



**Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación**

**Tesis:**

**Sistema Web Administrativo para la Preparatoria  
Benito Juárez García del Estado de Puebla**

**Presenta:**

**Rodolfo Hugo Jiménez Chávez**

**Que para obtener el título de  
Licenciado en Ciencias de la Computación**

**Asesora:**

**M.E. Carmen Cerón Garnica**

**Puebla de Zaragoza, Diciembre 2009**

# INDICE

<b>Introducción</b> .....	<b>1</b>
Objetivo General .....	2
Objetivos Específicos .....	2
Alcances y Limitaciones de la Tesis .....	3
<b>Capítulo 1 – Conceptos Generales en Aplicaciones Web</b> .....	<b>4</b>
1.1 Ingeniería de Software .....	4
1.2 Ciclo de Vida del Software .....	4
1.3 Sistema de Información .....	7
1.4 ¿Qué es una Base de Datos? .....	9
1.5 Normalización .....	11
1.6 Modelo Entidad-Relación .....	12
1.7 Base de Datos Relacionales .....	13
1.8 Esquema Cliente Servidor .....	14
1.9 ¿Qué es la WWW? .....	16
1.10 Tecnologías de Acceso a Bases de Datos .....	17
1.10.1 Páginas Dinámicas .....	17
1	
1.10.2 Active Server Pages .....	17
2	
1.10.3 Paginas Java Server .....	19
3	
1.10.4 Servlets .....	19
4	
1.10.5 SQL .....	20
5	
<b>Capítulo 2 – Análisis</b> .....	<b>21</b>
2.1 Planteamiento del Problema .....	21
2.2 Casos de Uso .....	22
2.2.1 Caso de Uso 1 .....	23
2.2.2 Caso de Uso 2 .....	24
2.2.3 Caso de Uso 3 .....	25
2.2.4 Caso de Uso 4 .....	26
2.2.5 Caso de Uso 5 .....	27
2.2.6 Caso de Uso 6 .....	28
2.3 Diagrama de Casos de Uso del Sistema .....	29
2.4 Diagramas de Secuencia .....	30
2.4.1 Diagrama de Secuencia de Entrada al Sistema .....	30
2.4.2 Diagrama de Secuencia de Alta de Equipos .....	31
2.4.3 Diagrama de Secuencia de Baja de Equipos. ....	32
2.4.4 Diagrama de Secuencia de Modificación de Equipos. ....	33
2.4.5 Diagrama de Secuencia de Ata de Docentes.....	34
2.4.6 Diagrama de Secuencia Consulta Índice General .....	35
2.4.7 Diagrama de Secuencia Consulta Nombre de Equipo.....	36
2.5 Clase de Presentación y Bases de Datos .....	37
<b>Capítulo 3 – Diseño</b> .....	<b>39</b>
3.1 Arquitectura del Sistema .....	39
3.2 Diagramas de Colaboraciones .....	40
3.2.1 Diagrama de Colaboraciones Alta de Equipos .....	40
3.2.2 Diagrama de Colaboraciones Baja de Equipo .....	40
3.2.3 Diagrama de Colaboraciones Modificación de Equipos .....	41
3.2.4 Diagrama de Colaboraciones Alta de Docentes.....	41
3.2.5 Diagrama de Colaboraciones Consulta por Índice General .....	42
3.2.6 Diagrama de Colaboraciones Consulta Nombre de Equipo .....	42
3.3 Diagramas de Actividades .....	43
3.3.1 Diagrama de Actividades Alta de Equipos.....	43
3.3.2 Diagrama de Actividades Baja de Equipos .....	43
3.3.3 Diagrama de Actividades Modificaciones de Equipos .....	44
3.3.4 Diagrama de Actividades Alta de Docentes.....	44
3.3.5 Diagrama de Actividades Consulta Índice General .....	45
3.3.6 Diagrama de Actividades Consulta Nombre de Equipo.....	45

3.4	Diagrama del Modelo E-R del Sistema .....	46
3.5	Diccionarios de Datos .....	47
3.6	Normalización .....	48
3.7	Relaciones de Tablas .....	49
<b>Capítulo 4 – Implementación .....</b>		<b>50</b>
4.1	Ambiente de Desarrollo .....	50
4.2	Creación de la Base de Datos .....	51
4.3	Pantalla de Entrada .....	52
4.4	Pantalla de Acceso al Sistema .....	53
4.5	Pantalla de Alta de Docentes.....	53
4.6	Pantalla Consultas de Docentes.....	54
4.7	Pantalla de Alta de Equipos .....	55
4.8	Pantalla Acceso Alumnos.....	55
4.9	Pantalla Sitios de Interés para Alumnos.....	56
4.10	Pantalla Base de Datos Bj.....	57
<b>Conclusiones .....</b>		<b>58</b>
<b>Perspectivas .....</b>		<b>59</b>
<b>Anexo A .....</b>		<b>60</b>
<b>Bibliografía y Referencias .....</b>		<b>62</b>

## LISTA DE FIGURAS

Figura 1.1	Proceso determinación de requerimientos sistema.....	5
Figura 1.2	Diseño Conceptual y Técnico.....	6
Figura 1.3	El Ciclo de la Información .....	8
Figura 1.4	Diagrama E-R .....	13
Figura 2.1	Diagrama de Casos de Uso del Sistema .....	29
Figura 2.2	Diagrama de Secuencias de Entrada al Sistema .....	30
Figura 2.3	Diagrama de Secuencias de Alta de Equipos .....	31
Figura 2.4	Diagrama de Secuencias de Baja de Equipos... ..	32
Figura 2.5	Diagrama de Secuencias de Modificación de Equipos .....	33
Figura 2.6	Diagrama de Secuencias de Alta de Docentes .....	34
Figura 2.7	Diagrama de Secuencias de Consulta Índice General .....	35
Figura 2.8	Diagrama de Secuencias de Consulta Nombre de Equipo .....	36
Figura 2.9	Clases de Presentación y Bases de Datos .....	37
Figura 2.10	Constitución del Sistema .....	38
Figura 3.1	Esquema General del Sistema .....	39
Figura 3.2	Diagrama de Colaboraciones para Alta de Equipos .....	40
Figura 3.3	Diagrama de Colaboraciones para Baja de Equipos .....	40
Figura 3.4	Diagrama de Colaboraciones para Modificación de Equipos .....	41
Figura 3.5	Diagrama de Colaboraciones para Alta de Docentes .....	41
Figura 3.6	Diagrama de Colaboraciones para Consulta por Índice General .....	42
Figura 3.7	Diagrama de Colaboraciones para Consulta por Nombre del Equipo.....	42
Figura 3.8	Diagrama de Actividades para Alta de Equipos .....	43
Figura 3.9	Diagrama de Actividades para Baja de Equipos .....	43
Figura 3.10	Diagrama de Actividades para Modificación de Equipos .....	44
Figura 3.11	Diagrama de Actividades para Alta de Docentes .....	44
Figura 3.12	Diagrama de Actividades para Consulta por Índice General.....	45
Figura 3.13	Diagrama de Actividades para Consulta por Nombre de Equipo.....	45
Figura 3.14	Diagrama E-R del Sistema .....	46
Figura 3.15	Relaciones de Tablas .....	49
Figura 4.1	Esquema de Ambiente de Desarrollo .....	51
Figura 4.2	Pantalla de Entrada .....	52
Figura 4.3	Pantalla Acceso al Sistema .....	53
Figura 4.4	Pantalla Primera Alta de Docentes.....	53
Figura 4.5	Pantalla Segunda Alta de Docentes .....	54
Figura 4.6	Pantalla Consulta de Docentes.....	54
Figura 4.7	Pantalla Alta de Equipos.....	55
Figura 4.8	Pantalla Acceso Alumnos.....	55
Figura 4.9	Pantalla Sitios de Interés para Alumnos .....	56
Figura 4.10	Pantalla Base de Datos Bj.....	57

# **AGRADECIMIENTOS**

## **A DIOS**

Por darme salud, sabiduría, habilidades para llegar hasta este momento, también por poner en mi camino a estas personas para lograrlo.

## **A MIS PADRES**

Por creer en mí, por su apoyo, desvelos, consejos consiguiendo este logro que también es de ustedes.

## **A MIS HERMANOS**

Por darme ejemplo y apoyo para conseguir lo anhelado en base a sacrificios y trabajo.

## **A MI NOVIA ARACELI**

Por su gran amor, por confiar siempre en mi y apoyarme en momentos difíciles de ausencia, por motivarme a mejorar cada día.

## **A MI ASESORA**

Por su dedicación apoyo incondicional y guía para culminar este trabajo de tesis, sin usted no lo habría logrado.

## **INTRODUCCION**

En estos tiempos la mayoría de los países están poniendo mayor interés y énfasis en la educación en todos los niveles por medio de la cual los países desarrollados se mantienen en ese nivel y los países en desarrollo buscan progresar y llegar a ese nivel. En México se están desarrollando infraestructuras para poder impartir la educación de una manera más ágil, versátil y al alcance de todos, a través de nuevas técnicas para el manejo, control y distribución de la información.

El sistema de educación superior en Puebla se ha ido fortaleciendo, creciendo y actualizando en gran número. Es por esto que existe la necesidad de una base de datos para el control de administrativo para la preparatoria Benito Juárez García del estado de Puebla, esto surge a partir de observar que se cuenta con mucha información pero la cual se encuentra en papel o recurso manual el cual a veces no se puede consultar o acceder a ella de manera rápida y eficiente y conlleva a que se pierda el interés y la búsqueda del desarrollo.

Los objetivos de este trabajo se enuncian a continuación.

### **OBJETIVO GENERAL**

Desarrollar una aplicación Web que incluya el control de una base de datos en el cual se pueda facilitar el manejo y control de información como lo es recursos físicos, cursos, academias, personal

### **OBJETIVOS ESPECIFICOS**

1. Contar con una aplicación Web en donde tener control de recursos para el cliente administrativo.
2. Consultar los cursos, academias y ubicación de los mismos.
3. Mandar por correo, temas o tareas para su evaluación.
4. Vincular bases de datos con páginas Web.

Este documento llevara como orden de capitulación el siguiente:

- Capitulo 1 se documentan los conceptos de los términos que se involucran en la realización de este proyecto denominándolo así Marco Teórico
- Capitulo 2 en este capítulo se documenta el objetivo general y los específicos, los casos de uso y diagramas de secuencia de los que dependerá el sistema auxiliándose de la metodología UML para el desarrollo del análisis del sistema.
- Capitulo 3 se realiza la arquitectura y los diagramas de colaboraciones y actividades del sistema dándole continuidad a la metodología UML, se expresa la forma de Entidad-Relación así como su diagrama y los diccionarios de datos, y la revisión de la normalización de las tablas resultantes.
- Capitulo 4 se muestra el ambiente de desarrollo y las herramientas que se utilizan así como la implementación del sistema así como las pruebas mostrando las pantallas del sistema.

Finalmente se presentan las conclusiones y las perspectivas.

## **ALCANCES Y LIMITACIONES DE LA TESIS**

Uno de los alcances que se pretende desarrollar en este documento de tesis es el desarrollar un sistema administrativo desarrollado en un ambiente cliente servidor utilizando como base el Web y muchas de sus herramientas de programación enfocadas a las bases de datos y al desarrollo de programación en HTML, se pretende que el sistema solo ofrezca los cursos y recursos a los clientes y estos hagan la consulta necesaria al sistema, los cursos, academias, y recursos a ofrecer estarán almacenados en una base de datos y el cliente podrá ver los atributos de los cursos y equipos para la elección de este.

Así mismo se pretende que el sistema este dirigido por un administrador el cual tendrá a su cargo las altas, bajas y modificaciones de los cursos, equipos, academias, salones de esta manera el administrador canalizará las consultas de los clientes.

Las principales limitaciones que este proyecto presentará serán que el sistema no contará con una base de datos propia acerca de los alumnos y servicios sino que será a través de un enlace con el departamento administrativo de la universidad.

Pues el propósito de este trabajo es aprender y demostrar el funcionamiento de herramientas de programación para HTML, la implementación de una base de datos a una página Web mediante el uso de PHP y MySQL, así como su implementación y pruebas del sistema.

# **1. CONCEPTOS GENERALES EN APLICACIONES WEB**

## **Marco Teórico**

Las secciones que comprenden el capítulo Marco Teórico tratan los conceptos relacionados con Ingeniería de Software, Base de datos, así como la plataforma de desarrollo y del Estado del Arte.

### **1.1 INGENIERIA DE SOFTWARE**

Es una disciplina de diseño y desarrollo de software de alta calidad, tiene un conjunto de elementos clave para mejorar la calidad de sus productos de software. Su finalidad es usar enfoques eficientes y productivos para generar soluciones efectivas a los problemas. Estos elementos son:

- Métodos
- Herramientas
- Procedimientos
- Paradigmas

Un método es un procedimiento formal para producir algún resultado.

Una herramienta es un instrumento o sistema automatizado para realizar algo de la mejor manera posible.

Un procedimiento es una combinación de herramientas y métodos que, juntas dan como resultado un producto particular.

Un paradigma representa un enfoque particular o filosofía de la construcción del software.

El término “Ingeniería de Software” implica efectuar un análisis a un determinado problema, es decir, desglosar el problema en piezas que se puedan comprender. De esta forma se puede describir un problema grande como una serie de problemas pequeños y sus interrelaciones para poder construir una solución a partir de componentes que se ocupen de diversos aspectos del problema basado en el análisis previo.

### **1.2 CICLO DE VIDA DEL SOFTWARE**

Es el proceso de desarrollo del software, es decir, describe la vida del producto del software desde su concepción hasta su implementación, entrega, utilización y mantenimiento.

El desarrollo del software por lo general involucra las siguientes etapas:

- Análisis y definiciones de los requerimientos.
- Diseño del sistema.
- Diseño de programas.
- Codificación.
- Prueba unitaria.
- Prueba de integración.
- Prueba del sistema.
- Mantenimiento.

### Análisis y definición de los requerimientos

Es la comprensión de lo que los clientes y los usuarios esperan que le haga el sistema, es decir, la comprensión del propósito y la función del sistema. Un requerimiento es una característica del sistema o una descripción de algo que el sistema es capaz de hacer con el objeto de satisfacer el propósito del sistema.

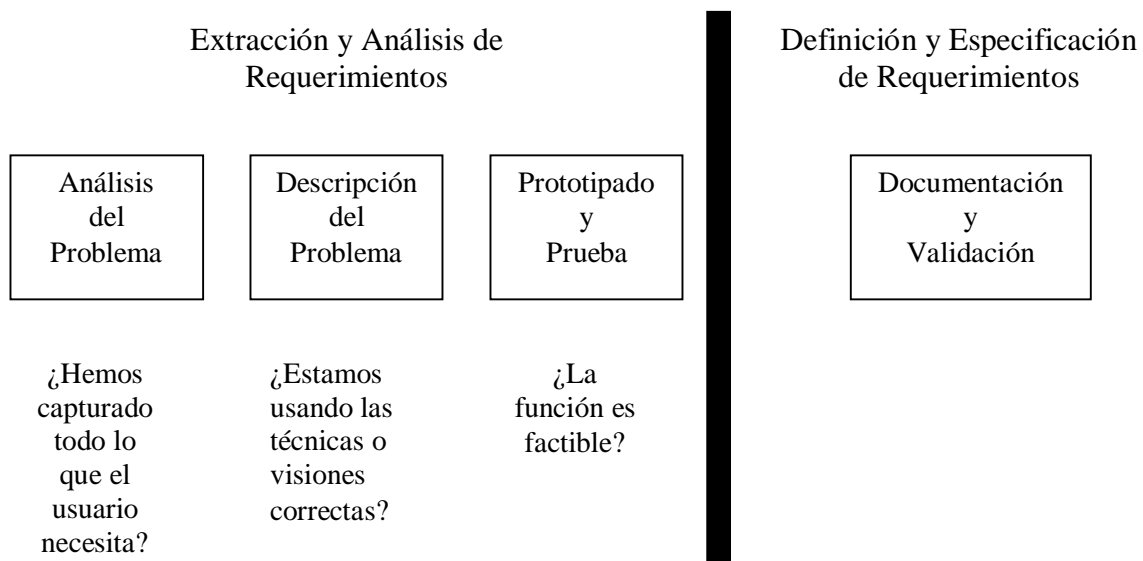


Figura 1.1 Proceso determinación requerimientos sistema

### Diseño del Sistema

Diseño es el proceso de transformación del problema en una solución.

Para el diseño de software se utiliza la especificación de requerimientos para definir el problema y para satisfacer los requerimientos planteados en la especificación. Para transformar

los requerimientos en un sistema que funcione, los diseñadores deben satisfacer tanto a los clientes como a los constructores de sistemas de su propio equipo de desarrollo.

El diseño es un proceso iterativo que consta de dos partes. Primero se elabora un diseño conceptual o diseño del sistema que le dice al cliente, exactamente que hará dicho sistema. Una vez que el cliente aprueba el diseño conceptual, se vuelve este diseño en un documento mucho más detallado, el diseño técnico, que permite que los constructores del sistema comprendan el hardware y el software concretos que se necesitan para resolver el problema del cliente. El proceso es iterativo porque en la realidad los diseñadores se mueven de una a otra de las diferentes actividades que abarcan la comprensión de requerimientos, la formulación de soluciones posibles, la comprobación de algunos aspectos de la solución, como la factibilidad, la presentación de alternativas a los clientes y la documentación del diseño de los programadores. Muchas veces el diseño se describe en un único documento, pero a veces se vuelve en dos documentos, como se ilustra en la figura 1.2.

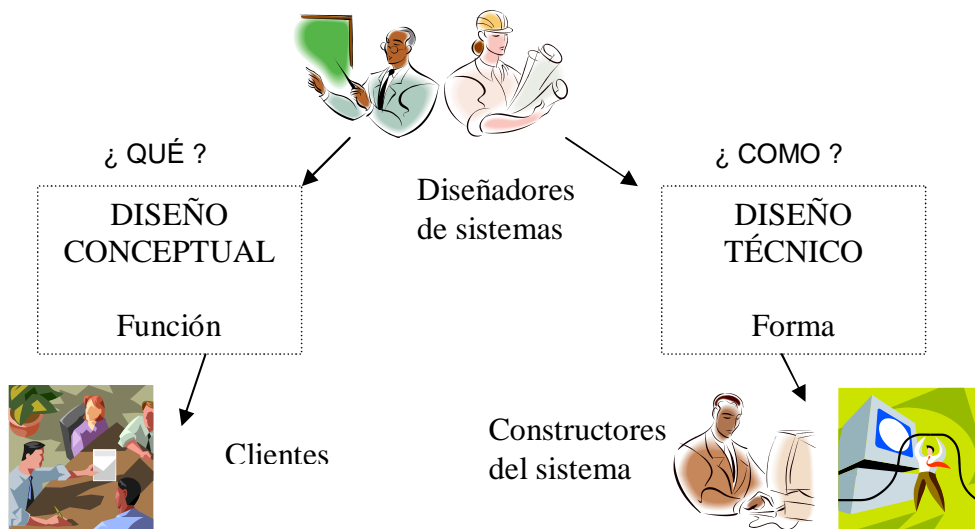


Figura 1.2 Diseño conceptual y técnico.

### Diseño de Programas

Comienza con los objetos y clases provenientes del diseño del sistema, pero es preciso perfeccionarlos y modificarlos para que se incluyan algunos aspectos relevantes a nivel de diseño.

### Codificación

Se centra en la implementación de la solución como software. Es decir deben escribirse los programas que implementan el diseño.

## **Pruebas de los programas**

Las pruebas se concentran en la búsqueda de defectos, y hay muchas maneras de hacer que los esfuerzos de la prueba sean más eficientes, estas son las pruebas unitarias en las cuales se verifica cada componente del programa en si mismo, aislado de los demás. La prueba de integración es el proceso de verificar que los componentes del sistema trabajan juntos conforme a lo descrito en las especificaciones del diseño del programa y del sistema.

## **Mantenimiento**

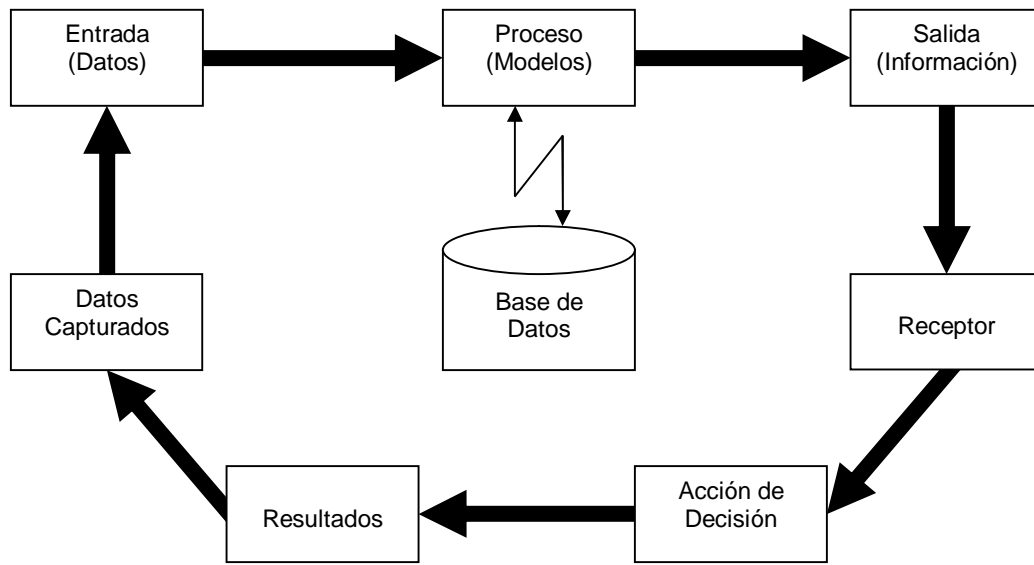
Una vez que el sistema esta completamente implementado y probado, se pone en marcha. El sistema esta ahora en la fase de mantenimiento en la que se llevan a cabo las siguientes tareas:

- Monitorización de las prestaciones del sistema. Si las prestaciones caen por debajo de un determinado nivel, puede ser necesario reorganizar la base de datos.
- Mantenimiento y actualización del sistema, cuando sean necesario, los nuevos requisitos que vayan surgiendo se incorporan al sistema, siguiendo de nuevo las etapas del ciclo de vida que se acaban de presentar.

## **1.3 SISTEMAS DE INFORMACION**

Un sistema de información coloca la información en un contexto significativo y útil y se ha comunicado a un receptor, quien la utiliza para tomar decisiones. La información implica la comunicación y recepción de inteligencia o conocimiento. Evalúa y notifica, sorprende y estimula, reduce la incertidumbre e influye sobre otros individuos y los estimula a la acción. Especialmente en la enseñanza, la información debe dar señales oportunas de aviso y anticipar el futuro.

La información esta compuesta de datos, imágenes, texto, documentos y voz, a menudo entrelazados en forma complicada, pero siempre organizados en un contexto significativo. Los datos que se van a procesar pueden ser datos de entrada, estar almacenados o ambos. Otro punto que se debe recordar es el ciclo de la información. Los datos se procesan mediante modelos para crear información; el receptor recibe la información y luego toma una decisión y actúa generando otras acciones o eventos, que a su vez crean diversos datos dispersos que se capturan y sirven como entrada; y el ciclo se vuelve a repetir



**Figura 1.3 El Ciclo de la Información**

La calidad de la información descansa sólidamente sobre tres pilares que son la exactitud, oportunidad y relevancia. Estos son los atributos claves de la información.

La exactitud significa que la información esté libre de errores, sea clara y refleje adecuadamente el sentido de los datos en los que se basa. Transmitiendo una imagen clara al receptor la cual puede requerir una presentación en forma grafica en vez de tabular.

El hacer llegar la información a los receptores dentro del marco de tiempo necesario es otro atributo clave de la calidad de la información. Obviamente, por lo general es de poco valor el periódico de ayer, las variaciones con respecto al estándar que se reportan después de que pueda tomarse una acción correctiva, o la información sobre las existencias uno o dos días después de los hechos. La oportunidad en la información significa simplemente que los receptores la puedan obtener cuando la necesitan.

La relevancia es el último atributo clave de la calidad de la información. En palabras sencillas ¿la información responde de manera especifica al receptor sobre el qué, por qué, dónde, cuándo, quién, cómo? Además, lo que es información relevante para un receptor, no lo es necesariamente para otro. La relevancia de la información de un pedido específico de un cliente, por ejemplo, variará entre los empleados de la compañía.

Los componentes esenciales de una organización pueden verse en función del área de trabajo, la cultura, la base de sus activos y los interesados y afectados. Para que una organización funcione sin obstáculos estos componentes deben estar orientados hacia los mismos objetivos y estar sincronizados entre sí. La información es el ingrediente clave que le permite a una organización lograr y mantener un estado de unidad y armonía.<sup>1</sup>

#### 1.4 ¿QUE ES UNA BASE DE DATOS?

La base de datos puede definirse como una colección de datos interrelacionados almacenados en conjunto sin redundancias perjudiciales o innecesarias; su finalidad es la de servir a una aplicación o más, de la mejor manera posible; los datos se almacenan de modo que resulten independientes de los programas que los usan; se emplean métodos bien determinados para incluir datos nuevos y para modificar o extraer los datos almacenados. Se dice que un sistema comprende una colección de bases de datos cuando éstas son totalmente independientes desde el punto de vista estructural.

Las bases de datos cumplen con algunos objetivos generales tales como:

- Desempeño
- Costo mínimo
- Redundancia mínima
- Capacidad de búsqueda
- Integridad
- Reserva y seguridad
- Afinación
- Migración de datos
- Simplicidad

**Desempeño.** Las bases de datos diseñadas para ser usadas por operadores de terminal deben asegurar un tiempo de respuesta adecuado para el diálogo entre el hombre y la terminal.

**Costo mínimo.** Con el fin de mantener bajo el costo hay que elegir técnicas que minimizan las necesidades totales de almacenamiento. Apelando a estas técnicas, la representación física de los datos en el almacén puede ser muy distinta de la representación que usa el programador.

<sup>1</sup> Diseño de Sistemas de Información

**Redundancia mínima.** Es necesario eliminar los valores de datos redundantes, siempre que resulte económico y controlar las incoherencias a que pueden dar lugar los valores redundantes. Para ello es importante identificar los datos que son utilizados en común por varias aplicaciones y almacenar una sola copia.

**Capacidad de búsqueda.** La capacidad para explorar una base de datos rápidamente y con diferentes criterios de búsqueda depende mucho de la organización física de los datos. Con algunas organizaciones los tiempos de búsqueda son excesivamente prolongados para poder considerar las respuestas como de tiempo real. Uno de los objetivos de la organización es, pues, el de lograr capacidad para la búsqueda rápida y flexible.

**Integridad.** Cuando una base de datos incluye información utilizada por muchos usuarios, es importante que no pueda destruirse los datos almacenados ni las relaciones que existen entre los distintos agregados. Ocasionalmente se producirán fallos de hardware y diversos tipos de accidentes. El almacenamiento de los datos y los procedimientos de actualización e inserción deben asegurar que el sistema pueda recuperarse de estas contingencias sin daños para los datos.

**Reserva y Seguridad.** Se refiere a la protección de los datos contra el acceso accidental o intencional por parte de personas no autorizadas y contra su indebida destrucción o alteración, así como a la reserva al derecho de los individuos y organismos para determinar por si mismos cuando, como y en que medida se permitirá la transmisión a terceros de la información que les concierne.

**Afinación.** La correcta afinación tiene dos requisitos. Se necesita la independencia física de los datos, así como se requieren medios para supervisar automáticamente el uso de la base de datos con el fin de que puedan hacerse los ajustes necesarios.

**Migración de datos.** Es deseable almacenar los datos de uso frecuente de manera que resulte fácil y rápido acceder a ellos. Los datos de uso ocasional se almacenarán, en cambio de manera más económica. Toda base de datos más o menos compleja tendrá múltiples niveles de facilidad de acceso.

**Simplicidad.** Los medios que se utilizan para representar la vista general de los datos deben ser concebidos de manera simple y nítida.<sup>2</sup>

## 1.5 NORMALIZACION

La normalización es un conjunto de reglas que sirven para ayudar a los diseñadores a desarrollar un esquema que minimice los problemas de lógica. Cada regla esta basada en la que le antecede. La normalización se adoptó porque el viejo estilo de poner todos los datos en un solo lugar, como un archivo o una tabla de base de datos, era ineficiente y conducía a errores de lógica cuando se trataba de manipular los datos.

Otra ventaja de la normalización de una base de datos es el consumo de espacio. Una base de datos normalizada puede ocupar menos espacio en disco que una no normalizada. Hay menos repeticiones de datos, lo que tiene como consecuencia mucho menor uso de espacio en disco.

Existen básicamente tres niveles de normalización: Primera Forma Normal (1NF), Segunda Forma Normal (2FM), y Tercera Forma Normal (3FN). Cada una de estas formas tiene sus propias reglas. Cuando una base de datos se conforma a un nivel se considera normalizada a esa forma de normalización.

**Primera Forma Normal.** La regla de la Primera Forma Normal (1FN) establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas. Se debe añadir un campo clave a la segunda tabla de forma que se establezca la relación.

La normalización ayuda a clarificar la base de datos y organizarla en partes más pequeñas y mas fáciles de entender. En lugar de tener que entender una tabla gigantesca y monolítica que tiene muchos diferentes aspectos, solo se tiene que entender objetos pequeños y más tangibles, así como las relaciones que guardan con otros objetos también pequeños.

**Segunda Forma Normal.** La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias palabras. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la clave de la tabla para identificarlos. Al haber alcanzado la Segunda Forma Normal, se puede disfrutar de algunas de las ventajas de las bases de datos relacionales, como por ejemplo se pueden añadir nuevas columnas a la tabla sin afectar las demás tablas, aplicando lo mismo para las otras tablas. Alcanzar este nivel de normalización permite que los datos se acomoden de una manera natural dentro de los límites esperados.

Una vez que ha alcanzado el nivel de la Segunda Forma Normal, se han controlado la mayoría de los problemas lógicos. Donde se pueden insertar un registro sin un exceso de datos en la mayoría de las tablas.

**Tercera Forma Normal.** La regla de la Tercera Forma Normal señala que hay que eliminar y separar cualquier dato que no sea clave. El valor de esta columna debe depender de la clave. Todos los valores deben identificarse únicamente por la clave. Esto le da más flexibilidad y proviene errores de lógica cuando inserta o borra registros. Cada columna en la tabla está identificada de manera única por la clave, y no hay datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir. Las reglas de normalización existen como guías para crear tablas que sean fáciles de manejar, así como flexibles y eficientes. Hay que tomar en cuenta que la normalización no es una ciencia exacta. Es subjetiva.

Existen seis niveles más de normalización que solo se mencionaran en este documento los cuales son Forma Normal Obice-Codd, Cuarta Forma Normal, Quinta Forma Normal o Norma Formal de Proyección-Unión, Forma Normal de Proyección-Unión Fuerte, Forma Normal de Proyección-Unión Extra Fuerte y Forma Normal de Clave de dominios. Estas formas de normalización pueden llevar las cosas más allá de lo que se necesita. Existen para hacer una base de datos realmente relacional. Tiene que ver principalmente con dependencias múltiples y claves relacionales.<sup>2</sup>

## 1.6 MODELO ENTIDAD-RELACION

El modelo de datos Entidad-Relación (E-R) está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre estos objetos. Una entidad es una cosa u objeto en el mundo real que es distinguible de otros objetos. Por ejemplo, cada persona es una entidad y las cuentas bancarias pueden ser consideradas entidades. Las entidades se describen en una base de datos mediante un conjunto de atributos, por ejemplo, los atributos número-cuenta y saldo describe una cuenta particular de un banco. Una relación es una asociación entre varias entidades como una relación impositor asocia un cliente con cada cuenta que tiene. El conjunto de todas las entidades del mismo tipo y el conjunto de todas las relaciones del mismo tipo se denominan conjuntos entidades y conjunto de relaciones, respectivamente.

Además de entidades y relaciones, el modelo E-R representa ciertas ligaduras que los contenidos de la base de datos deben cumplir. Una ligadura importante es la correspondencia de cardinalidades que expresa el número de entidades con la que otra entidad se puede asociar a través de un conjunto de relaciones.

La totalidad de estructuras lógicas de una base de datos se pueden expresar gráficamente mediante un diagrama E-R que consta de los siguientes componentes:

<sup>2</sup> Fundamentos de Bases de Datos

- Rectángulos, que representan conjuntos de entidades.
- Elipses, que representan atributos.
- Rombos, que representan relaciones entre conjuntos de entidades.
- Líneas, que unen los atributos con los conjuntos de entidades y los conjuntos de entidades con las relaciones.

Cada componente se etiqueta con la entidad o relación que represente. Como ilustración considérese parte de una base de datos de un sistema bancario consistente en clientes y cuentas que tienen esos clientes<sup>3</sup>.

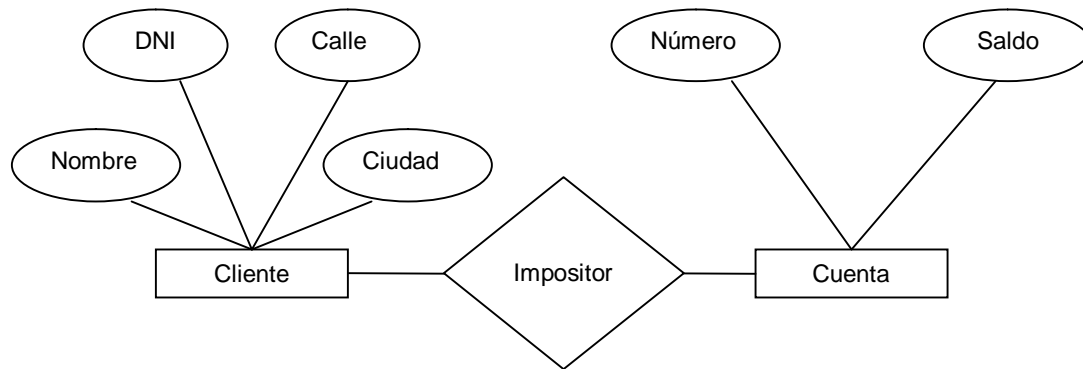


Figura 1.4 Diagrama E-R

## 1.7 BASES DE DATOS RELACIONALES

En una computadora existen diferentes formas de almacenar información. Esto da lugar a distintos modelos de organización de la base de datos: jerárquico, red, relacional y orientada a objeto.

Los sistemas relacionales son importantes porque ofrecen muchos tipos de procesos de datos, como: simplicidad y generalidad, facilidad de uso para el usuario final, períodos cortos de aprendizaje y las consultas de información se especifican de forma sencilla.

Las tablas son un medio de representar la información de una forma más compacta y es posible acceder a la información contenida en dos o más tablas.

Las bases de datos relacionales están constituidas por una o más tablas que contienen la información ordenada de una forma organizada. Cumplen las siguientes leyes básicas:

<sup>3</sup> Organización de las Bases de Datos

- Generalmente, contendrán muchas tablas.
- Una tabla sólo contiene un número fijo de campos.
- El nombre de los campos de una tabla es distinto.
- Cada registro de la tabla es único.
- El orden de los registros y de los campos no están determinados.
- Para cada campo existe un conjunto de valores posible.

El primer paso para crear una base de datos, es planificar el tipo de información que se quiere almacenar en la misma, teniendo en cuenta dos aspectos: la información disponible y la información que necesitamos.

La planificación de la estructura de la base de datos, en particular de las tablas, es vital para la gestión efectiva de la misma. El diseño de la estructura de una tabla consiste en una descripción de cada uno de los campos que componen el registro y los valores o datos que contendrá cada uno de esos campos.

Los campos son los distintos tipos de datos que componen la tabla, por ejemplo: nombre, apellido, domicilio. La definición de un campo requiere: el nombre del campo, el tipo de campo, el ancho del campo, etc.

Los registros constituyen la información que va contenida en los campos de la tabla, por ejemplo: el nombre del paciente, el apellido del paciente y la dirección de éste. Generalmente los diferente tipos de campos que se pueden almacenar son los siguientes: Texto (caracteres), Numérico (números), Fecha / Hora, Lógico (informaciones lógicas si/no, verdadero/falso, etc., imágenes.

En resumen, el principal aspecto a tener en cuenta durante el diseño de una tabla es determinar claramente los campos necesarios, definirlos en forma adecuada con un nombre especificando su tipo y su longitud.<sup>4</sup>

## **1.8 ESQUEMA CLIENTE-SERVIDOR**

El esquema cliente-servidor "es un modelo de computación en el que el procesamiento requerido para ejecutar una aplicación o conjunto de aplicaciones relacionadas se divide entre dos o más procesos que cooperan entre sí". Usualmente la mayoría del trabajo pesado se hace en el proceso llamado servidor y el (los) proceso(s) cliente(s) sólo se ocupa de la interacción con el usuario (aunque esto puede variar).

Los principales componentes del esquema cliente-servidor son entonces los Clientes, los Servidores y la infraestructura de comunicaciones.

Los Clientes interactúan con el usuario, usualmente en forma gráfica. Frecuentemente se comunican con procesos auxiliares que se encargan de establecer conexión con el servidor, enviar el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad.

Los Servidores proporcionan un servicio al cliente y devuelven los resultados. En algunos casos existen procesos auxiliares que se encargan de recibir las solicitudes del cliente, verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente. Además deben manejar los interbloqueos, la recuperación ante fallas, y otros aspectos afines. Por las razones anteriores la plataforma computacional asociada con los servidores es más poderosa que la de los clientes. Por esta razón se utilizan PCs poderosos, estaciones de trabajo, minicomputadoras o sistemas grandes. Además deben manejar servicios como administración de la red, mensajes, control y administración de la entrada al sistema, auditoria y recuperación y contabilidad. Usualmente en los servidores existe algún tipo de servicio de bases de datos.

Para que los clientes y los servidores puedan comunicarse se requiere una infraestructura de comunicaciones, la cual proporciona los mecanismos básicos de direccionamiento y transporte. La mayoría de los sistemas Cliente/Servidor actuales se basan en redes locales y por lo tanto utilizan protocolos no orientados a conexión, lo cual implica que las aplicaciones deben hacer las verificaciones. La red debe tener características adecuadas de desempeño, confiabilidad, transparencia y administración.

Como ejemplos de clientes pueden citarse interfaces de usuario para enviar comandos a un servidor, APIs para el desarrollo de aplicaciones distribuidas, herramientas en el cliente para hacer acceso a servidores remotos (por ejemplo: servidores de SQL) o aplicaciones que solicitan acceso a servidores para algunos servicios.

Como ejemplos de servidores pueden citarse servidores de ventanas como X-windows, servidores de archivos como NFS, servidores para el manejo de bases de datos, como los servidores de SQL, servidores de diseño y manufactura asistidos por computador, etc.

Además de lo anterior, el esquema Cliente/Servidor facilita la integración entre sistemas diferentes y compartir información, permitiendo, por ejemplo que las máquinas ya existentes puedan ser utilizadas pero utilizando interfaces más amigables al usuario. De esta manera

podemos, por ejemplo, integrar PCs con sistemas medianos y grandes, sin que todas las máquinas tengan que utilizar el mismo sistema operacional.

El esquema Cliente/Servidor también favorece la adaptación a cambios en la tecnología pues facilita la migración de las aplicaciones a otras plataformas y, al aislar claramente las diferentes funciones de una aplicación hace más fácil incorporar nuevas tecnologías en esta.

## 1.9 ¿QUE ES LA WWW?

El WWW es un esquema que trabaja bajo el sistema cliente-servidor. El cliente, también llamado browsers o visualizador por su capacidad de interacción, es el programa encargado de hacer una petición de documentos o información y una vez que la información es recibida la despliega al usuario. El servidor, por su parte, es el programa que una vez recibida la petición da respuesta enviando el documento requerido, es el encargado de almacenar información.

La WWW está compuesta por miles de intercambios de información de este tipo que se dan alrededor de todo el mundo, de ahí su nombre World Wide Web este es un sistema de información global, interactivo, dinámico, distribuido, gráfico, basado en hipertexto, con plataforma de enlaces cruzados, que se ejecuta en Internet.

Tanto servidores como clientes deben ser capaces de hablar y entender (comunicarse) a través de HTTP (HyperText Transfer Protocol o Protocolo de Comunicación y transmisión de documentos en hipertexto).

El lenguaje utilizado para crear y reconocer los documentos en hipertexto es el HTML Los documentos en HTML son archivos de código ASCII (8 bits) que contienen información sobre layout (estilos, títulos, listas, párrafos) y sobre hiperligas.

Hipertexto es básicamente lo mismo que un texto común, puede guardarse, leerse, editarse, etc., con una importante ventaja: hipertexto contiene conexiones a otros documentos, es decir, es un texto interactivo. A esas conexiones que se realizan simplemente seleccionando una palabra (generalmente resaltada del texto) se les conoce con el nombre de hiperligas.

Finalmente, hipermedia es una combinación de hipertexto y multimedia, esto es, las conexiones que realiza no son sólo a otros documentos de texto, sino a una variedad de formas de media como audio, imágenes y movimientos entre otros.

World Wide Web utiliza los llamados URLs (Uniform Resource Locators) para representar las conexiones en hipermedia y a otros servicios en red, como los documentos. El URL tiene dos

partes básicas: la primera especifica el método de acceso, luego se escriben dos slashes (//) y la segunda parte es la dirección de la computadora donde se localiza el documento o servicio deseado.<sup>5</sup>

## **1.10 TECNOLOGIAS DE ACCESO A BASE DE DATOS**

### **1.10.1 PAGINAS DINAMICAS**

Una de las principales ventajas que presenta el trabajar con páginas dinámicas es el poder almacenar los contenidos en bases de datos. De esta forma, podemos organizarlos, actualizarlos y buscarlos de una manera mucho más simple.

ASP nos ofrece una forma muy eficaz de interaccionar con estas bases de datos gracias al uso del componente ADO (ActiveX Data Objects) el cual permite acceder a dichas bases de una forma sencilla.

Este ADO no es más que un conjunto de objetos que, utilizados en conjunto, nos permiten explotar de una forma muy versátil las bases de datos de una aplicación.

Por otra parte, los scripts ASP deben establecer un dialogo con la base de datos. Este dialogo se lleva a cabo a partir de un idioma universal: el SQL (Structured Query Language) el cual es común a todas las bases de datos.

### **1.10.2 ACTIVE SERVER PAGES**

Active Server Pages (ASP) es una nueva tecnología creada por Microsoft, destinada a la creación de sitios Web. No se trata de un lenguaje de programación en sí mismo (ya que los ASP se pueden programar en VBScript, JavaScript, PerlScript o en varios otros lenguajes), sino de un marco sobre el que construir aplicaciones basadas en Internet.

La tecnología ASP apareció por primera vez con el servidor Internet Information Server 3.0 de Microsoft en Diciembre de 1996. La versión 4.0 de IIS (el Option Pack para NT 4.0) incluye la versión 2.0 de ASP, y la versión 5.0 de IIS, distribuida con Windows 2000, incluye ASP 3.0.

Los predecesores de ASP incluyen CGI y Perl. Las tecnologías de Microsoft predecesoras de ASP incluyen IDC y WebDB. Otras tecnologías que compiten con ASP son ColdFusion (Allaire), Java Server Pages (Sun Microsystems) y PHP (de libre distribución bajo Open System).

<sup>5</sup> [www.qualitrain.com.mx](http://www.qualitrain.com.mx)

Algunas de las características de ASP son:

- ASP es totalmente gratuito para Microsoft Windows NT o Windows 95/98.
- El código ASP se puede mezclar con el código HTML en la misma página (no es necesario compilarlo por separado).
- El código ASP se puede escribir con un simple editor de textos como el Bloc de notas de Windows.
- Cómo el código ASP se ejecuta en el servidor, y produce como salida código HTML puro, su resultado es entendible por todos los navegadores existentes.
- Mediante ASP se pueden manipular bases de datos (consultas, actualizaciones, borrados, etc.) de prácticamente cualquier plataforma, con tal de que proporcione un driver OLEDB u ODBC.
- ASP permite usar componentes escritos en otros lenguajes (C++, Visual Basic, Delphi), que se pueden llamar desde los guiones ASP.
- Sin modificar la instalación, los guiones ASP se pueden programar en JScript o VBScript (este último es el más usado porque mas programadores lo dominan), pero también existen otros lenguajes, como Perlscript y Rexx, que se pueden emplear para programar ASP.
- Se ha portado a la plataforma Java por Chili!Soft y Halcyon Software, lo que permite que ASP sea usado en casi cualquier sistema operativo.

Ventajas

Las principales ventajas que ofrece ASP son:

1. Permite acceder a bases de datos de una forma sencilla y rápida.
2. Las páginas se generan dinámicamente mediante el código de scripts, (guiones).
3. El código de script se ejecuta en el servidor, y no se depende del navegador que se emplee.
4. Desde una página ASP se pueden ejecutar servidores OLE en el servidor de Web, lo que abre un abanico de nuevas posibilidades sólo accesibles previamente usando CGI y filtros ISAPI: acceso a base de datos, acceso a ficheros, logging en el sistema, envío de correo, etc.

La tecnología ASP se emplea principalmente para crear aplicaciones interactivas que funcionan en Internet. <sup>6</sup>

### 1.10.3 PAGINAS JAVA SERVER

Una página Web JSP parece similar a una página Web estática; en su mayor parte es código HTML, pero con algunas etiquetas especiales encerrando código fuente java. En realidad, la página es un archivo de código fuente java utilizado para definir un servlet. Cuando la página se procesa por el servidor Web, está se convierte en autentico código fuente java, se compila y se ejecutan como un servlet.

Debido a que una página JSP es código fuente java, debe respetar los requerimientos del lenguaje. Una única línea errónea de java destruirá la página completa, no solo la línea afectada. La página debe, tras el filtrado por la lógica de JSP, ser como un código fuente válido para la implementación de un servlet. En otras palabras, no es fácil escribir una página JSP sin errores<sup>7</sup>.

### 1.10.4 SERVLETS

Los servlets son programas en java que son precompilados por el programador, entonces puestos en el servidor de paginas (web server) que los carga, se compila y ejecuta al momento que un usuario remoto lo solicita, generalmente a través de una página html, algunas de sus características son:

- Son independientes del servidor utilizado así como del sistema operativo.
- Son seguros, ya que como son llamados a través del Web Server la lógica del programa no es expuesta al cliente.
- Puede llamar a otros servlets o métodos de otros servlets para que de esta forma pueda distribuir la manera de realizar el trabajo y en caso de que alguno de los servlets tuviera un error no afecta el procesamiento del resto de los servlets.
- Son estables ya que se ejecutan independientemente del Web Server en caso de error no afectan el funcionamiento del Web Server.
- Permiten el uso de cookies y sesiones de tal forma que se puede guardar información específica acerca del usuario que esta utilizando el servicio y de esta forma mantener una sesión con el mismo usuario.
- Son persistentes en el sentido que permiten compartir información entre múltiples usuarios en una misma sesión.

<sup>6</sup> HTML Dinámico ASP y Java Script a través de Ejemplos

Permiten la implementación de arquitectura cliente servidor de tres capas, esto sirve como enlace entre el cliente y una o varias bases de datos, las cuales pueden estar físicamente en otro servidor.<sup>7</sup>

### 1.10.5 SQL

SQL se ha establecido como el lenguaje estándar de bases de datos relacionales, hay numerosas versiones de SQL. La versión original se desarrolló en IBM, este lenguaje, originalmente denominado Sequel, se implementó como parte del proyecto System R, a principios de 1970. El lenguaje sequel ha evolucionado desde entonces y su nombre ha pasado a ser SQL (Structured Query Language, Lenguaje estructurado de consultas). Actualmente numerosos productos son compatibles con el lenguaje SQL.

Sus principales componentes de SQL son:

**Lenguaje de definición de datos (LDD).** El LDD de SQL proporciona órdenes para la definición de esquemas de relación, borrado de relaciones, creación de índices y modificación de esquemas de relación.

**Lenguaje interactivo de manipulación de datos (LMD).** El LMD de SQL incluye un lenguaje de consulta, basado tanto en el álgebra relacional como en el cálculo relacional de tuplas. Incluye también órdenes para insertar, borrar y modificar tuplas de la base de datos.

**LMD incorporado.** La información incorporada de SQL se diseñó para el uso sin lenguajes de programación de propósito general, tales como PL/1, Cobol, Pascal, Fortran y C.

**Definición de vistas.** El LDD de SQL incluye órdenes para la definición de vistas.

**Autorización.** El LDD de SQL incluye órdenes para la especificación de las ligaduras de integridad que deben satisfacer los datos almacenados en la base de datos. Las actualizaciones que violen las ligaduras de integridad se rechazan.

**Control de transacciones.** SQL incluye órdenes para la especificación de comienzo y final de transacciones. Varias implementaciones permiten también bloqueo explícito de los datos para el control de la concurrencia.<sup>8</sup>

<sup>7</sup> Programación en Java

<sup>8</sup> Aprendiendo My SQL

## **2. ANALISIS**

### **2.1 PLANTEAMIENTO DEL PROBLEMA**

Respondiendo a las nuevas necesidades de preparatorias una preparatoria tiene la necesidad de controlar la información de su institución por ello desea obtener un Sistema Web Administrativo, en el cual se pueda ofrecer información tanto para alumnos, docentes de la misma institución como para el público fuera de la institución, ofreciéndoles ligas de servicio, control de inventario como lo es recursos físicos, salones, academias, en donde el objetivo principal es que el docente o alumno llene un formulario con sus datos personales, seleccione algunos de los servicios y obtenga la información necesaria para tener un control de la misma a través de la red, esta información a su vez será supervisada por un administrador el cual le dará su correspondiente mantenimiento de alta de datos como lo son equipos, cursos, salones, talleres o baja de los mismos y modificaciones.

Se pretende realizar una aplicación Web en donde poder consultar información de cursos, talleres, actividades, docentes, alumnos y servicios útiles para los mismos donde un administrador pueda manipular la información para tener listo un reporte acerca de con que recursos cuenta y darle seguimiento a dichos recursos para hacer solicitud de los mismos o en su caso poder controlar dichos recursos para con ello estar a la vanguardia verificando toda la información almacenada en el sistema por los usuarios y el mismo administrador y darles seguimiento todo esto vinculando bases de datos con páginas Web

## 2.2 CASOS DE USO

A continuación se muestra la lista de casos de uso del sistema:

No.	NOMBRE	DESCRIPCION
1.	Mantenimiento de alta de datos	El administrativo tendrá acceso al sistema para dar alta datos como equipos, salones, cursos.
2.	Mantenimiento de baja de datos	El administrativo tendrá acceso al sistema para dar baja datos que estén descompuestos o discontinuados.
3.	Mantenimiento de modificación de datos	El administrativo tendrá acceso al sistema para modificar datos existentes.
4.	Alta de Docente	El docente llenará un formato con sus datos personales para poder acceder.
5.	Consulta de equipos	Se mostrará una lista de los equipos dados de alta.
6.	Consulta de equipo por búsqueda de nombre	Se mostrará el equipo el cual tendrá clave del equipo, descripción del equipo, marca o y datos del equipo.

### 2.2.1 CASOS DE USO 1.

**Nombre:** Mantenimiento de Alta de Equipo  
**Objetivo:** Que el administrativo del sistema WEB pueda dar de alta equipos o recursos  
**Precondiciones:** Que el administrativo tenga cuenta de acceso al sistema WEB  
**Actores:** Administrativo.

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El administrativo abre un browser
	2	El administrativo introduce una dirección al sitio de administración de recursos físicos WEB
	3	El browser despliega la página de entrada a control de recursos
	4	El administrativo introduce su número de trabajador
	5	El sistema verifica el número de trabajador
Variación 1	6	Si el administrativo tiene acceso al sistema el browser muestra las opciones de mantenimiento
	7	El administrativo selecciona la opción de alta de equipo
	8	El sistema muestra la página de alta de equipo
	9	El administrativo completa el formulario de alta de equipo <ul style="list-style-type: none"> <li>I. Número de serie del equipo</li> <li>II. Número de serie interno</li> <li>III. Marca</li> <li>IV. Color</li> <li>V. Procesador</li> <li>VI. Disco duro</li> <li>VII. Memoria</li> <li>VIII. Velocidad</li> <li>IX. Estado</li> <li>X. Ubicación</li> </ul>
	10	El administrativo acepta mediante un botón el alta del producto
Variación 2	11	El administrativo decide dar de alta otro equipo o regresa a las opciones principales

**Estado de fin exitoso:** El sistema indicará que la acción de alta fue terminada con éxito.  
**Estado de fin fallido:** El sistema advertirá que la acción de alta no fue exitosa.  
**Variaciones:** Variación 1. En caso de no tener acceso al sistema este mostrara una advertencia de usuario no autorizado.  
 Variación 2. En caso de que el administrador seleccione las opciones principales el sistema dará por concluida el alta.  
**Casos de uso relacionados:** Consulta de Equipo por índice general  
 Consulta de Equipo nombre  
**Interfaz de usuario:** Mantenimiento de Alta de equipo.

### 2.2.2 CASOS DE USO 2.

**Nombre:** Mantenimiento de Baja de Equipo  
**Objetivo:** Que el administrativo del sistema WEB pueda dar de baja los equipos que se encuentren descontinuados o descompuestos  
**Precondiciones:** Que el administrativo tenga cuenta de acceso al sistema WEB  
**Actores:** Administrativo.

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El administrativo abre un browser
	2	El administrativo introduce una dirección al sitio de administración de recursos físicos WEB
	3	El browser despliega la página de entrada a control de recursos
	4	El administrativo introduce su número de trabajador
	5	El sistema verifica el número de trabajador
Variación 1	6	Si el administrativo tiene acceso al sistema el browser muestra las opciones de mantenimiento
	7	El administrativo selecciona la opción de baja de equipo
	8	El sistema muestra la página de baja de equipo
	9	El administrativo introduce el número de serie del equipo
	10	El administrativo acepta mediante un botón la baja del equipo
Variación 2	11	El administrativo decide dar de baja otro equipo o regresa a las opciones principales

**Estado de fin exitoso:** El sistema indicara que la acción de baja fue terminada con éxito.

**Estado de fin fallido:** El sistema advertirá que la acción de baja no fue exitosa.

**Variaciones:** Variación 1. En caso de no tener acceso al sistema este mostrara una advertencia de usuario no autorizado.  
 Variación 2. En caso de que el administrador seleccione las opciones principales el sistema dará por concluida la baja.

**Casos de uso relacionados:** Consulta de Equipo por índice general  
 Consulta de Equipo nombre

**Interfaz de usuario:** Mantenimiento de Baja de equipo.

### 2.2.3 CASOS DE USO 3.

**Nombre:** Mantenimiento de Modificación de Equipo  
**Objetivo:** Que el administrativo del sistema WEB pueda modificar equipos existentes  
**Precondiciones:** Que el administrativo tenga cuenta de acceso al sistema WEB  
**Actores:** Administrativo.

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El administrativo abre un browser
	2	El administrativo introduce una dirección al sitio de administración de recursos físicos WEB
	3	El browser despliega la página de entrada a control de recursos
	4	El administrativo introduce su número de trabajador
	5	El sistema verifica el número de trabajador
Variación 1	6	Si el administrativo tiene acceso al sistema el browser muestra las opciones de mantenimiento
	7	El administrativo selecciona la opción de modificación de equipo
	8	El sistema muestra la página de modificación de equipo
	9	El administrativo completa el formulario de alta de equipo <ul style="list-style-type: none"> <li>I. Número de serie del equipo</li> <li>II. Número de serie interno</li> <li>III. Marca</li> <li>IV. Color</li> <li>V. Procesador</li> <li>VI. Disco duro</li> <li>VII. Memoria</li> <li>VIII. Velocidad</li> <li>IX. Estado</li> <li>X. Ubicación</li> </ul>
	10	El administrativo acepta mediante un botón la modificación del equipo
Variación 2	11	El administrativo decide modificar otro equipo o regresa a las opciones principales

**Estado de fin exitoso:** El sistema indicará que la modificación fue terminada con éxito.  
**Estado de fin fallido:** El sistema advertirá que la modificación no fue exitosa.  
**Variaciones:** Variación 1. En caso de no tener acceso al sistema este mostrara una advertencia de usuario no autorizado.  
 Variación 2. En caso de que el administrador seleccione opciones principales el sistema dará por concluida la modificación.  
**Casos de uso relacionados:** Consulta de Equipo por índice general  
 Consulta de Equipo nombre  
**Interfaz de usuario:** Mantenimiento de Modificación de equipos.

## 2.2.4 CASOS DE USO 4.

**Nombre:** Alta de Docentes  
**Objetivo:** El Docente llenara un formato con sus datos personales para poder consultar los servicios  
**Precondiciones:** No hay precondiciones  
**Actores:** Docente

---

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El docente abre un browser
	2	El docente introduce una dirección al sitio de la preparatoria
	3	El browser despliega la página de entrada al sitio WEB
	4	El docente selecciona la opción de acceso tecleando su numero de trabajador
	5	El docente completa el formulario de alta de docentes <ul style="list-style-type: none"> <li>I. Nombre de Docente</li> <li>II. Grado Académico</li> <li>III. Horario de Asesoría</li> <li>IV. Correo electrónico</li> </ul>
	6	El docente acepta mediante un botón el alta de docentes
	7	El sistema regresa a la página de entrada al sitio WEB

---

**Estado de fin exitoso:** El sistema indicará que la acción de alta de docentes fue terminada con éxito.  
**Estado de fin fallido:** El sistema advertirá que la acción de alta de docentes no fue exitosa.  
**Variaciones:**  
**Casos de uso relacionados:** Consulta de docentes dados de alta  
**Interfaz de usuario:** Alta de Docentes

### 2.2.5 CASOS DE USO 5.

**Nombre:** Consulta de equipo por el índice general  
**Objetivo:** Que el administrativo podrá consultar el catálogo de productos mediante un listado del índice general  
**Precondiciones:** Que el administrativo tenga cuenta de acceso al sistema WEB  
**Actores:** Administrativo

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El administrativo abre un browser
	2	El administrativo introduce una dirección al sitio de administración de recursos físicos WEB
	3	El browser despliega la página de entrada a control de recursos
	4	El administrativo introduce su número de trabajador
	5	El sistema verifica el número de trabajador
Variación 1	6	Si el administrativo tiene acceso al sistema el browser muestra las opciones de consulta
	8	El administrativo selecciona la opción de búsqueda por índice general
	9	El sistema muestra un desplegado mediante el índice general con opción de poder ser impresa
	10	El administrativo podrá elegir algunos de los equipos mediante botones de selección
Variación 2	11	El Administrativo acepta mediante un botón la acción a realizar
	12	El Administrativo decide regresar a las opciones principales o buscar por otro método.

**Estado de fin exitoso:** El sistema indicará que la acción y búsqueda fue terminada con éxito.  
**Estado de fin fallido:** El sistema advertirá que la acción de consulta no fue exitosa.  
**Variaciones:** Variación 1. En caso de no tener acceso al sistema este mostrara una advertencia de usuario no autorizado.  
 Variación 2. En caso de que el cliente seleccione las opciones principales el sistema dará por concluida la búsqueda  
**Casos de uso relacionados:** Consulta de Equipos dados de alta  
**Interfaz de usuario:** Consulta de equipos por el índice general

### 2.2.6 CASOS DE USO 6.

<b>Nombre:</b>	Consulta de equipos por búsqueda de nombre de equipo
<b>Objetivo:</b>	Se mostrara un equipo el cual tendrá nombre del equipo, descripción del equipo, marca, color, estado, ubicación
<b>Precondiciones:</b>	Que el administrativo tenga cuenta de acceso al sistema WEB
<b>Actores:</b>	Administrativo

FLUJO DE EVENTOS	EVENTO	PROCESO DEL CASO DE USO
	1	El administrativo abre un browser
	2	El administrativo introduce una dirección al sitio de administración de recursos físicos WEB
	3	El browser despliega la página de entrada a control de recursos
	4	El administrativo introduce su número de trabajador
	5	El sistema verifica el número de trabajador
Variación 1	6	Si el administrativo tiene acceso al sistema el browser muestra las opciones de búsqueda
	7	El administrativo selecciona la opción de búsqueda nombre del equipo
	8	El sistema muestra un equipo que corresponda al nombre del equipo
	9	El administrativo podrá elegir el equipo para realizar alguna acción
Variación 2	10	El administrativo acepta mediante un botón la acción
	11	El administrativo decide regresar a las opciones principales o buscar por otro método.

<b>Estado de fin exitoso:</b>	El sistema indicará que la acción de búsqueda fue terminada con éxito.
<b>Estado de fin fallido:</b>	El sistema advertirá que la acción de búsqueda no fue exitosa.
<b>Variaciones:</b>	Variación 1. En caso de no tener acceso al sistema este mostrara una advertencia de usuario no autorizado. Variación 2. En caso de que el cliente seleccione las opciones principales el sistema dará por concluida la búsqueda de equipo.
<b>Casos de uso relacionados:</b>	Consulta de Equipo dados de alta
<b>Interfaz de usuario:</b>	Consulta de equipo por búsqueda nombre de equipo

### 2.3 DIAGRAMA DE CASOS DE USO DEL SISTEMA

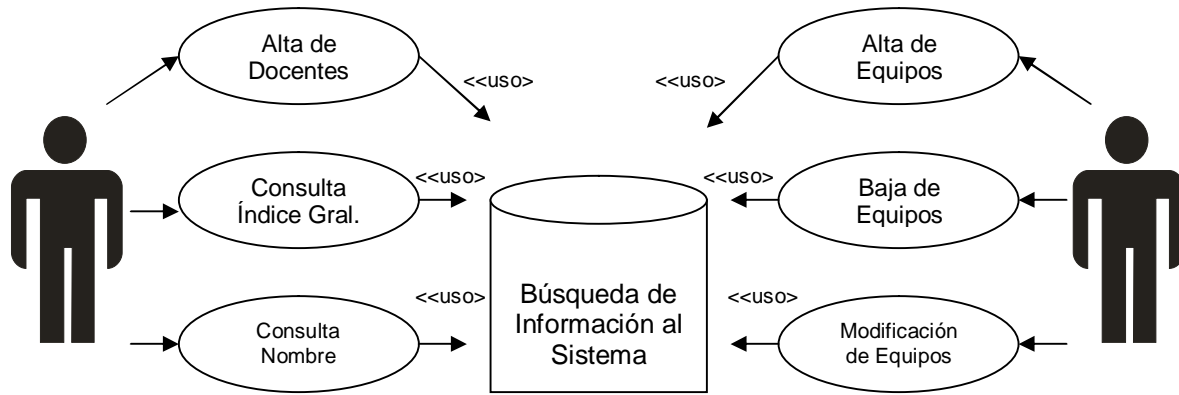


Figura 2.1 Diagramas de Casos de Uso del Sistema

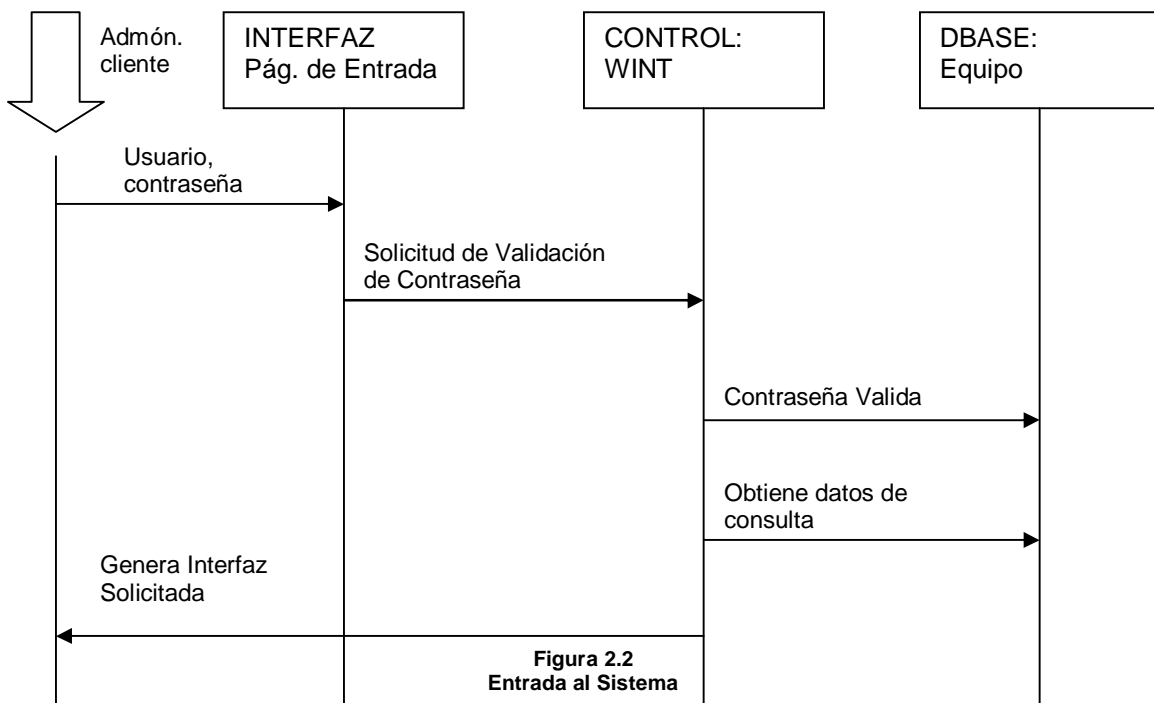
## 2.4 DIAGRAMAS DE SECUENCIA

En este punto se muestran los diagramas de secuencia que serán generados mediante los casos de uso que se describieron con anterioridad, en donde destacan los principales elementos como son:

- **AUTOR:** Es el individuo que utilizara el sistema y los módulos
- **INTERFAZ:** Se presentan los nombres de las páginas que serán presentadas en el browser para que el actor interactúe con el sistema
- **CONTROL:** Se muestra el nombre del servidor que presta atención a las peticiones del cliente o actor.
- **DBASE:** Hará referencia a los nombres de las instancias de las bases de datos con las que se llevaran acabo las operaciones.

### 2.4.1 DIAGRAMA DE SECUENCIA DE ENTRADA AL SISTEMA

Se muestra el flujo de la entrada para el sistema en donde el AUTOR puede ser el Administrador o Cliente entonces el administrador podrá dar de alta, baja o modificar equipos, así como la consulta de docentes dados de alta, mientras que el docente podrá dar de alta sus datos personales para después consultar catálogos de equipos por índice general, nombre del equipo, ubicación.



### 2.4.2 DIAGRAMA DE SECUENCIA DE ALTA DE EQUIPOS

El ACTOR es el administrador, la INTERFAZ es el alta de equipos, el CONTROL es WINT y la DBASE es equipos.

El sistema mostrará una página de registro de equipo, el administrativo captura los siguientes datos de equipos: Nombre del equipo, Número de serie, Número Control Interno, Marca, Procesador, Disco duro, Memoria, Velocidad, Color, Estado, Ubicación. Al terminar la captura se selecciona la inserción del registro con esto se conseguirá que se anexe a la base de datos catálogo de equipos.

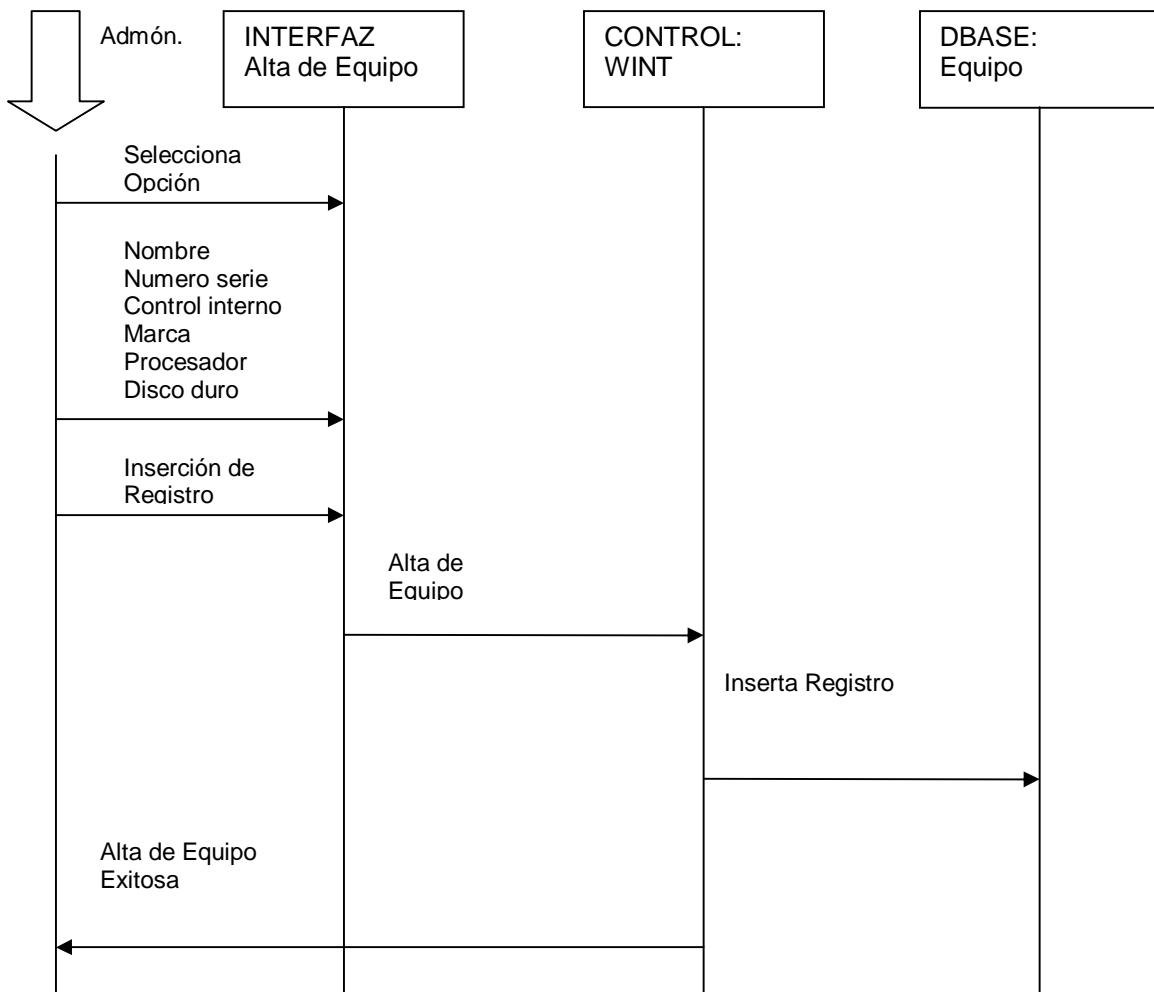


Figura 2.3  
Alta de Equipo

### 2.4.3 DIAGRAMA DE SECUENCIAS DE BAJA DE EQUIPOS

El ACTOR es el Administrativo, la INTERFAZ es baja de equipos, el CONTROL es WINT y DBASE es equipo.

El sistema mostrará la página de baja de equipo, el administrativo introducirá el nombre de equipo a dar de baja, se selecciona la baja del producto, borrándose de la base de datos equipo.

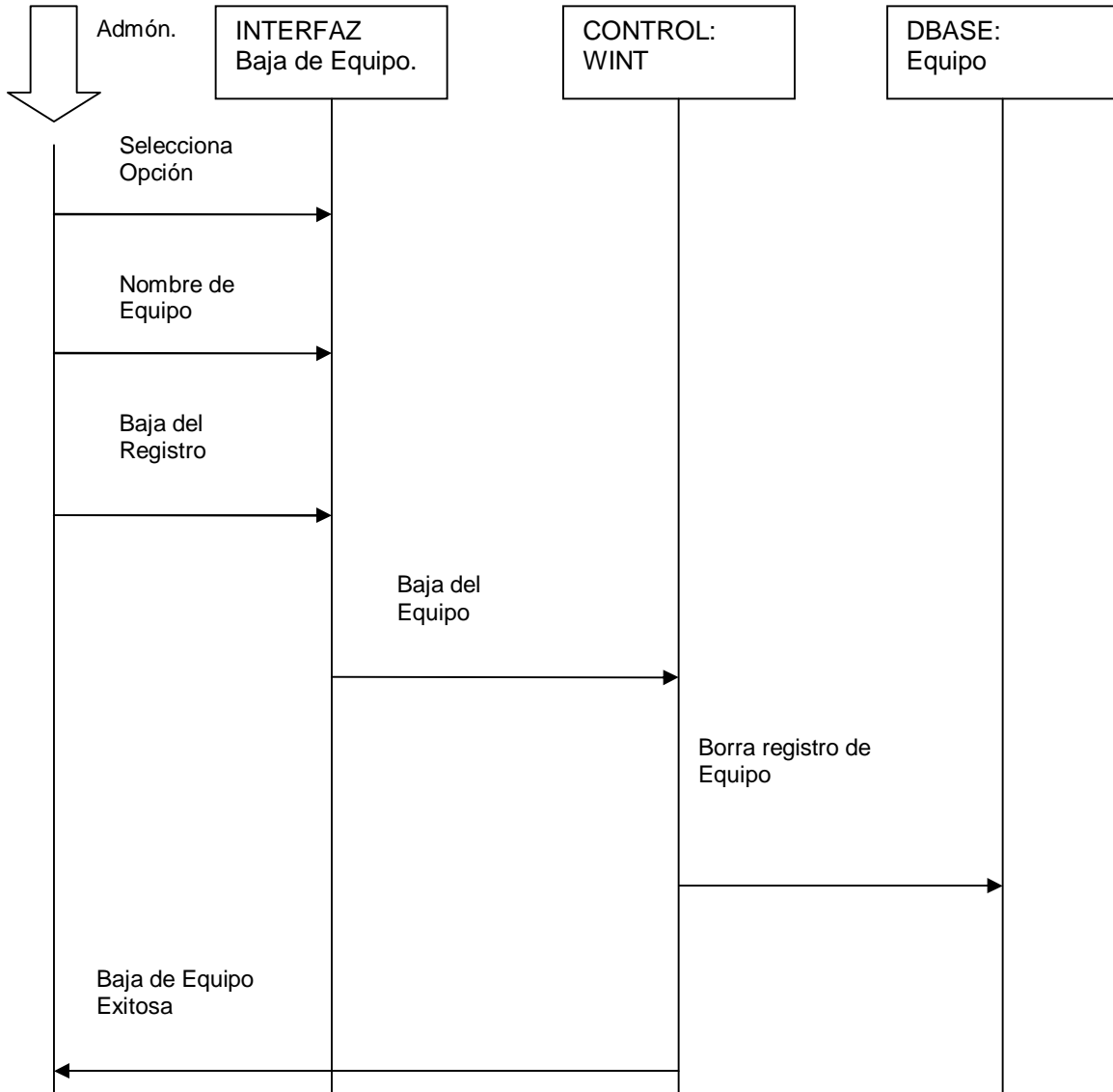


Figura 2.4  
Baja de Equipos

### 2.4.4 DIAGRAMA DE SECUENCIAS MODIFICACION DE EQUIPOS

El flujo de la modificación de equipos esta dado por el AUTOR que en este caso es el Administrativo, la INTERFAZ es Modificación de Equipos, CONTROL es WINT y DBASE es Equipos.

Se muestra la página de modificación de equipos con la información previamente capturada y el administrativo podrá modificar el Nombre del equipo, Número de serie, Número Control Interno, Marca, Procesador, Disco duro, Memoria, Velocidad, Color, Estado, Ubicación, del Equipo.

Al terminar la modificación del registro, se seleccionara la modificación de la base de datos

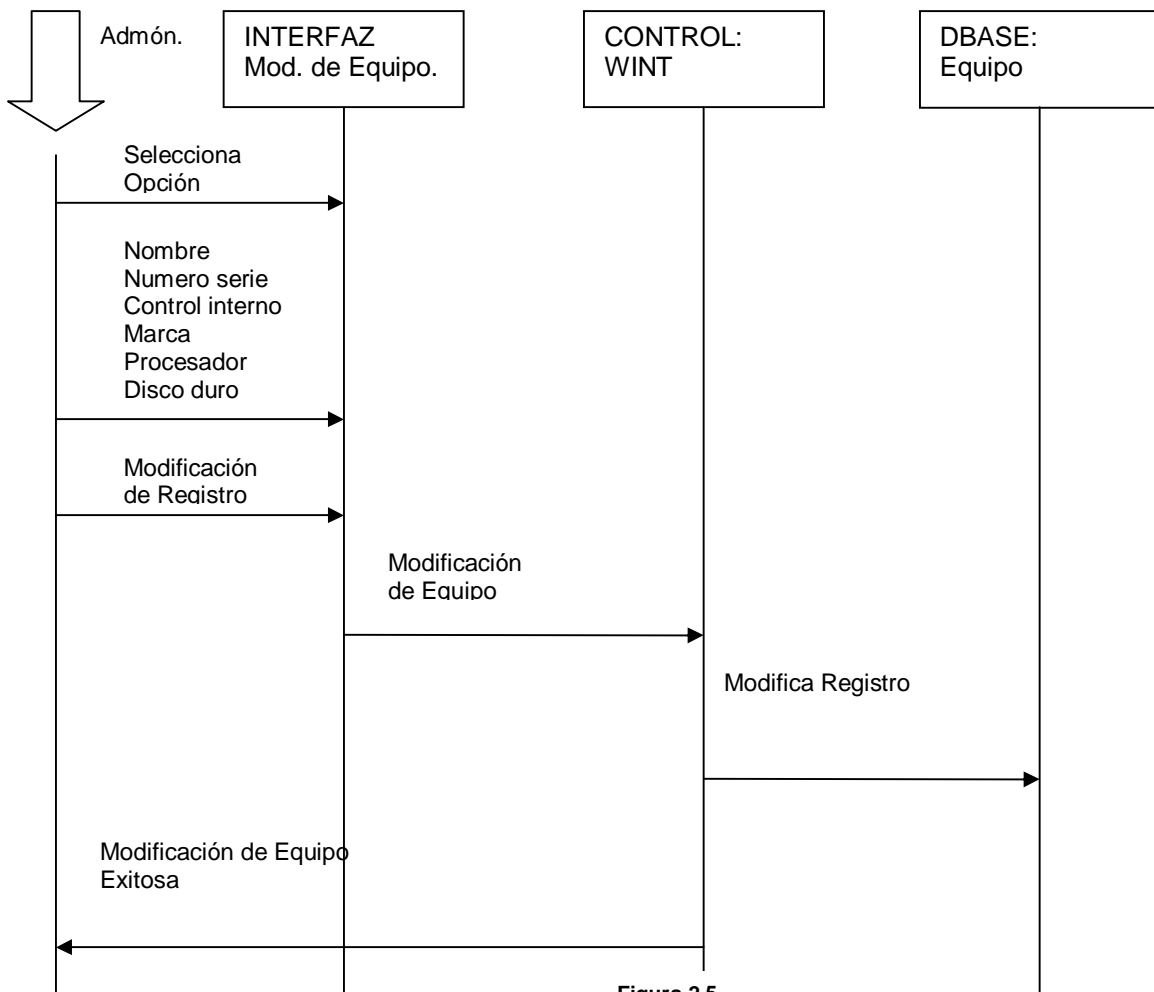


Figura 2.5  
Modificación de Equipos

### 2.4.5 DIAGRAMA DE SECUENCIA DE ALTA DE DOCENTES

El flujo de la alta de docentes está dado por el AUTOR que en este caso es el docente, la INTERFAZ es Alta de Docentes, CONTROL es WINT y DBASE es Docentes.

Se mostrará una página donde el docente podrá dar de alta sus datos personales tales como: Nombre de Docente, Grado Académico, Academia a la que pertenece, Horario de Asesoría, Correo electrónico, a modo de que queden almacenados en la base de datos docentes, para una consulta posterior, seleccionando al término alta de docentes

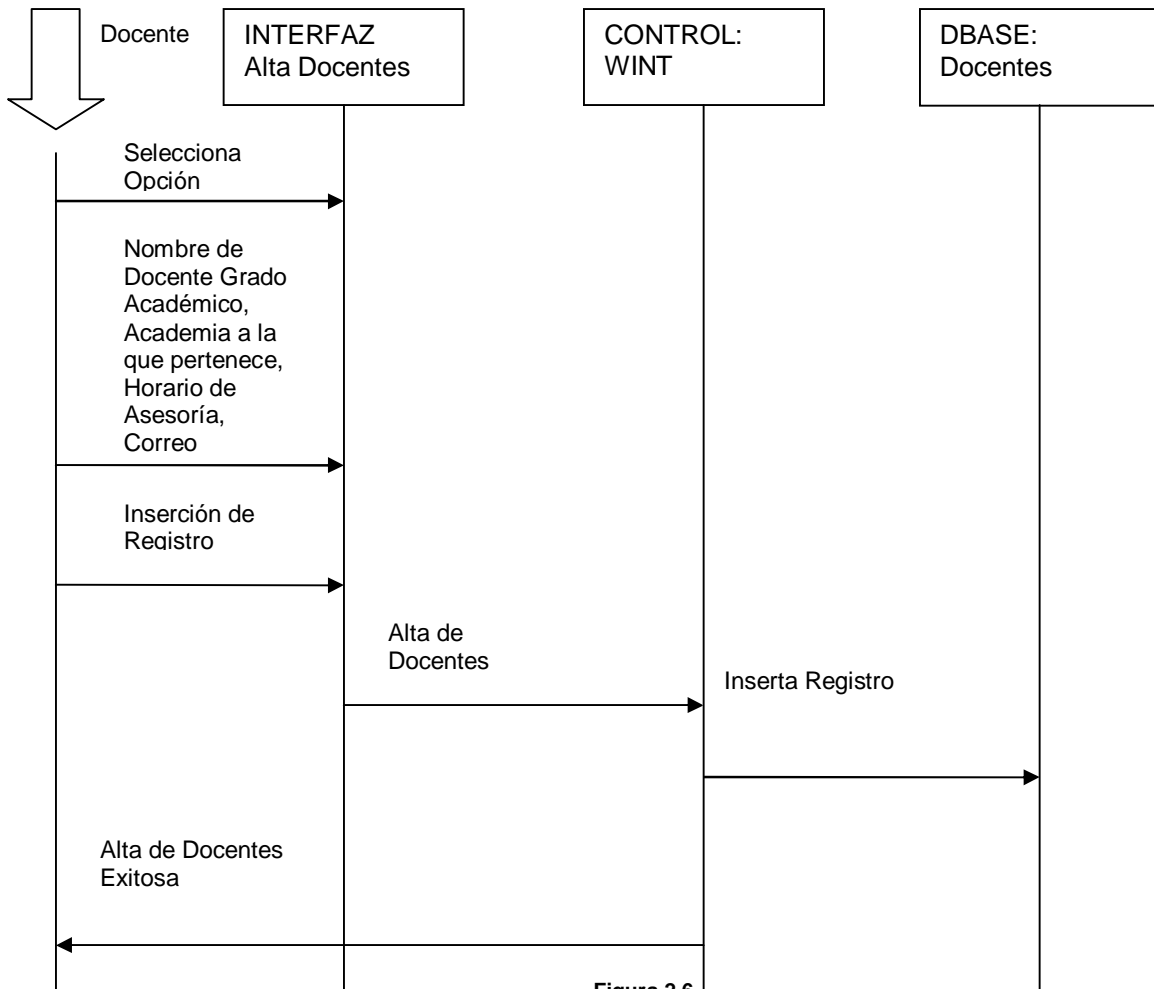


Figura 2.6  
Alta de Docentes

### 2.4.6 DIAGRAMA DE SECUENCIA CONSULTA INDICE GENERAL

El flujo de la consulta por índice general está dado por el AUTOR que ahora en este caso es el Administrativo la INTERFAZ es Índice General, CONTROL es WINT y DBASE es Equipos.

Aparecerá una página dirigida al administrativo en la cual podrá hacer consulta de equipos desde la base de datos Equipos, en donde solo el usuario seleccionara la opción de consulta por índice general y aparecerá un listado de todos los equipos, con cada uno de su información adicional.

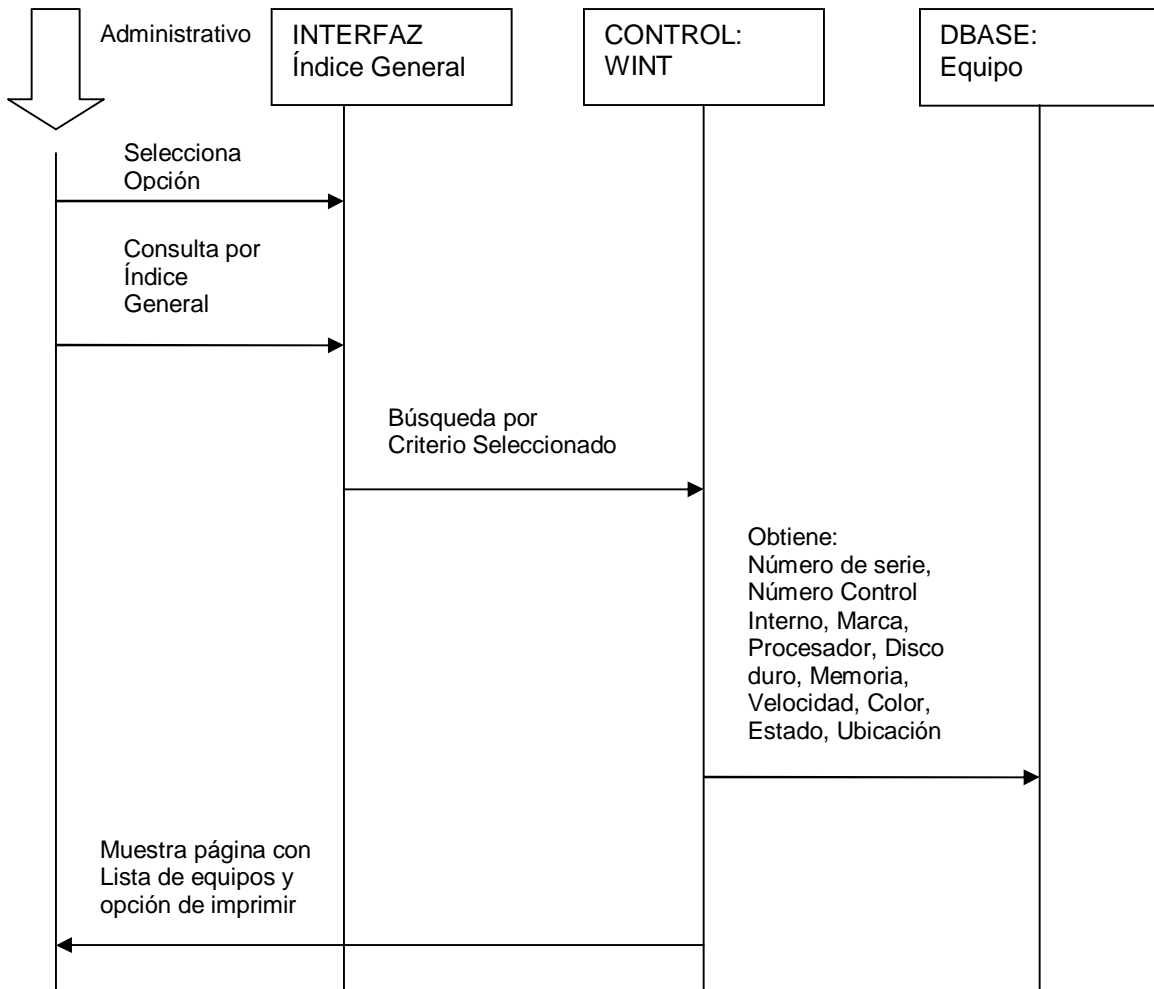
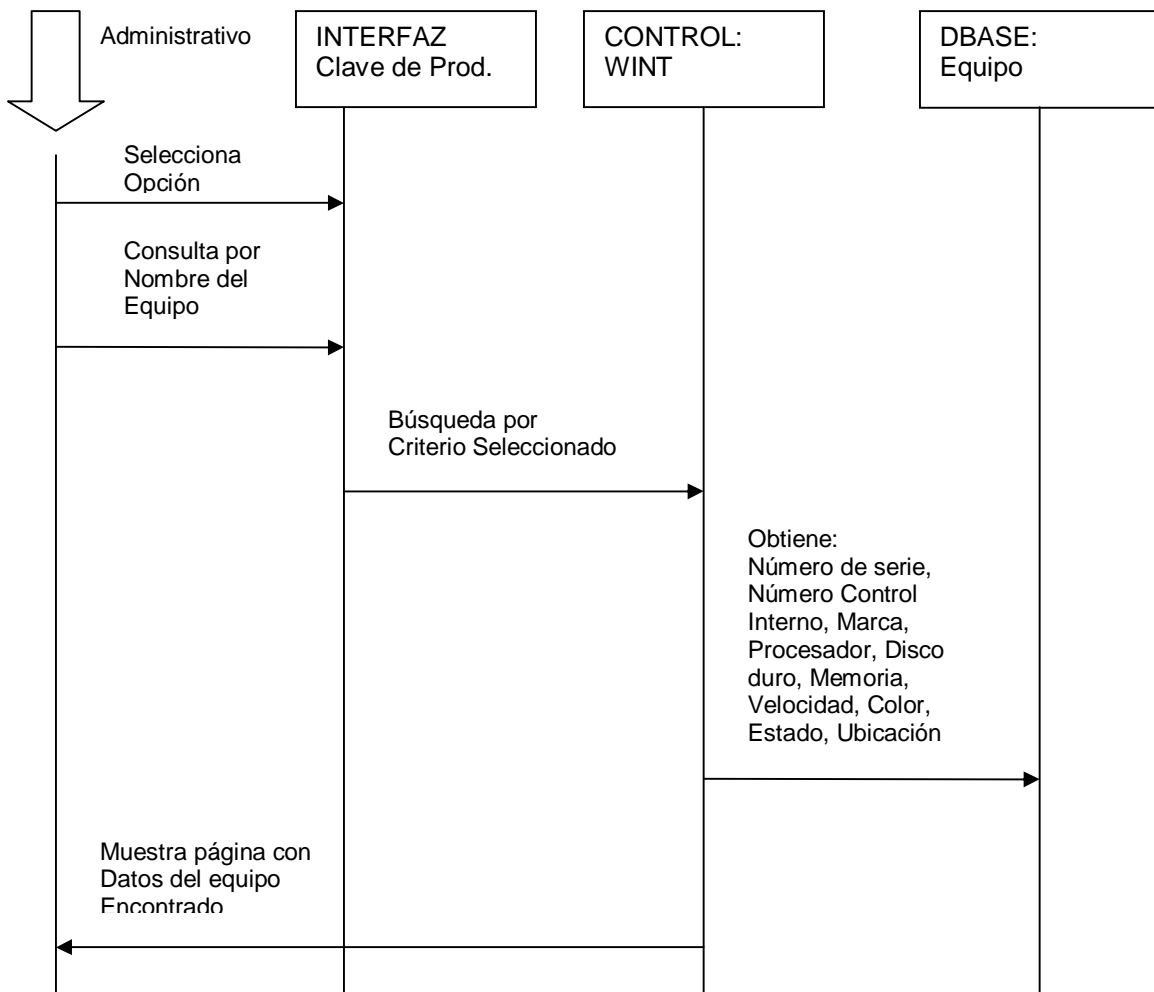


Figura 2.7  
Consulta Índice General

### 2.4.7 DIAGRAMA DE SECUENCIA CONSULTA POR NOMBRE DE EQUIPO

El flujo de la consulta por nombre de equipo está dado por el AUTOR que será el Administrativo, la INTERFAZ es Nombre del Equipo CONTROL es WINT y DBASE es Equipo.

Se mostrará una página que el Administrativo pueda manipular una búsqueda de algún producto en específico insertando como criterio de búsqueda el nombre del equipo para que despliegue los atributos faltantes del equipo como son Número de serie, Número Control Interno, Marca, Procesador, Disco duro, Memoria, Velocidad, Color, Estado, Ubicación.



**Figura 2.8**  
Consulta Nombre del Equipo

## 2.5 CLASES DE PRESENTACION Y BASES DE DATOS

Inicialmente se procederá a tratar de localizar los actores que se encuentren involucrados en el sistema; como es claro y a simple vista los actores son el Administrador y el Cliente, a su vez se pretende encontrar a los sustantivos del sistema que son los datos que se ocupen para así poder generar formularios, y distribución de pantallas o posibles menús para el sistema.

ACTORES	CASOS DE USO					
	1	2	3	4	5	6
Docente				•	•	•
Administrativo	•	•	•			

SUSTANTIVOS	CASOS DE USO					
	1	2	3	4	5	6
Dirección WEB	•	•	•	•	•	•
Nombre de Usuario	•	•	•		•	•
Opción Alta de Equipos	•					
Nombre de Equipo	•	•	•		•	•
Descripción del Producto	•		•		•	•
Ubicación	•		•		•	•
Opción Baja de Equipo		•				
Opción Modificación de Equipo			•			
Alta de Docentes				•		
Nombre del Docente				•		
Grado Académico				•		
Academia a la que pertenece				•		
Horario de Asesoría				•		
Correo Electrónico				•		
Usuario Nuevo				•		
Opción Consulta Índice					•	
Opción Consulta Nombre del Equipo						•
Opción Consulta de Docentes						

Figura 2.9 Clases de Presentación y Bases de Datos

Una vez recolectados los Actores y Sustantivos mediante los Casos de Uso del sistema se encuentran con las siguientes agrupaciones generando así una vista preliminar de las posibles Bases de Datos

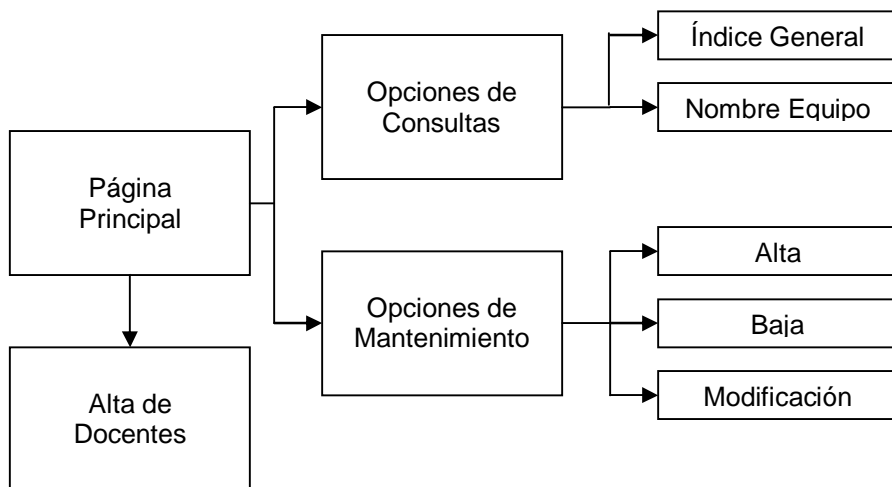
**Equipos**

- Nombre del Equipo
- Número de Serie
- Número de Control Interno
- Marca
- Procesador
- Disco Duro
- Memoria
- Velocidad Herz
- Color
- Estado
- Ubicación

**Datos de Docentes**

- Nombre del Docente
- Grado Académico
- Horario de Asesoría
- Correo Electrónico

Y con respecto a la organización del sistema se sigue que se implementen opciones para las búsquedas y el mantenimiento



**Figura 2.10 Constitución del Sistema**

Cabe mencionar que cada caso de uso es necesario tener notificaciones del Sistema de acción exitosa o fallida.

### 3. DISEÑO.

#### 3.1 ARQUITECTURA DEL SISTEMA

La arquitectura que se usara para el sistema será cliente-servidor, donde el lado del cliente se estará ejecutando un browser o navegador HTML el cual constará de una interfaz gráfica y será el encargado de mostrar todas las pantallas con las cuales el usuario tendrá contacto con el sistema para su manejo. Cabe mencionar que el browser se encargará de todas las peticiones necesarias hacia el servidor.

Así mismo del lado del servidor se encontrará un Servidor Web el cual se encargará de atender las peticiones del cliente y éste a su vez tendrá acceso a las bases de datos analizando así el Servidor Web las peticiones de páginas y consultas de datos que se reciban.

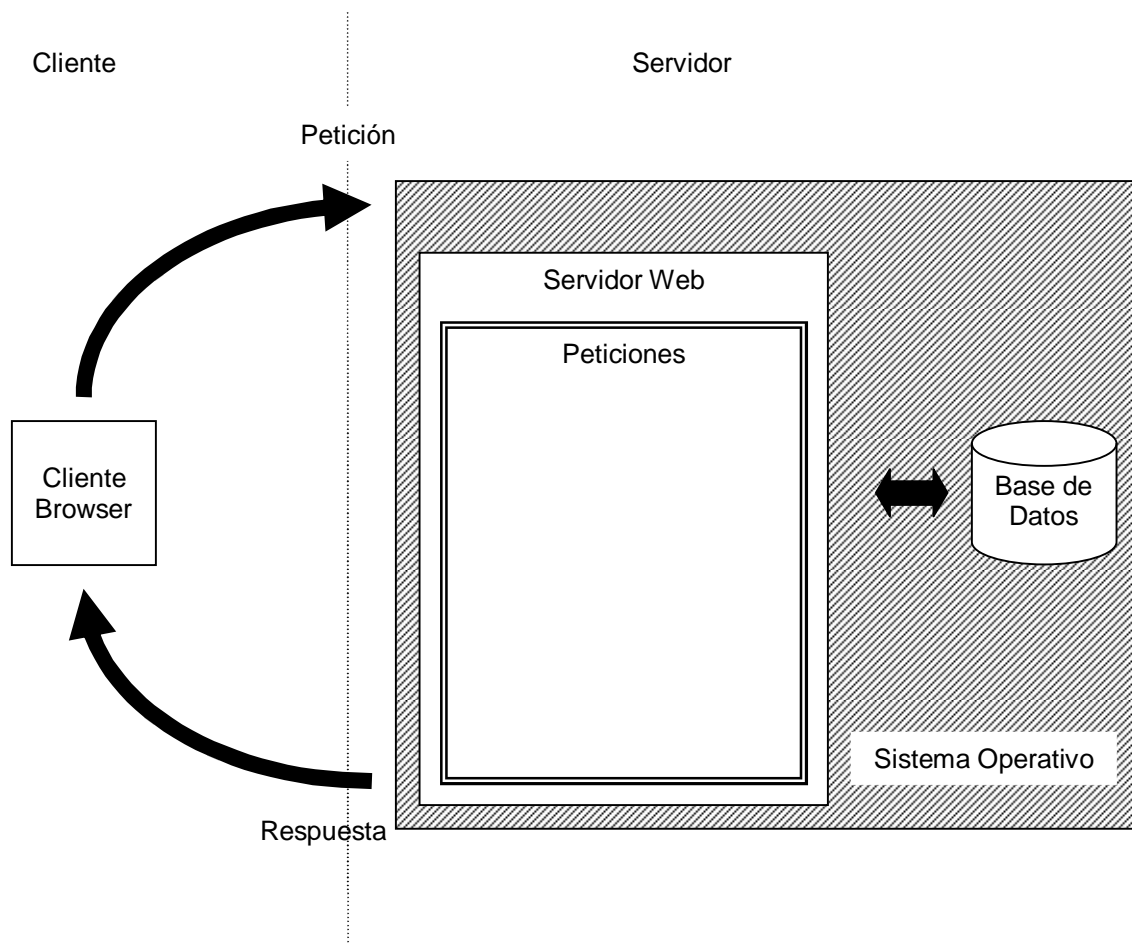
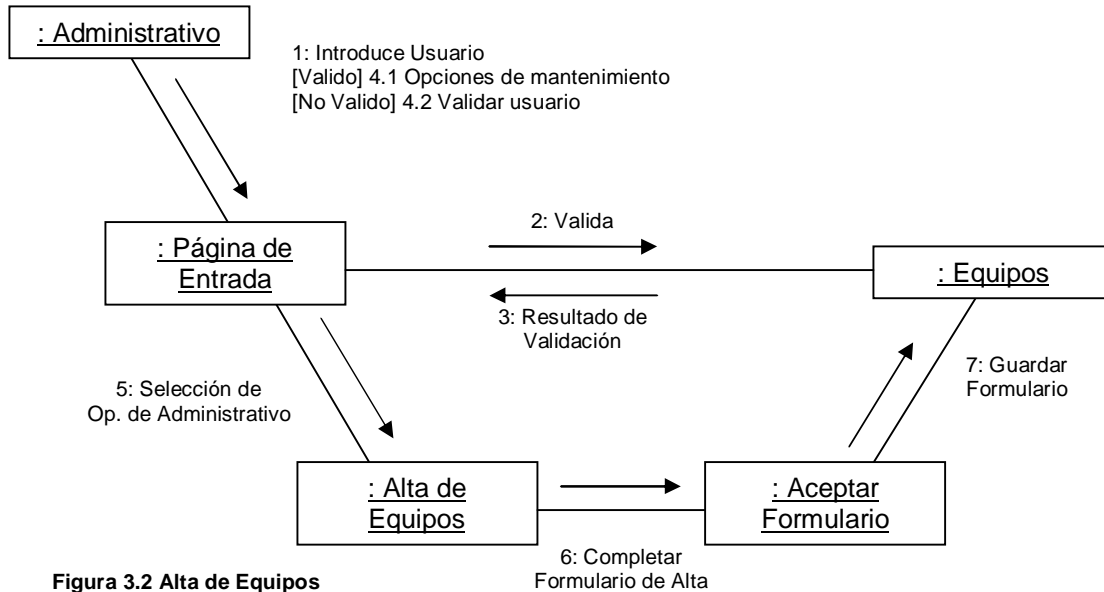


Figura 3.1 Esquema General del Sistema

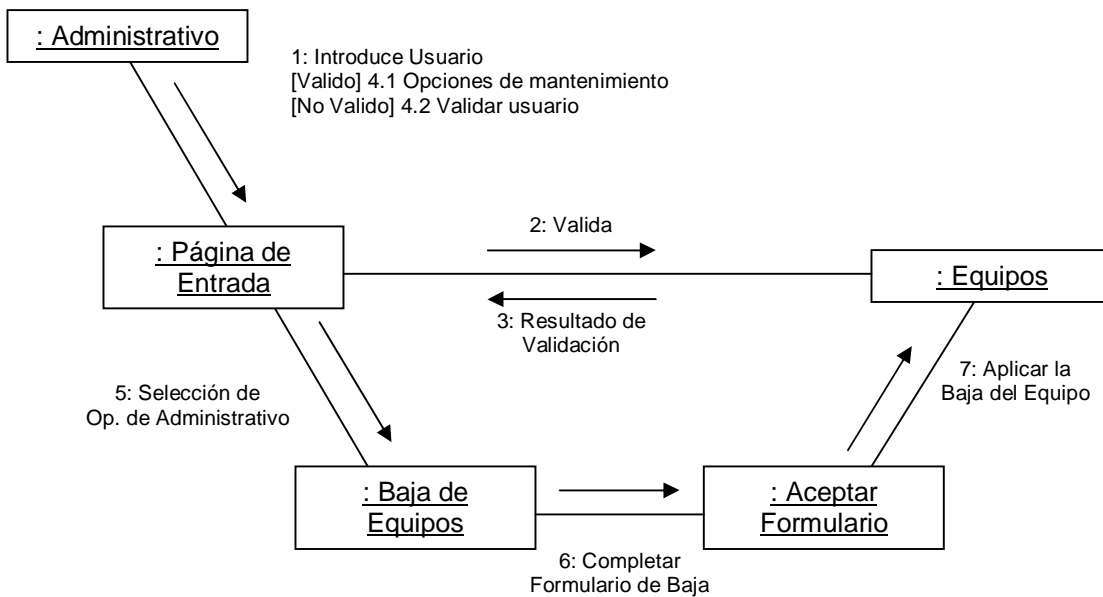
### 3.2 DIAGRAMAS DE COLABORACIONES

Aquí se tratará de mostrar como los objetos colaboran entre si, se muestran los objetos junto con los mensajes que se envían entre ellos destacando la sucesión de las interacciones, así como el contexto y organización general de los objetos que interactúan, organizando dicho diagrama de acuerdo al espacio.

#### 3.2.1 DIAGRAMA DE COLABORACIONES PARA MANTENIMIENTO DE ALTA DE EQUIPOS



#### 3.2.2 DIAGRAMA DE COLABORACIONES PARA MANTENIMIENTO DE BAJA DE EQUIPOS



### 3.2.3 DIAGRAMA DE COLABORACIONES PARA MODIFICACION DE EQUIPOS

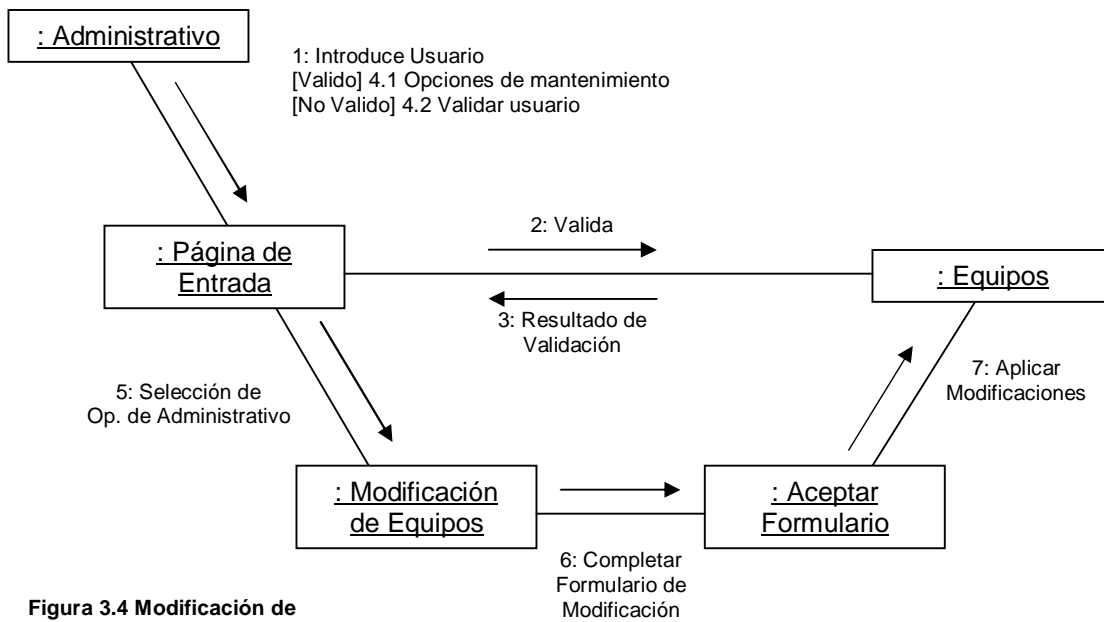


Figura 3.4 Modificación de Equipos

### 3.2.4 DIAGRAMA DE COLABORACIONES PARA ALTA DE DOCENTES

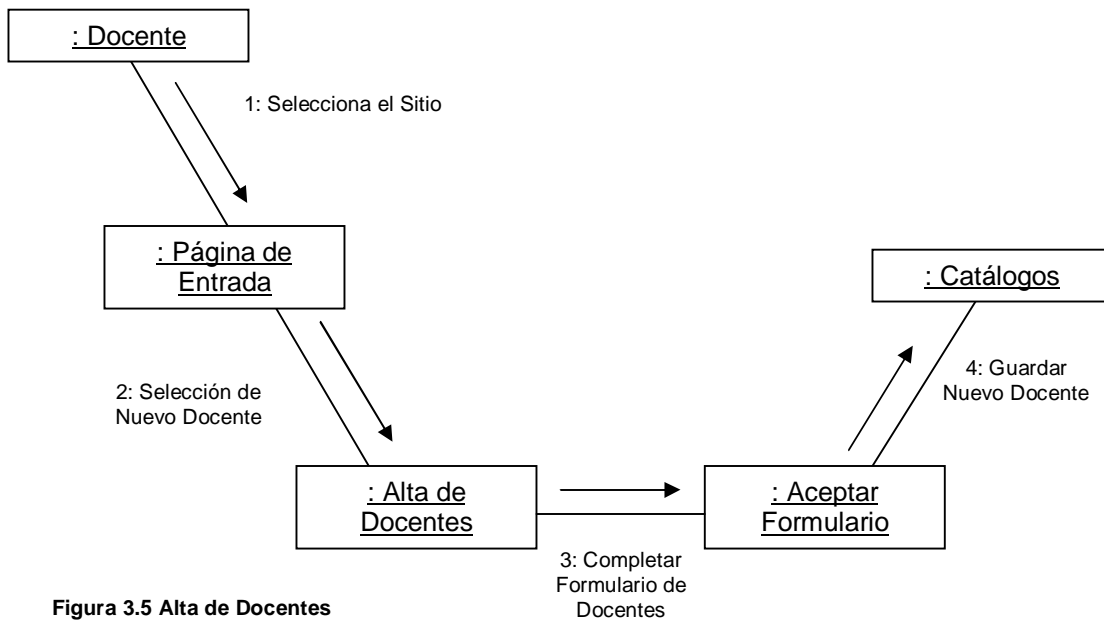


Figura 3.5 Alta de Docentes

### 3.2.5 DIAGRAMA DE COLABORACIONES PARA CONSULTA POR INDICE GRAL.

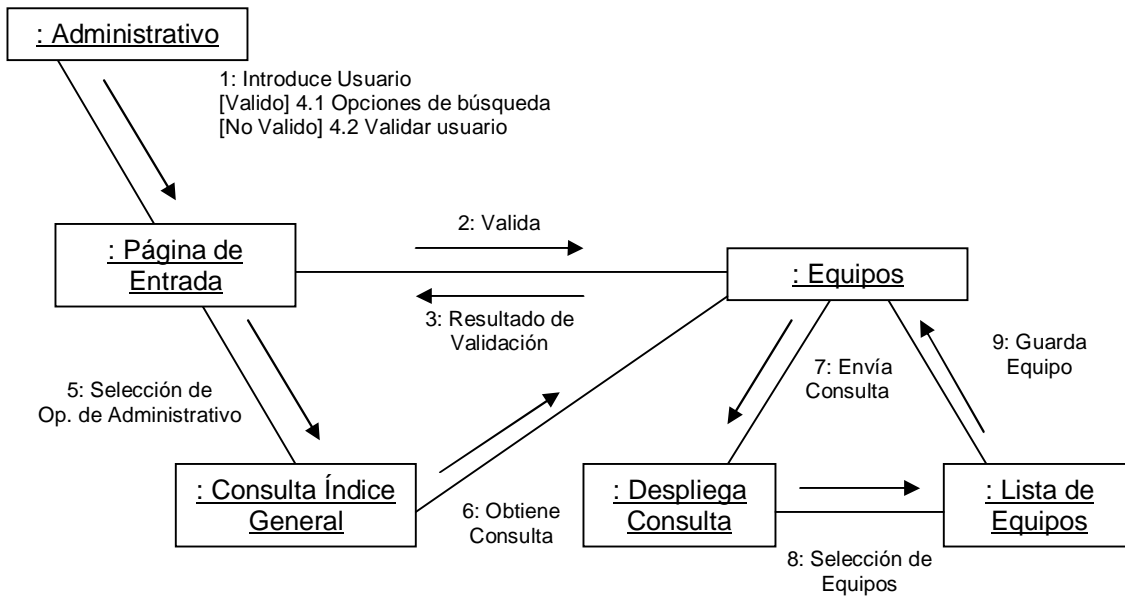


Figura 3.6 Consulta Índice General

### 3.2.6 DIAGRAMA DE COLABORACIONES PARA CONSULTA POR NOMBRE DEL EQUIPO

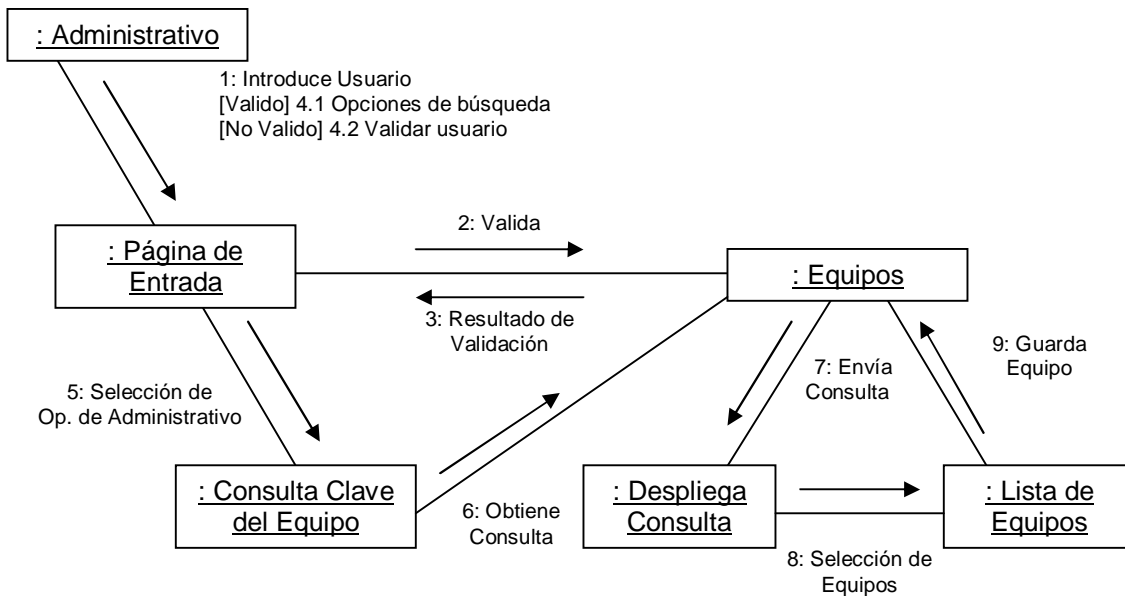


Figura 3.7 Consulta Clave del Equipo

### 3.3 DIAGRAMAS DE ACTIVIDADES

Los diagramas de actividades sirven para mostrar una visión simplificada de lo que ocurre durante una operación o proceso con el fin de expresar los puntos de decisión y bifurcaciones. El procesamiento dentro de una actividad se lleva a cabo y al realizarse se continúa con la siguiente actividad.

#### 3.3.1 DIAGRAMA DE ACTIVIDADES PARA ALTA DE EQUIPOS

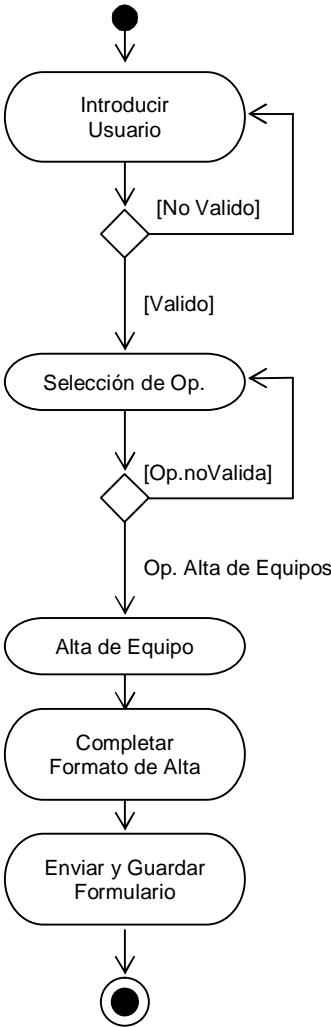


Figura 3.8 Alta de un Equipo

#### 3.3.2 DIAGRAMA DE ACTIVIDADES PARA BAJA DE EQUIPOS

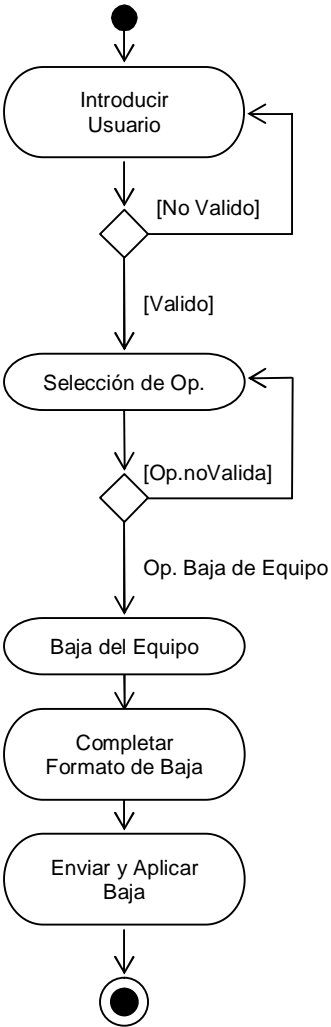
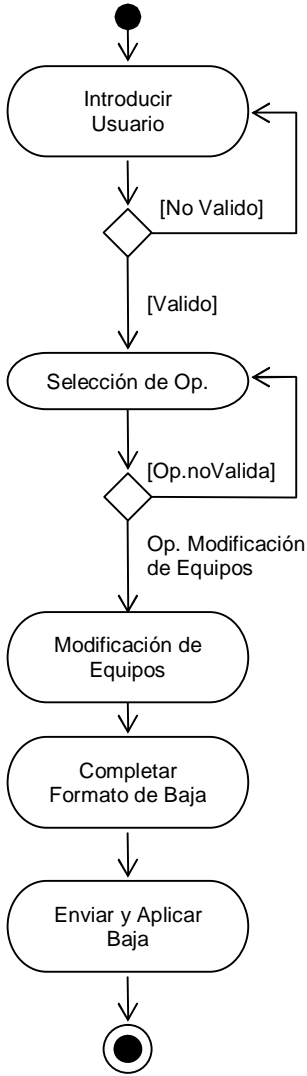


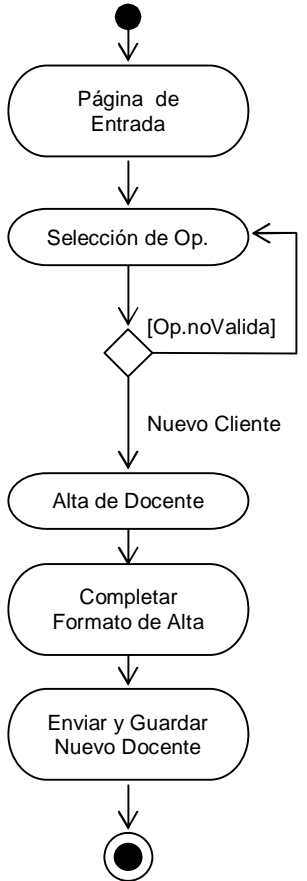
Figura 3.9 Baja de Equipo

**3.3.3 DIAGRAMA DE ACTIVIDADES PARA MODIFICACION DE EQUIPOS**



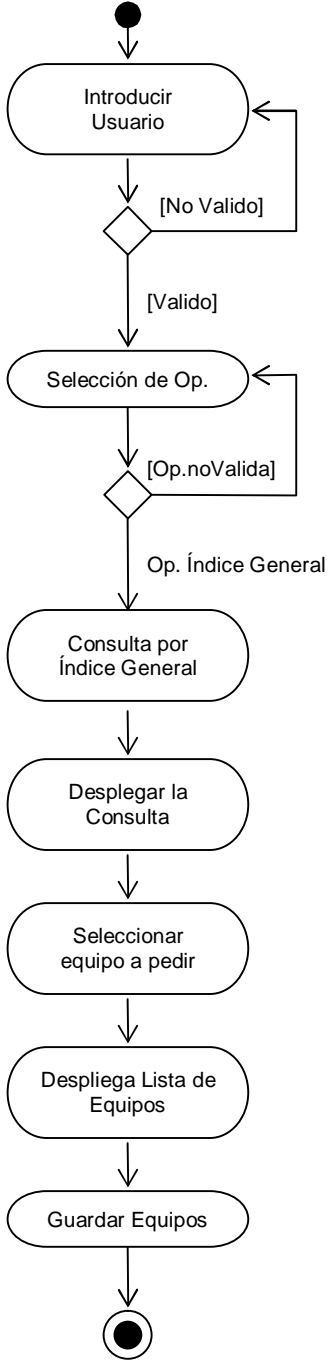
**Figura 3.10**  
**Modificación de Equipos**

**3.3.4 DIAGRAMA DE ACTIVIDADES PARA ALTA DE DOCENTES**



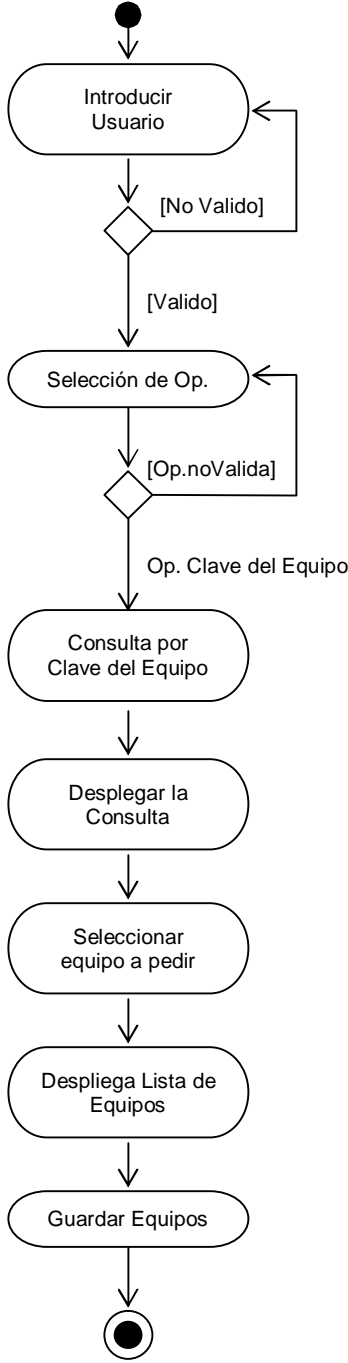
**Figura 3.11**  
**Alta de Docentes**

**3.3.5 DIAGRAMA DE ACTIVIDADES PARA CONSULTA POR INDICE GENERAL**



**Figura 3.12**  
Consulta Índice General

**3.3.6 DIAGRAMA DE ACTIVIDADES PARA CONSULTA POR NOMBRE DEL EQUIPO**



**Figura 3.13**  
Consulta Clave del Equipo

**3.4 DIAGRAMA DEL MODELO E-R DEL SISTEMA**

Respondiendo al modelo Entidad-Relación se muestra la estructura lógica de las bases de datos expresándola gráficamente, donde se pueden encontrar entre otros, las entidades que son Equipos, Datos del Docente, y respectivamente sus atributos y las relaciones entre conjuntos de entidades

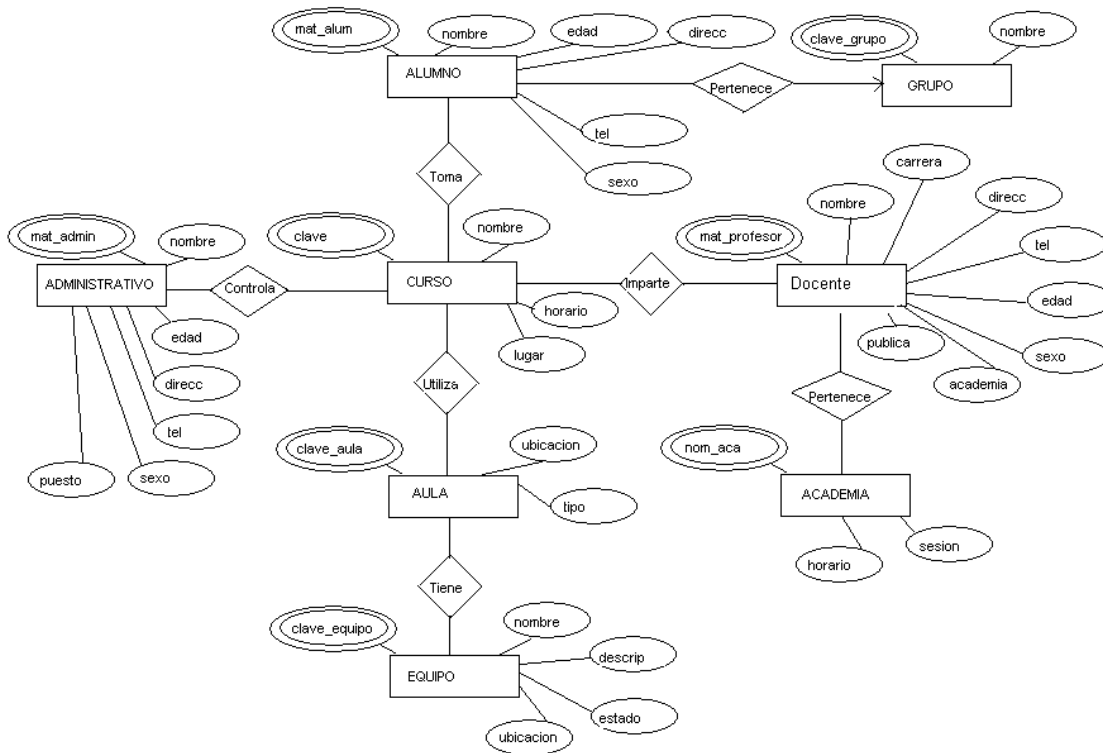


Figura 3.12 Modelo Entidad Relación del Sistema

### 3.5 DICCIONARIO DE DATOS

Se muestran los diccionarios de los datos para las bases, en donde se muestra el tipo de dato y su longitud, así como las llaves de cada tabla y si pueden ser nulas durante la captura es decir datos que no alteren de manera significativa el proceso de la información así como los campos que será utilizados en las bases de datos y una descripción de lo que representaran.

<b>EQUIPOS</b>				
<b>CAMPO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>NULO</b>	<b>LLAVE</b>
Numserie	Número de Serie del Equipo	Varchar(20)	No	Si
Nom	Nombre del Equipo	Char(10)	No	
Numprep	Número Control Interno	Varchar(20)	Si	
Mca	Marca	Char(10)	Si	
Proc	Procesador	Varchar(10)	Si	
Hd	Disco Duro	Varchar(6)	Si	
Mem	Memoria RAM	Varchar(7)	Si	
Herz	Velocidad Herz	Varchar(7)	Si	
Color	Color	Char(20)	Si	
Edo	Estado	Char(8)	Si	
Ubica	Ubicación	Varchar(15)	Si	

### 3.6 NORMALIZACION

Tomando en cuenta las tablas anteriores para el sistema y revisándolas conjuntamente con las reglas de normalización de las tablas para bases de datos relacionales se encuentra que la primera tabla incorpora una dependencia transitiva la cual puede dar problemas de inserción y actualización, para evitar esos conflictos a la hora de manipular dicha tabla se propone que la tabla se subdivida de la siguiente forma:

<b>EQUIPOS</b>				
<b>CAMPO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>NULO</b>	<b>LLAVE</b>
Numserie	Número de Serie del Equipo	Varchar(20)	No	Si
Nom	Nombre del Equipo	Char(10)	No	
Numprep	Número Control Interno	Varchar(20)	Si	
Mca	Marca	Char(10)	Si	
Color	Color	Char(20)	Si	
Edo	Estado	Char(8)	Si	
Ubica	Ubicación	Varchar(15)	Si	

<b>MANEJA EQUIPO</b>				
<b>CAMPO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>NULO</b>	<b>LLAVE</b>
Numserie	Número de Serie del Equipo	Varchar(20)	No	Si
Proc	Procesador	Varchar(10)	Si	
Hd	Disco Duro	Varchar(6)	Si	
Mem	Memoria RAM	Varchar(7)	Si	
Herz	Velocidad Herz	Varchar(7)	Si	

### 3.7 RELACIONES DE TABLAS

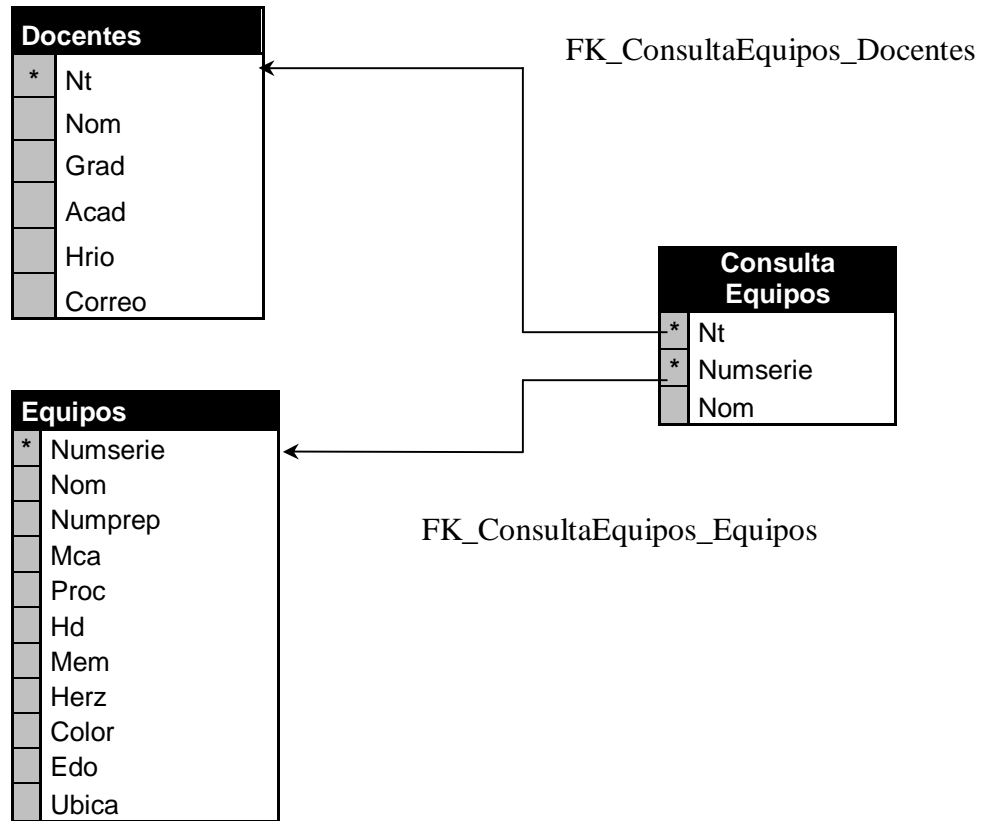


Figura 3.13 Relación de Tablas

## 4. IMPLEMENTACION

### 4.1 AMBIENTE DE DESAROLLO

Con lo que respecta al ambiente de desarrollo en la parte del Servidor se utilizaran los siguientes componentes: Apache Server, que es la plataforma de implementación la cual provee de un servicios denominado Internet Information Server (IIS) que permite compartir fácilmente documentos e información a través de una intranet de una compañía o de Internet. Con IIS, es posible distribuir aplicaciones confiables basadas en Web e incorporar aplicaciones y datos existentes al Web. Otra característica de IIS, son las Páginas Active Server.

Las Páginas Active Server (ASP) son un entorno de secuencias de comandos del lado del servidor que permite crear aplicaciones de servidor Web dinámicas e interactivas. Con ASP, se puede combinar páginas HTML, secuencias de comandos y componentes COM+ como sea necesario para crear aplicaciones basadas en Web flexibles y eficaces.

Con los servicios de componentes COM+ se pueden distribuir y administrar aplicaciones a través de una interfaz gráfica de usuario o automatizar tareas administrativas con un lenguaje de secuencias de comandos o de programación.

A su vez durante la programación en ASP se auxilia de JavaScript que es un lenguaje de alto nivel basado en objetos para la generación de documentos Web dinámicos esto para realizar las interacciones entre el usuario y las páginas HTML

La herramienta encargada de la base de datos es MySQL phpMyAdmin. Este motor de base de datos cuenta con la característica de operar la componente de almacenamiento de datos de los sitios Web. El modelo de programación está integrado con ActiveX Data Objects (ADO). para desarrollar aplicaciones Web.

Del lado del Cliente se utiliza como aplicación un browser o navegador que pueda desplegar código HTML para poder acceder y recuperar las páginas expuestas por el servidor para este fin se utiliza MS Internet Explorer y Mozilla Firefox soportando el lenguaje JavaScrip para ambas aplicaciones.

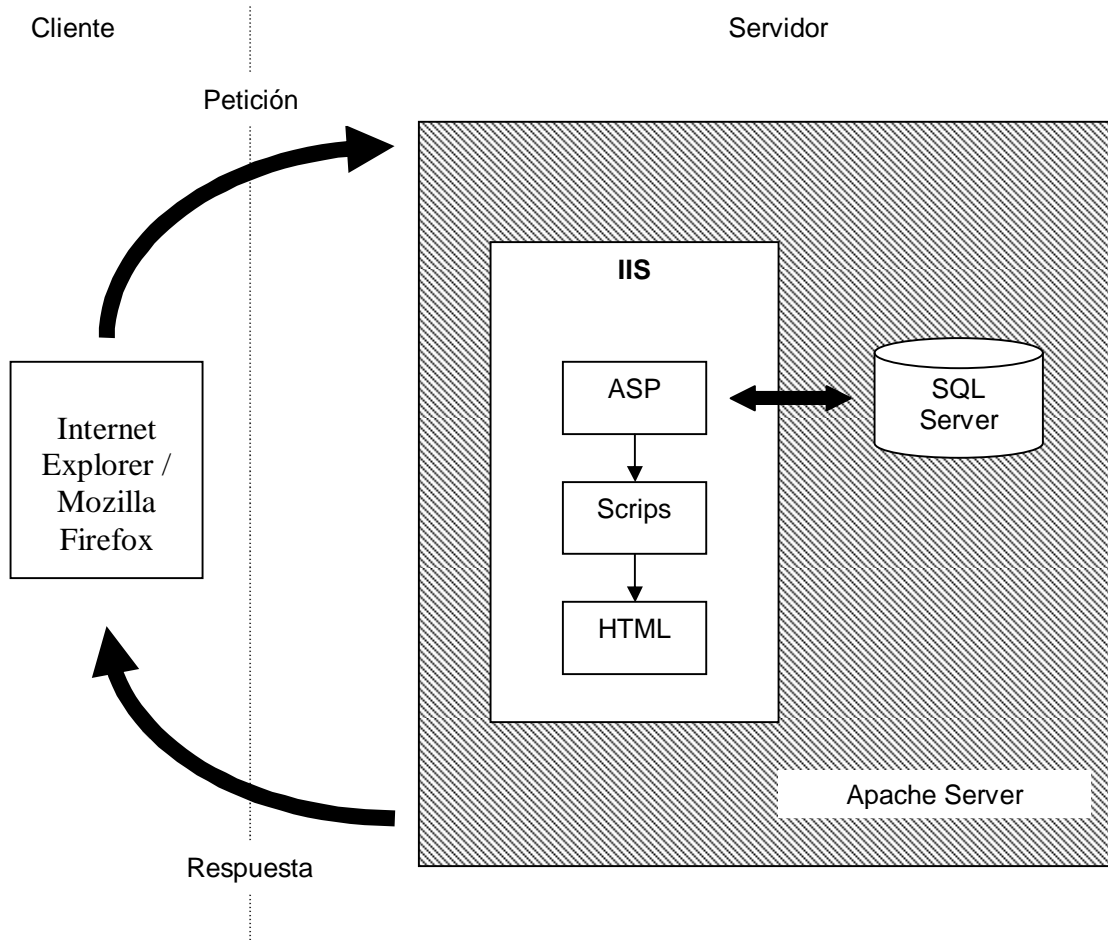


Figura 4.1 Esquema de Ambiente de Desarrollo

## 4.2 CREACION DE LA BASE DE DATOS

Una vez descrito el Ambiente de desarrollo se procedió a la creación de una nueva base de datos en donde se especifica el nombre de la base de datos, el tamaño máximo de la base de datos, el nombre de los archivos de datos y el nombre de registro de transacciones, posteriormente se crearon las tablas de la base de datos, la cual se puede realizar mediante dos formas la primera es utilizando el Administrador Corporativo de MySQL Server en donde se pueden crear las tablas de forma gráfica y la segunda es mediante el Analizador de Consultas de SQL Server 2000 y consiste en escribir las instrucciones del lenguaje MySQL para crear así las tablas. Una vez hecha la base de datos y las tablas fue necesario crear permisos y privilegios de usuario.

Para este capítulo se mostrarán las pruebas del Sistema de Pedidos en donde se mostrará su funcionamiento y los resultados que devuelve el sistema, para ello se empezará a probar mediante las acciones que pueda realizar un cliente.

### 4.3 PANTALLA DE ENTRADA

En esta pantalla se muestra una presentación del sistema como bienvenida y en su frame izquierdo se despliega el lugar donde nos encontramos, un menú el cual tiene las opciones de Acerca de la Preparatoria Historia, Plan de Estudios, Talleres, Tutores y Academias.

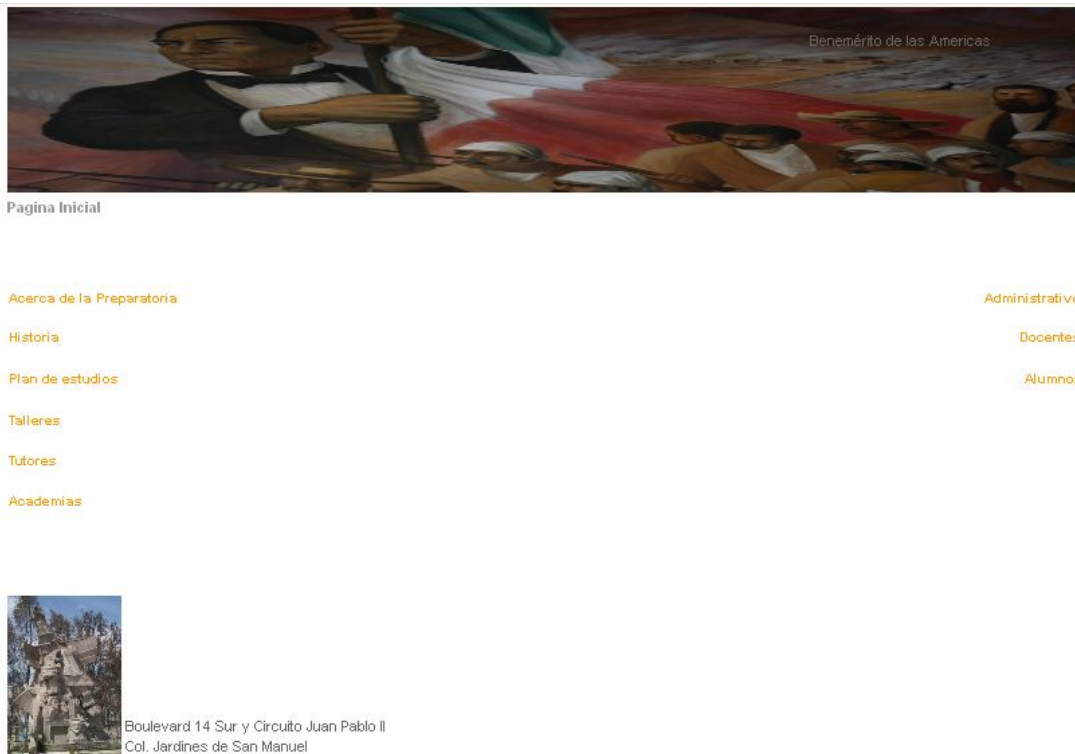


Figura 4.2 Pantalla de Entrada

#### 4.4 PANTALLA DE ACCESO AL SISTEMA

El Número de trabajador es la validación que hace el sistema, la identificación del usuario ya registrado en caso de que no sea posible identificarse el sistema le dará la opción de registrarse.

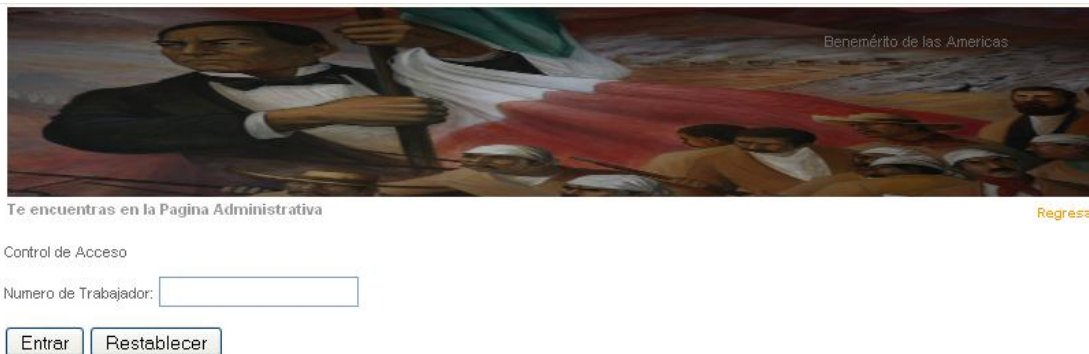


Figura 4.3 Acceso al Sistema

En caso de que el usuario intente entrar al sistema y no se valide su clave le notificara que no existe el usuario y le da la opción de registrarse al sistema.

#### 4.5 PANTALLA DE ALTA DE DOCENTES

En Esta pantalla podrá dar de Alta Docentes mediante el menú en el frame izquierdo aparecerá la siguiente pantalla, si el usuario no llena los campos.

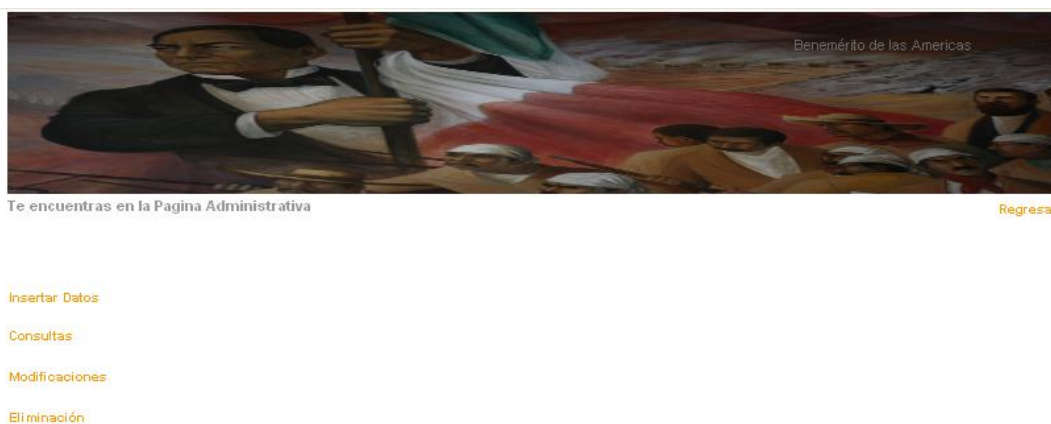



Figura 4.4 Primera Pantalla Alta de Docentes



Te encuentras en la Pagina Docentes [Regresar](#)

Datos de Docente

Nombre:

Grado Academico:

Academia a la que pertenece:


Horario de Asesoría:

Correo Electronico:

Figura 4.5 Pantalla Segunda Alta de Docentes

#### 4.6 PANTALLA CONSULTAS DE DOCENTES

Si el usuario es validado de manera correcta aparecerá un menú de consultas para poder hacer los pedidos así como a continuación se muestra.



Te encuentras en la Pagina Docentes [Regresar](#)

Control de Acceso

Numero de Trabajador:

[Lista de Maestros](#)

[Cursos](#)

[Mandar Correo](#)

Figura 4.6 Consultas de Docentes

#### 4.7 PANTALLA ALTA DE EQUIPOS

Aquí se insertará un equipo mas a la base de datos Equipos, una vez completado el formulario, éste será guardado.

Benemérito de las Americas

Regresar

Te encuentras en la Pagina Administrativa

Datos de Equipo " COMPUTADORA CPU "

Nombre:

Numero de Serie:

Numero Control Interno:

Marca:

Procesador:

Disco Duro:

Memoria RAM:

Velocidad Herz:

Color:

Estado:

Ubicacion:

Aceptar Restablecer

Figura 4.7 Alta de Equipos

#### 4.8 PANTALLA ACCESO ALUMNOS

Se muestra la pantalla en la cual podrán registrarse para poder acceder a las pantallas de sitios de Interés y también podrán mandar correo acerca de alguna Tarea a entregar.

Benemérito de las Americas

Regresar

Te encuentras en la Pagina Alumnos

Datos de Alumno

Nombre:

Matricula:

Fecha de Ingreso:

Ligas de Interes

Mandar Correo o Tarea

Figura 4.8 Acceso Alumnos

#### 4.9 PANTALLA SITIOS DE INTERES PARA ALUMNOS

Se muestra la pantalla de servicios para los alumnos de la Preparatoria, con ella podrán acceder al sitio de su elección para realizar algún trámite necesario.

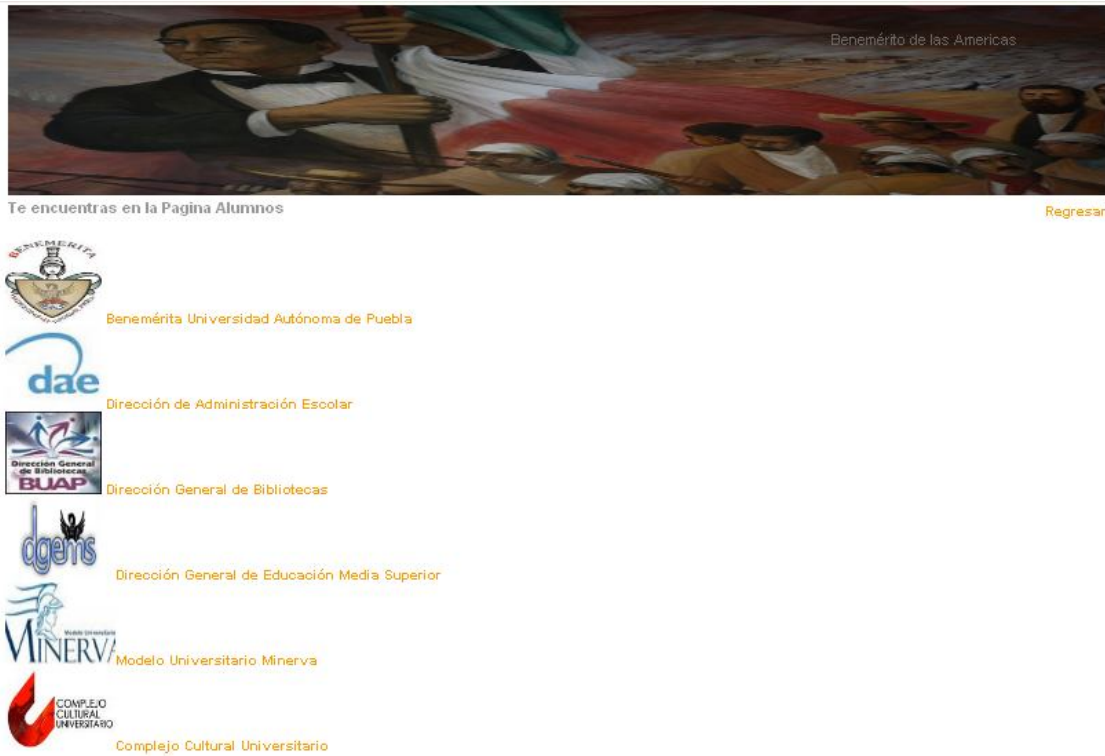


Figura 4.9 Sitios de Interés

### 4.10 PANTALLA BASE DE DATOS BJ

Se muestra la pantalla de la base de datos de nuestro sistema en la cual se pueden observar las tablas que componen dicha base de datos.

The screenshot shows the phpMyAdmin interface for a database named 'bj'. The left sidebar lists the database and its tables. The main area displays a table of database information with the following data:

Tabla	Acción	Registros	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> academias	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> administrativo	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> alumnos	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> cursos	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> docentes	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> equipo	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> personal	[Icons]	1	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> preparatoria	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<input type="checkbox"/> salones	[Icons]	0	InnoDB	latin1_swedish_ci	16.0 KB	-
<b>9 tabla(s)</b>	<b>Número de filas</b>	<b>1</b>	<b>MyISAM</b>	<b>latin1_swedish_ci</b>	<b>144.0 KB</b>	<b>0 Bytes</b>

Below the table, there are controls for 'Marcar todos/as / Desmarcar todos' and 'Para los elementos que están marcados:'. At the bottom, there is a section for 'Crear nueva tabla en la base de datos bj' with input fields for 'Nombre:' and 'Número de campos:', and a 'Continuar' button.

Figura 4.10 Base de Datos Bj

## CONCLUSIONES

En el presente trabajo de tesis se puede ultimar que se alcanzo el objetivo general que fue desarrollar una aplicación Web con conexión a una Base de Datos y también los objetivos específicos, en donde se logró poder consultar los equipos, cursos, talleres, docentes, alumnos, salones y academias. Se desarrollo un control de los recursos físicos con los que cuenta la Preparatoria en el cual se puede hacer operaciones de mantenimiento y recuperación, vinculando de esta forma una base de datos en el Web.

Todo esto se pudo crear gracias a la tecnología de PHP y al uso de HTML, del lenguaje Java Script, Lo cual primeramente fue necesario aprender a utilizar cada una de estas herramientas para poder realizar este proyecto, más aun el saber utilizar MySQL para la creación de la Base de Datos.

En lo particular este trabajo de tesis me dejo un buen conocimiento de cómo funciona principalmente Internet, los sitios Web como se implementan tanto los estáticos como los dinámicos, el saber construir una base de datos que no tenga problemas en su funcionamiento a lo que se refiere mantenimiento y recuperación, también logré aprender a realizar el modelado de sistemas mediante UML, pues es una gran ayuda porque es flexible y tiene opción de corregir errores durante la marcha, también fue de una gran ayuda el tener conocimiento del lenguaje MySQL que independientemente de la ayuda en el Web, tiene muchas aplicaciones más.

## **PERSPECTIVAS**

En este sistema se podrán hacer mejoras tanto en su interfaz como en la implementación de nuevos módulos que puedan enriquecer y facilitar su uso, también se puede hacer la implementación de una seguridad mayor en aplicaciones Web y ofrecer la publicación de artículos personales de los docentes en línea para brindar una enseñanza con excelencia y compartirla a público en general.

## ANEXO A

### CODIGO FUENTE DEL SISTEMA

#### ALTA DE DOCENTES

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Alta Docentes</title>
</head>

<body>
DATOS
<form id="form2" name="form2" method="post" action="inserta.php">

  <p>
    <label></label>
    <label>Nombre
    <input name="nom" type="text" id="nom" />
    </label>
  </p>
  <p>
    <label>Grado Academico
    <input name="grad" type="text" id="grad" />
    </label>
  </p>
  <p>
    <label>Academia a la que pertenece
    <input name="acad" type="text" id="acad" />
    </label>
  </p>
  <p>
    <label>Horario Asesoría
    <input name="hrio" type="text" id="hrio" />
    </label>
    <label></label>
  </p>
  <p>
    <label>Correo Electronico
    <input name="correo" type="text" id="correo" />
    </label>
  </p>

  <p>
    <label></label>
  </p>

  <p>
    <label>
    <input type="submit" name="Submit" value="Enviar" />
    </label>
  </p>
  <p>
  <label>
  <INPUT TYPE="Reset" VALUE="Restablecer">

</label>
</form>
</body>
</html>

```

## CONSULTA POR NOMBRE DE EQUIPO

```

<html>
<head>
    <title>
        Consulta por nombre
    </title>
</head>
<body>
<?php
    $link = mysql_connect ("localhost", "root", "password");
    mysql_select_db("bj",$link);
    $result =mysql_query("SELECT * FROM equipo", $link);
?>
<center>
<?
    $rows=mysql_num_rows($result);
    $i =0;
    while ($i < $rows){
?>
        <table width="314">
            <tr>
                <td width="129">
                    Numero de Serie:
                </td>
                <td width="173"> <?php
                    echo mysql_result ($result, $i, "numserie") . "<br>"; ?>
            </td>
            </tr>
            <tr>
                <td>
                    Nombre:
                </td>
                <td><?php
                    echo mysql_result($result, $i, "nom")."<br>"; ?>
                </td>
            </tr>
            <tr>
                <td>
                    Descripcion:
                </td>
                <td><?php
                    echo mysql_result($result, $i, "desc")."<br>"; ?>
                </td>
            </tr>
            <tr>
                <td>
                    Estado:
                </td>
                <td><?php
                    echo mysql_result($result, $i, "edo")."<br>"; ?>
                </td>
            </tr>
        </table>
        <br />
<?
    $i++;
}
?>
<form name="form1" method="post" action="busqueda.html">
    <label>
        <input type="submit" name="Submit" value="Regresar">
    </label>
</form>
</center>
</body>
</html>

```

## BIBLIOGRAFIA Y REFERENCIAS

- Aprendiendo My SQL, Mark Masla Kowski, Prentice Hall 1999
  - Diseño de Sistemas de Información, Burch, Grudnitski, Limusa 1998
  - Fundamentos de Bases de Datos, Silberschatz, Korth, Sudarshan, Mc. GrawHill 1998
  - HTML Dinámico ASP y JavaScript a través de ejemplos, Bobadilla, Alcocer, Alonso, Gutiérrez, Alfaomega 2001
  - Kit de Recursos de Intranet, Prakash Ambegaonkar, Mc. GrawHill 2000
  - Organización de las bases de datos, James Martin, Prentice Hall 1999
  - Programación en Java, Nathan Meyers, Prentice Hall 2001
  - Web DataBase Primer Plus, Piroz Mohseni, Waite Group Press. 2000
- 
- <http://www.aspespanol.com/tut-sql/index.html>
  - <http://www.asptutor.com/asp/vart.asp?id=112>
  - <http://webdia.cem.itesm.mx/ac/rtrejo/Interfaz/hipertex.html>
  - <http://www.dcc.uchile.cl/~psalinas/uml/introduccion.html>
  - <http://www.desarrolloweb.com>
  - <http://www.forsdelweb.com/f18/tablas-dinamicas-399882/>
  - <http://www.isc.uniandes.edu.co/~revista/articulos/cliser.html>
  - <http://www.lavariante.com>
  - <http://MSDN.microsoft.com/library/en-us/>
  - <http://www.pntic.mec.es/ies2000/iessql.htm#SQL>
  - <http://support.microsoft.com>
  - <http://www.webestilo.com/html/cap3a.phtml>
  - <http://www.websapespanol.com/tutorial/ASP.htm>