



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Sistema de teleimágenes
clínica: consulta y diagnóstico a
distancia**

TESIS

Que para obtener el título de
Ingeniero en Computación

P R E S E N T A N

Andrea Guadalupe Vargas Grajeda
Luis Antonio Méndez Tierri

DIRECTORA DE TESIS

M.C. Amanda O. Gómez González



Ciudad Universitaria, Cd. Mx., 2003

ÍNDICE

INTRODUCCIÓN	1
I. IMAGENOLOGÍA CLÍNICA	4
1.1. Historia	4
1.2. Misión, estructura y funciones	6
1.2.1. Facultades de la Imagenología	14
II. IDENTIFICACIÓN DE LA PROBLEMÁTICA DE LA IMAGENOLOGIA	15
2.1. Necesidades	15
2.2. Problemática	15
2.3. Organización actual	16
2.4. Objetivo	18
III. FUNDAMENTOS TEÓRICOS	19
3.1. Proceso de desarrollo de un sistema informático	19
3.1.1. Metodologías más utilizadas	20
a) Ciclo de Vida Clásico	21
b) Construcción de Prototipos	22
c) Modelo en Espiral	22
d) Ciclo de Vida Clásico Modificado	23
3.2. Técnicas de desarrollo de sistemas	23
3.2.1. Técnicas estructuradas	24
a) Programación estructurada	25
b) Diseño estructurado	26
c) Análisis estructurado moderno	28
d) Modelización de datos	30
e) Ingeniería de información	32
3.2.2. La técnica del Desarrollo Conjunto de Aplicaciones (DCA)	33

3.2.3. Las técnicas de prototipos y desarrollo rápido	34
3.3 ¿Qué metodología y técnicas elegir?	37
Ciclo de vida clásico	37
Construcción de prototipos	37
Modelo en espiral o evolutivo	37
IV. MÉTODO DE DISEÑO ESTRUCTURADO DE YOURDON	40
4.1. Marco para la modelización de procesos	41
4.1.1. Diagrama de flujo de datos	41
4.1.1.1. Características del Diagrama de Flujo de Datos	42
a) Enfoque descendente	43
b) Cubrir los detalles	43
c) Dibujar de nuevo los diagramas	43
4.1.2. Diccionario de Datos	44
4.2. Propiedades	45
a) Modularidad	45
b) Cohesión	45
c) Acoplamiento	46
4.3. Marco para la modelización de datos	47
4.3.1. Base de datos	47
4.3.2. Tipos de consulta	50
4.3.3. Modelo Entidad-Relación (E-R)	51
4.3.4. Normalización	52
4.4. Etapas del ciclo de vida	53
4.4.1. Estudio de viabilidad	54
4.4.2. Análisis del sistema	54
4.4.3. Diseño del sistema	54
4.4.4. Implementación ó producción	55
4.4.5. Pruebas del sistema	55
a) Pruebas de Unidad	55
b) Pruebas de Integración	56
c) Pruebas de Volumen	56
d) Pruebas de Validación	57
4.4.6. Control de Calidad	57
4.4.7. Mantenimiento	58

V. ANALISIS Y DISEÑO ESTRUCTURADO DE YOURDON	60
5.1. Análisis del sistema	60
5.2. El DFD del Sistema de Teleimagenología Clínica	60
5.3. Diagrama entidad-relación (DER)	64
5.4. Diccionario de datos del Sistema de Teleimagenología Clínica	65
5.5. Diseño de módulos del Sistema de Teleimagenología Clínica	69
VI. ANÁLISIS DE LA TECNOLOGÍA PARA LA IMPLEMENTACIÓN DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA	74
6.1. Servicios que ofrece la red del ISSSTE	74
6.2. El protocolo de comunicación TCP/IP	75
6.2.1. Historia del Protocolo de Comunicación TCP/IP	75
6.2.2. ¿Cómo trabaja el Protocolo TCP/IP?	76
6.2.3. Direcciones IP	77
6.2.4. Principales características del Protocolo TCP/IP	79
6.3. Plataformas de Hardware	79
6.3.1. Procesadores CISC	80
6.3.2. Procesadores RISC	81
6.3.3. Multiproceso	82
6.3.4. Multiprocesamiento Simétrico-Asimétrico	83
6.3.5. El equipo ideal (Cliente)	84
6.3.5.1. Características deseables en los equipos Cliente	85
6.4. Plataformas de Software	89
6.4.1. Sistemas Operativos de Red	90
6.4.1.1. Linux	91
6.4.1.2. Windows NT	92
6.4.2. Sistema Administrador de Base de Datos Relacional (RDBMS)	97
6.4.2.1. Sybase	97
6.4.2.2. MS SQL Server	98
VII. DESARROLLO E IMPLANTACIÓN DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA	102
7.1. Arquitectura Cliente/Servidor	102
7.2. Lenguaje de programación	106
7.2.1. MS Visual Basic 6.0	107
7.2.2. Programación orientada a eventos	107

7.3. Creación de la base de datos para el sistema de Teleimagenología Clínica	110
7.4. Acceso a datos con MS Visual Basic 6.0	115
7.5. Seguridad del Sistema de Teleimagenología Clínica	118
7.6. Distribución del sistema vía FTP	120
7.7. Propuesta de implantación	123
7.8. Alcances	126
VIII. PRUEBAS REALIZADAS AL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA	127
a) Pruebas de Unidad	127
b) Pruebas de Integración	127
c) Prueba de Volumen	128
d) Prueba de Validación	128
CONCLUSIÓN	130
BIBLIOGRAFÍA	131

INTRODUCCIÓN

La imagen es un elemento fundamental en el actual quehacer médico, por su papel en el estudio diagnóstico del paciente. Muchas de las imágenes que se estudian en Medicina proceden de aplicaciones basadas en rayos X. La radiografía simple es la técnica inicial de imagen por excelencia, llegando a ser el primer examen diagnóstico que se realiza después de la historia clínica de la mayoría de pacientes.

La IMAGENOLOGÍA tiene un brillante porvenir, gracias a las posibilidades que ofrece la informática para obtener, conservar y transmitir imágenes.

En el ISSSTE (Instituto de Seguridad y Servicios Sociales de los Trabajadores del Estado) en el área de Imagenología se encontraron una serie de problemas relacionados con la consulta y diagnóstico de los diferentes campos de la Imagenología, cuando estos eran requeridos de una distancia considerable, se detectaron grandes tiempos de espera para obtener el diagnóstico, así como un gran costo de transporte de los derechohabientes y en el envío de los expedientes así como una gran pérdida de los mismos, y una equivocada entrega de expedientes a sus doctores correspondientes debido a que muchos pacientes tienen uno o más doctores.

La propuesta de solución para resolver el problema expuesto, consiste en crear un sistema para computadora el cual maneje toda la información de los derechohabientes que fueron enviados a realizarse un estudio en el área de Imagenología Clínica. Al contar con esta herramienta se facilitarían los diagnósticos, sin importar la distancia en que se encuentren los especialistas, siempre y cuando estén capacitados para el uso de medios de comunicación, deberán de estar debidamente identificados en el sistema para poder apoyar el diagnóstico, indicar tratamiento, mantener en orden las imágenes capturadas y contar a su vez con un documento que capture la causa de la consulta, y la opinión del especialista distante, y que dicho documento pueda imprimirse como anexo al expediente del paciente.

A continuación se expone la estructura general de la tesis que se presenta, titulada: **TELEIMAGENOLOGÍA CLÍNICA: Consulta y Diagnóstico a distancia.**

CAPÍTULO I .- Imagenología Clínica.

En este capítulo se expone, la historia, misión, estructura y funciones, así como las facultades de la IMAGENOLOGÍA.

CAPÍTULO II .- Identificación de la problemática de la Imagenología.

Se exponen las necesidades de la comunidad médica en el área de Imagenología, se plantea la problemática existente y como puede ser atacada, así como el objetivo principal a cumplir con el sistema a realizar.

CAPÍTULO III .- Fundamentos teóricos.

En el se examinan las metodologías y técnicas mas utilizadas para el desarrollo de un sistema informático, sus ventajas y desventajas de cada una de ellas, así como sus relaciones, para así proponer, al final, la metodología y técnica a seguir para el desarrollo del sistema de Teleimagenología Clínica.

CAPÍTULO IV .- Método de diseño estructurado de Yourdon.

En este capítulo se analiza a detalle la metodología estructurada de desarrollo de sistemas de Yourdon, pues es la metodología a seguir para el desarrollo del presente proyecto.

CAPÍTULO V .- Análisis y diseño estructurado de Yourdon.

Se aplica tanto la metodología como la técnica seleccionada (metodología estructurada de Yourdon) para el desarrollo del sistema, por ejemplo: El diagrama de flujo de datos (DFD), el diagrama entidad relación (DER), el diccionario de datos (DD), etc.

CAPÍTULO VI .- Análisis de la tecnología para la implementación del sistema de Teleimagenología Clínica.

Se hace un estudio de la Red del ISSSTE y del protocolo de comunicación estándar que utiliza (TCP/IP). Se analizan el hardware (el equipo Servidor y los equipos Cliente) y el software (Sistema Operativo de Red y el Sistema Administrador de Base de Datos Relacional), sobre los cuales se pondrá en operación el sistema a realizar. Se realiza un estudio comparativo de mercado de las diferentes plataformas de hardware y software, y se propone la opción mas adecuada para la realización del sistema.

CAPÍTULO VII .- Desarrollo e implantación del sistema de Teleimagenología Clínica.

Se trata lo relacionado con la arquitectura Cliente/Servidor, las características y capacidades del lenguaje de programación utilizado para el desarrollo del sistema, seguridad y distribución del mismo, así como también trata sobre la creación de la base de datos y el acceso a los datos de la misma. Por último se da la propuesta de implantación así como los alcances del sistema final.

CAPÍTULO VIII .- Pruebas realizadas al sistema de Teleimagenología Clínica.

Se menciona el tipo de pruebas realizadas al sistema final, como son: Pruebas de unidad, de integración, de volumen y de validación, para así asegurar y garantizar la calidad del mismo.

Finalmente se da la conclusión acerca de la realización del sistema de Teleimagenología Clínica.

I. IMAGENOLOGÍA CLÍNICA

1.1. HISTORIA

Muchas de las imágenes que se estudian en Medicina proceden de aplicaciones basadas en rayos X. Los rayos X fueron descubiertos por W. K. Roentgen en 1895. El 8 de noviembre de 1895 Roentgen estaba trabajando con los tubos Hittorf Crookes y descubre una nueva radiación que atraviesa el papel, la madera, y la tela, no visible, a la cual llama radiación X.

Roentgen se encontraba trabajando los rayos catódicos en un tubo de descarga gaseosa de alto voltaje. A pesar de que el tubo estaba dentro de una caja de cartón negro, Roentgen vio que una pantalla de platinocianuro de bario, que casualmente estaba cerca, emitía luz fluorescente siempre que funcionaba el tubo. Tras realizar experimentos adicionales, determinó que la fluorescencia se debe a una radiación invisible más penetrante que una radiación ultravioleta.

El 22 de diciembre de 1895 obtiene la primera radiografía de la mano de su esposa Bertha, en la cual se observa el anillo que llevaba Bertha en el momento de hacerle la radiografía, y el 28 de diciembre realiza su comunicación a la sociedad Física Médica de Wurzburg Titulada "Sobre una nueva especie de rayos".



Fig. 1.1 Primera Radiografía tomada por Roentgen

El descubrimiento de los rayos X, por el cual Roentgen recibió el premio Nobel en 1901, ha tenido repercusiones en muchas áreas del conocimiento:

- En física de Materia Condensada, la difracción de rayos X por un cristal nos da información de la localización de átomos en ese cristal.
- En Biofísica Molecular, los estudios de difracción de rayos X han ayudado a elucidar la estructura del ADN (Ácido Desoxirribonucleico) y de un gran número de proteínas. El automatizar y digitalizar este proceso ha permitido crear un enorme banco de datos tal, que ahora el crecimiento de mapas de rayos X de estructuras biológicas aumenta a una tasa de centenares de mapas por año, beneficiando directamente el diseño de nuevas medicinas.
- En Medicina, los rayos X son fundamentales en diagnóstico y terapia. La tecnología de tomografías de rayos X computarizada, en constante avance, permite obtener imágenes cada vez mas claras de tumores.

Los rayos X tienen unas propiedades en las que se basa su uso en Medicina: el poder de penetración en la materia, el efecto luminiscente, que consiste en la capacidad de los rayos X que al incidir sobre ciertas sustancias, éstas emitan luz, y el efecto fotográfico, que es la capacidad que tienen de producir el ennegrecimiento de las emulsiones fotográficas, una vez reveladas y fijadas éstas. Esta es la base de la imagen radiológica.

La radiografía simple es la técnica inicial de imagen por excelencia, llegando a ser el primer examen diagnóstico que se realiza después de la historia clínica de la mayoría de pacientes. Sus indicaciones son múltiples: la Rx de tórax ante cualquier síntoma cardiorrespiratorio, la Rx simple de cualquier parte del cuerpo accidentada, la placa simple de abdomen ante molestias del aparato digestivo, la radiografía simple de cráneo en traumatismos craneoencefálicos, etc.

Entre las técnicas de imagen, destaca por su uso prioritario la TAC (Tomografía Axial Computarizada), escáner o TC, basada en la emisión de radiaciones electromagnéticas, característica que comparte con las Rx, y que la diferencia de la RMN (Resonancia Magnética Nuclear) o IRM (Imagen por Resonancia Magnética), que se basa en la capacidad de algunos núcleos para absorber ondas de radiofrecuencia cuando son sometidos al efecto de un campo magnético. Respecto a otras técnicas como los rayos X y la TAC, la RMN se usa

cada vez más por sus ventajas, como permitir cortes más finos, y en varios planos, ser más sensible para demostrar accidentes vasculares cerebrales, tumores y otras patologías, y no utilizar radiaciones ionizantes. Como desventajas tiene su mayor coste económico, el prolongado tiempo para obtener las imágenes y el tener que excluir a portadores de marcapasos y otros objetos extraños intracorpóreos.

La Ecografía es otra técnica de imagen, que utiliza las ondas ultrasónicas para producir imágenes de estructuras internas del cuerpo humano o del producto en desarrollo dentro de la madre.

La Medicina Nuclear se basa en la introducción de un isótopo radiactivo por diferentes vías para ver su distribución por el organismo. Posteriormente una cámara de escintigrafía registra la radiactividad del órgano en estudio, y a continuación se obtienen unas imágenes llamadas Gammagrafías, que tienen la virtud de ofrecer información funcional del cuerpo humano. Actualmente han surgido nuevas técnicas con un constante mayor uso, como la PET (Tomografía por emisión de positrones), y la SPECT (Tomografía por emisión de fotón único), con mayores resultados en algunas patologías, como las neurológicas.

La IMAGENOLOGÍA tiene un brillante porvenir, gracias a las posibilidades que ofrece la informática para obtener, conservar y transmitir imágenes.

1.2. MISIÓN, ESTRUCTURA Y FUNCIONES

La imagen es un elemento fundamental en el actual quehacer médico, por su papel en el estudio diagnóstico del paciente. Por lo que la IMAGENOLOGÍA tiene como Misión:

“ Ofrecer una reproducción de una parte del cuerpo humano que permita al médico acercarse al diagnóstico de la enfermedad, y así poder instaurar un tratamiento “.

En cuanto a lo que se refiere a su estructura está integrada por nuevas tecnologías, designadas globalmente como métodos de imagen, las cuales son:

- **RADIOLOGÍA:** Es la rama de la medicina que utiliza sustancias radioactivas, radiación electromagnética y ondas sonoras para crear

imágenes del cuerpo, sus órganos y estructuras con fines de diagnóstico y tratamiento.

Los diversos tipos de procedimientos radiológicos pueden agruparse según los medios en los que se producen sus imágenes: imagen por transmisión, imagen por reflexión o imagen por emisión.

Imagen por transmisión:

Los rayos X, la TC (Tomografía Computarizada) y la fluoroscopia son exámenes radiológicos cuyas imágenes se producen por transmisión. En las imágenes por transmisión, se produce un rayo de fotones de alta energía y se pasa a través de la estructura del cuerpo que está siendo examinada. El rayo pasa muy rápido a través de los tipos de tejido menos densos, como las secreciones acuosas, la sangre y la grasa, dejando un área oscurecida en la película de rayos X. Los tejidos conectivos y musculares (ligamentos, tendones y cartílago) aparecen en gris. Los huesos aparecerán en blanco.

- **Rayos X:** Para estudios de tórax, abdomen y músculo-esquelético. Estudios especiales de mielografía, artrografía e histerosalpingografía.



Fig. 1.2 Rx de pelvis

- **Tomografía Computarizada (TC):** Es un procedimiento de diagnóstico por imagen que utiliza una combinación de rayos X y tecnología computarizada para obtener imágenes de cortes transversales (a menudo denominadas "rebanadas"), tanto horizontales como verticales, del cuerpo. Un escáner TC muestra imágenes detalladas de cualquier parte del cuerpo, incluyendo los huesos, los músculos, la grasa y los órganos. El escáner TC es más detallado que los rayos X regulares. El escáner TC también reduce al mínimo la exposición a la radiación. La TC puede realizarse para ayudar a diagnosticar tumores, estudiar hemorragias internas o buscar otras lesiones o daños.



Fig. 1.3 TAC torácico de un paciente con una infección de vías respiratorias

- **Fluoroscopia:** Es un estudio de las estructuras del cuerpo en movimiento - similar a una "película" de rayos X. Se hace pasar un haz continuo de rayos X a través de la parte del cuerpo que va a examinarse, y se transmite a un monitor parecido a una televisión de forma que pueda verse en detalle la parte del cuerpo y su movimiento. La fluoroscopia se utiliza en muchos tipos de exámenes y procedimientos, como los rayos X con bario, la cateterización cardíaca y la colocación de catéteres intravenosos. En los rayos X con bario, la

fluoroscopia le permite al médico ver el movimiento de los intestinos a medida que el bario los recorre. En una cateterización cardíaca, la fluoroscopia le permite al médico ver el flujo de sangre a través de las arterias coronarias con el fin de evaluar la presencia de bloqueos arteriales. Para la inserción de un catéter intravenoso, la fluoroscopia ayuda al médico a guiar el catéter en la ubicación específica en el interior del cuerpo.



Fig. 1.4 Retinografía y Angiografía por fluoresceína

- **MASTOGRAFÍA:** Es un examen de rayos X de los senos. Se usa para detectar y diagnosticar enfermedades del seno en las mujeres que tienen problemas de los senos, como un bulto, dolor o secreción del pezón, y en las mujeres que no tienen ningún síntoma de enfermedad del seno.

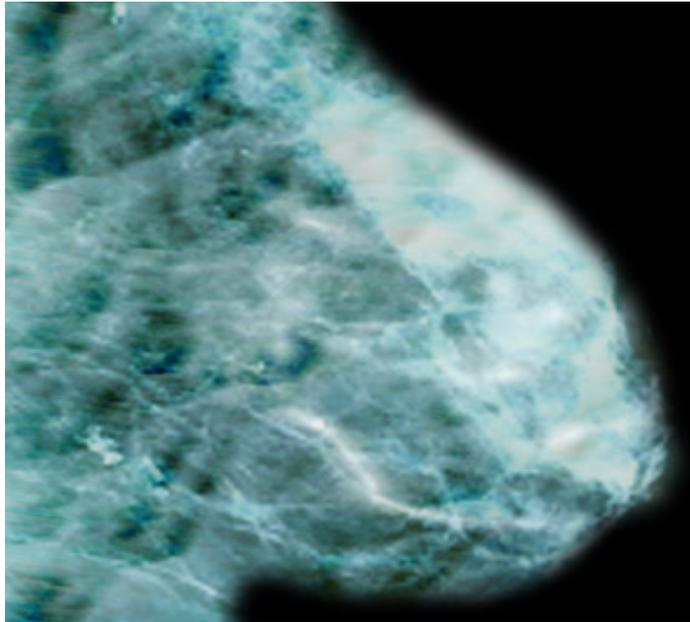


Fig. 1.5 Mamografía con pequeñas lesiones de fibrosis quística

Imagen por reflexión:

La imagen por reflexión se refiere al tipo de imagen producida al enviar sonidos de alta frecuencia a la parte del cuerpo u órgano que se está estudiando. Estas ondas sonoras "rebotan" en los diversos tipos de tejidos y estructuras del cuerpo a velocidades variables, dependiendo de la densidad de los tejidos presentes. Las ondas sonoras rebotadas se envían a una computadora que analiza las ondas sonoras y produce una imagen visual de la parte o estructura del cuerpo. La ecografía es un ejemplo de imagen por reflexión. La ecografía es una técnica de imagen basada en ultrasonidos de amplio uso en Medicina. El diagnóstico por ultrasonidos se basa en la detección de los ecos que provienen del interior del organismo.

- **ULTRASONIDO:** Utiliza ondas sonoras de alta frecuencia y una computadora para crear imágenes de los vasos sanguíneos, tejidos y órganos. Los ultrasonidos se utilizan para ver el funcionamiento de los órganos internos y para evaluar el flujo sanguíneo a través de diversos vasos. Los procedimientos del ultrasonido se utilizan a menudo para

examinar muchas partes del cuerpo, como el abdomen, los senos, la pelvis femenina, la próstata, el escroto, la tiroides y paratiroides, y el sistema vascular. Durante el embarazo, se hacen ultrasonidos para evaluar el desarrollo del feto.



Fig. 1.6 Ecografía de un riñón atrofico en un paciente hipertenso

Imagen por emisión:

La imagen por emisión se produce cuando un escáner detecta diminutas partículas nucleares o energía magnética y son analizadas por una computadora para producir una imagen de la estructura u órgano del cuerpo que está siendo examinado. Los estudios de medicina nuclear utilizan la emisión de partículas nucleares procedente de sustancias nucleares introducidas en el cuerpo específicamente para el examen. Los estudios de imágenes por resonancia magnética (IRM), por ejemplo, se obtienen utilizando un imán grande para provocar cambios en el tejido del cuerpo con el fin de detectar la energía magnética en la parte o estructura del cuerpo que está siendo examinada.

- **Resonancia Magnética (IRM):** Es un procedimiento de diagnóstico que utiliza una combinación de imanes grandes, radiofrecuencias y una computadora para producir imágenes detalladas de los órganos y las estructuras internas del cuerpo. La IRM se utiliza a menudo: para examinar el

corazón, el cerebro, el hígado, el páncreas, los órganos reproductores femeninos o masculinos, y otros tejidos blandos, para evaluar el flujo sanguíneo, para detectar tumores y diagnosticar muchas formas de cáncer, para evaluar infecciones, para evaluar lesiones en huesos y articulaciones.



Fig. 1.7 Resonancia Magnética. Litiasis de la vesícula biliar

- **IMÁGENES DE PATOLOGÍA:** Las imágenes de patología son el estudio de los tejidos extirpados de pacientes vivos durante una cirugía para ayudar a diagnosticar una enfermedad y determinar un plan de tratamiento.

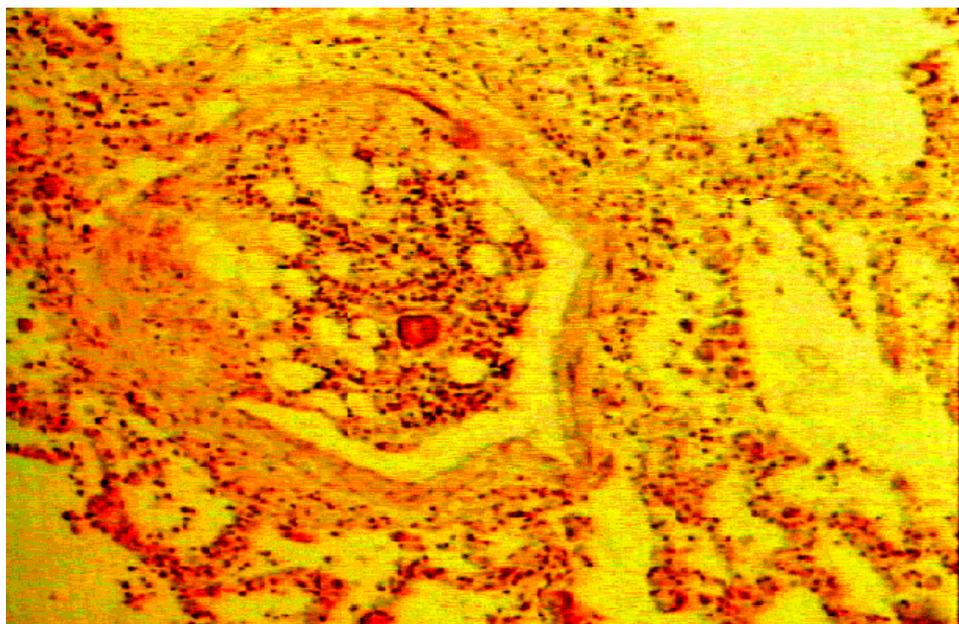


Fig. 1.8 Embolia en la medula ósea, pulmón.

- **FOTOGRAFÍAS MÉDICAS:** Son las fotografías de pacientes, utilizadas en estudios dermatológicos, psiquiátricos, ortopédicos, neurológicos, etc.

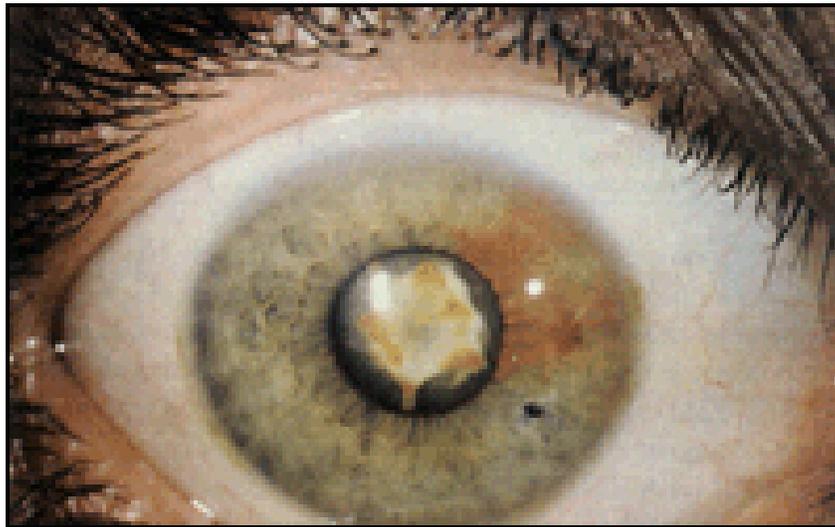


Fig. 1.9 Catarata secundaria a proceso uveítico.

De acuerdo a lo anterior, entre las funciones que tiene la IMAGENOLÓGÍA se encuentran las siguientes:

- Permitir la observación de las imágenes, impresas en una placa fotográfica o en papel, de cualquier estructura del cuerpo, sin molestias, sin riesgos, con una nitidez tal que es aceptada por los especialistas para el diagnóstico.
- Permitir el apoyo en el diagnóstico y tratamiento de enfermedades.
- Permite la detección y caracterización de las enfermedades basándose fundamentalmente en las propiedades físicas y fisiológicas como fuente principal de contraste.

- Las imágenes ayudan a diagnosticar y por lo tanto a emitir pronósticos y prescribir tratamientos con conocimiento de causa.

1.2.1. FACULTADES DE LA IMAGENOLOGÍA

Las facultades de la IMAGENOLOGÍA son:

- Revelar al médico la información necesaria para el diagnóstico exacto.
- Mostrar la eficacia del funcionamiento del cuerpo, sus órganos internos y estructura.
- Detección oportuna de numerosos padecimientos, para identificar el grado de extensión de una enfermedad y la posible afectación de otros órganos, con gran exactitud en el diagnóstico.

II. IDENTIFICACIÓN DE LA PROBLEMÁTICA DE LA IMAGENOLOGÍA

2.1. NECESIDADES

Debido a una reestructuración en el ISSSTE para una reducción de los gastos, dicho organismo se propuso adaptar su estructura, entre los departamentos que se observó que era conveniente dar un cambio es el de la Imagenología dado que este era uno de los que más gasto ejercía, para ello se fue adaptando a las nuevas necesidades de la tecnología que al paso del tiempo se le fueron exigiendo a este organismo.

Para llevar a cabo este propósito se sistematizó la consulta a los expedientes clínicos referidos con la Imagenología, a fin de enfocar el servicio más eficiente y un menor tiempo de espera, para ello, se inició la búsqueda de nuevas formas de fusionar estos objetivos para así cumplir con la minimización de los gastos.

2.2. PROBLEMÁTICA

En el presente, al realizar un estudio sobre la estructura y las funciones del ISSSTE en el área de Imagenología se encontraron una serie de problemas relacionados con la consulta y diagnóstico de los diferentes campos de la Imagenología, cuando estos eran requeridos de una distancia considerable. Por lo tanto el área de telemedicina se impuso la tarea de buscar mecanismos para mejorar y ordenar estos problemas así como de orientar sus funciones. Dichos mecanismos deberían de contener en esencia la prioridad de brindar un mejor servicio y que éste, estuviera guiado a todos los derechohabientes. Este sistema automatizará la consulta de los expedientes de los diversos pacientes de IMAGENOLOGÍA para así reducir tiempos de espera, hacer diagnósticos a distancia, evitar pérdida de expedientes, y llevar un orden en las consultas realizadas así como la seguridad de las mismas.

Con el propósito de que la telemedicina tenga una administración acorde a los lineamientos que en la actualidad le marca el ISSSTE y las leyes mexicanas,

se empezó una reestructuración y sistematización de las funciones del área de Imagenología Clínica. Se organizó y estructuró programas de acción e interacción de todos y cada uno de los elementos del área de Imagenología Clínica.

En esta etapa de reorganización se detectaron grandes tiempos de espera para obtener el diagnóstico, así como un gran costo de transporte de los derechohabientes y en el envío de los expedientes así como una gran pérdida de los mismos, y una equivocada entrega de expedientes a sus doctores correspondientes debido a que muchos pacientes tienen uno a más doctores. Por lo tanto se pretende homogenizar dicha información y que el doctor autorizado del ISSSTE pueda contar con ella para su consulta, desde cualquier parte del mundo.

2.3. ORGANIZACIÓN ACTUAL

Con la cooperación de la coordinación de informática y el Administrador de la base de datos se instaló en el Área de Telemedicina un plan para el desarrollo de un programa de computó que lograra la vinculación integral entre todos los campos de la Imagenología y sus actividades de las mismas.

En cuanto a la desvinculación del entorno se iniciaron mecanismos de relación con los hospitales y dependencias del ISSSTE en todo el país, a fin de beneficiar a un mayor número de derechohabientes con los servicios que esta dependencia ofrece. De igual manera, se realizaron acciones buscando una comunicación más abierta entre los Doctores encargados de dependencias del ISSSTE en poblados lejanos con los doctores encargados de realizar los diferentes estudios del Área de Imagenología Clínica.

Así mismo, las acciones de Imagenología empezaron a regirse bajo el propósito de llevar estas acciones a todas y cada una de las dependencias del ISSSTE, privilegiando las prácticas de las mismas, para ello se estableció una estrategia de trabajo basado en la coordinación y descentralización de las actividades del centro del país, (dado que aquí es donde se tiene la tecnología para la realización de estos estudios médicos), y en el establecimiento de programas concretos que permitan la distribución de la tarea en todas las dependencias del país.

Los diferentes campos del área de Imagenología a los cuales se les aplicó los diferentes planes de trabajo son:

- **Tomografía.** Permite al médico tener cortes visuales finos, axiales o coronales, para valorar con mayor detalle la patología de su paciente.
- **Mastografía.** Estudios de la mama o seno de la mujer en edad fértil. Tecnología de punta para la búsqueda de quistes, malignos o benignos, y para la toma de biopsias o muestras de tejido
- **Ultrasonido.** Muy utilizado para la valoración integral de la mujer gestante, así como para el rastreo folicular o búsqueda en hueco pélvico y valoración de hígado, vías biliares, páncreas, bazo y riñones.
- **Radiología.** Para estudios de tórax, abdomen y músculo-esquelético. Estudios especiales de mielografía, artrografía e histerosalpingografía.
- **Resonancia Magnética.** La resonancia magnética es el procedimiento que permite ver mejor los tejidos blandos, es decir, todos los componentes de la columna vertebral que no son hueso, como el disco intervertebral, la médula espinal, las raíces nerviosas o la eventual existencia de fibrosis posquirúrgica.
- **Fluoroscopia.** La fluoroscopia es un estudio de las estructuras del cuerpo. Se hace pasar un haz continuo de rayos X a través de la parte del cuerpo que va a examinarse.
- **Imágenes de Patología.** Las imágenes de patología son el estudio de los tejidos extirpados de pacientes vivos durante una cirugía para ayudar a diagnosticar una enfermedad y determinar un plan de tratamiento.
- **Fotografías Médicas.** Son las fotografías de pacientes, utilizadas en estudios dermatológicos, psiquiátricos, ortopédicos, neurológicos, etc.

Las necesidades de información que ha de cubrir este programa es vincular las dependencias se puede resumir en dos tareas fundamentales: las cuales son:

La Consulta de Expedientes Clínicos y el Diagnóstico de los mismos englobando con esto a una importante minimización de los costos.

Por lo tanto el problema a resolver consiste en crear un sistema para computadora el cual maneje toda la información de los derechohabientes que fueron enviados a realizarse un estudio en el área de Imagenología Clínica. Al contar con esta herramienta se facilitaran los diagnósticos, sin importar la distancia en que se encuentren los especialistas, siempre y cuando estén capacitados para el uso de medios de comunicación, deberán de estar debidamente identificados en el sistema (para esto cada Doctor que intente ingresar al sistema deberá contar con una clave y contraseña que será otorgada por el Administrador de la base de dato), para poder apoyar el diagnóstico, indicar tratamiento, mantener en orden las imágenes capturadas y contar a su vez con un documento que capture la causa de la consulta, y la opinión del especialista distante, con la debida identificación por medio de su código, y que dicho documento pueda imprimirse como anexo al expediente del paciente.

2.4 OBJETIVO

El objetivo principal de este trabajo es la creación de un sistema seguro y eficiente para el manejo de la información por computadora, reducir el tiempo de espera en las consultas de los expedientes, apoyar las consultas y segundas opiniones médicas, para consulta de bancos de imágenes, de una forma sencilla, amigable, mediante el empleo de gráficos y de acceso controlado, utilizando la plataforma Windows, además este sistema deberá funcionar bajo la arquitectura Cliente/Servidor, estando este ultimo en las oficinas centrales del ISSSTE y los clientes en todas y cada una de las dependencias del ISSSTE a nivel nacional. También este sistema deberá tener la posibilidad de emitir reportes impresos.

En los siguientes capítulos se define como se atacara el problema para tener el sistema necesario para el manejo de la gran cantidad y gran variedad de la información que se tiene en la Imagenología.

III. FUNDAMENTOS TEÓRICOS

3.1. PROCESO DE DESARROLLO DE UN SISTEMA INFORMÁTICO

Los elementos de un sistema basado en computadora son [Pressman]:

- SOFTWARE: Los programas de computadora, las estructuras de datos y la documentación asociada, que sirven para realizar el método lógico, procedimiento o control requerido.
- HARDWARE: Los dispositivos electrónicos (por ejemplo: UCP, memoria) que proporcionan la capacidad de computación y los dispositivos electromecánicos (por ejemplo: sensores, motores, bombas) que proporcionan las funciones del mundo exterior.
- GENTE: Los individuos que son usuarios y operadores del software y del hardware.
- BASES DE DATOS: Una colección grande y organizada de información a la que se accede mediante el software y que es una parte integral del funcionamiento del sistema.
- DOCUMENTACIÓN: Los manuales, los impresos y otra información descriptiva que explica el uso y/o la operación del sistema.
- PROCEDIMIENTOS: Los pasos que definen el uso específico de cada elemento del sistema o el contexto procedimental en que reside el sistema.

Actualmente, el software ha superado al hardware como la clave del éxito de muchos sistemas basados en computadora. Lo que diferencia a una compañía de su competidora es la suficiencia y oportunidad de la información dada por el software (y bases de datos relacionadas).

Existen tres elementos clave que facilitan al gestor controlar el proceso del desarrollo de un sistema, los cuales son: métodos, herramientas y procedimientos.

Varias metodologías están disponibles que abarcan los tres elementos antes mencionados, para el desarrollo de un sistema basado en computadora, las cuales se mencionan a continuación:

- a) Ciclo de vida clásico o modelo en cascada
- b) Construcción de prototipos
- c) Modelo en espiral o evolutivo
- d) Ciclo de vida clásico modificado

Independientemente de la metodología elegida, el proceso de desarrollo de un sistema contiene tres fases genéricas:

- **Definición:** Durante esta fase, el que desarrolla el sistema intenta identificar qué información ha de ser procesada, qué función y rendimiento se desea, qué interfaces han de establecerse, qué restricciones de diseño existen y qué criterios de validación se necesitan para definir un sistema correcto.
- **Desarrollo:** Durante esta fase el que desarrolla el software intenta descubrir cómo han de diseñarse las estructuras de datos, cómo ha de traducirse el diseño a un lenguaje de programación y cómo ha de realizarse la prueba.
- **Mantenimiento:** Se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas por la evolución del entorno del sistema y a las modificaciones debidas a los cambios de los requisitos del cliente dirigidos a reforzar o ampliar el sistema.

3.1.1. Metodologías más utilizadas

METODOLOGÍA: Es una versión amplia y detallada del desarrollo de un sistema, la cual incluye:

- Tareas paso a paso por fase.
- Funciones individuales y en grupo.
- Productos resultantes por tarea.
- Técnicas de desarrollo por tarea.

El propósito del ciclo de vida es: planear, ejecutar y controlar el desarrollo de un sistema. El ciclo de vida define las fases y las tareas esenciales para el

desarrollo de sistema, sin importar el tipo de envergadura del sistema que se intenta construir.

A continuación se analizan las metodologías más utilizadas:

a) Ciclo de Vida Clásico

El ciclo de vida clásico exige un enfoque sistemático y secuencial del desarrollo del sistema que comienza en el nivel de la ingeniería del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento.

Ingeniería de Sistemas: Debido a que el software es siempre parte de un sistema mayor, se comienza estableciendo los requisitos de todos los elementos del sistema y luego se asignan al software. Este planteamiento del sistema es esencial cuando el software debe interrelacionarse con otros elementos, tales como hardware, personas y bases de datos.

Análisis: Proceso de recopilación de los requisitos el cual se centra e intensifica especialmente para el software. Se debe comprender el ámbito de la información del software, la función, el rendimiento y las interfaces requeridas.

Diseño: El proceso de diseño traduce los requisitos en una representación del software. El diseño se documenta y forma parte de la configuración del software.

Codificación: En este proceso se realiza la traducción del diseño en una forma legible para la máquina. Si el diseño se realiza de una manera detallada, la codificación puede realizarse mecánicamente.

Prueba: La prueba se centra en la lógica interna del software, comprobando sentencias, funciones externas, entradas y resultados esperados.

Mantenimiento: El mantenimiento del software aplica cada uno de los pasos precedente del ciclo de vida a un programa existente en vez de a uno nuevo.

b) Construcción de Prototipos

La *construcción de prototipos* es un modelo que facilita la creación del sistema a construir mediante una de las tres formas siguientes.

- 1) Un prototipo en papel o un modelo basado en computadora que describa la interacción hombre-máquina, de forma que facilite al usuario la comprensión de cómo se producirá tal interacción.
- 2) Un prototipo que implemente algunos subconjuntos de la función requerida del programa deseado.
- 3) Un programa existente que ejecute parte o toda la función deseada, pero que tenga otras características que deban ser mejoradas en el nuevo trabajo de desarrollo.

Como en todos los métodos de desarrollo de software, la construcción de prototipos comienza con la recolección de los requisitos. Luego se produce un “diseño rápido”. El diseño rápido se enfoca sobre la representación de los aspectos del software visibles al usuario (por ejemplo, métodos de entrada y formatos de salida). El diseño rápido conduce a la construcción de un prototipo. El prototipo es evaluado por el cliente y se utiliza para refinar los requisitos del software a desarrollar. Se produce un proceso interactivo en el que el prototipo es “afinado” para que satisfaga las necesidades del cliente, al mismo tiempo que facilita al que lo desarrolla una mejor comprensión de lo que hay que hacer.

c) Modelo en Espiral

El modelo en espiral ha sido desarrollado para cubrir las mejores características tanto del ciclo de vida clásico, como de la creación de prototipos, añadiendo al mismo tiempo un nuevo elemento: el análisis de riesgo. El modelo es representado mediante una espiral que define cuatro actividades principales, representadas por cuatro cuadrantes:

- Planificación: Determinación de objetivos, alternativas y restricciones.
- Análisis de riesgo: Análisis de alternativas e identificación y/o resolución de riesgos.
- Ingeniería: Desarrollo del producto de “siguiente nivel”.
- Evaluación del cliente: Valoración de los resultados de la ingeniería.

Con cada iteración alrededor de la espiral se construyen sucesivas versiones del software, cada vez más completas. Durante la primera vuelta alrededor de la espiral se definen los objetivos, las alternativas y las restricciones, y se analizan e identifican los riesgos. Si el análisis de riesgo indica que hay una incertidumbre en los requisitos, se puede usar la creación de prototipos en el cuadrante de ingeniería para dar asistencia tanto al encargado del desarrollo como al cliente.

El cliente evalúa el trabajo de ingeniería y sugiere modificaciones. En base a los comentarios del cliente se produce la siguiente fase de planificación y de análisis de riesgo. En cada bucle alrededor de la espiral, la culminación del análisis de riesgo resulta en una decisión de “seguir o no seguir”. Si los riesgos son demasiado grandes, se puede dar por terminado el proyecto.

Cada vuelta alrededor de la espiral requiere ingeniería, que se puede llevar a cabo mediante el enfoque del ciclo de vida clásico o de la creación de prototipos.

d) Ciclo de Vida Clásico Modificado

Tomando en cuenta los elementos del ciclo de vida clásico previamente expuestos, y con la intención de hacerlo aplicable a la mayoría de las situaciones, surge el *ciclo de vida clásico modificado* que proporciona más flexibilidad entre las fases, permitiendo iteraciones entre ellas, según sean necesarias; trata de acomodar cambios en los requerimientos y disminuir los efectos de errores en cada etapa del desarrollo.

3.2. TÉCNICAS DE DESARROLLO DE SISTEMAS

A continuación se definirá que es una técnica.- Una técnica es un método que aplica herramientas y reglas específicas para completar una o más fases del ciclo de vida del desarrollo de sistemas. Uno de sus sinónimos habituales es el *paradigma*.

Las técnicas son, en su mayoría, sólo aplicables a una parte del ciclo de vida total. Por tanto, no pueden, por sí mismas, reemplazar al ciclo de vida. Algunas de las más conocidas técnicas de desarrollo de sistemas pretenden introducir en el ciclo de vida una precisión y un rigor propios de la ingeniería.

Son de destacar dos puntos importantes. En primer lugar, una auténtica metodología debe acompañar al ciclo de vida completo del desarrollo de sistemas, incluido el soporte de sistemas. Y segundo, la mayor parte de las metodologías modernas incluyen el uso de varias técnicas de desarrollo (con sus herramientas asociadas).

3.2.1. Técnicas estructuradas

Las técnicas estructuradas son métodos formales de división de un problema de empresas en fragmentos y relaciones manejables, y la ulterior reunión de estos fragmentos y relaciones (posiblemente, con añadidos y eliminaciones) en una solución de empresa (e informática) útil para resolver el problema. Uno de sus sinónimos es *métodos estructurados*.

En cierto sentido, las técnicas estructuradas utilizan el método de “divide y vencerás” para resolver problemas relacionados con el desarrollo de software y sistemas. A continuación se mencionarán las técnicas más populares entre los desarrolladores de sistemas:

- Programación estructurada.
- Diseño estructurado (varios métodos).
- Análisis estructurado moderno.
- Modelización de datos.
- Ingeniería de información.

La mayoría de las técnicas estructuradas de hoy en día se centran en dos visiones de los sistemas: **Actividades** y **Datos**. Recordemos que los bloques elementales **Actividades** se basaban en el concepto entrada-proceso-salida (EPS). Como el proceso es el núcleo de la transformación entrada-salida, estas técnicas reciben con frecuencia el nombre de **técnicas orientadas a procesos**. Las técnicas de procesos elaboran modelos de sistemas que se basan en el estudio de los procesos y/o sus entradas y salidas. El diseño estructurado y el análisis estructurado moderno son ejemplos de estas técnicas.

Las **técnicas orientadas a datos** establecen modelos de sistemas que se basan en una organización y un acceso ideales a los datos del sistema, independientemente de cómo se utilicen estos datos para satisfacer las

necesidades de información (salidas). La modelización de datos y la ingeniería de información son ejemplos de técnicas orientadas a datos.

A continuación se verán a detalle estas técnicas estructuradas:

a) Programación estructurada

La programación estructurada es una técnica orientada a procesos para el diseño y la escritura de programas con mayor claridad y consistencia. En esencia, la programación estructurada sugiere que el esquema lógico de cualquier programa debería ser escrito con un conjunto limitado de estructuras de control.

Esta técnica tiene que ver sólo con la lógica y la codificación de los programas. Propone que los programas deberían diseñarse de manera que pudieran leerse de principio a fin con un mínimo de ramificaciones. En particular, los programas bien estructurados se escriben exclusivamente con diversas combinaciones de tres **estructuras de control limitadas**. Estas tres estructuras básicas son:

- **Una secuencia de instrucciones.** Implementa los pasos de procesamiento esenciales de la especificación de cualquier algoritmo.
- **Una selección de instrucciones.** Permite seleccionar un procedimiento basado en alguna ocurrencia lógica (Esta construcción recibe a menudo el nombre de if-then-else o construcción de casos).
- **Una iteración de instrucciones.** Se repite sobre la base de ciertos criterios (construcción que se da en dos formas básicas repeat-until y do-while).

Esta construcción puede repetirse en otras construcciones. Una característica importante de las estructuras es que cada construcción debe tener la propiedad de disponer de una sola entrada y una sola salida. Ello quiere decir que sólo puede haber un punto de entrada y un punto de salida en una estructura dada. El código estructurado se lee como ésta página, de arriba abajo, sin referencias hacia atrás. Esto hace que sea más fácil de leer, probar, depurar y mantener.

Como técnica estructurada, la programación estructurada sólo da apoyo a algunas partes de las fases de diseño, implantación y soporte del ciclo de vida.

Los programas estructurados son más fáciles de escribir, probar y depurar, y mucho más sencillos de mantener.

b) Diseño estructurado

El diseño estructurado es una técnica orientada a procesos utilizada para fragmentar un programa grande en un conjunto jerarquizado de módulos y obtener un programa informático más fácil de implantar y mantener (cambiar), entre sus sinónimos (aunque técnicamente inadecuados) se encuentran *diseño descendente de programas* y *programación estructurada*.

El enfoque básico en el diseño estructurado es la conversión sistemática de los diagramas de flujo de datos en cartas de estructura.

El primer paso en el diseño estructurado es la revisión y refinamiento de los diagramas de flujo de datos desarrollados durante la definición de requisitos y el diseño externo. El segundo paso es determinar si el sistema está centrado en la transformación u orientado por las transacciones, y derivar una carta de estructura de alto nivel. En un sistema centrado en transformación, el diagrama de flujo de datos contiene los segmentos de entrada, procesamiento y salida que se convierten en los subsistemas de entrada, procesamiento y salida en la carta de estructura.

El tercer paso en el diseño estructurado es la descomposición de cada subsistema en módulos (grupo de instrucciones: un párrafo, un bloque, un subprograma o una subrutina), utilizando principios generales tales como el acoplamiento, la cohesión, cubrimiento de la información, niveles de abstracción, abstracción de datos, y otros criterios de descomposición. La estructura descendente de estos módulos se desarrolla conforme a diversas reglas y directrices de diseño.

La descomposición de las funciones de procesamiento en módulos debe continuarse hasta que cada módulo no contenga ningún subconjunto de elementos que pueda utilizarse solo, y hasta que cada módulo sea lo suficientemente pequeño como para que su instrumentación pueda alcanzarse en una sola vez. El diseño estructural, normalmente implica numerosas iteraciones y revisiones antes de que se obtenga una estructura apropiada [Fairley].

Diferentes escuelas de pensamiento han desarrollado su propia técnica para conseguir diseños bien estructurados. Entre estas escuelas se incluyen las siguientes:

1. **Yourdon-Constantine.** Esta técnica obtiene la estructura de software ideal por el estudio de flujo de datos a través de las funciones de programa necesarias. (Nota: Esta técnica describe la estructura descendente de los módulos en forma de un árbol invertido.)
2. **Warnier-Orr.** Esta técnica obtiene la estructura de software ideal por el estudio del contenido de las salidas y las entradas. (Nota: Esta técnica describe la disposición jerárquica descendente en forma de series de llaves de izquierda a derecha.)
3. **Jackson.** Esta técnica también obtiene la estructura de software ideal por el estudio del contenido de las salidas y las entradas. (Esta técnica, extendida en Europa, describe la disposición jerárquica descendente de los módulos en forma de un árbol invertido, similar al de Yourdon-Constantine.)

Todas estas técnicas se consideran técnicas de procesos, dado que su propósito es diseñar procesos y, específicamente, procesos de software.

A continuación se explican las propiedades con las que cuenta el **diseño estructurado de Yourdon**:

- a) Los módulos deben tener una fuerte cohesión; es decir, cada módulo debería comportar una y solo una función. Así, los módulos serán reutilizables en futuros programas.
- b) Los módulos deben estar debidamente acoplados; es decir, han de tener una dependencia mínima unos de otros. Ello reduce al mínimo el efecto que futuros cambios en un módulo pueden producir en otros módulos.

El modelo de software obtenido del diseño estructurado de Yourdon recibe el nombre de **diagrama de estructuras**. El diagrama de estructuras se obtiene del estudio del flujo de datos a través del programa. El diseño se lleva a cabo durante la fase de diseño del ciclo de vida. No cubre todos los aspectos del diseño; por ejemplo, el diseño estructurado no servirá de ayuda para diseñar las entradas, la base de datos o los archivos. Las ventajas obtenidas del diseño estructurado son numerosas. En primer lugar, los programas que se desglosan conforme al diseño estructurado son más fáciles de leer y de probar por equipos de programadores múltiples, ya que las interfaces entre los módulos también estarán bien definidas y limitadas por las reglas, los módulos cuya prueba sea correcta deberían funcionar igual de bien cuando se integraran en el sistema global. En segundo lugar, los sistemas y los programas desarrollados con diseño estructurado son más fáciles de mantener. En tercer lugar, una de las ventajas, con frecuencia olvidada, del

diseño estructurado es que los módulos de programas desarrollados conforme a la técnica suelen ser reutilizables. Ello se debe a que han sido escritos con cohesión.

Debido a las características antes mencionadas, para el desarrollo de nuestro proyecto, elegimos la técnica de **diseño estructurado de Yourdon**.

c) Análisis estructurado moderno

El análisis estructurado es una actividad de construcción de modelos. Mediante una notación que es única del método de análisis estructurado, se crean modelos que reflejan el flujo y el contenido de la información (datos y control).

El análisis estructurado divide un sistema en procesos, entradas, salidas y archivos. Elabora modelos del tipo entrada-proceso-salida orientados a flujos para un problema o una solución de empresa.

Los principales impulsores del análisis estructurado son Tom DeMarco, Chris Gane, Trish Sarson y Ed Yourdon.

A medida que fluye por un sistema basado en computadora, la información se transforma. El diagrama de flujo de datos (DFD) es una técnica gráfica que representa el flujo de la información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida.

Como los diagramas de flujo de datos describen el flujo de datos entre procesos, el análisis estructurado recibe también el nombre de método de *flujo de datos*. Por otra parte, en su mayoría, los especialistas consideran que los DFD son *modelos de procesos* y centran su análisis en dichos procesos (de ahí que el análisis estructurado sea en realidad un método orientado a procesos).

El analista debería producir varios DFD en su análisis estructurado. Estos DFD pueden diferir en lo que se refiere a:

- ❖ Si el modelo corresponde al sistema actual o al sistema propuesto.
- ❖ Si el modelo corresponde a los detalles de implantación del sistema (a veces denominado sistema físico) o a los fundamentos del sistema (sistema lógico).

Así pues, el modelo obtenido con los DFD ha de corresponder a: (1) la implantación del sistema actual; (2) las bases del sistema actual; (3) las bases del sistema propuesto, o (4) la implantación del sistema propuesto.

El concepto de **sistema lógico**, a veces denominado **sistema esencial**, fue creado por el análisis estructurado. El análisis estructurado anima al analista a que defina *qué* debería hacer un sistema (el **sistema lógico**) antes de decidir *cómo* diseñarlo o implementarlo (**sistema físico**). Por tanto, esta técnica obliga al analista a considerar primero la “solución de empresa” y sólo más tarde, la “solución técnica”. Al separar el sistema lógico del físico se obtienen las siguientes ventajas:

- 1) El analista identifica con mayor precisión las necesidades de la empresa y de los usuarios finales, al no preocuparse antes de tiempo por las cuestiones tecnológicas.
- 2) El analista muestra mayor tendencia a concebir soluciones alternativas más creativas, en lugar de soluciones basadas en el sistema existente. Las soluciones creativas pueden promover ventajas del tipo competitivo en la compleja situación económica actual.

Recientemente, Ed Yourdon introdujo una versión mejorada del análisis estructurado denominada **análisis estructurado moderno**. Este método elimina la modelización detallada del sistema actual (tanto lógica como física).

Adviértase que el interés central de todas las etapas se ha desplazado claramente hacia el nuevo sistema, prestándose escasa atención al sistema actual. Las diferentes variedades de análisis estructurados se encuadran o bien en el método original o bien en el método moderno. En la actualidad, la mayoría de los expertos admite que, como mínimo, debería dedicarse menos tiempo a la modelización detallada del sistema actual. Ello se debe a que: (1) la cantidad de tiempo a menudo no justifica el costo; (2) el sistema actual tiende a condicionar o limitar la creatividad cuando cambia el contexto desde el sistema actual al nuevo sistema, y (3) los usuarios y los directivos muestran poca paciencia a la hora de modelizar un sistema que, según las expectativas, será modificado o sustituido.

El análisis estructurado fue la primera técnica estructurada que se utilizó para estudiar de modo específico las fases del análisis de sistemas del ciclo de vida. Pero sólo examina dichas fases desde el punto de vista de las especificaciones o la modelización. El analista debe acudir a otras técnicas para recoger los hechos, analizar los problemas y las oportunidades del sistema,

identificar las necesidades de los usuarios, identificar las soluciones candidatas y evaluar la viabilidad de estas soluciones.

Debería advertirse que el análisis estructurado y el diseño estructurado son técnicas integradas. El método de diseño estructurado de Yourdon proporciona estrategias formales para obtener diagramas de estructuras de programas a partir de diagramas de flujo de datos elaborados adecuadamente durante el análisis estructurado. En su conjunto, esta técnica se suele también llamar *ingeniería de software*.

d) Modelización de datos

En la actualidad, las técnicas orientadas a procesos han recibido el complemento de las técnicas orientadas a datos.

La **modelización de datos** es una técnica que se centra únicamente en los datos (y, por tanto, satisface uno de los principios fundamentales de análisis), representando una “red de datos” existente para un sistema dado. La modelización de datos es útil para aplicaciones en las que los datos y las relaciones que gobiernan los datos son complejos. A diferencia del enfoque de análisis estructurado, la modelización de datos considera los datos independientemente de cómo se procesen dichos datos para producir información.

La modelización de datos se usa ampliamente en aplicaciones de bases de datos. Proporciona al analista y al diseñador de la base de datos una amplia visión de los datos y de las relaciones que gobiernan los datos. Dentro del contexto del análisis estructurado se puede usar la modelización de datos para representar el contenido de los almacenes de datos y de las relaciones que existen entre ellos. Por ejemplo, un elemento de datos de una estructura de archivo (un almacén de datos) puede ser un puntero a otra estructura de archivo (un almacén de datos diferente). La notación del análisis estructurado no proporciona mecanismos directos para representar este tipo de relaciones. La modelización de datos sí.

El modelo de datos se compone de tres piezas de información interrelacionadas: el objeto de datos (también llamado “entidad”), los atributos que describen el objeto de datos, y la relación que conecta objetos de datos entre sí.

Objetos de datos: Un objeto de datos es una representación de cualquier composición de información compuesta que deba comprender el sistema (software). Por *composición de información*, entendemos todo aquello que tiene un número de propiedades o atributos diferentes. Los objetos de datos pueden ser:

- 1) Cosas tangibles (por ejemplo, materiales, suministros, máquinas, vehículos y productos).
- 2) Funciones (como clientes, proveedores, empleados y accionistas).
- 3) Sucesos (como pedidos, devoluciones, contratos, viajes, accidentes o pagos).
- 4) Lugares (como oficinas o almacenes).

Atributos: Los atributos definen las propiedades de un objeto de datos y toman una de las tres características diferentes. Se pueden usar para (1) dar nombre a una instancia del objeto de datos, (2) describir la instancia, o (3) hacer referencia a otra instancia de otra tabla. Además, uno o varios atributos se definen como un identificador, es decir, el atributo identificador supone una “clave” cuando queramos encontrar una instancia del objeto de dato. Por ejemplo, consideremos la entidad PACIENTE. Algunos de los atributos que describen una presencia de PACIENTE son: CURP DEL PACIENTE, NOMBRE DEL PACIENTE, EDAD, SEXO, DIRECCIÓN, TELÉFONO, etc. Eventualmente, puede desearse almacenar estos atributos, o campos, en un archivo o base de datos de pacientes. También eventualmente, se diseñarían entradas para capturar estos atributos, y podrían diseñarse numerosas variedades de salidas para resumir informaciones sobre estos y otros atributos.

Relaciones: Los objetos de datos se conectan entre sí de muchas formas diferentes. Se identifican las actividades de empresa que tienen lugar entre las entidades; por ejemplo, PACIENTES “TIENEN ESTUDIOS DE” ULTRASONIDO. PACIENTES y ULTRASONIDO son entidades. “TIENEN ESTUDIOS DE “ es una actividad de empresa que asocia presencias de las entidades PACIENTES y ULTRASONIDO. ¿Dónde radica la importancia de estas asociaciones?. No sólo queremos almacenar datos sobre PACIENTES y ULTRASONIDO, sino también conocer qué PACIENTES tienen un estudio de ULTRASONIDO. Normalmente, el analista compone una imagen o **modelo de datos** de estas entidades, relaciones y atributos.

En algunos métodos de modelización de datos, el analista utiliza también técnicas formales para asegurar que el modelo de datos sea lo suficientemente flexible para adaptarse a las necesidades actuales y futuras que se basen en los mismos datos. Esta técnica es la normalización, que dice lo siguiente: La normalización es un método de análisis de datos que organiza los atributos de datos de manera que se agrupen entre sí para formar entidades estables, flexibles y adaptables.

Las ventajas extraídas de la modelización de datos son importantes. Si los archivos y la base de datos de los nuevos sistemas se construyen de acuerdo al modelo de datos, poseerán las siguientes propiedades:

- 5) Contendrán datos actualizados y precisos.
- 6) Satisfarán todas las necesidades actuales.
- 7) Satisfarán los requisitos futuros sin necesidad de cambios drásticos en el sistema, ya que los datos ya se han añadido, o pueden añadirse fácilmente, a las entidades apropiadas.

En otras palabras, los procesos y las aplicaciones pueden evolucionar en torno al modelo de datos, con la realización de ligeros cambios en el propio modelo,

Muchos libros representan la modelización de datos sólo como una técnica de *base de datos*. En realidad, funciona igualmente bien en el caso de archivos convencionales y datos distribuidos. La mayoría de los libros aboga por la construcción de base de datos y sistemas corporativos de gran envergadura. En la práctica, la técnica funciona con los mismos buenos resultados en base de datos de aplicaciones de menor tamaño que pueden ser fácilmente integradas en fecha posterior.

Aunque la modelización de datos es bastante útil, antes o después habrán de diseñarse también los modelos de procesos. Pueden utilizarse herramientas y técnicas orientadas a procesos, como el análisis estructurado moderno, para estudiar estas necesidades de procesos y, de este modo, servir de complemento a las técnicas de modelización de datos.

e) Ingeniería de información

Además de combinar la modelización de datos y de procesos, la ingeniería de información pone un especial acento en la importancia de la planificación de sistemas. La **ingeniería de información** es una técnica basada en los datos, pero también sensible a los procesos, aunque la técnica propone un equilibrio entre los métodos orientados a datos y orientados a procesos, se basa claramente en los datos; primero se elaboran los modelos de datos y, después, los de procesos. Esta es la primera técnica estructurada de las que se han abordado que cubre casi todo el ciclo de vida. En realidad, la única fase no incluida en la ingeniería de información es el soporte de sistemas. Particular interés en la ingeniería de información es su tratamiento de la planificación de sistemas. La ingeniería de información define la **planificación de sistemas** como la mejora de una organización a través de la tecnología de información.

En la ingeniería de información, el centro primordial son los datos almacenados. Los analistas, los programadores y otros profesionales de la informática son responsables de diseñar todos los almacenes de datos y de asegurar que dichos datos sean capturados, almacenados y mantenidos adecuadamente. También diseñan e implantan las principales salidas de información del sistema. Los usuarios finales satisfacen muchas necesidades adicionales de información por medio del estudio y la aplicación de informes fáciles y agradables, y de lenguajes de consulta. Las técnicas de ingeniería de información parecen estar adquiriendo cada vez mayor fuerza entre los desarrolladores de sistemas, ello se debe a su capacidad para equilibrar e integrar verdaderamente las perspectivas de los datos y los procesos.

3.2.2. La técnica del Desarrollo Conjunto de Aplicaciones (DCA)

Mientras que las técnicas anteriormente tratadas ponen especial acento en los procesos y los datos, una técnica cada vez más popular denominada desarrollo conjunto de aplicaciones centra su interés principalmente en las personas.

El **desarrollo conjunto de aplicaciones (DCA)** es una forma de trabajo altamente estructurada que lleva a los usuarios, los directivos y los especialistas en sistemas de información a definir y especificar conjuntamente las necesidades de los usuarios, las opciones técnicas y los diseños externos (entradas, salidas y pantallas).

Clientes e ingenieros de software, a menudo, tienen inconscientemente una forma de pensar en términos de “nosotros y ellos”. Más que trabajar en equipo para identificar y refinar los requisitos, cada parte define su propio “territorio” y se comunica a través de una serie de notas, impresos formales, documentos y sesiones de preguntas y respuestas. La historia nos ha demostrado que este enfoque no es muy efectivo. Abundan los malentendidos, se pierde información importante y nunca se establece una relación de trabajo satisfactoria.

El DCA intenta requerir mayor participación de los usuarios y los directivos en el ciclo de vida del desarrollo de sistemas. Del aumento de esta participación pueden obtenerse numerosas ventajas:

- 1) El DCA suele mejorar la relación entre usuarios, directivos y profesionales en sistemas de información.

- 2) El DCA suele ampliar la cultura informática de los usuarios y los directivos, así como la cultura en la empresa y sus aplicaciones, y de los especialistas en sistemas de información.
- 3) El DCA sitúa la responsabilidad de la resolución de los conflictos donde corresponda (haciéndola recaer tanto en los usuarios como en los directivos).
- 4) El DCA reduce por lo general el tiempo transcurrido en el desarrollo de los sistemas, gracias a la síntesis de entrevistas múltiples en el trabajo conjunto estructurado.
- 5) El DCA reduce normalmente el costo del desarrollo de sistemas al conseguir definir las necesidades y establecer sus prioridades correctamente desde el principio.
- 6) El DCA consigue, por lo general, sistemas de más valor y mejoras en la satisfacción de los usuarios/directivos. Hace aumentar la confianza de los usuarios y los directivos y su apoyo al proyecto.
- 7) El DCA consigue normalmente sistemas menos costosos de mantener, dado que su primera versión ya satisface las necesidades.

Para tener éxito, el DCA impone algunos requisitos previos. Primero, los directivos deben mostrar la voluntad de liberar a sus subordinados de su trabajo cotidiano (o retribuir las posibles horas extras) para que tomen parte en las sesiones. Segundo, los directivos deben ofrecerse ellos mismos para participar en las sesiones. Los directivos han de promover también un ambiente de cooperación y escucha con sus subordinados durante las sesiones. Finalmente, los encargados de dirigir y registrar las sesiones deben tener una formación suficiente para centrar y reordenar las conversaciones, así como mediar en los conflictos y en las disputas.

3.2.3. Las técnicas de prototipos y desarrollo rápido

Técnica de Prototipos:

- Los **prototipos de viabilidad** se utilizan para probar la viabilidad de una tecnología específica aplicable a un sistema de información. Sobre la base de los resultados del prototipo, puede adoptarse una decisión sobre si se sigue adelante o no con el desarrollo de la solución técnica.
- Los **prototipos de necesidades** (a veces denominados prototipos de descubrimiento) se utilizan para “descubrir” las necesidades de

los usuarios con respecto a la empresa. Pretenden simular la forma de pensar de los usuarios. Su base es sencilla: los usuarios reconocerán sus necesidades cuando las vean.

- Los **prototipos de diseño** (también llamados prototipos de comportamiento) se utilizan para simular el diseño del sistema de información final. Mientras que los prototipos de necesidades se centraban sólo en el contenido, los prototipos de diseño lo hacen en la forma y el funcionamiento del sistema deseado.
- Los **prototipos de implantación** (a veces denominados prototipos de producción) constituyen una extensión de los prototipos de diseño donde el prototipo evoluciona directamente hacia el sistema de producción. En principio, los prototipos de implantación omiten normalmente detalles como la edición de datos, las seguridades y los mensajes de ayuda. Estos detalles pueden añadirse posteriormente, si se desea que el prototipo evolucione hacia un sistema de producción. Los prototipos de implantación adquirieron notoriedad con el auge del uso de lenguajes de cuarta generación (L4G) y generadores de aplicaciones. Estos lenguajes de base de datos y generadores suministran potentes herramientas para la rápida generación de prototipos de archivos, base de datos, pantallas, informes y similares.

Las ventajas que se obtienen del uso de prototipos son numerosas:

- 1) Los usuarios se hacen participantes más activos en el desarrollo del sistema. Suelen mostrarse más interesados en los prototipos de trabajo que en las especificaciones de diseño en papel.
- 2) La definición de necesidades se simplifica por el hecho de que muchos usuarios finales no comprenden o no son capaces de enunciar detalladamente sus necesidades hasta que ven un prototipo.
- 3) La probabilidad de que los usuarios finales aprueben un diseño y luego rechacen su implantación se reducirá notablemente.
- 4) Según se dice, el diseño mediante prototipos reduce el tiempo de desarrollo; sin embargo, algunos expertos cuestionan este ahorro de tiempo.

Desventajas del uso de prototipos:

- 1) Los prototipos suelen pasar a las fases de análisis y diseño con demasiada rapidez. Ello empuja al analista a pasar demasiado rápido a la codificación, sin haber comprendido bien las necesidades y los problemas.
- 2) Los prototipos pueden desalentar la consideración de soluciones técnicas alternativas. El analista suele seguir adelante con el primer prototipo alternativo que suscita una reacción razonablemente favorable en los usuarios.
- 3) Los sistemas implantados desde prototipos son con frecuencia poco flexibles para adaptarse a los cambios en las necesidades, y que han sido desarrollados "por encima".
- 4) No siempre es fácil cambiar los prototipos. Varios expertos han advertido un crecimiento en las bibliotecas de código L4G con deficiencias de diseño no estructurado, imposible de leer e inadecuadamente documentado.
- 5) Los prototipos rara vez se pulen. La tecnología utilizada puede impedir, en la práctica, su comprensión por el usuario y, en consecuencia, desalentar su participación.

Los prototipos deben actuar de complemento a las técnicas de especificación apropiadas. La técnica de **desarrollo rápido de aplicaciones** da respuesta a este problema.

Técnica de Desarrollo Rápido de Aplicaciones:

El **desarrollo rápido de aplicaciones (DRA)** es una combinación de diversas técnicas estructuradas (en especial, la ingeniería de información basada en datos) con técnicas de prototipos y de desarrollo, conjunto de aplicaciones cuyo fin es acelerar el desarrollo de sistemas. La técnica DRA requiere el uso interactivo de técnicas estructuradas y prototipos para definir las necesidades de usuario y diseñar el sistema final. Por el uso de técnicas estructuradas, el equipo de desarrollo primero construye los modelos previos de datos y de procesos de las necesidades. Los prototipos ayudan entonces al analista y a los usuarios a verificar dichas necesidades, y a pulir formalmente los modelos de procesos y de datos. El ciclo de hacer modelos, luego prototipos, luego modelos, más tarde prototipos, y así sucesivamente, produce en último término una relación combinada de necesidades de empresa y de diseño técnico.

3.3 ¿ QUÉ METODOLOGÍA Y TÉCNICAS ELEGIR ?

➤ CICLO DE VIDA CLÁSICO

El ciclo de vida clásico proporciona un modelo que sirve para dos propósitos; el primero, permite representar los procesos de concepción y producción en una forma gráfica y lógica; y el segundo, proporciona un marco de trabajo alrededor del cual las actividades de aseguramiento de calidad pueden ser construidas en una manera disciplinada.

➤ CONSTRUCCIÓN DE PROTOTIPOS

La construcción de prototipos como metodología, puede ser problemática porque el cliente ve funcionando lo que parece ser la primera versión del sistema e ignora que no se han considerado todos los aspectos de calidad o de mantenimiento a largo plazo. También puede suceder que, para crear el prototipo se utilice un Sistema Operativo o un Lenguaje de Programación inapropiados, simplemente porque ya está disponible y es conocido; puede que se implemente ineficientemente un algoritmo, sencillamente para demostrar su capacidad.

Aunque pueden aparecer problemas, la construcción de prototipos es una metodología efectiva para el desarrollo de sistemas (ingeniería de software). La clave está en definir al comienzo las reglas del juego; esto es, el cliente y el técnico deben estar de acuerdo en que el prototipo se construya para servir sólo como un mecanismo de definición de requisitos. Posteriormente, ha de ser descartado y debe construirse el sistema real, con todos los aspectos de calidad y mantenimiento.

➤ MODELO EN ESPIRAL O EVOLUTIVO

El modelo en espiral es actualmente el enfoque más realista para el desarrollo de software y de sistemas a gran escala. Utiliza un enfoque “evolutivo”, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo. Utiliza la creación de prototipos como un mecanismo de reducción del riesgo, pero, lo que es más importante, permite a quien lo desarrolla aplicar el enfoque de creación de prototipos en cualquier etapa de la evolución del producto. Mantiene el enfoque sistemático correspondiente a los pasos sugeridos por el ciclo de vida clásico, pero incorporándolo dentro de un marco de trabajo interactivo.

Pero, por otro lado, puede ser difícil convencer a grandes clientes de que el enfoque evolutivo es controlable. Requiere una considerable habilidad para la valoración del riesgo. Si no se descubre un riesgo importante, indudablemente surgirán problemas. Por último, el modelo en sí mismo es relativamente nuevo y no se ha usado tanto como el ciclo de vida o la creación de prototipos.

Metodología Elegida:

A continuación seleccionaremos, de entre las metodologías más utilizadas, la que implementaremos en nuestro sistema

De acuerdo a lo antes escrito, para el desarrollo del presente proyecto, nos apoyaremos fundamentalmente en el **ciclo de vida clásico** (Fig. 3.1), la cual es una metodología ampliamente usada y que tiene un lugar definido e importante dentro del trabajo realizado en el desarrollo de sistemas, suministra una plantilla en la que pueden colocarse los métodos para el análisis, diseño, codificación, prueba y mantenimiento.

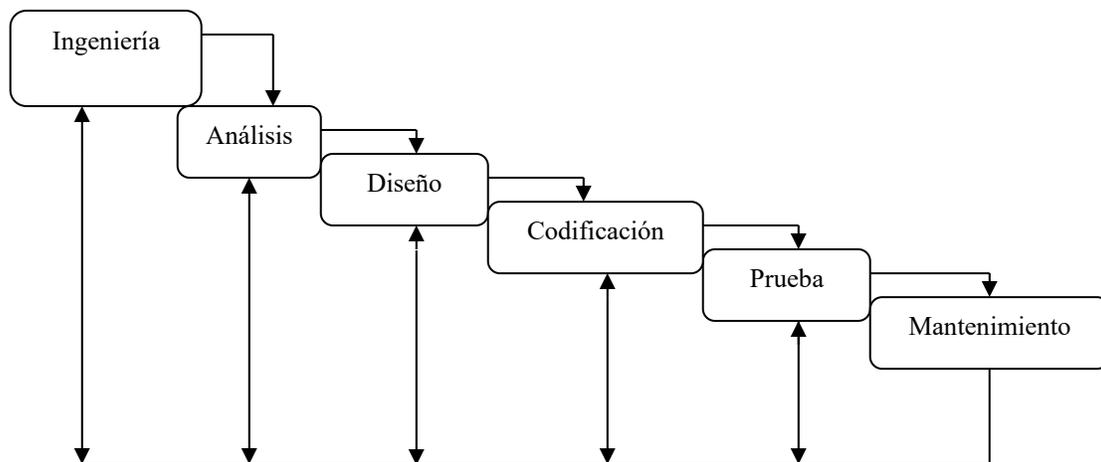


Fig. 3.1 El ciclo de vida clásico

Además, los pasos de esta metodología son muy similares a los pasos genéricos aplicables a todas las metodologías de desarrollo de sistemas.

Técnicas de Desarrollo de Sistemas Elegidas:

Tenemos que las **técnicas de diseño estructurado** y **análisis moderno de Yourdon** (orientadas a los procesos) y la **técnica de modelización de datos** (orientada a los datos), son las más conocidas y practicadas, por lo tanto; nuestro proyecto se centrará en las antes mencionadas.

Estas técnicas nos permitirán la construcción correcta de un sistema, mejorarán la calidad del mismo, reducirán costos, etc. Desde el punto de vista de eficiencia y de acuerdo a la problemática que se ha planteado, nos resultan las más adecuadas para implementar nuestro sistema. En el próximo capítulo se estudiarán éstas.

En la mayoría de los casos, las técnicas pueden y deben combinarse de forma que sea posible utilizar las ventajas de cada una en un mismo proyecto, y cumplir de igual manera con el ciclo de vida completo del desarrollo de sistemas.

IV. MÉTODO DE DISEÑO ESTRUCTURADO DE YOURDON

En este capítulo se analizarán a detalle las técnicas de Yourdon. Dentro del análisis y el diseño, el enfoque estructurado permite modelar los componentes del sistema por medio de símbolos gráficos, estableciendo una atención especial en los aspectos lógicos del sistema. Este enfoque formula las especificaciones funcionales para los módulos del software. La técnica estructurada de Yourdon, aporta una teoría que recoge todos los avances de la programación estructurada que durante décadas se ha venido instrumentando. Entre las características del proceso se encuentran las siguientes.

- Modelización de procesos.
- Modelización de datos.
- Etapas del ciclo de vida.

En la técnica de Yourdon el análisis y el diseño de un sistema se dividen en diversas etapas. Si examinamos su estructura se puede observar una mayor precisión en cuanto a los pasos a seguir a diferencia de otros métodos. A lo largo de sus obras Yourdon describe técnicas para la realización del análisis y el diseño estructurado de sistemas basado principalmente en los siguientes conceptos:

- Diagramas de flujo de datos para la representación de procesos.
- Diccionario de datos como base o soporte de información del sistema.
- Diagramas de transición de estados para la representación estructurada de las funciones a realizar en los procesos.
- Diagramas o mapas de estructura para la representación modular de los procesos y las variables intercambiadas entre ellos.
- Módulo Entidad–Relación para la representación conceptual de los datos.

Yourdon tiene el mérito de describirnos un proceso sistematizado y a la vez flexible en cuanto a la aplicación de las técnicas estructuradas. Particularmente las técnicas utilizadas por Yourdon para el desarrollo de cualquier sistema las explicaremos a continuación.

4.1. MARCO PARA LA MODELIZACIÓN DE PROCESOS

Es la etapa de análisis y especificación estructurada, examina los problemas mediante el modelo entrada-proceso-salida, es decir, mediante el establecimiento del flujo de la información. Esta etapa crea una representación del problema real con el objeto de establecer una solución apoyándose en herramientas como: el diagrama de flujo de datos, el diagrama de transición de estados las mínimas especificaciones y las tablas de decisión; herramientas que serán explicadas a continuación.

4.1.1. Diagrama de flujo de datos

El enfoque de flujo de datos permite crear una descripción grafica del problema, esta descripción se llama diagrama de flujo de datos (DFD) y establece los caminos que seguirán los datos a través de los procesos del sistema a desarrollar. Vamos a ver los formulismos de representación de los conceptos que adopto Yourdon.

Al utilizar el DFD es posible beneficiarse de tres maneras:

1. Contar con rapidez de una implantación técnica del sistema.
2. Comprender la relación existente entre los procesos del sistema.
3. La comunicación a los usuarios del estado actual del sistema mediante los DFD's

Para realizar un DFD Yourdon define la simbología que deberá ser usada para facilitar la comprensión del DFD de un sistema, *ver la Fig. 4.1*



Fig. 4.1

La entidad. *Representa algo que da y recibe datos del sistema; cada entidad se identifica por medio de un nombre apropiado.*

Flujo de Datos. *Representa el movimiento de los datos, con la punta se señala el destino de los mismos, un flujo de información bidireccional puede representarse por medio de dos flechas paralelas. Cada flecha también se define con un nombre apropiado.*

Proceso. *Indica la existencia de una acción de transformación. Los procesos siempre denotan una transformación de los datos, por lo tanto, el flujo de información que sale de un proceso, siempre tendrá un nombre diferente al que tenía al entrar.*

Almacén de Datos. *Representa el almacenamiento de la información. Se permiten anotaciones entre ambas líneas verticales del símbolo correspondiente. Es conveniente señalar que los DFD's pueden particionarse para mostrar mayor detalle.*

4.1.1.1. Características del Diagrama de Flujo de Datos.

Para la elaboración eficaz de un DFD se debe entender las siguientes características de un sistema informático.

1. Desarrollar el DFD mediante el enfoque descendente (top-down).
 - Hacer una lista de las entidades, los flujos de datos, los procesos y los almacenes de datos; esto determina los límites del sistema que se desarrolla.
 - Dibujar un DFD básico que ilustre exclusivamente los aspectos generales.
2. Cubrir los detalles.
3. Dibujar de nuevo los diagramas y volver a definir todos los símbolos por medio de nombres significativos.

A continuación los puntos mencionados anteriormente se verán a detalle.

a) Enfoque descendente.

Se debe iniciar un DFD a partir de la narración del sistema organizacional haciendo uso de cuatro categorías: entidad, flujo de datos, procesos y almacén de datos. Cuando se haya compilado la lista se puede elaborar un diagrama inicial.

El primer diagrama (nivel cero), incluirá lo básico de las entradas, los procesos y las salidas. Este será el diagrama más general.

b) Cubrir los detalles.

Posteriormente los DFD deberán incluir los detalles de cada uno de los procesos. Pero en el primer diagrama se debe definir las entradas y las salidas, estas se mantienen constantes en los diagramas consecutivos. El manejo de las excepciones se ignora en los primeros dos o tres niveles del dibujo del DFD.

El resto del diagrama original se descompone en diagramas que detallan los procesos, agregando en cada nivel inferior nuevos almacenes de datos y nuevos flujos de datos

c) Dibujar de nuevo los diagramas.

Después de que los DFD se aclaran, lo que sigue es volver a dibujarlos y rotularlos de manera significativa. Es decir, los rótulos no deberán ser muy genéricos. También se debe pensar en que las personas poco familiarizadas con el sistema sean capaces de comprender su contenido.

La efectividad de los DFD's se deben a su consistencia: cada vez que sea posible hay que unificar términos.

Los diagramas de flujo se emplean para el análisis y diseño de procesos. Utilizando un DFD desde el principio se pueden establecer los requisitos de la información y de esta manera, obtener una visión global del flujo de los datos del sistema. En este punto, bastara con dibujar los DFD's a mano y de manera somera.

Al detallar los DFD's, estos pueden ser útiles para la interacción con los usuarios puesto que ellos pudieran opinar sobre la realización del proceso y así

obtener una mayor precisión en los diagramas. Al final, los DFD's sirven para la documentación del sistema ya que ayudan a concebir la lógica de los flujos de datos de la organización. Para evitar el dibujo manual de los DFD's se pueden emplear herramientas integradas o instrumentos CASE (**Computer Assisted Software Engineering**).

4.1.2. Diccionario de Datos.

El diccionario de datos es una referencia útil para localizar nombre y atributos de los datos utilizados en el sistema, este deberá actualizarse cada vez que se hagan cambios en el sistema.

Los registros del diccionario de datos deberán contener información referente a:

- El nombre y el alias del dato.
- Las descripciones del dato.
- Los flujos de datos que se relacionan con el término.
- La longitud disponible en caracteres.
- Cualquier otra información pertinente.

- 1. Nombres y Alias.** Es necesario incluir la manera más común de denominar el dato en los programas del sistema, además de su alias. El registro de estos datos, facilita la comunicación y la referencia cruzada entre el sistema.
- 2. Descripción.** La descripción incluirá texto breve, pero informativo del dato elemental.
- 3. Flujos de datos.** Establecer el origen y el destino de la transferencia de datos.
- 4. Longitud del dato.** En el diccionario de datos también se establece la longitud permitida para el acceso de un dato. La longitud simple se da en función del número de caracteres impresos.
- 5. Información adicional.** Deberá agregarse al diccionario de datos cualquier información útil extra.

4.2. PROPIEDADES

El diseño estructurado de Yourdon pretende dividir un programa en un conjunto jerárquico de módulos en sentido descendente, dotado de las siguientes propiedades:

a) Modularidad.

Este concepto divide al software en nombres y ubicaciones determinadas, denominados módulos y que se integran para satisfacer requisitos del problema. Es decir, los módulos son los componentes procedimentales del sistema.

b) Cohesión.

Los módulos deben de tener una fuerte de cohesión; es decir la cohesión se refiere a **la fuerza de unión entre las funciones que realiza un modulo** y se mide considerando una escala que va desde la más débil hasta la más fuerte o deseada, en el siguiente orden.

1. Cohesión coincidental. Ocurre cuando los elementos dentro de un modulo, *no tienen relación aparente entre cada uno de ellos*; esto puede suceder cuando se segmenta arbitrariamente un programa en varios módulos.
2. Cohesión lógica. Esta cohesión *implica algunas relaciones entre los elementos de un módulo*. Un módulo unido lógicamente por lo común combina varias funciones relacionadas en una forma compleja e interrelacional. Un Ejemplo de este módulo es el que desempeña todas las funciones de entrada y salida del sistema.
3. Cohesión temporal. Se presenta cuando todos los elementos son ejecutados en un momento dado sin requerir ningún parámetro o lógica alguna para determinar que elemento debe ejecutarse. En ejemplo de esta cohesión se observa en un modulo dedicado a la iniciación del sistema.
4. Cohesión en la comunicación. Se refieren al conjunto de datos de entrada y salida. Un ejemplo de esta cohesión es la instrucción imprimir el archivo de salida, ya que esta instrucción establece una cohesión en la comunicación.
5. Cohesión secuencial. Se presenta cuando la salida de un elemento es la entrada para el siguiente. Por ejemplo. "Lea clave y contraseña del usuario y busque si es que existe el registro leído".

6. Cohesión funcional. Este tipo de cohesión representa un tipo fuerte y por lo tanto deseable de la relación entre los elementos de un modulo, debido a que todos ellos se encuentran encausados al desempeño de una sola función. Un ejemplo de este tipo es el que ejecuta la instrucción es “Escriba el registro en el archivo de salida”
7. Cohesión informacional. Ocurre cuando un módulo tiene una estructura de datos compleja, así como varias rutinas que manejen dicha estructura; cada rutina del modulo representa unión funcional; esta cohesión es la realización total de la abstracción de los datos. La cohesión informacional solamente requiere que el segmento con cohesión funcional sea ejecutado al ser llamado el modulo.

No es indispensable siempre tener el nivel mas alto de cohesión, sin embargo, lo importante es conseguir una cohesión alta y saber reconocer la cohesión baja, de forma que sea posible modificar el diseño del software para que disponga de una mayor dependencia funcional, de tal forma que los módulos sean reutilizables en futuros programas.

c) Acoplamiento.

Los módulos deben estar debidamente acoplados; es decir han de tener una dependencia mínima unos de otros. Esto reduce el mínimo efecto que futuros cambios en un módulo puedan producir en otros módulos.

El acoplamiento es una medida de la interconexión entre los módulos de un sistema. El acoplamiento depende de la complejidad de las interfases entre los módulos, del punto en el que se hace una entrada o referencia a un modulo y de los datos que pasan a través de estas interfases. En el diseño del software se busca el más bajo acoplamiento posible. La conectividad sencilla entre módulos da como resultado un software que es más fácil de comprender y menos propenso a errores que aparecen en una posición y se propagan a lo largo del sistema. El acoplamiento entre módulos puede ser considerado dentro de una escala del mas fuerte al más débil (que es le más deseable) de la siguiente manera.

- a) Acoplamiento del contenido. Ocurre cuando un modulo modifica los valores o las instrucciones de algún otro modulo; este acoplamiento es posible en los programas en lenguaje ensamblador.
- b) Acoplamiento en zonas compartidas. Es cuando, los módulos son conjuntados por medio de zonas globales para la estructura de datos, por ejemplo, cuando un programa se referencia a un bloque común de datos.

- c) Acoplamiento de control. Este acoplamiento incluye el paso de banderas de control, ya sea como parámetros o en forma global entre los módulos, de tal forma que un módulo controla la secuencia de proceso de otro.
- d) Acoplamiento por zona de datos. Este tipo de acoplamiento es similar a las zonas compartidas excepto que los elementos globales son compartidos en forma selectiva entre los diversos módulos que requieren los datos.
- e) Acoplamiento de datos. Incluye el uso de listas de parámetros para pasar los elementos entre los módulos.

4.3. MARCO PARA LA MODELIZACIÓN DE DATOS

Una vez que se analizó el marco de modelización de procesos, pasaremos a analizar la modelización de datos. Esta técnica representa las necesidades del usuario para un sistema en función de los datos del sistema, independientemente de cómo se procesen o se utilicen dichos datos para producir información. Las técnicas de modelización de datos representan al bloque elemental de datos.

4.3.1. Base de datos.

Es posible considerar dos tipos de almacenamiento de datos en un sistema de información basado en computadora: archivos individuales o elaborar una base de datos. Un sistema que hace uso de los archivos convencionales implica que los datos almacenados lleguen a ser redundantes, la actualización de los archivos lleva más tiempo y existirán problemas de integración de datos, ya que los cambios en un archivo, requieran la modificación de ciertos datos en otros archivos.

Antes de ver la utilidad de una base de datos, tendremos que definirla.

Base de datos. Es el conjunto de datos almacenados con una estructura lógica relacionada entre sí y que puede ser consultada y actualizada. Los datos están clasificados en campos y registros.

Por otro lado, una base de datos es una fuente central de datos significativos que pueden ser compartidos por varios usuarios y por varias aplicaciones y asegura que los datos requeridos estén siempre disponibles, además permite el mantenimiento preciso y consistente de los mismos. Es

oportuno en este momento establecer los conceptos básicos necesarios en el manejo de una base de datos.

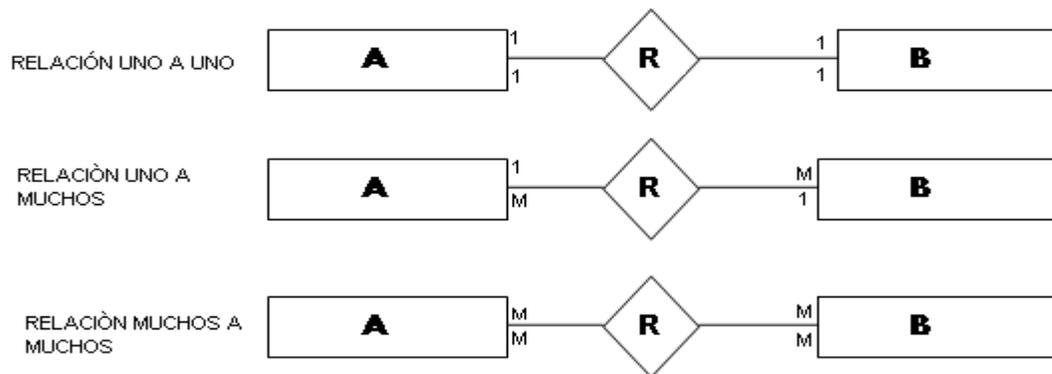
Entidades. Una entidad es cualquier objeto o evento que sea fuente de datos.

Relaciones. Las relaciones son asociaciones de datos entre las entidades. Existen tres tipos de relaciones.

Relación de uno a uno (1:1). Para una entidad existe otra sola entidad.

Relación de uno a muchos (1:M), para una entidad existen otras muchas entidades.

Relación de muchos a muchos (M:M), para muchas entidades existen otras varias entidades como se ve en la Fig. 4.2



Atributos. Un atributo es una característica de una entidad y pueden describirse varios atributos para una entidad.

Ejemplo. Nombre. CURP, Domicilio, etc.

Registros. Un registro es una colección de datos elementales que tienen algo en común con la entidad correspondiente. Los datos son las unidades más pequeñas en una base de datos. La palabra dato, puede aquí considerarse como un sinónimo de atributo, los tipos de datos pueden ser alfabéticos (todos ellos de longitud variable).

Ocasionalmente un dato puede ser llamado como un campo, sin embargo un campo representa algo físico y no lógico, por lo tanto esta nominación es incorrecta; además, un grupo de datos puede con juntarse en un campo.

Ejemplo. El campo fecha contiene los datos día, mes y año.

Un registro puede esquematizarse de la siguiente manera (figura 4.3).

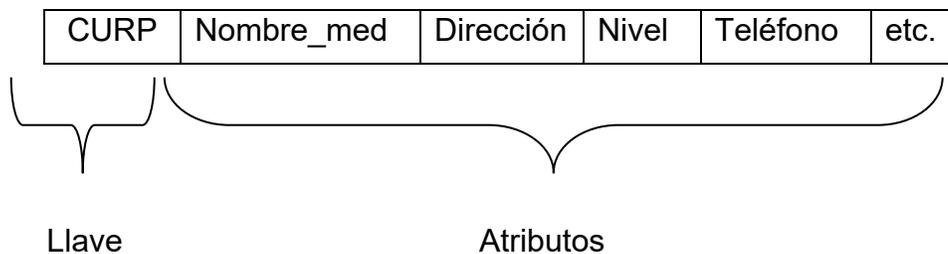


Fig. 4.3

Tabla. Una tabla es una colección de datos presentada en forma de matriz bidireccional, donde las filas son registros y las columnas campos.

Llaves. Una llave es un dato en un registro que se emplea como identificador para este. Existen dos tipos de llaves.

- Llave primaria
- Llave Foránea.

Llave primaria. Columna o grupo de columnas que identifican de manera exclusiva a un registro. La llave primaria debe cumplir con:

- No puede aceptar valores nulos.
- No pueden existir dos renglones con el mismo valor.
- No puede cambiarse el valor

Tipos de asignación de la llave primaria.

Asignada por el usuario. De esta forma el usuario proporciona el valor para la llave primaria del registro insertado. Por ejemplo el CURP del paciente a consultar.

Asignada por el sistema. De esta forma el sistema proporciona automáticamente el valor para la llave primaria cuando el usuario inserta un registro. Ejemplo los campos autonuméricos.

Todas las tablas deben contener una llave primaria, la cual puede estar compuesta de uno o más campos.

Llave Foránea. Es un atributo de una tabla la cual es llave primaria de otra tabla.

Nulo. Es cuando existe la ausencia de datos.

Consulta. Una consulta es el mecanismo que recoge una pregunta, simple, o compuesta, que se formula a la base de datos con el fin de obtener la información requerida de una o más tablas que cumplan con las condiciones enunciadas.

4.3.2. Tipos de consulta

El almacenamiento empleando bases de datos facilita la emisión de reportes adecuados a los usuarios mediante el empleo de consultas las cuales son preguntas acerca de la información contenida en dicha base. Pueden mencionarse algunos tipos de consultas básicas, relacionando a las entidades, atributos y valores en cada uno de los casos se conocen dos tipos de elementos y se intenta obtener el tercero.

- Tipo 1: Se cuenta con la entidad E y uno de sus atributos A, se desea buscar el valor correspondiente: $V = (E, A)$
- Tipo 2: $E = (V, A)$
- Tipo 3: $A = (V, E)$
- Tipo 4: $V = (E, \text{todos los } A)$
- Tipo 5: $\text{todas las } E = (V, \text{todos los } A)$
- Tipo 6: $\text{todos los } A = (V, \text{todas las } E)$

Empleado operadores aritméticos, comparativos y Booléanos las consultas pueden ser más complejas y precisas respecto a sus tipos básicos.

4.3.3. Modelo Entidad–Relación (E-R)

Para realizar el modelo conceptual de datos, Yourdon propone un esquema de entidades y relaciones del sistema, pasando después a normalizar esta estructura.

Veamos el ejemplo de la relación entre DOCTORES y PACIENTES: Un doctor puede tener varios pacientes y un paciente varios doctores.

Los diagramas E-R son una técnica para representar gráficamente la estructura lógica de una base de datos. Como tal ofrecen una forma sencilla de comunicar los rasgos prominentes del diseño de cualquier base de datos. Básicamente en un diagrama del modelo E-R se deben representar los conceptos siguientes.

Entidad. La componen una serie de datos que agrupados, tienen un cierto significado para la empresa. Se representa por medio de un rectángulo en cuyo interior figura el nombre de la entidad. (Fig. 4.4)

Representación de una Entidad



Fig. 4.4

Cada entidad esta compuesta por una serie de atributos los cuales son representados por círculos enlazados con las entidades.

Relaciones. Constituyen el enlace entre las diferentes entidades del sistema. Se representa mediante líneas que unen a las entidades o a través de un rombo en cuyo interior se coloca el nombre de la relación.

Ejemplo. Así entre un doctor y un paciente existirá la relación medico_pac (Fig. 4.5)

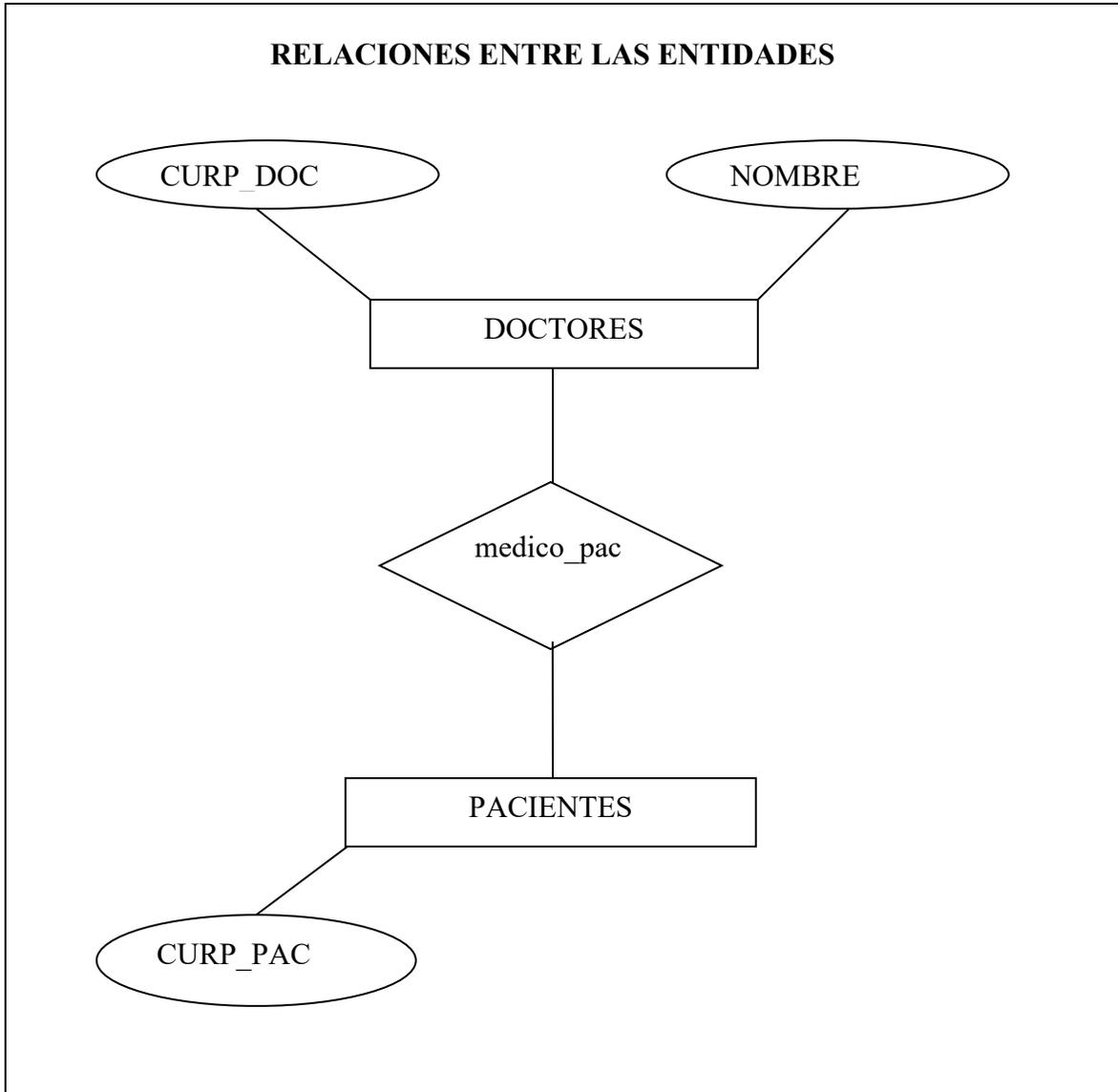


Fig. 4.5

4.3.4. Normalización

La normalización es la simplificación de las relaciones existentes entre los datos de las aplicaciones y los almacenamientos usados por los usuarios; de tal manera, que se establezcan estructuras de datos de menor

tamaño. Se emplean cuatro pasos para simplificación de los datos, llamadas formas normales (FN).

La primera forma normal (FN1) elimina los grupos de datos repetidos e identifica la llave primaria, en este punto puede ser necesario desglosar alguna relación en dos o más relaciones.

La segunda forma normal (FN2) se aplica para asegurar que todos los atributos no llave sean dependientes de la llave primaria. Si existen dependencias se colocan en otra relación.

La tercera forma normal (FN3) elimina cualquier dependencia transitoria (Es cuando los atributos no llave son dependientes de otros atributos no llave)

4.4. ETAPAS DEL CICLO DE VIDA

Ed Yourdon define las siguientes etapas y niveles en el ciclo de vida de los sistemas informáticos (Fig. 4.6)

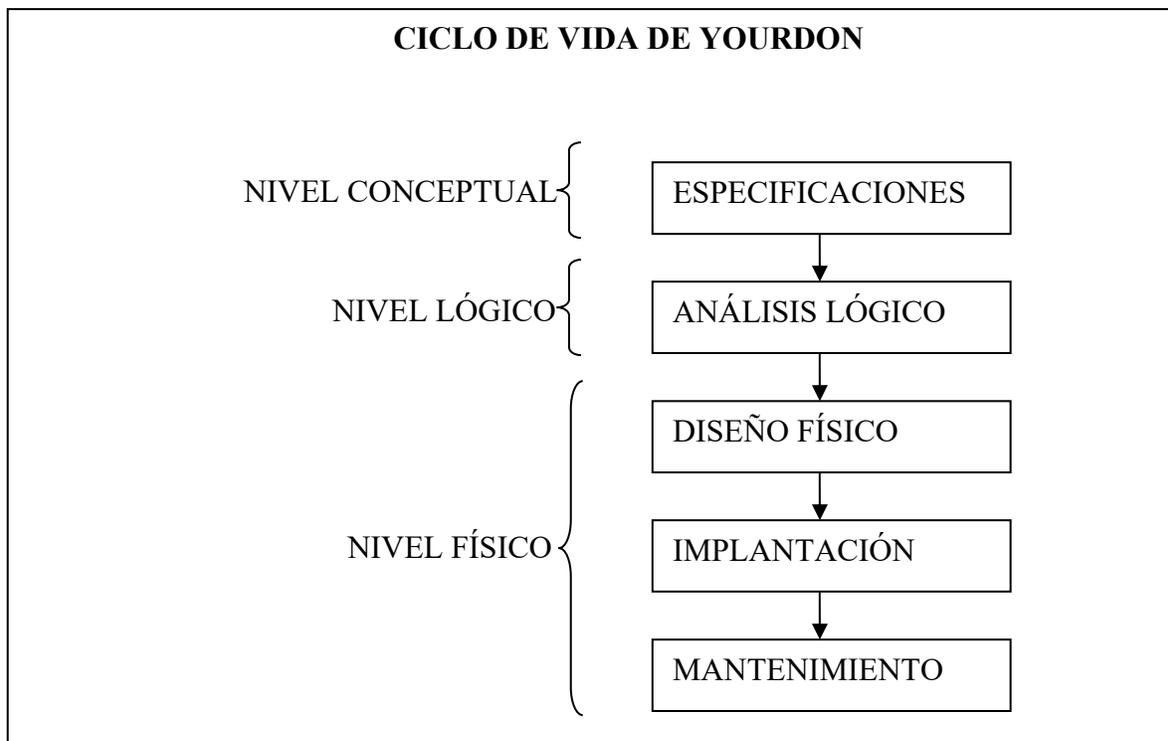


Fig. 4.6

Finalmente, las técnicas descritas a lo largo de estas etapas, se distinguen las siguientes actividades a realizar ya que se han planteado como bases de la programación estructurada, pero que son retomadas por el método de Yourdon.

4.4.1. Estudio de viabilidad

Debe hacer un estudio de la situación actual representando las mismas a través de DFD de primer nivel o diagramas de contexto en los que de forma simple, se indiquen los procesos más relevantes. La descripción de la situación actual del problema en estudio es indispensable para identificar las necesidades a considerar: la información que se desea obtener, la información que se desea obtener, la información que se va a suministrar, las funciones y el rendimiento requeridos. Es útil establecer la diferencia entre lo que “necesita” el cliente y lo que “quiere” el cliente.

En este estudio de la situación actual se identificarán las deficiencias del mismo como puntos a resolver por el nuevo sistema. Esta actividad debe ocupar entre un 5 y un 10% del total del proyecto.

4.4.2. Análisis del Sistema

En esta fase se debe representar mediante las técnicas de diagramas de flujo, modelo Entidad-Relación, etc., el sistema a desarrollar. En esta parte también se debe tener en cuenta los requerimientos de los usuarios relativos a cambios de o funcionalidad del sistema y esta labor es facilitada por las herramientas CASE de diagramación y por técnicas de prototipos de sistema.

4.4.3. Diseño del Sistema

En esta fase, se pasa de un nivel conceptual descrito anteriormente a un nivel de representación lógico de los datos mediante un diseño dependiente del modelo de base elegida, y una estructuración de los procesos utilizando diagramas de estructura de los mismos y generando las especificaciones del sistema correspondiente.

4.4.4. Implementación ó producción

Comprende a la generación de código y el montado e integración de todos los módulos.

4.4.5. Pruebas del Sistema

Las pruebas se tienen que realizar con la totalidad del sistema hasta llegar a la aceptación del mismo por parte del usuario. En esta fase se harán pruebas de integración y de funcionamiento.

La prueba del proyecto es un elemento importantísimo para garantizar la calidad del software y representa una confirmación final de las especificaciones del diseño y de la codificación.

Una prueba de sistema debe considerar los siguientes elementos.

- La prueba es un proceso de ejecución de un programa con la intención de descubrir errores.
- La prueba es adecuada si se cuenta con una alta probabilidad de mostrar un error no descubierto antes.

a) Pruebas de Unidad.

La prueba de unidad, centra el proceso de verificación en la menor unidad del diseño, es decir, en el módulo. Las pruebas que se realizan son de:

- *Interfaz.* Para asegurar que la información fluye de forma adecuada hacia y desde la unidad del programa que esta siendo probada.
- *Estructuras de datos.* Para asegurar que los datos se mantienen temporalmente conservando su integridad durante todos los pasos de ejecución del algoritmo.
- *Condiciones límite.* Para asegurar que el módulo funciona correctamente en los limites establecidos como restricciones de procesamiento.

- *Caminos independientes.* Para asegurar que todas las sentencias del módulo se ejecutan por lo menos una vez.

También deben diseñarse pruebas para detectar errores en las estructuras de datos locales de cada módulo, en las siguientes categorías.

- a) Caracterización impropia o inconsistente.
- b) Iniciación o valores impropios erróneos.
- c) Nombres de variables incorrectos.
- d) Tipos de datos inconsistentes.
- e) Desbordamiento o direccionamiento.

De la misma manera, deben establecerse caminos para detectar errores debidos a cálculos o comparaciones incorrectos o flujos de control inadecuados.

- a) Comparaciones entre tipos de datos distintos
- b) Operadores lógicos o de presencia incorrectos.
- c) Variables o comparadores incorrectos.
- d) Terminación inapropiada o existencia de bucles.

b) Pruebas de integración.

La prueba de integración, se establece para construir la estructura del programa mientras que, simultáneamente se realizan pruebas para detectar errores asociados con la interacción. Para agrupar a los módulos se debe seguir la técnica denominada integración incremental que consiste en probar pequeños segmentos para que los errores sean más fáciles de aislar y de corregir.

c) Pruebas de Volumen.

La prueba de volumen, intenta detectar errores en el software debidos a condiciones anormales, demandando recursos en cantidad, frecuencia o volúmenes anormales. Algunos ejemplos de pruebas para estos casos son:

- Generar diez interrupciones por segundo, cuando las normales son una o dos.
- Incrementar las frecuencias de datos de entrada en magnitud.

- Casos en los que se requieren el máximo de memoria o de otros recursos.
- Casos para probar la memoria virtual.
- Casos de exclusiva búsqueda de datos en disco.

d) Pruebas de Validación.

La validación del software se consigue demostrándose la conformidad con los requisitos, es decir, que el software funciona de acuerdo con las expectativas razonables del cliente.

Cuando se ha realizado el software para un cliente, es necesario realizar la *prueba de validación*, para que el cliente establezca conformidad del sistema con todos los requisitos.

Para realizar esta prueba, es necesario la instalación del producto terminado, posteriormente se lleva acabo la capacitación tanto al personal técnico como a los propios usuarios. Se entiende por producto terminado, al sistema listo para ser empleado en el equipo para el cual fue diseñado, junto con el manual del usuario. El sistema deberá estar listo para comenzar su operación de manera inmediata.

4.4.6. Control de Calidad

La calidad de los sistemas de software se torna cada día más importante debido a que las inversiones en sistemas continúan creciendo tanto en valor absoluto como en valor porcentual. El objetivo de esta actividad es garantizar que el producto final sea de un buen nivel de calidad y que cumpla con los estándares fijados.

La aceptabilidad de los productos de software es de vasto alcance. Su evaluación se basa en el grado de calidad presente tanto en aspectos de productividad como en características del método de desarrollo y del producto mismo.

La tarea de control de calidad no deja de representar una inversión considerable de tiempo y dinero. Sin embargo, los resultados esperados pagan con creces la inversión.

La expectativa es que el esfuerzo extra redunde en:

- Una mayor satisfacción del usuario,
- Una disminución considerable en la cantidad de errores presentes en el producto,
- Una reducción en los costos de mantenimiento debidos a errores.
- Un aumento en la confianza depositada en el equipo de desarrollo,
- Una mayor facilidad en la administración y generación de nuevas versiones.

En concepto de calidad implica que: Los requisitos de software son la base de las medidas de calidad, hay ciertos seguimientos especificados que definen los criterios de desarrollo y existe un conjunto de requisitos implícitos que frecuentemente no se mencionan, como el deseo de un buen mantenimiento.

4.4.7. Mantenimiento

El mantenimiento del software se caracterizaba como un medio por el cual se tenía la posibilidad de reutilizarlo y que tuviera aplicaciones en diferentes áreas. El mantenimiento del software existente puede dar cuenta de más del 60 por ciento de las inversiones efectuadas por una organización de desarrollo, y ese porcentaje sigue ascendiendo a medida que se produce más software. El cambio es algo inevitable cuando se construyen sistemas basados en computadoras; por tanto, debemos desarrollar mecanismos para evaluar, controlar y realizar modificaciones. El mantenimiento del software es algo que va mucho más allá. Se puede definir el mantenimiento describiendo las cuatro actividades que se emprenden cuando se publica un programa para su utilización:

- Mantenimiento correctivo.
- Mantenimiento adaptativo.
- Mejoras o mantenimiento de perfeccionamiento.
- Mantenimiento preventivo o reingeniería.

El mantenimiento correctivo. Modifica de alguna manera el funcionamiento de ciertas partes del programa

El mantenimiento adaptativo, es el que se requiere debido a los cambios en el ambiente del programa.

Mejoras o mantenimiento de perfeccionamiento, significa cambio que mejora al sistema de alguna manera sin cambiar su funcionalidad.

El mantenimiento preventivo o reingeniería. Trate de prever distintos errores que se puedan presentar en el funcionamiento del sistema.

El mantenimiento es una iteración del proceso de desarrollo y se deben aplicar los estándares de comparación. Los nuevos requerimientos deben ser validados, los componentes del sistema deben ser rediseñados e implementados y parte, o la totalidad del sistema, debe ser probada.

V. ANÁLISIS Y DISEÑO ESTRUCTURADO DE YOURDON

5.1. ANÁLISIS DEL SISTEMA

Basados en los antecedentes mostrados en el capítulo I donde se trató el funcionamiento de la Imagenología Clínica y de acuerdo con la problemática tratada en el capítulo II, se realizó una recopilación de información de datos para hacer un análisis de información que debería de manejar el sistema a desarrollar.

La información que el sistema manejara es la siguiente: sistematizar la consulta a los expedientes clínicos referidos con la Imagenología, a fin de enfocar el servicio más eficiente y un menor tiempo de espera.

Además también se debe de contar con un módulo de usuarios (médicos), realizar reportes en pantalla, a disco o bien a impresora, así como contar con un módulo de información común (catálogos).

En base a lo anterior se realizó un diseño general para el manejo y captura de la información, se contemplaron también los distintos tipos usuarios que pueden acceder al sistema, los privilegios o permisos que tendrán, así como la forma de obtener reportes de manera más fácil.

Las dependencias del ISSSTE en el área de Imagenología son ocho: Tomografía, Mastografía, Ultrasonido, Radiología, Resonancia Magnética, Fluoroscopia, Imágenes de Patología, y Fotografías Médicas.

A continuación se describirán y se aplicarán tanto la metodología como la técnica seleccionadas para el desarrollo e implementación del sistema.

5.2. EL DFD DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA

El Diagrama de Flujo de Datos, trata de mostrar de una manera clara los procesos que se realizan dentro del sistema y el almacenamiento de los datos recopilados por el mismo (Fig. 5.1).

DIAGRAMA DE FLUJO DE DATOS DE TELEIMAGENOL O G I A CL I N I C A

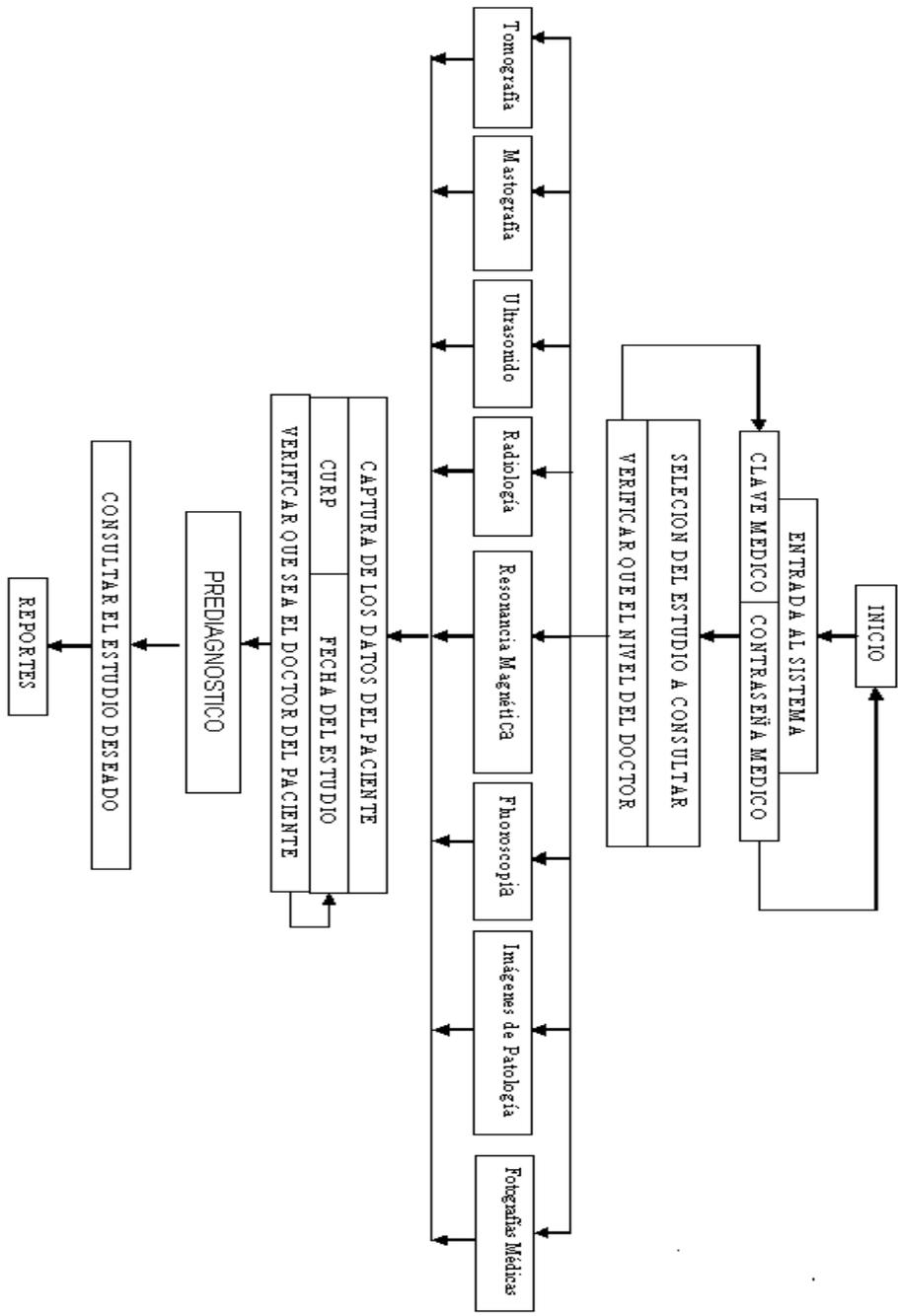


Fig. 5.1

El sistema esta dividido en cuatro bloques:

- Entrada del Médico
- Selección de Estudio a Consultar.
- Captura de los Datos del Paciente.
- Consulta del Estudio Deseado.

Entrada del Médico. El primero nos muestra la parte donde se verifica el tipo de Médico que entrara al sistema y de acuerdo con los permisos que tenga podrá acceder para consultar o actualizar la información. (Fig. 5.2).



Fig. 5.2

Selección de Estudio a Consultar. En este bloque se le indica al sistema el estudio que se desea consultar, para tener un mejor control del sistema y de la información, y se verifica el nivel del Médico con la cual se le indica al sistema si se tiene permisos de ver los estudios clínicos. (Fig. 5.3).

Los niveles de los Médicos se clasifican en:

- **Nivel 1:** Estudios de Rayos X
- **Nivel 2:** Mastografía, Ultrasonido, Fotografías Médicas
- **Nivel 3:** Tomografía, Fluoroscopia, Imágenes de Patología, Resonancia Magnética

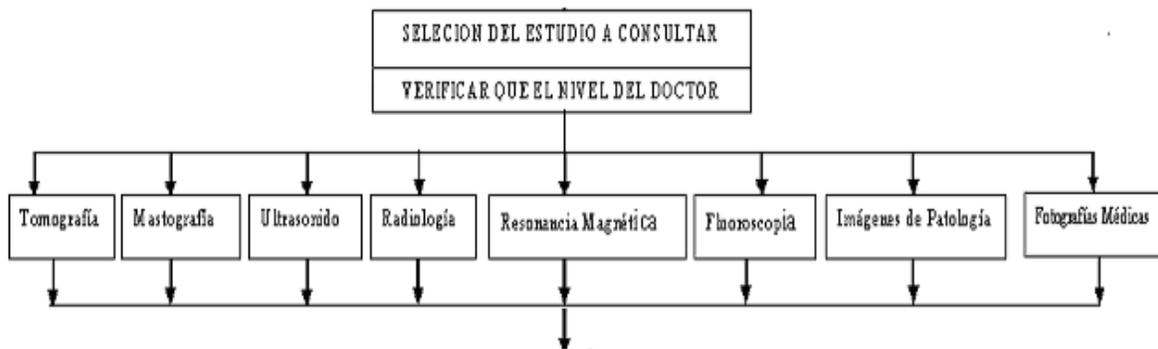


Fig. 5.3

Captura de los datos del paciente. En esta etapa se capturan el CURP del paciente y la fecha en que fue realizado el estudio. En esta etapa también se verifica que el Médico que está solicitando los estudios, sea su paciente. En caso de no serlo se le mandará un aviso y no se podrá acceder a los estudios (Fig. 5.4).



Fig. 5.4

Consulta del Estudio Deseado. Una vez que se pasan las tres etapas anteriores, se llega a esta etapa la cual es el objetivo principal del sistema, ya que nos muestra las imágenes de los estudios clínicos, pasando antes por un prediagnóstico el cual será llenado por el Médico consultante. Para así llegar al reporte impreso. (Fig. 5.5).



Fig. 5.5

5.3. DIAGRAMA ENTIDAD-RELACIÓN (DER)

Retomando los fundamentos teóricos descritos en los capítulos III y IV, y aplicándolas a nuestro sistema tendremos como resultado el Diagrama Entidad - Relación (DER) para el sistema de Teleimagenología Clínica, mostrado en la (Fig. 5.6).

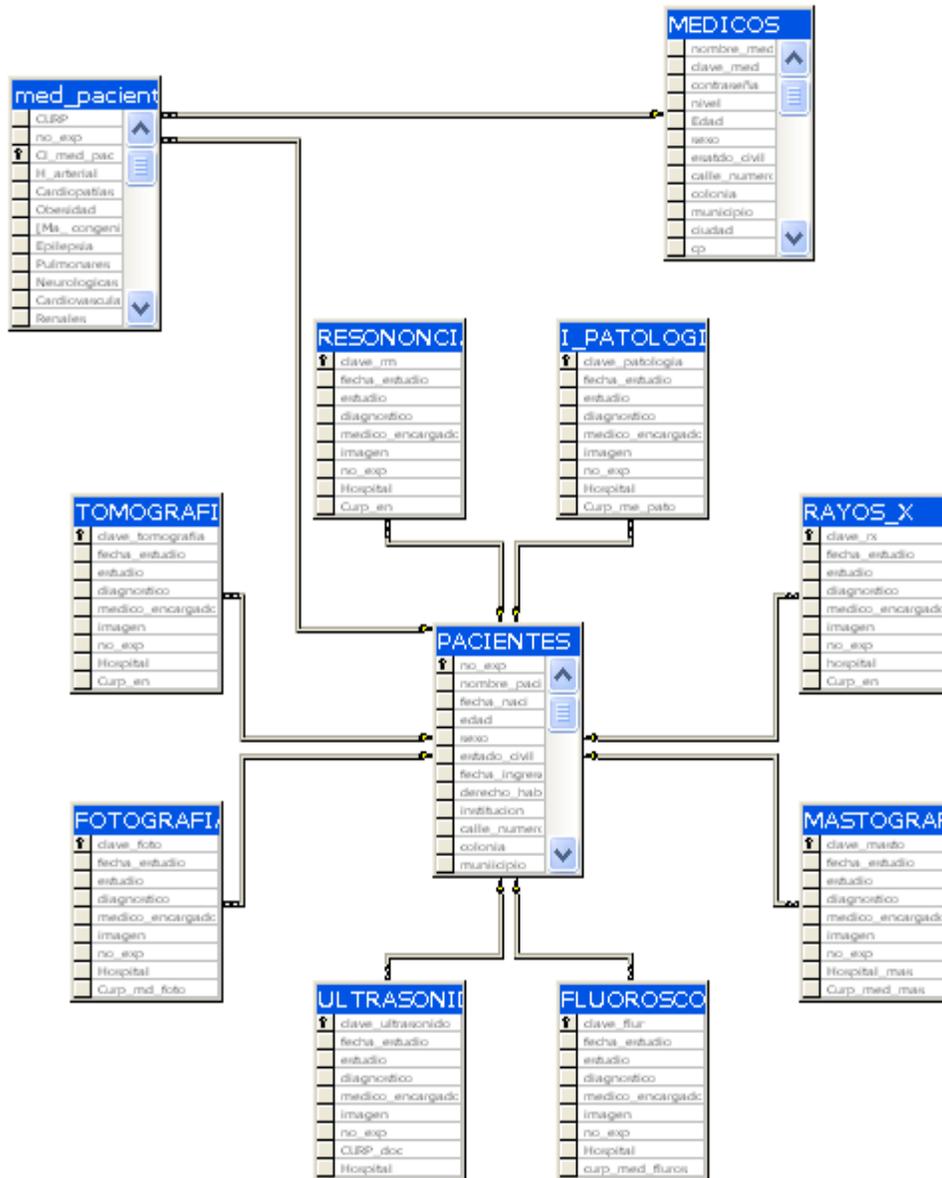


Fig. 5.6

5.4. DICCIONARIO DE DATOS DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA.

El diccionario de datos de la Teleimagenología Clínica recopila los nombres de las tablas principales y atributos de las mismas, así como las características de cada uno de ellos. Además de establecer la relación con cada uno de las dependencias del ISSSTE relacionadas con la Imagenología Clínica, utilizados por el sistema.

Tablas de datos de la Teleimagenología Clínica

A continuación describiremos las tablas encargadas de recibir los distintos tipos de datos que se manejan dentro del sistema. Los datos aquí manejados, como se mencionó anteriormente son pensando en un esquema general y se utilizarán para cualquier dependencia del ISSSTE, que este relacionada con la Imagenología clínica.

Estas tablas están formadas por dos partes, la primera es la de los Pacientes, Usuarios (Médicos) que como ya se menciono son los que podrán entrar al sistema y su relación entre estas tablas, y la segunda son los Estudios Clínicos que se usarán dentro del mismo.

➤ Primera Parte.

MÉDICOS.

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
MÉDICOS	1	PK	CURP_MED	CHAR	18	NO
	2		Nombre_med	VARCHAR	70	NO
	3		Clave_med	VARCHAR	8	NO
	4		Contraseña	VARCHAR	8	NO
	5		Nivel	SMALLINT	2	NO
	6		Edad	NUMERIC	4	SI
	7		Sexo	CHAR	2	SI
	8		Estado_civil	VARCHAR	10	SI
	9		Calle_numero	VARCHAR	60	SI
	10		Colonia	VARCHAR	40	SI
	11		Municipio	VARCHAR	30	SI
	12		Ciudad	VARCHAR	30	SI
	13		Cp	VARCHAR	10	SI
	14		Telefono	VARCHAR	20	SI

PACIENTES

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
PACIENTES	1	PK	NO_EXP	CHAR	18	NO
	2		Nombre_paciente	VARCHAR	70	NO
	3		Fecha_naci	SMALLDATETIME	4	SI
	4		Edad	SMALLINT	2	SI
	5		Sexo	CHAR	2	SI
	6		Estado_civil	VARCHAR	10	SI
	7		Fecha_ingreso	SMALLDATETIME	4	SI
	8		Der_habiencia	BIT	1	NO
	9		Institucion_per	VARCHAR	50	SI
	10		Calle_num	VARCHAR	60	SI
	11		Colonia	VARCHAR	30	SI
	12		Municipio	VARCHAR	30	SI
	13		Ciudad	VARCHAR	30	SI
	14		Cp	VARCHAR	10	SI
	15		Telefono_casa	VARCHAR	50	SI
	16		Telefono_oficina	VARCHAR	50	SI

MED_PACIENTES

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
MED_PACINTES	1	PK*	CURP	CHAR	18	NO
	2	PK*	NO_EXP	CHAR	18	NO
	3		FECHA	SMALLDATETIME	8	NO
	4		H_arterial	BIT	1	SI
	5		Cardiopatias	BIT	1	SI
	6		Obesidad	BIT	1	SI
	7		Ma_congenitas	BIT	1	SI
	8		Epilepsia	BIT	1	SI
	9		Pulmonares	BIT	1	SI
	10		Neurologicas	BIT	1	SI
	11		Cardiovasculares	BIT	1	SI
	12		Renales	BIT	1	SI
	13		Endocrinas	BIT	1	SI
	14		Diabetes	BIT	1	SI
	15		Hematologicas	BIT	1	SI
	16		En_congenitas	BIT	1	SI
	17		Quirurgicos	BIT	1	SI
	18		Traumaticos	BIT	1	SI
	19		Alergicos	BIT	1	SI
	20		Intoxicaciones	BIT	1	SI
	21		Hospitalizaciones	BIT	1	SI
	22		Tabaquismo	BIT	1	SI
	23		Alcoholismo	BIT	1	SI
	24		Toxicomanias	BIT	1	SI
	25		Ex_trabajo	BIT	1	SI
	26		In_sueros	BIT	1	SI
	27		Prótesis	BIT	1	SI

28	Llenada	BIT	1	SI
29	Recomendaciones	NTEXT	16	SI

➤ Segunda Parte.

FLUOROSCOPIA

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
FLUOROSCOPIA	1	PK	CLAVE_FLUR	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_flu	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

FOTOGRAFÍAS MÉDICAS

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
FOTOGRAFÍAS MÉDICAS	1	PK	CLAVE_FOTO	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_foto	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

IMÁGENES DE PATOLOGÍA

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
IMÁGENES DE PATOLOGÍA	1	PK	CLAVE_PATOLOGIA	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_pato	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

MASTOGRAFÍA

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
MASTOGRAFÍA	1	PK	CLAVE_MASTO	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_mas	VARCHAR	30	NO
	8		Curp_md_mas	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

RAYOS X

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
RAYOS X	1	PK	CLAVE_RX	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_rx	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

RESONANCIA MAGNÉTICA

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
RESONANCIA MAGNÉTICA	1	PK	CLAVE_RM	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_rm	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

TOMOGRAFÍA

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
TOMOGRAFÍA	1	PK	CLAVE_TOMOGRAFIA	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_tomo	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

ULTRASONIDO

TABLA	CAMPO	LLAVE	NOMBRE ATRIBUTO	TIPO	LONGITUD	NULO
ULTRASONIDO	1	PK	CLAVE_ULTRASONIDO	INT	4	NO
	2		Fecha_estudio	SMALLDATETIME	4	NO
	3		Estudio	VARCHAR	100	NO
	4		Diagnostico	NTEXT	16	NO
	5		Médico_encargado	VARCHAR	70	NO
	6		Imagen	IMAGE	16	NO
	7		Hospital_res	VARCHAR	30	NO
	8		Curp_md_ultra	CHAR	18	NO
	9	PK*	No_exp	CHAR	18	NO

5.5. DISEÑO DE MÓDULOS DEL SISTEMA DE TELEIMAGENOLÓGÍA CLÍNICA

Para el desarrollo del diseño de módulos lo dividiremos en cuatro bloques:

- Entrada del Médico (Acceso al sistema).
- Selección de Estudio a Consultar.
- Captura de los Datos del Paciente.
- Consulta del Estudio Deseado.

1. Acceso al sistema.

En este primer bloque manejamos el acceso ya que el Médico tendrá que introducir una clave y una contraseña, y en base a esto tendrá ciertos privilegios, ya que serán almacenados y cada que se requieran pueden ser consultados para permitir el acceso o negarlo a la entrada de un bloque posterior.

En esta parte capturamos sus permisos y la dependencia a la cual pertenece. Los permisos de un usuario varían desde el nivel que el Médico tenga para poder ver un estudio Médico, hasta si es su paciente del estudio requerido, si el Médico no concluye de introducir los datos en dos minutos el sistema mandara un aviso y se cerrara el programa (Fig. 5.7).

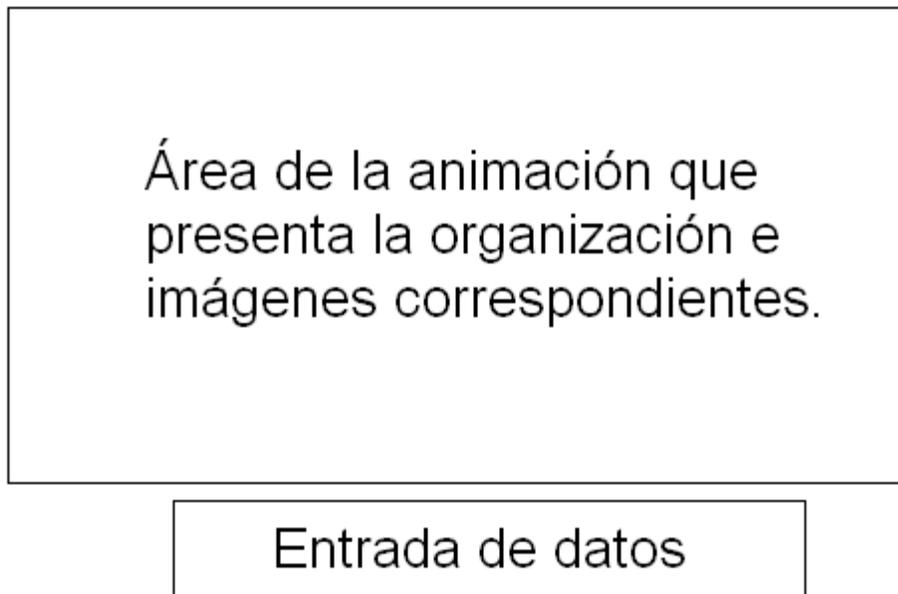


Fig. 5.7

2. Selección de Estudio a Consultar.

Este módulo contiene los diferentes tipos de estudios que maneja el sistema, el Médico puede tener acceso a ellos dando un click sobre el de su preferencia, el sistema validara el nivel del Médico para ver si tiene los permisos suficientes para ver dicho sistema, en caso contrario el sistema mandara un aviso y tendrá que seleccionar un estudio correspondiente a su nivel correspondiente. (Fig. 5.8).

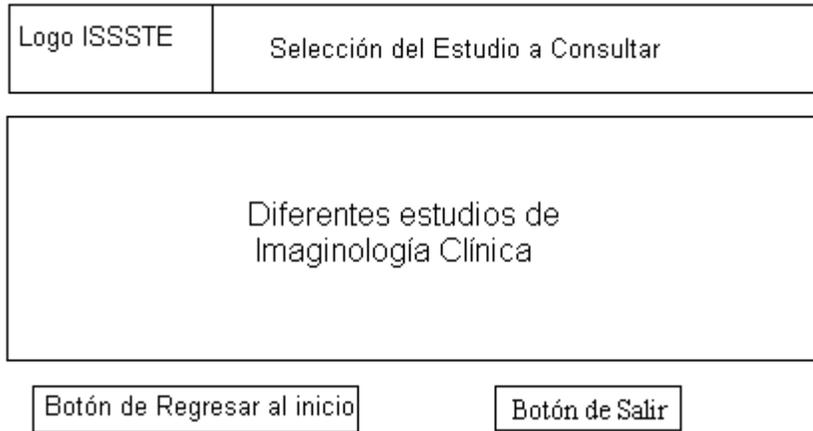


Fig. 5.8

3. Captura de los datos del paciente.

Ahora bien en el módulo de Captura de los datos del paciente tenemos una parte en donde el Médico le dará los datos del paciente al sistema, el sistema verificara que el doctor consultante sea su Médico del paciente, en caso de serlo pasara al módulo cuatro, y en caso contrario le mandara un aviso y no le permitirá el acceso al módulo siguiente.

La parte superior se encuentra el logo del ISSSTE, en medio se encuentran las casillas para la captura de los datos del paciente y en la parte inferior tendremos un botón de regresar al menú de estudio y otro de buscar al paciente requerido (Fig. 5.9).

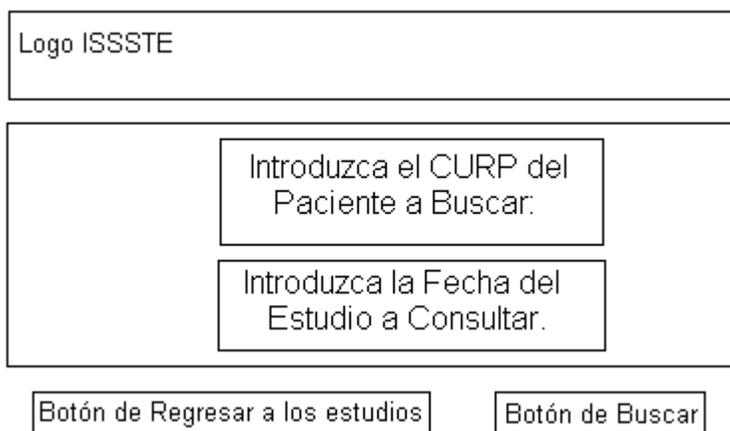


Fig. 5.9

4. Consulta del Estudio Deseado.

Finalmente en el módulo de Consulta del estudio deseado se compone de dos fases, la primera es un prediagnóstico de donde el Médico consultante llenara los campos para así tener un reporte mas completo, en la segunda parte muestra el estudio deseado el cual se forma de siete componentes los cuales son: (Fig. 5.10)

1. En menú principal, el cual tiene los campos de Archivo Opciones y Acerca.

- ❖ Archivo. Cuenta con las opciones de:
 - Imprimir el reporte impreso.
 - Ver el reporte.
 - Salir del Sistema.
- ❖ Opciones.
 - Regresar al Menú de los Estudios.
 - Buscar otro Expediente.
- ❖ Acerca.
 - Muestra los Datos del Sistema.

2. Datos del paciente.

- ❖ Nombre del paciente.
- ❖ Número de Expediente.
- ❖ Edad.
- ❖ Sexo.

3. Estudio. El cual indica el estudio realizado.

4. Diagnostico. El cual tiene el diagnóstico indicado por el Médico a distancia.

5. Médico Encargado. Indica quien fue el Médico que dictamino el diagnóstico del punto 4.
6. Imagen. Muestra la imagen del estudio realizado.
7. Botones.
 - ❖ Botón de Buscar otro Paciente.
 - ❖ Botón de Regresar al Menú de los Estudios Clínicos.

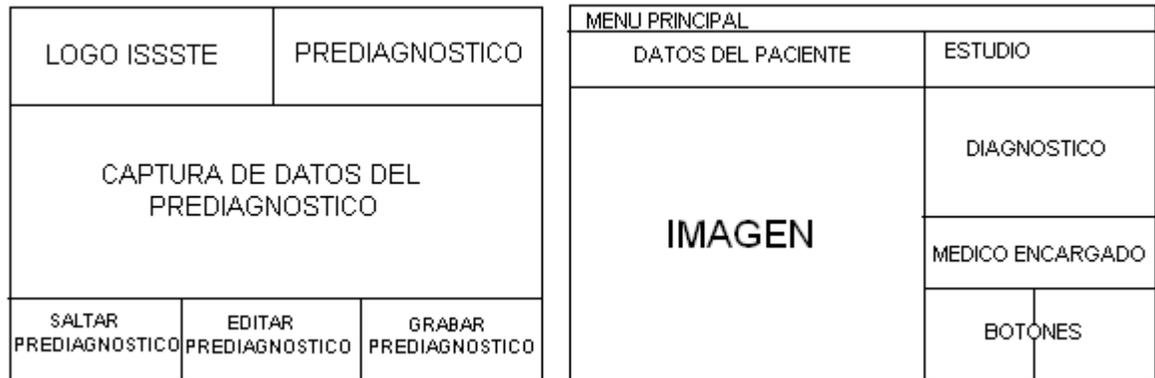


Fig. 5.10

En el capítulo siguiente definiremos el sistema operativo a utilizar, el manejador de bases de datos y el lenguaje de desarrollo que utilizaremos para la realización del sistema, ya que una de las condiciones principales del sistema es que corra en red, es decir bajo arquitectura **Cliente/Servidor**. Así como también debemos considerar el equipo de cómputo que se tiene dentro del ISSSTE.

VI. ANÁLISIS DE LA TECNOLOGÍA PARA LA IMPLEMENTACIÓN DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA

Para la elaboración de la presente tesis se abordan, en este capítulo, los temas sobre los cuales se pondrá en operación el sistema en cuestión, es decir, el equipo de cómputo y el software a utilizar.

6.1. SERVICIOS QUE OFRECE LA RED DEL ISSSTE

Entre los servicios brindados por la RED del ISSSTE destacan los que se mencionan a continuación:

➤ **Consulta tu Información Constitucional**

Es un servicio de consulta de información en línea, sobre todo lo relacionado con los seguros, servicios, créditos, y prestaciones que por ley le corresponden a cada trabajador.

➤ **Cita Médica Telefónica e Internet**

Todas las Clínicas de Medicina Familiar del ISSSTE en el Distrito Federal y en varios estados de la República Mexicana, donde se atiende al mayor número de derechohabientes, operan con el sistema de citas médicas telefónicas e Internet, lo que permite ahorrar en promedio 3.5 horas de espera a cada usuario. Se atienden vía telefónica a más de 25 mil usuarios y se otorgan cerca de 4 mil 100 citas al día.

➤ **SIPE-WEB**

El Sistema Integral de Prestaciones Económicas (SIPE), que permite actualizar en línea la vigencia de derechos de todos los cotizantes y de sus familiares, así como agregar información a cada uno de los registros de los trabajadores, cuyos datos pueden ser consultados a través de Internet por el público interesado. A este programa se suma una moderna página de Internet con información de lo más completo del sector público, porque incluye siete servicios en línea, los cuales de manera interactiva los derechohabientes y los ciudadanos en general pueden consultar y recibir respuesta de los servicios.

➤ **ISSSTE-TEL**

Este programa comenzó a operar en octubre de 1996 para cubrir la necesidad que el Instituto tenía de contar con un servicio telefónico para informar,

orientar y captar las inconformidades, sugerencias y felicitaciones sobre los servicios, seguros y prestaciones que otorga el ISSSTE a su derechohabiente.

Los estados que cuentan con estos servicios son: D.F., Baja California Norte, Campeche, Chihuahua, Coahuila, Colima, Durango, Hidalgo, Jalisco, Morelos, Oaxaca, Puebla, Querétaro, Tabasco, Quintana Roo, Tlaxcala, Zacatecas.

6.2. EL PROTOCOLO DE COMUNICACIÓN TCP/IP

Aunque poca gente sabe lo que es TCP/IP todos lo emplean indirectamente y lo confunden con un solo protocolo cuando en realidad son varios, de entre los cuales destaca y es el más importante el protocolo IP. Bajo este nombre (TCP/IP) se esconde uno de los protocolos más usados del mundo, si bien es cierto que existen diferentes tipos de protocolos, entre los cuales los de mayor uso en empresas pequeñas y medianas, así como en grandes corporativos son los siguientes:

- TCP/IP
- IPX/SPX
- NetBeui
- DLC

Sin lugar a dudas, debido al gran auge y crecimiento que ha tenido Internet en los últimos años, el protocolo que se perfila como estándar mundial es el TCP/IP, adoptándolo de igual forma dentro de la red del ISSSTE. Por tal motivo la aplicación que se desarrolló en la presente tesis, tomó como base el uso de este protocolo.

6.2.1. Historia del Protocolo de Comunicación TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol). En el año de 1973 la DARPA (**Defense Advanced Research Projects Agency**) inició un programa de investigación de tecnologías de comunicación entre redes de diferentes características. El proyecto se basaba en la transmisión de paquetes de información, y tenía por objetivo la interconexión de redes. De este proyecto surgieron dos redes: Una de investigación, ARPANET, y una de uso exclusivamente militar, MILNET. Para comunicar las redes, se desarrollaron varios protocolos. El protocolo de Internet y los protocolos de control de transmisión.

Posteriormente estos protocolos se englobaron en el conjunto de protocolos TCP/IP. En 1980, se incluyó en el UNIX 4.2 de BERKELEY, y fue el protocolo militar estándar en 1983. Con el nacimiento en 1983 de INTERNET, este protocolo se popularizó bastante, y su destino va unido al de Internet. ARPANET dejó de funcionar oficialmente en 1990.

El propósito de TCP/IP es proporcionar los protocolos necesarios de transmisión de información independientemente de la red o equipo utilizado, aplicándose en organizaciones que precisaban establecer conexiones entre distintos tipos de LAN. Dada la flexibilidad que caracteriza un protocolo de esa naturaleza, TCP/IP ha sido elevado casi de forma inmediata al dominio de la Internet, ya que el modelo básico en Internet es el modelo Cliente/Servidor. El Cliente es un programa que le solicita a otro que le preste un servicio. El Servidor es el programa que proporciona este servicio. La arquitectura de Internet está basada en capas. Esto hace más fácil implementar nuevos protocolos. El conjunto de protocolos TCP/IP, al estar integrado plenamente en Internet, también dispone de este tipo de arquitectura. El modelo de capas de TCP/IP es algo diferente al propuesto por ISO (**International Standard Organization**) para la interconexión de sistemas abiertos OSI (**Open System Interconnection**).

6.2.2. ¿Cómo trabaja el Protocolo TCP/IP?

El protocolo TCP/IP define una red de intercambio por paquetes, lo que significa que la información se transmite entre las computadoras en partes: una parte está conformada por los paquetes que contienen una “carga” de datos y, la otra, por la información de cabecera que identifica el envío, constituida por la dirección de destino y los códigos de corrección de errores. Si bien algunas órdenes requieren tan sólo un único paquete, las transmisiones de archivos de gran extensión se dividen en múltiples paquetes, debiendo incluir una secuencia de números indicativos del orden según el cual los paquetes se deben reensamblar en el sistema de destino.

Una de las características más importantes del protocolo TCP/IP es su esquema de direccionamiento de red, el cual proporciona una forma de direccionar diferentes redes, así como los nodos de las mismas. Este esquema es extensivo, dando cabida a millones de posibles direcciones en una escala global. De hecho, las direcciones Internet son registradas y asignadas por el NIC (**Network Information Center**), a fin de evitar conflictos aunque no es esencial registrar las direcciones, si tan sólo se trata de establecer una conexión entre redes en un ámbito reducido.

Como ya se mencionó en un principio, TCP/IP está compuesto por un conjunto de diferentes protocolos. Cada uno de ellos cae dentro de un esquema de estratos que se compara a menudo con el modelo OSI (**Open System Interconnection**). A continuación se describe cada uno de estos protocolos:

- **TCP (Transmission Control Protocol)**. El protocolo TCP proporciona un servicio de comunicación que forma un circuito, es decir, que el flujo de datos entre el origen y el destino parece que sea continuo. TCP proporciona un circuito virtual el cual es llamado una conexión. El protocolo TCP corresponde al estrato de Sesión y Transporte del modelo OSI. TCP recibe información de las aplicaciones que operan en niveles superiores en la pila de protocolos, y es responsable del empaquetado y la transmisión a través de la red. TCP pasa a IP los paquetes que crea.
- **IP (Internet Protocol)**. El protocolo IP corresponde al estrato Red en el modelo OSI. Este se encarga de crear los paquetes de información, y es el responsable de añadir las direcciones IP de origen y destino. Algo que IP no puede proporcionar es la garantía de que los paquetes llegaran a su destino, y en el orden adecuado. Por tal motivo, TCP añade información a cada uno de los paquetes que este módulo crea, en la que incluye datos de identificación y ubicación. Si IP pierde un paquete, TCP es el responsable de determinar cuál es el paquete perdido, y de volverlo a enviar.

6.2.3. Direcciones IP

La dirección **Internet Protocolo (IP)** de un nodo es una dirección lógica, que es independiente de la dirección física asignada a la tarjeta de red por el fabricante de la misma. La dirección IP es también independiente de la configuración de la red. La dirección IP tiene la misma forma, no importando el tipo de red que se usa. Este formato único es un valor numérico de 4 bytes (32 bits), que sirve para identificar, tanto a la red, como al nodo de la misma. Cada dirección IP debe ser única y consta de cuatro números decimales, separados por puntos. Como se menciono anteriormente, las direcciones IP no dependen de ningún tipo de red en concreto. Ello significa que los paquetes pueden atravesar diferentes tipos de redes. En cada uno de los tipos de red, el protocolo TCP/IP se encarga de asignar la dirección IP a un nodo físico, para poder realizar el tránsito. Los paquetes contienen las direcciones IP del remitente, lo cual permite que el destinatario pueda reclamar los datos, en caso de ser necesario. La dirección IP permite identificar tanto la red como el nodo que corresponde al remitente. Una

forma de definir direcciones IP consiste en unirse a la comunidad DARPA Internet, contactando con el NIC (**Network Information Center**) mencionado en párrafos anteriores. Si no se está interesado en registrar oficialmente una red, se pueden elegir números arbitrarios que se rijan por el esquema de direccionamiento IP (cuatro números decimales, separados por puntos). Sin embargo, es aconsejable que se utilice el esquema DARPA Internet, por si en el futuro se precisase entrar en contacto con algún punto externo. La dirección IP de 4 bytes está dividida en dos partes: una de ellas, sirve para identificar a la red, mientras que la otra, identifica a la computadora (nodo). De esta manera, la parte que corresponde a la red, debe ser siempre la misma para todos los nodos que la conforman, en tanto que la parte que identifica al nodo, ha de ser única y diferente para cada equipo conectado a dicha red.

Existen diversos esquemas para la asignación de estos números:

1. Esquema de direccionamiento Clase A.

El primer byte es la dirección de la red, y los tres últimos la del nodo. El rango para el primer byte es de 1 a 126, lo cual permite un total de 126 redes diferentes, con 16 millones de nodos cada una.

2. Esquema de direccionamiento Clase B.

Los dos primeros bytes identifican a la red, y los dos últimos, el nodo. El rango para el primer byte es de 128 a 191, quedando el segundo para identificaciones más finas. Esto permite un total de 16,000 redes con 65,000 nodos.

3. Esquema de direccionamiento Clase C.

Los tres primeros bytes identifican a la red, y el último es la dirección del nodo, lo cual permite un total de 2 millones de redes diferentes, con 254 nodos cada una.

Es posible dividir una red en varias subredes, para así poder hacer uso de múltiples tipos de medios, o bien para reducir la congestión, disminuyendo el número de puestos de trabajo de la red. Cuando existen subredes, las direcciones IP constan de una dirección de red, una dirección de subred y una dirección de nodo. La parte de dirección que corresponde al nodo en la dirección IP, se divide para incluir las direcciones de la subred y del nodo. Para las redes externas, la red total seguirá apareciendo como un único conjunto, con una única dirección de red.

6.2.4 Principales características del Protocolo TCP/IP

Entre las principales características que posee el protocolo TCP/I, se pueden mencionar las siguientes:

- Independencia de tecnologías de redes (Soporta múltiples tecnologías).
Ya que no está basado sobre el hardware de ningún vendedor en especial. Los protocolos TCP/IP definen la unidad de transmisión como datagrama y especifican cómo transmitir datagramas en una red.
- Interconexión universal (Puede funcionar en máquinas de cualquier tamaño).
Una red de TCP/IP permite la comunicación de cualquier pareja de computadoras que forman parte de la red. A cada computadora se le asigna una dirección, que es reconocida por todos los equipos integrantes de la red. Cada datagrama lleva la dirección de origen y destino. Las computadoras intermedias usan la dirección de destino para tomar decisiones de ruteo.
- Acuse de recibo entre computadoras que dialogan (Reconocimientos de extremo a extremo).
Los protocolos de TCP/IP proporcionan un acuse de recibo sólo entre las dos computadoras que dialogan y no entre máquinas sucesivas que forman parte de la trayectoria.

6.3. PLATAFORMAS DE HARDWARE

Como al principio del capítulo, en esta ocasión se analizará lo referente a las características del hardware, para poder llevar a cabo nuestro objetivo, por lo tanto, nos daremos a la tarea de explicar las particularidades de éste e iremos proponiendo el más viable tanto para el Servidor como para los equipos Clientes. En cuanto al equipo que fungirá como Servidor, se hace necesario exigir, las siguientes características que analizaremos a continuación.

Cabe mencionar que, dada la dimensión del sistema a generar, la plataforma a utilizar debe de ser robusta, confiable, eficiente, escalable y con un alto rendimiento, de tal forma que consienta un buen número de usuarios simultáneamente.

Así, se realizó una investigación comparativa de las tecnologías existentes en el mercado, con el fin de elegir la que mejor se apegue a las necesidades de nuestro sistema.

6.3.1. Procesadores CISC

Durante los años 60, 70 y 80's la evolución de los procesadores se caracterizó por un enriquecimiento del repertorio de instrucciones, tanto en cantidad como en modos de direccionamiento, se creía que la mejor forma de crear un procesador era incrementando su complejidad, como parte de un esfuerzo por emular fielmente las instrucciones proporcionadas por un lenguaje de alto nivel. A este tipo de procesadores se les llama CISC (**Complex Instruction Set Computer**), como por ejemplo los Intel x86, Motorola 68000 y Digital VAX. Un procesador CISC se caracteriza por tener un set de instrucciones amplio, una gran variedad de modos de direccionamiento y el uso extensivo de microcódigo.

En la teoría, un procesador CISC parece una buena opción ya que ejecuta directamente en **hardware** las instrucciones de **software**. Sin embargo, el análisis de cientos de programas en código objeto reveló los siguientes hechos.

- Los compiladores muy frecuentemente no hacen uso completo de las instrucciones proporcionadas por el procesador (alrededor del 80% de las operaciones se realizan con sólo un 20% de las instrucciones del repertorio).
- Un diseño complejo lleva a implementaciones ineficientes de las instrucciones.
- El uso de las instrucciones simples como **LOAD**, **STORE**, **ADD**, **SUB** y **BRANCH** son mucho más frecuentes que el de las instrucciones más complejas.

6.3.2. Procesadores RISC

Los procesadores RISC (**Reduced Instruction Set Computer**), como ejemplo la arquitectura SGI MIPS, Sun Sparc, Digital Alpha, PowerPC y HP PaRISC, se basan en simplificar al máximo el conjunto de instrucciones. En este tipo de arquitecturas, la complejidad queda a cargo del compilador. Los principales puntos de diseño de un procesador RISC son:

- Conjunto reducido de instrucciones. Sus instrucciones son sencillas, realizan únicamente las operaciones básicas. Idealmente, se busca que se ejecute una o más instrucciones por cada ciclo de reloj.
- No existen modos de direccionamiento. Tienen arquitectura (**LOAD/STORE**). Esto quiere decir que los accesos a la Memoria Principal (MP) son para extraer instrucciones, datos y para almacenar éstos últimos. Con esto se simplifica el diseño del procesador.
- Existe un mayor espacio para incorporar el chip de memoria.
- Tienen un número grande de registros de CPU.
- Menor consumo de potencia.
- Tiempos más cortos de diseño.

La “era RISC” ha alcanzado a todos los fabricantes de semiconductores: AMD, Intel, MIPS, Motorola, etc., y todos ellos son productos usados por fabricantes de computadoras y estaciones de trabajo: Apple, DEC, HP, IBM, SUN, etc. y sus correspondientes clónicos.

El tiempo de diseño de estos productos se reduce sensiblemente, lo que disminuye su costo final, además, son globalmente más eficaces, de menores dimensiones y más bajo consumo, ofreciendo siempre claras ventajas técnicas frente a los más avanzados CISC.

6.3.3. Multiproceso

La industria informática, ha tenido siempre un objetivo primordial, repetido a lo largo de toda su cadena (fabricantes de semiconductores, fabricantes de sistemas y usuarios): **La búsqueda de la velocidad**. Para alcanzar este objetivo se han invertido grandes cantidades de recursos, hasta alcanzar los límites físicos del silicio.

Hoy es posible construir sistemas, que aún teniendo procesadores más “lentos” que otros, ofrezcan un rendimiento significativamente superior. Son los sistemas multiprocesador, que como su nombre lo indica, incorporan varios procesadores para llevar a cabo las mismas funciones.

Ahora, y en gran medida gracias a los procesadores de arquitectura RISC, el soporte multiprocesador es una solución integrada y fácilmente disponible en estaciones de trabajo de sobremesa, que resuelve, a través de hardware VLSI (Very large Scale Integration), los complejos problemas de compartición de recursos (memoria compartida) de aquellas primeras máquinas.

Evidentemente, estas mejoras en el hardware, para ser funcionales, requieren importantes desarrollos en el software, y de hecho, muchos Sistemas Operativos admiten extensiones multiproceso (SCO, Solaris, System V, Windows NT, Unix, Linux, etc.), que proporcionan paralelismo “en bruto” (asignando múltiples tareas a múltiples procesadores) a nivel del Sistema Operativo.

El multiproceso no es algo difícil de entender: más procesadores significan más potencia computacional. Un conjunto de tareas puede ser completado más rápidamente si hay varias unidades de proceso ejecutándolas en paralelo; adicionalmente, las plataformas multiproceso son fácilmente escalables.

Ahora bien, en el caso de nuestra aplicación, estamos conscientes de que las aplicaciones de Base de Datos son típicamente intensas en manejo de Entradas/Salidas, por lo tanto, las arquitecturas de hardware que permiten la utilización de múltiples procesadores comúnmente otorgan un rendimiento superior a aquellas que no lo poseen.

6.3.4. Multiprocesamiento Simétrico-Asimétrico

Estas dos arquitecturas tienen diferencias entre sí, la más marcada es el rendimiento que manifiesta cada una de ellas. El multiprocesador simétrico, trata a todos los procesadores como iguales. Cualquier procesador puede hacer el trabajo de cualquier otro, y las aplicaciones se dividen en subprocesos que pueden ejecutarse de manera concurrente en cualquier procesador disponible, mejora el rendimiento de la aplicación misma y también el rendimiento total del sistema. Pero lo más importante es que los sistemas simétricos requieren aplicaciones que puedan aprovechar el paralelismo de múltiples subprocesos. Entre los Sistemas Operativos compatibles con el multiprocesamiento simétrico se encuentran UNIX, OS/2 y Windows NT y entre las aplicaciones, Microsoft BackOffice y también los administradores de base de datos SQL Server, Oracle, Sybase e Informix.

Por su parte en el multiprocesamiento asimétrico, hay un procesador (maestro) en el cual se ejecuta el Sistema Operativo, los demás procesadores ejecutan el resto de las tareas. La ventaja de éste es que al aumentar más procesadores se tiene que hacer un cambio mínimo y fácil para el manejo de éstos y en general se eliminan muchos problemas de integridad de datos. La gran desventaja es que al haber sólo una copia del Sistema Operativo en un solo procesador (maestro) cuando este procesador falla todo el sistema falla porque todos los recursos que son manejados por el Sistema Operativo no pueden ser accedidos.

Dado que la plataforma a utilizar debe de ser confiable, robusta y con un alto rendimiento, para el caso del **Sistema de Teleimagenología**, se recomienda seleccionar la arquitectura de multiprocesamiento simétrico ya que mejora el rendimiento de la aplicación así como el rendimiento total del sistema.

Hasta aquí se han analizado los requerimientos del Servidor sobre el cual se levantará el sistema, por tanto, se sugiere que el Servidor tenga las siguientes características:

- Que el equipo se pueda escalar, tanto en memoria, capacidad en disco duro, así como en el número de procesadores, a medida que crezcan las necesidades del sistema.
- Que acepte multiprocesamiento simétrico.
- Que optimice los procesos de entrada y salida, ya que accedaremos a una base de datos.

De esta manera, se tiene que el equipo para el **Servidor** que se usará para desarrollar el **Sistema de Teleimagenología**, es el siguiente:

- HP Netserver E800 es un servidor con:
 - Doble procesador Pentium III a 800 MHz
 - Dos discos duros de 9.1 GB cada uno
 - 384 MB SDRAM, expandible hasta 2GB
 - High Speed CD-ROM drive
 - Tarjeta de Red Fast Ethernet 10/100 integrada

Se eligió este equipo pues a parte de cumplir con los requerimientos técnicos y expectativas que se necesitan para poder llevar a cabo el desarrollo de nuestra aplicación, es el más accesible, ya que se tiene fácil acceso a él pues contamos con el apoyo del laboratorio de UNICA (Unidad de Servicios de Cómputo Académico) de Fundación UNAM en la Facultad de Ingeniería, para el desarrollo del presente proyecto.

Actualmente en el mercado existen equipos más potentes, sin embargo, como requerimiento mínimo para el equipo del **Servidor** recomendamos al HP Netserver E800.

6.3.5. El equipo ideal (Cliente)

La tecnología en general, y especialmente la tecnología informática y de las telecomunicaciones, avanza sin cesar; los equipos compatibles PC, cada vez son más rápidos, tienen más capacidad de almacenamiento y de proceso de la información, pueden realizar más trabajos, nos proporcionan el acceso a grandes cantidades de información y sus precios son accesibles.

Actualmente en nuestro país, la mayoría de los centros educativos, culturales, hospitales, empresas, etc., tanto privados como públicos, están dotados de equipos compatibles PC, y la mayor parte de los programas didácticos existentes están diseñados para estas máquinas. Por lo antes escrito, consideramos que para el propósito del presente trabajo, se pueden aprovechar los equipos PC.

Una vez propuesto el tipo de equipo para poder ejecutar la aplicación en cuestión, queda por determinar las características específicas con las que deberán contar los equipos Cliente, como son: La velocidad con la que procesa la información, cantidad de memoria, capacidad de almacenamiento, el rendimiento gráfico y comunicación con otros equipos.

6.3.5.1. Características deseables en los equipos Cliente

Los equipos informáticos actuales son modulares. Están constituidos por múltiples componentes fabricados por distintas empresas pero que funcionan perfectamente adaptados entre sí con la ayuda de las funcionalidades que proporcionan los programas del BIOS (**Basic Input/Output System**) y del Sistema Operativo que se instale (Windows, MS/DOS, Linux, etc.). Esta modularidad facilita en muchos casos su futura ampliación o escalabilidad, permitiendo adaptar los equipos a los requerimientos de los nuevos programas mediante la ampliación de su procesador, disco duro y memoria o la conexión de un nuevo periférico. A continuación se analizarán las principales características de estos equipos en cuanto al rendimiento que conviene exigir para obtener el máximo aprovechamiento del sistema a realizar.

➤ La Velocidad:

La velocidad real de trabajo de los equipos está determinada por las características de todos los elementos que configuran el sistema informático, pero depende especialmente de cuatro factores:

- a) El tipo de microprocesador (cerebro del equipo), que es capaz de realizar unas determinadas operaciones básicas y dispone de unos registros y buses para conducir la información. En estos momentos los microprocesadores más utilizados son los Pentium III, con registros internos de 64 bits, que comercializa la empresa Intel y los K7 de la empresa AMD de similares características. Aunque existen ya los Pentium IV, los cuales pueden procesar a una mayor velocidad.

Conviene además que el microprocesador esté insertado en un zócalo o socket tipo ZIF (**Zero Insertion Force**), que facilitará su sustitución si llega el momento de actualizarlo.

- b) La velocidad del reloj interno del microprocesador (se mide en megahercios, MHz), que determina las operaciones que el microprocesador es capaz de realizar cada segundo. Un microprocesador a 900 MHz es capaz de realizar novecientos millones de operaciones cada segundo.
- c) La arquitectura de la tarjeta principal que constituye la Unida Central del equipo y la velocidad de los buses a través de los cuales circula la información.
- d) La memoria caché, más rápida que la memoria RAM (**Random-Access Memory**), almacena los datos que utiliza el microprocesador con más frecuencia.

➤ **La Memoria y Proceso de la Información**

La capacidad de proceso de la información de un equipo depende especialmente del microprocesador y de la memoria RAM que tenga el equipo. En la memoria RAM es donde se almacenan los programas que se están ejecutando. Un equipo con poca memoria RAM solo podrá ejecutar programas pequeños. Con la llegada del Sistema Operativo Windows y de los entornos multimedia los programas cada vez son más grandes y necesitan más memoria, de manera que los equipos Cliente no deberían tener menos de 64 MB de memoria RAM.

Esta memoria debe estar acorde con las características de los buses de la tarjeta principal, y además conviene que sea tipo SDRAM (Synchronous Dynamic Random Access Memory) que van en conectores DIMM (**Dual In-line Memory Module**) de 168 contactos que es más rápida que las anteriores memorias EDO (**Extended Data Out**) con conectores SIMM (**Single In-line Memory Module**) de 72 contactos o SRAM (**Static Random Access Memory**) con conectores SIMM de 30 contactos.

➤ **La Capacidad de Almacenamiento**

La capacidad de almacenamiento de los equipos viene dada por las características de las unidades de disco que se instalen. El equipo debe disponer al menos de las siguientes unidades:

- 1) Unidad lectora de disquetes convencionales de 3 ½ pulgadas con 1'44 MB de capacidad máxima (HD, alta densidad), que puede leer también disquetes DD (**Double Density** de 720 K).
- 2) Disco Duro de gran capacidad; por ejemplo un disco duro de 20 GB (1 Gigabyte = 1000 Megabytes = 1 000 000 Kbytes). De tecnología IDE (**Integrated Drive Electronics**) que si además son Ultra-DMA (**Direct Memory Access**) aumentan muy considerablemente su velocidad de trabajo pudiendo llegar a 66M/seg. También hay discos duros SCSI (**Small Computer System Interface**) que aún son más rápidos (hasta 80 M/seg), pero su precio es mayor.
- 3) Lector de CD-ROM (Compact Disk – Read Only Memory) o lector de DVD (**Digital Versatile Disc**). Hasta hace muy poco todos los equipos multimedia incluían un lector de CD-ROM, que permite leer discos ópticos de 650 y 700 MB, reproducir discos compactos de audio y visualizar fotografías digitales. Una unidad con una velocidad x48 proporciona una transferencia de unos 7.2 M/seg. No obstante hoy en día ya puede ser una mejor opción adquirir un DVD. Los lectores DVD (**Digital Video Disk**) además de leer los discos convencionales de CD-ROM y audio, pueden leer los nuevos discos DVD de 17 G. Un DVD con velocidad x10 proporciona una transferencia de 15 M/seg.

➤ **Rendimiento Gráfico**

El rendimiento gráfico de los equipos depende de dos factores:

- a) La tarjeta de video, que controla el monitor, será tipo AGP (**Accelerated Graphics Port**). La tarjeta incluirá 16 MB o 32 MB de memoria de video, que asegurará la presentación de imágenes, video y gráficos con una resolución aceptable.
- b) El monitor, ha de ser capaz de responder a los requerimientos de la tarjeta de video. Puede ser una pantalla de baja radiación de 15

pulgadas de diámetro y el tamaño de sus puntos será de 0.28 mm. o menor (cuanto menor sea este tamaño mayor será el número de puntos y por lo tanto también será mejor la definición de las imágenes que presente).

➤ **Comunicación**

Para que nuestro equipo pueda conectarse a una red, necesitamos de un dispositivo electrónico llamado Tarjeta de Red o bien de un Módem a continuación se explica cada uno de éstos:

- La Tarjeta de Red, es lo que se necesita para conseguir una conexión entre más de un equipo para conformar una red. Normalmente las tarjetas de red modernas para equipos PC, suelen ir conectadas a un **slot** (ranura de expansión) **PCI** (Peripheral Component Interconnect), y no necesitan de ninguna configuración previa, éstas se conectan a una velocidad de 10/100 Mbps; de igual forma las hay antiguas que van conectadas a un **slot ISA** (Industry Standard Architecture), una desventaja es que a éste tipo de tarjetas se le tiene que configurar las interrupciones y las direcciones de Entrada/Salida, pero por otro lado son más baratas que las PCI, éstas se conectan a una a velocidad de 10 Mbps.
- El Módem es un dispositivo de Entrada/Salida por medio del cual la computadora puede establecer contacto con otras computadoras a través de la línea telefónica. Para realizar el proceso de comunicación, es necesario que cada computadora tenga su módem el cual debe estar programado con la misma velocidad de transmisión. El término Módem procede de Modulador/Demodulador que resume la función del módem, es decir, los datos que una computadora debe enviar están formados por bits, estos bits se transmiten de uno en uno por el puerto serie al módem, éste convierte estos datos digitales en señales analógicas de modo que puedan circular por la línea telefónica, modula los datos. El módem que se encuentra en el otro extremo de la línea telefónica y recibe estas señales de frecuencia las convierte en señales digitales, bits, decimos que demodula los datos, y los transmite por el puerto serie de uno en uno al PC. La mayoría de los módems utilizan un grupo de órdenes o comandos denominados comandos Hayes o comandos AT, debido a que todos ellos empiezan con las letras AT (por ejemplo ATDT significa realizar la marcación por tonos o ATDP por pulsos).

En resumen, las características deseables para los equipos Cliente son las siguientes:

- Que el equipo se pueda escalar, tanto en memoria, capacidad en disco duro, así como en el procesador, a medida que crezcan las necesidades del sistema y que éste tenga capacidad de comunicación con otros equipos.

De acuerdo a estas características, el sistema que se creará puede ser instalado en un equipo de cómputo con un procesador que contenga los requerimientos **mínimos** que a continuación se mencionan:

- Procesador Pentium II o superior
- Memoria RAM 64 MB en adelante
- Un espacio libre en disco duro de 30 MB

Lógicamente mientras el equipo Cliente sea más potente, obtendremos un mejor aprovechamiento del sistema a realizar, siendo así que nosotros proponemos el siguiente equipo para los **Clientes**, pues en éste se desarrollará y probará el **Sistema de Teleimagenología Clínica**.

HP VECTRA

- Procesador Pentium III a 750 MHz
Memoria RAM de 64 MB. Caché 512 KB
Disco duro de 10 GB

6.4. PLATAFORMAS DE SOFTWARE

Una vez descrito y seleccionado el hardware, pasamos a seleccionar el Sistema Operativo de Red y el Sistema Administrador de Base de Datos Relacional (RDBMS) para el Servidor sobre el que se ejecutará el Sistema de Teleimagenología; para ello se tienen dos alternativas en lo que se refiere a

Sistemas Operativos de Red, que “a priori” parecen válidas para Servidores PC, éstas son Linux y Windows NT, para el caso de RDBMS (**Relational Database Management System**), se verán Sybase y SQL Server. A continuación se estudiarán estas herramientas teniendo en cuenta que ambas pueden aprovechar al máximo las tecnologías que se han estudiado a lo largo de este capítulo; por consiguiente describiremos las capacidades de éstas.

6.4.1. Sistemas Operativos de Red

El sistema operativo, es el instrumento indispensable para hacer de la computadora un objeto útil. Un sistema Operativo es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al procesador central para que este realice las tareas necesarias y específicas para completar una orden.

Un Sistema Operativo (SO) es en sí mismo un programa de computadora. Sin embargo, es un programa muy especial, quizá el más complejo e importante en una computadora. El SO despierta a la computadora y hace que reconozca a la CPU (**Central Processing Unit**), la memoria, el teclado, el sistema de video, las unidades de disco, los puertos y conectores (serie y paralelo) de entrada y salida, los cuales son vías por medio de las cuales la computadora se comunica con los dispositivos externos antes mencionados. Además, proporciona la facilidad para que los usuarios se comuniquen con la computadora y sirve de plataforma a partir de la cual se corran programas de aplicación.

Los sistemas Operativos de Red empleados normalmente son UNIX, Linux, Macintosh OS, OS/2, Windows 2000 y Windows NT. El UNIX y sus clones permiten múltiples tareas y múltiples usuarios. Otros Sistemas Operativos multiusuario y multitarea son OS/2, desarrollado inicialmente por Microsoft e IBM, Windows NT y Windows 2000 desarrollado por Microsoft. El sistema Operativo multitarea de Apple se denomina Macintosh OS.

A continuación se analizarán los Sistemas Operativos de Red Linux y Windows NT.

6.4.1.1. Linux

Linux es una versión de UNIX libremente distribuible e independiente, para plataformas con procesadores X86, Motorola 68k, Digital Alpha, Sparc, Mips y Motorola Power PC.

En la actualidad, este Sistema Operativo es utilizado por miles de usuarios para desarrollo de software, redes y para plataformas de usuarios finales. Linux, entre los miles de Sistemas Operativos alternos que existen, se ha convertido en una opción interesante, independientemente de que estas vengan de UNIX o de las más conocidas donde se encuentra Windows 2000 y NT.

Linux está disponible en Internet en cientos de Servidores FTP, pues el programa es libre. Algunas distribuciones son: Caldera, Debian, Slackware, Red Hat, etc.

Linux, maneja los archivos de forma jerárquica, de la misma forma que DOS (**Disk Operating System**), con la diferencia que DOS está diseñado para procesadores X86 que no soportan verdaderas capacidades de múltiples tareas.

➤ **Ventajas**

1. Precio.
2. Estabilidad, no se traba a cada rato.
3. Seguridad, es mucho más seguro que otros Servidores.
4. Compatibilidad, reconoce la mayoría de los otros Sistemas Operativos en una red.
5. Velocidad, es mucho más veloz para realizar las tareas.
6. Pose el apoyo de miles de programadores a nivel mundial.
7. El paquete incluye el código fuente, lo que permite modificarlo de acuerdo a las necesidades del usuario.
8. Ideal para la programación, ya que se puede programar en Linux para distintas plataformas.
9. Un sistema de crecimiento rápido.
10. Se puede usar en casi cualquier computadora, desde una 386.
11. Multitareas REAL.
12. Puede manejar múltiples procesadores. Incluso hasta 16 procesadores.
13. Casi libre de virus (intentos de virus), se conoce un virus para Linux.
14. Maneja discos duros de hasta 16 TeraBytes.

15. Se consiguen parches con facilidad, además de ser gratuitos.
16. Se posee el apoyo de millones de usuarios a nivel mundial.
17. Los fabricantes de Hardware le están dando su apoyo, como IBM y Compaq.
18. Vendedores y desarrolladores implementan un sistema de certificación para Linux.
19. La corporación DATA Internacional predice que el crecimiento de este programa será del orden de un 25 por ciento anual en el nuevo milenio.

➤ **Desventajas**

1. Linux no cuenta con una empresa que lo respalde, por lo que no existe un verdadero soporte como el de otros Sistemas Operativos.
2. Linux corre el riesgo de llegar a fragmentarse como fue el caso de UNIX.
3. Algunas empresas pueden llegar a ayudar a Linux con la intención de mejorar sus relaciones públicas, aunque en el fondo no tengan ninguna intención de utilizarlo fielmente.

6.4.1.2. Windows NT

Con Windows NT, Microsoft ha expresado su dedicación a desarrollar software no sólo para PC de escritorio sino también para poderosas estaciones de trabajo y Servidores de Red y Bases de Datos. Microsoft Windows NT no es necesariamente un sustituto de DOS ni una nueva versión de éste; es, en conjunto, un nuevo Sistema Operativo diseñado desde sus bases para las máquinas más modernas y capaces disponibles.

Windows NT de Microsoft ofrece características propias que ningún otro Sistema Operativo para PC ofrece, con excepción de Unix. Además de las características tradicionales de estricta seguridad de sistema, red interconstruida, servicios de comunicación y correo electrónico interconstruidos, herramientas de administración y desarrollo de sistema y una GUI (Graphical User Interface),

Windows NT puede correr directamente aplicaciones de Windows de Microsoft y de Unix.

Windows NT, es un Sistema Operativo de 32 bits, que puede hacer completo uso de los procesadores de estas características. Esto le permite sacar ventaja de los microprocesadores más avanzados, como los Intel 80486 y posteriores, así como de las computadoras con tecnología RISC. Además de ser multitarea, está diseñado para tomar ventaja del multiproceso simétrico.

El rendimiento general, es decir, la combinación del rendimiento del microprocesador, transferencia de datos y acceso a memoria es, en definitiva, lo que este Sistema Operativo ha venido a mejorar sensiblemente.

La multitarea significa que el Sistema Operativo puede realizar varias tareas al mismo tiempo. En Windows NT existe la posibilidad de concurrencia de diversas tareas, de manera que si, por ejemplo, una de ellas queda parcialmente parada como consecuencia de un lento acceso al disco, el microprocesador puede concentrar su atención en otras tareas. Básicamente, lo que se consigue es que no se malgasten ciclos de procesamiento. El beneficio de esto es que los usuarios pueden realizar otras tareas mientras tiene lugar un acceso al disco en segundo plano, o bien, se prepara un trabajo de impresión.

El multiprocesamiento simétrico es una característica exclusiva de Windows NT, que permite sacar todo el partido de los procesadores múltiples. Windows NT es el primer Sistema Operativo que está preparado para realizar un uso eficaz de este tipo de tecnología. Los Sistemas Operativos anteriores se limitaban a asignar tareas específicas a cada microprocesador individual, como por ejemplo, para la Entrada/Salida en una red local.

El multiprocesamiento asimétrico, en donde un microprocesador se dedica exclusivamente a una tarea específica, da lugar a que éste permanezca inactivo en cuanto finaliza su tarea. En el multiprocesamiento simétrico, el Sistema Operativo puede asignar diferentes tareas a un mismo microprocesador, de esta manera, si uno de ellos termina su trabajo antes que otro, el Sistema Operativo podrá ocuparlo en otra actividad, por esto, ofrece un rendimiento superior.

Las características de trabajo en red local con las que cuenta Windows NT permiten compartir archivos de un sistema con cualquier usuario en un entorno de red, así como conectarse a directorios compartidos con otros sistemas.

El Sistema Operativo de Red Windows NT presenta una arquitectura del tipo Cliente/Servidor, ésta arquitectura se analizará ampliamente en el próximo

capítulo, ya que es un requisito indispensable que tendrá que cumplir el sistema a desarrollar en el presente trabajo.

En cuanto a la seguridad en Windows NT, se tienen dos cosas básicamente. El control total en el acceso al sistema y a los archivos o subdirectorios que hay en el sistema.

En el primer punto, el control sobre el acceso al sistema se refiere al manejo de user names y passwords para poder acceder al Sistema Operativo, de esta manera se mantienen a los usuarios sin autorización fuera del sistema. El siguiente nivel de seguridad en cuanto a este punto se refiere, son los privilegios que tiene un usuario, todos los usuarios o grupos de usuarios sobre los directorios y archivos del sistema, por ejemplo, el acceso a los archivos del sistema de NT está estrictamente limitado al Administrador del Sistema, mientras que las aplicaciones comunes como lo son hojas de cálculo o procesadores de palabras pueden ser accedidos por todos los usuarios.

Windows NT cuenta con un extenso sistema de control de seguridad para el acceso a archivos. El propósito de la seguridad en Windows NT es brindarle el acceso sólo a aquellos usuarios que están autorizados, controlar el acceso concurrente a archivos, a los directorios y a los recursos del sistema. La seguridad en los sistemas Windows NT debe ser configurada por el Administrador del Sistema siendo necesario para todos los sistemas un Administrador (incluyendo los sistemas monousuarios). El Administrador establece los nombres de usuario, crea grupos de usuarios, asigna los usuarios a los grupos, controla los passwords, permite los niveles de acceso a las funcionalidades del sistema; en pocas palabras el Administrador controla todos los puntos de acceso al sistema.

El Administrador puede controlar el acceso específico a ciertas funciones del sistema, especialmente aquellas que afectan el funcionamiento del mismo, este sistema de control es llamado la política de derechos del usuario. De esta manera el Administrador a través de esta política puede controlar las labores que efectúa un usuario tanto local como remotamente.

La integridad del sistema en NT: Entendamos por integridad del sistema a la habilidad del mismo de permanecer activo cuando una de sus aplicaciones falla. Windows NT está diseñado para prevenir la caída catastrófica del sistema en caso de que algunas de sus aplicaciones fallen y para esto establece los siguientes cuatro mecanismos de protección de memoria:

- 1) Espacio de direcciones separado: Cada proceso maneja sus propias direcciones virtuales y el sistema prohíbe el acceso a espacios de memoria de otros procesos.
- 2) Modos de Kernel y usuarios separados: Todas las aplicaciones corren en modo de usuario, por lo tanto, está prohibido el acceso o modificación del código o datos del sistema que residan en el kernel.
- 3) Banderas de páginas: Cada página de la memoria virtual tiene una bandera la cual determina cómo puede ser accesada en modo usuario y en modo kernel.
- 4) Seguridad de los Objetos: El manejador virtual de la memoria crea un tipo especial de objeto llamado objeto-sección el cual funciona como una ventana hacia la memoria virtual, por lo tanto, cada vez que un proceso accesa un objeto-sección, el sistema determina si el proceso tiene los permisos de lectura y/o escritura sobre éste. Dentro de la integridad del sistema Windows NT establece políticas y procedimientos de protección, el acceso a recursos de esta manera protege a los procesos de caer en estados muertos cuando compiten por recursos.

Windows NT Server, cuenta con el Microsoft Internet Information Server que es un Servidor Web que permite publicar información en una Intranet de la organización o en Internet. El IIS (**Internet Information Server**) transmite la información mediante el Protocolo de Transferencia de Hipertexto (http). Internet Information Server puede configurarse también para proporcionar servicios de Protocolo de Transferencia de Archivos (FTP) y Gopher. El servicio FTP permite que los usuarios transfieran archivos a y desde su sitio Web El servicio Gopher utiliza un protocolo controlado por menús para encontrar documentos. El protocolo Gopher ha quedado en buena parte reemplazado por el protocolo http (**hypertext Transfer Protocol**). El propósito de Windows NT es el de ser un Sistema Operativo diseñado para correr en distintas plataformas soportando los siguientes procesadores:

- La familia Intel X86
- 680x0 de motorola
- El MIPS 400
- El ALFA de Dec.
- El HP-PA de Hewlett Packard
- Los SPARC RISC processors de Sun Microsystems
- El RS/6000 de IBM
- Unas futuras versiones del Powerpc (Apple, IBM y Motorola)

La independencia de plataformas está basada en el concepto de desarrollar un kernel específico para cada uno de los distintos procesadores que sirva de interfaz entre el hardware específico y las llamadas al sistema de NT.

Desde hace unos años atrás esta existiendo un debate continuo en el campo de los Servidores: Linux o NT. Ambas plataformas, Linux y NT son totalmente opuestas en todos los niveles: Visualmente son diferentes, corren de manera diferente, soportan diferente hardware, y tienen costo diferente.

La más obvia diferencia entre NT y Linux es que la interfaz de NT con su Servidor Web IIS intenta traer la interfaz de Windows, más familiar con la mayoría de usuarios, al entorno de Servidores. Idealmente un Webmaster podría mantener un sistema NT usando su ratón y haciendo click. Lo que se puede ver sin discusión es su facilidad de manejo. Sí, es un Sistema Operativo muy sencillo de usar, quien tenga bases en la administración de una red de computadoras lo puede manejar perfectamente sin problemas, y todo gracias a su entorno gráfico que con el mouse maneja casi todo, lanzar programas, minimizar, maximizar y cerrarlos con solo apretar con el mouse en los respectivos íconos que se encuentran sobre la ventana. Prueba de ello, es su uso masivo, alrededor del 90% a nivel mundial.

Ahora bien, dada la magnitud del sistema a desarrollar, se requiere de un Sistema Operativo robusto; capaz de soportar a un gran número de usuarios en forma simultanea; que maneje adecuadamente la memoria y el multiprocesamiento simétrico, y, sobre todo, lo más importante, que aproveche al máximo el equipo Cliente (con la interfaz gráfica Windows 95, 98, 2000 y NT). Es por esto, que nosotros proponemos que el Sistema Operativo de Red Windows NT Server sea el elegido para poder desarrollar bajo esta plataforma el **Sistema de Teleimagenología Clínica**.

Podemos decir que; en cuanto a costo definitivamente Linux es la mejor opción, pero por otro lado, debemos estar concientes de que Linux no cuenta con un verdadero soporte como es el caso de Windows NT, así también, sabemos que NT es el primer Sistema Operativo que está preparado para realizar un uso eficaz del multiprocesamiento simétrico, por lo tanto, NT permite el aprovechamiento al máximo de los procesadores múltiples; es por todo esto que insistimos en seleccionar a Windows NT como el Sistema Operativo para poder llevar a cabo nuestro proyecto.

Una vez identificados y analizados los posibles Sistemas Operativos a utilizar, se requiere hacer también un análisis de los Sistemas Administradores de Bases de Datos Relacionales (RDBMS), que se acoplan a los Sistemas

Operativos antes mencionados; estos RDBMS's deben de ser robustos y tienen que manejar altos niveles de seguridad, mecanismos de respaldo, entre otras características. De tal manera que a continuación, se analizan los RDBMS's, basados en los Sistemas Operativos antes descritos.

6.4.2. Sistema Administrador de Base de Datos Relacional (RDBMS)

A continuación se dan las características principales de un RDBMS:

- 1) Representación de los datos por medio de tablas relacionadas.
- 2) Utiliza lenguaje de cuarta generación.
- 3) Flexibilidad: La modificación de los datos y los cambios a la estructura de la base de datos resulta muy sencilla.

Además el RDBMS es el responsable de:

- 1) Mantener las relaciones entre la información y la base de datos.
- 2) Asegurarse de que la información es almacenada correctamente, es decir, que las reglas que definen las relaciones entre los datos no sean violadas.
- 3) Recuperar toda la información en un punto conocido en caso de que el sistema falle. Por lo tanto, se procederá a realizar un análisis de los RDBMS siguientes:

6.4.2.1. Sybase

Sybase es una base de datos relacional diseñada para aplicaciones críticas y en línea, que requieren un alto rendimiento escalable, se basa en el modelo relacional. El lenguaje de consultas básicas es SQL. Sybase emplea una arquitectura **multithreaded** (multitareas) en la que la base de datos, más que el Sistema Operativo, es la que programa y controla las tareas para obtener una máxima eficiencia. Múltiples sentencias SQL pueden aumentarse con la programación de constructores, tales como lógica condicional, llamadas a procedimientos y variables locales, estos pueden combinarse en un objeto de base de datos llamado un procedimiento de almacenamiento. Los procedimientos pueden regresar hileras de datos y mensajes de error, además de regresar valores en variables de programación en el programa de aplicación. El Servidor Sybase también soporta disparadores como objetos independientes, estos tienen las capacidades de los procedimientos con tres extensiones importantes: 1) Ellos no

pueden ejecutarse directamente, sólo responden al cumplimiento de una condición. 2) Un disparador puede restaurar o modificar los resultados de una transacción del usuario. 3) El disparador puede ver los cambios hechos a los datos.

Los procedimientos de almacenamiento (**stores procedures**) y los disparadores (**triggers**) convierten a Sybase en un Servidor programable.

Las rutinas de mantenimiento de la base de datos son realizadas mientras que el sistema está totalmente operando. Sybase también se protege contra fallas del sistema con su propio mecanismo de tolerancia a fallas implantadas en el software. La interoperabilidad de Sybase permite a cualquier aplicación Cliente trabajar cooperativamente con cualquier fuente de datos a través de diferentes plataformas de hardware, red local y Sistema Operativo. El API (**Application Program interface**) del **Open Client** de Sybase es un método bastante simple y flexible para acceder una gran variedad de fuentes de datos. **El Open Server** de Sybase permite a cualquier fuente de datos responder a las peticiones de los clientes y comportarse como si fuera un Servidor inteligente. Los procedimientos remotos de Sybase RPC's (Remote Procedur Calls) son un mecanismo de mensajes para la comunicación entre cualquier Cliente y el Servidor.

6.4.2.2. MS SQL Server

El MS SQL Server es un sistema Administrador de Base de Datos Relacional inteligente, basado en la arquitectura Cliente/Servidor (como se mencionó ésta arquitectura se analizará a fondo en el próximo capítulo). Entre las ventajas que tiene MS SQL Server están su perfecta integración con el Sistema Operativo Windows NT Server. Este Servidor de Base de Datos es un programa que almacena datos estructurados en forma de tablas relacionales, escucha un puerto TCP/IP a través del cual acepta conexiones de Clientes autenticados, admite comandos en lenguaje SQL y devuelve al Cliente a través de la red los datos resultantes del procesamiento de los comandos SQL, es decir, SQL Server puede recibir conexiones (Clientes que desean manipular datos) desde cualquier equipo conectado a Internet.

La seguridad y garantía del procesamiento de transacciones en SQL Server lo hacen ideal para las más exigentes aplicaciones de misión crítica. Las capacidades avanzadas de SQL Server tales como: Trigger, Procedimientos

Almacenados, Reglas, Defaults y su tecnología de llamado a procedimiento remoto (RPC), hacen de SQL Server la herramienta ideal para el desarrollo de aplicaciones dorsales (**back-ends**) en estaciones de trabajo, minicomputadoras y mainframes. Su arquitectura diseñada para el alto rendimiento, multitarea, asegura un rendimiento consistente y estable que se mantiene aún cuando el número de usuarios aumenta. Los puentes SQL Server (**gateways**), también proveen la tecnología clave para la integración de microcomputadoras con minicomputadoras y mainframes de larga escala. Una de las grandes ventajas de SQL Server para Windows NT es que al igual que la mayoría de los sistemas desarrollados por la compañía Microsoft, se ejecutan sobre ambientes gráficos que ayudan tanto al Cliente para el desarrollo de aplicaciones, como al Administrador para tener mejor controlado el sistema.

Características principales de SQL Server:

- Administración remota de los recursos del sistema.
- Arquitectura de 32 bits.
- Capacidad para generación automática de copias de seguridad.
- Ejecución de procedimientos almacenados en forma remota.
- Un robusto soporte de tolerancia a fallas
- El reforzamiento centralizado de la integridad de los datos minimiza el mantenimiento, programación y reduce los costos, asegurando que los datos son correctos.
- Interoperabilidad dinámica con la red local.
- Monitoreo gráfico del rendimiento del sistema.
- Interfaces de programación remota como: **DB-Library** y **ODBC**.
- Un complejo y robusto manejo de transacciones con capacidad de **roll-forward** y **roll-back** que permiten la recuperación automática de transacciones al reinicio del sistema.

En el pasado, las diferencias entre las distintas clases de software y hardware eran relativamente claras: *mainframe*, *minicomputadora* y PC. En la medida en que el software RDBMS se ha vuelto más poderoso y el hardware basado en PC para multiprocesamiento simétrico de alto rendimiento (SMP) más económico, los usuarios han encontrado mejores opciones para elegir la mejor plataforma de aplicaciones Cliente/Servidor. La razón más importante para elegir Microsoft SQL Server es la escalabilidad.

Como SQL Server se basa en un núcleo paralelo de base de datos **multithreaded** (multitareas), puede aprovechar procesadores adicionales. Hasta hace muy poco, los sistemas de hardware de multiprocesamiento eran costosos y complejos. En muchos casos era necesario emplear paralelamente software especial de Bases de Datos y Sistemas Operativos para obtener un mayor soporte

de multiprocesamiento simétrico (SMP). En los últimos cinco años a aumentado el poder del procesamiento disponible para los usuarios en el hardware de bajo costo. Hoy en día, SMP es una arquitectura dominante y SQL Server se enfoca a estas plataformas

En combinación con sistemas de hardware poderoso y de bajo costo de HP, Compaq, Digital y otras marcas, Microsoft SQL Server y Windows NT Server proporcionan un excelente rendimiento.

Después de haber analizado y propuesto la plataforma a utilizar para el desarrollo del **Sistema de Teleimagenología**, a lo largo del presente capítulo; se concluye que la plataforma que mejor se adapta a los requerimientos técnicos y económicos es un **Servidor**, (con **características mínimas** del HP NetServer E800), con el Sistema Operativo de Red **Windows NT Server versión 4.0** y como Manejador de Base de Datos se seleccionó a **MS SQL Server en su versión 7.0**.

HP Netserver E800

Hardware

- HP netserver E800 es un Servidor
- Doble procesador Pentium III 800 MHz
- Dos discos duros de 9.1 GB cada uno
- 384 MB SDRAM, expandible hasta 2 GB
- High Speed CD-ROM drive
- Unidad de Cinta

Software

- MS Windows NT 4.0
- MS SQL Server 7.0

Por lo tanto, el **software** del **Ciente** a utilizar, tiene que ser totalmente compatible con el software seleccionado hasta este momento para el Servidor, a fin de garantizar la operación y manejo del **Sistema de Teleimagenología Clínica** por parte de los usuarios Cliente; de igual forma proponemos que el Sistema Operativo sea un ambiente gráfico como lo es Windows, así que nuestro **Ciente** propuesto sería el siguiente.

Hardware

- Procesador Pentium III a 750 MHz
- Memoria RAM de 64 MB. Caché 512 KB
- Disco duro de 10 GB

Software

Windows (95, 98, 2000 Millenium o NT)

En el siguiente capítulo se analizará la arquitectura Ciente/Servidor, el Lenguaje de Programación para el desarrollo del sistema, el cual deberá ser perfectamente compatible con el sistema Operativo y con el RDBMS, así también, se creará la Base de Datos del sistema, se analizará la forma de accesar a los datos de la misma, se verá la seguridad en el sistema y la distribución del mismo.

VII. DESARROLLO E IMPLANTACIÓN DEL SISTEMA DE TELEIMAGENOLÓGÍA CLÍNICA

En el capítulo anterior, se analizó en que plataforma de hardware y software, se realizaría el sistema, en este capítulo se tratará lo relacionado a la arquitectura Cliente/Servidor (C/S); ya que uno de los requisitos para llevar a cabo este proyecto, es que el manejo del **Sistema de Teleimagenología Clínica**, sea sobre un ambiente Cliente/Servidor, además de que se pueda acceder a la información vía red del ISSSTE e Internet, pues se necesita que el Médico autorizado del ISSSTE pueda contar con ella para su consulta, desde cualquier parte del mundo. De igual manera en éste capítulo se trata lo referente al lenguaje de programación utilizado para el desarrollo del sistema, seguridad, y distribución del mismo.

7.1. ARQUITECTURA CLIENTE/SERVIDOR

En el capítulo anterior se mencionó que Windows NT y SQL Server, están basados en la arquitectura C/S, esto nos permite separar la carga de trabajo en tareas que corran en computadoras tipo Servidor y tareas que corran en computadoras tipo Cliente:

- El Cliente es responsable de la parte lógica y de presentar la información al usuario.
- El Servidor distribuye los recursos disponibles (tales como procesadores, memoria, operaciones de disco, etc.) entre las múltiples peticiones del Cliente. Véase la Fig. 7.1.

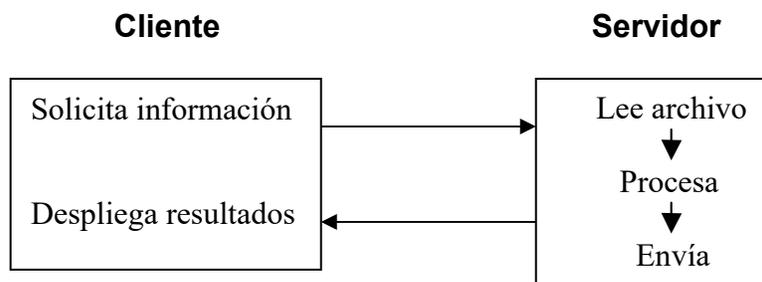


Fig. 7.1.

Por otro lado la arquitectura C/S, es el nuevo paradigma de la programación. Dentro de poco todo será C/S, como antes lo fue "orientado a

objetos". La programación Cliente/Servidor se utiliza cuando queremos realizar aplicaciones que utilicen redes y que comuniquen entre sí a varios equipos de cómputo. Ambas partes de la aplicación se comunican utilizando algún protocolo de red, en nuestro caso, el protocolo es el TCP/IP, el cual es el estándar en la red del ISSSTE. La justificación de este paradigma es la minimización del tráfico de red. El ancho de banda de las redes de datos, sobre todo de Internet, es actualmente el recurso más valioso por lo escaso y caro que es. Si hay que optimizar un programa para algo, ese algo es utilizar el menor tráfico de red posible para evitar que la red se haga lenta y de esta manera economizar ancho de banda.

La arquitectura computacional Cliente/Servidor es la primera solución en tecnología informática que satisface las presiones de costo-rendimiento de una empresa actual de alto nivel. Por esto se observa una clara tendencia de migración hacia este modelo.

Los motivos que nos llevaron a evaluar la posibilidad de implantar ésta arquitectura C/S dentro del desarrollo del sistema pueden ser varios, pero los principales son los siguientes:

Flexibilidad para satisfacer las necesidades del Sistema.

- Reducción de costos, en cuanto a la adquisición y aprovechamiento del equipo de cómputo y comunicaciones.
- Identificación de la solución que se adecuó más a las necesidades de la empresa y que crezca a la par de los requerimientos.
- Incremento a la productividad del personal.
- Aprovechamiento del equipo existente que no se utiliza al 100% de su capacidad.
- Consolidación de los procesos más confiables, ya que los recursos de cómputo pueden distribuirse en varios Servidores y los Clientes pueden configurarse de modo tal que cualquier aplicación o archivo de datos pueda ser accesado desde cualquier parte. Además, el aislamiento a fallas que se pueden incluir en este ambiente reduce la pérdida de tiempo.

Una ventaja adicional en el caso del proyecto en cuestión, es que se puede obtener un aprovechamiento total de los recursos dado que todos los equipos

(Servidores y Clientes), a pesar de no estar físicamente en el mismo lugar, estarán interconectados a través de la Red del ISSSTE.

Ventajas de la Arquitectura Cliente/Servidor:

- Con la arquitectura Cliente/Servidor se pueden integrar diferentes tipos de equipos y Sistemas Operativos en un ambiente único de procesamiento.
- Anteriormente, la capacidad de cómputo de una empresa se medía directamente por la capacidad del Mainframe o de las Minis con las que contaba. En el ambiente Cliente/Servidor la capacidad está construida por todos los equipos interconectados, entre los que se incluyen las PC's.
- Otra ventaja de esta arquitectura es la considerable reducción en el tráfico de la red, ya que las operaciones de procesamiento se jerarquizan y la red da prioridad a las necesidades del usuario, transmitiéndole sólo la información realmente útil. Esto constituye un estímulo al uso de sistemas abiertos dado que tanto Clientes como Servidores corren en diferentes plataformas de hardware y software, lo cual permite que las empresas compren productos de diferentes proveedores, sin interferir en el desempeño de las demás aplicaciones y equipos instalados en la red.

El ambiente Cliente/Servidor permite una estructuración, para que los recursos de cómputo se utilicen de la mejor manera posible, con lo cual se reducen los costos de manera considerable. Para lograr esto, se deben llevar a cabo ciertas medidas, las cuales se mencionan a continuación:

1. Separación de Tareas: Las funciones de cómputo se pueden sincronizar con los recursos de tal manera que el Servidor procese sólo las que son más complejas o que se requieren operaciones intensas, liberando al Cliente de esa labor, dándole la oportunidad de ejecutar otras tareas simultáneas. Con esta configuración puede integrarse una estrategia que equilibre la combinación y costo de los dispositivos de cómputo.
2. Comportamiento de Periféricos: La combinación de estos dispositivos, también puede realizarse al hacer que sus características coincidan con las funciones que se desea ejecutar y como la arquitectura C/S es abierta, permite la incorporación de un dispositivo especializado para realizar tareas muy específicas.

3. Comportamiento de Herramientas: Los Clientes pueden compartir diversas aplicaciones con la gran ventaja de que éstas pueden ser controladas de manera centralizada por un mismo Administrador. Con esta ventaja, se elimina la necesidad de tener que instalar, configurar y dar de alta la aplicación en cada Cliente cada vez que surja una nueva versión de software.

4. Acceso a la Información: A pesar de que una empresa se encuentre geográficamente separada, la arquitectura C/S permite el acceso a la información a nivel del Servidor de manera transparente, siempre y cuando existan estándares comunes.

Para el caso del presente proyecto, se pretende implantar un estándar, en cuanto al software que van a utilizar tanto el Servidor como los equipos Cliente. Esto es, se eligió como estándar una familia de productos que contara con una interfase gráfica, sencilla y amigable, para Administradores y Usuarios finales, tal como el ambiente Windows.

Por tal motivo, como ya se menciona en el capítulo anterior, el software ha utilizar para éste proyecto es de un sólo fabricante, la empresa Microsoft. En las estaciones Cliente, deberá instalarse MS Windows 95/98/2000/XP; mientras que los Servidores contarán con MS Windows NT Server 4.0 y MS SQL Server 7.0 como Sistema Operativo de Red y Manejador de Base de Datos, respectivamente.

En esta arquitectura, el Cliente maneja la interfase del usuario, construye los requerimientos del Servidor y controla la presentación de resultados, incluso puede contener la lógica de la aplicación. Cuando el Servidor recibe una solicitud de servicio, realiza el proceso, regresa el resultado al Cliente y provee el servicio de administración de datos. Tanto el Cliente como el servidor pueden ejecutar el trabajo, así como la lógica de la aplicación. De esta manera, el sistema a desarrollar, se desempeñará como se describe a continuación: el Servidor se encarga de ejecutar todos aquellos procesos que para el Cliente podrían representar mucho tiempo de proceso, de tal forma que el Servidor se encarga de realizar el trabajo pesado y el Cliente se dedica a presentar los resultados a los usuarios finales.

Por otro lado, la Arquitectura Cliente/Servidor presenta ciertas desventajas que a continuación se mencionan:

- Hay una alta complejidad tecnológica al tener que integrar una gran variedad de productos. Por una parte, el mantenimiento de los sistemas es más difícil pues implica la interacción de diferentes partes de hardware y de software, distribuidas por distintos proveedores, lo cual dificulta el diagnóstico de fallas.
- Es más difícil asegurar un elevado grado de seguridad en una red de Clientes y Servidores que en un Sistema Centralizado. Se deben hacer verificaciones en el Cliente y en el Servidor. También se puede recurrir a otras técnicas como el encriptamiento.
- En un ambiente Cliente/Servidor, los recursos de cómputo de la red enfrentan cargas de trabajo muy grandes respecto a los Sistemas Centralizados, por lo que el rendimiento no es el mismo.

En lo que se refiere al **Sistema de Teleimagenología Clínica**, la arquitectura Cliente/Servidor arrojará resultados muy favorables, ya que se pretende que el Servidor se encuentre en las oficinas centrales del ISSSTE y que los Clientes estén en todas y cada una de las dependencias del ISSSTE a nivel nacional, sin embargo, la información también se puede consultar desde cualquier parte del mundo.

7.2. LENGUAJE DE PROGRAMACIÓN

Después de haber elegido el Sistema Operativo de Red y el Manejador de Base de Datos Relacional, seleccionaremos la herramienta de desarrollo, para lo cual, ocuparemos MS Visual Basic 6.0, que como las herramientas anteriores, es de la empresa Microsoft, así aseguramos una perfecta integración entre las mismas y por lo tanto, tomar ventajas de sus características al desarrollar el **Sistema de Teleimagenología Clínica**. A continuación se estudiarán las características más sobresalientes de esta herramienta visual de desarrollo.

7.2.1. MS Visual Basic 6.0

MS Visual Basic 6.0 es uno de los Lenguajes de Programación que más entusiasmo despierta entre los programadores de PC's. Visual Basic 6.0 es también llamado lenguaje de cuarta generación (4GL), esto quiere decir, que un gran número de tareas se realizan sin escribir código, simplemente con operaciones gráficas realizadas con el ratón sobre la pantalla. Visual Basic 6.0 es también un programa basado en objetos, aunque no orientado a objetos como C++ o Java. La diferencia está en que Visual Basic utiliza objetos con propiedades y métodos, pero carece de mecanismos de herencia y polimorfismo propios de los verdaderos lenguajes orientados a objetos como C++ o Java.

Visual Basic, es un programa orientado a eventos, es decir, cuando Visual Basic se ha ejecutado, lo único que hace es quedarse a la espera de las acciones del usuario, que en este caso son llamadas **eventos**. El usuario dice si quiere abrir y modificar un archivo existente, o bien comenzar a crear un archivo desde el principio. Visual Basic, pasa la mayor parte del tiempo esperando las acciones del usuario (eventos) y respondiendo a ellas. Las acciones que el usuario puede realizar en un momento determinado son variadísimas, y exigen un tipo especial de programación: **la programación orientada a eventos**, Visual Basic 6.0 la hace especialmente sencilla y agradable. Visual Basic 6.0 está orientado a la realización de programas para **Windows**, pudiendo incorporar todos los elementos de este entorno informático: ventanas, botones, cajas de diálogo, cajas de texto, botones de opción, botones de selección, barras de desplazamiento, gráficos, menús, etc.

A continuación se explicarán, a detalle, algunos conceptos de la programación orientada a eventos.

7.2.2. Programación orientada a eventos

La aplicación Visual Basic de Microsoft puede trabajar de dos modos distintos: modo de Diseño y modo de Ejecución. En modo de Diseño el usuario construye interactivamente la aplicación, colocando controles en el formulario, definiendo sus propiedades, y desarrollando funciones para gestionar eventos.

La aplicación se prueba en modo de Ejecución. En este caso el usuario actúa sobre el programa (introduce eventos) y prueba cómo responde el programa. Hay algunas propiedades de los controles que deben establecerse en modo de diseño, pero muchas otras pueden cambiarse en tiempo de ejecución desde el programa escrito en Visual Basic 6.0. También hay propiedades que sólo pueden establecerse en modo de ejecución y que no son visibles en modo de diseño.

Todos los conceptos mencionados anteriormente se explicarán en los apartados siguientes:

- **Formularios y Controles:** Cada uno de los elementos gráficos que pueden formar parte de una aplicación típica de Windows 95/98/2000 es un tipo de **control**: los botones, las cajas de diálogo, las cajas de texto, las cajas de selección desplegadas, los botones de opción, los botones de selección los menús y otros tipos de elementos son controles para Visual Basic 6.0. Cada control debe tener un nombre a través del cual se puede hacer referencia a él en el programa. Visual Basic proporciona nombres por defecto que el usuario puede modificar.

En la terminología de Visual Basic 6.0 se llama **formulario** (form) a una ventana. Un formulario puede ser considerado como una especie de contenedor para los controles. Una aplicación puede tener varios formularios, pero un único formulario puede ser suficiente para las aplicaciones más sencillas. Los formularios deben tener también un nombre, que pueda crearse siguiendo las mismas reglas para los controles.

- **Objetos y Propiedades:** Los formularios y los distintos tipos de controles son entidades genéricas de las que pueden haber varios ejemplares concretos en cada programa. Cada formulario y cada tipo de control tienen un conjunto de **propiedades** que definen su aspecto gráfico (tamaño, color, posición en la ventana, tipo y tamaño de letra, etc.) y su forma de responder a las acciones del usuario (si está activo o no, por ejemplo). Cada propiedad tiene un **nombre** que ya viene definido por el lenguaje. Por lo general, las propiedades de un **objeto** son datos que tienen valores lógicos (true, false) o numéricos concretos, propios de ese objeto y distintos de las de otros objetos de su clase. Así pues, cada clase, tipo de objeto o control tiene su conjunto de propiedades, y cada objeto o control concreto tiene unos valores determinados para las propiedades de su clase. Se puede acceder a una propiedad de un objeto por medio del nombre del objeto a que pertenece, seguido de un punto y el nombre de la propiedad.
- **Nombres de Objetos:** En principio cada objeto de Visual Basic 6.0 debe tener un nombre, por medio del cual se hace referencia a dicho objeto. El nombre puede ser el que el usuario desee, e incluso Visual Basic 6.0 proporciona nombres por defecto para los diversos controles. Los nombres por defecto no son adecuados porque hacen referencia al tipo de control, pero no al uso que de dicho control está

haciendo el programador. Existe una convención ampliamente aceptada que es la siguiente: se utilizan siempre tres letras minúsculas que indican el tipo de control, seguidas por otras letras (la primera mayúscula, a modo de separación) libremente escogidas por el usuario, que tienen que hacer referencia al uso que se va a dar a ese control.

- **Eventos:** Ya se ha dicho que las acciones del usuario sobre el programa se llaman eventos. Son eventos típicos el oprimir un botón del ratón, el hacer doble click sobre el nombre de un archivo para abrirlo, el arrastrar un icono, o simplemente mover el ratón. Cada vez que se produce un evento sobre un determinado tipo de control, Visual Basic 6.0 arranca una determinada función o procedimiento que realiza la acción programada por el usuario para ese evento concreto. Estos procedimientos se llaman con un nombre que se forma a partir del nombre del objeto y el nombre del evento, separados por el carácter (`_`), como por ejemplo `txtBox_click`, que es el nombre del procedimiento que se ocupará de responder al evento `click`, en el objeto `txtBox`.

- **Métodos:** Los **métodos** son funciones que también son llamadas desde el programa, pero a diferencia de los procedimientos no son programadas por el usuario, sino que vienen ya pre-programadas con el lenguaje. Los **métodos** realizan tareas típicas, previsibles y comunes para todas las aplicaciones. De ahí que vengan con el lenguaje y que se libere al usuario de la tarea de programarlos. Cada tipo de objeto o de control tiene sus propios métodos.

- **Proyectos y Archivos:** Cada aplicación que se empieza a desarrollar en Visual Basic 6.0 es un nuevo **proyecto**. Un **proyecto** comprende otros componentes más sencillos, por ejemplo los **formularios** (que son las ventanas de la interfase de usuario de la nueva aplicación) y los **módulos** (que son conjuntos de funciones y procedimientos sin interfase gráfica de usuario). Un proyecto se compone siempre de **varios archivos** (al menos dos) y hay que preocuparse de guardar cada uno de ellos en el directorio adecuado y con el nombre adecuado. Existe siempre un archivo con extensión `*.vbp` (Visual Basic Project), éste archivo del proyecto contiene toda la información de conjunto. Además hay que crear un archivo para **cada formulario y por cada módulo** que tenga el proyecto. Los archivos de los formularios, con la extensión `*.frm` y los segundos

con la extensión ***.bas** y si son módulos de clase con la extensión ***.cls**.

En el **Sistema de Teleimagenología Clínica**, Visual Basic 6.0, se ha utilizado como lenguaje de desarrollo final, esto es, la aplicación desde el Cliente hace consultas hacia un Servidor de base de datos (SQL Server7.0); éste último realiza las tareas que le piden los Clientes autenticados y regresa resultados a los mismos para que sean desplegados por los usuarios finales; de esta forma se aprovecha al máximo la arquitectura Cliente/Servidor entre equipos remotos.

7.3. CREACIÓN DE LA BASE DE DATOS PARA EL SISTEMA DE TELEIMAGENOLÒGIA CLÍNICA

En este proyecto, primero creamos la Base de Datos con el Administrador de Base de Datos SQL Server 7.0, para después desarrollar el sistema con Visual Basic 6.0.

El almacenamiento de los datos es la esencia de los sistemas de información. Los objetivos generales del diseño de la organización del almacenamiento de datos son que se logre:

- Disponibilidad de los datos
- Almacenamiento eficiente de los datos
- Actualización eficiente
- Recuperación dirigida de la información

Para que el almacenamiento de los datos sea eficiente, se debe vigilar el cumplimiento de algunos requisitos. Primero, los datos deben estar disponibles para cuando el usuario final desee usarlos. Segundo, los datos deben de ser precisos y consistentes (la base de datos debe tener una integridad). Tercero, el almacenamiento, actualización y grabado de los datos debe de ser ágil, sencillo y confiable.

La esencia de una base de datos es el Sistema Administrador de la Base de Datos, el cual permite la creación, modificación y actualización de la

información; la recuperación de los datos y la emisión de los reportes. A la persona responsable de asegurar que la información satisfaga los objetivos programados se le denomina Administrador de la Base de Datos. Una de las ventajas de las bases de datos, es que al compartir los datos se realizará de manera más sencilla y confiable si éstos aparecen una vez y no en varios archivos, dado que esto ocasionaría redundancia en los datos. Finalmente el enfoque de base de datos, tiene la ventaja de permitir que los usuarios expongan sus puntos de vista sobre los datos, sin necesidad de preocuparse de la estructura presente de la base de datos o de su ubicación física. No obstante los beneficios del enfoque de base de datos, también deben considerarse las desventajas que presenta, entre estas últimas, las que resaltan son las siguientes: los datos deben almacenarse en un sólo lugar, lo que trae como consecuencia que los datos sean más vulnerables a accidentes y, por tanto, se hará necesario un proceso de respaldo completo y constante. Además, también existe el riesgo de que la persona que administre la base de datos pueda convertirse en el único privilegiado o habilitado para estar cerca de la información, esto justificable hasta cierto punto, por cuestiones de seguridad.

Finalmente, el modelo lógico de la base de datos debe transformarse en su correspondiente diseño físico, que es lo que nos ocupa en éste momento; el diseño físico considera la forma de almacenamiento de los datos y de sus interrelaciones, a sí como la mecánica de acceso a los datos.

Existen tres tipos básicos de base de datos con una estructuración lógica: el tipo jerárquico, el de red y el de relación. Para el **Sistema de Teleimagenología Clínica**, se utilizó el enfoque relacional, ya que además de ser el más conocido y utilizado, es el que mejor se adapta a las características del software que se eligió para el desarrollo del presente proyecto, en este caso es el RDBMS (SQL Server 7.0).

Una base de datos relacional consiste en una o más tablas bidimensionales, las cuales se refieren como relaciones. Los renglones de las tablas representan los registros y las columnas contienen los atributos. El mantenimiento de las tablas de una base de datos relacional es bastante simple al compararlo con el mantenimiento de una estructura jerárquica o de red. Una de las ventajas principales de la estructura relacional, es que las consultas específicas se manejan de una manera muy eficiente.

Con el fin de que las estructuras relacionales sean útiles y manejables, las tablas relacionales primero deben “normalizarse”. La normalización es el proceso de transformación de los almacenamientos de datos (tablas), en conjuntos estables de estructuras de datos de menor tamaño, además de ser más sencillas, tales estructuras son más estables. Las estructuras de datos normalizadas son más fáciles de mantener. Una vez que están normalizadas las tablas, el siguiente paso dentro del proceso de desarrollo del presente sistema, fue construir la base de datos, para esto, nos basamos en el diccionario de datos que se presento en el capítulo V. Para construir la base de datos, se utilizó una herramienta de SQL

Server llamada SQL Enterprise Manager, la cual cuenta con una interfaz gráfica, vía esta herramienta, realizamos toda la administración de las bases de datos del sistema y las de los usuarios como la que se construyó, para efectos del presente proyecto. Para poder llevar a cabo dicha tarea, fue necesario realizar los siguientes pasos:

1) Crear los dispositivos (devices) de almacenamiento y respaldo.

Los devices son archivos donde va a ser almacenada la base de datos. Existen tres tipos diferentes de devices: el de datos, el de log y el de respaldo. En los dos primeros residen los datos de la base de datos, mientras que el tercero se ocupa para crear los respaldos. Para el **Sistema de Teleimagenología Clínica** se creó un device de datos de 500 MB y un device de log de 50 MB, actualmente la base tiene 88 pacientes almacenados y ocupa 9 MB de espacio, debido a que se tiene planeado tener aproximadamente 4000 pacientes registrados, la base de datos tiene la posibilidad de crecer aún más, esto es, que no está restringido su crecimiento a 500 MB. El Administrador de la base de datos es quien decidirá si restringe el crecimiento de esta o no, para realizar esta operación únicamente se tiene que hacer uso de la herramienta de administración y escribir los datos de acuerdo a las necesidades tal como lo muestra la figura siguiente:

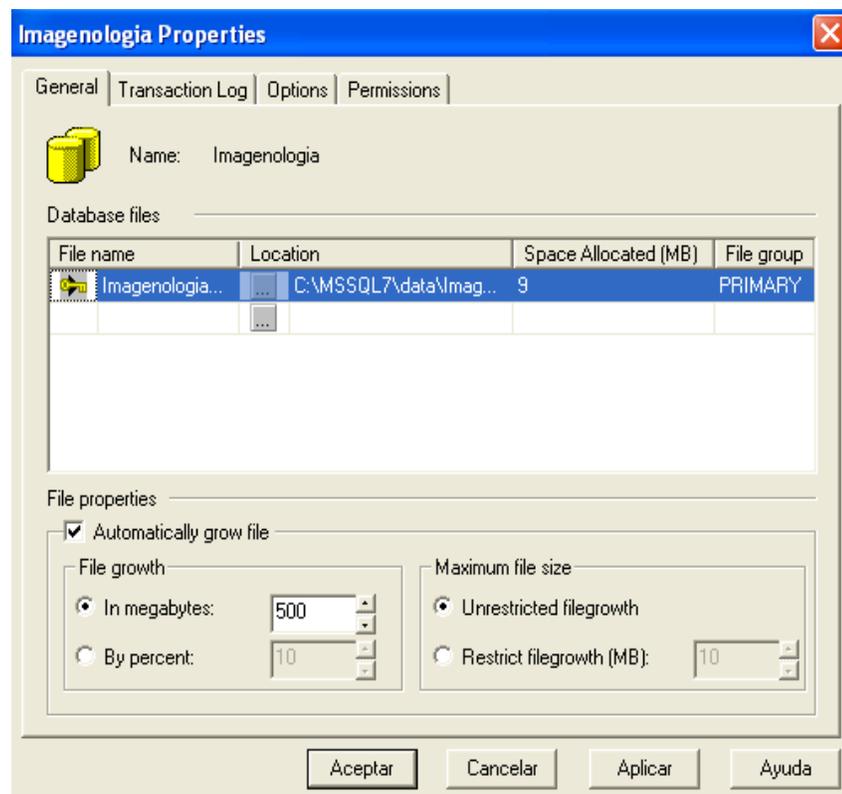


Fig. 7.2.

2) Generar las bases de datos.

Una vez que los devices fueron creados, se procede a la creación de la base de datos, donde únicamente se le tiene que indicar qué device se utilizará para guardar los datos y cuál para el log.

3) Crear las tablas.

Una vez generada la base de datos se tiene que crear, una a una, cada tabla que utiliza el sistema. La siguiente figura es un ejemplo:

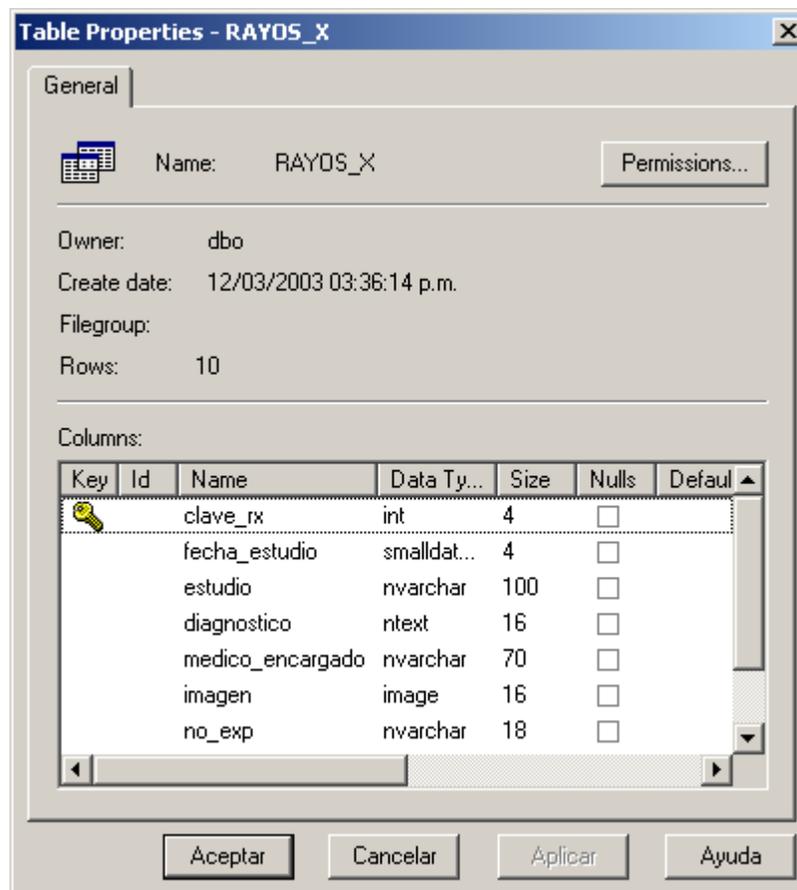


Fig. 7.3.

4) Poner datos en las tablas.

Para agregar datos a la tabla hay dos formas, con el asistente o bien con el comando Insert.

5) Crear usuarios y definir grupos.

Al igual que en el Sistema Operativo (Windows NT), en el RDBMS, se deben dar de alta los usuarios y los grupos de usuarios que tendrán acceso a los diferentes objetos de bases de datos que existan en el Servidor (Fig. 7.4.). Para el presente sistema se crearon 3 tipos de usuarios: A) El Administrador que tiene todos los privilegios sobre la base de datos. B) El usuario que tiene permisos de sólo lectura sobre la base y C) El usuario que tiene permisos de escritura.

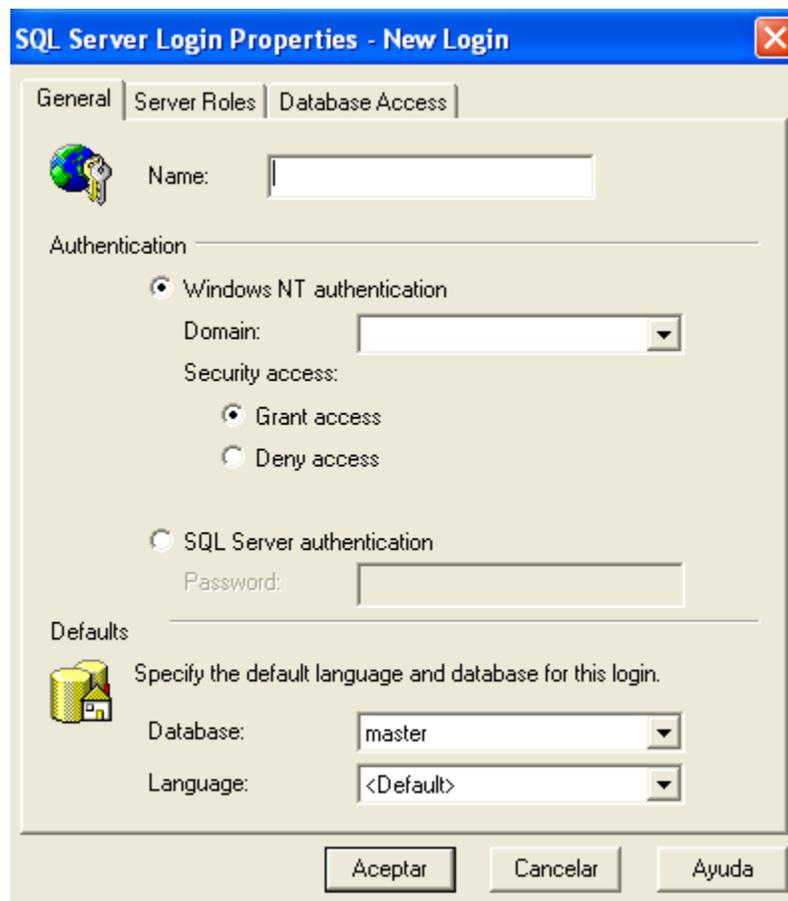


Fig. 7.4.

Para el manejo de claves de usuario dentro del sistema, se creó una tabla llamada Médicos, (debido a que los usuarios son Médicos) donde se tiene la información más importante relacionada a cada usuario como nombre, clave, contraseña, curp, nivel, etc. La creación de dicha tabla, obedeció fundamentalmente a que no era conveniente crear una cuenta de usuario al nivel de Windows NT o SQL para cada usuario que fuera accesar al sistema. Esto se debe a que con el uso de ODBC, el Cliente puede comunicarse a la base de datos utilizando cualquier herramienta que haga uso de ODBC, lo que afectaría severamente el carácter confidencial de la información.

6) Asignar permisos a los objetos (base de datos, tablas, etc.).

Los permisos son las acciones que un usuario o un grupo de usuarios, tienen permitido efectuar para un determinado objeto de la base de datos. Un usuario puede tener permiso para SELECT e INSERT sobre registros en una tabla determinada, pero puede carecer de permisos para DELETE o UPDATE. Para otorgar permisos a los objetos y establecer un esquema de seguridad en una base de datos, se utiliza el comando GRANT con el que se especifica qué usuarios tienen qué permisos sobre qué objetos de la base de datos.

7.4. ACCESO A DATOS CON MS VISUAL BASIC 6.0

Ya que se tiene la base de datos y en ella la información necesaria, el primer paso para tener acceso a la información de una base de datos consiste en establecer una conexión con el origen de datos.

Visual Basic, puede acceder a los datos contenidos en una base de datos, vía tres tecnologías. A continuación se explica que es cada una de ellas y cuales son sus ventajas en su uso:

- a) **DAO** (Data Access Objects), es el primer modelo de objetos de acceso a datos que presentó Visual Basic en su versión 3. esta tecnología permite una completa administración de la base de datos, muy en especial del formato (MDB Access) nativo. Cuenta con

capacidades transaccionales, multiusuarios, seguridad (suficientes en muchos casos), y de red. Su uso es recomendado para bases de datos locales y en entornos de red donde la concurrencia de usuarios no sobrepase los 15 ó 20, además de que la cantidad de datos no sobrepase 1 GB.

- b) **RDO** (Remote Data Objects), es una interfaz de acceso a datos para ODBC orientada a objetos, que incorpora un estilo sencillo de DAO y cuya interfaz expone prácticamente toda la flexibilidad y eficacia de bajo nivel de ODBC. Sin embargo, RDO presenta limitaciones al no proporcionar un acceso apropiado a las bases de datos Jet o ISAM, y al solo permitir el acceso a bases de datos relacionales a través de los controladores ODBC existentes.

- c) **ADO** (ActiveX Data Objects), es el sucesor de DAO y RDO. ADO es un modelo más fino que DAO. Muchas de las funcionalidades de DAO están consolidadas en un solo objeto y de ésta manera ADO ofrece un modelo de objetos más simple. Accesar datos a través de ADO es la visión de Microsoft para el futuro. ADO es mejor para aplicaciones Cliente/Servidor, también permite un fácil acceso a muchas fuentes de bases de datos. Visual Basic 6.0 y SQL Server 7.0, trabajan con la tecnología ADO de una manera natural.

ActiveX Data Objects (ADO) es una tecnología amigable y de fácil uso para agregar acceso a bases de datos. Utilizamos ADO para escribir secuencias de comandos compactas y escalables que conecten con bases de datos compatibles con Open Database Connectivity ODBC (Conectividad Abierta de Base de Datos) y orígenes de datos compatibles con Object Linking & Embedding Data Base (OLE DB).

ADO, sin embargo, no se comunica directamente con la base de datos, sino que accede a ella a través de una interfase intermediaria, estas interfases son ODBC y OLE-DB. ODBC, fue creado para acceder en lo posible a cualquier tipo de datos siempre y cuando sea del tipo base de datos relacional, mientras que OLE-DB, puede acceder a cualquier tipo de datos, incluido, claro esta, las bases de datos relacionales. El proveedor OLE-DB, también permite acceder a bases de datos a través de ODBC, que es el caso del presente proyecto, siendo así que, ocuparemos el proveedor "OLE-DB provider for ODBC", el cual nos permite conectarnos a cualquier fuente de datos ODBC.

Antes de crear secuencias de comandos para bases de datos se tiene que proporcionar una forma para que ADO encuentre, identifique y se comunique con una base de datos; esto es, se debe crear un archivo de acceso a datos, que no es otra cosa que, el controlador de bases de datos (programas que pasan información desde una aplicación, para este proyecto en Visual Basic, a una base de datos, es este caso SQL Server), éstos utilizan un Data Source Name (Nombre de Origen de Datos), para encontrar e identificar una base de datos ODBC en particular. Normalmente, los DSN contienen información de configuración de la base de datos, seguridad de usuarios y ubicación, y puede tener la forma de una entrada en el registro del sistema de Windows 95, 98, 2000 y NT o de un archivo de texto.

Existen tres tipos de DSN: *Usuarios*, *Sistema* o *Archivo*; enseguida se explica en que consiste cada uno de éstos.

Los DSN de *usuario* y de *sistema* residen en el registro del sistema de Windows 95, 98, 2000 y NT. Los DSN de *sistema* permiten que todos los usuarios que han iniciado una sesión en un Servidor concreto tengan acceso a una base de datos, siempre y cuando tengan permisos de Administrador; mientras que los DSN de *usuario* limitan la conectividad con la base de datos a los usuarios que hayan creado su propio DSN, es decir, van a poder acceder a la base de datos aquellos usuarios que dispongan de una Clave y Contraseña válidas en SQL Server. Los DSN de *archivo*, que tienen la forma de archivos de texto, proporcionan acceso a varios usuarios y son fácilmente transferibles entre un Servidor y otro mediante la copia de los archivos DSN. Para el **Sistema de Teleimagenología Clínica**, ocuparemos los DSN de usuario, ya que nos garantizan seguridad en el acceso al sistema, o sea que, solamente tiene acceso a la base de datos quien tenga una Cuenta o Clave y Contraseña válidas en el sistema.

Como se mencionó anteriormente, el primer paso para tener acceso a una base de datos es, establecer una conexión con el origen de datos. ADO, requiere un proveedor (data consumer) que pueda entender la estructura de los datos a leer y nos la entregue en formato "recordset" para nuestra manipulación. ADO proporciona el objeto **Connection**, que es el objeto que se utilizará para establecer y administrar las conexiones, de nuestra aplicación y la base de datos SQL Server. El objeto **Connection** incorpora diversas propiedades y métodos que utilizamos para abrir y cerrar conexiones con la base de datos, y para enviar consultas de actualización de la información, estas consultas, se realizan con el lenguaje de consulta estructurado (SQL). Aunque el objeto **Connection** simplifica la tarea de conectar con una base de datos y emitir una consulta, el objeto **Connection** tiene sus limitaciones. En concreto, con el objeto **Connection** no se pueden crear secuencias de comandos que recuperen y presenten información de una base de datos; se tiene que saber exactamente las modificaciones que desea realizar en la base de datos y después implementar las modificaciones como consultas.

Para recuperar datos, examinar resultados y modificar la base de datos del **Sistema de Teleimagenología Clínica**, ADO tiene el objeto **Recordset**. El objeto **Recordset** tiene las funciones necesarias para, dependiendo de las restricciones de las consultas, recuperar y presentar un conjunto de filas, o *registros*, de una base de datos. El objeto **Recordset** mantiene la posición de cada registro devuelto por la consulta, lo que permite “recorrer” los resultados de uno en uno.

Concluyendo, para la realización del presente trabajo, se ocupó el lenguaje de programación MS Visual Basic 6.0 con la tecnología de acceso a datos ADO, como herramienta de desarrollo, esto nos presentó ventajas, ya que se pudo obtener el máximo provecho de la tecnología propuesta en los capítulos anteriores, como es el Sistema Operativo de Red y el RDBMS, así como la arquitectura Cliente/Servidor, utilizando el protocolo de comunicación TCP/IP; sencillamente la justificación de la utilización de la tecnología antes mencionada, es por razones de rendimiento y minimización del tráfico en la red.

Ya que se logró agregar, modificar y analizar la información de la base de datos, se hace necesario, recuperarla y presentarla en pantalla, o bien de una forma impresa y/o almacenarla en disco; para esto, Visual Basic cuenta con una herramienta generadora de reportes que es el Microsoft Data Report V.6.0, que nos permite realizar reportes sencillos de una manera rápida.

7.5. SEGURIDAD DEL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA

Las consideraciones sobre seguridad en los sistemas informáticos son un componente cada vez más importante en el desarrollo de los sistemas de información de la actualidad. Los cambios generados por las redes informáticas y las nuevas tecnologías introducen nuevos enfoques, riesgos y requerimientos que hacen necesario incorporar la seguridad en el proceso del diseño.

Debido a que el sistema trabaja sobre la Red, la información de la base de datos está expuesta a todos los usuarios de Internet, tanto nacionales como internacionales, y puede existir alguien que con malas intenciones intente filtrarse, por tal motivo, es de suma importancia que ninguna persona ajena, pueda consultar, modificar o borrar información de la base de datos del sistema.

SQL Server está integrado con el sistema de seguridad de Windows NT. Esta integración permite acceder tanto a Windows NT como a SQL Server con la misma Clave y Contraseña, si así lo decide el Administrador del Sistema. Además SQL Server une las características de encriptación que tiene Windows NT para la seguridad en red.

SQL Server valida a los usuarios con 2 niveles de seguridad; **1) Autenticación de la clave y 2) Validación de permisos** en la base de datos de cuentas de usuario y de roles. La autenticación identifica al usuario que está usando una cuenta y verifica sólo la habilidad de conectarse con SQL Server. El usuario debe tener permiso para acceder a determinada base de datos en el Servidor. Los permisos controlan las actividades que el usuario tiene permitido realizar en la base de datos del SQL Server.

1) Autenticación de la clave: Un usuario debe tener una cuenta para conectarse al SQL Server. Este reconoce 2 mecanismos de autenticación: Autenticación de SQL Server y de Windows NT. Cada uno tiene un diferente tipo de cuenta.

- Autenticación de SQL Server: Cuando se usa, un Administrador del Sistema SQL Server define una Clave y una Contraseña, para determinado usuario que necesite acceder a la base de datos. El usuario debe suministrar tanto la Clave como la Contraseña cuando se conecta al SQL Server, vía ODBC.
- Autenticación de Windows NT: Cuando se usa, el usuario no necesita de una cuenta de SQL Server, para conectarse a la base de datos, siempre y cuando esta cuenta tenga privilegios de Administrador, de igual manera esto es mediante ODBC.

Cuando SQL Server está corriendo en Windows NT, como es el caso del presente proyecto, en el sistema Administrador de SQL Server podemos especificar que modo de autenticación vamos a utilizar, estos modos con los siguientes:

- Modo de autenticación de Windows NT.- Sólo está autorizada la autenticación de Windows NT. Los usuarios no pueden usar cuentas de SQL Server.
- Modo Mixto.- Cuando se usa este método de autenticación, los usuarios se pueden conectar a SQL Server con la autenticación de Windows NT o con las de SQL Server.

2) Validación de permisos en la base de datos de claves de usuario y de roles: Después de que los usuarios han sido autenticados, y por consiguiente se les ha permitido conectarse al SQL Server, deben tener Claves válidas en la base de datos. Las claves de usuario y los roles, identifican permisos para ejecutar tareas. Las cuentas de usuario son específicas para cada base de datos existentes en el SQL Server. Los roles nos permiten reunir a los usuarios en una sola unidad a la cual se le pueden aplicar permisos. Dentro de cada base de datos, se asignan permisos a las Cuentas de usuarios y a los roles para permitir o limitar ciertas acciones. SQL Server acepta comandos después de que un usuario ha accedido a la base de datos.

SQL Server realiza los siguientes pasos cuando valida permisos: 1) Cuando el usuario realiza una acción, tal como ejecutar un comando, o elegir una opción de un menú, los comandos son enviados al SQL Server. 2) Cuando SQL Server recibe un comando, checa que el usuario tenga permisos de ejecutar dicha instrucción. 3) Después, SQL Server realiza cualquiera de las siguientes acciones: a) Si el usuario no tiene los permisos adecuados, SQL Server devuelve un error. b) Si el usuario tiene los permisos adecuados, SQL Server realiza la acción.

7.6. DISTRIBUCIÓN DEL SISTEMA VÍA FTP

El sistema, hasta aquí, cuenta con la seguridad necesaria, para que ningún intruso pueda acceder a la base de datos sin los permisos necesarios. Por tanto, está listo para que nuestros Clientes puedan instalar en sus sistemas locales la aplicación (**Sistema de Teleimagenología Clínica**). Dado que el sistema tiene que trabajar sobre la red del ISSSTE e Internet se hace necesario que el Servidor que contiene nuestra base de datos, sea un Servidor FTP, para que los Clientes en donde quiera que estén, puedan conectarse a él y copiar los archivos necesarios en sus sistemas locales, para posteriormente, instalar la aplicación. Aquí se tienen que poner dos cosas en claro; una es el cómo se va a hacer el Servidor FTP y dos cómo es que se van a empacar y distribuir los archivos necesarios, para que los clientes puedan instalar la aplicación en cuestión.

Primero, veamos lo referente al **Servidor FTP**, al momento de estar instalando el Sistema Operativo Windows NT, se nos pregunta si queremos

instalar el IIS (Internet Information Server), y únicamente hay que seleccionar el o los servicios que se quieran habilitar como son:

- Gopher: Menús jerárquicos para búsqueda de información en Internet a través de conexiones transparentes al usuario.
- FTP: Transferencia de archivos de una máquina remota a otra local o viceversa.
- WWW (World Wide Web): Es uno de los servicios más interesantes y completos que se pueden encontrar en Internet. Este servicio aprovecha la tecnología multimedia para presentar información mediante el uso de imágenes, texto y sonido desde cualquier punto de Internet a la computadora del usuario.

FTP fue uno de los primeros protocolos usados en las redes TCP/IP y en Internet. FTP se utiliza para transferir archivos de un equipo de una red a otro equipo de cualquier red, para copiar archivos desde un equipo Cliente a un servidor sólo se puede usar FTP. En el presente proyecto, los Clientes remotos necesitan transferir archivos en ese sentido, así que se hace necesario usar FTP. Además, como se requiere que los archivos de la aplicación estén disponibles para todos los usuarios remotos, FTP es un servicio extremadamente fácil de instalar y mantener. Una vez instalado, se asigna el servicio FTP a los archivos; y no es necesaria ninguna configuración adicional. Los archivos disponibles mediante FTP pueden tener cualquier formato, como archivos de documentos, archivos multimedia o archivos de aplicación. El servicio FTP requiere que los usuarios inicien una sesión para poder usar el servicio. Como se puede apreciar, desde el momento en el que los Clientes quieren copiar los archivos de la aplicación, en sus sistemas locales, tienen que contar con una Clave y Contraseña válidos para poder acceder al Servidor de FTP y contenedor de la base de datos. De igual forma, se puede configurar el número de conexiones simultáneas permitidas y el periodo de tiempo permitido para las conexiones. Para configurar los requisitos de inicio de sesión para el servicio FTP se usa el Administrador de servicios de Internet (IIS). Si el servicio FTP se configura para inicios de sesiones anónimas, los Clientes pueden iniciar sesiones con el nombre de usuario "anonymous", cosa que no es conveniente por cuestiones de seguridad. Los Clientes de FTP también pueden iniciar sesión con un nombre de Usuario y una Contraseña de Windows NT, que es la manera como se puede acceder al Servidor de FTP que contiene nuestra aplicación, esto es, en **Propiedades del servicio FTP** del Administrador de Servicios de Internet (IIS), se desactivan las casillas de verificación **Permitir conexiones anónimas** y **Permitir sólo conexiones anónimas**, para evitar que usuarios ajenos al sistema accedan al Servidor IIS.

Al desactivar estas casillas de verificación, cualquier cuenta que sea "anónima" no podrá iniciar sesión. Esto es útil por cuestiones de seguridad; de tal forma que sólo las cuentas válidas en el sistema, que para tal efecto haya creado el Administrador del Servidor, tendrán acceso al Servidor IIS. Además estas

cuentas deben ser validadas en el Administrador de Usuarios de Windows NT, debiendo contener ciertos privilegios, para poder hacer uso del servicio de FTP.

La manera de configurar los directorios de FTP, es la siguiente: De forma predeterminada, todos los subdirectorios del Directorio Particular están disponibles para el servicio de FTP, de tal manera, que sólo debemos poner los archivos o subdirectorios en el Directorio Particular que queramos compartir con los Clientes, para nuestro caso se compartirán los archivos de la aplicación y todo lo que está en torno a ella.

Por otra parte, el servicio de FTP, cuenta con los permisos de lectura y escritura para el Directorio Particular de forma predeterminada, para el caso de la aplicación, se manejó el permiso de lectura, ya que los Clientes no tienen porque modificar la información que se les comparte, únicamente tienen que copiarla a sus sistemas locales, para posteriormente instalar, si así lo desean el **Sistema de Teleimagenología Clínica**, en sus sistemas locales.

En segundo término veamos cómo se empaquetará y distribuirá la aplicación en los Clientes remotos.

Visual Basic 6.0, cuenta con un Asistente de empaquetado y distribución, el cual nos permite crear paquetes de instalación para nuestra aplicación; es necesario haber guardado, compilado y creado el ejecutable, antes de ejecutar este asistente. El asistente de empaquetado y distribución ofrece numerosas opciones para la distribución de la aplicación, incluyendo la creación de:

- Paquetes estándar, diseñados para ser instalados mediante un programa de instalación (setup.exe), que es el caso del **Sistema de Teleimagenología Clínica**.
- Paquetes de Internet, diseñados para ser descargados desde un sitio Web
- Archivos de dependencia que puede distribuir con sus componentes.

El asistente comprime los componentes de la aplicación en un paquete formado por archivos "cabinet" (.cab). Se pueden crear uno o más archivos (.cab), dependiendo del tamaño de la aplicación.

Empaquetar una aplicación consiste en crear un paquete que permita instalar la aplicación, en el equipo del usuario final. Por otro lado la distribución del paquete supone la creación de los medios de distribución y copia de todos los archivos necesarios a una ubicación donde los usuarios puedan tener acceso a ellos, para esto, está el Servidor FTP, el cual es el Servidor de Distribución y es ahí donde también se aloja la base de datos, para nuestra aplicación en cuestión.

La primera etapa en la creación de un programa de instalación personalizado es determinar los archivos que deben empaquetarse y distribuirse. Todas las aplicaciones de Visual Basic, requieren un conjunto mínimo de archivos, conocido como archivos de preinstalación, que son necesarios antes de poder instalar la aplicación. Además, todas las aplicaciones de Visual Basic necesitan archivos específicos de la aplicación como un archivo ejecutable (.exe), archivo de datos, controles Activex (.ocx), mapa de bits (archivos .bmp) o archivos de librerías (.dll), es decir, existen tres categorías principales de archivos necesarios para ejecutar y distribuir nuestra aplicación: 1) Archivos de tiempo de ejecución. 2) Archivos de instalación. 3) Archivos específicos de la aplicación.

Después de llevar a cabo el proceso de empaquetado y de generar el medio de distribución de la aplicación, se probó el programa de instalación, asegurándose de hacerlo en un equipo que no tuviera Visual Basic. Una vez que se terminó con la instalación, se ejecutó el programa, para comprobar que se comportaría de la forma esperada.

7.7. PROPUESTA DE IMPLANTACIÓN

Dado que el objetivo principal a alcanzar es la creación de un sistema seguro y eficiente para el manejo de la información por computadora, reducir el tiempo de espera en las consultas de los expedientes, apoyar las consultas y segundas opiniones médicas, para consulta de bancos de imágenes, de una forma sencilla, amigable, mediante el empleo de gráficas y de acceso controlado, y siendo que ésta información tendrá que ser manejada a través de la red del ISSSTE e Internet, tanto para consulta como para actualización, a lo largo de los pasados capítulos se ha propuesto el uso de las tecnologías, tanto en hardware como en software, que más se adecuan a las necesidades, para poder cumplir con todas las expectativas que debe tener el **Sistema de Teleimagenología Clínica**; por lo anterior, en términos generales nuestra propuesta de solución es la siguiente:

- 1) Para alcanzar el objetivo, se planteó el desarrollo de un sistema bajo la arquitectura Cliente/Servidor, que contará con un origen de datos sobre un equipo que soportará el multiprocesamiento simétrico utilizando como Sistema Operativo de Red a Windows NT Server y a SQL Server como Administrador de Base de Datos Relacionales y el protocolo TCP/IP de comunicación entre los Clientes y el Servidor. En cuanto al equipo Cliente, se utilizará la interfase gráfica Windows en cualquiera de sus versiones (95, 98, 2000 NT o Millenium, XP), y Visual Basic como Lenguaje de Programación. En este sistema,

cuando el Servidor recibe una solicitud de servicio, realiza el proceso, regresa el resultado al Cliente y provee el servicio de administración de datos. De esta manera, el **sistema**, se desempeñará como se describe a continuación: El Servidor se encarga de ejecutar todos aquellos procesos que para el Cliente podrían representar mucho tiempo de proceso, de tal forma que el Servidor se encarga de realizar el trabajo pesado y el Cliente se dedica a presentar los resultados a los usuarios finales, mediante la aplicación realizada en Visual Basic. Mediante el **Sistema de Teleimagenología Clínica** se pretende el compartir los recursos que se tienen, tanto de software y hardware, en la red del ISSSTE, y por lo tanto en Internet, esto a su vez, trae como consecuencia que haya un mejor rendimiento y minimización del tráfico en la red.

- 2) Los siguientes puntos presentan las características a detalle que debe cumplir el **Sistema de Teleimagenología Clínica**.
 - I. Manejo bajo un ambiente Cliente/Servidor.- Originalmente, el sistema se pensó bajo una plataforma C/S, la cual permitiría la correcta distribución de las funciones de una máquina potente (Servidor) y otra con menos potencialidad (Cliente), esto es, los datos debían residir en el Servidor, quien puede procesarlos con más facilidad, mandando al Cliente la información ya procesada y éste último presentarla al usuario final, quien a su vez tendría la posibilidad de guardarla en un archivo, consultarla en pantalla o bien imprimirla según sus necesidades.
 - II. Acceso a la información vía red del ISSSTE e Internet.- El sistema presentaría, una característica muy importante y especial, ésta es que la aplicación, debía correr sobre cualquier computadora Cliente conectada a Internet, la única restricción es que los Clientes contarán con una interfase gráfica, como lo es Windows en cualquiera de sus versiones. Por lo anterior, tanto el Servidor como el Cliente debían usar un protocolo de comunicación estándar (TCP/IP).
 - III. Soporte multiusuario.- Otra consideración importante, es que el sistema daría servicio a un gran número de usuarios, todo aquel médico que tuviera permisos, en cada una de las dependencias del ISSSTE a nivel nacional.

- IV. Medidas de seguridad en el acceso a la información.- La seguridad dentro del sistema juega un papel muy importante y debido a que el sistema trabaja en Internet, la información de la base de datos está expuesta a todos los usuarios de Internet a nivel mundial, por lo tanto, no se descarta la idea de que alguien con malas intenciones intente filtrarse. Es por esto, que el sistema cuenta con una doble seguridad de acceso, una de ellas es la que nos ofrece Windows NT y por otra parte el sistema controla el nivel de acceso a los usuarios válidos, esto es, para poder entrar el sistema se tiene que configurar un ODBC, mismo que pide una Clave y Contraseña que el usuario tiene que saber, aparte dependiendo del tipo de Clave es como va a poder hacer movimientos dentro del sistema. En cuanto al nivel de disponibilidad de la información, para el **Sistema de Teleimagenología Clínica**, se realizan varias medidas de seguridad, como el monitoreo constante de los movimientos que se realizan en la base de datos; así como un programa de respaldo de la información en cinta diario, del Sistema Operativo y la Base de Datos.
- V. Medidas de seguridad para la integridad de la información.- En los niveles de integridad y consistencia, se protege la información a ser consultada o modificada; además de que el sistema se comporte en la forma esperada, estos mecanismos suelen ser transparentes para los usuarios, ya que desde el diseño de la base de datos, se estructuro de tal manera, que fuera estable y lógica, para evitar redundancia de la información e inconsistencia en los datos.
- VI. Consulta e impresión.- La aplicación tenía que contar con la opción de impresión de la información consultada en pantalla (reportes), una vez que el usuario realizara determinada búsqueda en el sistema.
- VII. Respaldo de la información consultada en el Disco Duro o en Disquetes.- El **Sistema de Teleimagenología Clínica** cuenta con la posibilidad de que la información visualizada en pantalla (reportes), pueda ser almacenada en el disco duro de la máquina o bien en disquetes o cualquier parte de la red local del sistema propio.

7.8. ALCANCES

Debido a que el ISSSTE cuenta con varias dependencias en toda la Republica Mexicana, el uso del **Sistema de Teleimagenología Clínica** estaría facilitando el que cada una de las dependencias contaran con un sistema que almacene, despliegue y lleve el orden de cualquier imagen que se utilice para el diagnóstico clínico, como son: Radiografías, Tomografías, Resonancias Magnéticas, Mastografías, etc.

En resumen, se lograría la vinculación integral entre todos los campos de la Imagenología y sus actividades; siendo así que se contaría con un sistema que automatiza la consulta de los expedientes de los diversos pacientes de IMAGENOLOGIA para así reducir tiempos de espera para hacer diagnósticos a distancia, evitar perdidas de expedientes y llevar un orden en las consultas realizadas, y con códigos de seguridad que garanticen la confidencialidad y la ética médica.

VIII. PRUEBAS REALIZADAS AL SISTEMA DE TELEIMAGENOLOGÍA CLÍNICA

Las pruebas que se realizaron al Sistema fueron:

- Prueba de Unidad.
- Pruebas de Integración.
- Pruebas de Volumen.
- Pruebas de Validación.

a) Pruebas de Unidad.

La primera prueba realizada al sistema fue la de unidad la cual se realiza sobre los procedimientos contenidos dentro de un módulo, cada uno de manera independiente, a fin de localizar posibles errores de codificación y lógica que estén contenidos en el código del sistema. En este momento es cuando se detecta si una llamada a una subrutina esta bien generado, o un procedimiento del Servidor, así como los datos que este ultimo regresa.

En el caso del **Sistema de Teleimagenología Clínica**, los primeros procedimientos que se probaron fueron los de consulta, es decir todos aquellos que mostraban información directamente en la pantalla del usuario; posteriormente se probo con la generación de los reportes impresos, y por último, con aquellos que guardaban el diagnóstico del Médico consultante.

b) Pruebas de Integración.

En esta prueba se evaluó el conjunto de conjunto de procedimientos que conforman un módulo, por ejemplo, la consulta de un expediente de Tomografía. Este módulo esta integrado por varios procedimientos que realizan cada uno una tarea especifica, pero que a la vez interactúan dependiendo del buen funcionamiento de los otros procedimientos.

Para llevar a cabo esta evaluación, se realizó una matriz de pruebas para ensayar el ciclo completo (consulta, almacenamiento y reporte) de cada una de las operaciones relacionadas con las áreas de Teleimagenología Clínica

En esta etapa fue en donde el sistema presentó más errores principalmente en el proceso de los reportes, ya que este fue un proceso solicitado al final y se tuvo que hacer una reestructuración de la información y manera de cómo se manejaban los datos, ya que al probar el ciclo completo surgieron opciones que no se habían contemplado anteriormente.

En esta evaluación se observó como funciona el sistema completo en un ambiente de trabajo. Una vez que todos los módulos estaban listos y probados individualmente se podía pensar que el sistema estaba listo, sin embargo, después de este periodo de pruebas se detectó algunos errores en la presentación de las pantallas, y en el arrastre de datos se vieron que existirán variables de sobra, corrigiendo esta anomalía enseguida.

c) Prueba de Volumen.

Con las pruebas de volumen se asegura el funcionamiento adecuado del sistema en condiciones de gran demanda de recursos en cantidad y frecuencia. Para este tipo de pruebas, tuvimos que trabajar con volúmenes cercanos a los que se piensa tener en el sistema. Lo anterior se realizó en las salas de de Computo de la Facultad de Ingeniería; lo que se hizo fue instalar la aplicación en los Clientes (Máquinas de las Salas), y la base de datos de prueba se colocó en uno de los Servidores (Medusa) de las mismas para poder realizar las pruebas. También se dio de alta la información en cada uno de los módulos que conforman la aplicación, de igual forma se realizaron las consultas, guardaron información y se imprimieron los reportes. En estas pruebas permitieron comprobar el funcionamiento de cada uno de los módulos, y se pudo constatar que el sistema resultó sin complicaciones.

d) Prueba de Validación.

En esta prueba se validó la información correcta de cada uno de los módulos, o sea la que el sistema está solicitando en cada uno de ellos y no la que el usuario cree que acaba de recibir.

Por otra parte en el diseño mismo de las ventanas (interfaz de usuario) de la aplicación se especificó el tipo de datos que deberá aceptar el campo, por ejemplo, caracteres alfanuméricos, numéricos, de fecha, etc., por lo tanto, el sistema mismo no permitirá que a un campo de fecha se le teclee un tipo de dato diferente. La última prueba fue el tiempo de espera y fueron validados dos casos:

1. Conexión LAN, de las salas de cómputo (UNICA) de la Facultad de Ingeniería, en donde el tiempo de espera fue inmediato.
2. La segunda fue con una conexión vía MODEM en donde fue validado a una velocidad de 32 KBPS obteniéndose un tiempo de espera que variaba desde los 20 segundos hasta 50 segundos.

Los objetivos de las pruebas fueron valorar, mejorar la calidad del sistema, siguiendo la idea que la mejor manera de disminuir sensiblemente el número de errores en un programa es encontrarlos y eliminarlos durante el análisis y el diseño. La confiabilidad del sistema implica un conjunto de actividades que incluyen diversas pruebas que garantizan el apego a las especificaciones de requisitos, principios generales de facilidad de manejo del sistema, estándares en el lenguaje de programación, estándares organizacionales del ISSSTE, documentación y expectativas del usuario.

CONCLUSIÓN

La aplicación de un programa como el planteado en el presente trabajo, facilita el que cada una de las dependencias del ISSSTE puedan contar con un sistema que almacena, despliega y lleva el orden de cualquier imagen que se utilice para el diagnóstico clínico, como son Radiografías, Tomografías, Resonancias Magnéticas, Mastografías, etc.; siendo así que se estaría logrando la vinculación integral entre todos los campos de la IMAGENOLOGÍA y sus actividades, además de que se contaría con un sistema que automatiza la consulta de los expedientes de los diversos pacientes de IMAGENOLOGÍA reduciendo así tiempos de espera, evitar pérdida de expedientes, hacer diagnósticos a distancia y llevar un orden en las consultas realizadas, con códigos de seguridad que garantizan la confidencialidad y la ética médica.

En este sistema se aprovechó al máximo toda la tecnología en cuanto a comunicaciones se refiere, utilizando la arquitectura Cliente/Servidor y el protocolo de comunicación TCP/IP; la utilización de esta tecnología es justificada por razones de rendimiento y minimización del tráfico en la red, por consiguiente, contar con un sistema eficiente, para el manejo de la información del paciente de área de IMAGENOLOGÍA CLÍNICA.

Debido a que el objetivo fundamental de este proyecto fue proporcionar un sistema seguro y eficiente para reducir tiempos de espera en las consultas de los expedientes, apoyar las consultas y segundas opiniones médicas, consulta de bancos de imágenes, de una forma sencilla, amigable y de acceso controlado; atendiendo a la calidad del producto final y empleando los métodos, herramientas y técnicas que juzgamos convenientes para la solución del problema por resolver, se concluye que este sistema cumple satisfactoriamente dicho objetivo.

El aprendizaje que se obtuvo, además del relacionado con la metodología empleada, es que la Ingeniería en Computación es una actividad que involucra de manera intensiva a la gente, pues se desarrollo un software que será usado por personas quienes nos establecieron sus necesidades y expectativas del producto final.

Finalmente se recomienda para trabajos futuros y la implementación masiva del programa, que será aplicado con una metodología de prioridad hacia la Imagenología y siempre evaluando el producto, siendo los usuarios finales los que retroalimenten las necesidades y la incorporación de mejoras que se irán presentando con la evolución de la tecnología.

BIBLIOGRAFÍA

- [1] Aguilar Díaz Cruz Sergio, Montoya Cervantes Francisco Javier et all, “**Sistema de Información Deportiva**”, Tesis, Ciudad Universitaria 2002.
- [2] Blanchard B. S. and Fabricky W. J., “**Systems Engineering and Analysis**”, editorial Prentice-Hall, USA 1981.
- [3] Fairley Richard E., “**Ingeniería de Software**”, editorial McGraw-Hill, México 1988, páginas 178, 179.
- [4] Jeffrey L. Whitten, “**Análisis y Diseño de Sistemas de Información**”, editorial McGraw-Hill, Colombia 1996.
- [5] Kendall K. E. and Kendall J. E., “**Análisis y Diseño de Sistemas**”, editorial Prentice-Hall Hispanoamericana, México 1991.
- [6] Martin J., “**Fourth Generation Languages**”, volumen 1, editorial Prentice-Hall, USA 1985.
- [7] McCall J., Richards P. and Walters G., “**Factors in Software Quality**”, editado por Ntis, USA 1997.
- [8] Myers G., “**The Art of software Tests**”, editorial Wiley, USA 1979.
- [9] Pressman Roger S., “**Ingeniería del Software**”, editorial McGraw-Hill, tercera edición, México 1993, páginas 26-32, 140, 141, 217-219, 270, 271.
- [10] Ross Nelson, “**Guía Completa de Visual Basic para Windows**”, editorial McGraw-Hill, México 1995.
- [11] Stevens W., Myers G. and Constantine L., “**Structured Design**”, editado por MIT, USA 1959.
- [12] Swanson E. B., “**The Dimensions of Maintenance**”, editado por IEEE, USA 1976.
- [13] Tanenbaum Andrew S. “**Redes de Computadoras**”, editorial Prentice-Hall, tercera edición, México 1997.
- [14] Tanenbaum Andrew S. “**Sistemas Operativos Modernos**”, editorial Prentice-Hall, primera edición, México 1995.

- [15] Tsai Alice Y. H., “**Sistemas de Base de Datos (administración y uso)**”, editorial Prentice-Hall Hispanoamericana, México 1990.
- [16] Yourdon E. and Constantine L., “**Structured Design**”, editorial Prentice-Hall, USA 1979.
- [17] Yourdon E., “**Techniques of Program Structure and Design**”, editorial Prentice-Hall, USA 1979.
- [18] William R. Vaughn, “**Programación de SQL Server 7.0 con Visual Basic 6.0**”, editorial McGraw-Hill, México 1999.

SITIOS DE INTERNET CONSULTADOS

- [1] <http://radiologia.nezit.com.ar/>
Mediante el desarrollo de las distintas páginas, pretendo, generar un espacio de intercambios con estudiantes y técnicos, transmitir conceptos teóricos, informar bibliografía específica, aspirando a contribuir y enriquecer los aspectos relacionados sobre el tema.
- [2] http://www.medspain.com/ant/n16_oct_nov00/dcoporimagen.htm
Diagnostico Por Imagen. La imagen es un elemento fundamental en el actual quehacer médico, por su papel en el estudio diagnóstico, del paciente. La imagen médica nos ofrece una reproducción de una parte del cuerpo humano que permite al médico acercarse al diagnóstico de la enfermedad, y así poder instaurar un tratamiento.
José Eduardo González Díaz
- [3] <http://colombiamedica.univalle.edu.co/Vol28No1/Radiaciones.html>
La génesis del uso de las radiaciones en la medicina
Roberto Cuenca, Ing.
- [4] <http://www.methodisthealth.com/spanish/radiology/index.htm>
La radiología es la rama de la medicina que utiliza sustancias radioactivas, radiación electromagnética y ondas sonoras para crear imágenes del cuerpo, sus órganos y estructuras con fines de diagnóstico y tratamiento.
- [5] <http://www.issste.gob.mx>
El Instituto de Seguridad y Servicios Sociales de los Trabajadores del Estado.

- [6] <http://www.maestrosdelweb.com/editorial/articulo.asp?linux>
Linux, otra opción en sistemas operativos
- [7] <http://www4.uji.es/~al019803/Tcpip.htm>
Protocolos TCP/IP
Juan Salvador Miravet Bonet
- [8] <http://live.linux.org.ve/NTvsLinux/metachart.html>
NT vs Linux
Extensa documentación que muestra las diferencias entre estos dos Sistemas Operativos.
- [9] <http://www.lawebdelprogramador.com>
Esta página, está diseñada para que tanto programadores aficionados, como programadores profesionales, y todos los que tengan alguna habilidad frente al ordenador.
- [10] <http://www.abox.com/productos.asp?pid=54>
Servidor de red HP NetServer E800
- [11] <http://www.ub.es/comporta/acronimo.htm>
Diccionario de Acrónimos. Una serie de siglas nos invaden cada día a través de los medios de comunicación, y muy especialmente si se trata de informática y de Internet.
- [12] http://www.cica.es/seguridad/DOCUMENTACION/guia_NT.es.html
Guía de seguridad en Windows NT
- [13] <http://math.unl.edu.ar/~martin/articulo.html>
Porqué usa la gente Windows?
Martín A. Marqués
- [14] http://www.geocities.com/elplanetamx/Sistema_Operativo.html
Sistema Operativo
- [15] <http://www.geocities.com/SiliconValley/8195/noscs.html>
Sistemas operativos para redes cliente servidor
- [16] <http://hp.es/>
Pagina HP en español.
- [17] www.comaq.com
Los diez mejores sitios: Computadoras, finanzas, hogar, etc.
- [18] <http://www.consulintel.es/Html/Tutoriales/Articulos/risc.html>

Procesadores RISC, multiproceso y caché
Autor: Jordi Palet

- [19] <http://www.hp.es/cpsd/docs/producto/htm/sere800.htm>
Página de hp
- [20] <http://h18000.www1.hp.com/products/servers/proliantml350/>
Página de hp
- [21] http://www.condusef.gob.mx/informacion_sobre/curp/curp.htm
Clave Única de Registro de Población (CURP)