

## REPOSITORIO ACADÉMICO DIGITAL INSTITUCIONAL

**Sistema de control de expedientes para un consultorio médico**

**Autor: Francisco Javier Gutiérrez Torres**

**Tesina presentada para obtener el título de:  
Ing. En Sistemas Computarizados [sic]**

**Nombre del asesor:  
Manoella Atziry Hernandez Montiel**

Este documento está disponible para su consulta en el Repositorio Académico Digital Institucional de la Universidad Vasco de Quiroga, cuyo objetivo es integrar organizar, almacenar, preservar y difundir en formato digital la producción intelectual resultante de la actividad académica, científica e investigadora de los diferentes campus de la universidad, para beneficio de la comunidad universitaria.

Esta iniciativa está a cargo del Centro de Información y Documentación “Dr. Silvio Zavala” que lleva adelante las tareas de gestión y coordinación para la concreción de los objetivos planteados.

Esta Tesis se publica bajo licencia Creative Commons de tipo “Reconocimiento-NoComercial-SinObraDerivada”, se permite su consulta siempre y cuando se mantenga el reconocimiento de sus autores, no se haga uso comercial de las obras derivadas.



**UNIVERSIDAD VASCO DE QUIROGA**



**ESCUELA DE SISTEMAS  
COMPUTARIZADOS**

**“ Sistema de control de  
expedientes para  
un consultorio médico ”**

**TESINA**

**Que para obtener el Título de :  
LICENCIADO EN SISTEMAS  
COMPUTARIZADOS**

**Presenta :  
Francisco Javier Gutiérrez Torres**

**Asesor :  
L. S. C. Manoella Atziray Hernández  
Montiel**

**CLAVE 16PSU0014Q  
ACUERDO 952006**

**Abril 1999  
Morelia, Mich.**



# INDICE

1. INTRODUCCION.....	1
2. ANTECEDENTES.....	4
2.1 ANTECEDENTES TECNICOS.....	4
2.1.1 Definición de Computadora.....	4
2.1.2 Clasificación de las computadoras.....	5
2.2 ANTECEDENTES DEL SISTEMA.....	9
3. OBJETIVOS.....	15
3.1 OBJETIVO GENERAL.....	15
3.2 OBJETIVOS PARTICULARES.....	15
4. ANALISIS DE SISTEMAS DE INFORMACION.....	16
4.1 SISTEMAS.....	16
4.2 CARACTERISTICAS DE LOS SISTEMAS.....	18
4.3 TIPOS DE SISTEMAS.....	18
5. CICLO DE VIDA DEL DESARROLLO DE SISTEMAS.....	21
5.1 INVESTIGACION PRELIMINAR.....	21
5.2 DISEÑO DEL SISTEMA.....	25
5.2.1 Descripción general de la Base de Datos.....	27
5.3 DESARROLLO DE SOFTWARE.....	29
5.3.1 Programación Orientada a Objetos.....	29
5.3.2 Visual Fox Pro.....	34
5.3.3 Bases de Datos.....	42
5.4 PRUEBA DEL SISTEMA.....	45
5.5 PUESTA EN MARCHA.....	46

<b>6. CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>48</b>
<b>6.1 CONCLUSIONES.....</b>	<b>48</b>
<b>6.2 RECOMENDACIONES.....</b>	<b>49</b>
<b>INTRODUCCION.</b>	
<b>BIBLIOGRAFIA.....</b>	<b>50</b>
<b>APENDICE.....</b>	<b>53</b>

En mi caso me he dedicado a la medicina tradicional para tratar a los pacientes con problemas de salud. En mi caso me he dedicado a la medicina tradicional para tratar a los pacientes con problemas de salud.

El lenguaje ha surgido como un medio de comunicación entre los seres humanos. El lenguaje ha surgido como un medio de comunicación entre los seres humanos.

A medida que los problemas se vuelven cada vez más complejos, surgen lenguajes más potentes que los resuelven de un modo más práctico y sencillo. A medida que los problemas se vuelven cada vez más complejos, surgen lenguajes más potentes que los resuelven de un modo más práctico y sencillo.

Los programas realizados con los Lenguajes Procedurales tradicionales son difíciles de modificar y mantener. La forma de resolver este problema ha sido el desarrollo de una metodología denominada Desarrollo Estructurado.

# 1

## INTRODUCCION.

Con el avance de la tecnología en nuestros tiempos se presentan necesidades cada vez mayores, sin importar el giro del negocio. En mi caso me enfoque a un consultorio médico, para llevar el control adecuado de sus pacientes, así como un registro de los medicamentos y la emisión de recetas.

Se ha sugerido que la Programación Orientada a Objetos surgió como resultado del caos que ocurrió cuando los lenguajes tradicionales trataron de enfrentarse a los grandes problemas.

A medida que los problemas se vuelven cada vez más complejos, surgen Lenguajes más potentes que los resuelven de una manera mas práctica y sencilla. En la evolución de cada lenguaje, llega un momento en el que el programador comienza a tener dificultades a la hora de manejar programas que sean de cierto tamaño y sofisticación. La complejidad aumenta a medida que se dispone de más recursos.

Los programas realizados con los Lenguajes Procedurales tradicionales pueden ser difíciles de modificar y mantener. La forma de resolver este problema fue a través de una metodología denominada Desarrollo Estructurado.

El Desarrollo Estructurado requiere de un alto grado de prudencia y planificación para que el proyecto quede correctamente y sin errores.

La Técnica Orientada a Objetos tiende a enfatizar el diseño de los Sistemas a desarrollar, incluso pensar que la Programación Orientada a Objetos es un desarrollo mejor estructurado. Un Lenguaje Orientado a Objetos enfatiza los tipos de datos y las operaciones intrínsecas de forma tal que se hace más sencillo su manejo.

En la Programación Orientada a Objetos, los datos no fluyen libremente por el Sistema, ya que están protegidos de alguna modificación accidental. A diferencia del enfoque de procedimientos que invocaba a una función que modifica a un dato, en el Lenguaje Orientado a Objetos se envía un mensaje al Objeto; lo que permite una dinámica del sistema mas organizada.

En la actualidad se pueden desarrollar diferentes Sistemas para diversas Profesiones, Organismos, Instituciones, etc.

La rama de la Medicina desde siempre ha jugado un papel muy importante en la vida del ser humano.

Desde nuestros antepasados han existido personas que se dedican a curar gente, conocidas científicamente como Doctores, quienes son los que laboran en la rama de la Medicina.

La Medicina es un campo muy amplio en el cual se pueden desarrollar varios sistemas, una aplicación importante es el control que deben de tener los doctores con los expedientes de los pacientes, así como, los medicamentos que deben recetar.

## 2.1 ANTECEDENTES TECNICOS.

Pensando en las necesidades que presenta el consultorio médico, el desarrollo de esta Tesina esta enfocado en el registro y control del expediente de los pacientes y medicamentos, para un Medico Otorrinolaringologo de la ciudad de Salamanca, Guanajuato.

## CONCLUSIONES

El sistema desarrollado es capaz de interpretar y ejecutar una serie de acciones basadas en reglas, lo que permite la automatización de la información y a la toma de decisiones.

## 2

# ANTECEDENTES.

## 2.1 ANTECEDENTES TECNICOS.

### 2.1.1 DEFINICION DE COMPUTADORA.

- \* Máquina capaz de efectuar una secuencia de operaciones mediante un programa, de tal manera, que se realiza un procesamiento sobre un conjunto de datos de entrada, obteniéndose otro conjunto de datos de salida.
- \* Máquina universal, capaz de interpretar y ejecutar una serie de operaciones elementales, relativas al tratamiento de la información y a la resolución de tareas.

### COMPONENTES.

- A. Una parte física (Hardware). El hardware está configurado por componentes microelectrónicos avanzados, que proporcionan el soporte físico necesario para la interpretación y ejecución de instrucciones o algoritmos elementales.

B. Una parte lógica (Software). El Software consiste en un conjunto de instrucciones comprensibles a la máquina, que conforman un programa o programas aptos para desarrollar una tarea o aplicación concreta.

La potencia de una computadora viene dada por la velocidad de desarrollo o ejecución de instrucciones y no por la complejidad de su circuitería e instrucciones que suelen ser sencillas.

### **2.1.2 CLASIFICACION DE LAS COMPUTADORAS.**

Las computadoras se clasifican de acuerdo a su tamaño:

- \* Supercomputadoras.
- \* Macrocomputadoras.
- \* Minicomputadoras.
- \* Microcomputadoras o PC's.

#### **SUPERCOMPUTADORAS.**

Una Supercomputadora es el tipo de computadora más potente y más rápida que existe en un momento dado. Estas máquinas están diseñadas para procesar enormes cantidades de información en poco tiempo y son dedicadas a una tarea específica.

Así mismo son las más caras, sus precios alcanzan los 30 millones de dólares y más. También cuentan con un control de temperatura especial, esto para disipar el calor que algunos componentes alcanzan a tener.

Unos ejemplos de tareas a las que son expuestas las Supercomputadoras son los siguientes:

1. Búsqueda y estudio de la energía y armas nucleares.
2. Búsqueda de yacimientos petrolíferos con grandes bases de datos sísmicos.
3. El estudio y predicción de tornados.
4. El estudio y predicción del clima de cualquier parte del mundo.
5. La elaboración de maquetas y proyectos de la creación de aviones, simuladores de vuelo, etc.

Debido a su precio, son muy pocas las Supercomputadoras que se construyen en un año.

### **MACROCOMPUTADORAS.**

Las Macrocomputadoras son también conocidas como Mainframes. Los Mainframes son grandes, rápidos y caros sistemas que son capaces de controlar cientos de usuarios simultáneamente, así como cientos de dispositivos de entrada y salida. Los Mainframes tienen un costo que va desde 350,000 dólares hasta varios millones de dólares.

De alguna forma los Mainframes son más poderosos que las Supercomputadoras porque soportan más programas simultáneamente. Pero las Supercomputadoras pueden ejecutar un sólo programa más rápido que un Mainframe.

En el pasado, los Mainframes ocupaban cuartos completos o hasta pisos enteros de algún edificio; hoy en día, un Mainframe es parecido a una hilera de archiveros en algún cuarto con piso falso, esto para ocultar los cientos de cables de los periféricos, y su temperatura tiene que estar controlada.

### **MINICOMPUTADORAS.**

En 1960 surgió la Minicomputadora, una versión más pequeña de la Macrocomputadora, al ser orientada a tareas específicas, no necesitaba de todos los periféricos que necesita un Mainframe, y esto ayudo a reducir el precio y costos de mantenimiento.

Las Minicomputadoras, en tamaño y poder de procesamiento, se encuentran entre los Mainframes y las estaciones de trabajo. En general, una Minicomputadora es un sistema de multiproceso (varios procesos en paralelo) capaz de soportar de 10 hasta 200 usuarios simultáneamente. Actualmente se usan para almacenar grandes Bases de Datos, automatización industrial y de aplicaciones multiusuario.

## MICROCOMPUTADORAS.

Las Microcomputadoras o Computadoras Personales (PC's) tuvieron su origen con la creación de los Microprocesadores. Un Microprocesador es "una computadora en un chip", o sea un circuito integrado independiente.

El término PC se deriva de que para el año de 1981, IBM®, sacó a la venta su modelo "IBM PC", la cual se convirtió en un tipo de computadora ideal para uso "personal", de ahí que el término "PC" se estandarizó y los clones que sacaron posteriormente otras empresas fueron llamados "PC y compatibles", usando procesadores del mismo tipo que las IBM, pero a un costo menor y pudiendo ejecutar el mismo tipo de programas.

Existen otros tipos de Microcomputadoras, como la Macintosh®, que no son compatibles con la IBM, pero que en muchos de los casos se les llaman también "PC's", por ser de uso personal. En la actualidad existen variados tipos en el diseño de PC's: Computadoras personales, con el gabinete tipo minitorre, separado del monitor. Computadoras personales portátiles "Laptop" o "Notebook".

Computadoras personales más comunes, con el gabinete horizontal, separado del monitor. Computadoras personales que están en una sola unidad compacta el monitor y el CPU.

Las computadoras "laptops" son aquellas computadoras que están diseñadas para poder ser transportadas de un lugar a otro.

## 2.2 ANTECEDENTES DEL SISTEMA.

El Doctor tiene la especialidad en otorrinolaringología. El lugar donde da atención a sus pacientes, es su consultorio particular donde atiende a personas enfermas de la nariz, oído y laringe.

Algunas enfermedades que se pueden citar son:

- \* Amigdalitis.

Es la infección por virus de los folículos linfoides de las amígdalas.

Se divide en agudas y crónicas.

- \* Desviación Septal Obstruida.

Es la deformación del Septum nasal que produce obstrucción parcial o total de una de las dos fosas nasales.

- \* Epitaxis traumática.

Es cuando se presenta hemorragia por las fosas nasales. Se divide en agudas y crónicas

\* Faringitis.

Infección aguda de la faringe que puede ser epidémica o no y que puede ser viral o bacteriana. Incluye una serie de padecimientos infecciosos de la laringe y regiones vecinas, los padecimientos del tejido linfoide serán consideradas separadamente.

\* Laringitis.

Es la inflamación aguda de la mucosa laringea. Se divide en aguda y crónica.

\* Catarro del oído medio.

Es la acumulación del liquido en el oído medio en ausencia del proceso infeccioso. La cantidad del liquido es variable y se produce cuando el drenaje normal de la trompa de Eustaquio esta alterado.

\* Sinusitis.

Infección aguda de la mucosa que recubre las cavidades de los senos paranasales, y que se acompaña de acumulación de la secreción y exudados dentro de los mismos.

A continuación se mencionan algunos nombres genéricos y presentación de algunos medicamentos.

Acido Acetilsalicílico.	Tabletas de 500 mg.
Dimetilpirazolona.	Tabletas de 500 mg.
Penicilina G.	Frasco de ampulas de 8000 U.
Eritromicina.	Cápsulas de 250 mg.
Dicloxacilina.	Cápsulas de 250 mg.
Ampicilina.	Solución 1 %, Frasco, gotero de 30 ml.
Felineprina	Frasco, gotero de 10 ml.
Xilometozolina pseudoefredina.	Tabletas de 60 mg.

\* Tablero.

Todos los medicamentos pertenecen a un grupo de medicinas determinado.

El consultorio se encuentra ubicado en la ciudad de Salamanca, Guanajuato. La administración del consultorio esta formada por el Doctor y una secretaria.

El consultorio consta de un recibidor donde se encuentran los archivos, en donde son guardados los expedientes de los pacientes, estos se encuentran ordenados alfabéticamente de acuerdo al apellido paterno.

La actividad que desempeña la secretaria, es el dar la cita a los pacientes, quienes la solicitan personalmente o por teléfono.

Las consultas se dan de acuerdo a la hora que es solicitada por el paciente, y son tomados sus datos del paciente, en un libro de citas, dando un promedio de 20 minutos por cita.

Cuando un paciente se presenta por primera vez a la consulta, la secretaria se encarga de abrir el expediente del paciente anotando sus datos personales como son:

- \* Nombre completo.
- \* Edad.
- \* Dirección.
- \* Teléfono.
- \* Comentarios.

Posteriormente el paciente es pasado a atención médica, donde es atendido y después se elabora una receta, si es necesario. Si el Doctor por el momento no recuerda el nombre correcto de un medicamento o la dosis que se debe de tomar y su presentación, consulta el diccionario de especialidades farmacéuticas; y posteriormente la secretaria anota al paciente para su nueva cita si es necesario.

Como sabemos, el área de la Medicina es muy amplia, existiendo desde la Medicina general hasta las diferentes especialidades que nos podemos encontrar.

El desarrollo de esta aplicación se puede implementar a cualquier rama de la Medicina ya que todo médico requiere llevar un control de expedientes y un catálogo de medicinas.

Debido a que el Doctor da consulta a una gran cantidad de pacientes, existen varios problemas a los que está enfrentando actualmente. El no encontrar los expedientes, por diversas razones:

- A. El Doctor tenga un expediente que la secretaria este buscando y viceversa.
- B. Que el expediente se encuentre en el archivo equivocado.
- C. Que existan expedientes duplicados por error humano.

Este último caso en particular, ocasiona gran descontrol en la atención de un paciente, ya que no se sabría el avance que a tenido el paciente con respecto a su enfermedad, el recetar medicamentos que le ocasionen daños alérgicos, el repetir cierto medicamento, etc. Ocasionando un seguimiento de un tratamiento inadecuado.

En ocasiones los pacientes terminan su tratamiento y después de un tiempo regresan a una nueva consulta, y se ve en la necesidad de crear o abrir un nuevo expediente, y no se sabe si el paciente ya estuvo en cierto tratamiento para conocer su historial.

Por otro lado también existe un Diccionario de especialidades farmacéuticas donde se encuentran todas las medicinas que están circulando en la actualidad; en muchas ocasiones dejan de existir o cambian de nombre y el estar consultando el diccionario de especialidades farmacéuticas ocasiona perdida de tiempo.

Sé penso que con un sistema que permita tener el control de pacientes, expedientes médicos, así como del catálogo de medicamentos, y nos permita elaborar las recetas de una manera más rápida, se solucionaría la problemática que enfrenta el Doctor ya que se evitaría la duplicidad de expedientes y se tendría un control mas preciso de las consultas.

Diseñar e implementar un sistema para llevar un control y registro del expediente médico de cada uno de los pacientes que acuden con el Doctor, para su atención médica.

### 3.2 OBJETIVOS PARTICULARES.

1. Tener un control adecuado de todos los pacientes que consultan.
2. Controlar los errores humanos en la duplicidad de expedientes.
3. Agilizar los tiempos de búsqueda de los expedientes.
4. Modernizar los procesos administrativos.
5. Colocar el consultorio médico a la vanguardia de la tecnología.

# 3

## OBJETIVOS.

### 3.1 OBJETIVO GENERAL.

Diseñar e implementar un sistema para llevar un control y registro del expediente médico de cada uno de los pacientes que acuden con el Doctor, para su atención médica.

### 3.2 OBJETIVOS PARTICULARES.

- \* Tener un control adecuado de todos los pacientes que toman consulta.

- \* Disminuir los errores humanos en la duplicidad de expedientes.

- \* Agilizar los tiempos de búsqueda de los expedientes.

- \* Mejorar los procesos administrativos.

- \* Colocar el consultorio médico a la vanguardia de la tecnología.

## 4

# ANALISIS DE SISTEMAS DE INFORMACION.

Un Sistema de información es un conjunto de personas, datos y procedimientos que interactúan entre sí para alcanzar un objetivo común para apoyar a las actividades de la organización.

El Análisis de Sistemas se refiere al proceso de examinar una situación de una empresa con la intención de mejorarla mediante nuevos procedimientos y métodos.

## 4.1 SISTEMAS.

“ Un sistema es un conjunto de componentes que interactúan entre sí para alcanzar un objetivo. “

Los Sistemas son todo lo que rodea al ser humano; por ejemplo, se sienten sensaciones físicas originadas por un complejo sistema nervioso, un conjunto de partes que incluye al cerebro, espina dorsal, nervios y células sensitivas especiales debajo de la piel, que trabajan conjuntamente para hacer sentir calor, frío, comezón, etc.

4.2 El hombre se comunica por medio del lenguaje, que es un Sistema altamente desarrollado de palabras y símbolos que tienen significado; vive de acuerdo con un sistema económico en el cual los bienes y servicios se intercambian por otros de valor comparable.

Con frecuencia no se advierte; pero un negocio también es un Sistema. Sus partes tienen nombres como mercadotecnia, producción, ventas, investigación, embarque, contabilidad y personal. Estos componentes trabajan todos juntos para crear una utilidad que beneficie a los empleados y a los accionistas de la firma.

Cada una de estas partes es un Sistema en sí mismo. Cuando se comienza a ver lo abundante que son los Sistemas, no nos sorprende darse cuenta que cada Sistema del negocio depende de una o más entidades abstractas llamadas Sistemas de Información.

### 4.3 TIPOS DE SISTEMAS

Por medio de estos Sistemas los datos pasan de una persona o departamento a otro y puede realizarse cualquier cosa, desde comunicaciones entre oficinas y comunicaciones telefónicas, hasta un Sistema de computadora que genere informes periódicos para diferentes usuarios.

Los Sistemas de Información, sirven a todos los sistemas de un negocio. Ellos son el lazo que mantiene unidos a diferentes componentes en forma total, que pueden trabajar de manera efectiva hacia el mismo objetivo.

## 4.2 CARACTERISTICAS DE LOS SISTEMAS.

interactúan con su medio ambiente

El objetivo de un sistema es la razón de su existencia. Para alcanzar sus objetivos, los sistemas interactúan con su medio ambiente, el cual está formado por todos los objetos que se encuentran fuera de las fronteras de los Sistemas. Siempre el "SICOME" un sistema que necesita estar en interrelación con su medio ambiente, la información se procesa de la siguiente manera: Por ejemplo, el sistema de arranque de un automóvil tiene un propósito claramente establecido: encender el combustible que puede transformarse en energía, que también utilizan otros sistemas del automóvil.

ENTRADA

La característica del Sistema de Control Médico "SICOME" es:

- \* Llevar un control del expediente médico de sus pacientes y que también sea utilizado en otros consultorios médicos sin importar su especialidad.

Información.

Registrado

Decisiones.

Recuperado.

Procesado.

## 4.3 TIPOS DE SISTEMAS.

Los sistemas que interactúan con su medio ambiente que reciben entrada y producen salida son Sistemas Abiertos, en contraste con los sistemas que no interactúan con sus alrededores son Sistemas Cerrados.

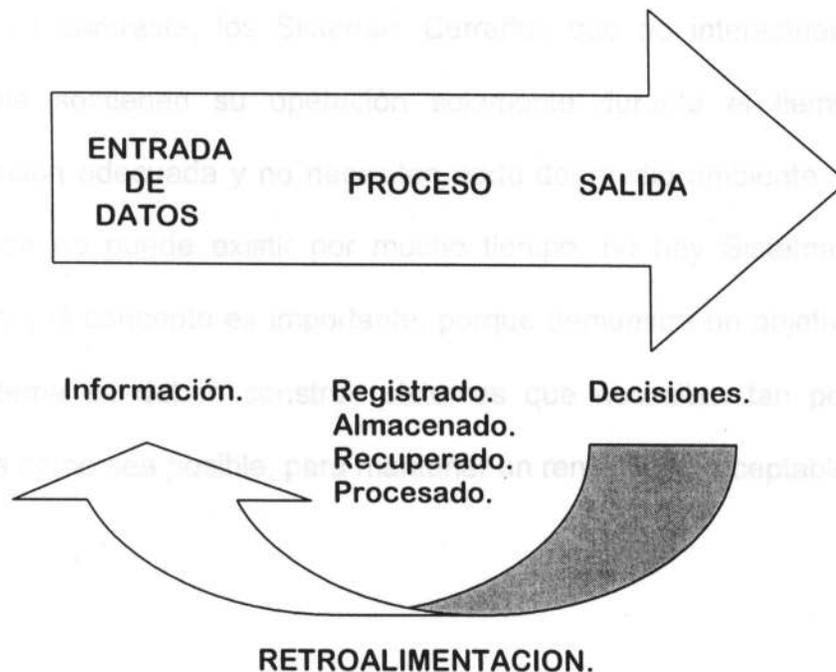
El tratamiento de la información puede ser sencillo, como la realización de una suma, o laborioso como la automatización de su método de cálculo; en ambos

casos la computadora sólo realiza la fase de proceso, es decir, toma los datos de entrada junto con el algoritmo y proporciona un resultado de salida.

Todos los sistemas en marcha son abiertos; por lo tanto, los Sistemas Cerrados existen solamente en forma conceptual. El elemento de control se relaciona con la diferencia entre si los Sistemas son Abiertos o Cerrados.

El "SICOME" es un Sistema Abierto, por que necesita que todas sus partes interactuen con su medio ambiente.

Siendo el "SICOME" un sistema que necesita estar en interrelación con su medio ambiente, la información se procesa de la siguiente manera:



El tratamiento de la información puede ser sencillo, como la realización de una suma, o laborioso como la automatización de su centro de cálculo; en ambos casos la computadora sólo realiza la fase de proceso, es decir, toma los datos de entrada junto con el algoritmo y proporciona un resultado de salida.

Los Sistemas que pueden ajustar sus actividades a niveles aceptables continúan funcionando; los que no pueden hacerlo, se detienen. El concepto de interacción dentro de un medio ambiente que caracteriza a los Sistemas Abiertos es esencial para el control. Por medio de la recepción de la entrada y la evaluación de la misma, un Sistema puede determinar qué tan bien está operando.

En contraste, los Sistemas Cerrados que no interactúan con el medio ambiente sostienen su operación solamente durante el tiempo que tengan información adecuada y no necesiten nada del medio ambiente. Dado que esta condición no puede existir por mucho tiempo, no hay Sistemas Cerrados; sin embargo, el concepto es importante, porque demuestra un objetivo del diseño de un sistema se deben construir sistemas que necesiten tan poca intervención externa como sea posible, para mantener un rendimiento aceptable.

1. Investigación preliminar.
2. Diseño del sistema.
3. Desarrollo de software.
4. Prueba de los sistemas.
5. Puesta en marcha.

## 5.1 INVESTIGACION PRELIMINAR.

¿Cuántas veces se está en situaciones en donde una pregunta si no existe una mejor manera de hacer algo? Una compañía en crecimiento, puede contener a los sistemas de información computarizados como una forma de crecer continuamente, sin tener dificultades en varios procesos, como por ejemplo en el proceso de pedidos de clientes.

## 5

# CICLO DE VIDA DEL DESARROLLO DE SISTEMAS.

El Ciclo de Vida del desarrollo de sistemas es el conjunto de actividades de los analistas, diseñadores y usuarios, que necesitan llevarse a cabo para desarrollar y poner en marcha un sistema de información.

El Ciclo de Vida del desarrollo de sistemas consiste en las siguientes actividades:

1. Investigación preliminar.
2. Diseño del sistema.
3. Desarrollo de software.
4. Prueba de los sistemas.
5. Puesta en marcha.

## 5.1 INVESTIGACION PRELIMINAR.

¿Cuántas veces se está en situaciones en donde se pregunta si no existe una mejor manera de hacer algo? Una compañía en crecimiento, puede contemplar a los sistemas de información computarizados como una forma de crecer continuamente, sin tener dificultades en varios procesos, como por ejemplo en el proceso de pedidos de clientes.

Una petición se puede iniciar por varias razones, pero la clave es que alguna persona de la empresa, ya sea un gerente, un empleado o un especialista de sistemas, inicie el requerimiento de un sistema de información.

Cuando esto se realiza, empieza la primera etapa del ciclo de vida del un sistema:

La Investigación preliminar.

Esta primera etapa del ciclo se divide en tres actividades:

#### **1.- CLARIFICACION DE REQUERIMIENTOS:**

El analista debe de observar en forma objetiva lo que ocurre en la empresa, ya que muchas veces los requerimientos no están claramente establecidos, por lo que, el proyecto requerido debe examinarse para determinar precisamente lo que desea la empresa.

Se realizo una entrevista con el Doctor y sé cuestionaron todas sus necesidades que se presentaban en el consultorio. Se estuvo observando por un tiempo y realizando por escrito los pasos y la metodología que se realizaba con anterioridad.

## 2.- ESTUDIO DE FACTIBILIDAD:

Es determinar si el proyecto solicitado es factible. Los aspectos para determinar la factibilidad del proyecto son:

- a) **Factibilidad Técnica:** Se debe de investigar si se puede realizar el trabajo para el proyecto con el equipo actual, el personal y el software disponible.

Se observo que cuentan con el equipo y el software adecuado a los requerimientos del Sistema.

Cuenta con una computadora Pentium, con las siguientes características y Software precargado:

- \* Pc PENTIUM II 350 Mhz. (Olivetti-45)
- \* 64 MB de Memoria.
- \* Windows 95
- \* Office 97 PRO
- \* Impresora Epson 440 Stylus.

- b) **Factibilidad Económica:** ¿Qué beneficios tendrá la creación del Sistema en cuanto a costo - beneficios?

La realización del Sistema "SICOME" beneficiara al control del manejo de los expedientes de todos sus pacientes. Resultando con esto una disminución en cuanto al gasto generado por la papelería, y mejorando lo administrativo del consultorio.

c) **Factibilidad Operativa:** Se debe de investigar si el sistema que se desarrolla se pondrá en marcha, si habrá resistencia de los usuarios en cuanto a este.

No se presento ninguna resistencia debido a lo sencillo del Sistema "SICOME".

### 3.- APROBACION DEL REQUERIMIENTO:

En muchas empresas tienen varios proyectos que se encuentran en marcha, por lo que la gerencia debe de decidir que proyectos son más importantes y entonces se programan.

Posteriormente, cuando se terminan dichos proyectos, puede iniciarse el desarrollo de la aplicación propuesta.

El resultado de estas actividades será aprobar el requerimiento para una atención posterior o rechazarlo como no factible.

Se observo que el sistema propuesto cubrirá con todas las necesidades señaladas por el Médico encargado del consultorio, siendo la aprobación inmediata para el desarrollo del sistema propuesto, por su factibilidad.

## 5.2 DISEÑO DEL SISTEMA.

El Diseño de un Sistema de información produce los elementos que establecen los requerimientos identificados durante el Análisis del Sistema. A esta etapa se le conoce también con el nombre de Diseño Lógico.

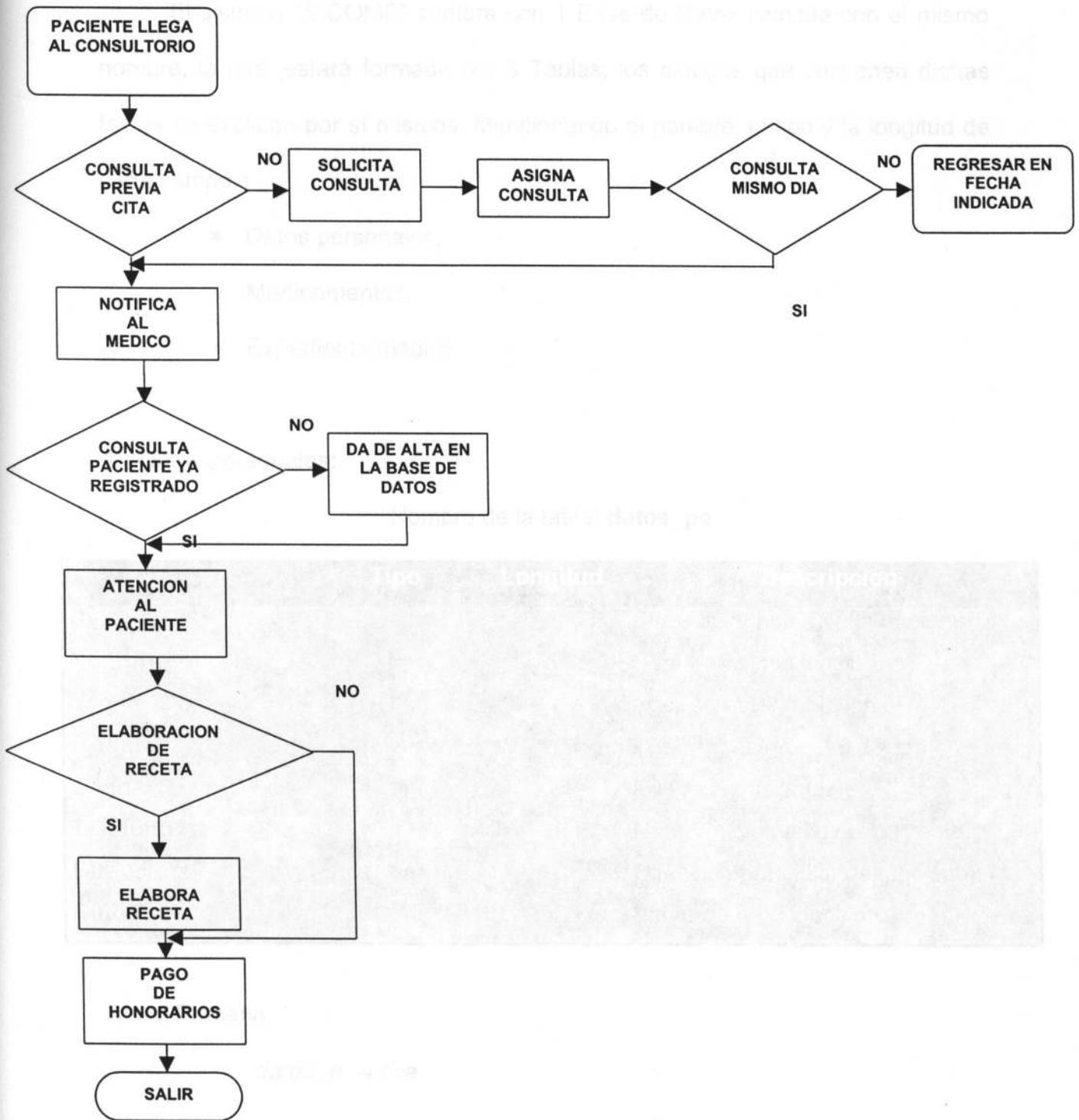
El primer paso en el Diseño de Sistemas es identificar los informes y las salidas que el Sistema producirá; a continuación los datos específicos de cada uno de éstos se señalan incluyendo su localización exacta sobre el papel, la pantalla de despliegue o cualquier otro medio.

El diseño también describe los datos calculados o almacenados que se introducirán a la Base de Datos. Los datos y los procedimientos de cálculo se describen con detalle.

Se seleccionan las estructuras de los archivos y los dispositivos de almacenamiento, como son discos o cintas magnéticas o papel.

Los procedimientos deben de mostrar cómo se van a procesar los datos y cuales van a ser las salidas. Los documentos que contiene las especificaciones del diseño se pueden representar por medio de los diagramas, tablas y símbolos especiales. Como a continuación se muestra:

## DIAGRAMA DE FLUJO DEL "SICOME"



## 5.2.1 DESCRIPCION GENERAL DE LA BASE DE DATOS.

El sistema "SICOME" contara con 1 Base de Datos llamada con el mismo nombre, la cual estará formada por 3 Tablas; los campos que contienen dichas tablas se explican por sí mismos. Mencionando el nombre, el tipo y la longitud de cada campo.

- \* Datos personales.
- \* Medicamentos.
- \* Expediente medico.

### 1. Datos personales:

Nombre de la tabla: **datos\_pe**

Nombre	Tipo	Longitud	Descripción
Cve	Num	4	Clave asignada a cada paciente.
Nombre	Char	20	Nombre del paciente.
a_pat	Char	20	Apellido paterno del paciente.
a_mat	Char	20	Apellido materno del paciente.
Edad	Num	2	Edad del paciente.
Teléfono	Char	20	Número telefónico del paciente.
Dirección	Char	20	Dirección particular del paciente.
Ciudad	Char	15	Lugar de procedencia.

Llave Primaria:

*datos\_p* → *cve*

Llave Alterna:

*Orden* → *A\_Pat* + *A\_Mat* + *Nombre*

## 2. Medicamentos:

Nombre de la tabla: **medicam**

Nombre	Tipo	Longitud	Descripción
<b>cve_m</b>	Num	5	Clave asignada al medicamento.
<b>n_medicina</b>	Char	15	Nombre del medicamento.
<b>Laboratorio</b>	Char	15	Nombre del laboratorio.
<b>Present</b>	Char	10	Presentación del medicamento.
<b>Descrip</b>	Mem	4	Descripción del medicamento.
<b>Coment</b>	Mem	4	Observaciones del medicamento.

Llave Primaria:

*medicam* → *Cve\_m*

## 3. Expediente medico:

Nombre de la tabla: **exp\_medi**

Nombre	Tipo	Longitud	Descripción
<b>Cve</b>	Num	4	Clave asignada a cada paciente.
<b>Fecha</b>	Date	8	Fecha de la consulta.
<b>Num_cons</b>	Num	5	Número de la consulta
<b>Coment</b>	Mem	4	Cuadro médico del paciente.
<b>cve_m</b>	Num	5	Clave asignada al medicamento.

Llave Primaria:

*exp\_medi* → *dtos(fecha) + str(cve,4)*

## **5.3 DESARROLLO DE SOFTWARE.**

De acuerdo al estudio del sistema desarrollado se puede decir que un Lenguaje Orientado a Objetos, mejora la programación en el sentido de que proporciona herramientas que reflejan con mayor exactitud la forma en que concebimos el mundo real. Con la programación basada en objetos es posible reproducir la complejidad de los sistemas del mundo real y desarrollar aplicaciones a gran escala de una forma más eficaz.

En la actualidad es muy importante saber sobre objetos, ya que de acuerdo al gran avance que esta teniendo todo lo que se maneja a través de la computadora es necesario tener conocimiento del Análisis, Diseño y Desarrollo a través de los objetos y la gran cantidad de aplicaciones que se pueden desarrollar gracias a ellos.

### **5.3.1 PROGRAMACION ORIENTADA A OBJETOS.**

"La Programación Orientada a Objetos es un método de implementación en donde los programas se organizan como una colección cooperativa de objetos, cada uno de los cuales representa una instancia de alguna clase, y cuyas clases pertenecen a una jerarquía de clases unidas a través de una relación de herencia."

Un Lenguaje Orientado a Objetos debe soportar y encausar la Programación Orientada a Objetos. Se debe hacer la distinción entre los lenguajes que promueven la Programación Orientada a Objetos (P.O.O.) y aquellos que simplemente la permiten. En teoría, se puede programar Orientado a Objetos en lenguajes que no fueron originalmente diseñados con ese propósito, pero él hacerlo constituye una labor extraordinaria.

La Programación Orientada a Objetos no es una técnica de codificación como una técnica de empaquetado del código; sino una forma con la que los Programadores de Código pueden encapsular la funcionalidad para suministrar a los que la consumen.

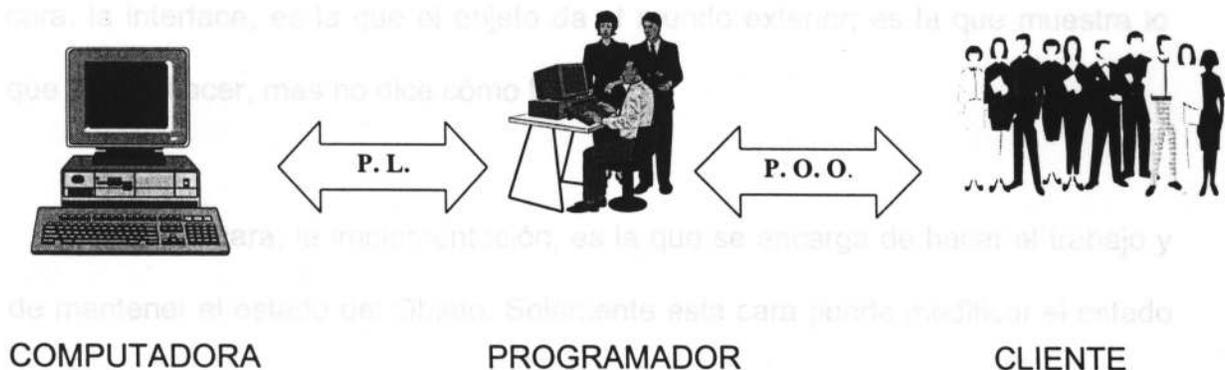
### **VENTAJAS DE LA PROGRAMACION ORIENTADA A OBJETOS.**

La Programación Orientada a Objetos es un entorno de Desarrollo de Sistemas. Ayuda al diseñador a reducir, reutilizar y reciclar código y objetos.

- \* **Objetos reutilizables:** Una vez que se ha desarrollado y depurado una clase, esta puede añadirse a otras aplicaciones sin necesidad de modificarla más.
- \* **Desarrollo más rápido:** Como los objetos son reutilizables, pueden ahorrarse tiempo en tener que escribir largas y aburridas cadenas de código, añadiendo objetos al programa.

- \* Menor código: Como se generan muy pocas líneas de código, existe una menor probabilidad de que se produzcan errores.
- \* Un mantenimiento más fácil y reducido: Como los cambios no son muy necesarios, el mantenimiento de la aplicación se reduce.
- \* Facilidad de modificar objetos: Puede realizarse un cambio en un objeto con un impacto mínimo sobre el resto del sistema.

Lo que separa la Programación Orientada a Objetos de la Programación Lineal (P.L.) es el aumento del énfasis que se hace en la relación entre Clientes y Programadores de Código.



Las herramientas de la Programación Lineal resaltan la relación entre el Programador y su código, mientras que la Programación Orientada a Objetos pone énfasis en la relación entre el Programador y Cliente del Código.

Parece ser universalmente aceptado que con un Lenguaje sea considerado Orientado a Objetos debe contar con por lo menos las siguientes tres propiedades:

- \* Encapsulamiento.
- \* Herencia.
- \* Polimorfismo.

### **ENCAPSULAMIENTO.**

El Encapsulamiento es la propiedad que los Objetos tomaron prestada de sus primos los tipos de datos abstractos. Un Objeto tiene dos caras. La primera cara, la interface, es la que el objeto da al mundo exterior; es la que muestra lo que puede hacer, mas no dice cómo lo hace.

### **POLIMORFISMO**

La otra cara, la implementación, es la que se encarga de hacer el trabajo y de mantener el estado del Objeto. Solamente esta cara puede modificar el estado del Objeto.

## **HERENCIA.**

Es una descripción abstracta de los datos y comportamiento que comparten objetos similares. La Herencia es la manera de establecer relaciones entre las distintas clases que conforman a un sistema Orientado a Objetos.

Debido a que las clases tienen cierto comportamiento que comparten a través de una jerarquía, se logra reducir el código de un programa y se promueve la reutilización.

El atractivo de la Herencia consiste en que al requerir un nuevo componente para un programa en lugar de diseñarlo y construirlo desde cero, se busca una clase que proporcione una funcionalidad lo más parecido a la deseada; una vez encontrada, se crea una nueva clase que herede de ésta, y se agregan y/o modifican únicamente aquellos detalles que sean precisos para obtener el componente que originalmente se estaba necesitando.

## **POLIMORFISMO.**

En un programa con Objetos las acciones ocurren cuando se les mandan mensajes a los Objetos. El Polimorfismo es una característica que permite a distintos Objetos responder al mismo mensaje de manera única. Un nombre de variable puede contener a diferentes momentos referencias a objetos de distintas clases que tienen un mismo antecesor. Cuando esto ocurre, dicho nombre de variable puede responder a un conjunto de operaciones comunes de diferentes maneras.

### 5.3.2 VISUAL FOX PRO.

Aunque Visual Fox Pro sigue permitiendo la programación por procedimientos, se ha ampliado la capacidad del lenguaje para proporcionar la potencia y la flexibilidad propias de la Programación Orientada a Objetos. Con Visual Fox Pro, siendo que es un lenguaje visual te permite desarrollar aplicaciones de una manera más rápida y con un número menor de líneas utilizando la Programación Orientada a Objetos. Visual Fox Pro esta orientado a Objetos y Eventos.

#### OBJETOS.

Estos nos permiten no utilizar demasiado código en nuestros programas. Por medio de estos podemos reutilizar lo que ya esta hecho y facilitar así la programación ya que solamente debemos de pegar los objetos sobre la forma en que estamos trabajando.

#### EVENTOS.

Estos nos permiten ejecutar cierto código en el momento en que algo le suceda a los objetos, es decir, que haya un evento sobre los objetos existentes en nuestro programa. Ej. Si se da click sobre un objeto queremos que mande un mensaje diciendo "Diste click sobre el objeto X".

A continuación se muestra el Sistema "SICOME" una vez ya terminado y listo para la siguiente etapa del Análisis del Sistema

El sistema "SICOME" esta compuesto de las siguientes pantallas.

### **Pantalla Principal.**

Al inicio del sistema se presentara la siguiente pantalla donde nos presenta el nombre del sistema, y las opciones de:



- ◆ Entrar al sistema: Opción de acceso al sistema "SICOME".
- ◆ Salir del sistema: Opción de salida del sistema "SICOME".

## Pantalla de Pacientes.

En la pantalla de pacientes, nos permite manipular los datos del paciente y se presentan las siguientes opciones.

The screenshot shows a window titled 'PACIENTES' with a close button in the top right corner. On the left, there is a medical bag icon. The main area contains a form with the following fields and values:

Clave	2
Nombre	JUAN
Apellido Paterno	SALGADO
Apellido Materno	RUIZ
Edad	56
Telefono	12-34-89
Dirección	SOLARES # 45
Ciudad	MORELIA

On the right side of the window, there is a vertical menu of buttons:

- ALTAS
- BAJAS
- MODIFICA
- CONSULTA
- ANT. (with a left arrow)
- SIG. (with a right arrow)
- PANTALLA OPCIONES

- ◆ Altas: Nos permite dar de alta un nuevo paciente.
- ◆ Bajas: Nos permite dar de baja a un paciente.
- ◆ Consulta: Nos muestra una lista general de todos los pacientes.
- ◆ Modifica: Nos permite realizar cambios en los pacientes.
- ◆ Anterior: Nos permite visualizar los datos del paciente anterior.
- ◆ Siguiente: Nos permite visualizar los datos del siguiente paciente.
- ◆ Pantalla de Opciones: nos regresa a la pantalla de opciones del sistema "SICOME".



## Pantalla de Medicamentos.

Permite manipular los datos de los medicamentos y presenta las siguientes opciones.

- ◆ Altas: Nos permite dar de alta un nuevo medicamento en el catalogo.
- ◆ Bajas: Nos permite dar de baja un medicamento del catálogo.
- ◆ Consulta: Nos muestra una lista general de todos los medicamentos en el catalogo.
- ◆ Modifica: Nos permite realizar cambios en los medicamentos.
- ◆ Anterior: Nos permite visualizar los datos del medicamento anterior.
- ◆ Siguiente. Nos permite visualizar los datos del siguiente medicamento.
- ◆ Pantalla de Opciones: Nos regresa a la pantalla de Opciones del Sistema "SICOME".

## Pantalla de Medicamentos.

Permite manipular los datos de los medicamentos y presenta las siguientes opciones.

- ◆ Altas: Nos permite dar de alta un nuevo medicamento en el catalogo.
- ◆ Bajas: Nos permite dar de baja un medicamento del catálogo.
- ◆ Consulta: Nos muestra una lista general de todos los medicamentos en el catalogo.
- ◆ Modifica: Nos permite realizar cambios en los medicamentos.
- ◆ Anterior: Nos permite visualizar los datos del medicamento anterior.
- ◆ Siguiente. Nos permite visualizar los datos del siguiente medicamento.
- ◆ Pantalla de Opciones: Nos regresa a la pantalla de Opciones del Sistema "SICOME".

## Pantalla de Expediente Medico.

Permite hacer la manipulación de consultas del paciente.

**EXPEDIENTE MEDICO**

Fecha

Consulta #

**Observaciones**

Clave

CLAVE	MEDICAMENTO	DESCRIPCION

ALTAS  
CONSULTA  
MODIFICA  
RECETA  
ANT. SIG.  
PANTALLA OPCIONES

- ◆ Altas: Nos permite dar de alta el examen médico realizado al paciente.
- ◆ Consulta: Nos muestra una lista general de todos los exámenes realizados por paciente.
- ◆ Modifica: Nos permite realizar cambios en el expediente médico.
- ◆ Anterior: Nos permite visualizar los datos del expediente anterior.
- ◆ Siguiente. Nos permite visualizar los datos del siguiente expediente.
- ◆ Receta: Nos permite ir al formato de la receta para despues mandarla a imprimir.
- ◆ Pantalla de Opciones: Nos regresa a la pantalla de Opciones del Sistema "SICOME".

## Pantalla de Mantenimiento.

Permite realizar un mantenimiento en la Base de Datos.



- ◆ Ordenar: Procedimiento para la regeneración de índices en los archivos de datos.
- ◆ Respaldar: Procedimiento para Respaldo de los archivos de datos en unidades de disco flexible.
- ◆ Regresar a Opciones: Nos regresa a la pantalla de Opciones del Sistema "SICOME".

### 5.3.3 BASE DE DATOS.

Visual Fox Pro utiliza una mejor manera de controlar sus Bases de Datos y las relaciones entre las tablas que utiliza, siendo una Base de Datos un contenedor de tablas asociadas a un mismo elemento.

#### DEFINICION.

Es un grupo de datos agrupados, contenidos en un lugar, y que tienen una relación entre sí. Tienen extensión .DBC y pueden contener una o más Tablas.

Ej. Una base de datos que contenga la información del consultorio.

#### TABLA.

Es una sub-agrupación de los datos. Es un nivel en que se clasifican los datos de acuerdo a su contenido. Almacena datos en Registros(filas) y Campos(columnas). Cada Tabla puede estar almacenada en su propio archivo, con una extensión .DBF.

Ej. Tabla de Medicamentos, que contiene los datos de los medicamentos.

Las tablas se dividen en Campos y Registros.

#### REGISTRO.

Cada registro contiene la información de un determinado individuo de la tabla. Ej. Todos los datos de un medicamento.

**CAMPOS.**

Son los datos en sí. Una tabla tiene varios campos, cada registro tiene ese mismo número de campos.

**Ejemplo:**

A continuación se presenta una tabla de medicamentos, para conocer de una forma más fácil las definiciones señaladas con anterioridad.

**TABLA DE MEDICAMENTOS.**

	CAMPO 1	CAMPO 2	CAMPO 3	CAMPO 4
Registro 1	Penicilina G	Feb. 2000	Bayer.	Tabletas.
Registro 2	Ampicilina	Mar. 1999	Pfizer	Gotas.
Registro 3	Felineprina	Sep. 2001	Cima	Gotas.
Registro 4	Dicloxacilina.	Ene. 2002	Hoechst	Cápsulas.
Registro 5				

Suponiendo lo siguiente:

Campo 1 = Nombre del Medicamento.

Campo 2 = Fecha de Caducidad.

Campo 3 = Nombre del Laboratorio.

Campo 4 = Presentación.

5.4 PROBLEMAS DEL SISTEMA

Como podemos ver, en nuestra tabla de Medicamentos tenemos datos de alta 4 medicamentos en los registros 1, 2, 3 y 4, el registro 5 está vacío. Por lo tanto podemos concluir que las tablas tienen muchos registros y todos los registros se componen del mismo número de campos.

## 5.4 PRUEBA DEL SISTEMA.

En esta etapa el Sistema utilizado en forma experimental se pone a prueba durante un período determinado, para asegurar que el software no falle, es decir, que trabaje de acuerdo a las especificaciones y de la manera en la que el usuario espera obtener su información.

En la Prueba del Sistema se examinan los datos de entrada de procesamiento y los resultados para localizar algunos problemas inesperados.

Es preferible detectar cualquier falla o anomalía antes de que la empresa ponga en marcha el nuevo sistema. La Prueba debe ser realizada por personas diferentes a aquellas que desarrollaron el sistema (programadores), ya que de esta manera se asegura una mayor confiabilidad, lo que origina un software de mayor calidad.

El "SICOME" Se puso a prueba durante 3 semanas. Resultando satisfactorio en su proceso de Prueba.

El Sistema "SICOME" se puso en prueba a partir de la segunda semana del mes de Marzo, completando con así el Ciclo de Vida del Desarrollo de un Sistema.

## 5.5 PUESTA EN MARCHA.

Esta etapa es la última del Ciclo de Vida del desarrollo de sistemas, pero no es el fin del sistema. Cuando el personal de sistemas verifica y pone en uso el equipo nuevo, se instala la nueva aplicación, se entrena al personal que manejará el sistema y construyen los archivos de datos que se necesiten.

Cuando estas actividades terminan, entonces se dice que el sistema está puesto en marcha.

Los desarrolladores del sistema pueden escoger una parte (un área o departamento) de la empresa para probar el nuevo sistema con sólo una o dos personas; a su vez este puede estar trabajando en forma paralela con el sistema anterior para comparar resultados (beneficios).

Una vez instalado, la aplicación se utilizará por muchos años, sin embargo, las empresas, el personal y el medio ambiente cambiarán a través del tiempo. La Puesta en Marcha es un proceso continuo.

El Sistema "SICOME" se puso en marcha a partir de la segunda semana del mes de Marzo, completando con así el Ciclo de Vida del Desarrollo de un Sistema.

## **CAPACITACION.**

Dar los conocimientos necesarios a todo el personal que vaya a hacer uso del sistema "SICOME", para su adecuado funcionamiento.

## **6.1 CONCLUSIONES.**

Se les dio capacitación durante dos semanas, obteniendo rápidos resultados por la facilidad con la que resulto el manejo del Sistema "SICOME".

## **DOCUMENTACION.**

Mantener por escrito (o por otro medio), la información completa de las funciones que realiza cada proceso del Sistema "SICOME".

Se entrego un Manual del Usuario donde se detallan las funciones del "SICOME", para posibles consultas.

## CONCLUSIONES Y RECOMENDACIONES.

### 6.1 CONCLUSIONES.

- \* El "SICOME" es sencillo y explícito, logrando un manejo fácil de comprender y de aprender.
  
- \* El "SICOME" no solo beneficiara en el área administrativa, sino que además redundara en la satisfacción del Médico.
  
- \* El "SICOME" nos permite agilizar los tiempos de búsqueda de los expedientes y de los medicamentos.

Es indudable que el avance de la tecnología, al servicio del hombre mejorara el nivel de la administración y vida de la sociedad, por ello debemos ir a la vanguardia para en un futuro estar a nivel del primer mundo.

## 6.2 RECOMENDACIONES. BIBLIOGRAFIA

### ANALISIS Y DISEÑO DE INFORMACION

- \* Actualizar los datos de los pacientes si se requiere.

James A. Shea.

Editorial Mc Graw-Hill.

- \* Mantener actualizado el catálogo de medicinas que existen actualmente.

Segunda Edición.

1992.

- \* Realizar respaldos semanales de la información.

### ANALISIS Y DISEÑO ORIENTADO A OBJETOS.

Garner Martin / James J. Odell.

Editorial Prentice-hall.

Primera Edición.

1994.

### INFORMÁTICA PRESENTE Y FUTURO

H. Sanders, Donald.

Editorial Mc Graw-Hill.

Tercera Edición.

1990.

### INTRODUCCION A LA COMPUTACION

Guillermo Lavín Gutierrez.

Editorial Mc Graw-Hill.

Tercera Edición.

1991.

INTRODUCCION A LA INFO **BIBLIOGRAFIA.**

Dr. Duffy.

**ANALISIS Y DISEÑO DE INFORMACION.**

Editorial Iberoamericana.

James A. Sean.

Segunda Edición.

Editorial Mc Graw~Hill.

1993.

Segunda Edición.

1992.

MANUAL DEL PROGRAMADOR.

Microsoft Visual Fox Pro.

**ANALISIS Y DISEÑO ORIENTADO A OBJETOS.**

For Visual 3.0

James Martin / James J. Odell.

Editorial Prentice~hall.

Primera Edición.

PROGRAMACION ORIENTADA A OBJETOS

1994.

Editorial Addison-Wesley / Diaz de Santos.

Segunda Edición.

**INFORMATICA PRESENTE Y FUTURO.**

1988

H. Sanders, Donald.

Editorial Mc. Graw~Hill.

VISUAL FOX PRO 5.0

Tercera Edición.

Manual de programación.

1990.

Lee & John Pinter.

Editorial Mc Graw~Hill.

**INTRODUCCION A LA COMPUTACION.**

Primera Edición.

Guillermo Levin Gutiérrez.

Editorial Mc Graw~Hill.

Tercera Edición.

1991.

**INTRODUCCION A LA INFORMATICA.**

**Tim Duffy.**

**Editorial Iberoamericana.**

**Segunda Edición.**

**1993.**

**MANUAL DEL PROGRAMADOR.**

**Microsoft Visual Fox Pro.**

**Versión 3.0**

**1995.**

**PROGRAMACION ORIENTADA A OBJETOS.**

**Editorial Addison-Wesley / Díaz de Santos.**

**Segunda Edición.**

**1986.**

**VISUAL FOX PRO 5.0**

**Manual de programación.**

**Les & John Pinter.**

**Editorial Mc Graw-Hill.**

**Primera Edición.**

**1997.**

## CONSULTA INTERNET.

### ANALISIS DE SISTEMAS DE INFORMACION.

Home.

<http://fipesmi.misiones.org.ar/users/educohen/analisis.htm>

1998.

### OTORRINOLARINGOLOGIA.

J. Aldasoro Martín.

<http://www.cvt.es/USERS/aldasoro/>

1998.

APENDICE.

### PRESENTACION DE VISUAL FOX PRO 5.0

Alberto Rodríguez.

<http://www.fpress.com/revista/num9610/oct96.htm>

1996.

### PROGRAMACION ORIENTADA A OBJETOS

Pedro Ayala Mujica.

Centro de Sistemas e Informática.

[http://www.lasalle.edu.co/~payalam/p\\_objeto.htm](http://www.lasalle.edu.co/~payalam/p_objeto.htm)

1997.

## APENDICE

LOS BOTONES DE:

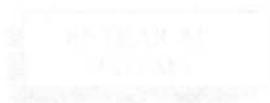


SE HEREDARON LOS PROCEDIMIENTOS Y LAS PANTALLAS QUE LES OTORGAN.

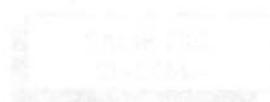
# APENDICE.

\*\*\*\* PANTALLA PRINCIPAL \*\*\*\*

SET DEFAULT TO 'C:\WFPI\FRANCISCO'  
CLOSE TABLES ALL  
CLOSE DATABASE  
SET DELETED ON



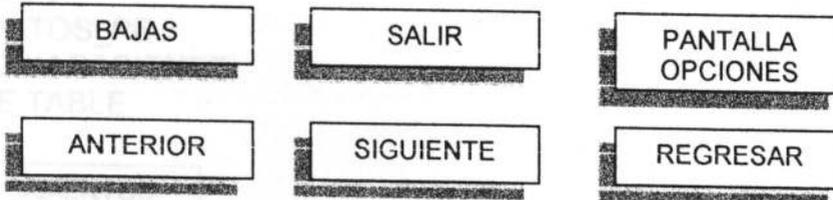
OPCIONES



THISFORM RELEASE

## APENDICE

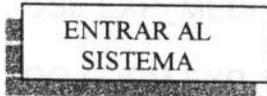
LOS BOTONES DE:



SE HEREDARON LOS PROCEDIMIENTOS A LAS PANTALLAS QUE LOS UTILIZAN.

### \*\*\*\*\* PANTALLA PRINCIPAL \*\*\*\*\*

SET DEFAULT TO "C:\VFP\FRANCISCO"  
CLOSE TABLES ALL  
CLOSE DATABASE  
SET DELETED ON



DO FORM OPCIONES



THISFORM.RELEASE

\*\*\*\*\* PANTALLA OPCIONES \*\*\*\*\*

PACIENTES

```
SELECT 1
USE DATOS_PE
DO FORM PACIENTE
CLOSE TABLE
```

MEDICAMENTOS

```
SELECT 1
USE MEDICAM
DO FORM MEDICAM
CLOSE TABLE
```

EXPEDIENTES

```
SELECT 1
USE DATOS_PE

SELECT 2
USE MEDICAM

SELECT 3
USE EXP_MEDI

DO FORM EXP_MEDI

CLOSE TABLES ALL
```

MANTENIMIENTO

```
DO FORM MANTEN
```

PRINCIPAL

```
THISFORM.RELEASE
```

\* SE LE PASAN LOS \*\*\*\*\* PANTALLA PACIENTES \*\*\*\*\* LAS DE TEXTO

\* SE HABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBCLAVE.VALUE = VCLAVE
THISFORM.TBCLAVE.ENABLED = .F.
THISFORM.TBNOMBRE.ENABLED = .F.
THISFORM.TBA_PAT.ENABLED = .F.
THISFORM.TBA_MAT.ENABLED = .F.
THISFORM.TBEDAD.ENABLED = .F.
THISFORM.TBTELEFONO.ENABLED = .F.
THISFORM.TBDIRECCION.ENABLED = .F.
THISFORM.TBCIUDAD.ENABLED = .F.
```

THISFORM.TBCLAVE.SETFOCUS

\* SE LE PASAN LOS VALORES EN BLANCO A LAS CAJAS DE TEXTO

```
THISFORM.TBCLAVE.VALUE = CVE
THISFORM.TBNOMBRE.VALUE = NOMBRE
THISFORM.TBA_PAT.VALUE = A_PAT
THISFORM.TBA_MAT.VALUE = A_MAT
THISFORM.TBEDAD.VALUE = EDAD
THISFORM.TBTELEFONO.VALUE = TELEFONO
THISFORM.TBDIRECCION.VALUE = DIRECCION
THISFORM.TBCIUDAD.VALUE = CIUDAD
```

THISFORM.TBCLAVE.SETFOCUS

ENDIF

THISFORM.REFRESH()

ALTAS

LOCAL VCLAVE

• SE HABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBCLAVE.ENABLED = .T.
THISFORM.TBNOMBRE.ENABLED = .T.
THISFORM.TBA_PAT.ENABLED = .T.
THISFORM.TBA_MAT.ENABLED = .T.
THISFORM.TBEDAD.ENABLED = .T.
THISFORM.TBTELEFONO.ENABLED = .T.
THISFORM.TBDIRECCION.ENABLED = .T.
THISFORM.TBCIUDAD.ENABLED = .T.
GO BOTTOM
VCLAVE = CVE + 1
```

- SE LE PASAN LOS VALORES EN BLANCO A LAS CAJAS DE TEXTO

```

APPEND BLANK
THISFORM.TBCLAVE.VALUE = VCLAVE
THISFORM.TBNOMBRE.VALUE = NOMBRE
THISFORM.TBA_PAT.VALUE = A_PAT
THISFORM.TBA_MAT.VALUE = A_MAT
THISFORM.TBEDAD.VALUE = EDAD
THISFORM.TBTELEFONO.VALUE = TELEFONO
THISFORM.TBDIRECCION.VALUE = DIRECCION
THISFORM.TBCIUDAD.VALUE = CIUDAD
THISFORM.TBCLAVE.SETFOCUS
THISFORM.TBCLAVE.SETFOCUS

```

BAJAS

```

IF MESSAGEBOX("DESEAS DAR DE BAJA ESTE PACIENTE",4+32)=6
  DELETE
  =MESSAGEBOX ("BAJA EXITOSA")
  SKIP - 1
  IF BOF()
    GO TOP
  ENDIF
ENDIF
THISFORM.REFRESH()

```

MODIFICA

- \* SE INHABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```

THISFORM.TBCLAVE.ENABLED = .F.
THISFORM.TBNOMBRE.ENABLED = .T.
THISFORM.TBA_PAT.ENABLED = .T.
THISFORM.TBA_MAT.ENABLED = .T.
THISFORM.TBEDAD.ENABLED = .T.
THISFORM.TBTELEFONO.ENABLED = .T.
THISFORM.TBDIRECCION.ENABLED = .T.
THISFORM.TBCIUDAD.ENABLED = .T.

```

PANTALLA DE OPCIONES

THISFORM.RELEASE

SE LE PASAN LOS VALORES DE LOS CAMPOS A LAS CAJAS DE TEXTO

```
THISFORM.TBCLAVE.VALUE = CVE
THISFORM.TBNOMBRE.VALUE = NOMBRE
THISFORM.TBA_PAT.VALUE = A_PAT
THISFORM.TBA_MAT.VALUE = A_MAT
THISFORM.TBEDAD.VALUE = EDAD
THISFORM.TBTELEFONO.VALUE = TELEFONO
THISFORM.TBDIRECCION.VALUE = DIRECCION
THISFORM.TBCIUDAD.VALUE = CIUDAD
```

THISFORM.TBCLAVE.SETFOCUS

CONSULTAS

```
DO FORM CON_PAC
THISFORM.REFRESH()
```

ANTERIOR

```
SKIP - 1
IF BOF()
  GO TOP
  =MESSAGEBOX("PRIMER PACIENTE")
  ENDIF
THISFORM.REFRESH()
```

SIGUIENTE

```
SKIP
IF EOF()
  SKIP -1
  =MESSAGEBOX("ULTIMO PACIENTE")
  ENDIF
THISFORM.REFRESH()
```

PANTALLA DE  
OPCIONES

THISFORM.RELEASE

## \*\*\*\*\* PANTALLA DE MEDICAMENTOS \*\*\*\*\*

\* SE INHABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.REFRESH()
THISFORM.TBCVE_M.ENABLED = .F.
THISFORM.TBN_MÉDICINA.ENABLED = .F.
THISFORM.TBLABORATORIO.ENABLED = .F.
THISFORM.TBPRESNT.ENABLED = .F.
THISFORM.EDESCRIP.ENABLED = .F.
THISFORM.ECOMENT.ENABLED = .F.
```

• SE LE PASAN LOS VALORES DE LOS CAMPOS A LAS CAJAS DE TEXTO

```
THISFORM.TBCVE_M.VALUE = CVE_M
THISFORM.TBN_MÉDICINA.VALUE = N_MÉDICINA
THISFORM.TBLABORATORIO.VALUE = LABORATORIO
THISFORM.TBPRESNT.VALUE = PRESENT
THISFORM.EDESCRIP.VALUE = DESCRIP
THISFORM.ECOMENT.VALUE = COMENT
```

ALTAS

LOCAL VCLAVE

\* SE HABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBCVE_M.ENABLED = .T.
THISFORM.TBN_MÉDICINA.ENABLED = .T.
THISFORM.TBLABORATORIO.ENABLED = .T.
THISFORM.TBPRESNT.ENABLED = .T.
THISFORM.EDESCRIP.ENABLED = .T.
THISFORM.ECOMENT.ENABLED = .T.
GO BOTTOM
VCLAVE = CVE_M + 1
```

• SE LE PASAN LOS VALORES EN BLANCO A LAS CAJAS DE TEXTO

APPEND BLANK

```
THISFORM.TBCVE_M.VALUE = VCLAVE
THISFORM.TBN_MÉDICINA.VALUE = N_MÉDICINA
THISFORM.TBLABORATORIO.VALUE = LABORATORIO
THISFORM.TBPRESNT.VALUE = PRESENT
THISFORM.EDESCRIP.VALUE = DESCRIP
THISFORM.ECOMENT.VALUE = COMENT
THISFORM.TBCVE_M.SETFOCUS
```

CONSULTAS

DO FORM CON\_MED  
THISFORM.REFRESH()

MODIFICA

\* SE INHABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

THISFORM.TBCVE\_M.ENABLED = .F.  
THISFORM.TBN\_MEDICINA.ENABLED = .T.  
THISFORM.TBLABORATORIO.ENABLED = .T.  
THISFORM.TBPRESNET.ENABLED = .T.  
THISFORM.EDESCRIP.ENABLED = .T.  
THISFORM.ECOMENT.ENABLED = .T.

• SE LE PASAN LOS VALORES DE LOS CAMPOS A LAS CAJAS DE TEXTO

THISFORM.TBCVE\_M.VALUE = CVE\_M  
THISFORM.TBN\_MEDICINA.VALUE = N\_MEDICINA  
THISFORM.TBLABORATORIO.VALUE = LABORATORIO  
THISFORM.TBPRESNET.VALUE = PRESENT  
THISFORM.EDESCRIP.VALUE = DESCRIP  
THISFORM.ECOMENT.VALUE = COMENT  
THISFORM.TBN\_MEDICINA.SETFOCUS

DFEC = DATE()

• SE LE PASAN LOS VALORES EN BLANCO A LAS CAJAS DE TEXTO

APPEND BLANK  
THISFORM.TBFECHA.VALUE = DFEC  
THISFORM.TBCONS.VALUE = NUM\_CONS  
THISFORM.ECOMENT.VALUE = COMENT  
THISFORM.TBCLAVE.VALUE = CVE  
THISFORM.TBFECHA.SETFOCUS

OPERACIONES

DO FORM DESPQRS  
THISFORM.PREFECHO

\*\*\*\*\* PANTALLA DE EXPEDIENTES \*\*\*\*\*

\* SE INHABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBFECHA.ENABLED = .F.  
THISFORM.TBCONS.ENABLED = .F.  
THISFORM.ECOMENT.ENABLED = .F.  
THISFORM.TBCLAVE.ENABLED = .F.
```

\* SE LE PASAN LOS VALORES DE LOS CAMPOS A LAS CAJAS DE TEXTO

```
THISFORM.TBFECHA.VALUE = FECHA  
THISFORM.TBCONS.VALUE = NUM_CONS  
THISFORM.ECOMENT.VALUE = COMENT  
THISFORM.TBCLAVE.VALUE = CVE  
THISFORM.TBFECHA.SETFOCUS
```

ALTAS

LOCAL DFEC

\* SE HABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBFECHA.ENABLED = .T.  
THISFORM.TBCONS.ENABLED = .T.  
THISFORM.ECOMENT.ENABLED = .T.  
THISFORM.TBCLAVE.ENABLED = .T.
```

DFEC = DATE()

• SE LE PASAN LOS VALORES EN BLANCO A LAS CAJAS DE TEXTO

• APPEND BLANK

```
THISFORM.TBFECHA.VALUE = DFEC  
THISFORM.TBCONS.VALUE = NUM_CONS  
THISFORM.ECOMENT.VALUE = COMENT  
THISFORM.TBCLAVE.VALUE = CVE  
THISFORM.TBFECHA.SETFOCUS
```

CONSULTAS

```
DO FORM LISTCONS  
THISFORM.REFRESH()
```

MODIFICA

\* SE INHABILITAN LAS CAJAS DE TEXTO PARA QUE NO PUEDAN EDITAR

```
THISFORM.TBFECHA.ENABLED = .F.  
THISFORM.TBCONS.ENABLED = .T.  
THISFORM.ECOMENT.ENABLED = .T.  
THISFORM.TBCLAVE.ENABLED = .T.
```

\* SE LE PASAN LOS VALORES DE LOS CAMPOS A LAS CAJAS DE TEXTO

```
THISFORM.TBFECHA.VALUE = FECHA  
THISFORM.TBCONS.VALUE = NUM_CONS  
THISFORM.ECOMENT.VALUE = COMENT  
THISFORM.TBCLAVE.VALUE = CVE  
THISFORM.TBFECHA.SETFOCUS
```

RECETA

• IMPRESION DE RECETA

```
SET CONSOLE OFF  
SET PRINT ON  
REPORT FORM RECETA TO PRINT  
SET PRINT OFF  
SET CONSOLE ON
```

```
IF MESSAGEBOX("INSERTAR CDIGO", 1+32) = 1  
    RUN COPY A:\DB  
    RUN COPY A:\DBC  
    RUN COPY A:\FP1  
    MESSAGEBOX("RESPALDO EXITOSO")
```

```
ENDIF  
THISFORM.Refresh()
```

REGISTRAR  
OPCIONES

```
THISFORM.Refresh()
```

\*\*\*\*\* PANTALLA DE MANTENIMIENTO \*\*\*\*\*

ORDENAR

```
CLOSE TABLES ALL
CLOSE DATABASE
=MESSAGEBOX ("INDEXANDO")
```

```
SELECT 1
USE EXP_MEDI
PACK
INDEX ON DTOS(FECHA)+STR(CVE,4) TAG EXP_MED
```

```
USE DATOS_PE
PACK
INDEX ON CVE TAG DATOS_P
```

```
INDEX ON A_PAT + A_MAT + NOMBRE TAG ORDEN
```

```
USE MEDICAM
PACK
INDEX ON CVE_M TAG MEDICAM
```

```
CLOSE TABLES ALL
CLOSE DATABASE
```

RESPALDAR

```
IF MESSAGEBOX("INSERTAR DISCO",1+32)=1
    RUN COPY A:\*.DBF
    RUN COPY A:\*.DBC
    RUN COPY A:\*.FPT
    =MESSAGEBOX ("RESPALDO EXITOSO")
ENDIF
THISFORM.REFRESH()
```

REGRESAR A  
OPCIONES

```
THISFORM.RELEASE()
```