



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE INGENIERÍA

SISTEMA DE ADMINISTRACIÓN MÉDICA

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A N:

**ESPINOSA CERDA LUIS URBANO
ESPINOSA VERA JORDANA RUTHVE
PEDRAZA SERNA MARÍA DEL PILAR**

DIRECTOR DE TESIS:

M.I JUAN CARLOS ROA BEIZA

MÉXICO, D. F. 2006





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

ÍNDICE

	PAG.
CAPÍTULO I. ENTORNO OPERATIVO.....	1
1.1 GENERALIDADES.....	2
1.2 ELEMENTOS QUE INFLUYEN EN LA ADMINISTRACIÓN DE MÉDICOS ...	5
1.3 ELEMENTOS QUE INFLUYEN EN EL REGISTRO DE PACIENTES E HISTORIAL CLÍNICO.....	8
1.4 ELEMENTOS QUE INFLUYEN EN EL REGISTRO DE CITAS	14
1.5 ORGANIGRAMA DE OPERACIÓN TÍPICO DE UN CONSULTORIO	20
CAPÍTULO II. TEORÍA BÁSICA.....	22
2.1 BASES DE DATOS RELACIONALES Y MODELO CLIENTE-SERVIDOR ..	23
2.2 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE VISUAL BASIC 6	35
2.3 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE MYSQL	43
2.4 SEGURIDAD DEL SISTEMA OPERATIVO XP	50
2.5 REDES Y COMUNICACIONES	61
CAPÍTULO III. PLANTEAMIENTO DEL PROBLEMA Y SOLUCIÓN.....	71
3.1 PROBLEMÁTICA ACTUAL	72
3.2 REQUERIMIENTOS GENERALES Y PARTICULARES	78
3.3 RECOPIACIÓN Y ANÁLISIS DE LA INFORMACIÓN	82
3.4 IDENTIFICACIÓN DEL PROBLEMA	91
3.5 UBICACIÓN DE LAS OPCIONES DE SOLUCIÓN	100
CAPÍTULO IV. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA	113
4.1 APLICACIÓN DE LA METODOLOGÍA ELEGIDA	114
4.1.1 DIAGRAMA DE CONTEXTO.....	121
4.1.2 DIAGRAMA DE FLUJO DE DATOS (DFD) Y DIAGRAMA DE PROCESOS DE DATOS (DPD).....	122
4.1.3 DICCIONARIO DE DATOS	134

4.1.4 DIAGRAMA ENTIDAD-RELACIÓN.....	143
4.1.5 NORMALIZACIÓN	145
4.2 DISEÑO Y GENERACIÓN DEL BACK-END.....	149
4.3 DISEÑO Y GENERACIÓN DEL FRONT-END	173
4.4 INTEGRACIÓN Y PRUEBAS DEL SISTEMA	179
4.5 OBTENCIÓN DE REPORTES Y TIPOS DE MANTENIMIENTO	187
MANUAL TÉCNICO.....	192
MANUAL DE USUARIO.....	204
CONCLUSIONES.....	249
BIBLIOGRAFÍA	252

INTRODUCCIÓN

En el presente trabajo de tesis se realiza un análisis, diseño y desarrollo de un Sistema de Administración Médica (SISAM), el cual tiene como objetivo ofrecer una herramienta para el mejor control de datos de manera electrónica en un consultorio médico.

Se realiza un análisis inicial sobre el entorno en el cual se implementará el sistema, esto es, las generalidades y los elementos que influyen en el trabajo diario dentro de un consultorio médico.

Posteriormente se presenta una investigación sobre las herramientas computacionales con las cuales contamos para el desarrollo del sistema, analizando las ventajas y desventajas de éstas.

En un tercer capítulo se presenta el planteamiento del problema al cual nos enfrentamos, estableciendo la solución más viable a través de una revisión a los requerimientos generales y particulares del usuario, ya que son estos elementos los que nos ayudan a determinar el flujo de información que tendrá el sistema y por lo tanto podremos darle la funcionalidad requerida.

Por último se muestra el desarrollo del sistema que comprende la elección de la metodología empleada, los diagramas de contexto y flujo de la información, así como la presentación final tanto de la base de datos diseñada como la interfaz de usuario. Se realizan las pruebas debidas al sistema para garantizar el buen funcionamiento de éste y se proporcionan los manuales tanto técnico como de usuario para el manejo guiado y confiable del sistema.

CAPÍTULO I

ENTORNO OPERATIVO

1.1 GENERALIDADES

En la actualidad los grandes volúmenes de información que se manejan en todos los sectores demandan de un estricto control de la misma, a fin de mantenerla segura, confiable y disponible en cualquier momento, es por esto que, la información debe ser administrada correctamente para mantener su calidad de información.

La administración y gestión medica, persigue la satisfacción de las necesidades y demandas de salud; las actividades que ofrecen son servicios en condiciones apropiadas (calidad), protegiendo los intereses económicos de la institución y logrando la generación de capital mediante un mecanismo de operación que persigue la utilización equilibrada de los recursos humanos como (médicos, enfermeras, técnicos, administrativos), recursos materiales (instalaciones, equipos), recursos financieros, y recursos tecnológicos (sistemas y procedimientos).

Dado el mundo globalizado en que vivimos, surge para las clínicas la necesidad de implementar sistemas de administración médica, que permitan estar a la vanguardia tecnológica, automatizando sus actividades y optimizando sus recursos. Al mismo tiempo convertirse en una empresa que brinde un alto nivel de calidad en su servicio.

Un sistema es un conjunto de componentes que interaccionan de manera lógica entre si para lograr un objetivo en común.

La finalidad de tener un sistema en una empresa, es que forme parte del plan global de la misma, permitiendo alcanzar los objetivos del control y gestión de la información.

La misión de los sistemas de administración médica, es proveer un nivel excepcional de cuidado al paciente y moldear el sistema médico del mañana a través de innovaciones clínicas, investigación biomédica y servicios de salud.

La visión de los sistemas de administración médica, redefinirá los tradicionales modelos de asistencia médica convirtiéndose en un sistema verdaderamente integrado y auto-regulado, utilizando prácticas de medicina basadas en evidencia para producir magníficos resultados clínicos.

En el contexto competitivo actual, el principal objetivo de un sistema de administración médica, es proporcionar las herramientas necesarias para llevar un control óptimo de su operación, desde la programación de citas médicas, registro de pacientes, creación de su expediente, etc.

Esta solución está enfocada a los profesionistas médicos, administradores de clínicas u hospitales, que requieran contar con una herramienta que les proporcione información oportuna sobre la disponibilidad de médicos, horarios y control absoluto del historial de cada uno de sus pacientes.

Un sistema de gestión y administración médica puede contener los siguientes módulos:

- Módulo de administración de usuarios.
- Módulo de registro y control de citas médicas.
- Módulo de registro de médicos
- Módulo de registro de pacientes y expedientes clínicos.

Los beneficios que provee este tipo de sistema son:

- Permite una gestión administrativa, ágil y eficiente.
- Reducción de costos de operación.

- Contar con el expediente clínico actualizado de los pacientes.
- Tener confidencialidad en los datos
- Respalos de la información.

Tecnología y Especificaciones Técnicas

Una de las primeras observaciones que debe realizarse para la utilización de los recursos, es verificar y evaluar tanto el software como el hardware con los que se cuentan. De acuerdo a las necesidades económicas y tecnológicas se requieren evaluar las diversas herramientas para el desarrollo de un sistema como son:

- Plataformas para el desarrollo de aplicaciones
- Manejadores de Base de Datos
- Sistemas Operativos
- Requerimientos de Hardware

Adaptaciones

Los sistemas deben estar diseñados para llevar acabo actualizaciones que permitan ir ajustando las necesidades operacionales de la clínica que surgen día a día, tanto tecnológicamente como en el almacenamiento, seguridad y control de la información.

- Adaptaciones a la medida para cumplir al 100% con las necesidades de su operación.
- Desarrollo para equipos móviles.
- Adaptaciones para acceso vía web.
- Adaptaciones de software y hardware actual.
- Adaptaciones de seguridad informática.
- Adaptaciones de redes híbridas.

1.2 ELEMENTOS QUE INFLUYEN EN LA ADMINISTRACIÓN DE MÉDICOS

Los elementos administrativos con los que el médico tiene relación son los siguientes:

- Ideas
- Tareas
- Personas

Las funciones generales que todo elemento administrativo debe cumplir son: el análisis de problemas, la toma de decisiones y el establecimiento de comunicación en todos los niveles.

Los pasos del proceso administrativo se describen a continuación:

- Planeación. Es alcanzar un objetivo y cumplirlo, entre las características que tiene son:
 - Formular Políticas. Esto se refiere a tomar decisiones permanentes sobre asuntos importantes.
 - Fijar procedimientos. Estandarización de métodos.
 - Presupuesto. Asignación recursos.
 - Programar. Establecer prioridades.
 - Desarrollar estrategias para alcanzar metas.
 - Fijar objetivos. Determinar resultados esperados.
 - Pronosticar. Prever situaciones en curso.

- Organización. Consiste en analizar la forma de distribuir el trabajo. Las características son:

- Fijar requerimientos para cada puesto. Conocer las cualidades del personal.
 - Crear descripciones de cada puesto. Fijar autoridad y responsabilidad.
 - Delinear las relaciones. Se ve que líneas de enlace son las mejores para coordinar.
 - Establecer una estructura de la organización.
- Integración. Selección del personal adecuado.
- Seleccionar. Reclutar personal calificado.
 - Orientar. Familiariza al personal con el trabajo.
 - Adiestrar. Mejorar el desempeño con instrucción y práctica.
 - Desarrollar. Mejorar conocimientos, aptitudes y habilidades.
- Dirección. Se refiere a las actividades que se tienen que realizar y a las decisiones que se tienen que tomar para alcanzar objetivos.
- Delegar. Asigna responsabilidades.
 - Motivar. Inspira al personal a tomar la acción deseada.
 - Coordinar. Relaciona esfuerzos eficazmente.
 - Superar diferencias. Soluciona conflicto y alienta iniciativa.
 - Administrar el cambio. Estimula creatividad e innovación.
- Control.
- Fijar sistemas de información. Datos críticos y cuando se requieren.
 - Planear estándares de actuación. Situaciones que se presentarán.
 - Mide resultados. Grado de desviación de las metas.
 - Toma medidas correctivas. Ajustar planes y asesorar.

Teniendo en mente lo anterior, podemos avanzar en el conocimiento del proceso administrativo de una clínica o consultorio médico.

El médico es el actor de mayor relevancia en el proceso de atención médica, ya que éste brinda el conocimiento y las soluciones para el tratamiento, detección y control de enfermedades del paciente.

Un consultorio médico puede estar formado por varios médicos, los cuales pueden tener diversas especialidades. Cada médico administra el expediente de un determinado grupo de pacientes de acuerdo a la especialidad del mismo.

Para que un médico pueda brindar una consulta clínica, es necesario que se establezca el horario de atención de pacientes, ya que sin éste, no se podrá generar cita alguna. El horario es el primer elemento que influye en la administración de médicos.

El segundo elemento que influye en la administración, es la especialidad del médico, ya que determinará el tipo de pacientes al que puede atender.

1.3 ELEMENTOS QUE INFLUYEN EN EL REGISTRO DE PACIENTES E HISTORIAL CLÍNICO

El registro de pacientes se entenderá como la toma de datos personales de un individuo que requiere servicios médicos para posteriormente programar una cita, la cual se convertirá en una consulta cuando la persona sea atendida por el médico.

El objetivo del registro es formar el expediente clínico de cada uno de los pacientes, a fin de conocer un poco más de cada individuo y sus padecimientos, así como también llevar un control sobre las consultas que va teniendo. El expediente clínico está formado por formatos preestablecidos que se pueden consultar y agregar en cada consulta.

El expediente clínico se forma de los siguientes elementos:

- **Historia Clínica:** Incluye los datos generales del paciente además de un cuestionario con los antecedentes de enfermedades del mismo y sus familiares así como los padecimientos que se observen al momento de llenar este documento, ya que esto quedará como base para que el doctor atienda al paciente en consultas futuras. Este formato es llenado por el médico o la enfermera y está formado por 2 hojas. Ver figura 1.3.1
- **Órdenes y Notas Médicas:** En este formato se registran los padecimientos que se encuentran en la consulta. Se agrega un formato de nota médica cada vez que el paciente acuda a una consulta. Este formato es llenado por el médico. Ver figura 1.3.2
- **Hoja de Referencia y Contrarreferencia:** Se usa cuando el paciente requiere ser trasladado a alguna otra institución médica y viceversa; el expediente clínico incluye un formato de hoja de referencia y contrarreferencia que será utilizado en caso de ser necesario. Este formato es llenado por el médico. Ver figura 1.3.3

- **Receta:** Aunque no se trata de un formato que se incluya en el expediente, se hará mención de ella ya que es un documento que el médico llena para el paciente en cada consulta, aquí el doctor indica al enfermo cuáles son los medicamentos que deberá de utilizar, así como el tratamiento a seguir. Este formato es llenado por el médico. Ver figura 1.3.4
- **Análisis y Estudios de Gabinete:** Son los resultados y la interpretación de las pruebas de laboratorio que haya solicitado el médico al paciente en caso de que el doctor lo haya considerado necesario para tratar algún padecimiento. Los formatos son variados ya que dependerá del tipo de estudio y del laboratorio que los haya realizado. Sólo se anexan al expediente dínico en algunos casos con observaciones por parte del médico.

Formatos del expediente clínico:

HISTORIA CLINICA	
INTERROGATORIO DIRECTO <input type="checkbox"/>	INTERROGATORIO INDIRECTO <input type="checkbox"/>
<i>FICHA DE IDENTIFICACIÓN</i> FECHA: _____	
NOMBRE: _____ No. EXP: _____	
SEXO: _____ EDAD: _____ EDO. CIVIL: _____	
OCUPACIÓN: _____ LUGAR DE NACIMIENTO: _____	
RESIDENCIA HABITUAL: _____	
DOMICILIO: _____	
RELIGIÓN: _____	
ANTECEDENTES HEREDOFAMILIARES: _____	

ANTECEDENTES PERSONALES NO PATOLÓGICOS: _____	

ANTECEDENTES PERSONALES PATOLÓGICOS: _____	

ANTECEDENTES GINECOOBSTETRICOS: _____	

ANTECEDENTES TERAPEUTICOS: _____	

PADECIMIENTO ACTUAL: _____	

REVISIÓN POR APARATOS Y SISTEMAS: _____	

Figura 1.3.1 Formato de historia clínica hoja1.

En el registro de pacientes interviene el personal administrativo en este caso la secretaria y/o la enfermera, que es la encargada de tomar los datos del paciente y verificar si éste acude por vez primera o ya ha recibido los servicios de la clínica en otras ocasiones, en el caso de las personas que acuden por primera ocasión forma el expediente con los formatos antes mencionados y se los hace llegar al médico; para los pacientes que ya han recibido atención médica al menos una vez se revisa y agrega en caso de ser necesario al expediente clínico los formatos faltantes para la consulta (órdenes y nota médica para cada consulta y hoja de referencia y contrarreferencia cuando sea requerida), al igual que en el caso anterior se hace llegar al doctor el expediente del paciente.

El registro de un paciente puede ser por teléfono o en persona, en ambos casos el registro se hace con la secretaria y/o enfermera.

En el siguiente diagrama se muestra la interacción de estos elementos (paciente, secretaria y médico) en el proceso del registro de un paciente y la generación de su expediente clínico.

Se consideran dos posibilidades, la primera que el paciente acude por primera vez y la segunda que es un paciente con expediente clínico. El punto a validar es que exista una cita, en el primer caso se considera como un elemento de condición para poder comenzar a formar el expediente clínico ya que al no contar con una cita no existirá una consulta y de acuerdo a lo que se ha descrito anteriormente es en la consulta donde se llena los documentos correspondientes al expediente clínico; en el segundo caso no se muestra a las citas como una condición ya que al no ser la primer cita del paciente se conoce el procedimiento, solo se verifica la misma.

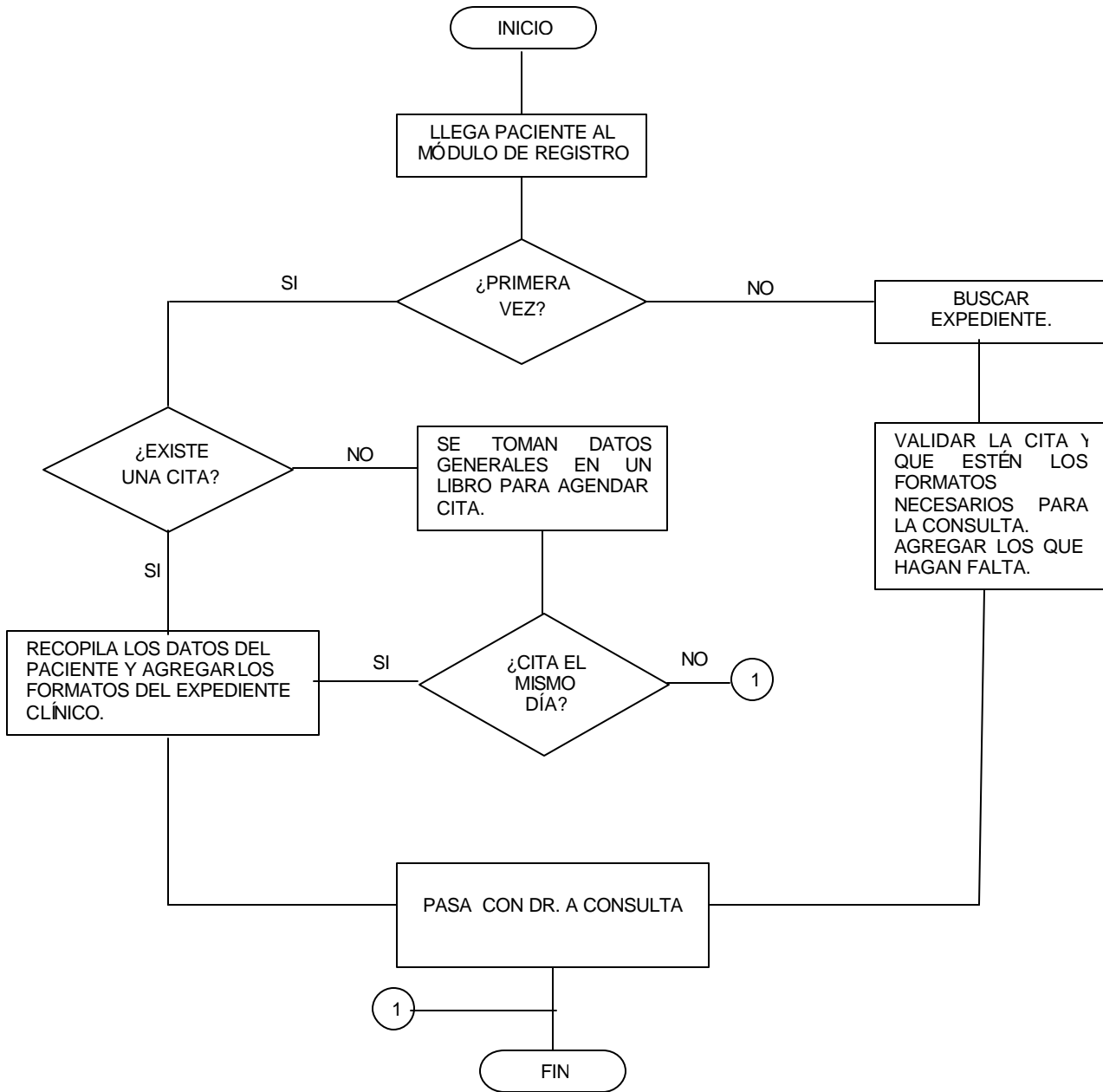


Figura 1.3.5 Registro de un paciente e historial clínico.

1.4 ELEMENTOS QUE INFLUYEN EN EL REGISTRO DE CITAS

Una cita médica consiste básicamente en reservar determinado tiempo de un médico para hacer a éste preguntas acerca de ciertas inquietudes médicas o padecimientos clínicos que pudiéramos tener en un momento de nuestra vida.

Actualmente, existen dos medios por los cuales una persona puede concertar una cita con un médico, el primero y más común es por vía telefónica, el segundo es la vía presencial.

La concertación de una cita médica a través del teléfono, es hoy en día, el medio más usado por las personas, ya que el tiempo invertido para este proceso es muy corto.

La generación de una cita de manera presencial, es la segunda vía que solemos emplear para tratarnos con un médico. El hacer uso de este medio implica invertir tiempo en el traslado, tiempo de atención por el personal de recepción que atiende en un consultorio médico, y un costo extra por traslado. Este tipo de concertación de cita es empleada por personas que tienen la necesidad de ser atendidos en ese momento.

En una cita médica intervienen tres actores, el primero y más importante es la persona que requiere del servicio, el segundo es el personal de recepción que está conformado principalmente por una secretaria o recepcionista quien genera la cita entre el paciente y el médico, finalmente, el tercer actor involucrado en el proceso es el médico tratante.

La figura 1.4.1 muestra gráficamente las entidades que intervienen en el proceso de una cita médica.

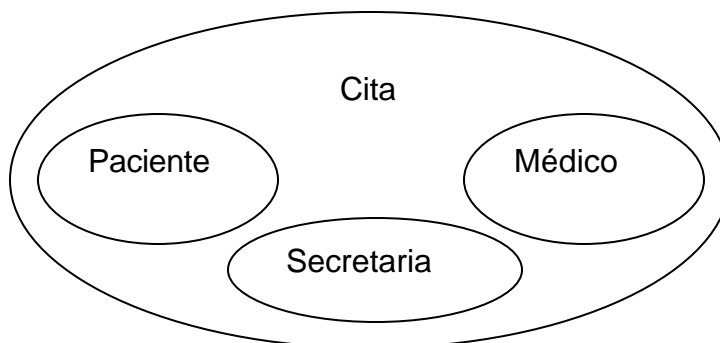


Figura 1.4.1 Actores que intervienen en una cita médica.

Las principales funciones que realiza el personal de recepción son las siguientes:

- Brindar informes al(los) paciente(s) sobre la(s) especialidad(es) que se atienden en el(los) consultorio(s).
- Especificar el(los) horario(s) de atención del(los) médico(s).
- Registro de datos generales del(los) paciente(s).
- Registro y administración de las citas del(los) médico(s).

Las funciones que realiza el médico tratante son las siguientes:

- Establece horario(s) de atención.
- Revisa la agenda médica del día.
- Genera el expediente del paciente.
- Realiza interrogatorio de rutina.
- Revisa clínicamente al paciente.
- Informa padecimiento al paciente.
- Brinda tratamiento al paciente.
- Genera receta médica.
- Transfiere al paciente en caso de que se requiera.
- Solicita estudios clínicos y de gabinete.

Proceso de una Cita por Teléfono

En la tabla 1.4.1 se expone el proceso de una cita vía telefónica consiste en lo siguiente:

No.	Actor	Actividad
1	Paciente	Llama por teléfono e informa al encargado de recepción el tipo de servicio que requiere.
2	Encargado de Recepción	Recibe la llamada del paciente, lo interroga acerca del servicio que solicita.
3	Encargado de Recepción	Pregunta al paciente si es la primera vez que solicita este servicio.
4	Encargado de Recepción	Si es la primera vez que el paciente solicita este servicio, registra sus datos personales. Continúa con el paso 6.
5	Encargado de Recepción	Si se trata de una consulta subsecuente, genera una cita. Continúa con el paso 9.
6	Encargado de Recepción	Brinda información sobre los médicos y horarios disponibles.
7	Paciente	Indica el nombre del médico, horario y fecha en que solicita la cita.
8	Encargado de Recepción	Registra cita del paciente en la fecha y hora convenida.
9	Paciente	Termina llamada.

Tabla 1.4.1 Procesos de una cita por teléfono.

Proceso de una Cita Presencial

La tabla 1.4.2 explica el proceso de una cita de manera presencial.

No.	Actor	Actividad
1	Paciente	Asiste a las instalaciones del consultorio médico.
2	Paciente	Solicita al encargado de recepción informes sobre el servicio deseado.
3	Encargado de Recepción	Brinda informes al paciente sobre los servicios que se ofrecen.
4	Paciente	Solicita referencias sobre el personal médico y horario de atención del mismo.
5	Encargado de Recepción	Proporciona información del personal médico y horario.
6	Encargado de Recepción	Pregunta al paciente si es la primera vez que solicita este servicio. Si es la primera vez continúa con el paso 8.
7	Encargado de Recepción	Si se trata de una consulta subsecuente continúa con el paso 9.
8	Encargado de Recepción	Brinda información sobre el(los) médico(s) y horario(s) disponible(s).
9	Paciente	Indica el nombre del médico, horario y fecha en que solicita la cita.
10	Encargado de Recepción	Registra cita del paciente en la fecha y hora convenida.
11	Paciente	Termina la solicitud de cita.

Tabla 1.4.2 Proceso de una cita presencial.

Proceso de una consulta

Una vez establecida la cita, el paso siguiente es que el paciente acuda a entrevistarse con el médico en el lugar, día y hora establecidos. Es aquí, donde la cita se convierte en una consulta.

La tabla 1.4.3 explica el proceso de una consulta médica:

No.	Actor	Actividad
1	Paciente	Acude a la cita en el lugar, fecha y hora establecidos.
2	Paciente	Se anuncia con el encargado de recepción y espera ser atendido.
3	Encargado de Recepción	Informa al médico de la presencia del paciente.
4	Enfermera	Pesa, mide y toma signos vitales del paciente.
5	Enfermera	Pregunta al paciente si es primera vez que asiste.
6	Enfermera	Si es primera vez, elabora expediente y anexa datos referentes al peso, talla, signos vitales, etc.
7	Paciente	Pasa al consultorio para ser atendido.
8	Médico	Si es primera vez, interroga al paciente y elabora hoja de historia clínica. Continúa con el paso 10.
9	Médico	Si no es primera vez continúa con el paso 11.
10	Médico	Interroga al paciente para determinar antecedentes.
11	Médico	Interroga y examina al paciente para conocer padecimiento.
12	Médico	Realiza exploración física completa al paciente.
13	Médico	Informa al paciente sobre su padecimiento, da instrucciones para su tratamiento, elabora nota y receta médica.
14	Médico	Si se requiere transferir al paciente se genera hoja de referencia.

No.	Actor	Actividad
15	Médico	Si se requiere de estudios de laboratorio y/o gabinete se anexa en la receta el estudio a realizar.
16	Médico	Si se requiere de otra consulta se informa al paciente para que solicite otra cita.
17	Paciente	Recibe instrucciones y documentos como: <ul style="list-style-type: none"> ➤ Receta médica ➤ Hoja de referencia ➤ Hoja de estudios de laboratorio y/o gabinete
18	Paciente	Procede a efectuar el pago de la consulta.
19	Encargado de Recepción	Recibe pago por consulta médica y registra movimiento.
20	Paciente	Si es necesaria otra cita, la tramita ese mismo día, o bien, la genera por vía telefónica.
21	Paciente	Si no es necesario realizar estudios de laboratorio y/o gabinete y no es una transferencia, se dirige a farmacia, o bien, a su hogar.

Tabla 1.4.3 Proceso de consulta médica

1.5 ORGANIGRAMA DE OPERACIÓN TÍPICO DE UN CONSULTORIO

El objetivo del diseño organizacional es mostrar la forma en que se establecen las relaciones de trabajo en el consultorio o clínica, lo cual es útil para determinar la interrelación al interior de la misma. Lo cual se muestra a través de un diagrama de bloques en la figura 1.5.1

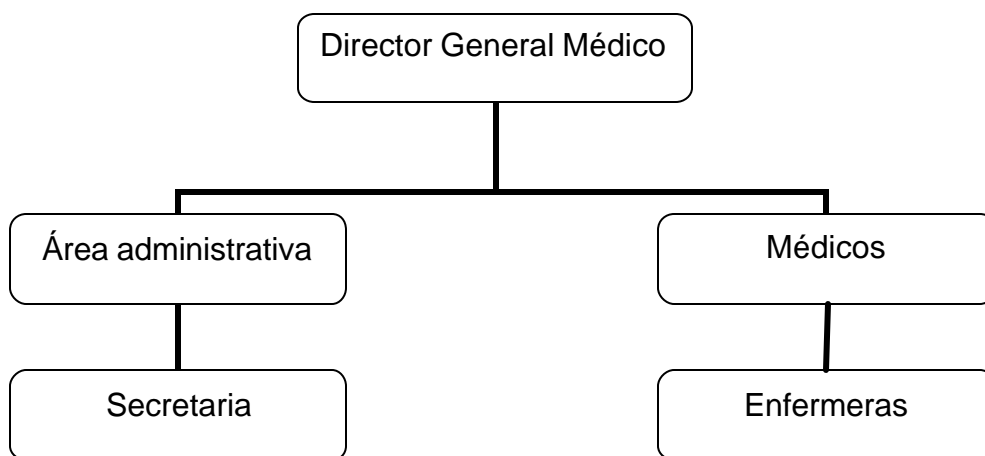


Figura 1. 5.1 Organigrama de operación típico de un consultorio.

El Director General Médico es la cabeza del consultorio, es quien se encarga de gestionar la atención del área de salud, la calidad de atención que se brinda a los pacientes dentro de éste, así como la organización del personal a su cargo. De él depende el funcionamiento eficaz del consultorio.

Otra de las entidades que forma parte del organigrama son los médicos quienes tienen como responsabilidad la atención eficiente y controlada de los pacientes, elaboran las notas médicas del día de acuerdo a la exploración física y al diagnóstico encontrado al paciente. Una vez que atienden al paciente realizan sus anotaciones en los formatos mostrados en el apartado 3 de este capítulo, una vez

determinado el diagnóstico se elabora la receta médica y si es necesaria la atención del paciente en algún otro centro se elabora una hoja de referencia para remitirlo a alguna otra institución.

Por otra parte los médicos también elaboran notas sobre los resultados de análisis de laboratorio y estudios de gabinete solicitados al paciente.

La relación que los médicos mantienen con la parte administrativa es para poder llevar el control de las citas y los expedientes de los pacientes.

El área administrativa es responsable de gestionar y revisar las actividades realizadas por el personal a su cargo, en este caso la secretaria que es la encargada de los trámites relacionados con la cobranza de las consultas. También se encarga del registro de todas las citas.

Por otra parte se encuentran las enfermeras que se encargan de ayudar a los médicos en el registro de los datos personales del paciente como parte del expediente clínico del mismo, apoyan al médico en el interrogatorio previo a la primera consulta con la finalidad de llenar la historia clínica donde anotan todos los antecedentes patológicos y no patológicos. Todo esto se recaba de manera manual en los formatos mostrados en el apartado 3 de este capítulo. Dentro de sus funciones también se pueden mencionar las de tomar nota de los signos vitales del paciente al momento de la consulta.

CAPÍTULO II

TEORÍA BÁSICA

2.1 BASES DE DATOS RELACIONALES Y MODELO CLIENTE-SERVIDOR

Las bases de datos más difundidas en la actualidad son las denominadas Bases de Datos Relacionales. Los principales motivos del éxito de este modelo son fundamentalmente los siguientes:

- Ofrecen sistemas simples y eficaces para representar y manipular los datos.
- Se basan en un modelo, el relacional, el cual cuenta con sólidas bases teóricas.
- El rápido entendimiento por parte de usuarios que no tienen conocimientos profundos sobre sistemas de bases de datos.

El modelo relacional fue propuesto por E. F. Codd en 1970. Gracias a su coherencia y facilidad de uso, el modelo se ha convertido en los años subsecuentes en el más usado para la producción de Manejadores de Bases de Datos (MBD).

Características de una Base de Datos Relacional

- Están basadas en el modelo de relaciones, es decir, una tabla bidimensional constituida por líneas (tuplas) y columnas (atributos).
- Cada tabla tiene un nombre que la identifica unívocamente.
- Las entradas en las tablas tienen un solo valor (son atómicos); no se admiten valores múltiples, por lo tanto la intersección de un renglón con una columna tiene un solo valor y nunca un conjunto de valores.
- Todas las entradas de cualquier columna son de un solo tipo.
- Cada columna posee un nombre único.
- El orden de las columnas no es de importancia para la tabla.

- Cada atributo tiene un dominio, que es una descripción física y lógica de valores permitidos.
- No existen 2 filas en la tabla que sean idénticas.
- La información en las bases de datos son representados como datos explícitos, no existen apuntadores o ligas entre las tablas.
- Las tablas poseen una llave primaria o foránea.
- No redundancia o redundancia controlada.
- Existe consistencia en los datos.
- Contienen reglas de integridad de datos.
- Tener algún tipo de relación entre tablas.

Algebra Relacional

Las operaciones de álgebra relacional manipulan relaciones. Esto significa que estas operaciones usan uno o más relaciones existentes para crear una nueva relación. Esta nueva relación puede entonces usarse como entrada para una nueva operación.

El álgebra relacional consta de nueve operaciones:

- **Unión:** Esta operación permite combinar datos de varias relaciones. No siempre es posible realizar consultas de unión entre varias tablas, para poder realizar esta operación es necesario e imprescindible que las tablas a unir tengan las mismas estructuras, es decir, que sus campos sean iguales.
- **Intersección:** Permite identificar filas que son comunes entre dos relaciones.
- **Diferencia:** Permite identificar filas que están en una relación y no están en la otra.

- **Producto:** Consiste en la realización de un producto cartesiano entre dos tablas dando como resultado todas las posibles combinaciones entre los registros de la primera y los registros de la segunda.

- **Selección:** Consiste en recuperar un conjunto de registros de una tabla o de una relación indicando las condiciones que deben cumplir los registros recuperados, de tal forma que los registros devueltos por la selección han de satisfacer todas las condiciones que se hayan establecido. Esta operación es la que normalmente se conoce como consulta.

- **Proyección:** Es un caso concreto de la operación selección, esta última devuelve todos los campos de aquellos registros que cumplen la condición que se ha establecido. Una proyección es una selección en la que seleccionamos aquellos campos que deseamos recuperar.

- **Reunión:** La reunión se utiliza para recuperar datos a través de varias tablas conectadas unas con otras mediante cláusulas JOIN, en cualquiera de sus tres variantes INNER, LEFT, RIGHT. La operación reunión se puede combinar con las operaciones selección y proyección.

- **División:** La operación división es la contraria a la operación producto .

- **Asignación:** Esta operación algebraica consiste en asignar un valor a uno o varios campos de una tabla.

Cálculo Relacional

El cálculo relacional usa un enfoque completamente diferente al álgebra relacional, no obstante, los dos lenguajes son lógicamente equivalentes. Esto significa que cualquier consulta que pueda resolverse en un lenguaje también puede resolverse en el otro. Las soluciones para toda consulta en este tipo de cálculo se determinan por:

- Una lista de resultados
- Una sentencia de cualificación

Una lista de resultados son aquellos registros que cumplen las condiciones que deseamos.

La sentencia de cualificación contiene las condiciones que deseamos que cumplan los registros de la lista de resultados.

La diferencia entre el cálculo y el álgebra radica en que el cálculo realiza la operación en un único paso, sin necesidad de tener que obtener tablas intermedias, el álgebra realiza las operaciones paso a paso.

Normalmente el cálculo relacional se apoya en algún lenguaje de interrogación de bases de datos como puede ser el SQL .

Dentro del cálculo relacional existen tres operaciones básicas que permiten manipular los datos de una tabla. A continuación se mencionan cada una de estas operaciones.

Operación JOIN (Unir)

Las consultas multitabla o JOINS, también denominadas combinaciones o composiciones, permiten recuperar datos de dos tablas o más, según las relaciones lógicas entre ellas.

Una condición de combinación (o composición) define la forma en que dos tablas se relacionan en una consulta al:

- Especificar la columna de cada tabla que debe de usarse para la combinación. Una condición de combinación típica especifica una clave externa de una tabla y su clave asociada en otra tabla.
- Especificar un operador lógico ($=<>$, etc.) para usarlo en los valores de comparación de las columnas.

Operación INSERT (Insertar)

La operación INSERT consiste en agregar tuplas o renglones en una tabla. Cada sentencia INSERT añade un único registro a la tabla.

Operación DELETE (Eliminar)

La operación DELETE consiste en eliminar una o varias filas o renglones de una tabla.

Las Interrelaciones

Las interrelaciones son las relaciones que existen entre varias tablas del sistema. Existen tres formas de interrelaciones dependiendo de la cardinalidad con la que se combinan los elementos de ambas tablas.

- **Interrelaciones Uno a Uno (1:1).** Una interrelación es de uno a uno entre a tabla A y la tabla B cuando a cada elemento de la tabla A se le asigna un único elemento de la tabla B y para cada elemento de la tabla B contiene un único elemento en la tabla A.

- **Interrelaciones Uno a Varios (1:n).** Una interrelación es de uno a varios entre las tablas A y B cuando un elemento de la tabla A posee varios elementos relacionados en la tabla B y cuando un elemento de la tabla B posee un único elemento relacionado en la tabla A.

- **Interrelaciones Varios a Varios (n:n).** Una interrelación es de varios a varios entre las tablas A y B cuando un elemento de la tabla A posee varios elementos relacionados en la tabla B y cuando un elemento de la tabla B posee varios elementos relacionados en la tabla A.

Normalización

Antes de proceder a la normalización de la tabla lo primero que debemos de definir es una clave, esta clave deberá contener un valor único para cada registro (no podrán existir dos valores iguales en toda la tabla) y podrá estar formado por un único campo o por un grupo de campos.

A continuación se lista cada una de las formas normales:

- **Primera Forma Normal (1NF).** Se dice que una tabla se encuentra en primera forma normal (1NF) si y solo si cada uno de los campos contiene un único valor para un registro determinado.

- **Segunda Forma Normal (2NF).** La segunda forma normal compara todos y cada uno de los campos de la tabla con la clave definida. Si todos los

campos dependen directamente de la clave se dice que la tabla está en segunda forma normal (2NF).

- **Tercera Forma Normal (3NF).** Se dice que una tabla está en tercera forma normal si y solo si los campos de la tabla dependen únicamente de la clave, dicho en otras palabras los campos de las tablas no dependen unos de otros.

- **Forma Normal de Boyce-Codd (FNBC)**

Una tabla está en FNBC sí y sólo si las únicas dependencias funcionales elementales son aquellas en las que la clave primaria determinan un atributo.

Es una etapa en que se deben agrupar los datos por afinidad, formando tablas las cuales se relacionan entre si mediante campos comunes.

- **Cuarta Forma Normal (4NF).**

Está en forma normal de Boyce-Codd y se eliminan las dependencias multivaluadas y se generan todas las relaciones externas con otras tablas u otras bases de datos.

El Modelo Cliente-Servidor

La arquitectura cliente-servidor llamado modelo cliente-servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia y permita simplificarlas.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes. Los clientes interactúan con el usuario, usualmente en forma gráfica. Frecuentemente se comunican con procesos auxiliares que se encargan de

establecer conexión con el servidor, enviar el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad.

Los servidores proporcionan un servicio al cliente y devuelven los resultados. En algunos casos existen procesos auxiliares que se encargan de recibir las solicitudes del cliente, verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente. Además deben manejar los interbloques, la recuperación ante fallas, y otros aspectos afines.

Para que los clientes y los servidores puedan comunicarse se requiere una infraestructura de comunicaciones, la cual proporciona los mecanismos básicos de direccionamiento y transporte.

La figura 2.1.1 muestra gráficamente los principales componentes del modelo cliente-servidor.

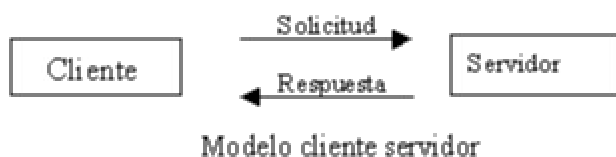


Figura 2.1.1 Diagrama del modelo cliente-servidor.

Características del Modelo Cliente-Servidor

- El Cliente y el Servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.
- Las funciones de Cliente y Servidor pueden estar en plataformas separadas, o en la misma plataforma.
- Un servidor da servicio a múltiples clientes en forma concurrente.
- Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean

por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.

Modelo Cliente-Servidor de una Capa

Las aplicaciones cliente-servidor de una capa agrupan la lógica de presentación (interfaz de usuario), la lógica de aplicación y la fuente de datos (base de datos) en una sola máquina, la cual actúa simultáneamente como cliente y servidor (ver figura 2.1.2).



Figura 2.1.2 Representación del Modelo Cliente-Servidor de una Capa.

Modelo Cliente-Servidor de dos Capas

Las aplicaciones cliente-servidor de dos capas, agrupan la lógica de presentación y la lógica de aplicación en la máquina cliente y acceden a fuentes de datos compartidos a través de una conexión de red que se encuentran en el servidor de datos (ver figura 2.1.3).

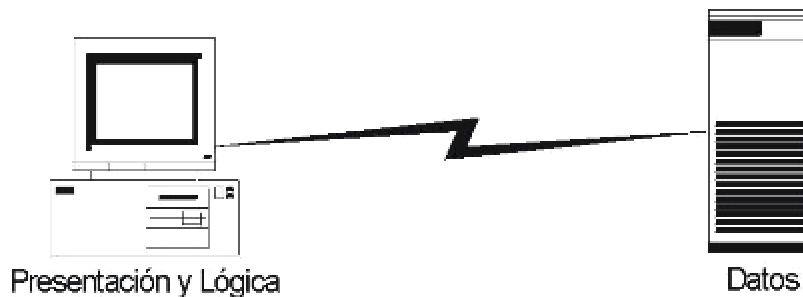


Figura 2.1.3 Representación del Modelo Cliente-Servidor de dos Capas.

Modelo Cliente-Servidor de tres Capas

En las aplicaciones cliente-servidor de tres capas, la presentación, la lógica de aplicación y la fuente de datos están conceptualmente separados (ver figura 2.1.4).

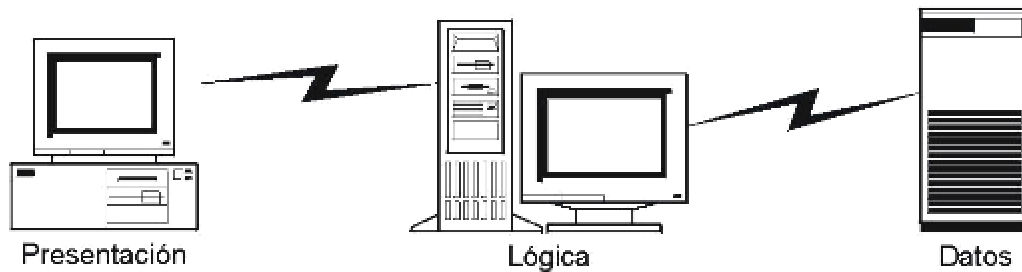


Figura 2.1.4 Representación del Modelo Cliente-Servidor de tres Capas.

Modelo Cliente-Servidor de “n” Capas

En las aplicaciones cliente-servidor de “n” capas, la presentación, la lógica de aplicación y la fuente de datos, además de estar conceptualmente separados, se pueden subdividir (ver figura 2.1.5).

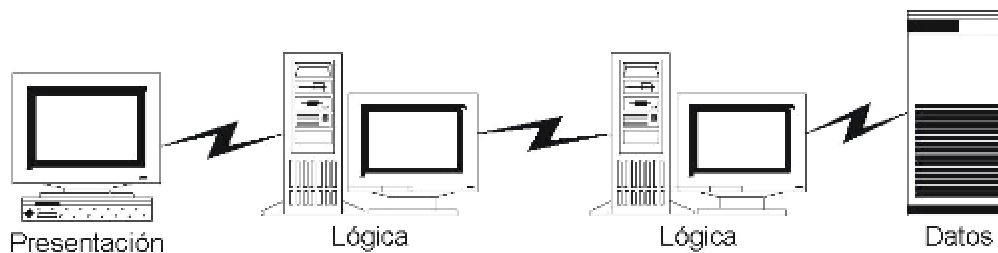


Figura 2.1.5 Representación del Modelo Cliente-Servidor de “n” Capas.

Ventajas del Modelo Cliente-Servidor

- El cliente no necesita tanta potencia de procesamiento, parte del proceso se reparte con los servidores.
- Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre para otra conexión.
- Uno de los aspectos que más ha promovido el uso de sistemas Cliente-Servidor, es la existencia de plataformas de hardware cada vez más baratas. Esta constituye a su vez una de las más palpables ventajas de este esquema, la posibilidad de utilizar máquinas considerablemente más baratas que las requeridas por una solución centralizada, basada en sistemas grandes. Además, se pueden utilizar componentes, tanto de hardware como de software, de varios fabricantes, lo cual contribuye a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- El esquema cliente-servidor facilita la integración entre sistemas diferentes y comparte información permitiendo, por ejemplo, que las máquinas ya existentes puedan ser utilizadas haciendo uso de interfaces más amigables al usuario. De esta manera, podemos integrar computadoras personales con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.
- Al favorecer el uso de interfaces gráficas interactivas, los sistemas construidos bajo este esquema tienen mayor interacción más intuitiva con el usuario. El uso de interfaces gráficas para el usuario, el esquema cliente-servidor presenta la ventaja, con respecto a uno centralizado, de que no es siempre necesario transmitir información gráfica por la red pues esta puede residir en el cliente, lo cual permite aprovechar mejor el ancho de banda de la red.

- El mantenimiento y desarrollo de aplicaciones del modelo cliente-servidor es más rápido, ya que se pueden emplear las herramientas existentes (por ejemplo los servidores de SQL o las herramientas de más bajo nivel como los **sockets**¹ o el **RPC**²).
- La estructura inherentemente modular facilita además la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional, favoreciendo así la escalabilidad de las soluciones.
- El esquema cliente-servidor contribuye además, a proporcionar, a los diferentes departamentos de una organización, soluciones locales, pero permitiendo la integración de la información relevante a nivel global.

Desventajas del Modelo Cliente-Servidor

- Se cuenta con muy escasas herramientas para la administración y ajuste del desempeño de los sistemas.
- Es importante que los clientes y los servidores utilicen el mismo mecanismo (socket o RPC), lo cual implica que se deben tener mecanismos generales que existan en diferentes plataformas.
- Requiere de estrategias para el manejo de errores y para mantener la consistencia de los datos.
- La complejidad en la seguridad es alta, es decir, se deben hacer verificaciones en el cliente y en el servidor.
- El desempeño se ve afectado por el exceso de tráfico en la red.
- La distancias entre los clientes y los servidores contribuye en la respuesta de la información.

¹ Interfaz de comunicación que sirve como mecanismo entre dos procesos cualesquiera que pertenezcan a un mismo sistema, o bien, a dos sistemas diferentes.

² Llamada a Procedimiento Remoto.

2.2 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE VISUAL BASIC 6.0

Características

La palabra "Visual" hace referencia al método que se utiliza para crear la **GUI (Interfaz grafica de usuario)** y **Basic "Beginners Ali-Purpose Symbolic Instruction Code"** (**Código de Instrucción con propósitos simbólicos para principiantes**), hace referencia al lenguaje de programación "Basic". Visual Basic es un ambiente gráfico de desarrollo de aplicaciones para el sistema operativo Microsoft Windows.

Las aplicaciones creadas con Visual Basic, están basadas en objetos y son manejadas por eventos. Visual Basic se deriva del lenguaje Básico, el cual es un lenguaje de programación estructurado. Es por tanto un término medio entre la programación tradicional, formada por una sucesión lineal de código estructurado y la programación orientada a objetos, combina ambas tendencias, ya que no podemos decir que pertenezca por completo a uno de esos dos tipos de programación.

Visual Basic, es un conjunto de herramientas completo para construir e integrar rápidamente aplicaciones confiables y también aplicaciones basadas en servicios de Web, proporciona un soporte revolucionario para múltiples lenguajes en un solo ambiente, aprovechando diseñadores compartidos para componentes visuales y no visuales, XML y HTML, así como un depurador unificado.

Las características de acceso a datos, permiten crear bases de datos, aplicaciones cliente, y componentes de servidor escalables para los formatos de las bases de datos mas conocidas, incluidos SQL y otras de ámbito empresarial como Oracle, etc.

Incluye funciones completas para trabajar con bases de datos para maximizar la productividad del desarrollador. También tiene una sofisticada gestión de errores para la frecuente tarea de evitar que las aplicaciones fallen.

Visual Basic por si mismo hace sencilla la creación de grandes programas mediante las técnicas modulares de programación, lo cual significa que se puede dividir un programa en módulos, más sencillos de manejar y por tanto menos sensibles a errores. Los módulos realizan una o varias tareas relacionadas entre sí, teniendo una interfaz bien definida con el resto del programa, por lo cual pueden ser codificados y verificados de forma totalmente independientes.

Visual Basic se enfoca en todo el ciclo de vida de desarrollo, desde planeación y diseño, pasando por el desarrollo y pruebas, hasta una implementación y administración continua.

IDE (Entorno Integrado de Desarrollo)

Un entorno de desarrollo integrado es un programa compuesto por un conjunto de herramientas para el programador, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI.

En la figura 2.2.1 se muestra el IDE de Visual Basic 6.0.

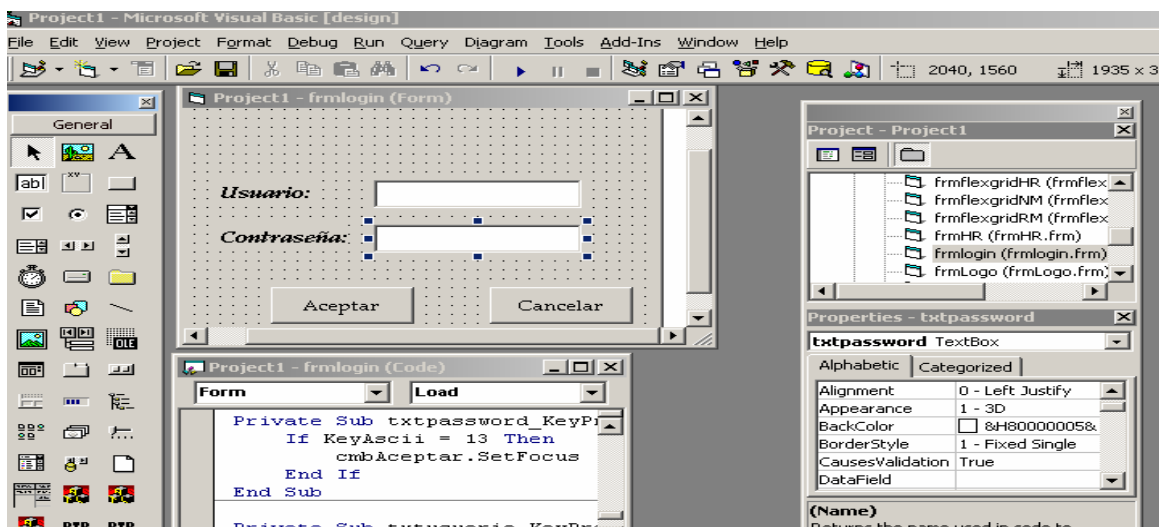


Figura 2.2.1 Entorno Integrado de Desarrollo.

A continuación se describe cada uno de los elementos que componen el IDE de Visual Basic 6.0:

Barra de Menús y Barra de Herramientas Estándar

Presenta los comandos que se usan para trabajar con Visual Basic. Además de la barra de menús como Archivo, Edición, Ver, Ventana y Ayuda (ver figura 2.2.2), y contiene otra barra de herramientas estándar (ver figura 2.2.3), para tener acceso a funciones específicas de programación, como Proyecto, Formato o Depuración y permite un acceso directo a muchas de las operaciones más frecuentes utilizadas durante el desarrollo de aplicaciones.

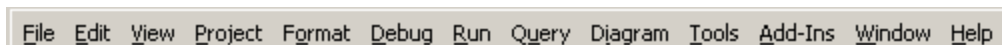


Figura 2.2.2 Barra de Menú.



Figura 2.2.3 Barra de Menú Estándar.

Barra de Herramientas No Estándar

Contiene todos los objetos y controles que se pueden añadir a los formularios para crear aplicaciones, como botones, etiquetas, cajas de texto, etc. (ver figura 2.2.4)

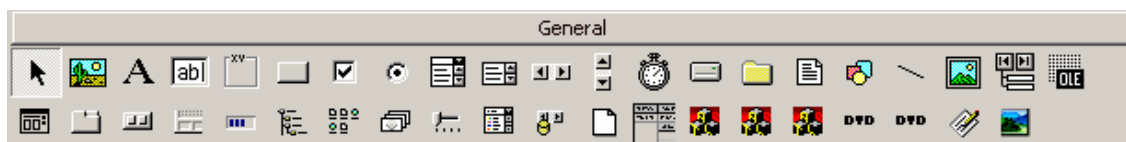


Figura 2.2.4 Barra de Herramientas No Estándar.

Diseñador de Formularios

Funciona como una ventana en la que se puede personalizar el diseño de la interfaz de usuario de una aplicación. (ver figura 2.2.5)

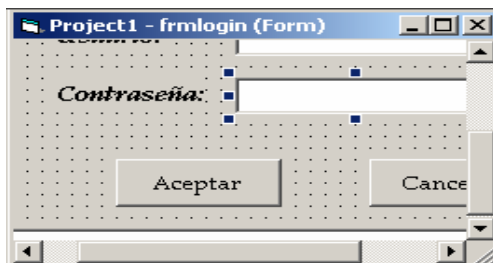


Figura 2.2.5 Diseñador de Formularios.

Explorador de Proyectos

Muestra la lista de los archivos del proyecto actual como formularios, módulos, etc. Un proyecto es una colección de archivos que utiliza para construir una aplicación. (Ver figura 2.2.6)

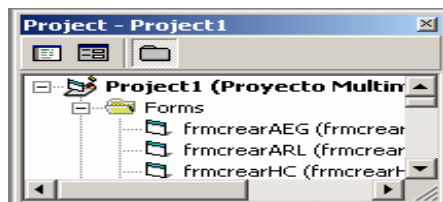


Figura 2.2.6 Explorador de Proyectos.

Ventana de Propiedades

Lista los valores de las propiedades del formulario o control seleccionado que pueden ser modificados durante el diseño del formulario o control (ver figura 2.2.7)

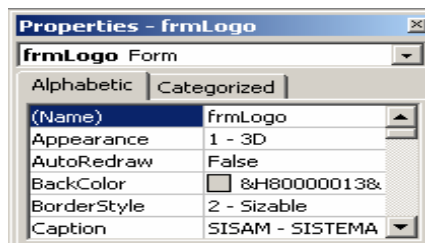


Figura 2.2.7 Ventana de Propiedades.

Editor de Código

Funciona como un editor para escribir el código o sentencias de la aplicación, la ventana o editor de código proporciona de manera automática información relevante a medida que se ingresa código. (Ver figura 2.2.8)

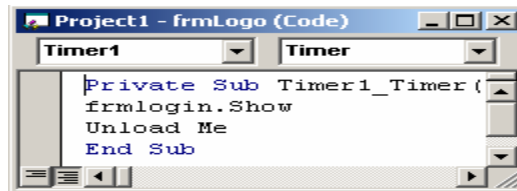


Figura 2.2.8 Editor de Código.

Debugger (Depurador)

Es una potente herramienta que permite la depuración de los programas realizados, su principal característica es que permite ejecutar parcialmente el programa deteniendo la ejecución en el punto deseado y estudiando en cada momento el valor de cada una de las variables, de esta manera facilita el descubrimiento de fuentes de errores.

Obtención de Ayuda del Sistema

Visual Basic proporciona una variedad de recursos para ayudarle a encontrar la información que necesite cuando se encuentre trabajando dentro del entorno de desarrollo.

Ventajas

- Es una herramienta de desarrollo Visual, simple y fácil de aprender.
- Centrado en un motor de formularios poderoso que facilita el rápido desarrollo de aplicaciones gráficas, mediante la gran gama de controles incorporados junto al lenguaje que ahorran tiempo de desarrollo.
- Combina la sucesión lineal de código estructurado, y la programación orientada a objetos (POO),

- Permite trabajar en 2 distintos modos: modo diseño y modo de ejecución
- No requiere de manejo de punteros.
- Las ramas de Visual Basic X ha muerto. Microsoft ha comenzado con su versión .NET, completamente diferente del Visual Basic original. Visual Basic .NET compila hacia el mismo código final que C#. Visual Basic .NET es mucho más potente y alberga muchas más funcionalidades que esta versión.
- Es un lenguaje RAD, centrado en conseguir en el menor tiempo posible los resultados deseados, por eso mismo su mayor uso está en las pequeñas aplicaciones como gestión de empresas, etc. Así como robustas aplicaciones o aplicaciones distribuidas en servicio Web.
- Tiene una implementación de POO (Programación Orientada a Objetos)
- Permite el tratamiento de mensajes de Windows.
- Permite crear controles personalizados fácilmente del mismo modo que el diseño de formularios.
- Posee varias bibliotecas para manejo de bases de datos, destacando ADO.
- Proporciona un soporte revolucionario para múltiples lenguajes, en un solo ambiente, aprovechando diseñadores compartidos para componentes visuales y no visuales, XML y HTML.
- Tiene un depurador unificado con una sofisticada gestión de errores para la frecuente tarea de evitar que las aplicaciones fallen.
- Permite generar aplicaciones cliente-servidor, para los formatos de las bases de datos más conocidas y otras bases de datos de ámbito empresarial.
- Visual Basic permite la creación de grandes programas mediante técnicas modulares, lo cual puede dividir un programa en módulos interactuando entre si, son mas sencillos de manejar y por lo tanto menos sensibles a errores.
- El desarrollador debe plantear primeramente lo que el usuario vera en pantalla.

Desventajas

- Es propietario de Microsoft, por tanto nadie que sea del equipo de desarrollo de esta compañía decide la evolución del lenguaje.
- Sólo existe un compilador & IDE, llamado igual que el lenguaje.
- Sólo genera ejecutables para Windows.
- Los ejecutables generados son relativamente lentos.
- No es adecuado para aplicaciones grandes, multimedia, videojuegos, editores gráficos, etc.
- No permite características de programación avanzada.
- No permite generar librerías dinámicas (DLL).
- Sólo permite el uso de funciones de librerías dinámicas (DLL) stdcall.
- Para que los ejecutables funcionen necesita una DLL llamada MSVBVM60.DLL: **MicroSoft Visual Basic Virtual Machine 60** (versión).
Provee todas las funciones y características implementadas en el lenguaje.
- Algunas funcionalidades están indocumentadas.
- No soporta tratamiento de procesos como parte del lenguaje.
- No tiene manejo de excepciones.
- No incluye operadores a nivel de bits.
- No permite el manejo de memoria dinámica, punteros, arrays, etc. como parte del lenguaje.
- No avisa de ciertos errores o advertencias, como la conversión de tipos inadecuada.
- El tratamiento de mensajes de Windows es básico e indirecto.
- La gran gama de controles incorporados son, sin embargo, muy generales, lo que lleva a tener que reprogramar nuevos controles para una necesidad concreta de la aplicación.
- Los controles personalizados no mejoran la potencia de la API de Windows, y en determinados casos acudir a ésta será el único modo de conseguir el control personalizado deseado.
- La forma de programación que plantea Visual Basic ha ocasionado que muchos programadores de Visual Basic practiquen malas costumbres, entre las más comunes:

- **Arrays (Arreglos)** empezados en 1
- Variables globales, Variables sin declarar, Variables de tipo indefinido y Variables con nombres inexpresivos
- Código innecesario, Código ilegible, Código repetido y Falta de funciones y procedimientos.
- Uso incorrecto de la API de Windows, Uso de goto y etiquetas,
- Uso de controles como simples contenedores de datos.
- Dependencia de los controles a la hora de programar

2.3 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE MySQL.

Características

MySQL es un manejador de bases de datos relacional, se manejan los datos en tablas separadas en lugar de almacenar todos los datos en un solo archivo, lo que añade velocidad y flexibilidad, es una aplicación capaz de ejecutar varios subprocesos a la vez lo cual le permite soportar una gran carga de forma muy eficiente. MySQL fue creado, es distribuido y soportado por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Es considerado un producto de software libre lo cual hace posible que tanto programadores como usuarios contribuyan a través de su experiencia a mejorarlo.

Aunque MySQL es considerada la base de datos de software libre más popular, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario,

Este manejador de bases de datos es, probablemente, el más usado en el mundo del software libre, considerado de muy buen desempeño que le hace consumir pocos recursos de CPU y de memoria RAM por lo que es de gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación, configuración y estabilidad.

MySQL es moderado en la utilización de recursos debido a su tamaño y permite, mediante las opciones de configuración del servidor, ocupar sólo unos cuantos kilobytes de memoria del sistema manteniendo al mismo tiempo un rendimiento adecuado.

Ventajas

MySQL fue originalmente desarrollada para manejar bases de datos muy grandes en forma más rápida que otras soluciones de bases de datos disponibles y ha estado en uso intenso en sistemas de producción de alta demanda por años.

Está constantemente en desarrollo, lo que ofrece un rico y muy actualizado conjunto de funciones y características, así también ofrece interfaces con muchos tipos ODBC para exportar de otras aplicaciones como MS Access, DBase, Fox, etc.

- Es rápido
- Escalable
- Seguro
- Conectividad
- Portable
- Diseño Multicapa

A continuación se describen con más detalle algunas de las ventajas y del manejador de bases de datos MySQL:

- Rapidez
 - El código de MySQL fue escrito teniendo como meta principal un excelente desempeño.
 - Posee un sistema de direccionamiento de memoria rápido basado en **threads** (subprocesos o hilos).
 - Tablas en memoria que son usadas como tablas temporales.
- Escalabilidad³
 - Soporte a grandes bases de datos. Se usa MySQL Server con bases de datos que contienen 50 millones de registros. También existen

³ <http://dev.mysql.com/doc/refman/5.0/es/features.html>

usuarios que usan MySQL Server con 60.000 tablas y cerca de 5.000.000 de registros.

- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2). Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT.

➤ Seguridad⁴

- Posee un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

➤ Conectividad⁵

- Los clientes pueden conectarse con el servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT, 2000, XP, ó 2003). los clientes pueden usar **named pipes** (conexión de software temporal entre dos programas o comandos) para la conexión. En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix.
- En MySQL 5.0, los servidores Windows soportan conexiones con memoria compartida si se inicializan con la opción **shared-memory** (memoria compartida). Los clientes pueden conectar a través de memoria compartida usando la opción `protocol=memory`.
- La interfaz para el conector ODBC (MyODBC) proporciona a MySQL soporte para programas clientes que usen conexiones ODBC (Open Database Connectivity). Por ejemplo, puede usar MS Access para conectar al servidor MySQL. Los clientes pueden ejecutarse en

⁴ ibidem

⁵ ibidem

Windows o Unix. El código fuente de MyODBC está disponible. Todas las funciones para ODBC 2.5 están soportadas, así como muchas otras.

- La interfaz para el conector J MySQL proporciona soporte para clientes Java que usen conexiones JDBC. Estos clientes pueden ejecutarse en Windows o Unix. El código fuente para el conector J está disponible.

➤ Portabilidad⁶

- Escrito en C y en C++
- Probado con un amplio rango de compiladores diferentes
- Funciona en diferentes plataformas.
- Usa GNU Automake, Autoconf, y Libtool para portabilidad.
- APIs (Interfaz de Programación de Aplicaciones) disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- Uso completo de multi-hilo mediante hilos del kernel. Pueden usarse fácilmente múltiple CPU's si están disponibles.
- Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si se desea añadir una interfaz SQL para una base de datos propia.
- Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas.
- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado (linkado) en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

⁶ *ibidem*

➤ Diseño Multicapa

El diseño de MySQL Server es multi capa, con módulos independientes. Algunos de los últimos módulos se listan a continuación con una indicación de lo bien probados que están:

- Replicación. Hay grandes grupos de servidores usando replicación en producción, con buenos resultados. Se trabaja para mejorar características de replicación en MySQL 5.x.
- InnoDB tablas. El motor de almacenamiento transaccional InnoDB es usado en grandes sistemas de producción con alta carga de trabajo.
- BDB tablas. El código Berkeley DB es muy estable, pero todavía se está trabajando para mejorarlo con el interfaz del motor de almacenamiento transaccional BDB en MySQL Server.
- MyODBC 3.51. Usa ODBC SDK 3.51 y es usado en sistemas de producción ámpliamente. Algunas cuestiones surgidas parecen ser cuestión de las aplicaciones que lo usan e independientes del controlador ODBC o la base de datos subyacente.

Desventajas

Como ya se ha comentado MySQL surgió como una necesidad de un grupo de personas sobre un manejador de bases de datos rápido, por lo que sus desarrolladores fueron implementando únicamente lo que precisaban, intentando hacerlo funcionar de forma óptima. Es por ello que, aunque MySQL se incluye en el grupo de sistemas de bases de datos relacionales, carece de algunas de sus principales características y aunque algunas de ellas ya se han incluido en las versiones más recientes no se puede asegurar la estabilidad y buen funcionamiento de estas:

- Subconsultas
- Triggers
- Procedimientos almacenados
- Transacciones
- Integridad referencial
- Índices.

A continuación se presenta una breve descripción de cada una de las características mencionadas como desventajas.

- Subconsultas
 - Tal vez ésta sea una de las características que más hacía falta, aunque gran parte de las veces que se necesitan, es posible reescribirlas de manera que no sean necesarias.
 - Triggers
 - Aunque éstos ya se incluyen en la última versión, no podemos hablar de una gran estabilidad ya que los triggers reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
 - Procedimientos almacenados
 - Los procedimientos almacenados fueron implementos hasta la versión 5, la cual aún presenta algunos problemas de estabilidad.
 - Transacciones
 - A partir de las últimas versiones ya hay soporte para transacciones, aunque no por defecto (se ha de activar un modo especial).
 - Integridad referencial
 - Aunque admite la declaración de claves foráneas en la creación de tablas, internamente no las trata de forma diferente al resto de los campos.
 - Índices
 - MySQL utiliza a menudo índices para realizar operaciones de clasificación rápidamente sin examinar todas las filas.
-
-

- Los índices agilizan la recuperación pero hace más lentas las operaciones de insertar, borrar, así como las actualizaciones de valores en las columnas indexadas(es decir la mayor parte de las operaciones que implican escritura) porque la escritura no afecta sólo a la fila de datos, sino a menudo también a los índices.
- Los índices se actualizan cada vez que se modifica la columna o columnas que utiliza. Por ello no es aconsejable usar como índices columnas en las que serán frecuentes operaciones de escritura (INSERT, UPDATE, DELETE).
- Si tiene muchos índices, los ficheros pueden alcanzar el tamaño máximo de fichero antes que el de datos.

2.4 SEGURIDAD DEL SISTEMA OPERATIVO XP

Windows XP es la versión más confiable de Windows hasta la fecha, el cual se basa en el código central de software utilizado en Windows 2000 y Windows NT Workstation. Este código, conocido como núcleo de Windows NT, o el nuevo motor de Windows, proporciona un nuevo nivel de confiabilidad y estabilidad ya que también contiene nuevas características en el kernel de sistema operativo que hacen que Windows XP sea más eficaz.

Windows XP permite hacer más, en menos tiempo y con menor dificultad. Este nuevo sistema operativo ofrece la seguridad que Windows 9x no puede y también permite a los usuarios ejecutar aplicaciones que Windows 2000 no puede ejecutar. Otro de los objetivos de esta versión es la compatibilidad con otras versiones de Windows. Además Windows XP no permite que ninguna aplicación tire el sistema. Proporciona un mejor monitoreo de manera que el usuario pueda ejecutar mejor las aplicaciones del sistema permitiendo el máximo número de éstas.

El nuevo motor de Windows ofrece las siguientes características:

- Una tecnología de sistema operativo superior, incluidas multitarea, tolerancia a errores y protección de la memoria del sistema, características que permiten evitar y solucionar problemas y garantizar un funcionamiento sin problemas.
- La capacidad de recuperar el trabajo en muchos casos, por ejemplo si se produce un error en el programa antes de poder guardar el equipo.
- Protección de la memoria del sistema para ayudar a evitar que se produzcan errores en software de poca calidad que provocan inestabilidad en el funcionamiento del equipo.
- Además, normalmente no será necesario reiniciar Windows XP al instalar software nuevo, a diferencia de las versiones anteriores de Windows.

Existen dos ediciones de Windows XP: Profesional y Home, de acuerdo a sus nombres la versión profesional cubre las necesidades de los que trabajan en un entorno profesional, proporciona características que no tiene la edición Home; la versión Home cubre las necesidades de aplicaciones ligeras.

Para abordar las cuestiones de seguridad se considerará la edición profesional ya que es la más completa en este sentido.

Windows XP Professional incluye distintas características con las que las empresas pueden proteger archivos, aplicaciones y otros recursos seleccionados. Estas características incluyen listas de control de acceso (ACL), grupos de seguridad y directiva de grupo, y también las herramientas necesarias para que las empresas las configuren y administren. En general se ofrece una infraestructura de control de acceso potente y flexible para las redes empresariales.

El sistema operativo Windows XP también incluye plantillas de seguridad predefinidas que las empresas pueden implementar sin modificaciones o utilizar como base para una configuración de seguridad más personalizada.

Antes de mencionar las funciones de dichas plantillas mencionaremos algunos conceptos para manejar la seguridad del sistema operativo Windows XP:

Dominio

Grupo de equipos que forman parte de una red y comparten una base de datos de directorio común. Un dominio se administra como una unidad con reglas y procedimientos comunes. Cada dominio tiene un nombre único.

Un dominio de Active Directory es un conjunto de equipos definido por el administrador de una red basada en Windows. Estos equipos comparten una base de datos común de directorios, directivas de seguridad y relaciones de seguridad con otros dominios. Un dominio de Active Directory permite el acceso a las

cuentas de usuario y de grupo centralizadas que mantiene el administrador del dominio. Un bosque de Active Directory está compuesto de uno o varios dominios, cada uno de los cuales puede abarcar más de una ubicación física.

Un dominio DNS es cualquier árbol o subárbol del espacio de nombres DNS. Aunque los nombres de los dominios DNS suelen corresponder a los dominios de Active Directory, no deben confundirse con los dominios de Active Directory.

Dominio de recurso

Dominio de Windows NT 4.0 que se utiliza para alojar servicios de archivo, de impresión y otros servicios de aplicación.

Árbol de consola

Panel izquierdo de una Microsoft Management Console (MMC) que muestra los elementos de la consola. De forma predeterminada, es el panel izquierdo de una ventana de consola, pero se puede ocultar. Las características de una consola están determinadas por los elementos del árbol de la consola y su organización jerárquica.

Árbol de dominios

En DNS, la estructura jerárquica de árbol invertida que se utiliza para indexar los nombres de dominio. El propósito y el concepto de los árboles de dominios son similares a los de los árboles de directorios utilizados por los sistemas de organización de archivos de los equipos para el almacenamiento en disco.

Por ejemplo, cuando varios archivos se almacenan en disco, los directorios se pueden utilizar para organizar los archivos en grupos lógicos. Cuando un árbol de dominios tiene una o varias ramas, cada una puede organizar los nombres de dominio utilizados en el espacio de nombres en grupos lógicos.

En Active Directory, es una estructura jerárquica de uno o varios dominios, conectados mediante relaciones de confianza, transitivas y bidireccionales, que forma un espacio de nombres contiguo. Varios árboles de dominio pueden pertenecer al mismo bosque.

El árbol y la raíz de la consola

El árbol de la consola es una estructura jerárquica que se encuentra en el panel izquierdo de una consola de MMC. Muestra los elementos que están disponibles en una consola. Estos elementos pueden ser carpetas, complementos, controles, páginas Web y otras herramientas.

En una consola nueva, el único elemento del árbol de la consola es una carpeta con la etiqueta Raíz de la consola. Puede crear las funciones administrativas de la consola si agrega elementos a la consola.

No es necesario que ubique una ventana de consola en la raíz de la consola; puede ubicarla en cualquier elemento del árbol de la consola. Esto oculta los elementos del árbol por encima de la raíz y activa la ventana de la nueva raíz de consola y cualquier herramienta administrativa por debajo de ésta.

Contenedores en el árbol de la consola

Un contenedor es cualquier elemento del árbol de la consola al que se agregan objetos. Haciendo click en el signo más se expandirá un contenedor donde se podrá ver su contenido, y en el signo menos se contraerá el contenedor.

El contenedor básico es una carpeta, que se puede utilizar para agrupar elementos relacionados de un árbol de consola. Sin embargo, también puede utilizar MMC para agregar herramientas a elementos que no son carpetas y que también se convertirán en contenedores.

La raíz de la consola es un contenedor, cuya única función es contener el árbol de la consola. El elemento superior de un complemento es, normalmente, una carpeta u otro contenedor.

Otro tipo de elemento que puede contener otros elementos es un elemento de visión. Cuando hace clic en un elemento de visión, éste muestra una lista, texto o información gráfica en el panel de detalles en lugar de elementos adicionales. Las páginas Web son elementos de visión ya que requieren un explorador, que sólo se puede abrir en el panel de detalles. No se debe agregar elementos a los elementos de visión porque, si un usuario oculta el árbol de la consola, posiblemente no se dé cuenta de que un elemento de visión contiene otros elementos. Por el contrario, si dispone de un conjunto de elementos de visión, puede agregar una carpeta al árbol de la consola y agruparlos en la carpeta. De este modo, un usuario puede seleccionar entre todos estos elementos cuando se oculte el árbol de la consola.

Un tipo de elemento que no puede contener otros elementos es una hoja. Cuando se hace click en una hoja, ésta enumera los elementos en el panel de detalles. Normalmente, son elementos contenidos individualmente en el árbol de la consola. Sin embargo, cuando el número de elementos es muy elevado, un complemento utiliza una hoja en lugar de un contenedor.

Agregar una funcionalidad nueva a una consola

Es posible que ninguno de los complementos que proporciona el comando **Agregar o quitar complemento** ofrezca exactamente las posibilidades que se necesitan; se pueden intentar algunas de las siguientes soluciones posibles:

- Buscar complementos o extensiones adicionales que Microsoft u otros fabricantes que se hayan publicado en el Web.
- Crear su propia extensión para ampliar un complemento ya existente o bien desarrollar un complemento nuevo independiente.

Las empresas pueden aplicar las plantillas de seguridad en los siguientes casos:

- Cuando se crea un recurso, como una carpeta o archivo compartido, y aceptan la configuración predeterminada de la lista de control de acceso o implementan una configuración predeterminada.
- Cuando sitúan usuarios en los grupos de seguridad estándar, como Usuarios, Usuarios avanzados y Administradores, y aceptan la configuración predeterminada de la ACL que se aplica a dichos grupos de seguridad.
- Cuando utilizan las plantillas de directiva de grupo básica, compatible, segura o de alta seguridad que se incluyen en el sistema operativo.

La seguridad de Windows XP está basada en el sistema de archivos NTFS. Un sistema de archivos es el método para nombrar, almacenar y organizar archivos en el equipo. El sistema de archivos NTFS mejora la seguridad, puesto que permite cifrar archivos y carpetas y restringir el acceso a los mismos.

Algunas de las ventajas de utilizar el sistema de archivos NTFS son:

- La tecnología de Sistema de archivos cifrados (EFS) de Windows XP ayuda a proteger los datos importantes. Si cifra un archivo con EFS, sólo usted puede abrirlo y trabajar con él. EFS se basa en el cifrado de clave pública y aprovecha la arquitectura CryptoAPI de Windows XP. La configuración predeterminada de EFS no requiere de modificaciones, puede comenzar a cifrar los archivos inmediatamente. EFS genera automáticamente un par de claves de cifrado y un certificado para un usuario, si todavía no existe uno. Esto resulta útil en equipos portátiles porque, incluso si alguien lo encuentra o lo roba, no podrá tener acceso a los archivos del disco duro. EFS puede utilizar el Estándar de cifrado de datos expandido (DESX) o Triple-DES (3DES) como algoritmo de cifrado. Tanto el software RSA Base como RSA Enhanced, ambos incluidos en el sistema operativo por los proveedores de servicios de cifrado (CSP), se pueden utilizar para los certificados EFS y para el cifrado de las claves de cifrado simétrico. Si cifra una carpeta, todos los archivos y subcarpetas que se creen o agreguen a ella se cifrarán

automáticamente. Se recomienda cifrar a nivel de carpeta para evitar la creación de archivos temporales de texto sin formato en el disco duro durante la conversión de archivos.

- Puede utilizar EFS para cifrar archivos y carpetas sin conexión, y dispone de opciones nuevas para compartir archivos cifrados o deshabilitar los agentes de recuperación de datos. Además, es más fácil para administrar EFS mediante Directiva de grupo y utilidades de la línea de comandos. La Figura 2.4.1 muestra el lugar en que se crea la configuración de EFS.

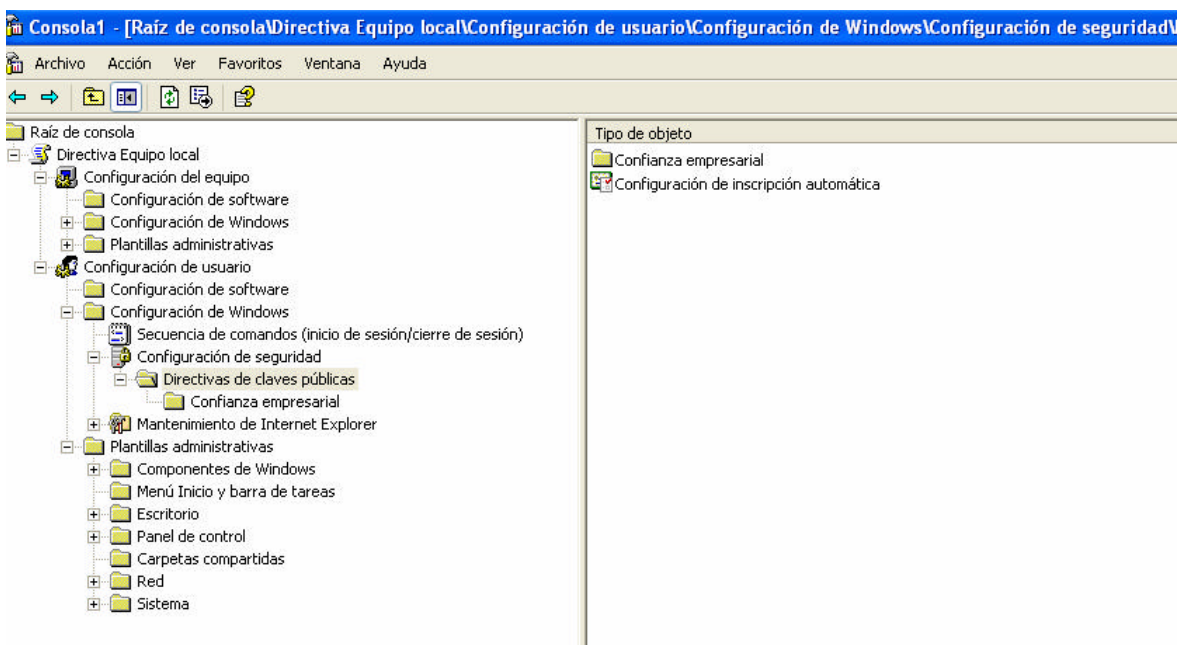


Figura 2.4.1 Configuración de seguridad local EFS.

- El sistema de archivos NTFS ofrece también una mejor compresión para el almacenamiento y ofrece compatibilidad para particiones y tamaños de archivo más grandes.

Windows XP es compatible con los sistemas de archivos NTFS, FAT16 y FAT32, de manera que puede ejecutar Windows XP y una versión anterior de Windows en un equipo con una configuración de arranque dual.

También proporciona acceso completo a los beneficios que proporciona Kerberos (protocolo de seguridad creado por MIT que usa una criptografía de claves simétricas para validar usuarios con los servicios de red, evitando así tener que enviar contraseñas a través de la red. Al validar los usuarios para los servicios de la red por medio de Kerberos, se frustran los intentos de usuarios no autorizados que intentan interceptar contraseñas en la red).

Windows XP soporta directivas locales y de grupo. Una directiva es un medio para definir la seguridad de un modo general por ejemplo, que todos los usuarios cambien sus contraseñas en determinado tiempo, determinar la longitud mínima de las contraseñas, controlar el acceso a los recursos y los datos; también ayudan a protegerse frente a virus y otros códigos maliciosos que se pueden extender a través del correo electrónico y en Internet.

Se pueden establecer directivas de restricción de software para:

- Hacer que el software se administre estrictamente, lo que permite decidir cómo, cuándo y dónde se puede ejecutar código.
- Hacer que el software no esté administrado, prohíbe ejecutar código específico.
- Aislar código no confiable en un área segregada, conocida como recinto de seguridad, hasta que compruebe que es seguro. Esto permite utilizar el software, incluso si no tiene claro que es seguro, a la vez que impide que dañe el sistema hasta que compruebe que sí es seguro.

Las directivas de restricción de software protegen también frente a datos adjuntos a mensajes de correo electrónico infectados, incluidos:

- Datos adjuntos de un archivo guardados en una carpeta temporal.
- Objetos y secuencias de comandos incrustados.

En la Figura 2.4.2 se muestra la configuración de directiva de restricción de software.

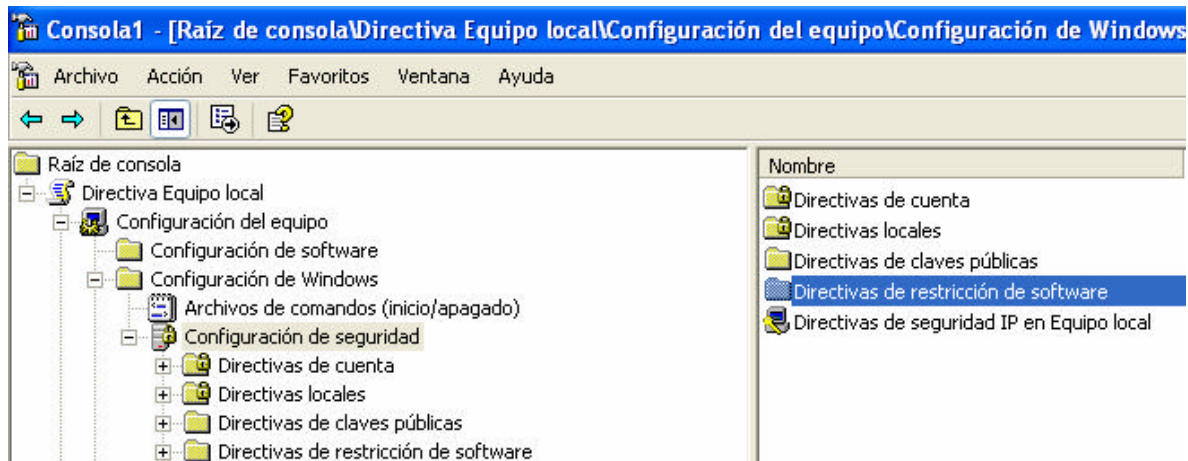


Figura 2.4.2 Directivas de restricción de software: Configuración de seguridad local.

En Internet, está protegido también frente a:

- Vínculos URL o UNC que pueden iniciar Internet Explorer, u otra aplicación, y descargar una página Web con una secuencia de comandos incrustada no confiable.
- Controles ActiveX descargados de Internet y que no son de confianza.

Algunos otros elementos que ofrece Windows XP para la seguridad son:

- Seguridad del Protocolo Internet (IPSec): El protocolo estándar que se utiliza para enviar y recibir información a través de Internet hace que los datos sean susceptibles de ser interceptados, modificados, reproducidos o falsificados. Windows XP utiliza IPSec para ofrecer una plataforma ideal para proteger las comunicaciones en una Intranet o en Internet. Antes de enviar los datos, IPSec negocia con el equipo al que está conectado, establece un nivel de seguridad apropiado para la sesión de comunicación.

Después, genera una clave de autenticación secreta y autentifica la identidad del otro equipo, antes de comenzar a intercambiar los datos de manera segura.

- Servidor de seguridad de conexión a Internet: Proporciona un alto nivel de seguridad para el hogar y para una pequeña red empresarial. Ya que utiliza filtros de paquetes activos, lo que significa que los puertos del servidor de seguridad se abren dinámicamente y permanecen abiertos sólo durante el tiempo necesario para permitir el acceso a los servicios que se utilizan. Impide a los piratas informáticos examinar los puertos y recursos de su equipo, incluidos los archivos y las impresoras compartidas. El ICF se habilita a nivel de conexión.
- Administración de cookies: Un cookie es un pequeño archivo que almacena un sitio Web en el equipo del usuario para ofrecer funciones de personalización, en Windows XP, puede establecer el nivel de privacidad que desee al explorar la Web, especificando si un sitio Web puede recopilar información acerca de usted a través de los cookies.
- Inicio de sesión personalizado: Cada una de las personas que utilice el equipo tendrá su propia interfaz, con su inicio de sesión y contraseña. Este nivel de seguridad garantiza que nadie pueda tener acceso a los documentos de cada individuo.
- Conexión compartida a Internet (ICS): Conecta varios equipos a Internet a través de una sola conexión. Los usuarios pueden compartir de forma segura entre varios equipos las conexiones DSL, de módem por cable o de línea telefónica. Un equipo, denominado el host ICS, se conecta directamente a Internet y comparte su conexión con el resto de los equipos de la red. Los equipos cliente se basan en el acceso a Internet que les proporciona el host ICS. Con ICS habilitado, la seguridad mejora porque el host ICS es el único equipo visible para Internet. Toda comunicación desde los equipos cliente a Internet debe pasar a través del host ICS, lo que permite ocultar sus direcciones a Internet. Los equipos cliente están

protegidos porque no pueden ser vistos desde el exterior de la red. El equipo que ejecuta ICS es el único que puede ser visto desde el lado público de la conexión. Además, el host ICS se encarga del direccionamiento de red. Se asigna a sí mismo una dirección permanente y proporciona el Protocolo de configuración dinámica de host (DHCP) a los clientes ICS. El host ICS asigna una dirección única a cada cliente y, de esta forma, permite que se comuniquen con otros equipos de la red.

- Uso de protocolos de red: La característica ICS de Windows XP proporciona Traducción de direcciones de red (NAT), DHCP y Servicio de nombres de dominio (DNS) a las redes domésticas, lo que elimina la necesidad de configuración de los clientes por parte del usuario.

En general Windows XP ofrece una variedad de herramientas con las que podemos estar tranquilos de que la información estará segura, aunque esto también dependerá del administrador y de los planes de seguridad que se tengan, para poder sacar provecho de las capacidades que ofrece este sistema operativo.

2.5 REDES Y COMUNICACIONES

Definición de una red

Una red es un conjunto de terminales, nodos, servidores y elementos de propósito especial que interactúan entre sí con la finalidad de intercambiar información y compartir recursos. En la figura 2.5.1 se muestra un ejemplo de una red.

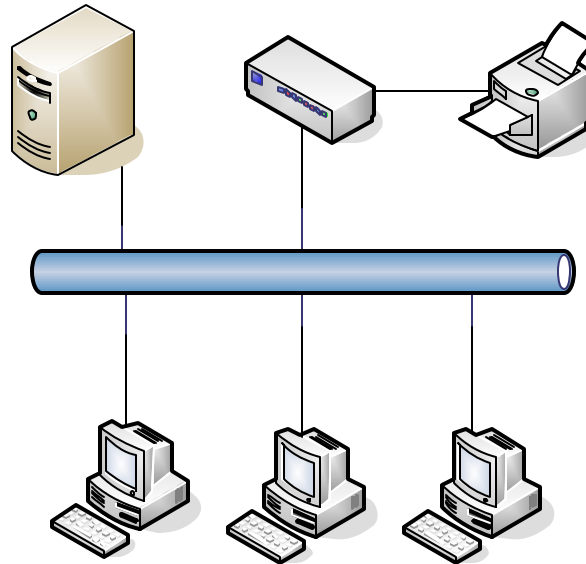


Figura 2.5.1 Ejemplo de una red.

Clasificación por Cobertura

De acuerdo a su medida (o cobertura), las redes son clasificadas de la siguiente manera:

- **Local Area Network (LAN).** Las cuales son típicamente extendidas hasta un kilómetro. Ejemplo de este tipo de redes son la red Ethernet o Token Ring.
- **Wide Area Network (WAN).** Redes de área amplia. Redes que tienen una cobertura mundial, por ejemplo Frame Relay, ATM, ISDN.

- **Metropolitan Area Network (MAN).** Redes de cobertura de 10, 20 ó 30 km. Actualmente las redes MAN no son muy utilizadas, ya que han sido absorbidas por las redes WAN y LAN, aunque actualmente se encuentran todavía.

Elementos de una Red

- **Repetidor:** Es un dispositivo de red que regenera la señal eléctrica que le llega con el fin de restituir su nivel original y así evitar los problemas que se pudieran producir por una excesiva atenuación. En ocasiones los repetidores se pueden utilizar para convertir la señal de un sistema de cableado en otro. Un repetidor podría tener una entrada coaxial y otra par trenzado.

La ventaja principal de un repetidor reside en la facilidad de operación, se limita a copiar bits de un segmento de red en otros. No requiere ningún tipo de operación especial. No entiende las direcciones de red, protocolos, servicios, etc. Se limita a repetir la señal de la red a un gran número de salidas/entrada. La mayor limitación del repetidor consiste en que no aísla los problemas del tráfico generados en la red en cada uno de sus segmentos. De hecho, si en uno de los segmentos se produce una colisión, ésta se propagará por todos los segmentos de la red. La operación mas inteligente que es capaz de realizar un repetidor consiste en aislar los segmentos de red en caso de rotura de cable en alguno de ellos.

- **Bridge (Puente):** Es un equipo que contiene dos puertos de comunicación y que crea tablas en memoria que contienen todas las direcciones de las tarjetas en la red, de ambos extremos, de tal manera que restringen el tráfico de datos de un segmento a otro, no permitiendo el paso de tramas que tengan como destino una dirección del mismo segmento al que pertenece la estación origen. Es conveniente el uso de

los mismos cuando se requiere llevar a cabo la interconexión de dos redes LAN locales o remotas.

- **Router (Ruteador):** Son dispositivos que permiten unir varias redes (más de dos, a diferencia de los puentes), tomando como referencia la dirección de red de cada segmento. Al igual que los puentes, los ruteadores restringen el tráfico local de la red permitiendo el flujo de datos a través de ellos solamente cuando los datos son diseccionados con esta intención.

- **Hubs (Concentrador):** Son los repetidores utilizados normalmente para la distribución de señal a través de par trenzado, formando una red en estrella.

- **Gateway:** Dispositivo de enlace entre redes de características diferentes. Sistema de hardware o software que hace de puente entre dos aplicaciones o redes incompatibles para que los datos puedan ser transferidos entre distintas computadoras. Se utilizan regularmente con los correos electrónicos que circulan entre los sitios de Internet y las redes privadas.

- **Backbone:** Es un enlace de gran caudal o una serie de nodos de conexión que forman un eje de conexión principal. Es la columna de una Red.

- **Switch:** Es un dispositivo de interconexión de redes de computadoras que opera en la capa 2 (nivel de enlace de datos) del modelo OSI (*Open Systems Interconnection*). Un switch interconecta dos o más segmentos de red, funcionando de manera similar a los puentes (bridges), pasando datos de un segmento a otro, de acuerdo con la dirección MAC de destino de los datagramas en la red. Los switches se utilizan cuando se

desea conectar múltiples redes, fusionándolas en una sola. Al igual que los bridges, dado que funcionan como un filtro en la red, mejoran el rendimiento y la seguridad de las LAN.

En la siguiente figura 2.5.2 se muestra un diagrama donde se describe la localización de dispositivos de una red.

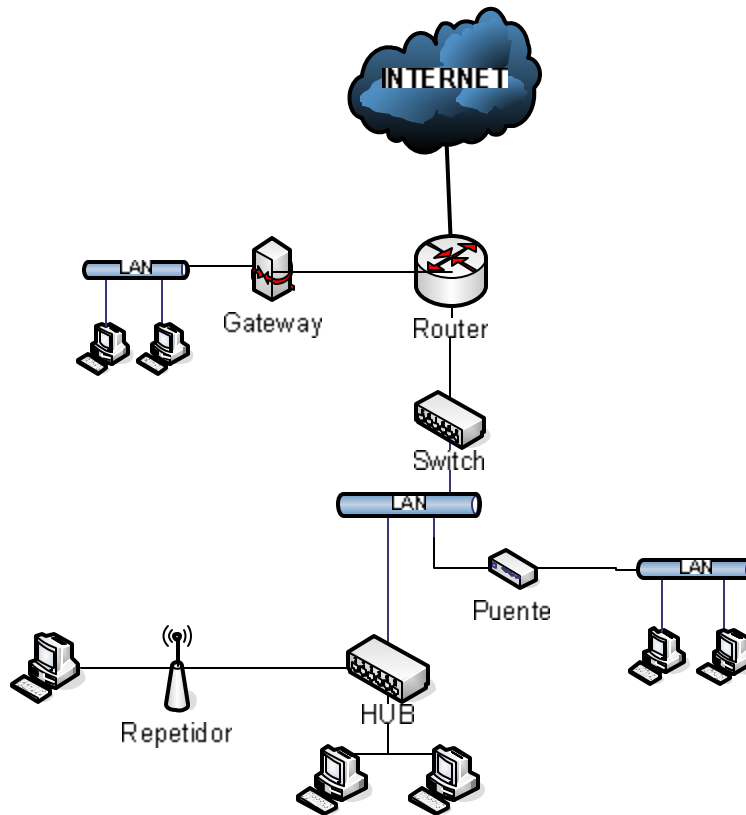


Figura 2.5.2 Diagrama de ubicación de dispositivos en una Red.

Topologías de Red

Entre los principales tipos de Topologías de red tenemos:

- **Topología de BUS:** Consiste en un cable con un terminador en cada extremo del que se cuelgan todos los elementos de una red. Todos los Nodos de la Red están unidos a este cable, el cual recibe el nombre de “Backbone Cable”. Las ventajas de esta topología son: es más fácil de conectar nuevos nodos a la red y requieren menos cable que una topología estrella. Sus desventajas son que toda la red se caería si hubiera una ruptura en el cable principal. Se requiere terminadores. Es difícil detectar el origen de un problema cuando toda la red cae. No se debe utilizar como única solución en un edificio. Ver la siguiente figura 2.5.3.

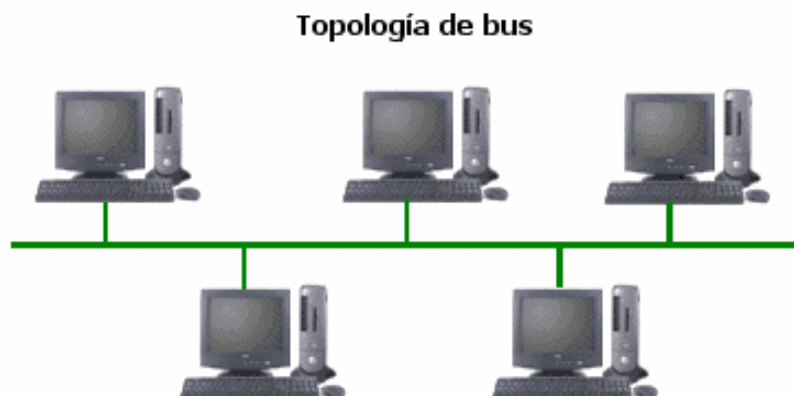


Figura 2.5.3 Topología de bus.

- **Topología de Estrella:** En esta topología todos y cada uno de los nodos de la red se conectan a un concentrador y/o switch. Los datos de estas redes fluyen del emisor hasta el concentrador, éste realiza todas las funciones de la red, además actúa como amplificador de los datos. Sus ventajas son: Gran facilidad de instalación. Posibilidad de desconectar elementos de red sin causar problemas. Facilidad para la detección de fallo y su reparación. Las desventajas serían: requiere más

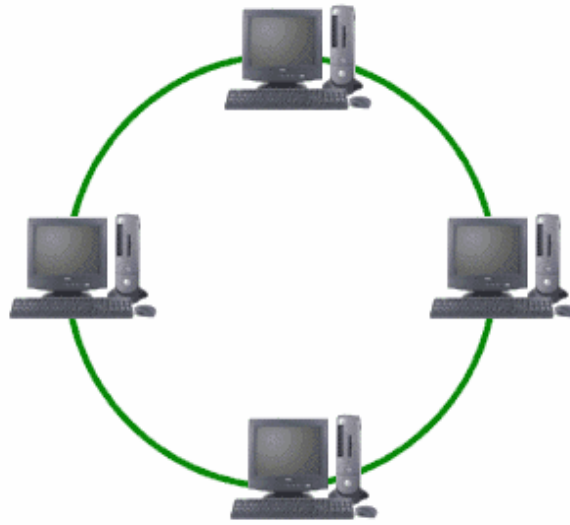
cable que la topología de bus, un fallo en el servidor o concentrador provoca el aislamiento de todos los nodos a él conectados. Se requieren hubs o concentradores. Vea figura 2.5.4.



Figura 2.5.4 Topología de estrella.

- **Topología Anillo:** Conecta computadoras en un simple círculo de cable. Las señales viajan alrededor del bucle en una dirección y pasan a través de cada computadora. A diferencia de la topología de bus, cada computadora actúa como un repetidor que impulsa la señal y la envía a la siguiente computadora. Debido a que la señal pasa a través de cada computadora, la falla de una computadora puede impactar a la red entera. Vea figura 2.5.5.

Topología en anillo

**Figura 2.5.5 Topología de anillo .**

- **Topología de Malla:** Cada estación está conectada con todas (red completa) o varias (red Incompleta) estaciones formando una estructura que puede ser regular (simétrica) o irregular. El costo en medios de comunicación depende del número de conexiones y suele ser elevado, ganando sin embargo en fiabilidad frente a fallos y en posibilidades de reconfiguración. El costo de instalación al aumentar el número de estaciones es también grande y sobre todo de difícil realización en una red ya instalada, lo que representa un gran inconveniente en redes locales. No se adapta a grandes dispersiones geográficas pero permite tráficos elevados con retardos de medios a bajos. Vea figura 2.5.6.

Topología en malla completa

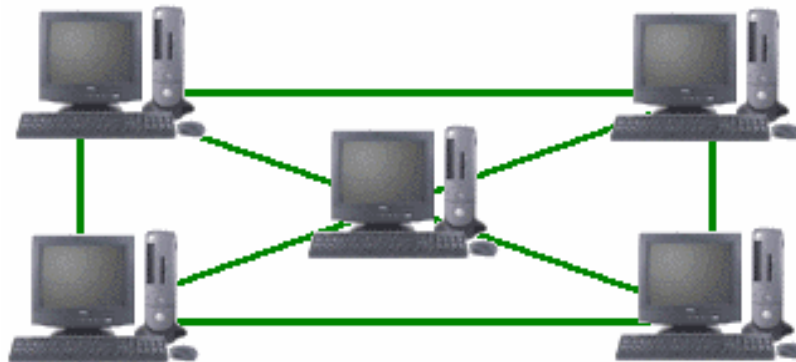


Figura 2.5.6 Topología de Malla.

Comunicaciones

Transferencia electrónica de información de un lugar a otro. Las comunicaciones de datos se refieren a las transmisiones digitales, y las telecomunicaciones, a transmisión análoga y digital, incluyendo voz y vídeo.

Modelo TCP/IP

TCP/IP proponen cuatro capas en las que las funciones de las capas de Sesión y Presentación son responsabilidad de la capa de Aplicación y las capas de Enlace de Datos y Física son vistas como la capa de Interfase a la Red. Por tal motivo para TCP/IP sólo existen las capas Interfase de Red, la de Intercomunicación en Red, la de Transporte y la de Aplicación. Como puede verse TCP/IP presupone independencia del medio físico de comunicación, sin embargo existen estándares bien definidos a nivel de Enlace de Datos y Físico que proveen mecanismos de acceso a los diferentes medios y que en el modelo TCP/IP deben considerarse la capa de Interfase de Red; siendo los más usuales el proyecto IEEE802, Ethernet, Token Ring y FDDI. Vea Tabla 2.5.1

Breve descripción del modelo TCP/IP

Capas de TCP/IP	Descripción del Modelo de Capas de TCP/IP
Capa de Aplicación.	Invoca programas que acceden servicios en la red. Interactúan con uno o más protocolos de transporte para enviar o recibir datos, en forma de mensajes o bien en forma de flujos de bytes.
Capa de Transporte.	Provee comunicación extremo a extremo desde un programa de aplicación a otro. Regula el flujo de información. Puede proveer un transporte confiable asegurándose que los datos lleguen sin errores y en la secuencia correcta. Coordina a múltiples aplicaciones que se encuentren interactuando con la red simultáneamente de tal manera que los datos que envíe una aplicación sean recibidos correctamente por la aplicación remota, esto lo hace añadiendo identificadores de cada una de las aplicaciones. Realiza además una verificación por suma, para asegurar que la información no sufrió alteraciones durante su transmisión.
Capa Internet.	Controla la comunicación entre un equipo y otro, decide qué rutas deben seguir los paquetes de información para alcanzar su destino. Conformar los paquetes IP que serán enviados por la capa inferior. Desencapsula los paquetes recibidos pasando a la capa superior la información dirigida a una aplicación.
Capa de Interfase de Red.	Emite al medio físico los flujos de bit y recibe los que de él provienen. Consiste en los manejadores de los dispositivos que se conectan al medio de transmisión.

Tabla 2.5.1 Características del Modelo TCP/IP

En la siguiente figura 2.5.7 se muestra el modelo arquitectónico del Modelo TCP/IP.

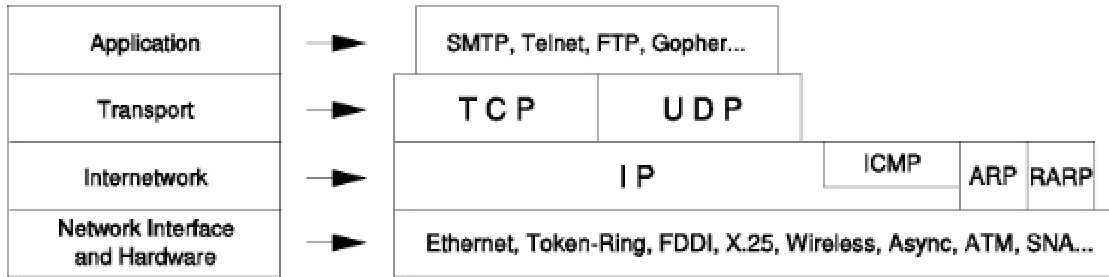


Figura 2.5.7 Modelo Arquitectónico de TCP/IP.

CAPÍTULO III

PLANTEAMIENTO DEL PROBLEMA Y SOLUCIÓN

3.1 PROBLEMÁTICA ACTUAL

Para comprender la problemática que se tiene en un consultorio médico, haremos uso del diagrama 3.1.1, el cual ilustra el primer proceso que se realiza en un consultorio médico. Este proceso es la generación de la cita médica, la cual se puede dar de dos formas; la primera es por la vía telefónica (lado izquierdo) y la segunda por vía presencial (lado derecho).

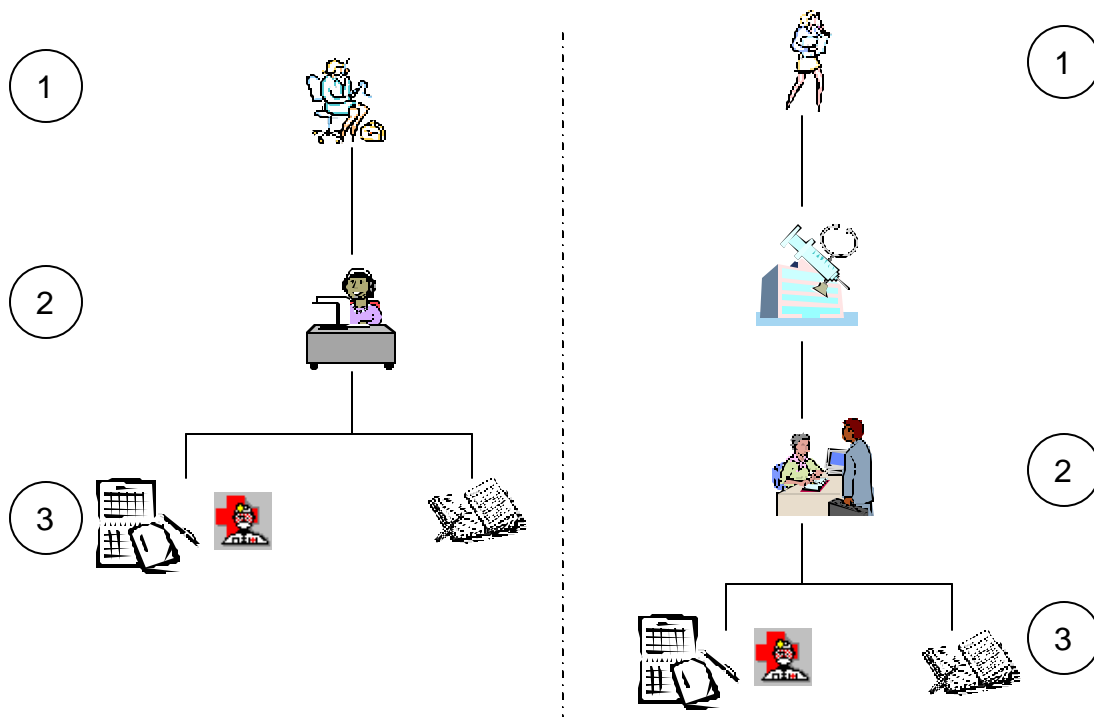


Diagrama 3.1.1 Proceso de Citas Telefónicas (izq.) y Presenciales (der.)

En el diagrama 3.1.1 se puede observar que existen bloques o niveles numerados del uno al tres. Estos bloques indican los principales puntos en donde se generan las problemáticas para el proceso de generación de una cita médica. A continuación, por cada bloque o nivel se listan las problemáticas existentes.

Problemáticas del Nivel 1 del Proceso de Citas Médicas (Paciente)

- Por el hecho de que el registro de los datos se realiza en un cuaderno u otro medio de papel semejante, obliga al paciente a repetir los mismos datos cada vez que solicita una cita.
- El paciente no ahorra tiempo en realizar una cita médica por teléfono debido a la repetitiva solicitud de sus datos.
- Para concertar una cita médica se requiere conocer el horario de atención del personal de recepción y del médico.

Problemáticas del Nivel 2 del Proceso de Citas Médicas (Persona de Recepción)

- El personal de recepción duplica los datos del paciente subsecuente (no nuevo) invirtiendo tiempo innecesario.
- La información que obtiene el personal de recepción por vía telefónica es breve, por lo que no logra obtener más datos generales del paciente.
- Para generación de información histórica, el personal de recepción debe hacer una búsqueda minuciosa en el cuaderno que utiliza para registro de datos, lo que implica que tome más tiempo en el proceso y se vuelva menos eficiente.
- El personal de recepción no cuenta con un mecanismo que le permita determinar si el paciente es nuevo o subsecuente.

Problemáticas del Nivel 3 del Proceso de Citas Médicas (Registro de Datos)

- El registro de los datos de las citas no tiene una estructura uniforme lo que dificulta su interpretación y la localización de un dato en específico.
- La caligrafía del personal de recepción puede ser ilegible para los otros miembros del consultorio.

- La información sobre el paciente puede ser registrada en diversos medios, por ejemplo, un cuaderno, **post-it** (Nota), un pedazo de papel, etc. Esto implica que se pueda perder la información, o bien, se invierta tiempo en registrarla en el medio correcto.
- Debido al estilo del registro de citas que cada personal de recepción tiene, implica que la información registrada no sea de fácil interpretación por parte de los otros miembros del consultorio.
- No existe un respaldo de los cuadernos de registro de citas médicas.
- El cuaderno de registro de citas está expuesto a eventualidades como derrame de líquidos, manchas por residuos alimenticios, etc.
- El personal de recepción se enfrenta con un serio problema al tener que modificar el registro de citas a causa de una eventualidad que tenga que ser atendida por el médico (emergencias de cirugías, problemas personales, salud, etc.).
- No es posible generar estadísticas acerca de los pacientes debido a la inconsistencia de los datos.

El segundo proceso que se realiza en un consultorio médico es la consulta del paciente. Para darnos una idea del proceso de consulta médica, haremos uso del diagrama 3.2.1 que ilustra los principales niveles o bloques.

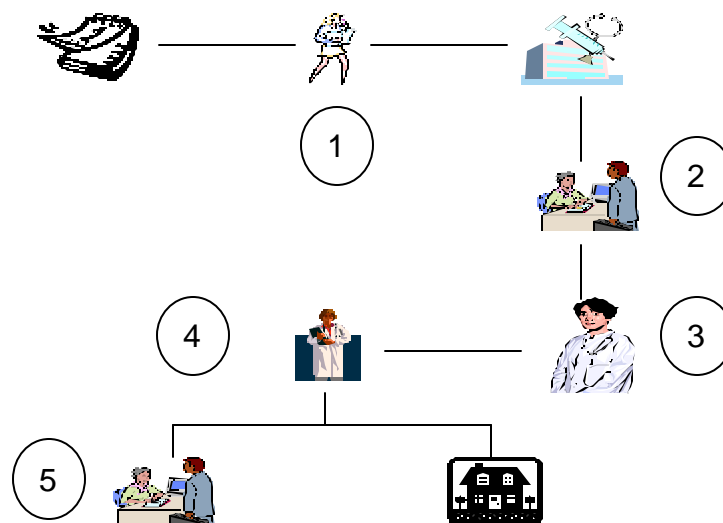


Diagrama 3.2.1 Proceso de Consultas Médicas.

Problemáticas del Nivel 1 del Proceso de Consultas Médicas (Paciente).

- Cuando el paciente llega tarde a su cita, produce un desfase en las citas que le siguen y a su vez provoca un descontrol sobre la administración del tiempo por parte del personal de recepción y del propio médico.
- Para el caso de consultas subsecuentes se ve obligado a acudir de nuevo con el personal de recepción.

Problemáticas del Nivel 2 del Proceso de Consultas Médicas (Personal de Recepción)

- Se emplea tiempo considerable en buscar y reunir los expedientes físicos de todos los pacientes que tendrán consulta.
- El personal de recepción está demasiado ocupado, ya sea bien por llamadas telefónicas, cobros a pacientes o por trámites administrativos.
- Para el caso de pacientes “nuevos”, el personal de recepción invierte tiempo en crear el expediente del paciente y solicitarle sus datos generales.
- El personal de recepción desconoce el tiempo que se demorará el médico en atender al paciente.
- No existe un mecanismo que permita al personal de recepción recordar al paciente su cita médica, y a su vez, confirmar la misma.
- Para el caso en donde el médico se ve en la necesidad de cambiar citas médicas por eventualidades (emergencias quirúrgicas, salud, etc.), el proceso de reestructuración de las citas se vuelve tedioso y complejo para el personal de recepción por lo mismo que se lleva en papel (cuaderno).
- El personal de recepción emplea un tiempo considerable en archivar los expedientes de los pacientes.

Problemáticas del Nivel 3 del Proceso de Consultas Médicas (Personal de Enfermería)

- Las mediciones de los signos vitales se escriben con puño y letra del personal de enfermería y esto ocasiona dificultad al interpretar los datos por parte de los otros integrantes del consultorio.
- El personal de enfermería se ve obligado anexar cada uno de los formatos que componen un expediente clínico cuando el paciente es nuevo, o bien, por extravío del historial.

Problemáticas del Nivel 4 del Proceso de Consultas Médicas (Médico)

- Desconoce que paciente es el que sigue y cual es la gravedad de su padecimiento o motivo de consulta.
- El médico no puede consultar de manera simultánea su agenda debido a que el personal de recepción la controla y está en un solo cuaderno.
- No existe respaldo de los expedientes físicos.
- Cada vez que surge un extravío o se traspapela un expediente, el médico dedica tiempo en recolectar nuevamente los datos necesarios.
- El médico invierte tiempo considerable en la revisión del expediente del paciente debido a la forma en que esté conformado u ordenado, al número de hojas que lo conformen, la cantidad de estudios clínicos y de gabinete, etc.
- Los formatos de registro de datos de los pacientes (generales, clínicos, etc.) se hacen en hojas de papel (con un cierto formato) y con la letra del médico o el personal de recepción (según sea el caso), lo que origina dificultad al interpretarlos (debido a la caligrafía empleada) y tiempo en llenarlos.
- Los documentos que extiende el médico al paciente (recetas, nota médica, etc.) no son del todo claros debido a la caligrafía empleada por éste mismo.

- El médico invierte tiempo de más cuando tiene que dar una copia de la nota médica al paciente, esto debido a que todo lo realiza de forma manual o escrita.
- Las recetas médicas no hacen mención al tipo de dietas y cuidados que en ocasiones son necesarias en el tratamiento del paciente y mucho menos tienen una estructura o formato estándar.
- No se hace entrega al paciente de una interpretación médica referente a los resultados de los estudios de laboratorio y gabinete debido a que se requiere tiempo de más para poder transcribirlos en papel y en términos coloquiales.
- La actitud del médico puede producir que el paciente no sea del todo franco y explícito en lo referente a los antecedentes clínicos y padecimientos.
- En el caso de que el paciente sea referido al médico, se desconoce su historial clínico y hay que generarlo, lo que implica que la consulta tome más tiempo de lo necesario.
- Por cada especialista del consultorio clínico que consulte el paciente, se le genera un expediente en particular provocando duplicidad.
- En caso de consultas subsecuentes, el médico no puede generarlas debido a que el personal de recepción es quien posee el cuaderno de citas, por ende, el paciente tiene que coordinarse con el personal de recepción y el médico.

3.2 REQUERIMIENTOS GENERALES Y PARTICULARES

Definición de requerimiento

Un requerimiento puede ser definido como una capacidad del sistema necesaria por el usuario para resolver un problema o alcanzar un objetivo.

Una buena identificación de requerimientos, una clara y específica definición de objetivos y un buen plan son claves en el éxito de un proyecto.

Para el desarrollo del sistema se consideran dos rubros:

- Requerimientos generales. Cubren las necesidades generales con las que debe contar el sistema.
- Requerimientos particulares. Cubren las necesidades específicas solicitadas por el usuario.

Requerimientos Generales

Entre los requerimientos generales se encuentran los siguientes:

- Contar con un sistema que permita la administración médica en un consultorio.
- El sistema deberá tener los siguientes módulos:
 - Citas.
 - Pacientes.
 - Médicos.
 - Administración de usuarios.

- Que el sistema permita tener centralizada la información.
- El sistema deberá permitir que más de un usuario pueda hacer uso del mismo a la vez.
- El sistema deberá proporcionar seguridad a la información.
- El sistema debe de presentar un entorno amigable y de fácil uso.
- El módulo de instalación debe ser relativamente fácil de ejecutar para que no presente rechazo por parte del personal encargado de su proceso.

Requerimientos Particulares

Los requerimientos particulares son los siguientes:

- Deberá existir un administrador del sistema que pueda crear cuentas de usuario y asignarles perfiles.
- La aplicación deberá solicitar un usuario y contraseña para acceder a la misma.
- El proceso de citas deberá permitir seleccionar la fecha y hora de acuerdo a la disponibilidad del médico elegido. Las citas podrán cancelarse en cualquier momento.
- El médico podrá consultar las citas agendadas.

- La información en el módulo de médicos deberá ser controlada únicamente por el administrador del sistema, quien dará de alta o modificará los datos.
- La información correspondiente a los médicos no podrá ser eliminada.
- El módulo de médicos deberá contar con un catálogo de turnos y especialidades.
- Se requiere que en el módulo de pacientes se cuente con el expediente completo del paciente, esto es, que contenga los siguientes elementos: datos generales, historia clínica, notas médicas, recetas médicas y hojas de referencia.
- Los datos del expediente no podrán ser eliminados.
- La búsqueda de un expediente podrá ser de diferentes maneras.
- La información del expediente deberá ser restringida, siendo el médico el único usuario que tendrá acceso a todo el módulo.
- Los datos generales de los pacientes serán la única parte del módulo a la que tendrán acceso usuarios diferentes a médico.
- Se podrán generar impresiones de los documentos más importantes.
- Es necesario poder consultar las notas, recetas y hojas de referencia generadas a los expedientes de los pacientes.
- El sistema tienen que mostrar una forma estándar en cuanto a tamaño, color, tipo y tamaño de letra y ubicación. Las ventanas tienen que

presentar colores no molestos para la vista y con una diferencia clara en el contraste entre las letras y el color del fondo de la ventana generando con ellos que la familiarización de los usuarios con el sistema sea más rápido.

- Los datos mínimos requeridos deberán identificarse.

- El sistema deberá estar en una red local.

3.3 RECOPIACIÓN Y ANÁLISIS DE LA INFORMACIÓN

El objetivo que se persigue con la recopilación de la información que se levanta al momento de acudir a una consulta, es realizar un análisis de los datos con los cuales se cuenta para determinar los que son útiles para cada uno de los puntos que intervienen en la construcción de un expediente clínico de manera electrónica, esto con la finalidad de que no haya redundancia de los mismos y los usuarios del sistema tengan acceso a ellos de manera sencilla y eficaz.

Citas

Las citas actualmente son agendadas por la secretaria del consultorio, quien lleva el control a través de un registro de los pacientes que atiende cada médico, en dicha agenda se recaban los siguientes datos:

- Fecha de la cita. La que corresponde al día
- Hora de la cita
- Nombre completo del paciente
- Médico al que consulta
- Teléfono del paciente

Historia Clínica

La historia clínica es el cuestionario realizado por el médico o la enfermera ya sea de manera directa o indirecta al paciente con el objetivo de contar por escrito con sus antecedentes patológicos, antecedentes no patológicos y con el padecimiento que actualmente tiene, el cuestionario consta de los siguientes campos que son llenados de manera manual y de forma abierta:

- Interrogatorio directo. Si es contestado por el mismo paciente.

- Interrogatorio indirecto. Si el paciente en ese momento no puede responder y el cuestionario es respondido por algún familiar o acompañante.
- Fecha.
- Nombre completo.
- Número de expediente. Este se le asigna en el momento del interrogatorio.
- Sexo.
- Edad.
- Estado civil.
- Ocupación.
- Lugar de nacimiento.
- Residencia habitual
- Domicilio.
- Religión.
- Antecedentes Heredo-familiares.
- Antecedentes familiares no patológicos.
- Antecedentes familiares patológicos.
- Antecedentes gineco-obstétricos.
- Antecedentes terapéuticos
- Padecimiento actual
- Revisión por aparatos y sistemas
- Inspección general
- Registro de signos vitales: tensión arterial, pulso, respiración, temperatura, peso y talla.
- Exploración física general: cabeza, tronco, cuello, abdomen, genitales y extremidades.
- Impresión diagnóstica.
- Estudios de laboratorio y gabinete que se sugieren practicar por el padecimiento actual.
- Tratamiento a seguir para combatir el padecimiento con el que llega el paciente.

- Definitivo. Diagnóstico general definitivo de acuerdo al tratamiento empleado.

Órdenes y Notas Médicas

Las notas médicas son escritas por el médico que realiza la consulta al paciente, consta de una o más hojas en donde va anotando el o los padecimientos encontrados al momento de la consulta, solo va diferenciando las notas, la fecha y hora en que fueron escritas, si encuentra necesario realizar algún estudio este será anotado en el campo de orden. Por lo que en resumen los campos que se utilizan actualmente en este formato son los siguientes:

- Nombre completo del paciente.
- Número de expediente.
- Fecha y hora en que se realiza la consulta.
- Órdenes. Si se tiene que realizar algún estudio o recomendación para el tratamiento.
- Notas. Resumen de padecimiento encontrado y tratamiento a seguir.

Hoja de Referencia y Contrarreferencia

La hoja de referencia y contrarreferencia se utiliza cuando es necesario remitir al paciente a otra institución, ya sea por no contar con el equipo necesario, o bien, requiera atención hospitalaria, o viceversa, que llegue algún paciente referido de alguna otra institución médica.

En estos casos, el médico que realiza la consulta llena la hoja con los siguientes datos:

- Nombre completo del paciente.
- Número de expediente. Si cuenta con él en el consultorio médico.
- Sexo.

- Fecha y hora de envío.
- Domicilio del paciente: calle y número, colonia, municipio o delegación, entidad federativa, código postal.
- Teléfono.
- Establecimiento al que envía.
- Establecimiento de recepción.
- Fecha y hora de recepción.
- Firma del médico que envía.
- Resumen clínico.
- Motivo del envío.
- Diagnóstico.
- Terapéutica empleada.
- Signos vitales: frecuencia cardiaca (fc), frecuencia respiratoria (fr), presión arterial (ta), temperatura.
- Firma del médico que envía.

Receta Médica

Las recetas médicas que se les dan a los pacientes en cada consulta tienen como finalidad indicarle cuales son los medicamentos que debe consumir para el tratamiento y las indicaciones necesarias en cuanto a frecuencia, horarios y duración del tratamiento.

Las recetas son personalizadas y tienen como encabezado el nombre del médico, especialidad, cédula profesional. La información propia de la receta incluye:

- Fecha de la consulta.
- Nombre completo del paciente.
- Clave de la receta. Dato que no siempre es llenado solo indica si es la primera, segunda o número de receta en la consulta.
- Edad

- Peso
- Talla
- Temperatura
- Tensión arterial
- Alergias. Si es que presenta alguna el paciente.
- Teléfono del médico
- Rp1. indicaciones de la receta
- Nombre y firma del médico.

Análisis y Estudios de Gabinete

Para los análisis y estudios de gabinete no existe algún formato, únicamente se le indica al paciente dentro de la misma receta que estudios se tiene que practicar. Una vez que el paciente llega a su siguiente cita con los resultados de los estudios, el médico realiza las notas que considera adecuadas dentro del formato de órdenes y notas médicas.

Análisis de la Información

Una vez que se realizó un análisis sobre la información ya presentada, se determina que en varios de los procesos que se realizan, existe redundancia de información lo cual ocasiona confusión, haciendo lentos e ineficaces los procesos.

Es importante identificar los datos que corresponden a cada uno de los puntos con los cuales se construye un expediente de manera electrónica, por lo que a continuación se presenta la información como debe recopilarse para hacer más rápidos y eficaces los procesos, los cuales tendrán como resultado un mejor manejo y control de la información de cada uno de los pacientes que acuden a recibir atención médica

Citas

Para generar una cita de manera ágil y sencilla es necesario conocer los posibles médicos que proporcionan la atención de acuerdo a la especialidad requerida. Una vez seleccionado el médico de preferencia, será necesario determinar la fecha y hora que se desea la cita. Para esto será necesario recopilar la siguiente información:

- Fecha de la cita. La que corresponde al día.
- Hora de la cita.
- Nombre completo del paciente.
- Médico al que consulta.
- Teléfono del paciente. Para localización y confirmación.

Pacientes

Debe existir un módulo en el cual se recopile la información necesaria para el registro de un nuevo paciente con los datos generales que serán utilizados en varios de los módulos que componen el expediente clínico sin necesidad de capturarlos cada vez que se utilicen. Los datos necesarios son:

- Nombre del paciente. Capturando su apellido paterno, materno y nombre por separado.
- Sexo.
- Fecha de nacimiento.
- Edad.
- Lugar de nacimiento.
- Estado civil.
- Ocupación.
- Dirección (calle y número, colonia, delegación o municipio, código postal, estado).

- Teléfono.
- Religión.

Historia Clínica

La historia clínica únicamente debe comprender la información correspondiente a los antecedentes clínicos patológicos y no patológicos del paciente, por lo que los siguientes datos deben ser los únicos por llenar:

- Expediente electrónico. El cual será único para cualquier movimiento que se realice independientemente del módulo en el que se encuentre el usuario.
- Fecha de creación de la historia.
- Interrogatorio. Puede ser directo e indirecto.
- Antecedentes familiares no patológicos.
- Antecedentes familiares patológicos.
- Antecedentes gineco-obstétricos.
- Antecedentes terapéuticos.
- Padecimiento actual.
- Nombre de la persona que realiza el interrogatorio, quien puede ser una enfermera o el mismo médico.

Nota Médica

Lo importante de las notas médicas es que en ellas se establece el padecimiento encontrado por el médico por lo que en ella deben anotarse:

- Fecha y hora de creación.
- Signos vitales y datos generales: peso, talla, frecuencia respiratoria, frecuencia cardiaca, pulso, tensión arterial, temperatura.

- Nota referente al estado del paciente de manera general y abierta al médico.
- Diagnóstico.
- Plan a seguir como tratamiento al padecimiento.
- Análisis y estudios de gabinete a los que deba someterse el paciente o bien los comentarios sobre los resultados de los análisis de laboratorio o los estudios de gabinete.

Hoja de Referencia y Contrarreferencia

La hoja de referencia se debe utilizar únicamente en los casos en los que el paciente deba ser atendido en alguna otra institución hospitalaria, sin embargo cuando un paciente es referido al consultorio, es necesario registrarlo como un nuevo paciente y por lo tanto generarle su expediente e irle creando sus notas médicas conforma pase a consultas, por lo que la hoja de referencia quedaría con la siguiente información:

- Fecha y hora de envío.
- Establecimiento al que envía.
- Resumen clínico.
- Motivo del envío.
- Diagnóstico.
- Terapéutica empleada.

Receta Médica

Como ya se ha mencionado la receta médica debe contener las indicaciones de los medicamentos que debe tomar el paciente como parte de su tratamiento con la explicación de cada uno de ellos y las indicaciones como dosis y horarios. No es necesario capturar toda la información que en esta aparezca pues ya los datos serán capturados en algún otro módulo como lo son los signos vitales en la nota

médica o los datos generales del paciente en el módulo de pacientes, por lo que los campos a capturar en esta parte únicamente comprenderán:

- Fecha en que se elabora la receta.
- Rp1. indicaciones de los medicamentos a tomar como parte del tratamiento.

Médicos

Es importante comentar que debe existir un módulo en el cual se capturen los datos generales de los médicos para que se empleen en los módulos que sean necesarios como la receta, la historia clínica, la nota médica con la siguiente información:

- Nombre completo del médico.
- Especialidad.
- Cédula profesional.
- Teléfono.
- Dirección.
- Horarios de atención del médico.

Administración del Sistema

Este proceso no se tiene actualmente, sin embargo es necesario que exista ya que en él se podrán determinar los accesos y permisos de cada uno de los usuarios que van a manejar la información del sistema. Para este proceso es necesario recopilar la siguiente información:

- Nombre del usuario.
- Identificador de usuario.
- Contraseña del usuario.
- Permisos sobre la información: agregar, consultar, modificar, eliminar.

3.4 IDENTIFICACIÓN DEL PROBLEMA

Como se menciona en el apartado 3.1 Problema Actual, el manejo de la clínica enfrenta una serie de problemas en los procesos que competen a la administración de la misma; siendo estos los que consideraremos para presentar una solución a través de un sistema de información que permita automatizar los procesos y mejorar los tiempos, así como la calidad del servicio de los miembros de la clínica hacia sus pacientes con el objetivo de estar a la vanguardia y ser reconocidos como una buena institución médica.

Se menciona que existen dos procesos básicos: registro de citas y consulta. Primero se analizará lo correspondiente a las situaciones que se presentan durante el proceso que denominamos cita. La figura 3.4.1 muestra tal proceso.

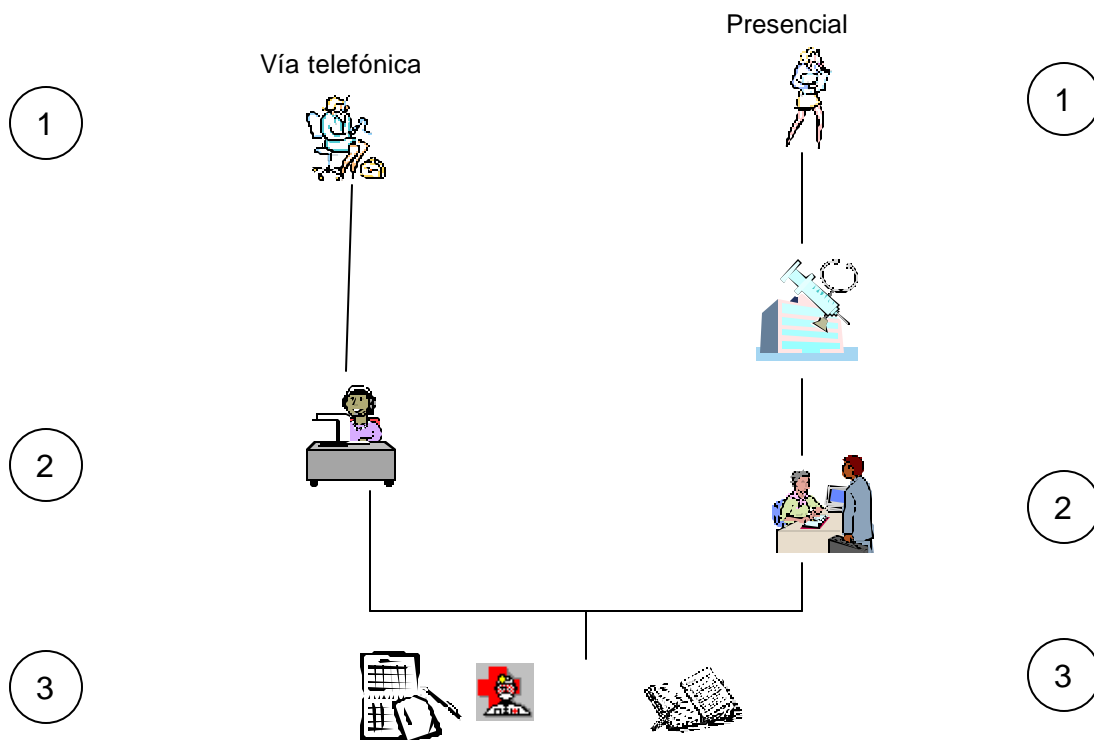


Figura 3.4.1 Proceso de citas.

Para nuestro objetivo centraremos el sistema de información en el nivel 3 que además es común a los dos tipos de proceso para registrar una cita, ya sea vía telefónica o presencial.

Se hará uso de una base de datos relacional para almacenar la información de la clínica en un solo lugar, en donde todos los involucrados tendrán acceso (médicos, enfermeras, personal administrativo).

El sistema tratará de cubrir todas las necesidades de la clínica, hablando propiamente de las citas, se desarrollará un módulo para este tópico que permitirá solventar los problemas expuestos en el apartado 3.1. Vea tabla 3.4.1

Problema	Solución
El registro de los datos de las citas no tiene una estructura uniforme lo que dificulta su interpretación y la localización de un dato en específico.	Los registros se mostraran en un formato sencillo y claro para todos los usuarios siguiendo un orden lógico. Esto mejora el proceso logrando que la captura de datos sea rápida y eficiente, al igual que la consulta de los mismos.
La caligrafía del personal de recepción puede ser ilegible para los otros miembros del consultorio.	Al hablar de un sistema de cómputo se soluciona el problema de caligrafía, los datos serán legibles para todos.
La información sobre el paciente puede ser registrada en diversos medios, por ejemplo, un cuaderno, post-it (Nota), un pedazo de papel, etc. Esto implica que se pueda perder la información, o bien, se invierta tiempo en registrarla en el medio correcto.	Usar una base de datos resuelve el problema de datos dispersos o que se pierda información. No habrá necesidad de que los datos tengan que rescribirse a las agendas. La información estará segura en un medio electrónico que permitirá respaldar los datos cuando el usuario (personal de la clínica) lo indique.

Problema	Solución
El personal de recepción se enfrenta con un serio problema al tener que modificar el registro de citas a causa de una eventualidad que tenga que ser atendida por el médico (emergencias de cirugías, problemas personales, salud, etc.).	De manera sencilla se podrán cancelar las citas que por alguna circunstancia no puedan ser atendidas en la fecha que estaban programadas, esto no causará conflicto alguno al resto de los datos ni a las demás citas.
Por el hecho de que el registro de los datos se realiza en un cuaderno u otro medio de papel semejante, obliga al paciente a repetir los mismos datos cada vez que solicita una cita.	Una vez que un paciente sea dado de alta en el sistema, no se tendrá que volver a cargar sus datos personales, es decir se eliminara la duplicidad y redundancia en la información. Con esto también se mejoran los tiempos de atención al paciente.
Para generación de información histórica, el personal de recepción debe hacer una búsqueda minuciosa en el cuaderno que utiliza para registro de datos, lo que implica que tome más tiempo en el proceso y se vuelva menos eficiente.	Las consultas al sistema para registrar una cita serán inmediatas, el tiempo que se invertirá en hacer la consulta a la base de datos será mínimo, además que los resultados siempre contendrán la información precisa.
No es posible generar estadísticas acerca de los pacientes debido a la inconsistencia de los datos.	En caso de que el personal requiera información de tipo estadístico sobre su personal o sobre sus pacientes se puede realizar, aunque por el momento este problema pasa a segundo plano.

Tabla 3.4.1 Principales Problemas del Proceso de Citas y Solución

Resumiendo, implantar una base de datos es la alternativa para ofrecer un sistema de información confiable, que permita dar seguimiento al manejo del proceso de registrar una cita.

Los beneficios que se tendrán en este proceso son:

- Se eliminan los registros a mano y en papel que pueden no ser confiables del todo, además que el realizarlos de manera física implica un gasto monetario y en tiempo que la clínica podrá ahorrar.
- El procedimiento para asignar una cita a un paciente será eficiente ya que:
 - Los datos de los pacientes se capturaran una sola vez, esto evitara que el registro de un paciente sea repetitivo, ahorrando tiempo.
 - Se evitaran duplicidades en el registro de los pacientes.
 - Se integraran los horarios de servicio de los médicos para poder asignarles citas.
 - Cada médico definirá la duración de sus consultas a fin de brindar la atención correcta a cada paciente y para organizar sus citas a fin de evitar empalmes entre las personas que requieran de atención médica.
 - Se tendrá la agenda de citas de los doctores actualizada conforme se registre una cita o se cancele.
 - Las opciones para ofrecer una cita al paciente se visualizaran de manera rápida, con lo que se hará esperar menos al individuo.
 - De manera automática se mostrarán las citas que se tengan registrada para la fecha actual.
 - Se permitirá hacer una consulta detallada de las citas que tenga programadas algún medico en particular.
 - Las consultas de las citas, así como los procesos para ingresar un paciente, dar de alta una cita o cancelarla, serán sencillos y transparentes para el usuario del sistema, agilizando los tiempos de espera del paciente.

- La información estará concentrada en un solo lugar y podrá ser consultada en cualquier momento tanto por la secretaria, como por los médicos.
- Se evitara la perdida de información
- Al tener la información correcta en menos tiempo, se ofrece al paciente un servicio de calidad que hará que el cliente de la clínica se sienta conforme con la atención recibida, ya que sin duda la atención del personal de recepción hacia los pacientes también cambiará al sentirse seguros de que la información que están proporcionando no ocasionará disgustos a los pacientes de la clínica.

Analizaremos los problemas del proceso dos, la consulta. De acuerdo al diagrama 3.1.2 del apartado 3.1, donde se describen 5 niveles de situaciones en las que se presentan problemas durante la consulta; nos enfocaremos a las problemáticas de los niveles 2, 3 y 4; esto debido a que las situaciones descritas en el nivel 1 se resuelven mejorando el proceso del registro de una cita, el cual ya se menciono, el nivel 5 hace referencia a si el paciente tiene que volver a sacar una cita o bien se puede ir a casa, en el caso de que deba registrar una nueva cita ya se presentaron tanto los problemas como las soluciones.

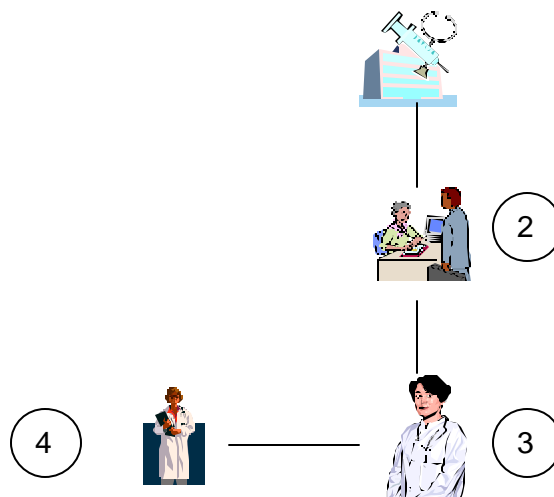


Diagrama 3.4.2 Proceso de Consultas

El sistema mantendrá relacionados todos los informes que se realicen dentro de una consulta, en este caso hablaremos del modulo paciente que contendrá todos los registros que forman parte del historial clínico y que son requeridos en las consultas, formando el expediente del paciente de manera electrónica.

Abordaremos las situaciones críticas de los procesos de consulta de tal manera que tenemos lo siguiente vea tabla 3.4.2

Problema	Solución
Se emplea tiempo considerable en buscar y reunir los expedientes físicos de todos los pacientes que tendrán consulta.	Al tener la información en una base de datos ya no será necesario hacer la búsqueda de los expedientes físicos, ya que el doctor tendrá disponibles los historiales clínicos de sus pacientes, con ingresar algún dato del paciente, como puede ser el nombre, apellido, número de expediente, etc.
Para el caso de usuarios “nuevos”, el personal de recepción invierte tiempo en crear expediente del usuario y solicitar al usuario datos generales.	Dado que los formatos que forman el expediente se crean de manera automática, el personal ya no invierte tiempo en hacer un expediente con lo necesario para que el médico atienda a la persona, al ingresar los datos generales una sola vez, estos se verán reflejados al momento que el doctor comience su consulta, por lo que no tendrá que ingresar nuevamente esos datos.

Problema	Solución
El personal de recepción emplea un tiempo considerable en archivar los expedientes de los usuarios.	Ya no será necesario archivar los expedientes, el médico será el que guarde los cambios en los registros de sus pacientes, que automáticamente serán almacenados en la base de datos.
Las mediciones de los signos vitales se escriben con puño y letra del personal de enfermería y esto ocasiona dificultad al interpretar los datos por parte de los otros integrantes del consultorio.	Al igual que en el registro de las citas el problema de la ilegibilidad de los datos, en todos los registros (historial médico, nota medica, recetas, etc.) se elimina, con el uso de un sistema de cómputo.
El personal de enfermería se ve obligado a anexar cada uno de los formatos que componen un expediente clínico cuando el paciente es nuevo, o bien, por extravío del historial.	Los formatos que se necesitan para la consulta se generarán en automático cada vez que el medico requiera uno, el historial tendrá la secuencia de las consultas, evitando se traspapelen o pierdan los registros del expediente.
No existe respaldo de los expedientes físicos.	El sistema podrá respaldarse cuando el personal de la clínica lo establezca, como se menciona en el proceso de citas.
Cada vez que surge un extravío o se traspapela un expediente, el médico dedica tiempo en recolectar nuevamente los datos necesarios.	No habrá perdida de información, cualquier información que requiera el medico acerca de un paciente estará disponible.

Problema	Solución
El médico invierte tiempo considerable en la revisión del expediente del paciente debido a la forma en que esté conformado u ordenado, al número de hojas que lo conformen, la cantidad de estudios clínicos y de gabinete, etc.	El médico podrá seleccionar cuál es el informe que necesita de sus pacientes, es decir, si necesita consultar la nota medica de la última consulta, historial clínico, recetas, estudios, etc.
Por cada especialista del consultorio clínico que consulte el usuario, se le genera un expediente en particular provocando duplicidad.	El expediente al encontrarse en una base de datos, podrá ser consultado por los médicos de la clínica, evitando duplicidades.

Tabla 3.4.2 Principales Problemas del Proceso de Consultas y Solución

Los beneficios que se tendrán en el proceso de consultas serán:

- Se evitará el gasto en los formatos en papel además del almacenamiento de los mismos.
- No habrá duplicidades en los expedientes de los pacientes, el expediente estará disponible para los médicos de la institución, permitiendo tener conocimiento de los padecimientos de un paciente, para mejorar la calidad en la atención que se le brinde.
- Todas las notas médicas, recetas, hojas de referencia, serán registradas por lo que se tendrá control sobre el padecimiento y tratamiento de los pacientes, así mismo, se puede realizar la impresión de estos informes para que el paciente tenga una copia en caso de que la requiera.
- El médico podrá consultar el expediente y las notas médicas del paciente de acuerdo a la necesidad que se presente durante la consulta.
- La información estará segura ya que no habrá pérdida de la misma además se pueden establecer fechas para realizar un respaldo de la misma.

El tener integrados los registros de las consultas, estos se verán reflejados en un ahorro de tiempo, y la mejora en la atención hacia el paciente, ya que no se le preguntara información que el haya proporcionado en consultas anteriores, se estarán evitando búsquedas innecesarias que puedan provocar que el paciente se desespere o se sienta a disgusto por la pérdida de su expediente.

Si se presentaran otros problemas adicionales a los antes descritos, en cualquiera de los dos procesos (citas y consulta) el sistema puede mejorarse, ya que no limita las funciones y permitirá adaptarse a las necesidades del momento.

3.5 UBICACIÓN DE LAS OPCIONES DE SOLUCIÓN

Para la construcción de un sistema de información la elección de herramientas es muy importante, en la actualidad existen muchas opciones que facilitan esta labor. Para llevar a cabo una buena elección se tienen que tomar en cuenta varios aspectos tales como el tamaño de la aplicación, el costo-beneficio, disponibilidad de equipo, tiempo de entrega del sistema y factores imprevistos, considerando estos factores se analizan algunas opciones de solución, para posteriormente seleccionar la que mejor se ajuste a los requerimientos del sistema que se desea desarrollar.

Opciones de Solución para la Elección del Back-End

Postgresql

Es un sistema de base de datos objeto-relacional que tiene las características de sistemas de bases de datos comerciales tradicionales. La última versión de PostgreSQL 8.1 y sus anteriores versiones, están distribuidas bajo la clásica licencia BSD (Sistema de Administración de Base de Datos PostgreSQL). No tiene restricciones acerca de la manera en que puede usarse el código fuente, es libre. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. Ingres fue más tarde desarrollado comercialmente por la Relational Technologies/Ingres Corporation. En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. La versión 1 fue liberada a un pequeño grupo de usuarios en junio de 1989, seguido por la versión 2 con un sistema de reglas reescrito en junio de 1990. Para la versión 3, liberada en 1991, el sistema de reglas fue reescrito nuevamente, pero también agregó soporte para múltiples administradores de almacenamiento y un sistema de consultas mejorado. Hacia 1993, Postgres había crecido inmensamente en

popularidad y poseía una demanda asfixiante de nuevas funcionalidades. Tras liberar la versión 4, la cual era una simple versión de limpieza, el proyecto fue abandonado.

En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periodo como Postgres95. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto y no es controlado por ninguna compañía.

Características

Algunas de sus principales características son:

PostgreSQL corre nativamente sobre sistemas operativos basados en Microsoft Windows NT tales como Win2000, WinXP y Win2003, cualquier plataforma moderna compatible con Unix debería ser capaz de ejecutar PostgreSQL y existe incluso un port para Novell Netware 6. Ofrece la integración con interfaces para lenguajes como Perl, TCL, Python, etc, Por omisión, PostgreSQL sólo permite conexiones desde el equipo local utilizando sockets de dominio Unix o conexiones TCP/IP. Cuenta con una herramienta llamada PL/pgSQL para gestionar las bases de datos.

Funciones

Las funciones permiten subir bloques de código que se ejecuten en el servidor. Estas funciones pueden escribirse en una variedad de lenguajes, algunos de los más importantes son PL/pgSQL, C, C++ y Java. Puede definirse si las funciones serán ejecutadas con los permisos del llamador o del usuario que definió la función.

Alta concurrencia

Mediante un sistema denominado MVCC (Acceso concurrente multiversión) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

Amplia variedad de tipos nativos

PostgreSQL provee nativamente soporte para:

Números de precisión arbitraria.

- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas).
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.
- Arreglos.

Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GIST de PostgreSQL. Algunos ejemplos son los tipos de datos GIS creados por el proyecto PostGIS.

Otras características:

- Claves ajenas también denominadas Llaves ajenas o Llaves Foráneas (*foreign keys*).
- Disparadores (*triggers*).
- Vistas.
- Integridad transaccional.
- Herencia de tablas.
- Tipos de datos y operaciones geométricas.

Firebird

Firebird es un sistema de administración de base de datos de código abierto, basado en la versión 6 de Interbase, cuyo código fue liberado por Borland. Su código fue reescrito de C a C++.

Los objetivos de la Fundación FirebirdSQL son:

- Apoyar y lograr el avance del manejador de base de datos relacional Firebird
- Proveer los mecanismos e infraestructura no comerciales para aceptar y administrar los fondos recaudados, e invertir tales fondos para promover el esfuerzo del desarrollo de esta base de datos
- Fomentar la cooperación y la afiliación de individuos, organizaciones sin fines de lucro y compañías comerciales involucradas o que estén planeando estar involucradas en el desarrollo, apoyo y promoción de los proyectos de software de Firebird y sus productos y actividades asociadas.)

Características

- Es multiplataforma, y actualmente puede ejecutarse en los sistemas operativos: Linux, HP-UX, FreeBSD, Mac OS , Solaris y Microsoft Windows .
- Ejecutable pequeño, con requerimientos de hardware bajos.
- Arquitectura Cliente/Servidor sobre protocolo TCP/IP y otros (embedded).
- Soporte de transacciones y claves foráneas.
- Es medianamente escalable .
- Buena seguridad basada en usuarios/roles.
- Diferentes arquitecturas, entre ellas el Firebird empotrado (*embedded server*) que permite ejecutar aplicaciones monousuario en ordenadores sin instalar el software Firebird.
- Bases de datos de sólo lectura, para aplicaciones que corran desde dispositivos sin capacidad de escritura, como cd-rooms.
- Existencia de controladores ODBC, OLEDB y JDBC
- Controlador nativo para PHP
- Requisitos de administración bajos, siendo considerada como una base de datos libre de mantenimiento, al margen de la realización de copias de seguridad y restauraciones periódicas.
- Pleno soporte del estándar SQL-92, tanto de sintaxis como de tipos de datos.
- Completo lenguaje para la escritura de disparadores (*triggers*) y procedimientos almacenados.
- Capacidad de almacenar elementos BLOB (*binary large objects*).

Tipos de servidor

Existen dos tipos de servidor Firebird para ser instalados: **Classic** y **Super server**. Si bien tienen varias diferencias menores entre si, la principal consiste en que el super server maneja hilos de ejecución individuales para cada conexión. Por lo

tanto para un número reducido de conexiones el recomendado sería el classic por que consumirá menor cantidad de recursos.

En caso de arquitecturas SMP, se debe utilizar el servidor classic porque el Supersever no tiene soporte para este tipo de arquitectura.

Los propios desarrolladores de Firebird recomiendan lo siguiente a la hora de decidirse por uno de estos servidores:

- En plataformas Windows seleccionar el Superserver.
- En Linux simplemente elegir cualquiera, según las conexiones estimadas. En la mayoría de las situaciones no se notará diferencias en el performance.

Podría considerarse un tercer tipo, el **Embebed**. Este consiste en una única DLL (de unos 2 Mb de tamaño) que contiene a todo el servidor. De esta forma se puede tener un DBMS completo disponible y distribuible junto con aplicaciones de usuario.

Características que ubican las funciones relevantes que debe tener un manejador de bases de datos para la implementación del Back-End.

Es complicado realizar un estudio del performance de las bases de datos, ya que depende del volumen de información que se va a manejar, seguridad, escalabilidad, etc. Se trata de ubicar los manejadores de bases datos mas utilizados y propuestos para su implementación.

CARACTERISTICAS	MYSQL 5.0	POSTGRESQL 8.1	FIREBIRD 2.0
Compatibilidad con Sistemas Operativos	↑	↑	↑
Integración con API	↑	↔	↔
Conectores (ODBC y JDBC)	↑	↑	↑
Características Fundamentales " ACID "	↑	↑	↑
Seguridad (Encriptación de Datos, Backups, Acceso a Usuarios)	↑	↔	↑
Acceso a Datos	↔	↔	↔

CARACTERISTICAS	MYSQL 5.0	POSTGRESQL 8.1	FIREBIRD 2.0
Tipo de Datos	↑	↑	↑
Soporte Técnico	↑	↓	↓
Rapidez	↑	↔	↔
Optimización de los Recursos	↑	↔	↔
Herramientas de Administración	↑	↔	↔
Costo y Licenciamiento	↑	↑	↑

Nota: La calificación es asignada, de acuerdo las características especificadas en la tabla, y se usan los siguientes símbolos para representar ese valor:

Excelente = ↑ Bueno = ↔ Regular = ↓

Opciones de Solución para la elección del Front-End

PowerBuilder

Se trata de una herramienta de desarrollo orientada a aplicaciones de gestión de bases de datos, multiplataforma, cliente/servidor, orientado a objetos y capaz de acceder homogéneamente a cualquier base de datos que soporte SQL (hoy en día el 99%). PowerBuilder, comenzó como un producto revolucionario de PowerSoft, ya que era una utilidad para crear pseudo-aplicaciones, que permitía ventanas, menús y poco más. No soportaba ni herencia, ni polimorfismo, y a duras penas hacía bien lo que se suponía que hacía. su especialización en el campo del desarrollo de bases de datos SQL, (su perfecta adaptación con Oracle y el éxito de uno de sus objetos: la DataWindow (por ahora, para hacernos una idea los formularios de Access se basan en este objeto).

No fue hasta la versión 4 (aún para Windows 16 bits) cuando se consolidó como una buena herramienta de programación, corrigiendo todos los errores y soportando (casi) completamente la orientación a objetos. A partir de la versión 5 (la primera que compilaba en 32 bits), comenzaron a desarrollar DPB (Distributed PowerBuilder), un sistema de programación distribuida para aplicaciones multi-capa.

También a partir de esta versión, se incluyó un conjunto de librerías base (o FrameWork) denominado PFC, es decir: PowerBuilder Foundation Classes. Aunque estas librerías no son imprescindibles para desarrollar aplicaciones, su utilización está muy extendida, ya que nos ofrece un repositorio de objetos muy completo que nos ahorrará muchas horas de trabajo, además de una filosofía muy ingeniosa de diseño.

A la vez que salía al mercado la versión 5, Sybase realizó la compra de PowerSoft, absorbiendo así la herramienta PowerBuilder, la base de datos SQL

Anyware y el compilador Watcom C++, entre otros productos. En la versión 6 se consolidó la arquitectura del DPB, hasta el punto que la programación distribuida es una de las características por las que PowerBuilder es conocido y apreciado.

También se comenzó a orientar la programación de internet y fue en la versión 7 cuando salió la HTML-Datawindow, un objeto capaz de generar páginas HTML a partir de consultas SQL en cualquier formato, además de importantes cambios en el entorno de desarrollo.

Ahora mismo ya está la versión 8 en el mercado, que ha aportado cambios menores (gestión de excepciones, cambios en el IDE, etc.), y ya están en la fase final de desarrollo la versión 9 (que promete soportar código C++ para creación de componentes).

Delphi

El nombre **Delphi**, hace referencia al oráculo de (**Delfos**). Borland eligió ese nombre para resaltar su principal mejora con respecto a su antecesor (Turbo Pascal), que sería su conectividad con bases de datos Oracle (oráculo, en inglés). El principal uso de Delphi, es para el desarrollo de aplicaciones visuales y de bases de datos cliente-servidor y multicapas. Debido a que es una herramienta de propósito múltiple, se usa también para proyectos de casi cualquier tipo, incluyendo aplicaciones de consola, CGI y servicios del sistema operativo.

Delphi inicialmente sólo producía ejecutables binarios para Windows (Win16 y Win32). En la última versión Delphi 2006 incluye en el mismo entorno de desarrollo (Delphi para Win32, Delphi para .NET, y C# para .NET.), Existe una versión de Delphi para sistemas Unix y Linux, denominada Kylix (de la cual existe un versión gratuita, aunque limitada). Sin embargo Kylix fue abandonada por Borland en su versión 3.0.

Delphi está basado en una versión moderna de Pascal, denominada Object Pascal, que expande las funcionalidades del primero:

- Soporte para la programación orientada a objetos (habitualmente llamada POO) también existente desde Turbo Pascal 5, pero mucho más evolucionada, en cuanto a Objetos en sí: clases, instancias. encapsulación declarando partes privadas, protegidas, públicas y publicadas de las clases, propiedades concepto nuevo que luego han adaptado muchos otros lenguajes. herencia simple y polimorfismo.
- Soporte para excepciones, mejorando sensiblemente el manejo de errores de usuario y del sistema.
- Programación manejada por eventos (**event-driven**), posible gracias a las ya mencionadas propiedades de los objetos.

A partir de la versión 7, se puede hablar de Delphi como un lenguaje.

Delphi introdujo la idea del uso de componentes, que son piezas reutilizables de código (clases) que pueden interactuar con el EID en tiempo de diseño y desempeñar una función específica en tiempo de ejecución. Desde un enfoque más técnico, se catalogan como componentes todos aquellos objetos que heredan desde la clase TComponent. Una gran parte de los componentes disponibles para Delphi son *controles* (derivados de TControl), que encapsulan los elementos de interacción con el usuario (botones, menús, barras de desplazamiento, etc.).

Delphi incluye una biblioteca de clases bien diseñada denominada VCL (*Visual Component Library*, Biblioteca de Componentes Visuales) y, en sus versiones 6 y 7, una jerarquía multiplataforma paralela denominada CLX. Ésta también se incluye en Kylix. Estas jerarquías de objetos incluyen componentes visuales y no visuales, tales como los pertenecientes a la categoría de acceso a datos, con los que puede establecerse conexiones de forma nativa o mediante capas intermedias (como ADO, BDE u ODBC) a la mayoría de las bases de datos relacionales

existentes en el mercado. La VCL también está disponible para el desarrollo en .NET.

Además de poder utilizar en un programa estos componentes estándar (botones, cuadrículas, conjuntos de datos, etc.), es posible crear nuevos componentes o mejorar los ya existentes, extendiendo la funcionalidad de la herramienta. En Internet existe un gran número de componentes, tanto gratuitos como comerciales, disponibles para los proyectos a los que no les basten los que vienen ya con la herramienta.

Tabla comparativa del Front-End, enfocado al soporte de ambas herramientas en distintas características y tecnologías:

Características	Delphi	PowerBuilder	VisualBasic
Orientación a objetos	Si	Si	Si
Arquitectura basada en servicios	No	Si	No
Gestión de excepciones	Si	Si	Si
Punteros	Si	No	No
Referencias	Si	Si	Si
Acceso a DLLs externas	Si	Si (sólo estático)	Si (estático)
Código nativo compilado	Si	No	Si
Código interpretado - máquina virtual	No	Si	Si
Código SQL en línea (SQL incrustado)	No	Si	Si
Acceso homogéneo a distintas bases de datos	Si	Si	Si
Multiplataforma	No	Si	No
Cliente/Servidor	Si	Si	Si
Programación distribuída	Si	Si	Si
Programación de librerías DLL	Si	No	No

Características	Delphi	PowerBuilder	VisualBasic
IDE programable (expertos, editores de propiedades, etc.)	Si	No	Si
Debugger gráfico	Si	Si	Si
Librería de componentes (nº de componentes)	Si	Si	Si
Librería de clases o Framework	Si	Si	Si
Herramientas de generación de SQL	Si	Si	Si
Programación CGI/ISAPI	Si	No	Si
<i>Applets</i> del Panel de Control	Si	No	Si
Servidores Web	Si	No	Si
Servicios NT	Si	No	Si

Nota: El significado de Si, quiere decir que cuenta con esa característica y No es lo contrario.

La selección del manejador de bases de datos y el lenguaje de programación para el desarrollo de un sistema depende en gran medida de la información a manejar.

De acuerdo al análisis del manejo de información que se tiene en un consultorio o clínica médica y de acuerdo a las características antes mencionadas en las tablas de características de los manejadores de bases de datos y de los lenguajes de programación, se optó por MySQL para el manejo del back-end y Visual Basic 6.0 para la construcción del front-end del sistema.

CAPÍTULO IV

DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

4.1 APLICACIÓN DE LA METODOLOGÍA ELEGIDA

Una metodología presenta el camino para desarrollar software de una manera sistemática.

Detallan y concretan las ideas generales de los modelos de ciclo de vida en que se basen, puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo.

La metodología indica cómo hay que obtener los distintos productos parciales y finales. Por lo que se puede definir como un conjunto de pasos y procedimientos que deben seguirse para el desarrollo de software.

Características Deseables de una Metodología

- Existencia de reglas predefinidas.
- Cobertura total del ciclo de desarrollo.
- Verificaciones intermedias.
- Planificación y control.
- Comunicación efectiva.
- Utilización sobre un abanico amplio de proyectos.
- Fácil formación.
- Herramientas CASE.
- Actividades que mejoren el proceso de desarrollo.
- Soporte al mantenimiento.
- Soporte a la reutilización del software.

Clasificación de las Metodologías

- Estructuradas.
 - Orientadas a procesos.

- Orientadas a datos: jerárquicas, no jerárquicas.
- Mixtas.
- Orientadas a objetos.
- Para sistemas de tiempo real.

Metodología Yourdon/De Marco

Para facilitar el uso de sistemas informáticos complejos en la ejecución de proyectos, Ed Yourdon ideó una metodología de análisis conocida como análisis estructurado cuya característica fundamental consiste en que el objeto principal del análisis es la información que se intercambia entre procesos, a diferencia del enfoque tradicional que se centraba en los procesos propiamente dichos. Posteriormente, Tom De Marco desarrolló y aplicó con excelentes resultados esta metodología.

Desde el punto de vista de su aplicación, las tareas básicas del Análisis Estructurado son las siguientes:

- Estudio del entorno físico existente. Esta tarea debe enfocarse desde el punto de vista de los datos o información que fluye entre procesos, más que desde el punto de vista de las personas ejecutoras. El resultado final es un modelo lógico sin relaciones físicas concretas.
- Desarrollo del nuevo modelo lógico propuesto, por medio de los diagramas de flujo de datos (DFD) correspondientes, el diccionario de datos y las descripciones de procesos.
- Cuando la complejidad de la actividad lo hace necesario, se desarrollan modelos lógicos alternativos, evaluándolos mediante los adecuados estudios comparativos y seleccionando la solución óptima.
- Elaboración de los documentos finales de los resultados.

Los elementos clave del análisis son los DFD's que se elaboran con los siguientes elementos básicos:

- Vectores: representan los flujos de información. La mayor parte de los datos se mueven entre procesos, pero también pueden hacerlo desde o hacia los archivos y desde o hacia personas o entidades. En los DFD's los vectores se identifican con la denominación de la información que se mueve por ellos.
- Burbujas: representan los procesos que consisten en la transformación de información. En los diagramas se asigna un índice a cada burbuja.
- Líneas rectas: representan los archivos o bases de datos. Los archivos son almacenes temporales de datos. Cualquier elemento que sirva para almacenar información se considera un archivo y en los DFD's se representa por una simple línea recta con su identificación incorporada.
- Cajas: representan emisores o receptores de información, normalmente personas o entidades, pertenecientes al ámbito del sistema. Suelen representar elementos del entorno del problema principal. Se incluyen únicamente como referencia del contexto físico exterior.

La Figura 4.1.1 es una muestra de un diagrama de flujo de datos que incluye los cuatro elementos. Los datos o la información X procedentes de la persona o entidad A se utilizan en el proceso B complementados con los datos Y procedentes del archivo o base de datos C dando como resultado la información Z.

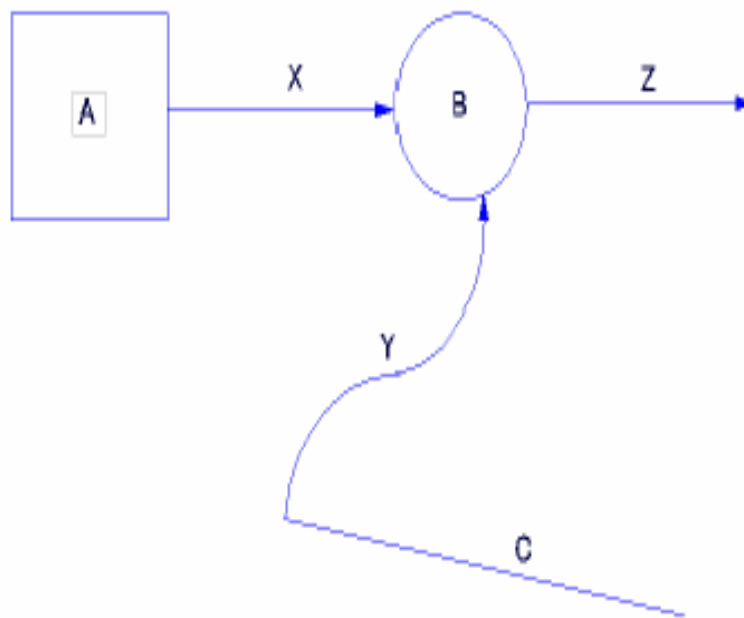


FIGURA 4.1.1 Elementos básicos de los diagramas de flujos de datos.

Metodología Merise

Esta metodología aporta un ciclo de vida más largo a los existentes hasta entonces que se materializa en un conjunto definido de etapas. Introducen dos ciclos complementarios: ciclo de abstracción y ciclo de decisión. El ciclo de abstracción se basa en la percepción de tres niveles de abstracción: conceptual, organizativo y físico. Además se definen dos niveles para cada nivel: un modelo de datos y otro de tratamientos.

Las fases de la metodología Merise son:

- Estudio preliminar.
 - Analizar la situación existente para proponer solución.
- Estudio detallado.
 - Definir a nivel funcional la solución.

- Implementación.
 - Estudio técnico: distribuir los datos en ficheros y los tratamientos en módulos de programas.
 - Producción de programas: codificar y pruebas
- Realización y puesta en marcha.
 - Instalación, formación del personal, etc.

Sus principales aportaciones son:

- Ciclo de vida más largo, con etapa de planificación
- Ciclo de abstracción a 3 niveles:
 - Nivel conceptual: Datos Modelo Conceptual de Datos y Procesos.
 - Modelo Conceptual de Tratamientos.
 - Nivel organizativo: Datos Modelo Lógico de Datos y Procesos Modelo Organizativo de Tratamientos.
 - Nivel físico: Datos Modelo Físico de Datos y Procesos Modelo Operativo de Tratamientos.

Metodología SSADM

El gobierno británico planteó la necesidad de crear una metodología y se desarrolló entre el Central Computing and Telecommunications Agency (CCTA) y Learmonth and Burchett Management Systems (LBMS), dando como resultado la metodología SSADM (Structures Systems Analysis and Design Method).

Los aspectos claves de SSADM son:

- Énfasis en los usuarios: sus requisitos y participación.
- Definición del proceso de producción: qué hacer, cuándo y cómo.
- Tres puntos de vista: datos, eventos, procesos.
- Máxima flexibilidad en herramientas y técnicas de implementación.

SSADM proporciona un conjunto de procedimientos para llevar a cabo el análisis y diseño, pero no cubre aspectos como la planificación estratégica ni entra en la construcción del código.

La figura 4.1.2 muestra esquemáticamente la metodología SSADM.

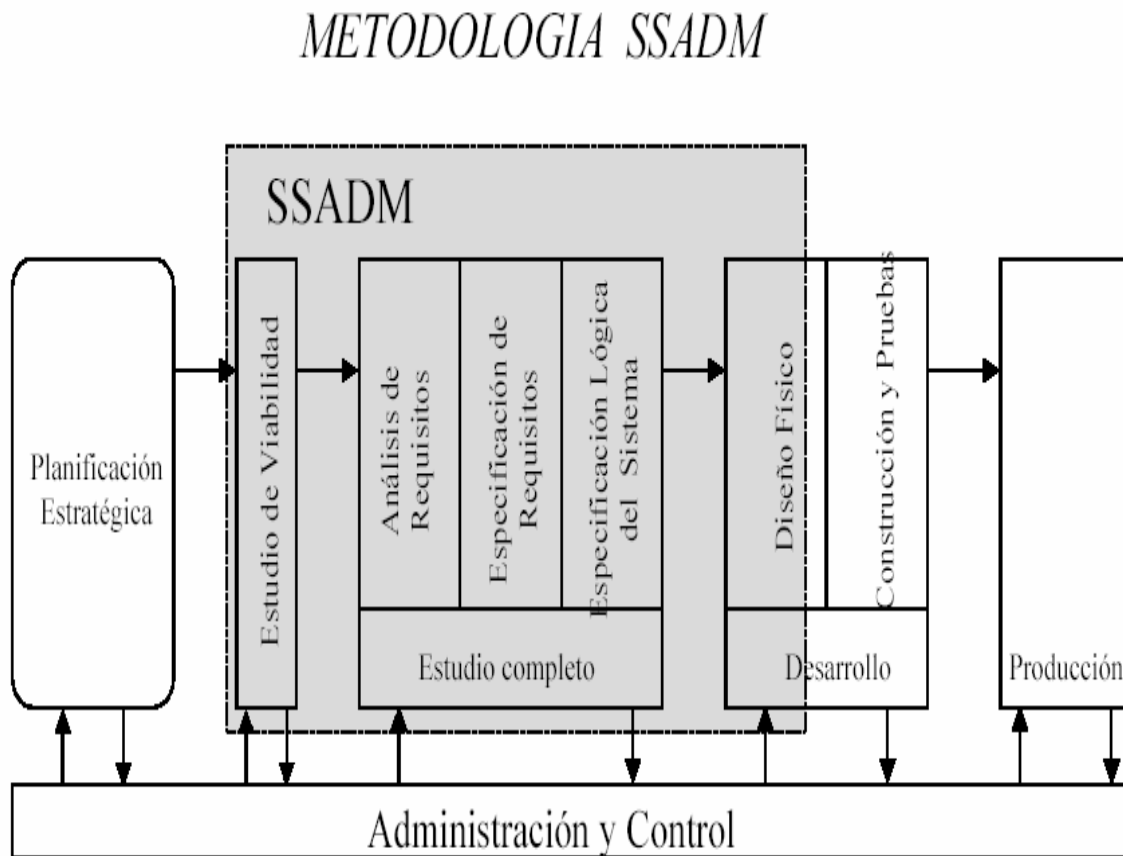


FIGURA 4.1.2 Metodología SSADM.

De las tres metodologías expuestas anteriormente se eligió la metodología Yourdon/De Marco debido a que se trata de un enfoque orientado al flujo de datos a través de los diagramas de datos y procesos, lo cual nos permite realizar un mejor análisis complementado con el uso del diccionario de datos y el diagrama de entidad-relación, todo esto nos proporciona herramientas para realizar un mejor diseño de la estructura de la programación, además de facilitar correcciones en el flujo de la información.

4.1.1 DIAGRAMA DE CONTEXTO

El diagrama de nivel 0 o diagrama de contexto representa el sistema en un solo proceso, que identifica cual es la función principal del sistema, mostrando además, los flujos de información que lo relacionan con otros sistemas: las entidades externas. El diagrama de contexto tiene una gran importancia puesto que resume el requisito principal del sistema de recibir ciertas entradas, procesarlas de acuerdo con determinada función y generar ciertas salidas. A partir del diagrama de contexto se puede ir construyendo nuevos diagramas que van definiendo con mayor nivel de detalle los flujos de datos y procesos de transformación que ocurren en el sistema, de forma que al final se obtiene una jerarquía de diagramas.

El diagrama de contexto del Sistema de Administración Médica se muestra en la figura 4.1.1.1.

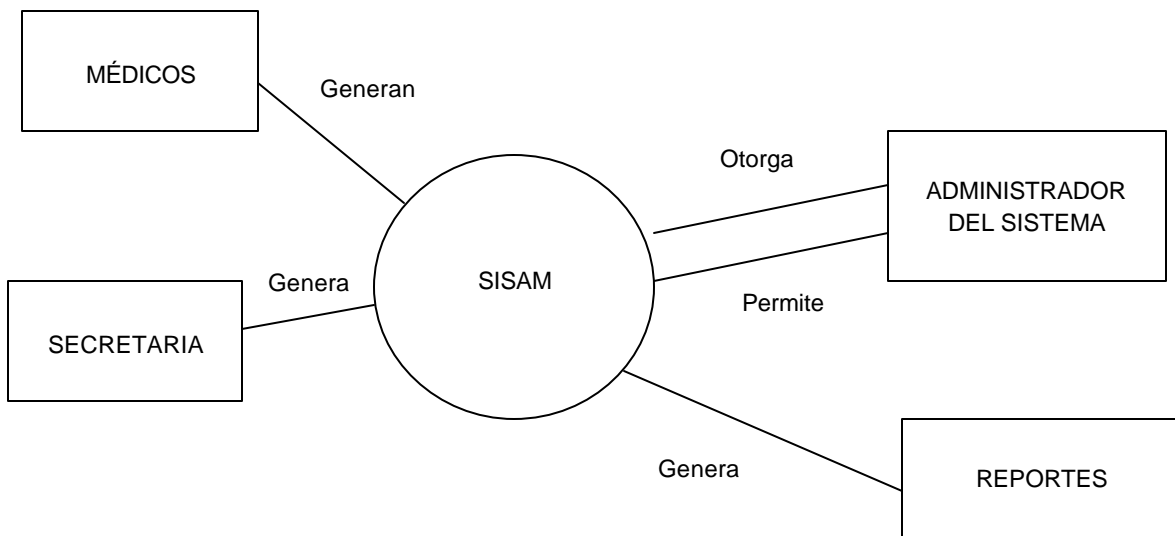


Figura 4.1.1.1 Diagrama de contexto.

4.1.2 DIAGRAMA DE FLUJO DE DATOS (DFD) Y DIAGRAMA DE PROCESOS DE DATOS (DPD)

El diagrama de flujo es un esquema para representar gráficamente un algoritmo. Se basan en la utilización de diversos símbolos para representar operaciones específicas. Se les llama diagramas de flujo porque los símbolos utilizados se conectan por medio de flechas para indicar la secuencia de operación.

Por otra parte el diagrama de proceso de datos es una representación gráfica que permite al analista definir entradas, procedimientos y salidas de la información en la organización permitiendo de ésta manera comprender los procedimientos existentes con la finalidad de optimizarlos reflejándolos en el sistema propuesto.

Por otra parte el diagrama de proceso de datos representa el flujo de información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida del sistema. Un DPD proporciona un mecanismo para el modelado funcional así como el modelado del flujo de información.

El diagrama 4.1.2.1 muestra una estructura general del sistema, después se presentan los diagramas de flujo de datos y de procesos de cada unos de los procesos principales del SISAM. Ver diagramas 4.1.2.1 a 4.1.2.12.

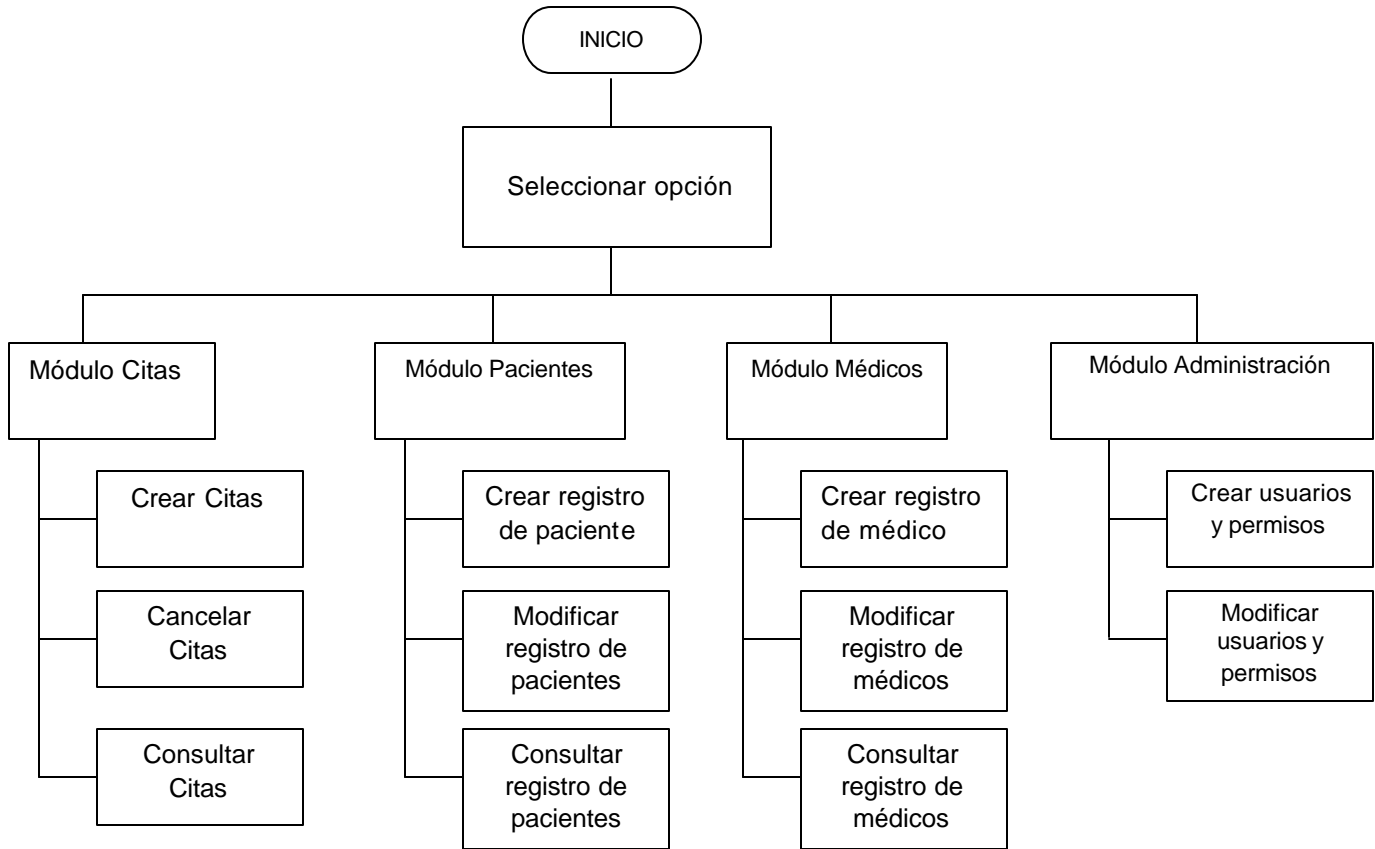


Figura 4.1.2.1 Diagrama de bloques general del sistema.

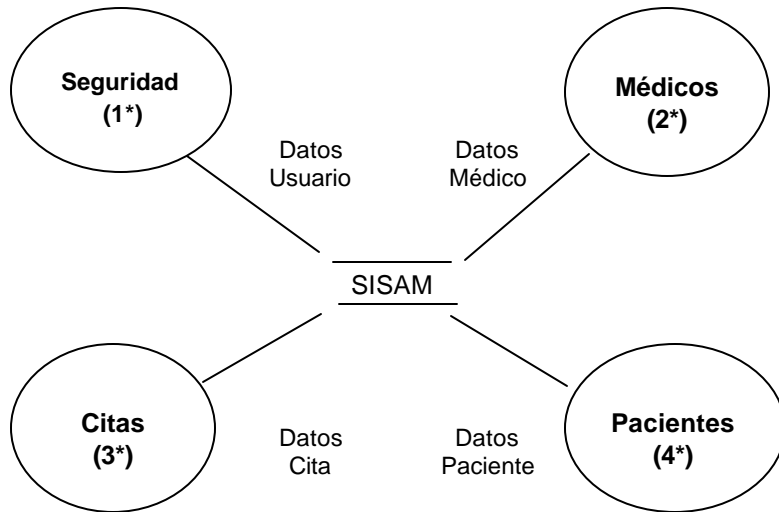


Figura 4.1.2.2 Diagrama de Proceso de Datos General del sistema.

OBSERVACIONES:

1* Ver diagrama de Proceso 4.1.2.4 y 4.1.2.6

2* Ver diagrama de Proceso 4.1.2.8

3* Ver diagrama de Proceso 4.1.2.10

4* Ver diagrama de Proceso 4.1.2.1

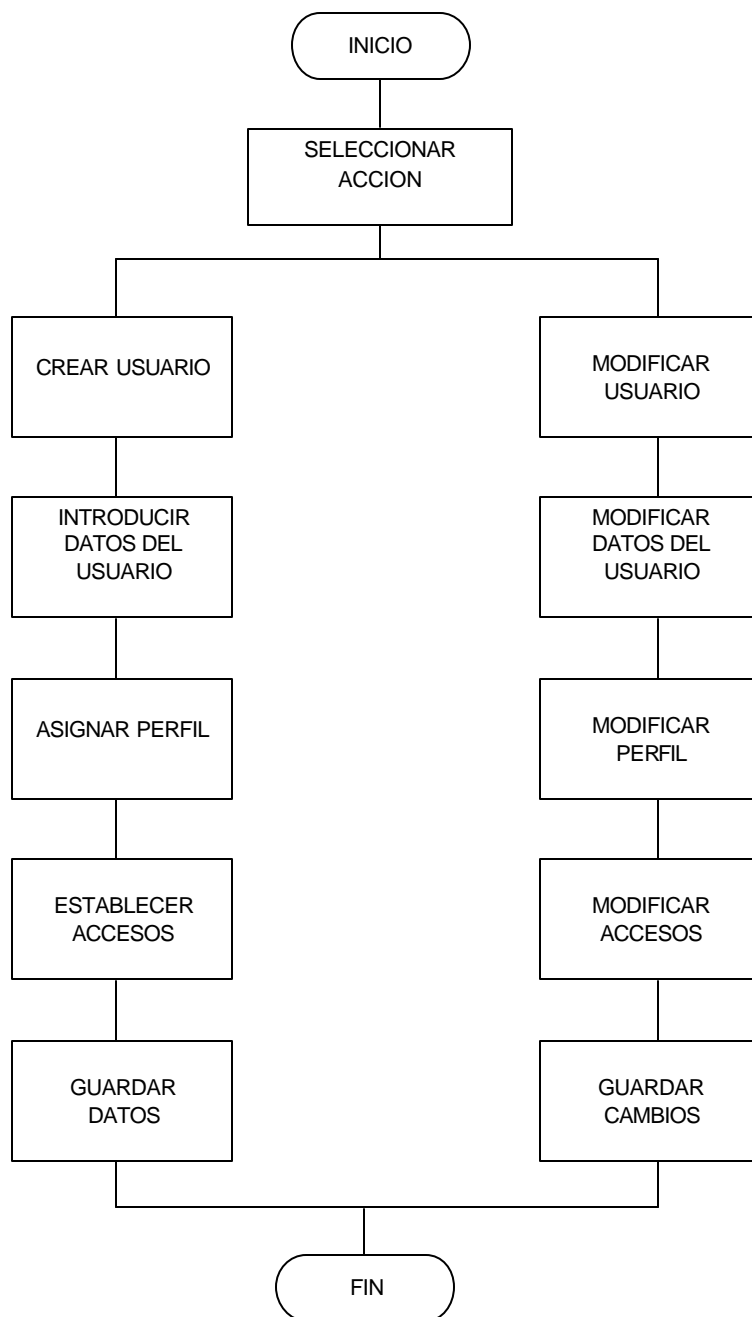


Figura 4.1.2.3 Diagrama de flujo del proceso Seguridad del Administrador.

Del diagrama 4.1.2.2 Proceso de Seguridad (1*), Figura 4.1.2.4

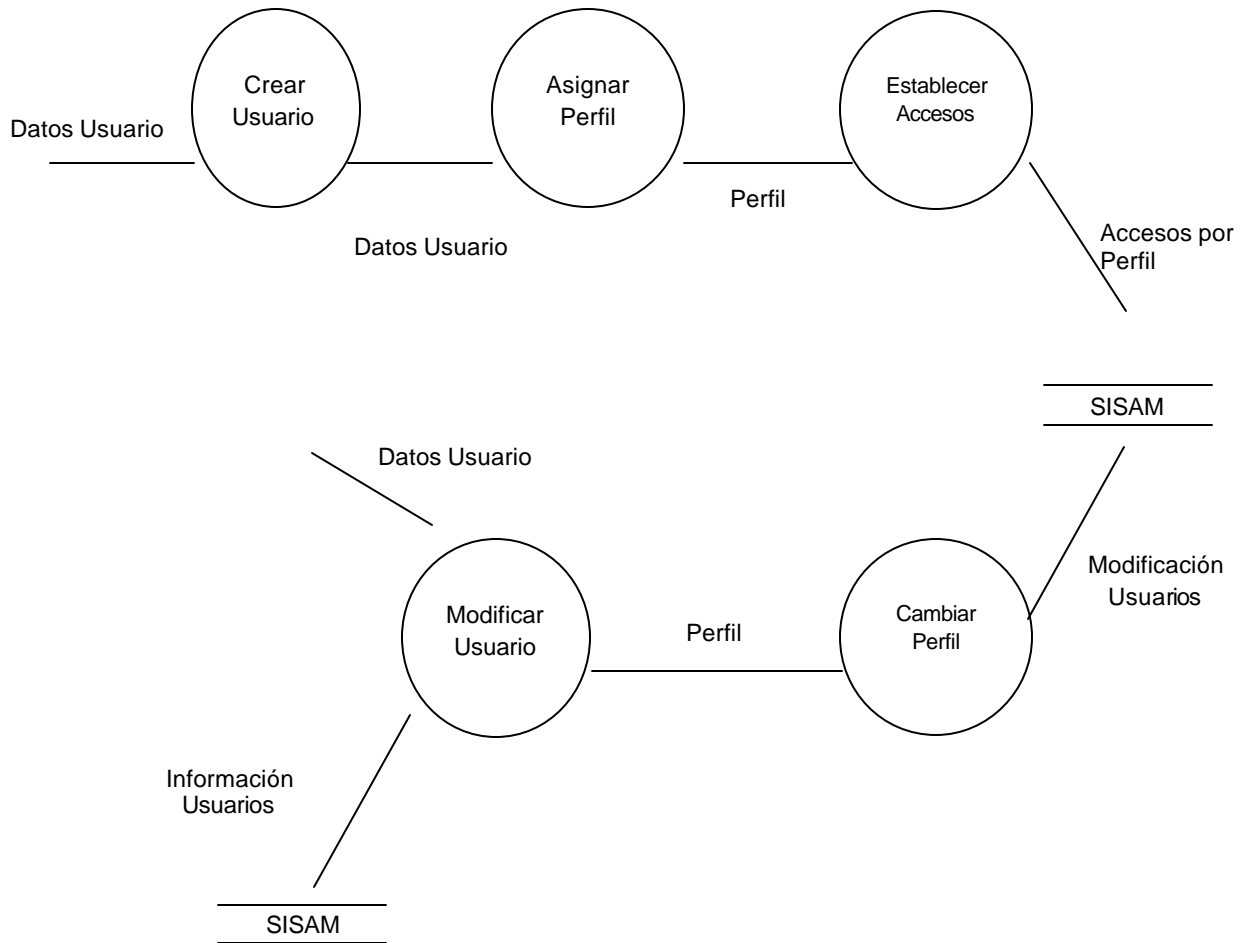


Figura 4.1.2.4 Diagrama de proceso de datos de Seguridad del Administrador.

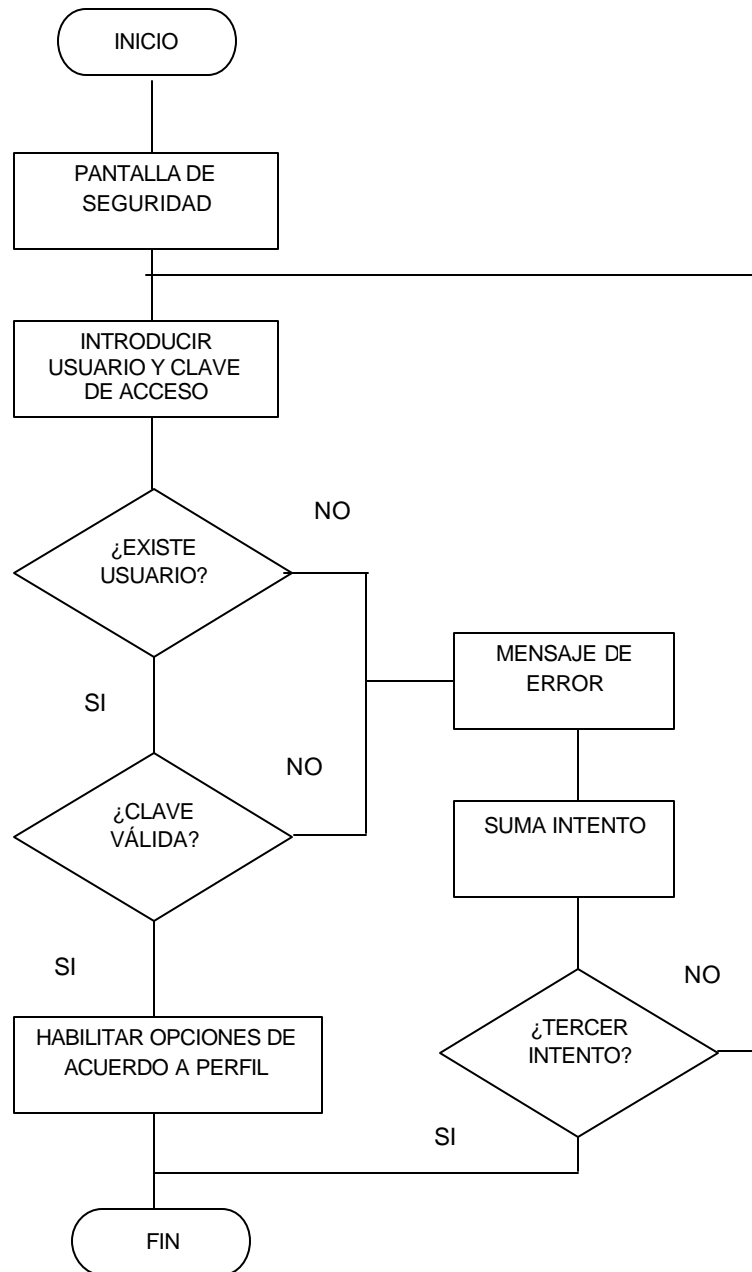


Figura 4.1.2.5 Diagrama de flujo de datos del proceso Seguridad del Usuario.

Del diagrama 4.1.2.2 Proceso de Seguridad (1*), Figura 4.1.2.6

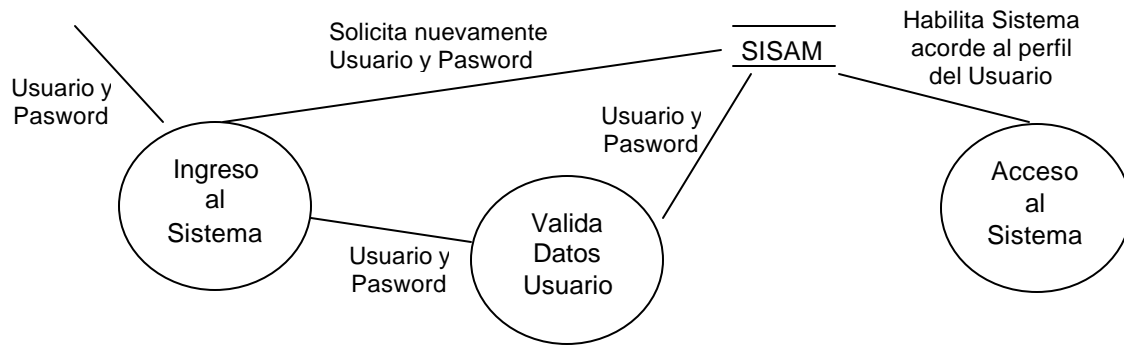


Figura 4.1.2.6 Diagrama de proceso de datos de Seguridad del Usuario.

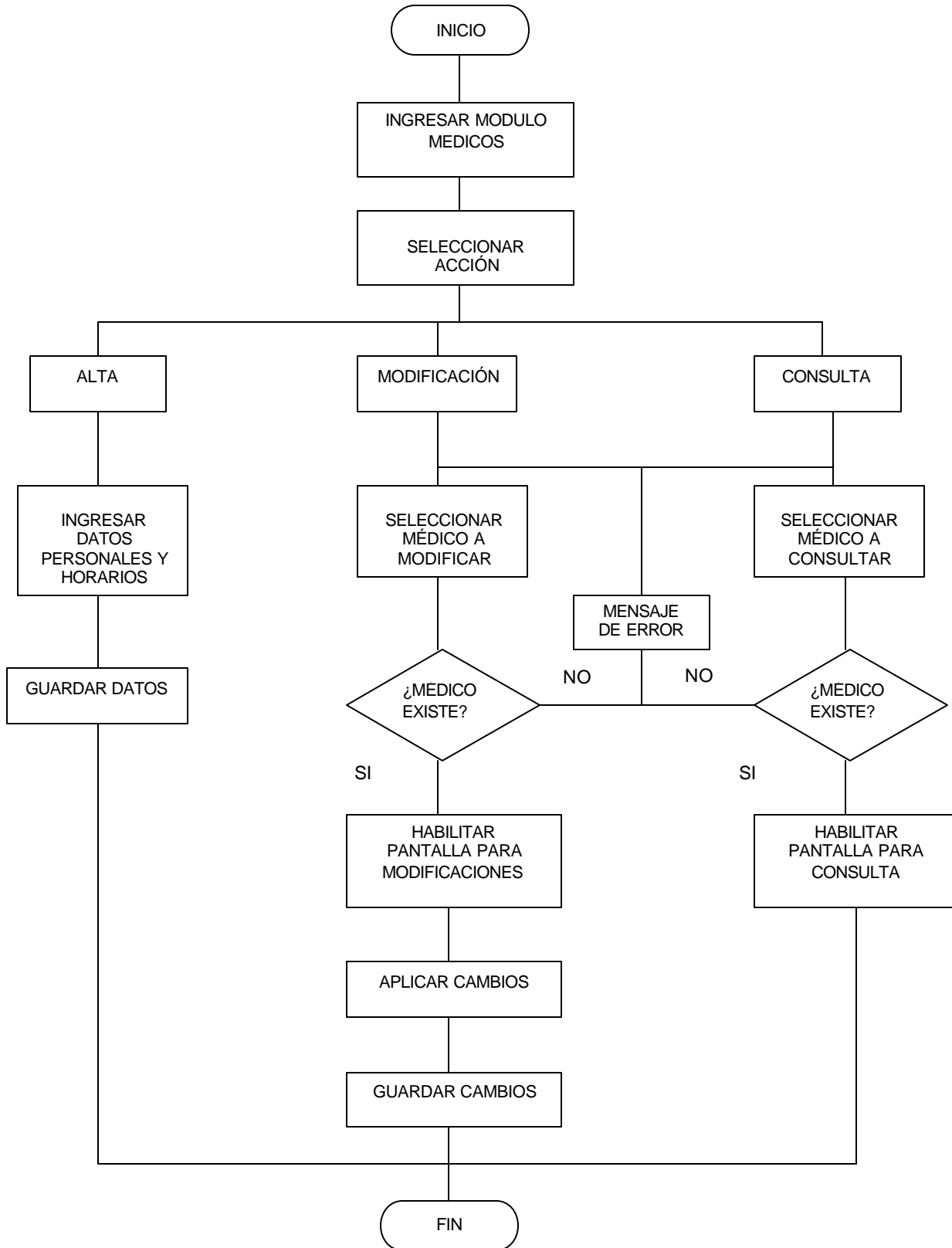


Figura 4.1.2.7 Diagrama de flujo de datos del proceso Médicos.

Del diagrama 4.1.2.2 Proceso de Médicos (2*), Figura 4.1.2.8

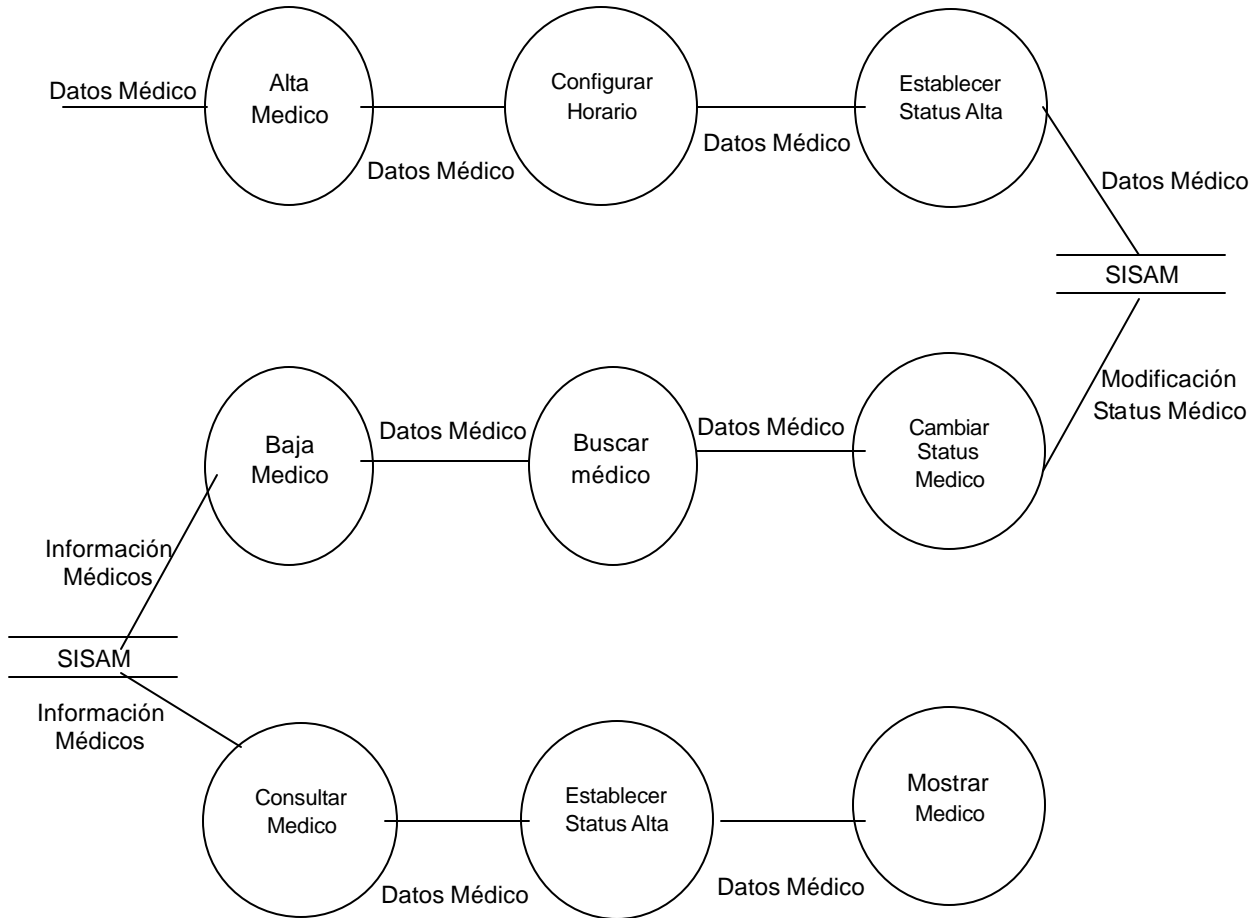


Figura 4.1.2.8 Diagrama de proceso de datos de Médicos.

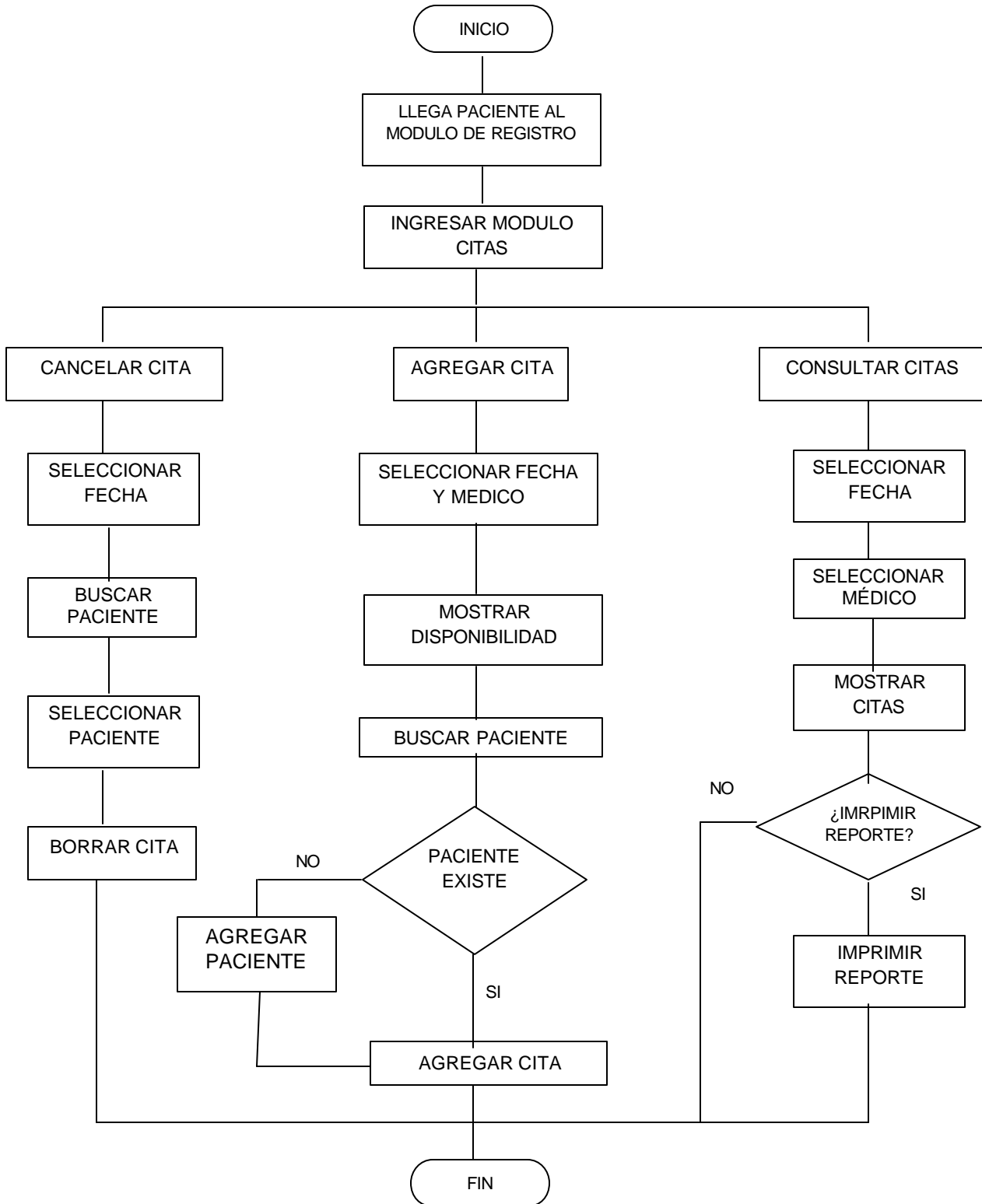


Figura 4.1.2.9 Diagrama de flujo de datos del proceso Citas.

Del diagrama 4.1.2.2 Proceso de Citas (3*), Figura 4.1.2.10

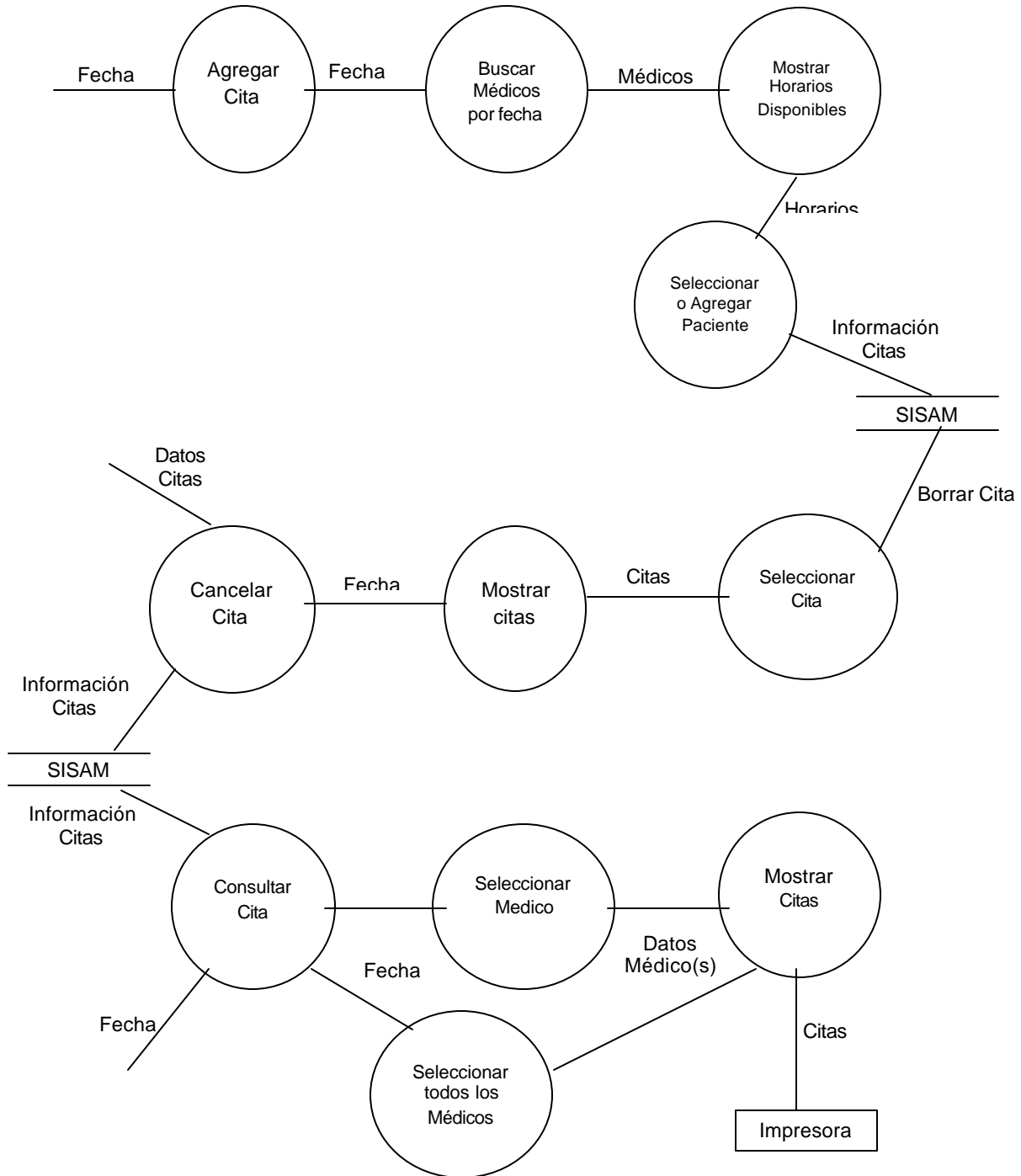


Figura 4.1.2.10 Diagrama de proceso de datos de Citas.

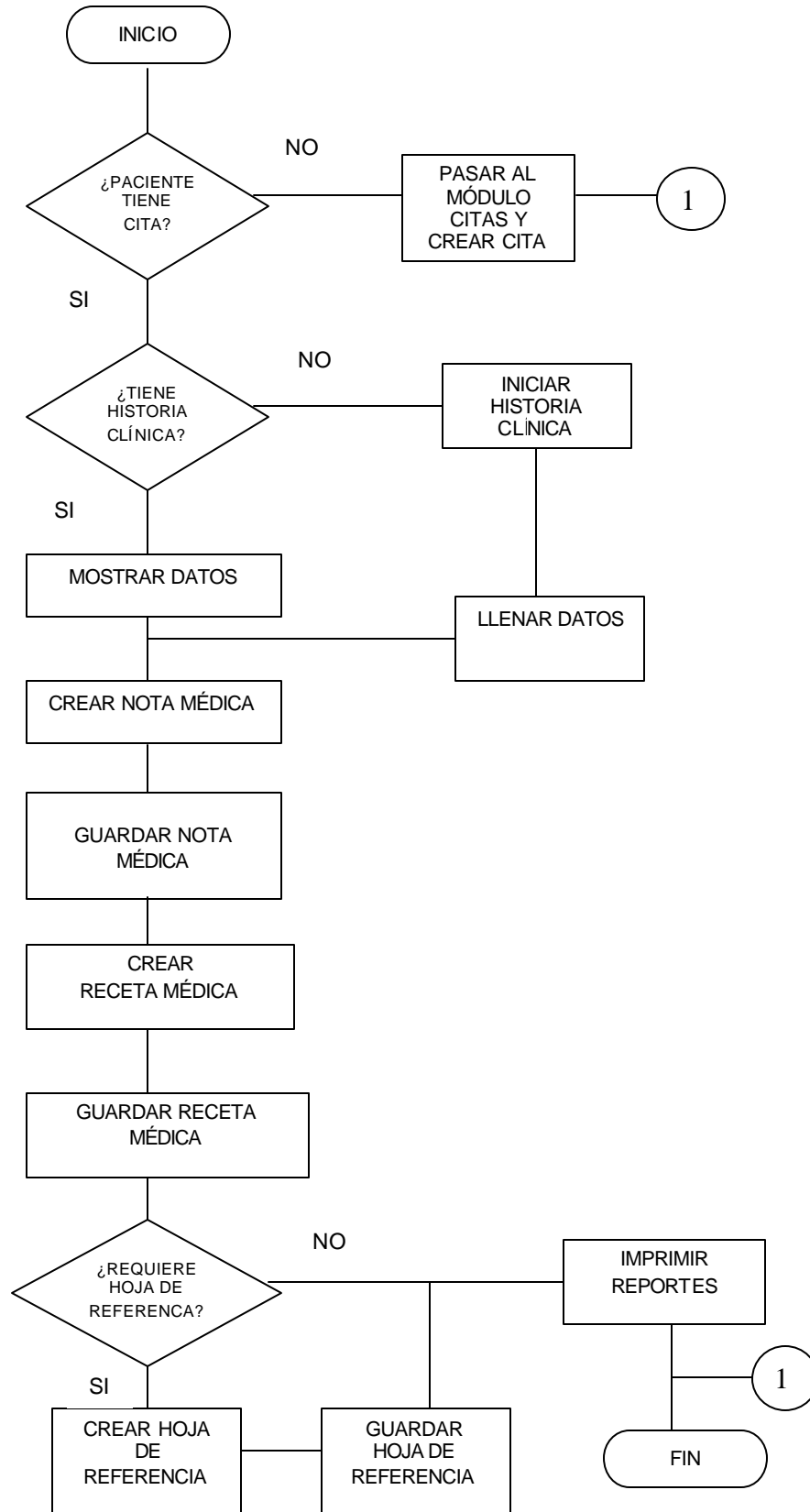


Figura 4.1.2.11 Diagrama de flujo de datos del proceso Pacientes.

Del diagrama 4.1.2.2 Proceso de Pacientes (4*), Figura 4.1.2.12

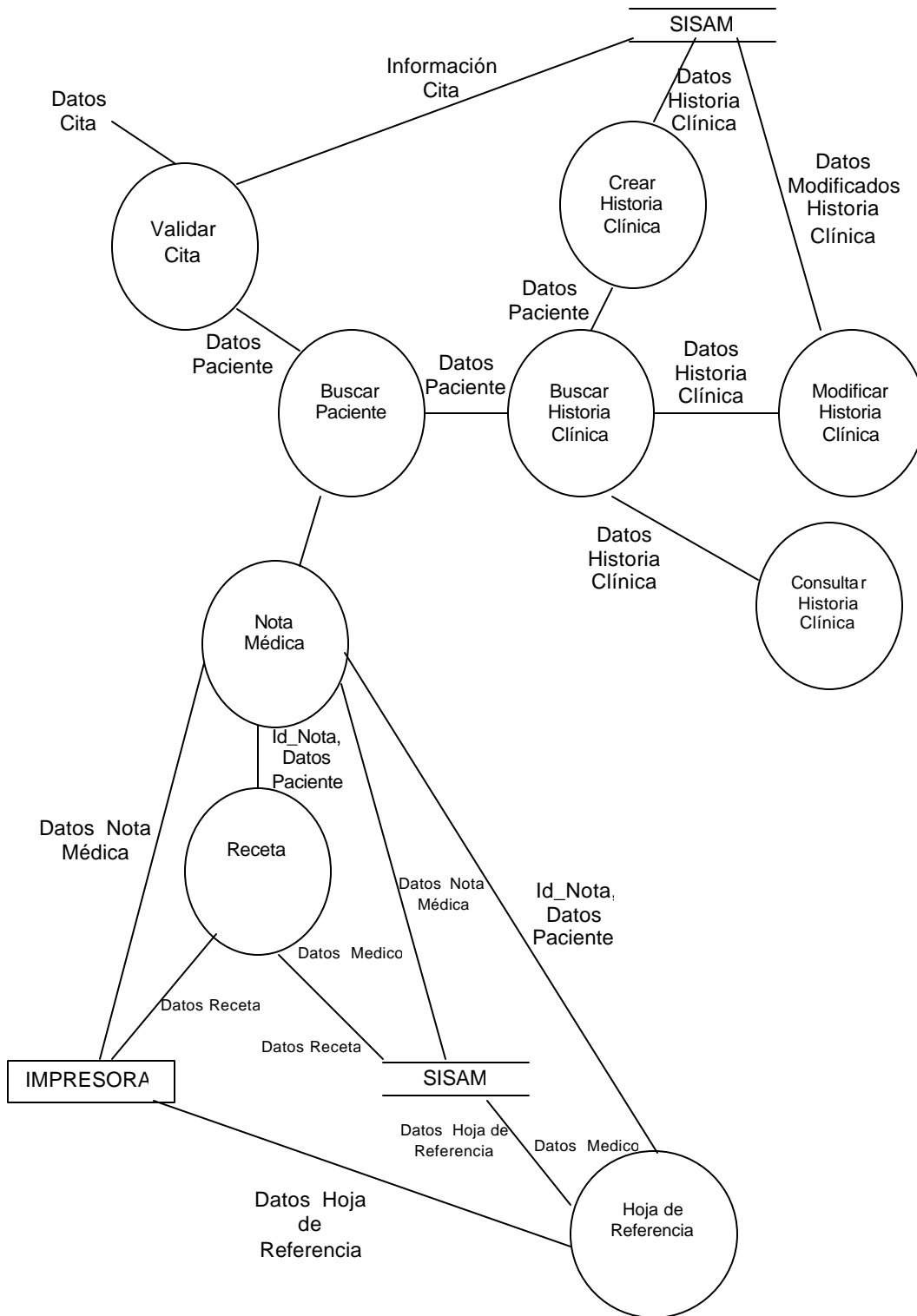


Figura 4.1.2.12 Diagrama de proceso de datos de Pacientes.

4.1.3 DICCIONARIO DE DATOS

Es una lista organizada que describe todos los elementos del sistema, con definiciones precisas para los analistas.

El contenido del diccionario de datos puede considerarse como “datos acerca de los datos” (los cuales en ocasiones reciben el nombre de metadatos), es decir objetos de la base de datos y no tan sólo datos en bruto.

Ver tablas 4.1.3.1 a 4.1.3.10.

CITA

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA- RELACIÓN	DESCRIPCIÓN
Idcita	id_cita	10	Int	PK		Identificador de cita
Idmedico	id_medic	10	Int	FK	MEDICO	Identificador de médico
Expediente	id_pcte	10	Int	FK	PACIENTE	Identificador de expediente del paciente
Hora_cita	hr_cita		Time			Hora de la cita del paciente
Fecha_cita	fec_cita		Date			Fecha de la cita del paciente

Tabla 4.1.3.1 Descripción Cita.

ESPECIALIDAD

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA- RELACIÓN	DESCRIPCIÓN
Idepecialidad	id_espec	10	Int	PK		Identificador de especialidad de médicos
Descripción	descrip	50	Varchar			Descripción de la especialidad.

Tabla 4.1.3.2 Descripción Especialidad.

GRUPO

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA- RELACIÓN	DESCRIPCIÓN
Idgrupo	id_gpo	10	Varchar	PK		Identificador del grupo al que pertenece cada usuario del sistema
Nombre	nom_gpo	30	Varchar			Nombre del grupo al que pertenece cada usuario del sistema.
Leer cita	cit_read	4	Tinyint			Permiso de grupo para consultar datos en el módulo de citas.
Agregar cita	cit_add	4	Tinyint			Permiso de grupo para agregar datos en el módulo de citas.
Actualizar cita	cit_upd	4	Tinyint			Permiso de grupo para actualizar datos en el módulo de citas.
Cancelar cita	cit_del	4	Tinyint			Permiso de grupo para cancelar datos en el módulo de citas.
Leer datos pacientes	pac_read	4	Tinyint			Permiso de grupo para consultar datos en módulo de pacientes.
Agregar datos pacientes	pac_add	4	Tinyint			Permiso de grupo para agregar datos en el módulo de pacientes.
Actualizar datos pacientes	pac_upd	4	Tinyint			Permiso de grupo para actualizar datos en el módulo de pacientes.
Cancelar datos pacientes	pac_del	4	Tinyint			Permiso de grupo para cancelar datos en el módulo de pacientes.

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA- RELACIÓN	DESCRIPCIÓN
Leer datos médico.	med_read	4	Tinyint			Permiso de grupo para consultar datos en módulo de pacientes.
Agregar datos médico.	med_add	4	Tinyint			Permiso de grupo para agregar datos en el módulo de médicos.
Actualizar datos médico.	med_upd	4	Tinyint			Permiso de grupo para actualizar datos en el módulo de médicos.
Cancelar datos médico.	med_del	4	Tinyint			Permiso de grupo para cancelar datos en el módulo de médicos.
Leer datos administrador	adm_read	4	Tinyint			Permiso de grupo para consultar datos en módulo de administrador.
Agregar datos administrador	Adm_add	4	Tinyint			Permiso de grupo para agregar datos en el módulo de administrador.
Actualizar datos administrador	Adm_upd	4	Tinyint			Permiso de grupo para actualizar datos en el módulo de administrador.
Cancelar datos administrador	Adm_del	4	Tinyint			Permiso de grupo para cancelar datos en el módulo de administrador.

Tabla 4.1.3.3 Descripción Grupo.

HISTORIA

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA- RELACIÓN	DESCRIPCIÓN
Idhistoria	id_histo	10	Int	PK		Número de historia clínica.
Expediente	id_pcte	10	Int	FK	PACIENTE	Identificador de expediente del paciente
Fecha_creación	fec_crea		Datetime			Fecha de creación de la historia clínica.
Fecha_modificación	fec_modi		Datetime			Fecha de modificación de la historia clínica.
Interrogatorio	interrog	10	Varchar			Tipo de interrogatorio: directo o indirecto.
Antecedentes no patológicos	ant_npat		Text			Antecedentes no patológicos del paciente.
Antecedentes patológicos	ant_pat		Text			Antecedentes patológicos del paciente.
Antecedentes ginecoobstétricos	ant_gine		Text			Antecedentes ginecoobstétricos del paciente en caso de ser mujer.
Antecedentes terapéuticos	ant_tera		Text			Antecedentes terapéuticos del paciente.

Tabla 4.1.3.4 Descripción Historia Médica.

HISTORIALOG

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA RELACIÓN	DESCRIPCIÓN
Idlog	id_log	10	Int	PK		Identificador de registro en la bitácora.
Idhistoria	id_histo	10	Int	FK	HISTORIA	Número de historia clínica que fue modificada.
Idmedico	id_medic	10	Int	FK	MEDICO	Identificador de médico que realizó la modificación de la historia clínica.
Fecha de modificación	fecha		Datetime			Fecha de modificación de la historia clínica.
Cadena de Modificación	datosact		Text			Cadena de datos modificados

Tabla 4.1.3.5 Descripción Modificaciones a Historia Médica.

HOJAREF

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Idhojaref	id_href	10	Int	PK		Identificador de la hoja de referencia.
idpaciente	id_pcte	10	Int	FK	PACIENTE	Expediente al que se le extiende hoja de referencia.
idmedico	id_medic	10	Int	FK	MEDICO	Identificador del médico que realizó la hoja de referencia.
Fecha	fecha		Datetime			Fecha en que se extendió la hoja de referencia.
Resumen	resumen	255	Text			Resumen del padecimiento del paciente.

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Motivo	motivo	255	Text			Motivo del envío a otra institución.
Transfiere a	transf_a	100	Varchar			Intitución a la que se envía al paciente.

Tabla 4.1.3.6 Descripción Hoja Referencia.

HRMEDICO

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Idturno	id_turno	10	Int	PK/FK	TURNO	Identificador del turno de médicos.
Idmedico	id_medico	10	Int	PK/FK	MEDICO	Identificador del médico.

Tabla 4.1.3.7 Descripción Horario Médico.

MEDICO

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Id Medico	id_medico	10	Int	PK		identificador de Médico
Id Especialidad	id_espec	10	Int	FK	ESPECIALIDAD	identificador de Especialidad
Id Usuario	id_user	30	Varchar	FK	USUARIO	identificador de Usuario
Nombre	nom_med	25	Varchar			Nombre de Medico
Apellido Paterno	app_med	25	Varchar			Apellido Paterno
Apellido Materno	apm_med	25	Varchar			Apellido Materno
Sexo	sexo	15	Varchar			Sexo
Cedula	cedula	20	Varchar			Cédula Profesional
Colegio	colegio	60	Varchar			Escuela de Procedencia
Calle	calle	60	Varchar			Dirección Medico
Colonia	colonia	45	Varchar			Dirección Medico
Código Postal	cp	5	Varchar			Dirección Medico
Municipio o Delegación	muni_del	30	Varchar			Dirección Medico
Ciudad	ciudad	30	Varchar			Dirección Medico
Estado	estado	20	Varchar			Dirección Medico
Tel. Celular	tel_cel	12	Varchar			Tel. celular

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Tel. Radio	tel_rad	12	Varchar			Biper
Tel. Casa	tel_casa	12	Varchar			Tel. casa
Tiempo Consulta	duracion	10	Int			Duración de cada consulta

Tabla 4.1.3.8 Descripción Médico.

NOTA

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Id nota	id_nota	10	Int	PK		identificador de Nota
Expediente	id_pcte	10	Int	FK	PACIENTE	Expediente del Paciente
Id medico	id_medic	10	Int	FK	MEDICO	identificador de Médico
Fecha creación	fec_crea		Datetime			Fecha de creación
Fecha modificación	fec_modi		Datetime			Fecha de modificación
Peso	peso	7	Varchar			Peso del Paciente
Talla	talla	6	smallInt			Talla del paciente
Frecuencia respiratoria	fre_resp	6	smallInt			Frecuencia respiratoria paciente
Frecuencia cardiaca	fre_card	6	smallInt			Frecuencia cardiaca paciente
Pulso	pulso	6	smallInt			Pulso paciente
Tensión arterial	ten_art	7	Varchar			Tensión arterial paciente
Temperatura	temp	5	Varchar			Temperatura paciente
Nota medica	nota		Text			Información de la consulta
Diagnostico	diagnost	255	Varchar			Diagnostico
Tratamiento	tramto		Text			Teratamiento
Análisis Laboratorio	anl_lab		Text			Análisis de Laboratorio
Análisis Gabinete	anl_gab		Text			Análisis de Gabinete

Tabla 4.1.3.9 Descripción Nota Médica.

PACIENTE

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Expediente	id_pcte	10	Int	PK		identificador de paciente, expediente
Nombre	nombre	25	Varchar			Nombre paciente
Apellido Paterno	ap_pater	25	Varchar			Apellido Paterno
Apellido Materno	ap_mater	25	Varchar			Apellido Materno
Sexo	sexo	15	Varchar			Sexo
Fecha de Nacimiento	fec_nac		Date			Fecha de Nacimiento
Edad	edad	3	Int			Edad
Lugar Nacimiento	lug_nac	50	Varchar			Lugar de Nacimiento
Estado Civil	edo_civ	15	Varchar			Estado Civil
Ocupación	ocupac	30	Varchar			ocupación
Calle	calle	60	Varchar			Dirección Paciente
Colonia	colonia	50	Varchar			Dirección Paciente
Municipio o Delegación	muni_del	50	Varchar			Dirección Paciente
Código Postal	cp	6	Varchar			Dirección Paciente
Estado	estado	25	Varchar			Dirección Paciente
Tel. Celular	tel_cel	13	Varchar			Tel. Celular
Tel. Casa	tel_casa	13	Varchar			Tel. Casa
Te. Trabajo	tel_tra	13	Varchar			Tel. Trabajo
Email	email	40	Varchar			Email
Religión	religion	30	Varchar			Religión

Tabla 4.1.3.10 Descripción Paciente.

RECETA

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Id receta	id_recet	10	Int	PK		identificador de Receta
Expediente	id_pcte	10	Int	FK	PACIENTE	Expediente del paciente
Id medico	id_medic	10	Int	FK	MEDICO	identificador de Médico
Fecha	fecha		Datetime			Fecha de expedición
Receta	receta		Text			Receta

Tabla 4.1.3.11 Descripción Receta.

TURNO

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Id turno	id_turno	10	Int	PK		identificador de turno
Horario inicio	hora_ini		Time			Inicio de Labores
Horario fin	hora_fin		Time			Fin de labores
Lunes	lu	4	TinyInt			Día laboral
Martes	ma	4	TinyInt			Día laboral
Miércoles	mi	4	TinyInt			Día laboral
Jueves	ju	4	TinyInt			Día laboral
Viernes	vi	4	TinyInt			Día laboral
Sábado	sa	4	TinyInt			Día laboral
Domingo	do	4	TinyInt			Día laboral

Tabla 4.1.3.12 Descripción Turno.

USUARIO

ATRIBUTO	ACRÓNIMO	LONGITUD	TIPO	LLAVE	TABLA-RELACIÓN	DESCRIPCIÓN
Id usuario	id_user	30	Varchar	PK		identificador de Usuario
Id grupo	id_gpo	10	Varchar	FK	GRUPO	identificador de Grupo
Nombre usuario	user_nom	30	Varchar			Nombre de Usuario
Apellido paterno del usuario	user_app	30	Varchar			
Apellido materno del usuario	user_apm	30	Varchar			
Fecha Alta	fec_alta		Datetime			Fecha de Alta
Fecha baja	fec_baja		Datetime			Fecha de Baja
Password de Usuario	user_pwd	30	Varchar			Password para el Usuario
Estatus	estatus	2	Varchar			Estatus de Usuario

Tabla 4.1.3.13 Descripción Usuario.

4.1.4 DIAGRAMA ENTIDAD-RELACIÓN

La definición de la estructura y respuesta de una base de datos tiene que ver en buena medida a la modelación de los datos que suministra el mundo real. Dicho de otra forma, nuestro entorno que motiva la base de datos debe ser concebido mediante un modelo que refleje claramente todos los objetos y sus restricciones. El diagrama entidad-relación se utiliza para describir los datos y sus relaciones de manera conceptual y de visión final.

A continuación, la figura 4.1.4.1 ilustra el Diagrama Entidad-Relación del sistema.

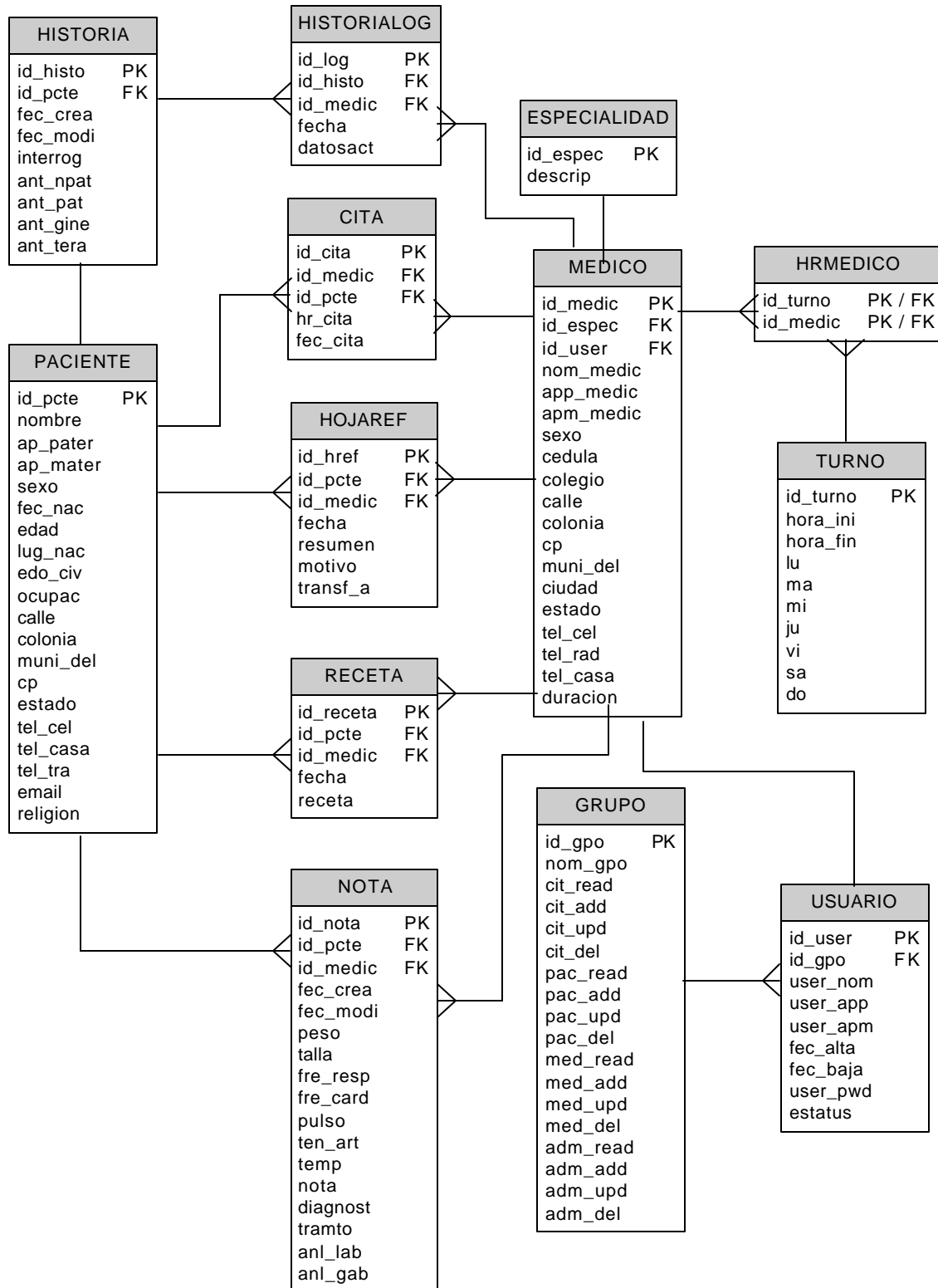


Figura 4.1.4.1 Diagrama Entidad-Relación.

4.1.5 NORMALIZACIÓN

Para mostrar el proceso de normalización que se llevó a cabo se recurre a tomar una muestra de datos.

Aplicando la primera forma normal, obtenemos la tabla 4.1.5.1 en la cual se almacenaron los datos de muestra, observando que contiene elementos atómicos o simples, quedando sin grupos repetitivos.

nombre del paciente
apellido paterno del paciente
apellido materno del paciente
sexo
edad
fecha de nacimiento
fecha de cita
hora de cita
peso
talla
pulso
nota médica
fecha de creación de nota médica
interrogatorio en historia clínica
antecedentes patológicos
antecedentes no patológicos
fecha de creación de historia clínica
fecha de modificación de historia clínica
nombre del médico
apellido paterno del médico
apellido materno del médico
cedula
especialidad

Figura 4.1.5.1 Tabla en 1ª. Forma normal

Para aplicar la segunda forma normal, es necesario tener las tablas en primera forma normal, se crearon las tablas necesarias para agrupar los datos y se establecieron las llaves primarias, una vez establecidas se revisó que cada uno de los atributos dependan únicamente de ésta, en este caso quedaron llaves primarias simples, y debido a que no tenemos llaves compuestas no existen dependencias parciales, por lo que las tablas se encuentran ya en segunda forma normal.

PACIENTE

id_pcte
nombre
ap_pater
ap_mater
sexo
edad
fec_nac

CITA

id_cita
fec_cita
hr_cita

NOTA

id_nota
fec_crea
peso
talla
pulso
nota

HISTORIA

id_histo
fec_crea
fec_modi
interrog
ant_npat
ant_pat

MEDICO

id_medic
nom_med
app_med
apm_med
cedula

ESPECIALIDAD

id_espec
descripcion

Para que una relación esté en tercera forma normal, cada atributo no llave debe depender totalmente de la llave primaria y no debe haber dependencias transitivas. Hay dependencias transitivas cuando un atributo no llave depende de uno o más atributos no llave.

PACIENTE

id_pcte
nombre
ap_pater
ap_mater
sexo
edad
fec_nac

CITA

id_cita
fec_cita
hr_cita
id_medic
id_pcte

Como esta tabla tiene relación con la tabla MEDICO y con la tabla PACIENTE, las llaves primarias id_pcte y id_medic pasan como llaves foráneas en calidad de atributo a la tabla CITA. Cada uno de los atributos depende completamente de id_cita.

NOTA

id_nota
fec_crea
peso
talla
pulso
nota
id_pcte
id_medic

Los atributos id_pcte y id_medic son llaves foráneas debido a la relación con la tabla PACIENTE y MEDICO, ya que una nota debe pertenecer a un paciente y la realiza un médico. Todos los atributos dependerán del id_nota.

HISTORIA

id_histo
fec_crea
fec_modi
interrog
ant_npat
ant_pat
id_pcte

La tabla HISTORIA se relaciona con la tabla PACIENTE, ya que que un paciente debe tener una historia clínica, por lo que el id_pcte es una llave foránea y pasa como atributo a HISTORIA. Todos los atributos dependen completamente del id_histo.

MEDICO

id_medic
nom_med
app_med
apm_med
cedula
id_espec

El atributo id_espec es una llave foránea por la relación de la tabla MEDICO con la tabla ESPECIALIDAD. Todos los atributos dependen únicamente del id_medic.

ESPECIALIDAD

id_espec
descripcion

Esta tabla cuenta únicamente con la llave primaria que identifica la especialidad y la descripción de dicha especialidad. Por lo que no hay dependencia transitiva.

4.2 DISEÑO Y GENERACIÓN DEL BACK-END

Instalación de MySQL Server 5.0 en Windows

Lo primero que se tiene que hacer es obtener la última distribución de MySQL de la dirección <http://www.mysql.com/downloads> y guardar el archivo de instalación en una carpeta de nuestro disco duro.

Una vez que tengamos el archivo de instalación de MySQL, procedemos a ejecutar el programa setup.exe para dar inicio al proceso del mismo.

La figura 4.2.1 muestra la pantalla de bienvenida del proceso de instalación y configuración de MySQL Server 5.0.



Figura 4.2.1 Pantalla de bienvenida del proceso de Instalación de MySQL .

En este punto es donde podemos decidir si continuamos o no con la instalación. Para abortar el proceso solo basta con dar un clic en el botón **Cancel** (Cancelar). Para nuestro caso, daremos clic en el botón **Next** (Siguiente).

El siguiente paso consiste en determinar el tipo de instalación que se desea. La figura 4.2.2 ilustra las diferentes opciones de instalación que se pueden realizar.

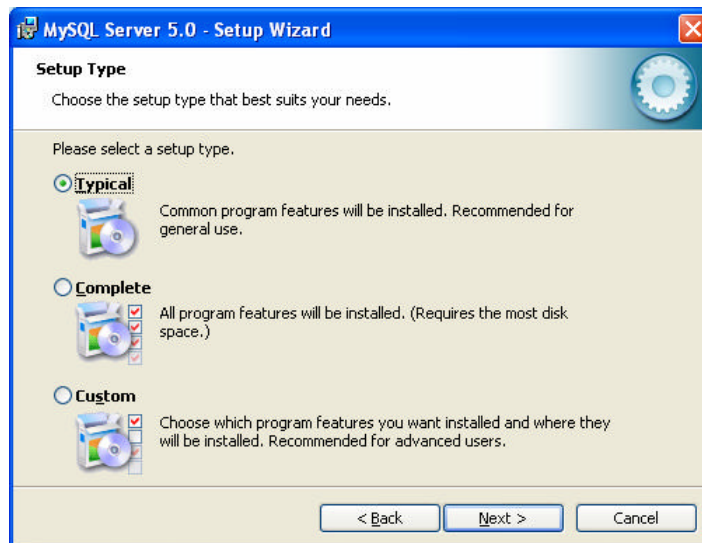


Figura 4.2.2 Pantalla de selección del tipo de instalación.

Una instalación **Typical** (Típica) consiste únicamente en instalar aquellos programas y características más comunes. Este tipo de instalación es recomendable para uso general.

Una instalación **Complete** (Completa), permite instalar todas las características y programas que conforman MySQL Server 5.0. Este tipo de instalación requiere de más espacio en disco duro.

Una instalación **Custom** (Personalizada) permite elegir al usuario las características y programas deseados por el mismo. Esta opción es aconsejable solo para usuarios expertos o con amplios conocimientos en MySQL.

Para nuestro caso haremos uso de la opción Complete, tal y como se muestra en la figura 4.2.3.

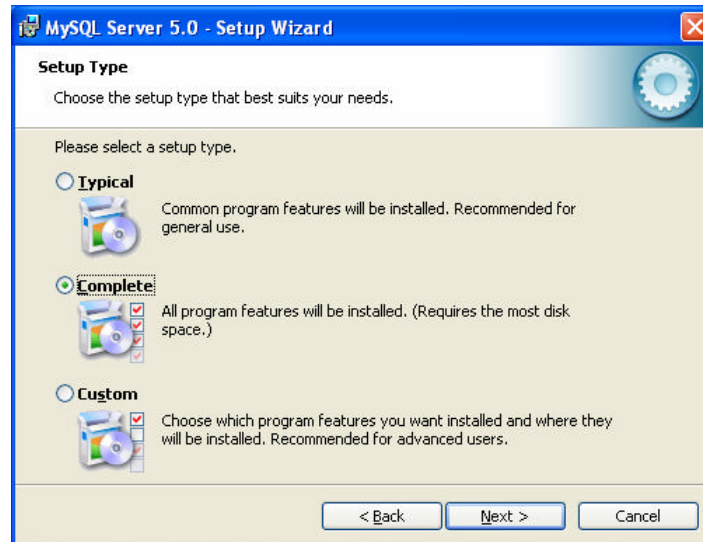


Figura 4.2.3 Pantalla de selección de instalación completa.

Una vez seleccionado el tipo de instalación, el siguiente paso es dar clic en el botón Next, para pasar a la siguiente pantalla. La siguiente pantalla (figura 4.2.4) nos indica el tipo de instalación que seleccionamos, así como la unidad de disco y ruta en donde se instalará el programa.

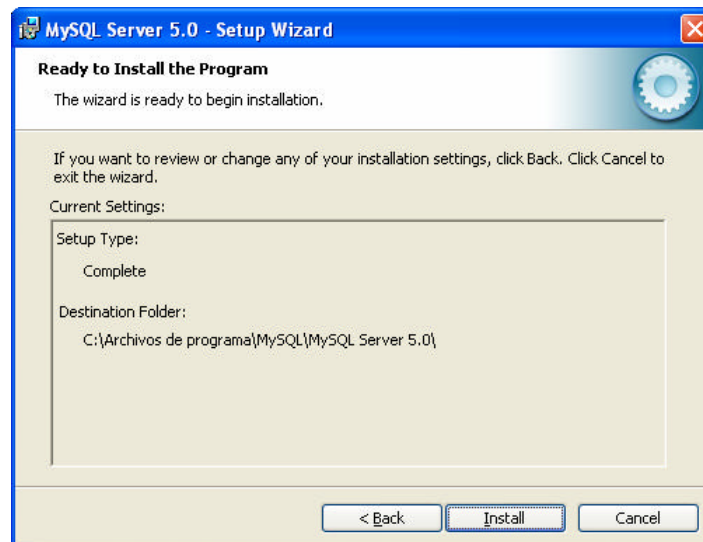


Figura 4.2.4 Pantalla de confirmación de la instalación.

Para iniciar con la instalación, es necesario dar clic en el botón **Install** (Instalar). La figura 4.2.5 muestra el proceso de avance de la instalación.

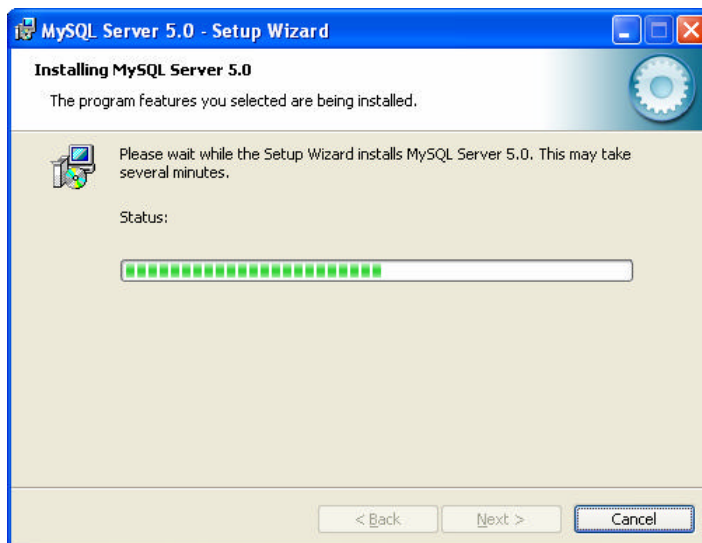


Figura 4.2.5 Pantalla de avance del proceso de instalación.

Una vez que el proceso de copiado de los programas de MySQL Server 5.0 a terminado, el siguiente paso es el de configurar la creación de una cuenta en el sitio oficial de MySQL, cuyo objetivo es el poder gozar de las actualizaciones que constantemente se generan, así como información complementaria del mismo. La pantalla 4.2.6 muestra el proceso de creación de la cuenta de actualización.

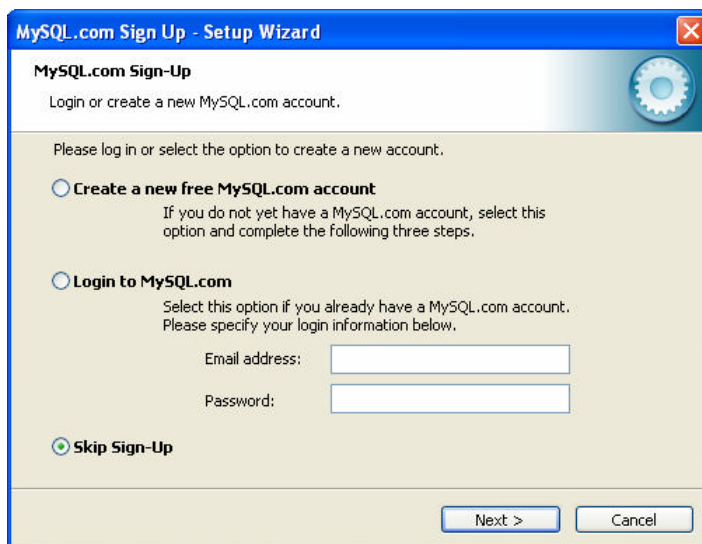


Figura 4.2.6 Pantalla de Configuración de una Cuenta en el sitio MySQL.com.

Para nuestro caso seleccionaremos la opción **Skip Sing-Up** (saltar registro) ya que no actualizaremos en línea nuestro MySQL Server. Una vez culminado el proceso de instalación, se muestra la pantalla de la figura 4.2.7 en donde se indica que la instalación se realizó de manera completa y exitosa.



Figura 4.2.7 Pantalla de Finalización de la Instalación.

Una vez culminado el proceso de instalación, el siguiente paso que debemos de realizar es el de configurar nuestro Servidor de Base de Datos. Para poder hacer esto, dentro de la pantalla 4.2.7, existe una caja de selección que nos permite indicar al programa de instalación, que deseamos proseguir inmediatamente con la configuración del Servidor de Base de Datos. Seleccionaremos dicha casilla y procedemos a dar clic en el botón **Finish** (Terminar).

Una vez que dimos clic en el botón Finish, el programa de instalación ejecuta automáticamente el programa de Configuración del Servidor de Base de Datos de MySQL Server. La figura 4.2.8 ilustra la pantalla de bienvenida del proceso de configuración de la Instancia del Servidor de Base de Datos.

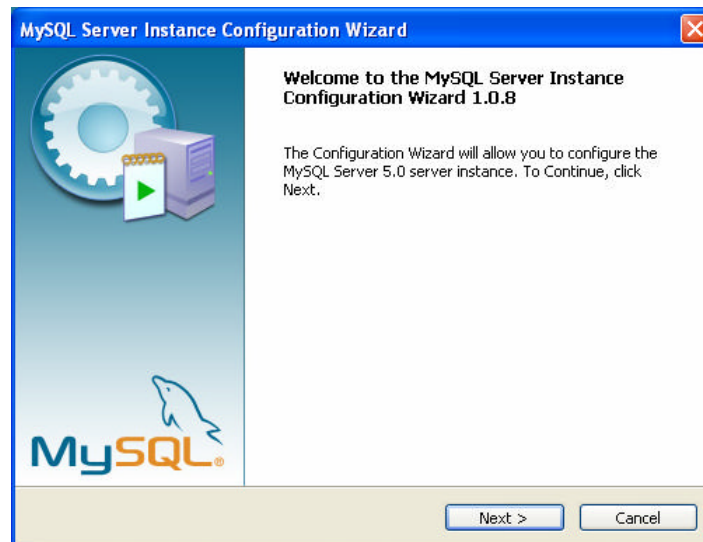


Figura 4.2.8 Pantalla de Configuración del Servidor de Base de Datos.

Para continuar con la configuración daremos clic en el botón Next. A continuación se presenta la pantalla de selección acerca del tipo de configuración que se desea llevar a cabo (figura 4.2.9).

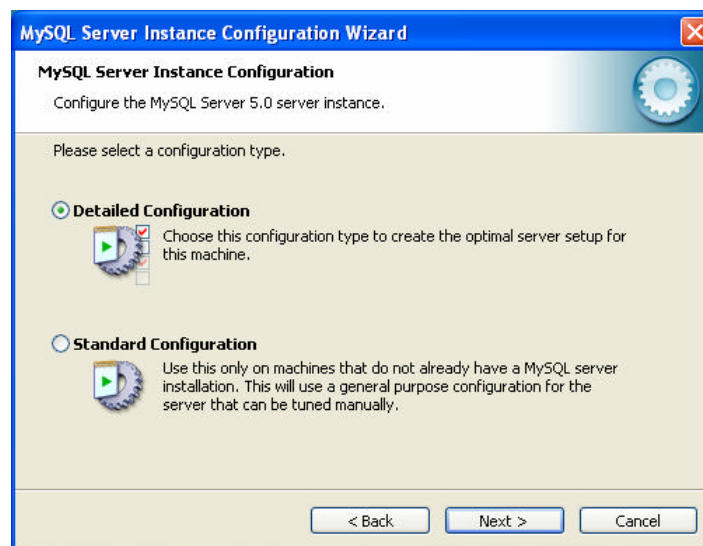


Figura 4.2.9 Tipo de Configuración del Servidor de Base de Datos.

La opción **Detailed Configuration** (Configuración Detallada) consiste en establecer paso a paso cada uno los aspectos de nuestro Servidor de Base de Datos (nombre del servidor, tipo de servidor, tipo de base de datos, puerto de

conexión, etc.). Este tipo de configuración es recomendable para expertos en MySQL.

La opción **Standard Configuration** (Configuración Estándar) solo establecerá los principales aspectos del Servidor de Base de Datos (tipo de servidor, tipo de base de datos, puerto de conexión, etc.). Esta opción es recomendable cuando queremos configurar manualmente el Servidor de Base de Datos sin la necesidad de una interface gráfica.

Para nuestro caso, seleccionaremos la configuración detallada y daremos clic en el botón Next.

La siguiente pantalla que se presenta nos permite establecer el tipo de máquina en donde se albergará el Servidor de Base de Datos de MySQL (figura 4.2.10).



Figura 4.2.10 Tipo de Servidor de Base de Datos.

La opción **Developer Machine** (Equipo de Desarrollo) nos permite indicar al Servidor de Base de Datos que debe hacer uso mínimo de la memoria del equipo y servicios del mismo. Se recomienda esta opción para el desarrollo de aplicaciones.

La opción **Server Machine** (Equipo Servidor) nos permite indicar al Servidor de Base de Datos que debe hacer uso medio de los recursos del equipo y memoria del mismo. Esta opción es recomendable para ambientes de producción.

La opción **Dedicated MySQL Server Machine** (Equipo Dedicado) nos permite indicar al Servidor de Base de Datos que debe hacer uso de los recursos y memoria del equipo al máximo. Esta opción es recomendable cuando en nuestro equipo exclusivamente correrá MySQL Server.

Para nuestro caso seleccionaremos la opción **Server Machine** y daremos clic en el botón **Next** para continuar.

El siguiente paso es establecer el tipo de uso que tendrá nuestra base de datos (figura 4.2.11).

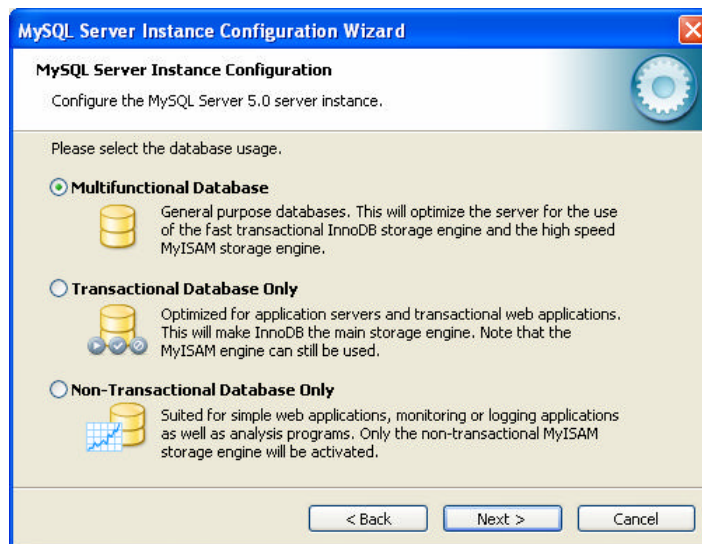


Figura 4.2.11 Tipo de uso de la base de datos.

La opción **Multifunctional Database** (Base de Datos Multifuncional) se debe elegir cuando se requieren bases de datos de propósito general.

La opción **Transactional Database Only** (Base de Datos Transaccional) se debe elegir cuando deseamos hacer uso de aplicaciones Web y en donde se requiera hacer uso de transacciones.

La opción **Non-Transactional Database Only** (Base de Datos no Transaccional) se debe elegir para aplicaciones Web de bajo nivel y en donde no exista una alta incidencia de acceso a la base de datos.

Para nuestro caso haremos uso de la opción Multifunctional Database y daremos clic en el botón Next.

El siguiente aspecto a configurar, es indicar la unidad y ruta en donde deseamos albergar la base de datos (figura 4.2.12). Se recomienda dejar la base de datos en la misma unidad y carpeta en donde se instaló el Servidor de Base de Datos.

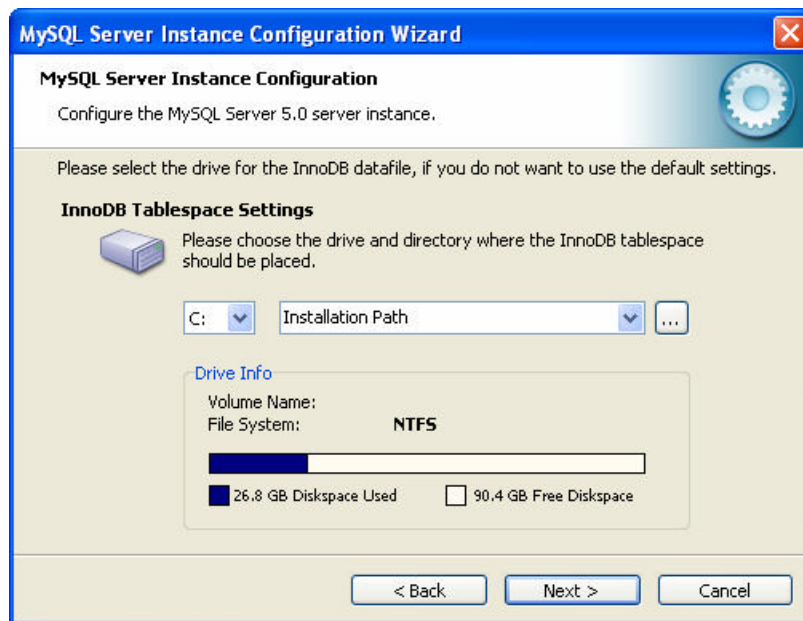


Figura 4.2.12 Especificación de la ruta de instalación de la Base de Datos.

Una vez que se indica la ruta en donde se albergará la base de datos, es necesario establecer el tipo de conexiones que soportará o que se permitirán hacia nuestro Servidor de Base de Datos (figura 4.2.13).

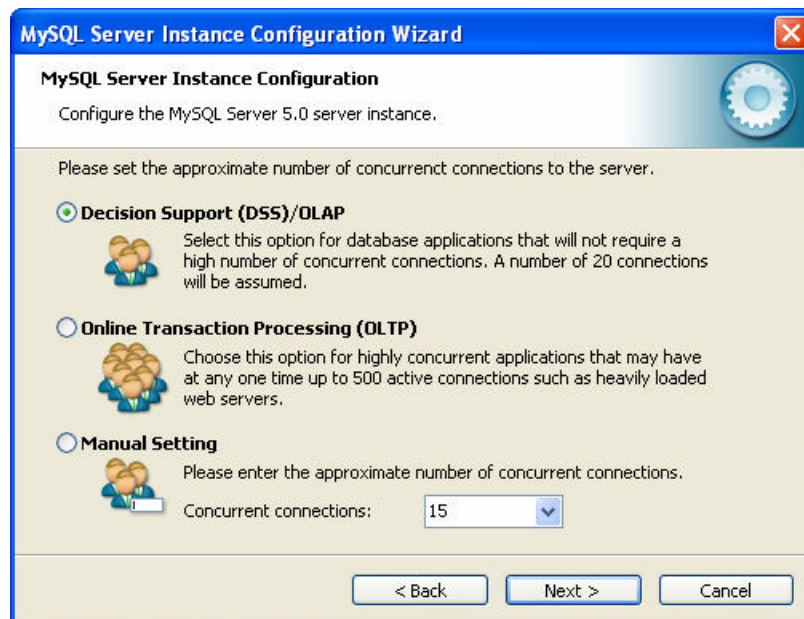


Figura 4.2.13 Selección del tipo y número de conexiones concurrentes.

La opción **Decision Support DSS/OLAP** (Procesos Analíticos en Línea), es recomendable para aplicaciones de base de datos que no requieran de un alto número de conexiones concurrentes. El número de conexiones asumidas para esta opción es de 20.

La opción **Online Transactional Processing (OLTP)** (Procesamiento de Transacciones en Línea) es recomendable para aplicaciones que pueden tener en cualquier momento hasta 500 conexiones activas, como lo son las aplicaciones Web.

La opción **Manual Setting** (Configuración Manual) es recomendable para establecer de manera manual el número de conexiones concurrentes que se desean tener.

Una vez elegido el tipo y número de conexiones que se desean tener, se debe dar clic en el botón Next para pasar a la siguiente pantalla.

La pantalla de configuración del puerto de conexión nos permite establecer el número de puerto TCP/IP (Protocolo de Control de Transmisión / Protocolo de Internet) que será utilizado por nuestro Servidor de Base de Datos para establecer comunicación con nuestra base de datos (figura 4.2.14). MySQL Server por defecto maneja el puerto 3306. Nosotros aceptaremos el propuesto y daremos clic en el botón Next para pasar a la siguiente pantalla.

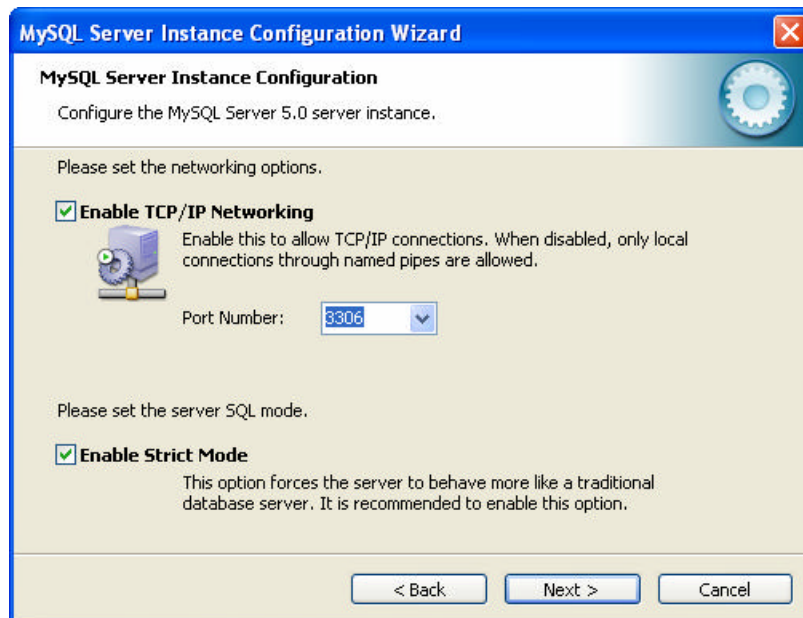


Figura 4.2.14 Selección del puerto de comunicación (TCP/IP).

Una vez establecido el puerto de comunicación de red, procederemos a configurar el juego de caracteres que nuestra base de datos puede albergar (figura 4.2.15). Para nuestro caso haremos uso de la primera opción.

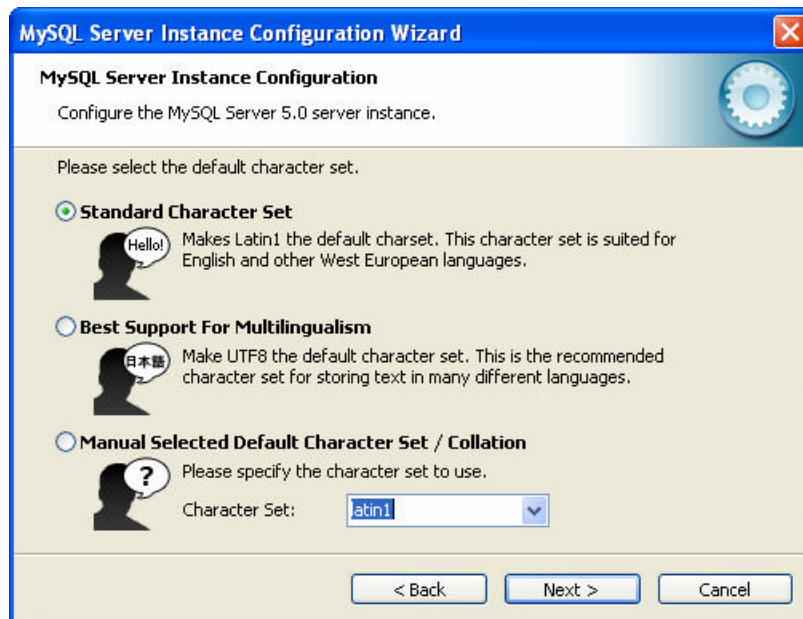


Figura 4.2.15 Selección del juego de caracteres.

Una vez seleccionado el juego de caracteres que deseamos que nuestra base de datos albergue, procederemos a indicar el nombre del Servicio de Windows con el cual deseamos arrancar nuestro Servidor de Base de Datos (figura 4.2.16).



Figura 4.2.16 Selección del nombre del Servicio de Windows.

También es posible agregar la ruta de MySQL dentro del **path** (ruta) de Windows con la finalidad de poder ejecutar los comandos de MySQL desde la línea de comando.

El paso siguiente es establecer el usuario que administrará nuestro Servidor de Base de Datos, así como su contraseña (figura 4.2.17).



Figura 4.2.17 Creación de la cuenta y contraseña del usuario administrador.

Es importante señalar que existe la posibilidad de crear una cuenta para acceso anónimo, pero por medidas de seguridad no procederemos a crearla. Para continuar con el proceso es necesario dar clic en el botón Next.

Finalmente, aparecerá la pantalla que muestra el avance de la configuración del Servidor de Base de Datos (figura 4.2.18).

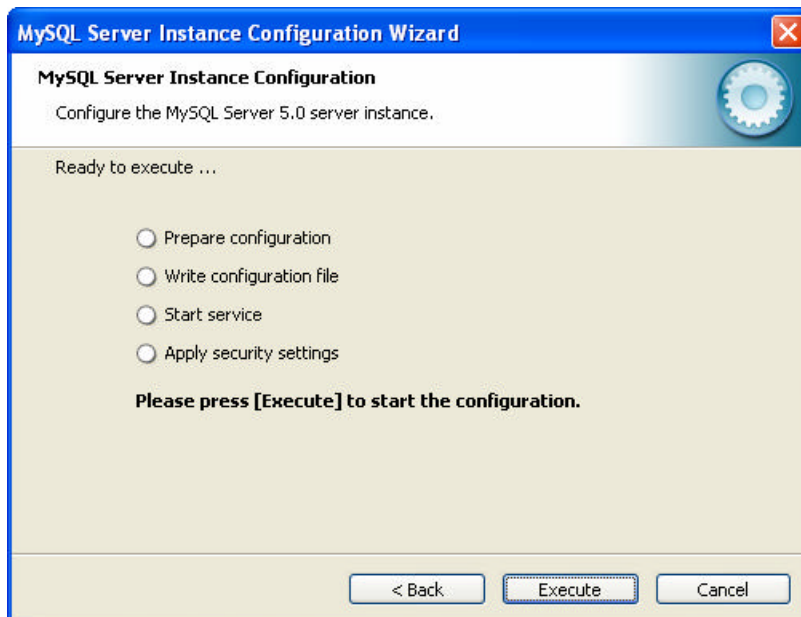


Figura 4.2.18 Pantalla de Avance de Configuración.

Para finalizar con la configuración, daremos clic en el botón **Execute** (Ejecutar). Durante el proceso es posible que nos aparezca la pantalla que muestra la figura 4.2.19, en donde MySQL nos notificará que no es posible acceder al Servidor de Base de Datos debido a que el **Firewall** (Corta Fuegos) de Windows está activo.

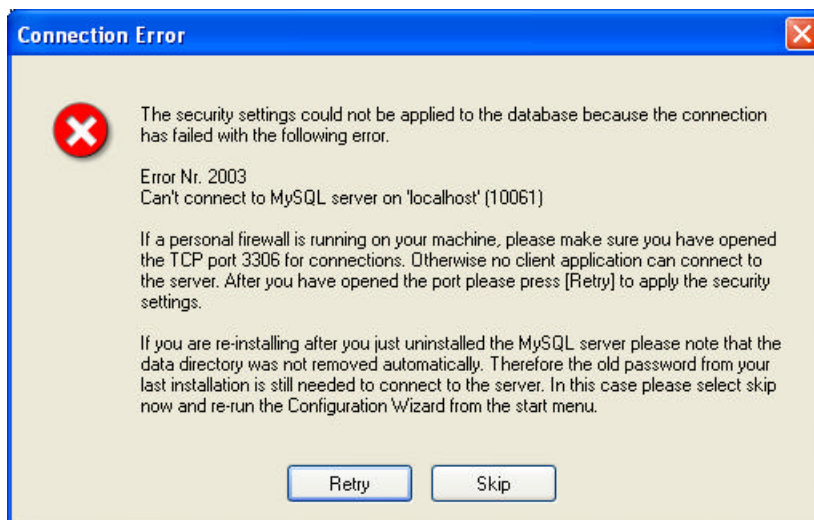


Figura 4.2.19 Error al acceder al Servidor de Base de Datos por el Firewall.

Para proseguir con la instalación daremos clic en el botón **Retry** (reintentar) para hacer caso omiso del error.

Finalmente, nos aparecerá la pantalla de la figura 4.2.20 indicándonos que el proceso de configuración ha sido exitoso y en donde daremos clic en el botón **Finish** para terminar.

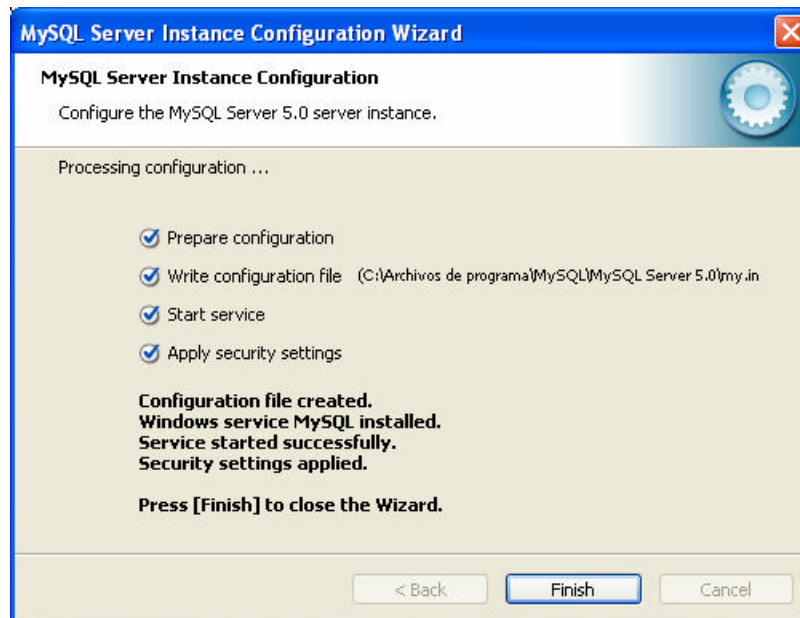


Figura 4.2.20 Proceso de configuración exitoso.

Creación de una Base de Datos

Existen dos formas de poder crear una base de datos en MySQL. La primera consiste en hacer uso de una utilidad gráfica denominada **MySQL Administrator** (Administrador de MySQL), la cual no viene implícitamente en la instalación de MySQL Server. Esta utilidad gráfica, se puede obtener en el sitio Web oficial de MySQL.

La segunda alternativa, es a través de la utilidad no gráfica denominada **MySQL Command Line Client** (Cliente de Línea de Comando MySQL), misma que viene integrada en la instalación de MySQL Server 5.0.

A continuación se explica como crear una base de datos en MySQL a través de las dos alternativas antes mencionadas.

Como Crear una Base de Datos en MySQL Command Line

Para poder crear una base de datos a través de esta utilidad es necesario seguir los siguientes pasos:

- Ejecutar la utilidad haciendo uso del acceso directo que se instala en Windows (figura 4.2.21)

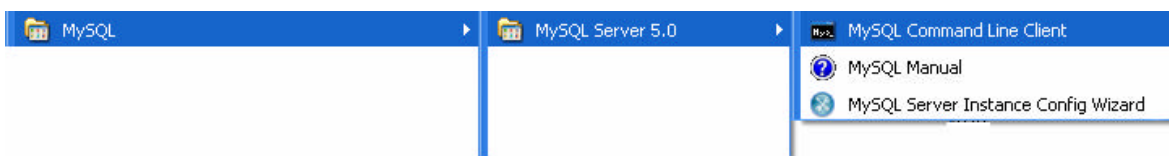


Figura 4.2.21 Utilería MySQL Command Line Client.

- Una vez ejecutada la utilidad aparecerá una pantalla de comando que solicitará la contraseña del usuario root para conectarse con el servidor (figura 4.2.22). Indicaremos la contraseña y presionaremos la tecla **Intro** (aceptar) para continuar.

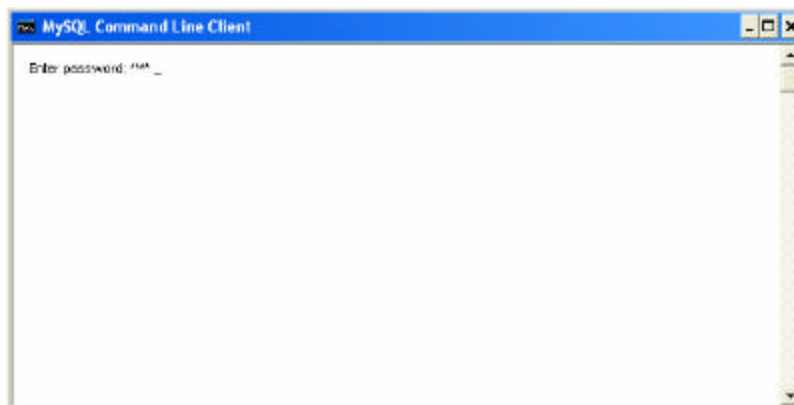


Figura 4.2.22 Acceso a MySQL Command Line Client.

- Una vez conectado al Servidor de Base de Datos, procederemos a teclear la siguiente instrucción:

```
create database <nombre_base>;
```

En donde <nombre_base> indica el nombre o seudónimo que deseamos dar a nuestra base de datos. MySQL nos responderá de la siguiente forma, indicando que la instrucción se ejecuto de manera exitosa.

```
Query OK, 1 row affected (0.32 sec)
```

- Conectarse al Servidor de Base de Datos haciendo uso de la cuenta del usuario root que fue configurada en la parte de la instalación de MySQL.

Como Crear una Tabla en MySQL Command Line Client

Para poder crear una tabla en una base de datos de MySQL, es necesario ejecutar la instrucción:

```
Create Table <nombre_tabla> (  
    <campo1> <tipo_dato1>,  
    <campo2> <tipo_dato2> ,  
    ...  
    ...  
    <campon> <tipo_daton>);
```

Antes de ejecutar el comando create table es necesario indicarle a MySQL en que base de datos deberá crear la tabla requerida, para esto haremos uso del comando use <nombre_base>

Para poder listar las bases de datos que se han creado haremos uso del comando `show databases`.

Para poder listar las tablas de una base de datos haremos uso del comando `show tables`.

Para poder ver la estructura de una tabla haremos uso del comando `describe <nombre_tabla>`.

Como Crear una Base de Datos en MySQL Administrator

Si se tiene instalada la utilidad gráfica de MySQL Administrator, la creación de una base de datos tiende a ser más fácil y práctica.

Para crear la base de datos, es necesario ejecutar la librería que se instala en el mismo grupo de programas de MySQL (figura 4.2.23).



Figura 4.2.23 Ruta para ejecutar MySQL Administrator.

Una vez que MySQL Administrator es ejecutado, nos solicitará los datos necesarios para establecer la conexión en el Servidor de Base de Datos (figura 4.2.24).

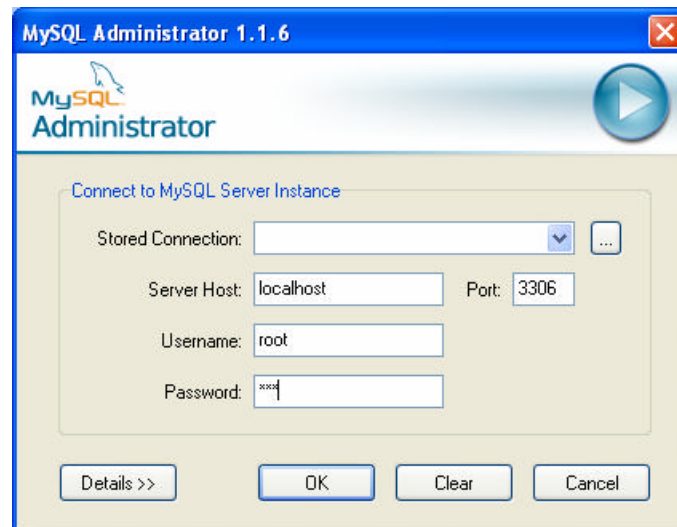


Figura 4.2.24 Conexión al Servidor de Base de Datos.

El primer dato solicitado es el nombre del **Server Host** (Servidor Huésped) en donde se aloja nuestra base de datos. Para nuestro caso el Servidor se llama **localhost** (Huésped Local).

El segundo dato solicitado, corresponde al **Port** (Puerto) de comunicaciones que nuestro Servidor de Base de Datos empleará como vínculo de enlace entre el cliente y la base de datos.

El tercer dato es el **Username** (Nombre de Usuario) en donde colocaremos el nombre del Administrador del Servidor de Base de Datos. Para nuestro caso es root.

El cuarto dato necesario para establecer la conexión es el **Password** (Contraseña) del usuario administrador (root).

Una vez introducidos estos datos procederemos a dar clic en el botón **OK** (Aceptar) para confirmar la conexión.

El Servidor de Base de Datos procederá a identificar al usuario, si es correcto entonces mostrará el Panel de MySQL Administrator (figura 4.2.25).

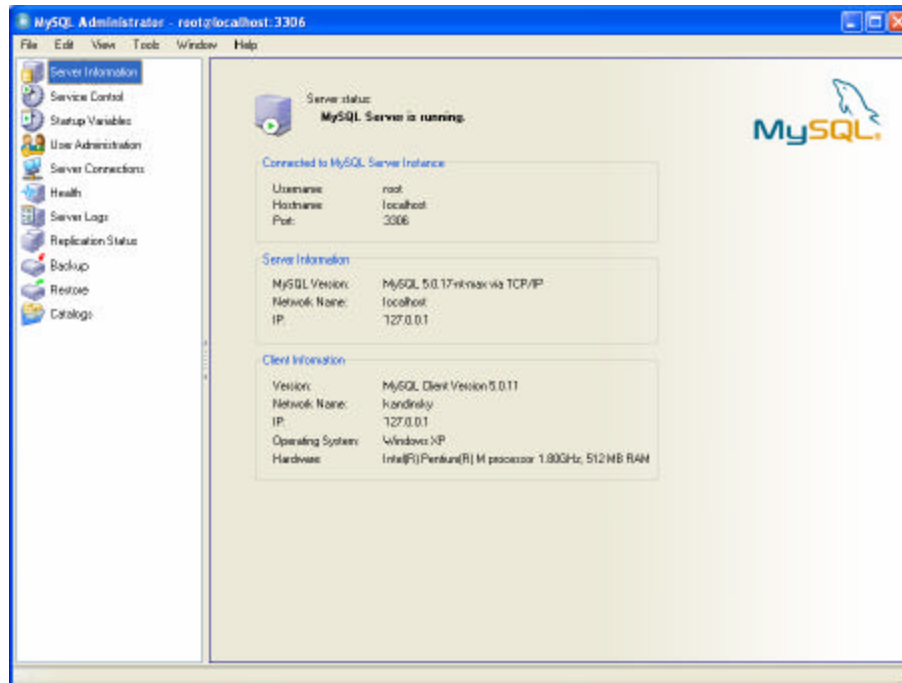


Figura 4.2.25 Panel de MySQL Administrator.

En la parte izquierda inferior del Panel, encontraremos la opción **Catalogs** (Catálogos), la cual nos permitirá crear una nueva base de datos. Dar un clic en esta opción para desplegar la lista de bases de datos (figura 4.2.26).

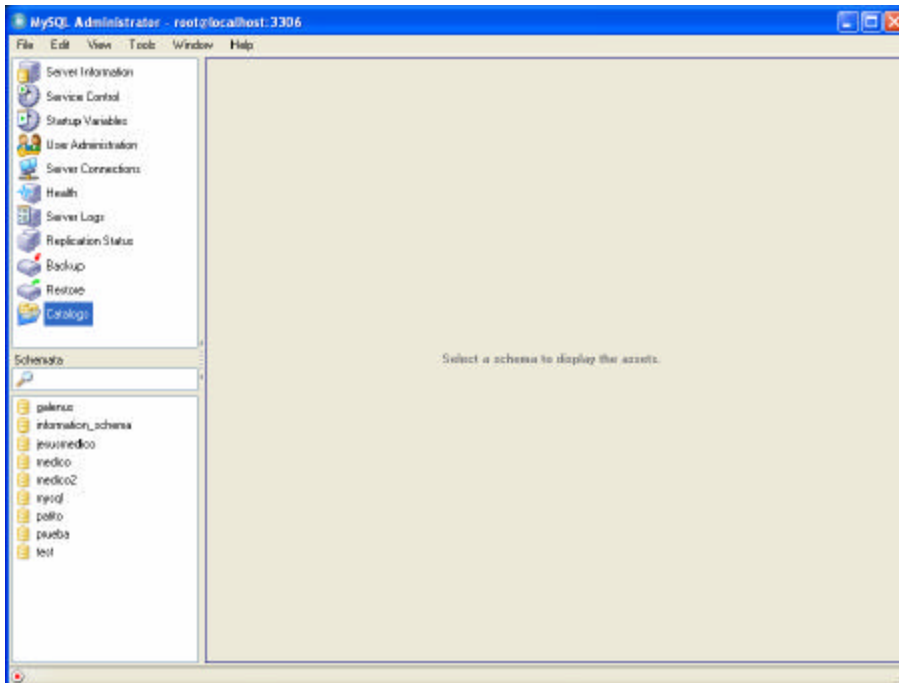


Figura 4.2.26 Lista de Bases de Datos.

Una vez desplegada la lista de bases de datos, posicionar el ratón sobre ésta y dar un clic con el botón izquierdo del ratón para desplegar el menú de contexto. Seleccionar la opción **Create New Scheme** (Crear Nuevo Esquema) para crear la nueva base de datos (figura 4.2.27).

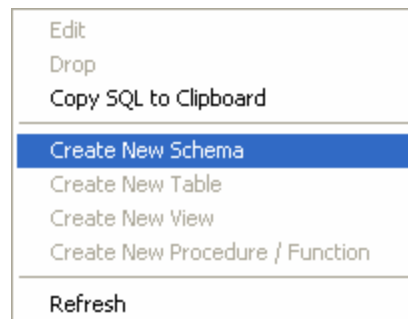


Figura 4.2.27 Menú de contexto para crear nuevo esquema.

Una vez seleccionada la opción, aparecerá una pantalla en donde se nos solicitará que introduzcamos el nombre de la base de datos (figura 4.2.28).

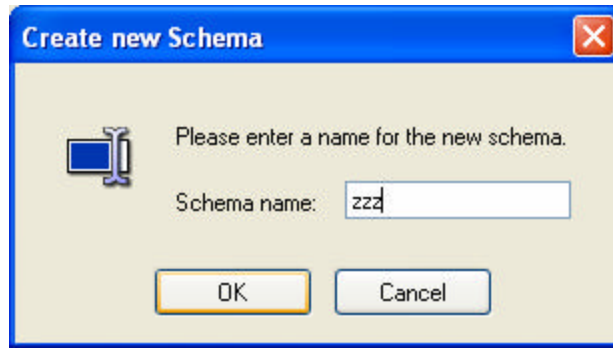


Figura 4.2.28 Solicitud del nombre de la base de datos.

Indicaremos el nombre de la base de datos y daremos clic en el botón OK. Observaremos que nuestra base de datos aparecerá dentro de la lista de bases de datos (figura 4.2.29).

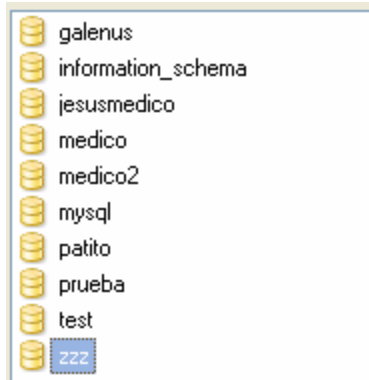


Figura 4.2.29 Lista de Bases de Datos.

Como Crear una Tabla en MySQL Administrator

Para poder crear una tabla, debemos primero seleccionar la base de datos en donde deseamos colocarla, dando un clic. Una vez seleccionada la base de datos, el panel que se encuentra del lado derecho a la lista de base de datos será visible para nosotros y nos mostrará la lista de tablas que constituyen a nuestra base de datos (figura 4.2.30).

Table Name ▲	Engine	Rows	Data length	Index length	Update time
cita	MyISAM	3	57 B	4 kB	2006-10-03 12:34:50
especialidad	MyISAM	1	64 B	2 kB	2006-10-03 15:48:34
grupo	MyISAM	3	88 B	2 kB	2006-10-03 12:34:50
historia	MyISAM	0	0 B	1 kB	2006-10-03 12:34:50
historialog	MyISAM	0	0 B	1 kB	2006-10-03 12:34:50
hojaref	MyISAM	7	296 B	4 kB	2006-10-03 12:34:50
hmedico	MyISAM	2	27 B	3 kB	2006-10-06 18:27:47
medico	MyISAM	0	192 B	4 kB	2006-10-06 18:27:47
nota	MyISAM	6	460 B	4 kB	2006-10-03 12:34:50
paciente	MyISAM	3	188 B	2 kB	2006-10-03 12:34:51
receta	MyISAM	5	228 B	4 kB	2006-10-03 12:34:51
turno	MyISAM	2	36 B	2 kB	2006-10-03 12:34:51
usuario	MyISAM	5	292 B	3 kB	2006-10-03 12:44:19

Figura 4.2.30 Lista de tablas de una base de datos.

Dar clic en el botón **Create Table** (Crear Tabla) que se encuentra en la parte inferior de la lista de las tablas, inmediatamente después aparecerá el formulario que nos permite definir los campos que la conformarán, el tipo de dato del campo, el campo(s) llave, el campo(s) índice(s), etc. (figura 4.2.31).

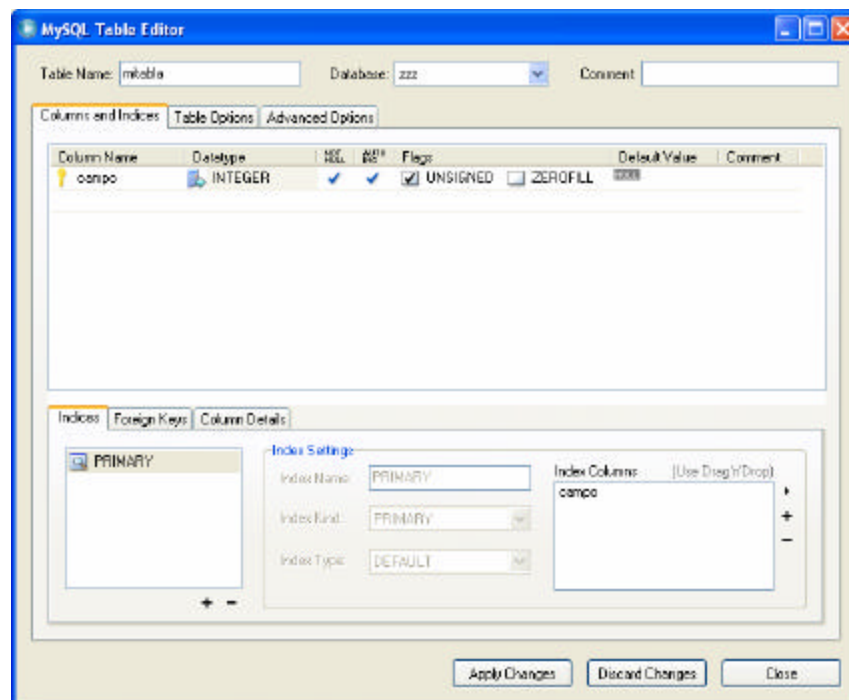


Figura 4.2.31 Pantalla de definición de tabla.

Una vez que definimos los campos e índices que constituirán nuestra tabla, procederemos a dar clic en el botón **Apply Changes** (Aplicar Cambios), si es que deseamos grabarlos, de lo contrario podemos dar clic en el botón **Discard Changes** (Descartar Cambios).

4.3 DISEÑO Y GENERACIÓN DEL FRONT-END

A continuación, la figura 4.3.1 muestra el diagrama de flujo para generar una aplicación en Visual Basic 6.0.

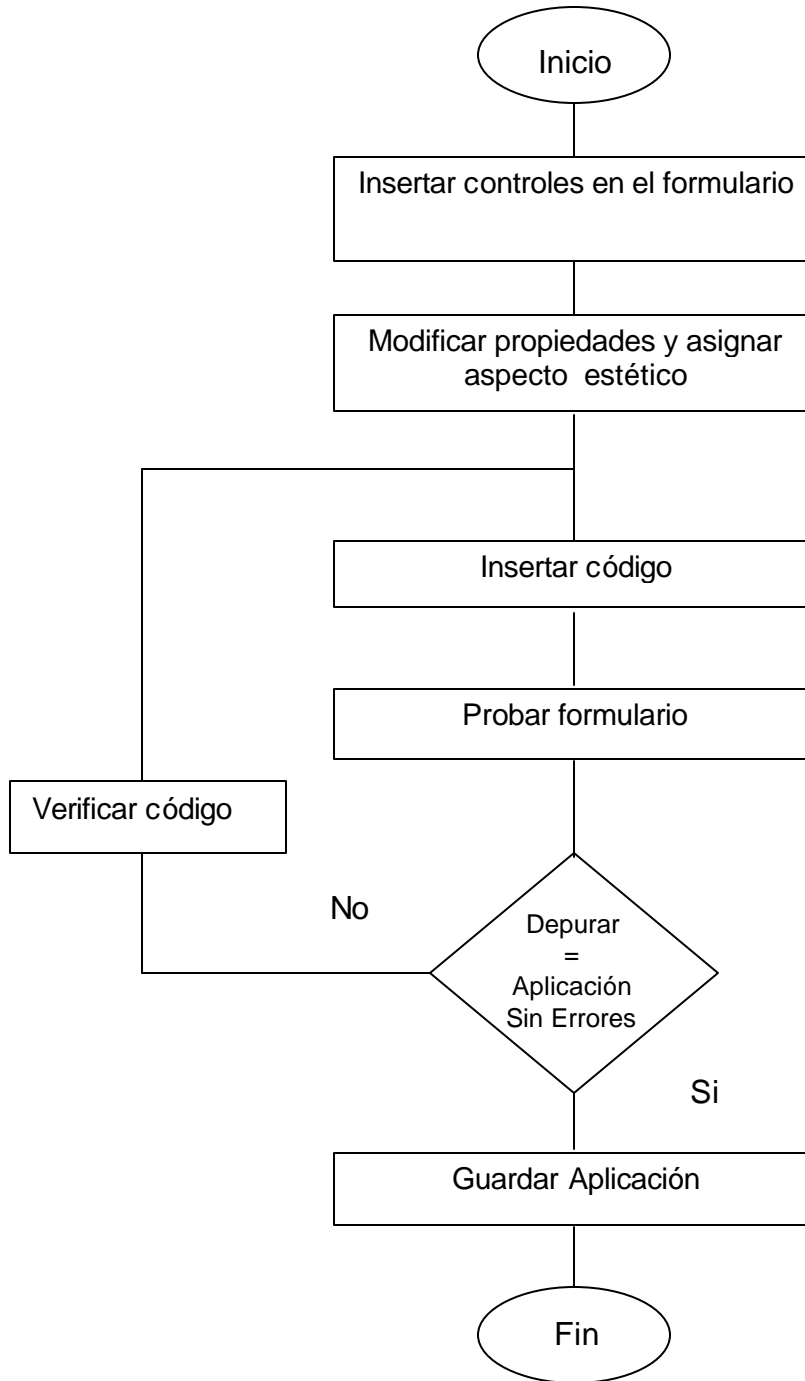


Figura 4.3.1 Diagrama de Flujo de una aplicación generada en Visual Basic 6.0.

Comparación de pantallas y controles de la aplicación

Existen dos componentes fundamentales en la creación de una aplicación desarrollada con un lenguaje orientado a eventos y objetos, los cuales son:

- Ventanas o formas (Form).
- Controles.

Las ventanas son el primer componente necesario para comenzar a crear una aplicación. Este componente tiene la peculiaridad de poder contener o albergar controles tales como:

- Textbox (caja o cuadro de texto).
- Label (etiqueta).
- Toolbar (barra de herramientas).
- Checkbox (caja de verificación).
- CommandButton (botón de pulsación).
- Combobox (cuadro o caja de lista).
- OptionButton (botón de opción).
- MSHFlexgrid (mallas o grid).
- SSTabed (folder).
- Calendar (control calendario).
- Frame (contenedor de controles).

A continuación se describe el uso de los controles empleados para el desarrollo del sistema.

Textbox. Este control es empleado para solicitar al usuario del sistema datos que son requeridos en cada módulo. Un ejemplo del uso de estos objetos lo podemos ver en la pantalla de acceso a SISAM (ver figura 4.3.1) en donde se solicita al usuario que introduzca su User Name y Password.

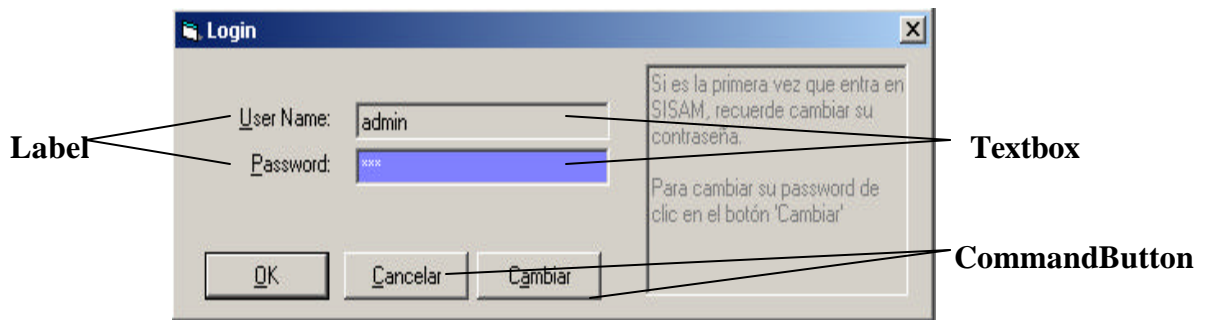


Figura 4.3.1 Pantalla de acceso a SISAM.

Label. Son empleados para describir o rotular algo. En nuestro caso los empleamos para rotular los campos que conforman cada ventana. La figura 4.3.1 muestra un ejemplo de este tipo de control.

Commandbutton. Este control permite ejecutar una determinada acción o proceso a través de un evento, tal como dar un clic sobre él, alejar el botón de él, etc. Un ejemplo de su uso lo podemos ver en la figura 4.3.1.

Frame. Este objeto es empleado para agrupar, contener y brindar un aspecto visual agradable. Un ejemplo del empleo de este objeto lo podemos ver en la pantalla de citas que ilustra la figura 4.3.2.

Toolbar. Es una barra que contiene botones que ejecutan una determinada acción o tarea al momento de dar un clic sobre la opción elegida. Esta permanece a la vista del usuario de manera fija, permitiendo su fácil acceso. Un ejemplo del uso de un toolbar lo podemos ver en la figura 4.3.2 en donde cada botón de la barra ejecuta un determinado módulo del sistema.

Calendar. Permite interactuar con el usuario de manera sencilla para seleccionar una fecha en específica. Este control es empleado en la pantalla de citas que ilustra la figura 4.3.2.

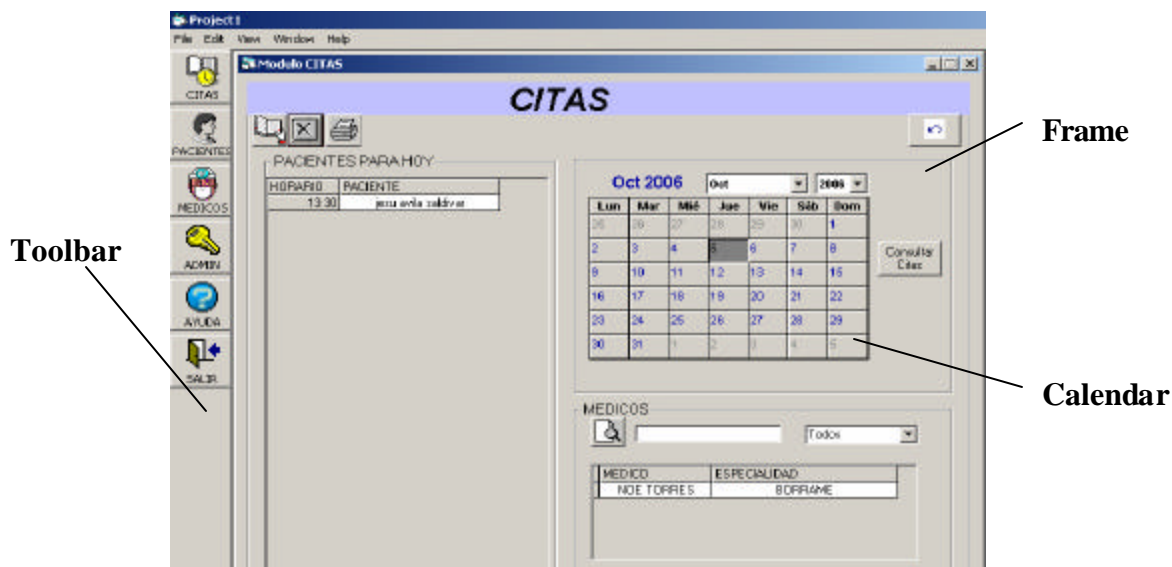


Figura 4.3.2 Pantalla de Citas.

MSHFlexgrid. Este control permite presentar los datos extraídos de la conexión de una base de datos en un formato de filas y columnas, permitiendo consultar más de un solo registro a la vez. Este objeto es empleado para mostrar los expedientes dentro de la pantalla de búsqueda de pacientes. Ver figura 4.3.3.

ComboBox. Es una caja que oculta una lista de opciones posibles. Es desplegada la lista al momento de dar clic sobre el control para seleccionar una opción. Este control se muestra en la figura 4.3.3.

CheckBox. Es una casilla de verificación empleada para seleccionar o no una opción. Para el caso de la pantalla de búsqueda de paciente con cita mostrada en la figura 4.3.3, nos permite elegir o no la opción buscar solo aquellas citas programadas en el día o todas.

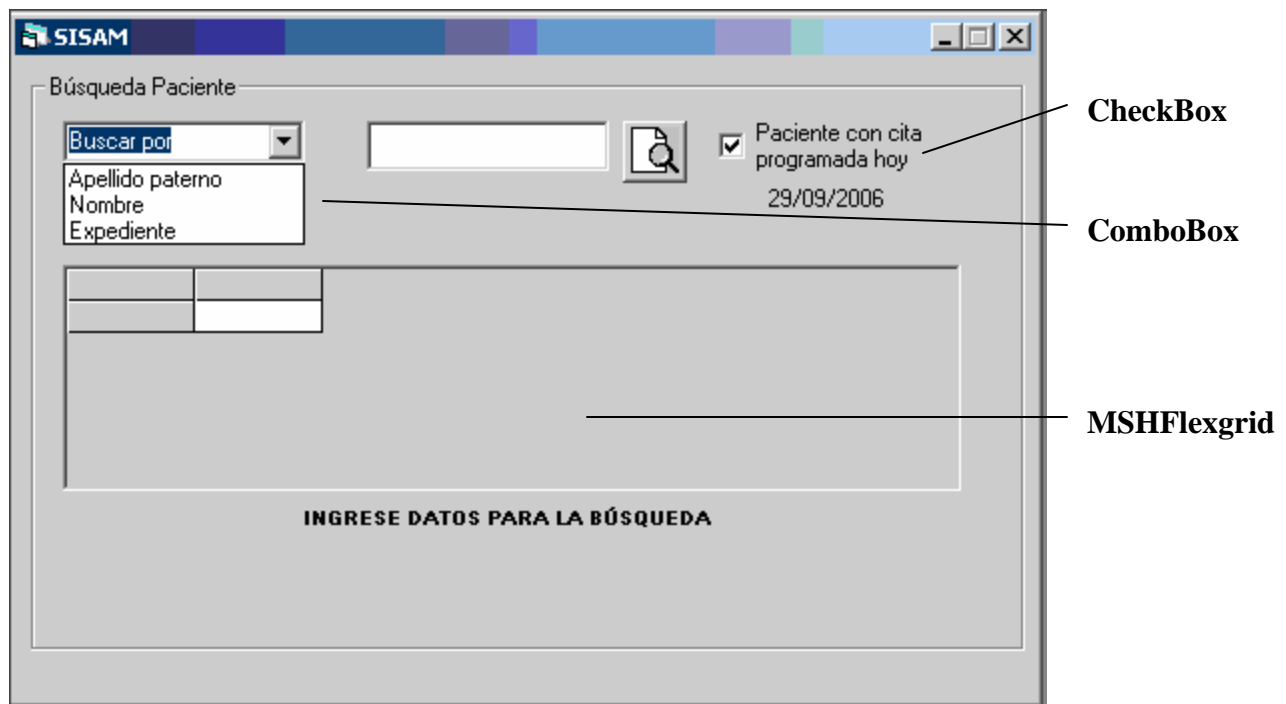


Figura 4.3.3 Pantalla de Consulta de Paciente.

SSTabed. Este control nos permite separar de manera ordenada la información y tiene el aspecto de folder. Para nuestro caso, la figura 4.3.4 muestra la pantalla del módulo de pacientes en donde existen diversos apartados que conforman el expediente del paciente.

OptionButton. Es empleado para seleccionar solo una opción de un conjunto de posibilidades. . Para nuestro caso, la figura 4.3.5 muestra la pantalla del módulo de pacientes en donde se hace emplea para determinar si el interrogatorio es de manera directa o indirecta.

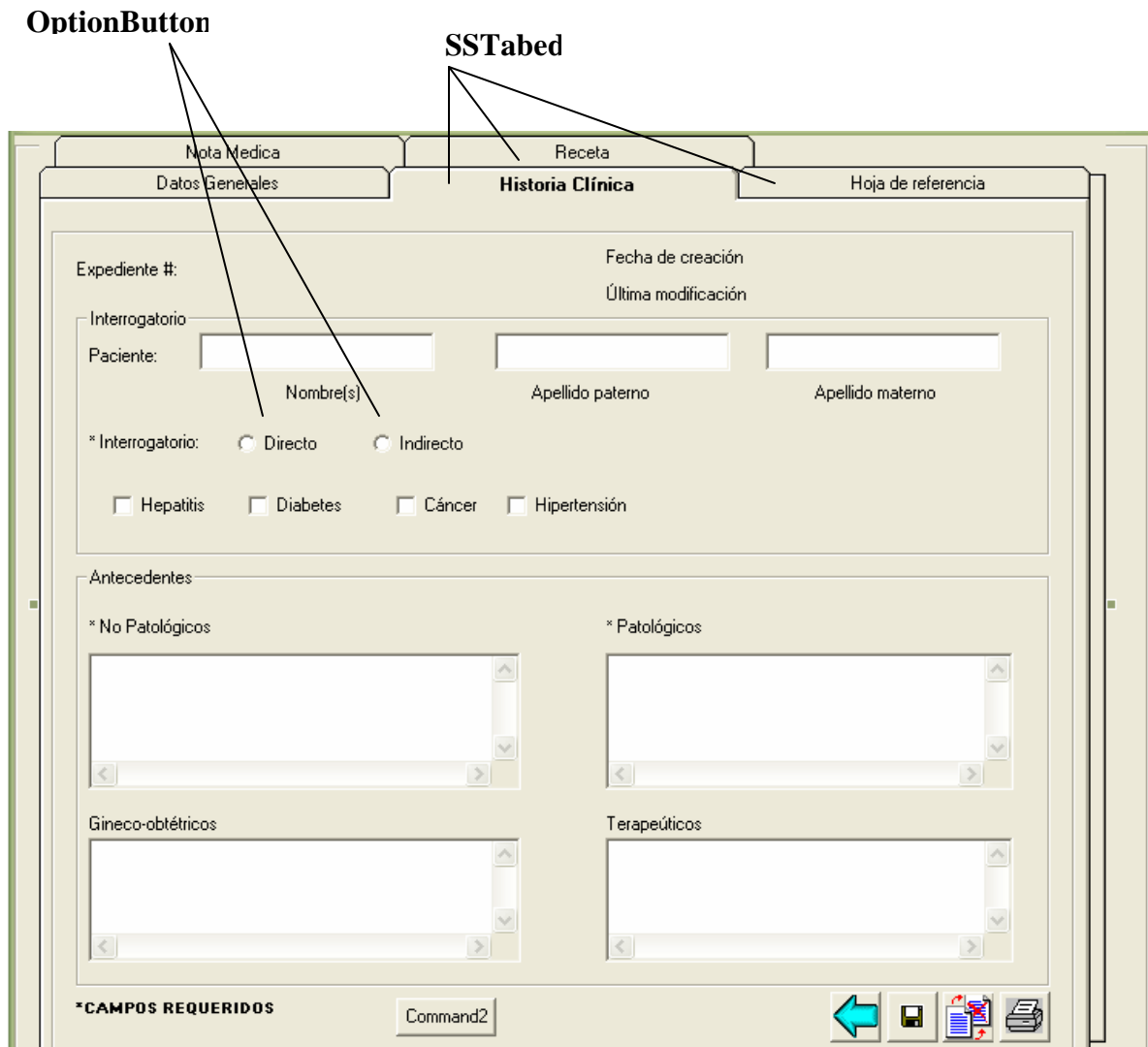


Figura 4.3.5 Pantalla de Pacientes.

El uso de los controles descritos en este apartado, fueron empleados para permitir una correcta distribución de la información que es mostrada en cada uno de los formularios que constituyen los módulos del SISAM, permitiendo con esto que sea una interfaz visual agradable para el usuario.

4.4 INTEGRACIÓN Y PRUEBAS DEL SISTEMA

Las pruebas constituyen una parte integral y vital del desarrollo del sistema. Se realizan con el propósito de descubrir defectos y se establecen para mejorar la calidad del sistema.

Las pruebas requieren que se descarten las ideas acerca de lo correcto que es el software desarrollado y que al descubrir los errores, se logre superar cualquier conflicto en el sistema.

Los criterios que se utilizan para la realización de los casos de pruebas del sistema, permiten detectar los errores y poder así proseguir con la corrección de los mismos. A continuación se lista los diferentes tipos de prueba:

- **Pruebas de caja negra (Funcional):** Esta prueba es un enfoque para probar si las especificaciones del componente probado son utilizadas para derivar casos de prueba. El componente es una caja negra y su comportamiento solo puede ser determinado al estudiar sus entradas y las salidas relacionadas a éstas. Son un enfoque complementario a las pruebas de caja blanca que Intentan encontrar errores de las siguientes categorías:
 - Funciones incorrectas o faltantes.
 - Errores de interfase.
 - Errores de estructura de datos o accesos de BD.
 - Errores de eficiencia.
 - Inicialización y/o terminación.

- **Pruebas de caja blanca (Estructural):** Este tipo de prueba es el contraste al de la caja negra, debido a que la persona que realiza la prueba puede analizar el código y también puede utilizar su

conocimiento acerca del código y de la estructura de un componente para poder obtener los datos de prueba. Los casos derivados pueden ser:

- Garantizar que todas las trayectorias independientes dentro de un módulo, hayan sido ejecutadas dentro de éste al menos una vez.
- Ejecutar todos los lados de las decisiones lógicas.
- Ejecutar todos los ciclos en sus límites operacionales.
- Ejecutar las estructuras de datos internas para asegurar su validez.

La figura 4.4.1 muestra el diagrama de las pruebas de caja blanca y caja negra.

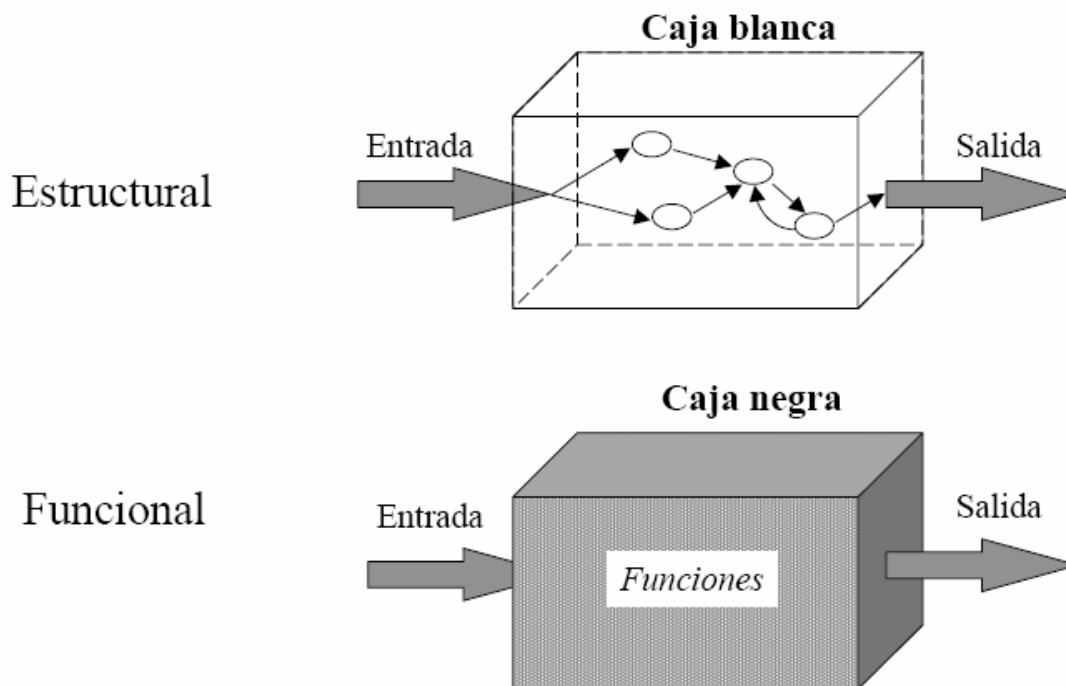


Figura 4.4.1 Diagrama de las pruebas de Caja Blanca y Caja Negra.

- **Pruebas unitarias o de módulo.** Estas pruebas enfocan el esfuerzo de verificación en las unidades más pequeñas de diseño de SW, es decir, el módulo, probando los caminos de control importantes con el fin de descubrir errores dentro de los límites de éste.

- **Pruebas de validación o aceptación.** La información contenida en una sección de la especificación de los requerimientos de software, forma la base para un enfoque de pruebas de validación. La validación del SW se logra por medio de las pruebas de caja negra que demuestran la conformidad con los requerimientos del usuario.

- **Pruebas ALFA y BETA.** Las pruebas Alfa son llevadas a cabo por el usuario en el lugar del desarrollo, donde el desarrollador participa como observador. Las pruebas Beta se llevan cabo por los usuarios finales, en el lugar de trabajo de éstos, y el desarrollador generalmente no está presente.

- **Pruebas de stress.** Las pruebas de stress ayudan a simular casos en los que el número de clientes o la recuperación masiva de datos de una base de datos aumentan. De este modo es posible evaluar tanto el tiempo de respuesta de un sistema como su capacidad de responder ante esos casos.

- **Pruebas de integración.** Es una técnica sistemática para construir la estructura del programa, mientras que al mismo tiempo se llevan a cabo las pruebas para detectar errores asociados con la interacción. Los tipos fundamentales de integración son:
 - **Integración incremental.** Se combina el siguiente módulo que se debe probar con el conjunto de módulos que ya han sido probados.

 - **Integración no incremental.** Se prueba cada módulo por separado y luego se integran todos de una vez y se prueba el programa completo

- **Ascendente.** Se comienza por los módulos hoja.
 - **Descendente.** Se comienza por el módulo raíz.
- **Pruebas de regresión.** Cada vez que se añade un módulo nuevo como parte las pruebas de integración, el SW cambia: se establecen nuevos caminos en el flujo de datos, pueden existir nuevas I/O, y se invoca a una nueva lógica de control, lo cual puede ocasionar problemas con funciones que ya trabajaban correctamente. Las pruebas de regresión consisten en volver a ejecutar un subconjunto de pruebas que se han llevado a cabo anteriormente, para asegurarse que los cambios no han ocasionado efectos colaterales indeseados.

Pruebas aplicadas al sistema

Para la validación de Usuarios, se dejaron los campos de Usuario y Password vacíos por lo que a la hora de aceptar datos nos genera un error. Ver figura 4.4.2.

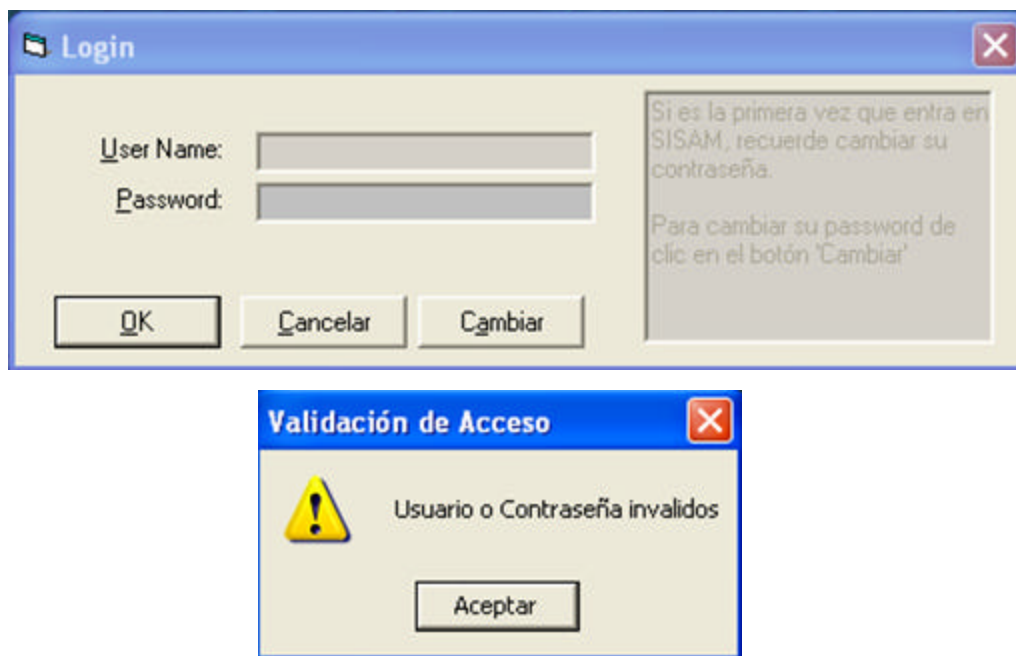


Figura 4.4.2 Error al dejar los campos en blanco.

De las pruebas que se realizaron para la validación de los datos fue el de introducir datos no permitidos en algunos campos, esto es, el de introducir letras en los campos que solo aceptan números, como es el caso de de los números telefónicos.

Como se nos muestra en la figura 4.4.3 en un principio en los campos nos aceptaba todo tipo de datos, y esto se corrigió por código indicando que solo se acepten ciertos caracteres ya sea números o letras dependiendo del campo. Ver la siguiente figura 4.4.3.

The screenshot shows a software window titled 'Project1' with a menu bar (File, Edit, View, Window, Help) and a sidebar with icons for CITAS, PACIENTES, MEDICOS, ADMIN, AYUDA, and SALIR. The main window is titled 'Módulo PACIENTES' and has three tabs: 'Nota Medica', 'Receta', and 'Historia Clínica'. The 'Datos Generales' tab is active, showing a form for patient data. The 'Datos Paciente' section includes fields for 'Paciente:' (containing '55555', circled in red), 'miranda' (Apellido paterno), and 'alvarado' (Apellido Materno). Below this are fields for 'Lugar de nacimiento' (df), 'Fecha de nacimiento' (7/5/1981), 'Sexo' (masculino), 'Edad' (25), 'Estado civil' (Seleccione opción), and 'Ocupación' (analista). The 'Domicilio' section includes fields for 'Calle y número' (w), 'Colonia' (sda), 'Delegación' (asd), 'CP' (222222), 'Estado' (Seleccione opción), 'Tel. casa' (1234), 'Tel. cel.' (123), 'Tel. trabajo' (123), 'correo electrónico', and 'Religion'.

Figura 4.4.3 Campo que acepta cualquier tipo de dato.

Otra de las pruebas que se realizaron al sistema fue la introducir en algunos campos letras, cuando sólo esta permitido la introducción de números, en este caso fue el campo de teléfono casa y esto mismo sucede con campos similares. Ver la figura 4.4.4.

The screenshot displays a medical system interface with a patient data form. The form is divided into sections: 'Datos Generales', 'Historia Clínica', and 'Hoja de referencia'. The 'Datos Generales' section includes fields for patient name (sergio, miranda, alvarado), birth date (7/5/1981), sex (masculino), age (25), marital status (Soltero(a)), and occupation (analista). A modal dialog box titled 'Project1' is overlaid on the form, displaying the error message: 'El campo teléfono de casa sólo debe contener números(sin espacios ni guiones)'. The 'Tel. casa' field contains 'eeeee'. Other fields include 'Tel. cel.' (0445578940392), 'Tel. trabajo' (84752123), 'correo electrónico', and 'Religion'. The 'Estado' dropdown is set to 'Chiapas'. At the bottom, there is a legend for '*CAMPOS REQUERIDOS' and a toolbar with navigation and action icons.

Figura 4.4.4 Error al introducir letras en donde solo acepta números.

También el sistema detecta como error cuando al querer guardad no se han introducido todos los datos que no acepta valores nulos. Ver figura 4.4.5.

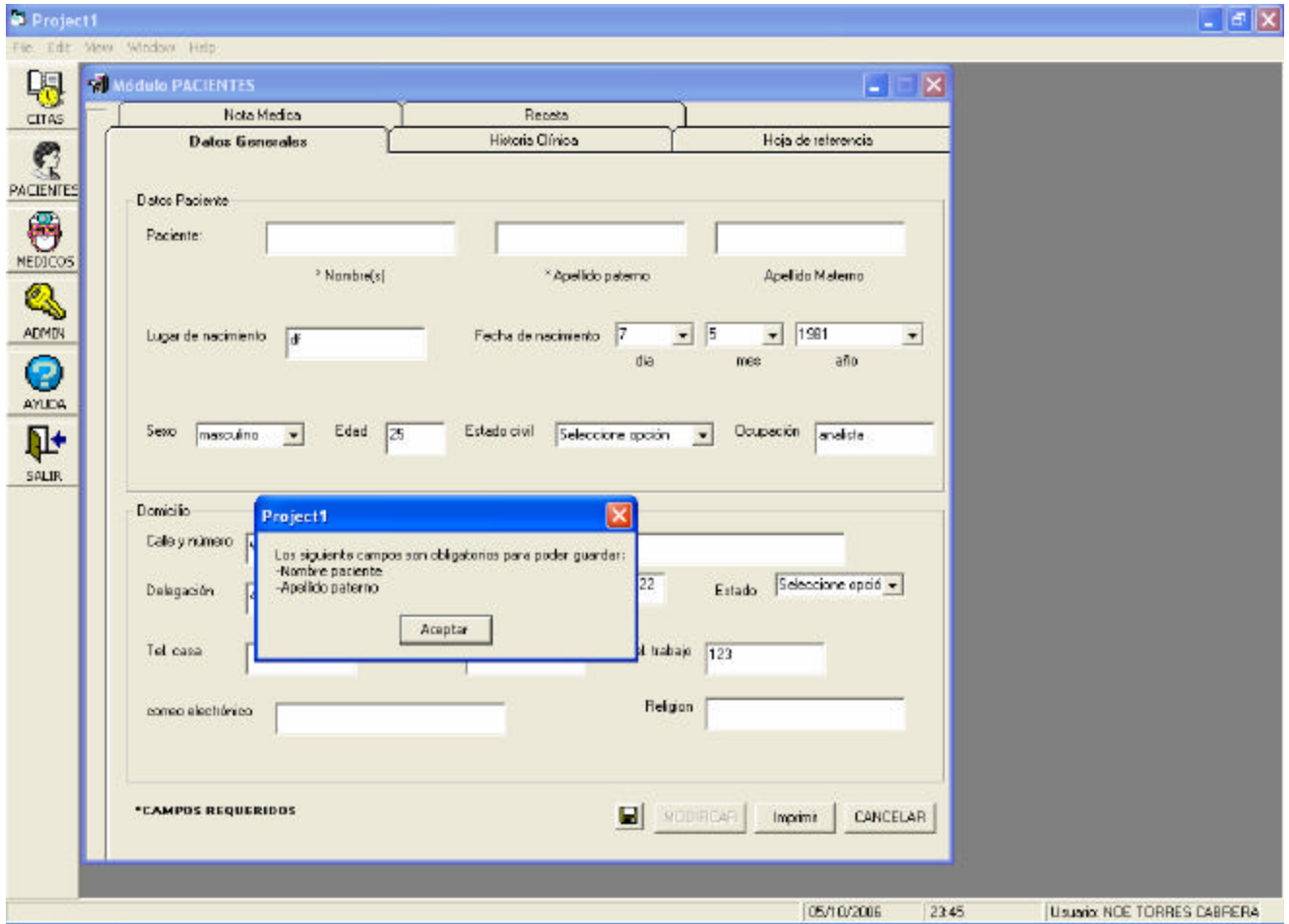


Figura 4.4.5 Error al faltar la introducción de datos.

Otra prueba que se hizo fue seleccionar un doctor y a la hora de querer generar una cita nos muestra que no hay médicos disponibles en esa fecha. Ver figura 4.4.6.

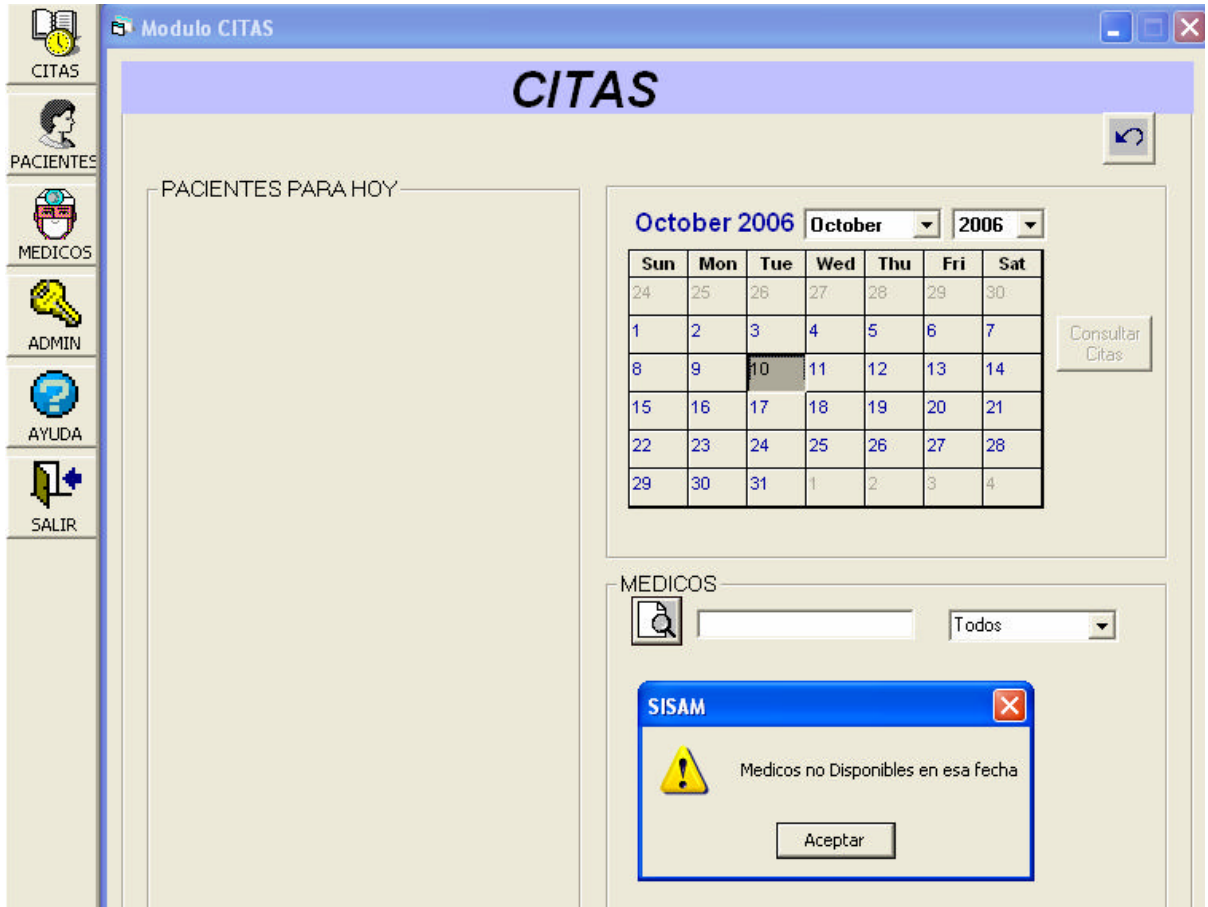


Figura 4.4.6 Muestra la falta de médicos disponibles.

4.5 OBTENCIÓN DE REPORTES Y TIPOS DE MANTENIMIENTO

Tipos de mantenimiento

Se clasifican de la siguiente forma:

- **Mantenimiento perfecto.** Esta actividad se da cuando se usa el software y se reciben nuevas modificaciones a funciones existentes. Este mantenimiento comprende también los cambios solicitados al programador del sistema.
- **Mantenimiento preventivo.** Esta actividad se realiza para modificar el software con la finalidad de hacer mejoras a algún proceso.
- **Mantenimiento adaptativo.** Es la actividad mediante la cual se modifica el software para que las interacciones sean adecuadas en su entorno cambiante. De este mantenimiento se pueden derivar los siguientes mantenimientos:
 - **Mantenimiento aumentativo.** Se da cuando se incluyen nuevas funciones al sistema que surgen de nuevas necesidades del usuario.
 - **Mantenimiento tecnológico.** Se da debido a los avances tecnológicos que se van dando en la informática.
 - **Mantenimiento correctivo.** Es la actividad de corrección de errores detectados en el uso del sistema.

Generación de Reportes

Los reportes presentados en el sistema es la relación de campos almacenados en la base de datos de registro de Citas, Recetas Médicas, Hojas de Referencia y Nota Médica.

Los reportes son documentos internos que indican los diversos procesos que se efectúan durante un día dentro de la clínica u hospital.

Los reportes sirven no solo para uso interno, sino que también son impresos para el uso del Paciente, éstos serían las Recetas en el cual se le indica al paciente como tomar el medicamento, a que hora y la cantidad, así como ciertas recomendaciones.

La creación de reportes es de gran ayuda para cualquier usuario por lo que es conveniente tener un buen control de todos estos reportes.

A continuación se presentan varios reportes generados con el sistema SISAM.

- La figura. 4.5.1 muestra el reporte de Receta Médica.

SISTEMA DE ADMINISTRACION MEDICA
SISAM

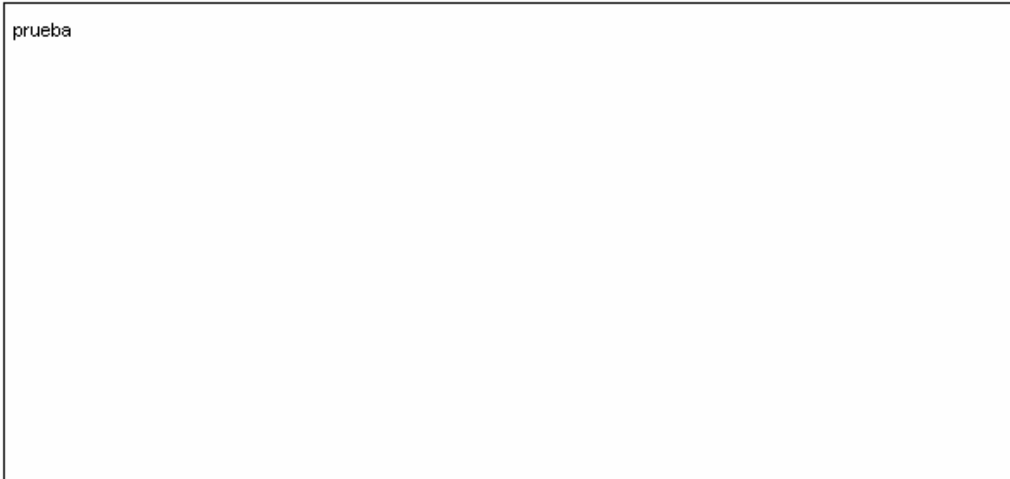
Dr(a). TERESA CARRILLO HUERTA
GINECOLOGIA
Cédula Profesional 2343875

Paciente: sergio miranda alvarado

Fecha: 26/09/2006 10:51:32

Rp:

prueba



Durango #290, Col. Roma

Del. Cuauhtémoc, C.P 05130

Tel(s): 0445598643111
57483920
23945757

Firma: _____

Figura 4.5.1 Receta Médica.

- La figura 4.5.2 muestra el Reporte de Hoja de Referencia

SISTEMA DE ADMINISTRACION MEDICA
SISAM

HOJA DE REFERENCIA

Dr(a). PILAR PEDRAZA SERNA

medico general

Cédula Profesional 657482123

Paciente: Sergio Miranda Alvarado

Fecha: 28/09/2006 23:17:53

Establecimiento al que envía: Clinica 25Motivo por el que se envía: Roptura de Femur

Resumen:

Se lastimo jugando soccer.
Favor de poner yeso.

Durango #290, Col. Roma

Del. Cuauhtémoc, C.P 05130

Firma: _____

Tel(s): 0445519284755

56712345

87412345

Figura 4.5.2 Hoja de Referencia.

- La figura 4.5.3 muestra el listado de las citas que se tienen.

**SISTEMA DE ADMINISTRACION MEDICA
SISAM**

Informe de Citas

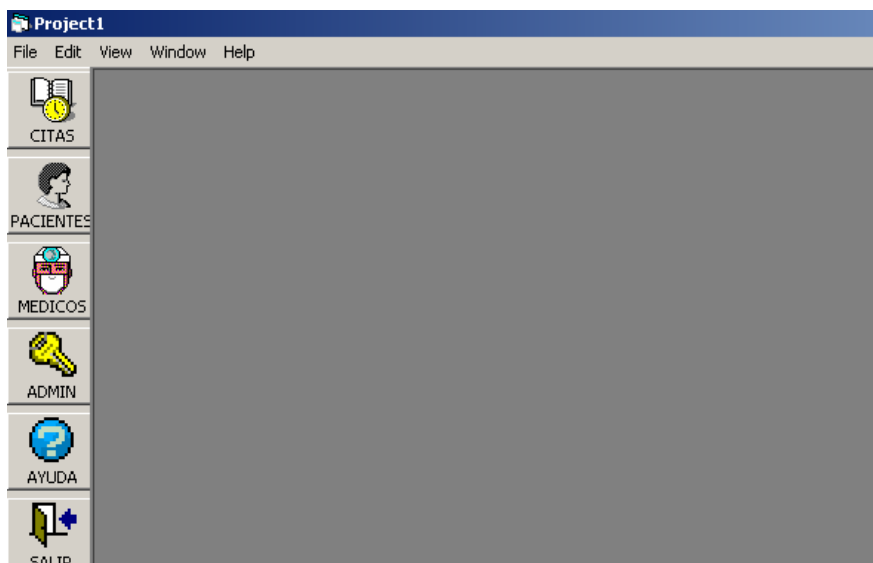
CITA	PACIENTE	MÉDICO	ESPECIALIDAD
Sergio	Miranda Alvarado 28/09/2006	PILAR PEDRAZ	medico general
Sergio	Miranda Alvarado 28/09/2006	PILAR PEDRAZ	medico general
Sergio	Miranda Alvarado 28/09/2006	PILAR PEDRAZ	medico general

Figura 4.5.3 Reporte de Citas.

MANUAL DE USUARIO

MANUAL DE USUARIO

El Sistema de Administración Médica (**SISAM**), es una aplicación enfocada a la administración de información de expedientes electrónicos en una clínica o consultorio médico. Los módulos que lo componen permiten automatizar el proceso de registro de citas, información de los pacientes y sus expedientes electrónicos, así como datos sobre los médicos y la administración de los usuarios como son sus perfiles y permisos de acceso al sistema.



Los cinco módulos que forman al SISAM son:



El módulo de Citas permite automatizar los procesos de alta y cancelación de la cita a un paciente.



El módulo de Pacientes permite crear el expediente electrónico de un paciente, el cual se compone por: datos generales, historia clínica (antecedentes), notas médicas, recetas médicas y hojas de referencia.



El módulo de Médicos permite administrar la información de los médicos: datos generales, especialidad, horario en el que labora.



El módulo Administración como su nombre lo indica administra la información correspondiente a los usuarios: nombre de usuario, clave de acceso y los permisos que cada uno de éstos tendrá para acceder a los demás módulos.



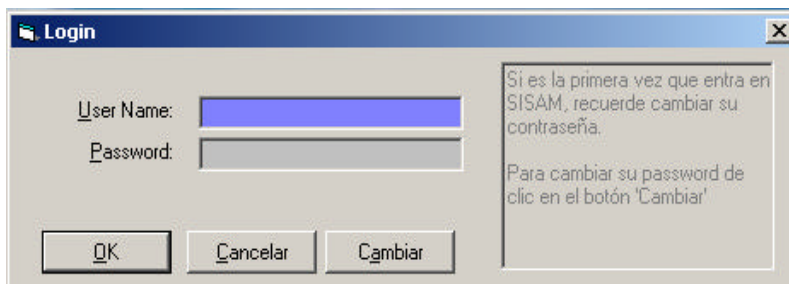
En el módulo de Ayuda, se muestra el manual de usuario, por si se tienen alguna duda en el manejo del sistema.



El botón Salir realiza el proceso de cerrar la aplicación.

SISAM (Sistema de Administración Médica)

Para acceder al sistema se despliega una ventana que nos solicita el nombre de usuario y la contraseña de acceso, si se ingresa incorrectamente los datos que se piden, al generar 3 intentos se cierra la aplicación, si es correcto el usuario y la contraseña mostrará la pantalla principal del sistema, de acuerdo con el perfil del usuario se tendrán determinados permisos para hacer uso del SISAM.



Si es la primera vez que entra al SISAM, se recomienda al usuario que cambie su contraseña, si da clic en el botón “cambiar”, se desplegará la siguiente ventana:



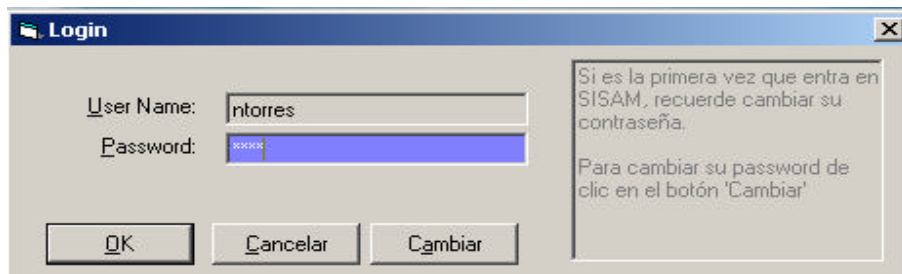
En esta pantalla se solicita la contraseña anterior y la nueva contraseña, así como la confirmación de esta última, si la nueva contraseña no coincide con la de confirmación al dar un clic en aceptar, el sistema mandará el siguiente mensaje:



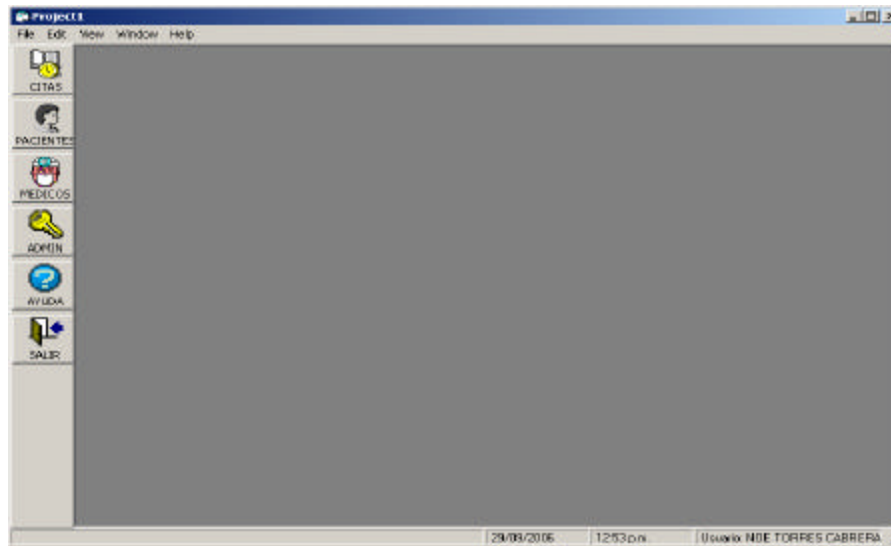
Si se ingresan los datos de manera correcta, se cambiará la contraseña con éxito y regresará a la ventana inicial.



Al haber efectuado el cambio de la contraseña, regresará a la ventana de acceso a SISAM, en donde se deberá capturar el nombre de usuario y la nueva contraseña para ingresar al sistema. Dar clic en el botón OK.



Una vez que el sistema valida que el usuario y contraseña son correctos se mostrará la pantalla principal del sistema, en la cual del lado izquierdo existe un panel que contiene seis botones que ejecutan los módulos del sistema: Citas, Pacientes, Médicos, Administración, Ayuda y Salir.



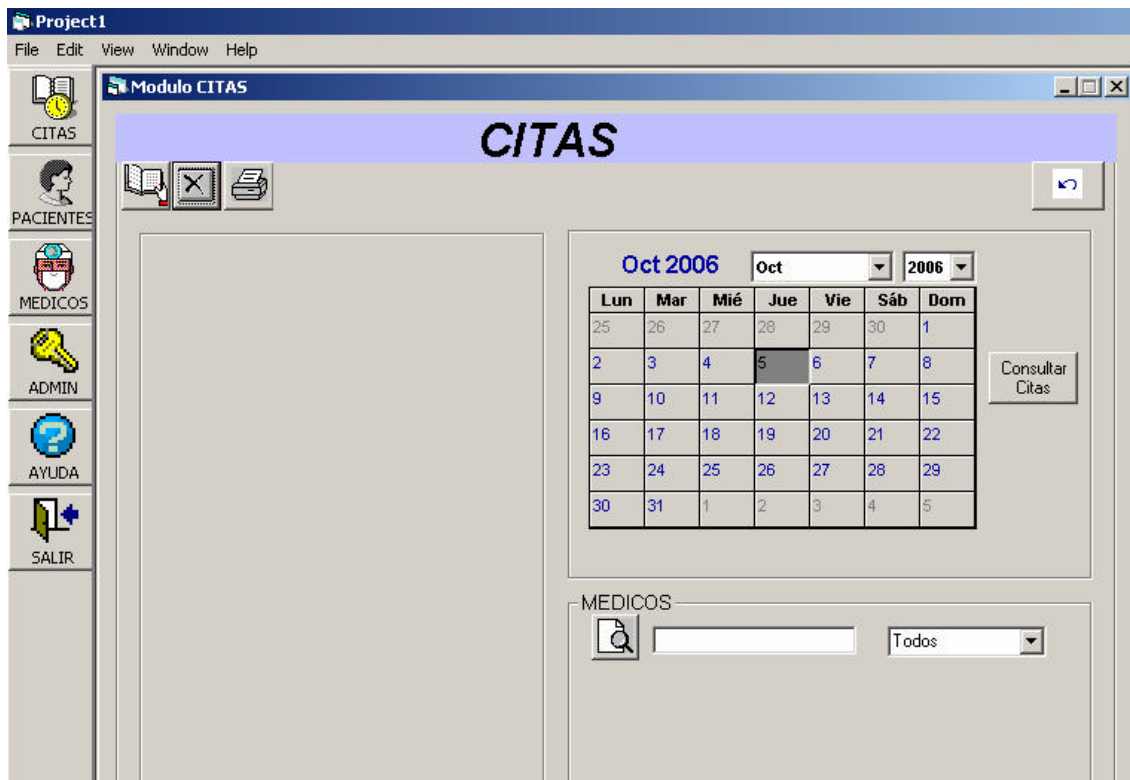
En la parte horizontal se encuentra la barra de estado en donde se indica la fecha actual, la hora y el usuario que ingreso al sistema.

MÓDULO DE CITAS

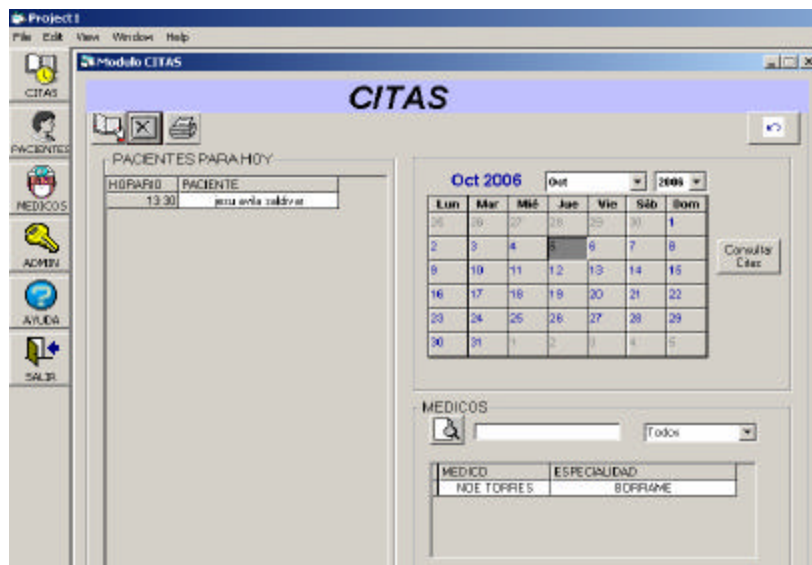
El módulo Citas permite automatizar los procesos de alta y cancelación de la cita a un paciente.



Al dar clic en el botón desde la pantalla principal del SISAM, se muestra la siguiente pantalla que está formada por: Calendario, Médicos, Pacientes.




Si existen citas programadas al día, al seleccionar el módulo de citas se mostrará la siguiente ventana:



Si no existen Citas programadas en la fecha actual, al entrar al módulo de citas desplegará el mensaje :

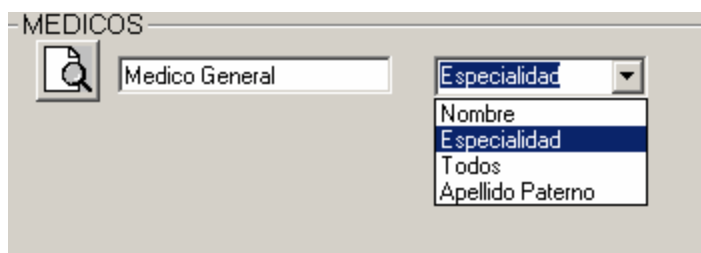


La forma en que se tiene que generar una cita es la siguiente:

1. Dar clic en el botón  **Agregar Cita.**
2. Elegir la fecha en el calendario en la que el paciente desea ser atendido. Dar un clic en el botón “Confirmar fecha”.

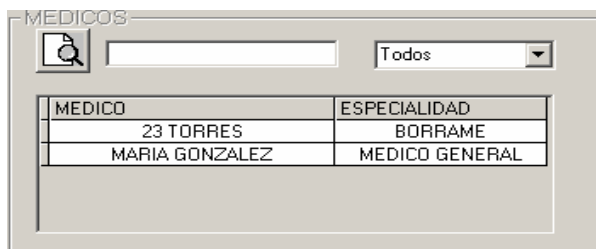


3. Una vez confirmada la fecha, ubicarse en el bloque de médicos para elegir al médico dependiendo de la disponibilidad de los mismos en la fecha elegida para la cita. La consulta de los médicos disponibles en la fecha elegida se puede llevar acabo de cuatro distintas formas: por especialidad, nombre, apellido paterno o todos. Capturar en la caja de texto que aparece en blanco, el dato que desea encontrar dependiendo de la consulta seleccionada.



4. Dar clic en el botón buscar médico 

5. Si hay un medico disponible se mostrará el doctor, de lo contrario mandará un mensaje indicando que no hay un doctor disponible para esa fecha.

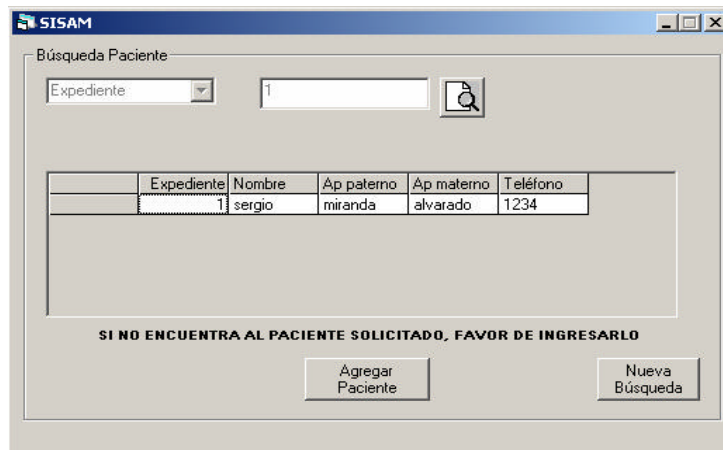


6.-Seleccionar al doctor, dando clic sobre el nombre del mismo, al momento de seleccionar al médico, en el bloque de pacientes se desplegarán los horarios disponibles que tiene en esa fecha.

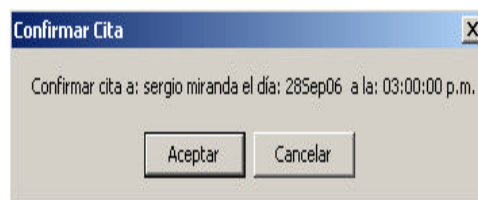


7. Seleccionamos el horario que quiere el paciente para su consulta dando un clic sobre éste, se mostrará una ventana para buscar al paciente. La búsqueda se puede hacer de varias formas ya sea por nombre, apellido paterno, o expediente. Capturar en la caja de texto que aparece en blanco, el dato que desea encontrar dependiendo de la consulta seleccionada.

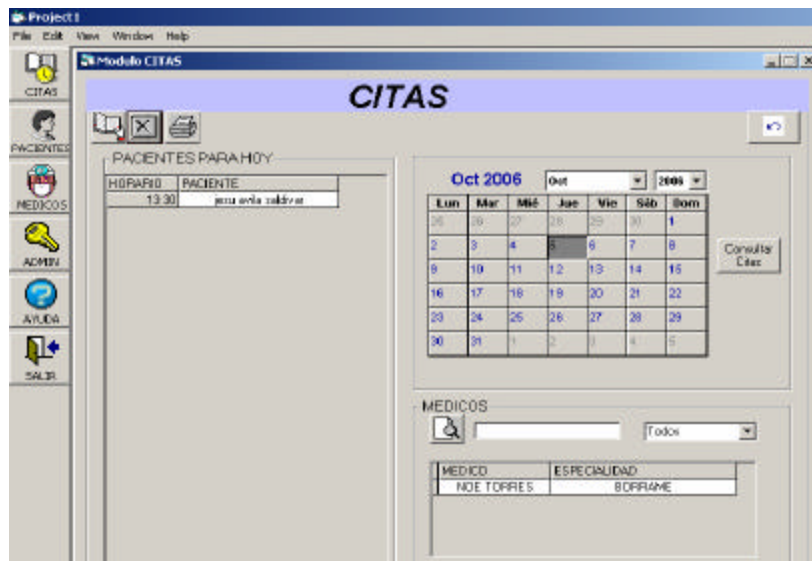
Dar clic en el botón buscar paciente 



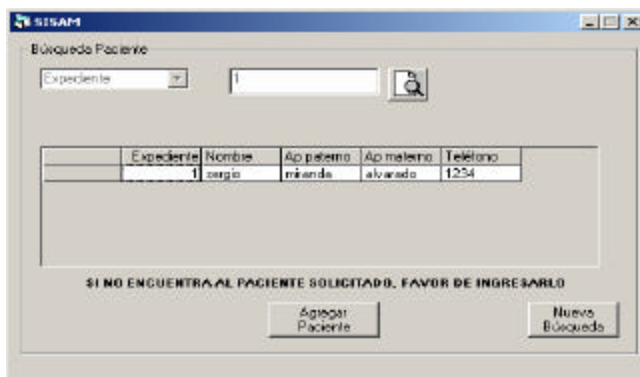
8. Si el paciente fue encontrado, dar clic sobre el paciente y se muestra la confirmación de la cita en la fecha y hora elegida. Presionar el botón “Aceptar” para confirmar la cita.




9. Una vez confirmada la cita se muestra a la pantalla principal del módulo de citas.

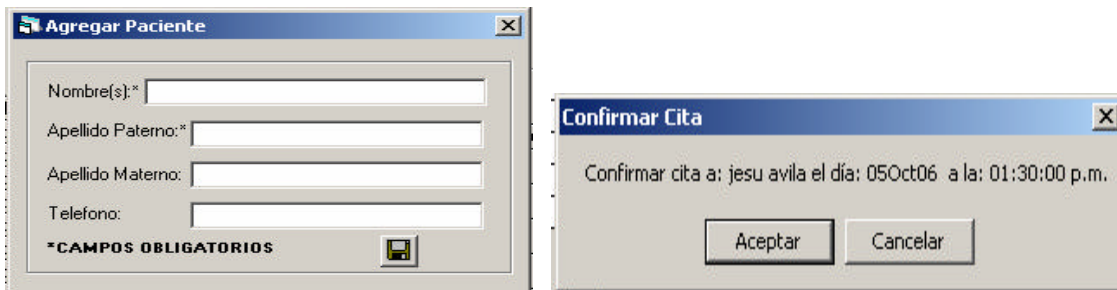



10. En caso de que el paciente acuda por primera vez a consulta, al realizar la búsqueda del paciente no se encontrarán datos y el sistema permitirá capturarlos. Dar clic en el botón “agregar paciente”.



11. Se mostrará una ventana que nos permite ingresar los datos del paciente:


nombre, apellidos, y teléfono y presionamos el botón . Una vez guardados los datos se muestra la pantalla de confirmación de la cita.

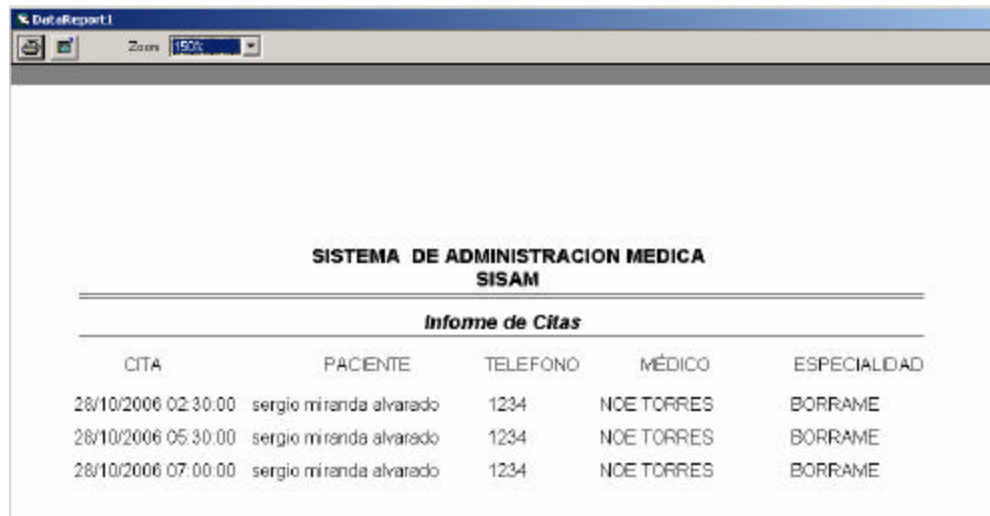



12. Si se quiere cancelar la cita generada, se debe elegir la cita que queremos cancelar y en seguida dar clic en el botón .

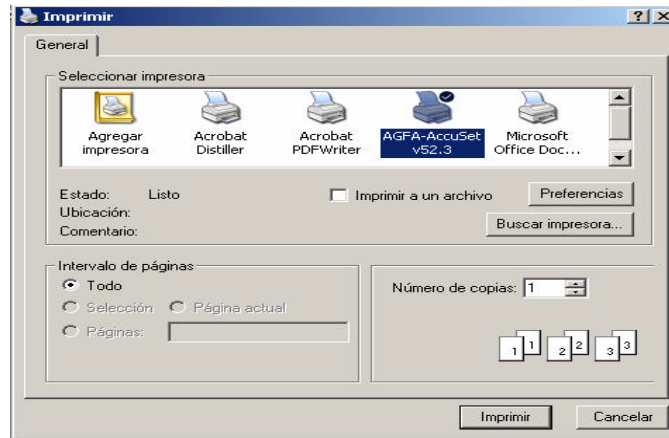
Este modulo también nos permite obtener el reporte de las citas programadas del día, mostrando la información de la fecha, nombre del paciente, nombre del médico y la especialidad del mismo.

Para obtener el reporte debe haber citas programadas en el día,

1. Dar clic en el botón ,
2. Se desplegará el reporte en vista previa:



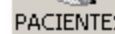
3. Al presionar el botón  que se encuentra en la parte superior izquierda, desplegará una ventana en la cual seleccionamos la impresora en donde se desea imprimir.

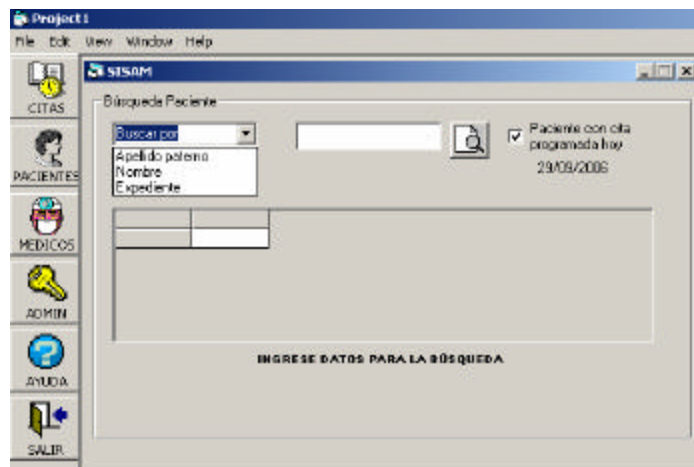



MÓDULO DE PACIENTES

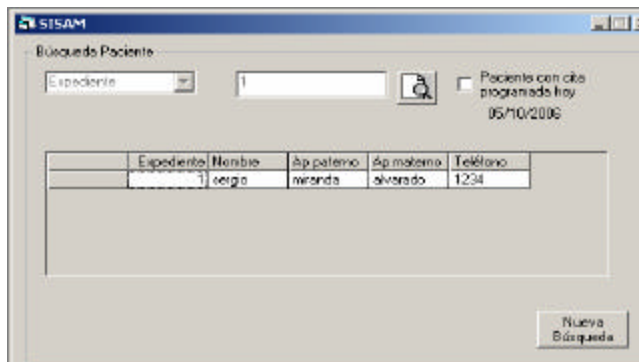
Este módulo permite al usuario generar el expediente electrónico del paciente. Se compone de: datos generales, historia clínica, notas médicas, recetas médicas y hojas de referencia.



1. Para entrar al módulo de Pacientes presionamos el botón  en la pantalla principal de SISAM. Una vez ingresando al módulo se muestra la pantalla que permite al usuario realizar la búsqueda del expediente del paciente, esta búsqueda se puede realizar de varias formas: por expediente, nombre o apellido paterno, una vez elegida el tipo de búsqueda se debe ingresar en la caja de texto en blanco el dato correspondiente al tipo de búsqueda. Para complementar la búsqueda el sistema puede realizarla sólo de los pacientes que tienen cita en la fecha actual marcando la casilla “Paciente con cita programada hoy”, o bien de todos los pacientes registrados en el SISAM para lo cual se debe dejar en blanco la casilla.

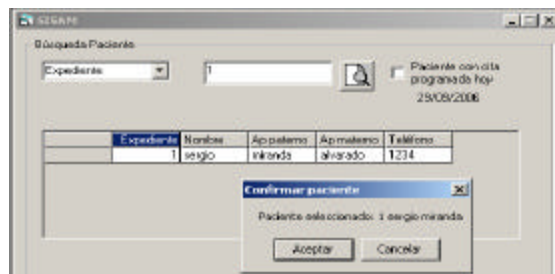


2. Dar clic en el botón  para traer la información solicitada.

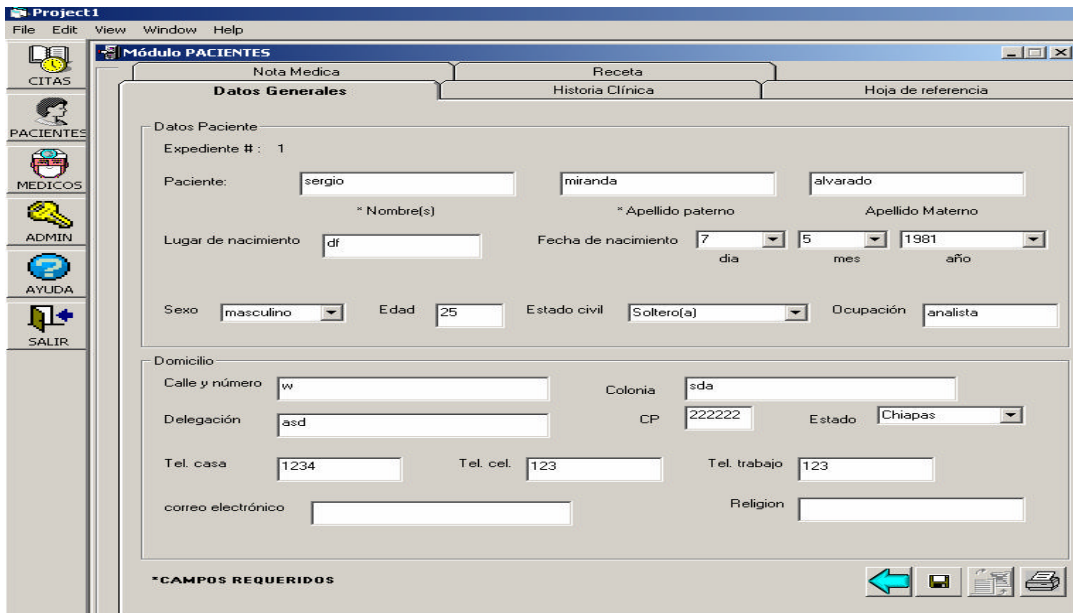


3. Una vez que se muestra la información, dar clic sobre el paciente deseado, se desplegará una ventana de confirmación, si el paciente elegido es el correcto damos clic en aceptar, de lo contrario damos clic en cancelar y seleccionamos el correcto o realizamos nuevamente la búsqueda presionando el botón “nueva búsqueda”.

4. Al seleccionar el paciente se despliega la pantalla siguiente:



5. Una vez confirmado el expediente del paciente con el cual se desea trabajar se mostrará la siguiente pantalla, la cual contiene los 5 elementos que conforman el expediente electrónico:



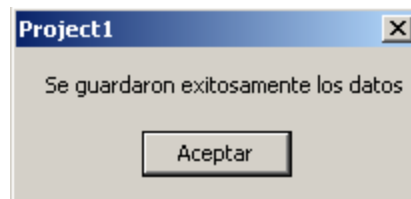
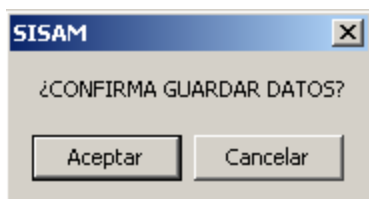
Datos Generales


Este apartado almacena la información del paciente, se divide en datos del paciente, describe los datos personales y domicilio. Es muy importante recalcar que si el paciente asiste por primera vez a la consulta, al confirmar en recepción su cita, la secretaria llevará a cabo la captura de los datos generales para generar su expediente.

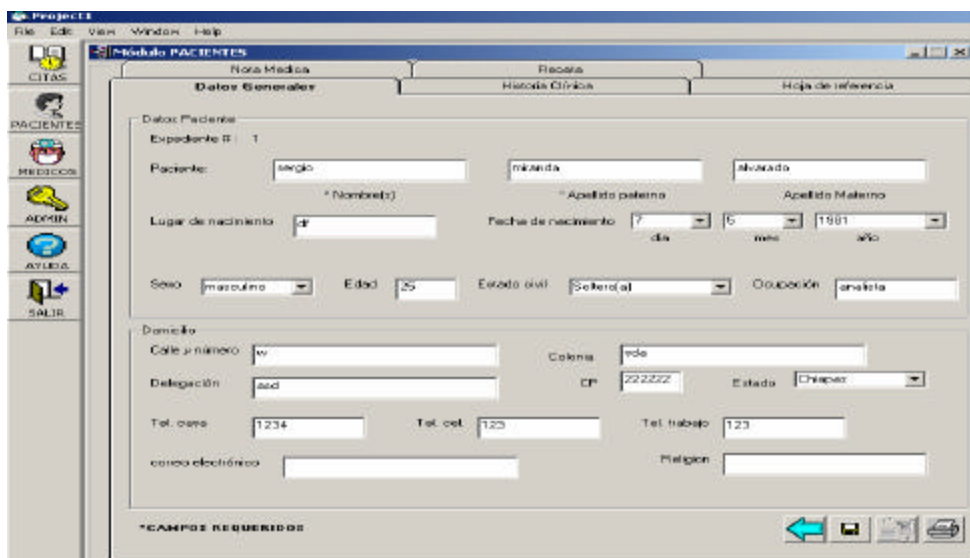
Es importante que cuando se capture la información, no se deje en blanco los campos requeridos, de otra manera no dejará guardar la información y enviará un mensaje indicando que se deben llenar los campos obligatorios. Al terminar de capturar los campos se deberá guardar la información.


Dar clic en el botón .

Se deberá confirmar si desea guardar la información, una vez confirmado se enviará un mensaje indicando que los datos fueron guardados exitosamente. Si se elige cancelar en la pantalla de confirmación los datos no serán guardados.



Para modificar la información almacenada se debe presionar el botón , y se activarán los campos permitiendo la modificación de cada uno de ellos.

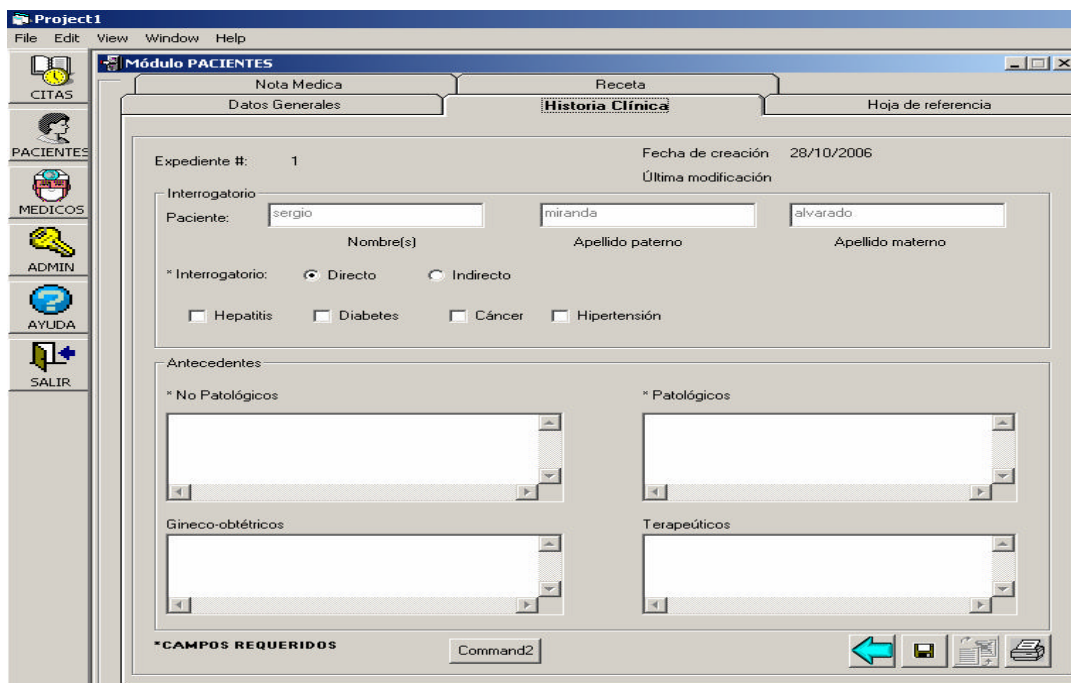


Para salvar los cambios presionar el botón .

Historia Clínica

La historia clínica almacena la información correspondiente a los antecedentes clínicos patológicos y no patológicos del paciente, esta información se obtiene cuando el médico cuestiona directamente al paciente o por alguna circunstancia se obtiene la información por medio de un familiar, amigo, etc.

Esta pantalla esta dividida en dos bloques, Interrogatorio el cual contiene el nombre completo del paciente, y tiene opciones para elegir si el cuestionamiento fue directo e indirecto y padecimientos, el segundo bloque Antecedentes contiene algunos campos para describir los padecimientos patológicos y no patológicos del paciente. Además contiene información como que número de expediente y fecha de creación o modificación de la historia clínica.



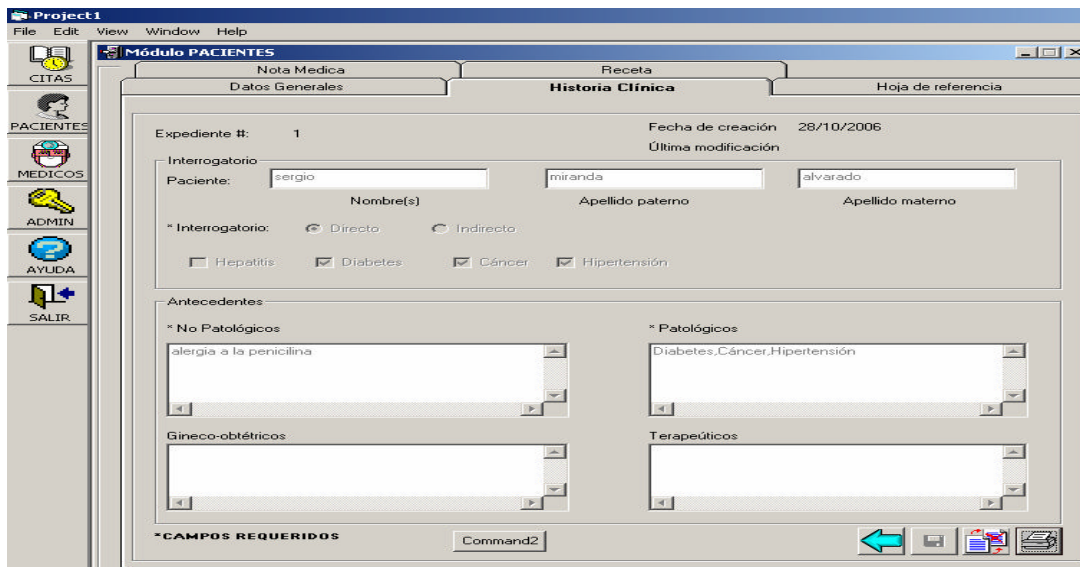
Es importante que cuando se capture la información, no se deje en blanco los campos requeridos, ya que no permitirá guardar la información enviando un


mensaje indicando que se llenen éstos. Una vez capturada la información dar clic


en el botón  .

Modificar Historia Clínica

Si ya se había guardado la información de historia clínica, esta pantalla presentará en cada campo la información que se ingresó, pero estarán deshabilitados, para no permitir escribir sobre ellos

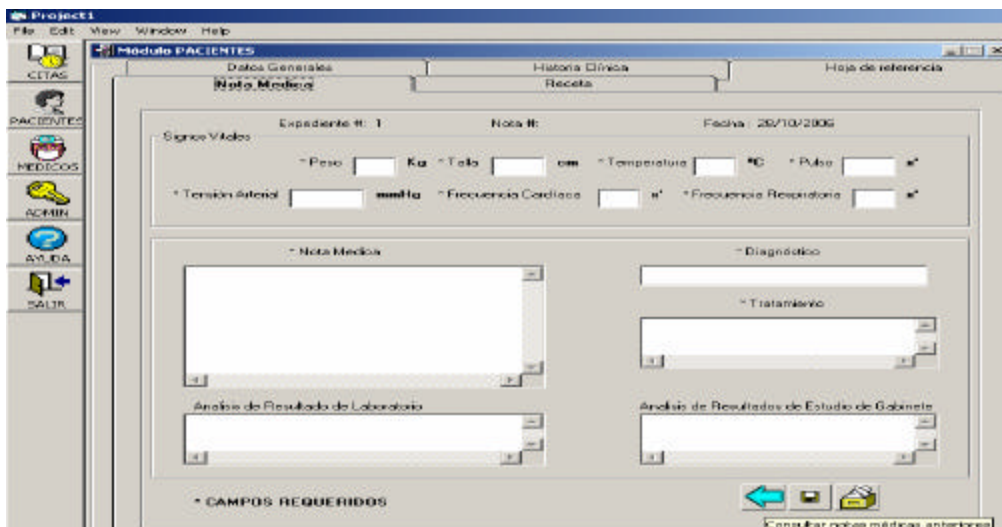


Para modificar la información tenemos que presionar el botón  , el cual activará nuevamente los campos, permitiendo escribir sobre el o los que necesiten corregir.

Para salvar los cambios dar clic en el botón  .

Nota Médica


Las notas médicas son escritas por el médico que realiza la consulta, en éstas se anotan los signos vitales que en ese momento presenta el paciente al ser revisado por la enfermera o el mismo médico, posteriormente éste evalúa y detecta el o los padecimientos encontrados en el paciente y en caso de que se haya practicado algún análisis o estudio la interpretación de los resultados la podrá capturar en los campos asignados en esta parte del expediente.



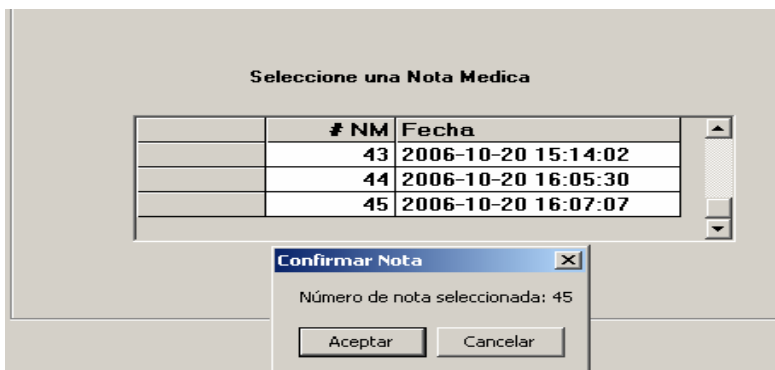
Al igual que en los otros bloques es necesario capturar los campos obligatorios.

Dar clic en el botón .

Consulta de Notas Médicas Anteriores

Para consultar alguna nota médica, presionar el botón , el cual ejecutará una ventana en la que muestra todas las notas existentes por número de nota y por fecha de creación.

Para consultar la nota médica, se da clic en la celda en donde está la nota que se desea, a continuación se mostrará una ventana que indica el número de nota que elegimos y la aceptación, de lo contrario permite seleccionar otra nota.



La información correspondiente a la nota elegida se mostrará en la pantalla principal de Nota Médica con la fecha de creación y hora. Se activarán los botones de terminar consulta, imprimir, consultar nota médica anterior.

The screenshot shows the 'Módulo PACIENTES' window with the 'Nota Médica' tab selected. The form contains the following fields and sections:


- Expediente #: 1, Nota #: 2, Fecha: 26/08/2008 02:21:15 p.m.
- Signos Vitales: * Peso: 25 Kg, * Est: 1.60 cm, * Temperatura: 37 °C, * Pulso: 55 x', * Tensión Arterial: 120 mmHg, * Frecuencia Cardíaca: 60 x', * Frecuencia Respiratoria: 12 x'.
- * Nota Médica: A large text area for entering the medical note.
- * Diagnóstico: A text area for the diagnosis.
- * Tratamiento: A text area for the treatment.
- Análisis de Resultados de Laboratorio: A text area for lab results.
- Análisis de Resultados de Estudio de Gabinete: A text area for office study results.
- * CAMPOS REQUERIDOS: A section indicating required fields.
- Buttons: A blue arrow button, a printer icon, a magnifying glass icon, and a car icon. A tooltip for the printer icon reads 'Consultar notas médicas anteriores'.

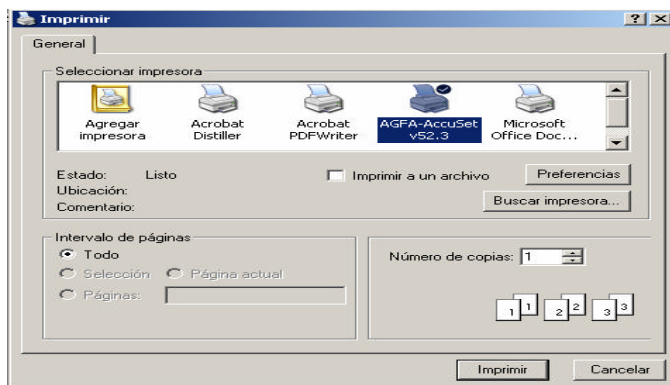
Este apartado también nos permite obtener el reporte de la nota médica generada o de la que se está consultando en ese momento, mostrando la información de la fecha, nombre del paciente, nombre del médico y la especialidad del mismo, además de la información de los signos vitales y la información del bloque de Nota Médica.




1. Dar clic en el botón ,
2. Se desplegará el reporte en vista previa.



3. Al presionar el botón  que se encuentra en la parte superior izquierda, desplegará una ventana en la cual seleccionamos la impresora en donde se desea imprimir.

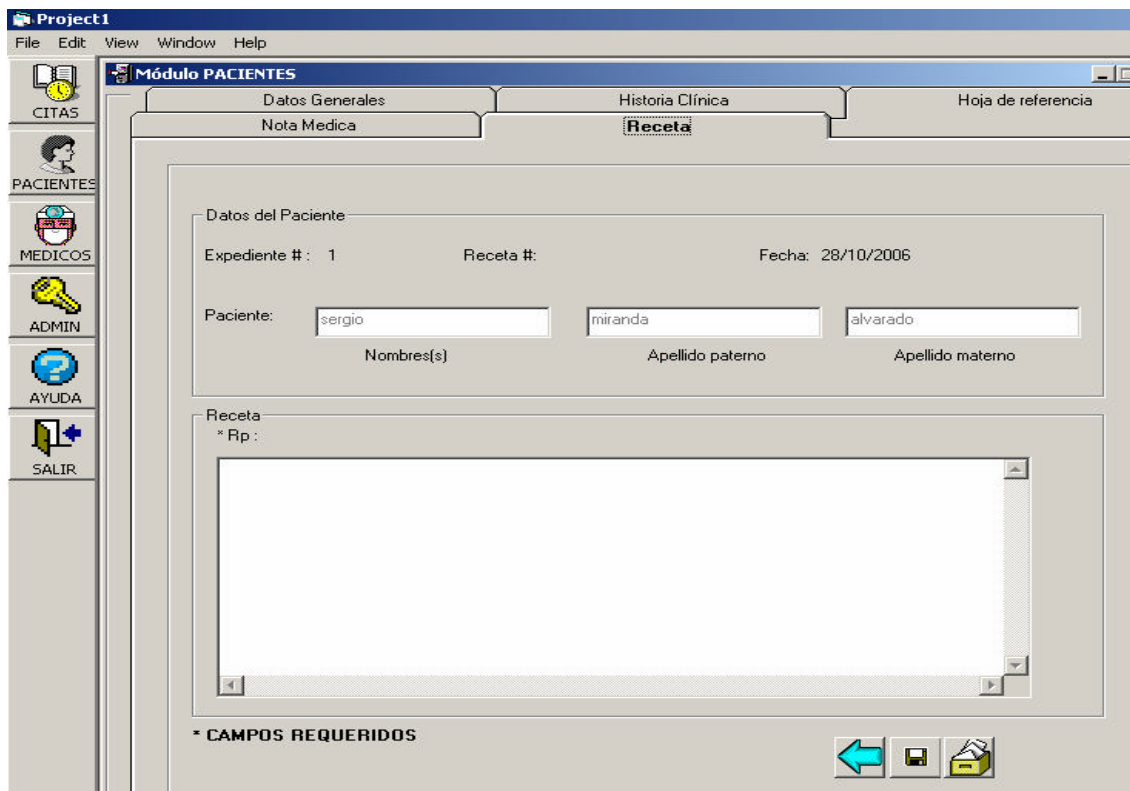


Si se desea terminar la consulta, se presionará el botón  ya que limpiará la información que se encuentra en los campos deshabilitados, permitiendo crear otra nota médica.


Receta Médica

Las recetas médicas que se les dan a los pacientes en cada consulta tienen como finalidad indicarle cuáles son los medicamentos que debe consumir para el tratamiento y las indicaciones necesarias en cuanto a dosis, horarios y duración del tratamiento.


Las recetas son personalizadas y tienen como encabezado el nombre del médico, especialidad, cédula profesional. La receta se imprime para proporcionársela al paciente. La pantalla comprende solo datos del paciente, el número de expediente, fecha y el campo para describir el tratamiento a seguir, dosis, medicamentos, etc.



Al igual que en los otros bloques es necesario capturar los campos obligatorios.

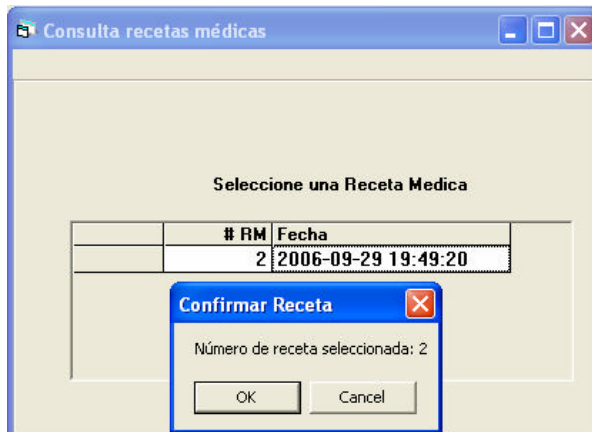
Dar clic en el botón .

Consulta de Recetas Médicas Anteriores

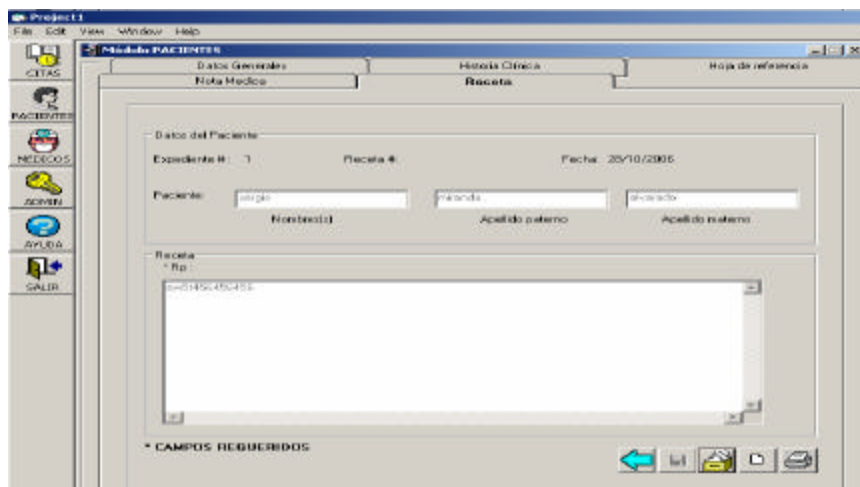
Para consultar una receta médica se debe presionar el botón , el cual ejecutará una ventana en la que muestra todas la recetas existentes por número de receta y fecha de creación.

Para consultar la receta medica, se da clic en la celda en donde esta la receta que se desea, a continuación se mostrará una ventana que indica el número de

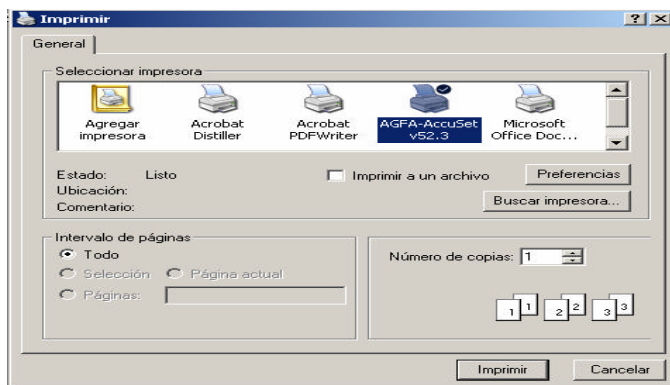
receta que elegimos y la aceptación, de lo contrario permite seleccionar otra receta.




Por lo que se desplegará en la pantalla de receta médica la información correspondiente a la seleccionada con la fecha y hora de creación los campos se encontraran deshabilitados para evitar la modificación de éstos. Se activarán los botones de terminar consulta, imprimir, consultar receta médica anterior.



Este apartado también nos permite obtener el reporte de la receta médica generada o de la que se está consultando en ese momento, mostrando la información de encabezado el nombre del médico, especialidad, cédula profesional, además de la información del paciente: nombre completo, fecha de creación de la receta y la descripción de su tratamiento.

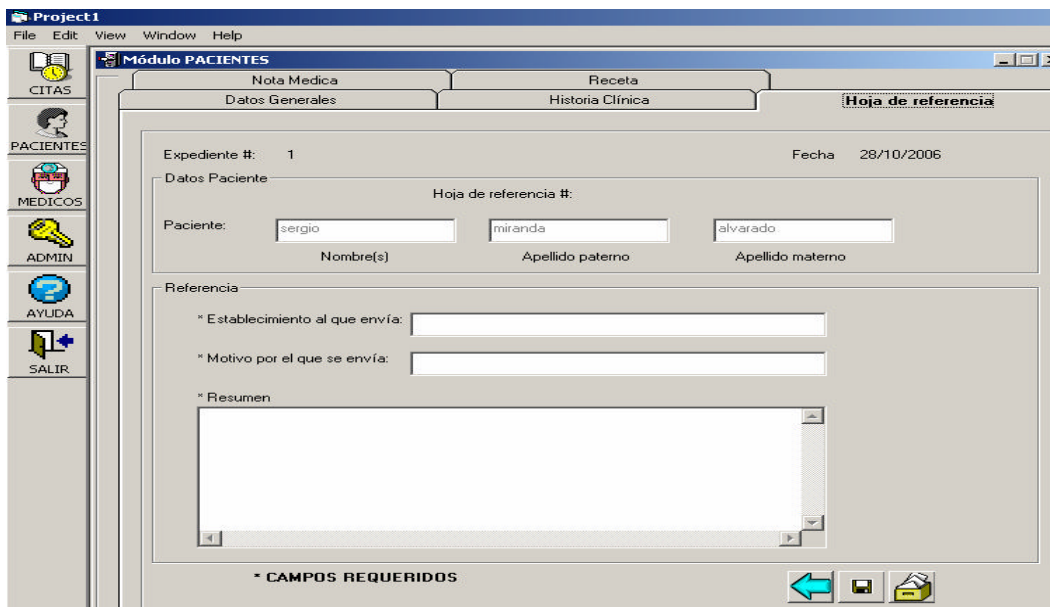


Si se desea terminar la consulta, se presionara el botón  ya que limpiará la información que se encuentra en los campos deshabilitados, permitiendo volver a escribir en el campo, para crear otra receta médica


Hoja de Referencia


La hoja de referencia, se utiliza cuando el paciente por alguna razón, ya sea que no se cuente con el equipo necesario o bien requiera atención hospitalaria, no pueda ser atendido en el consultorio se envíe a otra clínica.

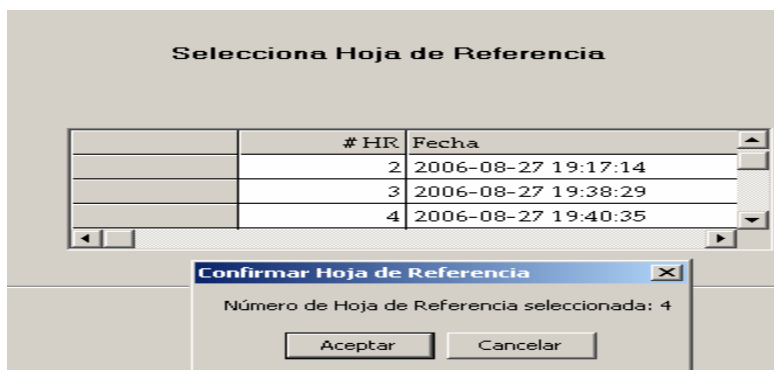
Esta pantalla esta dividida por datos del paciente y la propia hoja de referencia, la cual, indica el establecimiento al que se envía, el motivo, y la descripción de por que se esta enviando a otra clínica, también esta compuesta por el número de expediente y la fecha.



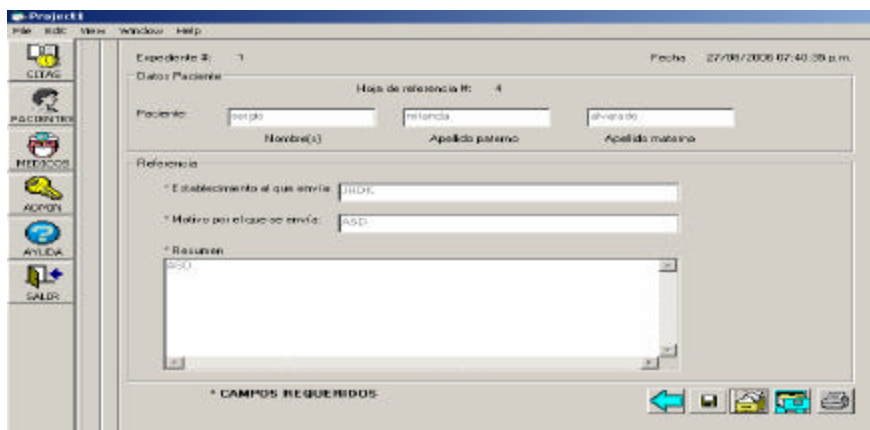
Al igual que en los otros bloques es necesario capturar los campos obligatorios.

Dar clic en el botón .

Para consultar una hoja de referencia se debe presionar el botón , el cual ejecutará una ventana en la que muestra todas las hojas de referencia existentes por número de hoja y fecha de creación. Para consultar la hoja de referencia dar clic en la hoja que se desea consultar, a continuación se mostrará una ventana que indica el número de hoja que elige y la aceptación, de lo contrario permite seleccionar otra.

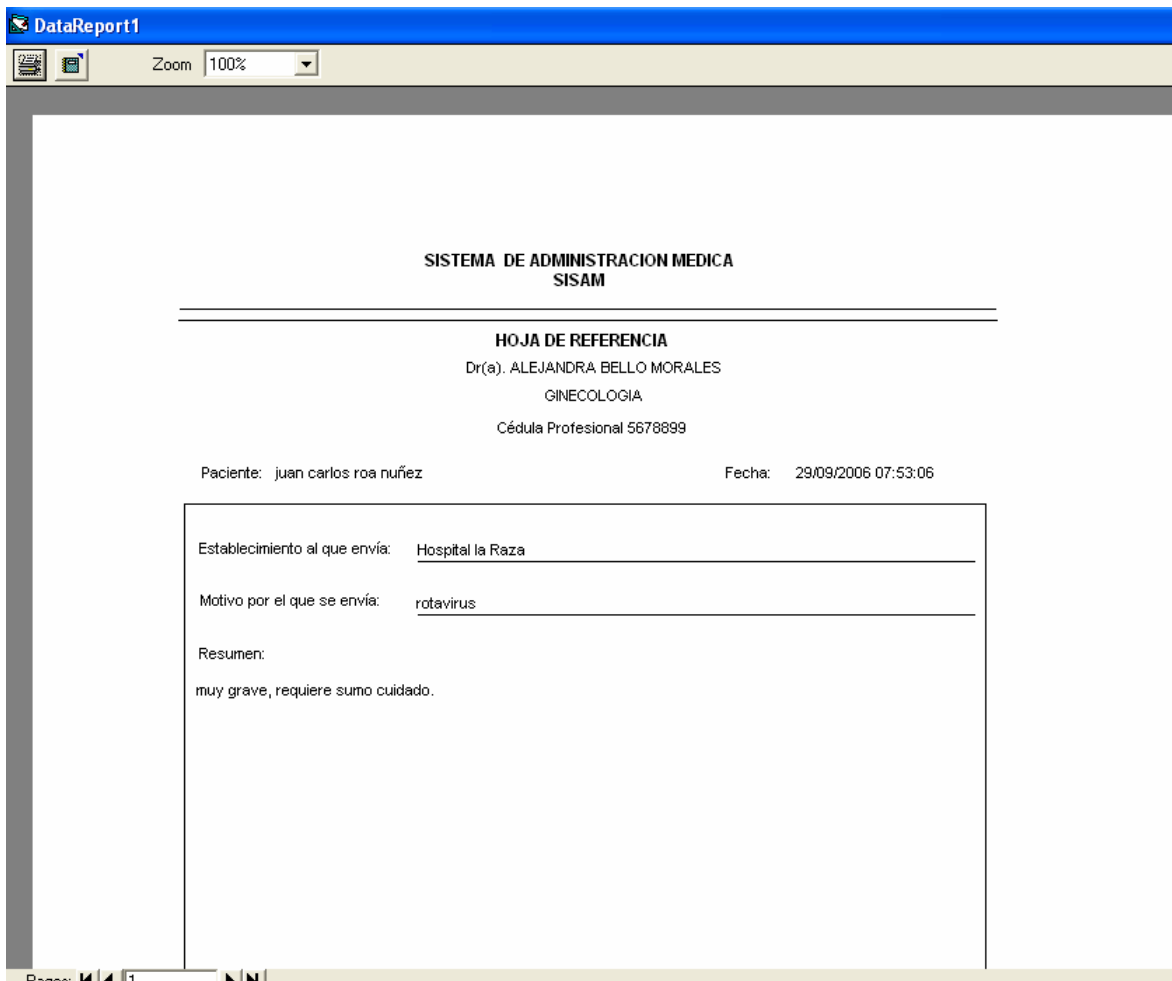



Una vez elegida la hoja de referencia a consultar se mostrará la información correspondiente a ésta en la pantalla de hoja de referencia médica con la fecha y hora de creación, los campos se encontrarán deshabilitados para evitar la modificación de estos. Se activarán los botones de terminar consulta, imprimir, consultar hoja de referencia anterior.

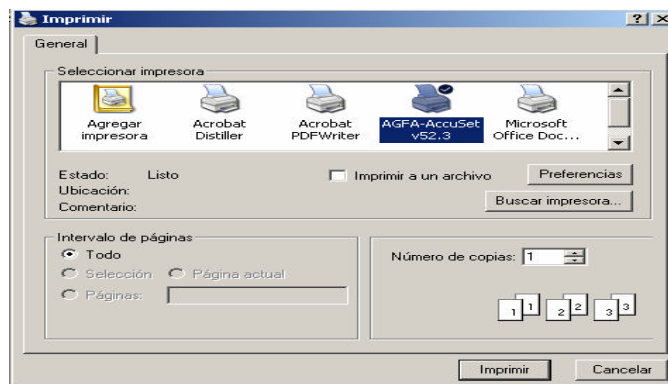



Este apartado también nos permite obtener el reporte de la hoja de referencia generada o de la que se está consultando en ese momento, mostrando la información de encabezado el nombre del médico, especialidad, cédula profesional, además de la información del paciente: nombre completo, fecha de creación de la hoja de referencia.

1. Dar clic en el botón ,
2. Se desplegará el reporte en vista previa.



3. Al presionar el botón  que se encuentra en la parte superior izquierda, desplegará una ventana en la cual seleccionamos la impresora en donde se desea imprimir.



Si se desea terminar la consulta, se presionará el botón  ya que limpiará la información que se encuentra en los campos deshabilitados, permitiendo volver a escribir en el campo, para crear otra hoja de referencia médica.

MÓDULO DE ADMINISTRACIÓN

SISAM está diseñado para que sus usuarios puedan acceder de la misma forma, independientemente del equipo desde donde se conecten.

El módulo de Administración permite crear las cuentas de los usuarios que tendrán acceso al sistema, así como la definición de las tareas que podrán o no realizar los usuarios.

La siguiente figura muestra la pantalla principal de administración del SISAM.



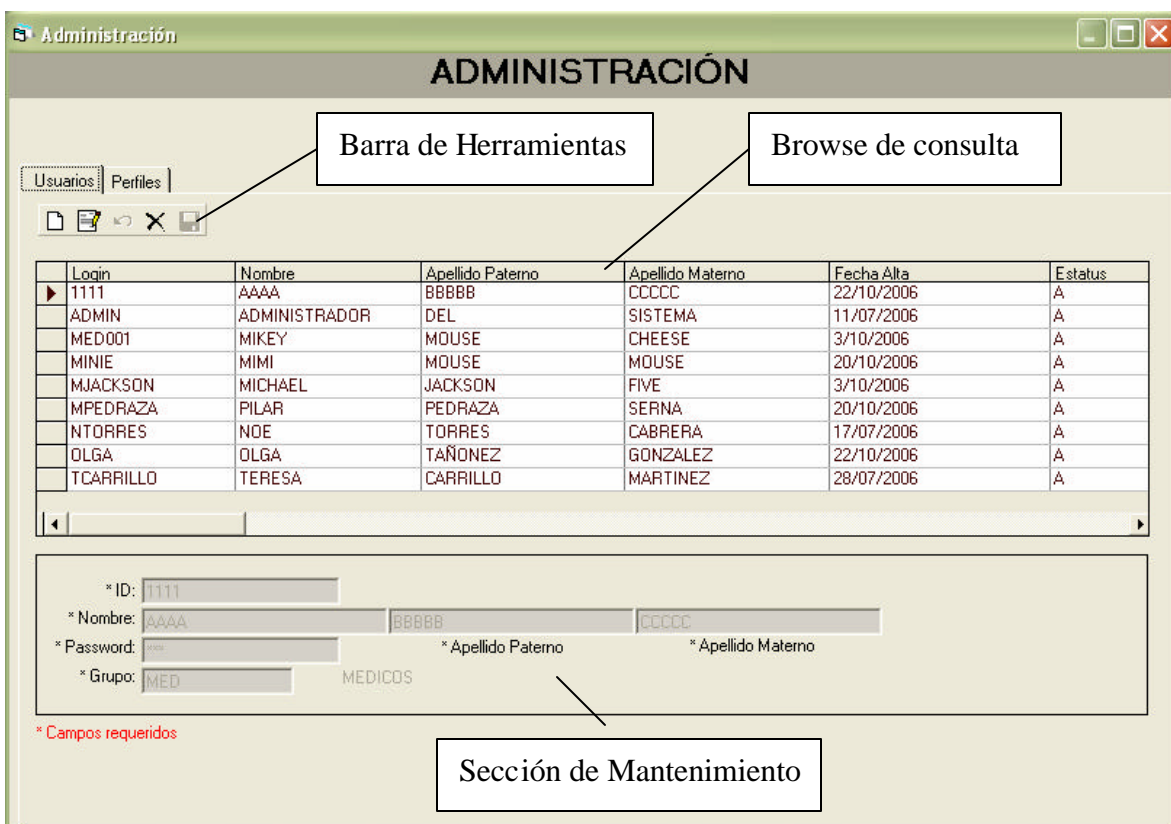
La ventana de administración está constituida por 2 pestañas, la primera denominada "Usuarios" y la segunda "Perfiles".

En la pestaña Usuarios se podrá definir cuales son los usuarios que podrán acceder al SISAM, mientras que en la pestaña Perfiles se determinarán las tareas a las que tendrán permiso.

Descripción de la Pestaña “Usuarios”


Esta pestaña está conformada por tres secciones principales:

- **Barra de Herramientas.** Permite seleccionar la operación deseada como Agregar, Actualizar, Cancelar operación, Borrar y Salvar.
- **Browse de Usuarios.** Visualiza el conjunto de usuarios del sistema.
- **Panel de Mantenimiento.** Permite capturar los datos e información correspondiente a los usuarios.




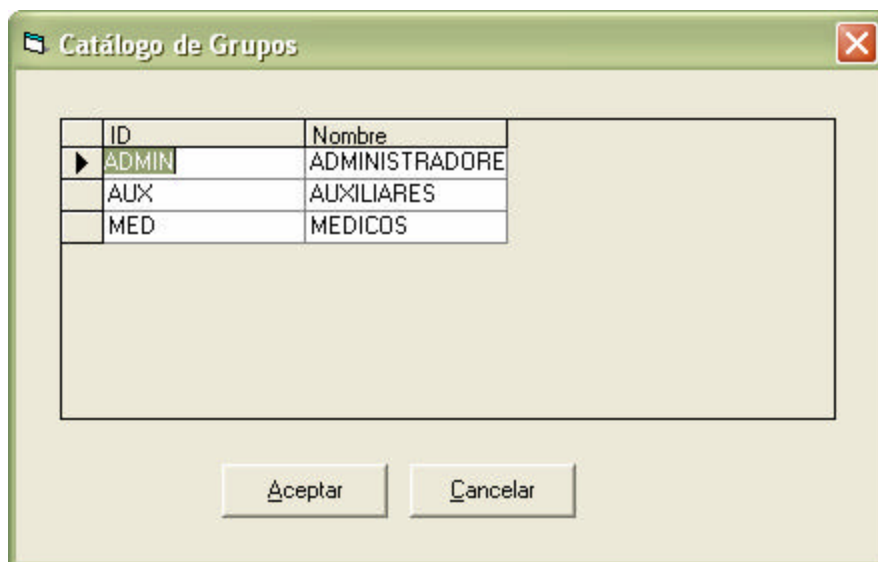
Alta de Usuario



A continuación se lista cada uno de los pasos a realizar para la creación de nuevos usuarios.

1. En la barra de herramientas de la pestaña “Usuarios”, pulse el botón nuevo  .
2. Introduzca el nombre, apellido paterno y materno del usuario.
3. Introduzca el ID y password con el que desea se valide el nuevo usuario al entrar al sistema.

Nota:

- El SISAM no es sensible a mayúsculas y minúsculas.
4. Seleccione el grupo al que pertenecerá el usuario. Para visualizar los grupos posibles, dar clic en el botón  . En la ventana del catálogo de grupos, dar clic sobre el registro elegido del browse de grupos y posteriormente dar clic en el botón Aceptar.




5. Dar clic en el botón salvar  para guardar los datos. Para cancelar la operación agregar, dar clic en el botón cancelar  de la barra de herramientas.



Nota:

- Los campos mínimos necesarios y obligatorios tienen antepuesto un asterisco (*).

Cambiar Datos de Usuario

1. Seleccionar la opción editar dando clic en el botón  de la barra de herramientas.
2. En el panel de mantenimiento de datos actualizar la información necesaria.


Nota:

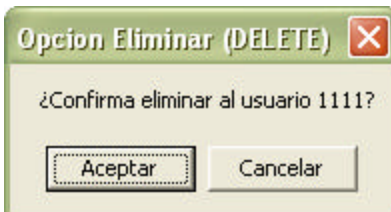
- Para el usuario **admin** solo será posible modificar el password.
3. Una vez que se hayan editados los datos del usuario, dar clic en el botón salvar  de la barra de herramientas. Para cancelar la operación editar, dar clic en el botón cancelar  de la barra de herramientas.

Eliminar un Usuario

1. Seleccione el usuario que desea eliminar dando clic sobre este en el browse de usuarios.

Login	Nombre	Apellido Paterno	Apellido Materno	Fecha Alta	Estatus
▶ ADMIN	ADMINISTRADOR	DEL	SISTEMA	11/07/2006	A
JSUAREZ	JORGE	SUAREZ	APM	5/10/2006	A
NTORRES	NOE	TORRES	CABRERA	17/07/2006	A
TCARRILLO	TERESA	CARRILLO	MARTINEZ	28/07/2006	A
XXXX	NOMBRE	APP	APM	2/10/2006	A

2. Seleccione la opción eliminar dando clic en el botón  de la barra de herramientas. El sistema solicitará que confirme la operación eliminar para el usuarios. Si deseamos confirmarla, daremos clic en el botón Aceptar, o bien, clic en el botón Cancelar para abortar la operación.

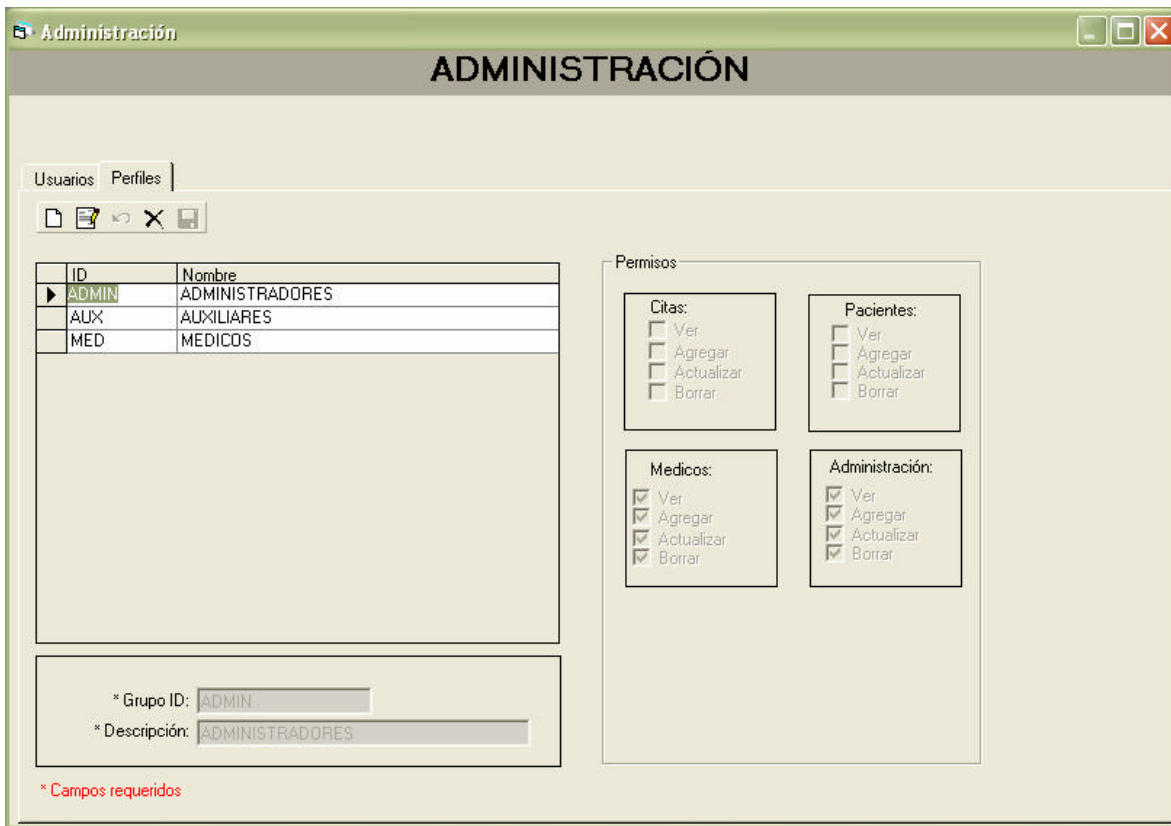
**Nota:**

- No es posible eliminar el usuario admin.

Descripción de la Pestaña Perfiles


Esta pestaña está conformada por tres secciones principales:

- **Barra de Herramientas.** Permite seleccionar la operación deseada como Agregar, Actualizar, Cancelar, Borrar y Salvar.
- **Browse de Grupos.** Visualiza los diferentes grupos de usuarios.
- **Panel de Mantenimiento y Permisos.** Permite capturar los datos e información correspondiente a los grupos, así como los permisos que tendrá el grupo con respecto a los módulos de citas, pacientes, médicos y administración.



Alta de Grupo

A continuación se lista cada uno de los pasos a realizar para la creación de nuevos grupos.

1. En la barra de herramientas de la pestaña "Perfiles", pulse el botón nuevo  .
2. Introduzca el ID y una descripción para el grupo.

Nota:

- El grupo ADMIN es el predefinido para los permisos de administración.
- No es posible crear otro grupo de administración.



3. En la sección Permisos, seleccionar solo aquellos que deseamos que posea el grupo en lo referente a los módulos de citas, pacientes, médicos y administración.

Permisos

Módulo	Ver	Agregar	Actualizar	Borrar
Citas:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Pacientes:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Medicos:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Administración:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Nota:


- Si no se especifica permiso alguno, el grupo no tendrá acceso a ningún módulo del sistema.

4. Dar clic en el botón salvar  para guardar los datos. Para cancelar la operación agregar, dar clic en el botón cancelar  de la barra de herramientas.



Nota:

- Los campos mínimos necesarios y obligatorios tienen antepuesto un asterisco (*).

Cambiar Datos de Grupos

1. Seleccionar la opción editar dando clic en el botón  de la barra de herramientas.
2. En el panel de mantenimiento de datos y perfiles actualizar la información necesaria.

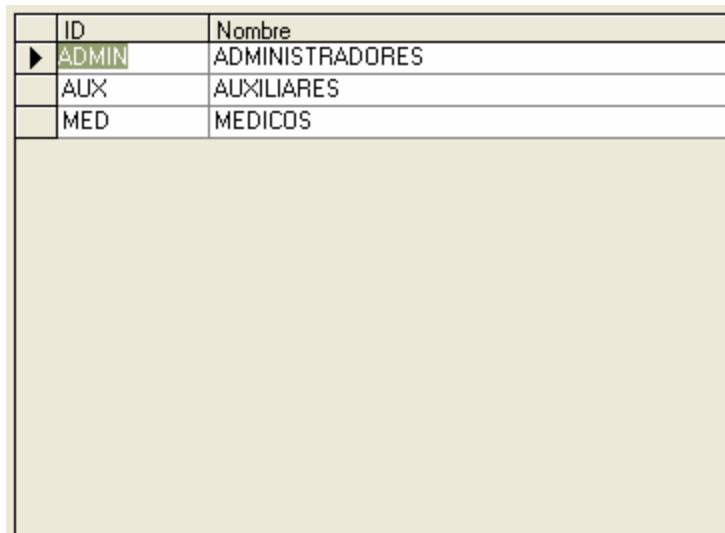
Nota:


- No es posible modificar los datos del grupo **ADMIN**.
3. Una vez que se hayan editados los datos del grupo, dar clic en el botón salvar  de la barra de herramientas. Para cancelar la operación editar, dar clic en el botón cancelar  de la barra de herramientas.

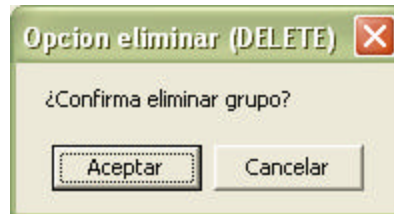
Eliminar un Grupo

1. Seleccione el usuario que desea eliminar dando clic sobre éste en el browse de grupos.

ID	Nombre
▶ ADMIN	ADMINISTRADORES
AUX	AUXILIARES
MED	MEDICOS



2. Seleccione la opción eliminar dando clic en el botón  de la barra de herramientas. El sistema solicitará que confirme la operación eliminar grupo. Si deseamos confirmarla, daremos clic en el botón Aceptar, o bien, clic en el botón Cancelar para abortar la operación.


**Nota:**

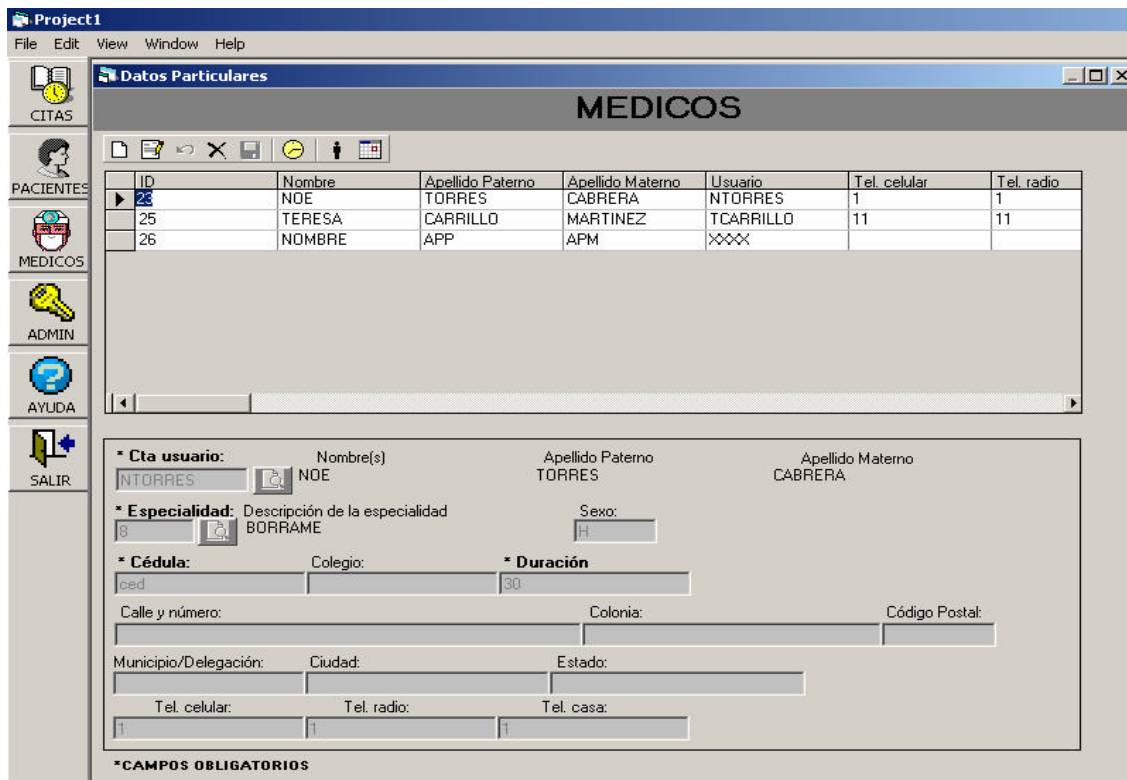
- No es posible eliminar un grupo para el cual existan usuarios asociados.

MÓDULO DE MÉDICOS

El módulo de Médicos permite almacenar los datos generales, turnos disponibles y especialidades con los que cuenta el médico, así mismo permite modificar y consultar la información capturada.




Al dar clic en el botón  de la pantalla general de SISAM, muestra una ventana donde permite capturar los datos generales del médico o ver los médicos existentes en el sistema. Esta misma permite acceder al catalogo de turnos y especialidades y dar mantenimiento a los médicos. La pantalla del módulo de médicos es la siguiente:

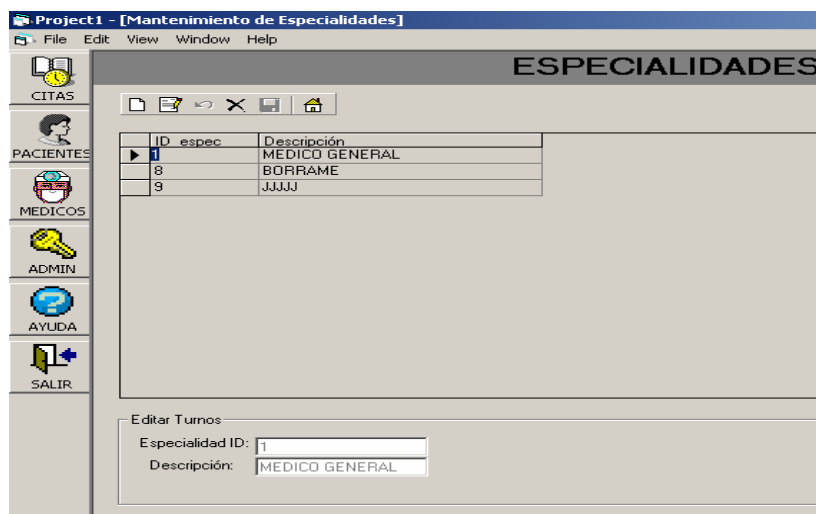


Esta pantalla se divide en dos bloques, el primero se muestran los médicos existentes en el sistema y mediante una barra de herramientas nos da las siguientes opciones: agregar nuevo médico, modificar, cancelar, guardar, asignar


horario, catalogo de turnos y de especialidades, el segundo bloque muestra la información general de los médicos donde los campos estarán deshabilitados para no permitir escribir sobre ellos.

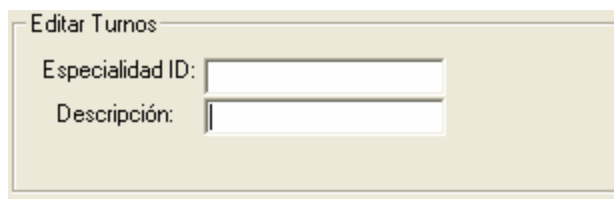
Para crear un médico se tiene que verificar que exista la especialidad y el o los turnos que el médico desea.

1. Dar clic en el botón  para que muestre la pantalla de mantenimiento de especialidades donde al igual que en la pantalla de médicos tenemos una barra de herramientas que permite agregar, modificar, deshacer, cancelar, guardar y regresar a la pantalla principal de médicos. La pantalla que se muestra es la siguiente:



Esta pantalla se divide en dos bloques el primero muestra las especialidades con las que cuenta el catalogo, el segundo bloque muestra el ID y de descripción de la especialidad donde los campos estarán deshabilitados para no permitir escribir sobre ellos.


2. Si la especialidad no existe dar clic en el botón  para crear una nueva especialidad. La pantalla donde se capturan los datos nuevos es la siguiente:




Editar Turnos

Especialidad ID:


Descripción:

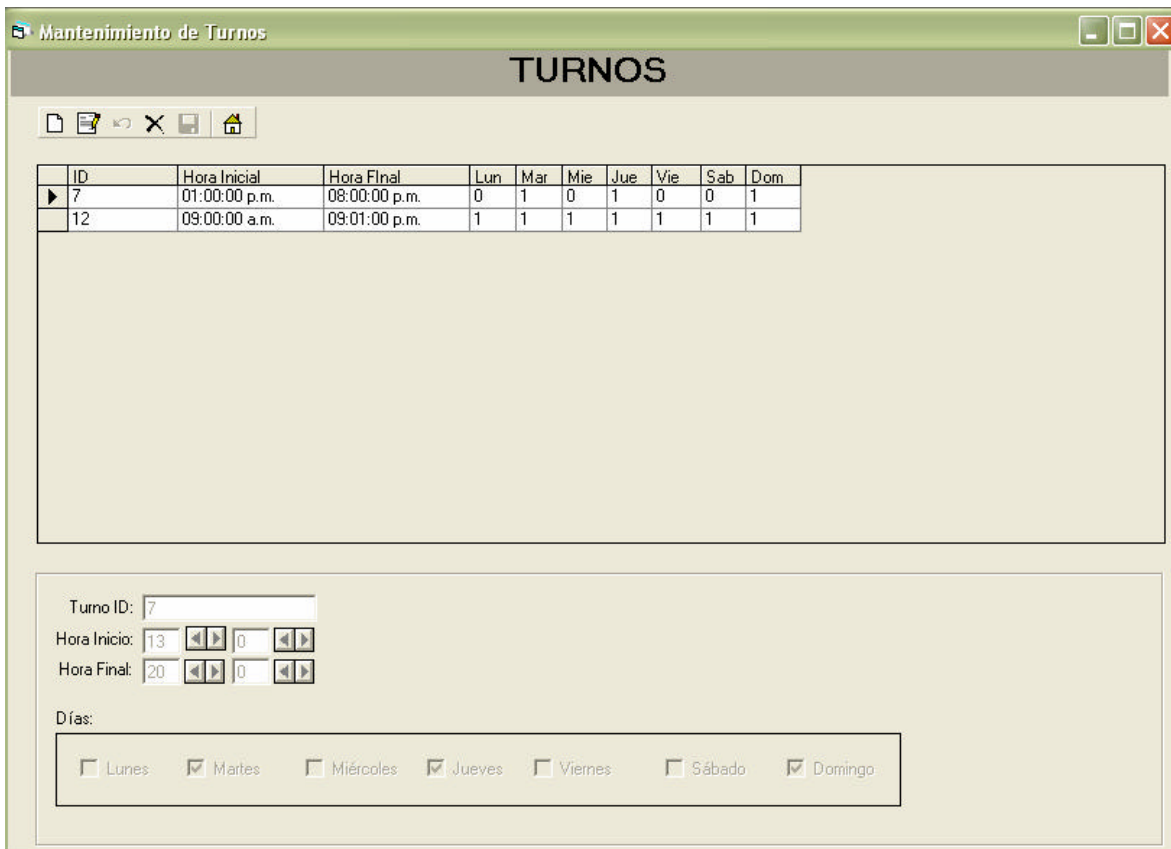
3. Dar clic en el botón .


4. Dar clic en el botón  para regresar a la pantalla principal de médicos.

5. Estando en esta pantalla dar clic en el botón  para acceder a mantenimiento de turnos. En esta se podrá verificar si existe el turno requerido por el médico o si se tiene que agregar.


6. Si el turno existe dar clic en el botón  para regresar a la pantalla principal de médicos.


7. Si no existe el turno dar clic en el botón  para ingresar el nuevo turno, en esta pantalla también se cuenta con una barra de herramientas que permite agregar, actualizar, deshacer, cancelar, guardar y volver la pantalla principal de médicos. La pantalla que se muestra es la siguiente:

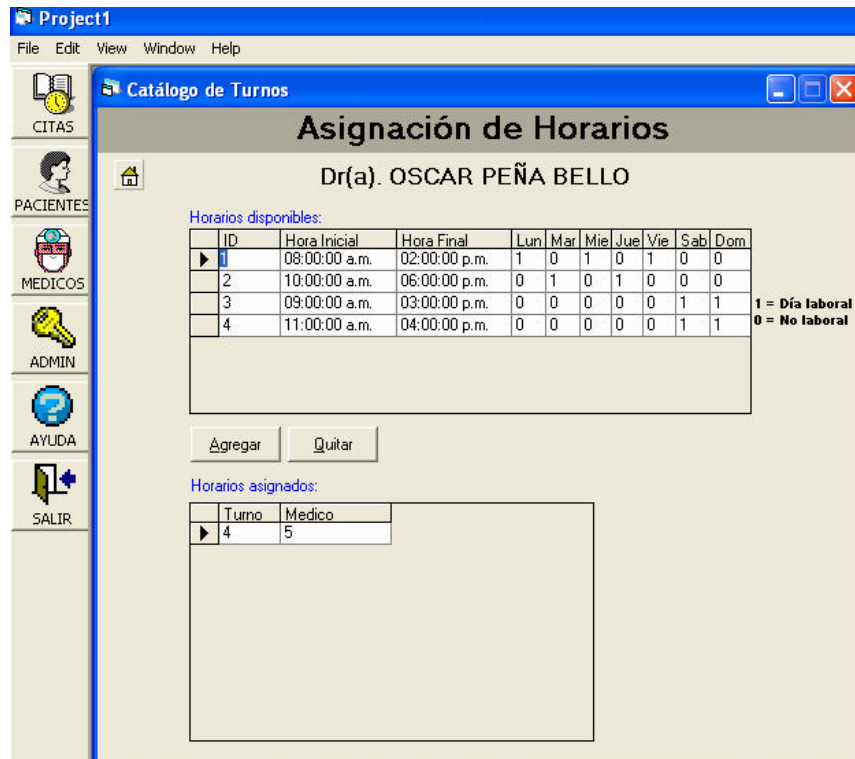


8. Dar clic en el botón .

9. Dar clic en el botón  para regresar a la pantalla principal de médicos.

10. Al regresar a la pantalla de médicos dar clic en el botón  para crear al médico, en esta pantalla se activan los campos donde se capturan los datos generales del mismo y se selecciona la especialidad. Al igual que en los otros bloques es necesario capturar los campos obligatorios.

11. Se asignan el o los turnos que va a tener el médico dando clic en el botón  en donde se muestra la siguiente pantalla:



12. Dar clic en el botón  para regresar a la pantalla principal de médicos. |

De esta manera se podrá agregar y modificar la información de los médicos al SISAM.

CONCLUSIONES

El Programa de Apoyo a la Titulación tiene un esquema de trabajo en el cual es posible que las personas que trabajamos podamos concluir nuestra carrera de una manera sencilla y eficaz, contando con el apoyo del asesor para dirigir este trabajo de tesis.

El desarrollo del tema tesis resulta una actividad interesante ya que representa la oportunidad de aplicar los conocimientos adquiridos durante la etapa escolar, la cual nos ha proporcionado las herramientas necesarias para ser profesionistas de calidad.

El llevar a cabo un proyecto de tesis, implica sin lugar a duda el trabajo en equipo, en el cual, los integrantes del mismo trabajamos de manera coordinada en la ejecución de esta tarea, tomando en cuenta que cada miembro contribuyó en el enriquecimiento de este trabajo con la aportación de ideas y conocimientos para su realización, completando así mismo la investigación y desarrollo del tema, teniendo una comunicación abierta que permitiera la unificación de criterios y finalmente mostrando un compromiso que lograra que el proyecto de tesis saliera lo mejor posible.

Hoy en día, es de suma importancia que las empresas cuenten con un sistema que les permita estar a la vanguardia tecnológica y así poder evitar quedar rezagadas y ser más competitivas. Así mismo, los sistemas son necesarios para el desarrollo y la organización de procesos y funciones empresariales.

El Sistema de Administración Médica cumple con el objetivo planteado, que es solucionar los problemas a los que se enfrenta una clínica en su administración derivado de tener la información dispersa; logramos implantar un sistema de información (una base de datos) el cual concentra la información con seguridad y confiabilidad en los datos.

Con SISAM se puede tener un mejor manejo y control de la información que se tiene en consultorios y pequeñas clínicas evitando malos manejos o pérdida de información.

El sistema es amigable, lo que facilita su uso, además de contar con el manual de usuario integrado en el mismo, lo que sirve de ayuda en caso de necesitarla.

Otra ventaja del sistema es que puede seguir creciendo, en cuanto cambien los requerimientos de los usuarios, se pueden realizar los ajustes que sean convenientes.

El sistema que hemos desarrollado ha dejado una enseñanza importante en el aspecto profesional, ya que como ingenieros en computación nos enfrentamos a problemas a los cuales se debe dar la solución óptima, la cual es posible hallar con un buen análisis, metodología y diseño de sistemas.

SISAM (Sistema de Administración Médica) ofrece una solución práctica a un problema real dentro de un consultorio médico, cumpliendo con los requerimientos solicitados por el cliente, y siguiendo una lógica de desarrollo que nos permitió ofrecer un sistema práctico y funcional.

BIBLIOGRAFÍA

BIBLIOGRAFÍA

- **Análisis y Diseño de Sistemas**
Kenneth E. Kendall y Julie E. Kendall
Ed. Prentice Hall
México, 1991

- **Aprendiendo Microsoft Windows XP en 21 lecciones avanzadas**
John Paul Mueller
Ed. Person Educación
México, 2002

- **Ingeniería del software. Un enfoque práctico**
Roger S. Presuman
Ed. McGraw-Hill
México, 2000
4ª. Edición

- **Microsoft Fundamentos de Redes Plus**
Curso Oficial de Certificación MCSE
Ed. Mcgraw Hill
España, 2000

- **Oracle 10g Administración y Análisis de Bases de Datos**
César Pérez
Ed. Alfaomega
México, 2005
1a. Edición

- <http://www.microsoft.com/mexico>
- <http://web.mit.edu/rhel-doc/3/rhel-rg-es-3/ch-kerberos.html>
- http://www.aepro.com/congresos/2004_1/pdf/ana.nieto@upct.es_512af97b56ecf7690b97e81646cdeb12.pdf
- <http://alarcos.infr.uclm.es/doc/ISOFTWAREI/Tema04.pdf#search=%22SSADM%22>
- <http://www.mysql.com/>
- <http://dev.mysql.com/doc/refman/5.0/es/features.html>
- http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec_14.htm
- http://www.sigonline.com/CID/detalle_curso.asp?Id=4810
- <http://www.geocities.com/agendamedicapro/>
- <http://es.wikipedia.org/wiki/Switch>
- <http://ditec.um.es/laso/docs/tut-tcpip/3376c21.html#figintlay>

MANUAL TÉCNICO

MANUAL TÉCNICO.

Como Respaldar una Base de Datos

Existen dos formas comunes de poder realizar un respaldo de una base de datos. La primera forma, consiste en hacer uso del cliente `mysqldump` o `mysqlhotcopy` a través de línea de comando del símbolo del sistema. La segunda forma, consiste en hacer uso de la utilería de MySQL Administrator, la cual cuenta con una interfaz gráfica amigable para el usuario, lo que la convierte en la opción más utilizada en el ámbito de respaldos de bases de datos.

A continuación, se describen brevemente cada una de las opciones de respaldo de bases de datos.

1. Uso de Clientes `mysqldump` y `mysqlhotcopy`

`mysqldump`

Este comando es utilizado por los administradores de bases de datos para realizar un **full backup** (Respaldo Total). Esta forma de respaldo, como su nombre lo indica, consiste en respaldar de manera total una base de datos.

El cliente `mysqldump` puede ser utilizado para descargar una base de datos o una colección de bases de datos para respaldos o para transferir los datos a otro Servidor del SQL (no necesariamente a un Servidor de MySQL). La descarga contiene instrucciones SQL para crear la tabla y/o para poblar la tabla.

Si todas las tablas que conforman la base de datos son del tipo MyISAM, entonces se debería considerar hacer uso del cliente `mysqlhotcopy` para que el respaldo sea más rápido.

Existen tres formas generales de invocar al cliente mysqldump desde la línea de comando del sistema operativo windows:

```
c:\> mysqldump [opciones] nombre_base [tablas]
c:\> mysqldump [opciones] --databases DB1 [DB2 DB3 ... ]
c:\> mysqldump [opciones] --all -databases
```

Si no se especifica tabla alguna o se usa la opción --databases o --all-databases, la base de datos entera se respaldará.

Para obtener la lista de opciones que mysqldump soporta, ejecutar el comando mysqldump --help.

A continuación se listan las opciones más comunes que se utilizan en el cliente mysqldump:

➤ --add-drop-database

Agrega una instrucción DROP DATABASE antes de cada instrucción CREATE DATABASE.

➤ --add-drop-table

Agrega una instrucción DROP TABLE antes de cada instrucción CREATE TABLE.

➤ --creates-options

Incluye todas las opciones específicas de tabla en las instrucciones CREATE TABLE.

➤ --databases, -B

Respalda varias bases de datos. Normalmente, mysqldump trata el primer nombre de argumento en la línea de comando como un nombre de base de datos y lo que sigue como nombre de las tablas. Con esta opción, se tratará todos los nombres de los argumentos como nombres de bases de datos. Las instrucciones CREATE DATABASE IF NOT EXISTS db_name and USE db_name son incluidas en el archive de salida antes de cada nueva base.

➤ --no-create-db, -n

Esta opción suprime la instrucción CREATE DATABASE /*!32312 IF NOT EXISTS*/ db_name que de lo contrario es incluida en el archivo de salida si la opción --databases o --all-databases es indicada.

➤ --opt

Esta opción es lo mismo que si se hubiera especificado --add-drop-table --add-locks --create-options --disable-keys --extended-insert --lock-tables --quick --set-charset. Esta opción hará una operación de respaldo rápida y produce un archivo de salida que puede ser recargado rápidamente en el Servidor MySQL. Esta opción se establece por defecto, pero puede ser deshabilitada con la opción --skip-opt. Para deshabilitar solo ciertas opciones habilitadas por --opt, solo se debe incluir --skip en la opción deseada; por ejemplo, --skip-add-drop-table o --skip-quick.

2. MySQL Administrator

Para poder crear un respaldo con MySQL Administrator, hacer clic en el acceso directo de éste, mismo que se encuentra como opción dentro del grupo de programas asociado a MySQL (ver figura 1.A).



Figura 1-A Ejecución de MySQL Administrator.

Una vez que se ejecuto el programa, MySQL Administrator solicitará el usuario y la contraseña para acceder al Servidor de Base de Datos (ver figura 2-A).

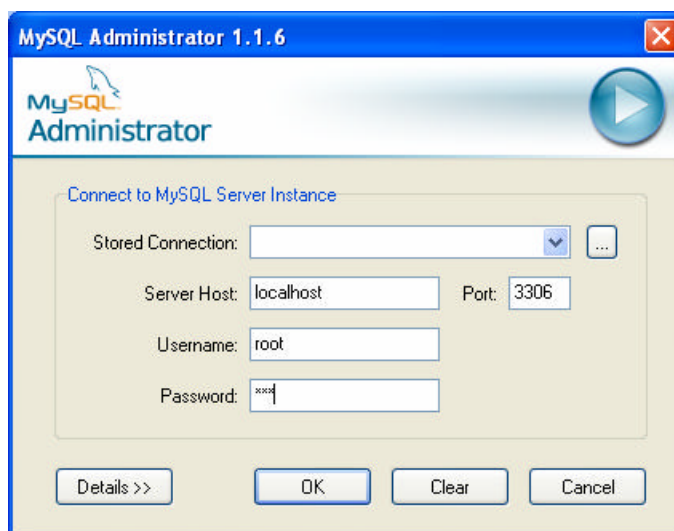


Figura 2-A Pantalla de Conexión al Servidor de Base de Datos.

Ya introducidos los valores correspondientes al usuario y la contraseña, dar clic en el botón OK para establecer la conexión. El usuario es identificado y autenticado en el Servidor de Base de Datos, es aquí, en donde el panel de MySQL Administrator nos aparece (ver figura 3-A).

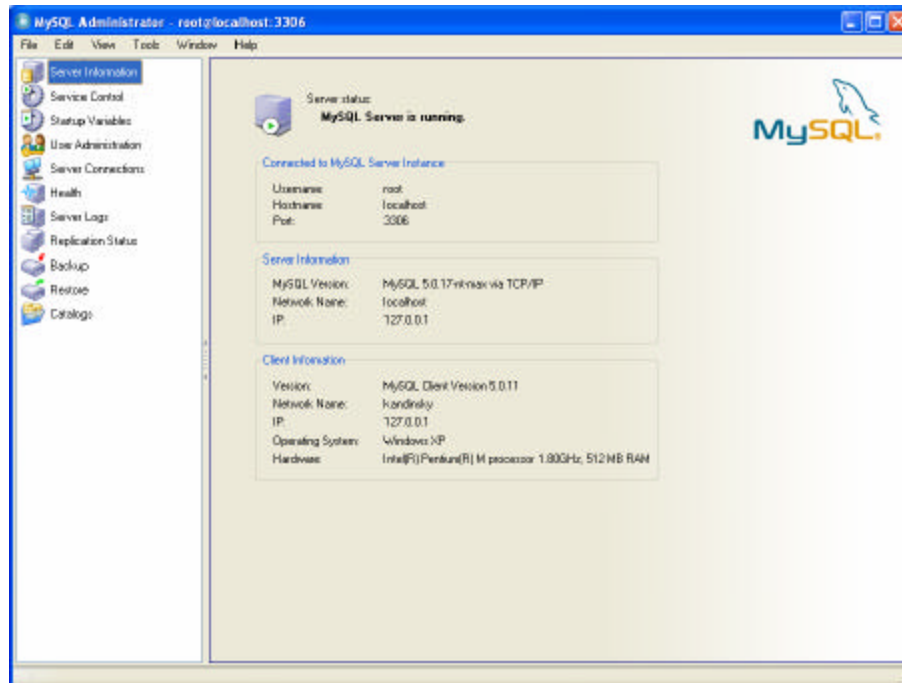


Figura 3-A Pantalla principal de MySQL Administrator.

Dentro del panel izquierdo de la pantalla principal de MySQL Administrator, existen varias opciones que son seleccionables. Una de ellas es la que se etiqueta como Backup. Al hacer clic en esta opción, dentro del panel derecho, aparecerá lo correspondiente al módulo de respaldos (ver figuras 4-A, 5-A y 6-A).

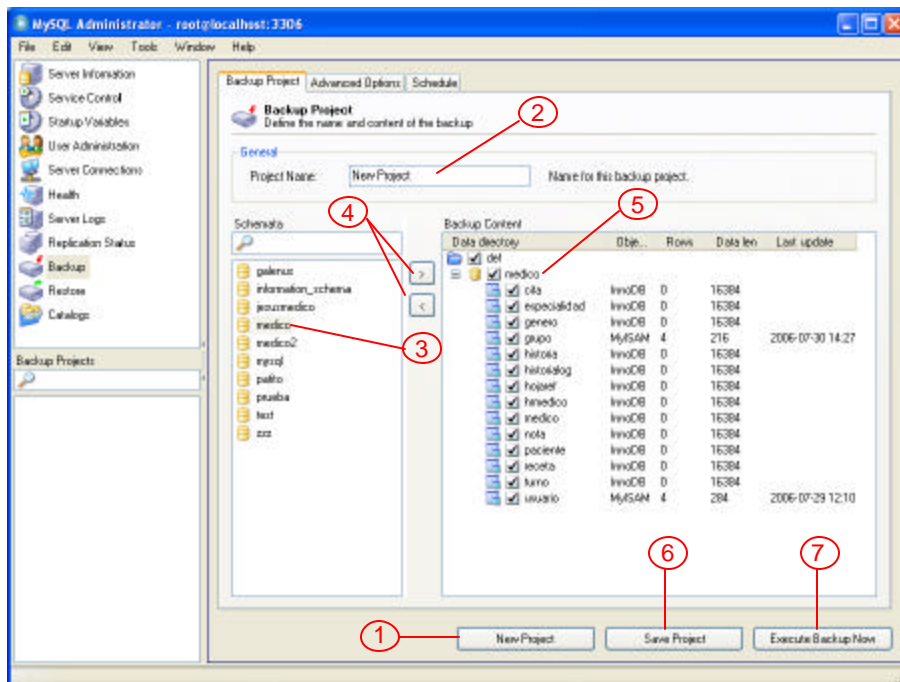


Figura 4-A Pantalla Principal de Respaldos.

A continuación se describe cada paso que se indica en la figura 4-A para poder realizar el respaldo de una base de datos.

- Paso 1. Dar clic en el botón New Project para crear un nuevo respaldo.
- Paso 2. Establecer el nombre del respaldo.
- Paso 3. En el panel que muestra la lista de bases de datos que se encuentran en nuestro Servidor de Base de Datos, seleccionar aquella que deseamos respaldar.
- Paso 4. Dar clic en el botón agregar ">" para incluir la base en el respaldo o bien clic en el botón quitar "<" para deshacer la inclusión de una base de datos en el proyecto del respaldo.
- Paso 5. Dentro del panel derecho que muestra la lista de tablas que conforma nuestra base de datos, seleccionar todas o solo aquellas que deseamos respaldar.

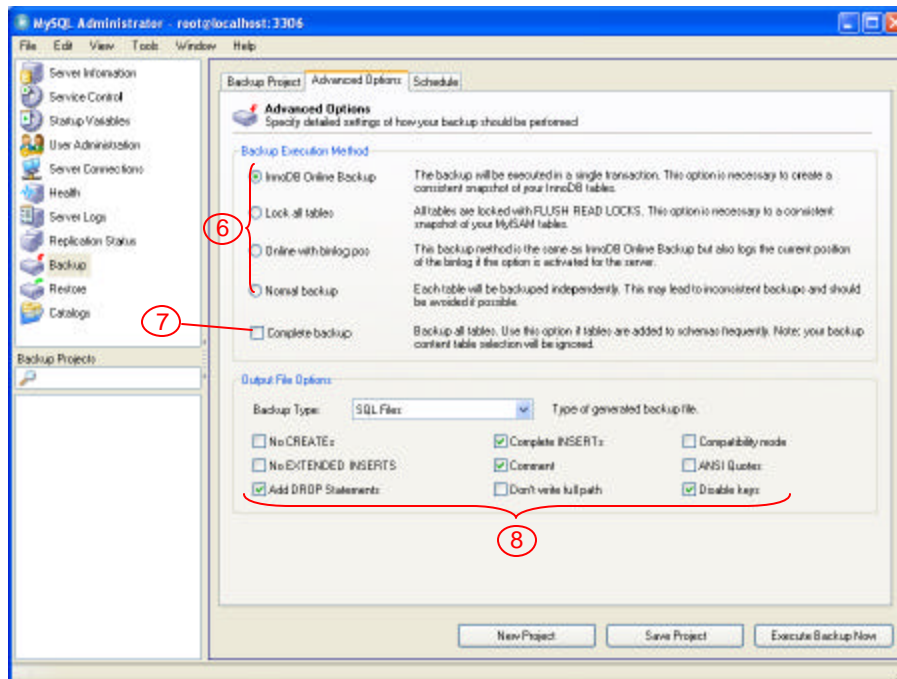


Figura 5-A Opciones de la pestaña Advanced Options de respaldos.

A continuación se describe cada paso que se indica en la figura 5-A para poder realizar el respaldo de una base de datos.

- Paso 6. En esta sección se indica el método de respaldo que se debe realizar con las tablas tomando como base su tipo.
- Paso 7. Marcar la casilla para indicar que el respaldo debe de hacerse de manera total (full backup).
- Paso 8. En la sección de las opciones que corresponden al archivo de salida, se puede indicar si se incluyen instrucciones para crear o no tablas, instrucciones para agregar datos o solo la estructura, agregar comentarios de las tablas, etc.

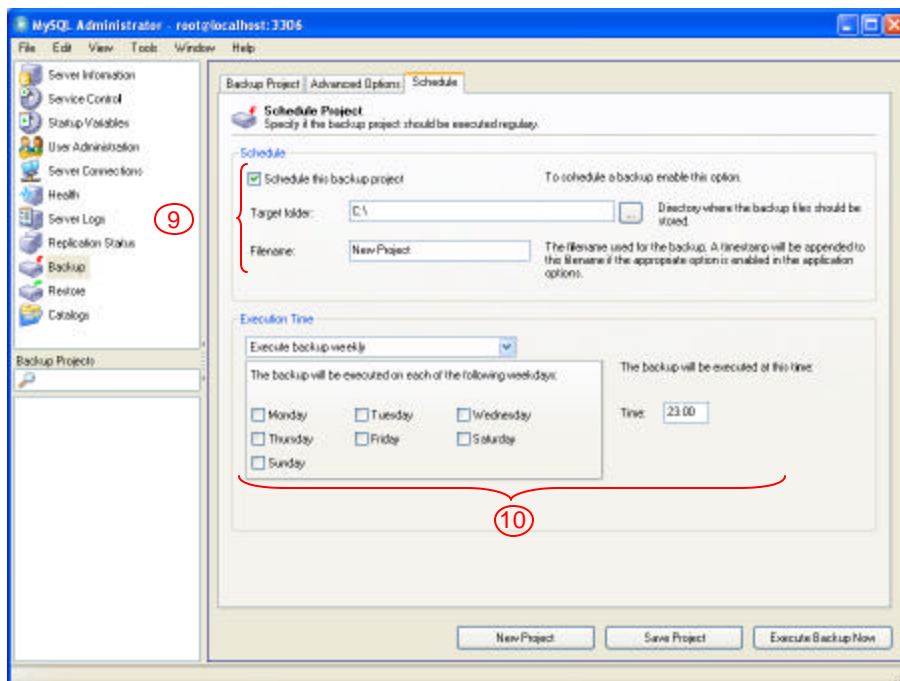


Figura 6-A Opciones de la pestaña Schedule de respaldos.

A continuación se describe cada paso que se indica en la figura 6-A para poder realizar el respaldo de una base de datos.

- Paso 9. La sección **Schedule** (Programación) permite definir la ruta y el nombre del archivo de salida que contendrá las instrucciones SQL para crear la base de datos.
- Paso 10. La sección **Execution Time** (Tiempo de Ejecución) permite definir el periodo y la hora en que deseamos que se ejecute la tarea de respaldo de la base de datos.

Como Restaurar una Base de Datos

1. Restaurar desde la Línea de Comando del Sistema Operativo

Estando en una sesión de Línea de Comando del sistema operativo, ejecutar la siguiente instrucción para restaurar una base de datos que fue previamente respaldada en un archivo **script** de SQL (Archivo de Instrucciones SQL).

```
C:\> mysql < backup_sunday_1_PM.sql
```

2. Restaurar desde MySQL Administrator

Para restaurar una base de datos a través de MySQL Administrator, es necesario dar clic en el botón **Restore** (Restaurar) que se encuentra en el panel izquierdo, e inmediatamente el panel derecho mostrará las pestañas correspondientes al proceso de restauración de una base de datos (ver figura 7-A).

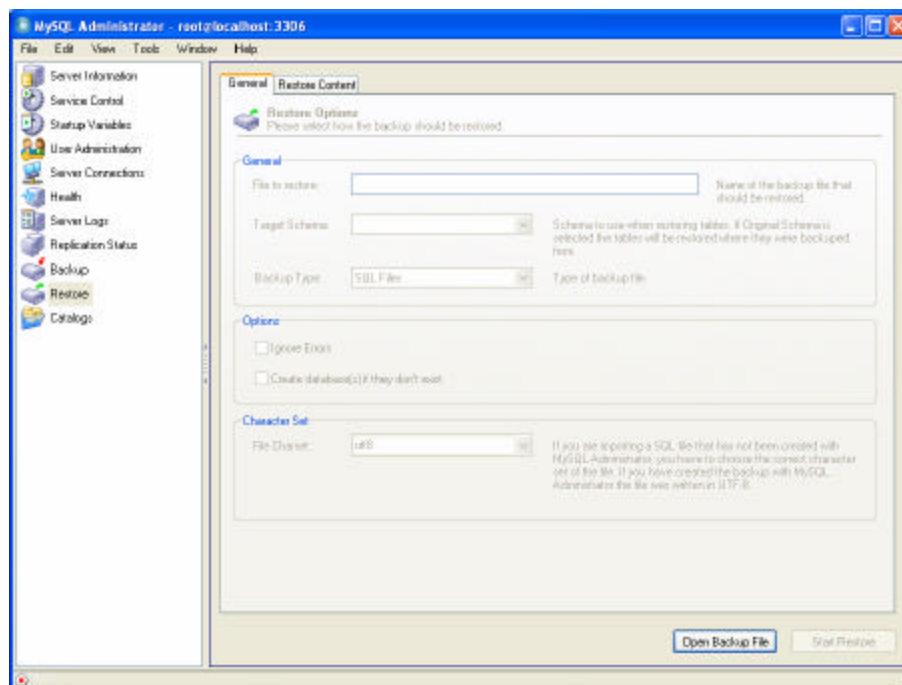


Figura 7-A Opción Restaurar Base de Datos.

Para configurar el proceso de restauración, se tiene que dar clic en el botón **Open Backup File** (Abrir Archivo de Respaldo) para indicar la ruta y seleccionar el nombre del archivo que contiene las instrucciones de SQL que permiten restaurar la base de datos.

Una vez seleccionado el archivo de restauración, es necesario indicar el Esquema Destino en donde restaurará la base de datos y el tipo de archivo de restauración. También es posible indicarle al programa de restauración que ignore los errores que se generen durante el proceso, así como la posibilidad de crear la base de datos si es que todavía no se crea. Es muy importante indicar también el juego de caracteres que almacenará nuestra base de datos (Ver figura 8-A).

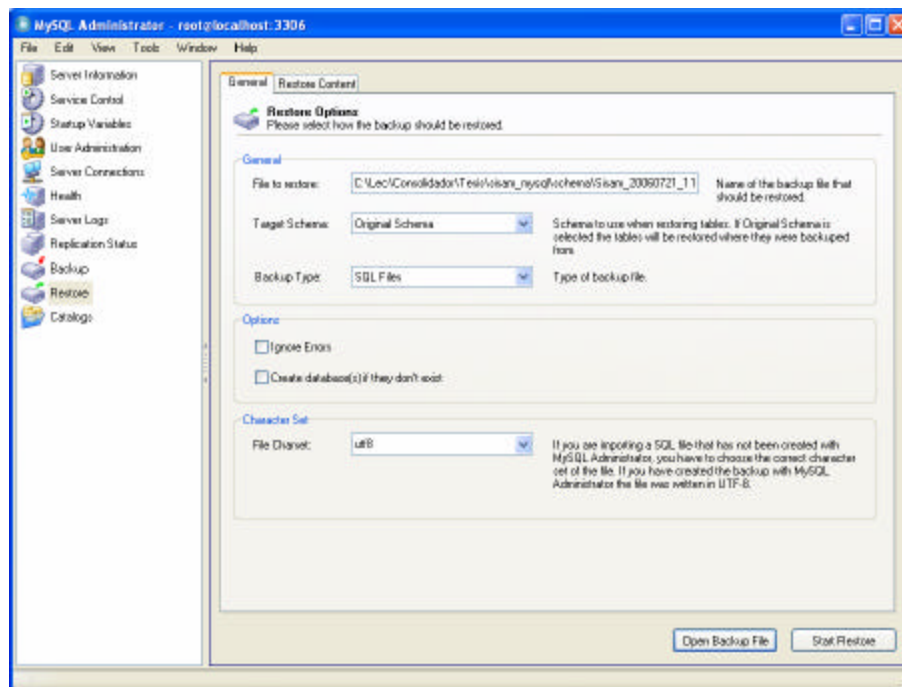


Figura 8-A Parámetros para restaurar una base de datos.

Finalmente, es necesario hacer clic en el botón **Start Restore** (Iniciar Restauración) para indicarle a MySQL Administrator que realice el proceso de restauración de la base de datos.

Como Levantar y Detener el Servidor de Base de Datos

Para poder detener el servicio del Servidor de Base de Datos existen dos formas de hacerlo.

- La primer forma consiste en utilizar la instrucción `mysqladmin` desde la línea de comando del sistema operativo.

```
c:\> bin/mysqladmin -u root shutdown
```

- La segunda opción es a través de MySQL Administrator. Estando dentro de esta utilería, dar clic en la opción **Service Control** (Control de Servicio) que se encuentra en el panel izquierdo de la pantalla principal de MySQL Administrator. Después de esto, aparecerá en el panel derecho la opción **Stop Service** (Detener Servicio), misma que se deberá elegir y que unos instantes después nos indicará que el Servidor de Base de Datos está detenido (ver figura 9-A). Para Levantar el servicio solo es necesario dar clic en el botón Start Service.

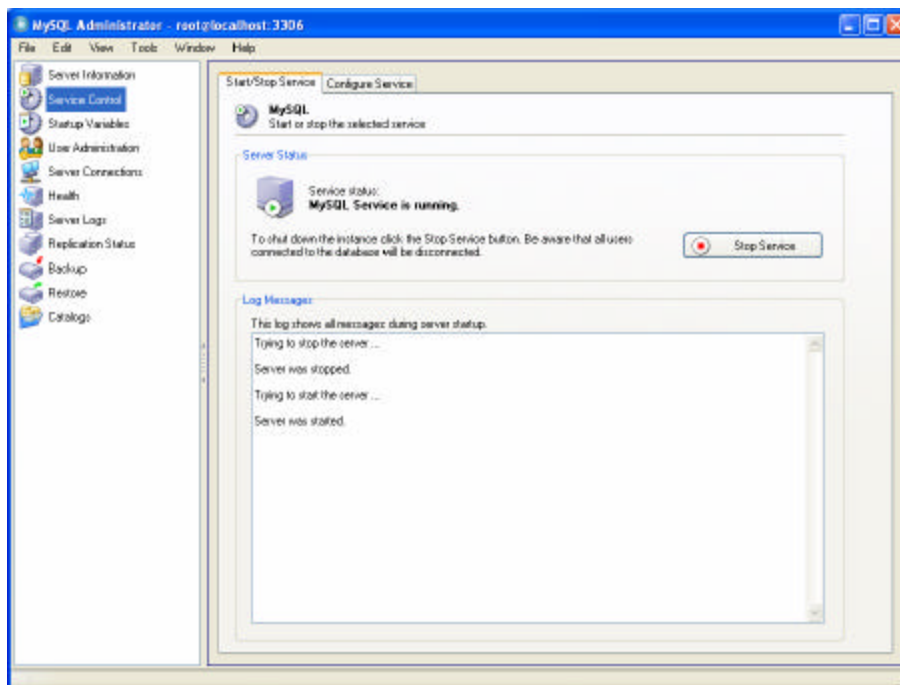


Figura 9-A Opción para Levantar y Detener el Servidor de Base de Datos.