "caso" es tratado separadamente por la FIIM bajo la referencia de una denuncia general. La primera acción, en cada caso, consistirá en identificar:

- a) *la compañía involucrada en el caso y la oficina principal o la compañía matriz y su localización, en caso de ser diferente;*
- b) *la Asociación Miembro a la que pertenece la casa central;*
- c) *la Asociación Miembro (si existe) del país en donde ha ocurrido la presunta infracción.*

Un resumen de cada caso comprendido en la denuncia, con copia de cualquier evidencia que apoye la denuncia (por ejemplo, una copia del anuncio que se ha denunciado como infractor del código) será enviada a la Asociación o Asociaciones Miembros, referidas anteriormente, con copia a la dirección de la compañía, tanto a nivel local como central. Se solicitará a las Asociaciones Miembros que consulten con la compañía y envíen su informe a la FIIM con los resultados de su investigación sobre el caso.

## **Compañías no Miembros de la FIIM**

Cuando un caso se refiera a una compañía que no pertenece (ni localmente, ni a través de su compañía matriz) a una Asociación Miembro de la FIIM, el caso no podrá ser procesado de modo formal.

Si la FIIM tiene una Asociación Miembro en el país donde ha ocurrido la presunta infracción, se solicitará a esta Asociación que trate el asunto, informalmente, con la compañía no miembro y le llame la atención acerca de la necesidad de respetar normas como las establecidas en el Código de la FIIM.

En aquellos países en los cuales no exista ninguna Asociación Miembro de la FIIM, se enviará información sobre el caso a la autoridad de registro local.

## **2. Plazos**

La carta remitida a la o a las Asociaciones Miembros indicará el plazo en que debe darse respuesta sobre el caso o casos bajo investigación. Este plazo será normalmente de 60 días naturales a partir de la recepción de la documentación.

En circunstancias excepcionales, se podrá permitir una prolongación de este plazo. Por ejemplo, cuando la Asociación Miembro del país en que ha ocurrido la presunta infracción proyecte someter el caso a los procedimientos de su Código nacional para los que existe un calendario definido. En tal eventualidad, el denunciante será informado de la extensión de los plazos.

## **3. Resolución**

Cuando la compañía admita haber infringido el Código, deberá aportar información sobre las medidas que hayan sido adoptadas o piense adoptar para corregir la infracción.

Cuando las alegaciones sean rechazadas, deberán especificarse con claridad las razones del rechazo y, cuando sea necesario, se aportarán datos que lo avalen (por

ejemplo, evidencia científica que demuestre las afirmaciones que han sido cuestionadas).

La respuesta o respuestas de la o de las Asociaciones Miembros será o serán utilizadas por la FIIM para preparar la resolución del caso, que será comunicada al denunciante, con copias a la compañía y a la o a las Asociaciones Miembros.

#### **4. Publicación de la Resolución**

Cuando una denuncia se haya substantivado y se haya comprobado la existencia de una infracción del Código, se hará pública inmediatamente la información que identifica la compañía implicada y el nombre del denunciante.

*Asimismo, la información podrá hacerse pública en aquellos casos en que la compañía no conteste en el plazo establecido.*

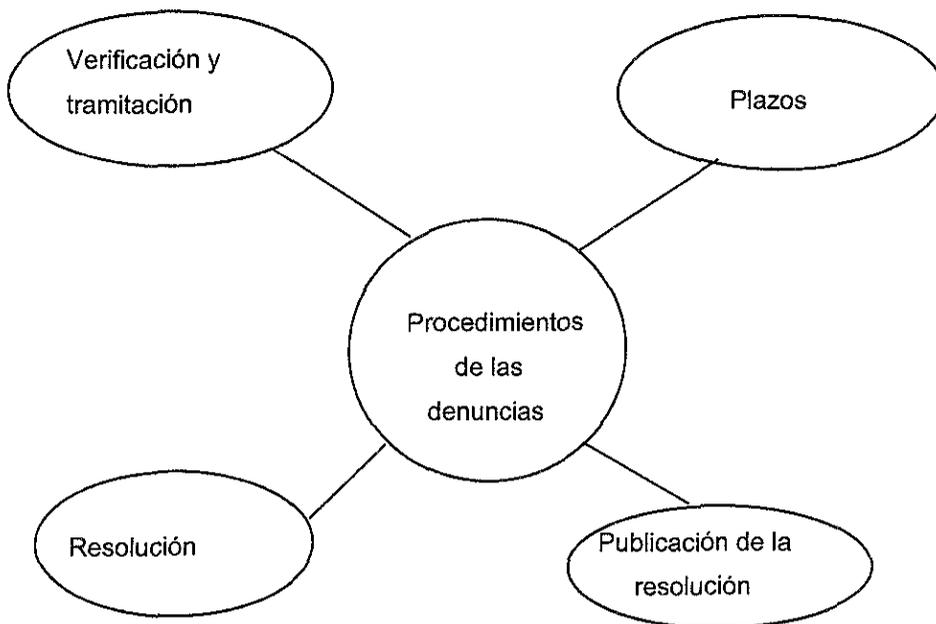
#### **Informes de Situación**

Informes de situación sobre el Código de la FIIM, que resuman todas las denuncias recibidas, serán publicados periódicamente y puestos en amplia circulación por los departamentos gubernamentales de salud, prensa técnica y revistas médicas importantes y Asociaciones Miembros de la FIIM.

En estos informes:

- a) *se nombrará a las compañías que han infringido el Código, describiéndose la naturaleza de la infracción y con mención del nombre del producto cuando ello sea apropiado;*
- b) *se informará sobre los casos que fueron considerados inválidos o los que no se juzgaron pertenecientes al ámbito de aplicación del Código, por ejemplo, aquellos en los que las compañías implicadas no eran miembros de la FIIM;*
- c) *se describirán brevemente los casos que fueron rechazados - porque no se encontró infracción del Código - pero sin mención del nombre del producto o de la compañía;*

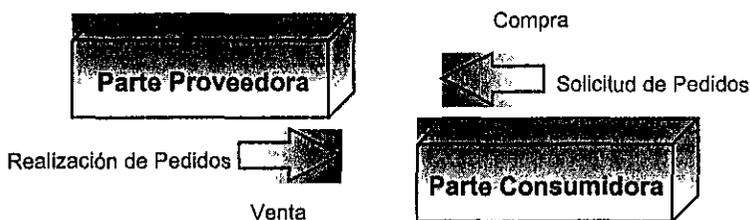
d) Se aportarán estadísticas acumulativas sobre la naturaleza y el resultado de los casos.



**Figura 1.3.2.1. Procedimientos de las denuncias en el Código de la FIIM**

## 1.4. REQUERIMIENTOS PARA LA REALIZACIÓN DE UN PEDIDO.

Para comprender de mejor forma todo lo que gira en torno a un pedido, es necesario tocar brevemente cada una de las partes que participan durante el proceso de realización del mismo, la parte proveedora o vendedora y la parte consumidora o compradora, para después enfocarnos a los requerimientos que resultan indispensables para su realización. En la Figura 1.4.1 mostramos ambas partes.



**Figura 1.4.1. Esquema de las partes involucradas.**

Comenzaremos con la parte que funge como proveedora, es decir, aquella entidad que se encarga de suministrar con base a una venta, sus productos, llámese un laboratorio, una farmacia, etc. Esta parte necesita tener amplio conocimiento sobre la dinámica del flujo de sus insumos, esto, para garantizar un suministro constante. Lo que quiere decir que se debe tener conocimiento de la información que fluye tanto de una petición para reaprovisionar sus almacenes como de una licitación hecha por una entidad externa -un consumidor- que pudiera tener impacto sobre su capacidad de producción así como las operaciones contables que se generen por consecuencia de las ventas, sin dejar fuera la dinámica de distribución de los productos. Se comienzan a ver por tanto, un conjunto de elementos de producción, de control de calidad, de almacenaje, de distribución, hasta llegar al lugar propio de consumo, que sólo si se comprenden adecuadamente podrán llevar a esta parte a cumplir con su papel; y sin perder de vista que como es una entidad que esencialmente se dedica a la venta, tiene que ser regulada por leyes vigentes tanto nacionales como internacionales, en caso que se dedique a la exportación. Es aquí, donde la ayuda de alguna herramienta que permita tener un mejor

control del flujo de la información puede ser la diferencia entre un proveedor y otro. Esta herramienta puede tener un efecto más evidente al facilitar un control centralizado con un suministro descentralizado, es decir, se puede dar el caso de tener que proveer productos a regiones distantes del centro de almacenaje y por tanto del de producción, por lo que es necesario descentralizar el o los almacenes con que se cuente, para garantizar una entrega oportuna. También resultaría evidente por el manejo de las normas de suministros, las cuales aplicarían para cualquier centro de distribución. No queremos decir que este tipo de control sea el que más convenga ya que también puede tener sus defectos, sobre todo por dirigir todos los pedidos hacia un punto central que puede estar sujeto a fallas. Sin embargo, ambos esquemas pudieran bien complementarse para obtener el mayor beneficio, ajustándose inclusive en grupos de insumos que se apeguen más a cada uno. En la Figura 1.4.2 mostramos el flujo de productos para un Proveedor.

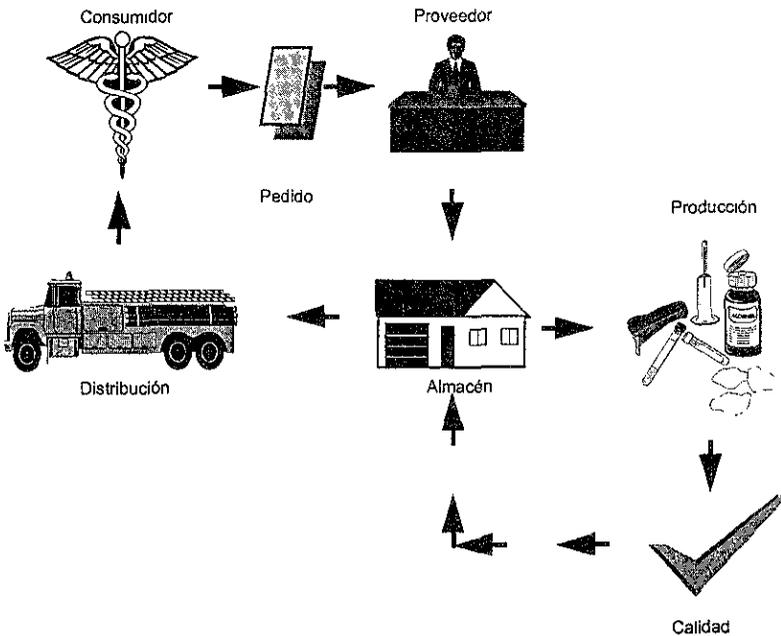
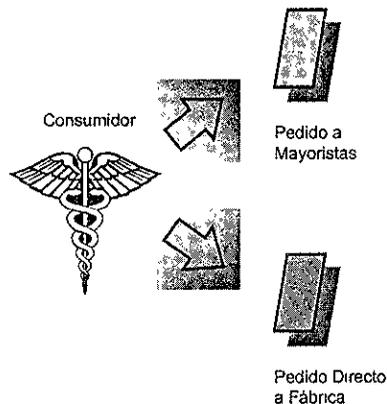


Figura 1.4.2. Flujo de productos para el Proveedor.

La otra parte que funge como solicitante y consumidora de los productos que el proveedor le proporciona, con base a una compra, llámese un hospital de asistencia pública o privado, una farmacia, etc. Es la encargada de solicitar un pedido que atienda a sus necesidades de abastecimiento y control interno. Hay que hacer notar que dentro de la misma se encuentran implícitos una serie de departamentos que pueden existir o no como entidades separadas una de otra, como el de compras que sería el punto de contacto indicado para realizar un pedido, de un almacén y hasta de una farmacia propia, en el caso de algunos hospitales. Así, aunque sabemos que el proceso de aprovisionamiento involucra el trabajo conjunto de una serie de departamentos los *agruparemos a todos para hacer referencia a ellos como una sola parte, pero sin perder de vista las funciones propias de cada uno de ellos.* Por lo que tenemos que mencionar primeramente a uno de los más importantes y que es el almacén, ya que éste tiene como responsabilidad vigilar el consumo de los artículos en general y sobre todo de aquellos cuya importancia sea evidente, de esta forma se puede llegar a detectar un consumo inadecuado ya sea por un pedido fuera de lo normal o un consumo desmedido dentro del mismo almacén. Tiene también la responsabilidad de realizar las requisiciones necesarias para mantener las reservas por encima de lo mínimo reglamentario. En lo que respecta al departamento de compras resalta su operación en forma centralizada para la adquisición de medicamentos, equipos, etc. Ya que debe converger hacia un lugar donde se lleva el control de los recursos financieros de la entidad. Mantiene además un inventario físico de los equipos, se encarga de revisar el almacenamiento o tiene conocimiento del mismo y algo muy importante, señala las especificaciones para la compra de materiales; entre otras cosas. Este departamento basa todas sus operaciones en órdenes de pedido que le hacen llegar de otros departamentos, las cuales una vez autorizadas pueden retornarse como copia al departamento que las ordenó, otra copia permanece en el mismo departamento de compras y el original es el que se manda al proveedor. Una vez que se recibió la mercancía en los términos especificados es turnada la factura al departamento de contabilidad para su registro y por supuesto su pago. Como se dijo con anterioridad, los departamentos -si existen- tienen funciones propias aunque necesariamente ligados uno al otro, sin embargo, puede darse el caso en que esas funciones se concentren en

uno sólo pero sin perder la mecánica básica que nos lleve a un abastecimiento efectivo. Es evidente establecer una política para el suministro, que sea lo más clara posible, evitando así, interpretaciones incorrectas; que maximice el rendimiento de los recursos financieros y humanos. Es así como surgen las ideas tales como el consumo a mayoristas y el consumo directo en fábrica de ciertos productos, que tenderán a reducir considerablemente los costos ya que se obtendrían descuentos por volumen y se facilitaría el control por la disminución en el número de pedidos, aunque se tendría que contemplar la existencia de un almacén de dimensiones considerables por lo que se debe de encontrar un punto de equilibrio entre los pedidos con base a volumen y los pedidos con base al surgimiento de la necesidad de un suministro constante, siendo necesario para ambos casos, contemplar las expectativas a corto, mediano y largo plazo del consumo de cada producto para la posible calendarización de los pedidos tomando en cuenta diversos factores como puede ser el presupuesto asignado para un cierto período, lo que nos llevaría a determinar prioridades de abastecimiento para cada producto o grupo de los mismos. En la Figura 1.4.3 se muestra la división de proveedores que puede contemplar un Consumidor.



**Figura 1.4.3 División de proveedores con base al tipo de productos ofrecidos.**

Como podemos observar, ambas partes manipulan para bien o para mal una amplia gama de variables que de una u otra manera las llevarán a satisfacer sus necesidades,

siendo obligación de ellas llevar un control adecuado de los procesos internos que les faciliten llegar al fin para el que fueron creadas. Pero debe quedar claro que como partes complementarias que encajan en un esquema económico deben estar regidas por una serie de leyes que regulen su interacción. Así como también una serie de procedimientos que concilien intereses comunes y que son aplicables dentro del marco de esas leyes.

### **1.4.1. Procedimientos para la Realización de un Pedido**

Podemos diferenciar los procesos de proveeduría de acuerdo a ciertas características del pedido, como son las compras con un valor relativamente bajo y que no están condicionadas por ningún tipo de requisito de competencia y las compras con un valor considerablemente superior a las primeras y que para cuya realización es necesario lanzar una convocatoria de concurso o licitación.

Compras con valor bajo   ⇒   Realización de cotización  
Compras con valor alto   ⇒   Realización de concurso o licitación.

Cabe hacer mención que el volumen no es una característica de peso que determine el valor comercial de un pedido ya que uno de escaso volumen puede llegar a tener un valor comercial mucho más elevado que uno de dimensiones mucho mayores, esto por supuesto, debido al valor en el mercado de el (los) producto (s) involucrado (s) o del nivel de urgencia con la que se requiera surtirlo. En gran parte de las instituciones de asistencia médica se trata de evitar al máximo la realización de compras con bajo valor comercial ya que en ocasiones llegan a convertirse en la mayor parte e involucran a gran cantidad de proveedores, ocasionando saturación en el control de los pedidos y en consecuencia en un punto crítico del proceso de proveeduría de suministros, sin embargo no dejan de ser menos importantes ya que suelen representar una parte importante en los ingresos de la parte proveedora. Además de que no requieren de un proceso de concurso, sino mas bien de una evaluación de cotizaciones, dependiendo de la institución, tomadas de un directorio de proveedores que genera y actualiza

continuamente. La selección de proveedores depende de la institución, es decir, para algunas esta selección requiere de un sondeo regular de cada uno de los productos que cada uno de sus proveedores distribuye de acuerdo a sus precios y/o condiciones de venta, a través de cotizaciones. Para otras, existen procedimientos que buscan simplificar lo anterior por medio de contratos abiertos o pedidos abiertos que constituyen una fórmula eficaz de mantener a ambas partes en un acuerdo común respecto a la lista de productos que el proveedor maneje en este contrato. En la figura 1.4.4 se muestra el esqueleto de un Pedido Abierto.

| <b>ISSSTE</b>   |                         |   |          |                 |  |                                   |
|---|-------------------------|---|----------|-----------------|--|-----------------------------------|
| SUBDIRECCIÓN GENERAL DE PROTECCIÓN AL SALARIO<br>SUBDIRECCIÓN DE ABASTO |                         | PEDIDO ABIERTO: 1543/18/01/016<br>FECHA DE AUTORIZACIÓN: 20/10/1997<br>FECHA DE EMISIÓN: 01/01/1997 |          |                 | PEDIDO                                   |                                   |
| PROVEEDOR: CONOCIDO.  |                         | R.F.C. : XXXX-000000-XXX  |          |                 |  |                                   |
| DOMICILIO: CONOCIDO.  |                         |   |          |                 |  |                                   |
| CIUDAD: MÉXICO      ESTADO: MÉXICO                                      |                         |   |          |                 |  |                                   |
| C.P. 0000      TELS. : 000-00-00  |                         |   |          |                 |  |                                   |
| LINEA: 18      FARMACÉUTICA   |                         |   |          |                 |  |                                   |
| PARTIDA<br>CÓDIGO   | DESCRIPCIÓN             | MEDIDA  | CANTIDAD | PRECIO<br>COSTO | PRECIO VTA.<br>UNITARIO<br>METROPOLITANO | PRECIO VTA<br>UNITARIO<br>FORÁNEO |
| 01  | Productos Farmacéuticos |   |          |                 |  |                                   |
| 01 0001   | Producto 1              | CAJA  | ##       | \$              | \$                                       | \$                                |
| 01 0002   | Producto 2              | CAJA  | ##       | \$              | \$                                       | \$                                |
| 01 0003   | Producto 3              | CAJA  | ##       | \$              | \$                                       | \$                                |
| 01 0004   | Producto 4              | CAJA  | ##       | \$              | \$                                       | \$                                |
| 01 0005   | Producto 5              | BOLSA   | ##       | \$              | \$                                       | \$                                |
| 01 0006   | Producto 6              | CAJA  | ##       | \$              | \$                                       | \$                                |
| Firma 1   |                         | Firma 2   |          |                 | Firma 3                                  |                                   |

**Figura 1.4.4 Esqueleto de un Pedido Abierto.**

La formalización de la operación se hace generalmente por la emisión de una orden de suministro o remisión oficial, que es un documento cuyo propósito y contenido puede variar de una institución a otra, mientras que en una significa un documento de pago en

otra puede representar un documento de cotejo y en otra una mezcla de ambas. Es necesario que dicha documentación contenga una serie de datos, como los que siguen: el nombre del organismo que emite la orden, la fecha, un número de identificación, el nombre y dirección del proveedor, lugar donde serán entregados los suministros, la fecha en que deben entregarse los abastecimientos, el medio de transporte, cuándo deben entregarse en un lugar distinto a aquel en que se encuentra el proveedor, la descripción de la mercancía o del material solicitado, su unidad de medida, su precio unitario, la cantidad requerida, su valor, su nombre, las firmas y los cargos de las personas que autorizan la orden de compra, la cuenta a la que debe cargarse el valor de la orden, así como los documentos que deben acompañar al despacho de la mercancía. Es importante tomar en cuenta el número de copias que deban hacerse con el objeto de que todas las partes ligadas conozcan el movimiento. El paso que sigue es que la parte consumidora reciba los diferentes productos ya sea mediante su departamento de almacén o uno, cualesquiera que sea el encargado de validar el contenido de los pedidos ya que en ocasiones es necesario recurrir a personal especialmente adiestrado para validar las especificaciones con las cuales se hizo la requisición. Es necesario también tomar en cuenta las cantidades, precios y marcas de los artículos que se reciben, de tal manera que si existe conformidad o no, la persona responsable pueda actuar oportunamente. Así cuando las especificaciones y las cantidades no corresponden al pedido, se debe hacer el rechazo de inmediato, en estos casos las compañías proveedoras exigen normalmente que se firme la llamada forma de remisión, con la cual se inician todos los trámites en relación con la oficina responsable de los pagos, que a su vez da conocimiento al departamento de contabilidad.

*Representan un proceso aparte, aquellas compras que requieren un concurso o licitación pública. En la figura 1.4.5. tenemos el ejemplo de una de ellas.*

**INSTITUTO NACIONAL DE ENFERMEDADES RESPIRATORIAS**

DIVISION DE RECURSOS MATERIALES  
LICITACION PUBLICA INTERNACIONAL  
CONVOCATORIA 007

En observancia a la Constitución Política de los Estados Unidos Mexicanos en su artículo 134, y de conformidad con la ley de Adquisiciones y Obras Públicas, se convoca a los interesados en participar en la(s) licitación(es) de carácter internacional para la contratación de material de curación, de conformidad con lo siguiente

| No. licitación  | Costo de las bases                    | Fecha límite para adquirir bases | Junta de aclaraciones  | Presentación de proposiciones y apertura técnica | Acto de apertura económica | Plazo de entrega  |
|-----------------|---------------------------------------|----------------------------------|------------------------|--|----------------------------|---|
| 12223001-010 97 | \$400<br>Costo en compraNET.<br>\$300 | 2/02/98                          | 2/02/98<br>12 00 horas | 9/02/98<br>12 00 horas                           | 23/02/98<br>12 00 horas    | Según calendario que aparece en el anexo 5 de las bases |

| Partida | Clave CABMS | Descripción                                 | Cantidad | Unidad de medida |
|---------|-------------|---|----------|------------------|
| 1       | 000000000   | Bata para cirujano desechable               | 3 000    | Pieza            |
| 2       | 000000000   | Equipo para vaciarse o/medidor de volúmenes | 8 000    | Pieza            |
| 3       | C48000072   | España gasa                                 | 2 250    | Pacúete          |
| 4       | 000000000   | Guantes para exploración                    | 160 000  | Pieza            |
| 5       | 000000000   | Equipo para vaciarse s/aguja                | 12 000   | Pieza            |

- \* Las bases de la licitación se encuentran disponibles para consulta y venta en Internet <http://compranet.gob.mx> o bien en Catzada de Tlalpan número 4502, colonia Sección XVI, código postal 14080, Tlalpan, Distrito Federal, con el siguiente horario: 9:00 a 14:00 horas
- \* La forma de pago es en convocatoria, cheque certificado o de caja a nombre del Instituto Nacional de Enfermedades Respiratorias, o en efectivo En compraNET: depósito en la cuenta 400 8513 487 de Bital, abierta a nombre de la Tesorería de la Federación
- \* Las bases de la licitación estarán a disposición de los interesados hasta siete días previos al acto de presentación de proposiciones y apertura técnica
- \* Los eventos se llevarán a cabo en las instalaciones del Instituto
- \* El idioma en que deberán presentarse las proposiciones será español
- \* La moneda en que deberán cotizarse las proposiciones será peso mexicano
- \* La entrega de muestras será el día 8/02/1998, en el almacén de farmacia de 10:00 a 13:00 horas
- \* El lugar de entrega será en almacén de farmacia, los días lunes a viernes en el horario: 9:00 a 13:00 horas
- \* Las condiciones de pago serán: 20 días

**Figura 1.4.5 Ejemplo de una licitación**

Existen reglamentaciones rígidas y precisas en las que se establecen las compras que entran en la categoría de licitación pública. Una vez elaborado un plan, un calendario de compras y se encuentre dentro del presupuesto aprobado para un período, se inicia el concurso en sí, mediante una inspección de acuerdo a cotizaciones que no comprometen a ninguna parte proveedora a adquirir algún compromiso con la parte consumidora, sino más bien, esta última se encarga de realizar un estudio de cada uno de los futuros proveedores así como en su conjunto pero sin promover un ambiente de competencia que propicie la alteración de sus precios antes de emitir la licitación en sí. Existen otros aspectos importantes, como son el ajuste de la compra que contempla las variaciones imprevistas del precio de los productos. A esto le sigue la elaboración de un proyecto de compra, en muchos casos conocido como Norma, que antecede al compromiso correspondiente que está en función de los requerimientos reglamentarios

de cada sistema de salud. Es entonces cuando se elabora la invitación para concursar a los posibles proveedores, esta invitación contiene las condiciones de la venta que se desea hacer, y que generalmente contiene lo siguiente: el número de la invitación, que sirve para llevar un control y es una forma de identificación para distinguirla de otras futuras, el organismo que hace la invitación, o en su caso la persona asignada para cualquier aclaración acerca de la misma, la fecha en que se emite la invitación, la dirección de la oficina de proveeduría, la descripción de los materiales o equipo requerido; esta es la parte más importante de la invitación, y en ella se debe indicar en forma clara la calidad de los artículos, de acuerdo con las descripciones o especificaciones de las que se trató anteriormente; éstas deben hacerse en lenguaje claro y sencillo, y las frases y palabras deben ser cortas y simples; en muchos casos se debe hacer referencia de planos y especificaciones que se anexan en la descripción del artículo de referencia. Debe indicarse también la unidad de medida para cada requisición, en la práctica comercial, y la que más se ajusta a las necesidades del servicio. Debe especificarse también la forma de presentación deseada, con el objeto de asegurar la uniformidad de los envases y facilitar el almacenamiento; así por ejemplo, se piden unidades que tengan 10, 50, 100 ó 1000 artículos, describiendo si son en cajas, tubos, latas, etc. Los líquidos pueden pedirse en frascos por dosis o en litros cúbicos o alguna medida equivalente; los productos inyectables pueden pedirse en dosis individuales o dosis múltiples. Fecha y forma de entrega de la siguiente manera: la fecha o plazo de entrega debe establecerse con base al tiempo que normalmente se usa para entregar un pedido; en los casos en que sea urgente la entrega, deberá indicarse que éste es un factor importante de un pedido. Cabe hacer notar que en prácticamente todos los casos se trata de que el fabricante o vendedor asuma parte o la totalidad de los costos de almacenaje y los riesgos por desuso; también pueden disminuirse estos problemas estableciendo un sistema de entrega parcial, pero siempre y cuando que asegure que los suministros serán recibidos oportunamente, para lo cual debe incluirse en la invitación al concurso el calendario correspondiente. Otro punto importante es establecer el lugar de la entrega, ya que en muchos casos se deben aprovechar los servicios de distribución del fabricante para que se haga responsable de la entrega de los artículos a todos aquellos lugares que se

conozcan como periférico. La invitación también debe especificar lo referente a los costos de desembarque en caso de que existan, la manipulación y el transporte hasta el lugar de recepción, y si es que se debe pagar algún tipo de impuestos. Se debe especificar el descuento por cantidad y en caso de lo que se conoce como pronto pago, en algunos otros casos en que existan envases que pueden ser devueltos y aprovechados implica que en caso de devolución se deberá realizar la disminución del precio pactado. En los contratos a largo plazo que involucren variaciones en el precio por cualquier causa generalmente se beneficia al comprador. La fecha más lejana en que se considera la oferta válida es un dato importante en caso de que se plantee una renegociación. Para efectos fiscales la invitación debe tocar el caso de las compras por adelantado, ya que las ofertas deberán extender su vigencia hasta el inicio del siguiente período fiscal, para lo cual es conveniente solicitar que los participantes del concurso sostendrían el mismo precio en caso de que se decida aumentar la cantidad. Debe indicarse que las cotizaciones se consideran como definitivas, y las responsabilidades en que se involucrará el vendedor en caso de modificarlas. Es necesario indicar los suministros que el comprador puede utilizar para pagar los materiales, es decir, si será en efectivo, al recibo de la mercancía, por medio de una carta de crédito, por medio de un depósito, etc. Así como el tipo de divisa para el pago. Es necesario también indicar si los suministros deberán despacharse por vías terrestres, marítimas o aéreas, o por cualquier combinación de las anteriores, lo cual viene determinado por la urgencia del pedido y el precio que se pactó pagar. En algunos casos también se establece que toda compra de cierta magnitud debe estar asegurada contra determinados riesgos. En algunos casos es necesario considerar ciertas fianzas que se establecen por medio de un pequeño depósito a favor de la institución, cuyo comprobante se anexaría a la cotización y cuyo propósito es el de asegurar la seriedad de la oferta. Este se conoce con el nombre de fianza de concurso y su monto debe reembolsarse al proveedor si es que no fue favorecido por la adjudicación del contrato. Otro tipo de afianzamiento es aquél que se realiza para el cumplimiento del contrato, que se exige a los ganadores de la licitación como requisito previo a la contratación; esta fianza es generalmente de mayor valor que la anterior. Respecto a las sanciones, se establece la posibilidad de que se pague una multa que consiste en una cantidad por un cierto período ya sea

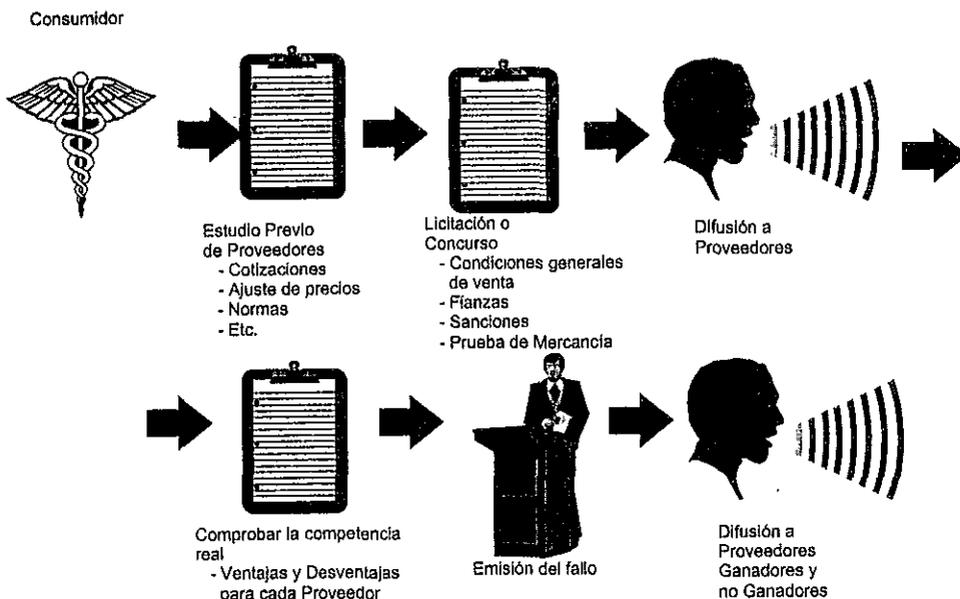
diario, semanal u otro que retrase la entrega de la mercancía, y también deben establecerse las condiciones en que se cancelará el contrato por demoras que impliquen en perjuicio para la parte solicitante o consumidora. También se incluye en la invitación las pruebas o análisis a los que podrían someterse los suministros de forma eventual, por lo que se puede tocar el caso de las importaciones y la posibilidad de que este análisis se lleve a cabo en el lugar de la fabricación o en el punto de embarque por organismos especializados, que pueden ser nacionales o internacionales según sea el caso, por asociaciones profesionales o inclusive por laboratorios privados. Con esto se evitarían las demoras y problemas que surgen al encontrarse que al llegar al punto de recepción resulta evidente la insatisfacción de las especificaciones del contrato. También es necesario establecer, en lo que se refiere a empaque y marca, la forma, el tamaño, el material, la resistencia y características de los envases que deban utilizarse para la protección de los diferentes artículos que en algunos casos tendrán una característica especial como puede ser impermeables, anticorrosivos, etc. Es frecuente que exista en algún lugar de la licitación el compromiso de que los proveedores aceptan los términos y las condiciones y que además están enterados de los mismos. La fecha de terminación de las ofertas por parte de los participantes es otro punto importante que se incluye en el documento; este requisito está ligado a la fecha de la apertura de las ofertas y no debe excluirse de ninguna manera, además de advertir que la apertura de las ofertas deberá ser presenciada por concursantes o representantes autorizados de éstos. En ocasiones es necesario solicitar algunos datos adicionales de los concursantes para validar si los materiales y/o equipo ofrecidos satisfacen las requisiciones y especificaciones, estos datos pueden incluir diagramas, planos, etc.

La distribución de la licitación se hace a través de una gran variedad de medios de publicidad disponibles o del dominio público. Los concursantes tienen derecho a retirar, substituir o modificar sus ofertas antes de la apertura. En caso de existir un cuestionamiento sobre las especificaciones, las condiciones y los términos que las mismas contienen, estos les deberán de ser contestados completamente no sólo al concursante que lo planteó sino a todos los demás sin distinción alguna. Antes de emitir lo que se conoce como fallo, es necesario levantar una acta de la apertura en la cual se

indicarán las ofertas recibidas, para pasar en seguida al proceso de evaluación de las cotizaciones, que consiste en la selección de la(s) oferta(s) presentada(s) por uno o varios concursantes y que resultan aceptables ya que se ajustan lo más posible a las especificaciones y condiciones de la licitación y que ofrezca al mismo tiempo el precio más favorable. Este proceso puede resumirse en una serie de pasos que básicamente son: verificar que existió una verdadera competencia, realizar un resumen de las cotizaciones en el que resalten sus ventajas y desventajas, una vez ordenadas de acuerdo al grado de calificación obtenido con base al precio y la calidad y que además se ajusten a las condiciones impuestas en ese concurso. Es común que una vez finalizado el proceso exista más de un finalista, y tal vez porque la licitación misma así lo estipula, por lo que la adjudicación del contrato podría resolverse con cualquiera de las tres siguientes formas: por la pugna entre diversos proveedores, por el reparto proporcional o por sorteo. Cualquiera fuera la resolución, esto debe de ser acentado en el acta antes mencionada. Todos los concursantes deben tener conocimiento de la adjudicación del contrato de suministros, para que en todo caso puedan recurrir a la apelación de acuerdo a la reglamentación bajo la que se encuentran las instituciones de atención médica. Si el proveedor que hace la oferta acepta dicha adjudicación se deberá de iniciar de inmediato a la preparación del contrato, del pedido y orden de compra, cumpliendo requisitos relativos a la fianza. En la figura 1.4.6 se muestran los principales pasos que sigue una Licitación.

Es conveniente que ambas partes mantengan un registro de las compras para la parte consumidora como de las ventas para la parte proveedora ya que éstos resultan de gran importancia para la preparación de informes que reflejen la situación en el manejo de suministros. Así es como podemos tener que por parte del proveedor es conveniente mantener un registro de sus consumidores que nos indiquen el nombre, la dirección, el teléfono, la experiencia que se ha tenido con él y un listado con productos que a cada uno se les ofrece, además de otras características que pueden variar de uno a otro. También es conveniente mantener un registro de cotizaciones otorgadas por cada uno de los clientes. Un registro de especificaciones para cada producto es de suma importancia. El registro de pedidos que puedan tener asociada una remisión o contrato

que nos muestre la forma en que se originó, cómo es que se desarrolló y en todo caso el estado de culminación. Un registro de licitaciones que le permita tener cada una por separado y por cliente para poder obtener un perfil de ellos en lo que respecta a esto. Pueden llevarse por separado otro tipo de registros, sin embargo eso depende en gran medida del nivel de control interno que se pretenda alcanzar.



**Figura 1.4.6 Principales pasos en el desarrollo de una Licitación.**

Por parte del consumidor se pueden mantener registros que tienen una estrecha relación con la contraparte que registra el proveedor. Así bien, puede llevar un registro de proveedores, uno por cada proveedor con su nombre, su dirección, su teléfono, su historial y un listado con las clases de productos que vende. Un registro de cotizaciones que se actualizan cada vez que un proveedor realiza una modificación, sobre todo en sus precios. Un registro de especificaciones por producto y con referencia a un proveedor para atender casos de reclamos o compras a futuro. Un registro de órdenes de compra que muestre cada orden de compra, la forma en que se originó, el número,

una breve descripción de los suministros que la misma contiene, la cuenta a la que se cargó, el vendedor y el valor de la orden. El registro de contratos es necesario para el caso de alguna reclamación o como referencia para contratos similares con determinado proveedor. El registro de procesos que muestre para cada orden de compra el estado en que se encuentre dicha operación, es decir, si se solicitaron ofertas, si se recibieron, si se adjudicaron, si se preparó el contrato, etc. Es decir, las operaciones más importantes. De manera similar que la parte proveedora el mantener un registro de licitaciones puede considerarse importante, guardándose en él toda la documentación relativa a los suministros. La determinación de otro tipo de registros depende como se mencionó con anterioridad del nivel de control que se busque alcanzar dentro de la institución.

#### **1.4.2. El Efecto de la Normalización y la Catalogación para la Elaboración de un Pedido.**

El proceso de normalización de artículos compete en mayor medida a la parte consumidora sin dejar de afectar a la proveedora. Para esta primera resulta mas evidente ya que se puede encontrar en una inspección a sus almacenes con que cuenta con una amplia variedad de productos que sirven para el mismo propósito. Por lo tanto, debe procurarse la selección conveniente de los artículos y materiales, que en forma obligatoria pueden ser usados por todas o algunas dependencias de atención médica. El proceso de normalización debe de ser una labor conjunta de varios departamentos, por lo que es necesario preparar una lista de los productos que se distribuyen de los diversos proveedores y que utiliza cada unidad de servicio médico, indicando su calidad, requisitos y costo, entre otras cosas. Esta lista puede ser revisada con el objeto de evitar duplicidad innecesaria, recomendar sustituciones en líneas de artículos similares e identificar los suministros que puedan reemplazarse entre sí, establecer una nomenclatura precisa para cada uno de los artículos que son de uso corriente o bien escoger el cuadro básico en caso de tratarse de medicamentos y revisar las especificaciones de cada artículo o material o hacerlas si es que no se tienen. Dentro de los requisitos que contiene un pedido de medicamentos, el cuadro

básico de medicamentos juega un papel muy importante ya que para instituciones de servicio médico que dependen directamente del gobierno resulta más efectivo aplicar este tipo de procedimientos de normalización sustentadas con dicho cuadro. Estos cuadros básicos consisten en seleccionar los productos que inspiran más confianza al personal médico de la institución, y generar una lista alfabética que se puede complementar por una clasificación terapéutica y la relación correspondiente para realizar las posibles sustituciones en caso de la existencia de productos similares, con objeto de que los departamentos de proveeduría y compras puedan llegar a sustituir a un proveedor por otro, aprovechando los beneficios que la competencia implica. A pesar de las críticas que se generan por el uso de este esquema, la tendencia de las instituciones médicas de gobierno es hacer un uso más extenso del mismo ya que es una forma de evitar la intermediación que tanto encarece a los productos además de tener bajo control el suministro de medicamentos de uso más frecuente. Sin embargo, es necesario tener en consideración que el establecimiento de cuadros básicos de medicamentos o formularios médicos influye de manera importante en la confianza que se le inspira al paciente al tratar de normalizar a tal grado, que las presentaciones a la vista sean muy similares, es por esto, que en muchas instituciones médicas en las que no se tiene un laboratorio propio, se tengan que seguir recibiendo pedidos con los empaques o envases comerciales, y únicamente se agrega en una de las caras de la etiqueta el número de la catalogación y de la codificación, haciéndose evidente que ese medicamento es proporcionado solamente por una institución médica.

Una vez que la normalización se lleva a efecto, es necesario informar a todas las partes correspondientes, pues de acuerdo a esto, se deberán de hacer los pedidos subsecuentes para derivar en un proceso de catalogación que consiste en un ordenamiento, de acuerdo a un criterio establecido para los productos que se destinan a un fin específico, es decir, consiste en una clasificación adecuada de los productos y en el diseño de un número de identificación para cada uno de ellos con objeto de simplificar el control, la distribución y sobre todo la adquisición. Tenemos un ejemplo de una lista de productos catalogados, como la siguiente:

1. Medicamentos.
2. Material de curación.
3. Lácteos.
4. Medicamentos libres.
5. Comestibles.

Primeramente se deben de agrupar todos los productos normalizados dentro de partidas y clases, donde cada clase debe de incluir productos homogéneos, tomando en cuenta su empleo, características químicas, su origen o su destino. Cada partida puede dividirse en subpartidas, que a su vez pueden tener un considerable número de artículos. Como por ejemplo, tenemos la siguiente división:

Partida 0, material quirúrgico.

Subpartida 0-01, suturas.

Artículos 0-0-01, catgut crómico del número uno.

Se puede fijar como regla general que debe decidirse que la clasificación a utilizar se base principalmente en el uso que le dará al catálogo, procurando que sea lo más sencilla posible y adecuada sobre todo para los fines de compra y distribución. La codificación puede ser numérica o alfanumérica y está limitada por la cantidad de artículos que se pretenda controlar, y tiene como punto de inicio la partida o división mayor, cada partida debe identificarse con el mismo número de la partida superior más el correspondiente de la partida en curso. Resulta evidente que la codificación genera ocurrencias únicas, no pueden existir dos artículos con la misma codificación, la codificación incluye las unidades de medida específicas para cada artículo y que se utilizan tanto para su reparto como para identificar como es que el proveedor los proporcionó. A conveniencia de los sistemas de control interno se pueden manejar otras características dentro de los catálogos codificados, como puede ser el manejo de los precios, ya que como éstos son variables tendrían que permanecer continuamente bajo inspección para su oportuna actualización. En muchas instituciones, sobre todo en lo que se refiere a catálogos de medicamentos y material de curación, existe por separado

el catálogo que muestra exclusivamente el nombre del principio activo del medicamento, las indicaciones, la dosificación, la presentación y la forma de empaque o envase, y otro catálogo en que se muestra el medicamento y la forma de empaque o envase, la dosificación, la presentación y el o los laboratorios que lo producen. Esto significa una ventaja para los proveedores ya que en caso de intervenir en el proceso de suministro algún intermediario no tendrá que preocuparse por redefinir sus pedidos. Cabe hacer mención que es necesario establecer un procedimiento administrativo para que el catálogo esté siempre vigente y sea lo más confiable posible.

El ciclo de generación y cumplimiento de un pedido lleva implícitos, por lo que se puede apreciar, una gran número de requisitos, esto influenciado en gran parte por la interacción de dos partes que tienen una estructura organizacional propia y una metodología de control propia. Por lo que la parte proveedora, se tiene que "acomodar" a dichos requisitos para tener presencia dentro de un mercado tan competido como el de los insumos farmacéuticos.

## 1.5. CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE VISUAL BASIC 4.0

### 1.5.1. Introducción a Visual Basic

Visual Basic es un poderoso sistema de programación gráfica que le permite crear aplicaciones Windows reales con código BASIC. Este sistema es un extraordinario avance para desarrollo de programas, porque combina la sintaxis simplificada de BASIC y GWBASIC con la estructura de programación QBASIC y QuickBasic.

El sistema de programación Visual Basic le permite crear *objetos*, establecer y cambiar sus propiedades para posteriormente, asignarles un código BASIC funcional.

La filosofía de la programación Visual Basic consiste, primero, en crear objetos como ventanas, iconos y menús y después elaborar procedimientos que sean llamados por cada uno de estos objetos. Esto es diferente del método tradicional de elaboración de un programa el cual exige estructuras para controlar el flujo del programa de un procedimiento a otro de manera lógica hasta que el programa termina. La programación por eventos es un método flexible y conveniente para escribir programas en Windows.

Por ejemplo, usted puede elaborar el código para un objeto que ha creado y después hacer copias del mismo con el código asignado; no tiene que escribir el código otra vez.

Ahora bien, definir Visual Basic como un *sistema de programación orientada a objetos* sería erróneo. Porque usted crea objetos, llamados *formas y controles*, que hacen que su aplicación funcione. Sin profundizar en el tema, estos objetos carecen de las propiedades de herencia y polimorfismo, las cuales tienen que estar presentes en un verdadero ambiente orientado a objetos. Además, usted está creando aplicaciones que se correrán en Windows, el cual es un ambiente *orientado a lo visual* que proporciona librerías de herramientas y objetos para aplicaciones de programación. La mayor ventaja de utilizar Windows es que proporciona una interfaz consistente y manejable

para desplegar información en muchas aplicaciones diferentes que comparten controles similares.

La filosofía de la interfaz Windows se basa en la utilización de metáforas visuales para realizar acciones y tareas.

El paradigma es simple: después de que usted ha utilizado una aplicación Windows, aprende fácilmente a utilizar otra ya que está familiarizado con la interfaz.

Al utilizar controles Windows como menús, iconos, barras de desplazamiento y cajas de diálogo, la información es presentada a los usuarios en un formato dinámico y visualmente interesante.

Visual Basic reduce drásticamente las dificultades y la cantidad de tiempo de la programación en Windows, usted puede construir una aplicación Windows dinámica en pocos días. Ya que este lenguaje es *interpretado* más que compilado, las aplicaciones están listas para correr tan pronto como se termine de escribir el código. En contraste, con un lenguaje compilado, un programa debe compilarse antes de que pueda ser ejecutado. Esto significa que un compilador convierte los programas al lenguaje de máquina. Esta conversión se realiza solamente una vez, cada vez que se corre el programa, el lenguaje de máquina ya está disponible y puede ser ejecutado de manera directa por la computadora.

### **1.5.2. Requerimientos de Hardware**

Antes de instalar el programa Visual Basic, hay que asegurarse de que la computadora cumpla con los siguientes requerimientos mínimos:

- Computadora IBM o PC compatible con un procesador 80386 ó más poderoso.
- Al menos 50 Mb de espacio disponible en disco duro para una instalación completa.
- Unidad de disco de 5.25" o 3.5", o una unidad de CD-ROM.

- Cualquier monitor VGA ó de alta resolución soportado por Microsoft Windows 3.1 o posterior.
- Al menos 8 Mb de memoria RAM, requeridos para Windows 95. (Esto variará dependiendo del tipo de librerías específicas o DLLs que incluya en sus aplicaciones).
- Ratón.
- Para versiones de Visual Basic de 32 bits, se deberá contar con Windows 95 o más reciente, o Windows NT 3.51 o más reciente.
- Para versiones de Visual Basic de 16 bits, se deberá contar con Windows versión 3.1 o más reciente en modo mejorado.

Para cualquier duda, consultar el archivo readme.doc que viene en el directorio raíz del disco de instalación.

### 1.5.3. Elaboración de Programas Orientados a Eventos

Cuando se inicia una aplicación, durante la corrida sólo las formas que usted creó y programó serán visibles. Una *forma* es una ventana que usted crea y adapta a sus necesidades para cada aplicación. Esta es la base de cualquier aplicación Visual Basic que eventualmente correrá como un programa independiente de Windows. Un *control*, es el nombre preciso para cualquier objeto que usted dibuje en una forma, pero también se refiere a la forma misma. Los botones de comando, menús, cajas de diálogo y barras de desplazamiento, son ejemplos de controles. Cuando un usuario activa un control, el código BASIC que tiene asignado el control, es invocado y realiza una tarea específica. Un *módulo* es una estructura para escribir el código que usted le asigna a una forma y sus controles. Después de dibujar los controles que usted desea que aparezcan en una forma, escribe el código funcional para cada control que los usuarios pueden invocar. Visual Basic utiliza la metáfora del "evento" para describir su paradigma de programación, siempre utilizará este enfoque orientado a los eventos cuando cree una aplicación. *Orientado a eventos* significa que todos los controles que usted dibuje en una forma especifican cómo se comportará la interfaz. Esto es, los

controles de Visual Basic esperan que sucedan eventos particulares antes de que respondan. Un *evento* es una acción que es reconocida por un control, es código escrito para cada control a fin de que algo suceda cuando los usuarios interactúan con la forma.

Mientras se está en Visual Basic y corre una aplicación, cada línea de código es interpretada y ejecutada por separado cada vez que aparece. Es esta ejecución individual de cada línea la que puede hacer que un programa interpretado corra con mayor lentitud que un programa compilado. Por consiguiente, el archivo terminado también es más grande. Por supuesto, un archivo compilado es un archivo ejecutable, que ya no es interpretado.

#### **1.5.4. Terminología de Visual Basic**

La necesidad de familiarizarse con la terminología de Visual Basic tiene su singular enfoque orientado a eventos, para programar aplicaciones. Por consiguiente, es útil para describir los elementos de una aplicación Visual Basic:

- *Control*: Un término general utilizado para describir cualquier forma o elemento gráfico que usted dibuje sobre una forma, incluyendo cajas de texto, cajas de listado, botones de comandos, cajas de imágenes, barras de desplazamiento e iconos. Una definición más precisa de un control, es la de datos acoplados con una serie de rutinas, conocida como *métodos*. Las únicas operaciones que pueden desarrollarse sobre un control son aquellas definidas como *métodos para él*.
- *Evento*: Acción reconocida por un control Visual Basic.
- *Forma*: Una ventana que usted crea y adapta a sus necesidades para su aplicación.

- *Método*: Una palabra de código Visual Basic que es similar a una función o instrucción, pero que siempre actúa sobre un control particular. Para cada control, Visual Basic predefine una serie de métodos que usted puede utilizar.
- *Procedimiento*: Un término que se refiere tanto a procedimientos **Sub** (Sub procedimientos) y **Function** (Función). Un procedimiento es sencillamente una secuencia de instrucciones Visual Basic que son ejecutadas en grupo durante la corrida. Existen dos tipos de procedimientos:

Procedimientos de eventos y procedimientos generales. *Los procedimientos de eventos* se limitan a las formas y los controles, mientras que *los procedimientos generales* se utilizan durante toda la aplicación y pueden ser requeridos por procedimientos de eventos.

- *Proyecto*: Conjunto de todos los archivos que forman la aplicación.
- *Propiedad*: Una característica o atributo de un control para cada tipo de control, Visual Basic define una serie de propiedades que se aplican solamente a ese control.
- *Definición*: El valor de una propiedad, se puede cambiar la definición de la mayoría de las propiedades mientras está construyendo una aplicación. El código de una aplicación que se corre también puede cambiar las definiciones.

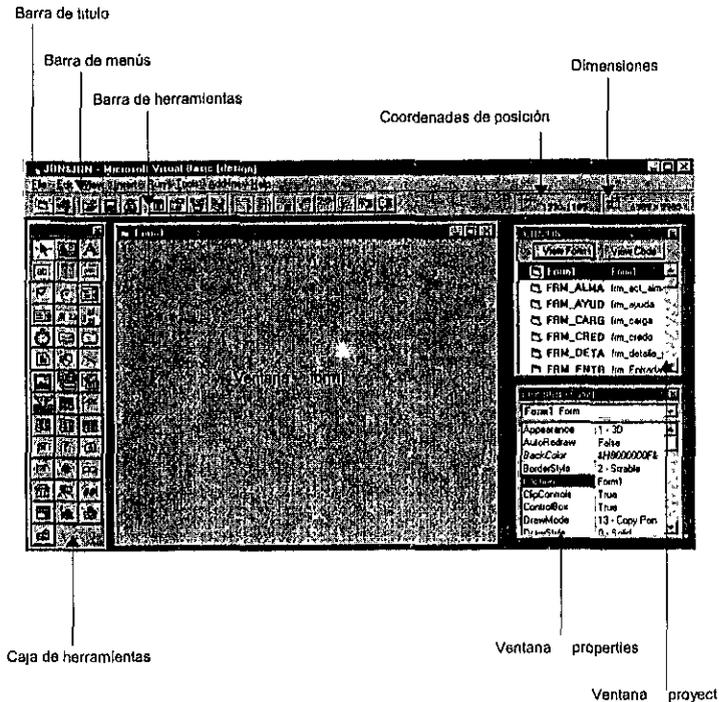
### 1.5.5. Acceso a la Barra de Menús

El ambiente de programación Visual Basic está formado por cinco elementos a los que usted accesa a fin de crear aplicaciones propias:

- Barra de menús.
- Barra de propiedades.
- Ventana Form.

- Ventana Project.
- Caja de herramientas.

Estas ventanas son lo primero que se observa cada vez que inicia Visual Basic como se muestra en la figura 1.5.5.1.



**Figura 1.5.5.1 El ambiente de programación Visual Basic**

### Barra de Menús

La barra de menús hace aparecer los nombres de los menús que están disponibles para utilizar en la ventana activa. Encima del menú de barra está la barra de título, los botones de maximización y minimización y la caja de control los cuales se muestran en la figura 1.5.5.2. Para entrar al menú hay varias maneras: puede hacer clic sobre un

nombre del menú para abrirlo y hacer que aparezcan sus comandos o también puede oprimir una tecla aceleradora de nombre de menú, reconocidas porque la letra esta subrayada. Por ejemplo, la tecla aceleradora para el menú **File** (Archivo) es **F**.



**Figura 1.5.5.2. Barra de menús y títulos de la barra.**

### **Características**

El sistema dinámico de ventanas, menús y herramientas le permite crear formas y controles, manipular archivos de proyectos, manipular texto, correr programas y depurar el código. Entre las características más interesantes de Visual Basic se incluyen las siguientes:

Cuando se comienza un programa durante la corrida, puede seleccionar comandos del menú **File** de la figura 1.5.5.2, para poner o quitar puntos de interrupción en el código y para ejecutar un programa a fin de evaluar cómo se ejecutan las instrucciones

- La ventana **Immediate** (Inmediato) le permite observar como se ejecuta una instrucción en particular mientras usted corre un programa.
- La ventana **Menu Design** (Diseño de menú) está disponible para crear menús con teclas aceleradoras, la figura 1.5.5.3 muestra las características de esta ventana.
- Se puede crear un sistema de ayuda completo, un tutorial y un índice por medio del menú **Help** (Ayuda).
- La ventana de propiedades le ofrece una manera conveniente de escoger propiedades de control y cambiar definiciones durante el diseño, la figura 1.5.5.4 muestra los elementos de esta barra.

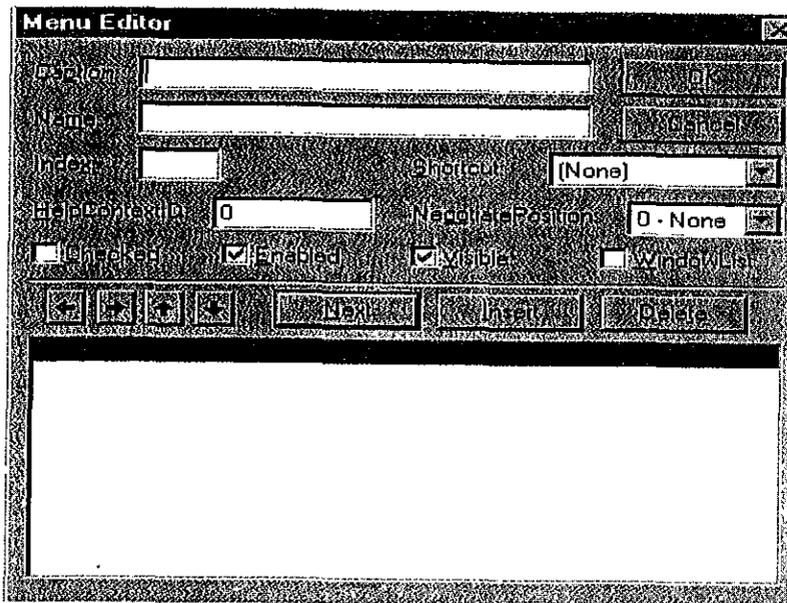


Figura 1.5.5.3. Ventana de diseño de menú.

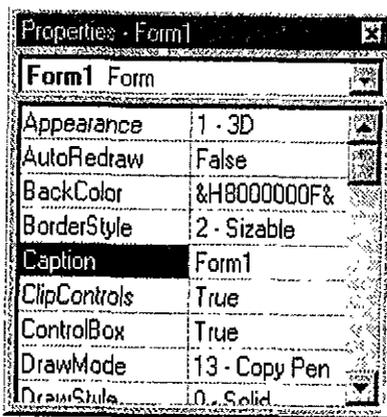
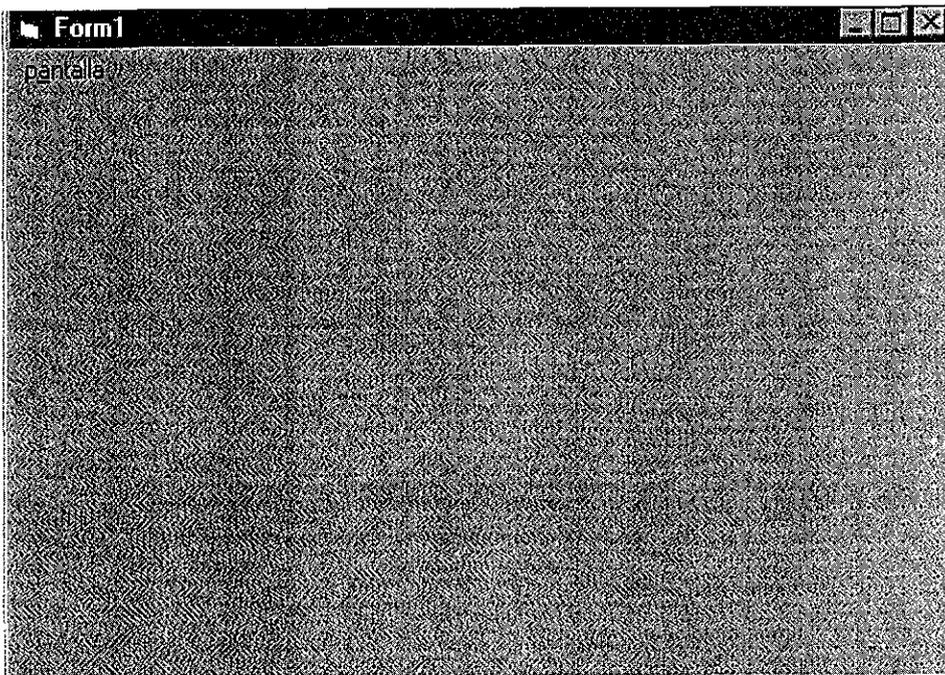


Figura 1.5.5.4. Ventana de propiedades.

- La caja de herramientas le permite crear 15 tipos diferentes de controles sobre una forma.
- Una ventana de forma, como muestra la figura 1.5.5.5, por omisión, aparecerá automáticamente cuando usted inicie Visual Basic. Cuando usted hace clic con el botón izquierdo del **mouse** (ratón, dispositivo de señalización o apuntador óptico) sobre una forma o control, aparece la ventana **Code** (Código) en la cual usted escribe el código funcional para una aplicación.
- La ventana Code, figura 1.5.5.6, muestra las cajas del listado de las cuales usted selecciona los nombres de controles y eventos, y los nombres de procedimientos generales.



**Figura 1.5.5.5. Ventana Forma de muestra.**

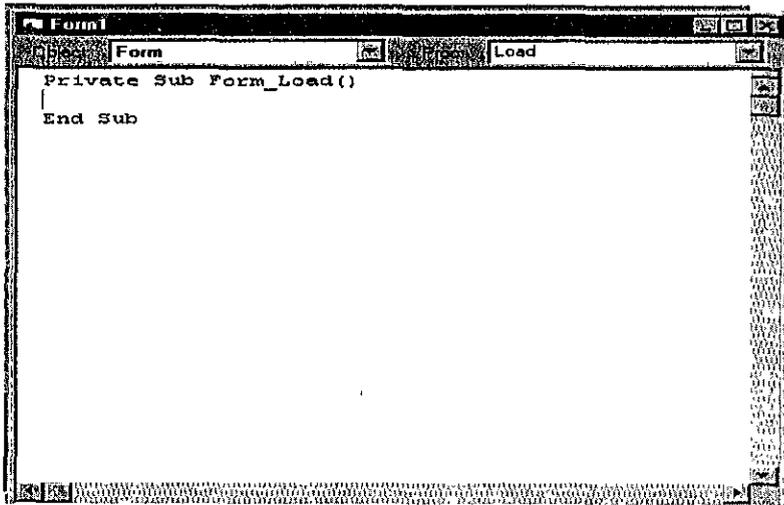


Figura 1.5.5.6. Ventana de código

- La ventana **Project** (Proyecto) organiza y muestra los nombres de todas las formas y módulos que son parte de una aplicación particular. Usted puede mostrar el código para una forma destacando el nombre del archivo en esta ventana y haciendo clic en el botón **View Code** (Vista del código); no tiene que mostrar primero la forma (o control) real para observar el código. La figura 1.5.5.7 muestra la ventana Project.

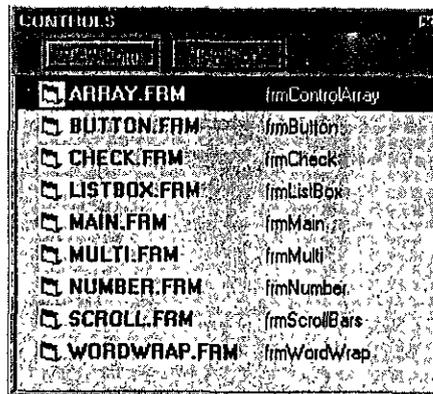


Figura 1.5.5.7. Ventana Project

## 1.5.6. Creación de Controles con la Caja de Herramientas

La caja de herramientas Visual Basic contiene un conjunto de 16 herramientas (las cuales muestra la figura 1.5.6.1) que puede utilizar para dibujar, mover o redimensionar los controles de su forma y a los cuales posteriormente les asignará el código BASIC funcional.

### Creación de un Control

Para crear cualquier control, siga estos pasos:

1. Haga clic sobre la herramienta **Text Box** (Caja de texto) para seleccionarla. El apuntador del ratón se convertirá en una cruz.
2. Coloque la cruz sobre la forma en la posición deseada.
3. Mantenga apretado el botón izquierdo del ratón y arrastre la cruz hacia abajo diagonalmente hacia la derecha hasta que el control tenga el tamaño deseado.
4. Suelte el botón del ratón. El encabezado `Text1` aparecerá sobre la caja de texto y *Selection Handles* aparece alrededor de la caja para indicar que está seleccionada.
5. Centre la caja de texto sobre la forma, como se muestra en la figura 1.5.6.2, arrastrándola.
6. Usted también puede arrastrar las manijas de selección para aumentar o disminuir el tamaño de la caja de texto.

### Modificación de las Propiedades de un Control

Ahora que ha creado un control de caja de texto, puede establecer las propiedades tanto para la forma como para la caja de texto para que aparezcan como usted desea.

Cambie **Caption** (Título) que aparece en la barra de título de la forma:

1. Seleccione la forma haciendo clic sobre ella (a menos que este ya seleccionada). El nombre `Form1` aparece en la caja de listado **Settings** (Valores) para la propiedad

Caption. (La caja de listados Settings esta a la mitad de la barra de propiedades, debajo de la barra de menú de Visual Basic).

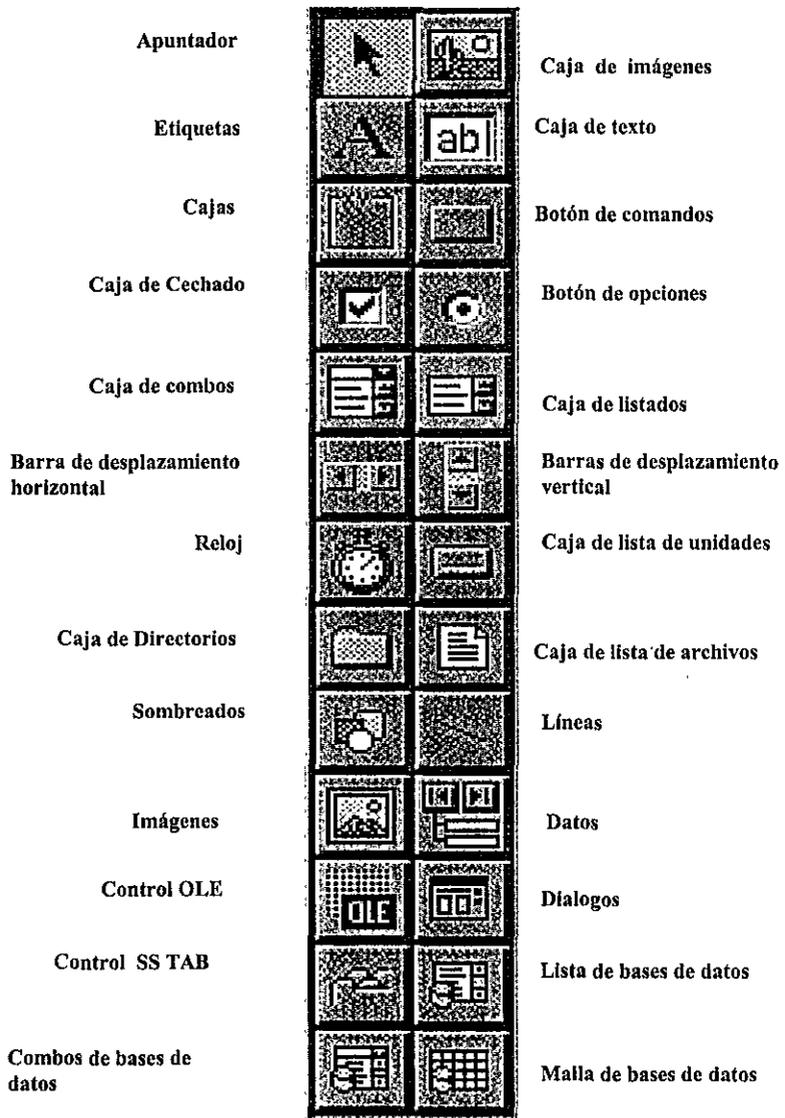
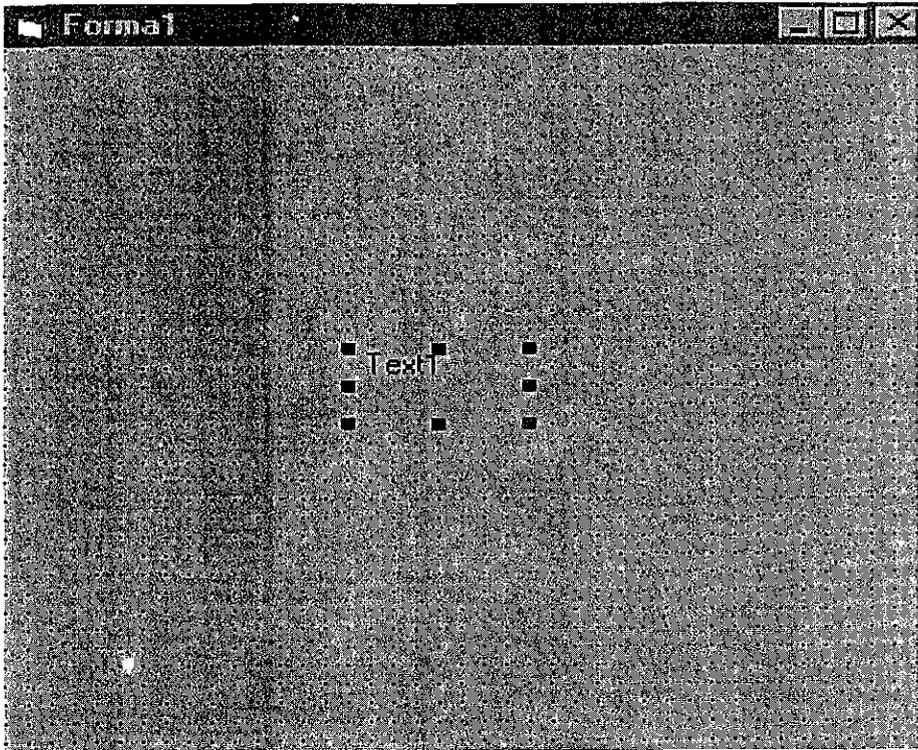


Figura 1.5.6.1. Caja de herramientas de Visual Basic



**Figura 1.5.6.2. Primer control que usted realiza, una caja de texto centrada sobre la forma**

2. Mueva el apuntador del ratón a la caja de listados Settings. Se convertirá en un cursor en forma de "I" al que se le llama rayo "I".
3. Haga resaltar el título haciendo doble clic sobre él colocando el rayo "I" al principio de la palabra y arrastrándolo a la derecha.
4. Oprima la tecla **Delete** (Borrar) para quitar Caption.
5. Teclee **Welcome** (Bienvenido) y presione **Enter** para confirmar la nueva definición, o haga clic sobre el botón Enter (botón izquierdo del ratón para configuración de mano derecha).

Ahora cambiemos la propiedad **Name** (Nombre) de la caja de texto para que usted pueda referirse a ella en un procedimiento:

1. Con la caja de texto sobre la forma seleccionada, haga clic sobre la caja de listado **Properties** (Propiedades), a la izquierda de la caja de listado **Settings**.
2. Seleccione la propiedad **Name**, si no ha sido seleccionada. El nombre **Text1** aparece en la caja de listado **Settings**.
3. Teclee **Display** en la caja de listado y oprima **Enter** para confirmar el cambio.

Ahora cambie la propiedad **Text** (Texto) del control:

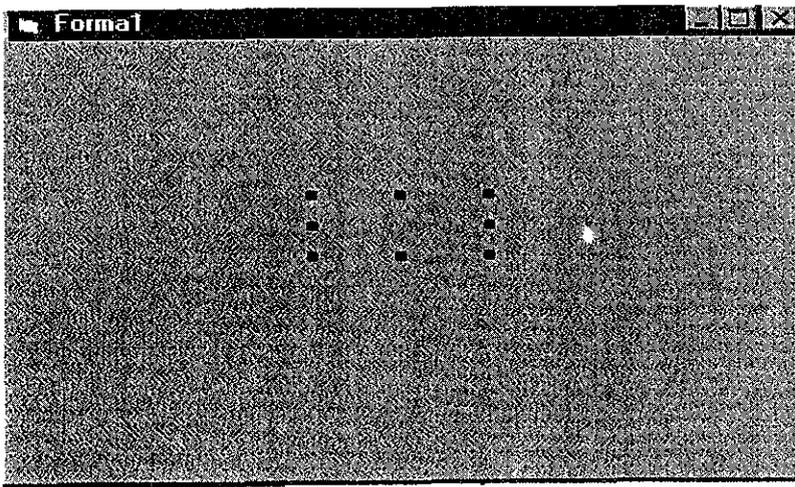
1. Seleccione la propiedad **Text** en la caja de listado **Properties**. El nombre de **Text1** aparece en la caja de listado **Settings**.
2. Haga un doble clic dentro de la caja de listado **Settings** para resaltar el texto.
3. Oprima la tecla **Delete** para quitar el texto.
4. Oprima **Enter** para confirmar el cambio.

Finalmente, cambie la propiedad **BorderStyle** (Estilo de Borde) del control:

1. Haga clic sobre la caja del listado de **Properties** que está abierta, mueva consecutivamente la lista, y seleccione la propiedad **BorderStyle**. La definición 1 aparece por omisión en la caja **Settings**, el cual es el valor por omisión de una sola línea compuesta.

2. Haga clic sobre la flecha, del lado derecho de la caja de listado Settings (la flecha indica que existen más definiciones para la propiedad). Aparece una lista.
3. Seleccione la definición Settings, la cual es el valor para no tener límites.
4. Haga clic sobre el botón Enter para confirmar el cambio.

La caja de texto deberá aparecer como la que se muestra en la figura 1.5.6.3.



**Figura 1.5.6.3. La apariencia del título ha sido cambiada modificando sus propiedades**

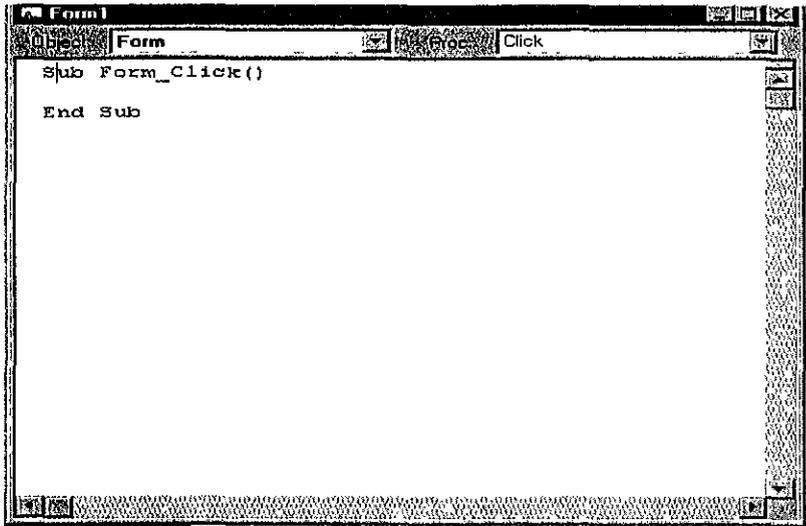
### **1.5.7. Escritura del Código Funcional**

Para escribir el código para un evento **Load** (Carga), siga estos pasos:

1. Haga doble clic en cualquier parte de la forma. La ventana de código aparecerá como se muestra en la figura 1.5.7.1. Una plantilla de código

aparece automáticamente mostrando la primera y la última línea del procedimiento de evento. En este ejemplo, el procedimiento de evento es identificado con el nombre `Form_click()`.

2. Haga clic sobre la flecha hacia abajo del lado derecho de la caja de listado Procedures.



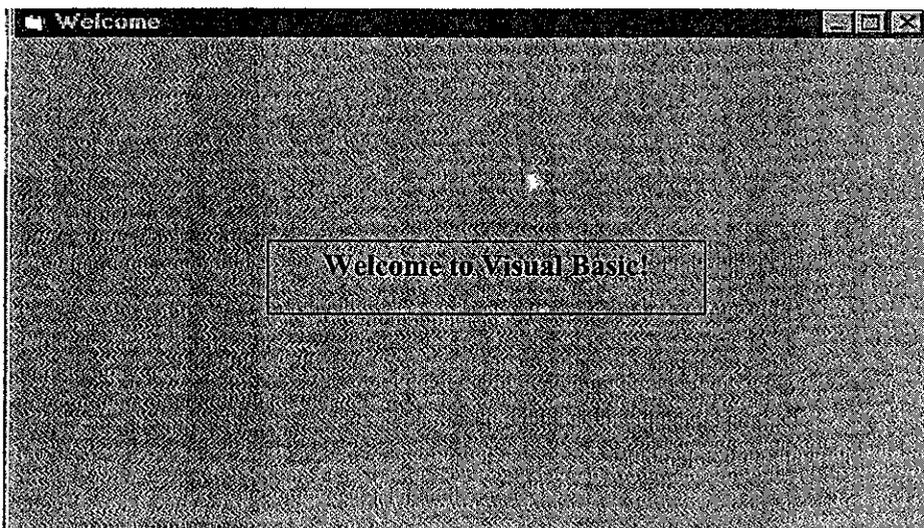
```
Form
Form
Proc Click
Sub Form_Click()
    End Sub
```

**Figura 1.5.7.1. El código de la ventana con el procedimiento de evento clic mostrado en la forma**

3. Seleccione **Load**. El nuevo evento aparecerá en la plantilla de código automáticamente.
4. Coloque la barra destellante de inserción entre las instrucciones `Sub` y `End Sub`.
5. Presione la tecla `Tab` (Tabulador) para darle sangría a esta línea (es un buen estilo de programación escribir las instrucciones con sangría).
6. Teclee

Display.Text="Welcome to Visual Basic!"

7. Haga doble clic sobre la caja de control para cerrar la ventana de código.
8. Abra el menú **Run** (Correr) y seleccione el comando **Start** (Ejecutar) para correr el programa. La aplicación hará aparecer el mensaje Welcome to Visual Basic!, como se muestra en la figura 1.5.7.2.
9. Abra el menú Run de nuevo y seleccione el comando **End** (Terminar) para detener la ejecución del programa.



**Figura 1.5.7.2. La forma muestra el mensaje "Welcome to Visual Basic!" cuando se corre la aplicación en la ventana Project.**

Para crear cualquier control automáticamente, sólo haga, doble clic sobre la herramienta en particular, aparecerá un control de tamaño por omisión en el centro de la forma. Para redimensionar un control, arrastre sus manijas de selección. Para reposicionar un control, sólo arrástrelo a otra posición dentro de la forma.

## 1.5.8. Utilización del Ambiente de Trabajo Visual Basic

¿Qué es una interfaz?

El término interfaz para el usuario es difícil de definir. Si usted pregunta a varios programadores ¿cómo lo definirían?. Es probable que obtenga respuestas muy diferentes porque no hay una interfaz para el usuario estándar para aplicaciones basadas en **DOS** (Disk Operating System). Aunque las aplicaciones Windows suelen reflejar especificaciones que los programadores pueden seguir para mantener una consistencia funcional entre las aplicaciones, la consistencia no es forzosa. Esto no quiere decir que las aplicaciones Windows no se apeguen a las convenciones generales de diseño. Pero se puede aplicar una cierta libertad, que con frecuencia resulta en aplicaciones Windows tan diferentes como los programadores que las crearon.

El área de las interfaces gráficas para el usuario abarca todo desde los aspectos filosóficos hasta psicológicos, y se ha convertido en una importante disciplina, captando la atención de programadores e investigadores en las áreas de la computación, ingeniería y psicología cognoscitiva. Lo que ha surgido es una visión del usuario que utiliza una computadora para realizar muchas tareas y cumplir con su trabajo.

Generalmente, la interfaz entre usted y la computadora consiste en una pantalla, un teclado y un ratón. Eso es lo que presenta información a usted y acepta información suya.

Usted interactúa con su computadora respondiendo con lo que ve en la pantalla, escribiendo comandos en el teclado o apuntando y haciendo clic con el ratón.

Mediante una aplicación, usted se comunica o más precisamente, tiene una *interfaz* con su computadora. De hecho, una aplicación Visual Basic como todas las aplicaciones Windows, se accesa mediante una interfaz gráfica con el usuario. Una interfaz es más

que una visión de objetos, textos, gráficas y patrones. Una interfaz gráfica para el usuario (o interfaz computadora-humano), tal como una aplicación, es el “diálogo”, que ocurre entre una computadora y sus usuarios. En este caso, el usuario interactúa con la computadora dándole comandos; la computadora, a su vez, le responde al usuario realizando una tarea. En otras palabras, la interfaz es aquel componente de una aplicación que traduce una acción del usuario con las consecuencias de sus acciones.

Visual Basic permite diseñar la interfaz de una aplicación utilizando la interfaz Windows para dar a los usuarios un método consistente de interacción con la computadora. Probablemente la consistencia sea la mayor ventaja de una interfaz. Todas las aplicaciones utilizan comandos y controles cuando usted aprende una aplicación Visual Basic. Una manera como Visual Basic mantiene la consistencia de una aplicación con otra, es mediante el uso de objetos y metáforas de la vida real. La metáfora de las ventanas es utilizada porque representa una experiencia de la vida real que la mayoría de nosotros comprendemos.

La creación de una aplicación efectiva comienza con un buen diseño y una visión amplia del contenido y propósito de la aplicación. La manera en que usted organiza una aplicación depende de la información, que ésta contiene y de cómo espera usted que los usuarios la utilicen y no sencillamente para que la observen.

Finalmente, una persona correrá la aplicación que usted ha creado *fuera de Visual Basic*, simplemente como cualquier otra aplicación Windows. Por tanto, es importante que se detenga y haga varias preguntas fundamentales que afectarán el diseño:

- ¿Cuál es el propósito de la aplicación?
- ¿Qué razón (o razones) tiene el usuario para utilizarla?
- ¿Qué información se presenta en la aplicación?
- ¿Cómo se da a entender esta información en la aplicación?
- ¿Cómo encuentra el usuario la información en la aplicación?
- ¿Qué indicaciones le da la aplicación al usuario?

Si toma en consideración el diseño de una aplicación antes de comenzar, verá mejores resultados y se ahorrará mucho tiempo (y dolores de cabeza).

Mediante el proceso de hacer claras las necesidades de los usuarios de su aplicación, usted la creará para que la utilicen todas las personas. Los usuarios serán capaces de transitar fácilmente por ella, siguiendo una trayectoria lógica, intuitiva y no lineal, marcada con condiciones claras. Una de las maneras de dar instrucciones es representarlas mediante controles. Por tanto una aplicación bien diseñada debe guiar a los usuarios de un punto a otro, permitiéndole salir en cualquier momento, convirtiéndose, los controles que usted coloque en su aplicación en señales del camino que los usuarios seguirán, para no perderse.

### **Guías para el Diseño de una Interfaz**

Existen diferentes maneras de enfocar la tarea de diseñar una interfaz y construir una aplicación. No obstante, usted puede seguir estos pasos generales:

1. Cree un nuevo proyecto (o utilice el proyecto elaborado cuando usted inició Visual Basic) para organizar las partes de su aplicación.
2. Elabore una forma para cada ventana en su aplicación
3. Dibuje los controles para cada forma.
4. Cree una barra de menús para la forma principal.
5. Establezca propiedades de forma y control.
6. Escriba procedimientos de evento y procedimientos generales.
7. Guarde su trabajo.
8. Depure su código.
9. Cree un archivo ejecutable para convertir el proyecto en una aplicación.
10. Cree un icono sobre el que pueda hacer clic para abrir la aplicación desde el Program Manager (Administrador de programas). (Usted puede crear su propio icono con la aplicación Icon Works que se incluye en Visual Basic o puede utilizar cualquiera de los cientos de iconos que se incluyen en la biblioteca de iconos de Visual Basic).

Para analizar el diseño de una interfaz, observemos nuevamente el ambiente de trabajo de Visual Basic, ¿qué es lo primero que nota?, ¿qué piensa de las diferentes ventanas conforme las explota?, ¿son fáciles de acceder los menús y comandos?, ¿son útiles las cajas de herramientas, la barra de propiedades, la forma por omisión y las ventana-project?, ¿todos estos elementos hacen atractiva o limitativa la utilización de Visual Basic? y lo que es más importante, ¿le ayudan a usted?

Sería ideal que el contenido de una aplicación y su diseño se complementaran uno al otro, en donde uno reflejará la intención del otro. Es decir, cualquier control que usted cree debe servir para una función obvia y debe ayudar a los usuarios a transitar por toda la aplicación.

### **1.5.9. Alcance de las Variables Locales y Globales**

Todos los lenguajes de programación de alto nivel funcionan sobre el principio de establecer un alcance de variables locales y globales. El *alcance de variables* se refiere a cómo una aplicación reconoce cada una de las variables que se han declarado en la aplicación. En Visual Basic, el alcance está determinado sobre la base de dónde declara usted las variables: en el módulo global, o en un módulo en la forma misma.

Las variables y constantes son con frecuencia declaradas en el módulo global de la aplicación. (Usted no puede ubicar procedimientos o funciones en el módulo global, únicamente puede declarar variables, constantes y tipos). El colocar las declaraciones aquí, le brinda a las variables su más amplio alcance, por eso el nombre de "global". Su colocación también las pone en el nivel más lejano de alcance en una aplicación. La ventaja de declarar variables en el nivel global, es que siempre mantienen su valor en toda la corrida de la aplicación. En otras palabras, las variables son persistentes.

Las variables a nivel modular se incluyen en la sección de declaraciones en un módulo separado, no en la sección de una forma. La ventaja de declarar las variables en un módulo, es que todos los procedimientos en la aplicación pueden reconocer y compartir

las variables. Esto es importante cuando usted tiene más de una forma en una aplicación refiriéndose a los mismos procedimientos generales y funciones.

Por ejemplo, usted solamente tiene que declarar la variable una vez en el procedimiento general de un módulo. Cualquier procedimiento de evento dentro de toda la aplicación puede entonces llamar al procedimiento general.

Las variables a nivel forma se incluyen en la sección de declaraciones de un procedimiento general de forma. Usted declara siempre las variables a nivel de forma cuando desea que varios procedimientos dentro de la forma compartan datos; una variable a nivel de forma es reconocida por todos los procedimientos asignados a la forma. Tanto los niveles de forma como las de nivel modular se consideran también persistentes; mantienen sus valores después de que un procedimiento ha sido ejecutado, pero sólo dentro de la forma.

Las variables locales se reconocen solamente por el procedimiento en el cual aparecen (de ahí el nombre de "local"). También tiene el nivel más cercano de alcance. Utilizar una variable local es una buena manera de declarar cualquier variable que es temporal, porque el valor de la variable puede con frecuencia cambiar en un procedimiento. Además, cada procedimiento puede declarar una variable con el mismo nombre; usted puede nombrar muchas variables locales con el mismo nombre, en tanto las coloque en procedimientos separados. Si usted cambia un nombre de variable, las otras variables con el mismo nombre no serán afectadas.

Es necesario tener en mente que pueden ocurrir conflictos cuando usted declara una variable local con el mismo nombre que le ha dado a una variable en otro nivel.

Como una variable local tiene el más cercano nivel de alcance, cualquier referencia a la variable será accesada primero. Si una variable con el mismo nombre existe a un nivel más alto, la instrucción en la cual se encuentra la variable podría no ejecutarse adecuadamente, dando como resultado un error. Cuando se depura un programa, no

hay nada más frustrante que descubrir que el culpable del conflicto es un nombre de variable. Cuando le sea posible declare una variable en un sólo lugar.

## 1.6. CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE ACCESS 7.0

### 1.6.1. Introducción a Access

Microsoft Access es un programa para el manejo de bases de datos relacionales que *permite el almacenamiento, agrupamiento y búsqueda rápida de todo tipo de datos indispensables en las labores diarias de una empresa que se combina con la facilidad del uso que permite Windows.*

Microsoft Access, cuenta con herramientas sencillas e intuitivas, que proporcionan la potencia de una base de datos de características complejas para tareas cotidianas de administración de datos.

Las bases de datos de Microsoft Access están constituidos por diversos objetos: *tablas, consultas, formularios, informes, macros y módulos.* Estos objetos se listan en la ventana de datos, donde se crean, se abren y se utilizan. (Ver figura 1.6.1.1)

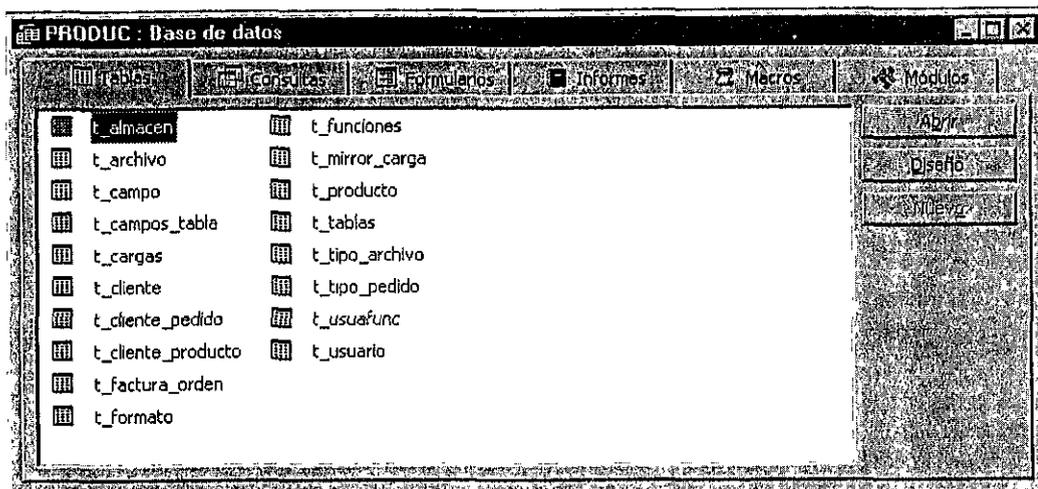


Figura 1.6.1.1. Objetos de Access

La estructura de Access, no es muy compleja, sin embargo requiere de varios elementos que lo hacen que sea potente y al mismo tiempo fácil de manejar por los usuarios, desde bases de datos sencillas como una agenda o un directorio telefónico, hasta aquellas grandes y complejas como un inventario o una base de datos que controle las compras, ventas, y nómina de una empresa. La figura 1.6.1.2 nos muestra la estructura de Access 97.

## **1.6.2. Diseño de una Base de Datos**

Para poder obtener el máximo provecho de las herramientas con que cuenta Access, es importante saber diseñar correctamente una base de datos.

Los pasos a seguir para el diseño de una base de datos son:

1. Determinar el propósito de la base de datos.
2. Determinar las tablas.
3. Determinar los campos.
4. Determinar las relaciones entre tablas.
5. Depurar el diseño.

Podemos mencionar además que una base de datos está formada por una o más tablas. Por lo general, varias tablas que se relacionan entre sí de alguna manera conforman una base de datos. Una tabla contiene filas y columnas. (Ver figura 1.6.2.1).

Al utilizar Microsoft Access, se puede administrar toda la información desde un único archivo de base de datos. Dentro de este archivo, se dividen los datos en contenedores de almacenamiento separados, denominados tablas; en los cuales se puede ver, agregar y actualizar los datos mediante formularios en línea; buscar y recuperar solamente los datos que se necesiten mediante consultas, analizar o imprimir los datos con el esquema deseado mediante informes.



Bases de datos: Una base de datos es un conjunto de información relacionada con un asunto o con una finalidad.



Tablas: Una tabla es un conjunto de datos sobre un tema específico. Las tablas organizan los datos en columnas (denominados campos) y filas (denominados registros).



Las consultas se usan para ver, cambiar y analizar datos de distintas maneras. También se pueden usar como origen de registros para formularios e informes.



Los formularios pueden crearse para la entrada de datos a una tabla, para ejecutar acciones con los datos introducidos en el formulario, o para abrir otros formularios o informes.



Un informe es una forma efectiva de presentar los datos en formato impreso. Como tiene control sobre el tamaño y el aspecto de todos los elementos de un informe, puede presentar la información

en la forma que se desee verla.



Una macro es un conjunto de una o más acciones para realizar una operación determinada, tal como abrir un formulario o imprimir un informe.



Un módulo es una colección de declaraciones y procedimientos de Access Basic para aplicaciones que se almacenan juntos como una unidad.

Un evento es una acción determinada que se produce en, o con, un objeto particular. Microsoft Access puede responder a una variedad de eventos: clics del mouse, cambios en los datos, formularios que se abren o se cierran, y muchos otros. Los eventos son normalmente el resultado de una acción del usuario.

**Figura 1.6.2.1. Estructura de Access**

El poder de manejo de una base de datos relacional como Access incide en su rápida búsqueda y extracción de información almacenada en tablas separadas.

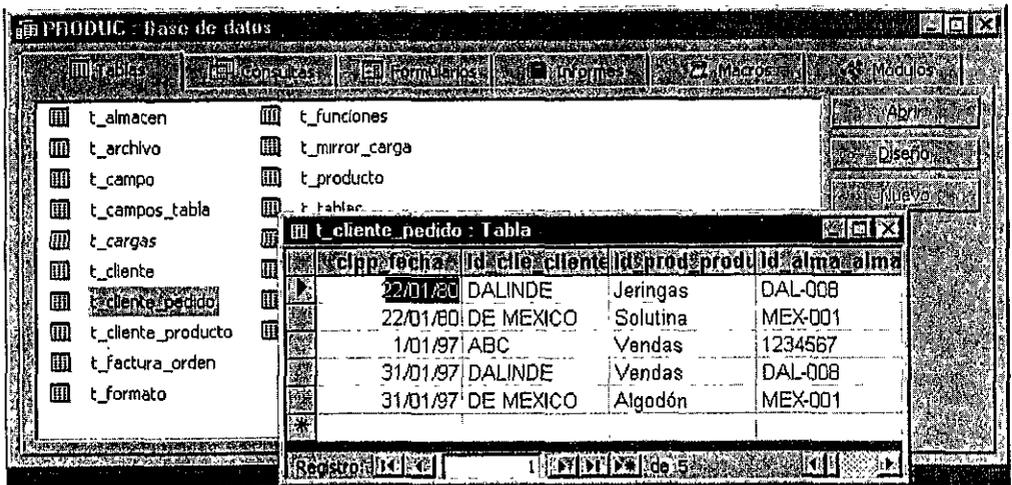
### 1.6.3. Objetos de Access

#### Características de las Tablas

Antes de crear cualquier otro objeto de una base de datos, se deben crear primero las tablas y la definición de tabla que se ha proporcionado en la figura 1.6.2.1., la cual hace referencia a una colección de datos acerca de un tema en particular y los datos son representados en una forma matricial en columnas y renglones o campos y registros respectivamente.

Cada fila de una tabla es un registro. El registro está formado por toda la información relacionada con una entrada en la tabla.

Cada columna de la tabla es un campo. Un campo es la porción más pequeña de información que contiene un registro, tal como se muestra en la figura 1.6.3.1.



| fecha    | id_cliente | cliente  | id_producto | producto | id_almacen | almacen |
|----------|------------|----------|-------------|----------|------------|---------|
| 22/01/80 | DALINDE    | Jeringas | DAL-008     |          |            |         |
| 22/01/80 | DE MEXICO  | Solutina | MEX-001     |          |            |         |
| 1/01/97  | ABC        | Vendas   | 1234567     |          |            |         |
| 31/01/97 | DALINDE    | Vendas   | DAL-008     |          |            |         |
| 31/01/97 | DE MEXICO  | Algodón  | MEX-001     |          |            |         |

Figura 1.6.3.1. Estructura de una tabla de Microsoft Access.

En una base de datos puede haber muchas tablas, cada una de las cuales almacena información sobre un tema distinto. Cada tabla puede contener datos de muchos tipos, incluyendo texto, números, fechas e imágenes.

Se definen relaciones entre tablas para recuperar datos de varias tablas en una consulta, formulario e informe.

Al diseñar las tablas es necesario especificar un identificador exclusivo de registros, mismo que se le denomina llave primaria. Ésta consiste en uno o más campos que identifican a cada registro almacenado en la tabla.

Las reglas de validación que se establezcan para las tablas y sus campos siempre estarán en vigor, tanto si cambian los datos en un formulario, como si se importan nuevos datos o se utiliza una consulta de acciones, una macro o código de Access para cambiar datos.

Estableciendo las propiedades de los campos en el modo de presentación Diseño de Tabla es posible controlar la apariencia de los datos, evitar su introducción incorrecta, especificar valores predeterminados y acelerar la búsqueda y la ordenación en la tabla.

Cuando se crea una tabla, ésta no está relacionada con las demás tablas de la base de datos. Para crear una relación, se debe abrir la ventana de relaciones y añadir las tablas o consultas que se desean relacionar.

Microsoft Access, cuenta con una ventana gráfica llamada relaciones, que permite realizar de forma visual un boceto del diseño de la base de datos y observar simultáneamente todas las relaciones de la base de datos.

Gracias a las actualizaciones y eliminaciones en cascada, ahora se puede actualizar un campo de una tabla y hacer que Access elimine automáticamente los registros relacionados de las tablas relacionadas.

## Características de las Consultas

Una consulta es la respuesta a una serie de preguntas acerca de datos almacenados en tablas e incluso en consultas.

Microsoft Access, cuenta con el asistente para consultas, que nos ayuda a crear consultas complejas para tareas frecuentes de administración de datos.

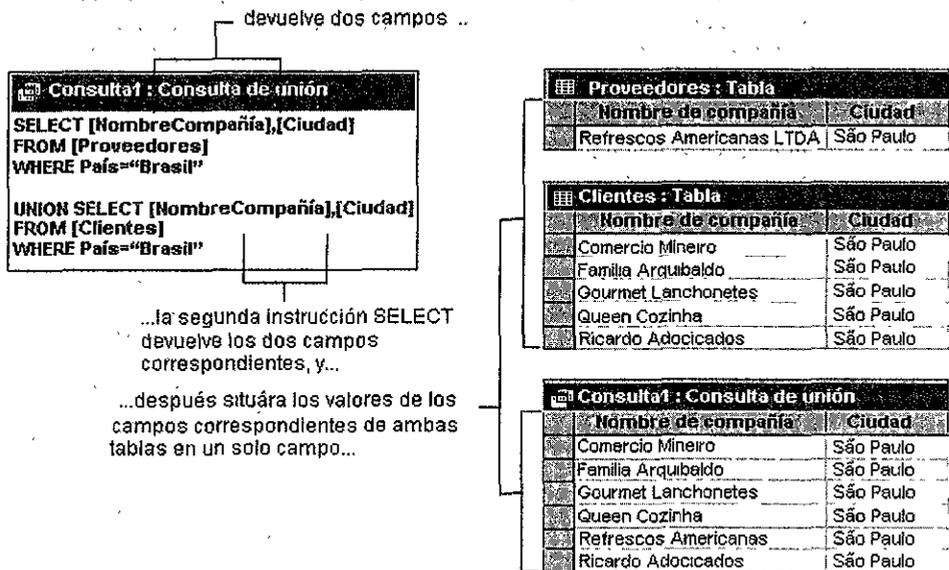
Se crean consultas para buscar y recuperar exactamente aquellos datos que cumplen determinadas condiciones que se especifican. Una consulta también puede actualizar o eliminar múltiples registros al mismo tiempo, así como realizar sobre los datos diversos cálculos incorporados o personalizados. Figura 1.6.3.2.

Aunque no se hayan establecido relaciones entre tablas, Access crea automáticamente combinaciones en una consulta entre tablas que contienen campos coincidentes.

Cuando una consulta incluye campos de varias tablas relacionadas, pueden actualizarse los datos a ambos lados de la combinación. También se puede definir una consulta de selección para limitar el número de registros devueltos.

Existen cuatro modos de presentación de consultas:

1. **Diseño:** Se utiliza para crear nuevas consultas o modificar las ya existentes.
2. **Hoja de datos:** Se usa para la presentación de datos recuperados de una consulta y en algunos casos agrega y edita datos.
3. **Presentación preliminar:** Para ver la apariencia que tendrán los datos de la consulta una vez impreso.
4. **Presentación SQL (Structured Query Language):** Para cuando se desee crear o modificar una consulta mediante inscripción SQL.



**Figura 1.6.3.2. Consulta de Microsoft Access.**

Access cuenta con los siguientes tipos de consulta:

- Selección: Son las más comunes, aquí se plantean preguntas sobre datos almacenados en las tablas y la respuesta es una hoja de datos, en donde se puede ver y cambiar los datos de las tablas subyacentes.
- Referencia cruzada: Presenta los datos con títulos en las filas y columnas, como en una hoja de cálculo. Con una consulta de tabla de referencias cruzadas se puede resumir gran cantidad de información en un formato de fácil lectura.
- Consulta de unión: Combina campos coincidentes de dos o más tablas.
- Consultas de definición de datos: Crea, modifica o elimina tablas de una base de datos Access utilizando inscripción SQL.
- Acción: Modifican muchos registros en una sola operación. Se utiliza para crear una nueva tabla, eliminar, agregar o modificar registros. A partir de la selección se actualizan, agregan y eliminan datos. Access provee cuatro tipos de consultas de

este tipo que son: consultas de creación de tablas, de eliminación, de agregación de datos y de actualización.

A la hora de diseñar las consultas, veremos que son muy flexibles, ya que podremos:

- Elegir campos de una o más tablas.
- Elegir registros mediante criterios que deben cumplir para que se incluyan en la hoja de respuesta dinámica de la consulta.
- Ordenar registros.
- Formular preguntas sobre datos de varias tablas.
- Realizar cálculos, es decir, crear nuevos campos que contengan el resultado de una operación aritmética sobre algún campo de todos o algunos registros.
- Usar una consulta como origen de los datos para formularios, informes y otras consultas. Utilizando una consulta se pueden incluir datos de varias tablas y establecer criterios para mostrar un solo conjunto limitado de datos. También se pueden introducir nuevos datos o modificar los existentes directamente en la hoja de datos de la consulta, o bien en un formulario que esté basado en esta consulta.
- Modificar los datos de las tablas, con las consultas de acciones puede actualizar, eliminar o agregar un grupo de registros simultáneamente. También puede utilizarlas para crear una nueva tabla que contenga registros de otra tabla o de un grupo de tablas existentes.

### **Características de Formularios.**

La creación de un formulario nos permite ver, introducir o cambiar datos directamente en una tabla de una manera sencilla. Cuando se abre un formulario, Microsoft Access recupera los datos de una o más tablas y los muestra en la pantalla usando el esquema que se eligió en el Asistente para Formularios o usando un esquema creado desde el principio. Figura 1.6.3.3.

Con Microsoft Access se pueden diseñar formularios fáciles de utilizar y que presenten la información de la forma deseada. Se pueden utilizar diversos elementos de diseño

(texto, datos, imágenes, líneas y colores) para crear exactamente el formulario deseado.

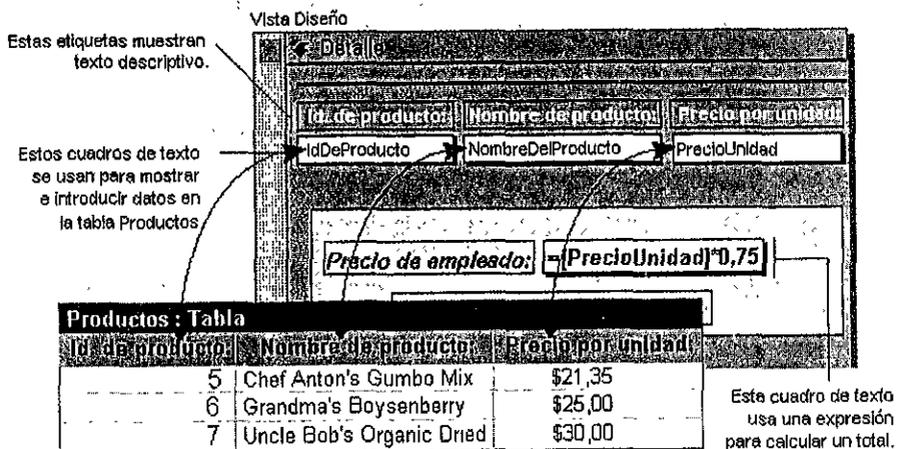
El uso de los formularios también es un modo eficiente de introducir los datos, pues ahorra tiempo y evita errores de mecanografía.

Los formularios se pueden presentar en los siguientes Modos :

- El Modo de Diseño se utiliza para crear nuevos formularios o modificar los ya existentes.
- El Modo de Formulario se utiliza para introducir, modificar y ver datos. En este modo generalmente se pueden ver al mismo tiempo todos los campos correspondientes a un registro.
- El Modo de Hoja de Datos se utiliza para introducir, modificar y ver datos utilizando para ello una hoja de datos. En este Modo de presentación se pueden ver muchos registros al mismo tiempo, pero puede que la hoja de datos sea más ancha que la pantalla, en cuyo caso no se podrán ver todos los campos correspondientes a cada registro sin desplazarse por el formulario.
- El Modo de Presentación Preliminar se utiliza para ver la apariencia que tendrá un formulario cuando se imprima.

Los formularios proporcionan una forma sencilla de ver los datos. Se pueden ver todos los valores de un registro en el Modo Presentación Formulario o bien se puede cambiar al Modo de Hoja de Datos para ver todos los registros del formulario.

El formulario ofrece el diseño más cómodo para introducir, modificar y ver los registros de la base de datos. Las herramientas de diseño para formularios de Access ayudan a diseñarlos de forma fácil y útil para:



**Figura 1.6.3.3. Formulario de Microsoft Access.**

- Presentar los datos en un formato atractivo, con fuentes especiales y otros efectos gráficos tales como colores y sombreado.
- Ofrecer una apariencia similar a los formularios en papel.
- *Calcular totales y presentarlos.*
- Contener gráficas de los datos presentados.
- Mostrar datos de varias tablas.
- Automatizar tareas que se realizan habitualmente.

Otros elementos de los formularios son los subformularios, que son un método para incluir en un formulario, información que proviene de más de una tabla. Un subformulario es un formulario dentro de otro formulario.

En Access, el formulario primario se denomina formulario principal y el formulario incluido en él se denomina subformulario.

Al utilizar un subformulario es sencillo ver la relación existente entre los registros de dos o más tablas.

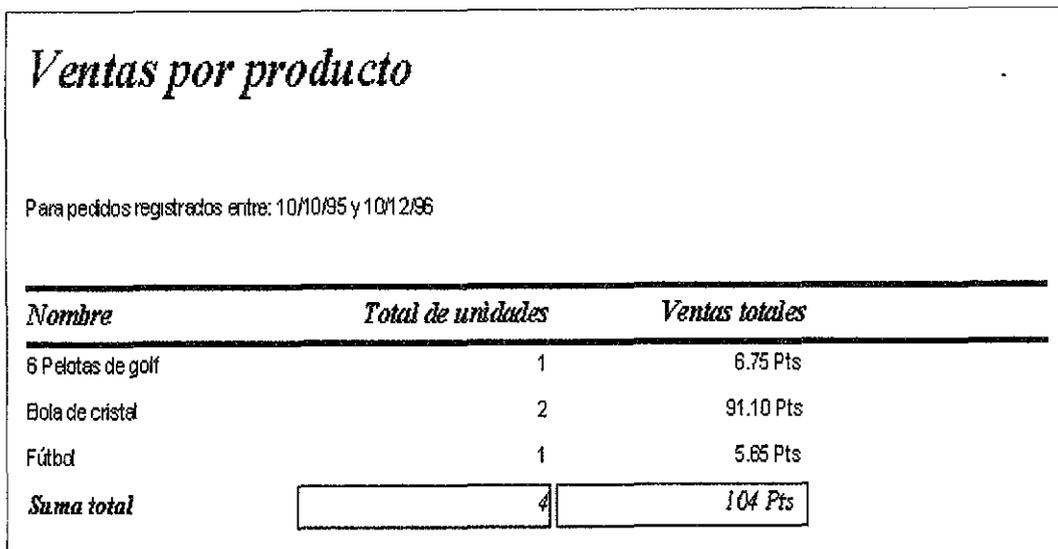
### Características de los Informes

Un informe es un conjunto de información que se puede organizar según determinados criterios y al que se le aplica formato de acuerdo con determinadas especificaciones.

Un informe constituye una forma de recuperar y presentar los datos como información significativa que se puede usar y distribuir. Ver figura 1.6.3.4.

Con Access se pueden diseñar informes que presenten la información de la forma deseada. Se pueden utilizar numerosos elementos de diseño (texto, datos, imágenes, líneas, cuadros y gráficos) para crear exactamente el informe que necesita.

Crear un informe es un método muy eficaz de presentar los datos en forma de documento impreso. Si bien también es posible imprimir los formularios y las hojas de datos, los informes brindan mayor flexibilidad a la hora de presentar información de resumen.



*Ventas por producto*

Para pedidos registrados entre: 10/10/95 y 10/12/96

| <i>Nombre</i>     | <i>Total de unidades</i> | <i>Ventas totales</i> |
|-------------------|--------------------------|-----------------------|
| 6 Pelotas de golf | 1                        | 6.75 Pts              |
| Bola de cristal   | 2                        | 91.10 Pts             |
| Fútbol            | 1                        | 5.65 Pts              |
| <b>Suma total</b> | <b>4</b>                 | <b>104 Pts</b>        |

Figura 1.6.3.4. Informe de Microsoft Access.

Los informes, al igual que otros elementos de Access se presentan de diversos modos:

- El modo de diseño, se utiliza para crear un informe o modificar alguno ya existente.
- El de presentación preliminar se utiliza para comprobar los datos de todo el informe.
- El modo muestra preliminar se utiliza para comprobar la fuente, el tamaño de la fuente y el diseño general de un informe.

Access ofrece varios métodos para obtener información de la base de datos: utilizar una consulta, formulario o un informe, por lo que se deberá seguir el método mas adecuado para la tarea que se requiera realizar.

Los informes se pueden crear para:

- Organizar y presentar los datos en grupos.
- Calcular totales parciales, totales de grupo, sumas de totales y porcentajes de totales.
- Incluir subformularios, subinformes y gráficas.
- Presentar los datos en un formato atractivo con imágenes, líneas y fuentes especiales

### **Características de las Macros**

Una macro realiza automáticamente una tarea o una serie de tareas. Cada tarea que se desea realizar con Access se denomina acción.

Access ofrece una lista de acciones, entre las que se puede elegir una o más para crear las macros. Cuando se ejecuta la macro, Access realizará todas las acciones en el mismo orden en que figura dentro de la macro, utilizando los objetos o los datos específicos.

Mediante el uso de macros se puede conseguir que los formularios, los informes y los demás objetos de la base de datos funcionen conjuntamente de forma aún más inteligente.

Las macros también son útiles para automatizar tareas rutinarias o repetitivas, como la impresión de informes semanales. Además, las macros ofrecen versátiles posibilidades de personalización tales como realizar tareas complejas de validaciones de los datos introducidos en formularios.

Las macros nos permiten:

- Hacer que los formularios e informes funcionen conjuntamente.
- Buscar y filtrar registros automáticamente.
- Establecer la exactitud de los datos, las macros también son idóneas para el manejo de datos y para su validación en formularios.
- Establecer propiedades de formularios, informes y controles. Se pueden utilizar las macros para establecer la mayor parte de las propiedades de formularios, informe y control.
- Automatizar transferencias de datos, es decir, para importar o exportar datos entre diferentes formatos de archivos.
- Crear nuestro propio entorno de trabajo personalizado.

### **Características de los Módulos**

Los módulos almacenan código de Access que puede escribirse para personalizar, mejorar y extender las posibilidades de la base de datos.

Para mayor información de código, consultar el manual de Microsoft Access.

### **Desarrollo de Access**

La primera versión de Microsoft Access apareció en 1992, como el primer sistema manejador de bases de datos relacionales (RDBMS) diseñado para el sistema operativo

Windows. Durante su vida, Microsoft Access ha continuado mejorando sus características y sobre todo en rapidez, facilidad y manejo de datos.

Cada versión que aparece de Access, ha introducido nuevas tecnologías revolucionadas que han cambiado el aprendizaje de los usuarios y del manejo de las bases de datos. Access es el primer manejador de bases de datos que soporta **OLE Automation** (Automatización del Ligado Dinámico de Objetos) que permite a los usuarios compartir información entre aplicaciones. Access versión 2.0 introduce herramientas tales como gráficas de relaciones y constructores de consultas. Las formas de Access y el Asistente para Reportes ayudan a los usuarios con poca o nada de experiencia a ser productivos rápidamente.

Como usuarios actualizados a Windows 95, Access es el primer RDBMS de 32 bits e *introduce tecnologías tales como las tablas y **Performance Analyzers** (Analizadores de funcionamiento) y **Briefcase Replication** (Portafolio de replicación), quienes manejan algunas de las tareas más complejas en el manejo de bases de datos automáticamente. Estas características crearon dificultades históricas de la tecnología de acceso a bases de datos para que los usuarios generaran negocios.*

Microsoft Access continúa su tradición de innovación. Aunque no se sabe si los usuarios estén conectados por una red de área local o a través de Internet o que estén trabajando en Microsoft. Access 97 se asegura que los usuarios puedan darse cuenta de los beneficios de usar una base de datos rápidamente. Con estas nuevas tecnologías integradas Access 97 fue diseñado para hacer más fácil para todos los usuarios encontrar respuestas, compartir información al mismo tiempo y crear soluciones rápidas.

### **Nuevas Características de Access 97**

Las nuevas características de Access 97 se enfocan en las herramientas que lo forman, las cuales están organizadas en tres grupos:

- Herramientas que ayudan a los usuarios a encontrar respuestas fácilmente.

Con las herramientas que ayudan a encontrar respuestas fácilmente, Microsoft Access 97 introduce nuevas características que hacen a los usuarios obtener respuestas a sus preguntas rápida y fácilmente.

Históricamente, las bases de datos han sido una herramienta poderosa de los usuarios, al igual que los manejadores de información y desarrolladores de soluciones, también cabe mencionar que las computadoras personales han sido trasladadas de los laboratorios de computadoras a los hogares y pequeños negocios. Sin embargo, las necesidades de los usuarios de bases de datos han cambiado.

Hoy los usuarios de bases de datos manejan muchos grupos de datos o uno simple que deben ser compartidos entre muchos usuarios. Los datos han ido creciendo, no solamente con textos o números, sino datos visuales y audibles. Los usuarios demandan una aplicación intuitiva y fácil de usar que ayude a obtener el mejor valor de su información.

- Elementos nuevos:

Asistente de Office: Es un paso tecnológico que unifica el asistente en línea del usuario, permite al usuario realizar preguntas en palabras relacionadas y proveer interactivamente sugerencias útiles.

El asistente es controlado a través de la automatización del lenguaje Microsoft Visual Basic para construcción de aplicaciones desarrolladas basadas sobre Access pudiendo tener ventajas sobre su funcionalidad. Ver figura 1.6.3.5.

- Elementos mejorados:

Asistente de Consultas: Ayuda al usuario a localizar exactamente la información que se busca, a pesar del nivel de experiencia en bases de datos en una forma rápida.

**Asistente de Base de Datos:** Esta herramienta permite a los usuarios sin ninguna experiencia en bases de datos, crear una o más de 20 aplicaciones de bases de datos funcionalmente completas.

**Filtros por forma:** Los usuarios pueden localizar la información que buscan para visualizarla, usando las formas, hojas de datos o formularios que trabajan diariamente.

**Asistente Analizador de Tablas:** Esta herramienta analiza la estructura de los archivos de datos llanos, es el encargado del proceso de normalización, además de que realiza y brinda recomendaciones para mejorar la base de datos relacional que el usuario puede aceptar o rechazar.

**Asistente de Formas y Reportes:** Estas herramientas crean tablas y reportes de información de bases de datos con pequeñas intervenciones de los usuarios.

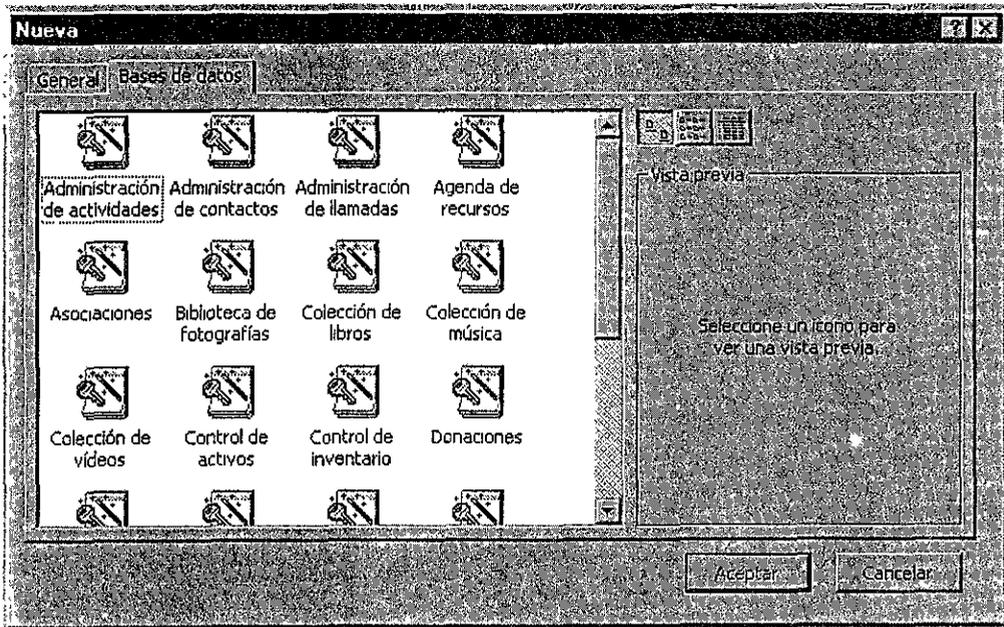
Además dentro de Microsoft Access, pueden utilizarse máscaras de entrada para definir patrones que deban seguir los datos cuando se introduzcan en formularios y hojas de datos, lo que confiere una mayor rapidez y precisión a la introducción de éstos.

Microsoft Access, puede ordenar rápidamente los registros de tablas, consultas y formularios, en forma ascendente o descendente. Puede ajustar la anchura de una columna de la hoja de datos para que se adapte a su valor más ancho. Además puede imprimir información detallada sobre el diseño de cualquier objeto de la base de datos.

#### **1.6.4. Herramientas para Compartir Información al Mismo Tiempo**

Hoy en día, acceder a la información correctamente en el momento correcto, puede significar la diferencia entre sucesos y fallas. La receta surge con la popularidad de Internet e Intranets. Como resultado de lo anterior Microsoft Access 97 incluyó nuevas características que permiten a los usuarios obtener más de sus ambientes computacionales ya instalados.

Desde un principio, las bases de datos han sido herramientas que pueden ser usadas por más de una persona. El propósito de las bases de datos es proveer una localización central para almacenar piezas de información relacionadas. Esta información puede ser organizada o vista en muchas formas como aplicaciones para datos.



**Figura 1.6.3.5. Asistente para crear bases de datos empleando plantillas.**

Las extensiones naturales de la funcionalidad de bases de datos están disponibles en Microsoft Access 97 ayudando a compartir a los usuarios información con un grupo de trabajo, en una Intranet o Internet.

Microsoft Access 97 es una de las primeras aplicaciones de bases de datos que soporta al almacenamiento de hiperligas como un tipo nativo, la cual ofrece muchos beneficios como ir a los **URLs** (Hipervínculos), o bien para localizar otros documentos de Office 97.

Elementos nuevos:

**Hiperligas:** Los usuarios pueden almacenar hiperligas en todas las bases de datos de Access para conectarse fácilmente a información no encontrada en el lugar donde reside.

**Salvar a HTML (Formato de páginas en Internet):** Los usuarios pueden publicar vistas estáticas de datos para su grupo de trabajo o para el **WEB** (Red de Internet).

**Asistente para publicar en el WEB:** Esta herramienta automatiza las publicaciones de información de bases de datos dinámicamente en el WEB.

**Importación y ligado de archivos HTML:** Los usuarios pueden ahora importar o atar sus bases de datos a páginas HTML y usar esta información dentro de las aplicaciones de Microsoft Access.

**Replicación en Internet:** Los usuarios pueden extender la capacidad de replicación introducida en Access 95 bajo Internet usado FTP (Protocolo de transferencia de archivos).

**Herramientas para la Creación de Soluciones Rápidas.** En suma para engrandecer el rendimiento, Access ha mejorado substancialmente el ambiente Access 97 y trae nuevas herramientas orientadas al desarrollo. Éstas mejoran el camino de los desarrolladores en rapidez y como crean sus aplicaciones.

Nuevas herramientas:

**Mejora el rendimiento:** Cambia el diseño de aplicaciones y la estructura de objetos, resultando en mejoras del rendimiento del sistema en más de 50%.

**Replicación parcial de tablas:** Los usuarios ahora pueden replicar un grupo seleccionado de registros de cualquier tabla.

Control de código fuente: Múltiples desarrolladores pueden crear ahora soluciones y mantener código y objetos consistentes usando el control de código fuente.

Herramientas mejoradas:

Asistente Analizador de Rendimiento: Esta herramienta analiza la estructura de la base de datos y sugiere las áreas para mejorarla.

Visual Basic para aplicaciones: Engrandecimiento del ambiente de diseño integrado haciendo a Access un ambiente de desarrollo más robusto para las aplicaciones de bases de datos.

Como parte de Microsoft Office 97, Access ofrece la posibilidad de manejar bases de datos obtenidas a través de Internet. Copiadas de la red a un equipo particular. Lo interesante es que permite ligar la base de datos a una hoja WEB, haciendo que cada vez sea consultada, el sistema la actualice automáticamente accedando la página correspondiente en WEB, sin importar si ocupa una o más páginas.

### **1.6.5. Especificaciones de uso de Access**

Para utilizar Microsoft Access se necesita:

- Computadora personal procesador 486 o superior (con multimedia de preferencia).
- Sistema operativo Windows 95, Windows NT como estación de trabajo 3.51 o superior.
- 12 Mb de memoria RAM para Windows 95 y 16Mb para estaciones de trabajo Windows NT.
- De 28 a 60 Mb de espacio en disco duro dependiendo de la configuración y 40 para la estación típica.
- Manejador de CD-ROM.
- CD-ROM que contenga Internet Explorer, manejadores adicionales y archivos con extensión **AVI** (Interfaz de Audio y Video).

- Adaptador de video VGA mínimo (se recomienda super VGA a 256 colores).
- Microsoft Mouse. Mocosoft IntelliMouse o cualquier puntero compatible.

Opciones o servicios adicionales para usar ciertas características:

- Modem con una velocidad de 9600 baudios (se recomienda 14400 baudios).
- Computadora multimedia, cuando se requieren efectos de sonido u otros efectos de Access.
- La característica de publicación al WEB requiere Microsoft Server para Windows NT o *Microsoft Personal WEB Server para Windows 95*.
- Alguna funcionalidad de Internet puede requerir acceso a Internet y el pago del servicio por separado.

## **2. PLANTEAMIENTO DEL PROBLEMA**

### **2.1. DEFINICIÓN DEL PROBLEMA**

Como en gran parte de las empresas que venden sus productos a una amplia variedad de clientes y que con procesos manuales, manipulan un considerable volumen de información; se corren riesgos inminentes, tales como el desconocimiento del flujo de la misma, su explotación e interpretación incorrecta, su archivamiento inadecuado o la imposibilidad de llevar a cabo cualquier cambio en el esquema de funcionamiento actual; que sin duda, afectan el rendimiento general de la Organización. Es necesario por tanto, identificar los puntos en donde tales empresas se encuentren con obstáculos que les impidan llevar un control interno eficaz.

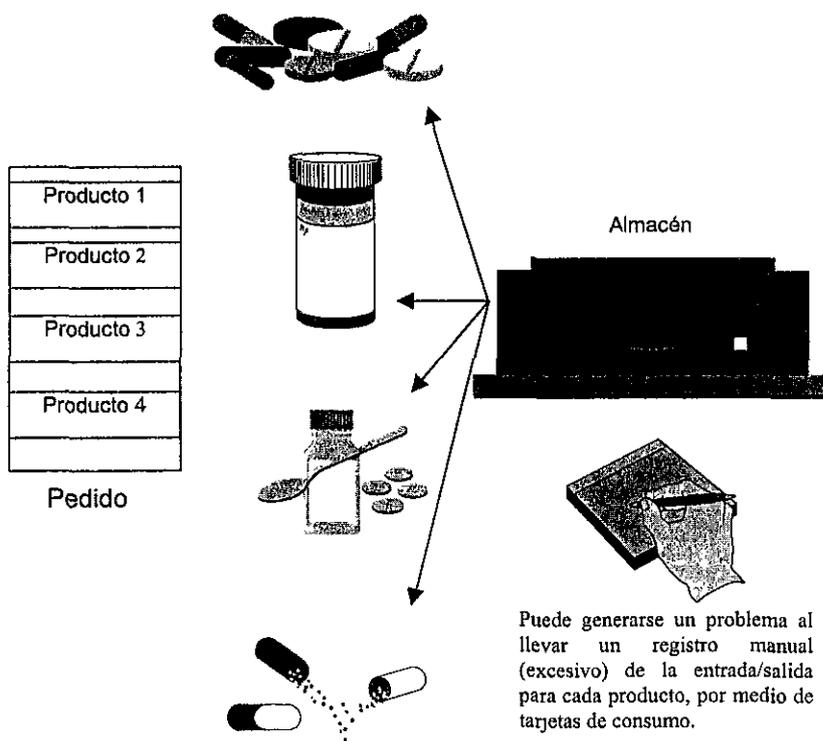
Los riesgos de enfrentarse frecuentemente con problemas generados por un control insuficiente de la información, no son deseables para ningún proveedor y mucho menos si tomamos en cuenta que se encuentran inmersos en un mercado con una demanda en continuo aumento y con una nutrida competencia, que a su vez tratará de esforzarse no sólo de evitar sus propios problemas sino de encontrar los medios necesarios para no quedarse atrás y ajustarse lo más rápidamente posible a los cambios que se puedan presentar. La necesidad de una correcta interacción del proveedor y los clientes resulta más evidente cuando se llegan a presentar problemas que los hagan entrar en contradicciones, los cuales pueden repetirse de un cliente a otro e ir incrementando el grado de descontrol de una o ambas partes. En el caso del proveedor, las diferencias que se tengan con un cliente no necesariamente serán evidentes al momento de cumplir con el pedido, sino que muchas veces, éstas pueden tener tras de sí un largo tiempo de haberse efectuado, por lo que para su aclaración habría que remontarse atrás un tiempo equivalente, para muchas veces llegar a la conclusión de que todo fue provocado por un error interno. Sin embargo, cualquiera que sea el caso que ocasione el error, se debe procurar una respuesta aceptable en ambas partes y que en un futuro no se vuelva a repetir o su ocurrencia sea mínima. No

es, sino cuando ocurren los problemas, que son más evidentes las deficiencias de una o las dos partes a la vez, muchas veces por incompatibilidad en las fuentes de información que fluyen entre ambas, y otras tantas por deficiencias de control interno propias de cada una. Obviamente las carencias para enfrentar los problemas comunes que involucren a un proveedor y a un cliente, no deberían tener una solución aislada de alguna de las partes, si es que se desean corregir de verdad; si no más bien una colaboración conjunta pudiese llevar a una solución más efectiva. Puede entonces surgir la necesidad de formar un medio común a través del cual puedan establecer un puente de comunicación común que les facilite la detección y corrección oportuna de sus problemas, mucho antes de que estos se agraven.

El procesamiento manual de datos puede complicarse con el transcurrir del tiempo por el incremento en la cantidad de información que se tiene que manejar para: registrar la *entrada/salida de productos de un almacén* debido al cumplimiento de algún pedido, para elaborar de forma continua un inventario de las existencias o aún para llevar un control de los clientes, entre otras cosas, que desembocarán en las expectativas del negocio dependiendo en gran medida del balance de esta información complementada con su situación financiera. De lo anterior, podemos decir que se trata de los puntos en donde se tienen que enfocar los esfuerzos para agilizar tanto los tiempos de respuesta como el control que cada uno requiera. La dinámica de un almacén puede arrojar datos muy interesantes desde el punto de vista del control, que pueden repercutir en datos que proporcionen información sobre costos de ciertos productos que no se consuman del todo, es decir, que permanezcan en el almacén un determinado tiempo tal que se deprecien con el transcurrir del tiempo, siendo que otros pueden tener un consumo completo, que por tanto influirá en el tipo de control que se aplique a cada producto. Así es como el paso por el almacén de algunos productos sería prácticamente nulo, no así para otros que deberían de incluirse frecuentemente en un inventario para en todo caso plantear su posible reemplazo por deterioro u obsolescencia. En el caso de los primeros se manejan comúnmente las tarjetas de consumo que resultan una forma *eficaz de controlar su existencia y paso por el almacén*, estas tarjetas son elaboradas



En la figura 2.1.2 se presenta un diagrama que muestra el registro de productos en el almacén para un proveedor, al momento de surtir un pedido.

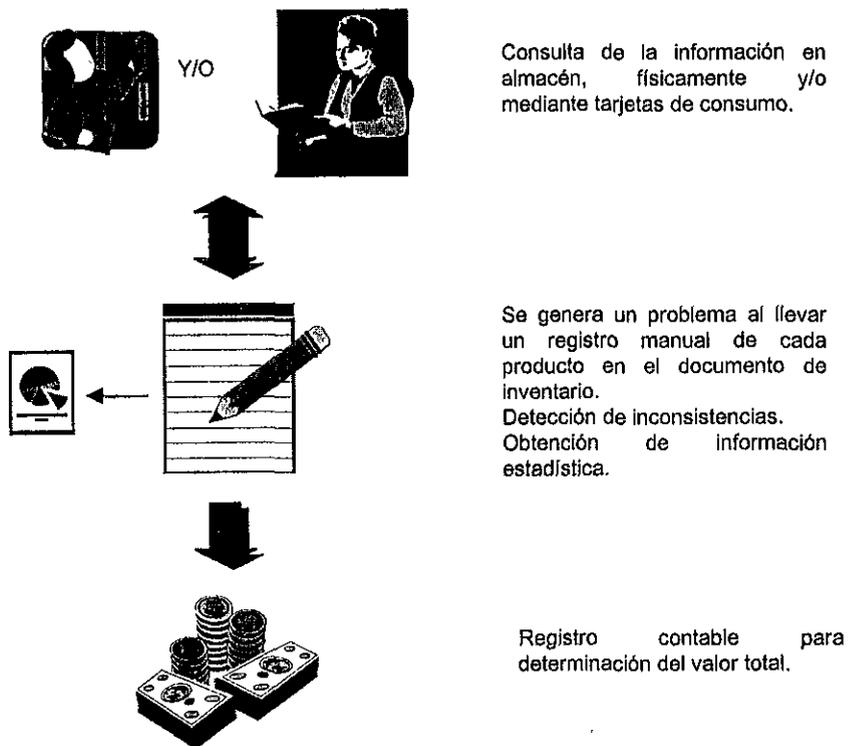


**Figura 2.1.2. Registro manual de entradas y salidas de productos del almacén.**

Para la elaboración continua de un inventario, la catalogación y ordenación apropiada de los productos puede llevar al proveedor a una mejor administración de sus recursos que le permitan tomar decisiones oportunas que le faciliten el cumplimiento a sus clientes. Este proceso lleva implícita una estrecha relación con el funcionamiento del

almacén ya que generalmente recurren a la misma fuente de información –las Tarjetas de Consumo-, que como hemos mencionado, puede prestarse a una mala interpretación o explotación, arrojando algunas inconsistencias en los resultados del mismo; otro punto a tratar son los tiempos que un inventario físico requiere para su ejecución y que son ocasionados muchas veces no por la revista física que se tenga que hacer de los productos sino por la consulta manual que puede consumir gran parte del tiempo, y la conciliación contra la información a la que se está recurriendo; el proceso se complica si se desea generar alguna información estadística que pueda ser de utilidad en la toma de decisiones de la empresa; la periodicidad con la que se elabore también es otro factor que influye en éste ya que muchas veces el período de tiempo entre uno y otro depende del mismo grado de desorganización que exista, ya que ningún proveedor estaría dispuesto a sacrificar demasiado en la realización de un inventario que sabe que le consumirá tiempo, esfuerzo y gasto de sus recursos humanos y financieros. Una vez que el inventario arroja sus resultados es necesario alimentar los registros contables para la evaluación de la existencia de mercancías ya que para la empresa representa dinero invertido que se sumará al que se posea en efectivo para arrojar en general el estado financiero de la entidad. En ocasiones los inventarios son considerados sólo como un conteo de productos siendo que pueden contribuir a representar la verdadera situación económica que guarda o guardó un Proveedor. Así como para el almacén pueden ser utilizadas herramientas manuales como las tarjetas de consumo, en el proceso de inventario se puede utilizar una herramienta análoga que permita comunicar a las partes correspondientes sus resultados, que posteriormente serán interpretados, o en todo caso, corregidos, para pasar finalmente a archivarlos. Algo muy importante que se debe asentar en una forma de inventario es el costo original de una mercancía así como el tiempo que se cree estará almacenada, en caso de que ésta tienda a permanecer por un tiempo en el almacén, ya que estos datos son muy útiles para el cálculo de la depreciación. No es raro que en ocasiones se requiera consultar estas formas de inventario de mercancías para rendir alguna aclaración o también para generar información estadística. En la figura 2.1.2 se presenta un diagrama que señala el curso para la elaboración del

## Inventario.



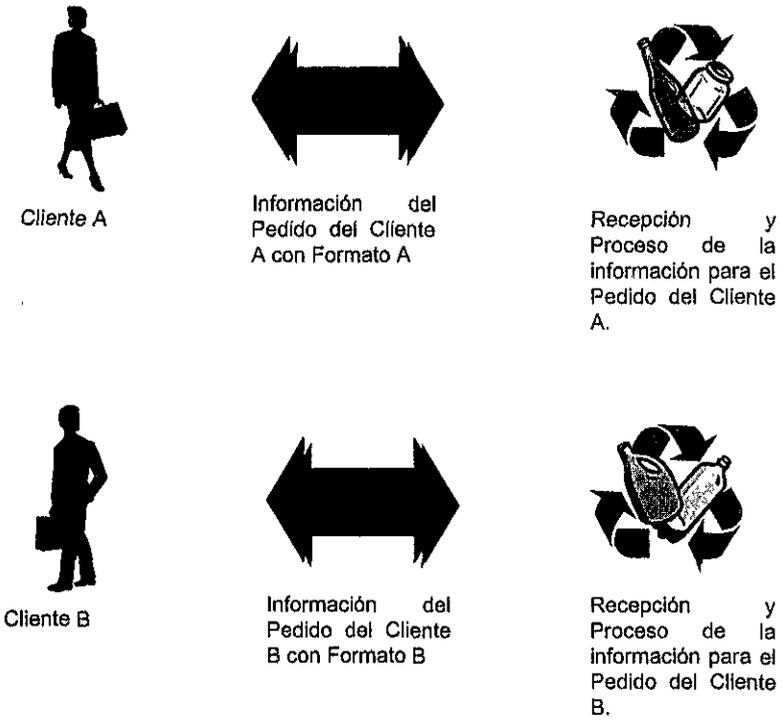
**Figura 2.1.3. Diagrama que señala el curso para la elaboración del inventario.**

De los procesos de almacenaje, proveeduría y de inventario se puede destacar un punto desfavorable que es inevitable durante su desempeño y que resulta necesario si es que ambos procesos se desempeñan en forma manual, éste es el de la redundancia en la captura de datos, es decir, dentro del almacén es necesario llevar un registro individual para cada producto con base en su entrada/salida y al realizar el inventario este producto será nuevamente registrado para asentar su estado en la forma correspondiente por lo que, tanto el producto, como sus atributos, tendrán que ser apuntados por lo menos una vez durante cada inventario. Si ahora, se incrementa

el número de mercancías, esto resultaría cada vez más complicado y requeriría, como se mencionó, mayor consumo de recursos humanos y financieros así como la tendencia a cometer más errores. Si no se tiene una fuente común hacia donde fluya y de donde fluya la información se tiende a caer en este doble registro. También puede presentarse el caso en que la decisión de entrada o no a un concurso para proveer a alguna institución dependa de la rapidez y confiabilidad con la que ambos procesos proporcionen su información para complementar a la financiera y generar una evaluación definitiva que facilite esa decisión.

El suministro a los clientes por parte del proveedor implica cumplir una serie de requisitos muchas veces dictados por instituciones gubernamentales y otras veces establecidos de común acuerdo entre el vendedor y el comprador. Una vez más el proceso manual de información que se maneja en ambas partes durante la realización del pedido por el consumidor y su cumplimiento por el proveedor, pueden generar inconsistencia al momento de cotejar la información de su contraparte, provocando en su caso, una o varias devoluciones de pedidos completos que requerirán de una revisión y evaluación cuidadosa del proceso de suministro y/o distribución de productos. Si sumamos a esto la diversidad de ventas que se pueden hacer: ventas a mayoristas, minoristas, ventas a instituciones médicas gubernamentales e inclusive exportaciones; así como la diversidad de clientes que cada uno de estos tipos de venta puede tener, y en donde cada uno puede proporcionar una lista propia que contenga los requisitos necesarios para su satisfacción; resulta cada vez más complejo adecuarse a esta variedad e implica tener un conocimiento particular de cada cliente y de los productos que consuma, dicho conocimiento se adquirirá sólo con base en el trato y la experiencia con él, lo cual no es deseable para ningún proveedor. Los clientes que consuman más frecuentemente, es decir, que soliciten más pedidos, son en los que más cuidado debe de tener un proveedor, aunque estos pedidos contengan frecuentemente los mismos tipos de productos cada vez que se solicitan variando sólo en la cantidad; sin embargo no siempre suele ser así ya que otras veces tanto la cantidad y frecuencia de pedidos como la diversidad de productos incluidos, dificulta el

proceso de suministro y por lo tanto de cotejo para ambas partes. Al cliente no basta con *satisfacerle su cuadro de requisitos establecidos, sino que además los tiempos de respuesta en caso de contingencias, consultas o aclaraciones sean aceptables* ya que pueden ser determinantes para que el proveedor sea incluido dentro de la lista de las instituciones consumidoras. Siempre que el vendedor carezca de un buen funcionamiento interno que le permita reaccionar en caso de fallas, tenderá a recurrir continuamente a gastos obligados para sobrellevarlas. En la figura 2.1.4 planteamos la idea del problema que existe al no tener formato común al momento de realizar un pedido y al momento de cumplirlo.



**Figura 2.1.4. Problemática por la no existencia de formato común.**

Los problemas presentados para una empresa de este tipo pueden tener variantes, dependiendo sobre todo de su esquema operativo, sin embargo, los definidos aquí pueden aplicarse a la mayoría de los casos. El problema fundamental radica en la integración adecuada de la información generada en los diferentes departamentos, su correcta explotación y sobre todo la interpretación y aprovechamiento que de ésta pueda efectuar la parte consumidora. Es importante apuntar que el incremento del procesamiento manual de la información, no implica necesariamente caer en la obsolescencia y en la ineficacia, aunque resulta evidente que conforme un proveedor incrementa la variedad de sus productos, de sus ventas y de sus clientes requerirá de inyectar mayores recursos que le permitan sostener su esquema o modificarlo de tal forma que no frene su crecimiento.

## **2.2. DESCRIPCIÓN DE OBJETIVOS**

Dado el incremento en la diversidad de productos que manejan las empresas farmacéuticas, surge la necesidad de crear un sistema que cuente con una adecuada clasificación de estos productos así como una integración con los servicios que se ofrecen para mantenerse en los niveles de competencia que el mercado presenta y responder eficientemente a las necesidades de los clientes.

Hemos propuesto realizar un sistema que tome en cuenta las modificaciones que se desean al esquema actual del control de productos, ya que no se tiene un control adecuado sobre ellos. Se realizó además un análisis sobre el tipo de información que se va a manejar y se consideraron también las necesidades y requerimientos de las empresas farmacéuticas.

### **2.2.1. Objetivos a Cubrir del Sistema**

- Controlar los movimientos de productos y servicios de la empresa
- Mantener un inventario actualizado de productos farmacéuticos
- Mejorar la comunicación cliente-empresa acortando y simplificando los procesos entre ellos
- Establecer al instante la situación de la empresa
- Permitir el desarrollo y actualización posterior o migración a otros sistemas

Para ello hemos propuesto un sistema que permita el registro de la información que maneja la empresa sobre una Base de Datos (BD).

Este sistema contará con una interfaz gráfica agradable para el usuario compuesta por iconos y ventanas vinculados entre sí de manera lógica para facilitar su uso. La pantalla inicial contemplará de manera general cualquier operación que el usuario desee realizar en el sistema. Las operaciones básicas a las que el usuario podrá

acceder en esta pantalla inicial serán:

- Importación de archivos.
- Exportación de archivos.
- Catálogos del sistema.
- Impresión de reportes.
- Ayuda general del sistema.
- Salida del sistema.

Se establecerán además formatos para la captura de información que agilicen la interacción con el cliente, ya que sólo se requerirá capturar la información básica. Con esto se logra que la información capturada en la BD no tenga redundancia y mantenga consistencia. Para la salida de información también se contemplan formatos preestablecidos que presenten claramente la información solicitada.

Se contará con un sistema de seguridad mediante el cual cada usuario que desee ingresar al sistema deberá identificarse debidamente. Se establecerán diferentes niveles de acceso de acuerdo al tipo de usuario que quiera ingresar, validando sólo las operaciones permitidas y su disponibilidad a la información.

La información del inventario de la empresa permitirá el autoabastecimiento de productos farmacéuticos y facilitará la visualización sobre la evolución de productos en el mercado. Se podrán efectuar reportes que permitan establecer al instante la situación que guarda la empresa y realizar los análisis que se consideren pertinentes para evaluar la conducción de la empresa. Se tiene contemplado también graficar los productos almacenados.

El sistema va a estar basado en programas del dominio público con el propósito de estandarizar su uso y facilitar su desarrollo posterior o migración a otros sistemas nuevos en caso de ser necesario. Utilizaremos el Sistema Operativo: Microsoft

Windows 95, con un Front-End gráfico desarrollado en Microsoft Visual Basic versión 4.0 para desarrollo en Windows de 16 bits y un Back-End para la BD, desarrollado en Microsoft Access 7.0. Estos paquetes son fácilmente adquiribles, se tienen instalados en la mayoría de computadoras personales, se puede obtener información sobre ellos, sus actualizaciones y además, software asociado en cualquier parte (internet, escuelas, librerías, etc.). La empresa posee las licencias de estos productos y es un requisito a cumplir en el sistema por parte de la empresa. El sistema tratará de conformarse gráficamente de tal manera que al usuario se le presente como una aplicación más de Windows 95.

Se contará con ayuda en línea sobre el uso del sistema, sobre la manera de capturar la información y sobre la manera de llevar a cabo los procesos que se deseen. Además se proporcionará información escrita sobre su uso.

Todo el código que atenderá los sucesos elegidos por el usuario estará escrito en Visual Basic 4.0, de manera que se pueda entender fácilmente al momento de integrar cada rutina que atiende a cada suceso. Los formatos de los reportes se harán en Crystal Report para Visual Basic, Versión 3.0.1.32.

La BD se implementará en Access 7.0 que esta basada en **Structured Query Language SQL** (Lenguaje Estructurado para Consultas), un lenguaje bastante compatible con diversas plataformas. Se proporcionará el diccionario de datos que permita al administrador del sistema conocer el sistema y poder desarrollar las adecuaciones que considere pertinentes así como su migración a otros programas o su implementación en internet.

El empleo de sistemas computarizados en la administración de recursos es la tendencia a seguir en la modernización de sistemas aprovechando sus ventajas y ahorros que representan para las empresas que deciden utilizarlos.

## 2.3. CONCEPTOS DE BASES DE DATOS RELACIONALES

### 2.3.1. Introducción

En años recientes, los sistemas administradores de bases de datos **DBMS** (Database Management System) han establecido como principios fundamentales el almacenamiento de datos para sistemas de información con alto flujo transaccional para aplicaciones comerciales de escritorio o ambiente PC. Actualmente, los sistemas de información requieren que su información no sólo se encuentre almacenada consistentemente, sino también que se tenga relacionada de una forma lógica, es de aquí la razón del surgimiento de los sistemas administradores de BD relacionales **RDBMS** (Relational Database Management System), proporcionando almacenamientos y recuperaciones de datos más sofisticados, con funciones de distribución empresarial o corporativa de datos para los sistemas de manejo de información. Comparando un sistema de archivos con un manejador de BD relacionales, este último proporciona organización de datos con una capacidad de integrar y manipular volúmenes masivos de datos operacionales dentro de un robusto sistema de información, dando inicio con esto a la evolución de las bases de datos con el inicio tecnológico del concepto cliente/servidor, **data warehousing** (Repositorio global e institucional de datos) y procesos analíticos en línea.

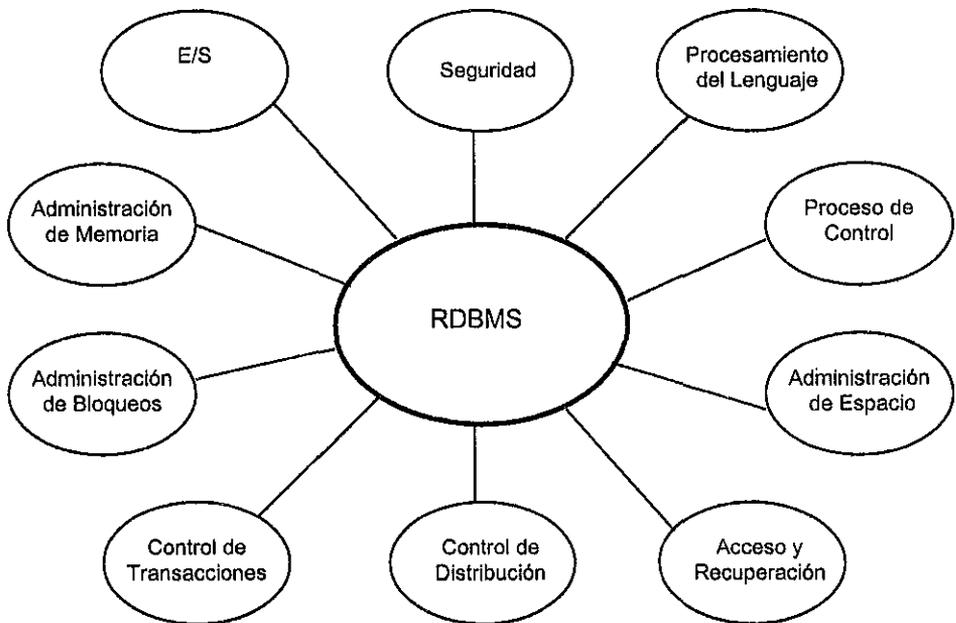
### 2.3.2. Componentes de un RDBMS

Un RDBMS tiene básicamente dos componentes importantes en su arquitectura, el **Kernel** (núcleo) el cual es el software de control y administración y el Diccionario de Datos el cual contiene el nivel de estructuras de datos usadas por el Kernel para manejo de la BD. Trataremos brevemente cada uno de estos componentes.

## Kernel

Virtualmente un RDBMS es un sistema operativo o un conjunto de subsistemas, diseñados específicamente para el control de los accesos de datos, sus funciones primarias son almacenar, entregar y asegurar los datos del usuario. Como un sistema operativo, el RDBMS maneja y controla los accesos para proporcionar un conjunto de recursos para las bases de datos de los usuarios concurrentes.

Un RDBMS mantiene su propia lista de usuarios autorizados y sus privilegios asociados, manejo de sus memorias y páginas, control de bloqueos (locking) para recursos utilizados concurrentemente, atención a las solicitudes de información de los usuarios y el manejo del espacio disponible para la BD. En la figura 2.3.2.1 se muestra los principales subsistemas del Kernel de un RDBMS para el manejo de BD.



**Figura 2.3.2.1. Un RDBMS y sus múltiples subsistemas.**

## **Diccionario de Datos**

Una diferencia fundamental entre un RDBMS y otra BD o sistema de archivos es en la forma de acceder los datos. Un RDBMS permite que se haga referencia física de los datos en una forma lógica, proporcionando flexibilidad en el desarrollo del código de las aplicaciones. Los programas que utilizan accesos a los datos de un RDBMS lo hacen a través de un software residente que proporciona el RDBMS para la atención a los requerimientos del usuario, este software se le conoce como Database Engine (Ingeniería de Bases de Datos), creando independencia entre la estructura actual de los datos y la definición lógica de los mismos. Esta independencia es posible, gracias al diccionario de datos del RDBMS.

### **2.3.3. Objetivos de los Sistemas de Bases de Datos**

Actualmente una de las actividades más importantes en la mayoría de las organizaciones, es la administración de datos. En la medida en que nos movemos hacia una sociedad más orientada a la información, un problema muy importante es cómo determinar la organización de los datos para maximizar su utilidad.

Sin embargo, mucha gente dedicada al procesamiento de datos no ha entendido completamente el problema de cómo organizar los datos para utilizar a su máxima capacidad los sistemas de bases de datos. El objetivo de estas notas es ofrecer una metodología que haga más fácil entender y seguir el proceso de organización de los datos.

Un RDBMS se compone de una serie de datos relacionados entre sí y de un conjunto de programas para tener acceso a esos datos. Los datos contienen información de determinada empresa.

El objetivo principal de un RDBMS es crear un ambiente en el que pueda almacenarse y recuperarse información en la BD en forma conveniente y eficiente. Estos sistemas

permiten evitar las principales desventajas de los sistemas de procesamiento de archivos tradicionales.

- Redundancia e inconsistencia de los datos.
- Dificultad para acceder a los datos.
- Aislamiento de los datos.
- Problemas de concurrencia.
- Problemas de seguridad.
- Problemas de integridad.

### **Abstracción de la Información**

Un RDBMS permite proporcionar a los usuarios una visión abstracta de la información. Es decir, el sistema oculta ciertos detalles relativos a la forma de cómo los datos se almacenan y mantienen. Tradicionalmente se definen tres niveles de abstracción.

- Niveles físico o esquema interno: Se describe como se almacenan realmente los datos.
- Nivel o esquema conceptual: Describe cuales son los datos reales que están almacenados y las relaciones existentes entre ellos.
- Nivel de vista o esquema del usuario: Es el nivel de abstracción más alto, en el cual se describe solamente una parte de los datos. El sistema puede proporcionar muchas vistas diferentes de la misma BD.

### **Modelos de Datos**

Para definir la estructura de una BD es necesario definir el concepto de modelo de datos. Primero, un modelo es una representación de objetos y eventos del mundo real. Un modelo de datos es una representación abstracta (una descripción) de datos por medio de sus entidades, actividades y sus asociados dentro de una organización.

### **Modelos Lógicos basado en Objetos**

Se utilizan para describir los datos en los niveles conceptual y de vista. Son flexibles y permiten especificar claramente las limitaciones de los datos. Algunos ejemplos son:

- ⇒ El modelo Entidad-Relación (ER)
- ⇒ El modelo semántico de datos

El modelo Entidad-Relación es el representativo de las clases de los modelos lógicos basados en objetos. Ha tenido bastante aceptación como modelo de datos apropiado para el diseño de bases de datos y se utilizan ampliamente en la práctica.

### **Modelos Lógicos basados en Registros**

Se utilizan para describir los datos en los niveles conceptuales y de vista. Sirve para especificar la estructura lógica general de la BD. Sin embargo, no permite especificar claramente las limitantes de los datos. Algunos son:

- Modelo relacional.
- Modelo de red.
- Modelo jerárquico.

### **La metodología ER y el Ciclo de Vida del Desarrollo de Sistemas**

La metodología ER, por todas sus ventajas y características, es una herramienta para que los analistas y diseñadores de sistemas puedan desarrollar un modelo lógico de la BD de su organización. Sin embargo, este diseño debe ser parte de toda una planeación.

### **2.3.4. Importancia de la Planeación de las Bases de Datos**

Tradicionalmente los sistemas de información nunca se planean o diseñan realmente, sino que evolucionan de una manera de "arriba hacia abajo" como sistemas

independientes de sistemas de información tradicionales que preguntan ¿qué procedimiento (programa de aplicación) se requiere para resolver ese problema particular como existe hoy? El problema con este enfoque es que los procedimientos requeridos seguramente cambiarán de acuerdo a los cambios en el ambiente de la organización.

Por otro lado, los administradores de los datos como recurso esencialmente preguntan: ¿qué requerimientos de bases cumplirán las necesidades de información de las empresas hoy y en el futuro? Una ventaja de este enfoque es que es menos probable que cambien los datos de una organización que sus procedimientos.

Por lo tanto, el reto de los administradores de datos como recursos, es diseñar bases de datos estables que sean relativamente independientes de las aplicaciones que se construyan sobre ella.

Para obtener el mayor beneficio de este enfoque la organización debe analizar su información y planear su BD cuidadosamente. Si se trata de tomar este enfoque sin esta planeación, los resultados muy bien podrían ser desastrosos.

Si se acepta que los datos son un recurso valioso para la organización, entonces se requiere una planeación completamente diferente. La organización debe desarrollar planes estratégicos para sus recursos de datos, tal y como lo hace para sus recursos humanos, financieros y materiales. El proceso de planeación debe ser de "arriba hacia abajo" para que los sistemas de información y la planeación de la BD se integren como los objetivos básicos de la empresa.

### **Proceso de Planeación de una Base de Datos**

El propósito de la planeación de la BD es desarrollar una estrategia o plan de largo plazo, para un ambiente de BD que soporte las necesidades de información, actuales y futuras de la organización. El plan de BD debe ser un subconjunto del plan de sistemas

de información de la organización, el cual es un subconjunto de todo el plan corporativo.

Esto visualiza al proceso de planeación como un proceso jerárquico, o de arriba hacia abajo, en naturaleza. Un enfoque alternativo es de abajo hacia arriba, en el que quien planea extrapola de los sistemas existentes y de las necesidades de información. Sin embargo, con este enfoque, no hay seguridad de que el plan de bases de datos concuerde con los planes de más alto nivel.

La planeación de la BD se debe establecer como un proyecto formal dentro de la organización. Normalmente se debería realizar una revisión del plan cada cinco años. La información anual de la planeación de los sistemas podría cambiar la dirección del plan con la posibilidad de hacer únicamente cambios menores.

### **Tareas en la Planeación de una BD**

Las principales tareas en la planeación de un ambiente de BD se mencionan enseguida. La planeación de una BD no es un proceso estrictamente secuencial. Casi siempre es necesario el refinamiento dentro de ciertos pasos en la medida que una organización aprende más acerca de sus requerimientos de datos.

**Tarea 0.** Obtener el compromiso de la dirección. Quien debe iniciar el proceso de planeación es la dirección. Aunque no lo haga, el equipo de estudio debe obtener la firma de compromiso de la alta administración. Existen varias razones para que este compromiso sea importante:

- Da la autoridad para comenzar el proceso de planeación.
- Asegura que habrá la disponibilidad de recursos y cooperación necesarios en el desarrollo del plan.
- Envía señales al resto de la organización de que el proceso de planeación es importante y que se le debe dar alta prioridad.

- Ayuda a asegurar que el plan final se realice.

Sin la firma del compromiso de la dirección, seguramente el proceso de planeación se pospondrá.

**Tarea 1. Inicio del estudio.** La primera tarea es definir las metas y objetivos del enfoque de bases de datos de la organización. Esto requiere establecer el alcance de la BD en términos de qué áreas del negocio o funciones, se atacarán. El equipo del proyecto debe identificar los beneficios que se consiguen con un ambiente de BD. Así los beneficios pueden establecer a manera de metas y objetivos que concuerden con los planes de largo plazo de la organización.

**Tarea 2. Establecer las funciones de la Administración de la BD.** Se debe definir las funciones del administrador de la BD desde un principio. La persona que sea, al menos candidatos para este puesto, debe tomar el papel de líder para los pasos siguientes de planeación. Se deberá adquirir herramientas de diseño y otras del tipo Computer Aided Systems Engineering CASE (Ingeniería de Sistemas Asistida por Computadora).

**Tarea 3 a 6. Realización del análisis de los sistemas del negocio.** El equipo de estudio identifica y documenta las funciones: procesos, actividades y entidades del negocio.

**Tarea 3.** Definición de funciones del negocio.

**Tarea 4.** Definición de procesos del negocio.

**Tarea 5.** Definición de actividades del negocio.

**Tarea 6.** Definición de las entidades del negocio.

**Tarea 7. Construcción de un modelo de información.** Este es el paso central de toda la estrategia del proceso de planeación. Este modelo esquematiza los principales

tipos de entidades de la organización y las asociaciones entre ellas. Es aquí en donde entra la metodología Entidad-Relación.

**Tarea 8. Desarrollo de un plan de distribución.** Si los datos estarán distribuidos en varias bases de datos físicas, se debe desarrollar un plan para distribución de datos.

**Tarea 9. Desarrollo del plan de realización.** Se establece una tabla de tiempos y responsabilidades para la realización de la BD. Se establecen prioridades para que el plan de bases de datos esté de acuerdo con todos los planes de los sistemas de información.

**Tarea 10. Revisión y evaluación final.** Se prepara un reporte final y se revisan todos los componentes, para asegurar que son consistentes. Se presenta el reporte a la alta gerencia y se establece con las responsabilidades para la actualización del plan.

El diseño detallado de la BD es un proceso de abajo hacia arriba. Se comienza con las vistas de los usuarios y se aplican las técnicas de normalización para desarrollar modelos de datos detallados y se verifican contra el modelo de información para asegurar que son completos y exactos.

### **2.3.5. Normalización**

El concepto principal, tomado del modelo relacional utilizado en el desarrollo del modelo conceptual, es el proceso de normalización, esto es, el proceso de agrupar a los campos de datos en tablas que representan a las entidades y sus relaciones. La teoría de la normalización está basada en la observación de que un cierto conjunto de relaciones tiene mejores propiedades en un medio de inserción, actualización y supresión, que las que tendrían otros conjuntos de relaciones conteniendo los mismos datos.

La razón de usar el procedimiento de normalización es asegurar que el modelo conceptual de la BD funcionará. Esto no significa que una estructura no normalizada no funcionará, sino que puede causar algunos problemas cuando los programadores de aplicación traten de modificar la BD. El administrador de la BD debe decidir, después de localizar las violaciones provenientes de la normalización, si las modificaciones afectarán la forma en que la BD funcionará.

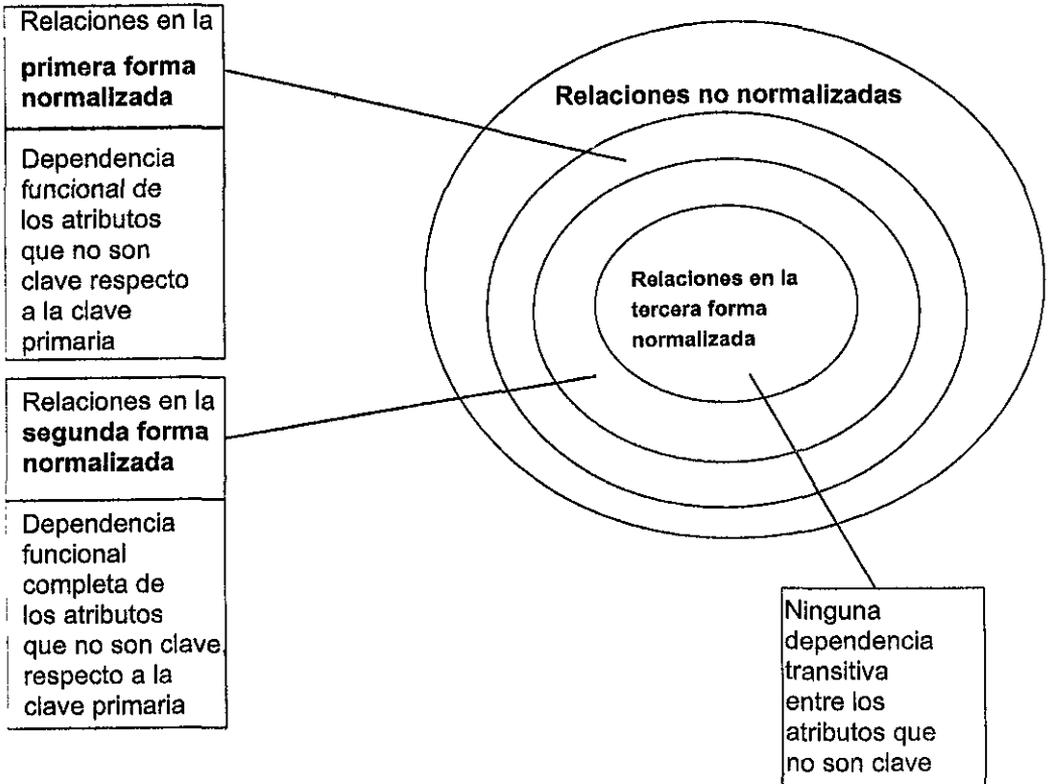
Un modelo de datos no normalizado consiste en registros utilizados por los programas de aplicación. El primer paso de la normalización consiste en transformar los campos de datos a una tabla de dos dimensiones. Lo que se requiere usualmente en este paso es la eliminación de ocurrencias repetidas de campos de datos, de tal manera que se obtenga un archivo fijo. Por ejemplo, si una declaración incluye espacio para el nombre de un empleado, su número, esposa y hasta diez hijos, el resultado será una tabla de 4 por 10, con cuatro columnas y diez renglones. Cada uno de los renglones tendrá el nombre y número del empleado, esposa y el nombre de uno de los hijos. Los diez renglones tendrán los nombres de los diez hijos. Este es sólo un paso preliminar que hace posible trasladarse a la segunda forma normalizada.

El segundo paso de la normalización es establecer las claves y relacionarlas con los campos de datos. En la primera forma normalizada, el renglón entero de la tabla (cadena) depende de todos los campos de claves. En la segunda forma normalizada, se hace un intento de establecer los campos de datos que están relacionados con alguna parte de la clave completa. Si los campos de datos sólo dependen de una parte de la clave, ésta y los campos conectados a la clave parcial son susceptibles de separarse en registros independientes. La división de la primera tabla normalizada, es una serie de tablas en las que cada campo sólo depende de la clave completa, se llama la segunda forma normalizada.

El tercer paso consiste en separar los campos de las segundas relaciones normales que, aunque dependan sólo de una clave, deben tener una existencia independiente

en la BD. Esto se hace de forma tal que la información sobre estos campos pueda introducirse separadamente a partir de las relaciones en las que se encuentra implicada.

En cada modelo de datos uno o más campos de datos se agrupan para representar entidades y sus relaciones. En los agrupamientos de los campos de datos pueden darse tres tipos generales de problemas, y la eliminación de cada uno de éstos da pie a las tres formas normalizadas de relaciones (tablas). Por lo tanto el proceso de normalización es una disciplina que consiste en agrupar a los campos de datos en un conjunto de relaciones (tablas). La figura 2.3.5.1 muestra las tres formas normalizadas.



**Figura 2.3.5.1.** Toda relación que está en la primera forma normalizada es un caso especial de una relación no normalizada. Pero no toda relación no normalizada está en la primera forma. Toda relación que está en la segunda forma normalizada es un caso especial de relación en la primera forma, pero no viceversa. Toda relación que está en la tercera forma es un caso especial de relación en la segunda forma normalizada, pero no toda relación en la segunda forma normalizada está necesariamente en la tercera forma.

Todas las relaciones normalizadas están en la primera forma normal, algunas relaciones de la primera forma normalizada están en la segunda forma y, finalmente, algunas relaciones de la segunda forma normalizada están en la tercera. El proceso de

normalización produce las tres formas de relaciones. Las relaciones de la tercera forma representan a las entidades y sus relaciones dentro de una empresa.

La primera, la segunda y la tercera formas normalizadas proporcionan mejoras sucesivas en las operaciones de inserción, supresión y actualización de la BD. Como veremos, el proceso de normalización hace que el diseñador entienda la semántica de los atributos y sus relaciones y como resultado influye en el proceso de consideraciones al hacer el análisis de los datos.

En los párrafos precedentes hemos visto someramente los pasos comprendidos al tomar una estructura no normalizada y convertirla hasta una tercera forma. El resto de esta sección muestra cómo hacer esto. Considere el ejemplo de la figura 2.3.5.2, los campos de datos o atributos son: número, nombre y dirección del paciente, número de licencia y nombre del cirujano, fecha de la cirugía y cirugía, medicina administrada después de la operación y su efecto secundario.

| Número del Paciente | Número de Licencia del cirujano | Fecha de la cirugía | Nombre del paciente | Dirección del paciente         | Nombre del cirujano | Cirugía                          | Medicina administrada después de la operación | Efecto secundario de la medicina |
|---------------------|---------------------------------|---------------------|---------------------|--------------------------------|---------------------|----------------------------------|---|----------------------------------|
| 1111                | 145                             | Enero 1, 1997       | John Write          | 115 New Street, New York, N.Y. | Beth Little         | Extirpación de cálculos biliares | Penicilina                                    | Prurito                          |
| 1111                | 311                             | Junio 12, 1997      | John Write          | 115 New Street, New York, N.Y. | Michael Diamond     | Extirpación de cálculos renales  |   |                                  |
| 1234                | 243                             | Abril 5, 1976       | Mary Jones          | 10 Main Street Rye, NY.        | Charles Field       | Extirpación de catarata          | Tetraciclina                                  | Fiebre                           |
| 1234                | 467                             | Mayo 10, 1977       | Mary Jones          | 10 Main Street Rye, NY.        | Patricia Gold       | Eliminación de trombosis         |   |                                  |

Clave Primaria

Atributo

Cadena o renglón

**Figura 2.3.5.2. Representación de datos utilizando un modelo relacional: "medio pseudohospitalario".**

Una columna o conjunto de columnas se llama "clave permitida" cuando sus valores identifican de manera única el renglón de la relación. Dado el número del paciente (1234), el número de licencia del cirujano (243) y la fecha de la cirugía (abril 5, 76), queda determinado de manera única el renglón (cadena) "1234243 abril 5, 1976 Mary Jones 10 Main Street, Rye, N. Y. Charles Field, Extirpación de catarata Tetraciclina Fiebre". Ya que ésta es la única clave permitida se convierte en una clave primaria (se considera que toda relación tiene sólo una clave permitida. Así, la única clave permitida es la clave primaria, y todo atributo que no forma parte de ésta es un atributo que no es clave).

Primera forma normalizada. En la primera forma, una relación es una "tabla". En cada intersección de un renglón (cadena) y en una columna sólo puede haber un valor en la tabla. Ningún grupo de valores está permitido en las intersecciones. Considere la figura 2.3.5.3. Esta es una relación "no normalizada". En los cruces de algunos renglones y columnas se llega a presentar más de un valor de la clave primaria, los valores de los atributos que no son clave no se pueden determinar de manera única.

| Número del Paciente | Número de Licencia del cirujano | Fecha de la cirugía | Nombre del paciente | Dirección del paciente   | Nombre del cirujano | Cirugía                          | Medicina administrada después de la operación | Efecto secundario de la medicina |
|---------------------|---------------------------------|---------------------|---------------------|--------------------------|---------------------|----------------------------------|---|----------------------------------|
| 1111                | 145                             | Enero 1, 1997       | John Write          | 15 New St., New York, NY | Beth Little         | Extirpación de cálculos biliares | Penicilina                                    | Prurito                          |
| 1111                | 311                             | Junio 12, 1977      |                     |                          | Michael Diamond     | Extirpación De Cálculos Renales  | -----   | -----                            |
| 1234                | 243                             | Abril 5, 1976       | Mary Jones          | 10 Main Str. Rye, NY.    | Charles Field       | Extirpación De Catarata          | Tetraciclina                                  | Fiebre                           |
| 1234                | 467                             | Mayo 10, 1977       |                     |                          | Patricia Gold       | Eliminación De Trombosis         | -----   | -----                            |

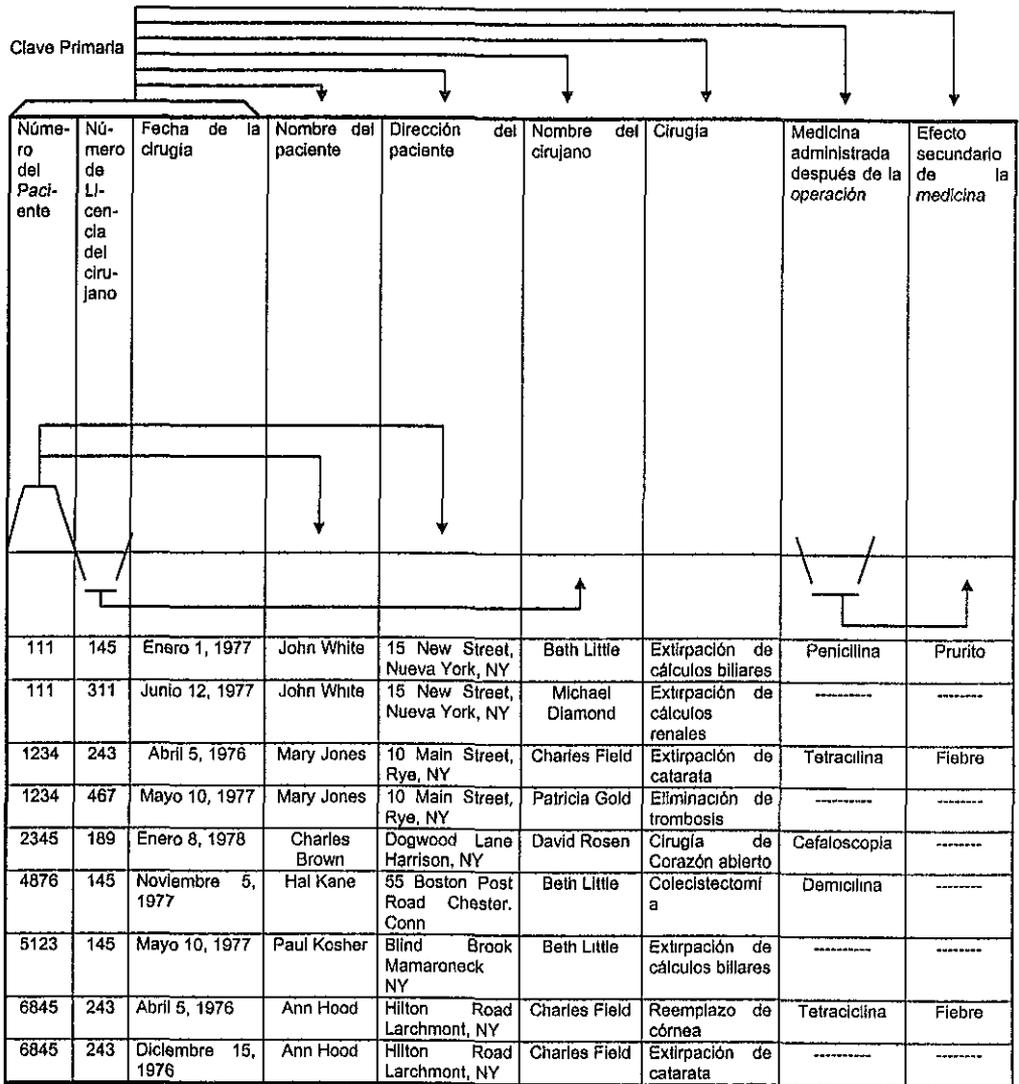
**Fig. 2.3.5.3. Esta es una relación "no normalizada". En el cruce de algunos renglones y columnas se presentan más de un valor.**

Transferimos la relación no normalizada de la figura 2.3.5.3 a una relación normalizada en la figura 2.3.5.4. En esta conversión repetimos los valores tomados por el número, nombre y dirección del paciente y, en su caso, los del número de licencia y nombre del cirujano. La figura 2.3.5.4 representa una relación en la primera forma normalizada. En cada intersección de un renglón (cadena) y una columna, solo hay un valor en la tabla, no grupos de valores. La figura 2.3.5.5 muestra la relación de una manera gráfica.

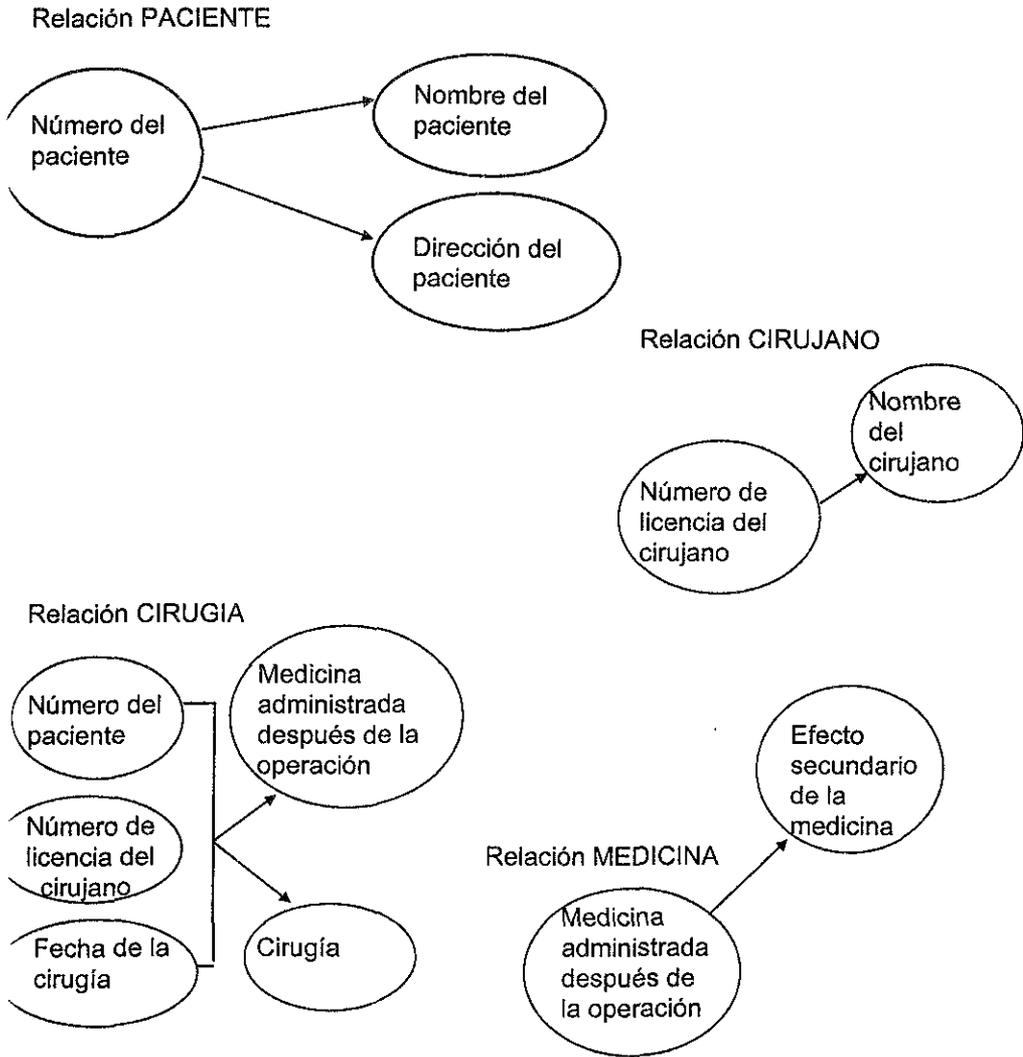
Cuando los valores tomados por el número del paciente, el número de licencia del cirujano y la fecha de cirugía se conocen, también se conocen los valores tomados por el nombre del paciente, la dirección del paciente, el nombre del cirujano, la cirugía, la medicina administrada después de la operación y el efecto secundario de la medicina. Así, la clave primaria está compuesta del número del paciente, el número de licencia del cirujano y la fecha de cirugía. No hay otra clave permitida para esta relación (tabla). Todos los atributos que no son clave en la relación son funcionalmente dependientes de la clave primaria, quedan determinados de manera única los valores tomados por los atributos que no son clave.

Fallas de almacenamiento de la primera forma normalizada. Las siguientes fallas sobrevendrán si dejamos nuestros datos en la primera forma normalizada (véase la figura 2.3.5.4).

Falla de inserción. Es posible que a un nuevo paciente no se le haya practicado una intervención quirúrgica en el hospital y que por lo tanto no se le haya asignado ningún cirujano. Como resultado de no habersele asignado ningún cirujano, la cadena del paciente no se puede introducir, esto es, la información sobre el paciente no se puede almacenar en la primera forma de la figura 2.3.5.4.



**Figura 2.3.5.4. Representación de datos utilizando un modelo relacional. Relación en la primera forma normalizada.**



**Figura 2.3.5.5. Una relación en la primera forma normalizada.**

Supongamos que deseáramos insertar los datos del paciente: nombre y dirección. No se debería conocer necesariamente el número de licencia del cirujano ni la fecha de la cirugía. Esto significa que, para identificar de manera única el nombre y la dirección del paciente son superfluos dos componentes de la clave primaria, el número de licencia

del cirujano y la fecha de cirugía. Si separamos el nombre del paciente y su dirección, junto con el número del paciente, en una relación (tabla) distinta, como en la figura 2.3.5.6a, esta falla de inserción se podrá rectificar.

Falla de inserción. Es posible que el hospital haya nombrado a un nuevo cirujano que aún no haya operado a ningún paciente de este hospital. Como consecuencia, en el renglón para este cirujano no habrá ningún valor asignado al número del paciente o a la fecha de la cirugía. Sólo una parte de la clave primaria está constituida del número de licencia del cirujano; el resto consiste en el número del paciente y en la fecha de la cirugía. Como consecuencia, no se puede introducir una nueva cadena para un nuevo cirujano, es decir, no se puede almacenar la información del cirujano. En este caso, el atributo que no es clave correspondiente al nombre del cirujano, está identificado de manera única con la clave primaria formada por el número del paciente + número de licencia del cirujano + fecha de la cirugía. Pero la identificación única del nombre del cirujano, la única parte necesaria de la clave primaria es el número de licencia del cirujano. Si separamos el nombre y el número de licencia del cirujano, como en la figura 2.3.5.6b, la falla en la inserción se rectifica.

Estas fallas ocurren debido a que algunas columnas sólo dependen una parte de la clave primaria, mientras que la clave primaria completa es necesaria para identificar de manera única a un renglón completo.

Falla de actualización. Si John White ingresa en el hospital por tercera vez, y si entre la segunda y tercera estancia se cambió de casa, nos gustaría actualizar su dirección en todos los renglones en los que aparece. (Queremos guardar sólo la información actualizada sobre los nombres y direcciones, con el fin de tener consistencia. Suponemos que el hospital no quiere guardar las antiguas direcciones). Este ejemplo muestra la dificultad de actualizar una relación en la primera forma normalizada, debido a que las cadenas en las que se debe reflejar el cambio, varían con el tiempo. Lo peor

que podría suceder sería tener algunas cadenas con la dirección antigua y algunas con la nueva.

Esta falla de actualización se puede resolver si la dirección del paciente sólo aparece una vez. Si separamos el nombre del paciente, su dirección y su número, como en la figura 2.3.5.6a la falla desaparece. La dirección de John White sólo aparecerá una vez en la tabla de PACIENTE, como en la figura 2.3.5.6a.

Falla la supresión (tipo 1). Supongamos que queremos borrar la información de un paciente después de que él o ella hayan muerto. Consideramos que el paciente Charles Brown fallece (véase la figura 2.3.5.4). Cuando suprimimos la cadena correspondiente a Charles Brown, toda la información relativa al cirujano David Rosen se pierde debido a que esta operación fue la única que él realizó. En algunas aplicaciones, la pérdida de información de este tipo podría tener efectos serios. Puesto que el renglón suprimido podría ser la única fuente de información sobre David Rosen. Para prevenir que algo así ocurra tendríamos que responsabilizar al usuario de la verificación de que la cadena por ser borrada no sea la única fuente de información de esta "categoría".

Tabla PACIENTE

| Número del paciente | Nombre del paciente | Dirección del Paciente               |
|---------------------|---------------------|--------------------------------------|
| 1111                | John White          | 15 New street, Nueva York, N.Y.      |
| 1234                | Mary Jones          | 10 Main Street, Rye, N.Y.            |
| 2345                | Charles Brown       | Dogwood Lane, Harrison, N.Y.         |
| 48776               | Hal Kane            | 55 Boston poston Road, Chester Conn. |
| 5123                | Paul Kosher         | Blind Brook, Mamaroneck, N.Y.        |
| 6845                | Ann Hood            | Hilton Road, Larchmont, N.Y.         |



**Figura 2.3.5.6a. Relación en la segunda forma normalizada. La clave primaria es el número del paciente. Los atributos que no son clave, nombre y dirección del paciente necesitan de la clave primaria completa para su identificación única. El nombre y la dirección del paciente sólo aparecen en esta relación.**

Tabla CIRUJANO



| Número de licencia del cirujano | Nombre del cirujano |
|---------------------------------|---------------------|
| 145                             | Beth Little         |
| 189                             | David Rosen         |
| 243                             | Charles Field       |
| 311                             | Michael Diamond     |
| 467                             | Patricia Gold       |

**Figura 2.3.5.6b. Relación en la segunda forma normalizada. La clave primaria es el número de licencia del cirujano. El atributo que no es clave, nombre del cirujano, necesita de la clave primaria para su identificación única. El nombre del cirujano sólo aparece en esta relación.**

Una posibilidad de resolver un problema de esta naturaleza es separar la información del cirujano que no dependa del paciente y viceversa. Esto se puede lograr teniendo dos relaciones (tablas), como en las figuras 2.3.5.6a y 2.3.5.6b, para PACIENTE y CIRUJANO, respectivamente.

Falla de supresión (tipo 2). Otra falla de supresión resulta de un atributo que no es clave, de la relación de la figura 2.3.5.4, que es funcionalmente dependiente de otro atributo que tampoco sea clave, de la misma relación. El efecto secundario de una medicina es funcionalmente dependiente sólo de la medicina administrada. Es posible que el prurito sea resultante de administrarle otra medicina. Los valores de atributo para esa cadena específica, cambian a otra medicina y a otro efecto secundario. Como consecuencia de este cambio, perdemos la información de que la penicilina produjo prurito. Puesto que esta cadena particular podría constituir la única fuente de información en la BD para esa categoría, esta pérdida no es deseable.

Para resolver estas fallas, separaremos la información sobre el paciente en una relación llamada PACIENTE, como en la figura 2.3.5.6a, y la información del cirujano en una relación llamada CIRUJANO, como en la figura 2.3.5.6b. La clave primaria en la relación PACIENTE de la figura 2.3.5.6a es el número del paciente, y la clave primaria en la relación CIRUJANO de la figura 2.3.5.6b es el número de licencia del cirujano. Los atributos restantes comprenden la relación PACIENTE-Y-CIRUJANO mostrada en la figura 2.3.5.6c, donde la clave primaria está compuesta del número del paciente, el número de licencia del cirujano y la fecha de la cirugía.

Tabla PACIENTE y CIRUJANO

| Número del Paciente | Número de licencia del cirujano | Fecha de la cirugía | Cirugía                          | Medicina administrada después de la operación | Efecto secundario de la medicina |
|---------------------|---------------------------------|---------------------|----------------------------------|---|----------------------------------|
| 1111                | 145                             | Enero 1, 1977       | Extirpación de cálculos biliares | Penicilina                                    | Prurito                          |
| 1111                | 311                             | Junio 12, 1977      | Extirpación de cálculos renales  | -----   | -----                            |
| 1234                | 243                             | Abril 5, 1976       | Extirpación de catarata          | Tetraciclina                                  | Fiebre                           |
| 1234                | 467                             | Mayo 10, 1977       | Eliminación de trombosis         | -----   | -----                            |

**Figura 2.3.5.6c. Relación en la segunda normalizada. La clave primaria es número de paciente + número de licencia del cirujano + fecha de la cirugía. Todo atributo que no es clave (por ejemplo, cirugía, medicina administrada después de la operación y efecto secundario de la medicina) requieren de la clave primaria para su identificación única.**

Se dice que las relaciones de las figuras 2.3.5.6a, 2.3.5.6b y 2.3.5.6c están en la segunda forma normalizada. Estas tres relaciones para nuestros datos son más deseables que la relación en la primera forma de la figura 2.3.5.4, debido a que una relación en la segunda forma normalizada elimina algunas de las fallas de almacenamiento encontradas en la relación que está en la primera forma. La figura 2.3.5.7 representa las tres relaciones de manera gráfica en la segunda forma normalizada.

Relación de la segunda forma normalizada. Se dice que una relación está en la segunda forma normalizada, cuando todo atributo que no sea clave es completamente dependiente de manera funcional de la clave primaria, es decir, todo atributo que no es clave necesita de la clave primaria completa para poder ser identificado de manera única, inversamente, una relación no está en la segunda forma si existe un atributo que no sea clave que no dependa completamente y de manera funcional de la clave primaria.

Toda relación en la segunda forma normalizada también presenta una relación en la primera forma.

Fallas de almacenamiento de la relación en la segunda forma normalizada. Las figuras 2.3.5.6a, 2.3.5.6b y 2.3.5.6c representan tres relaciones en la segunda forma. Algunas de las fallas que se tenían en la primera forma normalizada, como en la figura 2.3.5.4, se han eliminado en las relaciones que están en la segunda forma.

Inserción. Podemos introducir un nuevo paciente que todavía no ha sido operado en el hospital y al cual no se le ha asignado ningún cirujano, simplemente introduciendo la información del paciente en la relación de la figura 2.3.5.6a.

Inserción. La información referente a un nuevo cirujano que aún no ha operado a ningún paciente en el hospital, se puede introducir añadiendo simplemente una nueva cadena en la relación de la figura 2.3.5.6b.

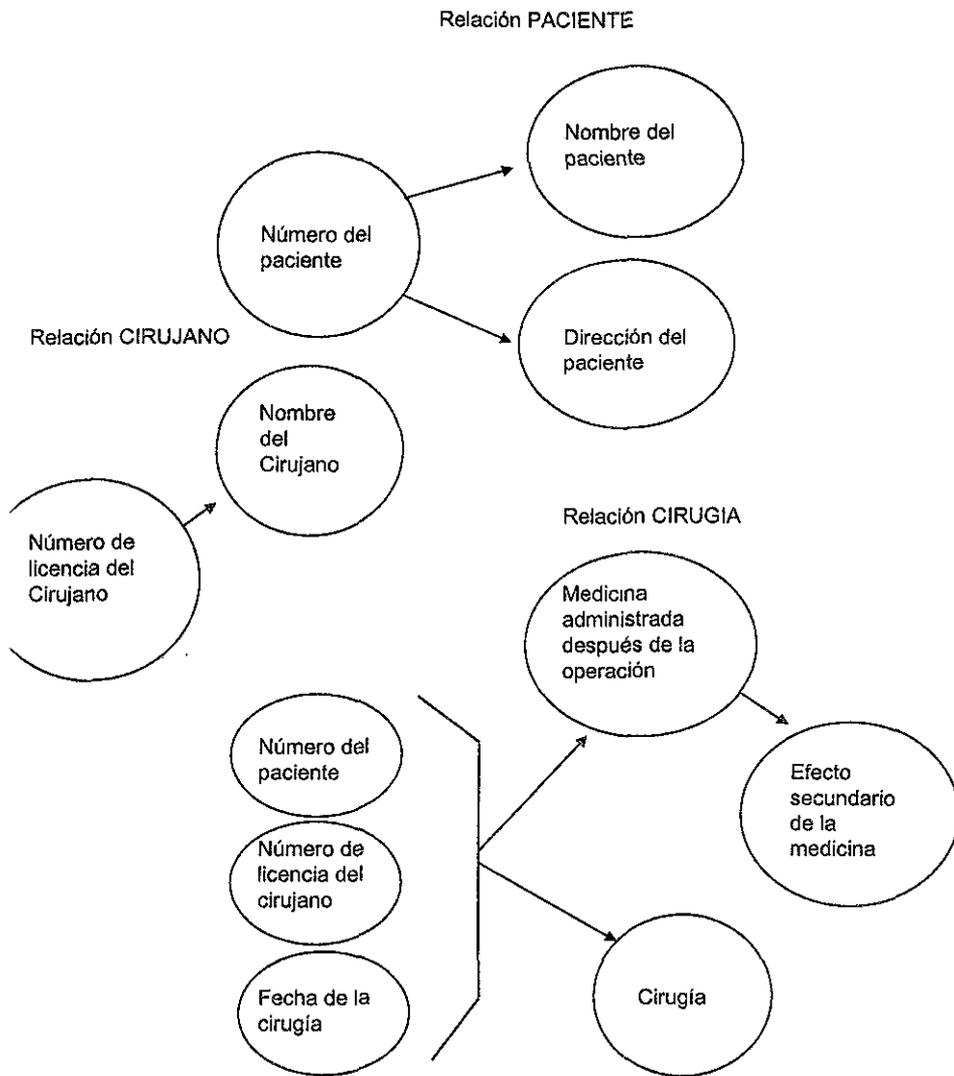
Supresión (tipo 1). Charles Brown fallece, la cadena correspondiente en las relaciones de las figuras 2.3.5.6a y 2.3.5.6c se pueden suprimir. La información referente al cirujano David Rosen permanecerá en la relación de la figura 2.3.5.6b.

Actualización. Si John White ingresa al hospital por tercera vez y tiene una dirección distinta, el único lugar en que el cambio en la dirección se tendrá que hacer será en la

relación de la figura 2.3.5.6a y no en la de la figura 2.3.5.6c. Parece que existen algunas fallas de almacenamiento en la relación de la figura 2.3.5.6c.

Falla de inserción. No podemos introducir el hecho de que una determinada medicina produzca determinados efectos secundarios a menos que ésta se le suministre a un paciente.

No podemos introducir una cadena en la relación de la figura 2.3.5.6c hasta que tengamos un paciente que haya sido operado y al cual se le haya suministrado dicha medicina.



**Figura 2.3.5.7. Tres relaciones en la segunda forma Normalizada. La clave primaria de la relación PACIENTE es el número del paciente. La clave primaria de la relación CIRUJANO es, el número de licencia del cirujano. La clave primaria de la relación CIRUGIA es el número del paciente + número de licencia del cirujano + fecha de la cirugía.**

Falla de supresión. La falla de supresión en la cual un atributo que no es clave de una relación depende funcionalmente de otro atributo que tampoco sea clave permanente. Si John White recibe alguna otra medicina como consecuencia del severo prurito debido a la penicilina y se actualizan los valores de atributo de la medicina administrada y su efecto secundario, perdemos la información de que la penicilina produjo prurito. Puesto que esa cadena particular era la única fuente de información en la BD para esa categoría, esta pérdida no es deseable.

Falla de actualización. El efecto secundario de una medicina aparece varias veces en la relación de la figura 2.3.5.6c. Si el fabricante de una determinada medicina cambia la fórmula de tal manera que el efecto secundario cambie, nos encontramos con dos posibilidades: 1. Buscar en toda la relación de la figura 2.3.5.6c y cambiar el valor del efecto secundario cada vez que la medicina se haya administrado ó; 2. Trabajar con algunas inconsistencias como consecuencia de cambiar el valor sólo en algunas cadenas. Las fallas de inserción, supresión y actualización son los resultados de la dependencia del atributo que es clave "efecto secundario de la medicina" de otro atributo que tampoco sea clave, "medicina administrada después de la operación". Este tipo de dependencia se llama "transitiva" (véase la figura 2.3.5.8).

La solución a estos problemas de la relación de la figura 2.3.5.6c, es reemplazarla con otras dos relaciones, como en las figuras 2.3.5.9c y 2.3.5.9d.

Puesto que no hay fallas de almacenamiento en las relaciones de las figuras 2.3.5.6a y 2.3.5.6b, las dejamos intactas en las figuras 2.3.5.9a y 2.3.5.9b, respectivamente.

Se debe notar que la conversión de la segunda a la tercera forma normalizada es semejante a la conversión de la primera a la segunda forma, excepto al trabajar con atributos que no son clave. En la conversión de 1FN (primera forma normal) a 2FN (segunda forma normal), observamos las relaciones entre los atributos que son clave y

los que no son. Sin embargo, en la conversión de 2FN a 3FN observamos las relaciones entre los atributos que no son clave.

Las relaciones PACIENTE, CIRUJANO, PACIENTE-Y-CIRUJANO y MEDICINA, que se muestran en las figuras 2.3.5.9a, 2.3.5.9b, 2.3.5.9c, 2.3.5.9d, respectivamente, están en la 3FN. La figura 2.3.5.10 muestra estas cuatro relaciones en forma gráfica.

Note, también, que MEDICINA puede ser una clave para contestar preguntas como “¿Quién está tomando o tomó tetraciclina?”.

Relación en la tercera forma normalizada. Se dice que una relación está en la 3FN si no existe ninguna dependencia funcional transitiva entre los atributos que no son clave. (Cuando un atributo que no es clave se puede determinar con uno o más atributos que tampoco lo sean, se dice que existe una dependencia funcional transitiva entre los dos. El efecto secundario de una medicina se puede determinar a partir de la medicina administrada. Así, el efecto secundario de una medicina depende funcional y transitivamente de la medicina administrada). Alternativamente, si existen algunos atributos que no siendo clave tengan dependencia funcional entre ellos, la relación no es de la tercera forma.

Como consecuencia de la creación de dos relaciones, como las mostradas en las figuras 2.3.5.9c, 2.3.5.9d hechas de la relación de la figura 2.3.5.6c, las fallas de almacenamiento se eliminan.

Inserción. Podemos introducir el hecho de que una medicina determinada tenga determinado efecto secundario en la relación de la figura 2.3.5.9d, sin que haya administrado la medicina a ningún paciente.

Dependencia transitiva

| Número del paciente | Número de licencia del cirujano | Fecha de la cirugía | Cirugía                         | Medicina administrada después de la operación | Efecto secundario de la medicina |
|---------------------|---------------------------------|---------------------|---------------------------------|---|----------------------------------|
| 1111                | 145                             | Enero 1, 1977       | Extracción de cálculos biliares | Penicilina                                    | Prurito                          |
| 1111                | 311                             | Junio 12, 1977      | Extracción de cálculos renales  | -----   | -----                            |
| 1234                | 243                             | Abril 5, 1976       | Extracción de catarata          | Tetraciclina                                  | Fiebre                           |
| 1234                | 467                             | Mayo 10, 1977       | Eliminación de trombosis        | -----   | -----                            |

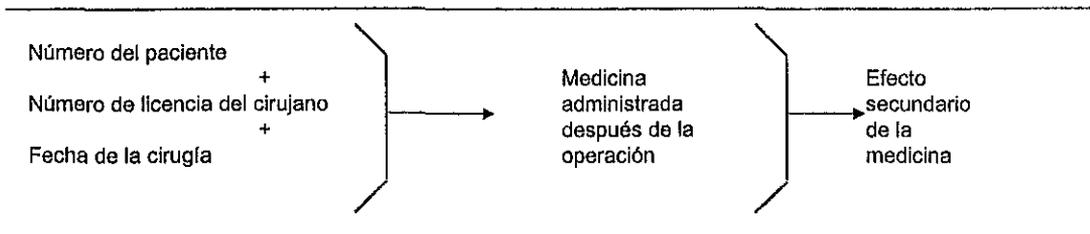


Figura 2.3.5.8. Representación de datos utilizando un modelo relacional.

Tabla PACIENTE

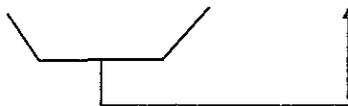


| Número del paciente | Nombre del paciente | Dirección del paciente             |
|---------------------|---------------------|------------------------------------|
| 1111                | John White          | 15 New Street, Nueva York, N.Y.    |
| 1234                | Mary Jones          | 10 Main Street, Rye, N.Y.          |
| 2345                | Charles Brown       | Dogwood Lane, Harrison, N.Y.       |
| 48776               | Hal Kana            | 55 Boston Post Road, Chester Conn. |
| 5123                | Paul Kosher         | Blind Brook, Mamaroneck, N.Y.      |
| 6845                | Ann Hood            | Hilton Road, Larchmont, N.Y.       |

**Figura 2.3.5.9a. Representación de datos utilizando el modelo relacional.**

Relación en la 3FN. La clave primaria es el número del paciente

| Número de licencia del cirujano | Nombre del cirujano |
|---------------------------------|---------------------|
| 145                             | Beth Little         |
| 189                             | David Rosen         |
| 243                             | Charles Fiedl       |
| 311                             | Michael Diamond     |
| 467                             | Patricia Gold       |



**Figura 2.3.5.9b. Representación de datos utilizando el modelo relacional.**  
 Relación en la forma normalizada, la clave primaria es el número de licencia del cirujano.

Supresión. John White recibe alguna otra medicina como resultado de haber presentado un severo prurito debido a la penicilina; el hecho de que haya sufrido ese efecto de la penicilina se puede registrar en la relación de la figura 2.3.5.9d.

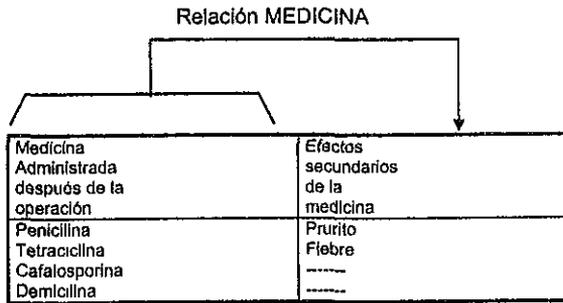
Actualización. El efecto secundario de cada medicina aparecerá sólo una vez en la relación de la figura 2.3.5.9d, resolviendo así la falla de actualización.

Es importante notar que, en el proceso de reducción de una relación en la 1FN a un conjunto de relaciones en la 3FN, no perdimos información. Esto significa que cualquier información que se pueda derivar de la relación en la figura 2.3.5.4 también se puede derivar de las relaciones en las figuras 2.3.5.9a, 2.3.5.9b, 2.3.5.9c y 2.3.5.9d. La relación de la figura 2.3.5.4 se puede construir combinando las relaciones de las figuras 2.3.5.9a a 2.3.5.9d. Otro punto que debe enfatizarse es que el proceso de reducción o normalización no es una función de los valores de los datos que aparecen en las relaciones en algún momento determinado: es una función de las relaciones entre los atributos. Para seguir el proceso de normalización, es absolutamente necesario que el diseñador de la BD entienda la semántica de la información. Dependiendo de las suposiciones sobre la dependencia funcional entre los atributos, el conjunto de relaciones en la 3FN para la empresa será diferente.

### Relación PACIENTE y CIRUJANO

| Número del paciente | Número de licencia del cirujano | Fecha de la cirugía | Cirugía                          | Medicina administrada después de la operación |
|---------------------|---------------------------------|---------------------|----------------------------------|---|
| 1111                | 145                             | Enero 1, 1977       | Extirpación de cálculos biliares | Penicilina                                    |
| 1111                | 311                             | Junio 12, 1977      | Extirpación de cálculos renales  | -----   |
| 1234                | 243                             | Abril 5, 1976       | Extirpación de catarata          | Tetraciclina                                  |
| 1234                | 467                             | Mayo 10, 1977       | Eliminación de trombosis         | -----   |

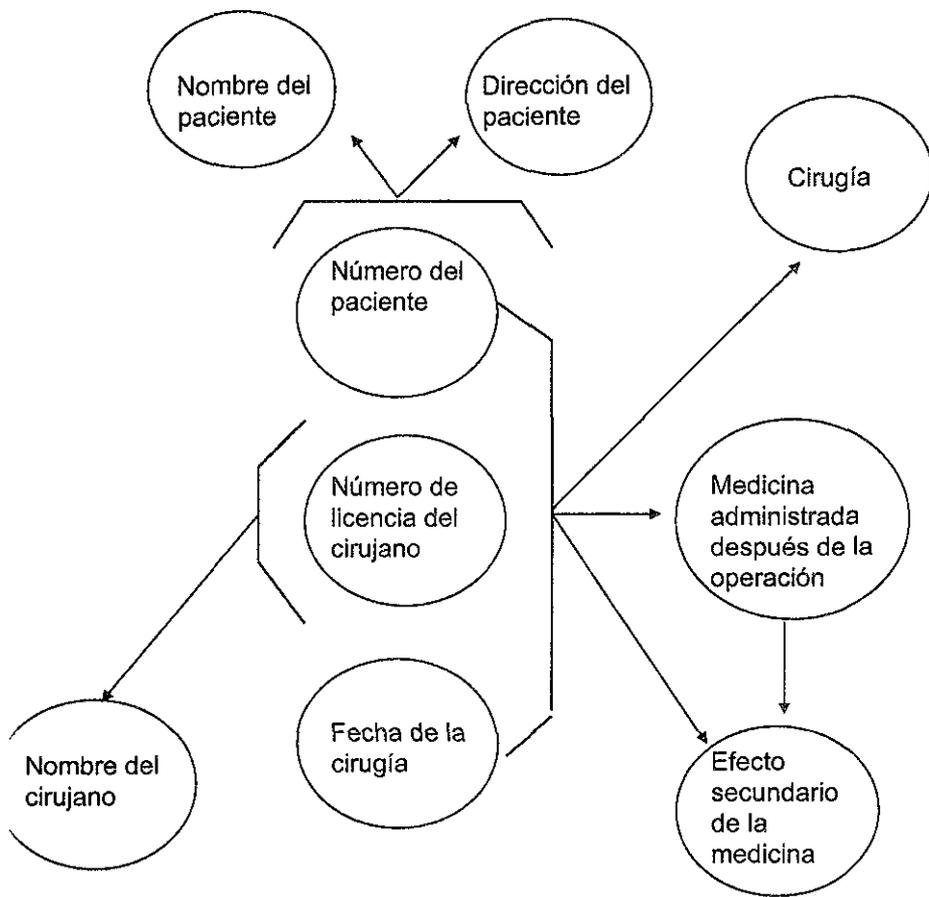
**Figura 2.3.5.9c. Representación de los datos utilizando el modelo relacional. Relación en la tercera forma normalizada. La clave primaria es el número del paciente + número de licencia del cirujano + fecha de la cirugía. Los atributos que no son clave son cirugía y medicina administrada después de la operación.**



**Figura 2.3.5.9d. Representación de datos utilizando el modelo relacional. Relación en la tercera forma normalizada. La clave primaria es medicina administrada después de la operación.**

1. Relación PACIENTE. La clave primaria es el número del paciente.
2. Relación CIRUJANO. La clave primaria es el número de licencia del cirujano.
3. Relación CIRUGÍA. La clave primaria es el número del paciente + el número de licencia del cirujano + la fecha de la cirugía.
4. Relación MEDICINA. La clave primaria es la medicina administrada después de la operación.

Como resultado del proceso de normalización, se derivan relaciones en la tercera forma normalizada. Estas relaciones representan a las entidades y sus relaciones y corresponden a la idea intuitiva de entidades: PACIENTE, CIRUJANO, CIRUGÍA y MEDICINA, estas cuatro relaciones representan al modelo conceptual.



**Figura 2.3.5.10. Cuatro relaciones en la tercera forma normalizada.**

## 2.4. METODOLOGÍAS DEL DISEÑO

### Problemas que se Presentan en el Desarrollo de sistemas

Los proyectos de software han tenido una serie de problemas debido a la falta de una metodología adecuada para desarrollarlos. Las dificultades que se presentan durante el desarrollo de un sistema, entre otras, son las siguientes:

1. Sistemas difíciles de administrar.
2. Resultados poco satisfactorios.
3. Programas poco confiables.
4. Sistemas inflexibles.
5. Sistemas difíciles de mantener.
6. Sistemas ineficientes.
7. Sistemas improductivos.

Las dificultades anteriores se salvan algunas veces de acuerdo con la habilidad del programador, pero siempre existen cuando no se sigue alguna técnica de diseño. Los orígenes de las dificultades son diversos, entre ellos se tiene por ejemplo:

1. Es difícil de administrar, debido a que cuando se va desarrollando los sistemas, estos van creciendo desmesuradamente y sin control, surgiendo problemas imprevistos, los cuales quedan ajenos al administrador, por lo que ya no pueden controlarlos.

Cuando los problemas son comentados al administrador, éste puede tomar decisiones improvisadas, lo que trae consigo, muchas veces, problemas diversos, tales como:

- a) Módulos repetidos.
- b) Cambio de políticas.

- c) Soluciones no factibles para el sistema.
- d) Etcétera.

2. Es muy mencionado que el desarrollo de software es poco productivo y poco profesional, ya que el programador tiene la idea, que una solución es la solución, lo cual no ocurre con otras ramas de la Ingeniería en las cuales se tienen distintas soluciones de las que se selecciona la mejor.

Lo que casi siempre se tiene al terminar un sistema son pocas satisfacciones pero muchas frustraciones, es común oír las siguientes frases:

- a) Lo hubiera hecho mejor de esta forma.
- b) Si yo hubiera sabido antes, no habría hecho esto.
- c) Tuve que hacer el programa varias veces, por problemas que se me presentaron.

3. La confiabilidad se ve afectada, ya que cuando se programa uno espera ciertos resultados, si éstos no aparecen, se va modificando el sistema hasta lograrlos, ocasionando varios cambios en la idea original, lo cual no garantiza que no se haya efectuado una transformación radical que pueda afectar al sistema, originando lo siguiente:

- a) Falle el sistema sin tener obvias razones.
- b) Produzca resultados erróneos en otras partes del sistema, que sería lo más grave. Como menciona Yourdon "Si un sistema trabaja en su totalidad, es un verdadero milagro".

4. Otro problema se presenta cuando el sistema se ha desarrollado, ya que por no seguir una técnica de diseño, éste se vuelve todo un laberinto, lo que ocasiona que se vuelva inflexible por el temor a modificaciones que produzcan resultados

incorrectos e inadvertibles.

Para resolver este problema, lo que se acostumbra es tener un grupo de programadores de tiempo completo, encargados del mantenimiento del sistema. El mantenimiento del sistema se refiere a las posibles modificaciones y a las actualizaciones del mismo. Nunca hay que confundirlo con el mantenimiento mecánico en el que existe desgaste físico, "un bit no se desgasta con el tiempo, ni es necesario poner dos IF en serie por si falla uno entra el otro".

5. También se menciona el alto costo de mantenimiento, el cual es elevado debido a que para hacer alguna modificación es necesario lo siguiente:
  - a) Estudiar a fondo todo el sistema.
  - b) Adaptar en lo posible el futuro cambio.
  - c) Analizar distintas alternativas.
  - d) Analizar las consecuencias del cambio a lo largo del sistema.

El alto costo se debe al tiempo invertido en hacerlo.

6. Se habla de ineficiencia desde dos puntos de vista:
  - a. Software.
  - b. Hardware.

Bastaría hablar de uno para darse cuenta de la ineficiencia verdadera, pero se hablará primero de eficiencia de hardware. Se dice o se decía que un sistema es eficiente, cuando se cumplen las siguientes condiciones:

- a) Ocupación de la menor cantidad de memoria posible.
- b) El menor tiempo del uso del procesador.

## 2.4.1. Las Diferentes Tendencias de las Metodologías de Desarrollo

Estas metodologías pueden clasificarse en pertenecientes a la escuela de ingeniería de software y en las pertenecientes a la escuela de ingeniería de la información; en orientadas a los procedimientos, a los datos o a la información, y en las que se desarrollan sistemas de tiempo real o de información.

### La Ingeniería de Software

La ingeniería de software es una fórmula **descendente** (top down) de la implantación por fases para el desarrollo de programas. Sus premisas básicas se representan en el cuadro 2.4.1.1. La ingeniería de software propone un proceso de desarrollo paso a paso que comienza con la visión funcional más general de lo que puede hacer el programa, descompone esta función en subfunciones y repite el proceso por cada subfunción hasta que todas las subfunciones sean lo suficientemente pequeñas para ser implantadas en código de programas. El resultado es un programa modular estructurado jerárquicamente.

Las premisas básicas son:

- Un planteamiento descendente (top down) de implantación por fases requiere un procedimiento gradual desde el nivel más alto de control y de definición de datos, descendiendo hasta los módulos funcionales y las estructuras de los datos.
- El proceso descendente requiere gran firmeza en los requerimientos.
- La solución del sistema se define en funciones; los datos se diseñan para preservar la funcionalidad del sistema y la independencia de los módulos del programa. Por tanto, normalmente los requerimientos de los datos se descubren a través del análisis de las funciones requeridas por el sistema.

#### Cuadro 2.4.1.1. Las premisas básicas de la ingeniería del software.

En la figura 2.4.1.2 se observan los pasos básicos seguidos de un método de ingeniería de software para un desarrollo. Nótese que el proceso comienza con una visión dinámica del flujo de datos antes de centrarse en una descomposición funcional.

Ahora se hablará de la eficiencia del software, lo cual implica hablar de:

- a) Programas claros.
- b) Modularidad del sistema.
- c) Programas documentados.

Lo que implica programas claros, que la modularidad del sistema ocasione que se pueda repetir código, lo cual trae consigo más espacio de memoria y que el texto del documento ocupe un espacio físico.

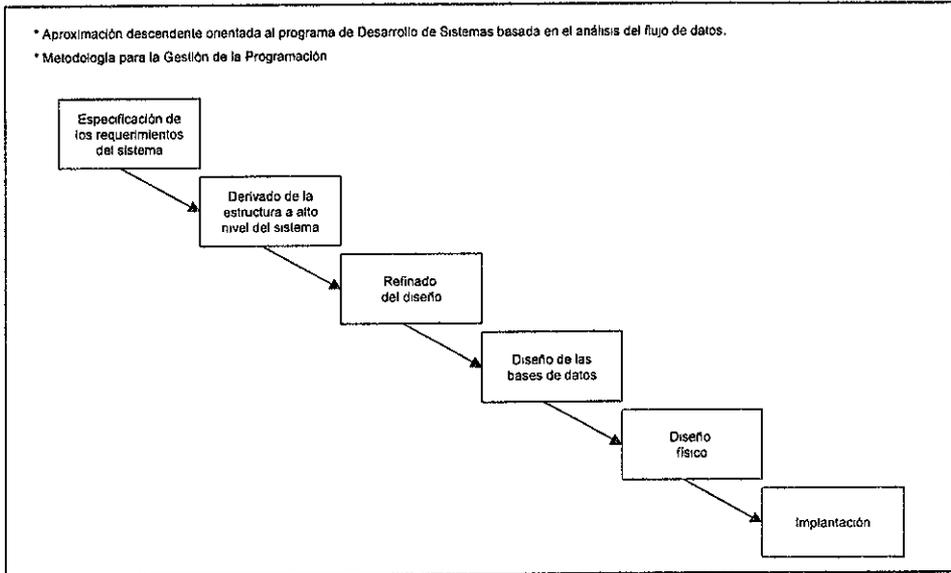
¿De lo anterior se puede concluir que al existir eficiencia en hardware, no se tiene eficiente en software o viceversa?

*No, ya que al repetirse el código no necesariamente trabaja más el procesador, ya que se eliminan muchos movimientos de disco a memoria. De igual manera se pueden refutar todos los demás puntos, pero basta saber que el costo de hardware es cada día menor a diferencia del costo de software que cada día es más caro. Además qué importa que un sistema ocupe más memoria a que tarde unos cuantos microsegundos más, si con eso se obtiene un sistema fácil de mantener.*

Los sistemas se vuelven improductivos, ya que se gasta la mitad del tiempo destinado al desarrollo del sistema, corrigiendo errores que los mismos programadores originan.

La construcción de software es un proceso que puede en ocasiones tornarse sumamente complejo, dependiendo de las características del sistema a ser desarrollado. Conforme se ha ido avanzando en esta disciplina, también se ha evidenciado la importancia de definir enfoques sistemáticos para el desarrollo de software. Cada uno de estos intentos por formalizar los procesos en la construcción de software es denominado como una metodología de desarrollo.

Obsérvese también que las consideraciones físicas y que el diseño lógico se separa de las consideraciones físicas y que el diseño lógico del programa precede al diseño físico. Los tipos de diagramas estructurados que se utilizan para representar un diseño de un programa de la ingeniería de software se muestran en el cuadro 2.4.1.3.



**Figura 2.4.1.2. Pasos básicos seguidos en la ingeniería de software.**

Los tipos de diagramas necesarios para soportar un planteamiento en ingeniería de software para desarrollo del sistema son:

- Diagrama de flujo de datos. Muestra los procesos funcionales desde el nivel más alto en un sistema y el flujo de datos desde la adquisición a través del proceso y las eventuales salidas.
- Diagramas estructurados en árbol. Muestran las relaciones entre los diseños del programa y muestran la estructura jerárquica de los datos.
- Diagrama detallado de la lógica del procedimiento. Muestra detalladamente la lógica del programa: secuencia, selección, iteración.

### **Tabla 2.4.1.3. Los diagramas de la ingeniería de software.**

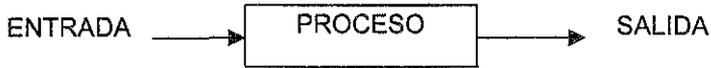
La ingeniería del software representa uno de los primeros intentos para reestructurar el proceso de desarrollo del software. Desde los años sesenta se ha avanzado en el cambio de desarrollo del software desde un arte manual a una disciplina casi mecanizada con el objetivo de introducir un enfoque normalizado al desarrollo de software. Los conceptos y técnicas involucrados en la ingeniería de software incluyen:

- El diseño ascendente (top down).
- La programación estructurada.
- La modularidad.
- El refinamiento progresivo y la descomposición funcional.
- La métrica de la calidad del software.
- La abstracción de los lenguajes de programación.

Las metodologías de análisis de DeMarco y Gane-Searson así como la metodología de análisis y diseño de Yourdon, son ejemplos de metodologías estructuradas populares pertenecientes a la escuela de ingeniería de software. La ingeniería de software es una disciplina general de desarrollo que puede aplicarse al desarrollo de los sistemas de información y a los sistemas de tiempo real.

## El Desarrollo Orientado al Procedimiento

La ingeniería de software está fundada sobre el modelo básico de entrada/proceso/salida de un sistema:



Los datos se introducen en el sistema como entradas, se opera con ellos (se transforman) por medio de un proceso y lo abandonan como salidas.

Este modelo básico se utiliza en todas las metodologías de diseño estructurado. Sin embargo, el orden en que se construye el modelo depende de la metodología particular que se sigue. Este tipo de desarrollo trata la parte del proceso como la fundamental del modelo del sistema y por tanto se describe como un enfoque del desarrollo del software orientado al procedimiento. Los datos se derivan de la función.

## El Desarrollo Orientado a los Datos

Es un enfoque orientado a los datos, la parte del modelado correspondiente a las entradas y salidas se considera como la más importante. Por tanto, las estructuras de los datos se definen primero y los componentes procedurales se derivan de las estructuras de los datos. El cuadro 2.4.1.4 muestra las premisas básicas de este enfoque de desarrollo dirigido por los datos, son diagramas de árbol estructurado jerárquicamente para representar las estructuras de los datos y de los programas, y el diagrama detallado de la lógica de procedimiento.

Las premisas básicas del enfoque dirigido por los datos son:

- La estructura del control del programa debe ser jerárquica y debe derivarse de la estructura de los datos del programa.
- El proceso de diseño consiste en definir primero las estructuras de los datos, mezclarlas luego todas en una estructura jerárquica de programa y después ordenar detalladamente la lógica procedural para que se ajuste a esa estructura.
- El diseño lógico debe estar separado del diseño físico y precederlo.
- Los tipos de diagramas necesarios para soportar un enfoque dirigido por los datos son:
- Diagrama estructurado en árbol jerárquico.
- – Diagrama detallado de la lógica del procedimiento.

#### **Cuadro 2.4.1.4. El enfoque dirigido por los datos para el desarrollo de sistemas.**

### **La Ingeniería de Información**

La ingeniería de información es un ejemplo de una formulación de desarrollo centrada en la información. En el cuadro 2.4.1.5 se exponen las premisas básicas de la ingeniería de la información.

Las premisas básicas de la ingeniería de la información son:

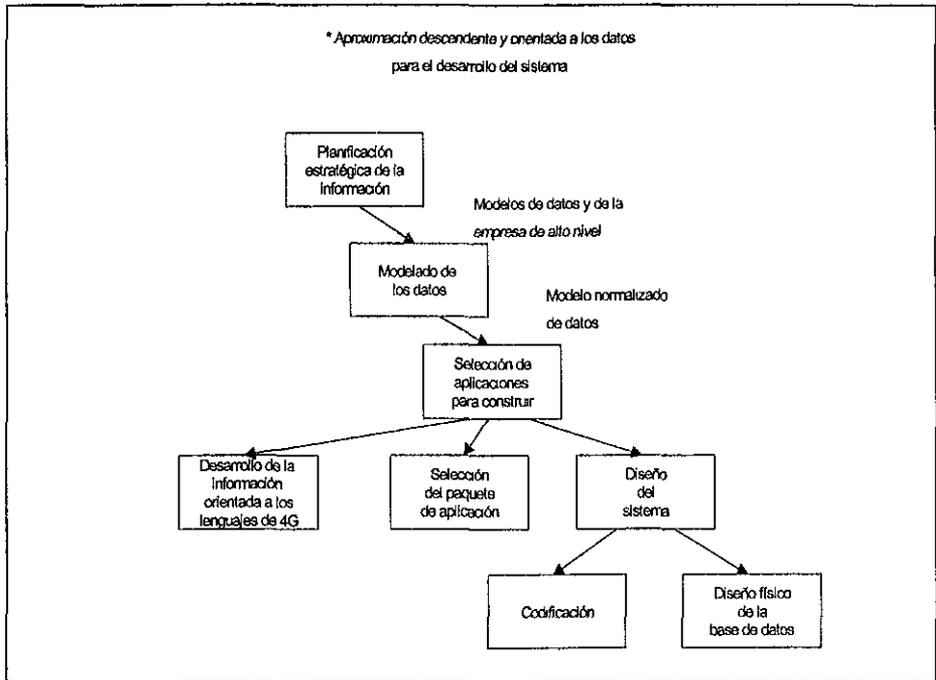
- Se necesita una estrategia general del desarrollo de sistema centrada en una planificación de los objetivos estratégicos del negocio para construir sistemas que satisfagan mejor las necesidades de la corporación.
- Sistemas de información que puedan integrarse mejor si los datos a compartir se controlan centralmente por ser parte de un mismo modelo lógico de datos.
- La presentación lógica de datos es relativamente estable, en tanto que los procedimientos que utilizan los datos cambian frecuentemente . Por tanto, el modelo lógico de datos, que refleja lo que es la organización, no la forma como trabaja, deberá ser la base del desarrollo del sistema.

#### **Cuadro 2.4.1.5. Las premisas básicas de la ingeniería de la información.**

La ingeniería de información es tanto un enfoque comercial de ingeniería como una formulación para la construcción de sistemas de software. Es una formulación comercial porque comienza con una planificación estratégica de la organización. Es una formulación de ingeniería porque promueve un procedimiento paso a paso para conseguir sistemas de información. La figura 2.4.1.6 describe los pasos lógicos de la formulación de la ingeniería de la información.

La ingeniería de la información es una disciplina más reciente que la ingeniería de software y en cierto sentido tiene un enfoque más amplio. Más que una disciplina para construir sistemas, es la construcción misma del sistema. Comienza por un nivel más alto (con la planificación estratégica) que la ingeniería de software. Sin embargo, en la fase de diseño del programa, su formulación del desarrollo de éste es básicamente la misma que la de ingeniería de software.

En otro sentido, sin embargo, la ingeniería de la información tiene un enfoque más estrecho que la ingeniería de software. Se emplea más aquella para construir sistemas de información, mientras que la ingeniería de software se emplea para desarrollar toda clase de sistemas (tanto de tiempo real como de información).



**Figura 2.4.1.6. Pasos básicos en ingeniería de información para desarrollo de sistemas.**

### **Sistemas de Tiempo Real**

Los sistemas de tiempo real típicamente son sistemas que controlan, y son controlados, por eventos externos. La respuesta a tales eventos (por ejemplo, señal, disparador, disponible/no disponible) de forma oportuna es una de las principales funciones de un sistema de tiempo real. Cuando el sistema reconoce un evento al recibir una señal de entrada, realiza ciertos cálculos y actividades lógicas para responder al evento produciendo una salida.

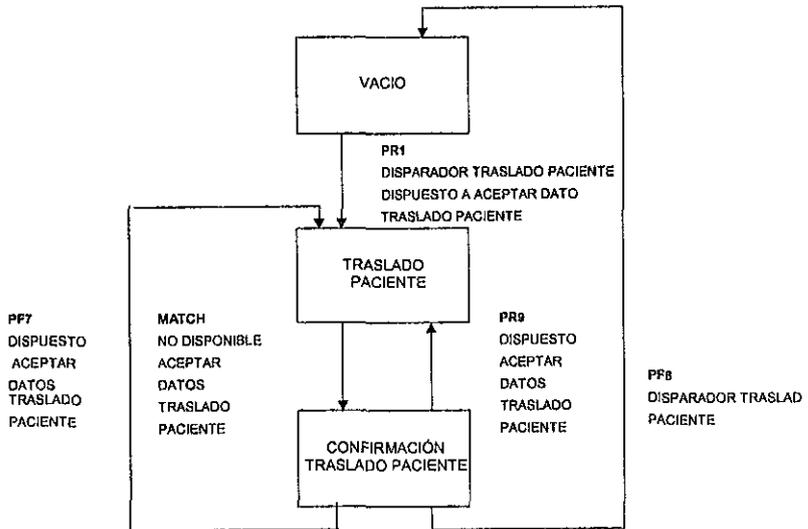
Los sistemas de tiempo real incluyen: la navegación aérea, las redes de comunicaciones, los sistemas de software, el control de procesos de fabricación y los procesos químicos entre otros.

Para especificar los requerimientos de un sistema de tiempo real se incluyen los conceptos para:

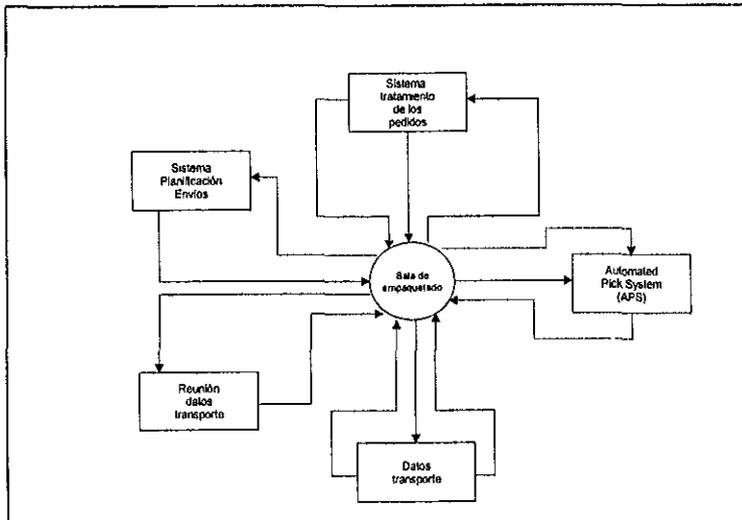
- El manejo de interrupciones.
- La comunicación y sincronización entre tareas.
- El proceso concurrente.
- La respuesta oportuna a los eventos externos.
- Los requerimientos y las restricciones de los sistemas de hardware.
- Las interacciones entre el sistema/entorno.
- Los datos continuos y los discretos.

Para representar los requerimientos de los sistemas de tiempo real se emplean tipos especiales de diagramas estructurados, entre los que se encuentran los flujos de control, de transición de estado, gráfica de contexto, matrices estado/suceso y las tablas de decisión. Por ejemplo, el diagrama de control de flujo se emplea para mostrar los procesos del sistema, el paso de las señales de datos de entrada/salida entre los procesos y las relaciones oportunas entre los procesos. El diagrama de transición mostrado en la figura 2.4.1.7 se utiliza para mostrar las condiciones y acciones que ocurren en el sistema al pasar de un estado a otro. La matriz proporciona una información más detallada al describir la lógica condicional empleada en un estado del sistema. El diagrama de contexto, como el mostrado en la figura 2.4.1.8, describe los límites del sistema.

La mayoría de estos diagramas son variaciones de los diagramas estructurados empleados para describir los sistemas de información. De modo similar, las metodologías de análisis y diseño utilizadas en sistemas de tiempo real son variaciones de las utilizadas en sistemas de información como por ejemplo los de Yourdon y DeMarco.



**Figura 2.4.1.7. Diagrama de transición de estados que muestra las condiciones y acciones que originan el cambio de un estado a otro del sistema.**



**Figura 2.4.1.8. Diagrama de flujo de control que muestra los límites del sistema.**

## 2.4.2. Estadísticos y Clasificación de Metodologías

En el cuadro 2.4.2.1 se listan las metodologías estructuradas de desarrollo más utilizadas de acuerdo con una investigación realizada entre más de mil empresas de EE.UU. En el mismo cuadro también se listan las principales metodologías que las organizaciones planean adoptar de acuerdo con la misma investigación.

| <b>LAS METODOLOGÍAS DE DESARROLLO<br/>MÁS UTILIZADAS EN ESTADOS UNIDOS</b> |       |
|--|-------|
| Análisis y Diseño Estructurado de Yourdon                                  | 29.1% |
| Análisis Estructurado de Gane-Searson                                      | 13.8% |
| Análisis Estructurado DeMarco  | 12.3% |
| Diseño Estructurado de Orr   | 7.2%  |
| Diseño Estructurado de Jackson   | 2.4%  |
| <br>   |       |
| <b>ELECCIÓN DE NORMALIZACIÓN DE<br/>METODOLOGÍAS DE DESARROLLO</b>         |       |
| Yourdon  | 24.9% |
| Gane-Searson   | 11.2% |
| Ingeniería de Información  | 11.2% |
| DeMarco  | 9.5%  |
| Orr  | 7.1%  |
| Jackson  | 3.6%  |

**Tabla 2.4.2.1. Las Metodologías de desarrollo más utilizadas.**

En el cuadro 2.4.2.2 que a continuación se presenta, se hace una clasificación de estas metodologías estructuradas.

| METODOLOGÍA                                      | CLASIFICACIÓN   |
|--|---|
| <b>DeMarco</b><br>Análisis estructurado          | <ul style="list-style-type: none"> <li>• Ingeniería de software</li> <li>• Orientados al procedimiento</li> <li>• Sistemas de información</li> <li>• Extensiones tiempo real</li> </ul> |
| <b>Gane-Searson</b><br>Análisis estructurado     | <ul style="list-style-type: none"> <li>• Ingeniería de software</li> <li>• Orientados al procedimiento</li> <li>• Sistemas de información</li> <li>• Extensiones tiempo real</li> </ul> |
| <b>Yourdon</b><br>Análisis y diseño estructurado | <ul style="list-style-type: none"> <li>• Ingeniería de software</li> <li>• Orientados al procedimiento</li> <li>• Sistemas de información</li> <li>• Extensiones tiempo real</li> </ul> |
| <b>Jackson</b><br>Diseño estructurado            | <ul style="list-style-type: none"> <li>• Ingeniería de software</li> <li>• Orientados al procedimiento</li> <li>• Sistemas de información</li> <li>• Extensiones tiempo real</li> </ul> |
| <b>Orr</b><br>Desarrollo de sistemas             | <ul style="list-style-type: none"> <li>• Ingeniería de software</li> <li>• Orientado a los datos</li> <li>• Orientado a las salidas</li> <li>• Sistemas de información</li> </ul>       |
| <b>Martin</b><br>Ingeniería de la información    | <ul style="list-style-type: none"> <li>• Ingeniería de la información</li> <li>• Orientado a la información</li> <li>• Sistemas de información</li> </ul>                               |

**Cuadro 2.4.2.2. Clasificación de las Metodologías.**

El objetivo de estas técnicas para diseño detallado es similar, pero cada una tiene características únicas para la definición y comunicación de las especificaciones del sistema. Se pueden utilizar en la comunicación de las especificaciones de diseño a los usuarios, aunque no necesariamente son fáciles de entender para una persona que no haya sido entrenada en la técnica. Otra ventaja de estos enfoques es que soportan el uso de programación estructurada. Un problema con todas las técnicas es que requieren una gran inversión en el entrenamiento y conversión de analistas hacia los nuevos procedimientos. Cada uno requiere más bien un compromiso completo y un cambio; no es fácil "prestar" elementos de cada uno y combinarlos.

Para la elaboración de este proyecto, se seleccionó la técnica de E. Yourdon o diseño estructurado del que hablaremos a continuación.

### **2.4.3. La Metodología de Análisis y Diseño Estructurado de Yourdon**

El estándar establecido para el desarrollo de sistemas de información en el Fidelist, es la metodología de análisis y diseño estructurado de Yourdon, la cual se clasifica como perteneciente a la escuela de la ingeniería de software, orientada a procedimientos y con soporte para sistemas de información, aunque posee extensiones para sistemas de información de tiempo real.

El contacto constante con esta metodología desde proyectos anteriores por parte de todo el personal involucrado en el desarrollo de sistemas, ha repercutido en una serie de beneficios, entre los cuales se pueden mencionar las siguientes:

- Los desarrolladores de sistemas han adquirido una amplia experiencia en el uso de la metodología, lo que les permite planear sus actividades de manera adecuada para construir sistemas útiles y eficientes, con la oportunidad requerida.

- Por su parte, los usuarios han llegado a comprender en buena medida el uso de la metodología, lo cual significa el establecimiento de un lenguaje común para el modelado en forma conjunta de los requerimientos reales de los sistemas.

Por supuesto que el Sistema de Autoabastecimiento Farmacéutico a Hospitales se trabajará con la metodología en cuestión, considerando que los beneficios mencionados repercutirán directamente en el proceso de desarrollo. Podemos considerar además que la naturaleza misma del sistema, donde existe alto flujo de información, permite que se adapte perfectamente para ser modelado mediante esta metodología.

Ya que la metodología de análisis y diseño estructurado de Yourdon será fundamental en el desarrollo de los capítulos posteriores, a continuación se realiza una explicación de las premisas básicas sobre las cuales se apoya.

Esta metodología de análisis de diseño estructurado usa los métodos de arriba-abajo y descomposición funcional para definir los requerimientos del sistema. Las especificaciones del sistema producidas por el proceso de análisis estructurado es un modelo descendente particionado del sistema a ser desarrollado. La descripción de los requerimientos del sistema a ser desarrollado se convierte en la liga entre el análisis y el diseño.

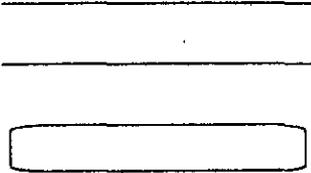
Los modelos que se realizan en el análisis del sistema son, en su mayoría, modelos de papel para el futuro sistema, o sea, representaciones abstractas de lo que al final será la combinación de hardware y software de computadora. Estos modelos enfatizarán ciertas propiedades críticas del sistema; con lo cual se prevé si la comprensión de los requerimientos fue la correcta, o no. Dando oportunidad de modificar o desechar el modelo o crear uno nuevo, de ser necesario.

En esta metodología se hace uso de las siguientes herramientas gráficas de modelado: Diagramas de Flujo de Datos, Diagrama de Entidad-Relación, Diagrama de Transición de Datos, Diagrama de Estructuras; las cuales ayudarán a concentrarse en las propiedades más sobresalientes del sistema, aclarar los cambios y correcciones de los requerimientos del usuario y respaldar con información documental el ambiente del usuario. Las herramientas mencionadas se describen a continuación:

### **El Diagrama de Flujo de Datos**

Describe los procesos (funciones o procedimientos) del sistema, es decir, la transformación de entradas y salidas. Esta es una herramienta que permite visualizar un sistema como una red de procesos funcionales, conectados entre sí por líneas de flujo y almacenamiento de datos. Los Diagramas de Flujo de datos (DFD) únicamente proporcionan un punto de vista de un sistema, el orientado a las funciones. Por lo anterior, si en un sistema las funciones no son el aspecto principal, es preferible concentrar la atención en otros diagramas que tengan más importancia.

Como puede apreciarse en el cuadro 2.4.3.1 que se muestra a continuación, los diagramas de flujo de datos se componen de procesos, flujos, agregados de datos y terminadores:

|  |  |
|--|--|
| <p><b>1. Procesos.</b> Representan las diferentes funciones que el sistema lleva a cabo. Éstas a su vez se transforman en entradas y salidas. El proceso se representa gráficamente por medio de un círculo, también llamado burbuja. El proceso se nombra o describe con una sola palabra, frase u oración sencilla de quién o qué lo está efectuando, más que describir el proceso mismo.</p>  |    |
| <p><b>2. Flujos.</b> Son conexiones entre los procesos e indican la información que se requiere como entrada y la información que genera como salida de cada proceso, representan datos en movimiento. Se representan lógicamente por medio de una flecha que entra o sale de un proceso. Pueden ser acompañados por un nombre señalado el significado del paquete de información que se mueve a lo largo del flujo. Si los datos se están moviendo hacia adentro o hacia fuera de un proceso (o ambas cosas).</p> |    |
| <p><b>3. Agregados o almacén de datos.</b> Son colecciones de datos (agregados) que el sistema guardará para ser llamados en un periodo de tiempo (archivos a bases de datos). Los agregados se utilizan para modelar una colección de paquetes de datos en reposo. Se denota por dos líneas paralelas o elipses.</p>  |   |
| <p><b>4. Terminadores.</b> Muestran las entidades externas con las que se comunicará el sistema, como por ejemplo grupos de personas, sistemas de cómputo externo y organizaciones externas. Gráficamente se representan con un rectángulo, y señalan entidades externas con las cuales el sistema se comunica.</p>  |  |

**Cuadro 2.4.3.1. Diagramas de flujo.**

Para un sistema de tiempo real, se manejan los flujos de control (es decir, señales o interrupciones), procesos de control (cuya labor sea la de coordinar y sincronizar las actividades de otros procesos del DFD), y agregados de control. Además, para mostrar con más detalle el diagrama de flujo de datos se hace uso de dos herramientas textuales de modelado: el diccionario de datos y la especificación de procesos.

Algunas reglas que ayudarán para elaborar un DFD con el mínimo de errores y cambios (por ejemplo, incompletos o lógicamente inconsistentes) y sean lo más entendibles por el usuario son las que a continuación se mencionan:

- Escogen nombres con significado para los procesos, flujos, almacenes y terminadores.
- Enumerar los procesos.
- Redibujar el DFD tantas veces sea necesario estéticamente.
- Evitar los DFD excesivamente complejos.
- Asegurarse que el DFD sea internamente consistente y que también lo sea con cualquiera DFD relacionado con él.

### **El Diagrama Entidad Relación**

Los diagramas de flujo de datos sólo muestran un aspecto principal de un sistema: sus funciones. No obstante, es necesario conocer a más detalle la relación que existe entre agregados de datos. Este aspecto es elaborado con la ayuda de otra herramienta llamada Diagrama de Entidad Relación (DER), el cual enfatiza las relaciones entre los almacenes de datos en el DFD, que de otra forma se hubieran visto sólo en la especificación del proceso.

Estos diagramas se utilizan para mostrar el tipo de relación existente entre entidades diferentes de un sistema, mismas que pueden ser "uno a uno" o "muchos a muchos". Un DER típico se muestra en la figura 2.4.3.2 , cada una de las cajas rectangulares corresponde a un almacén de datos en un DFD, y puede observarse que hay

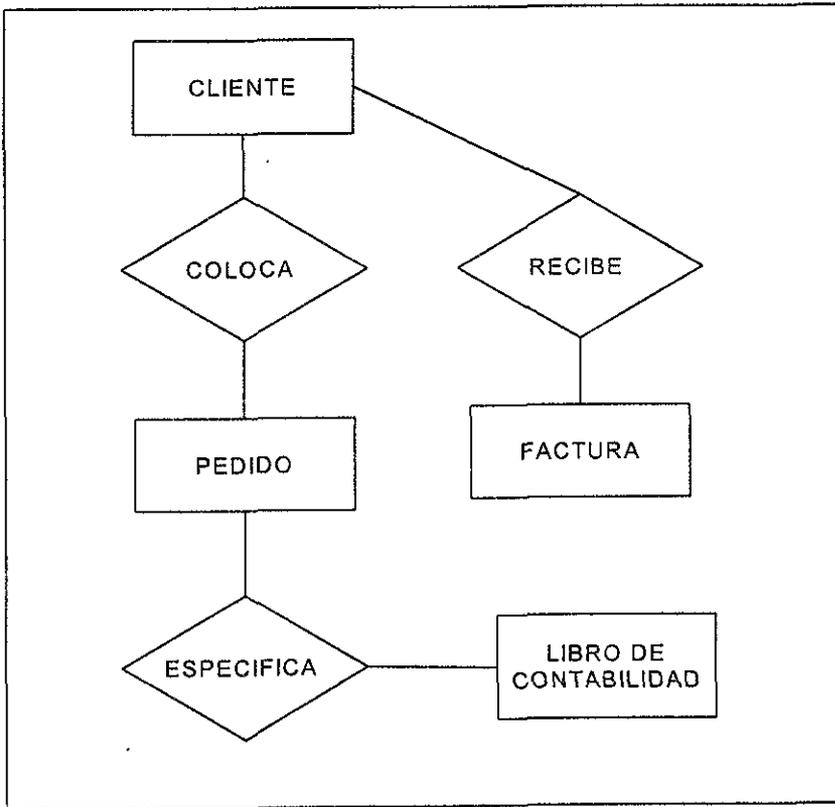
relaciones (conexiones) que normalmente en él no se aprecian. En ocasiones se encontrará que construir primeramente el modelo de datos, hace más fácil descubrir cuáles son las funciones requeridas.

El diagrama de entidad-relación consta de: Tipos de Objetos y Relaciones

1. **Tipos de Objetos o Entidades.** Se caracteriza porque cada una puede identificarse de manera única por algún medio, cada uno juega un papel necesario en el sistema que se construye y cada uno puede describirse por uno o más datos. Son representados por medio de un rectángulo en el diagrama y señalan una colección o conjunto de objetos (cosas) del mundo real; pueden ser identificados de manera única y ser descritos por uno o más atributos.
2. **Relaciones.** Son representadas por rombos y señalan una serie de conexiones o asociaciones entre los tipo objetos (entidades) que están conectados con la relación apropiada por medio de flechas. Cada instancia de la relación representa una asociación entre cero o más ocurrencias de un objeto y cero o más ocurrencias del otro. Una situación más común es ver múltiples relaciones entre múltiples objetos.

De igual forma, que en el diagrama de flujo de datos, es necesario acompañar el diagrama de entidad-relación con información textual detallada. De esta manera, también podrá usarse el diccionario de datos para mantener información apropiada acerca de objetos y relaciones.

Entre las reglas de refinamiento del DER se podrán mencionar las de llevar a la creación de tipos adicionales de objeto, mientras que otras llevarán a la eliminación de objetos y/o relaciones.



**Figura 2.4.3.2. Diagrama de Entidad-Relación típico.**

### **Diagrama de Transición de Estados**

Es la secuencia con la cual se tendrá acceso a los datos y se ejecutarán las funciones. Esto más que nada se refiere a que en algunos sistemas computacionales, la función N no puede llevara cabo su labor hasta que recibe la entrada que requiere; y esta entrada se produce como salida de una función N-1, y así sucesivamente.

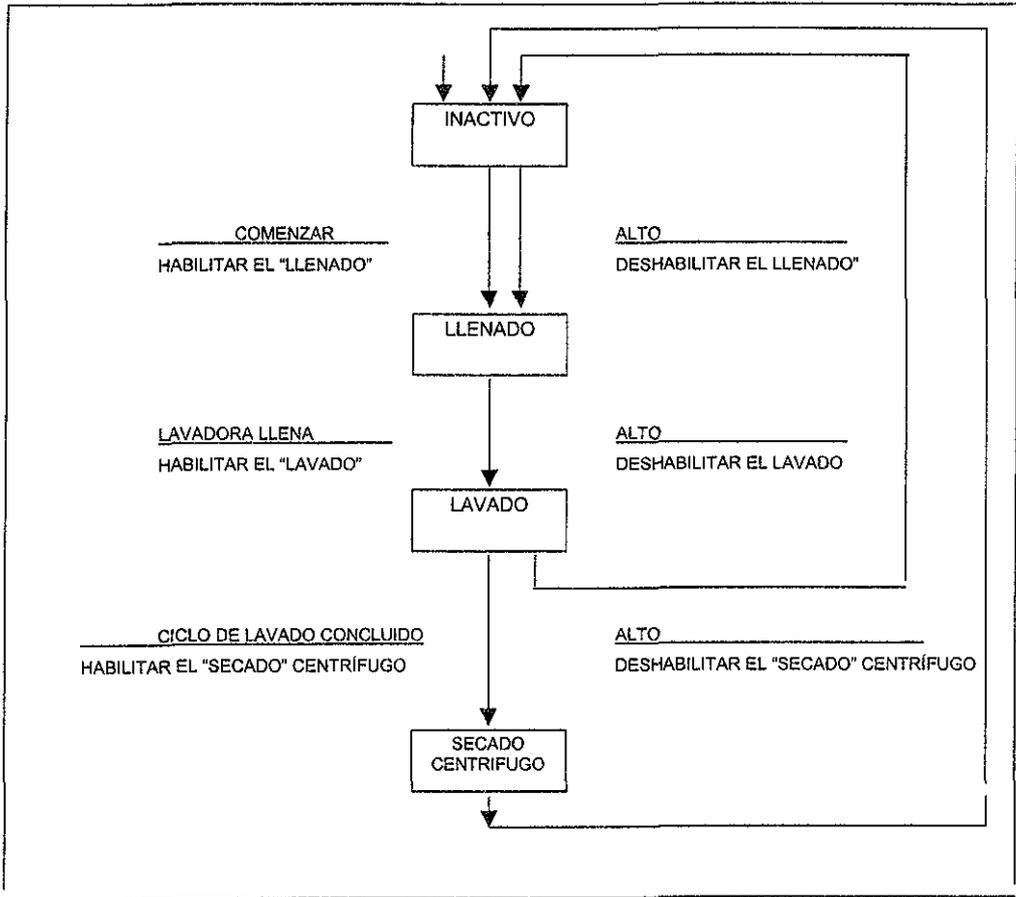
Los estados en los que se puede presentar el sistema ("escenarios" o "situaciones" reconocibles), representan entonces un periodo durante el cual, el sistema sigue algún comportamiento. Cualquier estado en el que el sistema se pueda encontrar sólo puede corresponder a periodos en los que: está esperando que algo ocurra en el ambiente

externo o está esperando a que alguna actividad presente en ese momento cambie a otro estado.

Además existen acciones asociadas con cada cambio de estado, esto es, respuestas, salidas o actividades que se llevan a cabo como parte de cambio de estado. Para realizar estos cambios de estados será necesario establecer una serie de condiciones en el sistema. Una condición es un acontecimiento en el ambiente externo que el sistema es capaz de detectar; típicamente es una señal, una interrupción a la llegada de un paquete de estados. Por lo que, las acciones que se muestran en un Diagrama de Transición de Estados (DTE) son respuestas regresadas al ambiente externo o bien cálculos cuyos resultados el sistema recuerda para poder responder algún acontecimiento futuro. En la figura 2.4.3.3 se presenta un DTE.

Existen dos formas para la construcción de un DTE, las cuales son las siguientes:

1. Comenzar por identificar todos los posibles estados del sistema y representan cada uno como una caja separada en una hoja de papel. Después, se puede explorar todas las conexiones con significado (es decir, los cambios de estados) entre las cajas.
2. Otra forma sería comenzar por el estado inicial y luego metódicamente ir siguiendo un camino hasta él o los estados restantes; después proseguir con los estados secundarios, los terciarios, etc.



**Figura 2.4.3.3. Diagrama de Transición de Estados.**

Algunas reglas que verifiquen la consistencia de un DTE preliminar son:

- Verificar cuidadosamente si no existe algún otro comportamiento, o alguna otra condición en la que el sistema pudiera estar, aparte de las que se han identificado.
- El sistema puede tener uno o más estados finales con múltiples entradas pero todos los demás estados deben tener un sucesor.

- Especificar el comportamiento del sistema ante condiciones inesperadas.

En la mayoría de los casos, el DTE representa una especificación del proceso para una función de control en un DFD. Presenta todo el sistema con un diagrama de una función del DFD, puede usarse el DTE para mostrar la secuencia de actividades en el sistema.

### **Diagrama de Estructuras**

Una herramienta gráfica de modelado que desarrollan los diseñadores de sistemas y no los analistas propiamente dicho, son los Diagramas de Estructuras (DE) que representan la jerarquía de software. En la figura 2.4.3.4 se muestra un diagrama típico, donde se aprecia que cada rectángulo representa un módulo. Las flechas que conectan los rectángulos representan las invocaciones de los módulos (llamados de subrutinas o llamados de procedimiento). Así también, se muestran los parámetros de entrada que se dan a cada módulo invocado, y los parámetros de salida devueltos por cada módulo cuando termina su labor y le devuelve el control al que lo llama.

Con el DE no señala los requerimientos del sistema y al usuario no le importa la descripción de la organización de subrutinas que usará el sistema en cuanto a software, por tanto, este diagrama es una herramienta exclusiva para los diseñadores de sistemas, pues modela un aspecto de la implantación del sistema.

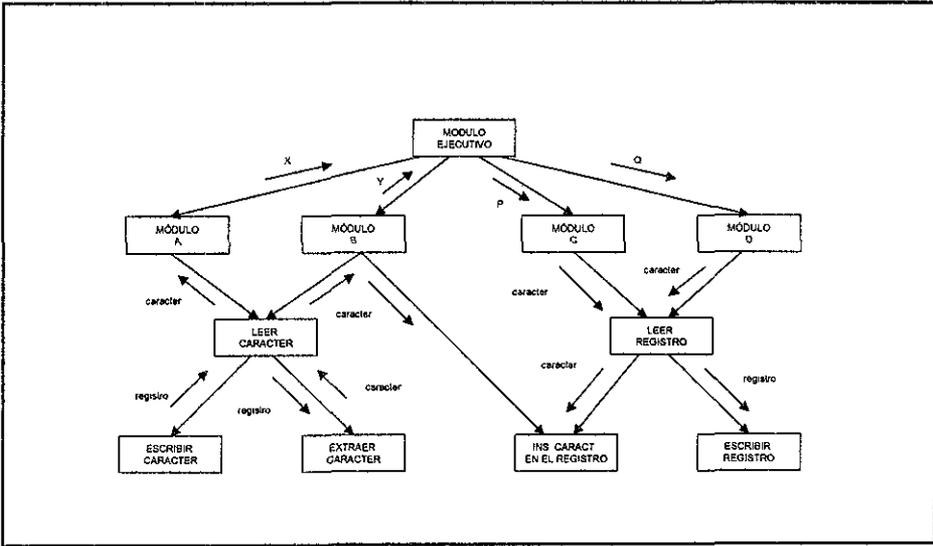


Figura 2.4.3.4. Diagrama de estructuras típico.

## 2.5. ANÁLISIS DEL SISTEMA

### 2.5.1. Situación Actual y Situación que se Pretende Alcanzar

En la fase del Análisis del Sistema es donde se realiza un estudio general del mismo, por ello definimos la situación actual y la situación que se pretende alcanzar con el sistema de Autoabastecimiento Farmacéutico a Hospitales. Es indispensable tener y conocer los requerimientos de la empresa, con el fin de desarrollar en una forma óptima el sistema que solicita y además dar soluciones.

Actualmente, en la empresa farmacéutica, se lleva un proceso para el abastecimiento a hospitales, el cual consiste en los siguientes pasos, mostrados en la figura 2.5.1.1.

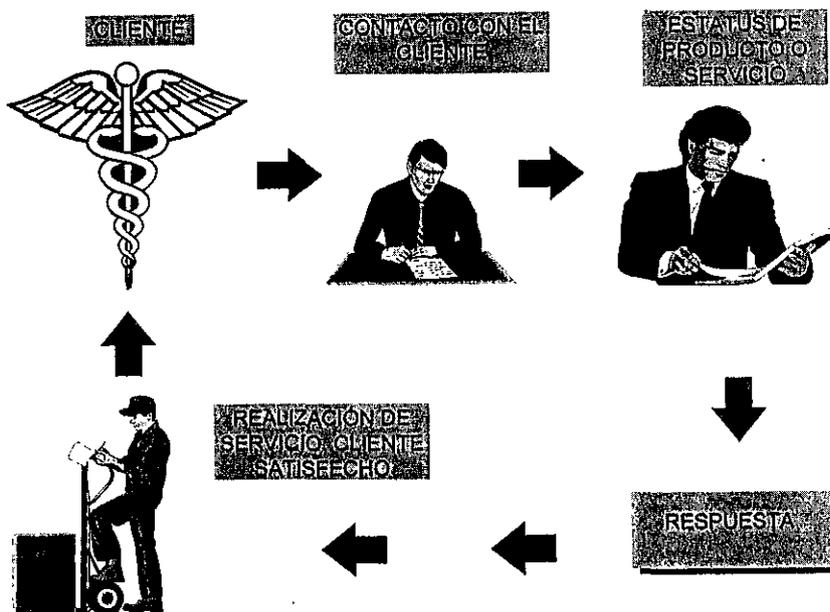


Figura 2.5.1.1. Esquema actual del proceso de abastecimiento.

El procedimiento de Abastecimiento inicia cuando un cliente quiere adquirir un servicio o producto de la empresa, por lo que surge la necesidad de conocer las características del cliente (en nuestro caso particular, los hospitales), por ejemplo: su nombre, su dirección, sus almacenes, etc. Una vez que se ha identificado el cliente, la empresa obsequia al cliente un catálogo de productos y servicios.

El cliente tiene tres medios de comunicación (teléfono, fax y personalmente) para informar sus requerimientos:

- a) Realizar pedidos.
- b) Consultar productos o servicios.

Los casos anteriores tienen un seguimiento diferente entre ellos. En el primer caso las peticiones se enviarán por escrito vía fax o con una persona específica (antes de la 1 p.m., pues de lo contrario se incrementará un día el tiempo de respuesta).

El seguimiento de las peticiones escritas se realiza en varios días, pues la empresa realiza un estudio minucioso del estatus del producto, por ejemplo: si existe en inventario y/o esta agotado.

Si tiene la cantidad disponible que se le ha requerido, en caso contrario se enviará en formato con una lista de los productos y cantidades requeridas al departamento de producción.

Verificar que los productos existentes cumplan con tamaño y capacidad solicitada.

Verificar qué productos son los primeros en salir del almacén de acuerdo al método **FIFO** (primero en entrar primero en salir).

Verificar que el costo del producto sea igual o menor al precio de la competencia.

Verificar si en el pedido hay un producto con descuento.

Determinar tiempos de distribución de pedidos para cada cliente.

Verificar que toda la documentación de cada producto o lote de productos sea la correcta.

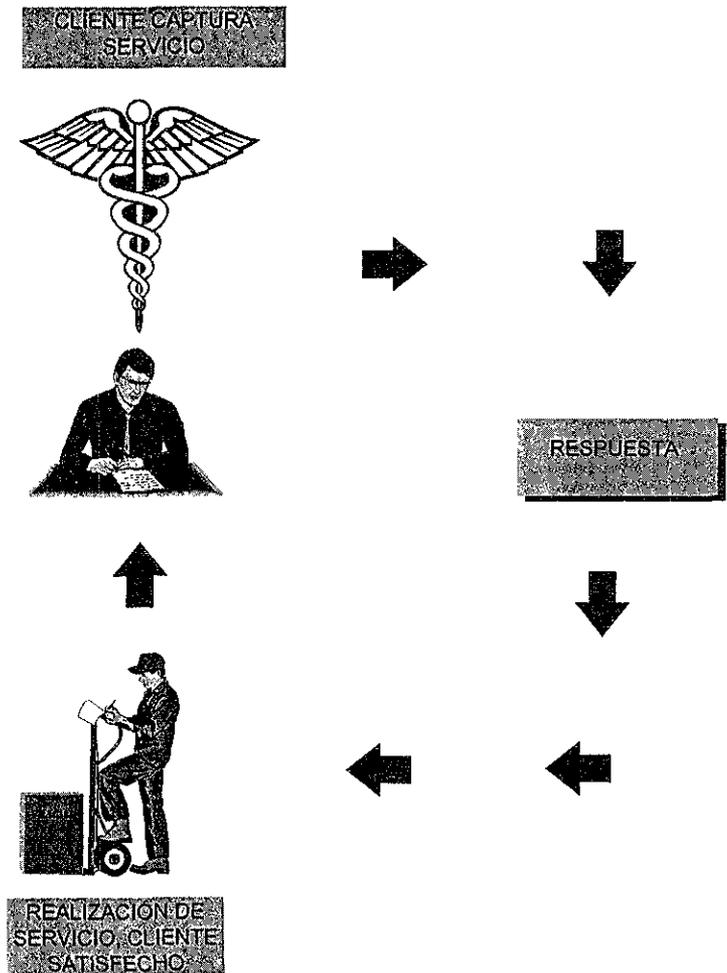
Todos los procedimientos tienen un seguimiento por medio de formatos, éstos serán enviados a los diferentes departamentos que estén involucrados con cada operación ya sea para una petición o un visto bueno.

Efectuado el paso anterior se le dá una respuesta escrita el cliente de estatus del producto, el cliente entonces toma una decisión de continuar el procedimiento del pedido.

En el segundo caso, Consulta de un Servicio o Producto, se utilizará el tercer medio de comunicación, el cliente con una llamada telefónica, se informa de una operación en particular del estatus del producto o servicio, la consulta se realiza en un mínimo de 4 horas y se contacta al cliente para darle el resultado.

El siguiente paso es realizar los reportes de operaciones hechas para satisfacer el pedido, esto es, documentar o listar las bajas en inventario, qué productos compró en una determinada fecha un determinado cliente, con qué costo al cliente adquirió un conjunto de productos, las facturas de compraventa, etc. Todos los reportes necesarios se cotejan con los departamentos involucrados y la mayoría de las veces estos resultados no cuadran entre sí.

Con el desarrollo del Sistema de Autoabastecimiento a Hospitales se desea que el proceso de abastecimiento funcione de una manera rápida, veraz y confiable. En la figura 2.5.1.2 se refleja la situación que se pretende alcanzar.



**Figura 2.5.1.2. Nuevo esquema del proceso de abastecimiento.**

Este nuevo proceso simplifica el abastecimiento de productos farmacéuticos, donde el cliente tiene una respuesta inmediata y participa en forma interactiva con el proveedor.

El proceso consiste en registrar al cliente en la BD del sistema, paso siguiente es generar un archivo tipo texto en el que se especifica el producto, hospital, almacén y

cantidad del producto para realizar el pedido (dependiendo del servicio que requiera el cliente), este archivo será tomado por el sistema para llenar la base de datos y dar la respuesta inmediata por medio de listados o reportes del estatus del producto. Todos en la empresa podrán tener acceso al sistema y así dar un control a los productos y el cuadro de formatos en cada departamento será el mismo. El sistema de Autoabastecimiento a Hospitales agilizará y facilitará el control de productos dando una seguridad y confianza en el proceso.

Al terminar la etapa de planteamiento de los objetivos se aclararán las necesidades del usuario y objetivos del proyecto. Sin embargo, hasta este momento no se ha constatado ni la factibilidad de objetivos, ni se han seleccionado entre las alternativas los métodos o medios para lograrlo.

La etapa de análisis de objetivos consiste en estudiar los objetivos conflictivos o antagónicos, señalar propiedades, constatar factibilidad y proponer los métodos de solución que sirvan de base para el diseño del sistema de autoabastecimiento.

### **2.5.2. Requerimientos del Usuario**

Al terminar la etapa de análisis de objetivos, se conocen las funciones del sistema y los métodos de solución que deberán ser empleados para satisfacer las necesidades del usuario. Esta información, reportada en los documentos de objetivos del proyecto y reporte de análisis se redactan en un lenguaje especializado, familiar a los analistas de sistemas. Los resultados de la etapa de especificación de requerimientos se resumen en un documento de especificaciones. Este documento deberá ser tangible, formal, completo y modificable.

Tangible. El documento debe ser claro a las tres partes involucradas: usuarios, analistas y especialistas en cómputo.

Formal. Las especificaciones se redactarán de tal manera que todo requerimiento se identifique explícitamente como tal, no dando lugar a malas interpretaciones, que redundarían en un producto que no satisfaga los requerimientos.

Completo. Debe cubrir todos aquellos aspectos que deben dejarse al criterio del diseñador. Todo lo que no sea explícitamente definido en este documento, dará el margen para que el diseñador decida la forma de tratarlo.

Modificable. Debe ser estructurado, redactado y almacenado de tal manera que admita cambios con un mínimo de esfuerzo y costo. Su alcance termina donde comienza el dominio del diseño del sistema. Determinar la forma como el sistema será realizado es el propósito de la fase de requerimientos del usuario.

Con los requerimientos del usuario, se definen los alcances y limitaciones del proyecto, esto hace importante la fase de análisis y diseño para el ciclo de vida del desarrollo de sistemas.

La información de operatividad, control de datos, factibilidad y alcance del proyecto son los elementos básicos para el diseñador, pues le permiten conocer toda la problemática y dar una óptima solución a la misma. Encontrar la óptima solución es responsabilidad tanto del diseñador como del usuario, ya que este último comunica al diseñador toda la información acerca de lo que debe hacer el sistema. Es de vital importancia recopilar toda la información posible, pues un mal diseño puede alargar el tiempo de desarrollo del sistema, tener una mala asignación del número de personas al proyecto, estimar o subestimar el costo del proyecto o definitivamente un reanálisis y rediseño del proyecto.

A continuación se listan los requerimientos que se pretenden cumplir con el Sistema de Autoabastecimiento a Hospitales:

1. Diseñar un sistema que permita tener bien identificado a cada cliente.
2. Evitar al acumulamiento de torres de papel donde se identifica el estatus de un producto (véase la figura 2.5.2.1).

| Clave<br>Producto | Costo | Existencia | Cantidad | Fecha<br>Caducidad | Descuento<br>(%) |
|-------------------|-------|------------|----------|--------------------|------------------|
| JER5              | 100   | S          | 10000    | 12/12/98           | 0                |
| GAS20             | 50    | N          | 0        |                    | 0                |
| PEN05             | 200   | S          | 500000   | 25/12/2020         | 50               |

**Figura 2.5.2.1. Estatus de productos solicitados.**

3. Que el sistema permita estandarizar las entradas y salidas de los archivos textos donde todos los clientes especifican el pedido a realizar.
4. El sistema debe contar con una BD que permita relacionar hospitales, almacenes, productos, pedidos, costos y descuentos.
5. Que el sistema permita la actualización y recuperación eficientes de datos.
6. El sistema debe contar con validación y verificación de la información para que no exista redundancia en la BD.
7. El sistema debe generar los tres tipos de archivos texto para realizar los pedidos.
8. El sistema debe permitir almacenar toda la información relacionada con los productos y asimismo procesarla de tal forma que sea organizada, resumida y detallada para ser comprendida fácilmente por cualquier persona.
9. El sistema debe presentarse en forma amigable y de un sencillo manejo de operación, pues no debe requerir de conocimientos complejos de computación.
10. El sistema debe ser flexible para la adición de nuevos módulos.
11. El sistema debe realizar una gráfica de los pedidos realizados.
12. Elaborar un manual del usuario donde indique la operación del sistema.
13. El sistema debe mostrar la visión de la empresa.

### **2.5.3. Recopilación y Análisis de la Información**

El primer paso en la recopilación de información es elevar la importancia que tiene el estudio a realizar para implantar el nuevo sistema, con lo anterior es obligatorio informar e identificar a todas las personas que están relacionadas con los procesos a automatizar dentro del proceso de autoabastecimiento, resaltar que su participación es indispensable y que deberán dar información lo más detallada y clara posible. Por otro lado también se les mencionarán los beneficios del sistema y que es lo que en particular esperan del sistema y con ello decidir en el análisis qué es lo posible o imposible para implantar el sistema.

El segundo paso es conocer qué personas están en contacto directo con el proceso de autoabastecimiento.

Entrevistar a cada persona de una forma muy particular para observar:

1. Qué información le llega y quién se la facilita o cómo la obtiene.
2. Cómo procesa la información para realizar su proceso o tarea.
3. A quién le envía su información procesada y cómo se la envía.

Identificar las aplicaciones existentes y si estas satisfacen las necesidades del proceso.

Identificar a las áreas o departamentos ajenos que se involucren en el uso y distribución de la información del pedido de productos.

Identificar los beneficios que puede aportar la implantación del sistema.

El proceso de abastecimiento se dividió en módulos para tener un mejor análisis, a continuación se describen:

| <b>Requerimiento</b>                  | <b>Seguimiento</b>  |
|---------------------------------------|---|
| Solicitud de Pedido                   | Obtener información del cliente. Identificar al cliente, esto es, registrar su dirección, sus almacenes y por supuesto sus requerimientos de consumo. Lo anterior se controla por formatos y se archivan en carpetas (véase la figura 2.5.3.1).   |
| Actualizar el inventario de productos | Relación de productos en existencia incluyendo costo, cantidad, tamaño, etc. Identificar productos faltantes y solicitarlos a producción para su registro en bodegas.   |
| Determinación de pedidos              | Relación de productos, verificar su existencia de lo solicitado en inventario. Identificar qué productos tienen descuento, costo de los productos, fechas de vencimiento, documentación de la elaboración del lote que se venderá, etc. Los pedidos se realizan por formatos en los cuales el cliente anota las cantidades y productos requeridos; el departamento de ventas pasa un informe al departamento de producción (véase la figura 2.5.3.2). |

| Clave<br>Cliente | Nombre  | Dirección        |
|------------------|---------|------------------|
| ANGE             | Ángeles | Av. Insurgentes  |
| INGL             | Inglés  | Av. Observatorio |
| ABC              | H. ABC  | San Fernando     |

| Clave<br>Cliente | Almacén                      | Clave Almacén |
|------------------|------------------------------|---------------|
| ABC              | Almacén uno de ABC           | ABC1          |
| INGL             | Inglés Sucursal Observatorio | INGO          |
| ABC              | Almacén dos de ABC           | ABC2          |

**Figura 2.5.3.1. Información relacionada al cliente y sus almacenes.**

| Clave de Producto | Producto                       |
|-------------------|--------------------------------|
| JER5              | Jeringa de 5ml.                |
| JER2              | Jeringa de 2ml.                |
| ALG50             | Algodón de 50gr.               |
| ALG1K             | Algodón de 1kg.                |
| VEN03             | Vendas de 3m.                  |
| GAS20             | Gasas esterilizadas de 20x8cm. |
| PEN05             | Ampolletas de Penicilina 5ml.  |

**Figura 2.5.3.2. Lista de productos solicitados por el cliente.**

Como es bien sabido todo proceso que se divida en subprocesos es mejor controlarlo, se trabajará módulo por módulo, así el control de cada uno de ellos será más sencillo y el número de tareas específicas será más pequeño, las divisiones se realizaron con supervisión de personal experto en el proceso de abastecimiento. Cabe mencionar que el proceso de abastecimiento consume el mayor tiempo en determinar los documentos que respaldan la realización del pedido, pues egresos e ingresos deben cuadrar en los reportes.

## **2.5.4. Propuestas de Solución**

### **2.5.4.1. Sistemas Administradores de Bases de Datos para PC's y elección de alguna (Características, Ventajas y Desventajas)**

El objetivo de un sistema de Base de Datos (BD) es simplificar y facilitar el acceso a los datos. Las vistas de alto nivel ayudan a lograrlo. Si el tiempo de respuesta para una consulta es demasiado largo, el valor del sistema se reduce. El funcionamiento del sistema depende de la eficiencia de las estructuras de datos utilizados para presentar los datos y de qué tan eficientemente pueda operar el sistema con esas estructuras. Como sucede en muchos otros aspectos de los sistemas de cómputo, deben hacerse concesiones, no sólo entre el espacio y el tiempo, sino también entre la eficiencia de un tipo de operación y la de otro.

La aparición de la computadora personal en el mercado, y su rápida evolución tecnológica, así como el abatimiento del costo, ha propiciado su uso en gran escala. En consecuencia ha surgido una gran cantidad de software de BD.

Por la facilidad de implantación del modelo relacional, el desarrollo de los administradores de BD se ha orientado a éste. El poder de la facilidad de uso son los aspectos fundamentales para los usuarios de bases de datos relacionales.

El manejador de BD es un módulo de programas que constituye la interfaz entre los datos de bajo nivel almacenados en la BD y los programas de aplicaciones y las consultas hechas al sistema. El manejador de BD es responsable de las siguientes tareas:

- **Interacción con el manejador de archivos.** Los datos sin procesar se almacenan en el disco mediante el sistema de archivos proporcionado normalmente por un sistema operativo convencional. El manejador de BD traduce las diferentes proposiciones en DML, a comandos de sistema de archivos de bajo nivel. Así, el manejador de BD se encarga realmente del almacenamiento, recuperación y actualización de los datos en la BD.
- **Implantación de la integridad.** Los valores de los datos almacenados en la BD deben satisfacer ciertos tipos de limitantes de consistencias. El administrador de la BD debe especificar estas limitantes en forma explícita. Si se especifican estas limitantes entonces el manejador de la BD puede verificar si las actualizaciones a ésta resultan en la violación de cualquiera de estas limitantes, y si es así se podrá realizar la acción apropiada.
- **Puesta en práctica de la seguridad.** Es preciso que todos los usuarios de la BD tengan acceso a todo su contenido. Es labor del manejador de la BD que éstos tengan acceso a todo su contenido y de que se cumplan estos requisitos de seguridad.
- **Respaldo y recuperación.** Un sistema de cómputo, como cualquier otro dispositivo mecánico o eléctrico, está sujeto a fallas. Es responsabilidad del manejador de la BD detectar estas fallas y restaurarla al estado que existía antes de presentarse la falla.

- **Control de concurrencia.** Cuando varios usuarios actualizan la BD en forma concurrente, es posible que no se conserve la consistencia de los datos. Es necesario que el sistema controle la interacción entre los usuarios concurrentes; lograr dicho control es una de las tareas del manejador de la BD.

Los sistemas administradores de BD relacionales (RDBMS) han sido altamente aceptados por la forma en que se manejan los datos, los sistemas relacionales ofrecen los siguientes beneficios:

- Permiten un sencillo acceso a los datos.
- Flexibilidad en el modelado de los datos.
- Disminuye la redundancia de datos y el nivel de almacenaje.
- Existe independencia entre el almacenaje físico y el diseño lógico de datos.
- Nivel alto en la manipulación de datos.

## **CLIPPER**

Este paquete para el desarrollo de bases de datos es, definitivamente para los programadores. Si bien carece de algunas de las excelentes funciones de generación de código que tienen los demás paquetes Clipper ofrece una riqueza de armas y capacidad que los programadores necesitan.

Dos de sus armas son un generador de reportes (RL) y una función (DBU) para la creación y manejo de los archivos de la BD, escritos en el propio lenguaje de programación Clipper, y se incluye el código fuente, el cual puede utilizarse como referencia, o modificarlo para añadirlo a las aplicaciones de la BD.

La pantalla de DBU enlista las opciones a través de su parte superior, junto con las teclas de funciones asignadas. El resto de la pantalla está dedicado a una

representación visual del panorama de la BD activa, la cual consiste en una columna partida en tres secciones. El primer grupo exhibe el nombre de la BD activa. Si al invocar el programa DBU se invoca un argumento, aparecerá la BD o la vista especificada. En la siguiente sección aparecerán los índices activos relacionados con la BD activa, y el grupo al fondo muestra los nombres de los campos para la BD activa.

El generador de reportes, RL no es tan flexible, no soporta un bosquejo WYSIWYG del reporte y tampoco permite ver la salida en forma preliminar conforme se vaya trabajando. La verdadera potencia de Clipper se basa en su codificación.

El compilador del programa es muy rápido además de tener algunas funciones muy interesantes, dentro de éstas se destacan las siguientes: soporta los llamados bloques de código, pedacitos de código ejecutable que se pueden almacenar como variables, o pasar a otros programas como argumentos, para ejecutar un bloque de código, se utiliza una función EVAL(); otra función útil es el uso hecho por Clipper de los archivos cerrados para el seguimiento, a fin de determinar cuáles archivos en un programa dependen de otros para operar correctamente y quedar al corriente.

Estando instalado este sistema, se puede invocar la función MiMake para llevar a cabo aquellas operaciones de compilación y enlace que se necesiten y mantener todos los archivos sincronizados, Clipper también soporta las funciones para leer los archivos binarios de DOS y escribir en ellos. También se aprecia el depurador de Clipper, el cual permite analizar el funcionamiento del código del programa, ejecutar comandos y revisar el estado de algunas variables, nombres de campos o expresiones en particular. Una ventana de estado enlista las bases de datos abiertas en todas las áreas del trabajo activas, así como los valores de todos los comandos Set.

## **FOXPRO**

La interfaz de FoxPro incluye menús presentados junto con una ventana de comandos conveniente, para utilizar los menús o escribir los comandos sin que

ninguno de los elementos de la interfaz interfieran con otro. Su soporte al ratón es el mejor de todos los paquetes basados en caracteres. Asimismo, la ventana de comandos mantiene una historia corrida de las instrucciones (al igual que Dbase III plus), esto facilita la repetición de los comandos utilizados a través de una sesión. También se puede seleccionar parte de la historia de los comandos, y anexarla a sus aplicaciones.

La ventana presenta gráficamente todas las áreas de trabajo disponible; se puede seleccionar un área de trabajo disponible, y abrir en ella una BD, accionando un botón de comando. FoxPro ofrece una caja de diálogo con una lista de los campos principales.

El constructor de pantalla comienza en blanco, en la cual se puede capturar el texto y colocar los campos, también se pueden crear botones de comando, casillas para marcar con "palomas", botones de "radio" y lista de extraer. Se pueden agregar pedacitos de código a cualquier objeto, incluso a los campos.

Desde la pantalla de establecimiento, se puede ejecutar código antes y después del programa pantalla de captura, cuando se genere el código para la pantalla diseñada, se pueden anexar otras pantallas a ella, lo cual ahorra tiempo una vez formulada una biblioteca de pantallas genéricas.

El funcionamiento de FoxPro es magnífico gracias a su tecnología exclusiva de *Rushmore*, siendo su único inconveniente su voluminosa documentación, ya que es un problema localizar la información rápidamente, lo cual no obstante disminuye gracias a su fuerte función de ayuda en línea.

## **INFORMIX-SQL**

Informix SQL es cien por ciento una BD con lenguaje de consulta estructurado (SQL), el aspecto del programa es muy austero: no se encontrará con pantallas de colores

múltiples con menús y ventanas a la vista. Las pantallas del paquete, estilo Lotus, impulsadas por menús, automatizan las operaciones más significativas de la BD, tales como la creación de tablas, la definición de los campos, etc. Informix también incluye un generador de reportes, y un sistema para la ejecución de archivos de la definición de formas diseñadas.

En informix se construye una forma, no moviendo un cursor sobre la pantalla con un ratón o teclas de fecha, sino escribiendo en la forma, un tipo de definición de pantalla acompañada con instrucciones ejecutables.

Dicho archivo comprende cinco partes: una sección de tablas serán accedidas por la que describe cada campo exhibido por la forma, y una sección opcional de instrucciones que define las operaciones que habrán de ser llevadas a cabo sobre los campos dentro de la forma.

Las secciones y tablas informan al sistema qué debe presentar, la sección de pantalla indica en donde presentarlo, la sección de atribuciones indica cómo presentarlo y la sección de instrucciones le dice al sistema qué hace antes, mientras y después de presentarlo.

Se incluye con Informix-SQL varios paquetes de servicios: BECHECK verifica la integridad de los índices, si encuentra una discrepancia entre un archivo de datos y uno de sus índices, le permite reformar el índice: DRLINK y DBLOAD son de utilidad para el traslado de los datos entre Informix y el mundo exterior de archivos de Lotus 1-2-3, Dbase, o ASCII: con DBSCHEMA, se pueden elaborar las instrucciones de SQL requeridas para crear una tabla o una BD.

## **PARADOX**

Una de las características más interesantes de PARADOX es su velocidad, destacándose su rápido tiempo de respuesta para la lectura y edición de las tablas.

Paradox soporta los formatos de importación y exportación más importantes, pero existen problemas en la importación almacenada en el formato ASCII. Con Paradox, es necesario planear con cuidado antes de precipitarse a la construcción de una forma, aplicando esto principalmente a las formas que acceden a múltiples tablas. El diseño de formas con tablas múltiples exige la creación de una forma maestra, después existe también la introducción, en dicha forma maestra de las formas incluidas en las otras tablas. Así que para crear la forma maestra completa, deben diseñarse las formas que se incluirán primero.

El lenguaje para el manejo de la BD PARADOX es PAL, aunque es erróneo representar a PAL como simplemente un DML. Lo que dificulta la programación con PAL es que maneja la transmisión de comandos como si el programador fuera un robot sentado en un teclado operando PARADOX; sin embargo, si simplemente no se quiere tener nada que ver con PAL, podrá instalarse el programador personal. Este programa es esencialmente un constructor de aplicaciones que a través de la creación de una aplicación completa con menús y formas ofreciendo el código PAL como su producto final.

## **ACCESS**

Access ha llegado a ser uno de los más populares administradores de bases de datos relacional lanzados en el medio informático, debido a todas las posibilidades que integra. Es el producto más escalable en su clase, implanta un impresionante grupo de características nuevas y mejoras a la facilidad de uso, incluyendo varios *Wizards* nuevos. Los *Wizard* son módulos orientados a ciertas tareas, que hacen una serie de preguntas, luego generan los objetos apropiados, según las respuestas. También posee un lenguaje de programación *ObjectPAL*.

Tiene el mejor rendimiento de búsqueda, impone reglas de integridad de datos a nivel de BD que le permiten determinar cómo los registros en tablas asociadas se ven

afectados cuando se modifica una matriz, reduciendo así la cantidad de código necesario para construir aplicaciones.

Cuando se crea una BD en Access todos sus componentes se guardan en un archivo único con extensión MDB. Pulsando sobre el botón New se comienza el proceso de crear un objeto del tipo seleccionado y es ahí donde los *Wizard* entran en acción, facilitando la creación de diseños funcionales. De los asistentes más potentes está el *Table Wizard*, que provee más de cuarenta modelos de tablas personales y de negocios, cada una con sus propios juegos de campos con formato para usar en su propia BD.

Los *Wizard* de botones de comandos frecuentemente determinan de antemano las respuestas de los eventos del ratón. Contiene características que permiten generar rápidamente reportes y formularios.

Tiene una ventana gráfica de Relationship que muestra todas las tablas en la BD con líneas entre ellas para representar las relaciones activas. Simplemente se arrastra de un campo en una tabla a otro en otra tabla para definir una relación y luego se personaliza el enlace.

Puede usar datos de varias bases de datos como: dBase III y IV, Foxpro 2.0 y 2.5, Paradox 3.x y 4.x así como Microsoft, Oracle, Sybase y el lenguaje SQL.

Access es una aplicación que se desarrolló bajo Windows, de esta forma facilita construir tareas rápidas por medio de ventanas y sus elementos.

Access nos ayuda a gestionar nuestra BD proporcionando una infraestructura eficiente para almacenar y recuperar la información, además genera automáticamente un formulario para complementar todos nuestros datos. Después de que hayamos

introducido algunos datos, podemos pedir a Access que visualice una parte seleccionada de información utilizando los procedimientos de: búsqueda, ordenación y consulta. También nos ayuda a imprimir solamente la parte de la información que deseamos ver en un informe. En cualquier momento, podemos personalizar la apariencia o editar el contenido de cualquier parte de nuestra BD.

### **Características de Access**

- Cambiar el nombre de la columna para que se ajuste mejor.
- Cortar el texto fácilmente utilizando un menú abreviado.
- Crear automáticamente una nueva tabla con el *asistente para tablas*.
- Filtrar una tabla para presentar solamente los registros que deseamos ver.
- Exportar solo de una tabla a una hoja de cálculo de Excel.
- Esquematizar el diseño de nuestra BD o ver todas las relaciones a la vez *utilizando la ventana de relaciones*.
- Establecer la propiedad de un campo fácilmente utilizando el botón general que nos ayuda a seleccionar las opciones.
- Ajustar la posición de los controles de un formulario desplazando un solo punto en la cuadrícula en un momento.
- Pero su mayor desventaja es que no puede crear archivos ejecutables (hasta el momento).

### **Elección de un Administrador de Base de Datos para PC`s**

Uno de los aspectos importantes de los administradores de bases de datos más recientes es que logran que entre la característica de poder que ofrecen la mayoría de los fabricantes, se cuenta con la posibilidad integrada de enlace en redes, la posibilidad de colocar un programa nuevo de software en la red, para ofrecer acceso compartido a datos y lenguajes de programación que se puedan utilizar para desarrollar aplicaciones adecuadas a las necesidades de cada uno de los usuarios, incluyendo menús y reportes y que ese poder sea más accesible con una serie de menús, editores de toda la pantalla y características automáticas que vuelvan

relativamente sencilla la realización de operaciones relacionales. Productos pioneros como Dbase por ejemplo, eran predominantemente controlados por líneas de comandos. Para aprovechar este software se tenía que aprender un gran número de comandos y procedimientos. Esto ya no sucede con los productos nuevos, tal es el caso de Access. La demanda del usuario de acceso a BD ha apresurado a los productores de software a diseñar interfaces realistas. Recientes adelantos técnicos (implantación de gráficos en Bases de Datos), como se mencionó anteriormente con Access se puede trabajar con la implementación de los avances OLE (Object Link Embedd) en versión 2.0, incorpora datos de Word 6.0, Excel 5.0, Power Point 3.0 y Microsoft Mail. Access es una aplicación que se desarrolló bajo Windows, de esta forma facilita construir tareas rápidas por medio de ventanas y sus elementos.

Para poder elegir la BD que se acerca a nuestras necesidades, fue necesario realizar un criterio de selección, el cual se muestra en las Tablas 2.5.4.1.1, 2.5.4.1.2, 2.5.4.1.3, 2.5.4.1.4, 2.5.4.1.5, 2.5.4.1.6, 2.5.4.1.7, 2.5.4.1.8, 2.5.4.1.9, 2.5.4.1.10, 2.5.4.1.11 y 2.5.4.1.12.

| PRODUCTO     | COMPANIA              |
|--------------|-----------------------|
| Clipper      | Nantucket             |
| FoxPro       | Fox Software, Inc.    |
| Informix-SQL | Informix              |
| Paradox      | Borland International |
| Access       | Microsoft Corp.       |

Para fines prácticos, se abreviarán con las siguientes letras a los productos

Clipper = C  
 FoxPro = F  
 Informix-SQL = I  
 Paradox = P  
 Access = A

| Tiempo de<br>Procesamiento de Batch | C | F | I | P | A |
|-------------------------------------|---|---|---|---|---|
| Bueno                               | X |   |   |   |   |
| Regular                             |   | X |   | X | X |
| Malo                                |   |   | X |   |   |

Tabla 2.5.4.1.1. Evaluación de tiempo de procesamiento en batch.

| Tiempo Importación y Generación de Índices | C | F | I | P | A |
|--|---|---|---|---|---|
| Bueno                                      |   | X |   | X |   |
| Regular                                    | X |   |   |   | X |
| Malo                                       |   |   | X |   |   |

Tabla 2.5.4.1.2. Evaluación de tiempo importación y generación de índices.

| Soporte de Núm. Máximos de archivos abiertos | C | F | I | P | A |
|--|---|---|---|---|---|
| Bueno  |   |   | X |   |   |
| Regular                                      | X | X |   | X | X |
| Malo   |   |   |   |   |   |

Tabla 2.5.4.1.3. Evaluación de soporte de núm. máximo de archivos abiertos.

| Facilidad de Programación | C | F | I | P | A |
|---------------------------|---|---|---|---|---|
| Bueno                     |   |   |   |   | X |
| Regular                   | X | X |   | X |   |
| Malo                      |   |   | X |   |   |

Tabla 2.5.4.1.8. Evaluación en la facilidad de programación.

| Compartición de los datos con otras bases de datos | C | F | I | P | A |
|--|---|---|---|---|---|
| Bueno  |   |   |   |   | X |
| Regular  | X | X | X | X |   |
| Malo   |   |   |   |   |   |

Tabla 2.5.4.1.9. Evaluación en la compartición de los datos con otras base de datos.

| Diversidad en Tipos de Datos | C | F | I | P | A |
|------------------------------|---|---|---|---|---|
| Bueno                        |   |   |   |   | X |
| Regular                      | X | X |   | X |   |
| Malo                         |   |   | X |   |   |

Tabla 2.5.4.1.10. Evaluación en la diversidad en tipos de datos.

| Manejo de Gráficos para Formularios | C | F | I | P | A |
|-------------------------------------|---|---|---|---|---|
| Bueno                               |   |   |   |   |   |
| Regular                             |   |   |   |   | X |
| Malo                                | X | X | X | X |   |

Tabla 2.5.4.1.11. Evaluación en el manejo de gráficos para formularios.

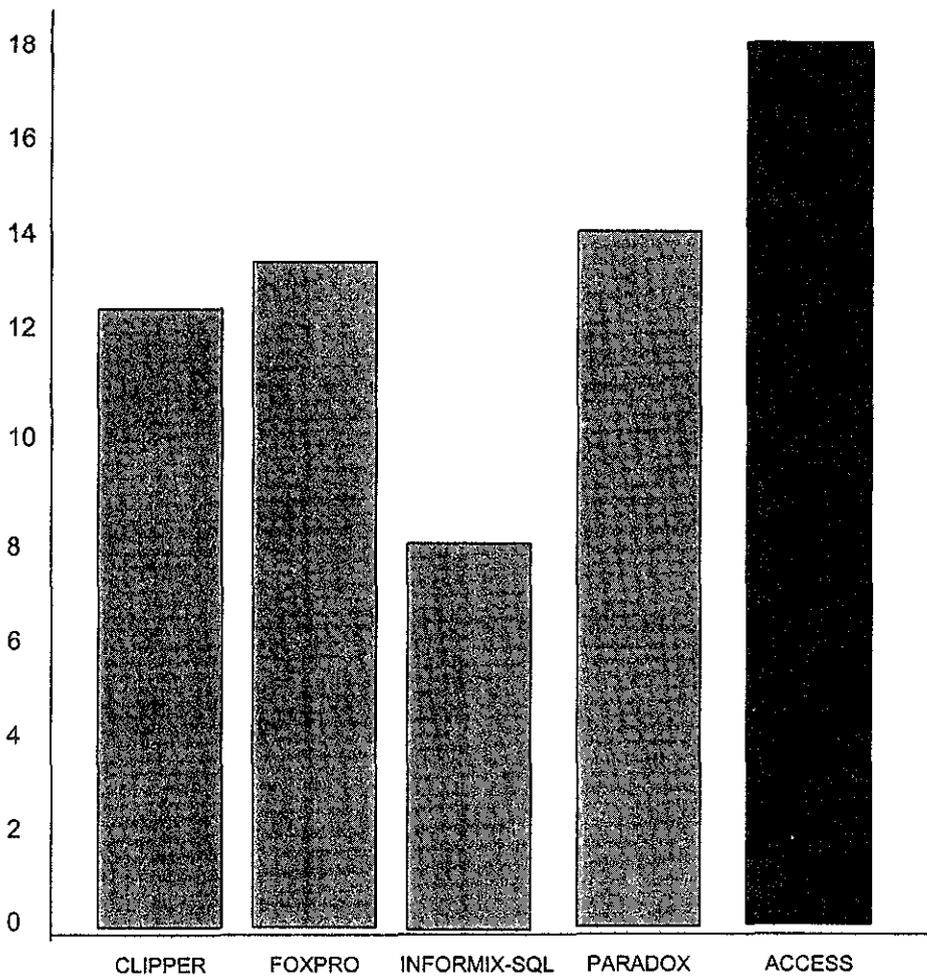
Para establecer la puntuación que obtuvo cada producto se determinó un valor para cada categoría determinada de la siguiente manera:

bueno = 2 puntos  
 regular = 1 punto  
 malo = 0 puntos

Obteniendo finalmente la siguiente tabla de puntuaciones: tabla 2.5.4.1.12

| Características                                  | C  | F  | I | P  | A  |
|--|----|----|---|----|----|
| Tiempo de procesamiento en Batch                 | 2  | 1  | 0 | 1  | 1  |
| Tiempo de importación y generación de índices    | 1  | 2  | 0 | 2  | 1  |
| Soporte de Num. Máximo de archivos abiertos      | 1  | 1  | 2 | 1  | 1  |
| Facilidad de Uso                                 | 1  | 2  | 1 | 2  | 2  |
| Proporciona Seguridad                            | 1  | 2  | 2 | 2  | 2  |
| Integridad en los datos                          | 2  | 1  | 1 | 2  | 2  |
| Presentación de Calidad en los Reportes y Formas | 1  | 1  | 1 | 1  | 2  |
| Facilidad de Programación                        | 1  | 1  | 0 | 1  | 2  |
| Compartición de Datos con otras bases de datos   | 1  | 1  | 1 | 1  | 2  |
| Diversidad de tipos de datos                     | 1  | 1  | 0 | 1  | 2  |
| Manejo de Gráficos para formularios              | 0  | 0  | 0 | 0  | 1  |
| Total  | 12 | 13 | 8 | 14 | 18 |

**Tabla 2.5.4.1.12. Evaluación para la puntuación final.**



**Tabla 2.5.4.1.13. Puntuación de la comparación.**

La elección de la BD se basó, además de todo lo anterior, en la disponibilidad comercial. Por lo antes mencionado y en relación con las capacidades de manejo, programación y de conocimiento, entre otras, asociadas, se terminó el utilizar Microsoft Access para el desarrollo del sistema.

## **2.5.4.2. Características de Herramientas Visuales Disponibles para el Front-End y Elección de la Óptima**

El front-end de un sistema es un programa de computadora que oculta detalles del acceso a datos. En un sentido, todo programa es un front-end que evita que los usuarios conozcan los detalles reales del intrincado funcionamiento de la computadora.

En otras palabras, un front-end es un procesador de formatos, es decir, un procesador de entrada/salida utilizado para realizar el formato de los datos de entrada y/o procesarlos. A veces se utiliza el término para referirse a un programa de comunicación.

### **Programación Visual**

El éxito de Windows en el entorno PC ha llevado a los programadores a desarrollar aplicaciones que respeten el "look", y las necesidades de esta interfaz gráfica. La llamada "programación visual" supone un paso más en la búsqueda de nuevas herramientas que faciliten el desarrollo de software para Windows.

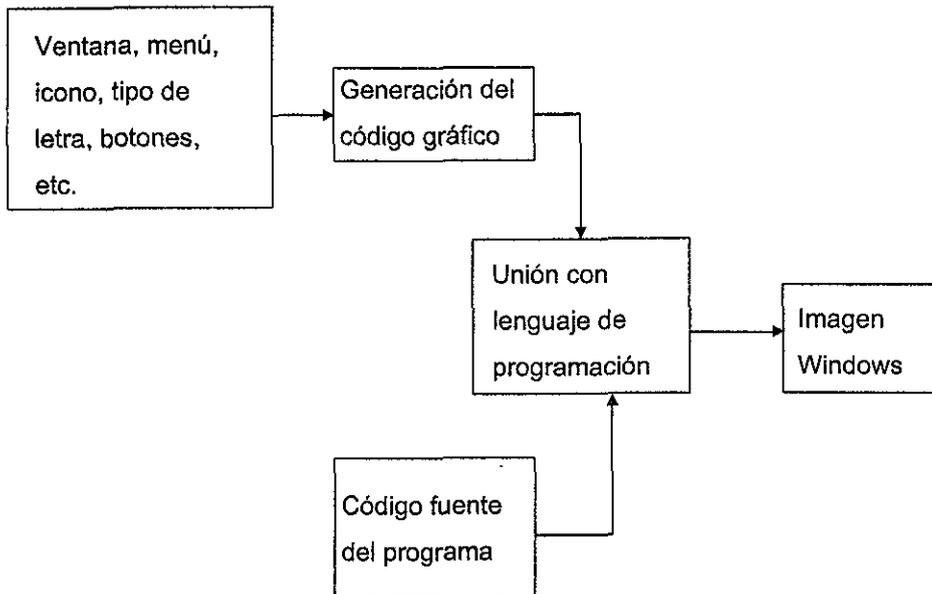
Empresas como Microsoft y Borland se han apresurado a lanzar al mercado versiones de sus productos con la palabra "Visual". Se puede decir que "Programación Visual" se entiende como un procesador de desarrollo interactivo de la interfaz de usuario para sus aplicaciones. La idea es que la "Programación Visual" permite que la mayor parte del proceso necesario para crear cualquier aplicación se pueda utilizar en forma "automática".

Así por ejemplo, como la mayoría de las aplicaciones de Windows disponen de menús desplegables, parece razonable desarrollar una herramienta para crearlos sin tener que recurrir a páginas de código indescifrable. Pero los beneficios no terminan aquí, debido a que lo mencionado anteriormente ya se encontraba disponible en los

editores de recursos, sin embargo actualmente se cuenta con mayor integración entre el desarrollo de aplicaciones y el diseño de la interfaz de usuario.

Los editores de recursos que pueden encontrarse en el mercado (en los compiladores de C para Windows siempre se incluye alguno) siguen las siguientes pautas:

Primero, se crea el recurso deseado que puede ser una ventana, un menú un icono, algún tipo de letra, un botón, una caja de texto o cualquier otro elemento típico de Windows, después se genera el código correspondiente al recurso y, por último con más o menos pasos intermedios, se acaba uniéndolo al resto del código fuente de un programa como se puede apreciar en la figura 2.5.4.2.1.



**Figura 2.5.4.2.1. Proceso para realizar una aplicación con ventanas.**

Con la programación utilizando herramientas visuales, se pretende simplificar al máximo todo el proceso, evitando cuantos pasos intermedios sean posibles. Lo más común es que se parta de la creación de una interfaz de usuario y se llegue a la escritura del código (de ser necesario) en algún lenguaje de presentación.

Por el momento, con la llegada de herramientas visuales para el desarrollo de aplicaciones, se ha dado un enorme paso para facilitar la vida de los desarrolladores; sin embargo, todavía no ha llegado el día en el que cualquier persona sea capaz de decirle a una computadora a través de un micrófono lo que quiere y que ésta lo haga. Hasta entonces, tendremos que seguir trabajando con las herramientas disponibles, muchas de las cuales ya han comenzado su transición a la "Programación Visual".

Desde que la informática abandonó el uso de tarjetas perforadas, la mayor preocupación de los programadores ha sido crear aplicaciones que "entren por los ojos"; es decir que no sólo fueran fáciles de usar, sino también atractivos. En pocos años hemos vivido una verdadera revolución en la manera de entender las aplicaciones, esta revolución ha venido acompañada de nuevas y poderosas PC's, así como de sistemas operativos totalmente gráficos (Windows NT) y de periféricos tales como modem, impresora láser, escáner, etc.

Además, la proliferación de las interfaces gráficas de usuario -los llamados ambientes de ventanas o GUI- no ha hecho más que acentuar lo que existía entre las técnicas de programación de los años ochenta y las actuales. Muchos han sido los desarrolladores que no han sabido adaptarse a los nuevos tiempos y se han quedado obsoletos, sin embargo, aquellos que han actualizado sus conocimientos reconocen que ahora cuentan con herramientas mejores y más poderosas para realizar su trabajo.

Al mismo tiempo, las herramientas de programación han evolucionado hasta convertirse en verdaderos ambientes de desarrollo integrados que se basan en la misma filosofía de sencillez que gobiernan las interfaces gráficas. En el campo de la

integración de herramientas, Borland fue pionera con su Turbo Pascal, cuyas primeras versiones aparecieron para el sistema operativo CPM de la familia de microprocesadores Z80 de Zilog. Por otra parte, Microsoft también ha sido pionera con el concepto de "Programación Visual" a través de su herramienta Visual Basic desde su versión para DOS, y que actualmente con sus versiones para Windows se ha convertido en un best seller.

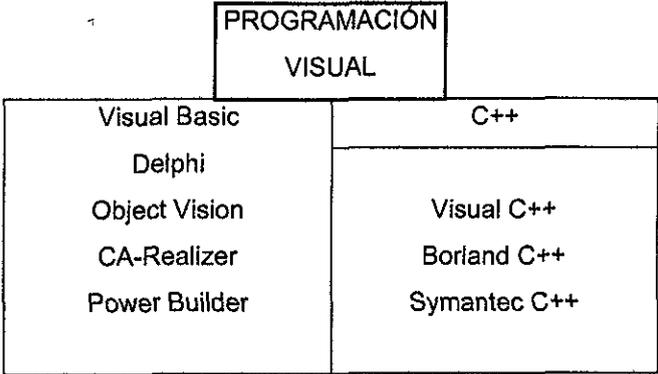


Figura 2.5.4.2.2. Paquetes y lenguajes que manejan la programación visual.

**Definición**

Anteriormente se mencionó que el término en sí era demasiado genérico y que procedía principalmente del nombre que las propias casas de software daban a sus **IDE** (Integrated Development Enviroment. Ambientes de Desarrollo Integrado) para lanzarlos publicitariamente con la idea de ser algo intuitivo y fácil de usar.

Sin embargo, con la aparición de los últimos productos de desarrollo de aplicaciones bajo ambientes gráficos se va haciendo necesaria una terminología adecuada que los clasifique de alguna manera.

Un ambiente integrado de desarrollo es aquel que permite desde un único programa realizar las tareas más habituales de desarrollo y depuración de una aplicación (incluido en algunos casos la optimización y realización automática de pruebas).

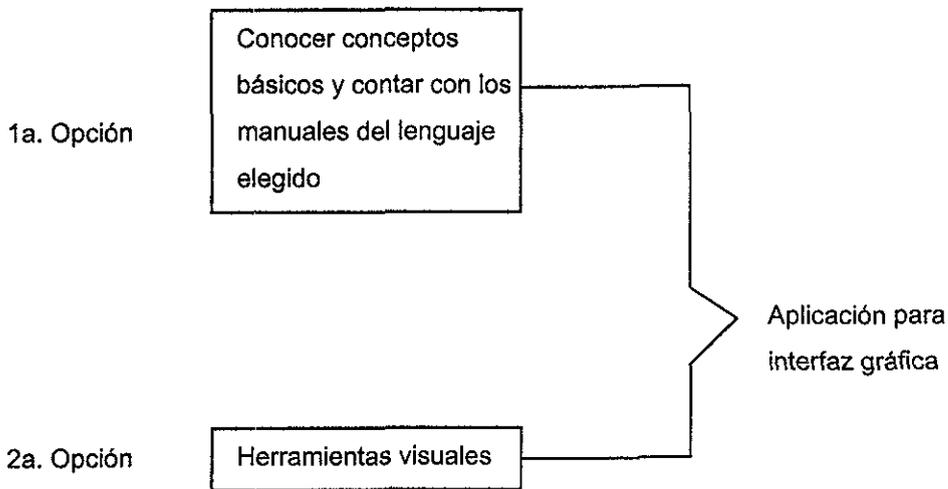
Si el ambiente integrado es de "Programación Visual", se habla de un producto que integra una serie de herramientas adicionales a las habituales y que son necesarias para el desarrollo de programas bajo una interfaz de usuario gráfica, todo ello con la mínima intervención del programador en las tareas más repetitivas y por tanto, más automatizadas.

Se puede decir que la "Programación Visual" se sitúa directamente por debajo de las llamadas herramientas CASE (Computer Aided Software Engineering). Los programas CASE permiten pasar directamente del diseño de una aplicación (diagramas de flujo, estructuras de datos) a producir la aplicación propiamente dicha.

### **Programar la Interfaz**

Con estas premisas, está claro que la "Programación Visual" ha pasado de ser una curiosa novedad meteorológica, a construirse como un conjunto de herramientas imprescindibles para el desarrollo de las complejas aplicaciones basadas en GUI's que se comercializan actualmente.

La complejidad de programar bajo una interfaz gráfica proviene del propio diseño de los ambientes. De hecho, desarrollar el programa más sencillo para un ambiente gráfico puede convertirse en una tarea desquiciante si no se dispone de las herramientas adecuadas.



**Figura 2.5.4.2.3. Desarrollo de una aplicación para una interfaz gráfica.**

Para afrontar el desarrollo de una aplicación bajo una interfaz gráfica (Windows u OS/2 en el mundo PC) pueden seguirse dos caminos. El primero consiste en enfrentarse con los conceptos básicos del ambiente, lo cual resulta una tarea tediosa y poco gratificante ya que a veces no es efectiva ante las funcionalidades ya existentes (OLE, Object Linking Embedding) o los nuevos sistemas operativos (Windows 95). El segundo camino se simplifica al utilizar las mencionadas herramientas visuales de desarrollo.

Estos compiladores simplifican el diseño al proporcionarnos formatos, tipos o patrones que pueden aplicarse a los desarrollos propios, al mismo tiempo que ofrecen una elevada confiabilidad debido a que se tratan de patrones probados en muchos otros desarrollos. Todo ello redundará en una mayor eficiencia y un gran ahorro de tiempo de codificación.

## VISUAL BASIC

Tomando como referencia el punto 1.5 "Características, ventajas y desventajas de Visual Basic 4.0":

### Ventajas

- Es un sistema productivo para crear soluciones bajo ambiente Windows.
- Posibilidad de incorporar controles preconstruidos por terceros (Custom Control).
- Acceso a bases de datos (Access, SQL Server, Oracle, FoxPro, etc.).
- Manejo de Multimedia.
- Permite ensamblar fácil y rápidamente una interfaz de usuario con componentes prefabricados.
- Ofrece una gran capacidad y velocidad en su Debugger.
- Permite manipular otras aplicaciones para utilizarlas como componentes en aplicaciones propias (Word, Excel, Project, etc.), siempre y cuando dichas aplicaciones soporten OLE Automation.
- Interfaz amigable al usuario.
- Manejo de ayudas en líneas robustas.
- Intérprete que permite la detección de errores al momento de editar el programa.
- Maneja aplicaciones Cliente – Servidor.
- Incluye motor de BD vía DAO (*Data Access Object*).
- Un usuario inexperto puede ser productivo con algunas semanas de utilización de Visual Basic.
- Puede ser utilizado como un servidor de OLE Automation local o remoto.
- Módulos de clases. Objetos y colecciones reutilizables.
- Procedimientos para propiedades.
- Compilación de aplicaciones en formato DLL para la reutilización de componentes.

- Aunque Visual Basic no cuenta con herencia, esto hace que los componentes reutilizados sean más fáciles de comercializar.
- Creación de Addins. Son servidores OLE que pueden integrarse al ambiente de desarrollo de Visual Basic para facilitar al programador la elaboración de aplicaciones.

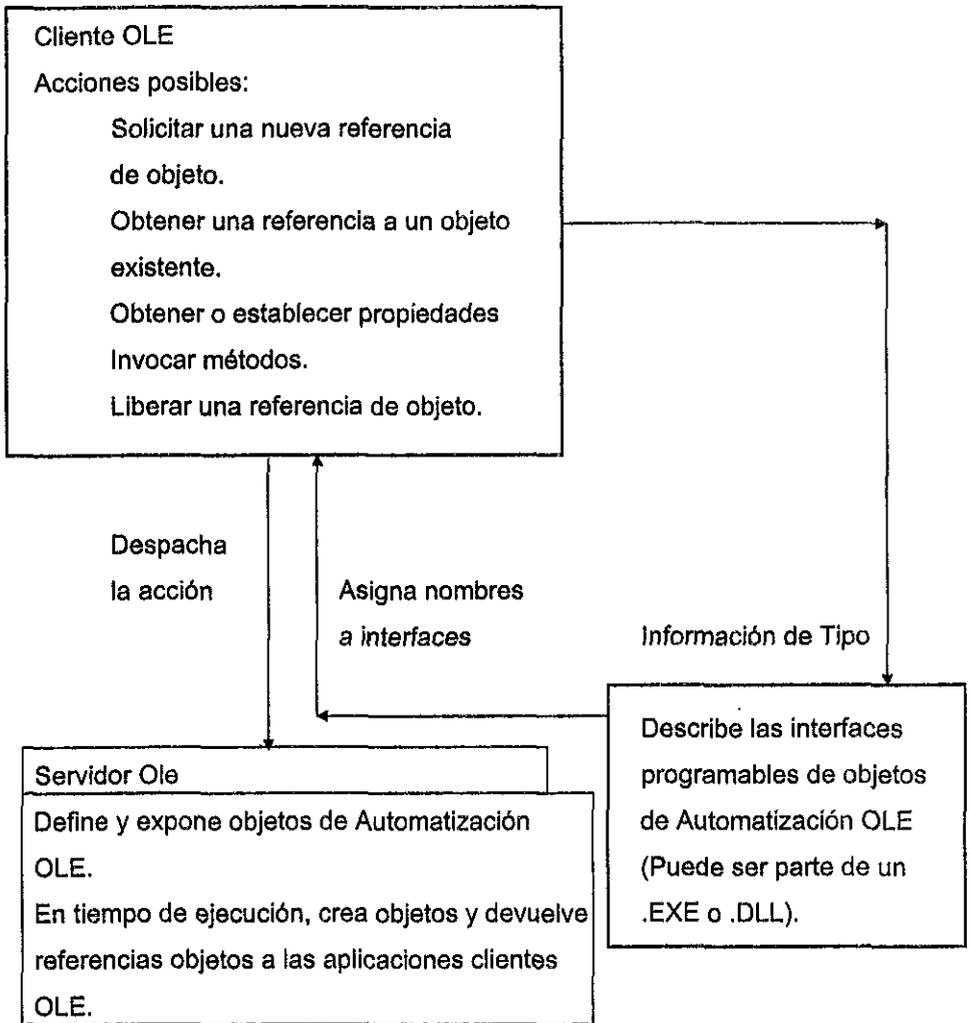
### **Desventajas**

- No cuenta con herencia, característica muy importante de la programación orientada a objetos.
- Por ser código interpretado, los ejecutables producidos por Visual Basic resultan lentos en su desempeño.
- Su código no es compatible con plataformas no Windows.
- La utilización de componentes remotos es muy lenta.

### **DELPHI**

Delphi es una herramienta para construir aplicaciones que utiliza el lenguaje de programación *Object Pascal* de Borland -sin lugar a dudas se le pudo haber llamado Visual Pascal.

Realizar una aplicación en Delphi es como crear una aplicación en Visual Basic, uno crea las formas y selecciona componentes de una caja de herramientas, para ponerlas dentro de las formas y se procede a programar los eventos necesarios para manipular las entradas del usuario.



**Figura 2.5.4.2.4. Los clientes OLE usan los objetos de automatización OLE de los servidores OLE.**

Por otra parte, Delphi es una aplicación basada en componentes bajo un ambiente de desarrollo que da soporte a una implementación rápida y eficiente de programas basados en Windows con un mínimo de código.

Muchos de los requerimientos tradicionales de la programación para Windows son manejados dentro de una librería de clases de Delphi, eliminando los problemas de tareas repetitivas de programación.

Delphi proporciona herramientas de diseño tales como formas para aplicaciones, de tal manera que se pueden crear y probar los prototipos de la aplicación.

Mediante el uso de una gran cantidad de componentes y la generación intuitiva de código, las aplicaciones se pueden modificar generando programas completos y propios del usuario.

Las herramientas de bases de datos de Delphi permiten el desarrollo de aplicaciones Cliente-Servidor. Se puede ver la información en tiempo real de tal manera que se conocen los resultados de las consultas inmediatamente.

### **Ventajas**

- Presenta programación estructurada y programación orientada a objetos.
- Requiere de una sintaxis precisa.
- Su compilador realiza su programa ejecutable automáticamente.
- Tiene edición multiarchivos.
- Su curva de aprendizaje es de nivel medio.
- Provee encapsulamiento para proyectos grandes o en grupo, lo cual significa que se tiene control de acceso a variables de una manera más sencilla que el esquema procedimiento - módulo - global utilizado en Visual Basic.
- Presenta un módulo separado de encapsulamiento donde las declaraciones en un archivo pueden o no formar parte de la interfaz.

## **Desventajas**

- Es una herramienta poco difundida.
- Su límite de cadenas es de 255 caracteres, sin embargo existen cadenas terminación cero que pueden ser de 65,535 caracteres de extensión, pero las operaciones que se pueden realizar con ellas no son tan poderosas como las de Visual Basic.
- El límite de 255 caracteres también se aplica a algunas propiedades de control.
- No existen arreglos re-dimensionables. Sin embargo, Delphi tiene colecciones que pueden cambiarse de tamaño, las cuales son más poderosas que los arreglos.

## **VISUAL en C**

Hasta hace poco los únicos que podían disfrutar de este tipo de herramientas en su trabajo eran los programadores de Visual Basic de Microsoft y Turbo Pascal para Windows de Borland. Sin embargo, sucede que la mayoría de los programadores profesionales prefieren el lenguaje C antes que Basic o Pascal.

Se puede decir sin temor a equivocarse que existen compiladores de C para todas las combinaciones posibles de microprocesadores, arquitecturas de hardware y sistemas operativos.

Si el C++ ha sido una evolución natural de C hacia la programación moderna, lo que aquí se llama "programación visual" podría considerarse como la evolución de los ambientes de programación integrados para el lenguaje C, que hasta hace relativamente poco tiempo no había abandonado el modo texto.

Que quede claro que no se está hablando de una innovación en el campo del lenguaje propiamente dicho, sino de una inevitable evolución hacia formas más cómodas y a la vez más seguras de programar, gracias a la facilidad de uso de las interfaces gráficas.

## **MICROSOFT VISUAL C++**

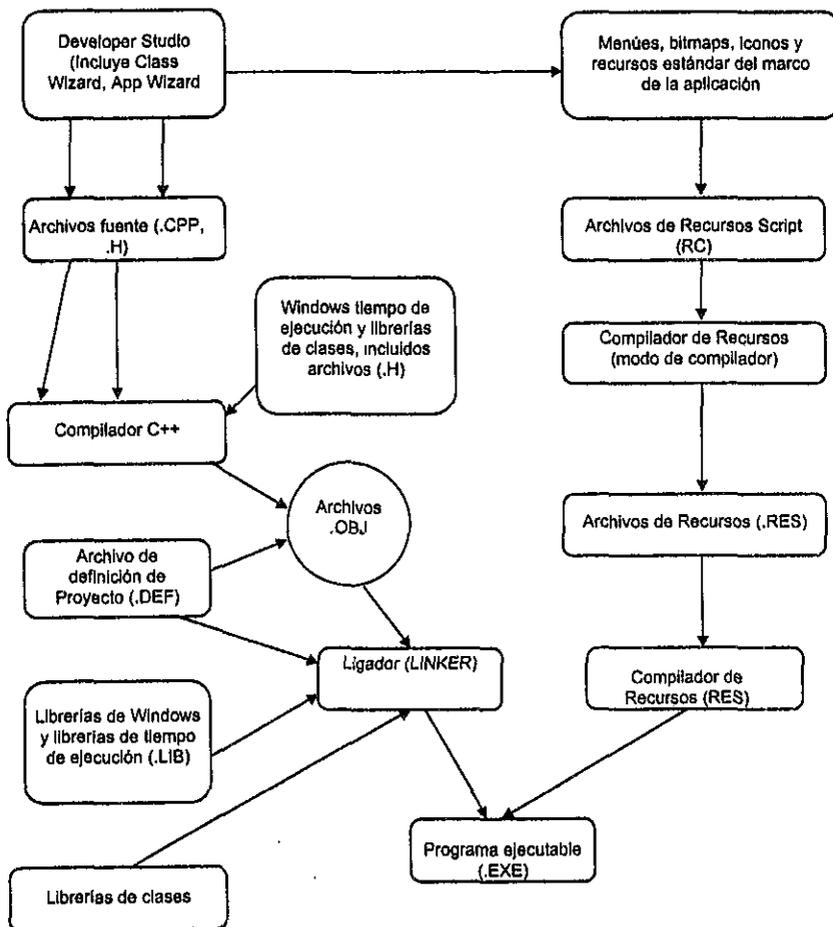
Aunque el lenguaje favorito de Microsoft es el Visual Basic para aplicaciones, son muchos los programadores que por muchos motivos prefieren el C o Pascal para sus aplicaciones. Para estos usuarios, existe una herramienta muy poderosa: Microsoft Visual C++.

Hasta hace unos años, Microsoft no había adoptado la filosofía de los ambientes integrados de programación en sus productos. Sus compiladores se invocaban desde la línea de comandos, teniendo que incluir multitud de parámetros para conseguir los resultados deseados. Sin embargo, su entrada a este campo ha sido espectacular. El nacimiento de Windows supuso un paso decisivo en general hacia el desarrollo de herramientas integradas e intuitivas, pero el más beneficiado en este aspecto ha sido Microsoft. Casi nadie esperaba que alcanzaría a Borland - pionero en ambientes bajo DOS- en cuanto a potencia y facilidad de uso en sus ambientes integrados de programación.

Visual C++ supone la competencia que Microsoft opone a Borland en este lenguaje que cuenta con tantos adeptos entre los programadores. Se supone que el inventor de Windows debe disponer de los mejores lenguajes de programación para las aplicaciones de este ambiente.

Una gran diferencia entre los programas elaborados en C a los programas elaborados bajo Windows C++, es que los programas en C llaman al sistema operativo para utilizar una salida, en cambio Windows utiliza las salidas a través de mensajes. En la figura 2.5.4.2.5, se presenta un diagrama del proceso de creación de una aplicación en Visual C++. Los programas elaborados en C son escritos directamente a la memoria de video y al puerto de impresión, la desventaja de esta técnica es que se necesita un driver o manejador de software para los periféricos.

Windows y C++ introdujeron un nivel de abstracción llamado la interfaz de dispositivos gráficos (GDI), para que así el programa no necesite conocer el tipo de tarjeta de video y el tipo de dispositivo de impresión que se tiene. En lugar de localizar el hardware, el programa llama a la función GDI la cual hace referencia a una estructura de datos llamada dispositivo de contexto (DC), Windows convierte el DC a un dispositivo físico y establece las instrucciones apropiadas de entrada/salida. El GDI es tan rápido como el acceso directo a vídeo y permite el uso a diferentes aplicaciones mientras se escribe.



**Figura 2.5.4.2.5. Proceso de creación de una aplicación Visual C++.**

Windows junto con C++ ofrece un manejador de memoria principal, resultando así que la memoria ya no es un problema. Simplemente se coloca la memoria que se necesita y Windows se encarga de lo demás; ejecuta el programa, administra los recursos, automáticamente lo cambia a disco y después lo coloca en memoria física, y los cambios son tan buenos que la computadora tendrá mucha memoria disponible.

| MICROSOFT VISUAL C++ |   |
|----------------------|---|
| VERSIÓN              | V4.0  |
| REQUERIMIENTOS       | <p>En IBM PC compatible, procesador Intel 486 o mayor</p> <p>100 Mb de espacio en HD.</p> <p>Mouse.</p> <p>Sistema operativo Microsoft Windows 95, NT 3.51 o mayor (32-bit).</p> <p>16 Mb de RAM para trabajar adecuadamente.</p> |
| COMENTARIOS          | <p>Requiere experiencia en la programación para Windows, pero para el programador profesional, su juego de herramientas visuales facilita la programación.</p>  |

**Figura 2.5.4.2.6. Ficha técnica de Visual C++.**

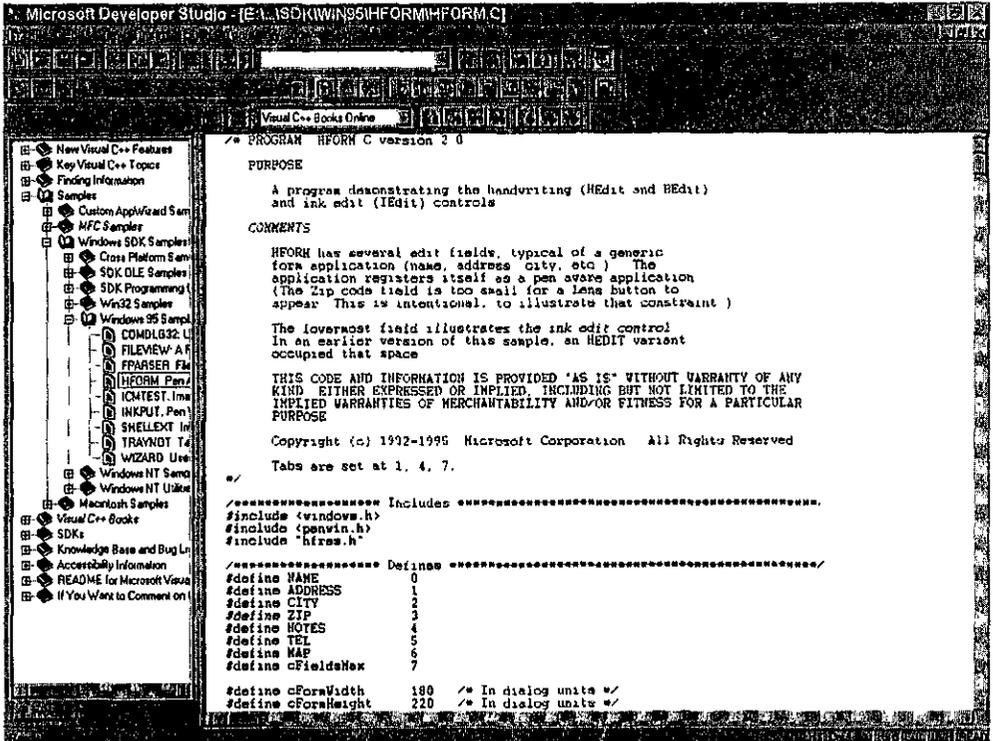


Figura 2.5.4.2.7. Interfaz de programación para Visual C++ 4.0.

## Ventajas

- Incluye librerías preconstruidas.
- *Controles visuales.*
- Reutilización de código total.
- Manejo completo del esquema de la programación orientada a objetos (herencia, polimorfismo, etc.).
- Todo el manejo de dispositivos de entrada/salida se hacen a través de librerías.
- Cuenta con un compilador, no con un intérprete.
- A través de él se pueden crear clientes de OLE.

- Es el único lenguaje que al salir una nueva versión de Windows, ya cuenta con las herramientas necesarias para trabajar con él.
- Manejo de multitarea.
- Rapidez en ejecución de programas.
- Permite obtener funcionalidades de otras aplicaciones para utilizarlas como componentes en aplicaciones propias, por ejemplo Word, Excel, Project, Visual Basic 4.0, etc.
- Manejo de OLE Automation. Permite la creación de servidores OLE.
- Total libertad para manejo de memoria y accesos al sistema operativo.
- Utilización de librerías MFC (Microsoft Foundation Classes) para el desarrollo de aplicaciones bajo plataforma Windows.

### **Desventajas**

- Curva de aprendizaje alta.
- Tiempos de compilación largos.
- Si se utiliza MFC se pierde la compatibilidad de código entre plataformas.

### **Selección del front-end**

En cuanto a la selección del software para la generación del front-end del sistema, se utilizará Visual Basic debido a todas las características que observamos y también aquellas que se mencionaron anteriormente tales como:

- Su curva de aprendizaje es baja.
- Su flexibilidad y velocidad son adecuadas.
- Su productividad es alta.
- Es altamente interactivo.
- La programación es accesible.
- Proporciona una facilidad en el uso propio del sistema.

## 2.5.5. Diagrama de Funcionalidad

El sistema llevará el control de usuarios, clientes, productos y servicios en una BD en Microsoft Access, que registrará la información generada y permitirá actualizar los movimientos realizados durante los procesos de operación. La interfaz con el usuario se realizará mediante un front-end gráfico en Visual Basic, como se muestra en la figura 2.5.5.1.

Las funciones sobre la BD así como las del sistema en general están controladas desde el front-end gráfico que es desde donde el usuario y el administrador interactúan con el sistema. Por lo expuesto, nos concretaremos a explicar las pantallas del mismo.

**Pantalla de Acceso.** Tiene como función el permitir el acceso sólo a usuarios registrados, si en tres ocasiones el usuario falla en su intento de ingresar al sistema, el sistema abortará su ingreso.

**Pantalla Principal.** Consta de 6 controles generales, las cuales tienen previsto cualquier proceso que el usuario desee realizar sobre el sistema:

- Importación de archivos.
- Exportación de archivos.
- Catálogos del sistema.
- Impresión de reportes.
- Ayuda general del sistema.
- Salida del sistema.

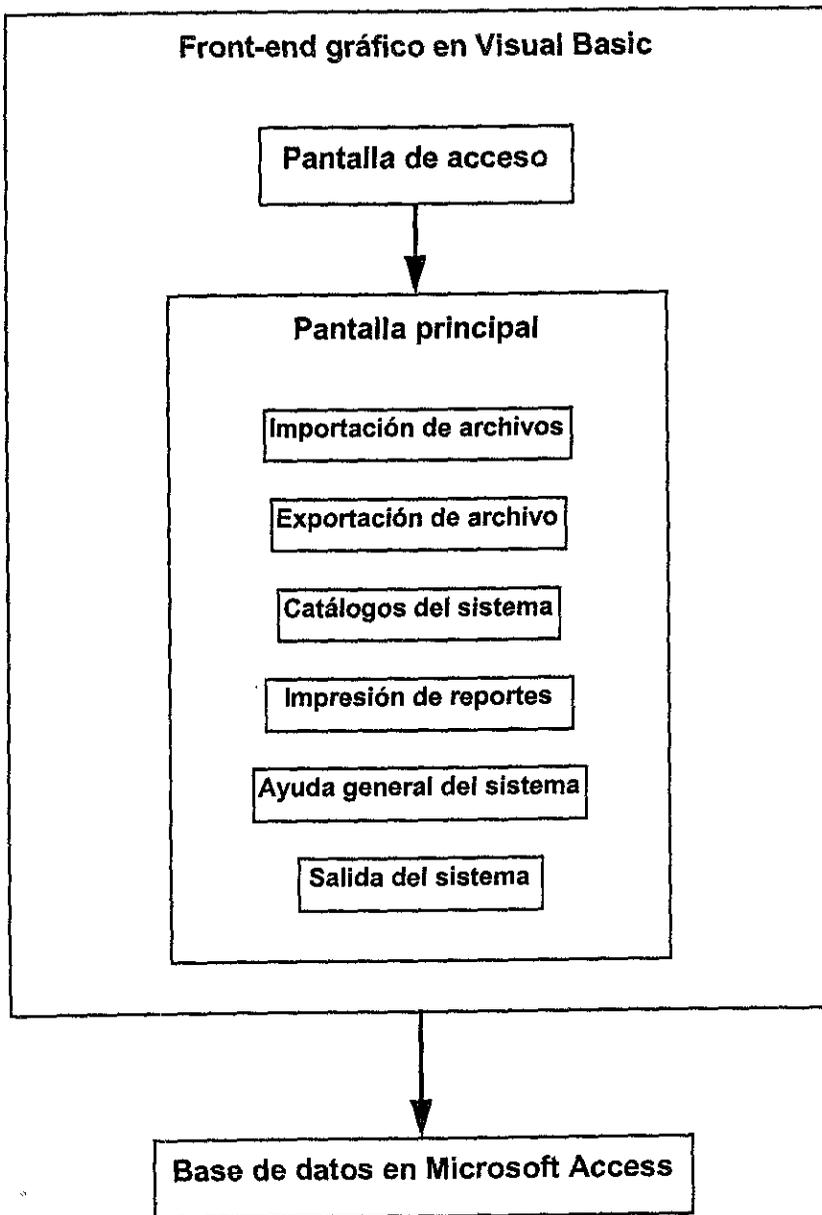


Figura 2.5.5.1. Diagrama de bloques de funciones del sistema.

Cada uno de estos controles, dependiendo de cual se seleccione mostrará una serie de opciones a través de las cuales el usuario se irá enfocando hacia el proceso específico que desea realizar. Al elegir alguna opción, el sistema mostrará una serie de pantallas predeterminadas donde el usuario ingresará información o escogerá sobre una serie de opciones para indicarle al sistema qué información desea ingresar o extraer.

El sistema responderá al requerimiento de información del usuario mostrándola en pantalla, en una impresión ó en un archivo. Los reportes emitidos por el sistema también estarán predeterminados en Crystal Reports. Si el usuario va a actualizar alguna información, el sistema le proporcionará un conjunto de pantallas donde el usuario la ingresará fácilmente.

Todos lo movimientos realizados por el usuario, serán registrados por el sistema como parte de la seguridad que el sistema ofrece.

En todo momento existirá la posibilidad de regresar a la pantalla principal si es el caso de que se hubiese elegido una pantalla errónea o simplemente se deséase salir del sistema.

A continuación detallaremos el conjunto de controles listados anteriormente, especificando las funciones asociadas a cada uno de ellos.

**Importación de Archivos.** Este control tiene como función, el permitir el ingreso de información manualmente. El usuario debe proporcionar la ruta del archivo que contiene la información a importar así, como la configuración del formato (posiciones, comas ó tabuladores). Al elegirlo, se dispondrán de las siguientes opciones:

- Archivo de productos consumidos por el almacén.
- Archivo de productos de SUN.

- Archivo de movimiento de productos.

La primera opción aceptará la importación de los movimientos de los productos en los almacenes de hospitales, normalmente estará activada para los usuarios de hospitales; la segunda proporcionará al usuario la manera de importar información manualmente desde los archivos de mantenimiento y la tercera será para importar información sobre productos facturados y en backorder.

**Exportación de Archivos.** Aquí se tiene contemplado el envío de archivos entre sistemas, desde esta empresa hacia hospitales y de los hospitales a esta empresa, se usará para alimentar bases de datos foráneas.

Los archivos que se podrán exportar serán:

- Archivo de mantenimiento de productos.
- Archivo para sistemas SUN.

**Catálogos del Sistema.** Tiene como función la actualización de productos, hospitales, almacenes, productos por cliente y de usuarios de manera local ya sea dentro de la empresa o en cada uno de los hospitales. Contemplará las siguientes funciones:

- Actualizador de productos.
- Actualizador de hospitales.
- Actualizador de almacenes.
- Actualizador de producto-cliente.
- Actualizador de usuarios.
- Actualizador de perfil de usuarios.

Los actualizadores de productos, hospitales y almacenes permitirán registrar las altas, bajas, cambios y consultas respecto a ellos en la BD, mientras que el actualizador de

producto-cliente lo hará a nivel detalle cliente. El actualizador de usuarios permitirá configurar la lista de usuarios junto con su password asociado, mientras que el otorgamiento de niveles de acceso a funciones se otorgará mediante el actualizador de perfil de usuario.

**Impresión de Reportes.** Se tienen contemplados los siguientes reportes:

- Listado de productos por cliente.
- Listado de precios de productos por cliente.
- Listado de pedidos.
- Listado de productos facturados y en backorder.
- Gráficas de pedidos de productos.

Los listados de productos, de precios y de pedidos nos mostrarán las características de cada uno de ellos por listados parciales o totales; en cuanto a los listados de productos facturados y en backorder nos mostrarán los reportes de los productos que no pudieron surtirse al hospital por rangos específicos. La gráfica de pedidos de productos nos mostrará la cantidad de pedidos ya sea por hospitales, almacenes o productos seleccionados, esto será utilizado para fines estadísticos.

**Ayuda General del Sistema.** Esta función proporcionará diferentes pantallas de ayuda e información del sistema.

**Salida del Sistema.** Es la puerta de salida del sistema.

**Base de Datos.** Estará directamente controlada desde el front-end y contendrá una serie de tablas donde se mantendrá la información procesada.

## **CAPÍTULO III**

# **DISEÑO E IMPLANTACIÓN DEL SISTEMA**

## **3. DISEÑO E IMPLANTACIÓN DEL SISTEMA**

### **3.1. DIAGRAMA DE FLUJO DE DATOS**

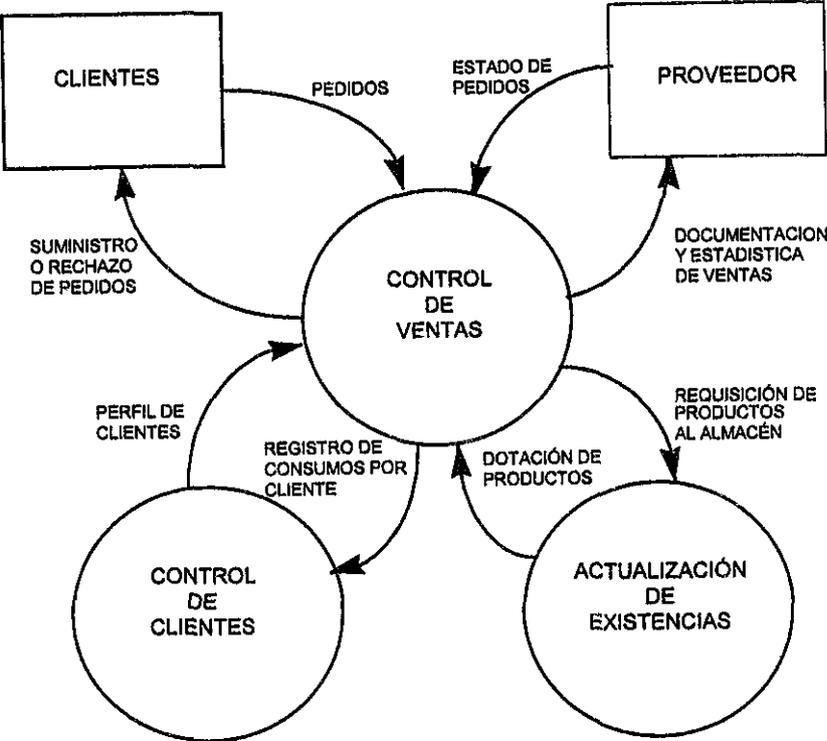
El diagrama de funcionalidad nos ayuda a establecer el perfil que esperamos del sistema, sin embargo, para el diseño es necesario establecer el curso que siguen los datos a través de ciertos puntos de proceso en los que podemos dividir a un sistema de información. Estos puntos de proceso pueden ser tantos como el nivel de detalle del flujo de datos que se quiera alcanzar. Para cada proceso que es alimentado de datos, se realiza una función determinada que generará otros, la cualidad y cantidad de datos obtenidos a la salida de cada proceso son las que determinarán su futuro destino, para así, convertirse en todo caso en la entrada de otro proceso.

El diagrama de flujo de datos ayuda a determinar la relación entre procesos, de acuerdo a los datos que fluyen entre uno y otro. Todas las posibles fuentes de datos externas deben ser consideradas en este diagrama, ya que son las que determinan en gran medida el comportamiento de el(los) proceso(s) con el(los) que tiene(n) el primer contacto. Así como también deben incluirse los datos que produzca el sistema global ya que estos pueden tornarse en la entrada de datos de otro sistema.

Las características que los datos vayan adquiriendo conforme viajan entre los procesos serán determinadas por las funciones de estos últimos. Es importante dar seguimiento a todo dato de entrada para que en su paso entre procesos se genere(n) su(s) correspondiente(s) salida(s).

El diagrama de flujo de datos que podemos observar en la figura 3.1.1 nos da una visión general del flujo de datos que se pretende obtener, está en un nivel 0. La descripción de cada proceso nos lleva a la elaboración de diagramas de flujo por niveles, es decir, conforme se avanza en el número de nivel, se tendrá más detalle tanto en procesos como en el flujo de datos. Así, tenemos que las figuras 3.1.2 y 3.1.3

corresponden a los niveles 1 y 2 respectivamente; y las figuras 3.1.4, 3.1.5 y 3.1.6 al nivel 3.



**Figura 3.1.1. Diagrama de flujo de nivel 0.**

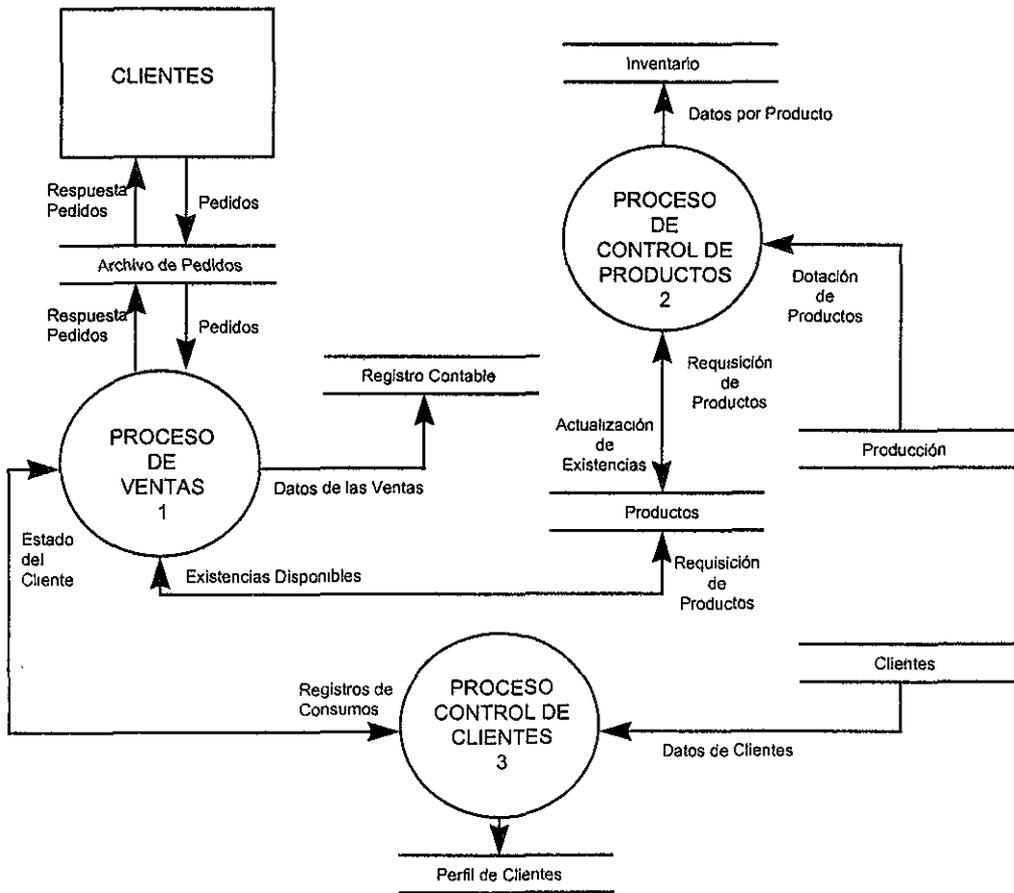


Figura 3.1.2. Diagrama de flujo, nivel 1.

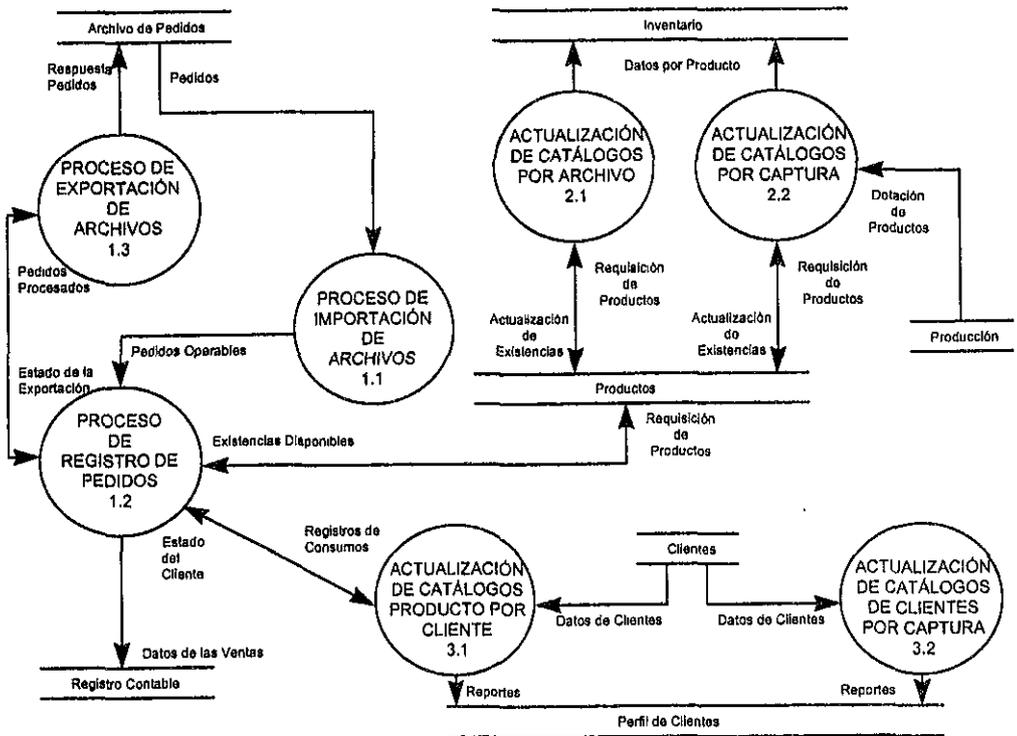


Figura 3.1.3. Diagrama de flujo de nivel 2.

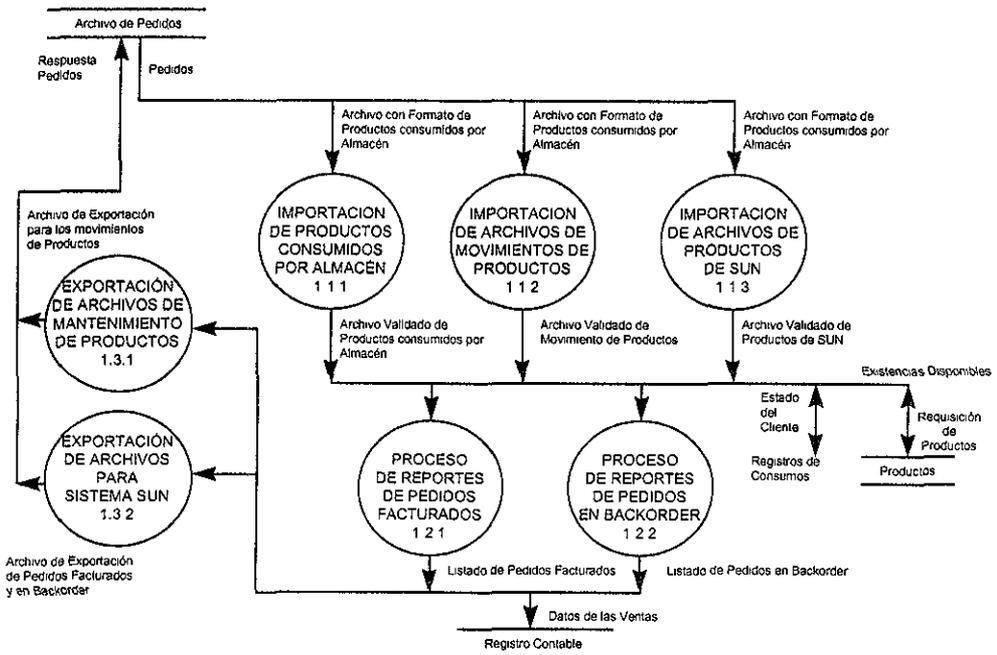


Figura 3.1.4. Diagrama de flujo de nivel 3. Control de ventas.

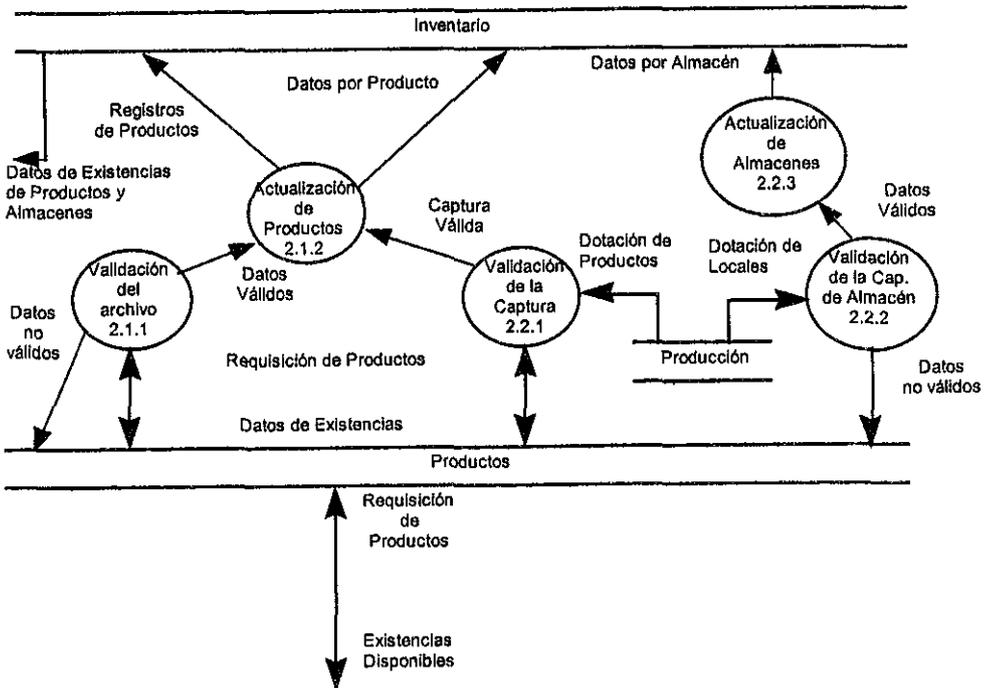


Figura 3.1.5. Diagrama de flujo de nivel 3. Control de catálogos.

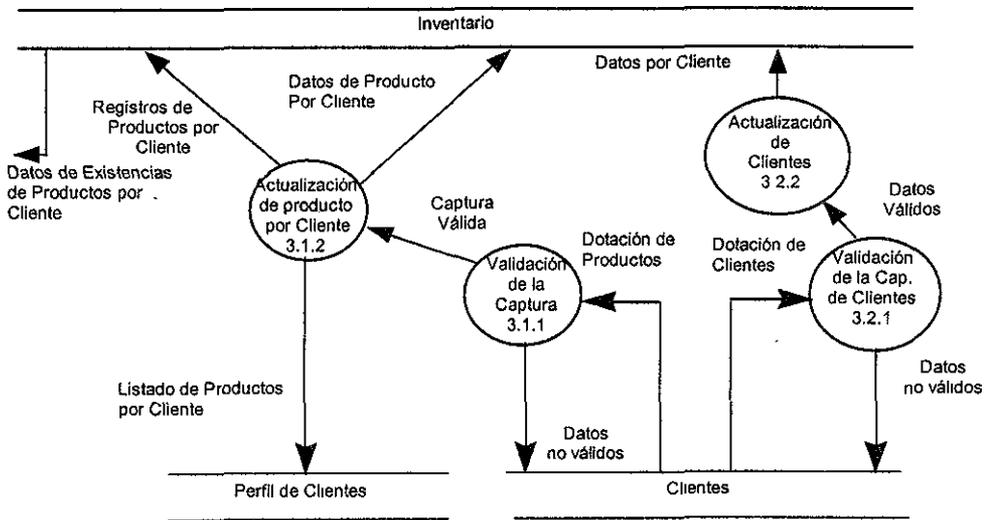


Figura 3.1.6. Diagrama de flujo de nivel 3. Control de clientes.

## 3.2. MODELO ENTIDAD-RELACIÓN Y NORMALIZACIÓN DE DATOS

El modelo de datos de Entidad-Relación (E-R), como ya se mencionó en capítulos anteriores, se basa en una percepción del mundo real que consiste en un conjunto de objetos básicos llamados *entidades* y de *relaciones* entre estos objetos. Este esquema representa la estructura lógica general de la base de datos.

### Entidad

Una entidad es un conjunto de cosas reales o abstractas, como una persona, un evento, o un cliente; acerca de la cual es importante almacenar información. Una entidad puede ser a menudo pensada como una tabla de una base de datos o como un archivo. Cada entidad debe tener un nombre *único dentro del sistema*, lo que indica que no deben existir dos entidades dentro del mismo sistema con el mismo nombre.

La entidad dentro de un modelo relacional se representa como una tabla, en donde el nombre de la tabla corresponde al nombre de la entidad. Además, cada tabla debe contener una columna que identifique de forma única a cada renglón de ésta. Esta columna recibe el nombre de llave primaria (PK), la cual no puede contener valores nulos, ni duplicados.

La figura 3.2.1 muestra dos entidades de la base de datos, *t\_cliente* y *t\_almacen*.

## t\_cliente

| id_clie_cliente | clie_descripcion  | clie_direccion       | clie_fecha | clie_telefono | clie_estatus_envio |
|-----------------|-------------------|----------------------|------------|---------------|--------------------|
| ESPA            | Hosp. Español     | Ejército Nac. N. 613 | 10/02/98   | 1234567       | Realizado          |
| INGL            | Hosp. Inglés      | Av Observatorio      | 15/02/98   | 2345678       | Incompleto         |
| ABC             | Hosp. ABC         | Periférico Sur       | 19/06/97   | 3456789       | Realizado          |
| HOIN            | Hosp. Infantil P. | Viaducto Río Becerra | 11/11/97   | 4567890       | Incompleto         |
| LOND            | Clínica Londres   | Durango No. 50       | 25/11/97   | 5678901       | Realizado          |

## t\_almacen

| id_cli_cliente | id_alma_almacen | alma_descripcion  | alma_direccion     | alma_usuario |
|----------------|-----------------|-------------------|--------------------|--------------|
| ABC            | ABC1            | Almacén 1 de ABC  | Durango No. 50     | Gte. Ventas  |
| ABC            | ABC 2           | Almacén 2 de ABC  | Londres No. 34     | Gte.         |
| INGL           | INGL1           | Almacén 1 de INGL | Av. Constituyentes | Director     |

**Figura 3.2.1. Entidades t\_cliente y t\_almacen.**

### Atributo

Es una información individual acerca de una entidad. Representa un tipo de característica, propiedad asociada a un conjunto de cosas reales o abstractas.

De la entidad t\_cliente\_pedido, que se muestra junto con la entidad t\_factura\_orden en la figura 3.2.2, tenemos como sus atributos id\_clie\_cliente, id\_alma\_almacen, alma\_descripcion, alma\_direccion y alma\_usuario.

### Relación

Una relación es una liga entre dos entidades, o entre una entidad así misma. Las relaciones pueden ser puntos comunes de referencia o semejanzas directas entre entidades. Las relaciones están representadas en el diagrama E-R como líneas conectando o uniendo a las entidades, estas relaciones deben caer dentro del alcance

del sistema, es decir, acerca de estas, el sistema deberá mantener, correlacionar y mostrar información.

Se puede observar que existe una relación entre la entidad *t\_cliente* y la entidad *t\_almacen* (figura 3.2.1), es decir, un cliente puede tener uno o muchos almacenes. Lo anterior especifica que el cliente "Hospital ABC" con *id\_clie\_cliente* = "ABC" tiene dos almacenes: ABC1 Y ABC2.

### **t\_factura\_orden**

| <i>clipp_norden</i> | <i>id_clpr_producto</i> | <i>clipp_fecha_factura</i> | <i>faor_cantidad_abas</i> | <i>faor_estatus_envio</i> | <i>faor_usuario</i> |
|---------------------|-------------------------|----------------------------|---------------------------|---------------------------|---------------------|
| 123                 | JER5                    | 10/01/98                   | 1000                      | Realizado                 | Responsable         |
| 123                 | JER2                    | 10/01/98                   | 1900                      | Realizado                 | Responsable         |
| 123                 | VEN5                    | 10/01/98                   | 3980                      | Realizado                 | Responsable         |
| 456                 | GAS3                    | 13/02/98                   | 10000                     | Realizado                 | Director            |
| 456                 | JER2                    | 13/02/98                   | 15000                     | Realizado                 | Director            |

### **t\_cliente\_pedido**

| <i>id_clie_cliente</i> | <i>id_alma_almacen</i> | <i>id_prod_producto</i> | <i>clipp_norden</i> | <i>clipp_ref_cliente</i> | <i>clipp_fecha</i> | <i>clipp_cantidad_pedido</i> |
|------------------------|------------------------|-------------------------|---------------------|--------------------------|--------------------|------------------------------|
| INGL                   | INGL1                  | JER2                    | 123                 | GYT-76                   | 10/01/98           | 1900                         |
| INGL                   | INGL1                  | JER5                    | 123                 | GYT-76                   | 10/01/98           | 1000                         |
| ABC                    | ABC2                   | VEN5                    | 456                 | GYT-76                   | 10/01/98           | 3980                         |
| ABC                    | ABC1                   | GAS3                    | 456                 | GYT-76                   | 13/02/98           | 10000                        |
| INGL                   | INGL1                  | VEN5                    | 800                 | QWE-89                   | 12/12/97           | 15000                        |

**Figura 3.2.2. Entidades *t\_factura\_orden* y *t\_cliente\_pedido*.**

Podemos observar que hay una relación entre la entidad *t\_cliente\_pedido* y la entidad *t\_factura\_orden* (figura 3.2.2), es decir, que un pedido puede tener sólo una factura. Significa que para el pedido 123, con tres productos en el mismo, tendrá una sola

factura con los datos del cliente, fecha y producto.

### t\_cliente\_producto

| id_clie_cliente | id_alma_almacen | id_prod_producto | clpr_descripcion | clpr_minimo | clpr_maximo | clpr_existencia | clpr_precio |
|-----------------|-----------------|------------------|------------------|-------------|-------------|-----------------|-------------|
| INGL            | INGL1           | JER2             | Jeringa 2m       | 1000        | 3300000     | 3300000         | 2.00        |
| INGL            | INGL1           | JER5             | Jeringa 5m       | 1550        | 5676000     | 5676000         | 5.80        |
| ABC             | ABC1            | VEN5             | Vendas 5m        | 2200        | 8677400     | 8677400         | 2.89        |
| ABC             | ABC 1           | GAS3             | Gasas 3cm        | 7800        | 6352300     | 6352300         | 3.60        |
| INGL            | INGL2           | PEN4             | Ampoll. 400      | 8900        | 4578500     | 4578500         | 4.00        |

### t\_producto

| id_prod_producto | prod_descripcion    | prod_precio | prod_fecha |
|------------------|---------------------|-------------|------------|
| JER2             | Jeringa de 2ml.     | 1           | 11/11/97   |
| JER5             | Jeringa de 5ml.     | 2           | 21/01/98   |
| VEN5             | Vendas de 5m.       | 2.5         | 17/11/97   |
| GAS3             | Gasas de 3cm.       | 1.8         | 19/10/97   |
| PEN4             | Ampolletas de 400u. | 6.79        | 30/09/97   |

**Figura 3.2.3. Entidades t\_cliente\_producto y t\_producto.**

Podemos observar en la figura 3.2.3, que la entidad t\_cliente\_producto tiene una relación con la entidad t\_producto. Significa que para cada clave de producto en la entidad t\_cliente\_producto, tenemos una descripción en la entidad producto.

Se definen los conjuntos de entidades:

- Cliente

Conjunto de todos los clientes de la empresa, cada uno se describe con los atributos id\_clie\_cliente, clie\_descripción, clie\_dirección, clie\_teléfono.

- Almacén  
 Conjunto de todos los almacenes que corresponden a los clientes de la empresa, cada uno se describe con los atributos `id_alma_almacen`, `id_clie_cliente`, `alma_descripcion`, `alma_direccion`, `alma_fecha`, `alma_estatus_envio`.
- Pedido  
 Conjunto de todos los pedidos que realizan los clientes a la empresa. Cada uno se describe por los atributos `id_clie_cliente`, `id_prod_producto`, `id_alma_almacen`, `clpp_cantidad_pedido`, `clpp_ref_cliente`, `clpp_estatus_pedido`, `clpp_fecha_factua`, `clpp_norden`.
- Usuario  
 Conjunto de todas las personas que trabajan en el sistema. Cada uno se describe con los atributos `id_usuario`, `usua_nombre`, `usua_password`.
- Función  
 Conjunto de todas las funciones que puede realizar un usuario dentro del sistema. Cada una se describe con los `id_usuario`, `id_func_funcion`, `func_usuario`.
- Productos  
 Conjunto de todos los productos que distribuye la empresa. Cada uno se describe con los atributos `id_prod_producto`, `prod_descripcion`, `prod_precio`, `prod_fecha`.
- Factura  
 Conjunto de todas las facturas realizadas por los clientes en un pedido. Cada una se describe con los atributos `clpp_norden`, `id_clpr_producto`, `clpp_fecha_factura`, `faor_cantidad_abas`.
- Tipo de Pedido  
 Conjunto de los tipos de pedido que maneja la empresa. Cada uno se describe con los atributos `id_tipo_estatus` y `tipo_descripcion`.

### 3.2.1. Limitantes de Mapeo

Un esquema E-R puede definir ciertas limitantes con las que debe cumplir los datos contenidos en la base de datos. Una limitante importante es la de las cardinalidades de mapeo que expresan el número de entidades con las que puede asociarse otra entidad mediante una relación. Véase la figura 3.2.1.1.

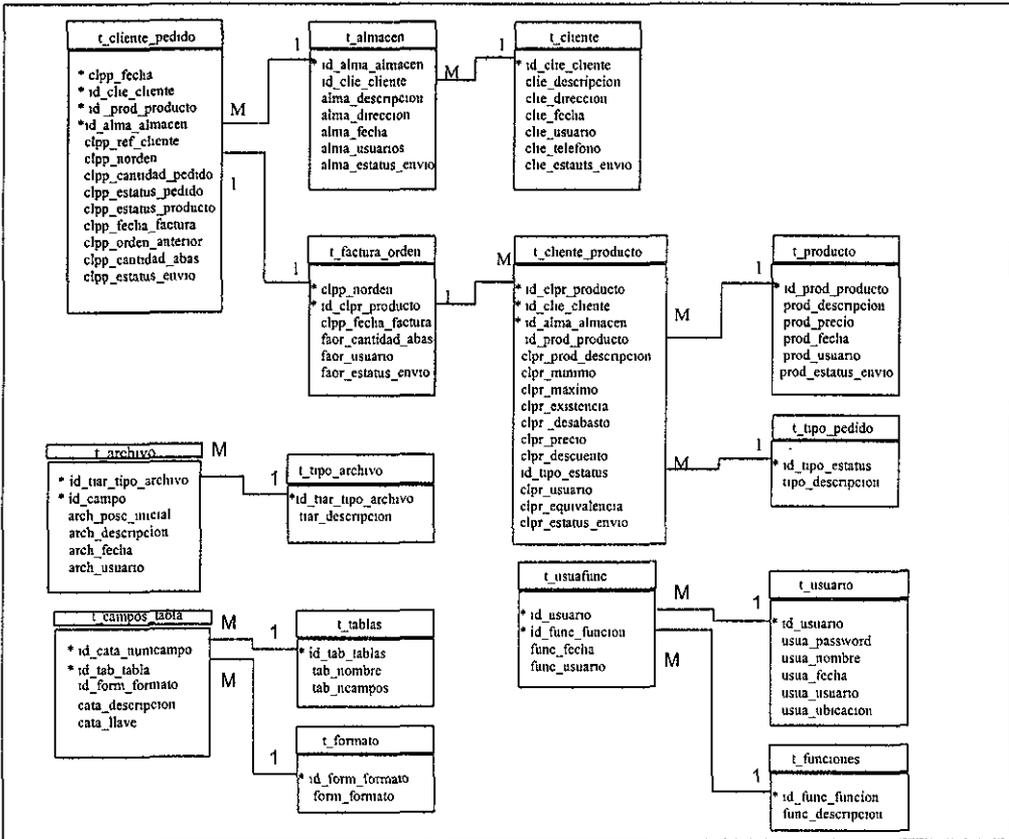


Figura 3.2.1.1. Limitantes de mapeo.

La entidad **t\_cliente\_pedido** tiene una relación de una a una con la entidad

t\_factura\_orden pues sólo existe una factura por cada pedido solicitado.

La entidad t\_cliente tiene una relación de uno a muchas con la entidad t\_almacen, podemos decir que un cliente puede tener muchos almacenes.

La entidad t\_almacen tiene una relación de una a muchas con la entidad t\_cliente\_pedido, podemos decir que un almacén puede tener muchos pedidos.

La entidad t\_cliente\_pedido tiene una relación una a una con la entidad t\_factura\_orden, podemos decir que un pedido sólo puede tener una factura y viceversa.

La entidad t\_cliente\_producto tiene una relación de muchas a una con la entidad t\_producto, de lo anterior podemos decir, que muchos productos de t\_cliente\_producto tienen solo una descripción de producto en la entidad t\_producto.

### **3.2.2. Diagrama Entidad-Relación**

La estructura lógica general de la base de datos puede expresarse en forma gráfica por medio de un diagrama E-R que se constituye por los siguientes elementos:

- Rectángulos, representan conjuntos de entidades.
- Elipses, representan atributos.
- Rombo, representan conjuntos de relaciones.
- Líneas, conectan los atributos con los conjuntos de entidades y los conjuntos de entidades a los conjuntos de relaciones.

Hay un número de tareas que deben ser realizadas durante el desarrollo de un diagrama de E-R y un modelo de datos completamente normalizado.

Estas tareas son las siguientes:

- Identificación y definición de entidades.
- Identificación de llaves primarias (PK).
- Identificación de llaves foráneas (FK).
- Definición de atributos.
- Definición de relaciones .
- Normalización.

En la figura 3.2.2.1 se muestra el diagrama Entidad-Relación del sistema. En dicho diagrama se muestran las relaciones y cardinalidad de las entidades.

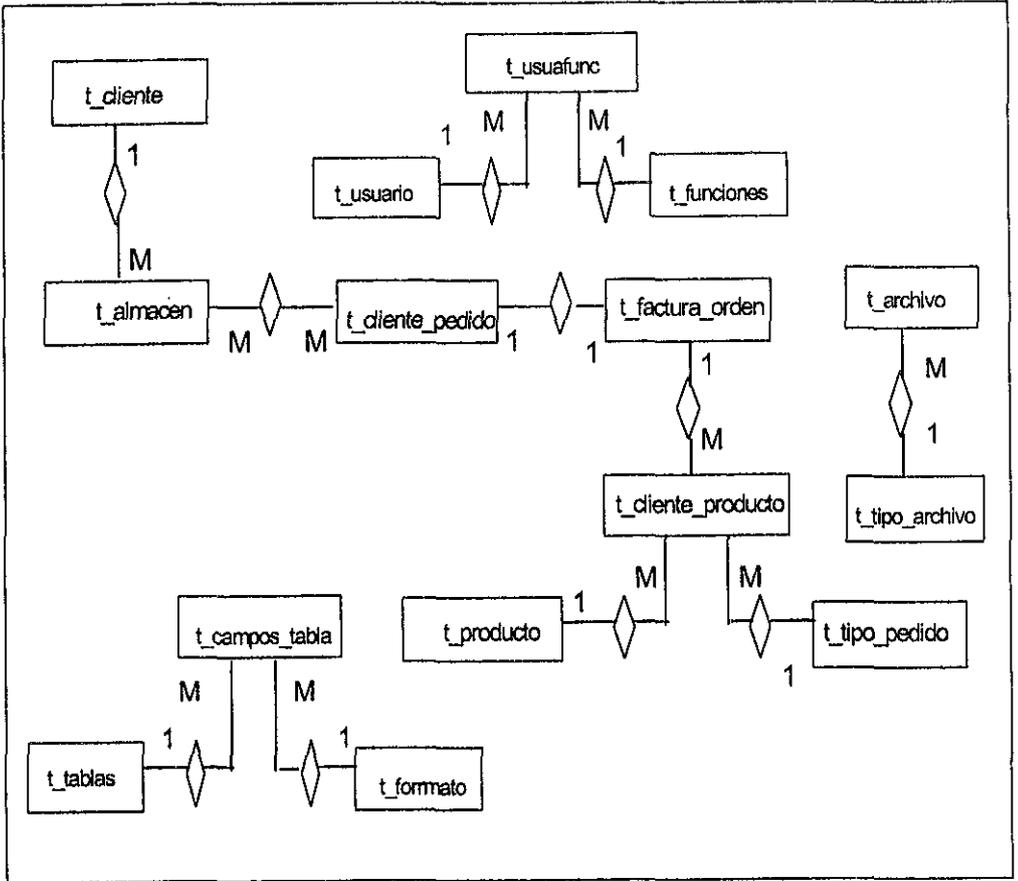


Figura 3.2.2.1. Diagrama E-R.

### **3.2.3. Normalización de Datos**

Codd ha identificado ciertas características estructurales en las relaciones que crean problemas de recuperación y actualización. Esas características indeseables pueden eliminarse descomponiendo una relación por otras, de estructuras deseables. Este proceso se conoce como Normalización. Existen tres tipos de propiedades indeseables:

- Grupos de Repetición.
- Uso múltiple del mismo atributo.
- Múltiple ocurrencia del mismo hecho.

Por lo anterior la normalización requiere de tres acciones sobre un atributo de una entidad. Estas son las siguientes:

- Primera Forma Normal.
- Segunda Forma Normal.
- Tercera Forma Normal.

#### **3.2.3.1. Primera Forma Normal**

La finalidad de la primera forma normal es simplificar la estructura de una relación asegurando que contenga solamente datos elementales y no agrupaciones de datos. Es decir cada identificador en una relación debe poseer un solo valor de cada uno de los atributos y no múltiples valores de estos.

La información que se manejará para la explicación de la normalización corresponde a una información en bruto a la cual se le aplicará dicho proceso.

Comenzaremos con la relación PEDIDO que se muestra en la figura 3.2.3.1.1.

## PEDIDO

| Cve.<br>Prod. | Producto        | Cve.<br>Cliente | Referencia | Almacón | Orden | fecha_pedido<br>o<br>fecha_factura | Cantidad | Costo/U | Importe |
|---------------|-----------------|-----------------|------------|---------|-------|------------------------------------|----------|---------|---------|
| JER5          | Jeringa de 5ml. | ABC             | GYT-76     | ABC1    | 150   | 10/01/98                           | 10000    | 3.25    | 32500   |
| JER5          | Jeringa de 5ml  | ABC             | GYT-76     | ABC1    | 170   | 13/02/98                           | 15000    | 3.25    | 48750   |
| GAS3          | Gasas de 30cm   | ABC             | GYT-76     | ABC2    | 456   | 12/12/98                           | 10000    | 5.5     | 55000   |
| GAS3          | Gasas de 30cm   | ABC             | GYT-76     | ABC1    | 456   | 13/02/98                           | 7000     | 5.5     | 38500   |
| VEN5          | Vendas de 5m    | ING             | QWE-89     | ABC2    | 100   | 10/01/98                           | 3960     | 4.98    | 19820.4 |
| VEN5          | Vendas de 5m    | ING             | QWE-89     | ING1    | 800   | 12/12/98                           | 3400     | 4.98    | 16932   |
| JER2          | Jeringa de 2ml. | ING             | XCV-93     | ING2    | 700   | 13/02/98                           | 1900     | 2.5     | 4750    |
| JER2          | Jeringa de 2ml. | ABC             | GYT-76     | ABC1    | 580   | 10/01/98                           | 2000     | 2.5     | 5000    |
| JER2          | Jeringa de 2ml. | ABC             | GYT-76     | ABC2    | 123   | 13/02/98                           | 40000    | 2.5     | 100000  |

**Figura 3.2.3.1.1. Relación PEDIDO.**

La relación PEDIDO posee múltiples valores para el atributo Clave Producto (Cve. Prod.) que es el identificador, por lo tanto no es una relación plana.

Se dice que una relación esta en primera forma normal si cada valor específico de un indicador existe uno y solo un valor de cada atributo. Es decir, no hay grupos "repetitivos". Pero se mantiene un alto grado de redundancia. Considerando el ejemplo anterior, la conversión de la relación a una forma plana procede de la manera como se muestra en la figura 3.2.3.1.2.

## PEDIDO

| Cve.<br>Prod | Producto           | Cve.<br>Cliente | Referen. | Almacén | Orden | fecha_pedido o<br>fecha_factura | Cantidad | Costo | Importe |
|--------------|--------------------|-----------------|----------|---------|-------|---------------------------------|----------|-------|---------|
| JER5         | Jeringa<br>de 5ml. | ABC             | GYT-76   | ABC1    | 150   | 10/01/98                        | 10000    | 3.25  | 32500   |
| JER5         | Jeringa<br>de 5ml  | ABC             | GYT-76   | ABC1    | 170   | 13/02/98                        | 15000    | 3.25  | 48750   |
| GAS3         | Gasas de<br>30cm   | ABC             | GYT-76   | ABC2    | 456   | 12/12/98                        | 10000    | 5.5   | 55000   |
| GAS3         | Gasas de<br>30cm   | ABC             | GYT-76   | ABC1    | 456   | 13/02/98                        | 7000     | 5.5   | 38500   |
| VEN5         | Vendas<br>de 5m    | ING             | QWE-89   | ABC2    | 100   | 10/01/98                        | 3960     | 4.98  | 19820.4 |
| VEN5         | Vendas<br>de 5m    | ING             | QWE-89   | ING1    | 800   | 12/12/98                        | 3400     | 4.98  | 16932   |
| JER2         | Jeringa<br>de 2ml. | ING             | XCV-93   | ING2    | 700   | 13/02/98                        | 1900     | 2.5   | 4750    |
| JER2         | Jeringa<br>de 2ml. | ABC             | GYT-76   | ABC1    | 580   | 10/01/98                        | 2000     | 2.5   | 5000    |
| JER2         | Jeringa<br>de 2ml. | ABC             | GYT-76   | ABC2    | 123   | 13/02/98                        | 40000    | 2.5   | 100000  |

**Figura 3.2.3.1.2. Conversión de la relación PEDIDO a una forma plana.**

Esta es la forma más común de llegar a la primera forma normal. La redundancia puede reducirse separándola en otro grupo a parte de la entidad debiendo contener el identificador de la entidad original para mantenerse relacionadas. En este caso se separa como identificador la clave del producto y como atributos producto y costo por unidad. Por lo que las tablas quedan como se muestran en las figuras 3.2.3.1.3 y 3.2.3.1.4.

## PEDIDO

| Cve. Prod. | Cve. Cliente | Referencia | Almacén | Orden | fecha_pedido o fecha_factura | Cantidad | importe |
|------------|--------------|------------|---------|-------|------------------------------|----------|---------|
| JER5       | ABC          | GYT-76     | ABC1    | 150   | 10/01/98                     | 10000    | 32500   |
| JER5       | ABC          | GYT-76     | ABC1    | 170   | 13/02/98                     | 15000    | 48750   |
| GAS3       | ABC          | GYT-76     | ABC2    | 456   | 12/12/98                     | 10000    | 55000   |
| GAS3       | ABC          | GYT-76     | ABC1    | 456   | 13/02/98                     | 7000     | 38500   |
| VEN5       | ING          | QWE-89     | ABC2    | 100   | 10/01/98                     | 3960     | 19820.4 |
| VEN5       | ING          | QWE-89     | ING1    | 800   | 12/12/98                     | 3400     | 16932   |
| JER2       | ING          | XCV-93     | ING2    | 700   | 13/02/98                     | 1900     | 4750    |
| JER2       | ABC          | GYT-76     | ABC1    | 580   | 10/01/98                     | 2000     | 5000    |
| JER2       | ABC          | GYT-76     | ABC2    | 123   | 13/02/98                     | 40000    | 100000  |

Figura 3.2.3.1.3. Relación PEDIDO en primera forma normal.

## PRODUCTO

| Cve. Prod. | Producto        | Costo/Unidad |
|------------|-----------------|--------------|
| JER5       | Jeringa de 5ml. | 3.25         |
| GAS3       | Gasas de 30cm   | 5.5          |
| VEN5       | Vendas de 5m    | 4.98         |
| JER2       | Jeringa de 2ml. | 2.5          |
| PEN4       | Ampolleta 4ml   | 6.80         |
| ALG1       | Algodón de 1kg  | 9.10         |
| ALG.5      | Algodón de 500g | 4.80         |
| GAS7       | Gasas de 7cm    | 3.5          |
| VEN10      | Vendas de 10m   | 9.00         |

Figura 3.2.3.1.4. Relación PRODUCTO en primera forma normal.

El identificador de la relación PRODUCTO es Cve. Prod.

### 3.2.3.2. Segunda Forma Normal

La prueba para determinar si una entidad está en Segunda Forma Normal es: que el valor de cualquier atributo que no es llave dependa de todos los atributos que forman la llave.

Para normalizar una entidad en la segunda forma, se crea una nueva entidad de los atributos que dependen parcialmente de una llave, siendo parte del identificador de esta nueva entidad el atributo del cual depende para mantenerla relacionada con la original.

En la tabla de la figura 3.2.3.1.3 observamos que hay un problema, por supuesto, es que tenemos un solo atributo "fecha\_pedido o fecha\_factura" el cual puede representar uno de dos hechos y no hay manera de saber a que hecho representa. Así mismo no hay forma de representar ambas fechas -la fecha del pedido y la fecha en que se factura el pedido- en situaciones cuando ambas fechas son conocidas.

La solución es no requerir de un solo atributo para que realice una doble tarea, sino por el contrario, permitir que atributos por separado, soporten hechos separados. Podríamos dejar el atributo "fecha\_pedido o fecha\_factura" y además agregar un atributo de tipo\_fecha el cual nos dirá si es para fecha\_pedido o para fecha\_factura.

Analizando bien la solución observamos que no es totalmente correcta ya que puede desencadenar confusión en la lógica para efectuar consultas. En realidad lo que hemos creado es otro tipo de error de normalización (el atributo tipo\_fecha no depende del atributo cve\_producto más bien dependería de la fecha que se esta informando), estos errores se ven con frecuencia cuando existe un atributo soportando más de un hecho. Para dar una solución a lo anterior la tabla quedaría como la que se muestra en la figura 3.2.3.2.1.

| Cve.<br>Prod | Cve.<br>Cliente | Referencia | Almacén | Orden | fecha_pedido | fecha_factura | Cantidad. | Importe |
|--------------|-----------------|------------|---------|-------|--------------|---------------|-----------|---------|
| JER5         | ABC             | GYT-76     | ABC1    | 150   | 10/01/98     | 13/01/98      | 10000     | 32500   |
| JER5         | ABC             | GYT-76     | ABC1    | 170   | 09/02/98     | 13/02/98      | 15000     | 48750   |
| GAS3         | ABC             | GYT-76     | ABC2    | 456   | 08/12/98     | 12/12/978     | 10000     | 55000   |
| GAS3         | ABC             | GYT-76     | ABC1    | 456   | 09/02/98     | 13/02/98      | 7000      | 38500   |
| VEN5         | ING             | QWE-89     | ABC2    | 100   | 10/01/98     | 13/01/98      | 3960      | 19820.4 |
| VEN5         | ING             | QWE-89     | ING1    | 800   | 08/12/98     | 12/12/978     | 3400      | 16932   |
| JER2         | ING             | XCV-93     | ING2    | 700   | 09/02/98     | 13/02/98      | 1900      | 4750    |
| JER2         | ABC             | GYT-76     | ABC1    | 580   | 10/01/98     | 13/01/98      | 2000      | 5000    |
| JER2         | ABC             | GYT-76     | ABC2    | 123   | 09/02/98     | 13/02/98      | 40000     | 100000  |

**Figura 3.2.3.2.1. Relación PEDIDO con una nueva entidad.**

Como observamos en la tabla de la figura 3.2.3.2.1 todavía no se cumple la segunda forma normal en que el valor de cualquier atributo que no es llave depende de todos los atributos que forman la llave. Por ejemplo: Orden y fecha de factura no depende de todos los atributos que forman al identificador o llave. Por lo anterior las tablas quedan como se muestran en las figuras 3.2.3.2.2 y 3.2.3.2.3.

## **PEDIDO**

| Cve.<br>Prod. | Cve.<br>Cliente | Referencia | Almacén | fecha_pedido | Cantidad | Importe |
|---------------|-----------------|------------|---------|--------------|----------|---------|
| JER5          | ABC             | GYT-76     | ABC1    | 10/01/98     | 10000    | 32500   |
| JER5          | ABC             | GYT-76     | ABC1    | 09/02/98     | 15000    | 48750   |
| GAS3          | ABC             | GYT-76     | ABC2    | 08/12/98     | 10000    | 55000   |
| GAS3          | ABC             | GYT-76     | ABC1    | 09/02/98     | 7000     | 38500   |
| VEN5          | ING             | QWE-89     | ABC2    | 10/01/98     | 3960     | 19820.4 |
| VEN5          | ING             | QWE-89     | ING1    | 08/12/98     | 3400     | 16932   |
| JER2          | ING             | XCV-93     | ING2    | 09/02/98     | 1900     | 4750    |
| JER2          | ABC             | GYT-76     | ABC1    | 10/01/98     | 2000     | 5000    |
| JER2          | ABC             | GYT-76     | ABC2    | 09/02/98     | 40000    | 100000  |

**Figura 3.2.3.2.2. Relación PEDIDO en segunda forma normal.**

## FACTURA

| Orden | Cve. Prod. | Fecha_factura |
|-------|------------|---------------|
| 150   | JER5       | 13/01/98      |
| 170   | JER5       | 13/02/98      |
| 456   | GAS3       | 12/12/978     |
| 456   | GAS3       | 13/02/98      |
| 100   | VEN5       | 13/01/98      |
| 800   | VEN5       | 12/12/978     |
| 700   | JER2       | 13/02/98      |
| 580   | JER2       | 13/01/98      |
| 123   | JER2       | 13/02/98      |

**Figura 3.2.3.2.3. Relación FACTURA en segunda forma normal.**

El identificador de la relación FACTURA es Orden.

### 3.2.3.3. Tercera Forma Normal

Se dice que una entidad se encuentra en Tercera Forma Normal si el valor de cada atributo depende de toda la llave y no de cualquier otro que no lo sea.

En esta forma normal, se buscan los atributos que están dependiendo de otro que no es una llave.

Para poner una entidad en tercera forma normal se crea una entidad con los atributos que no dependen de ningún atributo que forma la llave, siendo identificador de la nueva entidad el atributo del cual era dependiente.

Una entidad viola la segunda forma normal, si un hecho puede ser determinado conociendo sola una parte de la llave primaria de la entidad. Si un hecho puede ser

determinado conociendo el valor de algún atributo no-llave de la entidad, entonces la tercera forma normal es violada. De la tabla de la figura 3.2.3.2.2 observamos que importe es funcionalmente dependiente de la cantidad del producto y de su costo; por lo tanto conociendo el valor de la cantidad y del costo podemos derivar el importe. La tabla quedaría entonces como la que se muestra en la figura 3.2.3.3.1.

## PEDIDO

| Cve. Prod. | Cve. Cliente | Referencia | Almacén | fecha_pedido | Cantidad |
|------------|--------------|------------|---------|--------------|----------|
| JER5       | ABC          | GYT-76     | ABC1    | 10/01/98     | 10000    |
| JER5       | ABC          | GYT-76     | ABC1    | 09/02/98     | 15000    |
| GAS3       | ABC          | GYT-76     | ABC2    | 08/12/98     | 10000    |
| GAS3       | ABC          | GYT-76     | ABC1    | 09/02/98     | 7000     |
| VEN5       | ING          | QWE-89     | ABC2    | 10/01/98     | 3960     |
| VEN5       | ING          | QWE-89     | ING1    | 08/12/98     | 3400     |
| JER2       | ING          | XCV-93     | ING2    | 09/02/98     | 1900     |
| JER2       | ABC          | GYT-76     | ABC1    | 10/01/98     | 2000     |
| JER2       | ABC          | GYT-76     | ABC2    | 09/02/98     | 40000    |

**Figura 3.2.3.3.1. Relación PEDIDO en tercera forma normal.**

Para sumarizar las tres formas normales se tiene la siguiente regla:

“Una entidad esta en tercera forma normal si cada atributo no-llave depende de la llave primaria de la entidad, dicho de otro modo, cada atributo no debe depender de nada excepto del identificador único de su entidad, llave primaria para un valor.

### 3.3. DICCIONARIO DE DATOS

El diccionario de datos se utiliza para definir el significado, uso, características y otros datos relevantes de todos los campos, entidades de referencias cruzadas y las relaciones que existen entre ellos.

Se almacena lo que en el medio se le conoce como metadatos (datos definidos por datos) para todos los objetos que residen en nuestra base de datos. Este diccionario se almacena en una ubicación especial del resto de las tablas de trabajo o datos y es utilizada exclusivamente por el **kernel** (núcleo) del manejador de bases de datos que utilizamos, es decir, Access.

Este diccionario de datos también nos permite validar la existencia de los objetos definidos en la base de datos proporcionando un acceso y mapas de ubicación física de almacenamiento.

Es una base de datos que contiene datos acerca de los datos. Es importante señalar que el diccionario de datos se tiene que analizar para saber cómo está formada la base de datos en su estructura.

Tuvimos especial cuidado en la creación del diccionario de datos puesto que determina una ruta de acceso óptimo para guardar o extraer los datos. Normalmente el proveedor del manejador de la base de datos proporciona un algoritmo sofisticado para permitir la extracción de los datos proporcionando la mejor respuesta para el primer conjunto de renglones presentados o para todos los registros que son extraídos.

A continuación se presenta el diccionario de datos y la figura con las relaciones entre tablas, figura 3.3.1.

Tabla: t\_almacen. Esta tabla contiene las características específicas de envío del pedido.

| No. Campo | Nombre            | Tipo       | Longitud | Descripción                   |
|-----------|-------------------|------------|----------|-------------------------------|
| 1         | Id_alma_almacen   | Texto      | 10       | Clave del almacén             |
| 2         | Id_clie_cliente   | Texto      | 10       | Clave del cliente             |
| 3         | Alma_descripción  | Texto      | 50       | Descripción del pedido        |
| 4         | Alma_dirección    | Texto      | 50       | Dirección                     |
| 5         | Alma_fecha        | Fecha/hora |          | Fecha de pedido               |
| 6         | Alma_usuario      | Texto      | 10       | Usuario del almacén           |
| 7         | Alma_status_envio | Número     | Simple   | Indicador de envío del pedido |

Tabla: t\_archivo. Contiene las características de los archivos que envían los clientes respecto a su pedido.

| No. Campo | Nombre                 | Tipo       | Longitud     | Descripción                |
|-----------|------------------------|------------|--------------|----------------------------|
| 1         | Id_triari_tipo_archivo | Número     | Entero largo | Clave del archivo          |
| 2         | Id_campo               | Número     | Entero largo | Clave del campo            |
| 3         | Arch_posc_inicial      | Número     | Doble        | Posición inicial del campo |
| 4         | Arch_pos_final         | Número     | Doble        | Posición final del campo   |
| 5         | Arch_descripcion       | Texto      | 50           | Descripción del archivo    |
| 6         | Arch_fecha             | Fecha/hora |              | Fecha                      |
| 7         | Arch_usuario           | Texto      | 10           | Usuario                    |

Tabla: t\_campo. Contiene las características respectivas a los campos de los archivos de los pedidos de los clientes.

| No. Campo | Nombre           | Tipo     | Longitud | Descripción           |
|-----------|------------------|----------|----------|-----------------------|
| 1         | Id_campo         | Numérico | Simple   | Clave del campo       |
| 2         | Camp_descripción | Texto    | 50       | Descripción del campo |

Tabla: t\_campos\_tabla. Contiene las características de los campos de las tablas que el sistema utiliza.

| No. Campo | Nombre            | Tipo     | Longitud | Descripción  |
|-----------|-------------------|----------|----------|--|
| 1         | Id_tab_tabla      | Texto    | 8        | Clave del archivo o tabla dentro de la base de datos         |
| 2         | Id_cata_numcampo  | Numérico | Simple   | Número de campo y orden dentro del registro                  |
| 3         | Carta_descripcion | Texto    | 30       | Nombre del campo dentro de la definición de la base de datos |
| 4         | Id_form_formato   | Numérico | Doble    | Formato y longitud del campo                                 |
| 5         | Cata_llave        | Numérico | Simple   | Tipo de campo 1=si es llave 0= i es atributo de la tabla     |

Tabla: t\_cargas. Guarda las características propias de cada archivo de pedido recibido.

| No Campo | Nombre              | Tipo         | Longitud     | Descripción                    |
|----------|---------------------|--------------|--------------|--------------------------------|
| 1        | Id_archivo          | Texto        | 10           | Clave del archivo              |
| 2        | Carg_fecha          | Fecha/hora   |              | Fecha de recepción del archivo |
| 3        | Carg_nombre_archivo | Texto        | 30           | Nombre del archivo             |
| 4        | Carg_consecutivo    | Autonumérico | Entero largo | Número de archivo              |

Tabla: t\_cliente. Esta tabla contiene los datos del cliente que hizo el pedido.

| No Campo | Nombre             | Tipo       | Longitud | Descripción                       |
|----------|--------------------|------------|----------|-----------------------------------|
| 1        | Id_clie_cliente    | Texto      | 10       | Clave del cliente                 |
| 2        | Clie_descripción   | Texto      | 50       | Descripción del pedido            |
| 3        | Clie_dirección     | Texto      | 50       | Dirección                         |
| 4        | Clie_fecha         | Fecha/hora |          | Fecha en que se realizó el pedido |
| 5        | Clie_usuario       | Texto      | 10       | Usuario                           |
| 6        | Clie_telefono      | Texto      | 20       | Teléfono del cliente              |
| 7        | Clie_estatus_envio | Númérico   | simple   | Estatus de envío del pedido       |

Tabla: t\_cliente\_pedido. Contiene los datos particulares de cada pedido.

| No. Campo | Nombre                | Tipo       | Longitud | Descripción                          |
|-----------|-----------------------|------------|----------|--------------------------------------|
| 1         | Clpp_fech             | Fecha/hora |          | Fecha del pedido                     |
| 2         | Id_clie_cliente       | Texto      | 10       | Clave del cliente                    |
| 3         | Id_prod_producto      | Texto      | 10       | Producto a enviar                    |
| 4         | Id_alma_almacen       | Texto      | 10       | Clave del almacén                    |
| 5         | Clpp_ref_cliente      | Texto      | 10       | RFC del cliente                      |
| 6         | Clpp_norden           | Texto      | 10       | Número de orden                      |
| 7         | Clpp_cantidad_pedido  | Numérico   | Doble    | Cantidad del pedido                  |
| 8         | Clpp_estatus_pedido   | Numérico   | Simple   | Estatus del pedido                   |
| 9         | Clpp_estatus_producto | Numérico   | Simple   | Estatus del producto                 |
| 10        | Clpp_fecha_factura    | Fecha/hora |          | Fecha de la facturación              |
| 11        | Clpp_cantidad_abas    | Numérico   | Doble    | Cantidad de producto                 |
| 12        | Clpp_orden_anterior   | Texto      | 10       | Orden anterior                       |
| 13        | Clpp_estatus_envio    | Numérico   | Simple   | Estatus de envío del pedido anterior |

Tabla: t\_cliente\_producto. Esta tabla contiene los productos y sus respectivas cantidades del pedido.

| No. Campo | Nombre             | Tipo       | Longitud | Descripción                   |
|-----------|--------------------|------------|----------|-------------------------------|
| 1         | Clpp_norden        | Texto      | 10       | Número de orden               |
| 2         | Id_clpr_producto   | Texto      | 10       | Clave del producto a enviar   |
| 3         | Clpp_fecha_factura | Fecha/hora |          | Fecha de la facturación       |
| 4         | Faor_cantidad_abas | Numérico   | Doble    | Cantidad de producto          |
| 5         | Faor_usuario       | Texto      | 10       | Usuario                       |
| 6         | Faor_estatus_envio | Numérico   | Simple   | Estatus de envío del producto |

Tabla: t\_formato. Contiene el formato que deben tener los archivos que los clientes envían sobre su pedido.

| No. Campo | Nombre          | Tipo     | Longitud | Descripción                           |
|-----------|-----------------|----------|----------|---------------------------------------|
| 1         | Id_form_formato | Numérico | Doble    | Clave del formato de archivo recibido |
| 2         | Form_formato    | Texto    | 50       | Formato de archivo recibido           |

Tabla: t\_funciones. Contiene las funciones específicas que un pedido conlleva.

| No. Campo | Nombre           | Tipo  | Longitud | Descripción                    |
|-----------|------------------|-------|----------|--------------------------------|
| 1         | Id_func_funcion  | Texto | 10       | Clave de la función a realizar |
| 2         | Func_descripción | Texto | 50       | Función a realizar             |

Tabla: t\_mirroring\_carga. Contiene los pedidos realizados (datos históricos).

| No. Campo | Nombre         | Tipo     | Longitud | Descripción          |
|-----------|----------------|----------|----------|----------------------|
| 1         | Mica_almacen   | Texto    | 50       | Clave del almacén    |
| 2         | Mica_producto  | Texto    | 50       | Producto             |
| 3         | Mica_cliente   | Texto    | 50       | Cliente              |
| 4         | Mica_desabasto | Númérico | Doble    | Estatus del producto |

Tabla: t\_producto. Es el catálogo de productos disponibles para envío.

| No. Campo | Nombre             | Tipo       | Longitud | Descripción              |
|-----------|--------------------|------------|----------|--------------------------|
| 1         | Id_prod_producto   | Texto      | 10       | Clave del producto       |
| 2         | Prod_descripcion   | Texto      | 50       | Descripción del producto |
| 3         | Prod_precio        | Númérico   | Doble    | Precio del producto      |
| 4         | Prod_fecha         | Fecha/hora |          | Fecha                    |
| 5         | Prod_usuario       | Texto      | 10       | Usuario                  |
| 6         | Prod_estatus_envio | Númérico   | Simple   | Estatus de envío         |

Tabla: t\_tablas. Contiene las características de las tablas que el sistema utiliza.

| No. Campo | Nombre       | Tipo     | Longitud | Descripción                  |
|-----------|--------------|----------|----------|------------------------------|
| 1         | Id_tab_tabla | Texto    | 8        | Clave de la tabla            |
| 2         | Tab_nombre   | Texto    | 30       | Nombre de la tabla           |
| 3         | Tab_ncampos  | Numérico | Simple   | Número de campos de la tabla |

Tabla: t\_tipo\_archivo. Guarda la descripción de los formatos de los archivos correspondientes al pedido del cliente.

| No. Campo | Nombre               | Tipo         | Longitud     | Descripción             |
|-----------|----------------------|--------------|--------------|-------------------------|
| 1         | Id_tiar_tipo_archivo | Autonumérico | Entero largo | Clave del archivo       |
| 2         | Tiar_descripción     | Texto        | 30           | Descripción del archivo |

Tabla: t\_tipo\_pedido. Contiene la descripción de los diferentes tipos de pedidos que se realizan.

| No. Campo | Nombre           | Tipo     | Longitud | Descripción            |
|-----------|------------------|----------|----------|------------------------|
| 1         | Id_tipo_pedido   | Numérico | Simple   | Clave del pedido       |
| 2         | Tipo_descripcion | Texto    | 30       | Descripción del pedido |

Tabla: t\_usuafunc. Contiene las funciones asignadas a cada usuario del sistema.

| No. Campo | Nombre          | Tipo       | Longitud | Descripción                |
|-----------|-----------------|------------|----------|----------------------------|
| 1         | Id_usuario      | Texto      | 10       | Clave del usuario          |
| 2         | Id_func_funcion | Texto      | 10       | Tipo de función a realizar |
| 3         | Func_fecha      | Fecha/hora |          | Fecha                      |
| 4         | Func_usuario    | Texto      | 10       | Usuario                    |

Tabla: t\_usuario. Contiene las características particulares de los usuarios del sistema.

| No. Campo | Nombre         | Tipo       | Longitud | Descripción            |
|-----------|----------------|------------|----------|------------------------|
| 1         | Id_usuario     | Texto      | 10       | Clave del usuario      |
| 2         | Usua_password  | Texto      | 10       | Contraseña del usuario |
| 3         | Usua_nombre    | Texto      | 50       | Nombre del usuario     |
| 4         | Usua_fecha     | Fecha/hora |          | Fecha                  |
| 5         | Usua_usuario   | Texto      | 10       | Nombre del usuario     |
| 6         | Usua_ubicación | Texto      | 1        | Lugar de trabajo       |

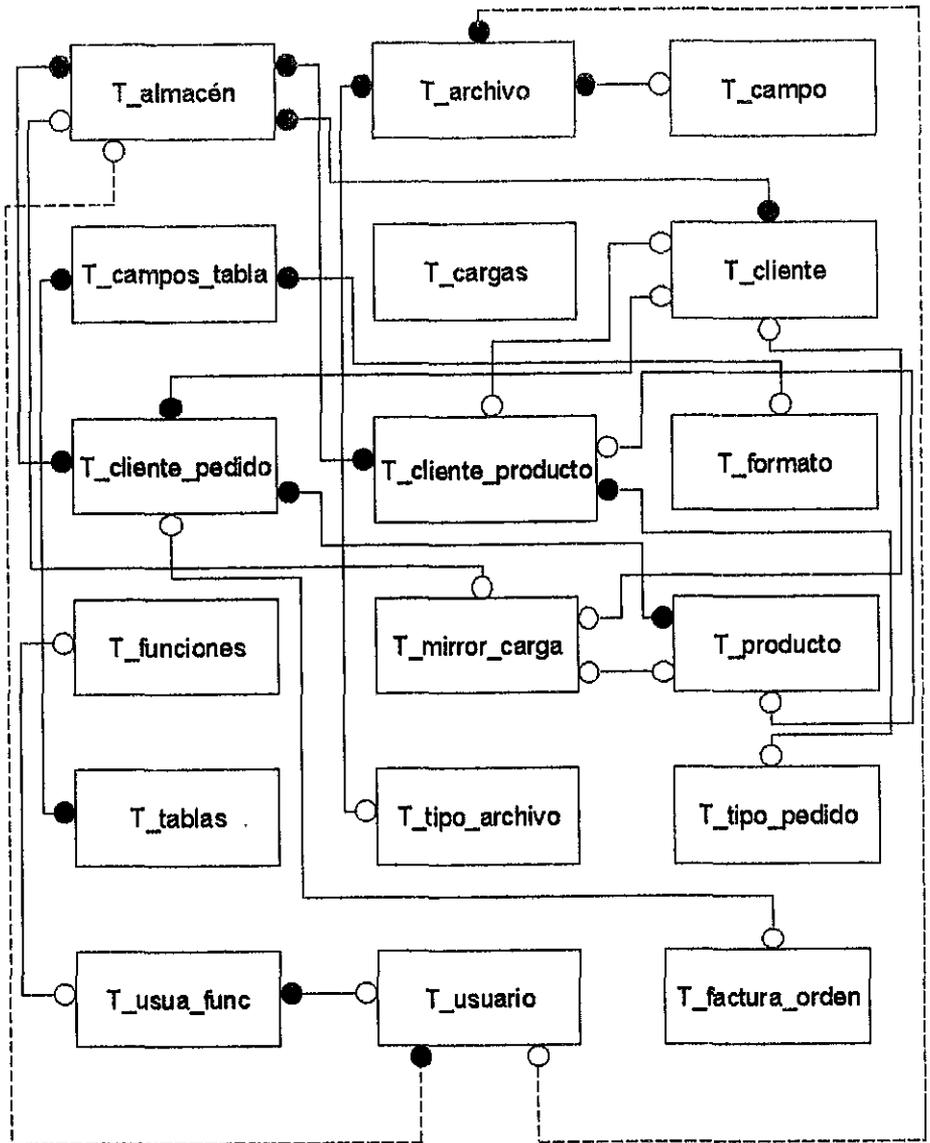


Figura 3.3.1. Relación entre tablas.

### **3.4. DISEÑO Y CONSTRUCCIÓN DE LA BASE DE DATOS**

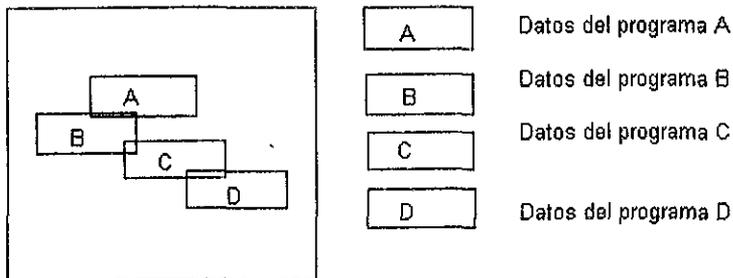
Los registros de una base de datos están situados generalmente en el archivo por el orden en que han sido introducidos, aunque podemos cambiarlos y ordenarlos como precisemos, por el campo que deseemos: alfabéticamente, numéricamente, etc.

A cada registro se le asigna automáticamente un número secuencial cuando se añade a la base de datos, por lo que se pueden identificar los registros por el número.

El contenido de una base de datos se obtiene combinando datos de todas las diferentes fuentes en una organización de tal manera que los datos estén disponibles para todos los usuarios y los datos redundantes puedan eliminarse o al menos minimizarse. En la figura 3.4.1 se muestra una base de datos como un recipiente de datos a ser compartidos por varios programas. El usuario podrá recobrar datos de cualquier parte de la base de datos ya que los archivos ahí almacenados están conectados directa o indirectamente.

El diseño de una base de datos se hace a partir del diagrama E-R e involucra los tres procesos siguientes:

1. Agrupamiento de los datos globales de una base de datos en uno o más registros de tipo conceptual (entidades).
2. Definición de las relaciones entre las entidades del modelo conceptual de la base de datos.
3. Transformación del modelo conceptual de la base de datos en un esquema mejorado que se puede describir mediante un DBMS específico para la implantación de la base de datos.



**Figura 3.4.1. Base de datos.**

### **Desarrollo de la Base de Datos**

La implantación de las tablas que forman la base de datos se creó en Microsoft Access se llevó a cabo de la siguiente manera:

- A.** Crear una base de datos en Microsoft Access.
- B.** Crear las tablas que componen a la base de datos.
- C.** Hacer las relaciones necesarias.

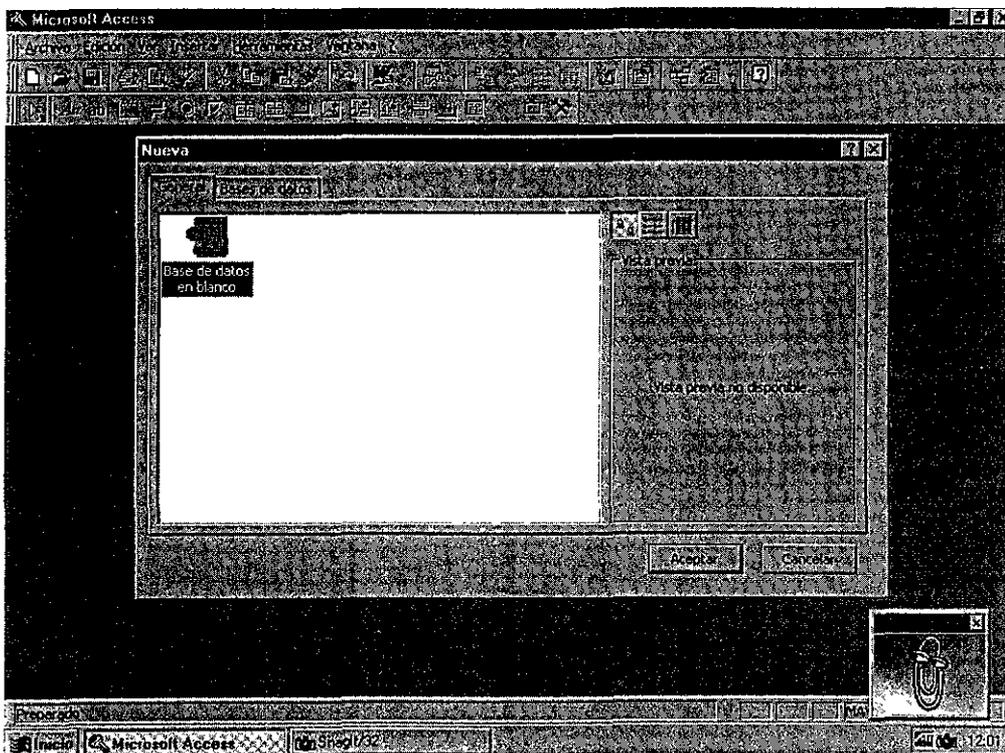
#### **A. Crear una Base de Datos en Microsoft Access**

Primeramente al entrar a Access aparece un menú como el que se muestra en la figura 3.4.2, seleccionamos la opción de Archivo y elegimos crear una nueva base de datos.



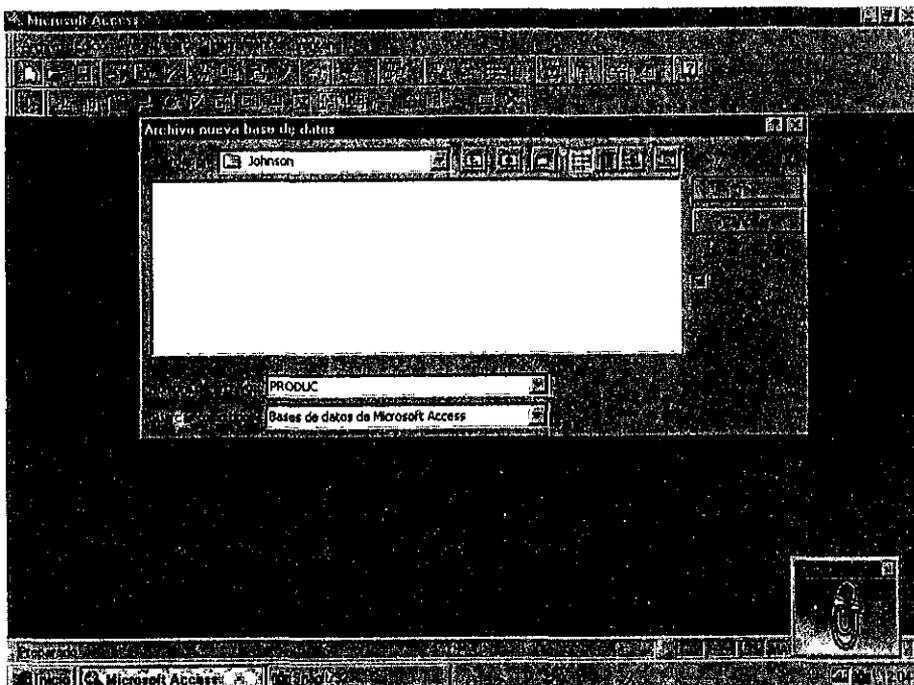
**Figura 3.4.2. Menú inicial de Microsoft Access.**

Posteriormente, al elegir la opción anterior aparecerá la ventana de asignación de bases de datos. Figura 3.4.3.



**Figura 3.4.3. Ventana de asignación de base de datos.**

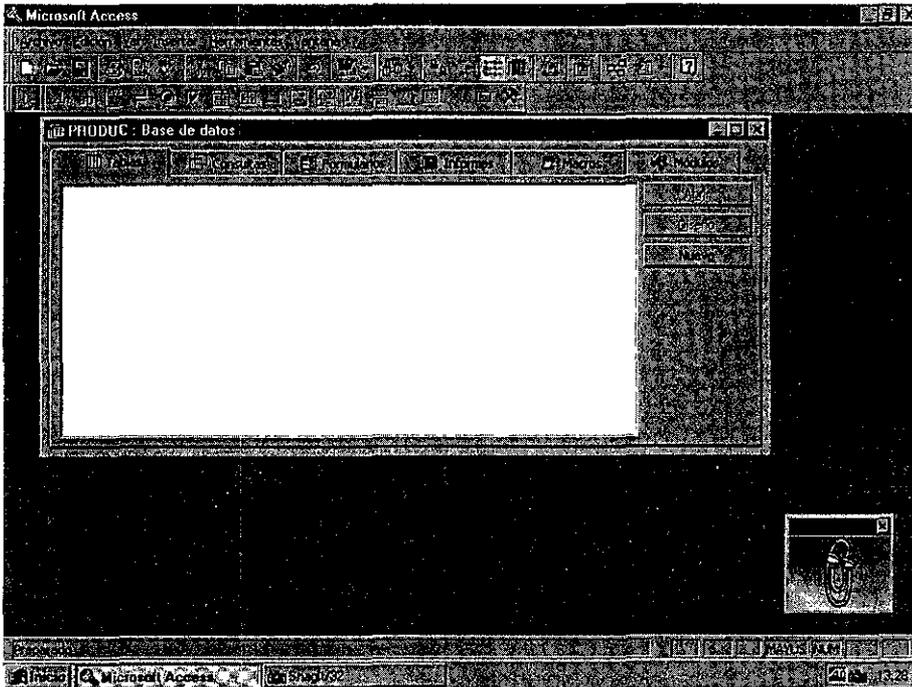
Esta opción nos va a pedir el nombre que le vamos a dar a la base de datos, el cual puede ser de 8 caracteres como máximo, en este caso es PRODUC. Una vez hecho esto, damos aceptar. Observar la figura 3.4.4.



**Figura 3.4.4. Creación de la base de datos PRODUC.**

En la figura 3.4.5 se muestra la ventana de la base de datos, con los respectivos elementos que la forman.

Microsoft Access nos presenta una ventana en la que nos muestra diferentes opciones: tablas, consultas, formas, reportes macros y módulos. Todas las opciones nos permiten crear un nuevo objeto o abrir uno ya existente.



**Figura 3.4.5. Ventana de la base de datos.**

Para utilizar una de las opciones de Microsoft Access se debe:

1. Elegir la opción deseada
2. Seleccionar uno de los tres botones siguientes:



para generar un nuevo objeto de la opción seleccionada



para ver la información de un objeto



para editar el objeto

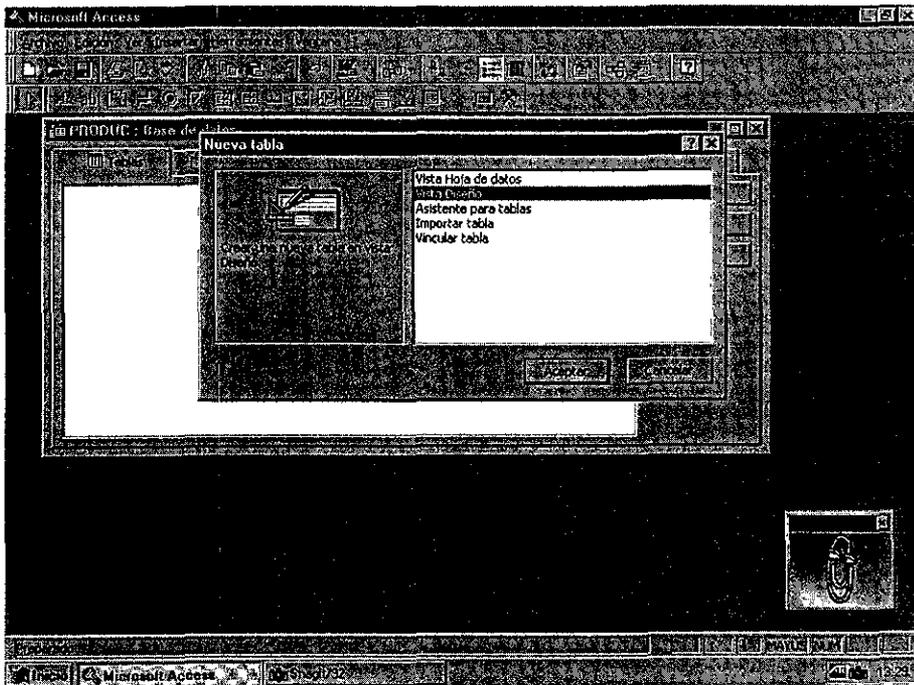
Ya disponemos de una base de datos vacía y entonces deberemos determinar y crear los diferentes objetos que la compondrán.

## **B. Creación de la Tabla**

Para poder crear la tabla, seleccionamos la opción de tablas y damos clic en la opción de **Nuevo**, como las demás, nos permite elegir si deseamos que el asistente nos auxilie, para crear el objeto con cierta estructura o no. Al seleccionar la opción de tabla en blanco, nos aparece la figura 3.4.6.

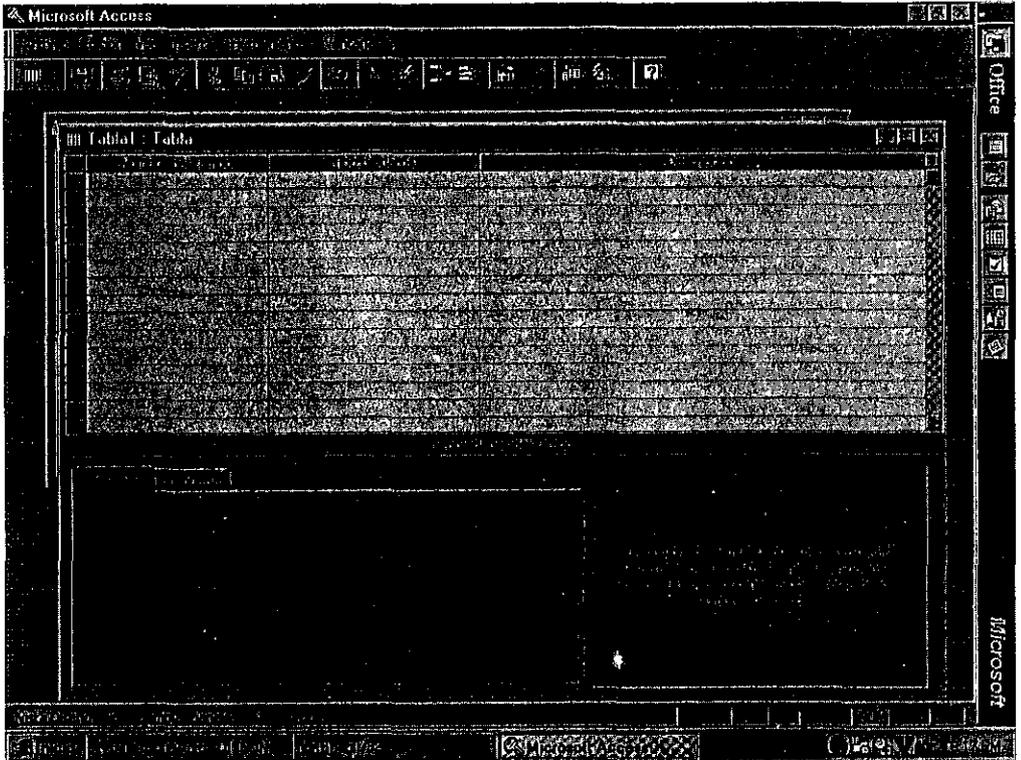
En la pantalla aparecerá el contenido que permite realizar varias tareas:

- **Modo de ver hoja de datos.** Esta opción permite visualizar la base de datos en forma de introducción de datos.
- **Modo de ver diseño.** Permite visualizar la tabla en modo diseño.
- **Asistente para tablas.** Esta opción permite crear una tabla de forma automática y asistida por el programa de ayuda y sugerencia de Access.
- **Importar tablas.** Esta opción permite importar tablas de otros paquetes.
- **Vincular tablas.** Permite vincular varias tablas de datos comunes o relacionados.



**Figura 3.4.6. Tareas para la elaboración de una tabla.**

Nosotros escogimos la opción de diseño de tabla, y aparece la pantalla de la figura 3.4.7.



**Figura 3.4.7. Diseño de una tabla.**

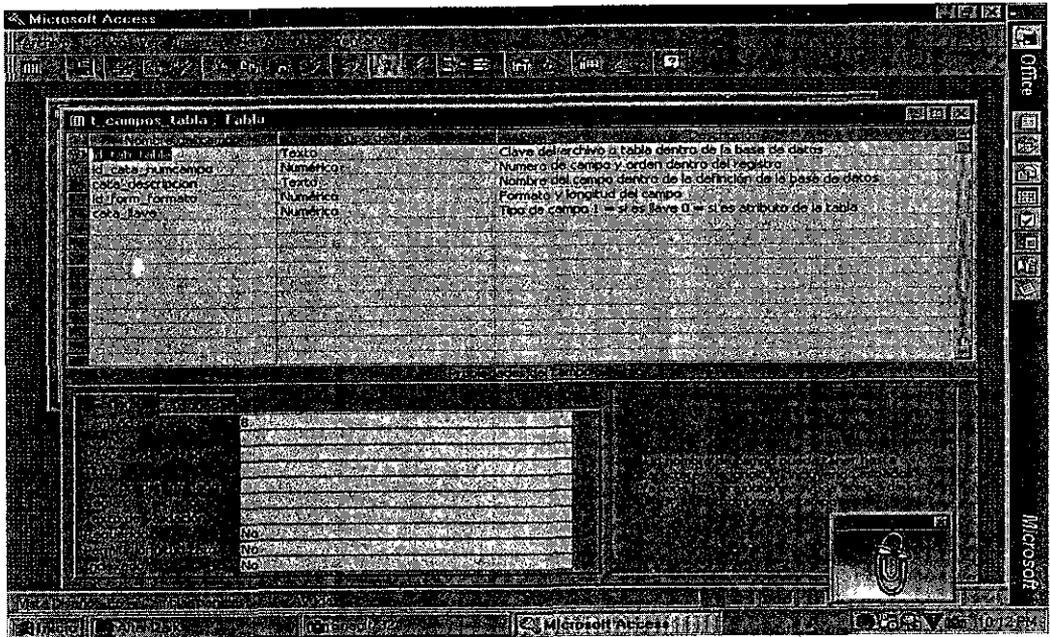
En la parte superior se tiene la sección de declaración de campos que esta compuesta por tres columnas, la primera para darle un nombre al campo, la segunda para darle el tipo de datos que deseamos y la tercera para dar el descriptivo de lo que se va a introducir en el campo.

En la parte inferior tenemos la sección de propiedades de los campos que nos permite adecuar las propiedades a nuestras necesidades.

Al entrar en esta opción también contamos con una barra de herramientas de la cual, utilizaremos principalmente el icono de especificar la llave primaria que debemos presionar después de seleccionar el o los campos que deseamos sea la llave de

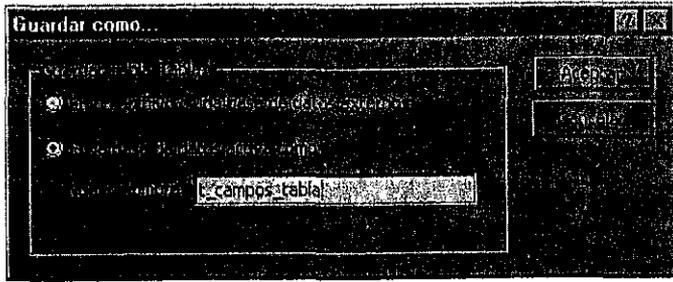
nuestra tabla. Asimismo, vamos a utilizar el icono de guardar el objeto activo, con el cual salvaremos con nombre nuestra tabla o bien las modificaciones realizadas a una ya existente.

La figura 3.4.8 muestra como creamos la tabla de líneas que forma parte de nuestro sistema. En esta tabla se muestran los campos que la forman con sus respectivas características.



**Figura 3.4.8. Tabla de catálogo de líneas de la base de datos.**

Una vez introducidos los campos de la tabla y validados todos los diferentes tipos de datos, se procede a cerrar la tabla en la cual aparece la pantalla de la figura 3.4.9 en donde se proporciona el nombre con el cual se quiere guardar.



**Figura 3.4.9. Ventana que muestra Access al salvar una tabla.**

Una vez explicada la manera en como se creó la tabla de t\_campos\_tabla se procedió a realizar de manera similar las otras tablas que conforman a la base de datos, en la figura 3.4.10 se muestra el nombre de todas las tablas creadas.

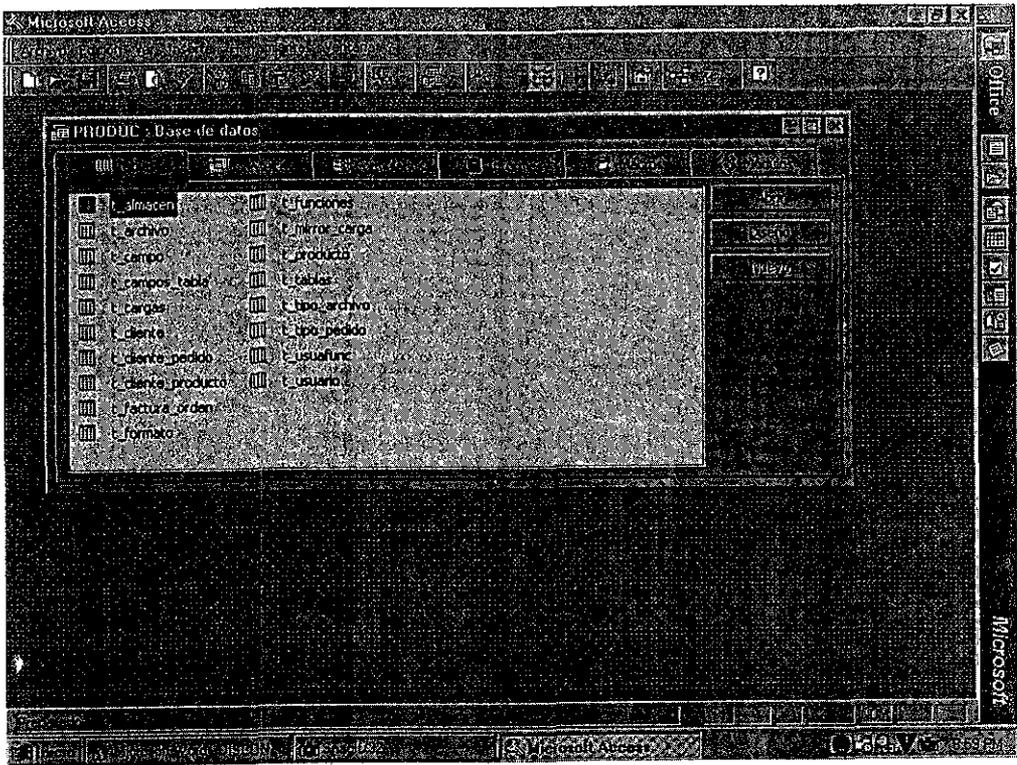
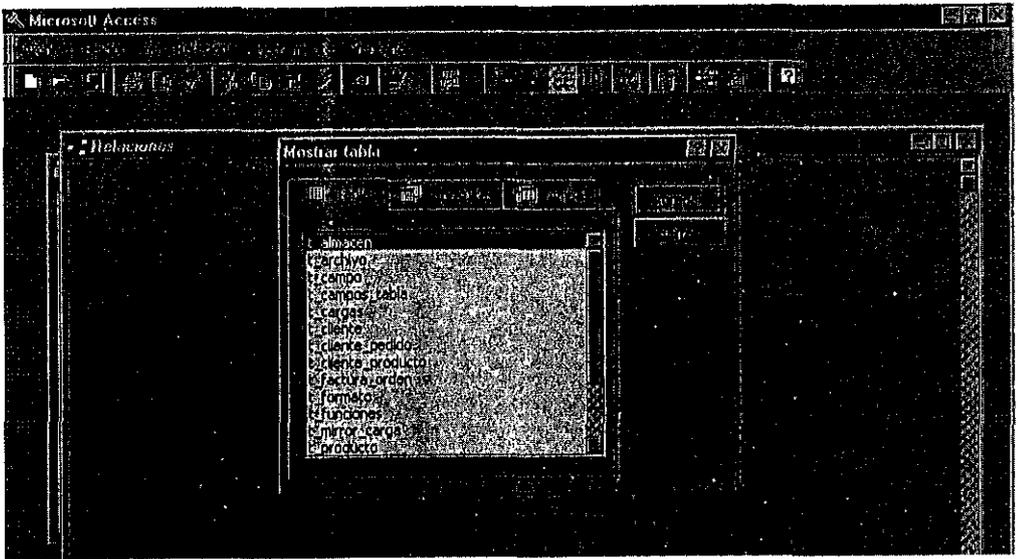


Figura 3.4.10. Ventana que muestra todas las tablas de la base de datos.

### C. Creación de las Relaciones Necesarias

En el menú de Microsoft Access se tiene la opción de relaciones, al presionar este comando nos aparece en la pantalla que se muestra en figura 3.4.11, en el cual existe la posibilidad de ir agregando las tablas que sean necesarias para determinar las relaciones que deseamos.



**Figura 3.4.11. Ventana con la que se agregan las tablas.**

Las tablas relacionadas se representan mediante una línea que conecta los campos coincidentes. Microsoft Access utiliza las relaciones para crear automáticamente combinaciones en las nuevas consultas y para relacionar los registros mostrados en los subformularios. Las tablas deben estar relacionadas por campos equivalentes (llaves). No es necesario que los campos equivalentes tengan el mismo nombre, pero sí deben contener el mismo tipo de datos. Normalmente, las tablas se relacionan vinculando el campo de clave principal de una tabla con un campo coincidente de otra tabla. Las relaciones de las tablas de nuestra base de datos se muestran en la figura 3.4.12.

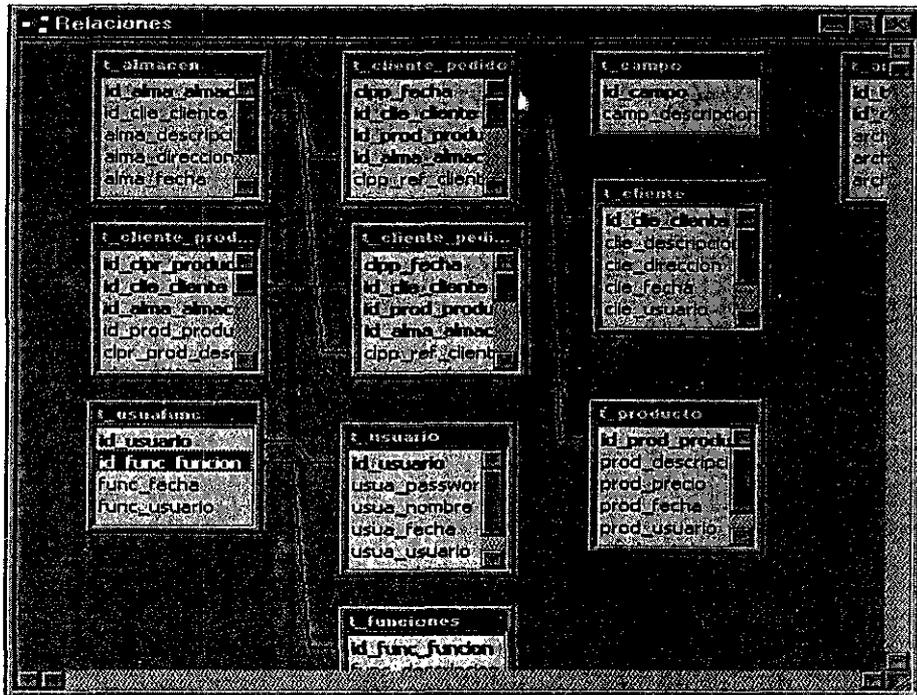
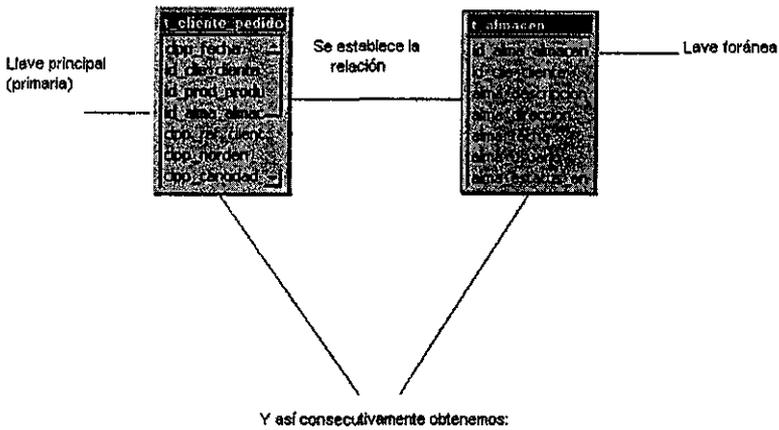


Figura 3.4.12. Construcción de las relaciones de las bases de datos.

### 3.5. DISEÑO Y CONSTRUCCIÓN DEL FRONT – END

Las aplicaciones en un ambiente Windows se pueden desarrollar mediante técnicas, herramientas de programación y el conocimiento de algún lenguaje de programación de alto nivel. En nuestro caso particular, tenemos el Front-End del sistema en un ambiente Windows, utilizando como herramienta de desarrollo Visual Basic 4.0 con el objetivo de presentar un ambiente gráfico y de fácil interacción con el usuario.

En un entorno gráfico el usuario interactua con los objetos que hay en la pantalla, ejecutando con ellos un evento o una acción que hace referencia a un procedimiento y *da respuesta a ese evento o acción*. El ambiente de Windows nos proporciona herramientas para el uso óptimo del sistema, por ejemplo:

**Multitarea.** Los equipos han logrado atender varias tareas simultáneamente a una alta velocidad.

**Intercambio de datos.** Intercambia información entre las aplicaciones mediante **Dynamic Data Exchange** (proceso estandarizado de intercambio).

**Memoria.** Windows es un sistema operativo que utiliza técnicas para la asignación de memoria no limitándose en tener un programa en memoria.

#### 3.5.1. Fases para el Desarrollo de Aplicaciones con Visual Basic

De manera genérica en el desarrollo de una aplicación con Visual Basic, se utilizan las siguientes fases:

- Establecer la forma o ventana y sus propiedades.
- Añadir objetos (controles) y sus propiedades.

- Escribir el código de cada objeto y de la ventana.
- Implantar procedimientos asociados a los eventos.
- Integrar las formas utilizadas en el sistema.
- Compilación del programa.
- Revisar diseño.
- Guardar el programa.
- Ejecución del programa.

El diseño del Front-End comienza con una ventana que es parte del medio homogéneo que proporciona Windows, es un contenedor de todos los objetos a utilizar, definiendo formas, colores y tamaños de los mismos. La forma posee propiedades, eventos y métodos que el programador y diseñador pueden utilizar para adecuarla y tener un mejor desarrollo del sistema. Véase la figura 3.5.1.1.

## Forma

### Propiedades:

**BackColor.** Permite cambiar de color el fondo de la forma, dando un color deseado en la tabla de colores.

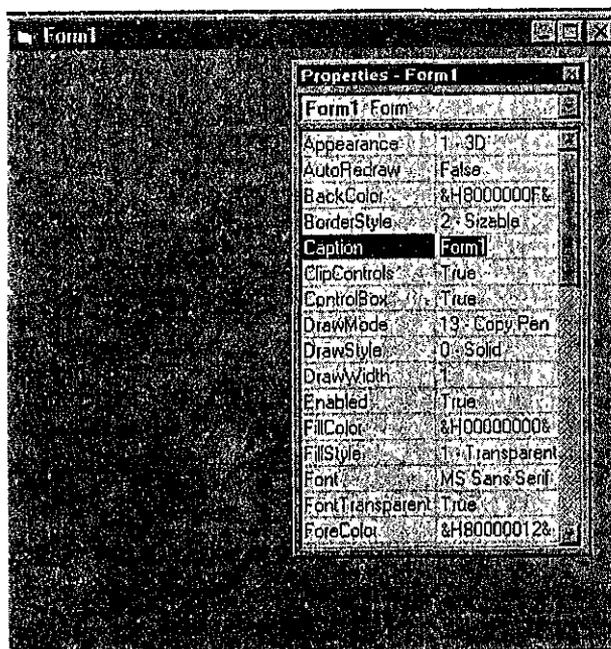
**Caption.** Permite dar un mensaje en el encabezado de la forma indicando al usuario en que pantalla se encuentra trabajando.

**Name.** Permite dar un nombre a la forma, por estandar el nombre comienza con frm más alguna descripción de la función de la forma, por ejemplo, frm\_act\_prod, significa que la forma tiene como función la actualización de los productos.

**Font.** Permite dar un formato especial a la fuente utilizada en la forma.

**Maxbutton.** Propiedad que permite agrandar en su máximo a la forma.

**Minbutton.** Permite reducir la forma en su mínima escala.



**Figura 3.5.1.1. Forma o ventana, lugar donde se conjuntan los controles u objetos.**

### Eventos:

**Load.** Primer evento que se realiza cuando trabajamos con una forma, aquí se inicializan los procedimientos y funciones que se requieren al cargar una forma solicitada.

**Mousemove.** Es un evento en el que al accionar el movimiento del mouse se hace referencia a un cierto procedimiento, por ejemplo, al mover el mouse sobre un icono con dicho evento se invoca al procedimiento de mensajes el cual muestra el mensaje de funcionamiento del icono.

Los controles utilizados en el sistema son los siguientes:

## Image

Es un control gráfico que permite por medio de imágenes dar presentación al sistema, la figura que se muestra puede ser un archivo \*.bmp, un icono o un archivo Windows metafile. Véase figura 3.5.1.2.

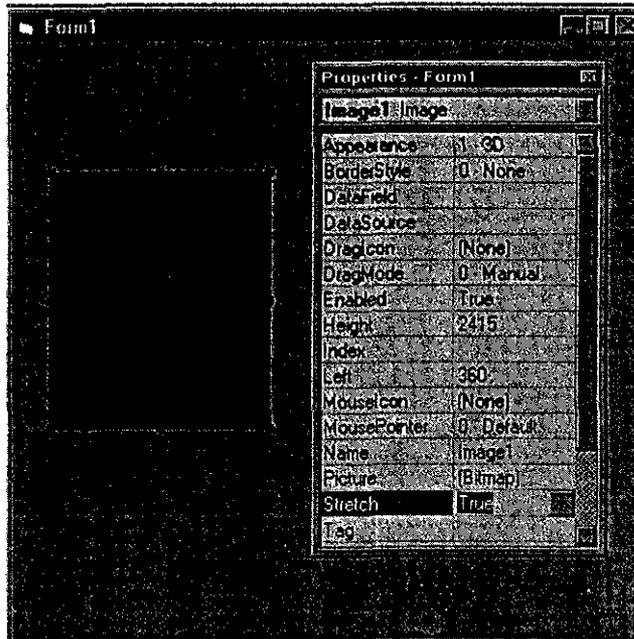


Figura 3.5.1.2. Control Image.

### Propiedades:

**Picture.** Indica el nombre y ruta de la imagen que se desplegará en el momento de la ejecución del sistema.

**Name.** Nombre del objeto que se utilizará en el código del sistema.

**Stretch.** Propiedad que permite dar una escala a la imagen.

**BorderStyle.** Permite dar un tipo de borde a la imagen, ésta puede ser en tercera dimensión.

## Textbox

Es un objeto muy importante, pues nos permite introducir la información a la base de datos, ya sea para actualizarla, modificarla o presentarla en pantalla. También es importante mencionar que este objeto por sí sólo es incapaz de actualizar la base de datos pues requiere de eventos y propiedades para realizar dicha operación. Véase la figura 3.5.1.3.

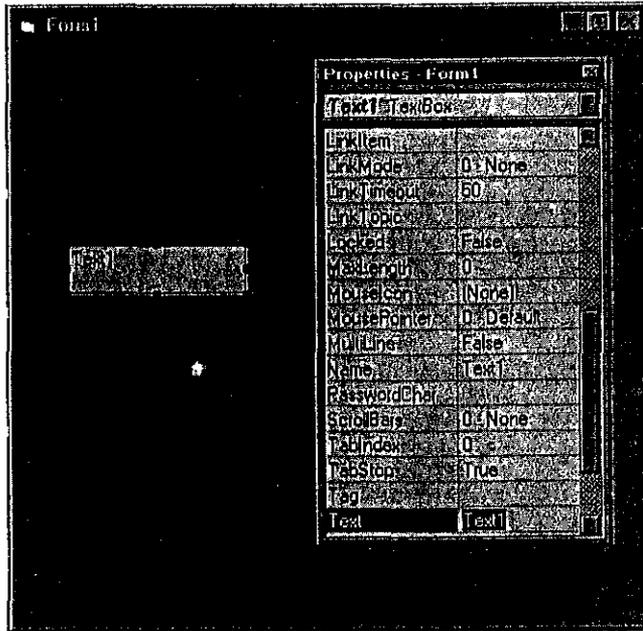


Figura 3.5.1.3. Control Textbox.

### Propiedades:

**Datafiled.** Cuando la caja de texto se asocia a un objeto "Data Control", siendo ésta una forma de enlazar la información con la base de datos.

**Fontname.** Tipo de fuente que se le asigna a la caja de texto.

**FontColor.** Color que se le asigna a la letra de la caja de texto.

**Maxlength.** Propiedad que permite dar un límite en el tamaño máximo de los caracteres en la caja de texto.

**Name.** Nombre de la caja de texto en el código del programa.

**Visible.** Propiedad que permite ocultar o hacer visible al campo, dependiendo de lo requerido en el diseño.

### Eventos:

**Click.** Evento que se ejecuta cuando el usuario da un clic con el mouse sobre el control, el cual se asocia con un procedimiento específico.

**KeyPress.** Evento que se ejecuta cuando se oprime una tecla específica, el evento va relacionado con un procedimiento y se indica con cuál tecla se ejecutará.

### Commandbutton

Control que se utiliza para iniciar, interrumpir, procesar o finalizar un procedimiento en el sistema. Véase la figura 3.5.1.4.

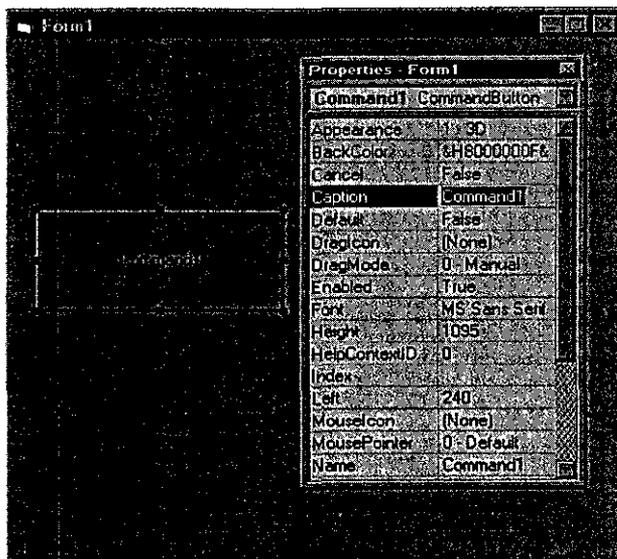


Figura 3.5.1.4. Control Commandbutton.

### Propiedades:

**Caption.** Mensaje que se despliega en el commandbutton al momento de la ejecución.

**Enabled.** Habilita al botón para su uso.

**Name.** Nombre que se le asigna al commandbutton en el diseño.

### Evento:

**Click.** Al dar un clic al mouse sobre el commandbutton se ejecuta un procedimiento.

### Sstab

Es un control que permite utilizar la presentación de varias pantallas en una sola forma, es decir, permite crear una serie de tabuladores, los cuales se pueden incrementar dependiendo de los requerimientos del sistema. Véase la figura 3.5.1.5.

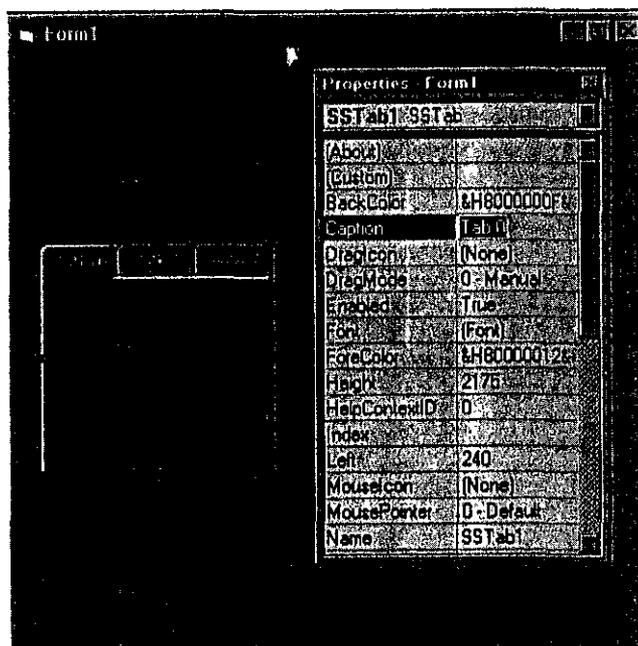


Figura 3.5.1.5. Control Sstab.

**Propiedades:**

**Font.** Propiedad que permite especificar tipo, tamaño y estilo de fuente dentro del objeto sstab.

**Caption.** Propiedad que indica el nombre de la información que se presenta en cada tabulador, el nombre se visualiza en la pestaña de cada tabulador.

**Style.** Formato de presentación del sstab, ya sea en forma de Msoffice o estilo Windows, los cuales no afectan la presentación.

**TabsOrientation.** Solo se emplea si se desea cambiar la orientación de los folders.

**TabsperRows.** Propiedad donde se indica cuantos folders por renglón debe presentar el tab.

**Eventos:**

**Click.** Es un evento que permite el desplazamientos por los diferentes folders, se realiza al dar un clic con el botón izquierdo sobre la pestaña deseada.

**Crystal Reports**

Es un control especial de Visual Basic que permite imprimir información de la base de datos sin la necesidad de hacer procedimientos complejos. Crystal Report se debe situar en la forma de donde se quiere obtener la información a imprimir. Además se requiere de la creación de un formato de reporte, es decir, un archivo \*.RPT que consta de campos especiales (número de página, fecha, etc.), campos de las bases involucrados en el reporte, campos de texto y de fórmulas. Las fórmulas son condiciones del reporte generadas desde el código de la forma en donde se encuentra el control de Crystal Report. Véase la figura 3.5.1.6.

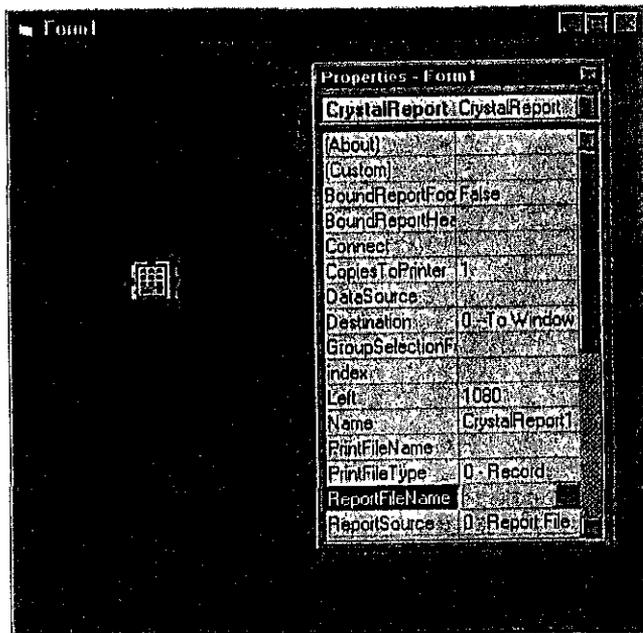


Figura 3.5.1.6. Control Crystal Report.

### Propiedades:

**ReportSource.** Propiedad que indica el origen de los datos de un archivo Report.

**ReportFileName.** Nombre del archivo reporte que se invocará al momento de su ejecución.

### Sspanel y Frame

Permite agrupar objetos de una misma entidad dándoles una presentación en tercera dimensión. Dentro de este objeto se colocan los objetos para darle relieve y un marco en tercera dimensión dependiendo del objeto empleado. Véase la figura 3.5.1.7.

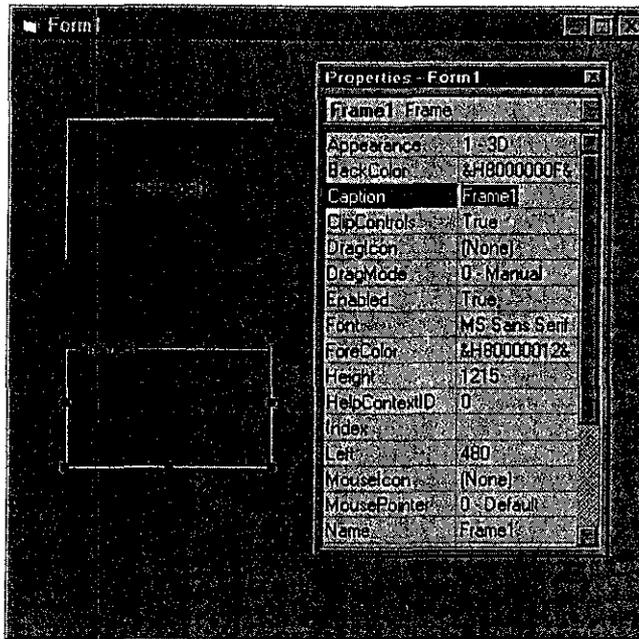


Figura 3.5.1.7. Controles Sspanel y Frame.

### Propiedades:

**Caption.** Nombre que se asigna al objeto.

**BackColor.** Color que se especifica para el fondo del objeto.

**BvelOuter.** Propiedad que permite dar el realce en tercera dimensión.

### Graph

Es un control que permite diseñar gráficas interactivas con las formas del sistema. Se puede decir que es la liga entre la aplicación y el **Graphics Server** (Servidor Gráfico) para cargar las librerías. Véase la figura 3.5.1.8.

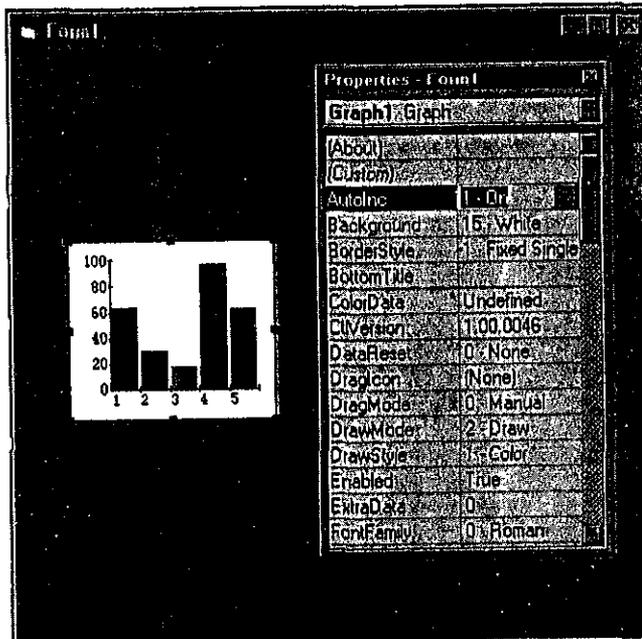


Figura 3.5.1.8. Control para gráficas.

**Propiedades:**

**Graphdate.** Se especifica con esta propiedad un conjunto de datos para representarlos en forma gráfica.

**Graphtype.** Propiedad que permite especificar el tipo de gráfica la cual puede ser: polar, Gantt, etc.

**Visible.** Propiedad que permite tener visible la gráfica en tiempo de ejecución.

**Numpoints.** Define el número de datos en cada intervalo de la gráfica.

**Numsets.** Define el número de intervalos en la gráfica.

## Label

Este objeto es empleado en todos los módulos del sistema, ya que permite visualizar etiquetas de información que corresponde a cada textbox, es decir, se localiza a un costado de textbox para informar de qué tipo de dato se trata. Véase la figura 3.5.1.9.

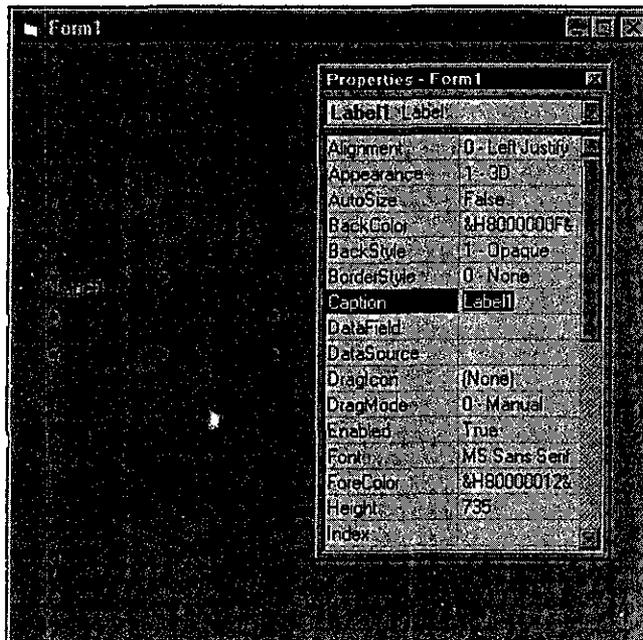


Figura 3.5.1.9. Control Label.

### Propiedades:

**Caption.** Propiedad que permite poner un mensaje de identificación a un campo. El control del label en la mayoría de los casos se localiza a un costado del campo a identificar.

**Font.** Propiedad que indica el tipo, tamaño, color y estilo de la fuente.

## Sscommand

Este objeto permite crear una presentación del botón en tercera dimensión, se puede incluir imágenes en la creación de menús por medio de botones, obteniendo una mejor presentación y un fácil manejo para el usuario. Véase la figura 3.5.1.10.

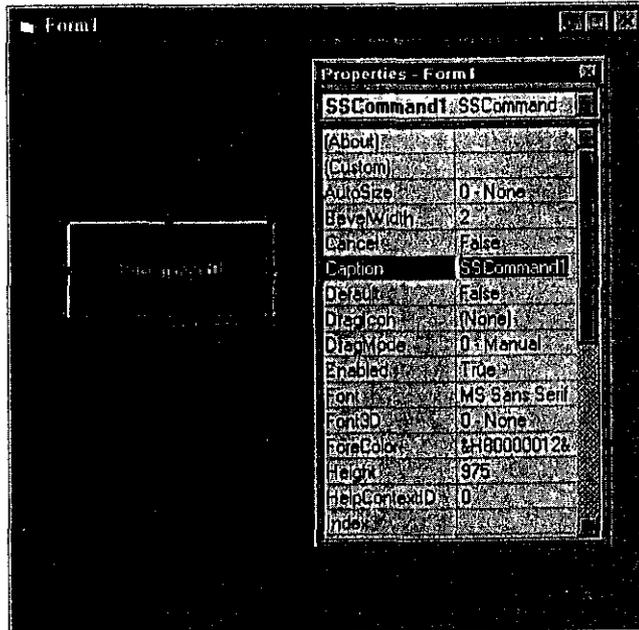


Figura 3.5.1.10. Control Sscommand.

### Propiedades:

**Caption.** Se coloca un mensaje para identificar la función del control, si así se requiere.

**Name.** Nombre que se le asigna al control para identificarlo en el código del sistema.

**Picture.** Es la imagen y ruta de una imagen que se pondrá en el ssccommand en sustitución al caption, es decir, representará por medio de la imagen la función del botón.

## Truegrid

Es un control anexo a Visual Basic que maneja la interface con la base de datos y la aplicación. Con Truegrid se puede visualizar en una ventana información de la base de datos de modo **browse** (navegación) o **edit** (edición), insertando solo aquellas columnas que se quiere visualizar. Véase la figura 3.5.1.11.

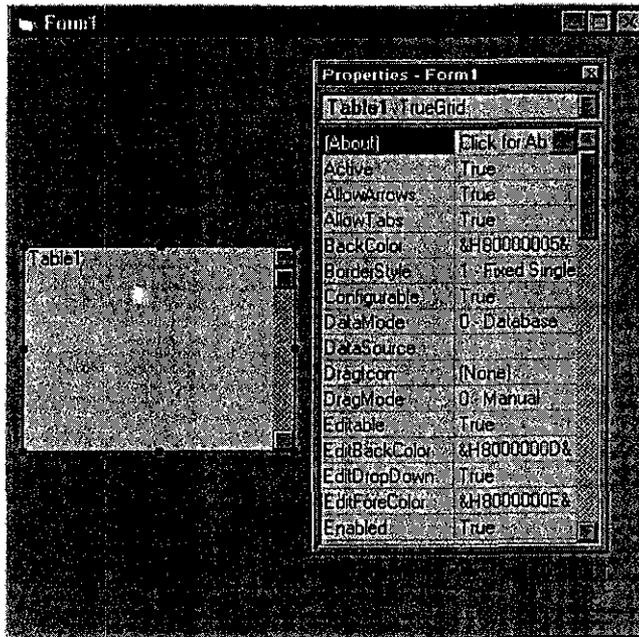


Figura 3.5.1.11. Control Truegrid.

### Propiedades:

**Rows.** Retorna el número de renglones en el truegrid.

**RowIndex.** Retorna el número de índice correspondiente al renglón actual.

**Editactive.** Es una propiedad que permite habilitar el modo edit en el truegrid o deshabilitarlo para mostrarlo en modo browse.

**Refresh.** Permite actualizar la información ya sea en columnas o renglones.

**Extendrightcol.** Permite extender el scrollbar a la derecha correspondiente a las columnas.

#### **Eventos:**

**Rowchange.** Indica los cambios modificados a un renglón de la tabla.

#### **Combobox**

Es un control que se puede utilizar de dos formas:

- Caja de texto. En ella se agrega información para formar una lista de selección para el usuario, es decir, el usuario introduce información inexistente en la base de datos para dicho campo.
- Seleccionador de una opción en una lista especificada en la base de datos.

Las propiedades del combobox y su representación en la forma se presentan en la figura 3.5.1.12.

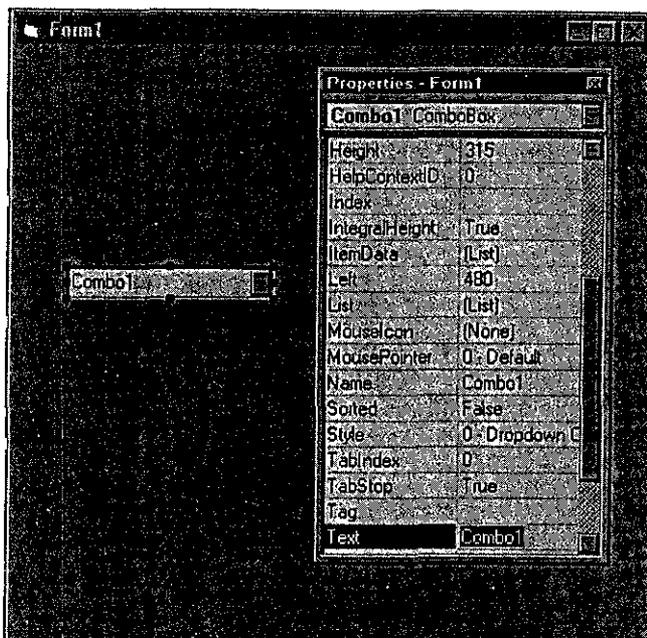


Figura 3.5.1.12. Control Combotext.

#### Propiedades:

**Font.** Propiedad que permite asignar el tipo, tamaño, estilo de una fuente.

**Itemdata.** Característica que se emplea para almacenar valores de referencia según las opciones almacenadas.

**List.** Es el almacenamiento de las opciones que se emplean dentro del combo.

**Name.** Es el nombre que identifica al combotext en el código del sistema.

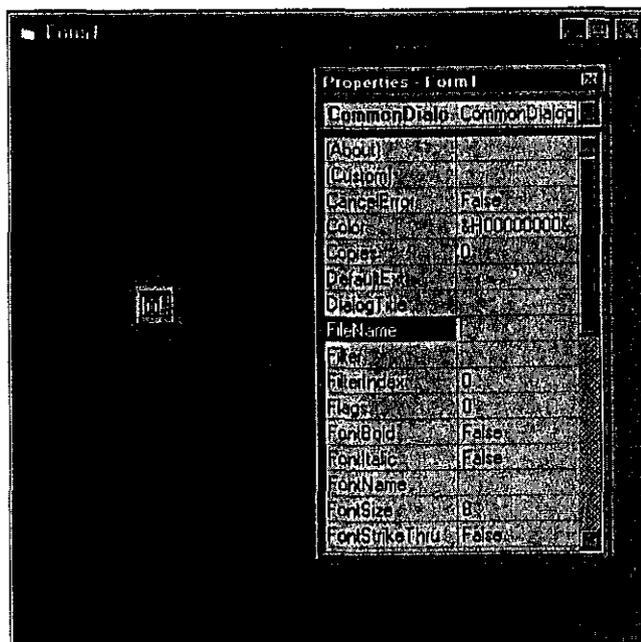
**Style.** Permite indicar cuál será el tipo de presentación en la pantalla. Por mencionar, el tipo que se utiliza en el sistema en el **Dropdownlist** (lista de selección de elementos de no-edición), ya que no permite en tiempo de ejecución agregar opciones en la lista ya establecida.

### Eventos:

**Click.** Evento que se ejecuta al oprimir el botón izquierdo del mouse sobre una opción de la lista del combotext.

### Commondialog

Es un control que permite invocar la ventana de directorios para que el usuario seleccione la ruta de un archivo texto para su exportación o importación. Véase figura 3. 5.1.13.



**Figura 3.5.1.13. Control Commondialog.**

### Propiedades:

**Name.** Es el nombre del control que se utiliza en el código del sistema.

**DialogTitle.** Es el mensaje que se despliega en la ventana de directorios.

**Fontsize.** Especifica el tamaño de la fuente.

**Initdir.** Es el drive por default.

**Cancelerror.** Retorna el valor de un error generado cuando el usuario selecciona el botón de Cancelar.

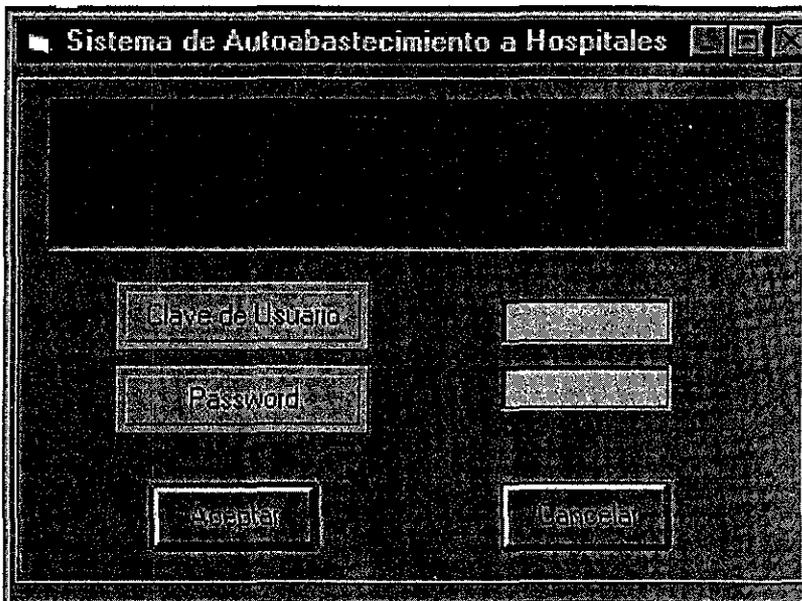
**Filter.** Selecciona sólo los archivos especificados en esta propiedad.

### **Pantallas Creadas para el Sistema**

Hasta el momento se han descrito los objetos utilizados en el sistema, a continuación describiremos las pantallas del sistema.

#### **Pantalla de Acceso**

La pantalla de acceso donde el usuario introduce nombre y contraseña del mismo, para poder acceder al sistema. En la pantalla de diseño se utilizaron los objetos textbox, sspanel, frame, ssccommand e image. Véase la figura 3.5.1.14.



**Figura 3.5.1.14. Pantalla de acceso.**

**Usos:**

**Textbox.** Se utiliza para insertar el nombre de usuario y su contraseña, este último utiliza la propiedad passwordchar para dejar en asteriscos los caracteres tecleados y así dar una seguridad al sistema.

**Sscommand.** Se emplean para cancelar o aceptar la operación de insertar los datos mediante los textbox, cuando se da aceptar se valida que el usuario se encuentre en la base de datos y posteriormente se obtiene el nivel de acceso, de lo contrario se mostrará un mensaje de "Acceso Denegado".

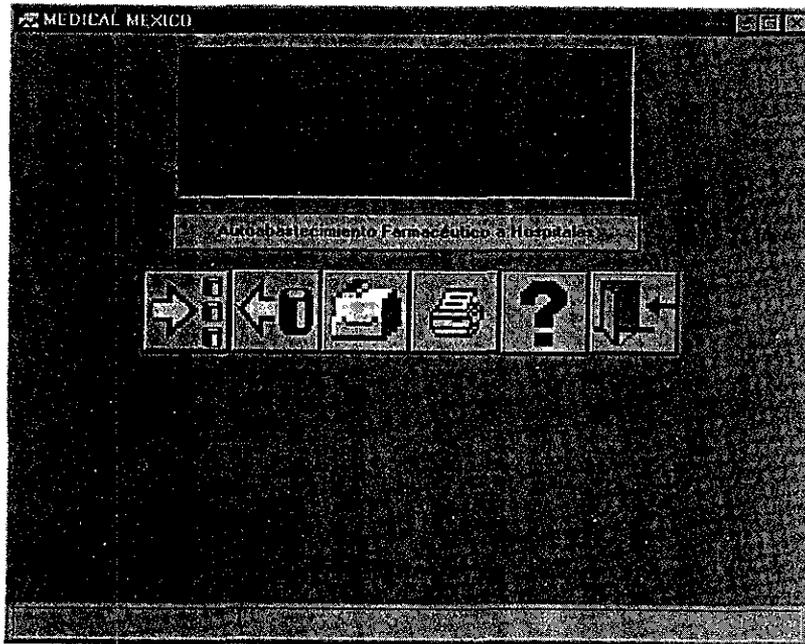
Por mencionar un ejemplo de acceso de nivel, si el nivel de acceso es 0000 el usuario tiene acceso a todos los módulos del sistema, si el nivel de acceso es 0010 se tiene acceso a la parte de Importación, si el acceso es 0020 sólo se tiene acceso a la parte de Exportación y así sucesivamente.

**Image.** Se utiliza para visualizar el logotipo de la empresa en el sistema.

**Sspanel y Frame.** Son empleados para agrupar a los objetos utilizados.

**Pantalla Principal**

La siguiente pantalla muestra los módulos que constituyen al sistema, en esta pantalla se utiliza image, sspanel, sscommand y commandbutton. Véase la figura 3.5.1.15.



**Figura 3.5.1.15. Pantalla principal.**

**Usos:**

**Image.** Permite visualizar el logotipo del sistema.

**Sspanel.** Se utiliza como un letrero en tercera dimensión y en él se despliega el nombre del sistema.

**Sscommnad.** Se utilizan para representar los módulos que existen en el sistema, con la propiedad picture se visualiza una imagen que identifica la función del módulo. Por ejemplo, el icono que tiene la impresora representa al módulo de reportes.

**Sscommnad.** Tienen un evento click con el cual ejecutan una acción para invocar a los submódulos. Para cada icono, es decir, para el módulo de reportes, se tienen cuatro reportes y una gráfica.

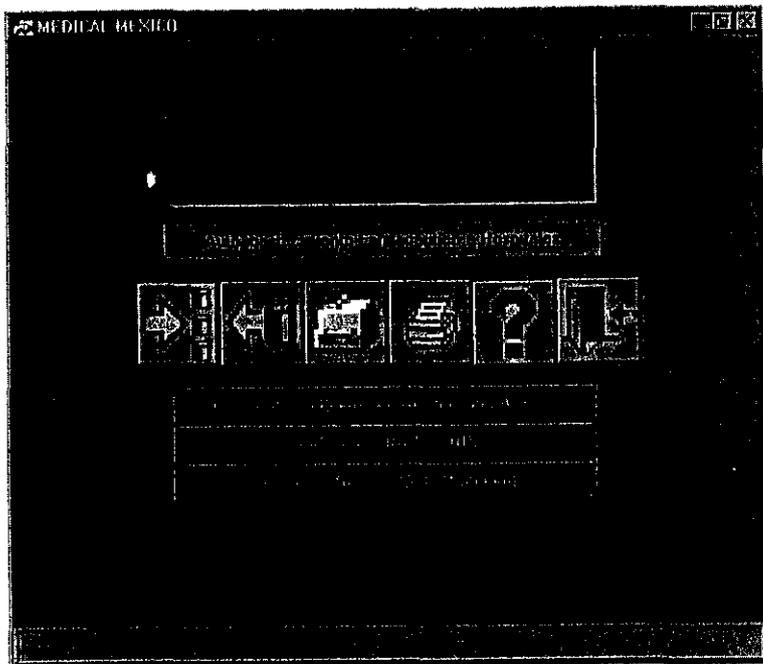
**Commandbutton.** Se utiliza para presentar un mensaje de los submódulos, que corresponden a un icono seleccionado y además ejecuta una acción particular al oprimir el botón izquierdo del mouse por el evento click.

## Pantalla de Importación

En esta pantalla se encuentra el módulo de importación, consiste de tres submódulos:

- Archivos de productos consumidos por almacén.
- Archivos de sistema SUN.
- Archivos de movimiento de productos.

Véase la figura 3.5.1.16.



**Figura 3.5.1.16. Pantalla de importación.**

En la pantalla de Archivo de Productos Consumidos por Almacén, se utilizaron los objetos sstab, truegrid, textbox, ssccommand, label y frame. La pantalla muestra dos

folders, en el primero, se pide el destino del archivo, mientras que en el segundo se asigna el tipo de archivo. Véase la figura 3.5.1.17.

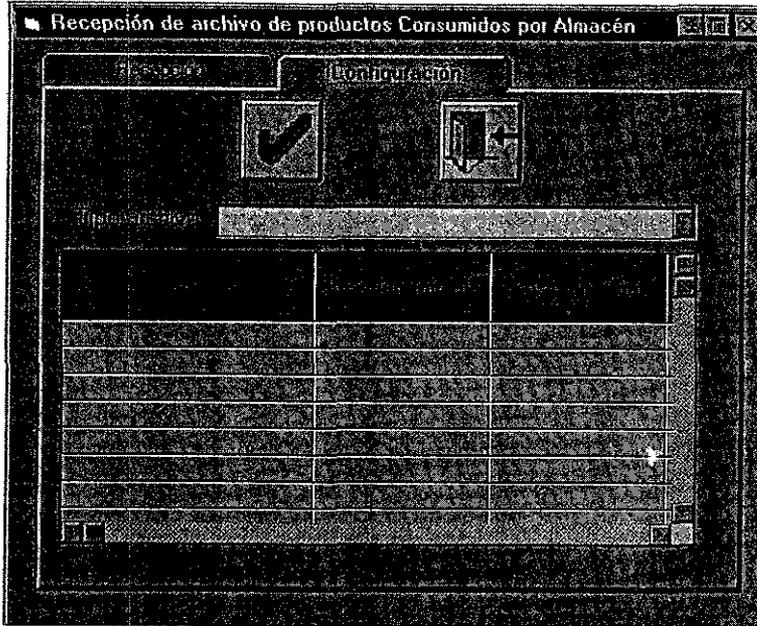


Figura 3.5.1.17. Pantalla de archivo de productos consumidos por almacén.

**Usos:**

**Sscommand.** Se emplean para aceptar la información introducida (icono representado por una paloma) o salir de la pantalla (icono representado por una puerta).

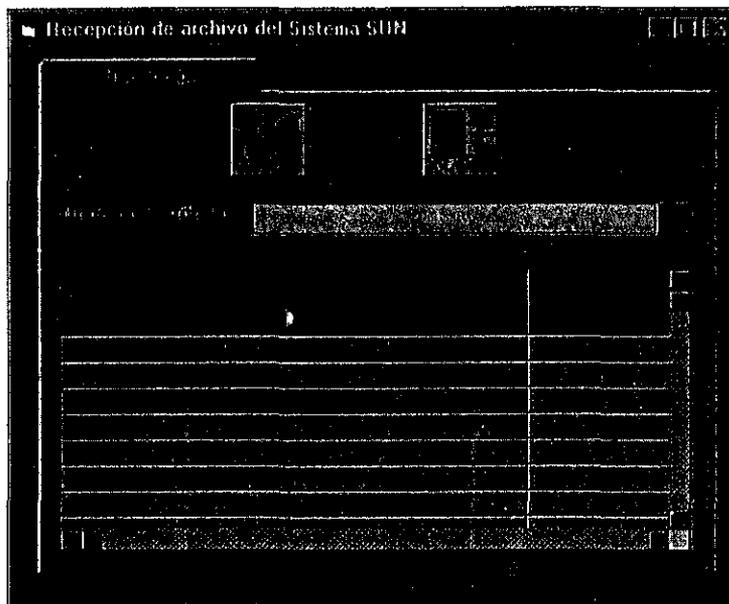
**Truegrid.** Se emplea para mostrar la información de la base de datos de los campos de un archivo texto insertado con anterioridad, indicando la posición inicial y final de los campos que constituyen a dicho archivo.

**Combotext.** Se emplea para seleccionar de una lista existente en la base de datos la configuración del archivo. Los tipos de configuración son: por posiciones, por tabuladores y comas.

**Sstab.** Se emplea para los folders que existen en esta pantalla.

**Label.** Es la etiqueta correspondiente al combotext.

Los submódulos Archivos de Sistema SUN y Recepción de Archivos de Productos Consumidos por Almacén tienen la misma pantalla de recepción la cual se describe a continuación. Véase la figura 3.5.1.18.



**Figura 3.5.1.18. Archivos de recepción.**

**Usos:**

**Sscommand.** Se emplean para aceptar los datos introducidos al presionar el icono de la izquierda o salir de la pantalla con el botón de la derecha.

**Textbox.** Se emplea para introducir el origen del archivo a importar.

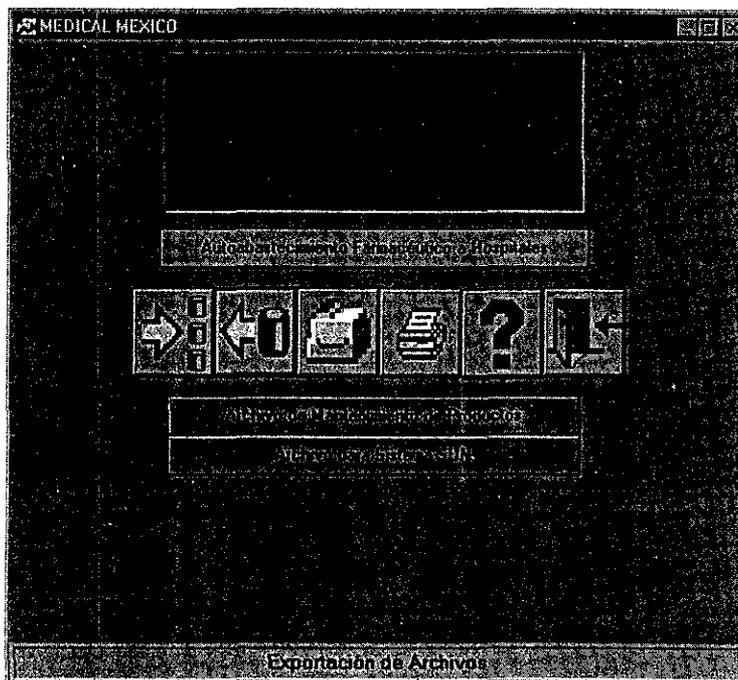
**Commondialog.** Se emplea para invocar a la pantalla de directorios y así poder seleccionar en donde se encuentra el archivo a importar. Una vez que se ha

seleccionado el directorio el control regresa al textbox para poder capturarlo. **Commdialog**. Este control en tiempo de ejecución es invisible.

**Truegrid**. Se emplea para mostrar información de los archivos recibidos y fechas de los mismos que ya existen en la base de datos.

### **Pantalla de Exportación**

En esta pantalla es donde se envían los archivos del cliente hacia la empresa farmacéutica. En esta pantalla se utilizan solo dos submódulos: Archivo de Mantenimiento de Productos y Archivo para Sistema SUN. Véase la figura 3.5.1.19.



**Figura 3.5.1.19. Pantalla de exportación.**

La pantalla para el submódulo Archivo de Mantenimiento de Productos tiene los mismos objetos utilizados en la pantalla de recepción de archivos para el submódulo

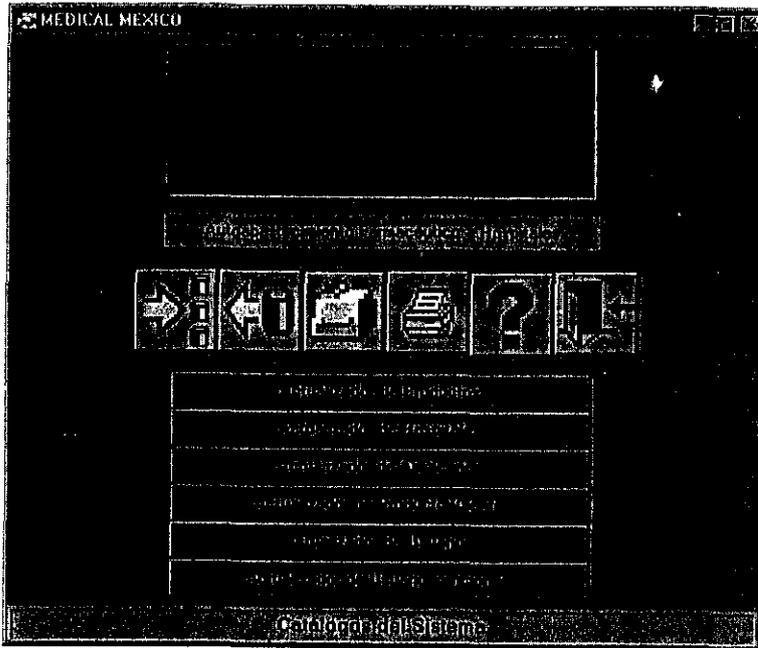
de Archivo de Productos Consumidos por Almacén, por tal razón se puede consultar en dicha pantalla los usos de los objetos utilizados.

Del mismo modo, para la pantalla del submódulo Archivo para Sistema SUN se tienen los mismos objetos, por lo que se puede consultar el uso de los mismos en la pantalla de Archivos de Productos SUN en el módulo de Importación.

### **Pantalla de Actualizaciones**

En esta pantalla se encuentran los submódulos de:

Productos, Hospitales o Clientes, Almacenes, Producto-Cliente, Usuarios y Perfil de Usuario. Véase la figura 3.5.1.20. La descripción de cómo se emplearon los objetos en esta pantalla se pueden consultar en la pantalla principal.



**Figura 3.5.1.20. Pantalla de actualizaciones.**

## Pantalla Actualizador de Productos.

En esta pantalla se actualizan los productos que la empresa fabrica y/o distribuye. En ella se utilizan los objetos: ssccommand, textbox y truegrid. Véase la figura 3.5.1.21.

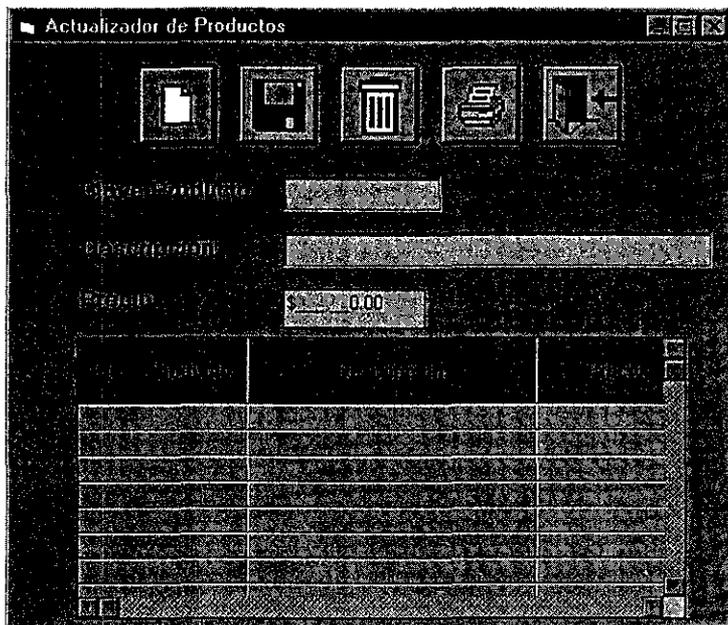


Figura 3.5.1.21. Pantalla de actualización de productos.

### Usos:

**Ssccommand.** Se emplean para ejecutar un procedimiento particular de cada uno de los iconos.

El icono con la hoja en blanco, significa que al oprimir el botón izquierdo del mouse sobre dicho icono se podrá habilitar los campos para un nuevo producto.

En el icono con el disco, guardará la información introducida en la base de datos.

El icono con el bote de basura, eliminará la información seleccionada desde la base de datos.

El icono con la impresora, en primera instancia, aparecerá una ventana con el producto seleccionado, en donde el usuario podrá imprimir el reporte del mismo.

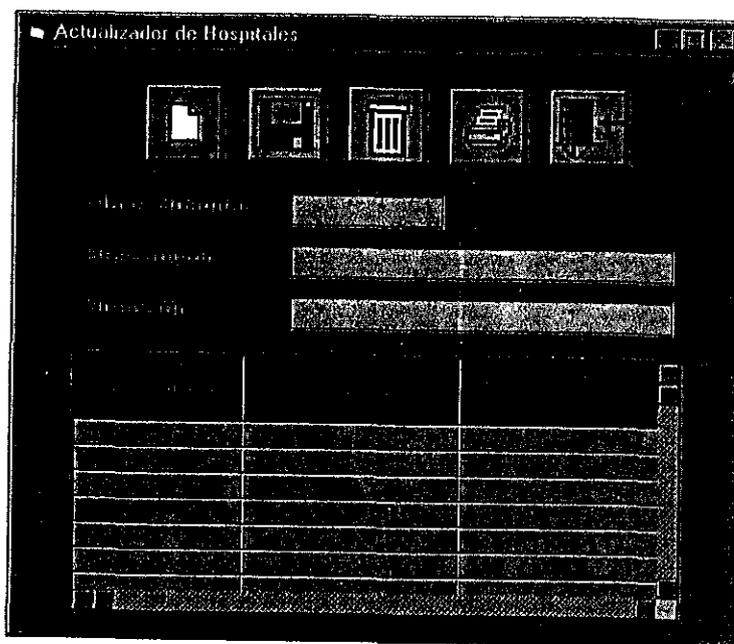
El icono con la puerta, ejecutará la salida de la pantalla de Actualización de Productos.

**Textbox.** Se emplean para introducir las características del producto, para realizar las operaciones de cada icono.

**Truegrid.** Se emplea para mostrar en una ventana los productos existentes en la base de datos y facilitar la selección de alguno de ellos para después ejecutar alguna operación sobre el mismo.

### **Pantalla de Actualización de Clientes**

En esta pantalla se actualiza la información referente a los clientes, en la pantalla utilizamos los objetos sscommand, textbox y truegrid. Véase la figura 3.5.1.22.



**Figura 3.5.1.22. Pantalla de actualización de clientes.**

## **Usos:**

**Sscommand.** Se emplean para ejecutar un procedimiento particular de cada uno de los iconos.

El icono con la hoja en blanco, habilitará los campos para un nuevo cliente.

En el icono con el disco, guardará la información del cliente en la base de datos.

El icono con el bote de basura, eliminará información de un cliente seleccionado en la base de datos.

El icono con la impresora, en primera instancia, aparecerá una ventana con la información del cliente seleccionado, en donde el usuario podrá imprimir el reporte del mismo.

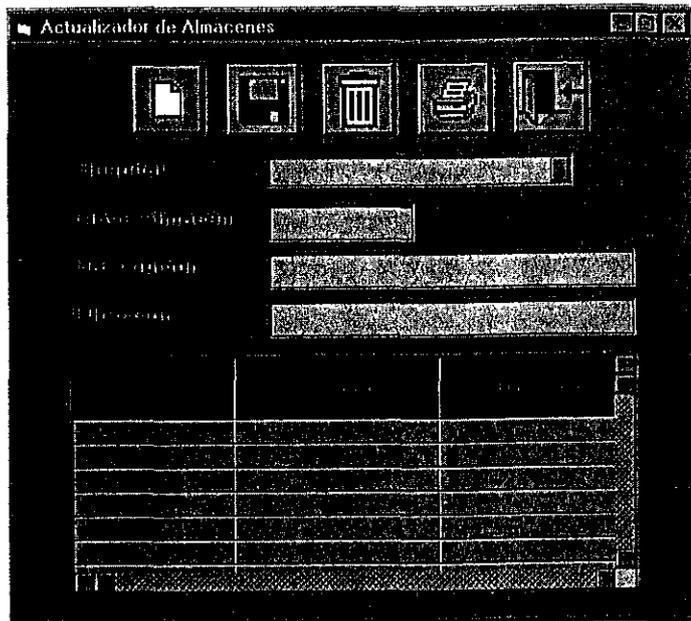
El icono con la puerta, ejecutará la salida de la pantalla de Actualización de Clientes.

**Textbox.** Se emplean para introducir las características del cliente, para realizar las operaciones de cada icono.

**Truegrid.** Se emplea para mostrar en una ventana los clientes existentes en la base de datos y facilitar la selección de alguno de ellos para después ejecutar alguna operación sobre el mismo.

## **Pantalla de Actualización de Almacenes**

En esta pantalla se actualiza la información referente a los almacenes del cliente, en la pantalla utilizamos los objetos sscommand, textbox, truegrid.y un combotext. Véase la figura 3.5.1.23.



**Figura 3.5.1.23. Pantalla actualizadora de almacenes.**

**Usos:**

**Sscommand.** Se emplean para ejecutar un procedimiento particular de cada uno de los iconos.

El icono con la hoja en blanco, habilitará los campos para un nuevo almacén.

En el icono con el disco, guardará la información de los almacenes del cliente en la base de datos.

El icono con el bote de basura, eliminará información de un almacén en la base de datos, que con anterioridad fue seleccionado.

El icono con la impresora, en primera instancia, aparecerá una ventana con la información de un almacén seleccionado, en donde el usuario podrá imprimir el reporte del mismo.

El icono con la puerta, ejecutará la salida de la pantalla de Actualización de Almacenes.

**Textbox.** Se emplean para introducir las características del almacén, para realizar las operaciones de cada icono.

**Truegrid.** Se emplea para mostrar en una ventana los almacenes existentes en la base de datos y facilitar la selección de alguno de ellos para después ejecutar alguna operación sobre el mismo.

**Combotext.** Se emplea para hacer un filtro de los almacenes de un cliente existente en la lista del combotext.

### Pantalla de Actualización de Producto-Cliente

En esta pantalla se actualiza los productos solicitados por el cliente, se utilizan los objetos sscommand, combotext y truegrid. Véase la figura 3.5.1.24.

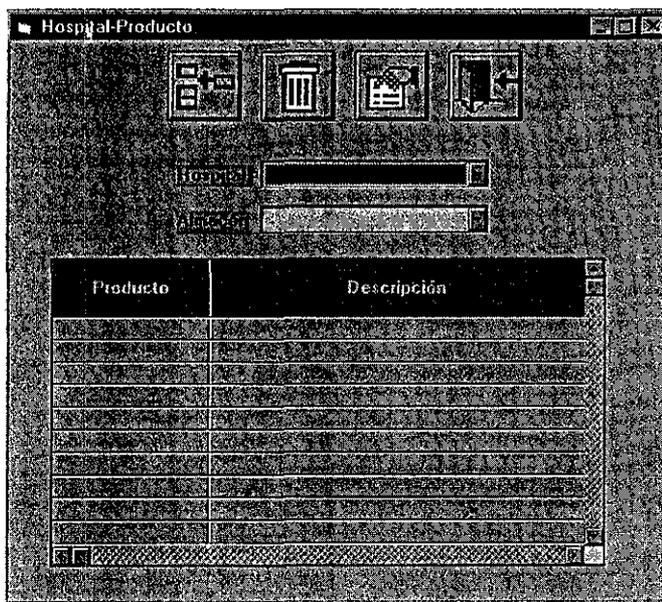


Figura 3.5.1.24. Pantalla de actualización de producto por cliente.

## **Usos:**

**Sscommand.** Se emplean para ejecutar un procedimiento particular de cada uno de los iconos.

El icono con el diagrama, mostrará la pantalla de producto.

En el icono con el disco, guardará la información en la base de datos de los productos solicitados por el cliente.

El icono con la puerta, ejecutará la salida de la pantalla de Actualización de Productos por Cliente.

El icono con la mano, mostrará la pantalla de producto.

**Truegrid.** Se emplea para mostrar en una ventana los productos solicitados ya existentes en la base de datos y facilitar la selección de alguno de ellos para después ejecutar alguna operación sobre el mismo.

**Combotext.** Se emplean para hacer filtros de los productos de un almacén de un cliente existente en la lista de los combotext.

## **Pantalla de Producto**

La pantalla maneja dos folders, el histórico y el detalle, en cada uno se tienen los siguientes objetos sscommand, textbox, sstab y combotext. Véase la figura 3.5.1.25.

Para el folder de detalle todos los campos en donde se requiere información se mantienen en forma edit, para poder capturar la información, mientras que en el folder histórico se hace una consulta y por lo tanto los campos son de sólo lectura.

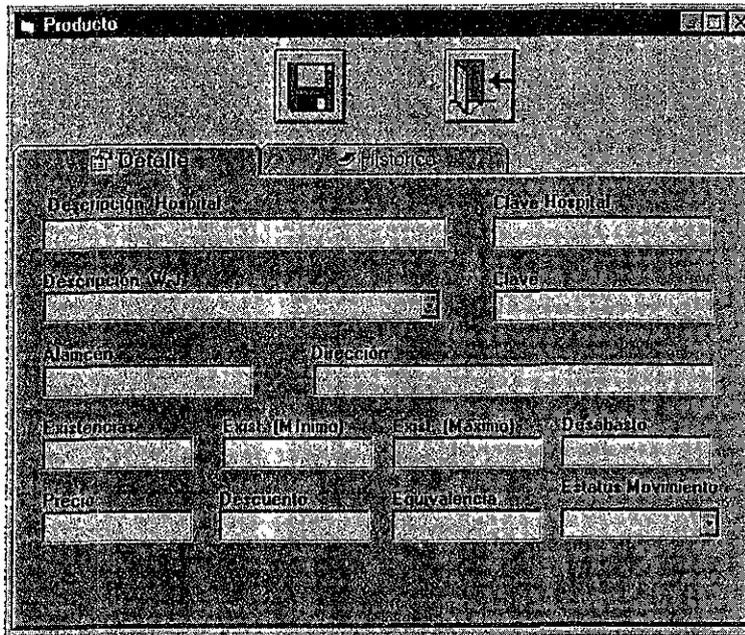


Figura 3.5.1.25. Pantalla producto.

#### Usos:

**Textbox.** Se emplean para capturar el estatus del producto.

**Combotext.** Se emplean para seleccionar la descripción del producto del cual queremos el estatus y para asignar el estatus de movimiento del mismo.

**Sscommand.** Se emplean para aceptar los datos introducidos en los textbox o salir de la pantalla.

#### Pantalla Actualización de Usuarios

Pantalla en la que se actualizan los usuarios del sistema en ella se requiere capturar el nombre del usuario, la clave del usuario y su contraseña. Por lo que requerimos los objetos de textbox y truegrid. Véase la figura 3.5.1.26.

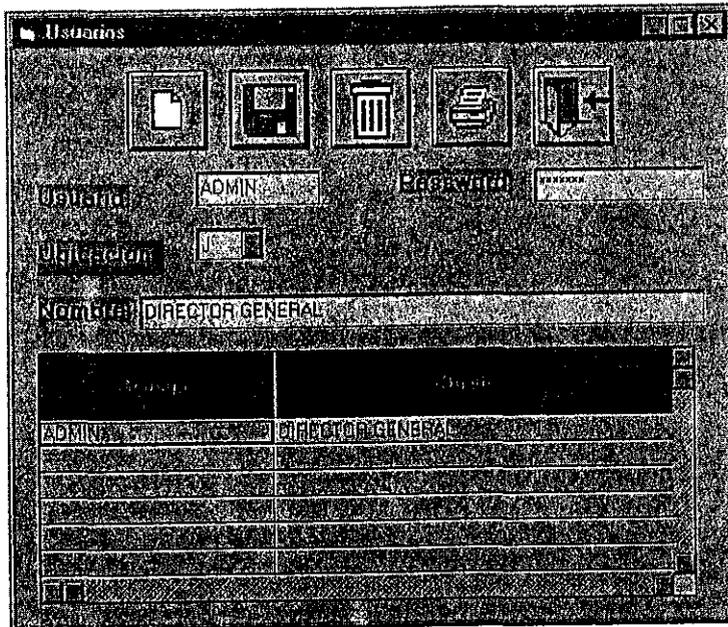


Figura 3.5.1.26. Pantalla de actualización de usuario.

**Usos:**

**Sscommand.** Se emplean para ejecutar un procedimiento particular de cada uno de los iconos.

El icono con la hoja en blanco, habilitará los campos para un nuevo usuario.

En el icono con el disco, guardará la información del usuario en la base de datos.

El icono con el bote de basura, eliminará información de un usuario en la base de datos, que con anterioridad fue seleccionado.

El icono con la impresora, en primera instancia, aparecerá una ventana con la información del usuario seleccionado, en donde el usuario podrá imprimir el reporte del mismo.

El icono con la puerta, ejecutará la salida de la pantalla de Actualización de Usuarios

**Textbox.** Se emplean para introducir las características del usuario, para realizar las operaciones de cada icono.

**Truegrid.** Se emplea para mostrar en una ventana los usuarios existentes en la base de datos y facilitar la selección de alguno de ellos para después ejecutar alguna operación sobre el mismo.

### Pantalla de Actualización de Funciones para los Usuarios

En esta pantalla se selecciona un usuario ya existente en la base de datos y se muestran sus funciones dentro del sistema. En ella utilizamos los objetos combotext, ssccommand y truegrid. Véase la figura 3.5.1.27.

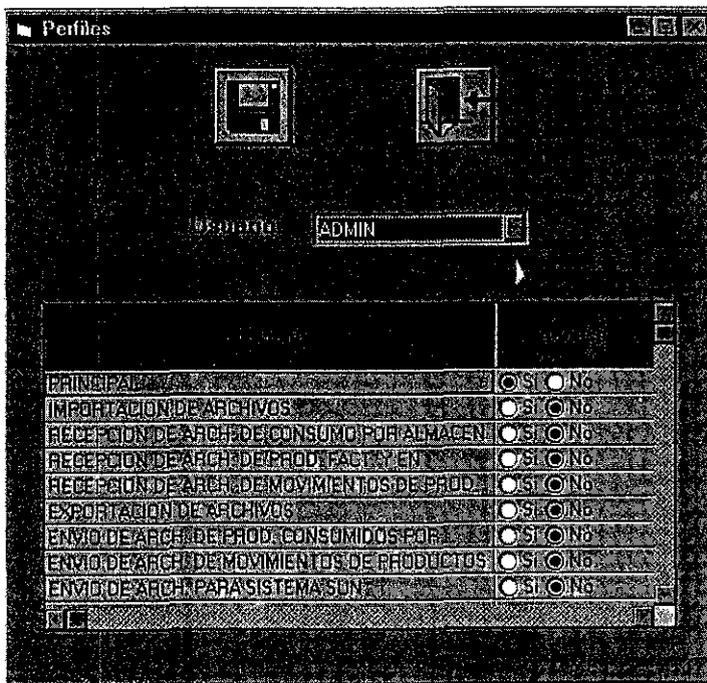


Figura 3.5.1.27. Pantalla de actualización de funciones para los usuarios.

#### Usos:

**Combotext.** Se emplea para seleccionar un usuario.

**Truegrid.** Se emplea para mostrar en una ventana las funciones que tiene un usuario en el sistema.

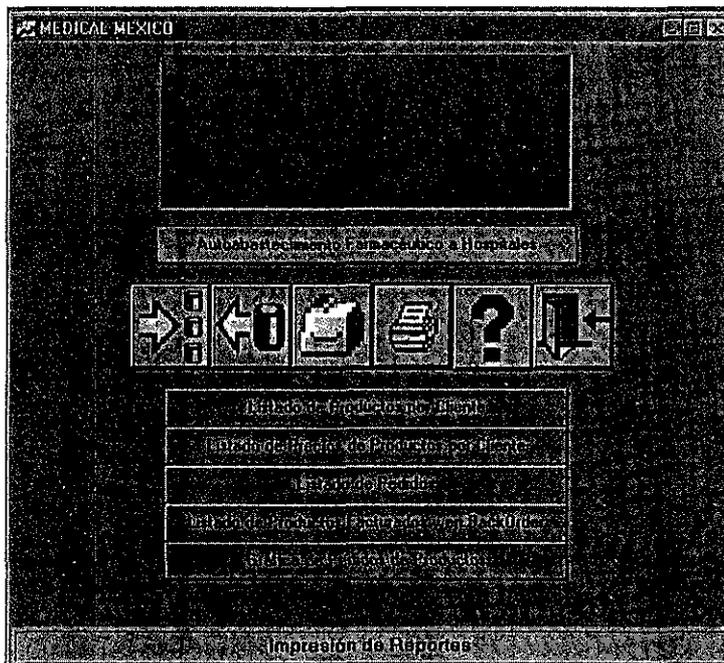
**Sscommand.** Se emplean para aceptar la información de las funciones del usuario y salir del sistema.

### **Pantalla de Reportes**

En la pantalla de reportes se tiene los submódulos:

- Reporte de Producto por Cliente.
- Reporte de Precios de Producto por Cliente.
- Reporte de Listado de Pedidos.
- Reporte de Productos Facturados.
- Gráfica de Producto-Cliente.

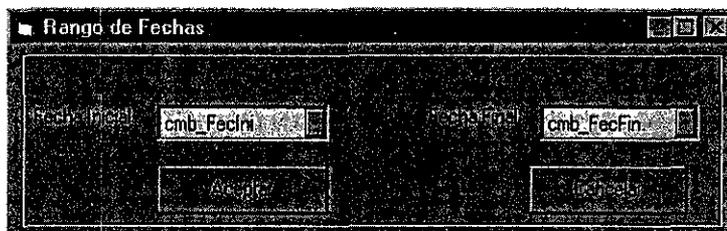
En la figura 3.5.1.28 se muestra los submódulos anteriores. Para consultar los objetos que se utilizan en esta pantalla se puede ver en la pantalla principal la descripción de cada uno de ellos.



**Figura 3.5.1.28. Pantalla de reportes.**

### **Pantalla de Rangos**

En algunos reportes es necesario hacer un rango de fechas (una fecha inicial y una fecha final con las validaciones de que la fecha final no debe ser mayor a la fecha actual en que se este manejando el sistema y además de que la fecha inicial no debe ser mayor a la final). Por lo que se crea una pantalla estándar para los mismos. En ella se utiliza combotext, ssccommand y sspanel. Véase la figura 3.5.1.29.



**Figura 3.5.1.29. Pantalla de rangos.**

## 3.6. PRUEBAS E INTEGRACIÓN DEL SISTEMA

### 3.6.1. Pruebas del Software

Las pruebas de software es un elemento crítico para la garantía o seguridad de la calidad de un sistema y representa la última revisión de las especificaciones, diseño y codificación del mismo.

El incremento visible en la creación y diseño del software, utilizado como un elemento más de un área para la que se requiere su automatización, así como, la atención a los costos asociados con las fallas del mismo, son fuerzas que motivan a planear correctamente todas las facetas de un sistema y a generar un buen conjunto de pruebas para su evaluación.

No es usual para las organizaciones de desarrollo de software, tener que gastar un alto porcentaje del esfuerzo total del proyecto en pruebas al sistema. Por una parte, las pruebas definidas por el personal que desarrollo el sistema así como por el usuario que finalmente lo utilizará, pueden costar algunas veces más que todos los otros pasos de la Ingeniería de Software combinada. En sistemas en tiempo real estos costos son mucho más altos. En desarrollos de sistemas con las nuevas herramientas que actualmente existen es extremadamente rápido, por lo que las pruebas a este tipo de sistemas son un porcentaje alto del tiempo total de desarrollo, en algunas ocasiones llegan a tener una mayor inversión de tiempo y dinero que el costo mismo del sistema.

Las pruebas pueden presentar interesantes anomalías por la ingeniería de software utilizada. Durante la temprana definición y desarrollo de las fases, la ingeniería intenta construir software desde un concepto abstracto a una implementación tangible. Cuando llegan las pruebas, el ingeniero crea una serie de casos de prueba que intentan demoler el software que ha sido construido. De hecho, las pruebas son un paso en el proceso de Ingeniería de Software que puede ser visto más como

destrutivo que constructivo.

Las pruebas son esenciales en el desarrollo de cualquier sistema de software, un sistema sin pruebas es únicamente una aplicación sin confirmación y podrá funcionar o no, pero sin pruebas, no existe ninguna forma de conocer esto antes de utilizarlo.

Los desarrolladores de software tiene que gastar aproximadamente el 40% del costo del desarrollo y tiempo en la planeación y diseño de las pruebas.

Para las pruebas del Sistema fijamos los siguientes objetivos:

1. La prueba es un proceso de ejecución de un programa con la intención de encontrar un error en él.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de encontrar un error no reconocible.
3. Una prueba satisfactoria es una que encuentra un error que el programador no localizó en las pruebas generales de su programa.
4. Llegar a una aceptación del Sistema por parte del usuario.

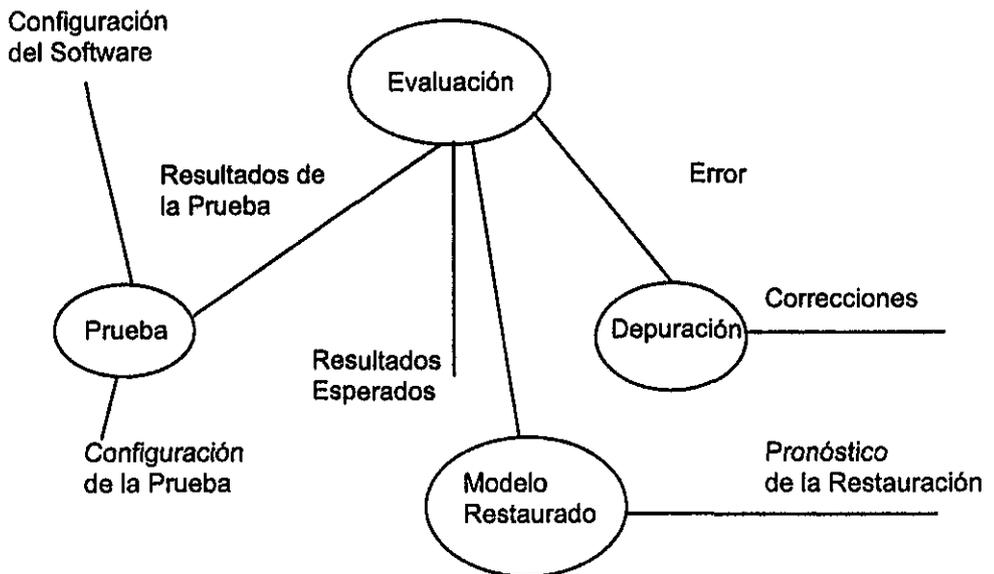
El objetivo real de las pruebas es: diseñar un conjunto de pruebas que sistemáticamente identifiquen diferentes tipos de errores y que lo realice con un mínimo gasto de tiempo y esfuerzo, y así obtener la aprobación por el usuario.

Si la prueba se conduce satisfactoriamente, ésta identificará los errores de software de la aplicación. Como un beneficio secundario, la prueba demostrará que todas las funciones del software funcionarán correctamente de acuerdo a las especificaciones del usuario.

En suma, la colección de datos para las pruebas está orientada a la funcionalidad del software, especificaciones del usuario, sin perder de vista las indicaciones y estándares de calidad definidas para el área o la empresa, todo esto como una meta

primordial de la etapa de pruebas.

El flujo de información para realizar las pruebas es el que se muestra en la figura 3.6.1.1. En esta figura se muestran dos tipos de entradas que son proporcionadas para el proceso de prueba. a) Configuración del Software que incluye la especificación de los requerimientos del producto y la especificación de su diseño y la segunda entrada b) Es la Configuración de la Prueba que incluye un plan y procedimientos de la misma. En este momento, la configuración de las pruebas es un subconjunto de la configuración del software.



**Figura 3.6.1.1. Flujo de información para realizar las pruebas.**

Todas las pruebas son revisadas en su ejecución y todos sus resultados son evaluados, esto es, el resultado de la prueba es comparado con el valor esperado de la misma. Cuando un dato erróneo es encontrado o identificado implicará que se tenga

que realizar **debug** (depuración) del código para precisar la procedencia y solución del error.

Las pruebas son aplicadas a diferentes niveles de acuerdo al ciclo de vida del desarrollo de software, sin embargo para cada una de las pruebas se definen diferentes objetivos y se consideran diferentes premisas de acuerdo al nivel que se necesite evaluar. El centro de todo el desarrollo de las pruebas, es encontrar errores, pero los diferentes tipos de errores que se encuentran son marcados para cada nivel evaluado.

### **3.6.1.1. Pruebas por Módulo o Unidad**

Las pruebas de unidad son el nivel más bajo en la tipificación de las exámenes del software, la función de la unidad básica del software es probada aisladamente (bajo el control de cualquier requerimiento global necesario para realizar la prueba).

La prueba de unidad es realizada por el programador cuando escribe el código. El propósito de la prueba de unidad es encontrar errores en unidades individuales de toda una aplicación, en sus datos o en su lógica de funcionamiento. Las pruebas pueden ser derivadas de el detalle lógico de la unidad con cualquier estructura de prueba adicional derivada a su vez del diseño físico, como el resultado de la identificación de la medición de todos los errores localizados o por localizar.

Para cada una de las unidades se probaron los siguientes aspectos:

- Se revisó la presentación de la pantalla, es decir, el tamaño de las fuentes y de los controles, la ortografía, el color, etc.
- Se probó que todos los controles que tienen asociado algún evento reaccionen adecuadamente, por medio del teclado y/o el ratón.

### **Pantalla de Password (Contraseña)**

- Se planteó una prueba modificando las variables globales que afectan directamente a este Módulo para comprobar que se comportara adecuadamente.
- Como es la pantalla en donde se captura la contraseña, se comprobó que al momento de errar por tercera vez en la captura de la clave del usuario y de la contraseña, el sistema mostrará un mensaje de "Acceso Denegado", saliendo de esta pantalla.

### **Pantalla Principal**

- Se comprobó que la pantalla presentara todas las opciones que se podrán manejar dentro del sistema.
- Se probó que aparecieran todas las opciones para cada uno de los botones.

#### **Botón de Importación de Archivos:**

Archivo de Productos Consumidos por el Almacén.

Archivo de Productos de SUN.

Archivo de Movimientos de Productos.

#### **Botón de Exportación de Archivos:**

Archivo de Mantenimiento de Productos.

Archivo para Sistema SUN.

#### **Botón de Catálogos del Sistema:**

Actualizador de Productos.

Actualizador de Hospitales.

Actualizador de Almacenes.

Actualizador de Producto-Cliente.

Actualizador de Usuarios.

Actualizador de Usuario-Perfil.

#### **Botón de Impresión de Reportes:**

Listado de Producto por Cliente.

Listado de Precios de Producto por Cliente.

Listado de Pedidos.

Listado de productos facturados y en **backorder** (pendientes).

Gráfica.

Ayuda:

Nuestro credo.

Visión estratégica.

Acerca de...

Salir:

Salida del sistema (no presenta ningún mensaje).

### **Importación de Archivos**

- Archivo de productos consumidos por el almacén.

Pestaña de Recepción:

Se probó la importación de archivos, presentándose en la pantalla su nombre, así como su fecha de carga.

Se probó la salida de esta opción.

Pestaña de Configuración:

Se probó la configuración de archivos con los tres diferentes delimitadores (posiciones, tabuladores y comas), para después proceder a su Recepción.

a) Por posiciones

Se probó que tanto la posición inicial como la posición final de cada campo fuera respetada por el código.

b) Por comas y tabuladores

Se probó que el código funcionara adecuadamente para diferenciar entre cada campo de acuerdo a estos delimitadores.

- Archivo de Movimiento de Productos.

Se probó que la recepción de este tipo de archivos fuera adecuada, así como también la salida de la opción.

- Archivo de productos de SUN.

Se probó la manera de importar información de los archivos facturados y en

backorder. También se probó que desplegara el histórico del archivo.

### **Exportación de Archivos**

- **Archivo de Mantenimiento de Productos**

Se probó la generación de archivos que serán explotados por las bases de datos externas, la presentación de su fecha y la salida de la opción.

- **Archivo para Sistema SUN**

Es esta unidad se hizo la prueba para la generación de este tipo de archivo, la presentación de su fecha, así como la salida de la opción.

### **Catálogos del Sistema**

Para cada catálogo aplicamos pruebas muy similares, ya que como es un módulo de mantenimiento de catálogos, sus funciones son parecidas en cada opción.

Se probó que las altas, bajas y cambios de los productos de la base de datos local fueran realizados correctamente:

**Botón de Nuevo:**

El botón de nuevo se probó que las cajas de la pantalla fueran limpiadas, para poder así capturar el producto.

**Botón de Registrar:**

El botón de registrar o dar de alta fue probado para confirmar que efectivamente la captura de una entidad nueva fuera insertada en la base de datos.

**Botón de Borrar:**

Se borraron varias entidades para confirmar que esta opción trabajara adecuadamente.

**Botón de Imprimir:**

La impresión se probó para aquellas entidades que se capturan o aparecen en ese momento en la pantalla.

**Salir:**

Se probó la salida de la opción, es decir, su regreso a la pantalla

principal.

Además para:

- Actualizador de Almacenes

Se probó además la correcta correlación para la actualización de el(los) almacén(es) de cada hospital.

- Actualizador de Producto-Cliente

La prueba se generó para comprobar que fueran válidas las altas, bajas y cambios de los productos por cada cliente.

- Actualizador de Usuarios

Se hicieron modificaciones a los mismos, así como altas y bajas para comprobarlos posteriormente en la Pantalla de Contraseña.

- Actualizador de Usuario-Perfil

Se probaron las modificaciones para cada nivel de seguridad dentro del Sistema, el complemento de esta prueba se realizó para cada una de las unidades en las que aplican dichos cambios.

### **Impresión de Reportes**

- Listado de Productos por Cliente

Con la prueba se confirmó que de este listado obtuviéramos el total de productos por cada hospital y cada almacén, nos diera el identificador del producto, la clave del hospital, la descripción, los precios y descuentos de los productos, la existencia máxima, mínima y el desabasto.

- Listado de Precios de Productos por Cliente

El reporte se probó para que por cada hospital y almacén se obtengan los precios de cada producto. En este reporte se verificó la salida del identificador del producto, la clave del hospital, su descripción, el precio de costo, sus descuentos, y el precio de venta.

- Listado de Pedidos

Para el rango de fechas para las cuales se requiere el reporte se probó la generación

de todos los pedidos por hospital y por almacén. En el reporte se obtuvieron el número de orden, descripción del producto, la cantidad abastecida, la cantidad pedida y la fecha de captura.

- **Listado de Productos Facturados y en Backorder**

De forma similar que en el reporte anterior, por medio de un rango de fechas se probó la generación del reporte de pedidos que no se pudieron surtir a los clientes, obteniendo la fecha del pedido, la descripción del producto, el número de orden, la cantidad del pedido facturado y la fecha de la factura.

- **Gráfica de Pedidos de Productos**

La generación de la gráfica de estadística de la cantidad de pedidos dependiendo del hospital, almacén y el producto seleccionado fue probada, es decir, se presentaron las ocurrencias de los productos gráficamente, para cada pedido.

Además para cada una de las opciones de generación de reportes se probó el evento de regreso a la pantalla principal.

### **Ayuda General del Sistema**

Como estas unidades son sólo de presentación de texto, sólo se comprobó su correcta redacción y su adecuado control de eventos.

### **3.6.1.2. Pruebas de Integración de Módulos o Unidades**

Este tipo de pruebas se encuentran en un nivel más alto que las pruebas por unidad, ya que se enfocan tanto a la estructura como a la funcionalidad de la programación entre dichas unidades.

#### **Pruebas de Caja Blanca**

Las pruebas de caja blanca es un método diseñado para crear casos para prueba de software, aprovechando el control estructurado del diseño procedural de la aplicación. *Utilizando los métodos de prueba de caja blanca, el ingeniero de software puede*

construir ejemplos para pruebas que garanticen que todas las rutas independientes de un módulo sean probadas al menos una vez, probar todas las decisiones lógicas en sus casos **TRUE** (VERDADERO) y **FALSE** (FALSO), probar todos los **loops** (ciclos) y sus fronteras dentro de operaciones definidas y probar la estructura de datos interna para asegurar su utilidad. A las pruebas de caja blanca también se les conocen como pruebas estructurales, porque utilizan las estructuras internas de un programa para derivar los casos de estudio.

Ya que se obtuvieron ciertos resultados para cada una de las unidades, construimos algunos ejemplos que nos dieron una visión más clara de qué es lo que deberíamos esperar de las entradas/salidas de cada una de estas unidades. Así, para el Módulo de Catálogos, en la opción de Actualizador de Usuarios, al capturar todos los datos que se requieren al dar de alta un nuevo usuario se confirmó la correcta inserción del mismo en la base de datos, es decir la correcta validación de los datos de entrada para generar su correspondiente salida; caso contrario que si no se capturan todos los datos, pues no se podrá realizar la inserción de ese usuario, dando como resultado de la prueba, la correcta validación de los datos de entrada para que los de salida no sean generados inconsistentemente.

### **Pruebas de Caja Negra**

El método de prueba de caja negra se enfoca en los requerimientos funcionales del software, es decir, habilita al ingeniero de software para derivar un conjunto de condiciones de entradas y que serán ejecutadas para todos los requerimientos funcionales de un programa.

Estas pruebas son complementarias a las técnicas de la caja blanca, ya que cubren diferentes clases de errores. A diferencia de las pruebas de caja blanca, que se ejecutan al inicio del proceso de pruebas; las de caja negra son aplicadas durante las etapas finales de las pruebas.

Las pruebas de caja negra también se conocen como pruebas funcionales, pues se basan en la definición de que, un programa debe realizar eficientemente las especificaciones definidas, sin importar la estructura de como lo hace.

Para este tipo de pruebas construimos ejemplos que nos ayudaron para la correcta integración de todas las unidades. Fuimos enlazando unidad por unidad para confirmar que nuestro diseño cumpliera con las funciones planteadas. La integridad de la información fue una parte importante a considerar ya que la funcionalidad, de una unidad a otra, depende en gran medida de esta integridad.

### **3.6.2. Pruebas Orientadas al Usuario**

Es virtualmente imposible para un desarrollador de software imaginar como el cliente realmente utilizará un programa. Instrucciones para utilizar el software pueden ser mal interpretadas; extrañas combinaciones de datos pueden ser regularmente usadas; la salida que parece clara para el desarrollador puede ser no entendible para un usuario. Cuando el software del cliente se construye, una serie de pruebas aceptables son conducidas para habilitar al cliente y validar todos sus requerimientos. Conducido por el usuario final, más que por el desarrollador del sistema, una prueba aceptable puede cambiar de una informal prueba de funcionamiento a planear y ejecutar sistemáticamente una serie de pruebas. En efecto, las pruebas de aceptación pueden ser conducidas por un período de semanas o meses, identificando errores que significan retrasos de la liberación del sistema.

#### **Prueba Alfa**

En un ambiente diseñado especialmente para esta prueba, se trató de recrear lo más posible la información cotidiana que procesan los usuarios, para que estos no fuesen influenciados adversamente. La presencia y participación de los analistas y desarrolladores del Sistema fue evidente. La prueba generó el punto de vista de ambas partes para ser encaminada a una complementación que mejoró algunos aspectos del

Sistema.

### **Prueba Beta**

Esta prueba nos arrojó resultados interesantes ya que fue en la que el usuario planteó casos que muchas veces no nos esperábamos. Aunque la retroalimentación entre los usuarios y nosotros no fue inmediata (ya que no nos encontrábamos en ese lugar) la generación de los resultados de la prueba hicieron evidentes algunos detalles que desembocaron en modificación de ciertas partes de la codificación. Uno de estos detalles fue la escasez de mensajes de error que orienten a los usuarios para una correcta operación del Sistema.

## **3.7. IMPLANTACIÓN**

La implantación del sistema tiene como objetivo el pasar del sistema actual al nuevo con una buena planificación para que el impacto del cambio no sea tan difícil para el usuario ya que el sistema se convierte en la forma de trabajar de la empresa. El problema más importante es convencer al usuario de los beneficios del nuevo sistema y desarrollar su capacidad de operarlo. Si en esta etapa no se logra que los usuarios utilicen el sistema se perderá una gran inversión.

### **3.7.1. Objetivos**

Los objetivos a cumplir en esta etapa de implantación serán:

- Asegurarse de que la instalación del sistema se lleve a cabo en los términos acordados entre la empresa y el usuario.
- Desarrollar un proceso de capacitación que permita al usuario obtener los beneficios esperados del sistema.
- Liberar el sistema a la vida productiva.
- Servir como mecanismo de control de actualizaciones del sistema.
- Ofrecer al usuario el soporte durante la vida productiva del sistema, para garantizar la funcionalidad del mismo.

### **3.7.2. Definición**

Esta etapa de implantación se define como el conjunto de actividades tendientes a lograr la capacidad del usuario para operar y tener en producción el Sistema de Autoabastecimiento a Hospitales.

Se proporcionará a los usuarios un entrenamiento con características pedagógicas adecuadas que permita asegurar la correcta operación, así como la explotación de todos los beneficios ofrecidos por el sistema.

Se espera que en el período de post-implantación se presenten modificaciones al sistema como consecuencia de cambios de organización, administrativo, jurídicos, fiscales, etc., para solucionar esto, se recurrirá al mantenimiento. El mantenimiento es la etapa de la vida del sistema durante la cual éste se encuentra en producción y donde ocurren eventualmente situaciones que obligan a modificar al mismo.

En este punto concluiremos la instalación iniciada en el punto 3.6 y explicaremos como se efectuará la capacitación. El proceso de post-implantación será explicado detalladamente más adelante en el mantenimiento del sistema.

### **3.7.3. Instalación**

Se acordó junto con la empresa cumplir los siguientes puntos para instalar de manera adecuada el sistema:

- Elaboración de un plan de instalación.
- Desarrollar procesos de instalación y archivos de conversión.

- Documentación para la liberación.
- Establecer acuerdo para la liberación.
- Establecer el ambiente de producción, soporte y seguimiento de la instalación.
- Realizar el Monitoreo de la producción.
- Transferir a operación.
- Efectuar una revisión post-instalación.

### **Elaboración de un Plan de Instalación**

Una vez que el sistema fue integrado y probado (en el punto 3.6), se tienen contempladas las siguientes actividades para dejarlo instalado y listo a incorporarse en la producción de la empresa:

- Importación y catalogación de archivos de productos en la base de datos del sistema (aproximadamente 3 semanas).
- Captura de usuarios, almacenes y hospitales y reconfiguración del sistema estableciendo niveles de acceso (1 semana).

La finalidad de esto es el llenado previo de información inicial necesaria para empezar a operar el sistema. Se contempla la presencia del administrador por parte de la empresa para adentrarlo en cómo se efectúa la instalación del sistema.

Se llevará una revisión de las actividades de instalación y se verificará a su vez la funcionalidad del sistema.

## **Desarrollo de Procesos de Instalación y Archivos de Conversión**

Para el proceso de importación de información inicial del sistema anterior al nuevo, se contará con un formato preestablecido de campos específicos separados por posiciones, comas o tabuladores para ingresar y catalogar la información con la finalidad de mantener la consistencia en la información almacenada en la base de datos; se usará una función específica de importación del sistema, para efectuar este proceso. En esta etapa se incluirá al administrador o los administradores del sistema como parte del proceso de introducción al sistema.

La captura de información de usuarios se realizará también a través de otra función específica del sistema, que presentará un formato donde se introducirán los datos y privilegios asignados.

Las pruebas al sistema fueron cubiertas en el punto 3.6, aquí sólo se llevará una revisión de la consistencia en la base de datos de la información inicial.

Se contará además con un plan de contingencia en caso de fallas que cubra los siguientes puntos:

- Mantener la información del sistema anterior, por un período predeterminado.
- Tener en espejo el disco duro que contiene las bases de datos y los programas.
- Contar con otra computadora de las mismas características (para poder realizar el cambio de discos en caso de fallas).

## **Documentación para la Liberación**

Se proporcionará a los operadores del sistema la siguiente documentación:

- Diseño del sistema.
- Especificaciones de procesos y funciones.
- Manual de usuario.
- Manual técnico.
- Listado de programas objeto y archivos del sistema.
- Diccionario de datos donde se especifican tablas y registros y su relación entre ellos.

### **Establecer Acuerdos para la Liberación**

Se acordarán con las áreas "aprobadoras" la fecha en las que entrarán los cambios del nuevo sistema y la fecha en las que se deberán entregar los manuales y documentación necesaria para la liberación del sistema.

### **Ambiente de Producción, Soporte y Seguimiento de la Instalación**

- El ambiente de producción será el propio de la empresa donde se tienen los almacenes y las terminales de atención a clientes. Se establecerá además un ambiente simulado donde se dará capacitación a usuarios y operadores, para no afectar el proceso de producción del sistema.
- Se tiene contemplado un programa de soporte por tres meses en el cual se dará asistencia a usuarios durante los horarios de producción de la empresa mientras se estabiliza el sistema. En este punto se supervisará el aprendizaje de los administradores del sistema y se les apoyará en la solución de problemas.

- Se verificará que todos los componentes hayan sido transferidos adecuadamente a las bases de datos (antes de correr cualquier proceso).
- Se llevará una bitácora donde se registrarán los pasos y tiempos de como se efectúa la instalación y se revisarán los comentarios y observaciones generados al respecto.

### **Monitoreo de Producción**

Se revisará durante un período de garantía, que el sistema cumpla con los niveles de servicios establecidos con la empresa (funcionalidad, estabilidad, etc.). Se documentará y llevará el control de posibles cambios de información del sistema derivados de este proceso.

### **Transferencia a Operación**

Se realizará el acta de liberación, que formalice la entrega del sistema.

### **Revisión Post-Instalación.**

(Esta información será sólo para uso de la empresa). Se analizarán y documentarán los siguientes puntos:

- Conclusiones y comentarios sobre diseño, ejecución, funcionalidad, recursos humanos y materiales.
- Análisis sobre la metodología ocupada y recomendaciones.
- Conclusiones de costo del proyecto contra los planes originales.

Se efectuará una revisión de aseguramiento de la calidad.

### **3.7.4. Capacitación**

Este proceso se desarrollará de acuerdo a los siguientes puntos con el propósito de proporcionar una buena capacitación al usuario y lograr la incorporación del sistema en el proceso de producción:

- Elaboración de un plan de capacitación de usuarios.
- Desarrollo de material de capacitación.
- Elaboración de la documentación para el usuario.
- Capacitación al usuario.

#### **Elaboración de un Plan de Capacitación de Usuarios**

- Se elaboró un programa de 4 semanas de duración para efectuar los cursos de capacitación en relación al sistema. Los horarios y fechas se establecerán en conjunto con los usuarios.
- Se determinaron los requerimientos de capacitación para el usuario final del sistema.
- Se estableció usar dos niveles de capacitación, uno para usuarios y en el otro operadores o administradores del sistema. Para los usuarios se tienen contemplados básicamente los procesos de captura y modificación a la información almacenada, mientras que en el caso de operadores o administradores, además a estos, se le capacitará en los procesos de instalación, operación y administración avanzada, dependiendo del puesto y las tareas que el personal desempeña.

- Se obtendrá el visto bueno de usuarios en el plan de capacitación.

### **Desarrollo de Material de Capacitación**

- El material de capacitación se elaboró con el visto bueno del usuario y consta de un manual de usuario, manual de instalación, especificaciones de procesos y funciones y manual de prácticas. Se realizó una capacitación piloto, para revisar el material antes de imprimirlo y distribuirlo.
- Se seleccionaron y capacitaron a los instructores.

### **Elaboración de la Documentación para el Usuario**

La documentación final para el usuario se obtiene de la información y observaciones generadas en las etapas de instalación y capacitación.

### **Capacitación al Usuario**

Se capacitará y evaluará a los usuarios al término del curso para determinar si se continuará con las siguientes etapas, o es necesario una recapacitación.

Al personal usuario del sistema, se le capacitará con respecto a:

- Uso y funciones completas del sistema.
- Niveles de seguridad.
- Información que se maneja y la manera de modificarla.

Al personal encargado de la operación y administración del sistema, se le capacitará además con respecto a:

- Estructura general del sistema.
- Funciones de cada módulo.
- Entidades entre datos existentes.
- Pasos y procedimientos en operación normal y emergencia.
- Funciones complejas del sistema.

Al personal del área administrativa que será dueña del sistema, se le capacitará con respecto a:

- Impacto que causará el sistema en los procedimientos del área o en sus relaciones con otras áreas.

Se establecerá un plan de capacitación continua sobre el funcionamiento u operación del sistema, dependiendo de las responsabilidades del usuario.

## **CAPÍTULO IV**

### **IMPACTO DEL SISTEMA**

## **4. IMPACTO DEL SISTEMA**

En este capítulo nos enfocaremos al mantenimiento que debe recibir el sistema una vez liberado. En los puntos 4.1.1, 4.1.2 y 4.1.3 se indica un mantenimiento que depende de nuevos requerimientos que solicite el usuario y en su momento se tratarán con las consideraciones descritas en dichos puntos. Pero el mantenimiento real, que debe considerarse durante toda la vida del sistema, es el que debe llevarse a cabo a la Base de Datos. En el punto 4.1.4 se indica a detalle los diferentes aspectos a considerar para el funcionamiento óptimo de la base.

### **4.1. MANTENIMIENTO DEL SISTEMA**

Partamos de una frase clave, software mantenible, que es la facilidad de que el software pueda expandirse o contraerse para satisfacer nuevos requerimientos o bien se puedan detectar errores, deficiencias u obsolescencias.

Hagamos una comparación básica, mantenimiento en términos de hardware es restaurar una condición inicial. Mantenimiento en software implica un producto nuevo, probablemente mejor, mayor o mas eficiente, pero nuevo, con la consecuente lista de acciones ligadas a esta situación: actualización de la documentación, adecuaciones, pruebas, implantaciones, nueva capacitación, etc.

Mantenimiento de software involucra aspectos y casos como: requerimientos o necesidades, correcciones, adaptaciones, reestructuraciones y engrandecimientos.

1. Requerir o necesitar es la especificación que lleva al mantenimiento.
2. Corregir es realizar todas aquellas actividades para que un módulo o conjunto de módulos que no efectúan la función que se pretende la realicen, modificando el software actual.
3. Adaptar es una modificación ocasionada por un cambio del hardware o software en que reside el sistema.

4. Engrandecimiento es una modificación para permitir al sistema realizar nuevas funciones con base en los nuevos requerimientos.
5. Reestructuración es una modificación que pretende mejorar la estructura interna del sistema conservando su comportamiento externo.

#### **4.1.1. Preventivo**

Entrando en materia, es común que el grupo encargado de dar mantenimiento a un sistema, se queje de que el grupo de desarrollo no previó la fase de mantenimiento de una manera adecuada, para evitar lo anterior se recomiendan una serie de requisitos que pretenden obtener un producto mantenible, esto es, las condiciones mínimas para lograr sistemas estándares para que también el mantenimiento tienda a ser homogéneo.

1. Los requerimientos deben de presentarse (por escrito) bien definidos y detallados, de tal manera que se puedan comprender fácilmente. A los requerimientos del usuario final se deben de agregar los de operación y producción. De preferencia se desean tener bien diferenciados los requerimientos mínimos y los opcionales.
2. Las especificaciones deben de poder probarse.
3. Se debe de analizar y diseñar, independientemente de la técnica usada, de tal manera que sea posible expandir, contraer y adaptar fácilmente; la explicación debe ser rica en ejemplos y prevenir cambios obvios.
4. La programación debe usar versiones y características estándares del lenguaje seleccionado. Se debe de documentar lo más posible explicándolo al grupo de personas que lo va a usar y nunca a si mismo.

Se debe documentar total y ampliamente, entregando dicha documentación tanto al usuario final como a las áreas involucradas, especialmente al encargado de mantenimiento.

Con estas condiciones mínimas, el mantenimiento deberá incluir las siguientes etapas:

1. Comprensión y compenetración del sistema actual.
2. Comprensión del objetivo de la modificación y su enfoque.
3. Implantación de la actualización o modificación.
4. Revalidación del software.

Cuando se inicien las etapas del mantenimiento, se debe tener presente la siguiente problemática:

1. La calidad de la documentación.
2. Las limitaciones de los recursos (durante todas las etapas).
3. La calidad del software original.
4. La dificultad para entender el sistema debido a su complejidad.

Para comprender mejor el último aspecto, definamos calidad como la cercanía del sistema con respecto a los requerimientos del usuario así como la prevención de los efectos; un software de calidad reúne las características de: confiabilidad, consistente, eficiente, conciso, mantenible, portable, entendible y económico.

#### **4.1.2. Correctivo**

Una vez familiarizados con la problemática que se presentará y pensando en las características anteriores, para ejecutar el mantenimiento de programas se necesita contar con los requerimientos siguientes.

##### **1. Técnicos:**

- a) Medio ambiente de prueba muy semejante o al menos capaz de simular el ambiente operacional.
- b) Recursos de cómputo suficientes con base en la situación real de operación.
- c) Insumos suficientes (datos prueba, verificadores, etc.)
- d) Herramientas de diagnóstico, prueba (de los que se habló en su momento) y mantenimiento.

e) Personal de las áreas involucradas con perfecto conocimiento de causa.

## **2. Conocimientos y Práctica**

Dada la responsabilidad del mantenimiento, es necesario eliminar la tendencia de destinar a los programadores nuevos a esta actividad, mientras adquieren experiencia y conocen el equipo. Hay que recordar que un mantenimiento mal ejecutado ocasiona grandes costos y problemas con algo que ya está produciendo, ya que se modifica su integridad. El personal del mantenimiento no sólo debe saber desarrollar sistemas, sino más que eso, mantener sistemas.

## **3. Elementos Técnico-Administrativos**

Buscando mantener la calidad y confiabilidad del producto, es necesario sujetarse a los estándares y verificar que se cumplan tomando en cuenta lo siguiente:

- a) Las prioridades y objetivos pretendidos.
- b) Verificación del cumplimiento de los estándares para mantener la calidad del producto.
- c) Exigir documentación oportuna y actualizada.
- d) Establecer auditorías con enfoque en control de calidad.

### **4.1.3. Control de Cambios**

1. Use en el análisis y diseño técnicas ascendentes y/o descendentes.
2. Plantee desde un principio un plan de revalidación. Es decir un plan de pruebas del software que sea sujeto de mantenimiento.
3. Utilice todas las herramientas disponibles, tales como auditores de código, generadores de referencias cruzadas, generadores de documentación, generadores de datos de prueba, programas de diagnóstico en línea, etc.

4. Trate de lograr una independencia de la máquina; así como un código lo más apegado con el lenguaje estándar.
5. Use estándares en el desarrollo del software.
6. Use técnicas adecuadas y actualizadas.
7. Opte por un producto mantenible sobre uno eficiente.
8. Utilice técnicas de Ingeniería de Software adaptándolas para su aplicación en su propio medio ambiente del mantenimiento.

**Directrices para el Control del Proyecto de Mantenimiento de Software:**

- a) Defina explícitamente las metas y prioridades del grupo de mantenimiento.
- b) Prepare un plan de soporte a mantenimiento y úselo para administrar el proceso.
- c) Organice revisiones periódicas de requerimientos, directamente con el usuario.
- d) Trabaje con grupos de mantenimiento con poca gente de buen nivel.
- e) Lleve a cabo controles y contabilice lo efectuado por cada grupo de mantenimiento.
- f) Intente desarrollar programotecas ad-hoc para la administración del mantenimiento, dadas las condiciones de su instalación.

Evalúe el mantenimiento con base en sus metas.

**Directrices para Control de lo Esperado en un Proyecto de Mantenimiento de Software**

- a) Defina las tareas de Mantenimiento en términos de requerimientos opcionales y futuros.
- b) Opte por la claridad y sencillez, aún sobre las integridad.
- c) Piense en un software mantenible aún después de modificarlo.
- d) Revise periódicamente el control de calidad del producto.
- e) Involucre al usuario en las revisiones.
- f) Mantenga actualizada la documentación.

**Para llevar a cabo un Plan de Mantenimiento adecuado interactue con el Usuario final y contemple los siguientes puntos:**

- a) Un procedimiento para reporte y corrección de fallas en software.
- b) Un procedimiento de cambios y su implantación.
- c) Requerimientos de protección para el mantenimiento y control de calidad de la aplicación en general.
- d) Un plan de revalidación, que incluye pruebas.
- e) Un plan de actualización de la documentación.
- f) Requerimientos de soporte técnico que incluyan, por ejemplo: herramientas técnicas disponibles, configuraciones, soporte e interfaces con el usuario, facilidades para prueba del sistema, manuales de usuario y de operación y documentación en general.
- g) Procedimientos para liberación de sistemas.

**Directrices a seguir durante el Análisis de Requerimientos para el Mantenimiento**

- a) Defina el objetivo del requerimiento.
- b) Defina el requerimiento en común acuerdo con el usuario.
- c) Contemple posibles cambios y consecuencias.
- d) Considere los efectos de las modificaciones.
- e) Determine compatibilidades presentes y futuras.
- f) Justifique las modificaciones en términos de costo, tiempo y riesgo.

Es necesario que el usuario apruebe antes de proceder definitivamente.

**Directrices para la Especificación del Proceso de Mantenimiento**

- a) Desarrolle especificaciones para las modificaciones, siguiendo los mismos estándares del desarrollo del software.
- b) Describa especificaciones factibles de probar,
- c) Identifique el ámbito a modificar.
- d) Estudie el impacto de la modificación y los recursos necesarios para soportarla.
- e) Antes de proceder a diseñar, es necesario que el usuario apruebe la propuesta.

### **Directrices para la Fase de Diseño del Mantenimiento**

- a) Analizar diseños compatibles con la filosofía original del diseño.
- b) Sea sencillo en el diseño, seleccione aquél que modifique el menor número de módulos y que éstos sean los menos complejos.
- c) Documente siguiendo el estándar del diseño original.
- d) Evalúe la flexibilidad del diseño para aislar funciones y proveer interfaces pensando en futuros cambios.
- e) Es necesario que el usuario apruebe antes de implantar, para dicha implantación se debe pensar en:
  - I. Interpretar correctamente el diseño de la modificación.
  - II. Minimizar el impacto de la modificación en el resto del software.

### **Directrices que Pretenden Conservar la correcta Estructura del Código**

- a) Use los estándares de la instalación.
- b) Siga las técnicas usadas en el análisis y diseño.
- c) Documente todos los cambios en el código y conserve las últimas versiones antes de modificar.
- d) Duplique el código en lugar de crear rutinas comunes.
- e) Codifique pensando en no degradar la mantenibilidad del software.
- f) Actualice la documentación.
- g) Revise código para conservar la calidad y los estándares.

### **Directrices para la Revalidación**

- a) Revalide empleando pruebas unitarias y siguiendo el orden del desarrollo.
- b) Trabaje con cada módulo, compare los resultados con los del desarrollo y busque discrepancias.
- c) Pruebe pensando en evitar cualquier tipo de consecuencia directa o indirecta.

Es necesario recordar que el mantenimiento además de correcciones de problemas incluye cambios, crecimientos, adecuaciones, actualizaciones y revisiones de la

aplicación para "preservar" su función operativa y productiva de manera oportuna y veraz.

Una vez que una aplicación de software es liberada a operación, es menester pensar que los reportes de fallas y modificaciones del sistema se presentarán como un resultado normal del uso del mismo esto contribuirá a que el ciclo de vida del sistema sea estable y sobretodo controlable; todo esto como consecuencia de la dinámica y versatilidad de la utilización del cómputo, pero siempre con la idea de que es un medio y no un fin.

#### **4.1.4. Consistencia de la Información**

Para tener una buena consistencia de la información en la base de datos hay que considerar los aspectos de la figura 4.1.4.1 que corresponden también a tener una buena administración de bases de datos, estos conceptos se explican a continuación:

##### **Administración de Bases de Datos**

Son las actividades requeridas para preservar la integridad y seguridad de una base de datos, como mantener los permisos de usuario, realizar copias de seguridad, compactar y reparar la base de datos.

##### **Base de Datos**

Conjunto de datos relacionados con un tema o propósito determinado. Una base de datos *contiene* tablas y también puede contener consultas e índices, así como relaciones entre tablas, criterios de validación de campos y tablas y vinculaciones con fuentes de datos externas.

##### **Integridad Referencial**

Reglas que se siguen para preservar las relaciones definidas entre las tablas cuando se introducen o eliminan registros. Si se exige la integridad referencial, Microsoft Access

impide agregar registros a una tabla relacionada cuando no existe un registro asociado en la tabla principal, realizar cambios en los valores de la tabla principal que resultarían en registros huérfanos en una tabla relacionada y eliminar registros de la tabla principal cuando existen registros relacionados coincidentes en una tabla relacionada.

Si selecciona para una relación las opciones "Actualizar en cascada los campos relacionados" o "Eliminar en cascada los registros relacionados", Microsoft Access permite realizar cambios y eliminaciones en la tabla principal, pero cambia o elimina los registros relacionados de la tabla relacionada para garantizar que las reglas se sigan cumpliendo.

La integridad referencial es un sistema de reglas que Microsoft Access utiliza para asegurarse de que las relaciones entre registros de tablas relacionadas son válidas y de que no se eliminan o cambian accidentalmente datos relacionados. Se puede establecer la integridad referencial cuando todas las condiciones siguientes se cumplen:

- El campo coincidente de la tabla principal es una clave principal o tiene un índice único.
- Los campos relacionados tienen el mismo tipo de datos.
- Ambas tablas pertenecen a la misma base de datos de Microsoft Access. Si las tablas están vinculadas, deben ser tablas de Microsoft Access. La integridad referencial no se puede aplicar a las tablas vinculadas de bases de datos de otros formatos.

Al aplicar la integridad referencial debe seguir las siguientes reglas:

- No se puede introducir un valor en el campo de la clave externa de la tabla relacionada que no exista en la clave principal de la tabla principal. Sin embargo, se puede introducir un valor nulo en la clave externa, indicando que los registros no están relacionados. Por ejemplo, no puede haber un pedido asignado a un cliente

que no existe, pero puede haber un pedido que no esté asignado a nadie introduciendo un valor nulo en el campo Id de cliente.

No se puede eliminar un registro de una tabla principal si existen registros coincidentes en una tabla relacionada. Por ejemplo, no se puede eliminar un registro de cliente de la tabla Clientes si hay pedidos asignados a ese cliente en la tabla Pedidos.

- No se puede cambiar el valor de una clave principal en la tabla principal si ese registro tiene registros relacionados. Por ejemplo, no puede cambiar el identificador de un cliente de la tabla Clientes si existen pedidos asignados a ese cliente en la tabla Pedidos.

Si desea que Microsoft Access le aplique estas reglas a una relación, seleccione la opción "Exigir la integridad referencial" al crear la relación. Si la integridad referencial está vigente y se rompe una de las reglas con las tablas relacionadas, Microsoft Access presenta un mensaje y no permite el cambio.

Puede ignorar las restricciones en contra de eliminar o cambiar registros relacionados y conservar la integridad referencial activando las casillas "Actualizar en cascada los campos relacionados" y "Eliminar en cascada los campos relacionados". Cuando la casilla "Actualizar en cascada los campos relacionados" está activada, cambiar el valor de una clave principal de la tabla principal actualiza automáticamente el valor coincidente en todos los registros relacionados. Cuando la opción "Eliminar en cascada los campos relacionados" está activada, al eliminar un registro de la tabla principal se eliminan todos los registros relacionados de la tabla relacionada. Para obtener más información sobre las opciones "Actualizar en cascada los campos relacionados" y "Eliminar en cascada los campos relacionados".

Para obtener información sobre cómo editar una relación existente y establecer la integridad referencial.

Para obtener información sobre cómo definir una nueva relación.

### **Utilizar las opciones “Actualizar en Cascada los Campos Relacionados” o “Eliminar en Cascada los Campos Relacionados”**

Para aquellas relaciones en las que la integridad referencial se aplica, se puede especificar si se desea que Microsoft Access automáticamente elimine y actualice en cascada operaciones para registros relacionados. Si activa estas opciones, las operaciones de eliminación y actualización que normalmente no serían permitidas por la integridad referencial lo serán. Cuando elimina registros o cambia el valor de las claves principales de una tabla principal, Microsoft Access hace los cambios necesarios en las tablas relacionadas para preservar la integridad referencial.

Si activa la casilla de verificación "Actualizar en cascada los campos relacionados" al definir una relación, cada vez que cambie la clave principal de un registro de la tabla principal, Microsoft Access actualizará automáticamente la clave principal al nuevo valor en todos los registros relacionados. Por ejemplo, si cambia un identificador de cliente en la tabla Clientes, el campo Id de cliente en la tabla Pedidos es actualizado automáticamente para cada uno de los pedidos de ese cliente de forma que la relación no se rompa. Microsoft Access actualiza en cascada sin presentar ningún mensaje.

Si activa la casilla de verificación "Eliminar en cascada los campos relacionados" al definir una relación, cada vez que elimine registros de la tabla principal, Microsoft Access eliminará automáticamente los registros relacionados de la tabla relacionada. Por ejemplo, si elimina el registro de un cliente de la tabla Clientes, todos los pedidos de ese cliente son eliminados automáticamente de la tabla Pedidos (incluidos los registros de la tabla Detalles de pedidos relacionados con los registros de Pedidos). Cuando elimine registros de un formulario u hoja de datos con la opción "Eliminar en cascada los campos relacionados" activada, Microsoft Access le avisará que los registros relacionados pueden ser eliminados. Sin embargo, cuando elimine registros mediante una consulta de eliminación, Microsoft Access eliminará automáticamente los registros

de las tablas relacionadas sin presentar ningún aviso. esto para obtener información sobre cómo editar una relación existente y establecer la integridad referencial.

### **Determinar las Relaciones de la Base de Datos**

Después de haber creado las diferentes tablas para cada asunto de la base de datos, necesita una forma de decirle a Microsoft Access cómo ha de combinar la información. El primer paso de este proceso es definir relaciones entre las tablas. Una vez hecho esto, puede crear consultas, formularios e informes para mostrar información de varias tablas a la vez. Por ejemplo, este formulario incluye información de estas tablas:

### **¿Cómo funcionan las Relaciones?**

Una relación trabaja haciendo coincidir los datos de los campos clave (normalmente un campo con el mismo nombre en ambas tablas). En la mayoría de los casos, esta será la "clave principal" de una tabla, que proporciona un identificador único para cada registro y que coincide con una entrada de la "clave externa" de la otra tabla. Por ejemplo, los clientes se pueden asociar con los pedidos de los que requirieron creando una relación entre los campos Id de cliente.

### **Relación Uno a Varios**

Una relación uno a varios es el tipo de relación más común. En este tipo de relaciones, un registro de la Tabla A puede tener muchos registros coincidentes en la Tabla B, pero un registro de la Tabla B no tiene más de un registro coincidente en la Tabla A. Por ejemplo, las tablas Proveedores y Productos tienen una relación uno a varios: cada proveedor suministra varios productos, pero cada producto sólo es de un proveedor.

### **Relación Varios a Varios**

En una relación varios a varios, un registro de la Tabla A puede tener varios registros coincidentes en la Tabla B y viceversa. Esto sólo es posible si se define una tercera tabla (denominada tabla unión) cuya clave principal consta de al menos dos campos: las claves externas de las Tablas A y B. Por ejemplo, las tablas Pedidos y Productos

tienen una relación varios a varios que se define creando dos relaciones uno a varios con la tabla Detalles de pedidos.

### **Relación Uno a Uno**

En una relación uno a uno, un registro de la Tabla A no puede tener más de un registro coincidente en la Tabla B y viceversa. Este tipo de relación no es habitual, debido a que la información relacionada de esta forma podría estar en una sola tabla. Podría usar la relación uno a uno para dividir una tabla grande, aislar parte de una tabla por motivos de seguridad o almacenar información que sólo se aplica a un subconjunto de la tabla principal. Por ejemplo, podría crear una tabla que registrara los empleados participantes en un partido de fútbol.

### **Definir Relaciones**

Defina una relación agregando las tablas que desea relacionar a la ventana Relaciones y arrastrando el campo clave de una tabla y soltándolo en el campo clave de la otra tabla.

El tipo de relación que crea Microsoft Access depende de cómo están definidos los campos relacionados.

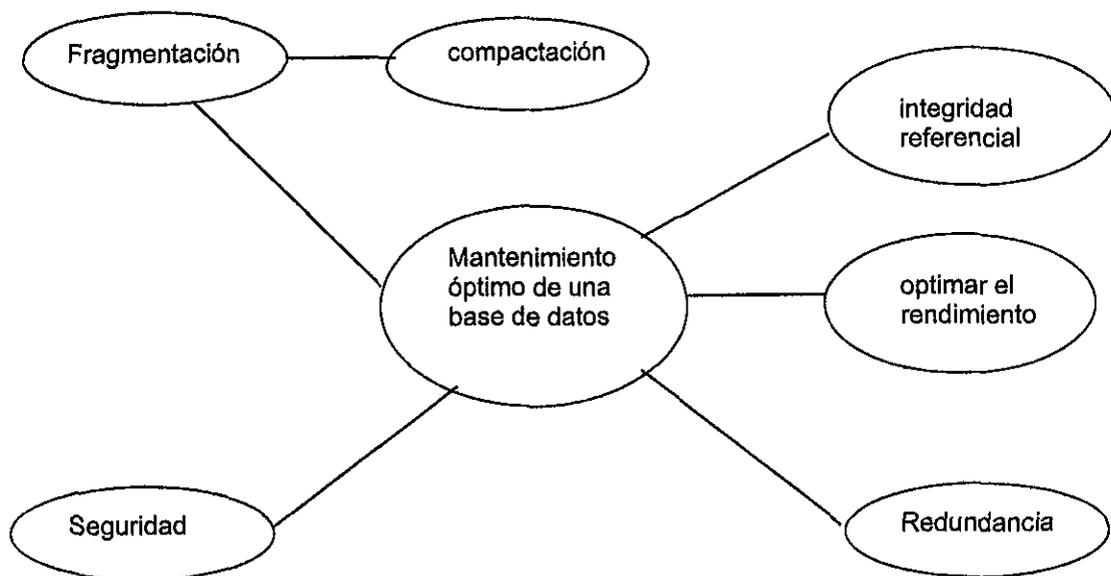
- Se crea una relación uno a varios si uno de los campos relacionados es una clave principal o tiene un índice único.
- Se crea una relación uno a uno si ambos campos relacionados son claves principales o tienen índices únicos.
- Una relación varios a varios es, en realidad, dos relaciones uno a varios con una tercera tabla cuya clave principal consta de dos campos: las claves externas de las otras dos tablas.

Nota: Si arrastra un campo que no sea una clave principal y no tenga un índice único a otro campo que tampoco sea una clave principal ni tenga un índice único, se crea una relación indeterminada. En las consultas que contengan tablas con una relación

indeterminada, Microsoft Access muestra una línea de unión predeterminada entre las tablas, pero no se exigirá integridad referencial y no habrá garantías de que los registros sean únicos en ninguna tabla.

### Fragmentación

Si borra tablas, su base de datos puede fragmentarse, lo que provoca un uso ineficiente del espacio en disco, a continuación se presenta la compactación como una una solución a este problema.



**Figura 4.1.4.1** Conceptos a considerar para tener un mantenimiento óptimo de una base de datos.

### Compactación

Compactar una base de datos para defragmentar el archivo y obtener espacio adicional en el disco

Si borra tablas, su base de datos puede fragmentarse, lo que provoca un uso ineficiente del espacio en disco. Al compactar la base de datos, se hace una copia de la misma y se reorganiza el espacio que ocupa en el disco.

1. Cierre la base de datos. Si se encuentra en un entorno multiusuario, asegúrese de que todos los usuarios hayan cerrado la base de datos.
2. En el menú Herramientas, elija Utilidades de la base de datos y después elija Compactar base de datos.
3. En el cuadro de diálogo Base de datos a compactar, especifique la base de datos que desea compactar.
4. En el cuadro de diálogo Compactar la base de datos, especifique el nombre, unidad y carpeta donde desea almacenar la base de datos compactada.

Se puede utilizar el mismo nombre para la base de datos compactada que para la base de datos de origen o bien se puede cambiar el nombre si desea crear un archivo diferente. Si usa el mismo nombre, unidad y directorio y si la base de datos se compacta con éxito, Microsoft Access reemplazará automáticamente el archivo de origen por la versión compactada.

#### Notas:

- Si elimina registros del final de una tabla que contiene un campo Autonumérico, al compactar la base de datos Microsoft Access le dará al próximo campo Autonumérico el valor del último contador sin borrar, más 1.
- Compactar una base de datos de las versiones 1.x ó 2.0 no la convierte al formato de Microsoft Access 95. Para convertir una base de datos de las versiones 1.x ó 2.0, cierre la base de datos; en el menú Herramientas elija Utilidades de la base de datos y luego elija "Convertir base de datos".

## **Optimar el Rendimiento General de las Tablas Evitando la Redundancia**

Puede conseguir los mejores resultados aplicando las siguientes indicaciones en el diseño de las tablas:

- Diseñe tablas sin datos redundantes. Una base de datos bien diseñada es un requisito previo para la recuperación y almacenamiento de datos. Si las tablas contienen datos redundantes, puede usar el Asistente para analizar tablas para dividir las tablas en tablas relacionadas y así almacenar los datos más eficientemente. Si desea obtener más información.
- Elija los tipos de campo apropiados. Puede ahorrar espacio en la base de datos y mejorar las operaciones de combinación eligiendo los tipos de campo apropiados. Al definir un campo, elija el tipo de datos mejor o Tamaño del campo (FieldSize) que mejor se adecúe a los datos del campo. Si desea obtener más información.
- Se crean unos índices para los campos que ordene, o combine o para los que establezca criterios. Puede mejorar considerablemente la velocidad de las consultas si ordena los campos a ambos lados de la combinación o si crea una relación entre esos campos y ordena alguno de los campos utilizado para establecer los criterios de la consulta. La búsqueda de registros mediante el cuadro de diálogo también es más rápida al buscar en un campo indexado.
- Los índices ocupan espacio en el disco y ralentizan la adición, eliminación y actualización de registros. En la mayoría de los casos, las ventajas de la velocidad de los índices en la recuperación de datos superan ampliamente las desventajas. Sin embargo, si la aplicación actualiza datos frecuentemente o tiene problemas de espacio de disco, sería aconsejable que limitara el número de índices.
- Además de tener en cuenta éstas indicaciones, antes de construir la base de datos, posteriormente se puede utilizar el Asistente para Rendimiento, que a continuación se describe:

## **Optimar el Rendimiento de la Base de Datos con el Asistente para Rendimiento**

1. Abra la base de datos que desee optimar.
2. En el menú Herramientas, elija Analizar y Rendimiento.
3. En la lista "Tipo de objeto", haga clic en el tipo de objeto de la base de datos que desee optimar. (Seleccione "Todos" para ver una relación de todos los objetos)
4. En la lista "Nombre del objeto", seleccione los objetos de la base de datos que desee optimar. (Haga clic en "Seleccionar todos" para seleccionar todos los objetos de la base de datos de ese tipo)
5. Repita los pasos 3 y 4 hasta que haya seleccionado todos los objetos que desee seleccionar y presione "Aceptar".

Se muestran tres tipos de optimización: Recomendación, Sugerencia e Idea. Al hacer clic en un elemento de la lista, se muestra información sobre la optimización propuesta en el cuadro "Notas del análisis" por debajo de la lista. Las sugerencias tienen unos inconvenientes adicionales que debería tener en cuenta antes llevarlas a cabo. Para ver una descripción de los inconvenientes, seleccione una sugerencia de la lista y lea la información que se presenta en el cuadro "Notas del análisis". Microsoft Access puede realizar las optimizaciones que sean recomendaciones y sugerencias, mientras que las ideas las tiene que realizar el usuario.

6. Seleccione cada recomendación o sugerencia que desee llevar a cabo y presione "Optimar".

Para llevar a cabo todas las recomendaciones o sugerencias, haga clic en "Seleccionar todos" y presione "Optimar". Para llevar a cabo una idea, seleccione esa optimización y siga las instrucciones que se muestran en el cuadro "Notas del análisis".

Nota: El Asistente para rendimiento no proporciona sugerencias sobre cómo mejorar el rendimiento de Microsoft Access o el sistema en el que se está ejecutando. Si desea obtener información sobre otras sugerencias acerca de cómo mejorar el rendimiento de otras áreas que no cubre el Asistente para rendimiento.

## **Seguridad**

Asegurar una base de datos. Microsoft Access ofrece dos métodos de asegurar una base de datos: *habilitando una contraseña para abrir una base de datos o mediante seguridad a nivel de usuario, que puede ser utilizada para limitar qué parte de la base de datos puede ser accedida o cambiada por el usuario.*

### **Habilitar una Contraseña**

El método más simple es habilitar una contraseña para abrir la base de datos. Una vez que se haya habilitado la contraseña, se mostrará un cuadro de diálogo que pida la contraseña cada vez que se abra la base de datos. Sólo los usuarios que escriban la contraseña correcta podrán abrir la base de datos. Este método es seguro (Microsoft Access encripta la contraseña de forma que se pueda acceder leyendo directamente el archivo de base de datos), pero sólo es aplicable para abrir una base de datos. Una vez que se ha abierto una base de datos, todos los objetos están disponibles para el usuario (a menos que se haya definido la seguridad a nivel de usuario, como se describe posteriormente en este tema). Para una base de datos que esté compartida entre un pequeño grupo de usuarios o sobre un sólo equipo, la habilitación de una contraseña es todo lo que se requiere.

**Precaución** No utilice una contraseña de base de datos si va a replicar una base de datos. Las bases de datos replicadas no pueden ser sincronizadas si están definidas las contraseñas de las bases de datos.

### **Seguridad a nivel de Usuario**

El método más flexible y extendido de asegurar una base de datos es llamado seguridad a nivel de usuario. Esta forma de seguridad es similar a los métodos usados en la mayoría de los sistemas de red. Los usuarios son obligados a identificarse y escribir una contraseña cuando inician Microsoft Access. Dentro del archivo de información de grupo de trabajo, están identificados como miembros de un grupo. Microsoft Access ofrece dos grupos predeterminados: Administradores (llamados el

grupo de Administradores) o Usuarios (llamados el grupo de Usuarios), aunque se pueden definir grupos adicionales.

Los permisos se conceden a los grupos y usuarios para regular cómo se les permite trabajar con cada objeto en una base de datos. Por ejemplo, a los miembros del grupo Usuarios puede que se les permita examinar, introducir o modificar datos en una tabla Clientes, pero no se les permita cambiar el diseño de esa tabla. Al grupo Usuarios se le podría permitir que sólo examinaran los datos de una tabla que contenga datos de pedidos y se les negara totalmente el acceso a una tabla Nóminas. Los miembros del grupo de Administradores tienen todos los permisos sobre todos los objetos de una base de datos. Se pueden instalar controles más ajustados creando sus propias cuentas de grupo, asignar los permisos adecuados a esos grupos y después agregar usuarios a esos grupos.

Las tres razones principales para utilizar la seguridad a nivel de usuario son para:

- Proteger la propiedad intelectual de su código.
- Impedir que los usuarios rompan inadvertidamente una aplicación cambiando código de objetos de los que depende la aplicación.
- Proteger los datos sensibles de la base de datos.

## **CONCLUSIONES**

## **Conclusiones:**

La solución de cualquier problema de Ingeniería, precisa conocer a fondo todos los aspectos involucrados para obtener la solución óptima. En nuestro caso fue necesario realizar una ardua tarea de investigación de las necesidades y la forma de operación de la empresa farmacéutica.

La realización del presente trabajo nos permitió aplicar los conocimientos adquiridos a lo largo de nuestra formación profesional a un problema real, orientado a la administración de productos dentro de una empresa farmacéutica. Además nos permitió afianzar, actualizar y agrandar los conocimientos teóricos sobre bases de datos y lenguajes de alto nivel.

Por otra parte, analizamos que la aplicación de la ingeniería de software a un problema real, es el medio más adecuado para solucionar todos los problemas que resulten del desarrollo de sistemas. Su aplicación además de ser extremadamente útil resulta muy enriquecedora.

Los programas empleados para el diseño del sistema son del dominio público, esto facilita su implantación en cualquier empresa. Además, el sistema es actualizable, se puede adecuar fácilmente a las necesidades específicas de cada empresa e implementarse en Internet.

El sistema cumple con las expectativas originales, ahora para su óptimo aprovechamiento es necesario cuidar la administración de la base de datos, como se indica en el capítulo correspondiente. Los datos contenidos en el sistema pueden migrarse fácilmente a otras bases de datos o importarse desde otras.

El sistema a automatizado el control de pedidos de esta empresa haciéndolo más rápido, confiable y con información actualizada, lo que representa un factor muy importante en la toma de decisiones de dicha empresa.

Durante el desarrollo del sistema nos percatamos de la importancia del análisis de datos, en este periodo encontramos conflictos con la organización y seguimiento de la información para las etapas iniciales de la automatización. En conclusión decimos que un buen análisis anula tiempos indeseados y se tiene un buen desarrollo del sistema.

Queremos resaltar que para desarrollar un sistema computarizado se requiere además de tener conocimientos profesionales, de una gran disciplina para realizar fase por fase el ciclo de vida de un sistema y con ello una organización, criterio y experiencia para resolver los conflictos que se presenten durante el desarrollo del sistema.

## **BIBLIOGRAFÍA**

## **Bibliografía:**

Introducción a la Programación de Visual Basic  
Kenyon Brown  
Grupo Noriega, 1994

Software Orientado a Objetos  
Ann L. Winblad  
Addison-Wesley, 1993

Programmer's Guide Visual Basic  
Microsoft

Lenguaje Reference Visual Basic  
Microsoft

Análisis y Diseño Orientado a objetos  
James Martin, James J. Odell  
*Prentice Hall Hispanoamericana S. A.*, 1974

Análisis y Diseño de Sistemas  
Kendall y Kendall  
Editorial Prentice Hall, 1991

Herramientas CASE-Metodología estructurada para el desarrollo de los sistemas  
William S. Davis  
Editorial Paraninfo

CASE Method  
Richard Borker  
Editorial Addison-Wesley

Ingeniería de Software  
Richard Fairley  
McGraw-Hill, 1994

Ingeniería de Software  
Richard Fairley  
Editorial MacGraw Hill  
1a. edición 1987

Ingeniería de Software, un enfoque práctico  
Roger S. Pressman  
Tercera edición, McGraw-Hill

Ingeniería de Software  
Ian Somer Ville  
Editorial Addison Wesley  
Iberoamericana  
2a. edición 1987.

Case la automatización del software  
Car McClure  
Serie Paradigma, Addison-Wesley Iberoamericana

Remington Farmacia  
Vol. 1,17 Edición  
Edit. Panamericana

Remington Farmacia  
Vol. 2, 17 Edición  
Edit. Panamericana

Indice de Medicamentos  
Barbara F. Mcvan  
Editorial Médica

Farmacotecnia Teoría y Práctica  
Tomo 1  
Jose Helman  
Editorial Continental

Martin James  
Organización de Base de Datos  
Prentice Hall, 1985

Brathwaite, Kenneth S.  
System Design in a Database Environment, 1st. Edition  
McGraw Hill Ranade IBM series 1989 cap. 3

Browning Dave  
Data Manager and Lans PC  
Tech Journal , Mayo 1987 (revista)  
pag. 56-70

**APÉNDICE A**  
**CÓDIGO DE PROGRAMACIÓN**

## PROGRAMACIÓN DEL SOFTWARE DEL SISTEMA

**\*\*\*\* Función de apertura de la Base de Datos a partir de un archivo de inicialización \*\*\*\***

Function Conecta\_db()

Dim basedatos As String

On Error GoTo controla

```
basedatos = ReadFromIni$("SERVIDOR", "DataBase", NULL_STRING)
If basedatos = "" Then
    MsgBox "no encuentre bd"
    basedatos = "produc.mdb"
End If
```

```
Path_db = ReadFromIni$("TRAYECTORIA", "PATH", NULL_STRING)
If Path_db = "" Then
    MsgBox "no encuentre la ruta"
    path = "c:\medical\based\"
End If
```

```
Path_rep = ReadFromIni$("TRAYECTORIA", "PATHREP", NULL_STRING)
If Path_rep = "" Then
    MsgBox "no encuentre ruta de reportes"
    path = "c:\medical\based\"
End If
```

```
Path_bmp = ReadFromIni$("TRAYECTORIA", "PATHBMP", NULL_STRING)
If Path_bmp = "" Then
    MsgBox "no encuentre ruta de BMPS"
    path = "c:\medical\bmps\"
End If
```

```
Path_recep = ReadFromIni$("TRAYECTORIA", "PATHREC", NULL_STRING)
If Path_recep = "" Then
    MsgBox "no encuentre ruta de recepción"
End If
```

' Ruta para envío de archivos a hospitales

```
Path_envio = ReadFromIni$("TRAYECTORIA", "PATHENV", NULL_STRING)
If Path_envio = "" Then
    MsgBox "no encuentre ruta de envío de archivos"
End If
```

' Ruta para recepción de archivos SUN

```
Path_sun1 = ReadFromIni$("TRAYECTORIA", "PATHSUN1", NULL_STRING)
If Path_sun1 = "" Then
```

```

    MsgBox "no encuentre ruta de Archivo Sun"
End If

' Ruta para envío de archivos SUN
Path_sun2 = ReadFromIni$("TRAYECTORIA", "PATHSUN2", NULL_STRING)
If Path_sun2 = "" Then
    MsgBox "no encuentre ruta de archivo sun"
End If

' Ruta para activar módulo de comunicación
Path_comunica = ReadFromIni$("TRAYECTORIA", "PATHCOM", NULL_STRING)
If Path_comunica = "" Then
    MsgBox "no encuentre ruta de shell de comunicaciones"
End If

Hora_env = ReadFromIni$("ENVIO", "HORAENV", NULL_STRING)
If Hora_env = "" Then
    MsgBox "No hay hora de envío"
End If

Path_db = Path_db & basedatos

Set db_medical = OpenDatabase(Path_db)
Conecta_db = True

Exit Function
controla:
    MsgBox "ERROR NUMERO " & Err.Description
    Conecta_db = False
End Function

```

**\*\*\*\*\* Establece el permiso al Sistema \*\*\*\*\***

```

Sub Verificar_Pass(Form As Form, txt_password As String, txt_CveUsua)
'
Dim dn_TipoArc As Recordset
Dim sql_TipoArc As String

sql_TipoArc = "select * from t_usuario where id_usuario = " & txt_CveUsua & " and usua_password = " &
txt_password & " "
Set dn_TipoArc = db_medical.OpenRecordset(sql_TipoArc, dbOpenSnapshot)
If dn_TipoArc.EOF Then
    MsgBox "Password o Clave Incorrectos", vbCritical, "Verificar"
    Form.txt_password = ""
    Form.txt_CveUsua = ""
    Form.txt_CveUsua.SetFocus

```

```

Exit Sub
Else
If Form.txt_password = dn_TipoArclusua_password And Form.txt_CveUsua = dn_TipoArclid_usuario
Then
    Id_Cve_Usua = dn_TipoArclid_usuario
    vgs_Usuario = Id_Cve_Usua
    Verificar_Funciones
    Unload frm_Entrada
    frm_main.Show vbModal
End If
End If
End Sub

```

**\*\*\*\*\* Establece las funciones de cada Usuario \*\*\*\*\***

```

Sub Verificar_Funciones()
'Sistema de Autoabastecimiento Farmacéutico a Hospitales
'Establece las Funciones de cada Usuario

Dim sn_TipoFun As Recordset
Dim sql_TipoFun As String
Dim sn_DesFun As Recordset
Dim sql_DesFun As String
Dim sql_Ubicacion As String
Dim sn_Ubicacion As Recordset

sql_TipoFun = "select * from t_usuafunc where id_usuario = " & Id_Cve_Usua & " ORDER BY id_usuario"
"
Set sn_TipoFun = db_medical.OpenRecordset(sql_TipoFun, dbOpenSnapshot)
If sn_TipoFun.EOF Then
    MsgBox "No hay Registros a Cargar", vbCritical, "Confirmar"
Exit Sub
Else
sn_TipoFun.MoveLast
vl_RowTot = sn_TipoFun.RecordCount
ReDim Arr_Funciones(1 To vl_RowTot)

sn_TipoFun.MoveFirst

'Lleno el arreglo con las funciones del Usuario
vl_LinRow = 1
Do While Not sn_TipoFun.EOF
    Arr_Funciones(vl_LinRow) = sn_TipoFun!id_func_funcion
    sn_TipoFun.MoveNext
    vl_LinRow = vl_LinRow + 1
Loop
sn_TipoFun.Close

```

End If

' Busca la ubicación del Usuario

sql\_Ubicacion = "select \* from t\_usuario where id\_usuario = " & Id\_Cve\_Usua & " "

Set sn\_Ubicacion = db\_medical.OpenRecordset(sql\_Ubicacion, dbOpenSnapshot)

If sn\_Ubicacion.EOF Then

MsgBox "No hay Registros a Cargar", vbCritical, "Confirmar"

Exit Sub

Else

vgs\_Ubicacion = sn\_Ubicacion!usua\_ubicacion

End If

End Sub

#### \*\*\*\*\* Generación de archivos Txt para sistema SUN \*\*\*\*\*

Sub abre\_archivo(Form As Form)

abre\_arch = Left\$(CurDir\$, 1)

aplicacion = abre\_arch + ":"

abre\_arch = Left\$(CurDir\$, 1)

aplicacion = abre\_arch + ":"

If Right(aplicacion, 1) <> "\" Then

aplicacion = aplicacion + "\"

End If

'archivo = aplicacion + "lok.ini"

Open Path\_sun1 & Format(Date, "ddmmyy") & ".txt" For Output As #1

End Sub

Dim J As Long

Dim I As Integer

For X = 0 To 1

vis\_sql = "select t\_cliente\_pedido.id\_prod\_producto, t\_producto.prod\_descripcion, "

vis\_sql = vis\_sql & "t\_cliente\_pedido.clpp\_fecha, t\_cliente\_pedido.clpp\_ref\_cliente, "

vis\_sql = vis\_sql & "t\_cliente\_pedido.clpp\_cantidad\_pedido, t\_cliente\_pedido.id\_cliente, "

vis\_sql = vis\_sql & "t\_cliente\_pedido.id\_alma\_almacen FROM t\_cliente\_pedido INNER JOIN "

vis\_sql = vis\_sql & "t\_producto ON t\_cliente\_pedido.id\_prod\_producto = "

vis\_sql = vis\_sql & "t\_producto.id\_prod\_producto WHERE "

vis\_sql = vis\_sql & "t\_cliente\_pedido.clpp\_estatus\_pedido = 0 and "

```

vls_sql = vls_sql & "t_cliente_pedido.clpp_estatus_producto = " & X
vls_sql = vls_sql & " ORDER BY id_clie_cliente, "
vls_sql = vls_sql & "id_alma_almacen "

```

```
Set sn_cliente_pedido = db_medical.CreateDynaset(vls_sql)
```

```
primer_reg = True
```

```
Do While Not sn_cliente_pedido.EOF
```

```

codigo_art = sn_cliente_pedido!id_prod_producto
desc_art = sn_cliente_pedido!prod_descripcion
cantidad_unid = sn_cliente_pedido!clpp_cantidad_pedido
ref_cliente = sn_cliente_pedido!clpp_ref - Cliente
Fecha = sn_cliente_pedido!clpp_fecha

```

```
If primer_reg = True Then
```

```

cuenta = 1
Cliente = sn_cliente_pedido!id_clie_cliente
almacen = sn_cliente_pedido!id_alma_almacen

```

```

abre_archivo frm_trans
genera_encabezado frm_trans

```

```

I = 1
primer_reg = False

```

```
End If
```

```

If sn_cliente_pedido!id_clie_cliente <> Cliente Or sn_cliente_pedido!id_alma_almacen <> almacen
Then

```

```

Cliente = sn_cliente_pedido!id_clie_cliente
almacen = sn_cliente_pedido!id_alma_almacen

```

```
Close #1
```

```

cuenta = cuenta + 1
abre_archivo frm_trans
genera_encabezado frm_trans

```

```
I = 1
```

```
End If
```

```

vls_str = "D" + Space(5) + Space(4) + Space(6) + Space(2)
vls_str = vls_str & Space(1) + Space(5) + codigo_art
vls_str = vls_str & Space(8) + Space(1) + Space(1) + Space(1)
vls_str = vls_str & desc_art + Space(2) + Str(cantidad_unid) + Space(16)

```

```

vls_str = vls_str & Space(16) + Space(23) + Space(5) + Space(8)
vls_str = vls_str & Space(8) + Space(7) + Space(7) + Space(5)
vls_str = vls_str & Space(5) + Space(10) + Space(1) + Space(1)
vls_str = vls_str & Space(19) + Space(34) + Space(15) + Space(1)
vls_str = vls_str & Space(1) + Space(1) + Space(1)
vls_str = vls_str & Space(15) + Space(23) + Space(25)

```

```
Print #1, vls_str
```

```

vls_sql1 = "update t_cliente_pedido set clpp_estatus_pedido = 1 where "
vls_sql1 = vls_sql1 & "clpp_estatus_pedido = 0 And id_clie_cliente = " & Cliente & ""
vls_sql1 = vls_sql1 & " and clpp_fecha = #" & Format(Fecha, "mm/dd/yy") & "#"
vls_sql1 = vls_sql1 & " and id_arma_almacen = " & almacen & ""
vls_sql1 = vls_sql1 & " and id_prod_producto = " & codigo_art & ""

```

```

'      #" & Format(CDate(fecha_asig), "mm/dd/yy") & "#"
'update t_cliente_pedido set clpp_estatus_pedido = 1 where
'clpp_estatus_pedido = 0 And id_clie_cliente = Cliente
'and clpp_fecha = fecha
'and id_arma_almacen = almacen
'and id_prod_producto = codigo_ar

```

```
db_medical.Execute vls_sql1
```

```
I = I + 1
```

```
sn_cliente_pedido.MoveNext
```

```
Loop
```

```
Close #1
```

```
Next
```

```
End Sub
```

#### \*\*\*\*\* Generación de gráficas \*\*\*\*\*

```
'Llenado del combo de almacén
```

```

vls_sql1 = "select id_arma_almacen FROM t_almacen "
vls_sql1 = vls_sql1 & "where id_clie_cliente = " + Cliente + ""

```

```
Set sn_grafica1_pedido = db_medical.CreateSnapshot(vls_sql1)
```

```
If sn_grafica1_pedido.EOF Then
```

```
Form.grafica.visible = False
MsgBox "Este Hospital no cuenta con Almacenes"
frm_grafica.cbx_almacen.Clear
frm_grafica.cbx_producto.Clear
Exit Sub
End If
```

```
frm_grafica.cbx_almacen.Clear
```

```
Do While Not sn_grafica1_pedido.EOF
```

```
    frm_grafica.cbx_almacen.AddItem sn_grafica1_pedido!id_alm_almacen
```

```
    'frm_grafica.cbx_producto.AddItem sn_grafica1_pedido!id_prod_producto
```

```
    sn_grafica1_pedido.MoveNext
```

```
Loop
```

```
    frm_grafica.cbx_almacen.ListIndex = 0
```

```
End Sub
```

```
Sub carga_grafica(Form As Form)
```

```
'Gráfica que muestra la estadística de pedidos de productos  
'para los Hospitales
```

```
    Dim sn_grafica_pedido As Snapshot
```

```
    vls_sql = "Select id_clie_cliente FROM t_cliente"
```

```
    Set sn_grafica_pedido = db_medical.CreateSnapshot(vls_sql)
```

```
Do While Not sn_grafica_pedido.EOF
```

```
    frm_grafica.cbx_hospital.AddItem sn_grafica_pedido!id_clie_cliente
```

```
    sn_grafica_pedido.MoveNext
```

```
Loop
```

```
    frm_grafica.cbx_hospital.ListIndex = -1
```

```
Exit Sub
```

```
End Sub
```

```

Sub genera_grafica(Form As Form)
' Generación de la gráfica de las diferentes cantidades de
pedidos de los almacenes
' Graficación de datos
  Dim J As Long
  Dim I As Integer

  genera_max_min Form

  Form.grafica.visible = True

  Form.grafica.GraphStyle = 4

  Form.grafica.NumPoints = 12
  Form.grafica.NumSets = 1
  Form.grafica.AutoInc = 1
  Form.grafica.DrawMode = 1

  For I% = 1 To Form.grafica.NumPoints
    Form.grafica.GraphData = arr_cantidad(I%)
  Next I%

  Form.grafica.GraphType = 6
  Form.grafica.DrawMode = 2

End Sub

Sub genera_max_min(Form As Form)
' Generación de la gráfica de las diferentes cantidades de
' pedidos de los almacenes
' llamado de los máximos y mínimos de productos

  Cliente = Form.cbx_hospital
  almacen = Form.cbx_almacen
  producto = Form.cbx_producto

  vis_sql = "select clpr_minimo, clpr_maximo FROM t_cliente_producto "
  vis_sql = vis_sql & " WHERE id_alma_almacen = " + almacen + " and "
  vis_sql = vis_sql & " id_cle_cliente = " + Cliente + " and id_prod_producto = " + producto + ""

  Set sn_grafica4_pedido = db_medical.CreateSnapshot(vis_sql)

  If sn_grafica4_pedido.EOF Then

    Form.grafica.visible = False
    MsgBox " No existen Máximos ni Mínimos definidos"
    Form.grafica.cbx_producto.Clear
    Exit Sub
  End If

```

```

Sub genera_grafica(Form As Form)
' Generación de la gráfica de las diferentes cantidades de
pedidos de los almacenes
' Graficación de datos
  Dim J As Long
  Dim I As Integer

  genera_max_min Form

  Form.grafica.visible = True

  Form.grafica.GraphStyle = 4

  Form.grafica.NumPoints = 12
  Form.grafica.NumSets = 1
  Form.grafica.AutoInc = 1
  Form.grafica.DrawMode = 1

  For I% = 1 To Form.grafica.NumPoints
    Form.grafica.GraphData = arr_cantidad(I%)
    Next I%

  Form.grafica.GraphType = 6
  Form.grafica.DrawMode = 2

End Sub

Sub genera_max_min(Form As Form)
' Generación de la gráfica de las diferentes cantidades de
' pedidos de los almacenes
' llamado de los máximos y mínimos de productos

  Cliente = Form.cbx_hospital
  almacen = Form.cbx_almacen
  producto = Form.cbx_producto

  vis_sql = "select clpr_minimo, clpr_maximo FROM t_cliente_producto "
  vis_sql = vis_sql & " WHERE id_alma_almacen = " & almacen & " and "
  vis_sql = vis_sql & " id_clie_cliente = " & Cliente & " and id_prod_producto = " & producto & ""

  Set sn_grafica4_pedido = db_medical.CreateSnapshot(vis_sql)

  If sn_grafica4_pedido.EOF Then

    'form.grafica.visible = False
    MsgBox " No existen Máximos ni Mínimos definidos"
    'frm_grafica.cbx_producto.Clear
    Exit Sub
  End If

```

```

maximo = sn_grafica4_pedidoiclpr_maximo
minimo = sn_grafica4_pedidoiclpr_minimo

```

```

Form.maxi.Caption = maximo
Form.mini.Caption = minimo

```

```
End Sub
```

**\*\*\*\*\* Llenado de productos de acuerdo al cliente y almacén \*\*\*\*\***

```
Sub carga_cbxproductos(Form As Form)
```

```

    Cliente = Form.cbx_hospital

```

```

    vls_sql = "select id_prod_producto FROM t_cliente_producto "
    vls_sql = vls_sql & " WHERE id_alma_almacen = " + almacén + " and "
    vls_sql = vls_sql & " id_clie_cliente = " + Cliente + ""

```

```
Set sn_grafica3_pedido = db_medical.CreateSnapshot(vls_sql)
```

```

    If sn_grafica3_pedido.EOF Then
        Form.grafica.visible = False
        MsgBox " No existen Productos para este Almacen"
        frm_grafica.cbx_producto.Clear
        Exit Sub
    End If

```

```
frm_grafica.cbx_producto.Clear
```

```
Do While Not sn_grafica3_pedido.EOF
```

```

    frm_grafica.cbx_producto.AddItem sn_grafica3_pedido!id_prod_producto

```

```
    sn_grafica3_pedido.MoveNext
```

```
Loop
```

```
    frm_grafica.cbx_producto.ListIndex = 0
```

```
End Sub
```

**\*\*\*\*\* Inicialización de Variables Globales para el Sistema \*\*\*\*\***

```
'Globalización de variables
```

```

' Variables de control de los Botones Principales
Global gi_opcion As Integer

```

```

Global gi_cmd As Integer
' Variables para el direccionamiento de Base de Datos,
' Reportes y Bmp's
Global Path_db As String
Global Path_rep As String
Global Path_bmp As String
Global Path_mail As String
Global Path_envio As String
Global Path_recep As String
Global Path_sun1 As String
Global Path_sun2 As String
Global Path_comunica As String

```

```
Global Hora_env As String
```

```

Global gs_cliente As String
Global Cliente As String
Global almacen As String
Global Fecha As String
Global vis_sql As String
Global vis_sql1 As String

```

```
'Declaraciones para el uso de funciones de manipulación
'de archivos de configuración *.ini
```

```
'Declare Function WritePrivateProfileString Lib "kernel" (ByVal Seccion As String, ByVal Entrada As
String, ByVal Valor As String, ByVal Archivo As String) As Integer
Declare Function GetPrivateProfileString Lib "Kernel" (ByVal IpAppliName$, ByVal IpKeyName$, ByVal
IpDefault$, ByVal IpReturnString$, ByVal nSize%, ByVal IpFileName$) As Integer
Const GsIniFile$ = "MEDICAL.INI"
Sub act_reg_prod(Form As Form)

```

**\*\*\*\*\* Actualización(ABC) de catálogos \*\*\*\*\***

```

Dim nLinTot As Integer
Dim vis_sql As String
Dim dn_update As Dynaset

```

```

val_cam_prod frm_act_prod
If vib_valida = True Then

```

```

    vib_valida = False
    Exit Sub
End If

```

```

vis_sql = "Select * From t_producto Where id_prod_producto=" + Form.txt_cve_prod + ""
Set dn_update = db_medical.CreateDynaset(vis_sql)
If dn_update.EOF Then

```

```

Exit Sub
End If

dn_update.Edit
dn_update!prod_descripcion = Form.txt_desc_prod.Text
dn_update!prod_precio = Form.txt_precio.Text
dn_update!prod_fecha = Format(Date, "dd/mm/yyyy")
dn_update!prod_usuario = vgs_Usuario
dn_update!prod_estatus_envio = 2
dn_update.Update

dn_update.Close
FlagEdit = 0

load_prod Form
End Sub

Sub borr_reg_prod(Form As Form)
Dim nLinTot As Integer
Dim vis_sql As String
Dim dn_update As Dynaset
Dim mensaje As String
Dim l As Variant

val_cam_prod frm_act_prod
If vib_valida = True Then
    vib_valida = False
    Exit Sub
End If

vis_sql = "Select * From t_cliente_producto Where id_prod_producto = " + Form.txt_cve_prod + ""
Set dn_update = db_medical.CreateDynaset(vis_sql)
If Not dn_update.EOF Then
    mensaje = "El Producto tiene asignación, No se puede borrar, favor de borrar primero la asignación"
    l = MsgBox(mensaje, 1 + 32 + 0, "Aviso...")
    Exit Sub
End If

vis_sql = "Select * From t_producto Where id_prod_producto=" + Form.txt_cve_prod + ""
Set dn_update = db_medical.CreateDynaset(vis_sql)
If dn_update.EOF Then
    Exit Sub
End If

dn_update.Edit
dn_update!prod_fecha = Format(Date, "dd/mm/yyyy")
dn_update!prod_usuario = vgs_Usuario
dn_update!prod_estatus_envio = 3
dn_update.Update

dn_update.Close

```

```
load_prod Form
```

```
End Sub
```

```
Sub ins_reg_prod(Form As Form)
```

```
' On Error GoTo Err_Unico
```

```
Dim LinRow As Long
```

```
Dim nI As Long
```

```
Dim vls_sql As String
```

```
Dim nRowTot As Long
```

```
Dim nJ As Integer
```

```
Dim I As Variant
```

```
Dim mensaje As String
```

```
Dim dn_prod As Dynaset
```

```
Dim sn_prod As SnapShot
```

```
Dim sn_prod_ren As SnapShot
```

```
val_cam_prod frm_act_prod
```

```
If vlb_valida = True Then
```

```
    vlb_valida = False
```

```
    Exit Sub
```

```
End If
```

```
'Determino si existen o no registros por medio de la bandera Global
```

```
If FlagNumReg = 1 Then
```

```
    LinRow = 0
```

```
Else
```

```
    LinRow = UBound(ArrCatProd, 2)
```

```
End If
```

```
' Verifica que el producto fue borrado pero sin haber sido enviado a la farmacéutica y se quiere volver a dar de ALTA
```

```
vls_sql = "Select id_prod_producto From t_producto Where id_prod_producto = " +  
Form.txt_cve_prod.Text + " and prod_estatus_envio = 3"
```

```
Set sn_prod_ren = db_medical.CreateSnapshot(vls_sql)
```

```
If Not sn_prod_ren.EOF Then
```

```
    mensaje = "El Producto ya fué dado de Baja anteriormente, pero no se ha actualizado en  
West&John " + Chr(13) + Space(20) + "¿Desea volverlo a dar de Alta?"
```

```
    I = MsgBox(mensaje, 1 + 32 + 0, "Aviso...")
```

```
    If I = vbOK Then
```

```
        Form.txt_cve_prod.SetFocus
```

```
ren_reg_prod Form
```

```
End If
```

```
Exit Sub
```

```
End If
```

```
' Verifica si ya existe el producto sin haber sido borrado
```

```

vls_sql = "Select id_prod_producto From t_producto Where id_prod_producto ="" +
Fom.txt_cve_prod.Text + ""
Set sn_prod = db_medical.CreateSnapshot(vls_sql)
If Not sn_prod.EOF Then
    mensaje = "El Producto ya Existe"
    I = MsgBox(mensaje, vbInformation, "Atento...")
    Exit Sub
End If

```

```

LinRow = LinRow + 1
ReDim Preserve ArrCatProd(1 To 6, 1 To LinRow) As String
ArrCatProd(1, LinRow) = Fom.txt_cve_prod.Text
ArrCatProd(2, LinRow) = Fom.txt_desc_prod.Text
ArrCatProd(3, LinRow) = Fom.txt_precio.Text
ArrCatProd(4, LinRow) = Format(Date, "dd/mm/yyyy")
ArrCatProd(5, LinRow) = vgs_Usuario
ArrCatProd(6, LinRow) = 1 'Estatus de alta de almacén

```

```

'Antes que nada borro los datos de la tabla
vls_sql = "Select * From t_producto"
Set dn_prod = db_medical.CreateDynaset(vls_sql)
If Not (dn_prod.EOF And dn_prod.BOF) Then
    While Not dn_prod.EOF
        dn_prod.Delete
        dn_prod.MoveNext
    Wend
End If
dn_prod.Close

```

```

vls_sql = "Select * From t_producto"
Set dn_prod = db_medical.CreateDynaset(vls_sql)

```

```

'Aqui hago la insercion hacia la tabla
nRowTot = UBound(ArrCatProd, 2)
For ni = 1 To nRowTot
    dn_prod.AddNew
    dn_prodlid_prod_producto = ArrCatProd(1, ni)
    dn_prodlprod_descripcion = ArrCatProd(2, ni)
    dn_prodlprod_precio = ArrCatProd(3, ni)
    dn_prodlprod_fecha = ArrCatProd(4, ni)
    dn_prodlprod_usuario = ArrCatProd(5, ni)
    dn_prodlprod_estatus_envio = ArrCatProd(6, ni)
    dn_prod.Update
Next

```

```

'Deshabilito el campo de la clave
Fom.txt_cve_prod.Enabled = False
'Le asigno 0 a la variable para registro nuevo

```

```

FlagNumReg = 0
FlagEdit = 0
'Refresco el Truegrid

```

```
load_prod Form
```

```

'Err_Unico:
' If Err = 3022 Then
'   MsgBox "El Almacén ya Existe "
'   form.txt_alma.SetFocus
'   Exit Sub
' End If

```

```
End Sub
```

```

Sub lim_reg_prod(Form As Form)
'LIMPIEZA

```

```

    Form.txt_cve_prod.Enabled = True
    Form.txt_cve_prod = ""
    Form.txt_desc_prod = ""
'   Form.txt_precio = ""
    Form.cbh_hosp.ListIndex = 0
    Form.txt_cve_prod.SetFocus
'   form.cmd_guarda.Enabled = True

```

```
End Sub
```

```
Sub load_prod(Form As Form)
```

```
**** CARGA
```

```

'Creo un Dynaset
Dim Sql_Prod As String
Dim nLinRow As Long
Dim LinRegTot As Long
Dim I As Variant
Dim mensaje As String

```

```
vib_valida = False
```

```
Sql_Prod = "Select * From t_producto Where prod_estatus_envio <> 3 Order By id_prod_producto"
```

```
Set sn_medical = db_medical.CreateSnapshot(Sql_Prod)
```

```

If sn_medical.EOF Then
    mensaje = "No hay registros de Productos"
    I = MsgBox(mensaje, 1 + 32 + 0, "Aviso...")
    Erase ArrCatProd
    Form.txt_cve_prod.Text = ""
    Form.txt_desc_prod.Text = ""

```

```

' Form.txt_precio = ""
nLinRow = 1
ReDim ArrCatProd(1 To 5, 1 To nLinRow) As String
Form.tbl_producto.Refresh
FlagNumReg = 1
Exit Sub
End If
'LLeno un arreglo para el truegrid

nLinRow = 1
Do While Not sn_medical.EOF
  ReDim Preserve ArrCatProd(1 To 6, 1 To nLinRow) As String
  ArrCatProd(1, nLinRow) = sn_medical!ld_prod_producto
  ArrCatProd(2, nLinRow) = sn_medical!prod_descripcion
  If IsNull(sn_medical!prod_precio) Then
    ArrCatProd(3, nLinRow) = 0
  Else
    ArrCatProd(3, nLinRow) = sn_medical!prod_precio
  End If
  ArrCatProd(4, nLinRow) = sn_medical!prod_fecha
  ArrCatProd(5, nLinRow) = sn_medical!prod_usuario
  ArrCatProd(6, nLinRow) = sn_medical!prod_estatus_envio
  sn_medical.MoveNext
  nLinRow = nLinRow + 1
Loop

'Asigno el contenido del arreglo al truegrid
Form.tbl_producto.Rows = nLinRow - 1
Form.tbl_producto.RowIndex = 1 'nLinRow - 1
Form.tbl_producto.EditActive = False
Form.tbl_producto.Refresh

End Sub

Sub ren_reg_prod(Form As Form)
'RENOVACION
  Dim nLinTot As Integer
  Dim vis_sql As String
  Dim dn_update As Dynaset

  val_cam_prod frm_act_prod
  If vib_valida = True Then
vib_valida = False
  Exit Sub
  End If

  vis_sql = "Select * From t_producto Where id_prod_producto=" + Form.txt_cve_prod + ""
  Set dn_update = db_medical.CreateDynaset(vis_sql)
  If dn_update.EOF Then
  Exit Sub
  End If

```

```

dn_update.Edit
dn_update!prod_descripcion = Form.txt_desc_prod.Text
dn_update!prod_precio = Form.txt_precio.Text
dn_update!prod_fecha = Format(Date, "dd/mm/yyyy")
dn_update!prod_usuario = vgs_usuario
dn_update!prod_estatus_envio = 2
dn_update.Update

```

```
dn_update.Close
```

```
load_prod Form
```

```
End Sub
```

```
Sub val_cam_prod(Form As Form)
```

**\*\*\*\*\* Validación \*\*\*\*\***

```
'Se valida que no estén vacios los campos
```

```
Dim i As Variant
```

```
Dim mensaje As String
```

```
If Form.txt_cve_prod.Text = "" Or Form.txt_desc_prod.Text = "" Or Form.txt_precio = "" Then
    mensaje = "Datos Incompletos"
```

```
    i = MsgBox(mensaje, vbInformation, "Aviso...!")
```

```
    vib_valida = True
```

```
End If
```

```
End Sub
```

```
' Parametros : forma frm_detalle_prod
```

```
Sub act_reg_clieprod(Form As Form)
```

```
    Dim vis_sql As String
```

```
    Dim dn_update As Dynaset
```

```
    Dim par_clpr_producto As String
```

```
    Dim par_clie_cliente As String
```

```
    Dim par_alma_almacen As String
```

```
' Validacion de los campos modificados en el detalle del producto
```

```
    val_campos_clieprod frm_detalle_prod
```

```
If vib_valida = True Then
```

```
    vib_valida = False
```

```
    Exit Sub
```

```
End If
```

```
' Lectura del registro seleccionado
```

```

vls_sql = "Select * from t_cliente_producto where id_clpr_producto = "
vls_sql = vls_sql + Form.txt_Hospital + " And id_clie_cliente = "
vls_sql = vls_sql + Arrclie_alma(vgn_ind_cliente) + ""
vls_sql = vls_sql + " And id_alma_almacen = " + Arralma_prod(vgn_ind_almacen) + ""
Set dn_update = db_medical.CreateDynaset(vls_sql)

```

```

If dn_update.BOF And dn_update.EOF Then
    Exit Sub
End If
If dn_update!clpr_estatus_envio = 3 Then
    Exit Sub
End If

```

' Actualización del registro con los datos modificados y estatus\_envio = 2

```

dn_update.Edit
dn_update!id_prod_producto = Form.txt_desc_medical
dn_update!clpr_prod_descripcion = Form.txt_desc_hospital
dn_update!clpr_minimo = Val(Form.txt_minimo)
dn_update!clpr_maximo = Val(Form.txt_maximo)
dn_update!clpr_existenci = Val(Form.txt_existencias)
dn_update!clpr_desabasto = Val(Form.txt_desabasto)
dn_update!clpr_precio = Val(Form.txt_precio)
dn_update!clpr_descuento = Val(Form.txt_descuento)
dn_update!id_tipo_estatus = Arrtipo_pedido(Form.cmb_estatus.ListIndex + 1)
dn_update!clpr_equivalencia = Val(Form.txt_equivalencia)
dn_update!clpr_fecha = Format(Date, "dd/mm/yyyy")
dn_update!clpr_usuario = "Enrique"
dn_update!clpr_estatus_envio = 2
dn_update.Update

```

' Llamada al proceso "generación de pedido" en caso de ser requerido

```

If Val(dn_update!clpr_desabasto) > Val(dn_update!clpr_minimo) Then

```

```

    par_clpr_producto = dn_update!id_clpr_producto
    par_clie_cliente = dn_update!id_clie_cliente
    par_alma_almacen = dn_update!id_alma_almacen

```

```

    genera_cliente_pedido par_clpr_producto, par_clie_cliente, par_alma_almacen
End If

```

```

dn_update.Close

```

```

End Sub

```

```

Sub actualiza_almacen(forma As Form)

```

```

Dim vfn_i As Integer

```

```

forma.cmb_almacen.Clear

```

```

vls_sql = "Select id_alma_almacen, alma_descripcion from t_almacen "

```

```

vls_sql = vls_sql & "where id_clie_cliente = "
vls_sql = vls_sql & "" & Arrclie_arma(forma.cmb_cliente.ItemData(forma.cmb_cliente.ListIndex)) & ""
Set sn_prod_cliente = db_medical.CreateSnapshot(vls_sql)
vfn_i = 1
If Not (sn_prod_cliente.EOF And sn_prod_cliente.BOF) Then
  While Not sn_prod_cliente.EOF
    ReDim Preserve Arralma_prod(vfn_i)
    forma.cmb_almacen.AddItem sn_prod_cliente!alma_descripcion
    forma.cmb_almacen.ItemData(forma.cmb_almacen.NewIndex) = vfn_i
    Arralma_prod(vfn_i) = sn_prod_cliente!id_arma_almacen
    sn_prod_cliente.MoveNext
    vfn_i = vfn_i + 1
  Wend
  sn_prod_cliente.Close
  forma.cmb_almacen.ListIndex = 0
Else
  Erase Arrprod_cliente
  ReDim Arrprod_cliente(1 To 2, 1 To 1)
  forma.tbl_prod_clie.Rows = 1
  forma.tbl_prod_clie.RowIndex = 1
  forma.tbl_prod_clie.EditActive = False
  forma.tbl_prod_clie.Refresh
End If

```

End Sub

Sub actualiza\_producto(forma As Form)

Dim vfn\_i As Integer

```

vls_sql = "Select id_clpr_producto, clpr_prod_descripcion, clpr_estatus_envio from "
vls_sql = vls_sql & "t_cliente_producto where id_clie_cliente = "
vls_sql = vls_sql & "" & Arrclie_arma(forma.cmb_cliente.ItemData(forma.cmb_cliente.ListIndex)) & ""
vls_sql = vls_sql & "and id_arma_almacen = "
vls_sql = vls_sql & "" & Arralma_prod(forma.cmb_almacen.ItemData(forma.cmb_almacen.ListIndex)) & ""
vls_sql = vls_sql & " And clpr_estatus_envio <> 3"
Set sn_prod_cliente = db_medical.CreateSnapshot(vls_sql)
vfn_i = 1

```

If sn\_prod\_cliente.BOF And sn\_prod\_cliente.EOF Then

Erase Arrprod\_cliente()

End If

If Not (sn\_prod\_cliente.EOF And sn\_prod\_cliente.BOF) Then

While Not sn\_prod\_cliente.EOF

ReDim Preserve Arrprod\_cliente(1 To 2, 1 To vfn\_i)

Arrprod\_cliente(1, vfn\_i) = sn\_prod\_cliente!id\_clpr\_producto

Arrprod\_cliente(2, vfn\_i) = sn\_prod\_cliente!clpr\_prod\_descripcion

sn\_prod\_cliente.MoveNext

```

        vfn_i = vfn_i + 1
    Wend
Else
    Erase Amprod_cliente()
End If

sn_prod_cliente.Close
forma.tbl_prod_cliente.Rows = vfn_i - 1
forma.tbl_prod_cliente.RowIndex = vfn_i - 1
forma.tbl_prod_cliente.EditActive = False
forma.tbl_prod_cliente.Refresh

End Sub

' Dar de baja lógica un registro de la tabla
' t_cliente_producto, marcando el registro con estatus de
' de envio = 3
' parametros : forma frm_cliente_prod

Sub borr_reg_clienteprod(Form As Form)

    Dim nLinTot As Integer
    Dim vis_sql As String
    Dim dn_update As Dynaset
    Dim mensaje As String
    Dim i As Variant

    ' Lectura del registro seleccionado

    vis_sql = "Select * from t_cliente_producto where id_clpr_producto = "
    vis_sql = vis_sql + Amprod_cliente(1, Form.tbl_prod_cliente.RowIndex) + ""
    vis_sql = vis_sql + "and id_cliente_cliente = "
    vis_sql = vis_sql + Arrcliente_almacena(Form.cmb_cliente.ItemData(Form.cmb_cliente.ListIndex))
    vis_sql = vis_sql + " and id_almacena_almacen = "
    vis_sql = vis_sql + Arralmacena_prod(Form.cmb_almacena.ItemData(Form.cmb_almacena.ListIndex))
    vis_sql = vis_sql + " and clpr_estatus_envio <> 3"
    Set dn_update = db_medical.CreateDynaset(vis_sql)
    If dn_update.EOF Then
        Exit Sub
    End If

    ' Actualización del registro (baja lógica) con estatus_envio = 3

    dn_update.Edit
    dn_update!clpr_fecha = Format(Date, "dd/mm/yy")
    dn_update!clpr_usuario = "Enrique"
    dn_update!clpr_estatus_envio = 3
    dn_update.Update

```

```
dn_update.Close
```

```
' Subrutina para volver a cargar los registros de la tabla
```

```
Form.cmb_cliente.Clear
```

```
Form.cmb_almacen.Clear
```

```
load_prod_clie Form
```

```
End Sub
```

```
Sub carga_cliente(forma As Form)
```

```
Dim vfn_i As Integer
```

```
'Carga la información con base al primer cliente que encuentra
```

```
vls_sql = "Select id_clie_cliente, clie_descripcion from t_cliente"
```

```
Set sn_prod_cliente = db_medical.CreateSnapshot(vls_sql)
```

```
If Not (sn_prod_cliente.EOF And sn_prod_cliente.BOF) Then
```

```
  vfn_i = 1
```

```
  While Not sn_prod_cliente.EOF
```

```
    ReDim Preserve Arrclie_alma(vfn_i)
```

```
    forma.cmb_cliente.AddItem sn_prod_cliente!clie_descripcion
```

```
    forma.cmb_cliente.ItemData(forma.cmb_cliente.NewIndex) = vfn_i
```

```
    Arrclie_alma(vfn_i) = sn_prod_cliente!id_clie_cliente
```

```
    sn_prod_cliente.MoveNext
```

```
    vfn_i = vfn_i + 1
```

```
  Wend
```

```
  sn_prod_cliente.Close
```

```
  forma.cmb_cliente.ListIndex = 0
```

```
End If
```

```
End Sub
```

```
Sub carga_detalle(forma As Form)
```

```
Dim vln_indice As Integer
```

```
vls_sql = "SELECT t_producto.id_prod_producto, t_producto.prod_descripcion, "
```

```
vls_sql = vls_sql + "t_cliente_producto.id_clpr_producto, "
```

```
vls_sql = vls_sql + "t_cliente_producto.clpr_prod_descripcion, t_cliente_producto.clpr_minimo, "
```

```
vls_sql = vls_sql + "t_cliente_producto.clpr_maximo, t_cliente_producto.clpr_existenci, "
```

```
vls_sql = vls_sql + "t_cliente_producto.clpr_desabasto, t_cliente_producto.clpr_precio, "
```

```
vls_sql = vls_sql + "t_cliente_producto.clpr_descuento, t_almacen.alma_descripcion, "
```

```
vls_sql = vls_sql + "t_cliente_producto.clpr_equivalencia, t_cliente_producto.id_tipo_estatus, "
```

```
vls_sql = vls_sql + "t_almacen.alma_direccion, t_cliente.clie_descripcion, "
```

```
vls_sql = vls_sql + "t_tipo_pedido.id_tipo_estatus, t_tipo_pedido.tipo_descripcion "
```

```
vls_sql = vls_sql + "FROM t_tipo_pedido INNER JOIN ((t_producto INNER JOIN "
```

```
vls_sql = vls_sql + "t_cliente_producto INNER JOIN t_almacen ON "
```

```
vls_sql = vls_sql + "t_cliente_producto.id_alma_almacen = t_almacen.id_alma_almacen) "
```

```
vls_sql = vls_sql + "ON t_producto.id_prod_producto = t_cliente_producto.id_prod_producto) "
```

```
vls_sql = vls_sql + "INNER JOIN t_cliente ON t_cliente_producto.id_clie_cliente = "
```

```
t_cliente.id_clie_cliente) "
```

```
vls_sql = vls_sql + "ON t_tipo_pedido.id_tipo_estatus = t_cliente_producto.id_tipo_estatus "
```

```

vls_sql = vls_sql + "WHERE ((t_producto.id_prod_producto=[t_cliente_producto].[id_prod_producto])
AND "

```

```

vls_sql = vls_sql & "(t_cliente_producto.id_clpr_producto = " & vgs_producto & ")"

```

```

Set sn_prod_cliente = db_medical.CreateSnapshot(vls_sql)

```

```

If Not (sn_prod_cliente.EOF And sn_prod_cliente.BOF) Then

```

```

    forma.txt_Hospital = sn_prod_cliente!id_clpr_producto

```

```

    forma.txt_desc_hospital = sn_prod_cliente!clpr_prod_descripcion

```

```

    forma.txt_desc_medical = sn_prod_cliente!id_prod_producto

```

```

    forma.txt_existencias = sn_prod_cliente!clpr_existenci

```

```

    forma.txt_maximo = sn_prod_cliente!clpr_maximo

```

```

    forma.txt_minimo = sn_prod_cliente!clpr_minimo

```

```

    forma.txt_desabasto = sn_prod_cliente!clpr_desabasto

```

```

    forma.txt_precio = sn_prod_cliente!clpr_precio

```

```

    forma.txt_descuento = sn_prod_cliente!clpr_descuento

```

```

    forma.txt_direccion = sn_prod_cliente!alma_direccion

```

```

    forma.txt_almacen = sn_prod_cliente!alma_descripcion

```

```

    forma.txt_equivalencia = sn_prod_cliente!clpr_equivalencia

```

```

    For vln_indice = 0 To forma.cmb_clave.ListCount - 1

```

```

        If forma.cmb_clave.List(vln_indice) = sn_prod_cliente!prod_descripcion Then

```

```

            forma.cmb_clave.ListIndex = vln_indice

```

```

            Exit For

```

```

        End If

```

```

    Next vln_indice

```

```

    For vln_indice = 0 To forma.cmb_estatus.ListCount - 1

```

```

        If forma.cmb_estatus.List(vln_indice) = sn_prod_cliente!tipo_descripcion Then

```

```

            forma.cmb_estatus.ListIndex = vln_indice

```

```

            Exit For

```

```

        End If

```

```

    Next vln_indice

```

```

' valores para la pestaña de historico

```

```

    forma.txt_nom_prod = sn_prod_cliente!clpr_prod_descripcion

```

```

    forma.txt_nom_hospital = sn_prod_cliente!cile_descripcion

```

```

End If

```

```

sn_prod_cliente.Close

```

```

End Sub

```

```

Sub carga_historico(forma As Form)

```

```

    Dim vfn_j As Integer

```

```

    Dim vls_sql As String

```

```

Dim sn_clie_pedido As Snapshot
Dim sn_almacen As Snapshot

vls_sql = "Select * from t_cliente_pedido where id_clie_cliente = "
vls_sql = vls_sql & Arrclie_alm(vgn_ind_cliente) & " "
vls_sql = vls_sql & "and id_prod_producto = "
vls_sql = vls_sql & forma.txt_desc_medical & " "
vls_sql = vls_sql & "order by id_clie_cliente, id_alm_almacen"

Set sn_clie_pedido = db_medical.CreateSnapshot(vls_sql)

vfn_j = 1
If sn_clie_pedido.BOF And sn_clie_pedido.EOF Then
    Erase Arrhist_producto()
End If

If Not (sn_clie_pedido.EOF And sn_clie_pedido.BOF) Then
    While Not sn_clie_pedido.EOF
        ReDim Preserve Arrhist_producto(1 To 5, 1 To vfn_j)
        Arrhist_producto(1, vfn_j) = sn_clie_pedido!clpp_fecha
        vls_sql = "Select id_alm_almacen, alma_descripcion from t_almacen "
        vls_sql = vls_sql & "where id_alm_almacen = " & sn_clie_pedido!id_alm_almacen & " "
        Set sn_almacen = db_medical.CreateSnapshot(vls_sql)
        Arrhist_producto(2, vfn_j) = sn_almacen!alma_descripcion
        sn_almacen.Close
        Arrhist_producto(3, vfn_j) = sn_clie_pedido!clpp_cantidad_pedido
        Arrhist_producto(4, vfn_j) = sn_clie_pedido!clpp_cantidad_abas
        Arrhist_producto(5, vfn_j) = sn_clie_pedido!clpp_cantidad_pedido -
sn_clie_pedido!clpp_cantidad_abas
        sn_clie_pedido.MoveNext
        vfn_j = vfn_j + 1
    Wend
    Else
        Erase Arrhist_producto()
    End If

    sn_clie_pedido.Close
    forma.tbl_history.Rows = vfn_j - 1
    forma.tbl_history.RowIndex = vfn_j - 1
    forma.tbl_history.EditActive = False
    forma.tbl_history.Refresh

End Sub

Sub carga_producto(forma As Form)
Dim vfn_j As Integer
Dim sn_pedido As Snapshot
vls_sql = "Select id_prod_producto, prod_descripcion from t_producto"
Set sn_prod_cliente = db_medical.CreateSnapshot(vls_sql)
If Not (sn_prod_cliente.EOF And sn_prod_cliente.BOF) Then
    vfn_j = 1

```

```

While Not sn_prod_cliente.EOF
  ReDim Preserve Arraima_prodh(vfn_j)
  forma.cmb_clave.AddItem sn_prod_cliente!prod_descripcion
  forma.cmb_clave.ItemData(forma.cmb_clave.NewIndex) = vfn_j
  Arraima_prodh(vfn_j) = sn_prod_cliente!prod_descripcion
  sn_prod_cliente.MoveNext
  vfn_j = vfn_j + 1
Wend
End If
sn_prod_cliente.Close

vis_sql = "Select id_tipo_estatus, tipo_descripcion from t_tipo_pedido"
Set sn_pedido = db_medical.CreateSnapshot(vis_sql)
If Not (sn_pedido.EOF And sn_pedido.BOF) Then
  vfn_j = 1
  While Not sn_pedido.EOF
    ReDim Preserve Arrtipo_pedido(vfn_j)
    forma.cmb_estatus.AddItem sn_pedido!tipo_descripcion
    forma.cmb_estatus.ItemData(forma.cmb_estatus.NewIndex) = vfn_j
    Arrtipo_pedido(vfn_j) = sn_pedido!id_tipo_estatus
    sn_pedido.MoveNext
    vfn_j = vfn_j + 1
  Wend
End If
sn_pedido.Close

End Sub

```

**APÉNDICE B**  
**MANUAL DE USUARIO**

## INTRODUCCIÓN

El propósito del presente manual, es orientar al usuario en el manejo del sistema de Autoabastecimiento Farmacéutico a Hospitales, mostrando cada una de las pantallas, sus procedimientos y operatividad de las mismas.

**INDICE**

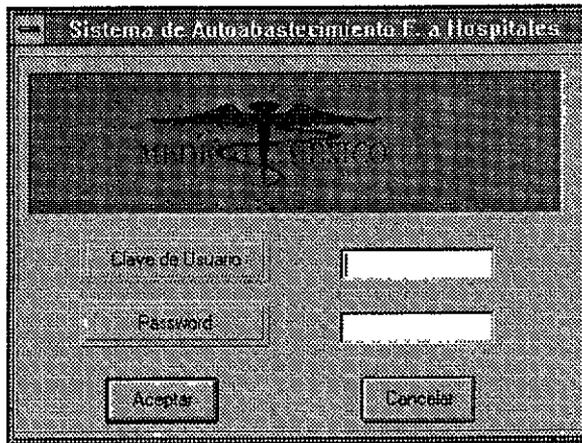
- **A. Pantalla Acceso**
- **B. Pantalla Principal**
  
- **1. Importación de Archivos**
  - 1.1 \* Archivo de Productos Consumidos por el Almacén
  - 1.2 \* Archivo de Productos de SUN
  - 1.3 \* Archivo de Movimientos de Productos
- **2. Exportación de Archivos**
  - 2.1 \* Archivo de Mantenimiento de Productos
  - 2.2 \* Archivo para Sistema SUN
- **3. Catálogos del Sistema**
  - 3.1 \* Actualizador de Productos
  - 3.2 \* Actualizador de Hospitales
  - 3.3 \* Actualizador de Almacenes
  - 3.4 \* Actualizador de Producto-Cliente
  - 3.5 \* Actualizador de Usuarios
  - 3.6 \* Actualizador de Usuario-Perfil
- **4. Impresión de Reportes**
  - 4.1 \* Listado de Producto por Cliente
  - 4.2 \* Listado de Precios de Productos por Cliente
  - 4.3 \* Listado de Pedidos
  - 4.4 \* Listado de Productos Facturados y en Backorder
  - 4.5 \* Gráfica
- **5. Ayuda General del Sistema**
- **6. Salir**

**A. Pantalla Acceso**

Es la primera pantalla en la ejecución del sistema y es en donde se debe teclear la clave del usuario y password correspondiente, posteriormente hacer clic en **Aceptar**.

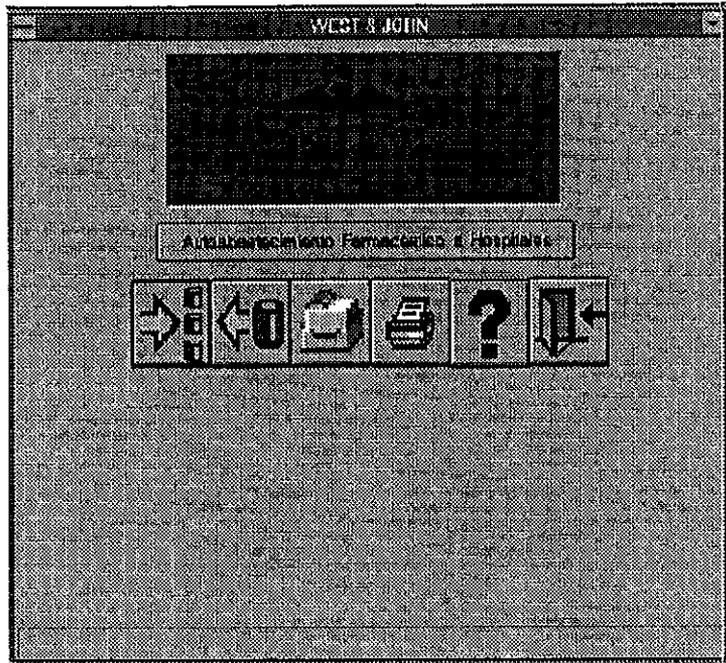
Se aceptarán tres oportunidades para teclear correctamente la clave y el password de lo contrario aparecerá el mensaje de "Acceso Denegado" terminado la ejecución del sistema.

Existe un Catálogo de Usuarios en donde se realizarán movimientos para el manejo de las claves y password de los usuarios.



## B. Pantalla Principal

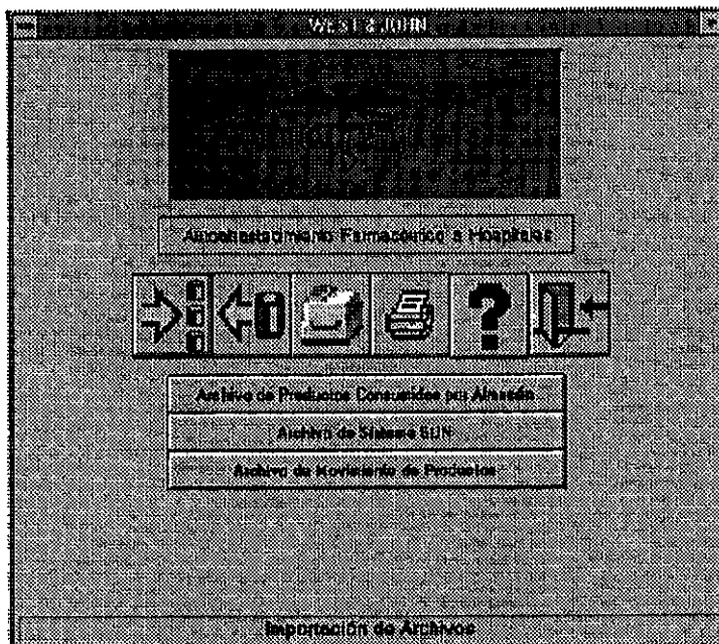
Aquí se presentan: los módulos del Sistema, cada uno de ellos se divide a su vez ( éstos se explicarán en las siguientes páginas ); el Logo de la Empresa y el Nombre del Sistema.



Nota: Al realizar un clic sobre cada botón o imagen, aparecerán en la parte inferior de la pantalla recuadros en los que se indica las opciones o submódulos del botón seleccionado.

- En la parte superior de la **Pantalla Principal** se encuentran los siguientes **botones**: (De izquierda a derecha)
  - **Importar:** Despliega 3 opciones con los archivos que va a recibir el Sistema.
    - \* Archivo de productos consumidos por el Almacén
    - \* Archivo de productos de SUN
    - \* Archivo de movimientos de productos
  - **Exportar:** Despliega las opciones de los archivos que envía West&John a Hospitales.
    - \* Archivo de mantenimiento de productos
    - \* Archivo para Sistema SUN
  - **Catálogos:** Despliega los catálogos con los que trabaja el Sistema.
    - \* Actualizador de productos
    - \* Actualizador de hospitales
    - \* Actualizador de almacenes
    - \* Actualizador de producto-cliente
    - \* Actualizador de usuarios
      - \* Actualizador de usuario-perfil
  - **Imprimir:** Despliega las opciones de los reportes del Sistema.
    - \* Listado de Producto por Cliente
    - \* Listado de Precios de Productos por Cliente
    - \* Listado de Pedidos
    - \* Listado de Productos Facturados y en Backorder
    - \* Gráfica
  - **Ayuda:** Presenta la Ayuda General del Sistema.
  - **Salir:** Salida del Sistema.

## 1. Importación de Archivos.



Al oprimir el botón de **Importación de Archivos** (primer botón de izquierda a derecha), se despliega las siguientes opciones :

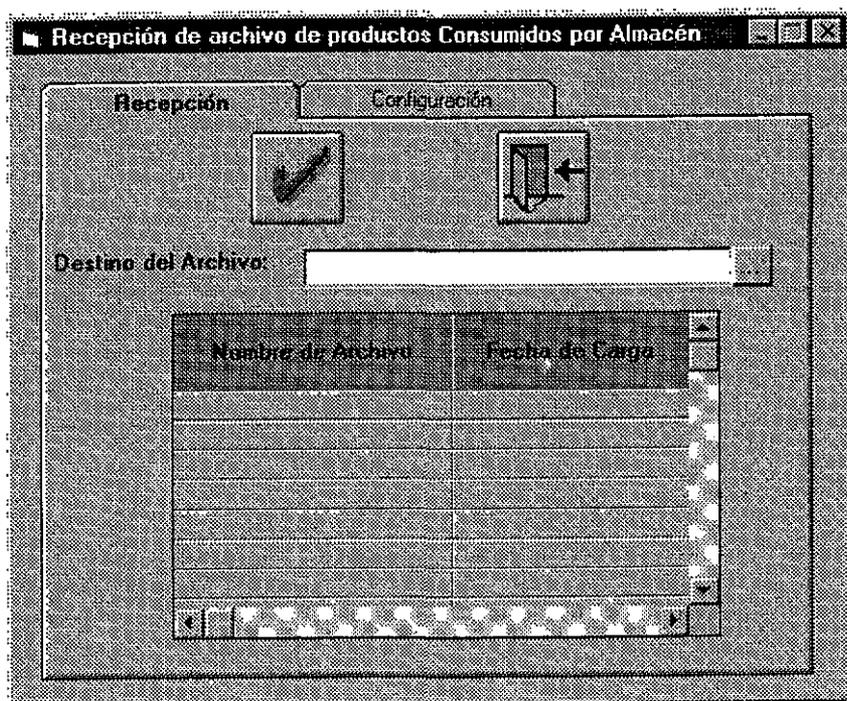
- **Archivo de productos consumidos por el Almacén**
- **Archivo de productos de SUN**
- **Archivo de movimientos de productos**

Según el archivo ha importar, deberá seleccionar mediante un clic el recuadro correspondiente.

Botón : **Importación de archivos**  
Opción : 1.1 **Archivo de productos consumidos por almacén**  
Pestaña : **Recepción**

Será activado para los usuarios de hospitales, ya que se aceptarán movimientos de los productos en el (los) almacén(s) del hospital.

Aquí se Importara el archivo de productos consumidos por el almacén, si es conocida la configuración del archivo, deberá oprimir "Aceptar" de lo contrario deberá acceder primero a la pestaña de configuración.



Se proporcionará la dirección de almacenamiento local del archivo de productos consumidos por almacén y desplegará el histórico del archivo.

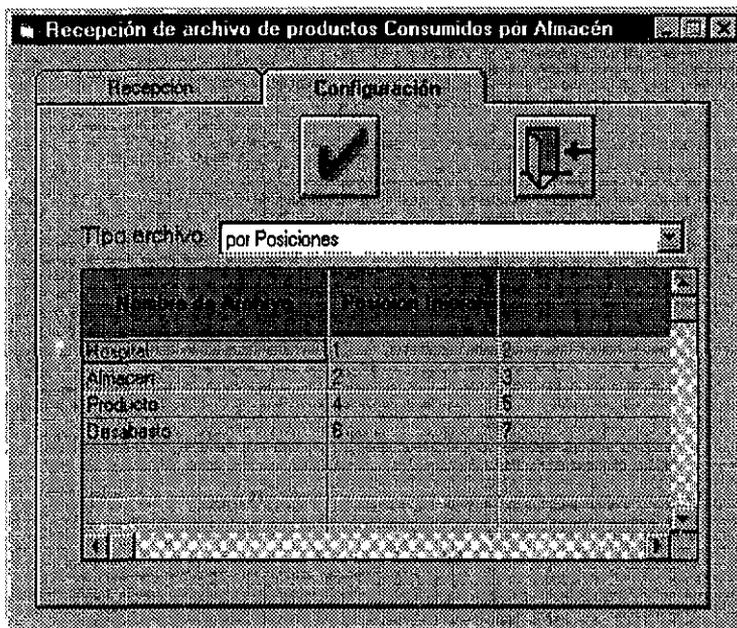
En la parte superior se muestran los siguientes botones: (De izquierda a derecha)

- **Aceptar:** Comenzará la recepción del archivo.
- **Salida:** Saldrá de la ventana actual.

### Pestaña Configuración.

Deberá ser la primera pestaña en acceder para asignar uno de los tres tipos de configuración al archivo de entrada. El acceso a dicha pestaña es mediante un clic en **configuración**, posteriormente en Tipo de archivo se seleccionará ; por Posiciones, Tabuladores o por Comas. Ya revisado el archivo, se regresa a la pestaña de **Recepción** y se oprime el botón **Aceptar**.

**Nota :** El archivo deberá estar bien detallado conforme a las posiciones que se requieren para cada campo.



Donde:

#### Por Posiciones :

Posición Inicial: Será la posición en donde empieza el campo dentro del registro  
ejem. xxxclavehospitalxxx      posición Inicial 4

Posición final: Será la posición en donde termina el campo dentro del registro  
posición Final 16

#### Por Comas y Tabuladores :

Se debe conocer la ubicación exacta de cada campo

ejem. xxx,clavehospital,xxx,producto

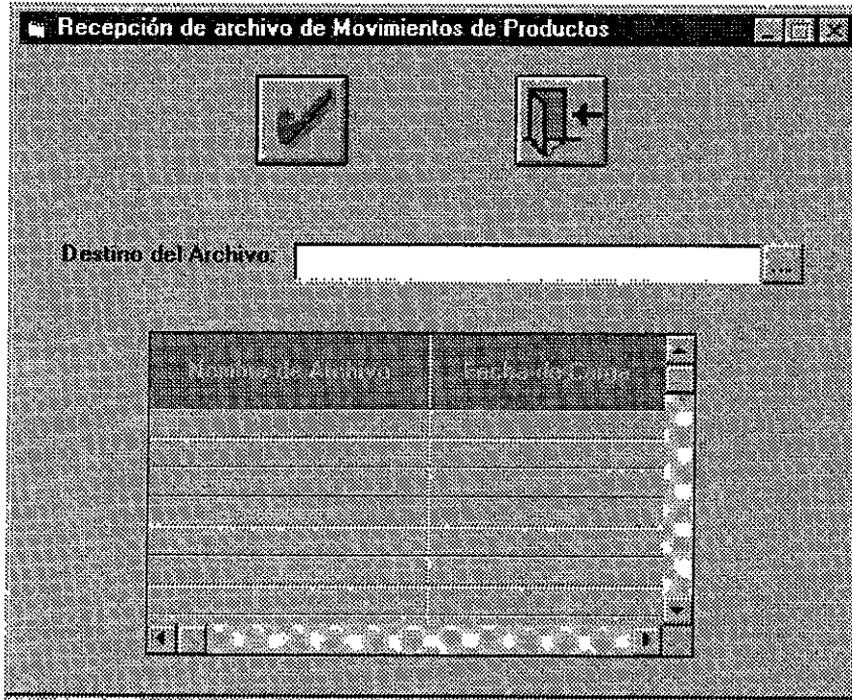
hospital posición 2

producto posición 4

Botón : **Importación de archivos**  
Opción : 1.2 **Archivo de Movimiento de Productos**

Recepción de archivo de mantenimiento de productos por el hospital

Proporciona al usuario la manera de Importar información manualmente de los archivos de mantenimiento

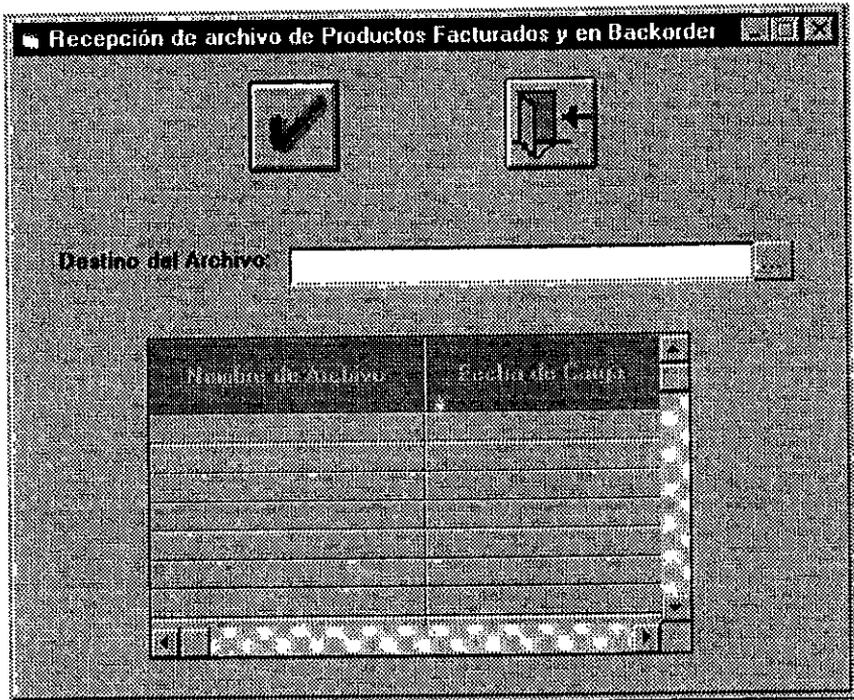


En la parte superior se muestran los siguientes botones: (De izquierda a derecha).

- **Aceptar:** Comenzará la recepción del archivo.
- **Salida:** Saldrá de la ventana actual.

Botón : **Importación de archivos**  
Opción : 1.3 **Archivo de sistema SUN**

Recepción de archivo de productos facturados y en backorder  
Proporcionar al usuario la manera de Importar información manualmente de los archivos facturados y en backorder



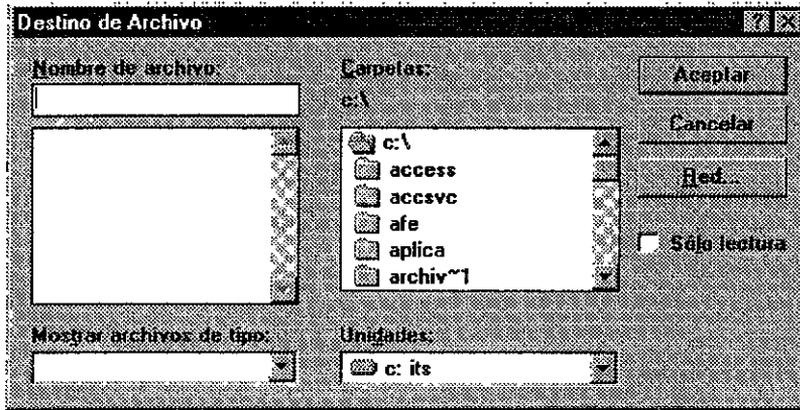
En esta ventana se proporcionará la dirección de almacenamiento local del archivo de productos facturados y en backorder y desplegará el histórico del archivo.

En la parte superior se muestran los siguientes botones: (De izquierda a derecha)

- **Aceptar:** Comenzará la recepción del archivo.
- **Salida:** Saldrá de la ventana actual.

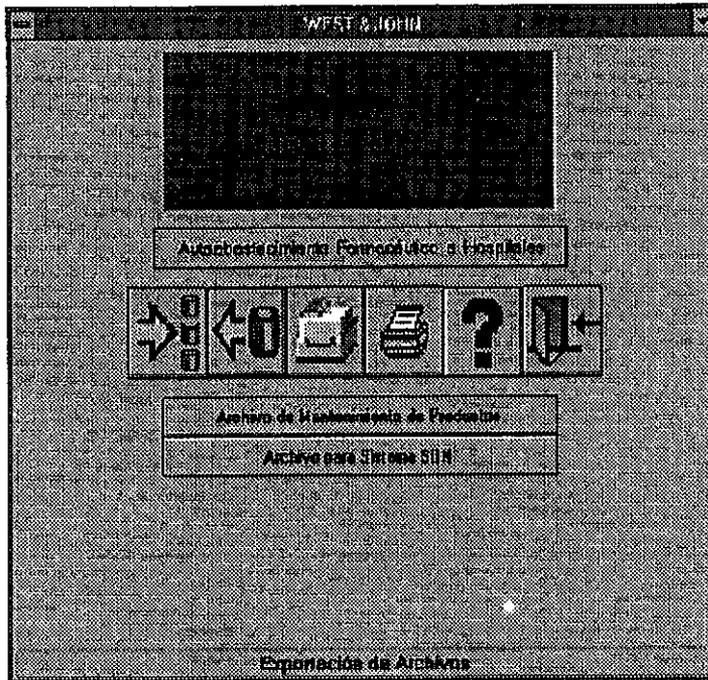
Destino del Archivo:

**Nota :** En cualquiera de las pantallas de recepción de archivo, se proporcionará la dirección oprimiendo el botón de la derecha, el cual desplegará la pantalla siguiente:



Al seleccionar el botón aceptar comenzará la recepción del archivo y éste quedará deshabilitado hasta que el proceso termine, volviéndose a activar.

## 2. Exportación de Archivos.



En esta pantalla se enviarán los archivos para West&John a los hospitales y de los hospitales a West&John, que son los archivos con los que se va alimentar la base de datos foránea.

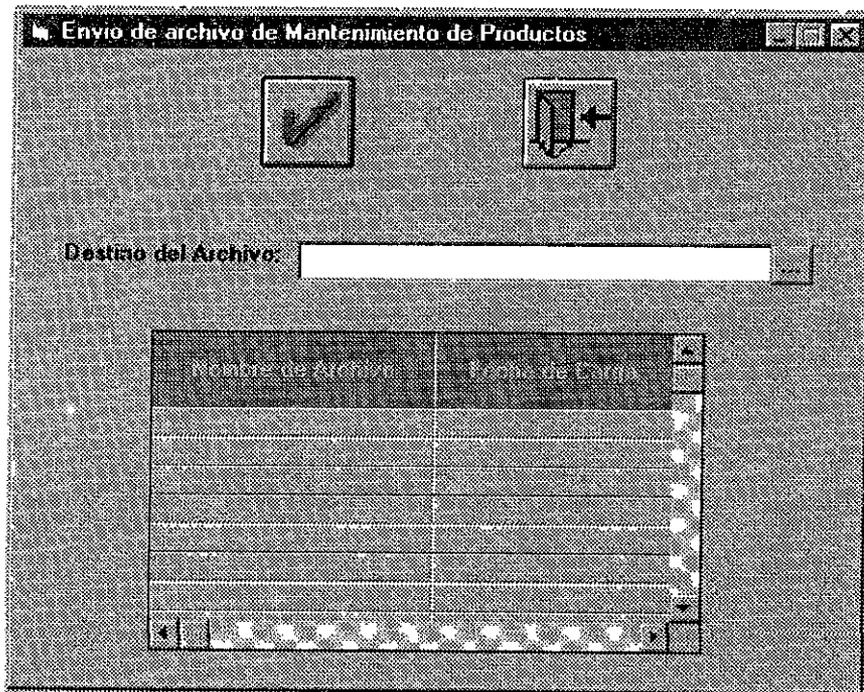
Al seleccionar el botón **Exportar** (el segundo de izquierda a derecha), aparecerá el menú con las siguientes opciones :

- **Archivo de mantenimiento de productos**
- **Archivo para Sistema SUN**

Botón : **Exportación de archivos**  
Opción : 2.1 **Archivo de Mantenimiento de productos**

Dentro de la pantalla Exportación de Archivos se proporcionará la dirección del archivo de mantenimiento de productos y desplegará el histórico del archivo.

Genera archivo de envío de productos con mantenimiento.  
Puede realizar la creación de archivo para envío de mantenimiento de productos.



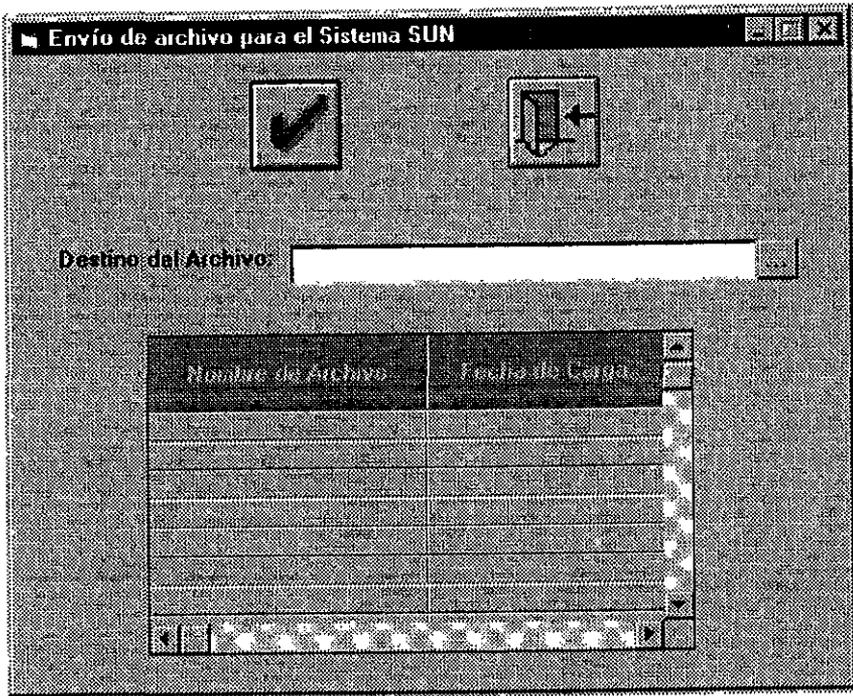
En la parte superior se muestran los siguientes botones: (De izquierda a derecha)

- **Aceptar:** Comenzará el envío del archivo.
- **Salida:** Saldrá de la ventana actual.

Botón : **Exportación de archivos**  
 Opción : **2.2 Archivo para Sistema SUN**

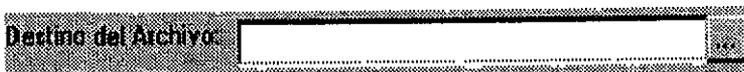
En esta pantalla se proporcionará la dirección del archivo para el Sistema SUN y desplegará el histórico del archivo. Primero deberá seleccionar el Destino del Archivo antes de oprimir el botón **Aceptar** ya que éste estará deshabilitado. Posteriormente al oprimir el botón **Aceptar** se desplegará una pantalla indicando el proceso, a su vez esta última pantalla tendrá la opción para cancelar dicho proceso.

Genera archivo para sistema SUN, permitiendo al usuario la generación manual del archivo para Sistema SUN

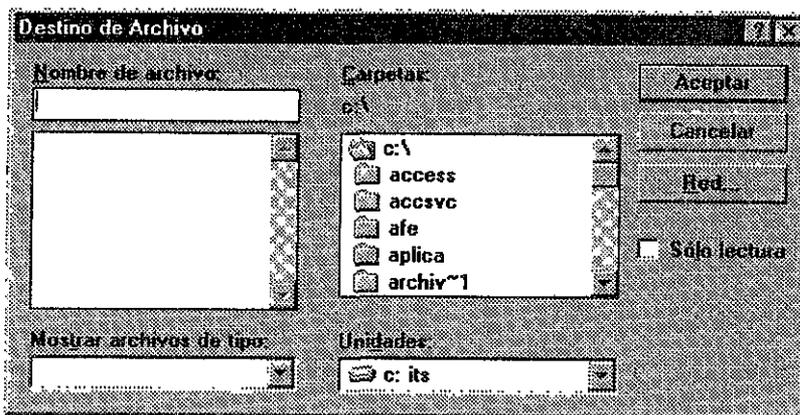


En la parte superior se muestran los siguientes botones: (De izquierda a derecha)

- **Aceptar:** Comenzará el envío del archivo.
- **Salida:** Saldrá de la ventana actual.

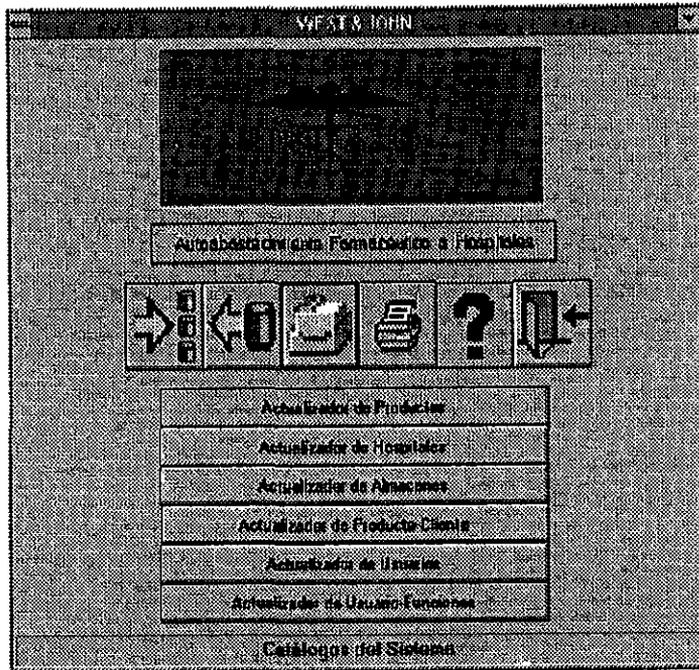


**Nota :** En cualquiera de las pantallas de envío de archivo, se proporcionará la dirección oprimiendo el botón de la derecha, con lo cual se desplegará la pantalla siguiente:



Al realizar un clic en el botón **Aceptar** comenzará el envío del archivo ha transmitir, quedará deshabilitado el botón en tanto esté transmitiendo la información.

### 3. Catálogos del Sistema.



En el módulo de Catálogos ( cuarto botón de izquierda a derecha ), se actualizará la base de datos con los archivos recibidos y de la información capturada en el sistema.

Al oprimir el botón de **Catálogos**, desplegará las siguientes opciones:

- **Actualizador de productos :** Activará la pantalla que actualiza la información de productos.
- **Actualizador de hospitales :** Activará la pantalla que actualiza la información de hospitales.
- **Actualizador de almacenes :** Activará la pantalla que actualiza la información de almacenes.
- **Actualizador de producto-cliente :** Activará la pantalla que actualiza la información de los productos consumidos por cliente.
- **Actualizador de usuarios :** Activará la pantalla que actualiza la información de los usuarios que manejan el sistema AFH.
- **Actualizador de usuario-perfil :** Activará la pantalla que actualiza la información de funciones asignadas a cada usuario.

Botón : **Catálogos del Sistema**  
 Opción : 3.1 **Actualizador de productos**

Permite al usuario realizar movimientos de productos dentro de la empresa. Los movimientos pueden ser : Altas, Bajas, Modificaciones e Impresiones de productos.



- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Nuevo:** Prepara campos para introducir información del producto.
  - **Salvar:** Guarda la información que ha sido capturada : Clave y Descripción del producto.
  - **Borrar:** Borra la información del producto que se encuentre seleccionado (letras rojas) en la tabla que muestra claves y descripciones de cada producto.
  - **Imprimir :** Despliega en pantalla los productos existentes, para imprimir en papel es necesario hacer clic en el icono de la impresora en la misma pantalla.
  - **Salir:** Sale del Catálogo de Productos a la Pantalla Principal.

Botón : **Catálogos del Sistema**  
 Opción : **3.2 Actualizador de Hospitales**

En esta pantalla se podrán hacer movimientos con los hospitales (clientes a los cuales abastece West&John). Los movimientos pueden ser : Altas, Bajas, Modificaciones e Impresiones de hospitales.

| Clave Hospital | Descripción        | Dirección        |
|----------------|--------------------|------------------|
| ABC            | HOSPITAL INGLES    | AV. OBSERVATORIO |
| DALINDE        | HOSPITAL DALINDE   | TUXPAN 4412      |
| DE MEXICO      | HOSPITAL DE MEXICO | VIADUETO         |
| IMSS           | MACDALENA DE LAS   | AV. J.P.N.       |

- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Nuevo:** Prepara campos para agregar un nuevo Hospital.
  - **Registrar:** Guarda la información del movimiento del Hospital realizado por el usuario.
  - **Borrar:** Borra la información del Hospital que se encuentre actualmente *seleccionado* en la tabla inferior (renglón que tenga letras rojas).
  - **Imprimir:** Impresión de los Hospitales presentados en la lista.
  - **Salir:** Sale del Catálogo de Hospitales a la Pantalla Principal.

Botón : **Catálogos del Sistema**  
Opción : 3.3 **Actualizador de Almacenes**

En esta pantalla se permitirá actualizar los almacenes con los que cuenta cada hospital para un mejor abastecimiento.

Permite al usuario realizar movimientos como: Altas, Bajas y Modificaciones de los diferentes almacenes de cada hospital.

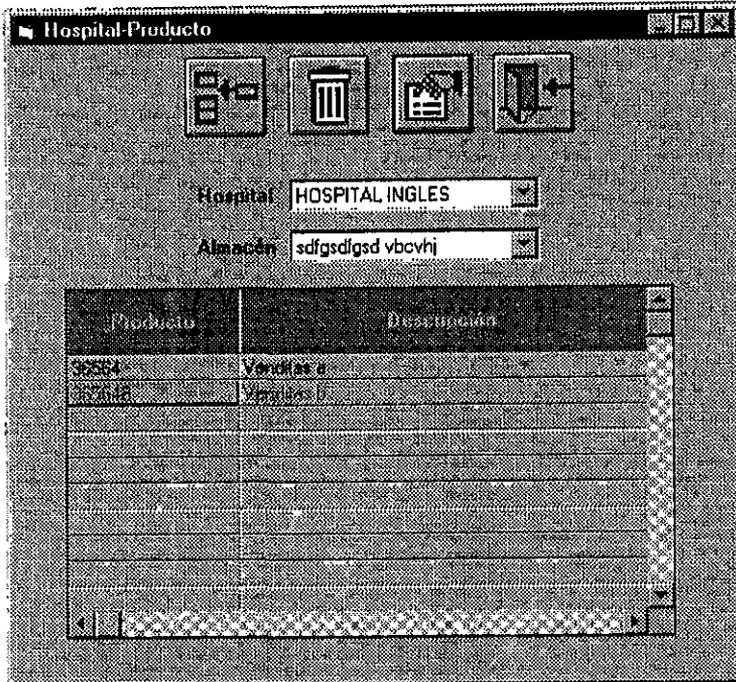
| Clave Almacén | Descripción | Dirección           |
|---------------|-------------|---------------------|
| 12345         | ALMACEN 1   | AV OBSERVATORIO S/N |
| 1234567       | ALMACEN 2   | AV DE LOS PIRULES   |

- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Nuevo:** Prepara los campos para un nuevo Almacén a la base de datos.
  - **Registrar:** Guarda la información del movimiento del Almacén realizado por el usuario.
  - **Borrar:** Borra la información del Almacén que se encuentre actualmente seleccionado.
  - **Imprimir:** Impresión de los Almacenes presentados en la lista de esta pantalla
  - **Salir:** Sale del Catálogo de Almacenes y regresa a la Pantalla Principal.

Botón : **Catálogos del Sistema**  
 Opción : 3.4 **Actualizador de Producto - Cliente**

En esta opción se permitirá hacer movimientos de los productos de la empresa y que internamente maneja el hospital.

Los movimientos que permite realizar son: Altas, Bajas y Modificaciones de los productos a nivel cliente.



- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Insertar producto:** Despliega la ventana para introducir la información requerida del producto.
  - **Borrar:** Borra la información del producto seleccionado en la tabla.
  - **Consulta de Producto:** Muestra la pantalla la información detallada del producto seleccionado.
  - **Salir:** Sale del Catálogo de Productos y regresa a la Pantalla Principal.

Al oprimir el botón de **Insertar Producto** se desplegará la ventana con los campos disponibles para introducir la información del producto.

| Detalle              |                 | Historico          |                              |
|----------------------|-----------------|--------------------|------------------------------|
| Descripción Hospital | Venditas b      | Clave Hospital     | 365648                       |
| Descripción          | Gasas de México | Clave              | Gasas                        |
| Almacén              | sdgdsdgsd vbcvh | Dirección          | vbnkghjkg, igh hikghjkg, hik |
| Existencias          | 848             | Exist. (Máximo)    | 5645                         |
|                      |                 | Exist. (Mínimo)    | 5465                         |
|                      |                 | Desabasto          | 87                           |
| Precio               | 869             | Descuento          | 979                          |
|                      |                 | Estatus Movimiento |                              |
|                      |                 | Pedido             |                              |

- En la parte superior se encuentran los siguientes botones (De Izquierda a Derecha):
  - **Registrar:** Guarda la información del Producto proporcionada por el usuario.
  - **Salir:** Regresa a la pantalla de consulta de los productos.

- Antes de pasar a esta pantalla en la lista de Hospitales de la opción Actualizador de Producto-Cliente se seleccionará el hospital del cual se quieren consultar los productos, éstos serán desplegados en una lista y así seleccionar el que se desee consultar.
- Esta pantalla cuenta con dos pestañas para la consulta del producto, en la primer pestaña la consulta está detallada, **Detalle**, se podrán consultar los almacenes que tienen este producto, así como sus cantidades y precios.
- En la pestaña de **Histórico** se podrá consultar el histórico del producto que está actualmente seleccionado y se desplegará la información del producto por fecha.

The screenshot shows a window titled "Producto" with two tabs: "Detalle" (selected) and "Histórico". The "Detalle" tab contains the following fields:

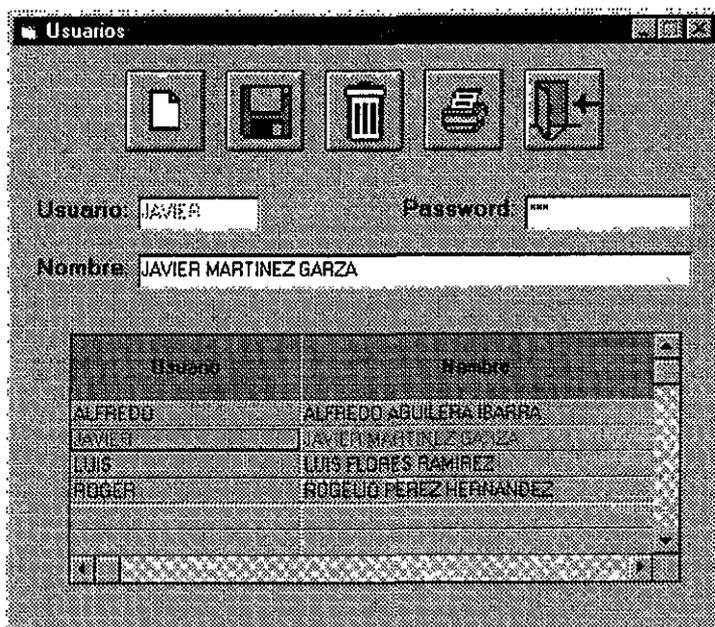
- Descripción Hospital:** Venditas b
- Clave Hospital:** 365648
- Descripción:** Gasas (dropdown menu)
- Clave:** Gasas
- Almacén:** sdfgsdfgsd vbcovhj
- Dirección:** ybnkghjkg,igh hkgkhkghk
- Existencias:** 848
- Exist. (Máximo):** 5645
- Exist. (Mínimo):** 5465
- Desabasto:** 87
- Precio:** 889
- Descuento:** 979
- Estatus Movimiento:** (empty)
- Pedido:** (empty)

- La consulta del producto se puede realizar de dos formas:
  - **Detalle**
  - **Histórico**
- En la parte superior se encuentran el siguiente botón:
  - **Salir:** Regresará a la pantalla de consulta de los productos.

Botón : **Catálogos del Sistema**  
Opción : 3.5 **Actualizador de Usuarios**

Aquí se hará la actualización de los usuarios del sistema, desde dar de alta, baja y cambios del mismo, así como un reporte del catálogo.

Permite al usuario la actualización de los mismos usuarios y su password para acceder al sistema de la Pantalla Password.



| Usuario | Nombre                  |
|---------|-------------------------|
| ALFREDO | ALFREDO ABUILERA IBARRA |
| JAVIER  | JAVIER MARTINEZ GARZA   |
| LUIS    | LUIS FLORES RAMIREZ     |
| ROGER   | ROGER PÉREZ HERNÁNDEZ   |

- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Nuevo:** Prepara los campos para un nuevo Usuario.
  - **Registrar:** Guarda la información del movimiento del Usuario realizado por él mismo.
  - **Borrar:** Borra de la tabla el renglón del Usuario que se encuentre seleccionado (letras rojas).
  - **Imprimir:** Impresión de los Usuarios presentados en la lista.
  - **Salir:** Salida del Catálogo de Usuarios.

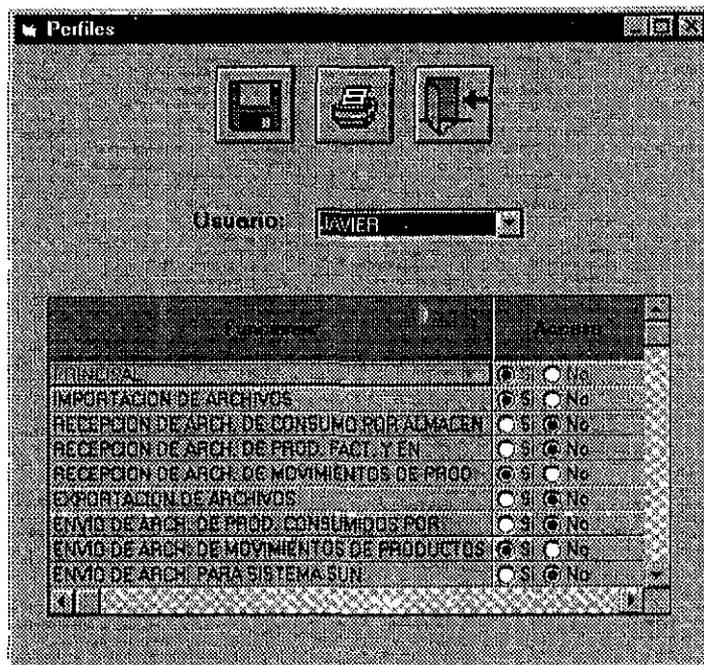
Botón : **Catálogos del Sistema**  
 Opción : 3.6 **Actualizador de Usuario - Perfil**

Actualización del perfil del usuario dentro del sistema.

En esta ventana se van a actualizar los perfiles por cada nivel de seguridad dentro del sistema.

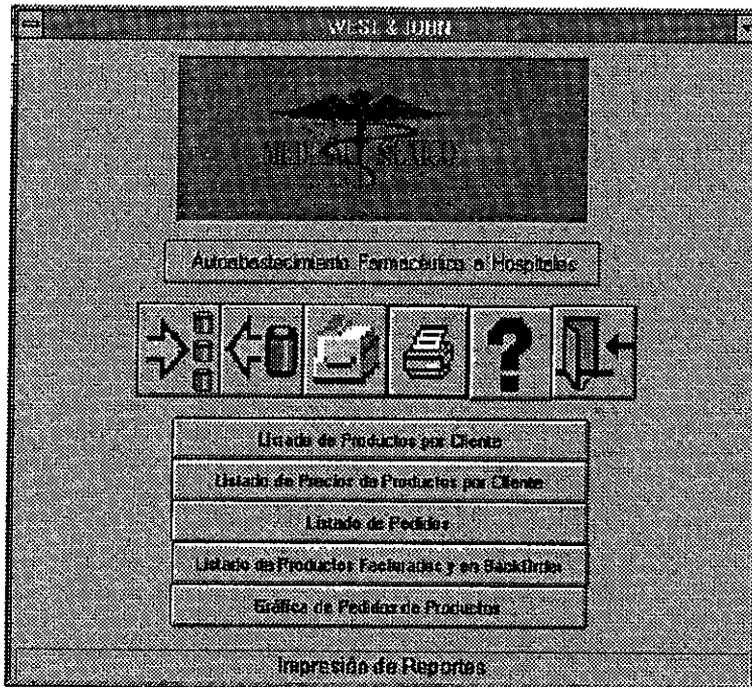
Permite al usuario mover los accesos a las diferentes funciones del sistema de acuerdo a los niveles.

- En la lista de Usuarios se seleccionará el usuario del cual se quieren consultar los perfiles éstos serán desplegados en la lista de abajo, si en la columna acceso tiene marcada la opción **Si** el usuario tendrá acceso a ese nivel.



- En la parte superior se encuentran los siguientes botones: (De izquierda a derecha)
  - **Registrar:** Guarda la información de las funciones asignadas a cada usuario.
  - **Imprimir:** Impresión de los funciones presentados en la lista.
  - **Salir:** Salida del Catálogo de Funciones.

#### 4. Impresión de Reportes



En el módulo de Reportes (quinto botón de izquierda a derecha), se tendrán los siguientes reportes :

Al oprimir el botón de **Impresión de Reportes**, se desplegarán las siguientes opciones:

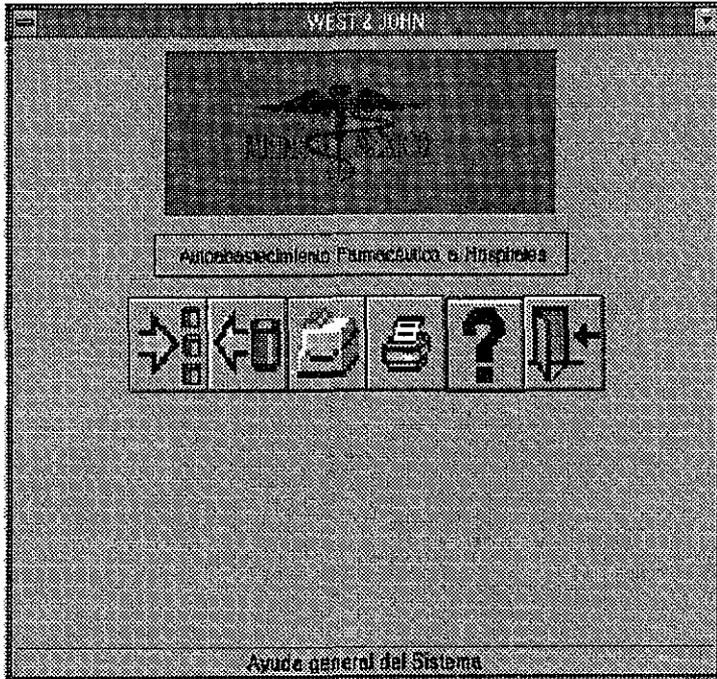
- **4.1 Reporte de Productos por Cliente** : Obtiene el total de los productos por cada hospital y por cada almacén. Nos mostrará el identificador del producto, la clave del hospital; la descripción, precios y descuentos de los productos; la existencia máxima, mínima y el desabasto.
- **4.2 Reporte de Precios de Productos por Cliente** : Por cada hospital y almacén obtienes los precios de cada producto. En este reporte tendremos, identificador del producto, clave del hospital; descripción, precio de la empresa y descuentos del producto; precio al hospital.
- **4.3 Reporte de Pedidos** : Al hacer click en esta opción desplegará una pantalla con rangos de fechas de las cuales se seleccionará un rango para el reporte. El reporte informara por cada hospital y cada almacén los pedidos de acuerdo al rango de fechas seleccionado.

En el reporte se obtiene el número de orden, descripción del producto, cantidad abastecida, cantidad pedida y fecha de factura.

- **4.4 Reporte de Productos Facturados y en BackOrder :** Al oprimir esta opción, desplegará una pantalla con rangos de fechas, de los cuales se seleccionará uno para obtener el reporte de lo que no se pudo surtir al hospital, de este reporte se obtiene la fecha del pedido, descripción del producto, número de orden, cantidad del pedido facturado y, la fecha de factura
- **4.5 Gráfica de Pedidos :** En esta opción se obtiene una Gráfica que muestra la estadística de pedidos por hospital, almacén y producto seleccionado.  
Al oprimir esta opción desplegará una pantalla en la cual se seleccionará el hospital, almacén y el producto, por último oprimir el botón "Graficar".  
Si no se seleccionará alguno de los tres campos se desplegará un mensaje donde indique cual de ellos no se informó. Para regresar a Pantalla Principal, se deberá oprimir el botón "Regresar".

## 5. Ayuda General del Sistema

Muestra todas las pantalla posibles para dar "Ayuda" al usuario con respecto al presente sistema. Esta ventana proporcionará información general del sistema.



Al oprimir el botón de ayuda, desplegará "Ayuda General del Sistema"

- Contenido
- Glosario
- Búsqueda

# **APÉNDICE C**

## **FORMATOS**

# INSTITUTO NACIONAL DE ENFERMEDADES RESPIRATORIAS

DIVISION DE RECURSOS MATERIALES  
LICITACION PUBLICA INTERNACIONAL

CONVOCATORIA. 007

En observancia a la Constitución Política de los Estados Unidos Mexicanos en su artículo 134, y de conformidad con la Ley de Adquisiciones y Obras Públicas, se convoca a los interesados en participar en la(s) licitación(es) de carácter internacional para la contratación de material de curación, de conformidad con lo siguiente:

| No. Licitación  | Costo de las bases                    | Fecha límite para entregar las bases | Fecha de apertura de las bases | Prescrita de proposiciones y ofertas | Acto de apertura de las bases | Plazo de entrega  |
|-----------------|---------------------------------------|--------------------------------------|--------------------------------|--------------------------------------|-------------------------------|---|
| 12223001-010-97 | \$400<br>Costo en compranET:<br>\$900 | 2/02/98                              | 2/02/98<br>12 00 horas         | 9/02/98<br>12 00 horas               | 23/02/98<br>12 00 horas       | Segun calendario que aparece en el anexo 5 de las bases |

| Partida | Clave CABMS | Descripción                                   | Cantidad | Unidad de medida |
|---------|-------------|---|----------|------------------|
| 1       | 0000000000  | Bata para cirujano desechable                 | 3 000    | Pieza            |
| 2       | 0000000000  | Equipo para venoclisis (medidor de volúmenes) | 8 000    | Pieza            |
| 3       | C480000072  | Espuma gasa                                   | 2 250    | Paquete          |
| 4       | 0000000000  | Gautes para exploración                       | 180 000  | Pieza            |
| 5       | 0000000000  | Equipo para venoclisis s/aguja                | 12 000   | Pieza            |

\* Las bases de la licitación se encuentran disponibles para consulta y venta en internet <http://compranet.gob.mx> o bien en Calzada de Tlalpan número 4502, colonia Sección XVI, código postal 14080, Tlalpan, Distrito Federal; con el siguiente horario: 9:00 a 14:00 horas

\* La forma de pago es en concavante, cheque certificado o de caja a nombre del Instituto Nacional de Enfermedades Respiratorias, o en efectivo

En compranET depósito en la cuenta 400 8513 467 de Bitá, abierta a nombre de la Tesorería de la Federación

\* Las bases de la licitación estarán a disposición de los interesados hasta siete días previos al acto de presentación de proposiciones y apertura técnica

\* Los eventos se llevarán a cabo en las instalaciones del Instituto

\* El idioma en que deberán presentarse las proposiciones será español

\* La moneda en que deberán cotizarse las proposiciones será peso mexicano

\* La entrega de muestras será el día 6/02/1998, en el almácn de farmacia de 10:00 a 13:00 horas

\* El lugar de entrega será en almácn de farmacia, los días lunes a viernes en el horario 9:00 a 13:00 horas.

\* Las condiciones de pago serán: 20 días.

| SUBDIRECCIÓN GENERAL DE PROTECCIÓN AL SALARIO |                         | ISSSTE                  |          | PEDIDO ABIERTO: 1543/18/01/016    |                           | PEDIDO                        |  |
|---|-------------------------|-------------------------|----------|-----------------------------------|---------------------------|-------------------------------|--|
| SUBDIRECCIÓN DE ABASTO                        |                         | SUBDIRECCIÓN DE ABASTO  |          | FECHA DE AUTORIZACIÓN: 20/10/1997 |                           | FECHA DE EMISIÓN: 01/01/1997  |  |
| PROVEEDOR: CONOCIDO.                          |                         | R.F.C.: XXXX-000000-XXX |          |                                   |                           |                               |  |
| DOMICILIO: CONOCIDO.                          |                         | ESTADO: MEXICO          |          |                                   |                           |                               |  |
| CIUDAD: MEXICO                                |                         |                         |          |                                   |                           |                               |  |
| C.P. 0000                                     |                         | TELS.: 000-00-00        |          |                                   |                           |                               |  |
| LINEA: 18                                     |                         | FARMACÉUTICA            |          |                                   |                           |                               |  |
| PARTIDA                                       | DESCRIPCIÓN             | MEDIDA                  | CANTIDAD | PRECIO COSTO                      | PRECIO VTA. METROPOLITANO | PRECIO VTA. UNITARIO FORANEEO |  |
| CÓDIGO  |                         |                         |          |                                   |                           |                               |  |
| 01  | Productos Farmacéuticos |                         |          |                                   |                           |                               |  |
| 10 001  | Producto 1              | CAJA                    | ##       | \$                                | \$                        | \$                            |  |
| 10 002  | Producto 2              | CAJA                    | ##       | \$                                | \$                        | \$                            |  |
| 10 003  | Producto 3              | CAJA                    | ##       | \$                                | \$                        | \$                            |  |
| 10 004  | Producto 4              | CAJA                    | ##       | \$                                | \$                        | \$                            |  |
| 10 005  | Producto 5              | BOLSA                   | ##       | \$                                | \$                        | \$                            |  |
| 10 006  | Producto 6              | CAJA                    | ##       | \$                                | \$                        | \$                            |  |
|   |                         | Firma 2                 |          | Firma 3                           |                           |                               |  |
|   |                         | Firma 1                 |          |                                   |                           |                               |  |

# INSTITUTO NACIONAL DE CARDIOLOGÍA

DIVISION DE RECURSOS MATERIALES

LICITACION PUBLICA INTERNACIONAL

CONVOCATORIA: 1297

En observancia a la Constitución Política de los Estados Unidos Mexicanos en su artículo 134, y de conformidad con la ley de Adquisiciones y Obras Públicas, se convoca a los interesados en participar en la(s) licitación(es) de carácter internacional para la contratación de material de curación, de conformidad con lo siguiente:

| No. de lotes       | Código de las bases                         | Fecha de publicación para adquirir bases | Fecha de presentación de propuestas | Fecha de apertura de propuestas | Fecha de entrega   |
|--------------------|---|--|-------------------------------------|---------------------------------|--|
| 82385647-010-56-97 | \$50 000<br>Costo en compraNET:<br>\$45 000 | 1/11/97                                  | 1/11/97<br>19:00 horas              | 15/11/97<br>15:00 horas         | 23/02/98<br>12:00 horas<br>Segun calendario que aparece en el anexo 5 de las bases |

| Partida | Clave CABMS | Descripción                  | Cantidad | Unidad de medida |
|---------|-------------|------------------------------|----------|------------------|
| 1       | 0000000123  | Jeringa desechable           | 50 000   | Pieza            |
| 2       | 0000000450  | Venda elastica 5cm x 5mts    | 13 000   | Pieza            |
| 3       | 000000072F  | Llave de 3 vias k-72         | 250      | Pieza            |
| 4       | 0000000023  | Sonda estom. de Levin No. 12 | 1 800    | Pieza            |
| 5       | FD00000000  | Hoja para bisturi            | 15 000   | Pieza            |

\* Las bases de la licitación se encuentran disponibles para consulta y venta en Internet: <http://compra.net.gob.mx> o bien en Calzada de Tlalpan número 4502, colonia Sección XVI, código postal 14080, Tlalpan, Distrito Federal, con el siguiente horario: 9:00 a 14:00 horas.

\* La forma de pago es, en comprobante, cheque certificado o de caja a nombre del Instituto Nacional de Cardiología, o en efectivo.

En compraNET: depósito en la cuenta 568 0800 652 de Bital, abierta a nombre de la Tesorería de la Federación.

\* Los eventos de la licitación estarán a disposición de los interesados hasta siete días previos al acto de presentación de proposiciones y apertura técnica.

\* Los eventos se llevarán a cabo en las instalaciones del Instituto

\* El idioma en que deberán presentarse las proposiciones será: español.

\* La moneda en que deberán cotizarse las proposiciones será: peso mexicano

\* La entrega de muestras será el día 6/12/1997, en el almacén de farmacia de 10.00 a 13.00 horas

\* El lugar de entrega será en: almacén de farmacia, los días lunes a viernes en el horario: 9:00 a 13:00 horas

\* Las condiciones de pago serán: 20 días.

| ISSSTE                 |                             | SUBDIRECCIÓN GENERAL DE PROTECCIÓN AL SALARIO |          | PEDIDO ABIERTO: 235/DF/345/98      |                                    |                               |
|------------------------|-----------------------------|---|----------|------------------------------------|------------------------------------|-------------------------------|
| SUBDIRECCIÓN DE ABASTO |                             | SUBDIRECCIÓN DE ABASTO                        |          | FECHA DE AUTORIZACIÓN: 15/06/1997  |                                    |                               |
| PROVEEDOR: MEDICAL     |                             | R.F.C.: MEDXX-451009-F41                      |          | FECHA DE EMISIÓN: 19/06/1997       |                                    |                               |
| DOMICILIO: CONOCIDO.   |                             | ESTADO: MEXICO                                |          | PRECIO VTA. UNITARIO METROPOLITANO |                                    |                               |
| CIUDAD: MEXICO D.F.    |                             | ESTADO: MEXICO                                |          | PRECIO VTA. UNITARIO FORANEJO      |                                    |                               |
| C.P. 0000              |                             | TELS. . 000-00-00                             |          | PRECIO COSTO                       |                                    |                               |
| LINEA 18               |                             | FARMACÉUTICA                                  |          | CANTIDAD                           |                                    |                               |
| PARTIDA                | DESCRIPCIÓN                 | MEDIDA  | CANTIDAD | PRECIO COSTO                       | PRECIO VTA. UNITARIO METROPOLITANO | PRECIO VTA. UNITARIO FORANEJO |
| CÓDIGO                 |                             |   |          |                                    |                                    |                               |
| 01                     | Productos Farmacéuticos     |   |          |                                    |                                    |                               |
| 70 514                 | Mantol al 20%               | CAJA  | 500      | \$120 000.00                       | \$240.00                           | \$245.00                      |
| 70 767                 | Sulfato de protamina        | CAJA  | 250      | \$78 750.00                        | \$315.00                           | \$325.00                      |
| 70 376                 | Insulina r simple           | FRASCO  | 1200     | \$252 000.00                       | \$210.00                           | \$205.00                      |
| 70 611                 | Tiopental sódico            | CAJA  | 542      | \$170 730.00                       | \$315.00                           | \$316.00                      |
| 70 766                 | Sulfato de magnesio al 100% | BOLSA   | 609      | \$60 960.90                        | \$100.10                           | \$100.10                      |
| 70 847                 | Diazepam                    | CAJA  | 1400     | \$245 000.00                       | \$175.00                           | \$152.00                      |
| 70 706                 | Solu-medrol                 | CAJA  | 253      | \$28 336.00                        | \$112.00                           | \$110.00                      |
| 70 705                 | Sol. Hartman                | FRASCO  | 254      | \$37 338.00                        | \$147.00                           | \$157.00                      |
| 70 697                 | Sol. Dextrosa 10%           | FRASCO  | 154      | \$60 050.00                        | \$325.00                           | \$450.00                      |
|                        |                             |   | Firma 2  |                                    |                                    | Firma 3                       |

**APÉNDICE D**  
**DICCIONARIO DE DATOS**

Tabla: t\_almacen

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 12        | Última actualización: | 04/04/98 |

**Columnas**

| Nombre                  | Tipo              | Tamaño |
|-------------------------|-------------------|--------|
| id_alma_almacen         | Texto             | 10     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | id_alma_almacen   |        |
| Column Order:           | Falso             |        |
| Column Width:           | 1                 |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 1                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_almacen         |        |
| id_cliente              | Texto             | 10     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | id_cliente        |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 2                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_almacen         |        |
| alma_descripcion        | Texto             | 50     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | alma_descripcion  |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 3                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_almacen         |        |
| alma_direccion          | Texto             | 50     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | alma_direccion    |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
|                         | Predeterminado    |        |

Tabla: t\_almacen

|                    |                         |                    |                   |
|--------------------|-------------------------|--------------------|-------------------|
|                    | Orden de combinación:   | Español            |                   |
|                    | Permitir longitud cero: | Falso              |                   |
|                    | Posición ordinal:       | 4                  |                   |
|                    | Requerido:              | Falso              |                   |
|                    | Tabla de origen:        | t_almacen          |                   |
| alma_fecha         |                         |                    | Fecha/Hora 8      |
|                    | Atributos:              | Tamaño fijo        |                   |
|                    | Campo de origen:        | alma_fecha         |                   |
|                    | Column Hidden:          | Falso              |                   |
|                    | Column Order:           | Predeterminado     |                   |
|                    | Column Width:           | Predeterminado     |                   |
|                    | Orden de combinación:   | Español            |                   |
|                    | Permitir longitud cero: | Falso              |                   |
|                    | Posición ordinal:       | 5                  |                   |
|                    | Requerido:              | Falso              |                   |
|                    | Tabla de origen:        | t_almacen          |                   |
| alma_usuario       |                         |                    | Texto 10          |
|                    | Atributos:              | Longitud variable  |                   |
|                    | Campo de origen:        | alma_usuario       |                   |
|                    | Column Hidden:          | Falso              |                   |
|                    | Column Order:           | Predeterminado     |                   |
|                    | Column Width:           | Predeterminado     |                   |
|                    | Orden de combinación:   | Español            |                   |
|                    | Permitir longitud cero: | Falso              |                   |
|                    | Posición ordinal:       | 6                  |                   |
|                    | Requerido:              | Falso              |                   |
|                    | Tabla de origen:        | t_almacen          |                   |
| alma_estatus_envio |                         |                    | Número (simple) 4 |
|                    | Atributos:              | Tamaño fijo        |                   |
|                    | Campo de origen:        | alma_estatus_envio |                   |
|                    | Column Hidden:          | Falso              |                   |
|                    | Column Order:           | Predeterminado     |                   |
|                    | Column Width:           | Predeterminado     |                   |
|                    | Lugares decimales:      | Ascendente         |                   |
|                    | Orden de combinación:   | Español            |                   |
|                    | Permitir longitud cero: | Falso              |                   |
|                    | Posición ordinal:       | 7                  |                   |
|                    | Requerido:              | Falso              |                   |
|                    | Tabla de origen:        | t_almacen          |                   |
|                    | Valor predeterminado:   | 0                  |                   |

Indices de tablas

| <u>Nombre</u>    | <u>Número de campos</u> |
|------------------|-------------------------|
| PrimaryKey       | 1                       |
| Agrupado:        | Falso                   |
| Cuenta distinta: | 12                      |
| Extranjero:      | Falso                   |
| Ignorar Nulos:   | Falso                   |

---

Tabla: t\_almacen

---

|            |                             |
|------------|-----------------------------|
| Nombre:    | PrimaryKey                  |
| Principal: | Verdadero                   |
| Requerido: | Verdadero                   |
| Única:     | Verdadero                   |
| Campos:    | id_alma_almacen, Ascendente |

#### Permisos de usuario

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

#### Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

---

 Tabla: t\_archivo
 

---

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 0         | Última actualización: | 04/04/98 |

**Columnas**

| Nombre                                | Tipo           | Tamaño |
|---------------------------------------|----------------|--------|
| id_tiar_tipo_archivo                  | Número (largo) | 4      |
| Atributos:                            |                |        |
| Campo de origen: id_tiar_tipo_archivo |                |        |
| Column Hidden: Falso                  |                |        |
| Column Order: 1                       |                |        |
| Column Width: Predeterminado          |                |        |
| Lugares decimales: 255                |                |        |
| Orden de combinación: General         |                |        |
| Permitir longitud cero: Falso         |                |        |
| Posición ordinal: 1                   |                |        |
| Requerido: Falso                      |                |        |
| Tabla de origen: t_archivo            |                |        |
| Valor predeterminado: 0               |                |        |
| id_campo                              | Número (largo) | 4      |
| Atributos:                            |                |        |
| Campo de origen: id_campo             |                |        |
| Column Hidden: Falso                  |                |        |
| Column Order: Predeterminado          |                |        |
| Column Width: Predeterminado          |                |        |
| Lugares decimales: 255                |                |        |
| Orden de combinación: General         |                |        |
| Permitir longitud cero: Falso         |                |        |
| Posición ordinal: 2                   |                |        |
| Requerido: Falso                      |                |        |
| Tabla de origen: t_archivo            |                |        |
| Valor predeterminado: 0               |                |        |
| arch_posc_inicial                     | Número (doble) | 8      |
| Atributos:                            |                |        |
| Campo de origen: arch_posc_inicial    |                |        |
| Column Hidden: Falso                  |                |        |
| Column Order: Predeterminado          |                |        |
| Column Width: Predeterminado          |                |        |
| Lugares decimales: 255                |                |        |
| Orden de combinación: General         |                |        |
| Permitir longitud cero: Falso         |                |        |
| Posición ordinal: 3                   |                |        |
| Requerido: Falso                      |                |        |
| Tabla de origen: t_archivo            |                |        |
| Valor predeterminado: 0               |                |        |

Tabla: t\_archivo

|                         |                   |                |    |
|-------------------------|-------------------|----------------|----|
| arch_posc_final         |                   | Número (doble) | 8  |
| Atributos:              | Tamaño fijo       |                |    |
| Campo de origen:        | arch_posc_final   |                |    |
| Column Hidden:          | Falso             |                |    |
| Column Order:           | Predeterminado    |                |    |
| Column Width:           | Predeterminado    |                |    |
| Lugares decimales:      | 255               |                |    |
| Orden de combinación:   | General           |                |    |
| Permitir longitud cero: | Falso             |                |    |
| Posición ordinal:       | 4                 |                |    |
| Requerido:              | Falso             |                |    |
| Tabla de origen:        | t_archivo         |                |    |
| Valor predeterminado:   | 0                 |                |    |
| arch_descripcion        |                   | Texto          | 50 |
| Atributos:              | Longitud variable |                |    |
| Campo de origen:        | arch_descripcion  |                |    |
| Column Hidden:          | Falso             |                |    |
| Column Order:           | Predeterminado    |                |    |
| Column Width:           | Predeterminado    |                |    |
| Orden de combinación:   | Español           |                |    |
| Permitir longitud cero: | Verdadero         |                |    |
| Posición ordinal:       | 5                 |                |    |
| Requerido:              | Falso             |                |    |
| Tabla de origen:        | t_archivo         |                |    |
| arch_fecha              |                   | Fecha/Hora     | 8  |
| Atributos:              | Tamaño fijo       |                |    |
| Campo de origen:        | arch_fecha        |                |    |
| Column Hidden:          | Falso             |                |    |
| Column Order:           | Predeterminado    |                |    |
| Column Width:           | Predeterminado    |                |    |
| Orden de combinación:   | Español           |                |    |
| Permitir longitud cero: | Falso             |                |    |
| Posición ordinal:       | 6                 |                |    |
| Requerido:              | Falso             |                |    |
| Tabla de origen:        | t_archivo         |                |    |
| arch_usuario            |                   | Texto          | 10 |
| Atributos:              | Longitud variable |                |    |
| Campo de origen:        | arch_usuario      |                |    |
| Column Hidden:          | Falso             |                |    |
| Column Order:           | Predeterminado    |                |    |
| Column Width:           | Predeterminado    |                |    |
| Orden de combinación:   | Español           |                |    |
| Permitir longitud cero: | Falso             |                |    |
| Posición ordinal:       | 7                 |                |    |
| Requerido:              | Falso             |                |    |
| Tabla de origen:        | t_archivo         |                |    |

---

 Tabla: t\_archivo
 

---

Índices de tablas

| <u>Nombre</u>    | <u>Número de campos</u>          |
|------------------|----------------------------------|
| PrimaryKey       | 2                                |
| Agrupado:        | Falso                            |
| Cuenta distinta: | 0                                |
| Extranjero:      | Falso                            |
| Ignorar Nulos:   | Falso                            |
| Nombre:          | PrimaryKey                       |
| Principal:       | Verdadero                        |
| Requerido:       | Verdadero                        |
| Única:           | Verdadero                        |
| Campos:          | id_tiar_tipo_archivo, Ascendente |
| Campos:          | id_campo, Ascendente             |

Permisos de usuario

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

Permisos de grupo

|        |   |
|--------|---|
| Admins | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
| Users  |   |

Tabla: t\_campo

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 0         | Última actualización: | 04/04/98 |

**Columnas**

| Nombre                  | Tipo              | Tamaño |
|-------------------------|-------------------|--------|
| id_campo                | Número (simple)   | 4      |
| Atributos:              |                   |        |
| Campo de origen:        | Tamaño fijo       |        |
| Column Hidden:          | id_campo          |        |
| Column Order:           | Falso             |        |
| Column Width:           | 1                 |        |
| Lugares decimales:      | Predeterminado    |        |
| Orden de combinación:   | 255               |        |
| Permitir longitud cero: | General           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 1                 |        |
| Tabla de origen:        | Falso             |        |
| Valor predeterminado:   | t_campo           |        |
| 0                       |                   |        |
| camp_descripcion        | Texto             | 50     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | camp_descripcion  |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 2                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_campo           |        |

**Índices de tablas**

| Nombre           | Número de campos     |
|------------------|----------------------|
| PrimaryKey       | 1                    |
| Agrupado:        | Falso                |
| Cuenta distinta: | 0                    |
| Extranjero:      | Falso                |
| Ignorar Nulos:   | Falso                |
| Nombre:          | PrimaryKey           |
| Principal:       | Verdadero            |
| Requerido:       | Verdadero            |
| Única:           | Verdadero            |
| Campos:          | id_campo, Ascendente |

---

Tabla: t\_campo

---

**Permisos de usuario**

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

**Permisos de grupo**

Admins

Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_campos\_tabla

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 55        | Última actualización: | 04/04/98 |

**Columnas**

| Nombre                  | Tipo   | Tamaño |
|-------------------------|--|--------|
| id_tab_tabla            | Texto  | 8      |
| Atributos:              |  |        |
| Campo de origen:        | Longitud variable  |        |
| Column Hidden:          | id_tab_tabla   |        |
| Column Order:           | Falso  |        |
| Column Width:           | 1  |        |
| Descripción:            | Predeterminado   |        |
| Orden de combinación:   | Clave del archivo o tabla dentro de la base de datos         |        |
| Permitir longitud cero: | Español  |        |
| Posición ordinal:       | Falso  |        |
| Requerido:              | 1  |        |
| Tabla de origen:        | Falso  |        |
|                         | t_campos_tabla   |        |
| id_cata_numcampo        | Número (simple)  | 4      |
| Atributos:              |  |        |
| Campo de origen:        | Tamaño fijo  |        |
| Column Hidden:          | id_cata_numcampo   |        |
| Column Order:           | Falso  |        |
| Column Width:           | Predeterminado   |        |
| Descripción:            | Predeterminado   |        |
| Lugares decimales:      | Numero de campo y orden dentro del registro                  |        |
| Orden de combinación:   | 255  |        |
| Permitir longitud cero: | Español  |        |
| Posición ordinal:       | Falso  |        |
| Requerido:              | 2  |        |
| Tabla de origen:        | Falso  |        |
| Valor predeterminado:   | t_campos_tabla   |        |
|                         | 0  |        |
| cata_descripcion        | Texto  | 30     |
| Atributos:              |  |        |
| Campo de origen:        | Longitud variable  |        |
| Column Hidden:          | cata_descripcion   |        |
| Column Order:           | Falso  |        |
| Column Width:           | Predeterminado   |        |
| Descripción:            | 1785   |        |
| Orden de combinación:   | Nombre del campo dentro de la definición de la base de datos |        |
| Permitir longitud cero: | Español  |        |
| Posición ordinal:       | Falso  |        |
| Requerido:              | 3  |        |
| Tabla de origen:        | Falso  |        |
|                         | t_campos_tabla   |        |
| id_form_formato         | Número (doble)   | 8      |

Tabla: t\_campos\_tabla

|                         |  |                        |          |
|-------------------------|--|------------------------|----------|
| Atributos:              | Tamaño fijo  |                        |          |
| Campo de origen:        | id_form_formato  |                        |          |
| Column Hidden:          | Falso  |                        |          |
| Column Order:           | Predeterminado   |                        |          |
| Column Width:           | Predeterminado   |                        |          |
| Descripción:            | Formato y longitud del campo                                 |                        |          |
| Lugares decimales:      | 255  |                        |          |
| Orden de combinación:   | Español  |                        |          |
| Permitir longitud cero: | Falso  |                        |          |
| Posición ordinal:       | 4  |                        |          |
| Requerido:              | Falso  |                        |          |
| Tabla de origen:        | t_campos_tabla   |                        |          |
| Valor predeterminado:   | 0  |                        |          |
| <b>cata_llave</b>       |  | <b>Número (simple)</b> | <b>4</b> |
| Atributos:              | Tamaño fijo  |                        |          |
| Campo de origen:        | cata_llave   |                        |          |
| Column Hidden:          | Falso  |                        |          |
| Column Order:           | Predeterminado   |                        |          |
| Column Width:           | Predeterminado   |                        |          |
| Descripción:            | Tipo de campo 1 = si es llave 0 = si es atributo de la tabla |                        |          |
| Lugares decimales:      | 255  |                        |          |
| Orden de combinación:   | Español  |                        |          |
| Permitir longitud cero: | Falso  |                        |          |
| Posición ordinal:       | 5  |                        |          |
| Requerido:              | Falso  |                        |          |
| Tabla de origen:        | t_campos_tabla   |                        |          |
| Valor predeterminado:   | 0  |                        |          |

Índices de tablas

| <u>Nombre</u>     | <u>Número de campos</u>      |
|-------------------|------------------------------|
| <b>PrimaryKey</b> | <b>2</b>                     |
| Agrupado:         | Falso                        |
| Cuenta distinta:  | 55                           |
| Extranjero:       | Falso                        |
| Ignorar Nulos:    | Falso                        |
| Nombre:           | PrimaryKey                   |
| Principal:        | Verdadero                    |
| Requerido:        | Verdadero                    |
| Única:            | Verdadero                    |
| Campos:           | Id_tab_tabla, Ascendente     |
| Campos:           | Id_cata_numcampo, Ascendente |

Permisos de usuario

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

---

Tabla: t\_campos\_tabla

---

Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_cargas

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 0         | Última actualización: | 04/04/98 |

Columnas

| Nombre   | Tipo           | Tamaño |
|--|----------------|--------|
| id_archivo   | Texto          | 10     |
| Atributos: Longitud variable<br>Campo de origen: id_archivo<br>Column Hidden: Falso<br>Column Order: 1<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_cargas                       |                |        |
| carg_fecha   | Fecha/Hora     | 8      |
| Atributos: Tamaño fijo<br>Campo de origen: carg_fecha<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_cargas                |                |        |
| carg_nombre_archivo  | Texto          | 30     |
| Atributos: Longitud variable<br>Campo de origen: carg_nombre_archivo<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 3<br>Requerido: Falso<br>Tabla de origen: t_cargas |                |        |
| carg_consecutivo   | Número (largo) | 4      |
| Atributos: Tamaño fijo, AutoIncremento<br>Campo de origen: carg_consecutivo<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado  |                |        |

---

 Tabla: t\_cargas
 

---

|                         |          |
|-------------------------|----------|
| Orden de combinación:   | Español  |
| Permitir longitud cero: | Falso    |
| Posición ordinal:       | 4        |
| Requerido:              | Falso    |
| Tabla de origen:        | t_cargas |

#### Índices de tablas

| <u>Nombre</u>    | <u>Número de campos</u>      |
|------------------|------------------------------|
| PrimaryKey       | 3                            |
| Agrupado:        | Falso                        |
| Cuenta distinta: | 0                            |
| Extranjero:      | Falso                        |
| Ignorar Nulos:   | Falso                        |
| Nombre:          | PrimaryKey                   |
| Principal:       | Verdadero                    |
| Requerido:       | Verdadero                    |
| Única:           | Verdadero                    |
| Campos:          | id_archivo, Ascendente       |
| Campos:          | carg_fecha, Ascendente       |
| Campos:          | carg_consecutivo, Ascendente |

#### Permisos de usuario

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

#### Permisos de grupo

|        |   |
|--------|---|
| Admins |   |
| Users  | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |

Tabla: t\_cliente

**Propiedades**

Def. actualizable: Verdadero  
Record Count: 0

Fecha de creación: 28/02/98  
Última actualización: 04/04/98

**Columnas**

| Nombre  | Tipo       | Tamaño |
|---|------------|--------|
| id_cliente  | Texto      | 10     |
| Atributos:<br>Campo de origen: id_cliente<br>Column Hidden: Falso<br>Column Order: 1<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_cliente                       |            |        |
| cliente_descripcion   | Texto      | 50     |
| Atributos:<br>Campo de origen: cliente_descripcion<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_cliente |            |        |
| cliente_direccion   | Texto      | 50     |
| Atributos:<br>Campo de origen: cliente_direccion<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 3<br>Requerido: Falso<br>Tabla de origen: t_cliente   |            |        |
| cliente_fecha   | Fecha/Hora | 8      |
| Atributos:<br>Campo de origen: cliente_fecha<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado  |            |        |

Tabla: t\_cliente

|                   |                         |                   |                 |    |
|-------------------|-------------------------|-------------------|-----------------|----|
|                   | Orden de combinación:   | Español           |                 |    |
|                   | Permitir longitud cero: | Falso             |                 |    |
|                   | Posición ordinal:       | 4                 |                 |    |
|                   | Requerido:              | Falso             |                 |    |
|                   | Tabla de origen:        | t_cliente         |                 |    |
| cli_usuario       |                         |                   | Texto           | 10 |
|                   | Atributos:              | Longitud variable |                 |    |
|                   | Campo de origen:        | cli_usuario       |                 |    |
|                   | Column Hidden:          | Falso             |                 |    |
|                   | Column Order:           | Predeterminado    |                 |    |
|                   | Column Width:           | Predeterminado    |                 |    |
|                   | Orden de combinación:   | Español           |                 |    |
|                   | Permitir longitud cero: | Falso             |                 |    |
|                   | Posición ordinal:       | 5                 |                 |    |
|                   | Requerido:              | Falso             |                 |    |
|                   | Tabla de origen:        | t_cliente         |                 |    |
| cli_telefono      |                         |                   | Texto           | 20 |
|                   | Atributos:              | Longitud variable |                 |    |
|                   | Campo de origen:        | cli_telefono      |                 |    |
|                   | Column Hidden:          | Falso             |                 |    |
|                   | Column Order:           | Predeterminado    |                 |    |
|                   | Column Width:           | Predeterminado    |                 |    |
|                   | Orden de combinación:   | Español           |                 |    |
|                   | Permitir longitud cero: | Falso             |                 |    |
|                   | Posición ordinal:       | 6                 |                 |    |
|                   | Requerido:              | Falso             |                 |    |
|                   | Tabla de origen:        | t_cliente         |                 |    |
| cli_estatus_envio |                         |                   | Número (simple) | 4  |
|                   | Atributos:              | Tamaño fijo       |                 |    |
|                   | Campo de origen:        | cli_estatus_envio |                 |    |
|                   | Column Hidden:          | Falso             |                 |    |
|                   | Column Order:           | Predeterminado    |                 |    |
|                   | Column Width:           | Predeterminado    |                 |    |
|                   | Lugares decimales:      | Ascendente        |                 |    |
|                   | Orden de combinación:   | Español           |                 |    |
|                   | Permitir longitud cero: | Falso             |                 |    |
|                   | Posición ordinal:       | 7                 |                 |    |
|                   | Requerido:              | Falso             |                 |    |
|                   | Tabla de origen:        | t_cliente         |                 |    |
|                   | Valor predeterminado:   | 0                 |                 |    |

Índices de tablas

| <u>Nombre</u>    | <u>Número de campos</u> |
|------------------|-------------------------|
| PrimaryKey       | 1                       |
| Agrupado:        | Falso                   |
| Cuenta distinta: | 0                       |
| Extranjero:      | Falso                   |
| Ignorar Nulos:   | Falso                   |

---

Tabla: t\_cliente

---

|            |                            |
|------------|----------------------------|
| Nombre:    | PrimaryKey                 |
| Principal: | Verdadero                  |
| Requerido: | Verdadero                  |
| Única:     | Verdadero                  |
| Campos:    | id_cle_cliente, Ascendente |

Permisos de usuario

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_cliente\_pedido

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 5         | Última actualización: | 04/04/98 |

**Columnas**

| Nombre   | Tipo  | Tamaño |
|--|---|--------|
| <b>clpp_fecha</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:       | Fecha/Hora<br><br>Tamaño fijo<br>clpp_fecha<br>Falso<br>1<br>Predeterminado<br>General<br>Falso<br>1<br>Falso<br>t_cliente_pedido                     | 8      |
| <b>id_clie_cliente</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:  | Texto<br><br>Longitud variable<br>id_clie_cliente<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>2<br>Falso<br>t_cliente_pedido  | 10     |
| <b>id_prod_producto</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen: | Texto<br><br>Longitud variable<br>id_prod_producto<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>3<br>Falso<br>t_cliente_pedido | 10     |
| <b>id_alma_almacen</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:   | Texto<br><br>Longitud variable<br>id_alma_almacen<br>Falso<br>Predeterminado<br>Predeterminado  | 10     |

Tabla: t\_cliente\_pedido

|                         |                      |                 |    |
|-------------------------|----------------------|-----------------|----|
| Orden de combinación:   | Español              |                 |    |
| Permitir longitud cero: | Falso                |                 |    |
| Posición ordinal:       | 4                    |                 |    |
| Requerido:              | Falso                |                 |    |
| Tabla de origen:        | t_cliente_pedido     |                 |    |
| clpp_ref_cliente        |                      | Texto           | 10 |
| Atributos:              | Longitud variable    |                 |    |
| Campo de origen:        | clpp_ref_cliente     |                 |    |
| Column Hidden:          | Falso                |                 |    |
| Column Order:           | Predeterminado       |                 |    |
| Column Width:           | Predeterminado       |                 |    |
| Orden de combinación:   | Español              |                 |    |
| Permitir longitud cero: | Falso                |                 |    |
| Posición ordinal:       | 5                    |                 |    |
| Requerido:              | Falso                |                 |    |
| Tabla de origen:        | t_cliente_pedido     |                 |    |
| clpp_norden             |                      | Texto           | 10 |
| Atributos:              | Longitud variable    |                 |    |
| Campo de origen:        | clpp_norden          |                 |    |
| Column Hidden:          | Falso                |                 |    |
| Column Order:           | Predeterminado       |                 |    |
| Column Width:           | Predeterminado       |                 |    |
| Orden de combinación:   | Español              |                 |    |
| Permitir longitud cero: | Falso                |                 |    |
| Posición ordinal:       | 6                    |                 |    |
| Requerido:              | Falso                |                 |    |
| Tabla de origen:        | t_cliente_pedido     |                 |    |
| clpp_cantidad_pedido    |                      | Número (doble)  | 8  |
| Atributos:              | Tamaño fijo          |                 |    |
| Campo de origen:        | clpp_cantidad_pedido |                 |    |
| Column Hidden:          | Falso                |                 |    |
| Column Order:           | Predeterminado       |                 |    |
| Column Width:           | Predeterminado       |                 |    |
| Lugares decimales:      | 255                  |                 |    |
| Orden de combinación:   | Español              |                 |    |
| Permitir longitud cero: | Falso                |                 |    |
| Posición ordinal:       | 7                    |                 |    |
| Requerido:              | Falso                |                 |    |
| Tabla de origen:        | t_cliente_pedido     |                 |    |
| Valor predeterminado:   | 0                    |                 |    |
| clpp_estatus_pedido     |                      | Número (simple) | 4  |
| Atributos:              | Tamaño fijo          |                 |    |
| Campo de origen:        | clpp_estatus_pedido  |                 |    |
| Column Hidden:          | Falso                |                 |    |
| Column Order:           | Predeterminado       |                 |    |
| Column Width:           | Predeterminado       |                 |    |
| Lugares decimales:      | 255                  |                 |    |
| Orden de combinación:   | Español              |                 |    |
| Permitir longitud cero: | Falso                |                 |    |

Tabla: t\_cliente\_pedido

|                         |                       |                 |    |
|-------------------------|-----------------------|-----------------|----|
| Posición ordinal:       | 8                     |                 |    |
| Requerido:              | Falso                 |                 |    |
| Tabla de origen:        | t_cliente_pedido      |                 |    |
| Valor predeterminado:   | 0                     |                 |    |
| clpp_estatus_producto   |                       | Número (simple) | 4  |
| Atributos:              | Tamaño fijo           |                 |    |
| Campo de origen:        | clpp_estatus_producto |                 |    |
| Column Hidden:          | Falso                 |                 |    |
| Column Order:           | Predeterminado        |                 |    |
| Column Width:           | Predeterminado        |                 |    |
| Lugares decimales:      | 255                   |                 |    |
| Orden de combinación:   | Español               |                 |    |
| Permitir longitud cero: | Falso                 |                 |    |
| Posición ordinal:       | 9                     |                 |    |
| Requerido:              | Falso                 |                 |    |
| Tabla de origen:        | t_cliente_pedido      |                 |    |
| Valor predeterminado:   | 0                     |                 |    |
| clpp_fecha_factura      |                       | Fecha/Hora      | 8  |
| Atributos:              | Tamaño fijo           |                 |    |
| Campo de origen:        | clpp_fecha_factura    |                 |    |
| Column Hidden:          | Falso                 |                 |    |
| Column Order:           | Predeterminado        |                 |    |
| Column Width:           | Predeterminado        |                 |    |
| Orden de combinación:   | Español               |                 |    |
| Permitir longitud cero: | Falso                 |                 |    |
| Posición ordinal:       | 10                    |                 |    |
| Requerido:              | Falso                 |                 |    |
| Tabla de origen:        | t_cliente_pedido      |                 |    |
| clpp_cantidad_abas      |                       | Número (doble)  | 8  |
| Atributos:              | Tamaño fijo           |                 |    |
| Campo de origen:        | clpp_cantidad_abas    |                 |    |
| Column Hidden:          | Falso                 |                 |    |
| Column Order:           | Predeterminado        |                 |    |
| Column Width:           | Predeterminado        |                 |    |
| Lugares decimales:      | 255                   |                 |    |
| Orden de combinación:   | Español               |                 |    |
| Permitir longitud cero: | Falso                 |                 |    |
| Posición ordinal:       | 11                    |                 |    |
| Requerido:              | Falso                 |                 |    |
| Tabla de origen:        | t_cliente_pedido      |                 |    |
| Valor predeterminado:   | 0                     |                 |    |
| clpp_orden_anterior     |                       | Texto           | 10 |
| Atributos:              | Longitud variable     |                 |    |
| Campo de origen:        | clpp_orden_anterior   |                 |    |
| Column Hidden:          | Falso                 |                 |    |
| Column Order:           | Predeterminado        |                 |    |
| Column Width:           | Predeterminado        |                 |    |
| Orden de combinación:   | Español               |                 |    |
| Permitir longitud cero: | Falso                 |                 |    |

Tabla: *t\_cliente\_pedido*

|                         |                    |                 |   |
|-------------------------|--------------------|-----------------|---|
| Posición ordinal:       | 12                 |                 |   |
| Requerido:              | Falso              |                 |   |
| Tabla de origen:        | t_cliente_pedido   |                 |   |
| clpp_estatus_envio      |                    | Número (simple) | 4 |
| Atributos:              | Tamaño fijo        |                 |   |
| Campo de origen:        | clpp_estatus_envio |                 |   |
| Column Hidden:          | Falso              |                 |   |
| Column Order:           | Predeterminado     |                 |   |
| Column Width:           | Predeterminado     |                 |   |
| Lugares decimales:      | 255                |                 |   |
| Orden de combinación:   | Español            |                 |   |
| Permitir longitud cero: | Falso              |                 |   |
| Posición ordinal:       | 13                 |                 |   |
| Requerido:              | Falso              |                 |   |
| Tabla de origen:        | t_cliente_pedido   |                 |   |
| Valor predeterminado:   | 0                  |                 |   |

**Indices de tablas**

| <u>Nombre</u>     | <u>Número de campos</u>      |
|-------------------|------------------------------|
| <i>PrimaryKey</i> | 4                            |
| Agrupado:         | Falso                        |
| Cuenta distinta:  | 5                            |
| Extranjero:       | Falso                        |
| Ignorar Nulos:    | Falso                        |
| Nombre:           | PrimaryKey                   |
| Principal:        | Verdadero                    |
| Requerido:        | Verdadero                    |
| Única:            | Verdadero                    |
| Campos:           | clpp_fecha, Ascendente       |
| Campos:           | id_cle_cliente, Ascendente   |
| Campos:           | id_prod_producto, Ascendente |
| Campos:           | id_alma_almacen, Ascendente  |

**Permisos de usuario**

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

**Permisos de grupo**

|                 |   |
|-----------------|---|
| Admins<br>Users | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-----------------|---|

Tabla: t\_cliente\_producto

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 1         | Última actualización: | 04/04/98 |

**Columnas**

| Nombre                  | Tipo               | Tamaño |
|-------------------------|--------------------|--------|
| id_clpr_producto        | Texto              | 10     |
| Atributos:              | Longitud variable  |        |
| Campo de origen:        | id_clpr_producto   |        |
| Column Hidden:          | Falso              |        |
| Column Order:           | 1                  |        |
| Column Width:           | Predeterminado     |        |
| Orden de combinación:   | Español            |        |
| Permitir longitud cero: | Falso              |        |
| Posición ordinal:       | 2                  |        |
| Requerido:              | Falso              |        |
| Tabla de origen:        | t_cliente_producto |        |
| id_clie_cliente         | Texto              | 10     |
| Atributos:              | Longitud variable  |        |
| Campo de origen:        | id_clie_cliente    |        |
| Column Hidden:          | Falso              |        |
| Column Order:           | Predeterminado     |        |
| Column Width:           | Predeterminado     |        |
| Orden de combinación:   | Español            |        |
| Permitir longitud cero: | Falso              |        |
| Posición ordinal:       | 3                  |        |
| Requerido:              | Falso              |        |
| Tabla de origen:        | t_cliente_producto |        |
| id_alma_almacen         | Texto              | 10     |
| Atributos:              | Longitud variable  |        |
| Campo de origen:        | id_alma_almacen    |        |
| Column Hidden:          | Falso              |        |
| Column Order:           | Predeterminado     |        |
| Column Width:           | Predeterminado     |        |
| Orden de combinación:   | Español            |        |
| Permitir longitud cero: | Falso              |        |
| Posición ordinal:       | 4                  |        |
| Requerido:              | Falso              |        |
| Tabla de origen:        | t_cliente_producto |        |
| id_prod_producto        | Texto              | 10     |
| Atributos:              | Longitud variable  |        |
| Campo de origen:        | id_prod_producto   |        |
| Column Hidden:          | Falso              |        |
| Column Order:           | Predeterminado     |        |
| Column Width:           | Predeterminado     |        |

Tabla: t\_cliente\_producto

|                         |                       |                |    |
|-------------------------|-----------------------|----------------|----|
| Orden de combinación:   | Español               |                |    |
| Permitir longitud cero: | Falso                 |                |    |
| Posición ordinal:       | 5                     |                |    |
| Requerido:              | Falso                 |                |    |
| Tabla de origen:        | t_cliente_producto    |                |    |
| clpr_prod_descripcion   |                       | Texto          | 50 |
| Atributos:              | Longitud variable     |                |    |
| Campo de origen:        | clpr_prod_descripcion |                |    |
| Column Hidden:          | Falso                 |                |    |
| Column Order:           | Predeterminado        |                |    |
| Column Width:           | Predeterminado        |                |    |
| Orden de combinación:   | Español               |                |    |
| Permitir longitud cero: | Falso                 |                |    |
| Posición ordinal:       | 6                     |                |    |
| Requerido:              | Falso                 |                |    |
| Tabla de origen:        | t_cliente_producto    |                |    |
| clpr_minimo             |                       | Número (doble) | 8  |
| Atributos:              | Tamaño fijo           |                |    |
| Campo de origen:        | clpr_minimo           |                |    |
| Column Hidden:          | Falso                 |                |    |
| Column Order:           | Predeterminado        |                |    |
| Column Width:           | Predeterminado        |                |    |
| Lugares decimales:      | 255                   |                |    |
| Orden de combinación:   | Español               |                |    |
| Permitir longitud cero: | Falso                 |                |    |
| Posición ordinal:       | 7                     |                |    |
| Requerido:              | Falso                 |                |    |
| Tabla de origen:        | t_cliente_producto    |                |    |
| Valor predeterminado:   | 0                     |                |    |
| clpr_maximo             |                       | Número (doble) | 8  |
| Atributos:              | Tamaño fijo           |                |    |
| Campo de origen:        | clpr_maximo           |                |    |
| Column Hidden:          | Falso                 |                |    |
| Column Order:           | Predeterminado        |                |    |
| Column Width:           | Predeterminado        |                |    |
| Lugares decimales:      | 255                   |                |    |
| Orden de combinación:   | Español               |                |    |
| Permitir longitud cero: | Falso                 |                |    |
| Posición ordinal:       | 8                     |                |    |
| Requerido:              | Falso                 |                |    |
| Tabla de origen:        | t_cliente_producto    |                |    |
| Valor predeterminado:   | 0                     |                |    |
| clpr_existenci          |                       | Número (doble) | 8  |
| Atributos:              | Tamaño fijo           |                |    |
| Campo de origen:        | clpr_existenci        |                |    |
| Column Hidden:          | Falso                 |                |    |
| Column Order:           | Predeterminado        |                |    |
| Column Width:           | Predeterminado        |                |    |
| Lugares decimales:      | 255                   |                |    |

Tabla: t\_cliente\_producto

|                         |                    |                        |          |
|-------------------------|--------------------|------------------------|----------|
| Orden de combinación:   | Español            |                        |          |
| Permitir longitud cero: | Falso              |                        |          |
| Posición ordinal:       | 9                  |                        |          |
| Requerido:              | Falso              |                        |          |
| Tabla de origen:        | t_cliente_producto |                        |          |
| Valor predeterminado:   | 0                  |                        |          |
| <b>clpr_desabasto</b>   |                    | <b>Número (doble)</b>  | <b>8</b> |
| Atributos:              | Tamaño fijo        |                        |          |
| Campo de origen:        | clpr_desabasto     |                        |          |
| Column Hidden:          | Falso              |                        |          |
| Column Order:           | Predeterminado     |                        |          |
| Column Width:           | Predeterminado     |                        |          |
| Lugares decimales:      | 255                |                        |          |
| Orden de combinación:   | Español            |                        |          |
| Permitir longitud cero: | Falso              |                        |          |
| Posición ordinal:       | 10                 |                        |          |
| Requerido:              | Falso              |                        |          |
| Tabla de origen:        | t_cliente_producto |                        |          |
| Valor predeterminado:   | 0                  |                        |          |
| <b>clpr_precio</b>      |                    | <b>Número (doble)</b>  | <b>8</b> |
| Atributos:              | Tamaño fijo        |                        |          |
| Campo de origen:        | clpr_precio        |                        |          |
| Column Hidden:          | Falso              |                        |          |
| Column Order:           | Predeterminado     |                        |          |
| Column Width:           | Predeterminado     |                        |          |
| Lugares decimales:      | 255                |                        |          |
| Orden de combinación:   | Español            |                        |          |
| Permitir longitud cero: | Falso              |                        |          |
| Posición ordinal:       | 11                 |                        |          |
| Requerido:              | Falso              |                        |          |
| Tabla de origen:        | t_cliente_producto |                        |          |
| Valor predeterminado:   | 0                  |                        |          |
| <b>clpr_descuento</b>   |                    | <b>Número (doble)</b>  | <b>8</b> |
| Atributos:              | Tamaño fijo        |                        |          |
| Campo de origen:        | clpr_descuento     |                        |          |
| Column Hidden:          | Falso              |                        |          |
| Column Order:           | Predeterminado     |                        |          |
| Column Width:           | Predeterminado     |                        |          |
| Lugares decimales:      | 255                |                        |          |
| Orden de combinación:   | Español            |                        |          |
| Permitir longitud cero: | Falso              |                        |          |
| Posición ordinal:       | 12                 |                        |          |
| Requerido:              | Falso              |                        |          |
| Tabla de origen:        | t_cliente_producto |                        |          |
| Valor predeterminado:   | 0                  |                        |          |
| <b>id_tipo_estatus</b>  |                    | <b>Número (simple)</b> | <b>4</b> |
| Atributos:              | Tamaño fijo        |                        |          |
| Campo de origen:        | id_tipo_estatus    |                        |          |
| Column Hidden:          | Falso              |                        |          |

Tabla: t\_cliente\_producto

|                    |                         |                    |                 |    |
|--------------------|-------------------------|--------------------|-----------------|----|
|                    | Column Order:           | Predeterminado     |                 |    |
|                    | Column Width:           | Predeterminado     |                 |    |
|                    | Lugares decimales:      | 255                |                 |    |
|                    | Orden de combinación:   | Español            |                 |    |
|                    | Permitir longitud cero: | Falso              |                 |    |
|                    | Posición ordinal:       | 13                 |                 |    |
|                    | Requerido:              | Falso              |                 |    |
|                    | Tabla de origen:        | t_cliente_producto |                 |    |
|                    | Valor predeterminado:   | 0                  |                 |    |
| clpr_fecha         |                         |                    | Fecha/Hora      | 8  |
|                    | Atributos:              | Tamaño fijo        |                 |    |
|                    | Campo de origen:        | clpr_fecha         |                 |    |
|                    | Column Hidden:          | Falso              |                 |    |
|                    | Column Order:           | Predeterminado     |                 |    |
|                    | Column Width:           | Predeterminado     |                 |    |
|                    | Orden de combinación:   | Español            |                 |    |
|                    | Permitir longitud cero: | Falso              |                 |    |
|                    | Posición ordinal:       | 15                 |                 |    |
|                    | Requerido:              | Falso              |                 |    |
|                    | Tabla de origen:        | t_cliente_producto |                 |    |
| clpr_usuario       |                         |                    | Texto           | 10 |
|                    | Atributos:              | Longitud variable  |                 |    |
|                    | Campo de origen:        | clpr_usuario       |                 |    |
|                    | Column Hidden:          | Falso              |                 |    |
|                    | Column Order:           | Predeterminado     |                 |    |
|                    | Column Width:           | Predeterminado     |                 |    |
|                    | Orden de combinación:   | Español            |                 |    |
|                    | Permitir longitud cero: | Falso              |                 |    |
|                    | Posición ordinal:       | 16                 |                 |    |
|                    | Requerido:              | Falso              |                 |    |
|                    | Tabla de origen:        | t_cliente_producto |                 |    |
| clpr_equivalencia  |                         |                    | Número (fargo)  | 4  |
|                    | Atributos:              | Tamaño fijo        |                 |    |
|                    | Campo de origen:        | clpr_equivalencia  |                 |    |
|                    | Column Hidden:          | Falso              |                 |    |
|                    | Column Order:           | Predeterminado     |                 |    |
|                    | Column Width:           | Predeterminado     |                 |    |
|                    | Lugares decimales:      | 255                |                 |    |
|                    | Orden de combinación:   | Español            |                 |    |
|                    | Permitir longitud cero: | Falso              |                 |    |
|                    | Posición ordinal:       | 18                 |                 |    |
|                    | Requerido:              | Falso              |                 |    |
|                    | Tabla de origen:        | t_cliente_producto |                 |    |
|                    | Valor predeterminado:   | 0                  |                 |    |
| clpr_estatus_envio |                         |                    | Número (simple) | 4  |
|                    | Atributos:              | Tamaño fijo        |                 |    |
|                    | Campo de origen:        | clpr_estatus_envio |                 |    |
|                    | Column Hidden:          | Falso              |                 |    |
|                    | Column Order:           | Predeterminado     |                 |    |

---

 Tabla: f\_cliente\_producto
 

---

|                         |                    |
|-------------------------|--------------------|
| Column Width:           | Predeterminado     |
| Lugares decimales:      | 255                |
| Orden de combinación:   | Español            |
| Permitir longitud cero: | Falso              |
| Posición ordinal:       | 19                 |
| Requerido:              | Falso              |
| Tabla de origen:        | f_cliente_producto |
| Valor predeterminado:   | 0                  |

Índices de tablas

| <u>Nombre</u>    | <u>Número de campos</u>     |
|------------------|-----------------------------|
| PrimaryKey       | 3                           |
| Agrupado:        | Falso                       |
| Cuenta distinta: | 1                           |
| Extranjero:      | Falso                       |
| Ignorar Nulos:   | Falso                       |
| Nombre:          | PrimaryKey                  |
| Principal:       | Verdadero                   |
| Requerido:       | Verdadero                   |
| Única:           | Verdadero                   |
| Campos:          | id_cpr_producto, Ascendente |
| Campos:          | id_cfie_cliente, Ascendente |
| Campos:          | id_alma_almacen, Ascendente |

Permisos de usuario

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

Permisos de grupo

|        |   |
|--------|---|
| Admins | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
| Users  | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |

Tabla: t\_factura\_orden

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 0         | Última actualización: | 04/04/98 |

Columnas

| Nombre   | Tipo  | Tamaño              |
|--|---|---------------------|
| <b>clpp_norden</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:        | Longitud variable<br>clpp_norden<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>1<br>Falso<br>t_factura_orden      | Texto<br>10         |
| <b>id_clpr_producto</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:   | Longitud variable<br>id_clpr_producto<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>2<br>Falso<br>t_factura_orden | Texto<br>10         |
| <b>clpp_fecha_factura</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen: | Tamaño fijo<br>clpp_fecha_factura<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>3<br>Falso<br>t_factura_orden     | Fecha/Hora<br>8     |
| <b>faor_cantidad_aba</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:   | Tamaño fijo<br>faor_cantidad_aba<br>Falso<br>Predeterminado<br>Predeterminado   | Número (doble)<br>8 |

Tabla: t\_factura\_orden

|                           |                    |                 |    |
|---------------------------|--------------------|-----------------|----|
| Lugares decimales:        | 255                |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 4                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_factura_orden    |                 |    |
| Valor predeterminado:     | 0                  |                 |    |
| <b>faor_usuario</b>       |                    | Texto           | 10 |
| Atributos:                | Longitud variable  |                 |    |
| Campo de origen:          | faor_usuario       |                 |    |
| Column Hidden:            | Falso              |                 |    |
| Column Order:             | Predeterminado     |                 |    |
| Column Width:             | Predeterminado     |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 5                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_factura_orden    |                 |    |
| <b>faor_estatus_envio</b> |                    | Número (simple) | 4  |
| Atributos:                | Tamaño fijo        |                 |    |
| Campo de origen:          | faor_estatus_envio |                 |    |
| Column Hidden:            | Falso              |                 |    |
| Column Order:             | Predeterminado     |                 |    |
| Column Width:             | Predeterminado     |                 |    |
| Lugares decimales:        | 255                |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 6                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_factura_orden    |                 |    |
| Valor predeterminado:     | 0                  |                 |    |

**Índices de tablas**

| Nombre           | Número de campos             |
|------------------|------------------------------|
| PrimaryKey       | 2                            |
| Agrupado:        | Falso                        |
| Cuenta distinta: | 0                            |
| Extranjero:      | Falso                        |
| Ignorar Nulos:   | Falso                        |
| Nombre:          | PrimaryKey                   |
| Principal:       | Verdadero                    |
| Requerido:       | Verdadero                    |
| Única:           | Verdadero                    |
| Campos:          | clpp_norden, Ascendente      |
| Campos:          | id_clpr_producto, Ascendente |

---

Tabla: t\_factura\_orden

---

Permisos de usuario

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

## Tabla: t\_formato

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 3         | Última actualización: | 04/04/98 |

Columnas

| Nombre   | Tipo           | Tamaño |
|--|----------------|--------|
| id_form_formato  | Número (doble) | 8      |
| Atributos:<br>Campo de origen: id_form_formato<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Descripción: Identificador del tipo de formato<br>Lugares decimales: 255<br>Orden de combinación: General<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_formato<br>Valor predeterminado: 0 |                |        |
| form_formato   | Texto          | 50     |
| Atributos:<br>Campo de origen: form_formato<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Descripción: Descripción del formato<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_formato   |                |        |

Índices de tablas

| Nombre           | Número de campos            |
|------------------|-----------------------------|
| PrimaryKey       | 1                           |
| Agrupado:        | Falso                       |
| Cuenta distinta: | 3                           |
| Extranjero:      | Falso                       |
| Ignorar Nulos:   | Falso                       |
| Nombre:          | PrimaryKey                  |
| Principal:       | Verdadero                   |
| Requerido:       | Verdadero                   |
| Única:           | Verdadero                   |
| Campos:          | id_form_formato, Ascendente |

---

Tabla: t\_formato

---

Permisos de usuario

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_funciones

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 27        | Última actualización: | 04/04/98 |

Columnas

| Nombre                  | Tipo              | Tamaño |
|-------------------------|-------------------|--------|
| id_func_funcion         | Texto             | 10     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | id_func_funcion   |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 1                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_funciones       |        |
| func_descripcion        | Texto             | 50     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | func_descripcion  |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | 5190              |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 2                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_funciones       |        |

Índices de tablas

| Nombre           | Número de campos            |
|------------------|-----------------------------|
| PrimaryKey       | 1                           |
| Agrupado:        | Falso                       |
| Cuenta distinta: | 27                          |
| Extranjero:      | Falso                       |
| Ignorar Nulos:   | Falso                       |
| Nombre:          | PrimaryKey                  |
| Principal:       | Verdadero                   |
| Requerido:       | Verdadero                   |
| Única:           | Verdadero                   |
| Campos:          | id_func_funcion, Ascendente |

---

Tabla: t\_funciones

---

**Permisos de usuario**

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

**Permisos de grupo**

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_miror\_carga

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 0         | Última actualización: | 04/04/98 |

**Columnas**

| Nombre  | Tipo   | Tamaño |
|---|--|--------|
| <b>mica_almacen</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:  | Texto<br>Longitud variable<br>mica_almacen<br>Falso<br>1<br>2700<br>Español<br>Falso<br>0<br>Falso<br>t_miror_carga                  | 50     |
| <b>mica_producto</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen: | Texto<br>Tamaño fijo<br>mica_producto<br>Falso<br>Predeterminado<br>2205<br>Español<br>Falso<br>1<br>Falso<br>t_miror_carga          | 50     |
| <b>mica_cliente</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:<br>Orden de combinación:<br>Permitir longitud cero:<br>Posición ordinal:<br>Requerido:<br>Tabla de origen:  | Texto<br>Tamaño fijo<br>mica_cliente<br>Falso<br>Predeterminado<br>Predeterminado<br>Español<br>Falso<br>2<br>Falso<br>t_miror_carga | 50     |
| <b>mica_desabasto</b><br>Atributos:<br>Campo de origen:<br>Column Hidden:<br>Column Order:<br>Column Width:   | Número (doble)<br>Tamaño fijo<br>mica_desabasto<br>Falso<br>Predeterminado<br>Predeterminado   | 8      |

---

Tabla: t\_mirroring\_carga

---

|                         |                   |
|-------------------------|-------------------|
| Lugares decimales:      | 255               |
| Orden de combinación:   | Español           |
| Permitir longitud cero: | Falso             |
| Posición ordinal:       | 3                 |
| Requerido:              | Falso             |
| Tabla de origen:        | t_mirroring_carga |
| Valor predeterminado:   | 0                 |

#### Permisos de usuario

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

#### Permisos de grupo

Admins

Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_producto

Propiedades

Def. actualizable: Verdadero      Fecha de creación: 28/02/98  
 Record Count: 0      Última actualización: 04/04/98

Columnas

| Nombre                  | Tipo              | Tamaño |
|-------------------------|-------------------|--------|
| id_prod_producto        | Texto             | 10     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | id_prod_producto  |        |
| Column Order:           | Falso             |        |
| Column Width:           | 1                 |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 1                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_producto        |        |
| prod_descripcion        | Texto             | 50     |
| Atributos:              |                   |        |
| Campo de origen:        | Longitud variable |        |
| Column Hidden:          | prod_descripcion  |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Orden de combinación:   | Predeterminado    |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 2                 |        |
| Tabla de origen:        | Falso             |        |
|                         | t_producto        |        |
| prod_precio             | Número (doble)    | 8      |
| Atributos:              |                   |        |
| Campo de origen:        | Tamaño fijo       |        |
| Column Hidden:          | prod_precio       |        |
| Column Order:           | Falso             |        |
| Column Width:           | Predeterminado    |        |
| Lugares decimales:      | Predeterminado    |        |
| Orden de combinación:   | 255               |        |
| Permitir longitud cero: | Español           |        |
| Posición ordinal:       | Falso             |        |
| Requerido:              | 3                 |        |
| Tabla de origen:        | Falso             |        |
| Valor predeterminado:   | t_producto        |        |
|                         | 0                 |        |
| prod_fecha              | Fecha/Hora        | 8      |
| Atributos:              |                   |        |
| Campo de origen:        | Tamaño fijo       |        |
| Column Hidden:          | prod_fecha        |        |
|                         | Falso             |        |

## Tabla: t\_producto

|                           |                    |                 |    |
|---------------------------|--------------------|-----------------|----|
| Column Order:             | Predeterminado     |                 |    |
| Column Width:             | Predeterminado     |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 4                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_producto         |                 |    |
| <i>prod_usuario</i>       |                    | Texto           | 10 |
| Atributos:                | Longitud variable  |                 |    |
| Campo de origen:          | prod_usuario       |                 |    |
| Column Hidden:            | Falso              |                 |    |
| Column Order:             | Predeterminado     |                 |    |
| Column Width:             | Predeterminado     |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 5                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_producto         |                 |    |
| <i>prod_estatus_envio</i> |                    | Número (simple) | 4  |
| Atributos:                | Tamaño fijo        |                 |    |
| Campo de origen:          | prod_estatus_envio |                 |    |
| Column Hidden:            | Falso              |                 |    |
| Column Order:             | Predeterminado     |                 |    |
| Column Width:             | Predeterminado     |                 |    |
| Lugares decimales:        | Ascendente         |                 |    |
| Orden de combinación:     | Español            |                 |    |
| Permitir longitud cero:   | Falso              |                 |    |
| Posición ordinal:         | 6                  |                 |    |
| Requerido:                | Falso              |                 |    |
| Tabla de origen:          | t_producto         |                 |    |
| Valor predeterminado:     | 0                  |                 |    |

Índices de tablas

| <u>Nombre</u>     | <u>Número de campos</u>      |
|-------------------|------------------------------|
| <i>PrimaryKey</i> | 1                            |
| Agrupado:         | Falso                        |
| Cuenta distinta:  | 0                            |
| Extranjero:       | Falso                        |
| Ignorar Nulos:    | Falso                        |
| Nombre:           | <i>PrimaryKey</i>            |
| Principal:        | Verdadero                    |
| Requerido:        | <i>Verdadero</i>             |
| Única:            | Verdadero                    |
| Campos:           | Id_prod_producto, Ascendente |

---

Tabla: t\_producto

---

**Permisos de usuario**

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

**Permisos de grupo**

Admins

Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_tablas

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 6         | Última actualización: | 04/04/98 |

Columnas

| Nombre   | Tipo            | Tamaño |
|--|-----------------|--------|
| id_tab_tabla   | Texto           | 8      |
| Atributos: Longitud variable<br>Campo de origen: id_tab_tabla<br>Column Hidden: Falso<br>Column Order: 1<br>Column Width: Predeterminado<br>Descripción: Clave de la tabla o archivo de la base de datos<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_tablas   |                 |        |
| tab_nombre   | Texto           | 30     |
| Atributos: Longitud variable<br>Campo de origen: tab_nombre<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Descripción: Nombre con el que se reconoce la tabla en la aplicación<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_tablas  |                 |        |
| tab_ncampos  | Número (simple) | 4      |
| Atributos: Tamaño fijo<br>Campo de origen: tab_ncampos<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Descripción: Numero de campos por cada registro del archivo<br>Lugares decimales: 255<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 3<br>Requerido: Falso<br>Tabla de origen: t_tablas<br>Valor predeterminado: 0 |                 |        |

---

 Tabla: t\_tablas
 

---

**Índices de tablas**

| <u>Nombre</u>    | <u>Número de campos</u>  |
|------------------|--------------------------|
| PrimaryKey       | 1                        |
| Agrupado:        | Falso                    |
| Cuenta distinta: | 6                        |
| Extranjero:      | Falso                    |
| Ignorar Nulos:   | Falso                    |
| Nombre:          | PrimaryKey               |
| Principal:       | Verdadero                |
| Requerido:       | Verdadero                |
| Única:           | Verdadero                |
| Campos:          | id_tab_tabla, Ascendente |

**Permisos de usuario**

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

**Permisos de grupo**

|        |   |
|--------|---|
| Admins | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
| Users  |   |

## D40

Tabla: t\_tipo\_archivo

### Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 3         | Última actualización: | 04/04/98 |

### Columnas

| Nombre                                | Tipo           | Tamaño |
|---------------------------------------|----------------|--------|
| id_tiar_tipo_archivo                  | Número (largo) | 4      |
| Atributos:                            |                |        |
| Tamaño fijo, Autoincremento           |                |        |
| Campo de origen: id_tiar_tipo_archivo |                |        |
| Column Hidden: Falso                  |                |        |
| Column Order: 1                       |                |        |
| Column Width: 1860                    |                |        |
| Orden de combinación: General         |                |        |
| Permitir longitud cero: Falso         |                |        |
| Posición ordinal: 1                   |                |        |
| Requerido: Falso                      |                |        |
| Tabla de origen: t_tipo_archivo       |                |        |
| tiar_descripcion                      | Texto          | 50     |
| Atributos:                            |                |        |
| Longitud variable                     |                |        |
| Campo de origen: tiar_descripcion     |                |        |
| Column Hidden: Falso                  |                |        |
| Column Order: Predeterminado          |                |        |
| Column Width: 2010                    |                |        |
| Orden de combinación: Español         |                |        |
| Permitir longitud cero: Falso         |                |        |
| Posición ordinal: 2                   |                |        |
| Requerido: Falso                      |                |        |
| Tabla de origen: t_tipo_archivo       |                |        |

### Índices de tablas

| Nombre                                   | Número de campos |
|--|------------------|
| PrimaryKey                               | 1                |
| Agrupado: Falso                          |                  |
| Cuenta distinta: 3                       |                  |
| Extranjero: Falso                        |                  |
| Ignorar Nulos: Falso                     |                  |
| Nombre: PrimaryKey                       |                  |
| Principal: Verdadero                     |                  |
| Requerido: Verdadero                     |                  |
| Única: Verdadero                         |                  |
| Campos: id_tiar_tipo_archivo, Ascendente |                  |

---

Tabla: t\_tipo\_archivo

---

**Permisos de usuario**

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

**Permisos de grupo**

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

---

 Tabla: t\_tipo\_pedido
 

---

Propiedades

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 2         | Última actualización: | 04/04/98 |

Columnas

| Nombre                  | Tipo  | Tamaño |
|-------------------------|---|--------|
| Id_tipo_estatus         | Número (simple)                                 | 4      |
| Atributos:              |   |        |
| Campo de origen:        | Tamaño fijo                                     |        |
| Column Hidden:          | id_tipo_estatus                                 |        |
| Column Order:           | Falso   |        |
| Column Width:           | Predeterminado                                  |        |
| Descripción:            | Predeterminado                                  |        |
| Lugares decimales:      | Clave numérica del tipo de pedido (consecutivo) |        |
| Orden de combinación:   | 255   |        |
| Permitir longitud cero: | General   |        |
| Posición ordinal:       | Falso   |        |
| Requerido:              | 1   |        |
| Tabla de origen:        | Falso   |        |
| Valor predeterminado:   | t_tipo_pedido                                   |        |
| Valor predeterminado:   | 0   |        |
| tipo_descripción        | Texto   | 30     |
| Atributos:              |   |        |
| Campo de origen:        | Longitud variable                               |        |
| Column Hidden:          | tipo_descripción                                |        |
| Column Order:           | Falso   |        |
| Column Width:           | Predeterminado                                  |        |
| Descripción:            | Predeterminado                                  |        |
| Orden de combinación:   | Descripción del tipo de pedido                  |        |
| Permitir longitud cero: | Español   |        |
| Posición ordinal:       | Falso   |        |
| Requerido:              | 2   |        |
| Tabla de origen:        | Falso   |        |
| Tabla de origen:        | t_tipo_pedido                                   |        |

Índices de tablas

| Nombre           | Número de campos            |
|------------------|-----------------------------|
| PrimaryKey       | 1                           |
| Agrupado:        |                             |
| Cuenta distinta: | Falso                       |
| Extranjero:      | 2                           |
| Ignorar Nulos:   | Falso                       |
| Nombre:          | Falso                       |
| Principal:       | PrimaryKey                  |
| Requerido:       | Verdadero                   |
| Única:           | Verdadero                   |
| Campos:          | Verdadero                   |
|                  | Id_tipo_estatus, Ascendente |

---

Tabla: t\_tipo\_pedido

---

**Permisos de usuario**

admin

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

**Permisos de grupo**

Admins

Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos

Tabla: t\_usuafunc

**Propiedades**

|                    |           |                       |          |
|--------------------|-----------|-----------------------|----------|
| Def. actualizable: | Verdadero | Fecha de creación:    | 28/02/98 |
| Record Count:      | 4         | Row Height:           | 240      |
|                    |           | Última actualización: | 04/04/98 |

**Columnas**

| Nombre   | Tipo       | Tamaño |
|--|------------|--------|
| id_usuario   | Texto      | 10     |
| Atributos:<br>Campo de origen: id_usuario<br>Column Hidden: Falso<br>Column Order: 1<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_usuafunc                   |            |        |
| id_func_funcion  | Texto      | 10     |
| Atributos:<br>Campo de origen: id_func_funcion<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_usuafunc |            |        |
| func_fecha   | Fecha/Hora | 8      |
| Atributos:<br>Campo de origen: func_fecha<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 3<br>Requerido: Falso<br>Tabla de origen: t_usuafunc      |            |        |
| func_usuario   | Texto      | 10     |
| Atributos:<br>Campo de origen: func_usuario<br>Column Hidden: Falso<br>Column Order: Predeterminado  |            |        |

---

 Tabla: t\_usuafunc
 

---

|                         |                |
|-------------------------|----------------|
| Column Width:           | Predeterminado |
| Orden de combinación:   | Español        |
| Permitir longitud cero: | Falso          |
| Posición ordinal:       | 4              |
| Requendo:               | Falso          |
| Tabla de origen:        | t_usuafunc     |

### Índices de tablas

| <u>Nombre</u>    | <u>Número de campos</u>     |
|------------------|-----------------------------|
| PrimaryKey       | 2                           |
| Agrupado:        | Falso                       |
| Cuenta distinta: | 4                           |
| Extranjero:      | Falso                       |
| Ignorar Nulos:   | Falso                       |
| Nombre:          | PrimaryKey                  |
| Principal:       | Verdadero                   |
| Requerido:       | Verdadero                   |
| Única:           | Verdadero                   |
| Campos:          | id_usuario, Ascendente      |
| Campos:          | Id_func_función, Ascendente |

### Permisos de usuario

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

### Permisos de grupo

|        |   |
|--------|---|
| Admins | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
| Users  |   |

Tabla: t\_usuario

Propiedades

Def. actualizable: Verdadero  
Record Count: 2

Fecha de creación: 28/02/98  
Última actualización: 04/04/98

Columnas

| Nombre  | Tipo       | Tamaño |
|---|------------|--------|
| id_usuario  | Texto      | 10     |
| Atributos:<br>Campo de origen: id_usuario<br>Column Hidden: Falso<br>Column Order: 1<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 1<br>Requerido: Falso<br>Tabla de origen: t_usuario                 |            |        |
| usua_password   | Texto      | 10     |
| Atributos:<br>Campo de origen: usua_password<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 2<br>Requerido: Falso<br>Tabla de origen: t_usuario |            |        |
| usua_nombre   | Texto      | 50     |
| Atributos:<br>Campo de origen: usua_nombre<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: 2520<br>Orden de combinación: Español<br>Permitir longitud cero: Falso<br>Posición ordinal: 3<br>Requerido: Falso<br>Tabla de origen: t_usuario             |            |        |
| usua_fecha  | Fecha/Hora | 8      |
| Atributos:<br>Campo de origen: usua_fecha<br>Column Hidden: Falso<br>Column Order: Predeterminado<br>Column Width: Predeterminado   |            |        |

Tabla: t\_usuario

|                |                         |                   |       |    |
|----------------|-------------------------|-------------------|-------|----|
|                | Orden de combinación:   | Español           |       |    |
|                | Permitir longitud cero: | Falso             |       |    |
|                | Posición ordinal:       | 4                 |       |    |
|                | Requerido:              | Falso             |       |    |
|                | Tabla de origen:        | t_usuario         |       |    |
| usua_usuario   |                         |                   | Texto | 10 |
|                | Atributos:              | Longitud variable |       |    |
|                | Campo de origen:        | usua_usuario      |       |    |
|                | Column Hidden:          | Falso             |       |    |
|                | Column Order:           | Predeterminado    |       |    |
|                | Column Width:           | Predeterminado    |       |    |
|                | Orden de combinación:   | Español           |       |    |
|                | Permitir longitud cero: | Falso             |       |    |
|                | Posición ordinal:       | 5                 |       |    |
|                | Requerido:              | Falso             |       |    |
|                | Tabla de origen:        | t_usuario         |       |    |
| usua_ubicacion |                         |                   | Texto | 1  |
|                | Atributos:              | Longitud variable |       |    |
|                | Campo de origen:        | usua_ubicacion    |       |    |
|                | Column Hidden:          | Falso             |       |    |
|                | Column Order:           | Predeterminado    |       |    |
|                | Column Width:           | Predeterminado    |       |    |
|                | Orden de combinación:   | Español           |       |    |
|                | Permitir longitud cero: | Falso             |       |    |
|                | Posición ordinal:       | 6                 |       |    |
|                | Requerido:              | Falso             |       |    |
|                | Tabla de origen:        | t_usuario         |       |    |

**Índices de tablas**

| <u>Nombre</u>    | <u>Número de campos</u> |
|------------------|-------------------------|
| PrimaryKey       | 1                       |
| Agrupado:        | Falso                   |
| Cuenta distinta: | 2                       |
| Extranjero:      | Falso                   |
| Ignorar Nulos:   | Falso                   |
| Nombre:          | PrimaryKey              |
| Principal:       | Verdadero               |
| Requerido:       | Verdadero               |
| Única:           | Verdadero               |
| Campos:          | id_usuario, Ascendente  |

**Permisos de usuario**

|       |   |
|-------|---|
| admin | Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos |
|-------|---|

---

Tabla: t\_usuario

---

Permisos de grupo

Admins  
Users

Eliminar, Leer permisos, Establecer permisos, Cambiar propietario, Leer definición, Escribir definición, Leer datos, Insertar datos, Actualizar datos, Eliminar datos