



**UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO**

---

---

**FACULTAD DE ESTUDIOS SUPERIORES  
ARAGÓN**

ADMINISTRACIÓN DEL INVENTARIO DEL  
HOSPITAL DE ESPECIALIDADES DEL CENTRO  
MÉDICO NACIONAL SIGLO XXI  
BASADO EN EL DIPLOMADO  
“ADMINISTRACIÓN DE BASES DE DATOS”

**TRABAJO ESCRITO**

EN LA MODALIDAD DE  
SEMINARIOS Y CURSOS DE ACTUALIZACIÓN  
Y CAPACITACIÓN PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

P R E S E N T A:

**HUGO ALBERTO CEDILLO QUIROZ**

ASESOR:  
MTRA. SILVIA VEGA MUYTOY



FES Aragón

MÉXICO 2012



Universidad Nacional  
Autónoma de México



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

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

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

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



ADMINISTRACIÓN DEL INVENTARIO DEL  
HOSPITAL DE ESPECIALIDADES DEL CENTRO MÉDICO NACIONAL SIGLO XXI  
BASADO EN EL DIPLOMADO "ADMINISTRACIÓN DE BASES DE DATOS"

---

<b>INTRODUCCIÓN .....</b>	<b>6</b>
<b>CAPÍTULO 1   INFORME GLOBAL DEL DIPLOMADO EN ADMINISTRACIÓN DE BASES DE DATOS.</b>	
1.1 Sistemas de información y el modelo de datos relacional ...	8
1.2 Sistemas manejadores de bases de datos relacionales (RDBMS) .....	12
1.3 SQL (Standar Query Language) .....	20
1.4 Acceso a datos a través de la programación de clientes ....	25
1.5 Fundamentos de sistemas operativos .....	27
1.6 Administración de bases de datos .....	28
1.7 Buenas prácticas en la función de la administración .....	31
1.7.1 SGSI .....	31
1.7.2 ITIL .....	32
1.7.3 ISO 20 000 .....	33
1.7.4 Plan de continuidad del negocio .....	34
1.7.5 COBIT .....	35
1.7.6 Auditoría Informática .....	35
1.8 Seguridad en bases de datos .....	36
1.9 Performance and Tunning .....	38
1.10 Tópicos avanzados de base de datos .....	39

## **CAPÍTULO 2 | PROYECTO PARA LA ADMINISTRACIÓN DE INVENTARIO A TRAVÉS DE UNA INTERFAZ WEB EN EL HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI**

2.1	Software necesario para el servidor .....	45
2.2	Preparando el servidor .....	51
2.3	Manual de Usuario para la Administración del Inventario de Equipo de Cómputo del Hospital de Especialidades CMN Siglo XXI .....	57
	<b>CONCLUSIONES</b> .....	71
	<b>FUENTES</b> .....	72
	<b>GLOSARIO</b> .....	73

La gran cantidad y diversidad de información relevante que se tiene en cualquier institución hoy en día, requiere de profesionales que tengan los conocimientos y habilidades para administrar de forma eficiente dicho recurso. Un mal uso de la información puede llevar a una incorrecta toma de decisiones, trayendo repercusiones innecesarias o catastróficas para las instituciones. Es por esta razón que se debe contar con personal preparado para el manejo de ésta, así como darle la importancia que requiere.

Un sistema de base de datos es básicamente un sistema computarizado para guardar registros; es decir, es un sistema cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información. La información en cuestión puede ser cualquier cosa que sea de importancia para el usuario u organización; en otras palabras, todo lo que sea necesario para auxiliarle en el proceso general de su administración.

La administración de bases de datos proporciona el conocimiento necesario para hacer un buen uso de la información almacenada en las mismas, a través del adecuado manejo y explotación de éstas haciendo uso de algún manejador de bases de datos.

En el presente trabajo, que consta de dos capítulos, se conjunta la teoría y la práctica para la adecuada interpretación de la información y la correcta toma de decisiones.

El primer capítulo es un informe del diplomado en administración de bases de datos, el cual como su nombre lo indica, trata absolutamente sobre la administración de bases de datos, desde sus conceptos básicos hasta temas especializados para la explotación y análisis de la información que estas contienen.

En el segundo capítulo se habla acerca del software requerido para la implementación de la aplicación así como del manual de usuario para la óptima y correcta administración del inventario del Hospital de Especialidades del Centro Médico Nacional Siglo XXI.

El objetivo general de este trabajo es explicar el qué, para qué y cómo se logra la administración de bases de datos, así como la implementación de una aplicación para la administración del inventario del hospital antes mencionado.

INFORME GLOBAL DEL DIPLOMADO  
EN ADMINISTRACIÓN DE BASES DE DATOS

---

## 1.1 SISTEMAS DE INFORMACIÓN Y EL MODELO DE DATOS RELACIONAL

La información es un recurso sumamente importante en cualquier organización. Debido a esto es necesario contar con el conocimiento necesario para su manejo, así como para una correcta y eficiente explotación. Los sistemas de información y el modelado de datos relacional son herramientas que permiten analizar y comprender cómo es procesada, distribuida y almacenada la información.

### SISTEMAS DE INFORMACIÓN

Un sistema de información es un conjunto de componentes que interactúan entre sí orientados al procesamiento y administración de datos con el objetivo de ayudar a la toma de decisiones.

El análisis de la información es un modelo de datos que consiste en la representación conceptual de la problemática que se desea resolver y cuya característica primordial es la claridad de su contenido.

La información se entiende como una entidad conceptual que permite reducir la incertidumbre acerca de algún estado o suceso.

Para que la información sea de valor debe contar con las siguientes características:

- **Accesible:** facilidad y rapidez con la que se obtiene información.
- **Clara:** se refiere a la integridad y entendimiento de la información.
- **Precisa:** que sea lo más exacta posible.
- **Propia:** debe haber relación entre el resultado y lo solicitado.
- **Flexible:** debe permitir adaptabilidad.
- **Verificable:** que se pueda examinar.
- **Organizada:** que esté estructurada.

### MODELO DE DATOS RELACIONAL

Un modelo de datos es un conjunto de herramientas conceptuales para describir los datos, las relaciones que existen entre ellos, la semántica asociada a los datos y las

restricciones de consistencia. En el modelo de datos relacional, los datos y las relaciones entre estos son representados a través de una colección de tablas, en las cuales los renglones (tuplas) equivalen a cada uno de los registros que contendrá la base de datos y las columnas corresponden a las características (atributos) de cada registro localizado en la tupla. En el modelo relacional el único elemento de representación es la tabla (véase figura 1.1).

Los datos se organizan en tablas llamadas relaciones, cada una de las cuales se implanta como un archivo. En terminología relacional, una fila en una relación representa un registro o tupla. Cada columna en una relación representa un campo o un atributo, así una relación se compone de una colección de tuplas (registros) cuyo grado está descrito por el número de atributos implantados como campos.



Fig. 1.1 Componentes de una tabla.

Este enfoque permite a los usuarios obtener información de la base de datos sin asistencia de sistemas profesionales de administración de información, es importante saber que las entradas en la tabla 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, nunca un conjunto de valores.

Todas las entradas de cualquier campo son de un solo tipo. Por ejemplo, un campo puede contener nombres de clientes y en otro puede tener fechas de nacimiento. Cada campo posee un nombre único, el orden de los campos no es de importancia para la tabla. Cada campo tiene un dominio, que es una descripción física y lógica de valores permitidos.

## REGLAS DE CODD

En 1984 Codd<sup>1</sup> publicó 12 reglas que un verdadero sistema relacional debería de cumplir. Un sistema podrá considerarse más relacional cuanto más siga estas reglas.

- |  |   |
|--|---|
| <b>Regla 0</b>   | Cualquier DBMS <sup>2</sup> que proclame ser relacional, deberá manejar, completamente, las bases de datos por medio de sus capacidades relacionales.   |
| <b>1 Regla de Información</b>                                      | Toda la información dentro de una base de datos relacional se representa de manera explícita a nivel lógico y exactamente de una sola manera, como valores en una tabla.  |
| <b>2 Regla de Acceso Garantizado</b>                               | Se garantiza que todos y cada uno de los datos (valor atómico) en una base de datos relacional pueden ser leídos recurriendo a una combinación del nombre de la tabla, valor de la llave primaria y nombre de la columna.   |
| <b>3 Manejo Sistemático de los Valores Nulos</b>                   | En un DBMS totalmente relacional se soportan los valores nulos (que son distintos a una cadena de caracteres vacía o de una cadena con caracteres en blanco o de cero o cualquier otro número), para representar información faltante o no aplicable de una forma consistente, independientemente del tipo de dato. |
| <b>4 Catálogo Dinámico en Línea Basado en el Modelo Relacional</b> | La descripción de la base de datos se representa en el nivel lógico de la misma forma que los datos ordinarios, de tal suerte que los usuarios autorizados puedan aplicar el mismo lenguaje relacional para consultarla, que aquel que emplean para con sus datos habituales.                                       |

<sup>1</sup> Edgar Frank Codd. Científico informático inglés, conocido por sus aportes a la teoría de bases de datos relacionales.

<sup>2</sup> Data Base Management System (Sistema Administrador de Base de Datos).

<b>5 Regla del Sublenguaje de Datos Completo</b>	<p>Se debe contar con un sublenguaje que contemple:</p> <ul style="list-style-type: none"> <li>• La definición de datos.</li> <li>• La definición de vistas.</li> <li>• La manipulación de datos.</li> <li>• Las restricciones de integridad.</li> <li>• Limitantes de transacción: iniciar (begin), realizar (commit), deshacer (rollback).</li> </ul>
<b>6 Regla de Actualización de Vistas</b>	Todas las vistas que teóricamente sean actualizables deberán ser actualizadas por medio del sistema.
<b>7 Inserción, Actualización y Eliminación de Alto Nivel</b>	La posibilidad de manejar una relación base o una relación derivada como un sólo operador se aplica a la lectura, inserción, modificación y eliminación de datos.
<b>8 Independencia Física de los Datos</b>	Los programas de aplicación y la actividad en terminales no deberán ser afectados por cambios en el almacenamiento físico de los datos o en el método de acceso.
<b>9 Independencia Lógica de los Datos</b>	Los programas de aplicación y la actividad en terminales no deberán ser afectados por cambios de cualquier tipo que preserven la información y que teóricamente permitan la no afectación en las tablas base.
<b>10 Independencia de la Integridad</b>	Las restricciones de integridad de una base de datos deberán poder definirse en el mismo sublenguaje de datos relacional y deberán almacenarse en el catálogo, no en los programas de aplicación.
<b>11 Independencia de la Distribución</b>	Un DBMS relacional tiene independencia de distribución.

## 12 Regla de la No Subversión

Si un sistema relacional tiene un lenguaje de bajo nivel (un sólo registro cada vez), ese bajo nivel no puede ser utilizado para suprimir las reglas de integridad y las restricciones expresadas en el lenguaje relacional de nivel superior (múltiples registros a la vez).

### 1.2 SISTEMAS ADMINISTRADORES DE BASES DE DATOS RELACIONALES (RDBMS)

Entre la base de datos física (los datos tal y como están almacenados en la realidad) y los usuarios del sistema, existe un nivel de programas denominado manejador de bases de datos o en la mayoría de los casos, el sistema administrador de bases de datos DBMS.

Un RDBMS es el conjunto de programas que permiten la creación, actualización y administración para una o varias bases de datos relacionales.

Algunas características de los RDBMS son:

- Facilitan la integridad, seguridad y acceso de los datos.
- Los datos se almacenan como mínima redundancia.
- Las aplicaciones son independientes del almacenamiento físico de los datos.

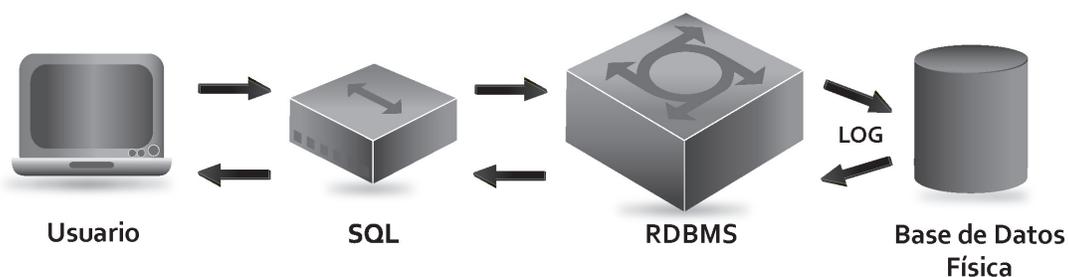
Un RDBMS debe permitir las siguientes condiciones para cualquier base de datos:

- Los datos han de estar almacenados juntos.
- Tanto los usuarios finales como los programas de aplicación no necesitan conocer los detalles de las estructuras de almacenamiento.
- Los datos son compartidos por diferentes usuarios y programas de aplicación; existe un mecanismo común para la inserción, actualización, borrado y consulta de los datos.

- Los procedimientos de actualización y recuperación comunes y bien determinados, habrán de ser capaces de conservar la integridad, seguridad y confidencialidad del conjunto de datos.
- Tanto datos como procedimientos pueden ser transportables conceptualmente a través de diferentes RDBMS.

La figura 1.2 ilustra la manera en que el usuario realiza una petición al RDBMS. Conceptualmente sucede lo siguiente:

1. El usuario realiza una petición a la base de datos empleando algún sublenguaje de datos, SQL<sup>3</sup> en la mayoría de los casos.
2. El RDBMS interpreta esa solicitud y la analiza.
3. El RDBMS inspecciona en orden el esquema externo de ese usuario, la correspondencia externa/conceptual asociada, el esquema conceptual, la correspondencia conceptual/interna y la definición de la estructura de almacenamiento.
4. El RDBMS ejecuta las operaciones necesarias sobre la base de datos almacenada y devuelve una respuesta al usuario.



*Fig. 1.2 Interacción entre usuario y RDBMS.*

<sup>3</sup> Structured Query Language (Lenguaje de Consulta Estructurado).

## COMPONENTES DE UN RDBMS

A continuación se describen los componentes de un RDBMS, ilustrados en la figura 1.3.

### Lenguaje de Definición de Datos (DDL)

Permite definir la estructura de la base de datos mediante comandos como crear (Create), eliminar (Drop), o alterar (Alter). Se utiliza para crear, eliminar o modificar tablas, índices, vistas, triggers y procedimientos.

### Lenguaje de Manipulación de Datos (DML)

Se utiliza para realizar la consulta y edición de la información contenida en la base de datos, esto implica: seleccionar (Select), insertar (Insert), borrar (Delete), modificar (Update).

### Lenguaje de Control de Datos (DCL)

Se utiliza para la definición de los privilegios de control de acceso y edición a los elementos que componen la base de datos (seguridad), es decir, permitir (Grant) o revocar (Revoke) el acceso.

Los permisos a nivel base de datos pueden otorgarse a usuarios para ejecutar ciertos comandos dentro de la base o para que puedan manipular objetos y los datos que puedan contener éstos.

### Diccionario de Datos (DD)

El contenido del diccionario de datos es un conjunto de metadatos que definen los objetos de una base de datos.

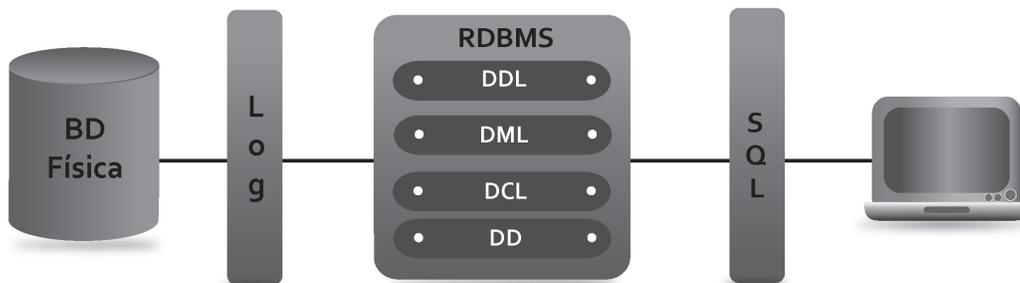


Fig. 1.3 Arquitectura de un RDBMS.

## TIPOS DE DATOS USADOS POR LOS RDBMS

Los tipos de datos SQL se clasifican en 13 tipos primarios y de varios sinónimos válidos reconocidos por dichos tipos de datos dependiendo del RDBMS, mostrados en la tabla 1.1.

Tipo de Dato	Sinónimos	Tamaño	Descripción
BINARY	VARBINARY BINARY VARBINARY BIT VARYING	1 byte por caracter	Se puede almacenar cualquier tipo de datos en un campo de este tipo. Los datos no se traducen (por ejemplo, a texto). La forma en que se introducen los datos en un campo binario indica cómo aparecerán al mostrarlos.
BIT	BOOLEAN LOGICAL LOGICAL 1 YESNO	1 byte	Valor Sí y No, y campos que contienen solamente uno de los dos valores.
BYTE	INTEGER1 TINYINT	1 byte	Un número entero entre 0 y 255.
COUNTER	AUTOINCREMENT SEQUENCE		Se utiliza para campos contadores cuyo valor se incrementa automáticamente al crear un nuevo registro.
MONEY	CURRENCY	8 bytes	Un número entero comprendido entre -922.337.203.685.477,5808 y 922.337.203.685.477,5807.
DATETIME	DATE TIME	8 bytes	Un valor de fecha u hora entre los años 100 y 9999.
DECIMAL	NUMERIC DEC	17 bytes	Un tipo de datos numérico exacto con valores comprendidos entre 1028 -1 y -1028 -1. Puede definir precisión (1-28) y la escala (0-precisión definida). La precisión y la escala predeterminadas son 18 y 0, respectivamente.
REAL	SINGLE FLOAT4 IEEESINGLE	4 bytes	Un valor de coma flotante de precisión simple con un intervalo comprendido entre - 3,402823E38 y -1,401298E-45 para valores negativos, y desde 1,401298E-45 a 3,402823E38 para valores positivos, y 0.

Tipo de Dato	Sinónimos	Tamaño	Descripción
FLOAT	DOUBLE FLOAT8 IEEEDOUBLE NUMBER	8 bytes	Un valor de coma flotante de precisión doble con un intervalo comprendido entre -179769313486232E308 y -4,94065645841247E-324 para valores negativos, y desde 4,94065645841247E-324 a 1,79769313486232E308 para valores positivos, y 0.
SMALLINT	SHORT INTEGER2	2 bytes	Un entero corto entre -32 768 y 32 767
INTEGER	LONG INT INTEGER4	4 bytes	Un entero largo entre -2.147.483.648 y 2.147.483.647
IMAGE	LONGBINARY GENERAL OLEOBJECT	Lo que se requiera	Desde cero hasta un máximo de 2.14 gigabytes.
TEXT	LONGTEXT LONGCHAR MEMO NOTE NTEXT	2 bytes por caracter. (Consulte manual)	Desde cero hasta un máximo de 2.14 gigabytes.
CHAR	TEXT(n) ALPHANUMERIC CHARACTER STRING VARCHAR VARYING NCHAR NATIONAL NATIONAL CHAR	2 bytes por caracter. (Consulte manual)	Desde cero a 255 caracteres.

Tabla 1.1 Tipos de datos usados por los RDBMS.

## SISTEMAS MANEJADORES DE BASES DE DATOS

Los sistemas manejadores de bases de datos se clasifican por su ámbito en: *comerciales* y *de software libre*. Así como también es posible clasificarlos por el volumen de información que pueden manejar en: *corporativos* y *departamentales*.

A continuación se describen las características principales de algunos manejadores de bases de datos.

## MICROSOFT SQL SERVER

Características:

- Compatibilidad con estándares de W3C, incluyendo XML, Xpath, XSL, HTTP.
- Obtiene código XML de las consultas realizadas con SQL.
- Manipulación de documentos XML.
- Manejo de bases de datos distribuidas.
- Manejo de varias particiones físicas para almacenamientos de datos flexibles.
- Permite realizar algunas tareas de mantenimiento y administración de la base de datos sin tener que dar de baja el servicio.
- Permite realizar acciones OLAP (Online Analyzing Processing), herramienta que permite analizar datos almacenados en una Base de Datos, por medio de cubos de información.
- Consulta y modificación de cubos virtuales de manera gráfica.
- Conectividad con clientes ODBC y JDBC.

Plataformas Soportadas:

- Sólo compatible con sistemas operativos Windows.

## ORACLE

Características:

- Ofrece varias plataformas de desarrollo para Internet y aplicaciones tradicionales, tales como: XML, Enterprise Java Engine, SQL y PL/SQL, C, C++, entre otras.
- Soporte Unicode.

- Extiende las habilidades de una base de datos para Internet.
- Amplía distintos mecanismos para protección de datos.
- Soporta OLTP y OLAP.
- Contiene mecanismos de gran funcionalidad y flexibilidad para compartir la información almacenada en la base de datos con otras bases de datos o aplicaciones.
- Conectividad con clientes ODBC y JDBC.
- Soporte para BLOB.
- Ofrece escalabilidad y performance sin modificar las aplicaciones instaladas.
- Soporta columnas con cifrado de datos.
- Permite replicación de bases de datos (bases de datos distribuidas).
- Ofrece distintas herramientas para la administración de la base de datos.
- Redefinición de tablas en línea.
- Respaldo y recuperación en línea.

#### Plataformas Soportadas:

- |            |             |
|------------|-------------|
| • Solaris. | • HP Alpha. |
| • HP-UX.   | • Linux.    |
| • AIX.     | • Windows.  |

## MySQL

#### Características:

- Soporta los estándares ANSI.
- Contiene esquemas de almacenamiento independiente que se pueden seleccionar de acuerdo a las necesidades.
  - InnoDB para transacciones y bloqueo de registros.
  - MyISAM sin transacciones.
- Soporte para SSL.
- Consultas con manejo de cache que puede incrementar el performance de la base de datos en un 200%.

- Permite manejo de replicación de bases de datos.
- Soporta indexado de texto.

#### Plataformas Soportadas:

- FreeBSD.
- Linux.
- Mac OS X.
- Solaris.
- Windows.

### PostgreSQL

#### Características:

- Base de datos de distribución libre.
- Velocidad.
- Confiabilidad.
- Flexibilidad.
- Bajo costo de operación.
- Conformación a estándares ANSI.
- Estrategia de almacenamiento MVCC para grandes volúmenes.
- Soporta replicación de bases de datos.
- Interfases nativas para ODBC, JDBC, C, C++, PHP, Perl, TCL, XML.
- Soporta SSL nativo.

#### Plataformas Soportadas:

- FreeBSD.
- Linux.
- Mac OS X.
- Solaris.
- Windows.

### 1.3 STRUCTURED QUERY LANGUAGE (SQL)

SQL es un lenguaje de consulta que se puede utilizar para organizar, gestionar y recuperar datos almacenados en una base de datos relacional, siendo adoptado como estándar en la industria en 1986.

SQL es un lenguaje fácil de entender ya que su estructura utiliza palabras en inglés, lo que lo hace fácil de aprender y utilizar, además, las instrucciones se enfocan a qué buscar, dejando al RDBMS la tarea de cómo recuperar la información.

#### EVOLUCIÓN DEL SQL

A continuación se describen las principales innovaciones que introdujeron las versiones más relevantes de SQL.

#### ANSI SQL 89

En 1989, tanto ANSI<sup>4</sup> como ISO<sup>5</sup>, publicaron estándares que definían al modelo relacional en el manejo de Bases de Datos.

Estos estándares son: ANSI X3.135-1989 e ISO/IEC9075:1989.

Características:

- Agregan la capacidad conocida como integridad referencial y la descripción de todo el modelo relacional.
- Se definió que el lenguaje SQL está compuesto por comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan para definir y manipular la base de datos.
- Se establecen los elementos de un RDBMS (DDL, DML y DCL), así como las instrucciones y sintaxis relacionadas con cada uno de ellos.
- Establecimiento de las cláusulas del comando **SELECT (FROM, WHERE, GROUP BY, HAVING, ORDER BY)**.
- Definición de los operadores lógicos: **AND, OR y NOT**.
- Definición de los operadores de comparación ( =, !=, >, < ).

<sup>4</sup> American National Standards Institute (Instituto Nacional de Normalización Estadounidense).

<sup>5</sup> International Organization for Standardization (Organización Internacional de Estandarización).

- Se determinan las funciones de agregado, tales como: **AVG, COUNT, SUM, MAX, MIN.**

## ANSI SQL 92

### Características:

- Toma todas las características definidas en el estándar ANSI SQL 89.
- Permite la definición de esquemas.
- Permite la definición de dominios por parte de los usuarios, es decir, tipos de datos definidos por el usuario.
- Menciona las consideraciones para realizar consultas sencillas, multi-tablas y subconsultas.
- Incluye los operadores **EXISTS** y **NOT EXISTS**.
  1. **SELECT** columns FROM table WHERE EXISTS (subquery);
  2. **SELECT** \* FROM table WHERE not exists (subquery);
- Contempla el uso de la palabra **DISTINCT** en una consulta.
- Menciona algunas consideraciones para el uso de las cláusulas **GROUP BY** y **HAVING**.
- Especifica la definición de vistas en una base de datos.

## ANSI SQL 99

### Características:

- Toma todas las características definidas en los estándares ANSI SQL 89 y 92.
- Incluye nuevos tipos de datos escalares: **BOOLEAN, CLOB** (objeto de caracteres largo) y **BLOB** (objeto binario grande).
- Presenta dos nuevos operadores de totales: **EVER** y **ANY**.
- Incorpora generadores de tipo de dato: **REF, ARRAY** y **ROW**.
- Soporta una opción **LIKE** en **CREATE TABLE**, lo cual permite que todas o algunas definiciones de columna de una nueva tabla sean copiadas a partir de otra ya existente.

- Incluye la cláusula **WITH** para introducir nombres abreviados para determinadas expresiones.
- Incorpora una nueva expresión de condición **IS DISTINCT** para la cláusula FROM.

## SQL 2003

El estándar SQL 2003 hace pequeñas modificaciones a todas las partes de SQL 99 e introduce algunas nuevas características como:

- Características relacionadas con XML.
- Ventana de funciones.
- Generador de la secuencia, lo que permite secuencias estandarizadas.
- Dos nuevos tipos de columna: valores auto-generados y la identidad de columnas.
- Nueva sentencia **MERGE**.
- Eliminación de los mal implementados tipos de datos **"BIT"** y **"BIT VARYING"**.

## ELEMENTOS DE UNA BASE DE DATOS RELACIONAL

Antes de comenzar a trabajar con SQL, es necesario conocer los elementos que intervienen en la definición de la información en una base de datos, para poder manipularla de manera adecuada.

### TABLAS

Las tablas son el elemento fundamental que compone a una base de datos relacional porque todo gira en torno a ellas. Las tablas son estructuras de almacenamiento que albergan la información en forma de registros (renglones) que deben de ser identificados de manera única y esto se logra a través de una llave primaria.

### LLAVES PRIMARIAS

Una llave primaria permite identificar de manera única un registro dentro de una tabla. Al crear una llave de este tipo se genera automáticamente un índice de valores únicos, por lo que los valores de los campos que involucra la llave primaria no se pueden repetir ni ser nulos.

## LLAVES FORÁNEAS

Las llaves foráneas son atributos de una tabla que hacen referencia a la llave primaria de otra tabla. Estas permiten establecer relaciones entre las distintas tablas que existen dentro de la base de datos. La mayoría de los RDBMS implementan restricciones (CONSTRAINTS) cuando se genera una llave foránea, de este modo asegura la integridad de la información almacenada en la base de datos.

## ÍNDICES

Un índice es una estructura de almacenamiento físico. Los índices ayudan al Servidor SQL a localizar datos y son transparentes para el usuario. El propósito principal de un índice es proporcionar un acceso más rápido a los datos, aunque en algunos casos su propósito es asegurar que el contenido de un campo sea único.

## REGLAS

Las reglas dentro de la base de datos, permiten definir condiciones que debe cumplir la información para que sea válida. En la práctica, muchas de estas reglas no se definen en la base de datos sino mediante la lógica de una aplicación que tiene acceso a la información, ya que el exceso de reglas puede disminuir el rendimiento del RDBMS en los procesos de inserción y modificación de información.

## MANIPULACIÓN DE DATOS

La mayor parte de las consultas con SQL gira entorno a cuatro comandos:

- **Select.** Permite seleccionar (recuperar) información de una tabla.
- **Insert.** Permite agregar información a una tabla.
- **Delete.** Permite eliminar información de una tabla.
- **Update.** Permite actualizar información que existe en una tabla.

Para el empleo de cualquiera de los comandos mencionados es indispensable tomar en cuenta dos puntos:

1. Para expresar un valor de tipo alfanumérico o fecha es requisito entrecomillarlo con comillas simples.
2. Todo valor que no se especifique entre comillas simples será interpretado como tipo de dato numérico.

## ALGEBRA RELACIONAL

SQL se basa en el álgebra relacional, por ello es importante conocer las operaciones del álgebra relacional y su relación con SQL.

### SELECCIÓN

La operación de selección genera un subconjunto de los renglones de una tabla, con base en un criterio (restricción) establecido. Esta operación del álgebra relacional es realizada por la cláusula WHERE.

### PROYECCIÓN

La proyección selecciona y genera un subconjunto con los atributos (columnas) indicados de una tabla. También es conocida como operación vertical. Esta operación es realizada por la cláusula SELECT.

### UNIÓN

La operación unión realiza la misma acción que en el álgebra de conjuntos. Esta operación se realiza con la cláusula UNION.

## PRODUCTO CARTESIANO

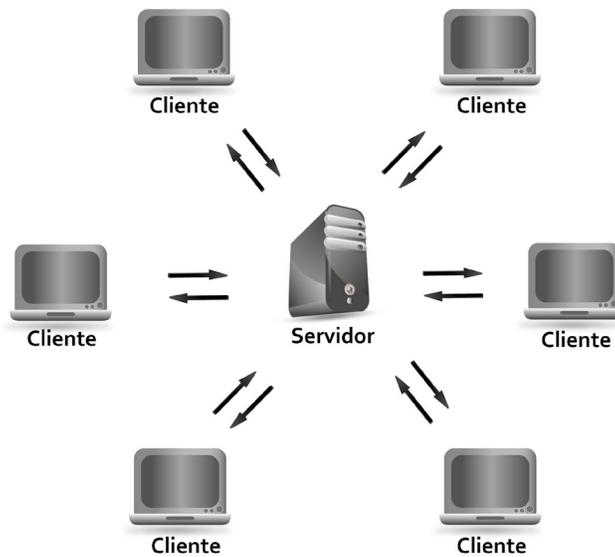
El producto cartesiano es el producto cruz entre 2 tablas. El resultado es la combinación de cada renglón de una tabla con cada renglón de la otra tabla. En SQL esta operación se lleva a cabo cuando se ocupa la cláusula FROM especificando 2 o más tablas y no se ha especificado una restricción que indique la relación entre las tablas.

## JOIN

La operación Join es en esencia un producto cartesiano donde se selecciona los renglones que satisfagan las condiciones indicadas que establecen la relación entre las tablas involucradas. Esta es una operación muy común en las bases de datos relacionales.

## 1.4 ACCESO A DATOS A TRAVÉS DE LA PROGRAMACIÓN DE CLIENTES

Las aplicaciones web, en su gran mayoría, funcionan basadas en la arquitectura cliente-servidor (veáse figura 1.4). Esta arquitectura es un sistema en que uno o más equipos (clientes) solicitan a una máquina central (servidor) que ejecute alguna tarea específica. Este modelo proporciona al usuario acceso transparente a las aplicaciones, datos, servicios o a cualquier otro recurso dentro del grupo de trabajo.



*Fig. 1.4 Arquitectura Cliente-Servidor.*

Los clientes interactúan con el usuario, usualmente de forma gráfica, y a su vez se comunican con procesos auxiliares que se encargan de establecer la conexión con el servidor, enviar la petición, recibir respuesta, manejar fallas y realizar actividades tanto de sincronización como 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 para satisfacer el pedido, recibir su respuesta y enviarla al cliente. La programación de clientes se puede realizar a través de diferentes lenguajes de programación siendo los más utilizados: PHP, JSP y ASP.

## PHP

PHP (acrónimo de PHP: Hypertext Preprocessor) es un lenguaje de programación interpretado del lado del servidor incrustado dentro de código HTML. El cliente recibirá los resultados de ejecutar algún script, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido. Existe un gran número de servidores y aplicaciones para montar servidores que soporten PHP, siendo Apache la opción más usada. Esto permite una compatibilidad con diferentes plataformas ya sean comerciales o libres.

## JSP

JSP (Java Server Pages) combina HTML<sup>6</sup> con fragmentos de Java para producir páginas web dinámicas. Una página JSP es un archivo de texto plano que consiste en contenido HTML o XML con elementos JSP. Cuando un cliente pide una página JSP de algún sitio web y no se ha ejecutado antes, la página es inicialmente pasada al motor de JSP, el cual compila la página convirtiéndola en Servlet, la ejecuta y devuelve el contenido de los resultados al cliente. Cuando ya existe el Servlet sólo se manda a llamar, sin necesidad de volver a compilarse.

Para usar aplicaciones con Servlets y JSP se deberá tener en el servidor: Java Servlet, Java Server Pages y un servidor web con capacidad para Servlets como Apache Tomcat.

---

<sup>6</sup> HyperText Markup Language (Lenguaje de Marcado de Hipertexto).

## ASP

ASP (Active Server Pages) es un lenguaje creado por Microsoft para la creación de páginas dinámicas generadas dinámicamente en el servidor. Estas páginas contienen código HTML y fragmentos de código que son utilizados para llevar a cabo la interacción entre el servidor y el navegador.

Para procesar una página ASP es necesario un servidor Web de Microsoft, siendo IIS (Internet Information Server) el más distribuido. Para plataformas Unix es necesario añadir software que actúe de intérprete, siendo algunos de los más conocidos: Chili-soft e Instant ASP.

## 1.5 FUNDAMENTOS DE SISTEMAS OPERATIVOS

El sistema operativo es el programa necesario para la interacción con la computadora, es un conjunto de programas que a través de alguna interfaz gráfica permiten al usuario hacer uso del hardware, así como la interacción con otros programas.

Las funciones básicas del sistema operativo son administrar los recursos de la máquina, coordinar el hardware, así como organizar archivos y directorios en los dispositivos de almacenamiento disponibles.

Cuando se habla de sistemas operativos se debe tener bien en claro los siguientes conceptos que resultan de base para cualquier sistema operativo:

- **Proceso:** es cualquier aplicación que se esté ejecutando.
- **Llamada a sistema:** son peticiones que realizan las aplicaciones solicitando algún servicio.
- **Archivos:** secuencia de bits.

Los principales componentes de un sistema operativo son:

- Administrador de procesos (scheduler).
  - Crear y destruir los procesos.
  - Parar y reanudar los procesos.
  - Ofrecer mecanismos para que se comuniquen y sincronicen.

- Administrador de dispositivos E/S.
  - Gestionar el almacenamiento temporal de E/S.
  - Servir las interrupciones de los dispositivos de E/S.
  
- Administrador de Memoria.
  - Conocer qué partes de la memoria están siendo utilizadas y por quién.
  - Decidir qué procesos se cargarán en memoria cuando haya espacio disponible.
  - Asignar y reclamar espacio de memoria cuando sea necesario.
  
- Sistema de Archivos.
  - Construir y eliminar archivos y directorios.
  - Ofrecer funciones para manipular archivos y directorios.
  - Establecer la correspondencia entre archivos y unidades de almacenamiento.
  - Realizar copias de seguridad de archivos.

## 1.6 ADMINISTRACIÓN DE BASES DE DATOS

La administración de la base de datos es un conjunto de tareas para la adecuada planeación, creación y gestión de éstas. Esta responsabilidad recae principalmente sobre el administrador de bases de datos.

### ADMINISTRADOR DE BASES DE DATOS

Lleva a cabo tareas administrativas ajenas a la aplicación que use la base de datos. El administrador de las bases de datos no es necesariamente único; el rol puede ser otorgado a cualquier número de cuentas individualmente y para ello las funciones deben estar centralizadas o muy bien coordinadas. Las tareas del DBA<sup>7</sup> incluyen:

- Decidir el contenido de la base de datos.
- Crear la estructura de almacenamiento y los métodos de acceso.
- Instalación y configuración del DBMS.
- Administrar y controlar la seguridad física y lógica de la base de datos.
- Otorgar permisos de acceso y prioridades a los diferentes usuarios.

<sup>7</sup> Data Base Administrator (Administrador de Bases de Datos).

- Modificar la base de datos o la descripción de la organización física.
- Monitorear el comportamiento y crecimiento de la base de datos.
- Realizar procedimientos de respaldo y depuración de la base de datos.
- Salvaguardar la documentación, respaldos y diccionario de datos tanto de la base de datos como de sus aplicaciones.
- Contar con procedimientos de contingencia y recuperación de la base de datos.
- Diagnóstico de problemas del sistema así como la corrección de los mismos.

## CREACIÓN Y MANEJO DE LAS BASES DE DATOS

Es responsabilidad del administrador de bases de datos participar en el diseño de la base de datos, así como el tener claro qué tipo de hardware se requiere para su implementación y que RDBMS es el que conviene utilizar.

La creación así como la eliminación de las bases de datos es llevada a cabo bajo la responsabilidad del administrador de bases de datos. Dentro de la creación se debe tener establecido el tipo de base de datos, así como la forma en la que será almacenada.

Una vez creada la base de datos se deberá configurar el RDBMS para obtener el mejor rendimiento. Existen muchos parámetros de configuración que afectan el comportamiento del sistema de base de datos.

## PERFILES DE USUARIO EN UNA BASE DE DATOS

Los perfiles sirven como medio para conceder privilegios sobre el RDBMS a un usuario que los requiera. Estos permisos pueden verse reflejados sobre objetos o sobre el mismo sistema. En los RDBMS ya vienen predeterminados algunos, sin embargo no en todos se pueden crear nuevos roles.

<b>Administrador de Base de Datos (DBA)</b>	Tiene control total sobre el RDBMS. Puede crear, modificar, borrar bases de datos, así como usuarios.
---	---

**Dueño de Base de Datos (DBO)**

Puede realizar las mismas tareas de gestión de bases de datos que un DBA pero solamente a las bases de datos de las cuales es dueño, no puede operar fuera de las bases de datos que le fueron asignadas.

**Oficial de Sistema de Seguridad (SSO)**

El Oficial del Sistema de Seguridad es responsable de la seguridad del sistema y tiene las siguientes funciones:

- Crear cuentas de usuario.
- Bloquear cuentas.
- Otorgar y revocar los roles de SSO y operador.
- Cambiar contraseña a cualquier cuenta en el sistema.
- Poner intervalos de expiración a las contraseñas.

**Operador (Oper)**

El operador del sistema es responsable de respaldar y recuperar todas las Bases de Datos del Sistema.

## RESPALDO Y RESTAURACIÓN DE BASES DE DATOS

Los motivos para hacer un respaldo van desde un desastre natural o robo, hasta el daño accidental por parte de un usuario. Sin un respaldo reciente no existe posibilidad alguna de recuperar los datos. Las bases de datos se deben respaldar periódicamente y en horarios que no comprometan el desempeño de las aplicaciones que hacen uso de éstas.

Una alternativa para respaldar, es hacer una copia de los archivos donde se almacenan los datos de la base de datos. Se puede utilizar cualquier método del sistema de archivos del sistema operativo en el que se encuentre instalado el RDBMS.

Otro método es utilizar SQL-Dump, la idea básica es generar un archivo de texto con sentencias SQL, para reconstruir la base de datos al mismo estado que tenía a la hora de la creación del respaldo.

Es importante la manera de restaurar los datos en el sentido de que sea rápida, segura y que además devuelva el funcionamiento normal a la empresa con un mínimo de retrasos.

## 1.7 BUENAS PRÁCTICAS EN LA FUNCIÓN DE LA ADMINISTRACIÓN

La implementación de buenas prácticas nos garantiza una mejor administración de la información así como la seguridad de ésta, también podrá mejorar los servicios y procesos a través de una adecuada documentación y metodología.

### 1.7.1 SGSI

El SGSI<sup>8</sup> trata, como su nombre lo indica sobre la gestión de la seguridad. La función de un SGSI es garantizar que los riesgos de la seguridad de la información sean conocidos, asumidos, gestionados y minimizados por la organización de una forma documentada, sistemática, estructurada, periódica y eficiente.

#### Ventajas

- Se evitan muchos errores o se detectan a tiempo gracias a los controles adecuados; y si se producen, se cuenta con los planes de incidencias para dar respuesta efectiva y en el tiempo mínimo.
- Se evita fuga de información o la dependencia con personas internas y externas.
- La gestión de la seguridad se convierte en un ciclo de vida metódico y controlado.
- El conjunto de acciones adoptadas reducirá al mínimo todo riesgo por robo, fraude, error humano (intencionado o no), mal uso de instalaciones y equipo a los cuales está expuesto el manejo de información.
- Crea conciencia y compromiso de seguridad en todos los niveles de la empresa, no sólo al implantarla, sino que será permanente pues se trata de un ciclo.
- Crea un marco legal que protegerá a la empresa en muchos flancos que antes no tenía cubiertos.
- La certificación de la seguridad puede ser una oportunidad de negocio más que un costo.

---

<sup>8</sup> Sistema de Gestión de la Seguridad de la Información.

## Desventajas

- Al iniciar este conjunto de tareas, no cabe duda que se está sobrecargando el ritmo habitual de trabajo de toda la organización.
- El cúmulo de actividades realizadas exige un mantenimiento.
- El mantenimiento del nivel alcanzado, requerirá inexorablemente un esfuerzo continuo de toda la organización al completo.

### 1.7.2 ITIL

ITIL<sup>9</sup> es un estándar para la administración de servicios por medio de procedimientos, roles, tareas, y responsabilidades que se pueden adaptar a cualquier organización de TI<sup>10</sup>. Genera una descripción detallada de mejores prácticas, que permitirán tener mejor comunicación y administración en la organización de TI. Proporciona los elementos necesarios para determinar objetivos de mejora y metas que ayuden a la organización a madurar y crecer.

## Ventajas

- Mejora la comunicación con los clientes y usuarios finales a través de los diversos puntos de contacto.
- Los servicios se detallan en lenguaje del cliente y con más detalle.
- Se maneja mejor la calidad y los costos de los servicios.
- La entrega de servicios se enfoca más al cliente mejorando con ello la calidad de los mismos y la relación entre el cliente y el departamento de TI.
- Una mayor flexibilidad y adaptabilidad de los servicios.
- La organización TI desarrolla una estructura más clara, se vuelve más eficaz, y se centra más en los objetivos de la organización.
- La administración tiene un mayor control, se estandarizan e identifican los procedimientos, y los cambios resultan más fáciles de manejar.
- A través de las mejores prácticas de ITIL se apoya al cambio en la cultura de TI y su orientación hacia el servicio, y se facilita la introducción de un sistema de administración de calidad.

<sup>9</sup> Information Technology Infrastructure Library (Biblioteca de Infraestructura de Tecnologías de Información).

<sup>10</sup> Tecnologías de Información.

## Desventajas

- Tiempo y esfuerzo necesario para su implementación.
- Que no se dé el cambio en la cultura de las áreas involucradas.
- Que no se vea reflejada una mejora, por falta de entendimiento sobre procesos, indicadores y como pueden ser controlados.
- Que el personal no se involucre y se comprometa.
- La mejora del servicio y la reducción de costos puede no ser visible.
- Que la inversión en herramientas de soporte sea escasa. Los procesos podrán parecer inútiles y no se alcanzarán las mejoras en los servicios.

### 1.7.3 ISO 20 000

La ISO 20 000 es el estándar internacional para la Administración de Servicios de TI.

## Ventajas

- La provisión de servicios TI es orientada al cliente y se mejora la relación con este.
- Los servicios son descritos más detalladamente en idioma del usuario.
- La calidad y el costo de los servicios son monitoreados ganándose control.
- La comunicación entre los usuarios y el departamento de sistemas se simplifica y se hace más clara consiguiendo una rápida mejora percibida.
- El departamento de TI desarrolla una más clara estructura, se hace más eficiente y focalizada en los objetivos corporativos.
- La administración es más controlada junto con los cambios que conlleva la misma.
- El nuevo mapa de procesos provee de un framework que permite decidir la tercerización de ciertos procesos en caso de ser convenientes.
- Se liderea un cambio cultural hacia la provisión de servicios.
- Provee un uniforme marco de referencia de comunicación interna y externa hacia proveedores junto con la estandarización e identificación de procedimientos.

## Desventajas

- Cuando el proceso de cambio es manejado sólo con recursos internos, se corre el riesgo de no poder cambiar el estado actual, porque implicaría marcar errores en lugar de oportunidades de mejora, y a los responsables en lugar de líderes del cambio.

- Un enfoque demasiado ambicioso puede llevar a la frustración, porque los objetivos no son cumplidos.
- Si la estructura de procesos se transforman en un objetivo en sí mismo, la calidad del servicio puede verse afectada. En este caso, los procedimientos se transforman en obstáculos burocráticos que quieren ser evitados.
- Puede no haber mejoras debido a la falta de entendimiento sobre el alcance de los procesos, los indicadores de desempeño o cómo estos deberían ser controlados.
- La mejora en la provisión de los servicios y la reducción de costos pueden no ser visibles.
- Dejar el desarrollo de las estructuras de procesos a sólo especialistas puede provocar el aislamiento de los mismos a través de opiniones sobre nuevas direcciones que no serán seguidas por los otros departamentos.
- Si hay una insuficiente inversión en herramientas de soporte, los procesos no van a ser mejorados, ya que requieren cierto grado de automatización.
- Recursos adicionales pueden ser requeridos si la organización ya tiene sobrecargados recursos para las actividades de manejo de servicios de TI.

#### 1.7.4 PLAN DE CONTINUIDAD DEL NEGOCIO

Comprende la planeación para recuperación de desastres así como la planeación para el restablecimiento del negocio.

##### Ventajas

- Identificar potenciales desastres y sus efectos mediante la evaluación del riesgo.
- Adoptar medidas preventivas para minimizar la probabilidad de ocurrencia.
- Desarrollar una respuesta organizada en caso de presentarse un desastre.
- Asegurar que los procesos de negocio fundamentales continúen durante el período de recuperación.

### 1.7.5 COBIT

Modelo de control interno dirigido a las necesidades de control de las TI. Es una herramienta innovadora para el gobierno de las TI que ayuda a la gerencia a comprender y administrar los riesgos asociados con estas.

El objetivo principal del proyecto COBIT <sup>11</sup> es el desarrollo de políticas claras y buenas prácticas para la seguridad y el control de TI.

#### Ventajas

- Contribuye a salvar las brechas entre riesgos de negocios, necesidades de control y aspectos técnicos.
- Es aplicable a todos los tamaños y tipos de organización.
- Es una herramienta válida para los administradores y los usuarios.
- Permite determinar el alcance de la tarea de auditoría e identificar los controles mínimos.
- Es posible emplearla como una herramienta de autoevaluación del área informática.
- Observa y se nutre de prácticamente la totalidad de los estándares y regulaciones internacionales.
- Incorpora los principios de gestión de la calidad total, reingeniería de empresas e integra los dos modelos de control: los orientados a las Tecnologías de la Información y los orientados a los objetivos empresariales.

#### Desventajas

- Está orientado a los auditores informáticos.
- Resulta un modelo ambicioso que requiere de profundidad en el estudio.

### 1.7.6 AUDITORÍA INFORMÁTICA

La auditoría informática es un proceso que mediante técnicas y procedimientos aplicados por personal ajeno a la organización, evalúa las funciones de las TI así como el cumplimiento de los objetivos institucionales.

<sup>11</sup> Control Objectives for Information and related Technology (Objetivos de Control para la Información y Tecnologías relacionadas)

El personal encargado de la auditoría emite su opinión al respecto y efectúa recomendaciones para mejorar el nivel en apoyo del cumplimiento de los objetivos.

#### Ventajas

- La auditoría informática sirve para mejorar ciertas características en la empresa como: desempeño, fiabilidad, eficacia, rentabilidad, seguridad y privacidad.

#### Desventajas

- Depende fuertemente de la habilidad y calidad del personal involucrado.

### 1.8 SEGURIDAD EN BASES DE DATOS

La importancia de la información concentrada en las bases de datos, hace de éstas un elemento imprescindible para las empresas, es por esto que resulta de suma importancia establecer e implementar medidas de seguridad para garantizar la integridad, confidencialidad y disponibilidad.

Una base de datos con un bajo nivel de seguridad compromete no solamente a la base de datos misma, sino también al sistema operativo. La protección del sistema operativo y de los servicios de red en un servidor de bases de datos tiene una importancia crítica.

Un manejador de base de datos debe proteger sus datos de:

- Accidentes, por ejemplo errores de captura o de programación.
- Uso malicioso de la base de datos.
- Fallas de hardware o software que corrompan los datos.

Para poder cumplir esto un manejador de bases de datos debe cumplir con los siguientes requisitos como mínimo:

- Autenticación de usuarios.
- Protección de acceso impropio.

- Integridad de la base de datos.
- Administración y protección de datos sensibles.
- Registro de eventos y auditoría.

Por parte del administrador de bases de datos es importante cumplir con algunas buenas prácticas como:

- Usar un sistema de detección de intrusos, especialmente en servidores de bases de datos en línea y de alto riesgo.
- Cambiar las contraseñas de las cuentas creadas por default durante la instalación asignando contraseñas fuertes a las mismas.
- Mantener actualizado el manejador de bases de datos con las versiones más recientes del software y de los parches de seguridad liberados por el fabricante.
- En las aplicaciones, restringir la ejecución de instrucciones de SQL dinámico.
- Impedir que las aplicaciones acepten instrucciones SQL de los usuarios y las ejecuten sobre la base de datos.
- Hacer que los usuarios consulten los datos mediante vistas en lugar de otorgarles acceso a las tablas base.
- Habilitar la auditoría de acceso al sistema operativo y al servidor de base de datos.
- Revisar el registro de auditoría buscando los eventos fallidos de acceso y buscar tendencias con la finalidad de detectar posibles intrusos.
- Monitorear cuidadosamente los registros (logs) de error y de eventos y disparar automáticamente alertas relacionadas con la seguridad y con errores. Proteger los archivos de registro (logs) mediante permisos apropiados de sistema operativo.
- En ambiente de bases de datos distribuidas, eliminar el acceso a los servidores que no se utilicen. Utilizar cuentas de acceso con mínimos privilegios para los servidores relacionados.

- Definir y aplicar una política de respaldo periódico. Almacenar los medios de respaldo en un lugar seguro. Realizar regularmente restauraciones de la base de datos a partir de los respaldos.

## 1.9 PERFORMANCE AND TUNNING

La optimización es una tarea compleja porque requiere un conocimiento de todo el sistema. Se podrían optimizar sólo algunos aspectos teniendo poco conocimiento del sistema o aplicación, pero cuanto más óptimo se quiera el sistema, más se tiene que conocer acerca del mismo.

El factor más importante para hacer un sistema rápido es su diseño básico. Además, es necesario conocer los procesos que realiza el sistema y cuáles son sus puntos débiles. En la mayoría de los casos los problemas están relacionados con el manejo y la administración de la memoria principal y discos duros.

### MEMORIA

La falta de memoria se traduce en un descenso en el desempeño del RDBMS, ya que el sistema se ve imposibilitado para resolver la gran cantidad de peticiones a las que se ve sujeto.

Teniendo una adecuada cantidad de memoria RAM<sup>12</sup> disponible para cachés se reduce el número de veces que el RDBMS tiene que leer datos del disco duro.

A continuación se lista una serie de pasos para maximizar la memoria:

1. Determine la cantidad total de memoria física en su computadora.
2. Reste la memoria requerida por el sistema operativo de la memoria física total.
3. Si la máquina no es sólo para el RDBMS, reste la memoria que es utilizada por otras aplicaciones así como la requerida por el entorno gráfico.
4. Estar seguro de que el RDBMS sólo usa la memoria disponible que queda.

---

<sup>12</sup> Random Access Memory (Memoria de Acceso Aleatorio).

## DISCOS DUROS

El aumento de las peticiones a disco puede generar un cuello de botella causando problemas en el desempeño del sistema. Este tipo de problema se hace más evidente cuando la cantidad de datos ha comenzado a crecer tanto que resulta imposible almacenarla en cache eficientemente.

Considerar los siguientes consejos acerca de cómo colocar los datos:

- Poner bases de datos con requerimientos de rendimiento crítico en dispositivos separados.
- Poner las bases de datos de servidor en dispositivos separados.
- Poner tablas con uso pesado en discos separados.
- Usar segmentos como necesidad para tablas críticas.
- Usar particiones como necesidad para consultas paralelas.
- Separar datos almacenados del log almacenado.
- Colocar dispositivos duplicados en discos separados.
- Usar segmentos para poner tablas e índices en sus propios discos.
- Las tablas particionadas equilibran el número de dispositivos físicos en un segmento.
- Colocar apropiadamente objetos de base de datos en dispositivos físicos mejora el rendimiento.

## 1.10 TÓPICOS AVANZADOS DE BASES DE DATOS

Una parte importante dentro de la administración de bases de datos es cómo es mostrada la información al usuario o a la alta dirección, quienes por lo general no cuentan con conocimientos técnicos relacionados con la administración de bases de datos, por lo cual es necesario emplear técnicas para facilitar el análisis y la comprensión de la información contenida en éstas.

## DATA MINING

El Data Mining (DM) es un mecanismo de explotación que consiste en la búsqueda de información valiosa en grandes volúmenes de datos, para apoyar en la toma de decisiones.

En el Data Mining se captan y procesan los datos con la esperanza de que de ellos surja una hipótesis apropiada. Se desea que los datos describan o indiquen por qué presentan determinada configuración y comportamiento.

La figura 1.5 ilustra las etapas principales del proceso de Data Mining que son:

1. **Determinación de los objetivos:** delimitar los objetivos que el cliente desea bajo la orientación del especialista en DM.
2. **Preprocesamiento de los datos:** se refiere a la selección, limpieza, enriquecimiento, reducción y transformación de las bases de datos. Esta etapa consume generalmente alrededor del 70% del tiempo total de un proyecto de DM.
3. **Determinación del modelo:** se comienza realizando un análisis estadístico de los datos y después se lleva a cabo una visualización gráfica de los mismos para tener una primera aproximación. Según los objetivos planteados y la tarea que debe llevarse a cabo, pueden utilizarse algoritmos desarrollados en diferentes áreas de la Inteligencia Artificial.
4. **Análisis de los resultados:** verifica si los resultados obtenidos son coherentes y los coteja con los obtenidos por el análisis estadístico y de visualización gráfica. El cliente determina si son novedosos y si le aportan un nuevo conocimiento que le permita considerar sus decisiones.

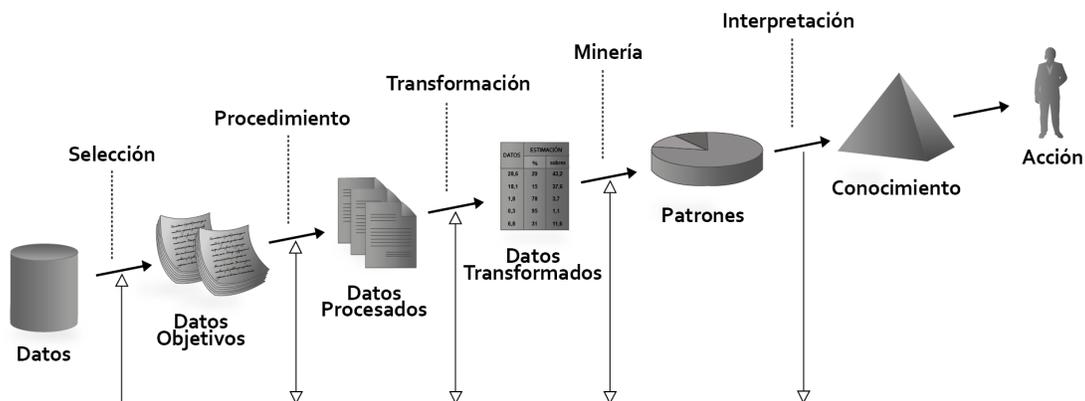


Fig. 1.5 Proceso de Data Mining.

## DATA WAREHOUSING

Data Warehousing (DW) es una combinación de conceptos y tecnología que cambian significativamente la manera en que es entregada la información a la gente de negocios. El objetivo principal es satisfacer los requerimientos de información internos de la empresa para una mejor gestión, con eficiencia y facilidad de acceso.

El DW intenta responder a la compleja necesidad de obtención de información útil sin el sacrificio del rendimiento de las aplicaciones operacionales, debido a lo cual se ha convertido actualmente en una de las tendencias tecnológicas más significativas en la administración de información.

La figura 1.6 ilustra la estructura básica de la arquitectura DW que incluye:

1. **Datos operacionales:** un origen de datos para el componente de almacenamiento físico DW.
2. **Extracción de Datos:** selección sistemática de datos operacionales usados para poblar el componente de almacenamiento físico DW.
3. **Transformación de datos:** Procesos para resumir y realizar otros cambios en los datos operacionales para reunir los objetivos de orientación a temas e integración principalmente.
4. **Carga de Datos:** inserción sistemática de datos en el componente de almacenamiento físico DW.
5. **Datawarehouse:** almacenamiento físico de datos de la arquitectura DW.
6. **Herramientas de Acceso al componente de almacenamiento físico DW:** herramientas que proveen acceso a los datos.

## BASES DE DATOS MULTIDIMENSIONALES

El modelado multidimensional es una técnica para modelar bases de datos de forma simple y entendibles al usuario final. La idea fundamental es que el usuario visualice fácilmente la relación que existe entre los distintos componentes del modelo.

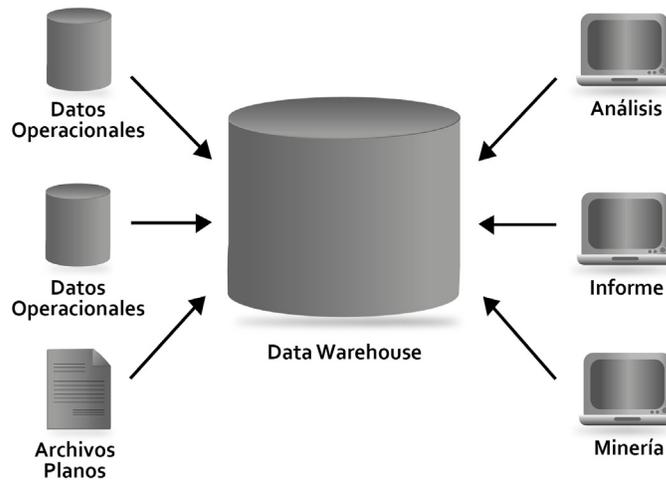


Fig. 1.6 Arquitectura Data Warehouse

En el modelo multidimensional cada eje corresponde a una dimensión particular. Entonces la dimensionalidad de la base estará dada por la cantidad de ejes (o dimensiones) que se le asocien.

Cuando una base de datos puede ser visualizada como un cubo de tres dimensiones (véase figura 1.7), es más fácil para el usuario organizar la información e imaginarse en ella cortando y rebanando el cubo a través de cada una de sus dimensiones para buscar la información deseada.

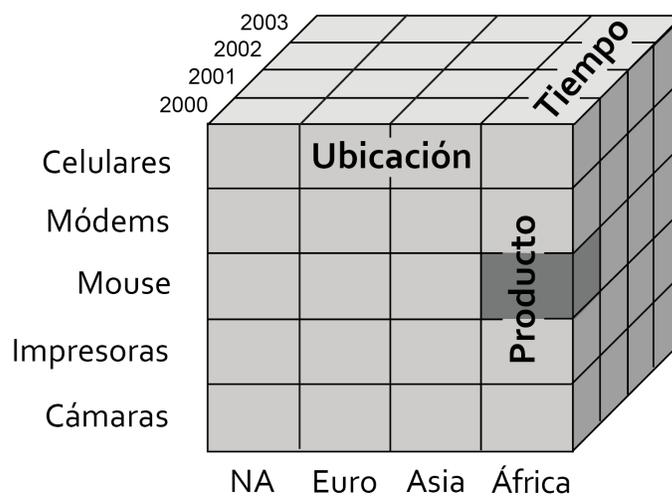


Fig. 1.7 Cubo de información.

PROYECTO PARA LA ADMINISTRACIÓN DE INVENTARIO  
A TRAVÉS DE UNA INTERFAZ WEB  
EN EL HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI

---

Basado en todo lo aprendido en el diplomado y cubriendo una necesidad del Hospital de Especialidades del Centro Médico Nacional Siglo XXI, se desarrolló una aplicación web para el levantamiento y administración del inventario de equipo de cómputo.

El Hospital de Especialidades CMN Siglo XXI no cuenta con un inventario actualizado del equipo de cómputo, lo cual repercute en el control del mismo; no saben con precisión con cuántos equipos se cuenta, ni cuántos ya no funcionan, tampoco saben exactamente la ubicación de éstos.

Debido a un requerimiento de alta dirección, surge la necesidad de contar con un inventario actualizado y verídico, ya que se requiere saber la cantidad de equipos con los que cuenta el hospital, así como su ubicación y antigüedad. La importancia de esta información es crucial para la adquisición de equipos nuevos, así como para tener la certidumbre del número de equipos que han quedado fuera de garantía. También es importante para la asignación de equipos a servicios que no cuentan con los equipos necesarios para su buen desempeño.

Con la aprobación del Jefe de División de Ingeniería Biomédica la Ingeniera Ema Evelia Gutiérrez Flores, se implementó dicho sistema, montando un servidor para el procesamiento de la información y el albergue de la base de datos. El sistema operativo utilizado es Linux Debian 6 Squeeze y el manejador de la base de datos es MySQL.

## 2.1 SOFTWARE NECESARIO PARA EL SERVIDOR

A continuación se describe el software indispensable para montar el servidor.

### DEBIAN GNU/LINUX

Debian GNU/Linux es una de tantas distribuciones de Sistemas Operativos Linux. La cual ha sido desarrollada por una extensa comunidad de programadores quienes pertenecen a una asociación llamada Proyecto Debian. Debian tiene como particularidad ser un sistema operativo compuesto totalmente por software libre.

Hablar de software libre es referirse a libertad, no a la ausencia de precio. Software libre significa tener la libertad para distribuir libremente copias de software libre, el poder recibir u obtener código fuente si se desea, poder modificar software o utilizar pedazos de código de algún software ya existente para crear nuevo software libre.

Actualmente Debian utiliza el núcleo Linux. Linux es una pieza de software creada en un principio por Linus Torvalds y soportada por miles de programadores a lo largo del mundo. Sin embargo, se está trabajando para ofrecer Debian con otros núcleos, en especial con el Hurd, el cual es software libre producido por el proyecto GNU<sup>13</sup>.

Debian viene con más de 29 000 paquetes (software precompilado y empaquetado en un formato amigable para una instalación sencilla) todos ellos de forma gratuita. Estos paquetes abarcan un gran número de aplicaciones, las cuales facilitan realizar un sin número de tareas, desde editar documentos de texto, administrar redes, manejar bases de datos, hasta poder desarrollar nuevo software libre.

El nombre Debian es una contracción de los nombres Debra e Ian Murdock, siendo este último quien fundó oficialmente el proyecto Debian el 16 de agosto de 1993. Hasta ese momento, el concepto de una distribución de Linux era nuevo. Ian pretendió que Debian fuera una distribución realizada de forma abierta y siguiendo el espíritu de Linux y GNU.

Debian es la única distribución que está abierta a las contribuciones de cada desarrollador y usuario que deseen participar con su trabajo. Y es la única distribución relevante de Linux que no es una entidad comercial. Es el único gran proyecto con una constitución, contrato social, y documento de directrices que organizan el proyecto.

---

13 GNU No Unix

Debian es también la única distribución que se microempaqueta y que utiliza una detallada información de las dependencias de cada paquete con respecto a otros para asegurar la consistencia del sistema cuando tiene lugar una actualización.

## MySQL

MySQL es el RDBMS de código fuente abierto más popular y utilizado en el mundo. Al ser de código abierto, permite a todos tener acceso al código fuente, con lo cual cualquiera puede contribuir con ideas, mejoras u observaciones. Gracias a esto MySQL pasó de ser una pequeña base de datos a una completa herramienta para la administración de éstas. Su rápido desarrollo se debe en gran medida a la contribución de mucha gente al proyecto, así como la dedicación del equipo de MySQL.

MySQL es un sistema de administración de bases de datos relacional (RDBMS). Se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y de distribuirlos para cubrir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales hasta grandes empresas y organismos administrativos. Originalmente fue creado y desarrollado en Suecia por: David Axmark, Allan Larsson y Michael "Monty" Widenius, este último finlandés.

El origen del nombre es algo incierto. Algunos lo adjudican a que un gran número de librerías utilizadas en el desarrollo de MySQL tenían el prefijo "my". Sin embargo, la hija del co-fundador Michael Widenius también se llama My. Lo único claro es el uso de SQL (Structured Query Language).

MySQL destaca de la mayoría de RDBMS por su velocidad ya que emplea un motor llamado MyISAM, el cual hace omisión de muchas características típicas en los RDBMS, como es el caso de la integridad referencial, la cual si el administrador de base de datos lo considera necesario, está disponible si se usa el motor InnoDB.

El desarrollo, distribución y soporte es llevado a cabo por MySQLAB. La cual es una compañía comercial fundada por los desarrolladores de MySQL, y que ahora pertenece a Oracle. Es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL.

## SERVIDOR HTTP APACHE

El servidor HTTP<sup>14</sup> Apache es producto de una colaboración de un gran número de desarrolladores alrededor del mundo, los cuales tienen como objetivo crear un robusto, competitivo y auténtico servidor HTTP (Web) con libre disponibilidad de implementación de código fuente. Este proyecto es conjuntamente manejado por un grupo de voluntarios esparcidos por todo el mundo, quienes haciendo uso del Internet planean y desarrollan el servidor, así como la documentación necesaria. Cientos de usuarios han contribuido con ideas, código y documentación al proyecto, el cual es parte de la ASF (Apache Software Foundation).

En Abril de 1995 el Grupo Apache lanza la primera versión del servidor (0.6.2), el cual utilizaba el httpd 1.3 de la NCSA<sup>15</sup> como base, al cual le añadieron todas las correcciones de errores publicadas hasta ese momento así como todas las mejoras relevantes que pudieron encontrar.

El lanzamiento del servidor Apache fue un gran éxito, pero el grupo sabía que tenían que realizar una revisión general del código así como rediseñarlo. Después de una extensa serie de pruebas, una nueva documentación y la adición de nuevas características, es lanzada la versión 1.0 de Apache el 1 de Diciembre de 1995. En menos de un año el servidor Apache pasó a ser el más utilizado en internet y aún retiene esa posición.

## PHP

El lenguaje PHP es un lenguaje de programación como tal, con variables, sentencias condicionales, ciclos, funciones, etc. No es un lenguaje de marcado como HTML o XML. Se asemeja más a JavaScript o a C.

A diferencia de JavaScript que se ejecuta en el navegador, PHP se ejecuta en el servidor, eso permite acceder a los recursos que tenga el servidor como por ejemplo una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador, como se observa en la figura 2.1. El resultado por lo general es una página HTML.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente de él, sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

---

<sup>14</sup> Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto).

<sup>15</sup> National Center for Supercomputing Applications (Centro Nacional de Aplicaciones de Supercomputación).

PHP fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI<sup>16</sup> escrito en C el cual utilizaba en sus páginas web para mantener un control sobre quien consultaba su currículum. La primera versión disponible para el público fue lanzada a principios de 1995 conocida como "Herramientas para Páginas Web Personales" (Personal Home Page Tools). A mediados de 1995 Rasmus rescribió el analizador sintáctico utilizado en PHP Tools, lo combinó con un programa que había escrito el cual procesaba los datos de un formulario llamado FI (Former Interpreter) y añadió soporte para mSQL (mini SQL). Fue así como PHP/FI vio la luz.

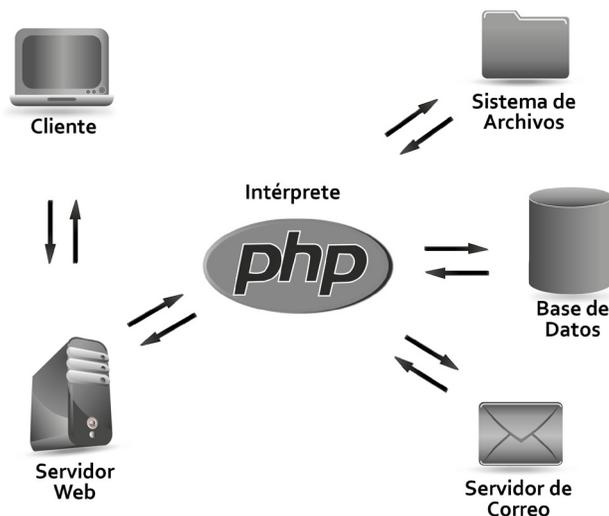


Fig. 2.1 Funcionamiento de PHP.

A mediados de 1997 el número de páginas que utilizaban PHP/FI era alrededor de 50 000. El proyecto dejó de ser obra personal de Rasmus, al cual habían ayudado un grupo de usuarios, y se convirtió en un proyecto de grupo mucho más organizado. El analizador sintáctico se reescribió desde el principio por Zeev Suraski y Andi Gutmans y este nuevo analizador estableció las bases para PHP versión 3. Gran cantidad de código de PHP/FI fue portado a PHP3 y otra gran cantidad fue escrito completamente de nuevo.

Actualmente PHP se encuentra en su versión 5, que utiliza el motor Zend desarrollado con mayor meditación para cubrir las necesidades actuales y solucionar algunos inconvenientes de la anterior versión. Algunas mejoras de esta nueva versión son su rapidez (gracias a que primero se compila y luego se ejecuta, mientras que antes se ejecutaba mientras se interpretaba el código), su mayor independencia del servidor web y un API<sup>17</sup> más elaborado y con más funciones.

16 Common Gateway Interface (Interfaz de Entrada Común).

17 Application Programming Interface (Interfaz de Programación de Aplicaciones).

## HTML

HTML es un lenguaje para describir la estructura de páginas web. Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomo dos herramientas ya existentes: el concepto de Hipertexto el cual permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es un lenguaje de programación, es un lenguaje de marcado. El cual consta de un conjunto de etiquetas de marcado con las que se describe la estructura de una página web.

Para crear páginas con HTML basta tener algún editor de texto plano como el bloc de notas (Windows), vim, nano o gedit (Linux). Los archivos creados deberán ser guardados con la extensión .htm o .html. Estos archivos pueden ser mostrados por los navegadores web como Microsoft Internet Explorer, Mozilla Firefox, Google Chrome, etc. El propósito del navegador de páginas web es leer los archivos HTML y mostrarlos como páginas web. El navegador no muestra las etiquetas HTML sólo las emplea para interpretar el contenido de la página.

Un documento escrito en HTML contendría básicamente lo siguiente:

<code>&lt;HTML&gt;</code>	Indica el inicio del documento.
<code>&lt;HEAD&gt;</code>	Indica el encabezado del documento.
<code>&lt;TITLE&gt;</code>	Indica el título del documento.
<code>&lt;/TITLE&gt;</code>	Indica el fin de la etiqueta del título.
<code>&lt;/HEAD&gt;</code>	Indica el fin de la etiqueta de encabezado.
<code>&lt;BODY&gt;</code>	Indica el cuerpo del documento.
<code>&lt;/BODY&gt;</code>	Indica el fin del cuerpo del documento.
<code>&lt;/HTML&gt;</code>	Indica el fin del documento.

## CSS

CSS<sup>18</sup> es un lenguaje utilizado para dar estilo a documentos HTML y XML separando la estructura de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores web controlar el estilo y el formato de múltiples páginas web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

---

<sup>18</sup> Cascade Style Sheets (Hojas de Estilo en Cascada).

En sus orígenes HTML no fue creado con la intención de poder dar formato a los documentos, se enfocaba totalmente a definir la estructura de estos. Con la aparición de etiquetas como `<font>` y los atributos de color, el desarrollo de sitios web grandes resultó en un proceso muy largo y costoso, ya que si se realizaba alguna modificación visual, ésta tenía que ser aplicada a cada una de las páginas que conformaban el sitio.

El organismo W3C<sup>19</sup>, encargado de crear todos los estándares relacionados con el desarrollo web, propuso la creación de un lenguaje de hojas de estilos específico para el lenguaje HTML.

En 1995 surge el lenguaje CSS, el cual fue resultado de la unión de dos propuestas: la CHSS<sup>20</sup> de Hakon Wium y la SSP<sup>21</sup> de Bert Bos. En ese mismo año el W3C decidió apostar por el desarrollo y estandarización de CSS y lo añadió a su grupo de trabajo de HTML. A finales de 1996, el W3C publicó la primera versión oficial, conocida como "CSS nivel 1".

CSS funciona a base de reglas, es decir, declaraciones sobre el estilo de uno o más elementos. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML.

La regla se compone principalmente de dos partes: un selector y una o varias declaraciones. El selector es normalmente el elemento HTML al cual se aplicará el estilo. Cada declaración consiste de una propiedad y un valor. La propiedad es el atributo de estilo que se desea cambiar, cada propiedad tiene un valor. Toda declaración de CSS termina con punto y coma, y el grupo de declaraciones va encerrado por llaves, como se ilustra en la figura 2.2.



Fig. 2.2 Componentes de una regla CSS.

19 World Wide Web Consortium (Consortio World Wide Web).

20 Cascading HTML Style Sheets (Hojas de Estilo para HTML en Cascada).

21 Stream-based Style Sheet Proposal (Propuesta de Hojas de Estilo en Flujo).

## 2.2 PREPARANDO EL SERVIDOR

A continuación el proceso de instalación del software.

### INSTALACIÓN DE DEBIAN GNU/LINUX

El proceso de instalación consiste en copiar los archivos del sistema operativo al disco duro del equipo.

Antes de instalar cualquier distribución Linux, se debe conocer el hardware dónde se va a realizar la instalación. Esta información puede ser obtenida, ya sea de las especificaciones del fabricante, o si el equipo ya cuenta con un Sistema Operativo instalado utilizando alguna herramienta de dicho sistema operativo.

Una vez que se tiene toda la información sobre el hardware que conforma el equipo y se confirma que es compatible con el sistema operativo Debian, se puede proceder con la instalación. Existen varios medios para llevar a cabo la instalación, puede ser con un CD, DVD o USB (si lo permite el BIOS del equipo).

Los requisitos mínimos recomendados para instalar Debian son los mostrados en la tabla 2.1.

Tipo de Instalación	RAM (Mínimo)	RAM (Recomendado)	Disco Duro
Sin Escritorio	64 MB	256 MB	1 GB
Con Escritorio	128 MB	512 MB	5 GB

*Tabla 2.1 Requisitos mínimos para instalar Debian.*

A continuación se describe el proceso de instalación utilizando un CD.

1. Dar enter a la opción INSTALL y proceder a la selección de idioma, ubicación y configuración del teclado.
2. Para la configuración de red se debe ingresar el nombre del equipo así como el dominio.
3. Un paso importante durante la instalación es la configuración de usuarios y contraseñas. Como primer paso hay que otorgar una contraseña al usuario *root*.

El usuario *root* (también conocido como supersuario) posee acceso completo al sistema. Es usado para instalar paquetes, instalar actualizaciones y realizar la mayoría de las tareas de mantenimiento del sistema.

Una vez realizado esto se debe crear el usuario a utilizar así como también asignarle la contraseña.

4. Seleccionar la zona horaria en la que te encuentras.

5. El particionado del disco duro es un punto crítico durante la instalación.

El particionamiento permite dividir el disco duro en secciones aisladas, donde cada sección se comporta como un propio disco duro. Un esquema de particionamiento recomendado es el siguiente:

- Una partición SWAP (de al menos 256 MB): las particiones swap son utilizadas para apoyar a la memoria RAM. En otras palabras, los datos son escritos a una partición swap cuando no hay suficiente memoria RAM para almacenar los datos que el sistema está procesando.
- Una partición boot (100 MB): la partición montada en `/boot` contiene el kernel del sistema operativo que permite que su sistema inicie, así como los archivos utilizados durante el proceso de arranque.
- Una partición root (500 MB a 5.0 GB): aquí es donde se localiza `/` (el directorio raíz). En esta configuración, todos los archivos (excepto aquellos almacenados en `/boot`) están en la partición raíz. Una partición raíz de 500 MB permite el equivalente de una instalación mínima, mientras que una partición raíz de 5.0 GB le permitirá realizar una instalación completa, seleccionando todos los grupos de paquetes.

6. Una vez definido el particionamiento del disco duro, comenzará el copiado de archivos al mismo.

7. Cuando haya terminado el copiado de archivos, se tendrá que elegir que programas se van a instalar, ya sea a través de otro CD o por medio de una réplica en red. Se desplegará un menú con diferentes opciones para agregar programas.

8. Terminada la descarga e instalación de todos los programas que se hayan elegido, se tendrá que instalar el gestor de arranque.

El gestor de arranque es el primer software que se ejecuta cuando arranca la computadora. Es responsable de la carga y de la transferencia del control al kernel. El kernel, por otro lado, inicializa el resto del sistema operativo.

9. Con la instalación del gestor de arranque termina la instalación, por lo cual se deberá retirar el CD de la unidad CD-ROM y reiniciar el equipo.

## INSTALACIÓN DE MySQL

La instalación de MySQL en Debian puede realizarse de distintas formas:

- Descargar directamente el paquete .deb de instalación de la página oficial de MySQL e instalarlo. Abrir una terminal como *root* y ejecutar:

```
# dpkg -i /ruta_del_paquete_deb/paquete.deb
```

- Haciendo uso de los repositorios. Ejecutar como *root* en una terminal:

```
# apt-get install mysql-common mysql-server mysql-client
```

## CREACIÓN DE LA BASE DE DATOS PARA EL INVENTARIO

Una vez instalado MySQL se puede acceder desde cualquier usuario ejecutando en una terminal:

```
$ mysql -u root -p
```

Se pedirá ingresar la contraseña del usuario *root* que se creó durante la instalación de MySQL. Nota: NO es el usuario *root* que se crea al instalar Debian.

Ya dentro de MySQL se crea la base de datos inventario. Hay que colocarse dentro de la misma.

```
CREATA DATABASE inventario;  
USE inventario;
```

De acuerdo al diagrama Entidad-Relación de la figura 2.3, se crean las tablas necesarias para la base de datos inventario.

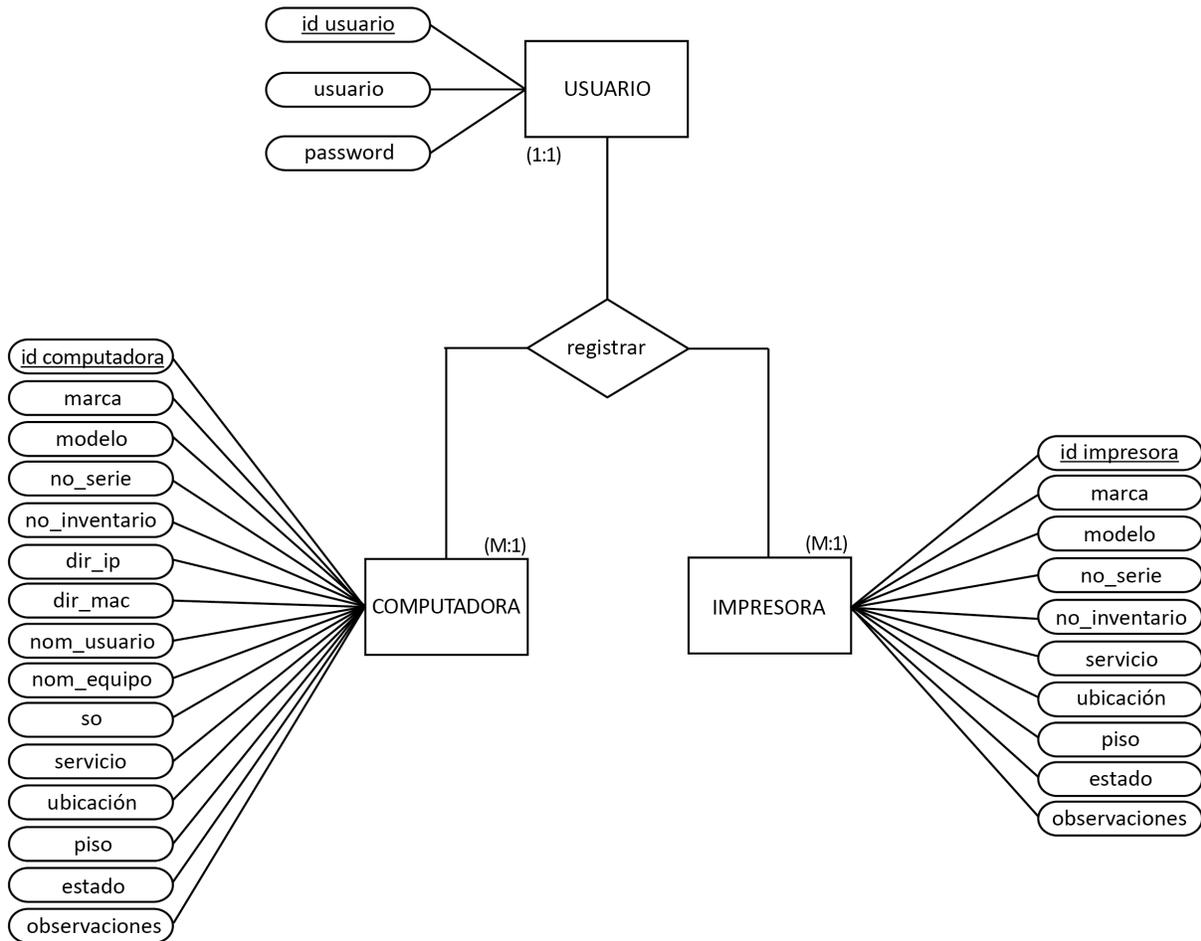


Fig. 2.3 Diagrama Entidad-Relación.

## CÓDIGO PARA LA CREACIÓN DE TABLAS

```
CREATE TABLE computadoras (  
id_computadora int(5) NOT NULL auto_increment,  
marca varchar(15),  
modelo varchar(15),  
no_serie varchar(25),  
no_inventario varchar(20),  
dir_ip varchar(15),  
dir_mac char(17),  
nom_usuario varchar(30),  
nom_equipo varchar(20),  
so varchar(5),  
servicio varchar(35),  
ubicacion varchar(40),  
piso int(1),  
estado char(1),  
observaciones varchar(150),  
PRIMARY KEY (id_computadora)  
);
```

```
CREATE TABLE impresoras (  
id_impresora int(5) NOT NULL auto_increment,  
marca varchar(15),  
modelo varchar(15),  
no_serie varchar(15),  
no_inventario varchar(20),  
servicio varchar(35),  
ubicacion varchar(40),  
piso int(1),  
estado char(1),  
observaciones varchar(150),  
PRIMARY KEY (id_impresora)  
);
```

```
CREATE TABLE usuarios (  
id_usuario int(1) NOT NULL auto_increment,  
usuario varchar(20) NOT NULL,  
password varchar(20) NOT NULL,  
PRIMARY KEY (id_usuario)  
);
```

En la tabla usuarios ingresar los siguientes datos:

```
INSERT INTO usuarios VALUES  
(1,'Biomedica','Biom3dic@');
```

## INSTALACIÓN DEL SERVIDOR HTTP APACHE

Al igual que con MySQL se puede instalar el Servidor HTTP Apache de distintas formas.

- Descargar el archivo con la versión binaria empaquetado de la página oficial de Apache, el cual tiene como extensión `.tar.gz`. Extraer el contenido del archivo ejecutando como *root* en una terminal:

```
# gzip -d httpd-2.2.21.tar.gz
# tar xvf httpd-2.2.21.tar
```

Moverse a la carpeta descomprimida. Configurar, compilar e instalar.

```
# cd httpd-2.2.21
# ./configure --prefix=/usr/local/apache2
# make
# make install
```

- Haciendo uso nuevamente de los repositorios. Abrir una terminal y como usuario *root* ejecutar:

```
# apt-get install apache2
```

Ya instalado el Servidor HTTP Apache copiar la carpeta inventario con los scripts que conforman la aplicación al directorio `/var/www/` que es el que asigna Apache en su instalación por default.

```
# cp -r inventario /var/www/
```

## INSTALACIÓN DE PHP

Para poder trabajar con scripts en PHP es necesario instalar tres cosas: el intérprete, la librería de Apache para soportar PHP y el módulo de PHP para poder establecer comunicación con MySQL.

Afortunadamente todo esto se puede realizar en una sola línea de comandos desde una terminal. Ejecutar como usuario *root* lo siguiente:

```
# apt-get install php5 libapache2-mod-php5 php5-mysql
```

## 2.3 MANUAL DE USUARIO PARA LA ADMINISTRACIÓN DEL INVENTARIO DE EQUIPO DE CÓMPUTO DEL HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI

La correcta elaboración de un inventario permite llevar un registro verídico de los bienes con los que se cuenta, así como poder llevar un control sobre los mismos.

Para la elaboración del inventario de equipo de cómputo se requiere la recopilación de datos físicos como: números de serie, número de inventario, modelo, etc., así como de datos del equipo a nivel lógico como: nombre de usuario, nombre de equipo, dirección IP, etc. Esta recopilación de información podría resultar engorrosa si se realizara con algún método antiguo de captación de datos como el llenado de hojas, para después en el mejor de los casos, vaciar todo a una hoja de cálculo.

Haciendo uso de la infraestructura de red con que cuenta el hospital, se puede realizar la recolección de datos que se requieren sin necesidad de llenar formatos a mano, logrando con esto un proceso más rápido y eficiente, concentrando toda la información en una base de datos a través de una interfaz web la cual funciona sin necesidad de instalar software adicional en los equipos utilizados para inventariar.

Con la siguiente aplicación web se logra agilizar la recolección de la información, así como poder concentrar en una base de datos la misma, pudiendo así facilitar su manipulación y control.

En el presente manual se indicará como operar correctamente la administración del inventario a través de dicha aplicación web.

### ADMINISTRACIÓN DEL INVENTARIO DE EQUIPO DE CÓMPUTO DEL HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI

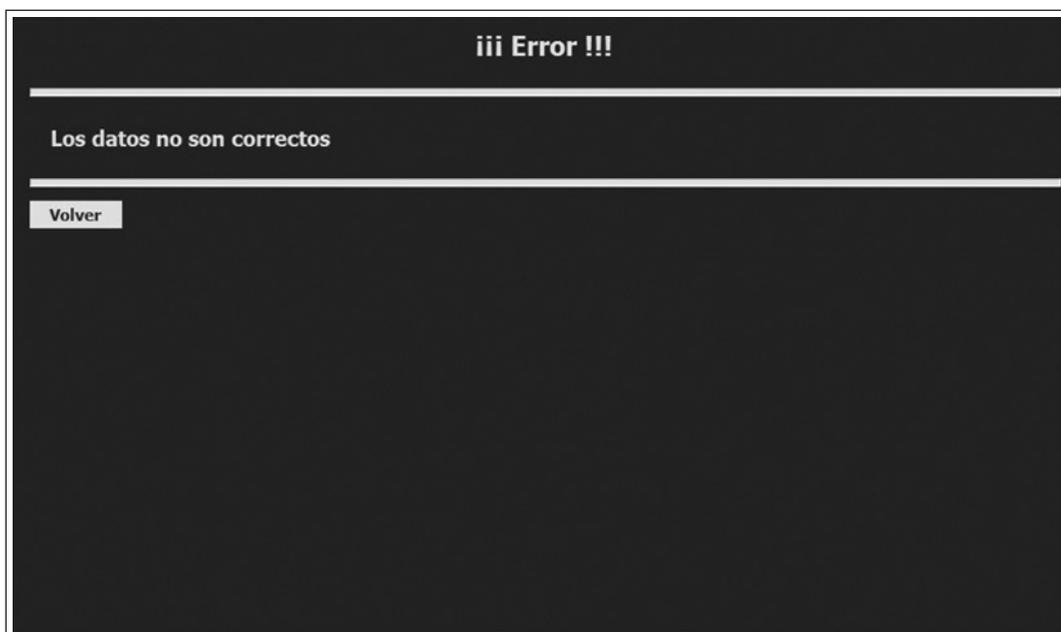
La aplicación para la administración del inventario no requiere de ninguna instalación de software, ya que es una aplicación web a la cual se puede acceder desde cualquier equipo que esté dado de alta en la red institucional.

Se procede a abrir el navegador de internet e ingresar a la dirección <http://11.47.37.80/inventario>, lo cual dirigirá a la pantalla de autenticación. (Fig. 2.3)



*Fig. 2.3 Pantalla de autenticación.*

Donde se tendrá que ingresar los datos siguientes: usuario **Biomedica** y como contraseña **Biom3dic@**. Dar click en el botón de **Login**; en caso de ingresar mal los datos se notificará con la pantalla de error. (Fig. 2.4)



*Fig. 2.4 Pantalla de error*

Una vez autenticado correctamente se desplegará la pantalla principal de la aplicación. (Fig. 2.5)



Fig. 2.5 Pantalla principal.

## REGISTRO

Para poder registrar un equipo de cómputo dar click a lo opción **REGISTRO** del menú que se encuentra del lado izquierdo; al hacerlo se abrirá en la misma ventana un menú (véase figura 2.6) en el cual se tendrá que especificar el tipo de equipo de cómputo que se va a registrar.



Fig. 2.6 Menú de registro.

Al escoger el tipo de equipo que se va a registrar, aparecerá en pantalla el formulario para la recolección de los datos del mismo. (Fig. 2.7).

HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI  
DIVISION DE INGENIERIA BIOMEDICA  
INVENTARIO

MENU  
REGISTRO  
ACTUALIZACION  
BAJA  
REPORTES

Registro de Computadoras

Marca	HP	Modelo	
No. Serie		No. Inventario	
Dir. IP		Dir. MAC	
Nombre Usuario		Nombre Equipo	
Windows	XP		
Servicio		Ubicacion	

HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI  
DIVISION DE INGENIERIA BIOMEDICA  
INVENTARIO

MENU  
REGISTRO  
ACTUALIZACION  
BAJA  
REPORTES

Windows	XP		
Servicio		Ubicacion	
Piso	PB		
Estado	● F ● NF		
Observaciones			

Enviar

Fig. 2.7 Formulario para el registro.

Una vez terminado el llenado del formulario dar click al botón de **Enviar** (véase figura anterior), aparecerá una pantalla que confirmará el registro del equipo y mostrará los datos ingresados. (Fig. 2.8)



Fig. 2.8 Confirmación del registro.

## ACTUALIZACIÓN

Para la actualización de algún equipo de cómputo seleccionar del menú de la izquierda con un click la opción de **ACTUALIZACION**; al igual que con el registro de equipo se debe especificar el tipo de equipo de cómputo que se va a actualizar. (Fig. 2.9)

Al haber seleccionado el tipo de equipo de cómputo que se va a actualizar, aparecerá en pantalla el formulario para realizar la búsqueda del equipo de cómputo que se desea modificar, teniendo como parámetros de búsqueda el número de serie, dirección MAC o servicio; siendo no necesario el llenado de los tres se recomienda realizar la búsqueda valiéndose del número de serie. (Fig. 2.10)



Fig. 2.9 Menú de actualización.

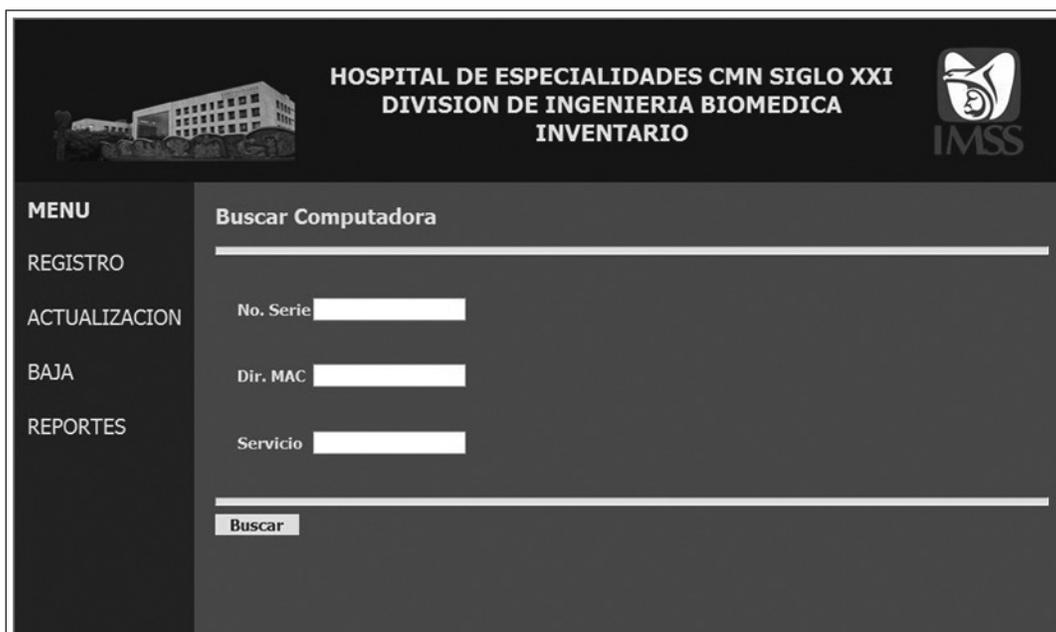


Fig. 2.10 Formulario para realizar búsqueda.

Una vez realizada la búsqueda aparecerá en pantalla una tabla con los datos de los equipos encontrados, así como un formulario para ingresar los nuevos datos del equipo que se desea modificar (véase figura 2.11). Es muy importante poner atención al campo **id** ya que es el que indica que equipo se va a modificar.



**HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI**  
**DIVISION DE INGENIERIA BIOMEDICA**  
**INVENTARIO**



**MENU**

REGISTRO

ACTUALIZACION

BAJA

REPORTES

**Resultado de la Búsqueda**

id	marca	modelo	no_serie	no_inventario	dir_ip	dir_mac	nom_usuario	nom_equipo	so	servicio	ubicacion
509	HP	DC7100	MXJ0480VXJ	2008987987	11.47.37.80	A1-B2-C3-D5-E6	Biomedica	mt002464wsbloss1	XP	INGENIERIA BIOMEDICA	BIOMEDICA

**Ingresar el id del equipo y sus nuevos datos**

id

Marca  Modelo



**HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI**  
**DIVISION DE INGENIERIA BIOMEDICA**  
**INVENTARIO**



**MENU**

REGISTRO

ACTUALIZACION

BAJA

REPORTES

Windows

Servicio  Ubicacion

Piso

Estado  F  NF

Observaciones

**Modificar**

Fig. 2.11 Resultado de la búsqueda y formulario para realizar actualización.

En caso de que no se haya encontrado equipo alguno al realizar la búsqueda también será notificado. (Fig. 2.12)



Fig. 2.12 Resultado de la búsqueda al no encontrar equipos.

Una vez capturados los nuevos datos del equipo dar click al botón **Modificar** (véase figura 2.11), saldrá una pantalla notificando la actualización. (Fig. 2.13)



Fig. 2.13 Confirmación de actualización.

## BAJA

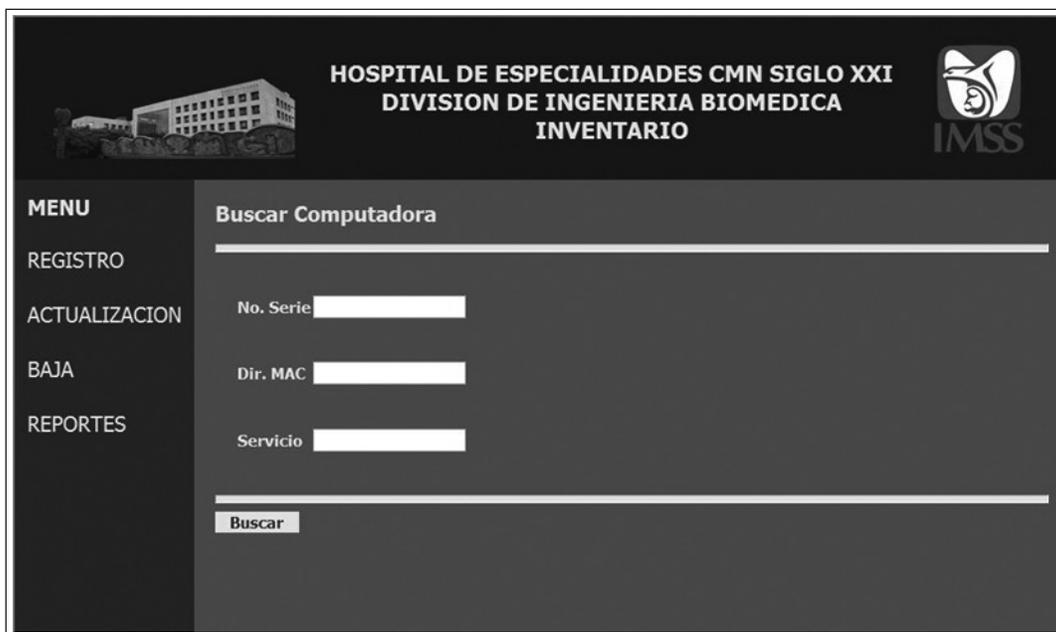
Al seleccionar **BAJA** del menú se tendrá que especificar que tipo de equipo se va a dar de baja. (Fig. 2.14)



The screenshot shows the 'BAJA' menu in the 'HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI DIVISION DE INGENIERIA BIOMEDICA INVENTARIO' system. The left sidebar contains the menu items: MENU, REGISTRO, ACTUALIZACION, BAJA (highlighted), and REPORTEES. The main content area is titled 'Borrar' and lists two equipment types: COMPUTADORA and IMPRESORA.

Fig. 2.14 Menú de baja.

Al igual que con el módulo de actualizar se debe realizar la búsqueda del equipo que se desea borrar. (Fig. 2.15)



The screenshot shows the search form in the 'HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI DIVISION DE INGENIERIA BIOMEDICA INVENTARIO' system. The left sidebar contains the menu items: MENU, REGISTRO, ACTUALIZACION, BAJA, and REPORTEES. The main content area is titled 'Buscar Computadora' and contains three input fields: 'No. Serie', 'Dir. MAC', and 'Servicio'. A 'Buscar' button is located at the bottom of the form.

Fig. 2.15 Formulario para realizar búsqueda.

Una vez realizada la búsqueda aparecerá una tabla con los datos de los equipos encontrados (veáse figura 2.16). La eliminación de un equipo se realizará a través de su id.

The screenshot shows the 'HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI DIVISION DE INGENIERIA BIOMEDICA INVENTARIO' interface. On the left is a 'MENU' with options: REGISTRO, ACTUALIZACION, BAJA, and REPORTE. The main area is titled 'Resultado de la Búsqueda' and contains a table with the following data:

id	marca	modelo	no_serie	no_inventario	dir_ip	dir_mac	nom_usuario	nom_equipo	so	servicio	ubicacion
69	HP	DC7100	MXJ44400LQ		172.23.147.15	00-11-85-62-19-24	vista95.umae02464	mto02464wsv95XP		CARDIOLOGIA	CONSUL

Below the table is a form titled 'Ingresa el id del equipo a borrar' with an input field for 'id' and a 'Borrar' button.

Fig. 2.16 Resultado de la búsqueda y formulario para realizar baja.

Si no se encontraron equipos con los datos proporcionados para su búsqueda será notificado. (Fig. 2.17)

The screenshot shows the same interface as Fig. 2.16, but the search results area displays the message: 'No se encontraron equipos con esos datos.' The menu and header are identical.

Fig. 2.17 Resultado de la búsqueda al no encontrar equipos.

Una vez ingresado el **id** del equipo que se desea eliminar dar click al botón **Borrar** (véase figura 2.16) Se notificará con la pantalla de equipo eliminado. (Fig. 2.18)

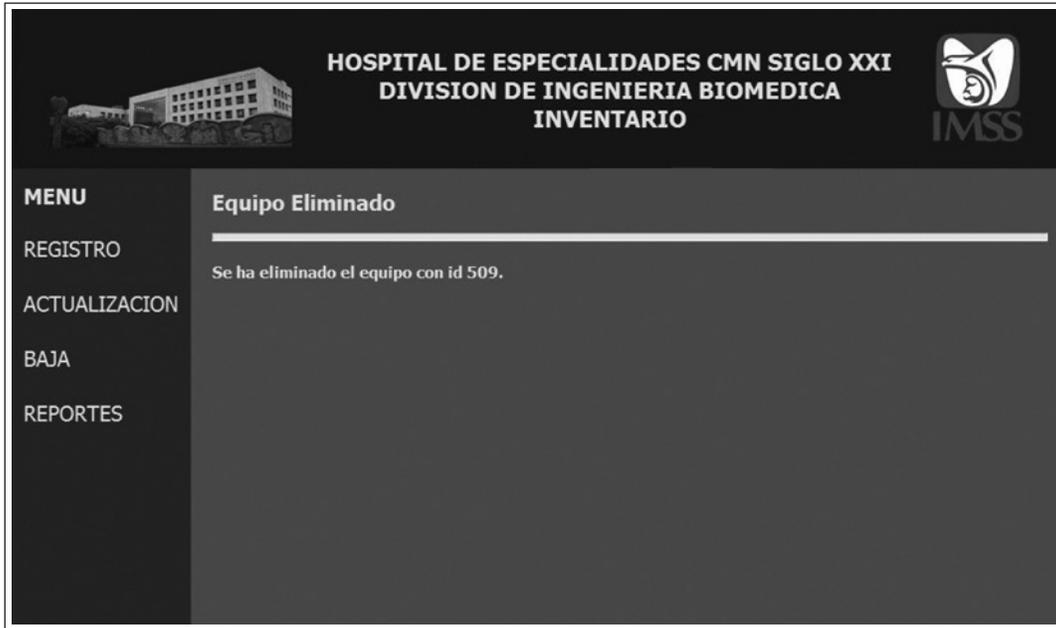


Fig. 2.18 Confirmación de equipo eliminado.



Fig. 2.19 Menú de reportes.

## REPORTES

Para acceder al módulo **REPORTES** al igual que con los otros módulos será a través del menú del lado izquierdo, teniendo también que especificar el tipo de equipo para generar el reporte. (Fig. 2.19)

Una vez especificado el equipo, se deberá seleccionar que tipo de reporte se desea generar, pudiendo seleccionar entre **PISO**, **SERVICIO** o **GENERAL**. (Fig. 2.20)



Fig. 2.20 Tipo de reporte.

Para generar un reporte por piso se deberá dar click a la opción **PISO** y seleccionar el piso que se desee. Sólo se puede obtener un reporte por piso a la vez. (Fig. 2.21)

Para generar un reporte por servicio debe dar click a la opción **SERVICIO** e ingresar el nombre del servicio. Sólo se puede obtener un reporte por servicio a la vez. (Fig. 2.22)

HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI  
DIVISION DE INGENIERIA BIOMEDICA  
INVENTARIO

MENU

REGISTRO

ACTUALIZACION

BAJA

REPORTES

Selecciona el Piso

Piso PB

Generar

Fig. 2.21 Reporte por piso.

HOSPITAL DE ESPECIALIDADES CMN SIGLO XXI  
DIVISION DE INGENIERIA BIOMEDICA  
INVENTARIO

MENU

REGISTRO

ACTUALIZACION

BAJA

REPORTES

Ingresa el Servicio

Servicio

Generar

Fig. 2.22 Reporte por servicio.

Los reportes serán generados al dar click al botón **Generar**, esto abrirá una ventana preguntando si se desea abrir o guardar el archivo (véase figura 2.23). El archivo generado tiene como extensión **.xls**, el cual deberá ser abierto con el programa Microsoft Excel.

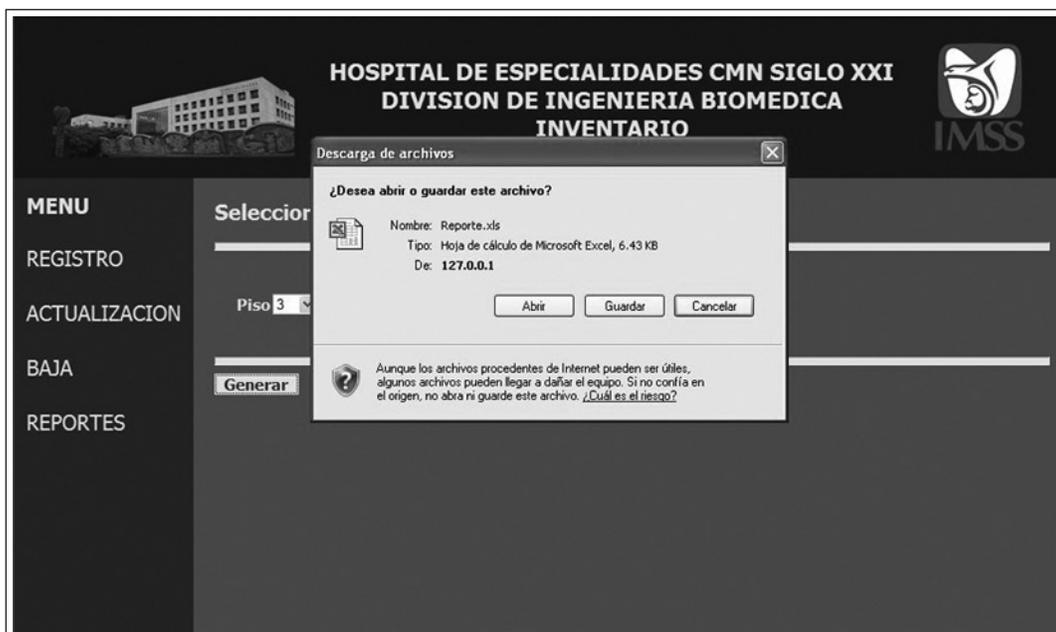


Fig. 2.23 Descarga de archivo.

La información actualmente juega un papel muy importante en cualquier organización u empresa. Por lo cual no se debe dejar de prestarle atención, ya que es posible identificar problemas o irregularidades al analizarla.

Las bases de datos no son solamente almacenes de datos. Concentran en sus registros información valiosa para auxiliar a las organizaciones en la toma de decisiones, ya que a través del análisis de la información que éstas contienen es posible comprender el comportamiento que se está presentando en el sistema y tomar medidas para mejorar o corregir el mismo.

El diplomado en administración en bases de datos me proporcionó de conocimientos sobre la gestión, optimización, explotación, análisis y aseguramiento de la información almacenada en bases de datos. Gracias a esto me es posible valorar la información como un activo de suma importancia y no sólo como un conjunto de datos.

Para el desarrollo de la aplicación no sólo me valí de lo aprendido en el transcurso de la carrera y el diplomado, hice uso del ingenio y análisis para definir cuál sería la forma más viable para su realización e implementación. Desarrollar una aplicación requirió un estudio detallado de la problemática que se presenta, así como comprender que es lo que se desea resolver.

Con la implementación de la aplicación para la administración del inventario en el Hospital de Especialidades del Centro Médico Nacional Siglo XXI se logró resolver la incertidumbre sobre los equipos de cómputo que se encuentran en el hospital, así como tener control sobre los mismos.

Apuntes del Diplomado “*Administración de Bases de Datos*”. Dirección General de Cómputo y de Tecnologías de Información y Comunicación. Del 18 de junio de 2010 al 15 de enero de 2011.

DATE, C. J. *Introducción a los sistemas de bases de datos*. Editorial PEARSON EDUCACIÓN, México, 2001.

LÓPEZ, Quijano José. *Domine PHP y MySQL*. Editorial Alfaomega: Ra-Ma, 2010.

MURPHY, PERSSON. *HTML y CSS*. Editorial Anaya Multimedia, Madrid, 2010.

Apache HTTP Server Project, 2011

Disponible en: [http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html)

Última consulta: diciembre 2011.

Guía de instalación de Debian GNU/Linux, 2010.

Disponible en: <http://www.debian.org/releases/stable/i386/>

Última consulta: diciembre 2011.

MEDINA, López Francisco. *Fundamentos del Sistema Operativo GNU/Linux Versión 2.2*.

México 2006. pp. 151

Disponible en:

[http://www.cepeu.edu.py/LIBROS\\_ELECTRONICOS\\_3/fundamentos\\_so.pdf](http://www.cepeu.edu.py/LIBROS_ELECTRONICOS_3/fundamentos_so.pdf)

Última consulta: noviembre 2011.

MySQL 5.0 Reference Manual, 2011.

Disponible en: <http://dev.mysql.com/doc/refman/5.0/en/index.html>

Última consulta: noviembre 2011

PHP Manual, 2012

Disponible en: <http://mx2.php.net/manual/en/index.php>

Última consulta: noviembre 2011.

- ASF:** Fundación de Software Apache (Apache Software Foundation).
- BLOB:** (Binary Large Object) Elemento utilizado en las bases de datos para almacenar datos de gran tamaño, generalmente son datos de tipo multimedia.
- JDBC:** (Java Database Connectivity) API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java.
- MBD:** Manejador de Bases de Datos.
- MVCC:** (Multiversion Concurrency Control) Método para el control de concurrencia.
- ODBC:** (Open Database Connectivity) API para acceder a datos en sistemas manejadores de bases de datos utilizando SQL.
- OLAP:** (OnLine Analytical Processing) Bases de datos orientadas al procesamiento analítico.
- OLTP:** (OnLine Transaction Processing) Bases de datos orientadas al procesamiento de transacciones.
- SSL:** (Secure Sockets Layer) protocolo criptográfico que proporciona comunicaciones seguras por una red.
- XML:** Lenguaje de marcas extensible (Extensible Markup Language).
- Xpath:** Lenguaje que permite construir expresiones que recorren y procesan un documento XML.
- XSL:** Lenguaje Extensible de Hojas de Estilo (Extensible Stylesheet Language).

<b>Caché:</b>	Segmento de memoria en la que se almacenan una serie de datos para su rápido acceso.
<b>Framework:</b>	Soluciones completas que contemplan herramientas de apoyo a la construcción(ambiente de trabajo o desarrollo) y motores de ejecución (ambiente de ejecución) de aplicaciones.
<b>Hardware:</b>	Todos los componentes físicos y tangibles que conforman un equipo de cómputo.
<b>Kernel:</b>	Núcleo del sistema operativo. Permite al usuario interactuar con el equipo de cómputo.
<b>Partición:</b>	Nombre que recibe cada sección aislada de un disco duro.
<b>Repositorio:</b>	Sitio donde se almacenan programas, puede estar en internet o en un medio extraíble.
<b>Script:</b>	Archivo que contiene código.
<b>Servlet:</b>	Programa que se ejecuta en un servidor.
<b>Software:</b>	Todo el conjunto intangible de datos y programas de la computadora.
<b>Triggers:</b>	Procedimiento que se ejecuta cuando se cumple una condición establecida al realizar una operación.
<b>Unicode:</b>	Estándar de codificación de caracteres diseñado para facilitar el manejo de la información sin importar la plataforma, programa, o idioma.
<b>Web:</b>	Sistema de distribución de información basado en hipertexto o hipermedios enlazados y accesibles a través de Internet.