

UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN

ESCUELA DE INGENIERÍA DE SISTEMAS

**“IMPLEMENTACIÓN DEL SISTEMA DE
ADMINISTRACIÓN DE ARCHIVOS TEXTUALES
DE LA CASA DE LA CULTURA NÚCLEO DEL
AZUAY – HEMEROTECA PIEDAD PAREDES DE
JARAMILLO”.**

**Tesis de Grado previa a la
obtención del Título de
Ingeniero de Sistemas.**

DIRECTOR:

Ing. Francisco Salgado

AUTORES:

**Sandra Abril Zambrano.
Susana Rodas Sumba.**

**Cuenca - Ecuador
2005**

Las ideas y opiniones vertidas en la presente Tesis, son de exclusiva responsabilidad de sus autores.

Sandra Abril Zambrano

Susana Rodas Sumba

AGRADECIMIENTO

Nuestro principal agradecimiento a **Dios**, por su bendición diaria, durante nuestros años de estudio.

A la nuestra querida **Universidad del Azuay**, por albergarnos en estos años de carrera universitaria y a nuestros profesores por entregarnos sus valiosos conocimientos.

Al **Ing. Francisco Salgado**, nuestro Director de Tesis, ya que sin su ayuda y apoyo incondicional no hubiera sido posible la culminación de este trabajo.

A la **Hemeroteca Piedad Palacios de Jaramillo – Casa de la Cultura Núcleo del Azuay**, que nos brindó la oportunidad de realizar este tema.

A la **Biblioteca del Banco Central**, que nos dió su apoyo para la obtención de la información de periódicos, para nuestra Base de Datos.

**Sandra
Susana**

DEDICATORIA

Esta Tesis va dedicada al Ser que con toda paciencia, amor y tolerancia estuvo junto a mi en todo momento, siendo mi apoyo y la razón por la cual he decidido superarme para su propio bien a mi hijo: Javier Sebastián.

A Javier, mi esposo, que con su sacrificio me apoyó para la realización de esta tesis.

A mis Padres Ramón y Adriana, como reconocimiento a su amor, sacrificio, dedicación y apoyo decisivo, para que logre alcanzar esta meta importante en vida.

A mis hermanos, por su cariño y comprensión, en especial a Juan Pablo, por su apoyo incondicional.

Sandra

Al culminar esta etapa de mi vida, dedico este trabajo a mis queridos Padres Arturo y Teresita, por el gran amor, sacrificio y entrega que me brindaron, durante mis años de estudiante.

A mi querido Esposo Marcelo, por su paciencia y apoyo constante durante estos años.

A mis hijas Ma. Gabriela y Maithe, por su tiempo compartido y su sacrificio durante la realización de esta tesis.

A mi familia, especialmente a mi hermana Tania, por su ayuda, para la culminación de este trabajo de tesis.

Susana.

ÍNDICE GENERAL

CAPÍTULO I

INVESTIGACIÓN DE CAMPO Y RECOPIACIÓN DE LA INFORMACIÓN

1.1	Introducción.....	1
1.2	Antecedentes.....	2
1.2.1	¿Porqué desarrollar la aplicación?.....	3
1.2.2	¿Para quién está orientado el software?.....	4
1.3	Procedimiento para el desarrollo.....	4
1.4	Teoría Referencial.....	5
1.4.1	¿Qué es un sitio web?.	5
1.4.2	Estructuras.....	6
1.4.3	Tipos de Estructuras	7
1.4.3.1	Estructura Jerárquica	7
1.4.3.2	Estructura lineal	8
1.4.3.3	Estructura lineal-jerárquica o mixta	9
1.4.3.4	Estructura de Frames	10
1.4.3.5	Estructura Web	11
1.4.4	Motores de Búsqueda	12
1.4.4.1	¿Qué son los motores de búsqueda?.	12
1.4.4.2	Historia de los buscadores	13
1.4.4.3	Componentes de un buscador	13
1.4.4.3.1	Programas robot	13
1.4.4.3.2	Un equipo de expertos	14
1.4.4.3.3	Un servidor de Web	14

1.4.4.4	Etapas de la búsqueda	14
1.4.4.4.1	Recolectar datos	14
1.4.4.4.2	Catalogar	15
1.4.4.4.3	Interfases	15
1.4.4.5	Tipos de búsquedas	16
1.4.4.5.1	Búsquedas temáticas	17
1.4.4.5.2	Búsquedas por palabras	18
	clave o automáticos	
1.4.4.6	Búsquedas avanzadas	20
1.5	Conclusiones	22

CAPÍTULO II

ESTUDIO Y REVISIÓN DE LAS HERRAMIENTAS PARA EL DESARROLLO DE LA APLICACIÓN

2.1	Introducción	23
2.2	Estudio de la configuración del Hardware a utilizar para el desarrollo	24
2.3	Estudio del Software a utilizar	24
2.3.1	Dreamweaver MX 2004	24
2.3.2	ASP. (Active Server Page)	25
2.3.3	MySQL	27
2.3.4	PHP	28
2.3.5	FireWorks	31
2.3.6	Adobe Acrobat Professional	32
2.3.7	Archivos de formato PDF	33
2.3.8	I-Filter	34

2.3.9 CSS Cascading Style Sheets	35
2.4 Internet Information Server	36
2.5 Index Server de Microsoft Windows	37
2.6 Conclusiones	40

CAPÍTULO III

DISEÑO DE LA ESTRUCTURA DE ARCHIVOS

3.1 Introducción	42
3.2 Bases de datos de texto no estructurado	43
3.2.1 Repositorios de información	43
3.3 Características del Motor de Búsqueda	44
3.3.1 Catálogos de Búsqueda utilizados por el Index Server.	45
3.3.2 Características básicas de las consultas	46
3.3.2.1 Consultas de lógica borrosa	47
3.3.3 Registro	48
3.3.4 Caracteres comodín	48
3.3.5 Consultas de texto libre	49
3.3.6 Nombres de propiedades	49
3.4 Diccionario de datos	50
3.5 Estructura de Directorios	55
3.6 Nombre de Archivos	56
3.7 Base de Datos para administración de archivos	56
3.8 Conclusiones	60

CAPÍTULO IV

ANÁLISIS Y DISEÑO DE LOS RECURSOS DE HARDWARE Y DE REDES REQUERIDOS PARA EL SISTEMA

4.1	Introducción	61
4.2	Determinación del equipo necesario para la implementación	62
4.3	Servidor principal	62
4.4	Red de computadores	63
4.4.1	Estación de trabajo	63
4.4.2	Estación para el Cliente	63
4.4.3	Unidades de entrada y salida de la información	64
4.4.3.1	Scanner	64
4.4.3.2	Impresora	64
4.5	Cotizaciones	64
4.6	Conclusiones	65

CAPÍTULO V

PROGRAMACIÓN DE LA APLICACIÓN WEB

5.1	Introducción	67
5.2	Diseño de la página web	68
5.2.1	Diseño Gráfico	68
5.2.2	Diseño Estructural	68
5.3	Estructura del Sitio Web	70

5.4	Interfaz de la aplicación Web y el servidor	71
5.5	Definición de las búsquedas	72
5.6	Definición del Front-End o interfase con el usuario	75
5.7	Conclusiones	81

CAPÍTULO VI

DEFINICIÓN DE LA ENTRADA DE LA INFORMACIÓN

6.1	Introducción	82
6.2	Tecnologías de Escaneo Óptico	83
6.2.1	Lector de marcas ópticas (OMR)	83
6.2.2	Sistemas de escaneo basados en el reconocimiento de caracteres ópticos (OCR)	84
6.2.3	Sistemas de escaneo basados en el reconocimiento inteligente de caracteres (ICR)	85
6.2.4	Tecnología de imagen	86
6.3	Guía para la subida de la información	87
6.3.1	Captura de la información a través del scanner	87
6.3.2	Utilización de un software que le permita realizar los cambios necesarios a la noticia deseada	87
6.3.3	Reconocimiento de texto usando OCR	88
6.3.4	Ingreso de la información a la base de datos	88
6.4	Conclusiones	90

CAPÍTULO VII

IMPLEMENTACIÓN DE LA APLICACIÓN.

7.1	Introducción	91
7.2	Captura de la información	92
7.3	Ingreso al mantenimiento de archivos	92
7.4	Página de Administración de Archivos	93
7.4.1	Ordenación de los archivos	95
7.4.2	Links de acceso	96
7.4.2.1	Periódico	96
7.4.2.2	Sección	98
7.4.2.3	Nuevo Archivo	99
7.4.3	Mantenimiento de la información de archivos	101
7.5	Manejo de las búsquedas	101
7.6	Conclusiones	105

CAPÍTULO VIII

CONCLUSIONES Y RECOMENDACIONES

8.1	Conclusiones	106
8.2	Recomendaciones	109

ÍNDICE DE FIGURAS

CAPÍTULO I

INVESTIGACIÓN DE CAMPO Y RECOPIACIÓN DE LA INFORMACIÓN

Figura 1.1. Modelo jerárquico organizado.....	6
Figura 1.2. Estructura Jerárquica.....	7
Figura 1.3. Estructura Lineal.....	8
Figura 1.4. Estructura lineal-jerárquica o mixta.....	9
Figura 1.5. Estructura de Frames.....	10
Figura 1.6. Estructura Web.....	11
Figura 1.7. Catalogo de búsquedas temáticas.....	18

CAPÍTULO III

DISEÑO DE LA ESTRUCTURA DE ARCHIVOS

Figura 3.1. Caracteres comodines.....	49
Figura 3.2. Consultas de texto libre.....	49
Figura 3.3. Nombre de propiedades de los archivos.....	50
Figura 3.4. Tabla de nombre de propiedades del Index Server.....	51
Figura 3.5. Tabla de Usuarios.....	57
Figura 3.6. Tabla de Información de archivos.....	58
Figura 3.7. Tabla de Nombres de Periódicos.....	59
Figura 3.8. Tabla de Nombres de Secciones.....	59

CAPÍTULO IV

ANÁLISIS Y DISEÑO DE LOS RECURSOS DE HARDWARE Y DE REDES REQUERIDOS PARA EL SISTEMA

Figura 4.1. Cotizaciones de Hardware y Software.....	64
--	----

CAPÍTULO V

PROGRAMACIÓN DE LA APLICACIÓN WEB

Figura 5.1. Estructura de Mantenimiento de Archivos.....	70
Figura 5.2. Estructura de Búsqueda de Archivos.....	70
Figura 5.3. Interfaz de búsqueda de Archivos.....	71
Figura 5.4. Interfaz de Administración de Archivos.....	71
Figura 5.5. Interfaz de Creación de Archivos.....	72
Figura 5.6. Formulario para Ingreso de las búsquedas.....	72
Figura 5.7. Página con resultado de la búsqueda.	74
Figura 5.8. Página para ingreso de la clave de acceso.....	76
Figura 5.9. Página para listar los archivos de periódicos.....	77
Figura 5.10. Página para grabación de los archivos de periódicos.....	77
Figura 5.11. Página para Mantenimiento de los archivos de periódicos.....	78
Figura 5.12. Página para listar los periódicos.....	78
Figura 5.13. Página para mantenimiento de los nombres de Periódicos.....	79
Figura 5.14. Página para listar de los nombres de Secciones.....	79
Figura 5.15. Página para mantenimiento de las Secciones.....	60

CAPÍTULO VII

IMPLEMENTACIÓN DE LA APLICACIÓN

Figura 7.1. Página de Acceso a Mantenimiento de archivos.....	93
Figura 7.2. Página de Administración de Archivos.....	94
Figura 7.3. Links de Acceso.....	96
Figura 7.4. Listado de nombres de Periódicos.....	96
Figura 7.5. Mantenimiento de Periódicos.....	97
Figura 7.6. Listado de nombres de Secciones.....	98
Figura 7.7. Mantenimiento de Secciones.....	99
Figura 7.8. Creación de Archivos Nuevos.....	100
Figura 7.9. Mantenimiento de Archivos	101
Figura 7.10. Formulario de Búsquedas.....	102
Figura 7.11. Formulario de Respuestas de Búsquedas.....	103
Figura 7.12. Ejemplo de Archivo de la Noticia.....	104

CAPÍTULO I

**INVESTIGACIÓN DE CAMPO Y
RECOPIACIÓN DE LA
INFORMACIÓN**

CAPÍTULO I

INVESTIGACIÓN DE CAMPO Y RECOPIACIÓN DE LA INFORMACIÓN

1.1 Introducción.

El presente capítulo describe la razón del desarrollo de esta aplicación, los objetivos que persigue el mismo, las personas para las que está dirigida la aplicación, indicaremos el procedimiento con el cual se construirá la aplicación. Es también importante conceptuar los tipos de estructuras web existentes y definir la que utilizaremos, así también daremos un breve detalle de los motores de búsqueda.

1.2 Antecedentes.

La Hemeroteca “Piedad Paredes de Jaramillo”, fué creada en el año 1990, y cuenta con un fondo bibliográfico de alrededor de 600 títulos en la sección de obras nacionales y 900 en la sección extranjera en los más variados idiomas: inglés, francés, italiano, portugués, alemán, rumano, noruego, ruso, etc., ello da cuenta de la presencia de nuestra institución en el extranjero, a través de la suscripción, el canje y la donación.

Cabe destacar que esta sección de la Biblioteca ha sido procesada analíticamente hasta el último artículo, incluyendo los suplementos dominicales de los respectivos diarios, en una base de datos con 60800 registros bibliográficos, la misma que se encuentra a disposición del público junto con la base de la Biblioteca; las que se pueden consultar a través de ficheros electrónicos en las salas de lectura o en el Internet en la siguiente dirección: <http://www.casaculturacuena.org.ec/biblioteca.html>.

En cuanto a periódicos, la Hemeroteca posee los diarios más destacados del país, y que en el esfuerzo por informar de la mejor manera su contenido, pone a disposición del público, a través de un mecanismo manual de búsqueda, una selección de artículos más relevantes en el campo de la cultura, la ciencia y la tecnología tomados de los siguientes diarios: El comercio, El Día, El Mercurio, El Sol, El Tiempo, El Telégrafo, El Universo, La Nación y Tiempos del Mundo.

1.2.1 ¿Porqué desarrollar la aplicación?

Actualmente la Hemeroteca “Piedad Paredes de Jaramillo” no posee una aplicación que permita manipular la cuantiosa información de los diferentes diarios del país, por lo que es necesario tener un sistema que permita acceder al mismo. Dicha información es muy valiosa y está siendo subutilizada debido a que no cuentan con un sistema de información que les permita administrarla eficientemente.

Es urgente la implementación de un software que evite la manipulación física de los diarios ya que el material de el que están hechos, el papel periódico, se deteriora fácilmente y está expuesto a cualquier siniestro. Por esta razón es necesario pasar esta información a un medio más seguro, como lo es un computador, del que se podrá además realizar respaldos que protejan mucho más esta información.

Lo que pretendemos desarrollar es una aplicación en Página Web que permita sustituir la manipulación del material físico con el manejo electrónico de la misma, permitiendo que el servicio al lector sea más provechoso. El acceso a los periódicos será través de un Sistema buscador de artículos en página WEB con hipervínculos que permitan acceder al texto.

Adicionalmente será necesario realizar un estudio técnico de redes y equipos, debido a que la Hemeroteca no cuenta con un sistema informático de comunicación, se

sugerirá la compra de un servidor, terminales y más dispositivos necesarios.

1.2.2 ¿Para quién está orientado el software?

El sistema está orientado para dar servicio al público, permitiéndoles la manipulación, búsqueda de archivos textuales e impresión de artículos de su interés. Será utilizado por los estudiantes de escuela, colegio y universidad, así como también profesores, profesionales y público en general, ya que la Hemeroteca cuenta con valiosa información económica, política, cultural, nacional, internacional y de opinión de nuestro país, recopilada desde el año 1940 en los diferentes diarios como: El Mercurio, El Tiempo, El Comercio, El Universo y El Hoy.

Además el software facilitará al personal de la Hemeroteca la manipulación de la información, permitiéndoles administrarla más eficientemente.

1.3 Procedimiento para el desarrollo

Hemos realizado la investigación de campo, recopilando la información y determinando el problema, ya expuesto anteriormente. Posteriormente nos enfocaremos en el estudio y revisión de las herramientas para el desarrollo de la aplicación, definiremos el software y la plataforma más adecuada. Una vez definido el ambiente de desarrollo realizaremos el diseño de la estructura de los archivos. Debido a que la Hemeroteca no posee una estructura adecuada para la implementación de la aplicación

realizaremos un análisis y diseño de los recursos de Hardware y Redes necesarios para el funcionamiento adecuado de la aplicación.

Posteriormente continuaremos con el diseño, definición de las búsquedas y Programación de la Página Web, además realizaremos una guía para utilización del sistema.

Definiremos el proceso a utilizar para la entrada de la información, es decir de los artículos de los periódicos.

Por último realizaremos las pruebas e implementación de la aplicación.

1.4 Teoría Referencial

Un sitio Web es un conjunto de documentos enlazados que poseen elementos compartidos: un mismo tema, un mismo diseño o un objetivo común.

1.4.1 ¿Qué es un sitio web?

Evidentemente es posible crear documentos individuales, pero éstos raramente tienen un fin en sí mismos, sino que están al servicio de un ente mayor: el sitio web. Un sitio agrupa páginas web que están relacionadas entre sí. El primer paso a la hora de crear un sitio consiste en planificarlo cuidadosamente.

1.4.2 Estructuras

A la hora de diseñar un site son muchos los puntos a tener en cuenta para lograr que sea atractivo en su estética y en su contenido y al mismo tiempo que posea la virtud de ser funcional. Es por eso que plantear una buena estructura para un site es un paso primordial. Y debe ser el primero antes de empezar. Para ello hay que tener en claro el tipo y la cantidad de contenido que figurará en el mismo.

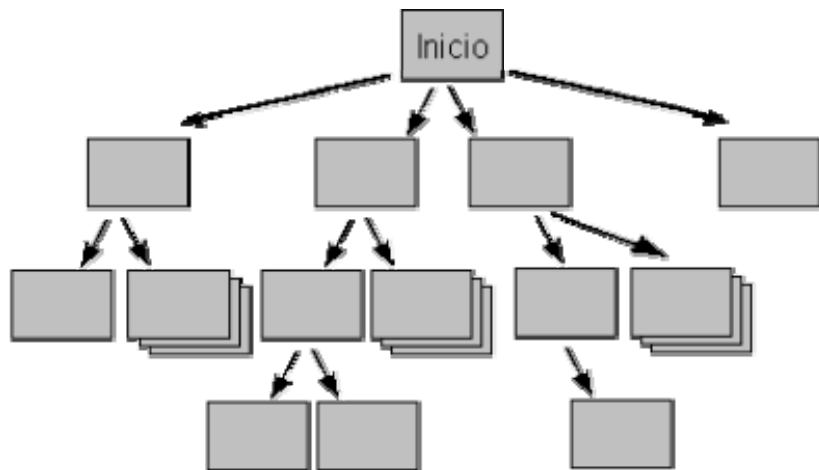


Figura 1.1. Modelo jerárquico organizado.

Es necesario tener en mente el análisis previo debidamente ordenado y secuenciado. Hay que moverse desde lo más importante hasta los detalles.

Por más simple que pueda resultar planificar la distribución de la información en un site, son muchos los que incurren en decisiones poco funcionales para el navegante.

Para que el internauta pueda recorrer el site con facilidad y acceder al contenido que le interesa se recomiendan cinco tipos de estructuras.

1.4.3 Tipos de Estructuras

La estructura de un sitio web se va a referir a la disposición entre los enlaces de las diferentes páginas que lo forman, es decir, al esquema general de disposición de las páginas entre sí y a la forma de acceso entre ellas.

Los principales tipos de estructura son:

1.4.3.1 Estructura Jerárquica

Que parte de una página principal mediante la que se puede acceder a diferentes páginas secundarias, a partir de las cuales podemos acceder a las páginas terciarias, y así sucesivamente. La disposición de un sitio de este tipo sigue el esquema general expresado en el siguiente gráfico:

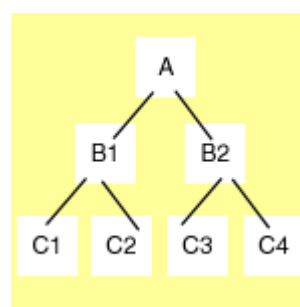


Figura 1.2. Estructura Jerárquica.

Si usamos una estructura de tipo jerárquica podemos crear un menú general en la página principal, mediante el que daremos acceso a las diferentes páginas de entrada a

las secciones, y en cada una de estas deberemos establecer otro menú desde el que el usuario pueda ir a cualquiera de las páginas que la componen. En cada una de las páginas individuales tendremos que implementar unos enlaces a las diferentes secciones principales y a la página de inicio.

1.4.3.2 Estructura lineal

En la que partiendo de una página inicial se van recorriendo las demás del sitio web secuencialmente, una detrás de otra. Es análoga en su disposición a la estructura de las páginas de un libro. Cada página posee un enlace a su anterior en la secuencia y otro a su siguiente. La representación gráfica es la siguiente:



Figura 1.3. Estructura Lineal.

El sistema de menús característico de este tipo de estructura sería el acceso a una página de entrada, desde la que podemos acceder únicamente a la página que le sigue en la secuencia establecida, y en esta encontraremos un pequeño menú, generalmente situado en la parte inferior o superior de la misma, desde el que podemos acceder tanto a la página anterior como a la siguiente en la secuencia, y así sucesivamente, hasta llegar a la última página, en la que sólo figurará un enlace a la página anterior.

Debido a sus repercusiones de diseño y navegación, este tipo de estructuras en su forma pura es raramente usado.

1.4.3.3 Estructura lineal-jerárquica o mixta

Que como su propio nombre indica es una mezcla de las dos anteriores, en la que partiendo de una página principal o de inicio se accede a diferentes páginas de entrada a secciones, a partir de las cuales la navegación es lineal. Por ejemplo, si accedemos a cualquier artículo desde las páginas cabeceras de sección, navegando luego uno a uno los diferentes capítulos que lo forman.

Su representación gráfica es la siguiente:

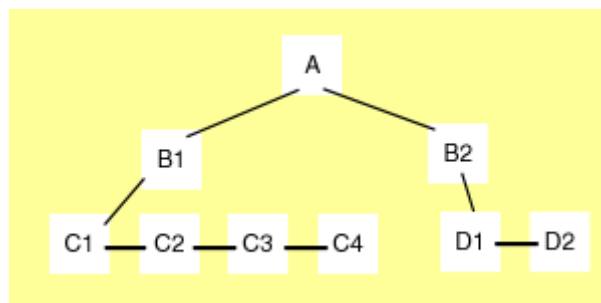


Figura 1.4. Estructura lineal-jerárquica o mixta.

En el caso de usar este tipo de jerarquía deberemos situar un menú en la página de inicio desde el que podamos acceder a las páginas de entrada a cada uno de los temas a tratar, y en cada una de las páginas que forman la secuencia del tema tendremos que establecer un link a la página anterior y otro a la siguiente. Como complemento podemos habilitar en cada una de ellas un enlace a la página que abre

la secuencia, y en cada una de estas otro a la página de inicio.

1.4.3.4 Estructura de frames

Que es la típica de una interfaz a base de frames y en la que el usuario dispone de un menú siempre presente desde el que puede acceder en todo momento a las páginas de entrada a las diferentes secciones del sitio web, a partir de las que puede navegar bien de forma jerárquica, bien de forma lineal, bien de forma mixta. Su representación gráfica es del tipo:

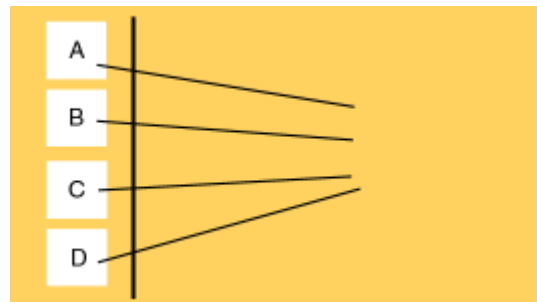


Figura 1.5. Estructura de Frames.

Este tipo de estructura se suele combinar con otro jerárquico o mixto que nos ayude a navegar dentro de los subtemas principales, a los que accedemos a través del frame lateral.

El sistema de menús consiste generalmente en un menú situado en un frame lateral, que nunca cambia, desde el que se accede a las diferentes secciones principales del sitio, cuyas páginas iniciales se cargan en el frame central, que suele ser el de mayor tamaño. Para acceder a las diferentes subsecciones y páginas se suelen establecer los

enlaces adecuados bien como submenús en el frame lateral, bien como menús individuales dentro de cada página de entrada a las secciones, dentro del frame principal.

1.4.3.5 Estructura web

En la que podemos estructurar las diferentes páginas con libertad total. Es la que da más facilidades a los diseñadores, pero puede resultar a veces demasiado confusa para los usuarios, ya que le permite visitar un sitio sin un rumbo fijo, pudiendo desde cualquier página acceder a los contenidos de un conjunto indeterminado de otras. No es aconsejable su uso, ya que suele resultar caótica. Su representación gráfica puede ser del tipo:

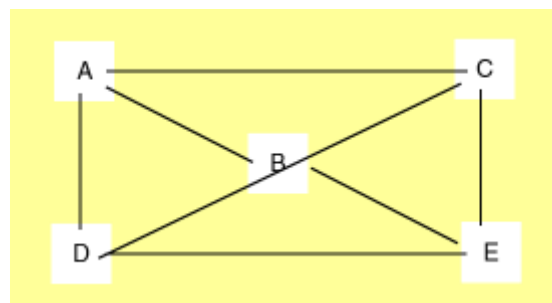


Figura 1.6. Estructura Web.

Estos son los tipos principales de estructura de un sitio web. Generalmente se suele usar un tipo mixto o un tipo frames, dependiendo siempre de la naturaleza propia del sitio.

1.4.4 Motores de Búsqueda

1.4.4.1 ¿Qué son los motores de búsqueda?

Los motores de búsqueda o search engines (en inglés) son programas computacionales cuya función es buscar constantemente páginas nuevas y actualizar sus registros acerca de viejas páginas. Estos programas también se les conocen como robots de búsqueda.

Estos programas leen el código HTML de las páginas, que es en realidad el código con el que se encuentran escritas todas las páginas que visitas en Internet. En base a la información encontrada, esta es clasificada y de estas se alimentan los grandes programas o motores de búsqueda de Internet, Motores como Google.com, webcrawler, yahoo, etc.

En todos estos sitios, pueden tener acceso a una búsqueda detallada o general sobre temas diversos, simplemente se deben de poner la palabra o tema que se busca y el buscador mostrará una cantidad muy grande de opciones (paginas web) que hablan sobre el tema. Así podrán encontrar de manera más sencilla aquello que se busca. Esto por supuesto para facilitar y agilizar el tiempo que se invierte en Internet.

Un buscador o un robot también se le conoce como arañas de búsqueda o web crawlers, ya que se mueven a través de la red de redes visitando sitios y extrayendo los enlaces que se encuentran en estos.

1.4.4.2 Historia de los buscadores

Nacieron de la necesidad de organizar la información anárquica contenida en la Internet. Se basaron en una herramienta llamada Gopher creada por la Universidad de Minesotta a principios de los '90. Fueron desarrollados por empresas, organizaciones o individuos e instalados en servidores conectados a Internet. Los buscadores más utilizados, como Yahoo o Altavista, forman parte de algunos de esos emprendimientos exitosos.

A medida que un buscador se hace más popular, comienza a concentrar más y más datos.

Pero sin duda las claves del arte de buscar no consisten en conectarse al buscador ni en recorrer miles de documentos sino en aprender a detallar los pedidos con la precisión necesaria para que el mecanismo de búsqueda brinde pocas opciones: formular la pregunta adecuada es el requisito fundamental para obtener la respuesta justa.

1.4.4.3. Componentes de un buscador

Los buscadores se componen de varias partes en delicado equilibrio e interacción, cada una con una función específica:

1.4.4.3.1 Programas robot que recolectan la mayor cantidad de información de todo el mundo y la almacenan en una enorme base de datos.

1.4.4.3.2 Un equipo de expertos en la organización de la información que la catalogan por temas (categorías) o palabras clave.

1.4.4.3.3 Un servidor de Web que, a través de un programa que examina la base de datos del buscador, recibe los pedidos de información del usuario y le devuelve los resultados.

1.4.4.4. Etapas de la búsqueda

1.4.4.4.1 Recolectar datos

Algunos buscadores poseen programas robots especiales llamados Wanderers (viajeros), spiders (arañas) o simplemente robots, que en forma automática recorren de página en página los links (enlaces) de la Web y almacenan todo lo que encuentran en su camino. El Inktomi y el Webcrawler son algunos de los buscadores basados en este método de recolección de datos.

Otros buscadores se basan en los datos que les envían los usuarios. Cuando alguien diseña un sitio, lo primero que debe hacer es darse a conocer, entre otros medios, a través de los buscadores, mediante un formulario (form) donde se ingresan los datos del sitio y los temas o palabras clave con los que está relacionado.

La mayoría de los buscadores, sin embargo, utilizan ambos métodos de recolección de datos combinados.

1.4.4.4.2 Catalogar

Una vez que está almacenada en la base de datos, la información de las páginas ingresa a la etapa de análisis efectuada, en general, por especialistas en bibliotecología o en la organización de unidades de información. La forma de ordenar los datos es fundamental para el éxito de una herramienta de búsqueda y difícilmente pueda programarse para que el ordenamiento se realice en forma automática. Las personas encargadas de esta tarea podrán verificar cada sitio que los robots almacenaron, recorrerlo y decidir en qué categorías estará incluido, qué palabras clave deben activarlo, etc. Este es un proceso delicado y juega un papel fundamental en la eficiencia de la herramienta.

1.4.4.4.3 Interfases

Después de estas dos primeras etapas, la información estará a disposición del que quiera consultarla. La búsqueda funciona del siguiente modo:

Conectarse con el buscador elegido. La elección depende de los gustos y necesidades personales. Dado que cada buscador tiene sus propios mecanismos para relevar información, los resultados serán diferentes en cada caso.

Ingresar una o varias palabras clave, luego dar la orden de comenzar o bien elegir un tema del menú desplegado por el buscador.

El servidor del buscador recibirá el pedido, recorrerá su base de datos buscando información que coincida con el requerimiento y generará un resultado.

En la pantalla aparecerá un documento Web encabezado por una leyenda indicativa de cuántos sitios se encontraron. Aparecerán también los nombres y direcciones de los primeros 10 o 20 sitios y (opcionalmente) una breve descripción del contenido de cada uno.

En ese momento se podrá elegir si se desea seguir alguno de esos enlaces cliqueando sobre ellos o solicitar los próximos diez sitios, o bien estrechar la búsqueda si la cantidad de sitios de resultado son demasiados. Estas opciones figuran usualmente al pie de la página del buscador.

1.4.4.5 Tipos de búsquedas

Existen dos formas básicas de buscar:

- a) **Búsquedas temáticas:** El buscador ofrece al usuario una serie de temas o áreas de interés. Una vez seleccionado uno de ellos, se abren nuevas opciones de subtemas y así, en pasos sucesivos, se va refinando la búsqueda hasta llegar a la información requerida.

- b) **Búsquedas por palabras clave o automáticas:** Se activan a partir de una o más palabras clave ingresadas por el usuario. El buscador ofrecerá, como

resultado, una página con la lista de los sitios que contienen los términos solicitados.

1.4.4.5.1 Búsquedas temáticas.

Están compuestos por dos partes: la base de datos que es construida por las direcciones de las páginas remitidas y una estructura jerárquica que facilita la consulta.

Por lo general, suelen ser más lentas porque el usuario debe ir afinando, tema por tema, su pedido hasta llegar al punto que le interesa. Pero tienen más precisión y son ideales cuando se necesita encontrar una gran variedad de fuentes sobre un mismo tema.

En las búsquedas temáticas, a las herramientas que organizan la información en temas se las llama catálogos (Net Directory en inglés).

<u>Arte y cultura</u> <u>Literatura, Teatro, Museos...</u>	<u>Internet y ordenadores</u> <u>WWW, Aplicaciones, Revistas...</u>
<u>Ciencia y tecnología</u> <u>Animales, Informática, Ingeniería...</u>	<u>Materiales de consulta</u> <u>Bibliotecas, Diccionarios...</u>
<u>Ciencias sociales</u> <u>Economía, Psicología, Historia...</u>	<u>Medios de comunicación</u> <u>Temas de actualidad, Periódicos, TV...</u>
<u>Deportes y ocio</u> <u>Fútbol, Deportes, Turismo...</u>	<u>Política y gobierno</u> <u>Países, Embajadas, Derecho...</u>
<u>Economía y negocios</u> <u>Para empresas, Para consumidores, Empleo...</u>	<u>Salud</u> <u>Medicina, Enfermedades...</u>
<u>Educación y formación</u> <u>Primaria, Secundaria, Universidades...</u>	<u>Sociedad</u> <u>Gastronomía, Culturas, Religión...</u>
<u>Espectáculos y diversión</u> <u>Cine, Actores, Música, ¡Genial!...</u>	<u>Zonas geográficas</u> <u>Países, Europa, CC.AA....</u>

Figura 1.7. Catálogo de búsquedas temáticas.

1.4.4.5.2 Búsquedas por palabras clave o automáticas

Se activan a partir de una o más palabras clave ingresadas por el usuario. El buscador ofrecerá, como resultado, una página con la lista de los sitios que contienen los términos solicitados.

Están compuestos por tres partes: los robots que recorren la red escrutándola. La base de datos que es construida por los robots y el motor de búsqueda que facilita la consulta a la base.

La búsqueda por palabra clave (keywords) suele ser más rápida aunque menos precisa que la anterior. Hay que

conectarse con algún buscador de los que se mencionan más adelante.

Si el usuario no sabe efectuar su pedido con precisión, la búsqueda por palabras clave puede llevar horas ya que un término puede estar presente en cientos de miles de sitios.

Para buscar, el sitio se presenta una línea en blanco para que el usuario ingrese la palabra o palabras que mejor describan el tema que busca.

Si se ingresa una sola palabra, por ejemplo música, aparecerán miles de enlaces con páginas que citan ese tema. Será imposible, obviamente, recorrerlas todas. Si, en cambio, se ingresan varias palabras, se podrá estrechar la búsqueda hacia subtemas más específicos.

Una vez ingresadas las palabras, se debe hacer clic sobre "buscar" (seek o find, según el buscador).

El buscador mostrará el resultado en forma de una página Web encabezada por una leyenda indicativa de cuantos sitios que satisfacen el pedido se encontraron. También aparecerán los nombres y direcciones de los primeros 10 o 20 sitios y opcionalmente una breve descripción del contenido de cada uno. En este momento se podrá elegir si se desea seguir cliqueando alguno de esos links o solicitar los próximos diez sitios, o bien estrechar su búsqueda (definir mejor las palabras clave) si la cantidad de

sitios de resultado son demasiados. Estas opciones figuran usualmente al pie de la página del buscador.

1.4.4.6 Búsquedas avanzadas

Cada herramienta de búsqueda utiliza diversas notaciones para incorporar una combinación compleja de palabras claves.

Muchas herramientas dejan cambiar la manera en que las palabras claves se combinan y por lo tanto dejan controlar la búsqueda. Para hacer esto, se necesita proveer la información adicional para ayudar a los motores de búsqueda a no hacer caso de sitios inaplicables y para incluir solo los relevantes. Esto se hace con el uso de lógica de Boolean y el uso de las palabras AND (y), OR (o), NEAR (cerca) y NOT (no).

Los operadores básicos que se deben usar son:

Comillas: Permiten “pegar” varias palabras en una expresión, de modo que si en una página no aparecen exactamente, sino en otro orden se deseche la información encontrada.

Mayúsculas y minúsculas: Al diferenciar mayúsculas y minúsculas se fuerza a que esas palabras aparezcan literalmente (Ej. “Ecuador”).

Incluida (+): El signo más delante de una palabra obliga a que la palabra que va a continuación esté incluida en la información encontrada.

Eliminar (-): El signo menos elimina páginas que contengan la palabra que va a continuación. [Si se emplea este guión para un término que contenga guiones, se debe escribir las dos palabras sin separarlas con un espacio (“deja-vu”).]

Raíz (*): El asterisco sirve para indicar raíces de palabras (por ejemplo, “ecuador” o “ecuatoriano”). En general, el asterisco reemplaza a una serie indeterminada de letras: “col*r” es tanto “color” como “colour”.

Se utiliza la expresión AND, para incluir una palabra (“rojo AND negro”), OR, para excluir una palabra, NEAR, para ubicar una palabra en un cierto contexto (“mesa NEAR venta”, para localizar sitios de venta de mesas, y no sobre que es un mesa) y la expresión NOT, para excluir alguna palabra.

1.5 Conclusiones.

Tenemos como objetivo principal la realización de un sistema de digitalización de periódicos, como un aporte para la Hemeroteca “Piedad Paredes de Jaramillo” de la Ciudad de Cuenca, pues el sistema que tienen actualmente es manual. Con este sistema podrá estar la información más importante de los periódicos que se encuentran embodegados desde el año 1940 al alcance de un público más amplio del que actualmente ahora está a disposición.

Lo que pretendemos desarrollar es una aplicación en Página Web que permita sustituir la manipulación del material físico con el manejo electrónico de la misma, permitiendo que el servicio al lector sea más provechoso y evitando así que dicha información pueda perderse por el deterioro de la misma. El acceso a los periódicos será través de un Sistema buscador de artículos en página WEB con hipervínculos que permitan acceder al texto.

La interfaz que utilizaremos será la lineal en la que partiendo de una página inicial se van recorriendo las demás del sitio web secuencialmente, una detrás de otra.

La forma de búsqueda será a partir de una o más palabras clave ingresadas por el usuario. El buscador ofrecerá, como resultado, una página con la lista de los archivos que contienen las noticias relacionadas con la palabra o frase.

CAPÍTULO II

**ESTUDIO Y REVISIÓN DE LAS
HERRAMIENTAS PARA EL
DESARROLLO DE LA
APLICACIÓN**

CAPÍTULO II.

ESTUDIO Y REVISIÓN DE LAS HERRAMIENTAS PARA EL DESARROLLO DE LA APLICACIÓN

2.1 Introducción

En este capítulo analizaremos la configuración del hardware y las herramientas necesarias para el desarrollo e implementación de nuestra aplicación entre las que describiremos brevemente las siguientes:

- Dreamweaver MX 2004
- ASP
- MySql
- PHP
- FireWorks
- Adobe Acrobat Professional.
- I-Filter
- CSS Cascading Style Sheets.
- Internet Information Server.
- Index Server de Microsoft Windows

2.2 Estudio de la configuración del Hardware a utilizar para el desarrollo

Los requisitos mínimos para realizar el desarrollo de nuestra aplicación son:

- Procesador Intel, Pentium 4 de 600 MHz o equivalente
- Windows 98 SE, Windows 2000, Windows XP, o Windows Server™ 2003 , Windows NT
- 128 MB de RAM (se recomienda 256 MB)
- 1 GB de espacio disponible en el disco

2.3 Estudio del Software a utilizar

Dentro del software necesario para el desarrollo de esta aplicación hemos pensado en un diseñador de páginas Web como el Dreamweaver MX 2004, en un entorno para crear y ejecutar aplicaciones web el ASP y un diseñador de páginas Web como es el Fireworks.

2.3.1 Dreamweaver MX 2004

Dreamweaver MX 2004 es un entorno profesional de creación de aplicaciones web. Una aplicación web es un conjunto de páginas que interactúan unas con otras y con diversos recursos en un servidor web, incluidas bases de datos.

Dreamweaver MX 2004 es un editor profesional para la creación y administración de los sitios y páginas web. Dado que incorpora las herramientas de diseño de páginas y

administración de sitios de Dreamweaver MX 2004 facilita la creación, administración y edición de páginas Web para múltiples plataformas y navegadores.

Con Dreamweaver podemos crear páginas HTML sin tener que preocuparnos por el código HTML, recordar todos los "tags" necesarios para componer nuestra página o tener que previsualizar en nuestra cabeza cual será el resultado compositivo del documento final. De esta manera, crear un documento HTML se convierte en una tarea menos parecida a programar y más parecida a maquetar. En resumen, podríamos decir que Dreamweaver es un programa de "maquetación" de páginas web, es además una herramienta completa para la gestión y desarrollo de sitios web completos.

Para crear aplicaciones Web en Dreamweaver, se necesitará lo siguiente:

- Un Servidor Web
- Un Servidor de Aplicaciones que se ejecute en un servidor Web, o un servidor Web que actúe también como servidor de aplicaciones como Internet Information Server (IIS).
- Una Base de Datos o Sistema de Base de Datos.
- Un Controlador de Base de Datos compatible con su sistema de Base de Datos.

2.3.2 ASP (Active Server Page)

Las Páginas Active Server (ASP, Active Server Pages), es un entorno para crear y ejecutar aplicaciones dinámicas e interactivas en la Web.

Se puede combinar páginas HTML, secuencias de comandos y componentes ActiveX para crear páginas y aplicaciones Web interactivas.

ASP es un lenguaje orientado a las aplicaciones en red creado por Microsoft que funciona del lado servidor. Es en efecto el servidor quien se ocupa de ejecutarlo, interpretarlo y enviarlo al cliente (navegador) en forma de código HTML.

Las páginas ASP comienzan a ejecutarse cuando un usuario solicita un archivo asp al servidor Web a través del explorador. El servidor web llama a ASP, que lee el archivo solicitado, ejecuta las secuencias de comandos que encuentre y envía los resultados al explorador del cliente.

Puesto que las secuencias de comandos se ejecutan en el servidor, y NO en el cliente, es el servidor el que hace todo el trabajo necesario para generar las páginas que se envían al explorador. Las secuencias de comandos quedan ocultas a los usuarios, estos solo reciben el resultado de la ejecución en formato HTML.

Desaparece por tanto el problema de si el cliente puede o no ejecutar sentencias de comandos, el servidor Web solo envía el resultado en código HTML standard interpretable por cualquier explorador.

Los archivos .ASP son archivos de texto normales, no es necesario ningún editor especial para crearlos, puede usarse cualquier editor que genere código ASCII.

Un archivo .ASP puede contener texto, código HTML, código ASP o cualquier combinación de estos. Si no contiene código ASP se comporta como un archivo .html normal.

2.3.3 MySQL

MySQL es un servidor de bases de datos relacionales muy rápido, multiusuario y multihilo(multitarea), usado sobre todo en Internet en conjunción con PHP. Es software libre (licencia GPL) y es mantenido por la compañía sueca MySQL AB.

La principal herramienta de MySQL es mysqladmin, la cuál como parece indicar su nombre es la encargada de la administración.

Todo el sistema de permisos de acceso al servidor, a las bases de datos y sus tablas, MySQL lo almacena en una tabla llamada mysql, es donde se guardarán todos los permisos y restricciones a los datos de nuestras bases de datos, la cual se componen de cinco tablas: host, user, db, tables_priv, columns_priv.

La tabla *user* contiene información sobre los usuarios, desde que máquinas pueden acceder a nuestro servidor MySQL, su clave y de sus diferentes permisos.

La tabla *host* nos informa sobre que máquinas podrán acceder a nuestro sistema, así como a las bases de datos que tendrán acceso y sus diferentes permisos.

Finalmente, las tablas `db`, `tables_priv`, `columns_priv` nos proveen de un control individual de las bases de datos, tablas y columnas (campos).

En el directorio `/benc` encontraremos ejemplos de script y SQL. En el directorio `/share` están los mensajes de error del servidor para los distintos idiomas. Los directorios `/include` y `/lib` contiene los ficheros `*.h` y las librerías necesarias, en `/bin` están los ficheros ejecutables y en `/data` encontraremos como subdirectorio cada una de las bases de datos que hayamos creado.

Para cada base de datos que nosotros creamos, MySQL crea un directorio con el nombre que le hemos asignado a la base de datos. Dentro de este directorio, por cada tabla que definamos MySQL va a crear tres archivos: `mitabla.ISD`, `mitabla.ISM`, `mitabla.frm`.

El archivo con extensión `ISD`, es el que contiene los datos de nuestra tabla, el `ISM` contiene información acerca de las claves y otros datos que MySQL utiliza para buscar datos en el fichero `ISD`. Y el archivo `frm` contiene la estructura de la propia tabla.

2.3.4 PHP

PHP es un lenguaje sencillo, de sintaxis cómoda y similar a la de otros lenguajes como C o C++, además es un software libre. PHP es una tecnología del lado del servidor, que funciona embebida (es decir, incrustada) dentro del código HTML de una página, dándole mayor dinamismo a la

misma, con acceso a bases de datos, creación de foros, libros de visita, rotación de banners, etc. Su sintaxis es heredada de C/Java y posee gran cantidad de funciones que permiten realizar todas las acciones que soporta el PHP.

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje "Open Source" ("código abierto") interpretado de alto nivel, especialmente pensado para desarrollos web y el cual puede ser embebido en páginas HTML y ejecutado en el servidor. La mayoría de su sintaxis es similar a C, Java y Perl y es fácil de aprender. La meta de este lenguaje es permitir escribir a los creadores de páginas web, páginas dinámicas de una manera rápida y fácil, aunque se pueda hacer mucho más con PHP.

Lo que distingue a PHP de la tecnología Javascript, la cual se ejecuta en la máquina cliente, es que el código PHP es ejecutado en el servidor. El servidor web puede ser incluso configurado para que procese todos los archivos HTML con PHP.

Existen tres campos en los que se usan scripts escritos en PHP.

Scripts del lado del servidor. Este es el campo más tradicional y el principal foco de trabajo. Se necesitan tres cosas para que esto funcione.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluyendo HP-UX, Solaris y

OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI.

De modo que, con PHP tiene la libertad de elegir el sistema operativo y el servidor de su gusto. También tiene la posibilidad de usar programación procedimental o programación orientada a objetos. Aunque no todas las características estándar de la programación orientada a objetos están implementadas en la versión actual de PHP, muchas bibliotecas y aplicaciones grandes (incluyendo la biblioteca PEAR) están escritas íntegramente usando programación orientada a objetos.

Con PHP no se encuentra limitado a resultados en HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y archivos XML.

Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir un interfaz vía web para una base de datos es una tarea simple con PHP. Las siguientes bases de datos están soportadas actualmente:

Adabas D	Ingres	Oracle (OCI7 and OCI8)
dBase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave	Direct MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

También cuenta con una extensión DBX de abstracción de base de datos que permite usar de forma transparente cualquier base de datos soportada por la extensión. Adicionalmente, PHP soporta ODBC (el Estándar Abierto de Conexión con Bases de Datos), así que puede conectarse a cualquier base de datos que soporte tal estándar.

2.3.5 FireWorks

Macromedia Fireworks MX es la forma más fácil de crear, optimizar y exportar gráficos interactivos en un entorno único y centrado en el web.

Macromedia Fireworks MX tiene las herramientas familiares que exigen los profesionales que hacen gráficos, reunidas en un solo entorno centrado en el web. Crea rápidamente gráficos web originales e interactivos, desde sencillos botones gráficos hasta complejos efectos de rollover y menús emergentes. Edita con facilidad e integra perfectamente archivos fuente en todos los principales

formatos gráficos, y exporta a proyectos de Macromedia Flash™ y Dreamweaver®. Fireworks MX cuenta con un juego completo de herramientas gráficas con un flujo de trabajo que promueve el trabajo en equipo y aumenta la productividad.

Fireworks MX 2004 permite que los usuarios importen archivos de todos los principales formatos gráficos y manipulen las imágenes vectoriales y de mapas de bits para crear gráficos e interactividad con rapidez. Las imágenes pueden exportarse fácilmente a Dreamweaver, Flash y aplicaciones de terceros.

2.3.6 Adobe Acrobat Professional

Acrobat Professional es un paquete que incluye diferentes utilidades para poder crear y modificar documentos PDF. La pieza clave es el *Acrobat*, un programa con una interfaz semejante a Acrobat Reader, pero con más opciones para retocar los contenidos, añadir índices y miniaturas, y otras opciones. *Acrobat Distiller* y *PDF writer*, también parte de Acrobat

En general, PDF Writer es más simple en su manejo, y permite también la incrustación de fuentes.

Adobe Acrobat Reader es un programa gratuito que nos permite leer o imprimir documentos en formato PDF. El acrónimo de Portable Document Format identifica a un tipo de documentos muy especial, aunque poco conocido y utilizado, en relación a sus características:

- Es multiplataforma: Un documento creado en una computadora macintosh puede ser visualizado e impreso en una PC con Windows, o con el sistema operativo OS2 o Unix. Y viceversa.
- Mantiene el aspecto exacto del documento, incluyendo las fuentes Tipográficas. No es necesario que estén instaladas en la computadora donde se mira el documento.
- Los documentos son de tamaño muy reducido. Mucho más pequeños que los documentos originales antes de ser pasados a formato PDF.
- Permite establecer protección para el contenido del documento, evitando por ejemplo, que pueda ser modificado.

2.3.7 Archivos de formato PDF

PDF es un formato portátil para documentos (Portable Document Format) desarrollado por Adobe Systems y muy usado en Internet debido a su versatilidad, facilidad de uso y tamaño pequeño.

Según se pregona extensamente en la Red, el formato de archivo PDF se ha convertido en el estándar para la distribución de documentación, tanto en intranets empresariales como en la Web, y parece igualmente un correcto método de distribución de archivos en pre-impresión, son muy populares en la red por varias razones:

- Un documento PDF tiene la misma apariencia, color, tipo de imprenta, gráficos y formato que un documento impreso.
- Los archivos PDF se pueden ver utilizando el navegador mismo o se pueden almacenar en la computadora para uso o impresión posterior.
- El programa Lector Acrobat (Acrobat Reader) se puede obtener gratis para la mayoría de los sistemas operativos.
- Si bien el programa Acrobat Reader no se puede usar para editar (modificar) un documento PDF, permite copiar texto del documento a otro archivo, y también efectuar búsquedas para localizar una palabra o texto.
- Pueden distribuirse por toda la Web, o mediante e-mails, o estar en CDs; pero este tipo de archivos es muy utilizado a la hora de compartir información gráfica o de texto, como por ejemplo, contratos, manuales, y hasta e-books.

2.3.8 I-Filter

Está diseñado para usuarios y administradores que desean indexar documentos Adobe PDF. Este permite a los usuarios realizar fácilmente búsquedas de texto.

Dentro de los beneficios tenemos:

- Está integrado con el sistema operativo.
- Provee una solución fácil para buscar en documentos Adobe PDF localizados en su computadora, en la red de la empresa y en la intranet de la empresa.

- Incrementa potencialmente la habilidad para localizar exactamente la información deseada.

Adobe PDF IFilter 6.0 soporta todas las versiones de archivos PDF, incluyendo aquellos creados Adobe Acrobat 6.0 (PDF 1.5).

IFilter es un filtrador de texto de Microsoft. Windows PDF IFilter 6.0 requiere uno de las siguientes ambientes:

- Microsoft® Windows® 2000 Professional with Service Pack 2
- Microsoft Windows XP Professional with Service Pack1.
- Microsoft Windows 2000 Server with Service Pack 3
- Microsoft Windows 2003 Server

2.3.9 CSS Cascading Style Sheets

Es un simple mecanismo para adicionar estilos a un formulario de página web, por ejemplo fondos, colores, espaciados, etc. Uno de los tipos fundamentales de CSS es los estilos de hojas en cascada, permite tener un estilo personal de hojas, lo que posibilita imponerse con éxito a diseñadores con experiencia.

Es utilizado para definir las reglas que tendrán los diferentes estilos, a fin de que no exista confusión.

El diseño de estilos de hojas es simple. Solamente necesita conocer los títulos HTML y algunas terminologías

básicas de publicidad, por ejemplo: para ver el texto de color azul, usted puede definirlo de la siguiente manera:

```
H1 { color: blue }
```

Ejemplo:

```
<HTML>
  <HEAD>
    <TITLE>title</TITLE>
    <LINK REL=STYLESHEET TYPE="text/css"
      HREF="http://style.com/cool" TITLE="Cool">
    <STYLE TYPE="text/css">
      @import url(http://style.com/basic);
      H1 { color: blue }
    </STYLE>
  </HEAD>
  <BODY>
    <H1>Headline is blue</H1>
    <P STYLE="color: green">While the paragraph is
green.
  </BODY>
</HTML>
```

Otro ejemplo:

```
H1 {
  font-weight: bold;
  font-size: 12pt;
  line-height: 14pt;
  font-family: helvetica;
  font-variant: normal;
  font-style: normal;
}
```

2.4 Internet Information Server

IIS engloba un conjunto de herramientas destinadas al control de servicios de Internet como el Web, FTP, correo y servidores de noticias. Además incluye el soporte necesario para la creación de páginas dinámicas en el servidor mediante el lenguaje ASP.

Los Servicios de Internet Information Server (IIS) simplifican la publicación de la información en Internet o en la intranet. IIS incluye una amplia gama de funciones

administrativas para controlar sitios Web y el servidor Web. Con funciones de programación como páginas Active Server (ASP), puede crear e implementar aplicaciones Web flexibles y escalables.

2.5 Index Server de Microsoft Windows

Index Server de Microsoft es un indexador completo de texto y un motor de búsqueda para Internet Information Server y Windows NT Server.

El servicio de Index Server es un servicio que extrae la información de un conjunto de documentos y la organiza de modo que sea más rápido y sencillo el acceso a ella mediante la función de búsqueda de Windows[®]XP, el formulario de consulta de Index Server o un explorador de Web. La información puede incluir texto contenido en un documento (su *contenido*) y las características y parámetros del documento (sus *propiedades*), como el nombre del autor. Una vez que se ha creado el índice, puede buscar o consultar el índice para obtener los documentos que contienen las palabras clave, frases o propiedades.

Index Server está diseñado para ser usado por un sólo servidor o en una Intranet o en Internet. Puede manejar una gran cantidad de consultas y proporciona actualizaciones y ayuda automáticas para los documentos de Microsoft Office.

Los requerimientos son Windows NT Server 4.0 o Windows NT Workstation 4.0, además necesita Internet Information Server.

Index Server es capaz de indexar información de texto en cualquier tipo de documento a través de filtros. Los filtros son proporcionados por HTML, texto y documentos de Microsoft Office.

Se podría buscar por ejemplo por autor, título, subtítulo o por otra frase, estas propiedades son además conocidas como propiedades de documentos OLE. Por ejemplo, un usuario podría buscar en un servidor todos los documentos que el o ella escribió, si la localización de ese documento ha sido olvidada. Igualmente, un usuario que conoce el autor de un artículo de periódico, pero no conoce su título, podría buscar por los artículos de periódico escritos por tal autor.

Index Server le permite restringir las consultas construidas a través de la combinación del número de "test".

Las restricciones disponibles para usar en alguna combinación incluye: Búsqueda de palabras y frases. Búsqueda por aproximación de palabras o frases son propiedades textuales. Por ejemplo buscar por una palabra que aparece en un resumen del documento. Buscar palabras o frases dentro de tipos de documentos específicos, como dentro de una hoja de Microsoft Excel o en una presentación de Microsoft PowerPoint. Se podrían usar parámetros como >, <=",">, > junto a una constante, como una fecha o un

tamaño de archivo. O podría usarse operadores Bolean, AND, OR, y AND NOT. Los meta caracteres como son "*", "?", y expresiones regulares también son permitidos.

2.6 Conclusiones

El software con el que trabajaremos es un su mayoría un software libre, lo que permitirá que Hemerotecas medianas y pequeñas puedan implementar este proyecto con una reducción significativa en costos.

En este capítulo hemos explicado brevemente las características del Hardware y el software que vamos a utilizar en el desarrollo de nuestra aplicación como son:

- Un diseñador de páginas Web, como es el **Dreamweaver MX 2004**
- Un entorno para crear y ejecutar aplicaciones web, **ASP**
- Un software que permita la creación de animaciones vectoriales como es **FireWorks**
- Un manejador muy sencillo de Base de Datos como es el **MySql**.
- **PHP**, un lenguaje "Open Source" ("código abierto") interpretado de alto nivel, especialmente pensado para desarrollos web y el cual puede ser embebido en páginas HTML, ejecutado en el servidor y que permita crear páginas dinámicas de una manera rápida y fácil.

- **IFilter**, un filtrador de texto de Microsoft, que indexa documentos Adobe PDF. Este permite a los usuarios realizar fácilmente búsquedas de texto.
- **CSS**, estilos de hojas en cascada, que permite tener un estilo personal de hojas en la página Web.
- Los Servicios de **Internet Information Server (IIS)** simplifican la publicación de la información en Internet o en la intranet. IIS incluye una amplia gama de funciones administrativas para controlar sitios Web y el servidor Web. Con funciones de programación como páginas Active Server (ASP), puede crear e implementar aplicaciones Web flexibles y escalables.
- Y un indexador completo de texto y motor de búsqueda, como es **Index Server de Microsoft Windows**.
- Además hemos realizado un análisis de los requisitos básicos de la configuración del hardware para desarrollo de la aplicación.

CAPÍTULO III

DISEÑO DE LA ESTRUCTURA DE ARCHIVOS

CAPÍTULO III

DISEÑO DE LA ESTRUCTURA DE ARCHIVOS

3.1 Introducción

Nuestro sistema está basado en el Index Server que es parte del Internet Information Server propio de los Sistemas Operativos, Windows NT, 2000 y XP.

En este capítulo definiremos la relación que tiene nuestro proyecto con el Index Server, las características de una base de datos de archivos de texto, además estableceremos el Diccionario de datos utilizado por el Index Server, las propiedades de los archivos de las que nos valdremos para identificarlos en la base de datos de archivos y la estructura de los directorios donde se encontrarán los archivos, que contienen las noticias, a partir de los cuales se realizará la búsqueda.

3.2 Bases de datos de texto no estructurado

Las bases de datos de texto no estructurado, como su nombre lo indica, no tienen estructura definida, ninguna parte del contenido tiene más importancia que otra. No contienen palabras clave que permitan diferenciar unas partes de otras en un documento. Es lo más parecido a un artículo periodístico, en donde la cadena del título se ha perdido.

Un ejemplo de este tipo de bases de datos son los documentos de texto plano, como los que están escritos en editores de texto, para recuperar este tipo de documentos sólo podemos hacerlo con el nombre del archivo. Otro claro ejemplo de bases de datos no estructuradas son los repositorios de información que se describirán a continuación.

3.2.1 Repositorios de información

Los repositorios de información almacenan grandes volúmenes de datos que cambian constantemente, no están estructurados, así que los datos y toda la información que almacenan, no se puede estructurar como en una base de datos tradicional, dado que es información volátil.

Esta manera de almacenar datos, sin estructura ni formato, obliga a que las búsquedas se realicen utilizando minería de datos, estadística o inteligencia artificial. Algunas herramientas para navegar en repositorios y recuperar documentos son los buscadores en WWW (World Wide

Web), como altavista, lycos, hotbot. En ambientes locales, y con dominios acotados, se utilizan programas como ``pebbles" que es un programa de datos.

Como la información almacenada en los repositorios no tiene estructura, (no cuentan con una organización de campos, registros, ni orden alfabético, o numérico), tampoco se puede contar con la seguridad de poder recuperar objetos o documentos ahí almacenados con anterioridad.

3.3 Características del Motor de Búsqueda

El motor de búsqueda nuestro está basado en el Internet Information Server y el Index Server.

INDEX SERVER interactúa con IIS de modo que pueda buscar información en un sitio Web mediante un formulario de consulta. Debido a su estrecha interacción con IIS, Index Server toma prestado el modelo que IIS utiliza para consultar un origen de datos de Open Database Connectivity (ODBC, Conectividad abierta de bases de datos).

Los Servicios de Internet Information Server de Microsoft (IIS) 4.0 y el Servicio de Index Server 2.0 (ambos forman parte del Microsoft Windows NT 4.0 Option Pack) se combinan para proporcionar el filtrado y la búsqueda de propiedades así como la indexación de texto y la búsqueda de datos de archivos.

Se dispone de todas estas capacidades de manera completamente independiente de SQL Server. En concreto, hay al menos dos formas de búsqueda que no usan SQL Server. Una de éstas utiliza un lenguaje de consulta específico del Servicio de Index Server; la otra es compatible con las consultas basadas en SQL dentro de ActiveX® Data Objects (ADO).

3.3.1 Catálogos de Búsqueda utilizados por el Index Server

El Servicio de Index Server almacena índices y valores de propiedades en un catálogo de búsqueda de texto. De manera predeterminada, se crea un catálogo de búsqueda de texto denominado Web cuando se instala el Servicio de Index Server. Se puede especificar más de un catálogo de búsqueda de texto pero este documento se limita al uso de Web y no tratará el proceso utilizado para crear catálogos adicionales de búsqueda de texto.

Un catálogo de búsqueda de texto determinado hace referencia a uno o más directorios virtuales de IIS. Un directorio virtual hace referencia a uno o más directorios físicos y, opcionalmente, a otros directorios virtuales. Una vez que un archivo real está vinculado al catálogo de búsqueda de texto por medio de un directorio virtual, se notifica al Servicio de Index Server cualquier nuevo archivo que tenga que indexarse y comienza el filtrado y la indexación de las propiedades y del contenido relacionados con esos archivos. También se notifican al Servicio de Index

Server los cambios posteriores en los archivos para que vuelva a filtrar y a indexar todos los archivos actualizados.

3.3.2 Características básicas de las consultas

Estas son las características básicas de una consulta:

- Alcance
- Restricción
- Conjuntos de Resultados

El **alcance** indica al motor de consulta dónde debe mirar cuando esté buscando. Describe el conjunto de documentos pertenecientes al cuerpo que se explorará. El alcance de una consulta especifica el conjunto de documentos que se deben explorar. Los alcances se suelen especificar mediante una ruta de acceso de directorio de un volumen de almacenamiento, como D:\Docs.

Index Server indexa los documentos según los sitios.

La **restricción** comprueba si se debe devolver un documento. Una restricción es un conjunto de condiciones que se pueden combinar mediante varios operadores.

Además de consultar el contenido, se pueden consultar propiedades almacenadas sobre los objetos. Estas propiedades incluyen el tamaño del archivo, la fecha de creación y de modificación, los nombres de los archivos, los autores, etc.

En las consultas puede utilizar los operadores de comparación estándar, estos incluyen =, >, <, >=, <= y != (no

igual que) para las propiedades numéricas y de texto. Además, para las propiedades de texto está disponible toda la funcionalidad de las consultas de contenido. Con los operadores booleanos (**AND**, **OR** y **NOT**) y los paréntesis, puede combinar libremente términos de restricciones.

3.3.2.1. Consultas de lógica borrosa

Index Server admite *consultas de lógica borrosa*, que contienen comodines simples (como los de MS-DOS®), y busca coincidencias de expresiones regulares contra propiedades de texto. Las consultas de contenido admiten coincidencias de un prefijo simple (por ejemplo, *per** devolverá *perro* y *percusión*). Index Server admite también desinencias gramaticales, que coinciden con las raíces y las formas derivadas de las palabras de la consulta. (Por ejemplo, *amar*** se expande a *amando*, *amado*, *amaba*, etc.) Aunque Index Server no implementa un auténtico procesamiento del lenguaje natural, sí admite el modo de texto libre.

El **conjunto de resultados** define la información que se debe devolver de una consulta. Index Server recopila las referencias encontradas de la consulta en conjuntos de resultados, que se devuelven. Se puede limitar el número máximo de referencias encontradas devueltas al cliente.

Además de las características básicas, hay otras características que permiten controlar la forma en que se devuelven y se muestran los resultados; por ejemplo, puede

especificar cómo se ordenarán los resultados. También puede:

- Limitar la consulta a alcances específicos.
- Buscar palabras y frases en el contenido del documento.
- Buscar palabras o frases que estén cerca de otra palabra o frase.
- Buscar palabras y frases con propiedades de texto (por ejemplo, @DocAuthor Rosa).
- Buscar propiedades con <, <=, =, =>, > contra una constante (por ejemplo, DATE > 01/01/05).
- Aplicar los operadores booleanos **AND**, **OR** y **NOT**.
- Buscar con comodines (por ejemplo, "*", "?" y expresiones regulares).
- Integrar completamente las búsquedas con el modelo de seguridad de Windows NT.
- Ordenar las referencias encontradas por cualidades.
- Devolver datos especificados de propiedades.

3.3.3 Registro

IIS registra todo el tráfico entre un cliente y el servidor. El registro estándar de IIS almacena la información de las consultas, como la dirección IP de la consulta y las consultas enviadas al servidor.

3.3.4 Caracteres comodín

Los operadores comodín ayudan a buscar páginas que contengan palabras similares a una palabra dada.

Para buscar	Ejemplo	Resultados
Palabras con el mismo prefijo	<i>ord*</i>	Las páginas con palabras que tengan el prefijo <i>ord</i> , como <i>orden</i> , <i>ordenación</i> , etc.
Palabras basadas en la misma raíz	<i>volar**</i>	Las páginas que contienen palabras basadas en la misma raíz que <i>volar</i> , como <i>volando</i> , <i>volado</i> , <i>voló</i> , etc.

Figura 3.1. Caracteres comodines.

3.3.5 Consultas de texto libre

El motor de consulta busca las páginas que mejor coincidan con las palabras y las frases contenidas en una consulta de texto libre. Para ello, se buscan automáticamente las páginas que coincidan con el significado de la consulta, no con la cita textual. En una consulta de texto libre, se pasan por alto los operadores booleanos y de proximidad y los caracteres comodín. Las consultas de texto libre tienen el prefijo *\$contents*.

Para buscar	Ejemplo	Resultados
Archivos que coincidan con el texto libre	<i>\$contents ¿cómo puedo imprimir en Microsoft Excel?</i>	Las páginas que mencionan la impresión y Microsoft Excel.

Figura 3.2. Consultas de texto libre.

3.3.6 Nombres de propiedades

Los nombres de las propiedades se escriben precedidos del carácter @ o por el signo de número (#).

Utilice @ para las consultas relacionales y # para las consultas de expresión regular.

Si no se especifica un nombre de propiedad, se asume @contents.

Las propiedades disponibles para los archivos incluyen:

Nombre de propiedad	Descripción
All	Coincide con palabras, con frases y con cualquier propiedad.
Contents	Palabras y frases contenidas en el archivo.
Filename	Nombre del archivo.
Size	Tamaño del archivo.
Write	Última vez que se modificó el archivo.

Figura 3.3. Nombre de propiedades de los archivos.

3.4 Diccionario de datos.

El diccionario de datos está basado en las propiedades del Index Server. Estas propiedades siempre están disponibles para las consultas. También puede haber propiedades adicionales disponibles según la configuración del servidor Web, las cuales describiremos a continuación:

Tabla de nombres de propiedades

Nombre descriptivo	Tipo de datos	Propiedad
A_HRef	DBTYPE_WSTR DBTYPE_BYREF	Texto de HTML HREF. Este nombre de propiedad se creó para Microsoft® Site Server y corresponde a la propiedad HtmlHRef de Index Server. <i>Se puede consultar pero no se puede recuperar.</i>
Access	VT_FILETIME	La última vez que se tuvo acceso al archivo.
All	(no aplicable)	Busca cada una de las propiedades de una cadena. <i>Se puede consultar pero no se puede recuperar.</i>
AllocSize	DBTYPE_I8	El tamaño de la asignación de disco del archivo.
Attrib	DBTYPE_UI4	Los atributos del archivo. Documentada en el SDK de Win32.
ClassId	DBTYPE_GUID	El Id. De clase del objeto, por ejemplo, WordPerfect, Word, etc.
Characterization	DBTYPE_WSTR DBTYPE_BYREF	La caracterización o resumen del documento. Efectuada por Index Server.
Contents	(no aplicable)	El principal contenido del archivo. <i>Se puede consultar pero no se puede recuperar.</i>
Create	VT_FILETIME	La hora en que se creó el archivo.
Directory	DBTYPE_WSTR DBTYPE_BYREF	La ruta física de acceso al archivo, sin incluir el nombre de archivo.
DocAppName	DBTYPE_WSTR DBTYPE_BYREF	El nombre de la aplicación que creó el archivo.
DocAuthor	DBTYPE_WSTR DBTYPE_BYREF	El autor del documento.
DocByteCount	DBTYPE_14	El número de bytes de un documento.
DocCategory	DBTYPE_STR DBTYPE_BYREF	El tipo de documento, como un memorando, una

		programación o unas notas de la aplicación.
DocCharCount	DBTYPE_I4	El número de caracteres del documento.
DocComments	DBTYPE_WSTR DBTYPE_BYREF	Los comentarios acerca del documento.
DocCompany	DBTYPE_STR DBTYPE_BYREF	El nombre de la empresa para la que se escribió el documento.
DocCreatedTm	VT_FILETIME	La hora en que se creó el documento.
DocEditTime	VT_FILETIME	El tiempo total empleado en modificar el documento.
DocHiddenCount	DBTYPE_14	El número de diapositivas ocultas de un documento de Microsoft® PowerPoint.
DocKeywords	DBTYPE_WSTR DBTYPE_BYREF	Las palabras clave del documento.
DocLastAuthor	DBTYPE_WSTR DBTYPE_BYREF	El último usuario que modificó el documento.
DocLastPrinted	VT_FILETIME	La hora en que se imprimió el documento por última vez.
DocLastSavedTm	VT_FILETIME	La hora en que se guardó el documento por última vez.
DocLineCount	DBTYPE_14	El número de líneas contenidas en un documento.
DocManager	DBTYPE_STR DBTYPE_BYREF	El nombre del administrador del autor del documento.
DocNoteCount	DBTYPE_14	El número de páginas con notas de un documento de PowerPoint.
DocPageCount	DBTYPE_I4	El número de páginas de un documento.
DocParaCount	DBTYPE_14	El número de párrafos de un documento.
DocPartTitles	DBTYPE_STR DBTYPE_VECTOR	Los nombres de las partes del documento. Por ejemplo, en los títulos de las partes de un documento de Excel son los nombres de las hojas de cálculo; en los títulos de las diapositivas de PowerPoint y en los documentos de Word para

		Windows son los nombres de los documentos incluidos en el documento maestro.
DocPresentation Target	DBTYPE_STR DBTYPE_BYREF	El formato de destino (35mm, impresora, vídeo, etc.) de una presentación de PowerPoint.
°DocRevNumber	DBTYPE_WSTR DBTYPE_BYREF	El número de versión actual del documento.
DocSlideCount	DBTYPE_I4	El número de diapositivas de un documento de PowerPoint.
DocSubject	DBTYPE_WSTR DBTYPE_BYREF	El tema del documento.
DocTemplate	DBTYPE_WSTR DBTYPE_BYREF	El nombre de la plantilla del documento.
DocTitle	DBTYPE_WSTR DBTYPE_BYREF	El título del documento.
DocWordCount	DBTYPE_I4	El número de palabras del documento.
FileIndex	DBTYPE_I8	El Id. único de un archivo.
FileName	DBTYPE_WSTR DBTYPE_BYREF	El nombre del archivo.
HitCount	DBTYPE_I4	El número de coincidencias (las palabras coincidentes con la consulta) encontradas en el archivo.
HtmlHRef	DBTYPE_WSTR DBTYPE_BYREF	El texto de HTML HREF. <i>Se puede consultar pero no se puede recuperar.</i>
HtmlHeading1	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H1. <i>Se puede consultar pero no se puede recuperar.</i>
HtmlHeading2	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H2. <i>Se puede consultar pero no se puede recuperar.</i>
HtmlHeading3	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H3. <i>Se puede consultar pero no se puede recuperar.</i>
HtmlHeading4	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H4. <i>Se puede consultar pero no se puede recuperar.</i>
HtmlHeading5	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H5. <i>Se</i>

		<i>puede consultar pero no se puede recuperar.</i>
HtmlHeading6	DBTYPE_WSTR DBTYPE_BYREF	El texto de un documento HTML en el estilo H6. <i>Se puede consultar pero no se puede recuperar.</i>
Img_Alt	DBTYPE_WSTR DBTYPE_BYREF	Texto alternativo para etiquetas . <i>Se puede consultar pero no se puede recuperar.</i>
Path	DBTYPE_WSTR DBTYPE_BYREF	La ruta completa de acceso al archivo, incluido el nombre de archivo.
Rank	DBTYPE_I4	La jerarquía de la fila. El intervalo es de 0 a 1000. Los números más grandes indican las mejores coincidencias.
RankVector	DBTYPE_I4 DBTYPE_VECTOR	Las jerarquías de los componentes individuales de una
ShortFileName	DBTYPE_WSTR DBTYPE_BYREF	El nombre breve (8.3) del archivo.
Size	DBTYPE_I8	El tamaño del archivo en bytes.
USN	DBTYPE_I8	El Número de secuencia de actualización (USN). Sólo para las unidades NTFS.
VPath	DBTYPE_WSTR DBTYPE_BYREF	La ruta de acceso virtual completa del archivo, incluido el nombre de archivo. Si hay más de una ruta de acceso posible, se elige la que mejor coincida con la consulta específica.
WorkId	DBTYPE_I4	El Id. interno del archivo. Utilizado en Index Server.
Write	VT_FILETIME	La última vez que se escribió el archivo.

Figura 3.4. Tabla de nombre de propiedades del Index Server.

3.5 Estructura de Directorios.

La estructura de los directorios estaría definida de la siguiente manera:

- Tipo de periódico: Nacional o Internacional
- Nombre del periódico
- Año de la noticia
- Mes de la noticia
- Día de la noticia
- Sección

Dentro de los nombres de periódicos tendríamos los siguientes:

- Mercurio - MER
- Tiempo - TIE
- Hoy - HOY
- Universo - UNI
- Comercio - COM

El año de la noticia sería de 4 dígitos. Ejemplo. 2005

El mes de la noticia sería de 2 dígitos. Ejemplo. 05

El día de la noticia sería de 2 dígitos. Ejemplo. 10

Dentro de las Secciones, tendríamos:

- Economía - ECO
- Política - POL
- Cultura - CUL

- Sociedad - SOC
- Iglesia - IGL
- Farándula - FAR
- Deportes - DEP

Ejemplo: NMER20050510ECO

3.6 Nombre de Archivos

El nombre de los Archivos estaría definido de la siguiente manera.

El primer dígito el tipo de periódico: Nacional o Internacional.

Los 3 siguientes caracteres lo componen el nombre del periódico:

Los 4 dígitos del año

Los 2 dígitos del mes

Los 2 dígitos del día

Los 3 primeros caracteres de la Sección

Los 9 dígitos para la secuencia de la noticia.

Ejemplo: MER20050510ECO000000001

3.7 Base de Datos para Administración de Archivos

Con la finalidad de administrar los archivos escaneados de los periódicos se creó una pequeña base de datos en MySql, la cual tiene la finalidad de guardar la

información más importante del archivo y posteriormente permitirá realizar el mantenimiento de la misma. Así como también restringirá el acceso a los usuarios que almacenarán la información.

Nuestra Base de datos se llama Biblioteca, la cual contiene 4 tablas:

- administrador
- información
- periodicos
- secciones

Administrador: En esta tabla se almacenarán los usuarios y sus claves que administrarán los archivos de periódicos:

Column Name	Datatype
 codigo_admin	 INTEGER
 user_admin	 VARCHAR(45)
 passw_admin	 VARCHAR(45)

Figura 3.5. Tabla de Usuarios.

codigo_admin: Es un campo numérico secuencial.

User_admin: Es el campo para identificar al usuario.

Passw_admin: Es la clave de acceso del usuario

Información: Aquí se registrarán los datos relacionados con cada uno de los archivos de los periódicos.

Column Name	Datatype
codigo_inf	INTEGER
periodico_inf	VARCHAR(45)
tipo_inf	VARCHAR(45)
fecha_inf	DATE
seccion_inf	VARCHAR(45)
archivo_inf	VARCHAR(100)
titulo_inf	VARCHAR(45)

Figura 3.6. Tabla de Información de archivos.

codigo_inf: Es un campo secuencial que numera a los registros de los periódicos.

periodico_inf: Es el nombre del periódico.

tipo_inf: Identifica si la noticia es Nacional o Internacional.

fecha_inf: Guarda la fecha de la noticia.

Seccion_inf: Sección a la que pertenece la noticia.

Archivo_inf: Guarda la ruta de acceso para el archivo que contiene la noticia.

Titulo_inf: Utilizado para una descripción corta del contenido de la noticia.

Periódicos: En esta tabla se almacenarán los diferentes nombres de periódicos que serán utilizados en la Base de Datos. Está conformado por los siguientes campos:







Column Name	Datatype
 codigo_per	 INTEGER
 nombre_per	 VARCHAR(45)
 nombre_largo_per	 VARCHAR(60)

Figura 3.7. Tabla de Nombres de Periódicos.

codigo_per: Es un campo secuencial que numera a los registros de los nombres de periódicos.

nombre_per: Es el nombre corto del periódico .

nombre_largo_per: Es el nombre largo del periódico .

Secciones: Los diferentes nombres de secciones que serán utilizados en la Base de Datos se almacenarán en esta tabla.

Está conformado por los siguientes campos:





Column Name	Datatype
 codigo_sec	 INTEGER
 nombre_sec	 VARCHAR(45)

Figura 3.8. Tabla de Nombres de Secciones.

codigo_sec: Es un campo secuencial que numera a los registros de los nombres de secciones.

nombre_sec: Es el nombre de la sección.

3.8 Conclusiones.

El motor de búsqueda nuestro está basado en el Index Server. Los Servicios de Internet Information Server y el Servicio de Index Server, ambos de Microsoft, se combinan para proporcionar el filtrado y la búsqueda de propiedades así como la indexación de texto y la búsqueda de documentos o texto de los diarios o periódicos que la Hemeroteca dispone para consultas.

La forma en que opera el Index Server es a través de un catálogo de búsqueda de texto determinado. Se notifica al Servicio de Index Server cualquier nuevo archivo que tenga que indexarse y comienza el filtrado y la indexación de las propiedades y del contenido relacionados con esos archivos. También se notifican al Servicio de Index Server los cambios posteriores en los archivos para que vuelva a filtrar y a indexar todos los archivos actualizados.

Con la finalidad de administrar los archivos escaneados de los periódicos se creó una pequeña base de datos en MySQL, la cual se llama "Biblioteca", y contiene 4 tablas: administrador, información, periódicos y secciones.

CAPÍTULO IV

**ANÁLISIS Y DISEÑO DE LOS
RECURSOS DE HARDWARE Y DE
REDES REQUERIDOS PARA EL
SISTEMA**

CAPÍTULO IV

ANÁLISIS Y DISEÑO DE LOS RECURSOS DE HARDWARE Y DE REDES REQUERIDOS PARA EL SISTEMA

4.1 Introducción

En este capítulo detallaremos las especificaciones de los equipos de red y la estructura que tendrían los mismos.

Además están incluidas cotizaciones de tres experimentadas Empresas de nuestra Ciudad en el área de venta de equipos y software necesario para que la aplicación funcione correctamente y sin problemas legales.

4.2 Determinación del equipo necesario para la implementación

Para el funcionamiento del Software de digitalización de los periódicos será necesario tener una red conformada por un servidor principal y al menos tres terminales conectadas por medio de una red Lan con Topología en Estrella. Además será imprescindible el uso de un scanner A3 y una impresora.

4.3 Servidor Principal

El servidor deberá tener básicamente una gran capacidad de almacenamiento puesto que se trata de almacenamiento de archivos gráficos en formato pdf., una muy buena velocidad de procesamiento y una unidad para realizar respaldos.

A continuación se encuentran detalladas las características técnicas del servidor principal.

- Servidor HP Proliant ML 110 G2 ultra-low Storage Server
- Disco duro de 320 GB de capacidad
- Procesador Intel Celeron 2.8 GHZ, 533 MHZ FSB
- Windows storage server 2003 Express edition
- Memoria Standard 512 MB, ampliable a 4 GB, PC3200 DDR SDRAM DIMM
- Memoria de ejecución de 400 MHZ
- Tarjeta de red SCSI incluida.
- Drive de respaldo HP StorageWorks DAT 72i

4.3 Red de Computadores

Será necesario además tener una red Lan de computadores enlazados al computador principal con Topología en Estrella.

Para realizar la digitalización de la información se debería tener al menos una estación de trabajo y dos terminales para uso del público o clientes.

4.4.1 Estación de trabajo

Dentro de las características mínimas de la estación de trabajo para la subida de la información serían:

- Procesador Pentium 4
- Memoria RAM 1 GB
- Tarjeta de red Fast Ethernet 10/100 MBPS

4.4.2 Estación para el Cliente

Este computador no requerirá demasiados recursos ya que el procesamiento se le realizara en el Servidor principal, como características básicas para este computador están:

- Procesador Pentium III en adelante
- Memoria RAM 256 KB
- Tarjeta de red Fast Ethernet 10/100 MBPS

4.4.3 Unidades de entrada y salida de la información

4.4.3.1 Scanner

El ingreso de la información se lo deberá realizar por medio de un Scanner A3, debido a que se trata de páginas de periódicos grandes, en donde este tamaño de Scanner nos facilitaría para realizar la captura de más noticias a la vez.

4.4.3.2 Impresora

Además el sistema permitirá enviar la noticia a una impresora Laser, con lo que el usuario podría llevarse por escrito cualquier tema de su interés. Para conseguir este fin, la impresora deberá ser para uso común dentro de la red, es decir una impresora de red.

4.5 Cotizaciones

COTIZACIONES DE HARDWARE Y SOFTWARE						
	EMPRESA "A"		EMPRESA "B"		EMPRESA "C"	
	CARACTERÍSTICAS	PRECIO	CARACTERÍSTICAS	PRECIO	CARACTERÍSTICAS	PRECIO
SERVIDOR	HP FROLIANT ML 110 CON PROCESADOR 2.3 GHZ MEMORIA 1GB HD 160 GB 1" 7200 RPM SATA CONTROLADOR DE ARREGLO DE DISCOS INTELIGENTE	\$ 3.390	HP FROLIANT ML 110 CON PROCESADOR INTEL PENTIUM 4 DE 3.2 GHZ HD DE 160 GB SCSI MEMORIA 1GB CONTROLADOR DE ARREGLO DE DISCOS INTELIGENTE	\$ 3.393	HP ML350T04 X3.2 2800-IM CON PROCESADOR INTEL XEON 3.2 GHZ MEMORIA 1GB HD 72 GB CONTROLADOR DE ARREGLO DE DISCOS INTELIGENTE	\$ 4.688
ESTACION DE TRABAJO	CLON CON PROCESADOR INTEL PENTIUM 4 DE 2,0 MHZ MEMORIA 1GB HD 80GB ULTRA ATA	\$ 772	CLON COMPUTADOR INTEL PENTIUM 4 DE 3.2 GHZ HD DE 80 GB MEMORIA 1GB	\$ 953	HP DC5100 P43.0 GHZ HD 80GB MEMORIA 512 MB	\$ 1.010
ESTACION DEL CLIENTE	CLON CON PROCESADOR INTEL PENTIUM 4 DE 2,6 MHZ MEMORIA 512 MB HD 80GB ULTRA ATA	\$ 722	CLON CON PROCESADOR INTEL PENTIUM 4 DE 2,6 MHZ MEMORIA 512 MB HD 80GB ULTRA ATA		HP PENTIUM 4 DE 2.6 MHZ MEMORIA 512 MB HD 80 GB	\$ 900
SCANNER	A3 EPSON PHOTO IMAGING B11E1600H1600 X1200 DPI	\$ 1.490	A3 FUJITSU FI 4520C 600 DPI X 600 DPI	\$ 4.110	A3 EPSON EXPRESSION 1600 XL COLOR FLATBED RESOLUCION 2400 X 4800 DPI	\$ 2.690
IMPRESORA	LEXMARK LASER E230 RESOLUCION 600 PUNTOS POR PULGADA	\$ 130	HP LASER JET 1010	\$ 183	HP LASER JET 1160 20PP E&M 133 MHZ 1200 DPI 16MB	\$ 227
SWITCH	ENCORE 8 PORTS	\$ 22	3-COM 8 PORTS RJ45 OFFICE CONT.	\$ 67	3-COM 8 PORTS RJ45 OFFICE CONT.	\$ 54
TARJETA DE RED	ENCORE 10/100	\$ 7	3-COM 10/100	\$ 24	3-COM 10/100	\$ 30
WINDOWS PARA EL SERVIDOR CON LICENCIA PARA 5 ESTACIONES	WINDOWS SVR STD 2003 WIN32 SPANISH CD 5 CLT	\$ 1.200	WINDOWS SVR STD 2003 WIN32 SPANISH CD 5 CLT	\$ 1.200	WINDOWS SVR STD 2003 WIN32 SPANISH CD 5 CLT	\$ 1.364
SOFTWARE PARA MANIPULACION DE ARCHIVOS	ADOBE PHOTOSHOP 7.0	\$ 100	ADOBE PHOTOSHOP 7.0	\$ 110	ADOBE PHOTOSHOP 7.0	\$ 115
SOFTWARE PARA CONVERSION EN PDF CON OCR	ADOBE ACROBAT 7.0 PROFESSIONAL	\$ 450	ADOBE ACROBAT 7.0 PROFESSIONAL	\$ 576	ADOBE ACROBAT 7.0 PROFESSIONAL	\$ 580
TOTAL DEL PROYECTO		\$ 8.290		\$ 10.516		\$ 11.763

Figura 4.1. Cotizaciones de Hardware y Software

4.6 Conclusiones

Hemos definido los equipos necesarios para la implementación del Sistema de digitalización de los periódicos, de la manera más adecuada, para el ingreso de los archivos de periódicos a la base de datos y para el uso del público de la Hemeroteca.

Dentro de los recursos de Hardware hemos previsto que es necesario tener una red Lan, con Topología en Estrella, con un Servidor Principal, una o más estaciones de trabajo para la subida de la información y terminales básicas para los clientes, un scanner y una impresora.

Además hemos realizado cotizaciones en tres Empresas de venta de hardware y software de la localidad. Por lo tanto, hemos concluido que todas las empresas nos han cotizado un servidor de marca HP, adecuado para este trabajo, la variación de precios esta básicamente en el modelo del servidor de la Empresa "C", ya que esta opción tiene un modelo más actual de servidor. El resto de computadores de trabajo en las Empresas "A" y "B" son clones, pero adecuados en su capacidad para este proyecto.

La Empresa "A", además ha abaratado precios debido a que ofrece equipos más económicos con marcas no muy acreditadas en nuestro medio, pero de buena calidad. Esta pudiera ser una muy buena opción en el caso de no tener mucho presupuesto para este proyecto.

La Empresa "B", igualmente ofrece un servidor de marca, estaciones de trabajo clones, pero posee la mejor opción de Scanner, que hemos podido cotizar en nuestra

ciudad, además el resto de equipos son de marcas conocidas. Esta es una buena opción ya que posee equipos de marca y no es tan costosa.

La Empresa "C" es la opción más costosa ya que todos sus equipos son de excelentes marcas, el servidor es el más conveniente, ya que su modelo es el más actualizado, su equipo técnico es experimentado y además ofrece más garantía. Todo dependería como ya hemos mencionado del presupuesto que se tenga, para poderse inclinar por esta opción.

.

CAPÍTULO V

PROGRAMACIÓN DE LA APLICACIÓN WEB

CAPÍTULO V

PROGRAMACIÓN DE LA APLICACIÓN WEB

5.1 Introducción

En este capítulo explicaremos a detalle como fue realizada la programación de la Página Web, para cubrir las necesidades de digitalización de los periódicos. Se indicarán los estándares de desarrollo utilizado, así como un breve detalle de las herramientas utilizadas para conseguir este fin.

Además indicaremos todas las características utilizadas en el diseño de las páginas web.

5.2 Diseño de la página web

5.2.1 Diseño Gráfico

La página web fue diseñada en FireWorks, que nos permite crear rápidamente gráficos web originales e interactivos, desde sencillos botones gráficos hasta complejos efectos de rollover y menús emergentes.

COLORES : Se emplearon tonos de colores en azules y naranjas para el diseño de nuestra página. Colores fuertes para los fondos de las etiquetas, títulos e información principal. Y los para los fondos se emplearon colores pasteles con la finalidad de tener una mejor visualización y no causar cansancio visual en el usuario.

BOTONES: Se emplearon botones estilizados y con relleno simple.

LETRA: El tipo de letra utilizada es Arial 10 y 12.

5.2.2 Diseño Estructural

Se utilizó el lenguaje PHP para poder acceder a la base de datos MySql, pues en ella tenemos las tablas de mantenimiento de los usuarios y la de características básicas de los archivos de periódicos. Además con esta herramienta podemos tener mayor rapidez en las búsquedas de los archivos de noticias, ya que dichas búsquedas se las realizan en el Servidor.

Además utilizamos Dreamweaver para la creación de la página Web, ya que con esta herramienta es posible crear páginas HTML sin tener que preocuparnos por el código HTML, de esta manera, crear un documento HTML se convierte en una tarea menos parecida a programar y más parecida a maquetear.

Utilizamos una base de datos de MySql para la administración de los archivos de los periódicos y para establecer seguridades en el uso de la página de subida de información. La definición de esto ya fue explicado en el Capitulo III, numeral *3.7 Bases de Datos para Administración de Archivos*.

La página fue desarrollada a través de un componente de estilos, con la finalidad de tener una sola definición inicial de todas las características y luego únicamente utilizar esta definición que posee dichas características, obteniendo así las páginas con los mismos detalles.

5.3 Estructura del Sitio Web

En el siguiente diagrama explicaremos la estructura que tiene nuestro sitio web:

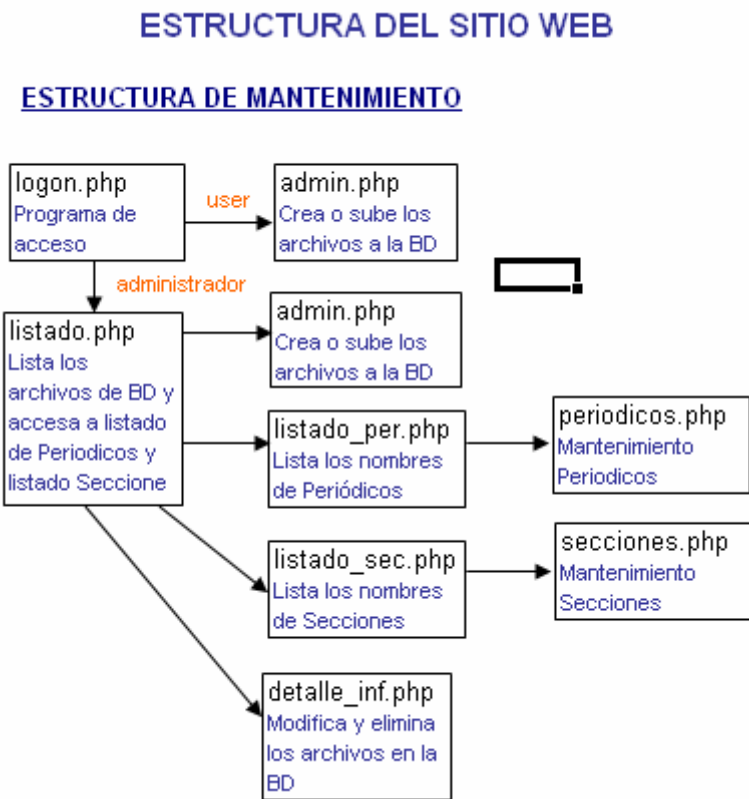


Figura 5.1. Estructura de Mantenimiento de Archivos.

ESTRUCTURA DE BÚSQUEDA



Figura 5.2. Estructura de Búsqueda de Archivos.

5.4. Interfaz de la aplicación web y el servidor

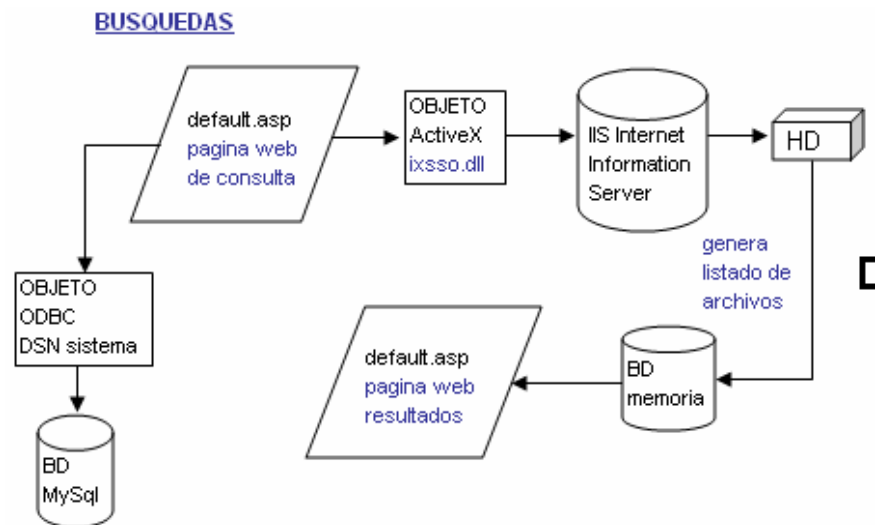


Figura 5.3. Interfaz de búsqueda de Archivos.

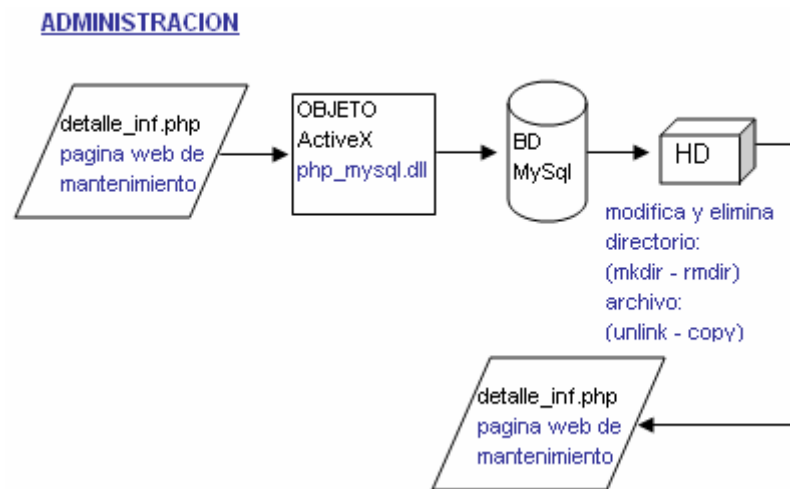


Figura 5.4. Interfaz de Administración de Archivos.

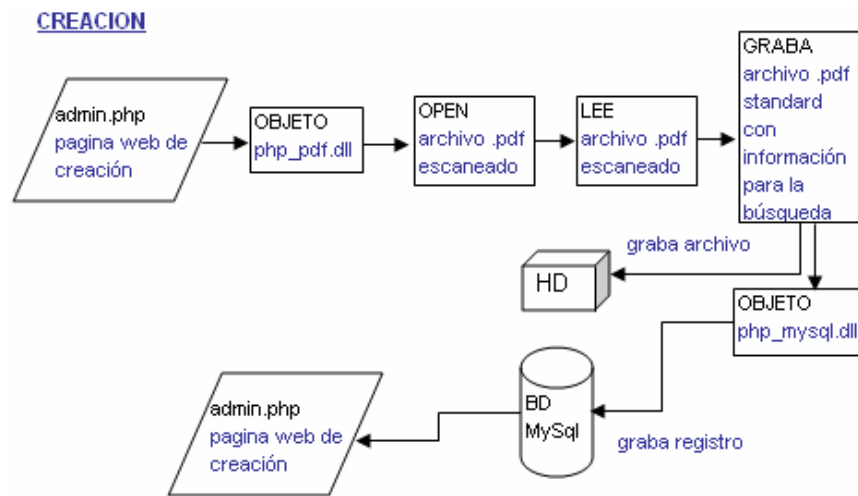


Figura 5.5. Interfaz de Creación de Archivos.

5.5 Definición de las búsquedas

Para realizar la búsqueda, en la página web se presenta una pantalla en la que se ingresa una lista o conjunto de filtros por los que el usuario desee encontrar los archivos de periódicos.

En esta pantalla se encuentran los siguientes casilleros:

Hemeroteca Virtual
Buscador de información

Buscar texto:

Tipo: -- Seleccione Tipo --

Periódico: -- Seleccione Periodico --

Seccion: -- Seleccione Sección --

Tamaño: Menor que tamaño

Fecha: tiempo

Ejecutar Limpiar

Figura 5.6. Formulario para Ingreso de las búsquedas.

Buscar Texto: Se ingresa la frase o palabra clave que mejor describa al tema que busca sin importar el uso de mayúsculas o minúsculas.

Tipo: En este casillero se puede o no seleccionar si el periódico es Nacional o Internacional para que filtre por alguno de ellos.

Periódico: Permite elegir el nombre del periódico en el que se realicen las búsquedas o puede omitirse para que busque en todos los periódicos.

Ejemplo: El mercurio, El Tiempo, El Comercio, Hoy, El Universo.

Sección: Si se desea que se filtre por alguna sección de periódico, debe realizar la selección en este casillero.

Ejemplo: Policial, Economía, Social, Política, Cultura, Finanzas, Ciudad, Desastres, Deportes, Iglesia, Farándula.

Tamaño: Se puede especificar si se desea filtrar archivos cuyos tamaños sean mayores o menores a uno digitado. En un casillero adyacente se tiene las siguientes opciones de tamaño: 100 bytes, 1K byte, 10K bytes, 100K bytes, 1M byte, 10M bytes, 100M bytes, Otro que debe ser ingresado

Fecha: Con este casillero se tiene la posibilidad de filtrar archivos cuya fecha de periódico sea mayor o igual a una fecha digitada. Además se puede elegir alguna de las siguientes opciones:

En el último día, en la última semana, en el último mes, en el último año.

Usted podría además combinar estas opciones para realizar la búsqueda de su interés.

Una vez ingresadas las palabras, se debe hacer clic sobre "Ejecutar".

El buscador mostrará el resultado en la página Web, encabezada por un breve resumen de la noticia, de los archivos que satisfacen el pedido. También aparecerán los nombres y direcciones de los primeros 10 archivos. En este momento se podrá elegir si se desea consultar alguno de esos links haciendo click en dicho link o solicitar los próximos diez archivos, como se muestra en la siguiente pantalla.

The screenshot displays the 'Biblioteca Virtual' search interface. At the top, there is a search bar with the text 'josefina' entered. Below the search bar are several filters: 'Tipo' (dropdown), 'Periódico' (dropdown), 'Sección' (dropdown), 'Tamaño' (dropdowns for 'Menor que' and 'tamaño'), and 'Fecha' (dropdown for 'tiempo'). There are 'Ejecutar' and 'Limpiar' buttons. The search results are displayed under the heading 'Resultados'. The first result is for the newspaper 'TIEMPO' with the title 'Dique Podría Reventar'. The section is 'DESASTRES' and the date is '05/07/2005 20:10:29'. The summary states: 'N. DESASTRES. La noche de ayer, a las 21 h00, se produjo un primer deslave del cerro o colina situada en la margen izquierda aguas abajo, del río Paute, en la confluencia con el Jadán, en el sector denominado "La Josefina" el cual comprende la carretera Cuenca-El Descanso Paute. El deslave continúa según las inf'. The file size is 128919 bytes. The file location is 'E:\netpub\wwwroot\BibliotecaVirtual\Periodicos\W7e\19930330\Des\Nie 19930330des20.Pdf'. The second result is for 'TIEMPO' with the title 'JOSEFINA'.

Figura 5.7. Página con resultado de la búsqueda.

5.6 Definición del Front-End o interfase con el usuario

En la construcción de la página web hemos diseñado 8 interfaces de usuario las cuales detallamos a continuación:

Logon.php: Es la interfaz de inicio al sistema de mantenimiento de los archivos de los periódicos.

Se ingresa el nombre del usuario y la clave las cuales son validadas en la base de datos.

Por razones de seguridad se crearon 2 usuarios: Administrador y User.

Administrador: Permite visualizar un listado completo de los archivos de periódicos alimentados en la base de datos, además de realizar un mantenimiento de los mismos es decir crear archivos, modificar las características principales de los archivos o eliminarlos.

User: Con este login accesa directamente a la creación de los archivos de periódicos.



Hemeroteca Virtual
Buscador de Información

Ingreso al Mantenimiento

Nombre de Usuario :

Clave :

Por favor ingrese el nombre de usuario y clave para el mantenimiento.

Figura 5.8. Página para ingreso de la clave de acceso.

Listado.php: Muestra un listado completo de los archivos de periódicos ingresados a la base de datos.

A través de esta página se puede realizar un mantenimiento de los archivos ya sea agregando nuevos “Grabar Nuevo”, modificándolos o eliminándolos de la base de datos.

Además permite crear nuevos tipos de periódicos y secciones, a través de los botones “Periódico” y “Sección”.

Hemeroteca Virtual
Buscador de información

Mantenimiento Periodico Sección Nuevo Archivo

Click en el título para Modificar el archivo o en Grabar Nuevo para generar uno nuevo.

21 ARTICULOS (3 PAGINAS) NOMBRES POR PAGINA
 < Primer << Anterior Siguiente >> Ultimo >|
 10 | 20 | 30 | 40

Codigo	Tipo	Titulo	Periodico	Fecha	Sección
2	N	AUCAS ASESINAN A 3 MISTONEROS	MERCURIO	1969-09-26	NAC
3	I	AVION VIAJA ALREDEDOR DEL MUNDO	MERCURIO	1970-06-25	SOC
4	N	BANCO DEL AZUAY JUNTA GENERAL ORDINARIA DE AC	MERCURIO	1969-07-01	ECO
5	I	400 DELINCUENTES DE GUERRA HAN SIDO YA EJECU	MERCURIO	1970-10-12	EDI
6	I	SUICIDOSE HIJA DEL ESCRITOR ESPAÑOL LUIS ARAQ	MERCURIO	1975-09-07	FAR
8	N	SOLO UN LOGRO SALVARSE MILAGROSAMENTE	MERCURIO	1959-09-26	POLICIAL
9	N	AGUA CUBRE CASAS EN LA JOSEFINA.pdf	TIEMPO	1993-04-01	DESASTRES
12	N	JOSEFINA	TIEMPO	1993-04-01	DESASTRES
13	N	PEDIDOS DE AUXILIO	TIEMPO	1993-04-04	DESASTRES
17	N	Cacerolazo	MERCURIO	2005-03-27	POLITICA

Figura 5.9. Página para listar los archivos de periódicos.

admin.php: Esta página permite crear los archivos de periódicos y tiene las características principales de la noticia.

Hemeroteca Virtual
Buscador de información

El archivo fue grabado en la ubicación solicitada.

Mantenimiento del Sistema de Archivos Nuevos

Codigo:

Titulo Noticia:

Tipo:

Periodico:

Fecha: Día Mes Año

Sección:

Archivo:

Figura 5.10. Página para grabación de los archivos de periódicos.

Detalle_inf.php Permite modificar y eliminar archivos de periódicos.

Hemeroteca Virtual
Buscador de información

Mantenimiento del Sistema de Archivos

Codigo:

Título Noticia:

Tipo:

Periódico:

Fecha: Día Mes Año

Sección:

Archivo: `periodicos/N/MER/19690926/NAC/NMER19690926NAC2.pdf`

[Modificar](#) [Eliminar](#)

Figura 5.11. Página para Mantenimiento de los archivos de periódicos.

listado_per.php Lista todos los nombres de los periódicos y le permite crear nuevos tipos de periódicos, a través del botón “Grabar Nuevo”.

Hemeroteca Virtual
Buscador de información

Mantenimiento de Periodicos [Grabar Nuevo](#)

Click en el título para Modificar el archivo

5 ARTICULOS (1 PAGINAS) NOMBRES POR PAGINA

[|< Primero](#) [<< Anterior](#) [Siguiente >>](#) [Ultimo >|](#) 10 | 20 | 30 | 40

Codigo	Nombre Corto	Nombre Largo
1	MERCURIO	EL MERCURIO
2	TIEMPO	EL TIEMPO
3	COMERCIO	EL COMERCIO
4	HOY	EL HOY
5	UNIVERSO	EL UNIVERSO

Figura 5.12. Página para listar los periódicos.

periodicos.php Realiza el mantenimiento de los registros de los periódicos.

Hemeroteca Virtual
Buscador de información

Mantenimiento de Periodicos

Codigo:

Nombre Corto:

Nombre Largo:

[Grabar](#) [Modificar](#) [Eliminar](#)

Figura 5.13. Página para mantenimiento de los nombres de Periódicos.

listado_sec.php Lista las Secciones y permite crear nuevas secciones, a través del botón “Grabar Nuevo”.

Hemeroteca Virtual
Buscador de información

Mantenimiento de Secciones [Grabar Nuevo](#)

Click en el título para Modificar el archivo

11 ARTICULOS (2 PAGINAS) NOMBRES POR PAGINA
 |< Primero << Anterior Siguiente >> Ultimo >| 10 | 20 | 30 | 40

Codigo	Nombre Sección
1	POLICIAL
2	ECONOMIA
3	SOCIAL
4	POLITICA
5	CULTURA
6	FINANZAS
7	CIUDAD
8	DESASTRES
9	DEPORTES
10	IGLESIA

Figura 5.14. Página para listar de los nombres de Secciones.

secciones.php Realiza el mantenimiento de las Secciones, que posteriormente serán utilizadas en el mantenimiento de los archivos de noticias.



Hemeroteca Virtual
Buscador de información

La sección fue creada.

Mantenimiento de secciones

Codigo:

Nombre Corto:

[Nuevo](#) [Modificar](#) [Eliminar](#)

Figura 5.15. Página para mantenimiento de las Secciones.

5.7 Conclusiones

La programación se la realizó en base a las necesidades de la Hemeroteca, considerando que los usuarios del sistema serán personas sin mayor conocimiento técnico, por lo que todas las pantallas fueron diseñadas de la manera más sencilla posible.

Hemos explicado a detalle como fue realizada la programación de la Página Web, para cubrir las necesidades de digitalización de los periódicos. Se ha indicado los estándares de desarrollo utilizado, así como un breve detalle de las herramientas utilizadas para conseguir este fin.

Se utilizó el lenguaje PHP para poder acceder a la base de datos MySql, así como Dreamweaver para la creación de la página Web, ya que con esta herramienta es posible crear páginas HTML sin tener que preocuparnos por el código HTML

Además se ha mostrado la Estructura del Sitio Web para comprender fácilmente la aplicación.

Se ha explicado gráficamente la conectividad entre el Servidor y la Aplicación para las diferentes instancias como son el Buscador de archivos, el mantenimiento de archivos y la creación de archivos.

Está definido el método de búsqueda de la información y las interfases gráficas de usuario para cada una de las instancias.

CAPÍTULO VI

DEFINICIÓN DE LA ENTRADA DE LA INFORMACIÓN

CAPÍTULO VI

DEFINICIÓN DE LA ENTRADA DE LA INFORMACIÓN

6.1 Introducción

En este capítulo detallaremos el proceso utilizado para la entrada de la información a la Base de Datos, desde la cual nuestro software realizará la búsqueda.

Para conseguir este objetivo, analizaremos las tecnologías de escaneo óptico compuestas por programas de cómputo y las bondades del equipo. Seleccionaremos la más idónea para nuestro proyecto de acuerdo a la tecnología utilizada actualmente en los equipos de escaneo modernos.

6.2 Tecnologías de Escaneo Óptico

Un dispositivo de escaneo óptico combina equipo y programas de cómputo especializados. Los equipos capturan una imagen y los programas convierten esa imagen en información que puede ser leída por la computadora.

Existen cuatro tipos de tecnologías de escaneo óptico:

- Lector de marcas ópticas (OMR por sus siglas en inglés)
- Reconocimiento de caracteres ópticos (OCR)
- Reconocimiento inteligente de caracteres (ICR)
- Tecnología de imagen

6.2.1 Lector de marcas ópticas (OMR)

Un OMR comprende típicamente ciertas clases particulares de marcas que pueden ser leídas por un escáner en un conjunto determinado de posiciones de una página. Los programas de cómputo utilizados por el escáner OMR está programada para reconocer el significado de las distintas marcas y de convertir las imágenes escaneadas en información legible por la computadora de acuerdo a su ubicación.

La tecnología OMR ha sido ampliamente utilizada desde la década de los 70's para distintos propósitos, incluyendo pruebas escolares y universitarias, censos, encuestas y sorteos, así como para votar. También se utiliza en lectores de código de barras, que se utilizan comúnmente en comercios, bibliotecas y escuelas.

La tecnología OMR es muy útil y eficiente para recabar información simple y predeterminada. Sin embargo no lo es para recabar información compleja o variable, como la que implica una gran cantidad de texto. Los sistemas OCR e ICR son más apropiados para estos propósitos.

6.2.2 Sistemas de escaneo basados en el reconocimiento de caracteres ópticos (OCR)

Estos sistemas toman imágenes escaneadas y utilizan programas de cómputo para reconocer las formas de los caracteres impresos o escritos a mano, como números o letras, y los almacenan como información que puede ser leída por la computadora. Los sistemas OCR son utilizados normalmente para convertir un texto impreso en legible para la computadora.

Otro uso importante del OCR es para la captura de información impresa en formatos. En lugar de capturarla manualmente, el OCR puede ser utilizado para convertir automáticamente la información de esos formatos en información legible para la computadora.

El OCR opera "capacitando" a los programas de escaneo para que reconozcan formas particulares como letras y números. Ya que los distintos tipos de letras tienen diferentes formas, los sistemas OCR tienen que ser capacitados para reconocer que un número o letra particular puede adoptar distintas formas. Dada la regularidad de los tipos impresos, es un proceso relativamente directo. Los sistemas OCR también pueden ser capacitados para reconocer la letra manuscrita. Sin embargo, dada la infinita

variedad de estilos, esta es una tarea mucho más complicada.

Los sistemas OCR iniciales tenían un índice relativamente alto de error cuando convertían el texto impreso en información legible para la computadora, particularmente el texto manuscrito. Esto requiere un alto nivel de supervisión humana para revisar y corregir la información que ha sido convertida. A medida que los equipos y programas de escaneo óptico mejoran a finales de la década de 1990, disminuyeron los índices de error. Sin embargo, la siguiente generación de sistemas de escaneo, los sistemas ICR, han ido más lejos en el incremento de sus niveles de precisión.

6.2.3 Sistemas de escaneo basados en el reconocimiento inteligente de caracteres (ICR)

Los sistemas ICR llevan a los OCR un paso adelante utilizando programas de cómputo para aplicar pruebas de inteligencia lógica a los caracteres escaneados para convertirlos de manera más confiable en información más legible para la computadora.

Los sistemas de ICR aplican reglas de ortografía, gramática y contexto para escanear los textos a fin de efectuar evaluaciones "inteligentes" sobre la interpretación correcta de la información. Esto permite una conversión mucho más precisa de los textos escaneados de la que realizan los sistemas OCR más simples, especialmente con el texto manuscrito.

Los programas ICR requieren computadoras rápidas y poderosas para desempeñarse de manera eficiente: Los sistemas ICR de alta confiabilidad solo estuvieron disponibles a mitad de la década de 1990 con el desarrollo de productos computacionales, económicos y poderosos.

A medida que se vuelvan más confiables los sistemas ICR, se incrementarán sus aplicaciones.

6.2.4 Tecnología de imagen

Además de capturar imágenes para convertirlas en información, los escáners también pueden capturar imágenes para almacenarlas como imágenes legibles para las computadoras. Fotografías, dibujos e imágenes de texto pueden ser almacenadas y reutilizadas en formatos legibles para computadoras.

Los formatos de papel pueden ser convertidos en imágenes y almacenados en formato electrónico. Luego se pueden bajar copias de las imágenes en una red de computadoras sin necesidad de acceder a las copias originales en papel.

La tecnología de imagen también puede ser utilizada para verificar la identidad. Las fotografías pueden ser digitalizadas y colocadas en tarjetas de identidad. Las imágenes de las huellas digitales o de rasgos faciales pueden ser digitalizadas y almacenadas en tarjetas inteligentes. Los sistemas de identidad basados en programas de cómputo pueden ser utilizados para comparar la imagen de la persona mostrada en la carta inteligente con

la codificada en la tarjeta para determinar si es la misma persona.

6.3 Guía para la subida de la información.

A continuación detallaremos los pasos a seguirse para la subida de la información a la base de datos.

6.3.1 Captura de la información a través del scanner:

El primer paso será escanear la página donde se encuentra la noticia permitiendo que se visualice lo más encuadrado posible en el lente del scanner y grabarlo en el formato jpg, puesto que este formato genera archivos de tamaños pequeños, lo cual optimizará el trabajo de almacenamiento en el Disco Duro.

Si es que el software del scanner permite manipular brillo y contraste es mejor que se configure de tal manera que el texto se aparezca lo más oscuro posible y el fondo lo más claro posible.

6.3.2 Utilización de un software que le permita realizar los cambios necesarios a la noticia deseada

Es necesario utilizar un software que permita seleccionar, recortar, encuadrar y darle atributos de brillo y color si no ha sido posible realizarlo directamente a través del software del scanner.

Podemos utilizar el Adobe PhotoShop, Microsoft PhotoEditor u otro software que manipule archivos gráficos.

Nosotros hemos utilizado Adobe PhotoShop por sus mayores beneficios y ventajas en el manejo de imágenes.

6.3.3 Reconocimiento de texto usando OCR

En este paso es imprescindible el uso de Acrobat Writer para que el archivo que está en formato de imagen (.jpg) se convierta en archivo de texto (.pdf).

Para conseguir este objetivo hemos investigado que este software permite reconocimiento óptico de caracteres en un archivo (.pdf) y además nos permite sobrescribir el archivo adicionando los atributos que utilizará nuestro software para la búsqueda de las noticias.

6.3.4 Ingreso de la información a la base de datos

Una vez realizados los pasos anteriores, el archivo se encuentra listo para agregarlo a nuestra base de datos de MySQL a través del administrador de archivos de nuestra aplicación. Este paso se explicará más detalladamente en el siguiente capítulo.

El programa ubica los archivos de la forma en que fue explicada en el *Capítulo III – 3.5 Estructura de Directorios*. Y los nombra de acuerdo a lo que se explicó en el *Capítulo III - 3.6 Nombre de Archivos*.

Por ejemplo la noticia de la muerte del Papa Juan Pablo II, del 02/abr/2005 del diario El Tiempo, nuestro software lo ubicará en el siguiente directorio:
Biblioteca\periodicos\N\TIE\20050402\IGL
Y grabará la noticia con el nombre: NTIE20050402IGL22.pdf

6.4 Conclusiones.

Hemos definido un proceso lo más sencillo posible para la conversión y subida de la información.

El proceso comprende desde la captura de la información a través del scanner, la utilización de un software que le permita seleccionar la noticia deseada, el reconocimiento de texto usando OCR y el ingreso de la información a la base de datos que nos permitirá visualizar la noticia en nuestro buscador.

De la experiencia que hemos tenido al subir la información de los periódicos antiguos y nuevos en el Banco Central, vemos la necesidad que se utilice un scanner con muy buena resolución y si es posible de formato A3 en adelante, ya que con estos modelos de scanner se puede capturar un área mayor del periódico y se pueda a su vez tener más noticias scaneadas en cada pasada, lo cual optimiza el tiempo de captura de la información.

Además sería conveniente que el software del scanner permita manipular la nitidez y el encuadramiento del archivo, para obviar el uso del Adobe Photoshop, citado en este capítulo.

CAPÍTULO VII

IMPLEMENTACIÓN DE LA APLICACIÓN

CAPÍTULO VII

IMPLEMENTACIÓN DE LA APLICACIÓN

7.1 Introducción

En este capítulo mostraremos los detalles de la implementación de las páginas Web, desarrolladas para la digitalización de los periódicos de la Hemeroteca.

Explicaremos ampliamente la manera más adecuada para la captura de la información de los periódicos, indicando gráficamente cada uno de los pasos a seguir.

Se indicará la forma de realizar el mantenimiento de los archivos que conforman nuestra base de datos de archivos textuales.

También enseñaremos como realizar las búsquedas de noticias o archivos.

7.2 Captura de la Información

La noticia deberá ser escaneada y manipulada la imagen por un software que admita realizar los cambios necesarios para que este archivo quede correctamente visualizado y encuadrado de tal forma que con el software del Adobe Acrobat Writer permita realizar una conversión de reconocimiento óptico de caracteres.

7.3 Ingreso al Mantenimiento de Archivos

Para acceder a nuestra aplicación debe realizarlo desde el Internet Explorer, digitando en la dirección: <http://localhost/biblioteca/logon.php>.

Aparecerá la siguiente pantalla con la que debe ingresar el nombre del usuario y su clave y presionar el boton “Ingresar”, como se indicó en el *Capítulo V – 5.2 Definición del Front-End o interfase con el usuario*.



Figura 7.1. Página de Acceso a Mantenimiento de archivos

Para el mantenimiento de la Base de Datos tenemos 2 niveles de seguridad:

- La persona que administrará y agregará dicha información (administrador).
- La persona que únicamente podrá agregar la información (user).

7.4 Página de Administración de Archivos

Una vez convertida la información en un archivo .pdf y realizado el proceso de reconocimiento óptico de caracteres (OCR), se puede en este momento ya agregarlo a la Base de Datos de Archivos Textuales.

Luego de haber ingresado, la aplicación muestra un listado completo de todos los archivos almacenados en la Base de Datos de Archivos Textuales.



Mantenimiento Periodico Sección Nuevo Archivo

Click en el título para Modificar el archivo o en Grabar Nuevo para generar uno nuevo.

21 ARTICULOS (3 PAGINAS) NOMBRES POR PAGINA
 |< Primero << Anterior Siguiente >> Ultimo >| 10 | 20 | 30 | 40

Codigo	Tipo	Titulo	Periodico	Fecha	Sección
2	N	AUCAS ASESINAN A 3 MISIONEROS	MERCURIO	1969-09-26	NAC
3	I	AVION VIAJA ALREDEDOR DEL MUNDO	MERCURIO	1970-06-25	SOC
4	N	BANCO DEL AZUAY JUNTA GENERAL ORDINARIA DE AC	MERCURIO	1969-07-01	ECO
5	I	400 DELINCUENTES DE GUERRA HAN SIDO YA EJECU	MERCURIO	1970-10-12	EDI
6	I	SUICIDOSE HIJA DEL ESCRITOR ESPAÑOL LUIS ARAQ	MERCURIO	1975-09-07	FAR
8	N	SOLO UN LOGRÓ SALVARSE MILAGROSAMENTE	MERCURIO	1959-09-26	POLICIAL
9	N	AGUA CUBRE CASAS EN LA JOSEFINA.pdf	TIEMPO	1993-04-01	DESASTRES
12	N	JOSEFINA	TIEMPO	1993-04-01	DESASTRES
13	N	PEDIDOS DE AUXILIO	TIEMPO	1993-04-04	DESASTRES
17	N	Cacerolazo	MERCURIO	2005-03-27	POLITICA

Figura 7.2. Página de Administración de Archivos

Esta página Web exhibe las características de los archivos de texto y son las siguientes:

Código: Es un número secuencial que el programa lo coloca automáticamente cuando se agrega un nuevo archivo. Al presionar click en esta columna, se realiza el acceso al mantenimiento de esta información.

Tipo: Corresponde al tipo del periódico sea este Nacional (N) o Internacional (I).

Título: Es el título de la noticia o una breve descripción de la misma.

Periódico: Corresponde al nombre del Periódico de la noticia.

Fecha: Es la fecha de la noticia.

Sección: Se encuentra la sección a la que corresponde la noticia.

El programa puede mostrar en grupos de 10, 20, 30 o 40 archivos por página. Así como también puede navegar a la primera, siguiente, anterior y última página

21 ARTICULOS (3 PAGINAS)

[|< Primero](#) [<< Anterior](#) [Siguiete >>](#) [Ultimo >|](#)

NOMBRES POR PAGINA

[10](#) | [20](#) | [30](#) | [40](#)

7.4.1 Ordenación de los archivos

Si deseamos ordenar el listado por los diferentes campos enunciados arriba, se lo puede realizar presionando click en los títulos de cada uno de ellos. Se puede realizar la ordenación por:

- Código
- Tipo
- Título
- Periódico
- Fecha
- Sección

7.4.2 Links de acceso

En esta página Web tenemos links ubicados en la parte superior como se muestra en la siguiente figura, que sirven para acceder a los mantenimientos de:

- Periódicos
- Secciones
- Archivos



Figura 7.3. Links de Acceso.

7.4.2.1 Periódico

Esta página permite visualizar un listado de los nombres de los periódicos existentes en la Base de Datos.

 The screenshot shows the 'Hemeroteca Virtual' interface. At the top, there is a blue banner with the text 'Hemeroteca Virtual' and 'Buscador de información' on the left, and an image of a newspaper and the word 'Periódico' on the right. Below the banner, the page title is 'Mantenimiento de Periodicos' and there is a 'Grabar Nuevo' button. A note says 'Click en el título para Modificar el archivo'. The main content area shows '5 ARTICULOS (1 PAGINAS)' and 'NOMBRES POR PAGINA' with navigation links. A table lists five newspapers with their codes, short names, and long names.

Código	Hombre Corto	Hombre Largo
1	MERCURIO	EL MERCURIO
2	TIEMPO	EL TIEMPO
3	COMERCIO	EL COMERCIO
4	HOY	EL HOY
5	UNIVERSO	EL UNIVERSO


Figura 7.4. Listado de nombres de Periódicos

Se puede Modificar o Eliminar registros, presionando click en el registro deseado.

Si deseamos agregar un nuevo nombre de periódico, tenemos el siguiente link de acceso:

[Grabar Nuevo](#)

Este link de acceso nos lleva a la siguiente pantalla:



The screenshot shows the 'Hemeroteca Virtual' application interface. At the top, there is a blue header with the text 'Hemeroteca Virtual' in orange and 'Buscador de información' in white. To the right of the header is an image of a stack of newspapers. Below the header, the main content area is white and contains the title 'Mantenimiento de Periódicos' in blue. Underneath the title, there are three input fields: 'Codigo:' with the value '6', 'Nombre Corto:' with the value 'MERCURIO', and 'Nombre Largo:' with the value 'EL MERCURIO'. At the bottom right of the form, there are three orange buttons labeled 'Grabar', 'Modificar', and 'Eliminar'.

Figura 7.5. Mantenimiento de Periódicos

Nuevo: Para agregar un registro hay que presionar el botón <<Nuevo>>, luego de ingresar los datos presionar el botón <<Grabar>>.

Modificar: Visualizará la página de mantenimiento de periódicos (Fig. 7.5) desde la cual tendrá acceso a modificar tanto el nombre corto como el nombre largo y para terminar con el cambio presione el botón <<Modificar>>.

Eliminar: Al igual que en la modificación se visualizará la página de la Fig. 7.5, presione el botón <<Eliminar>>.

7.4.2.2 Sección

Para visualizar un listado de los nombres de las secciones existentes en la Base de Datos, tenemos la siguiente página:



Hemeroteca Virtual
Buscador de información

Mantenimiento de Secciones [Grabar Nuevo](#)

Click en el título para Modificar el archivo

12 ARTICULOS (2 PAGINAS) NOMBRES POR PAGINA
 < Primero << Anterior Siguiente >> Ultimo >| 10 | 20 | 30 | 40

Codigo	Nombre Sección
1	POLICIAL
2	ECONOMIA
3	SOCIAL
4	POLITICA
5	CULTURA
6	FINANZAS
7	CIUDAD
8	DESASTRES
9	DEPORTES
10	IGLESIA

12 ARTICULOS (2 PAGINAS) NOMBRES POR PAGINA
 < Primero << Anterior Siguiente >> Ultimo >| 10 | 20 | 30 | 40

Figura 7.6. Listado de nombres de Secciones

Se puede Modificar o Eliminar registros, presionando click en el registro deseado.

Para agregar una nueva sección debemos presionar el siguiente botón:

[Grabar Nuevo](#)

Este link de acceso nos lleva a la siguiente pantalla:



Hemeroteca Virtual
Buscador de información

Mantenimiento de secciones

Codigo:

Nombre Corto:

Nuevo **Modificar** **Eliminar**

Figura 7.7. Mantenimiento de Secciones

Nuevo: Presione el botón <<Nuevo>> e ingrese el nombre corto de la sección para luego presionar el botón <<Grabar>>.

Modificar: Ingrese el nombre de la sección y presione el botón <<Modificar>> para realizar el cambio.

Eliminar: Se visualizará el registro y si desea borrarlo presione el botón <<Eliminar>>.

7.4.2.3 Nuevo Archivo

El procedimiento para agregar la información de las noticias a la Base de Datos se lo realiza a través de esta página.



The screenshot shows the 'Hemeroteca Virtual' interface. At the top, there is a blue header with the text 'Hemeroteca Virtual' in orange and 'Buscador de información' in white. To the right of the header is an image of a folded newspaper. Below the header, the main content area is titled 'Mantenimiento del Sistema de Archivos Nuevos'. The form contains several fields: 'Codigo:' with the value '33'; 'Titulo Noticia:' with an empty text box; 'Tipo:' with a dropdown menu showing '-- Tipo --'; 'Periódico:' with a dropdown menu showing '-- Nombre del Periodico --'; 'Fecha:' with three dropdown menus for 'Día' (15), 'Mes' (Julio), and 'Año' (2005); 'Sección:' with a dropdown menu showing '-- Nombre de la Sección --'; and 'Archivo:' with an empty text box and an 'Examinar...' button. At the bottom right of the form is a red 'Grabar' button.

Figura 7.8. Creación de Archivos Nuevos

Es necesario que se llene los siguientes campos:

Título Noticia: En este casillero debe ingresarse el título de la noticia o una breve descripción de la misma.

Tipo: Se tiene opción de elegir si el periódico es Nacional o Internacional.

Periódico: En este casillero debe elegirse el nombre del periódico al que corresponde la noticia.

Fecha: Compuesta por Día, Mes y Año de la noticia.

Sección: Es la sección a la que corresponde la noticia.

Archivo: En este casillero se guarda la ruta de acceso al archivo que contiene la noticia. Tiene un botón para seleccionar el archivo.

7.4.3 Mantenimiento de la información de Archivos

Para realizar cualquier eliminación o modificación a la información básica del archivo, es necesario desde el Listado de archivos presionar un click en el código del registro deseado. El sistema le presentará la siguiente página para realizar el cambio correspondiente:



Hemeroteca Virtual
Buscador de información

Mantenimiento del Sistema de Archivos

Codigo:

Titulo Noticia:

Tipo:

Periódico:

Fecha: Día Mes Año

Sección:

Archivo:

Figura 7.9. Mantenimiento de Archivos

7.5 Manejo de las búsquedas

El sistema ofrece una forma muy sencilla de realizar las búsquedas. Posee varios filtros que permiten encontrar

la información deseada de la manera más exacta posible. Una explicación más detallada sobre este tema puede encontrar en el *Capítulo 5.5 Definición de las búsquedas*.

Por ejemplo, para realizar una búsqueda de todas las noticias relacionadas con el “Desastre de la Josefina del año 1993”, colocamos la palabra “Josefina” en el casillero “Buscar Texto” como se indica a continuación:



Buscador de información

Buscar texto: JOSEFINA

Tipo: -- Seleccione Tipo --

Periódico: -- Seleccione Periódico --

Sección: -- Seleccione Sección --

Tamaño: Menor que tamaño

Fecha: tiempo

Ejecutar Limpiar

Resultados

Archivos 1 a 4 de 4 iguales a: JOSEFINA

Figura 7.10. Formulario de Búsquedas

El programa realizará la búsqueda en todos los archivos que en su contenido posean la palabra “Josefina” y mostrará una página de resultados como se observa en la siguiente figura:

1	<p>Periódico: TIEMPO Título: Dique podría reventar 90%</p> <p>Sección: DESASTRES Tipo: Nacional Fecha: 30/03/1993 22:01:49</p> <p>Resumen: N. DESASTRES. La noche de ayer, a las 21 h00, se produjo un primer deslave del cerro o colina situada en la margen izquierda aguas abajo, del río Paute, en la confluencia con el Jadán, en el sector denominado "La Josefina" el cual comprende la carretera Cuenca-El Descanso Paute. El deslave continúa según las inf Tamaño: 129470 bytes</p> <p>Ubicación: e:\inetpub\wwwroot\Biblioteca\periodicos\ntie\19930330\des\ntie19930330des33.pdf</p>
2	<p>Periódico: TIEMPO Título: Dique Podria Reventar 90%</p> <p>Sección: DESASTRES Tipo: Nacional Fecha: 30/03/1993 21:54:33</p> <p>Resumen: N. DESASTRES. La noche de ayer, a las 21 h00, se produjo un primer deslave del cerro o colina situada en la margen izquierda aguas abajo, del río Paute, en la confluencia con el Jadán, en el sector denominado "La Josefina" el cual comprende la carretera Cuenca-El Descanso Paute. El deslave continúa según las inf Tamaño: 129237 bytes</p> <p>Ubicación: E:\inetpub\wwwroot\Biblioteca\Periodicos\Ntie\19930330\Des\Ntie19930330des20.Pdf</p>
3	<p>Periódico: TIEMPO Título: UN AÑO DESFOGUE JOSEFINA 90%</p> <p>Sección: DESASTRES Tipo: Nacional Fecha: 30/04/1994 22:40:14</p> <p>Resumen: N. DESASTRES. l Hace un año fue el desfogue Mañana se cumple un año del violento desfogue de i aguas de los ríos Jadán y Cuenca que estuvieron presadas desde el 29 de marzo, fecha en la cual elcerro imuga se desplomó sobre el cauce de los dos ríos en el fión del Tahual y causó el peor desastre que haya lrido la Reg Tamaño: 203808 bytes</p> <p>Ubicación: e:\inetpub\wwwroot\Biblioteca\periodicos\ntie\19940430\des\ntie19940430des51.pdf</p>
	<p>Periódico: MERCURIO Título: JOSEFINA COMITE SOLIDARIDAD ENTREGA AYUDA 60%</p> <p>Sección: DESASTRES Tipo: Nacional Fecha: 06/04/1993 22:20:57</p>

Figura 7.11. Formulario de Respuestas de Búsquedas

Cada una de los archivos encontrados contiene los siguientes datos:

- Nombre del periódico
- Título de la Noticia
- Sección de la Noticia
- Tipo de Periódico
- Fecha de la Noticia
- Un breve resumen generado en forma automática por el buscador
- Un link de acceso al archivo de la noticia

Al presionar click ya sea en el Título de la noticia o en el link de la ubicación, se exhibirá el archivo a través del Acrobat Reader de la siguiente manera:



Figura 7.12. Ejemplo de Archivo de la Noticia

Una vez abierta la noticia seleccionada, se puede realizar algunas acciones como: Imprimir, Grabar, Ampliar, Reducir, Buscar, etc, comandos propios del Acrobat Reader.

Es importante notar que la búsqueda se realiza en todo el contenido del archivo y no sólo en el título de la noticia. En el ejemplo se muestra un archivo, en donde la palabra buscada “Josefina” se encuentra en el segundo párrafo.

7.6 Conclusiones

La aplicación desarrollada está dirigida a usuarios sin mayor conocimiento técnico, pudiendo encontrar en forma fácil y rápida las noticias existentes en la Base de Datos.

Este capítulo contiene básicamente un manual de usuario, tanto para la subida de la información a la base de datos como para la búsqueda de las noticias.

Hemos realizado pruebas para la implementación de este buscador en la Biblioteca del “Banco Central”, scaneando periódicos desde los más antiguos hasta los del año actual, a fin de encontrar la manera más idónea para la digitalización de la información.

De esta experiencia podemos recomendar que sería aconsejable utilizar un scanner que posea un software que permita manipular el brillo y el contraste de la noticia, para que pueda ser reconocido fácilmente por el buscador. Además es necesario realizar el Reconocimiento Óptico de Caracteres, tarea que lo realiza el Acrobat Writer, para que posteriormente sea subida a nuestra Base de Datos a través de la página Web

Cabe destacar que es mejor utilizar un scanner de alta resolución mínimo de 600 x 600 dpi de tamaño A3.

CAPÍTULO VIII

CONCLUSIONES Y RECOMENDACIONES

8.1 CONCLUSIONES

Luego de haber finalizado el desarrollo del presente proyecto de tesis y de acuerdo a los objetivos inicialmente planteados, podemos concluir que:

- Se ha logrado tener un método para la sistematización y digitalización de la cuantiosa y valiosa información que posee la Hemeroteca “Piedad Paredes de Jaramillo” y su posterior funcionamiento, rescatando de esta manera este material que se va deteriorando con cada día que pasa. Con este sistema podrá estar la información más importante de los periódicos que se encuentran embodegados desde el año 1940 al alcance de un público más amplio del que actualmente ahora está a disposición.
- El software con el que trabajamos es un su mayoría un software gratuito o de libre distribución, lo que permitirá que Hemerotecas medianas y pequeñas puedan implementar este proyecto con una reducción significativa en costos. Hemos utilizado las siguientes herramientas para el desarrollo e implementación de nuestra aplicación: Dreamweaver

MX 2004, ASP, MySql, PHP, FireWorks, Adobe Acrobat Professional, I-Filter, CSS Cascading Style Sheets, Internet Information Server, Index Server de Microsoft Windows.

- Hemos diseñado una Base de datos de archivos textuales, para la Gestión de la información existente en la Hemeroteca, la misma que podrá ser administrada fácilmente a través de otra base de datos relacional MySql, que nos permitirá acceder de manera sencilla y rápida.
- Hemos desarrollado una aplicación en página Web para la consulta, búsqueda e impresión de noticias por Temas, Nombre de periódico, Sección y Fechas al que pertenece. Nuestra aplicación usa las bondades que presta el Internet Information Server, a través de su poderosa herramienta como es el Index Server, que es un indexador completo de texto y un motor de búsqueda, el cual maneja una gran cantidad de consultas. Estos dos propios del Sistema Operativo, lo cual nos permite realizar las búsquedas en forma nativa.
- Se ha realizado un análisis de los equipos y de la red necesarios para implementar este software. El detalle de este estudio se encuentra en el Capítulo IV, además tenemos tres opciones de compra, que van desde una más barata y de marcas no muy utilizadas en el medio, hasta una opción más costosa pero de marcas muy reconocidas. La Hemeroteca tendría que

analizar su presupuesto para seleccionar la mejor opción.

- El proceso utilizado para la entrada de la información a la Base de Datos, desde la cual nuestro software realizará la búsqueda, comprende desde la captura de la información a través del scanner, la utilización de un software que le permita seleccionar la noticia deseada, el reconocimiento de texto usando OCR y el ingreso de la información a la base de datos que nos permitirá visualizar la noticia en nuestro buscador.
- Tenemos además un Manual de Usuario, tanto para la subida de la información a la base de datos como para la búsqueda de las noticias detallado en el Capítulo 7.

8.2 RECOMENDACIONES

De la experiencia adquirida en la recolección de la información realizada en la Biblioteca del Banco Central, podemos acotar que es conveniente utilizar un scanner de formato A3 ya que con este tamaño se puede scannear de forma más fácil y rápida la información de los periódicos.

Además es importante anotar que al momento de realizar la captura de la noticia recomendamos cambiar la configuración de la resolución de la imagen y colocarlo mínimo en 200 dpis para que posteriormente el Reconocimiento Óptico de Caracteres (OCR) pueda realizarse satisfactoriamente.

Del conocimiento que tenemos a través del Presidente de la Casa de la Cultura núcleo del Azuay, Dr. Efraín Jara Hidrovo, la Hemeroteca “Piedad Paredes Jaramillo” que es parte de la Casa de la Cultura, no posee los recursos necesarios para la ejecución de este proyecto. Por lo que recomendamos que se junten las Instituciones que también tienen este material almacenado como son además: la Biblioteca de la Universidad de Cuenca, la Biblioteca del Banco Central, la Biblioteca del Municipio y la Biblioteca de nuestra Universidad, para que realicen la ejecución de este proyecto abaratando de esta manera costos.

Pressman, Roger S., Ingeniería del Software: Un enfoque Práctico,
Editorial.

<http://www.microsoft.com/latam/technet/articulos/200010/art11/default.asp>
http://www.windowstimag.com/atrasados/2000/46_oct00/articulos/indexserver_web.htm

UNIVERSIDAD DEL AZUAY

DISEÑO DE TESIS

“IMPLEMENTACIÓN DEL SISTEMA DE ADMINISTRACIÓN DE ARCHIVOS TEXTUALES DE LA CASA DE LA CULTURA NÚCLEO DEL AZUAY – HEMEROTECA PIEDAD PAREDES DE JARAMILLO”.

DIRECTOR DE TESIS

ING. FRANCISCO SALGADO

INTEGRANTES

- ANLS. SANDRA ABRIL S.
- ANLS. SUSANA RODAS S.

JULIO 2003

DISEÑO DE TESIS

1.- TITULO DEL PROYECTO

“Implementación del Sistema de Administración de Archivos Textuales de la Casa de la Cultura Núcleo del Azuay – Hemeroteca Piedad Paredes de Jaramillo”.

2.- RESUMEN DEL PROYECTO

Con el desarrollo de esta tesis se pretende dotar a la Hemeroteca “Piedad Paredes de Jaramillo” de una aplicación en Página Web que permita sustituir la manipulación del material físico con el manejo electrónico de la misma, permitiendo que el servicio al lector sea más provechoso.

Con la implementación de esta tesis se va a poner a disposición del público periódicos a través de un Sistema buscador de artículos en página WEB con hipervínculos que permitan acceder al texto. Dicha información es muy valiosa y está siendo subutilizada debido a que no cuentan con un sistema de información que les permita administrarla eficientemente.

Actualmente la Hemeroteca no cuenta con un sistema informático de comunicación, por lo que será necesario realizar un estudio técnico de redes y equipos, sugerir la compra de un servidor, terminales y más dispositivos necesarios.

Adicionalmente el sistema contará con un administrador de la Base de Datos que permita darle mantenimiento.

3.- IMPACTO TECNOLÓGICO

Esta aplicación Web constituye un aporte tecnológico de gran utilidad para la manipulación de archivos textuales, permitiendo la captura de datos, la búsqueda y administración de los mismos, desde la página Web.

4.- IMPACTO SOCIAL

La aplicación Web facilitará a los usuarios finales y funcionarios de la Hemeroteca a fin de llevar un control de las publicaciones de los principales diarios del País, poniendo a disposición del público esta valiosa información.

Al haber escogido este tema de Tesis “Implementación del Sistema de Administración de Archivos Textuales de la Casa de la Cultura Núcleo del Azuay – Hemeroteca Piedad Paredes de Jaramillo”, adquiriremos conocimientos en lo que se refiere a la estructura, funcionamiento y administración de Hemerotecas en páginas Web. La misma que estará a disposición de otros desarrolladores que podrán consultar en la Biblioteca de la Universidad.

Actualmente existen otras hemerotecas en la ciudad que podrían mejorar sus servicios de atención al público valiéndose de los beneficios de esta aplicación.

5.- OBJETIVOS

5.1 OBJETIVO GENERAL

Diseñar y desarrollar un sistema de información Web para la Gestión de Archivos textuales de Hemerotecas, aplicada a la Hemeroteca “Piedad Paredes de Jaramillo” de la Casa de la Cultura Núcleo del Azuay.

5.2. OBJETIVOS ESPECIFICOS

- Sistematizar la información relativa al trabajo de una Hemeroteca y su funcionamiento digitalizado.
- Diseñar una Base de datos para la Gestión de la información de la Hemeroteca.
- Desarrollar una aplicación en página Web para la consulta, búsqueda e impresión de artículos por temas y fechas así como el mantenimiento, almacenamiento y recuperación de la base de datos.
- Realizar un estudio técnico de la red y los equipos necesarios para implementar este software.

6.- TEORIA REFERENCIAL

De manera resumida, nuestra propuesta de investigación abarca los siguientes conceptos:

Base de Datos de Archivos textuales:

- Es una colección organizada de datos o archivos de texto relacionados que se almacenan en un archivo.

Tecnología Cliente/Servidor:

- La arquitectura Cliente/Servidor es la plataforma abierta por excelencia, por la variedad de combinaciones de clientes y servidores que permite conectar en red.

La arquitectura Cliente/Servidor es un modelo para el desarrollo de sistemas de información, en el que las transacciones se dividen en elementos independientes que cooperan entre sí para intercambiar información, servicios o recursos.

En esta arquitectura la computadora de cada uno de los usuarios, llamada cliente, inicia un proceso de diálogo: produce una demanda de información o solicita recursos. La computadora que responde a la demanda del cliente, se conoce como servidor. Bajo este modelo cada usuario tiene la libertad de obtener la información que requiera en un momento dado proveniente de una o varias fuentes locales o distantes y de procesarla como según le convenga.

Motor de Búsqueda:

- Un buscador es un conjunto de programas instalados en un servidor conectado a Internet: un sitio más al que se puede acceder en la Web.

Esta herramienta, llamada también de varias formas (Search Engine, motor de búsqueda, robots inteligente, etc.) son programas tipo base de datos, con un software diseñado especialmente para buscar la información a través de los enlaces de las paginas Web. Algunos motores tiene programas muy sofisticados, pero que a nuestro fin, no importa el mecanismo interno por el cual lo realizan, ya que es muy complejo y técnico, pero a veces, el éxito de la búsqueda en un motor específico, estará determinado por el volumen y el contenido de la base de datos del motor, la velocidad de búsqueda, la facilidad del uso, etc.

7.- UNIVERSO DEL ESTUDIO:

Nuestro universo de estudio será el Sistema de Administración de Archivos Textuales de la Casa de la Cultura Núcleo del Azuay – Hemeroteca Piedad Paredes de Jaramillo.

8.- METODOLOGIA DE INVESTIGACION:

Para la recopilación de la información nos basaremos en lo siguiente:

- Entrevistas :
 - Sr. Paúl Solano

- Sra. Lucrecia Palacio
 - Sr. Rafael Montenegro
- Investigación de campo
 - Revisión del material existente en la Hemeroteca.
- Navegación en el Internet utilizando los buscadores

9.- PLAN OPERATIVO:

Fase 1

1. INVESTIGACION DE CAMPO

- 1.1. Recopilación de Información
- 1.2. Determinación del Problema

Fase 2

2. ESTUDIO Y REVISION DE LAS HERRAMIENTAS PARA EL DESARROLLO DE LA APLICACION

- 2.1. Estudio de la configuración del Servidor bajo la plataforma Windows NT
- 2.2. Estudio del Software a utilizar para la creación de Páginas Web
- 2.3. Estudio de los Graficadotes de las páginas Web
- 2.4. Estudio de las Herramientas eficaces (ASP) que permitirán conectarse de alguna forma a la Base de Datos para poder manipularlos desde la Web

Fase 3

3. DISEÑO DE LA ESTRUCTURA DE ARCHIVOS

- 3.1. Diseño de las estructuras de los archivos

Fase 4

4. ANALISIS Y DISEÑO DE LOS RECURSOS DE HARDWARE Y DE REDES REQUERIDOS PARA EL SISTEMA

- 4.1. Análisis y diseño de los recursos de Hardware
- 4.2. Análisis y diseño de los recursos de Redes

Fase 5

5. PROGRAMACION DE LA APLICACIÓN WEB

- 5.1. Diseño de la Página Web

- 5.2. Definición y programación de las búsquedas
- 5.3. Definición del Front-End o interfase
- 5.4. Construcción de la página
- 5.5. Programación de las búsquedas y atributos

Fase 6

6. INTERFAZ DE LA APLICACIÓN WEB Y EL SERVIDOR

- 6.1. Crear la interfaz entre el servidor, y el sistema de información desarrollado
- 6.2. Conectividad posterior de las búsquedas con la página Web

Fase 7

7. DEFINICION DE LA SUBIDA DE LA INFORMACIÓN

- 7.1. Definición del proceso.
- 7.2. Pruebas de subida de datos

Fase 8

8. PUESTA EN PRUEBA DE LA APLICACIÓN WEB

- 8.1. Verificación del funcionamiento correcto de la aplicación Web
- 8.2. Corrección de errores en caso que los hubiere

Fase 9

9. IMPLEMENTACION DE LA APLICACIÓN WEB

10.- RECURSOS HUMANOS:

Las personas que participarán en la elaboración del sistema son las siguientes:

- **Director de Tesis:**
Ing. Francisco Salgado
- **Director de Diseño:**
Lic. Vicente Córdova
- **Alumnos aspirantes al título de Ingeniero de Sistemas:**
Anls. Sandra Abril Z.
Anls. Susana Rodas S.

11.- RECURSOS MATERIALES

▪ **Hardware:**

El hardware necesario para que funcione nuestra aplicación Web depende de ciertos factores, como son el tamaño de las tablas y de los demás servicios que se están ejecutando en el servidor.

El hardware mínimo que se sugiere a usar es:

- Una computadora personal con un microprocesador Intel – Pentium IV o superior.
- Con disco duro de tecnología SCOSI 30 GB en adelante
- 512 MB de memoria RAM
- Un monitor 14”.

▪ **Software:**

El software de nuestra aplicación incluye lo siguiente:

- **HTML** Es el idioma estándar para documentos Hypertext se llama HTML (Hypertext, Markup Lenguaje). En él se define la estructura de los documentos por medio de una serie de elementos de control (llamados tags).
- **Java Script**, es un lenguaje de programación que se utiliza dentro del html. Lo interpreta el navegador y produce alguna acción determinada en la página web donde está insertado.
- **ASP.-** Es un archivo de texto con la extensión .asp que contiene cualquier combinación de lo siguiente: Texto, Etiquetas HTML, secuencias de comandos del servidor.
- **Dreamveawer.-** Diseñador de Paginas Web.
- **Genexus.-** Case con Inteligencia Artificial.

Cuenca, 28 de Julio de 2003.

Ingeniero

Miguel Moscoso Cobos

**DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
DE LA UNIVERSIDAD DEL AZUAY.**

Cuenca.-

De nuestra consideración:

Quienes suscribimos comunicamos a Usted que hemos procedido a revisar el Diseño de Tesis presentado por las estudiantes, Sandra Abril Zambrano y Susana Rodas Sumba, egresadas de la Escuela de Ingeniería de Sistemas, con el tema "IMPLEMENTACIÓN DEL SISTEMA DE ADMINISTRACIÓN DE ARCHIVOS TEXTUALES DE LA CASA DE LA CULTURA NÚCLEO DEL AZUAY – HEMEROTECA PIEDAD PAREDES DE JARAMILLO", como requisito previo a la obtención del Título de Ingenieros de Sistemas sobre el cual presentamos el siguiente informe:

1. El diseño cumple con los requisitos metodológicos básicos exigidos por la Facultad, en cuanto a la descripción del objeto de estudio, resumen del proyecto, situación actual, situación proyectada, justificación – impactos, formulación de problemas, objetivos, marco teórico, esquema tentativo, procedimientos metodológicos, recursos, cronograma y bibliografía necesaria para el desarrollo de la tesis.

Por las consideraciones anotadas, se emite un informe favorable y salvo su mejor criterio, se recomienda la aprobación.

Atentamente,

Lcdo. Vicente Córdova Mosquera
PROFESOR DE METODOLOGÍA

Ing. Francisco Salgado
DIRECTOR DE TESIS

Cuenca, 28 de Julio del 2003

Ingeniero

Miguel Moscoso

**DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
DE LA UNIVERSIDAD DEL AZUAY**

Cuenca.-

De nuestra consideración:

Nosotras, Sandra Abril Zambrano y Susana Rodas Sumba, egresadas de la Escuela de Ingeniería de Sistemas, nos dirigimos a Usted, y por su digno intermedio al Honorable Consejo de Facultad, para solicitarle la aprobación del Diseño de Tesis con el Tema "IMPLEMENTACIÓN DEL SISTEMA DE ADMINISTRACIÓN DE ARCHIVOS TEXTUALES DE LA CASA DE LA CULTURA NÚCLEO DEL AZUAY – HEMEROTECA PIEDAD PAREDES DE JARAMILLO", así como la asignación de Director.

Nos permitimos sugerir el nombre del Ing. Francisco Salgado Arteaga como Director de Tesis por cuanto ha sido el profesor de la Facultad con quien hemos tenido asesoría y además contamos con su aceptación.

.

Por la favorable acogida que brinde a la presente, le anticipamos nuestro más sincero agradecimiento.

Atentamente,

Anls. Sandra Abril Z.
Código:4046

Anls. Susana Rodas S.
Código:8834

ANEXO I

```

<?php require_once('conecta.php');
    $modi=0;
    mysql_select_db($database_conexion, $conexion);
    $query_Recordt = "SELECT max(codigo_inf) as codigo FROM informacion";
    $Recordt = mysql_query($query_Recordt, $conexion) or die(mysql_error());
    $row_Recordt = mysql_fetch_assoc($Recordt);
    $codigo=$row_Recordt['codigo']+1;

$msg="";
if (isset($_REQUEST['log']))
    if (isset($_REQUEST['exec']))
        $titulo=$_REQUEST['titulo'];
        $tipo=$_REQUEST['tipo'];
        $periodico=strtoupper(substr($_REQUEST['periodico'],0,3));
        $dia=$_REQUEST['dia'];
        $mes=$_REQUEST['mes'];
        $ano=$_REQUEST['ano'];
        $seccion=strtoupper(substr($_REQUEST['seccion'],0,3));
        if (!file_exists('periodicos/' . $tipo)){
            $carpetatmp= mkdir ('periodicos/' . $tipo, 0700);
        }
        if (!file_exists('periodicos/' . $tipo . "/" . $periodico )){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico, 0700);
        }
        if (!file_exists('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia )){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia , 0700);
        }
        if (!file_exists('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion)){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion, 0700);
        }

        $archivo= $tipo . $periodico . $ano . $mes . $dia . $seccion . $codigo;
        $uploaddir = 'periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion . "/";
        $uploadfile = $uploaddir . $archivo . ".pdf";

        if ($_FILES['userfile']['type']=="application/pdf"){
            if (is_uploaded_file($_FILES['userfile']['tmp_name'])){
                if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploadfile)) {
                    $msg="El archivo fue actualizado en la ubicación solicitada.\n";

                    $insertSQL = "INSERT INTO informacion (periodico_inf , tipo_inf, fecha_inf,
                    seccion_inf, archivo_inf, titulo_inf) VALUES (" . $_REQUEST['periodico'] . "," . $_REQUEST['tipo'] . "," . $_REQUEST['ano'] . "/" .
                    $_REQUEST['mes'] . "/" . $_REQUEST['dia'] . " , " . $_REQUEST['seccion'] . " , " . $uploadfile . " , " . $_REQUEST['titulo'] . ")";
                    mysql_select_db($database_conexion, $conexion);
                    $Result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());
                    $salida="";

                    $fd = fopen($uploadfile, "r");
                    while (!feof($fd)) {
                        $buffer = fgets($fd, 4096);
                        $salida=$salida . $buffer;
                    }
                    $salida= $salida;
                    $info="3 0 obj<</CreationDate(D:." . $_REQUEST['ano'] . $_REQUEST['mes'] .
                    $_REQUEST['dia'] . " .date ("H"). date("i").date("s") . "-05'00')/Subject(" . $_REQUEST['tipo'] . ") /Author(" . $_REQUEST['periodico']
                    .")/Creator(Adobe Acrobat 7.0)/Comments(" . $_REQUEST['ano'] . $_REQUEST['mes'] . $_REQUEST['dia'] . " .date ("H"). date("i").date("s")
                    .")/Keywords(" . $_REQUEST['seccion'] . ") /Producer(Adobe Acrobat 7.0 Paper Capture Plug-in)/ModDate(D:20050525104453-05'00')/Title("
                    $_REQUEST['titulo'] . ")>>.\r\n."endobj" . "\r\n."%%EOF";

                    $ts08 = fopen($uploadfile,"w");
                    fwrite($ts08,$salida . "\r\n" . $info);

                    fclose($fd);

                } else {
                    $msg="El archivo no fue grabado revise por favor el tamaño.\n";
                }
            }
        } else{
            $msg="<b>Error:</b> >> El formato del archivo proporcionado no es el correcto.";
        }
}

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Principal</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL=STYLESHEET HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">

```

```

<tr>
  <td width="50%" align="right" valign="top">
<form action="admin.php" method="post" enctype="multipart/form-data" name="form1">
<table width=97% align="right">
<tr align="right">
  <td colspan="3" class="textoper06"><?php echo $msg?></td>
</tr>
<tr>
  <td height="35" colspan="3" class="textoper03">Mantenimiento del Sistema de Archivos Nuevos</td>
</tr>
<tr>
  <td colspan="3" class="lineasup">&nbsp;</td>
</tr>
<tr>
  <td height="20" colspan="2" align="right" class="texto01">Codigo: </td>
  <td height="20"><font face="Arial, Helvetica, sans-serif">
    <input name="codigo" type="TEXT" class="campos" id="codigo" value="<?php echo $codigo?>" size="41" maxlength="100">
  </font></td>
</tr>
<tr>
  <td height="20" colspan="2" align="right" class="texto01">Titulo Noticia : </td>
  <td width="508" height="20"><font face="Arial, Helvetica, sans-serif">
    <textarea name="titulo" cols="41" rows="3" class="campos" id="titulo"></textarea>
    <input name="log" type="hidden" id="log" value="1">
    <input name="exec" type="hidden" id="exec" value="1">
  </font></td>
</tr>
<tr>
  <td height="20" colspan="2" align="right" class="texto01">Tipo :
  <td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
    <select name="tipo" class="campos" id="tipo">
      <option value="0">-- Tipo --</option>
      <option value="N">Nacional</option>
      <option value="I">Internacional</option>
    </select>
  </font>
</tr>
<tr>
  <td height="20" colspan="2" align="right" class="texto01">Peri&oacute;dico :
  <td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
    <?php
      mysql_select_db($database_conexion, $conexion);
      $query_rsFunc1 = "SELECT * from periodicos ";
      $rsFunc1 = mysql_query($query_rsFunc1, $conexion) or die(mysql_error());
      $row_rsFunc1 = mysql_fetch_assoc($rsFunc1);
?>
    <select name="periodico" class="campos" id="periodico">
      <option value="0">-- Nombre del Periodico --</option>
      <?php do {?>
        <option value="<?php echo $row_rsFunc1['nombre_per']?>"><?php echo $row_rsFunc1['nombre_largo_per']?></option>
      <?php }while ($row_rsFunc1 = mysql_fetch_assoc($rsFunc1));?>
    </select>
</font>
</tr>
<tr>
  <td height="20" colspan="2" align="right" class="texto01"> Fecha :</td>
  <td height="20" align="left" valign="middle"><span class="textoper04">D&iacute;a</span>
    <select name="dia" class="campos">
      <option value="0">--</option>
      <option value="01">1</option>
      <option value="02">2</option>
      <option value="03">3</option>
      <option value="04">4</option>
      <option value="05">5</option>
      <option value="06">6</option>
      <option value="07">7</option>
      <option value="08">8</option>
      <option value="09">9</option>
      <option value="10">10</option>
      <option value="11">11</option>
      <option value="12">12</option>
      <option value="13">13</option>
      <option value="14">14</option>
      <option value="15">15</option>
      <option value="16">16</option>
      <option value="17">17</option>
      <option value="18">18</option>
      <option value="19">19</option>
      <option value="20">20</option>
      <option value="21">21</option>
      <option value="22">22</option>
      <option value="23">23</option>
      <option value="24">24</option>
      <option value="25">25</option>
      <option value="26">26</option>
      <option value="27">27</option>
      <option value="28">28</option>
      <option value="29">29</option>
      <option value="30">30</option>
      <option value="31">31</option>
    </select>
    <span class="textoper04">Mes</span>
    <select name="mes" class="campos">
      <option value="0">-----</option>
      <option value="01">Enero</option>
      <option value="02">Febrero</option>
      <option value="03">Marzo</option>
      <option value="04">Abril</option>
      <option value="05">Mayo </option>
      <option value="06">Junio</option>
      <option value="07">Julio</option>
      <option value="08">Agosto</option>
      <option value="09">Septiembre</option>
      <option value="10">Octubre</option>

```

```

<option value="11">Noviembre </option>
<option value="12">Diciembre</option>
</select>
<span class="textoper04">A&ntilde;o</span>
<select name="ano" class="campos">
<option value="0">----</option>
<option value="1969">1969</option>
<option value="1970">1970</option>
<option value="1971">1971</option>
<option value="1972">1972</option>
<option value="1973">1973</option>
<option value="1974">1974</option>
<option value="1975">1975</option>
<option value="1976">1976</option>
<option value="1977">1977</option>
<option value="1978">1978</option>
<option value="1979">1979</option>
<option value="1980">1980</option>
<option value="1981">1981</option>
<option value="1982">1982</option>
<option value="1983">1983</option>
<option value="1984">1984</option>
<option value="1985">1985</option>
<option value="1986">1986</option>
<option value="1987">1987</option>
<option value="1988">1988</option>
<option value="1989">1989</option>
<option value="1990">1990</option>
<option value="1991">1991</option>
<option value="1992">1992</option>
<option value="1993">1993</option>
<option value="1994">1994</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
<option value="2005">2005</option>
<option value="2006">2006</option>
<option value="2007">2007</option>
</select>
</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01"> Secci&oacute;n :
<td width="508" height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<?php
mysql_select_db($database_conexion, $conexion);
$query_rsFunc2 = "SELECT * from secciones ";
$rsFunc2 = mysql_query($query_rsFunc2, $conexion) or die(mysql_error());
$row_rsFunc2 = mysql_fetch_assoc($rsFunc2);
?>

<select name="seccion" class="campos" id="seccion">
<option value="0">-- Nombre de la Secci&oacute;n --</option>
<?php do {>
<option value="<?php echo $row_rsFunc2[nombre_sec]?>"><?php echo $row_rsFunc2[nombre_sec]?></option>
<?php }while ($row_rsFunc2 = mysql_fetch_assoc($rsFunc2));?>
</select>
</font>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01"><input type="hidden" name="MAX_FILE_SIZE" value="6000000">
Archivo :</td>
<td height="20" align="left"><input name="userfile" type="file" class="campos" id="userfile">
</td>
</tr>
<tr>
<td colspan="3" align="right" class="lineainf">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td align="right" width="18">&nbsp;&nbsp;&nbsp;</td>
<td align="left" width="215" valign="bottom">&nbsp;&nbsp;&nbsp;</td>
<td align="right" valign="bottom"> <font face="Arial, Helvetica, sans-serif">
<input name="Grabar" type="button" class="botones" id="Grabar" value="Grabar" onClick="validar()">
</font></td>
</tr>
</table>
</form>
<script language="JavaScript" type="text/JavaScript">
document.form1.dia.value="<?php echo date("d")?>";
document.form1.mes.value="<?php echo date("m")?>";
document.form1.ano.value="<?php echo date("Y")?>";
function validar() {
if (document.form1.seccion.value==0 | document.form1.periodico.value==0){
return alert("Debe ingresar un periodico y una secci&oacute;n");
}
if (document.form1.userfile.value==0){
return alert("Debe ingresar un archivo");
}
if (document.form1.tipo.value==0){
return alert("Debe ingresar un tipo");
}
else{
document.form1.submit();
}
}
</script>

```



```
</td>
<td width="50%" align="right" valign="top"></td>
</tr>
</table>
</TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>
<?php } else {
?>
<p><font face="Arial, Helvetica, sans-serif" size="4" color="#CCCCCC">ADVERTENCIA:</font></p>
<blockquote>
<p><font face="Arial, Helvetica, sans-serif" size="2"> <font color="#000000" size="1">Usted
no tiene permisos para ingresar</font></font>

</p>
</blockquote>
<?php }?>
<?php //if ($modi==1){?>
<script language="JavaScript" type="text/JavaScript">
function modificar(path) {
//window.open('modificar.php?fecha=<?php echo $fecha ?>&path=' + path +
", 'Servicio ', width=350,height=300,screenX=200,screenY=200,scrollbars=no,toolbar=no,menubar=no,statusbar=yes,resizable=no,locationbar=
no,hotkeys=no");
}
</script>
<?php //} ?>
</BODY>
</HTML>
```

```
<?php
$hostname_conexion = "localhost:3306";
$database_conexion = "biblioteca";
$username_conexion = "root";
$password_conexion = "";
$conexion = mysql_connect($hostname_conexion, $username_conexion, $password_conexion) or
trigger_error(mysql_error(),E_USER_ERROR);
?>
```

```

<HTML>
<HEAD>
<SCRIPT LANGUAGE="VBScript" RUNAT="Server">
  <!--
    option explicit
  -->
</SCRIPT>
<TITLE>Biblioteca Virtual - Principal</TITLE>
<LINK REL=STYLESHEET HREF="is2style.css" TYPE="text/css">
<META NAME="DESCRIPTION" CONTENT="Sample query form for Microsoft Index Server">
<META NAME="AUTHOR" CONTENT="Index Server Team">
<META NAME="KEYWORDS" CONTENT="query, content, hit">
<META NAME="SUBJECT" CONTENT="sample form">
<META NAME="MS.CATEGORY" CONTENT="Internet">
<META NAME="MS.LOCALE" CONTENT="EN-US">
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<%
  DebugFlag = FALSE
  NewQuery = FALSE
  UseSavedQuery = FALSE
  SearchString = ""
  QueryForm = Request.ServerVariables("PATH_INFO")
    cl=0

  if Request.ServerVariables("REQUEST_METHOD") = "POST" then
    ' Propiedad de Búsqueda libre
    SearchString = Request.Form("SearchString")
    ' Propiedad de Periodico
    DocAuthorRestriction = Request.Form("DocAuthorRestriction")
    ' Propiedad de Tipo
    DocSubject = Request.Form("DocSubject")
    ' Propiedad de Sección
    DocKeywords = Request.Form("DocKeywords")
    FSRest = Request.Form("FSRest")
    FSRestVal = Request.Form("FSRestVal")
    FSRestOther = Request.Form("FSRestOther")
    FMMod = Request.Form("FMMod")
    ' Propiedad de Fecha Periódico
    filename = Request.Form("ano") & Request.Form("mes") & Request.Form("dia")
    ' Propiedad Adicionales de Búsqueda
    SortBy = Request.Form("SortBy")
    Colset = Request.Form("ColChoice")
    Scope = Request.Form("Scope")
    RankBase = Request.Form("RankBase")
    ' NOTE: this will be true only if the button is actually pushed.
    if Request.Form("Action") = "Ejecutar" then
      NewQuery = TRUE
    elseif Request.Form("pg") <> "" then
      UseSavedQuery=TRUE
      NextPageNumber = Request.Form("pg")
    end if
  end if
%>

</HEAD>

<BODY OnLoad="SetAll()" LANGUAGE=JavaScript bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0"
text="#003300" vlink="#996600">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap><script language=JavaScript>
<!--
function SetAll() {

  document.QForm.SearchString.value = "<%=SearchString%>";

  <%select case FSRest%>
  <%case " " %>
    document.QForm.FSRest.selectedIndex = 1;
  <%case else%>
    document.QForm.FSRest.selectedIndex = 0;
  <%end select%>

  <%select case FSRestVal%>
  <%case "any"%>
    document.QForm.FSRestVal.selectedIndex = 0;
  <%case "100"%>
    document.QForm.FSRestVal.selectedIndex = 1;
  <%case "1024"%>
    document.QForm.FSRestVal.selectedIndex = 2;
  <%case "10240"%>
    document.QForm.FSRestVal.selectedIndex = 3;
  <%case "102400"%>
    document.QForm.FSRestVal.selectedIndex = 4;

```

```

<%case "1048576"%>
document.QForm.FSRestVal.selectedIndex = 5;
<%case "10485760"%>
document.QForm.FSRestVal.selectedIndex = 6;
<%case "104857600"%>
document.QForm.FSRestVal.selectedIndex = 7;
<%case "other"%>
document.QForm.FSRestVal.selectedIndex = 8;
<%case else%>
document.QForm.FSRestVal.selectedIndex = 0;
<%end select%>

document.QForm.FSRestOther.value = "<%=FSRestOther%>";
document.QForm.dia.value = "<%=Request.Form("dia")%>";
document.QForm.mes.value = "<%=Request.Form("mes")%>";
document.QForm.ano.value = "<%=Request.Form("ano")%>";
document.QForm.DocAuthorRestriction.value = "<%=DocAuthorRestriction%>";
document.QForm.DocSubject.value = "<%=DocSubject%>";
document.QForm.DocKeywords.value = "<%=DocKeywords%>";

}
!-->
</script>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="70%" align="right"><form name="QForm" action="<%=QueryForm%>" method=POST>
<!-- -->
<table width=97% align="right">
<tr>
<td><font size="3" face="Arial, Helvetica, sans-serif" class="texto01"><b>Buscar texto : </b></font></td>
<td width="508"><font face="Arial, Helvetica, sans-serif">
<input type="TEXT" name="SearchString" size="51" maxlength="100" value="">
</font></td>
</tr>
<tr>
<td class="texto01">Tipo :
<td align="left"><font face="Arial, Helvetica, sans-serif">
<select name="DocSubject" class="campos" id="DocSubject">
<option value="">-- Seleccione Tipo --</option>
<option value="N">Nacional</option>
<option value="I">Internacional</option>
</select>
</font>
</td>
<tr>
<td class="texto01">Peri&oacute;dico :
<td align="left">
<%
set ObjConn = Server.CreateObject("ADODB.Connection")
ObjConn.Open "Provider=MSDASQL;DSN=biblioteca;UID=root;PWD=";
set rsfunc1= ObjConn.Execute("SELECT * from periodicos")
%>
<font face="Arial, Helvetica, sans-serif"> <select name="DocAuthorRestriction" class="campos"
id="DocAuthorRestriction">
<option value="">-- Seleccione Periodico --</option>
<%while (Not rsFunc1.EOF)%>
<option value="<%=rsfunc1("nombre_per")%>"><%=rsfunc1("nombre_largo_per")%></option>
<%rsFunc1.MoveNext
wend%>

</select>
</font>
</td>
<tr>
<td align="right">
<div align="left" class="texto01"><font face="Arial, Helvetica, sans-serif" size="2">Seccion:</font></div>
<td align="left" width="508"><font face="Arial, Helvetica, sans-serif">
<%
set ObjConn = Server.CreateObject("ADODB.Connection")
ObjConn.Open "Provider=MSDASQL;DSN=biblioteca;UID=root;PWD=";
set rsfunc2= ObjConn.Execute("SELECT * from secciones")
%>
<select name="DocKeywords" class="campos" id="DocKeywords">
<option value="">-- Seleccione Secci&oacute;n --</option>
<%while (Not rsFunc2.EOF)%>
<option value="<%=rsfunc2("nombre_sec")%>"><%=rsfunc2("nombre_sec")%></option>
<%rsFunc2.MoveNext
wend%>

</select>
</font>
</td>
<tr>
<td align="left"><div align="left"><font size="2" face="Arial, Helvetica, sans-serif" class="texto01">Tamaño : </font></div></td>
<td align="left" width="508"><font face="Arial, Helvetica, sans-serif">
<select name="FSRest" class="campos">
<option value=" &lt; " selected>Menor que</option>
<option value=" &gt; " >Mayor que</option>
</select>
<select name="FSRestVal" class="campos">
<option value="any" selected>tama&ntilde;o</option>
<option value="100">100 bytes </option>
<option value="1024">1K bytes </option>
<option value="10240">10K bytes </option>
<option value="102400">100K bytes </option>
<option value="1048576">1M byte </option>
<option value="10485760">10M bytes </option>
<option value="104857600">100M bytes </option>
<option value="other">Otro</option>
</select>
<input name="FSRestOther" type="TEXT" class="campos" value="" size="12" maxlength="9">
</font></td>

```

```

</tr>
<tr>
<td align="left" class="texto01">
<div align="left"></div> <span class="style2"><font face="Arial, Helvetica, sans-serif" size="2">Fecha</font></span><span
class="texto01"><font face="Arial, Helvetica, sans-serif" size="2">:</font></span><font face="Arial, Helvetica, sans-serif" size="2">&nbsp;</font></td>
<td align="left" width="508"><font face="Arial, Helvetica, sans-serif">
<span class="textoper04">D&iacut;a</span>
<select name="dia" class="campos">
<option value="">--</option>
<option value="01">1</option>
<option value="02">2</option>
<option value="03">3</option>
<option value="04">4</option>
<option value="05">5</option>
<option value="06">6</option>
<option value="07">7</option>
<option value="08">8</option>
<option value="09">9</option>
<option value="10">10</option>
<option value="11">11</option>
<option value="12">12</option>
<option value="13">13</option>
<option value="14">14</option>
<option value="15">15</option>
<option value="16">16</option>
<option value="17">17</option>
<option value="18">18</option>
<option value="19">19</option>
<option value="20">20</option>
<option value="21">21</option>
<option value="22">22</option>
<option value="23">23</option>
<option value="24">24</option>
<option value="25">25</option>
<option value="26">26</option>
<option value="27">27</option>
<option value="28">28</option>
<option value="29">29</option>
<option value="30">30</option>
<option value="31">31</option>
</select>
<span class="textoper04">Mes</span>
<select name="mes" class="campos">
<option value="">-----</option>
<option value="01">Enero</option>
<option value="02">Febrero</option>
<option value="03">Marzo</option>
<option value="04">Abril</option>
<option value="05">Mayo </option>
<option value="06">Junio</option>
<option value="07">Julio</option>
<option value="08">Agosto</option>
<option value="09">Septiembre</option>
<option value="10">Octubre</option>
<option value="11">Noviembre </option>
<option value="12">Diciembre</option>
</select>
<span class="textoper04">A&ntilde;o</span>
<select name="ano" class="campos">
<option value="">----</option>
<option value="1969">1969</option>
<option value="1970">1970</option>
<option value="1971">1971</option>
<option value="1972">1972</option>
<option value="1973">1973</option>
<option value="1974">1974</option>
<option value="1975">1975</option>
<option value="1976">1976</option>
<option value="1977">1977</option>
<option value="1978">1978</option>
<option value="1979">1979</option>
<option value="1980">1980</option>
<option value="1981">1981</option>
<option value="1982">1982</option>
<option value="1983">1983</option>
<option value="1984">1984</option>
<option value="1985">1985</option>
<option value="1986">1986</option>
<option value="1987">1987</option>
<option value="1988">1988</option>
<option value="1989">1989</option>
<option value="1990">1990</option>
<option value="1991">1991</option>
<option value="1992">1992</option>
<option value="1993">1993</option>
<option value="1994">1994</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
<option value="2005">2005</option>
<option value="2006">2006</option>
<option value="2007">2007</option>
</select>
</font></td>
</tr>
<tr>

```

```

<td align="right">&nbsp;  </td>
<td align="center" width="508"> <font face="Arial, Helvetica, sans-serif">
  <input type="SUBMIT" name="Action" value="Ejecutar" class="botones">
  <input type="RESET" name="Limpiar" value="Limpiar" class="botones">
</font></td>
</tr>
</table>
<input type="HIDDEN" name="SortBy" value="rank[dj]>
<input type="HIDDEN" name="ColChoice" value="1">
<input type="HIDDEN" name="Scope" value="/biblioteca/periodicos">
</form></td>
  <td width="30%" align="right"></td>
</tr>
</table>
</TD>
</TR>
</TABLE>
<%=Msgborrar%><%=MsgStatus%> <BR>
<%if DebugFlag then

    'Variables reservadas %>
    <PRE>
SearchString      = <%=SearchString%>
DocAuthorRestriction = <%=DocAuthorRestriction%>
    DocSubject      = <%=DocSubject%>
    DocKeywords     = <%=DocKeywords%>
FSRest            = <%=FSRest%>
FSRestVal         = <%=FSRestVal%>
FSRestOther       = <%=FSRestOther%>
FMMod             = <%=FMMod%>
filename          = <%=filename%>
SortBy            = <%=SortBy%>
Colset            = <%=Colset%>
Scope             = <%=Scope%>
NewQuery          = <%=CStr(NewQuery)%>
UseSavedQuery     = <%=CStr(UseSavedQuery)%>
    </PRE>
<%end if%>
<%
if NewQuery then
set Session("Query") = nothing
set Session("Recordset") = nothing
NextRecordNumber = 1
set Q = Server.CreateObject("ixsso.Query")

    Composer = ""
    TheQuery = ""
' Consulta de Texto libre buscado
if SearchString <> "" then
    if Left( SearchString, 1 ) <> "@" AND Left( SearchString, 1 ) <> "#" AND Left( SearchString, 1 ) <> "$" then
        TheQuery = "@Contents " + SearchString
    else
        TheQuery = SearchString
    end if

    Composer = " & "
end if
' Consual de tamaño de archivo
if FSRestVal <> "any" then
    if FSRestVal <> "other" then
        TheQuery = "(@Size " + FSRest + FSRestVal + ") " + Composer + TheQuery
    else
        TheQuery = "(@Size " + FSRest + FSRestOther + ") " + Composer + TheQuery
    end if

    Composer = " & "
end if
' Consulta de Periodico
if DocAuthorRestriction <> "" then
    TheQuery = "(@DocAuthor " + DocAuthorRestriction + ") " + Composer + TheQuery

    Composer = " & "
end if
' Consulta de Tipo
    if DocSubject <> "" then
        TheQuery = "(@DocSubject " + DocSubject + ") " + Composer + TheQuery
        Composer = " & "
    end if
' Consulta de la Seccion
    TheQuery = "(@DocKeywords " + DocKeywords + ") " + Composer + TheQuery

    Composer = " & "
end if
' Conusta de la Fecha
    if filename <> "" then
        TheQuery = "(#filename "*" + filename + "*.pdf) " + Composer + TheQuery
        Composer = " & "
    end if

' Consulta de Fecha
' if FMMod <> "" AND FMMod <> "any" then
' if FMMod <> "since" then
'     TheQuery = "(@Write > " + FMMod + ") " + Composer + TheQuery
' else
'     TheQuery = "(@Write > " + filename + ") " + Composer + TheQuery
' end if
' end if
%>
<%if DebugFlag then%>
TheQuery = <%=TheQuery%>

```

```

<%end if%>
<%
    Q.Query = TheQuery

    Q.SortBy = SortBy
' Scope es la restriccion donde buscar

    if Scope = "/biblioteca/periodicos" then
        set Util = Server.CreateObject("ixsso.Util")
        Util.AddScopeToQuery Q, Scope, "DEEP"
        set Util = nothing
    end if
    if ColSet = 1 then
        ' Columnas para el recordset o select.
        Q.Columns = "DocTitle, DocAuthor, DocSubject, DocKeywords, DocComments, create, vpath, path, filename, size, write,
characterization, rank, Access"
        RecordsPerPage = 10
    elseif ColSet = 2 then
        Q.Columns = "vpath, path"
        RecordsPerPage = 200
    elseif ColSet = 3 then
        Q.Columns = "filename, vpath, size, write"
        RecordsPerPage = 25
    end if

    set RS = Q.CreateRecordSet("nonsequential")

    ' Denfinimos tamano de paginado

    RS.PageSize = RecordsPerPage
    ActiveQuery = TRUE
    elseif UseSavedQuery then
        if IsObject( Session("Query") ) And IsObject( Session("RecordSet") ) then
            set Q = Session("Query")
            set RS = Session("RecordSet")
            if RS.RecordCount <> -1 and NextPageNumber <> -1 then
                RS.AbsolutePage = NextPageNumber
                NextRecordNumber = RS.AbsolutePosition
            end if
            ActiveQuery = TRUE
        else
            Response.Write "ERROR - No saved query"
        end if
    end if

    if ActiveQuery then
        if not RS.EOF then
%>

<form name="eliminar" method="post" action="<%=QueryForm%>" OnSubmit="pas();return valida">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td valign="bottom"></td>
<td align="left" valign="bottom" class="texto05"><%
        ' Resumen de registros encontrados
        LastRecordOnPage = NextRecordNumber + RS.PageSize - 1
        CurrentPage = RS.AbsolutePage
        if RS.RecordCount <> -1 AND RS.RecordCount < LastRecordOnPage then
            LastRecordOnPage = RS.RecordCount
        end if

        Response.Write "Archivos " & NextRecordNumber & " a " & LastRecordOnPage
        if RS.RecordCount <> -1 then
            Response.Write " de " & RS.RecordCount
        end if

        if SearchString <> "" then
            Response.Write " iguales a: " & chr(34) & "<!--"

            Response.Write SearchString & "</!--" & chr(34) & ".<P>"
        end if
%></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table>
<table width="100%" border=0 align="center" cellpadding="0" cellspacing="0" bgcolor="#E8E8E8" >
<colgroup width=105>
<% Do While Not RS.EOF and NextRecordNumber <= LastRecordOnPage

' This is the detail portion for Title, Abstract, URL, Size, and
' Modification Date.
' If there is a title, display it, otherwise display the filename.
' Graphically indicate rank of document with list of stars ("s).

        if NextRecordNumber = 1 then
            RankBase=RS("rank")

            if RankBase>1000 then
                RankBase=1000
            elseif RankBase<1 then
                RankBase=1
            end if

        end if

        NormRank = RS("rank")/RankBase

        if NormRank > 0.80 then
            stars = "rankbtn5.gif"
        elseif NormRank > 0.60 then
            stars = "rankbtn4.gif"

```

```

elseif NormRank > 0.40 then
    stars = "rankbtn3.gif"
elseif NormRank > .20 then
    stars = "rankbtn2.gif"
else stars = "rankbtn1.gif"

end if

dim ubc
dim tot
dim corte
' Ruta fisica del archivo cuando presionamos click
ubc= len(RS("path"))
tot= ubc - 30
corte2= right(RS("path"),tot)
corte = Replace(corte2, " ", "-")
parf1= Mid(RS("filename"), 5, 4)
parf2= Mid(RS("filename"), 9, 2)
parf3= Mid(RS("filename"), 11, 2)
fecha=parf1 & "/" & parf2 & "/" & parf3
' Inicio de salida de informacion segun las propiedades pedidas
%>
<tr class="RecordTitle">
<td width="26" rowspan="4" align="right" valign=bottom bgcolor="#AEC2E5" class="RecordTitle"><table width="26" height="110"
border="0" cellpadding="0" cellspacing="0" background="images2/buellet01.jpg">
<tr>
<td class="texto04"><%= NextRecordNumber%></td>
</tr>
</table> <font face="Arial, Helvetica, sans-serif" size="2">
</font></td>
<td valign="bottom" class="margen3"> <font size="2" face="Arial, Helvetica, sans-serif" class="margen3"><b class="RecordTitle">
</b></font>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="180"><table width="167" height="27" border="0" align="left" cellpadding="0" cellspacing="0"
background="images2/etiqueta.jpg">
<tr>
<td width="67" align="right" class="textoper01">Peri&oacute;dico: </td>
<td class="textoper02">&nbsp;</td>
<td class="textoper02"><%= Server.HTMLEncode(RS("DocAuthor"))%></td>
</tr>
</table></td>
<td width="50" align="left" class="textoper08"><font class="margen3"><b class="RecordTitle">
Titulo:
</b> </font></td>
<td colspan="6" align="left" class="texto04">
<%if VarType(RS("DocTitle")) = 1 or RS("DocTitle") = "" then%>
<a href="<%=corte2%>" class="RecordTitle"><%= Server.HTMLEncode( RS("filename") )%></a>
<%else%>
<a href="<%=corte2%>" class="RecordTitle"><%= Server.HTMLEncode(RS("DocTitle"))%></a>
<%end if%>
</b>
<%
' Construct the URL for hit highlighting
WebHitsQuery = "CiWebHitsFile=" & Server.URLEncode( RS("vpath") )
WebHitsQuery = WebHitsQuery & "&CiRestriction=" & Server.URLEncode( Q.Query )
WebHitsQuery = WebHitsQuery & "&CiBeginHilite=" & Server.URLEncode( "<b class=Hit=" )
WebHitsQuery = WebHitsQuery & "&CiEndHilite=" & Server.URLEncode( "</b>" )
WebHitsQuery = WebHitsQuery & "&CiUserParam3=" & QueryForm
WebHitsQuery = WebHitsQuery & "&CiLocale=" & Q.LocaleID

%>
</font></td>
</tr>
<tr>
<td>&nbsp;</td>
<td align="left" class="textoper08">Seccion:</td>
<td width="100" align="left" valign="top" class="margen3"><font class="textoper010"><%= Server.HTMLEncode( RS("DocKeywords")
)%>
</font></td>
<td width="50" align="left" valign="top" class="textoper08">Tipo:</td>
<td width="25" align="left" valign="top" class="textoper010"><font class="textoper010">
<%if Server.HTMLEncode( RS("DocSubject") ) = 1 then Response.write("Internacional") else Response.Write("Nacional") end
if%></font></td>
<td width="25" align="left" valign="top" class="textoper010">&nbsp;</td>
<td width="60" align="left" class="textoper01">Fecha:</td>
<td width="200" align="left" valign="top" class="textoper010"><%= fecha%></td>
</tr>
</table>
<font size="2" face="Arial, Helvetica, sans-serif" class="margen3"><b class="RecordTitle"> </b></font></td>
<td class="margen3"><font size="2" face="Arial, Helvetica, sans-serif" class="margen3">"></font></td>
</tr>
<tr>
<td valign=top class="margenex"><span class="textoper08">Resumen:</span> <span class="textoper09"><%=
Server.HTMLEncode(RS("characterization"))%>
<%if RS("size") = "" then%>
(tama&ntilde;o y fecha desconocida)
<%else%>
Tama&ntilde;o: <%=RS("size")%> bytes
<%end if%></span>
<p> </td>
<td valign=top class="margen">
</font></td>
</tr>
<tr>
<td valign=top class="margen"><span class="textoper08">Ubicaci&oacute;n: </span><i class="RecordStats"><a href="<%=corte2%>"
class="textoper07"><%=RS("path")%></a> </i></font></td>
<td valign=top class="margen">&nbsp;</td>
</tr>
<tr>
<td colspan="2" align="left" valign=top ></td>
</tr>
<tr bgcolor="#FFFFFF">

```



```

<td height="2" colspan="3" align="right" ></td>
</tr>
<%
    RS.MoveNext
    NextRecordNumber = NextRecordNumber+1
Loop
%>
</table>
<P align="right"><font face="Arial, Helvetica, sans-serif" size="2"><i class="RecordStats"> </i> </font>
<P align="right"><font face="Arial, Helvetica, sans-serif" size="2"><i class="RecordStats">
<%
'Variables de formulario de busqueda para el paginado
%>
<input type="hidden" name="test" value="">
<INPUT TYPE="HIDDEN" NAME="SearchString" VALUE="<%=SearchString%>">
<INPUT TYPE="HIDDEN" NAME="SortBy" VALUE="rank[d]">
<input type="HIDDEN" name="ColChoice" value="1">
<INPUT TYPE="HIDDEN" NAME="Scope" VALUE="/diseno/archivos">
<INPUT TYPE="HIDDEN" NAME="DocAuthorRestriction" VALUE="<%=DocAuthorRestriction%>">
    <INPUT TYPE="HIDDEN" NAME="DocSubject" VALUE="<%=DocSubject%>">
    <INPUT TYPE="HIDDEN" NAME="DocKeywords" VALUE="<%=DocKeywords%>">
<INPUT TYPE="HIDDEN" NAME="FSRest" VALUE="<%=FSRest%>">
<INPUT TYPE="HIDDEN" NAME="FSRestVal" VALUE="<%=FSRestVal%>">
<INPUT TYPE="HIDDEN" NAME="FSRestOther" VALUE="<%=FSRestOther%>">
<INPUT TYPE="HIDDEN" NAME="FMMod" VALUE="<%=FMMod%>">
<INPUT TYPE="HIDDEN" NAME="filename" VALUE="<%=filename%>">
</i><BR>
</font>
</form>
<span class="texto04">
<%
else ' NOT RS.EOF
if NextRecordNumber = 1 then
    Response.Write "<div class=texto03> No hay coincidencias iguales a lo especificado </div><P>"
else
    Response.Write "No hay mas archivos iguales a lo especificado<P>"
end if

end if ' NOT RS.EOF
%>

<!-- If the index is out of date, display the fact -->
<%if NOT Q.OutOfDate then%>
</span></font>
<P> <span class="texto04">El formulario se est&aacute; actualizando</i>
</span>
<P>
<span class="texto04">
<%end if%>

<%if Q.QueryIncomplete then%>
</span>
<P> <span class="texto04"><b>La busqueda es demasiado extensa para completar.</b></i><BR>
<%end if%>

<%if Q.QueryTimedOut then%>
</span>
<P> <span class="texto04"><b>Su busqueda tomara demasiado tiempo para completar.</b></i><BR>
<%end if%>
</span>
<TABLE align="left">
<!--
    This is the "previous" button.
    This retrieves the previous page of documents for the query.
-->
<%SaveQuery = FALSE%>
<%if CurrentPage > 1 and RS.RecordCount < -1 then ' BUGBUG - use RS.Supports(adMovePrevious)%>
<td align=left><form action="<%=QueryForm%>" method="post">
    <INPUT TYPE="HIDDEN" NAME="SearchString" VALUE="<%=SearchString%>">
    <INPUT TYPE="HIDDEN" NAME="DocAuthorRestriction" VALUE="<%=DocAuthorRestriction%>">
        <INPUT TYPE="HIDDEN" NAME="DocSubject" VALUE="<%=DocSubject%>">
        <INPUT TYPE="HIDDEN" NAME="DocKeywords" VALUE="<%=DocKeywords%>">
    <INPUT TYPE="HIDDEN" NAME="FSRest" VALUE="<%=FSRest%>">
    <INPUT TYPE="HIDDEN" NAME="FSRestVal" VALUE="<%=FSRestVal%>">
    <INPUT TYPE="HIDDEN" NAME="FSRestOther" VALUE="<%=FSRestOther%>">
    <INPUT TYPE="HIDDEN" NAME="FMMod" VALUE="<%=FMMod%>">
    <INPUT TYPE="HIDDEN" NAME="filename" VALUE="<%=filename%>">
    <INPUT TYPE="HIDDEN" NAME="SortBy" VALUE="<%=SortBy%>">
    <INPUT TYPE="HIDDEN" NAME="Colset" VALUE="<%=ColChoice%>">
    <INPUT TYPE="HIDDEN" NAME="Scope" VALUE="<%=Scope%>">
    <INPUT TYPE="HIDDEN" name="pg" VALUE="<%=CurrentPage-1%>" >
    <INPUT TYPE="HIDDEN" NAME="RankBase" VALUE="<%=RankBase%>">
    <input type="submit" value="<%=RS.PageSize%> Documentos Anteriores" class="botones">
</form></td>
<%SaveQuery = TRUE%>
<%end if%>
<!--
    This is the "next" button.
    This button retrieves the next page of documents for the query.
-->
<%if Not RS.EOF then%>
<td align=right><form action="<%=QueryForm%>" method="post">
    <INPUT TYPE="HIDDEN" NAME="SearchString" VALUE="<%=SearchString%>">
    <INPUT TYPE="HIDDEN" NAME="DocAuthorRestriction" VALUE="<%=DocAuthorRestriction%>">
        <INPUT TYPE="HIDDEN" NAME="DocSubject" VALUE="<%=DocSubject%>">
        <INPUT TYPE="HIDDEN" NAME="DocKeywords" VALUE="<%=DocKeywords%>">
    <INPUT TYPE="HIDDEN" NAME="FSRest" VALUE="<%=FSRest%>">
    <INPUT TYPE="HIDDEN" NAME="FSRestVal" VALUE="<%=FSRestVal%>">
    <input type="HIDDEN" name="FSRestOther2" value="<%=FSRestOther%>">
    <input type="HIDDEN" name="FMMod2" value="<%=FMMod%>">
    <INPUT TYPE="HIDDEN" NAME="filename" VALUE="<%=filename%>">
    <INPUT TYPE="HIDDEN" NAME="SortBy" VALUE="<%=SortBy%>">
    <INPUT TYPE="HIDDEN" NAME="Colset" VALUE="<%=ColChoice%>">

```

```
<INPUT TYPE="HIDDEN" NAME="Scope" VALUE="<%=Scope%>">
<INPUT TYPE="HIDDEN" name="pg" VALUE="<%=CurrentPage+1%>" >
<INPUT TYPE="HIDDEN" NAME = "RankBase" VALUE="<%=RankBase%>">
<% NextString = ""
   if RS.RecordCount <> -1 then
       NextSet = (RS.RecordCount - NextRecordNumber) + 1
       if NextSet > RS.PageSize then
           NextSet = RS.PageSize
       end if
       NextString = NextString & NextSet & " Documentos Siguientes"
   else
       NextString = NextString & " pagina de documentos"
   end if
%>
<input type="submit" value="<%=NextString%>" class="botones">
</form></td>
<%SaveQuery = TRUE%>
<%end if%>
</TABLE>
<!-- Display the page number -->
<div align="center" class="texto04">Pagina <%=CurrentPage%>
<%if RS.PageCount <> -1 then
   Response.Write " de " & RS.PageCount
end if %>
<%
' If either of the previous or back buttons were displayed, save the query
' and the recordset in session variables.
if SaveQuery then
   set Session("Query") = Q
   set Session("RecordSet") = RS
else
   RS.close
   Set RS = Nothing
   Set Q = Nothing
   set Session("Query") = Nothing
   set Session("RecordSet") = Nothing
end if
%>
<% end if %>
</div></td>
</tr>
</table>
<!--#include file ="is2foot.inc"-->

</BODY>
</HTML>
```

```

<?php require_once('conecta.php');
mysql_select_db($database_conexion, $conexion);
$query_Recordt = "SELECT max(codigo_inf) as codigo FROM informacion";
$Recordt = mysql_query($query_Recordt, $conexion) or die(mysql_error());
$row_Recordt = mysql_fetch_assoc($Recordt);
$codigo=$row_Recordt['codigo']+1;

$query_Recordset1 = "SELECT * FROM informacion WHERE codigo_inf=" . $_REQUEST['codigo_inf'] . "" ;
$Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

$msg="";
if (isset($_REQUEST['log'])){
    if (isset($_REQUEST['Modificar'])){
        $titulo=$_REQUEST['titulo'];
        $tipo=$_REQUEST['tipo'];
        $periodico=strtoupper(substr($_REQUEST['periodico'],0,3));
        $dia=$_REQUEST['dia'];
        $mes=$_REQUEST['mes'];
        $ano=$_REQUEST['ano'];
        $seccion=strtoupper(substr($_REQUEST['seccion'],0,3));
        if (file_exists('periodicos/' . $tipo)){
            $carpetatmp= mkdir ('periodicos/' . $tipo, 0700);
        }
        if (!file_exists('periodicos/' . $tipo . "/" . $periodico )){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico, 0700);
        }
        if (file_exists('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia )){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia , 0700);
        }

        if (!file_exists('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion)){
            $carpetatmp= mkdir ('periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion,
0700);
        }

        $archivo= $tipo . $periodico . $ano . $mes . $dia . $seccion . $codigo;
        $uploaddir = 'periodicos/' . $tipo . "/" . $periodico . "/" . $ano . $mes . $dia . "/" . $seccion . "/";
        $uploadfile = $uploaddir . $archivo . ".pdf";

        copy($row_Recordset1['archivo_inf'], $uploadfile);
        unlink($row_Recordset1['archivo_inf']);
        //rmdir (substr($row_Recordset1['archivo_inf'], 0,30));

        $insertSQL = "UPDATE informacion
SET periodo_inf = " . $_REQUEST['periodico'] . " ,
tipo_inf = " . $_REQUEST['tipo'] . " ,
fecha_inf = " . $_REQUEST['ano'] . "-" . $_REQUEST['mes'] . "-" .
$_REQUEST['dia'] . " ,
seccion_inf = " . $_REQUEST['seccion'] . " ,
archivo_inf = " . $uploadfile . " ,
titulo_inf = " . $_REQUEST['titulo'] . "

WHERE
codigo_inf=" . $_REQUEST['codigo_inf'];
mysql_select_db($database_conexion, $conexion);
$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

$msg="El archivo fue actualizado en la ubicación solicitada.\n";
    }

    if (isset($_REQUEST['Eliminar'])){
        $insertSQL = "DELETE FROM informacion
WHERE
codigo_inf=" . $_REQUEST['codigo_inf'];
mysql_select_db($database_conexion, $conexion);
$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

        unlink($row_Recordset1['archivo_inf']);

        $msg="El archivo fue eliminado en la ubicación solicitada.\n";
        //rmdir (substr($row_Recordset1['archivo_inf'], 0,30));
    }
}

$query_Recordset1 = "SELECT * FROM informacion WHERE codigo_inf=" . $_REQUEST['codigo_inf'] . "" ;
$Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Principal</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="STYLESHEET" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>

```

```

</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="70%" align="right" valign="top">
<form action="detalle_inf.php" method="post" enctype="multipart/form-data" name="form1">
<table width="97%" align="right">
<tr align="right">
<td height="40" colspan="3" class="textoper06"><?php echo $msg?></td>
</tr>
<tr>
<td height="40" colspan="3" class="textoper03">Mantenimiento del Sistema de Archivos </td>
</tr>
<tr>
<td colspan="3" class="lineasup">&nbsp;  </td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Codigo: </td>
<td height="20"><font face="Arial, Helvetica, sans-serif">
<input name="codigo_inf" type="TEXT" class="campos" id="codigo_inf" value="<?php echo $row_Recordset1['codigo_inf']?>"
size="41" maxlength="100">
</font></td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Titulo Noticia : </td>
<td width="508" height="20"><font face="Arial, Helvetica, sans-serif">
<input name="titulo" type="TEXT" class="campos" id="titulo" value="<?php echo $row_Recordset1['titulo_inf']?>" size="41"
maxlength="100">
<input name="log" type="hidden" id="log" value="1">
</font></td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Tipo :
<td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<select name="tipo" class="campos" id="tipo">
<option value="0">-- Tipo --</option>
<option value="N">Nacional</option>
<option value="I">Internacional</option>
</select>
</font>
</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Peri&oacute;dico :
<td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<?php
mysql_select_db($database_conexion, $conexion);
$query_rsFunc1 = "SELECT * from periodicos ";
$rsFunc1 = mysql_query($query_rsFunc1, $conexion) or die(mysql_error());
$row_rsFunc1 = mysql_fetch_assoc($rsFunc1);
?>
<select name="periodico" class="campos" id="periodico">
<option value="0">-- Nombre del Periodico --</option>
<?php do {?>
<option value="<?php echo $row_rsFunc1['nombre_per']?>"><?php echo $row_rsFunc1['nombre_largo_per']?></option>
<?php }while ($row_rsFunc1 = mysql_fetch_assoc($rsFunc1));?>
</select>
</font>
</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01"> Fecha :</td>
<td height="20" align="left" valign="middle"><span class="textoper04">D&iacute;a</span>
<select name="dia" class="campos">
<option value="0">--</option>
<option value="01">1</option>
<option value="02">2</option>
<option value="03">3</option>
<option value="04">4</option>
<option value="05">5</option>
<option value="06">6</option>
<option value="07">7</option>
<option value="08">8</option>
<option value="09">9</option>
<option value="10">10</option>
<option value="11">11</option>
<option value="12">12</option>
<option value="13">13</option>
<option value="14">14</option>
<option value="15">15</option>
<option value="16">16</option>
<option value="17">17</option>
<option value="18">18</option>
<option value="19">19</option>
<option value="20">20</option>
<option value="21">21</option>
<option value="22">22</option>
<option value="23">23</option>
<option value="24">24</option>
<option value="25">25</option>
<option value="26">26</option>
<option value="27">27</option>
<option value="28">28</option>
<option value="29">29</option>
<option value="30">30</option>
<option value="31">31</option>
</select>
<span class="textoper04">Mes</span>
<select name="mes" class="campos">
<option value="0">-----</option>
<option value="01">Enero</option>

```

```

<option value="02">Febrero</option>
<option value="03">Marzo</option>
<option value="04">Abril</option>
<option value="05">Mayo </option>
<option value="06">Junio</option>
<option value="07">Julio</option>
<option value="08">Agosto</option>
<option value="09">Septiembre</option>
<option value="10">Octubre</option>
<option value="11">Noviembre </option>
<option value="12">Diciembre</option>
</select>
<span class="textoper04">A&ntilde;o</span>
<select name="ano" class="campos">
<option value="0">----</option>
<option value="1969">1969</option>
<option value="1970">1970</option>
<option value="1971">1971</option>
<option value="1972">1972</option>
<option value="1973">1973</option>
<option value="1974">1974</option>
<option value="1975">1975</option>
<option value="1976">1976</option>
<option value="1977">1977</option>
<option value="1978">1978</option>
<option value="1979">1979</option>
<option value="1980">1980</option>
<option value="1981">1981</option>
<option value="1982">1982</option>
<option value="1983">1983</option>
<option value="1984">1984</option>
<option value="1985">1985</option>
<option value="1986">1986</option>
<option value="1987">1987</option>
<option value="1988">1988</option>
<option value="1989">1989</option>
<option value="1990">1990</option>
<option value="1991">1991</option>
<option value="1992">1992</option>
<option value="1993">1993</option>
<option value="1994">1994</option>
<option value="1995">1995</option>
<option value="1996">1996</option>
<option value="1997">1997</option>
<option value="1998">1998</option>
<option value="1999">1999</option>
<option value="2000">2000</option>
<option value="2001">2001</option>
<option value="2002">2002</option>
<option value="2003">2003</option>
<option value="2004">2004</option>
<option value="2005">2005</option>
<option value="2006">2006</option>
<option value="2007">2007</option>
</select>
</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01"> Secci&oacute;n :
<td width="508" height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<?php
mysql_select_db($database_conexion, $conexion);
$query_rsFunc2 = "SELECT * from secciones ";
$rsFunc2 = mysql_query($query_rsFunc2, $conexion) or die(mysql_error());
$row_rsFunc2 = mysql_fetch_assoc($rsFunc2);
?>
<select name="seccion" class="campos" id="seccion">
<option value="0">-- Nombre de la Secci&oacute;n --</option>
<?php do {?>
<option value="<?php echo $row_rsFunc2[nombre_sec]?>"><?php echo $row_rsFunc2[nombre_sec]?></option>
<?php }while ($row_rsFunc2 = mysql_fetch_assoc($rsFunc2));?>
</select>
</font>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01"><input type="hidden" name="MAX_FILE_SIZE" value="500000">
Archivo :</td>
<td height="20" align="left" class="textoper07"><a href="<?php echo $row_Recordset1[archivo_inf]?>" target="_blank"><?php echo
$row_Recordset1[archivo_inf]?></a></td>
</tr>
<tr>
<td colspan="3" align="right" class="lineainf">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td align="right" width="18">&nbsp;&nbsp;&nbsp;</td>
<td align="left" width="215" valign="bottom">&nbsp;&nbsp;&nbsp;</td>
<td width="508" height="40" align="right" valign="bottom"> <font face="Arial, Helvetica, sans-serif">
<input name="Modificar" type="SUBMIT" class="botones" id="Modificar" value="Modificar">
<input name="Eliminar" type="SUBMIT" class="botones" id="Eliminar" value="Eliminar">
</font></td>
</tr>
</tr>
</table>
</form>
</td>
<td width="30%" align="right"></td>
</tr>
<tr>
<td align="right">
<script language="JavaScript" type="text/JavaScript">
document.form1.periodico.value="<?php echo $row_Recordset1[periodico_inf]?>";
document.form1.tipo.value="<?php echo $row_Recordset1[tipo_inf]?>";
document.form1.seccion.value="<?php echo $row_Recordset1[seccion_inf]?>";
document.form1.dia.value="<?php echo substr($row_Recordset1[fecha_inf],8,2)?>";

```

detalle_inf.php

```
document.form1.mes.value="<?php echo substr($row_Recordset1['fecha_inf'],5,2)?>";  
document.form1.ano.value="<?php echo substr($row_Recordset1['fecha_inf'],0,4)?>";
```

```
</script></td>
```

```
<td>&nbsp;</td>
```

```
</tr>
```

```
</table>
```

```
</TD>
```

```
</TR>
```

```
<TR>
```

```
<TD ALIGN="RIGHT">&nbsp;</TD>
```

```
</TR>
```

```
</TABLE>
```

```
</td>
```

```
</tr>
```

```
<td valign="bottom"><?php include("is2foot.inc") ?></td>
```

```
</tr>
```

```
</table>
```

```
</BODY>
```

```
</HTML>
```

```

body {
    background-color: #FFFFFF;
}

A:link {
text-decoration: none
;color: #00349A
}
A:hover {
    COLOR: #FF9900
; text-decoration: none
}
A:visited {
    COLOR: #00349A; FONT-FAMILY: Arial, Helvetica, sans-serif
; text-decoration: none
; text-transform: capitalize
}

.texto01 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #666666;
    font-weight: bold;
}
.texto02 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 14px;
    color: #666666;
}
.botones {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #FFFFFF;
    background-color: #F96500;
    border-top-width: 1px;
    border-right-width: 1px;
    border-bottom-width: 1px;
    border-left-width: 1px;
    border-top-style: solid;
    border-right-style: solid;
    border-bottom-style: solid;
    border-left-style: solid;
    border-top-color: #FFCC00;
    border-right-color: #CC6600;
    border-bottom-color: #CC6600;
    border-left-color: #FFCC00;
}
.texto03 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 14px;
    color: #013299;
}
.bordes {
    border-bottom-width: 1px;
    border-top-style: none;
    border-right-style: none;
    border-bottom-style: solid;
    border-left-style: none;
    border-bottom-color: #FFFFFF;
}
.margen {
    padding-left: 10px;
}
.margen2 {
    padding-left: 5px;
}
.texto04 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #013299;
    font-weight: bold;
}
.margen3 {
    padding-top: 5px;
    padding-left: 10px;
}
.texto05 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #00359B;
    font-weight: bold;
    text-align: right;
    padding-right: 5px;
}
.textoper01 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    font-weight: bold;
    color: #FF9900;
}
.textoper02 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    font-weight: normal;
    color: #FFFFFF;
}
.textoper03 {
    font-family: Arial, Helvetica, sans-serif;

```

```

        font-size: 14px;
        font-weight: bold;
        color: #01329B;
    }
    .textoper04 {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 9px;
        font-weight: bold;
        color: #666666;
        vertical-align: middle;
    }
    .campos {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 10px;
        font-weight: normal;
        color: #666666;
        vertical-align: middle;
    }
    .textoper05 {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        font-weight: bold;
        color: #FF6600;
        vertical-align: middle;
    }
    .lineasup {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        font-weight: bold;
        color: #FF6600;
        vertical-align: middle;
        border-top-width: 1px;
        border-right-width: 1px;
        border-bottom-width: 1px;
        border-left-width: 1px;
        border-top-style: solid;
        border-right-style: solid;
        border-bottom-style: solid;
        border-left-style: solid;
        border-top-color: #02339A;
        border-right-color: #FFFFFF;
        border-bottom-color: #FFFFFF;
        border-left-color: #FFFFFF;
    }
    .lineainf {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        font-weight: bold;
        color: #FF6600;
        vertical-align: middle;
        border-top-width: 1px;
        border-right-width: 1px;
        border-bottom-width: 1px;
        border-left-width: 1px;
        border-top-style: solid;
        border-right-style: solid;
        border-bottom-style: solid;
        border-left-style: solid;
        border-top-color: #FFFFFF;
        border-right-color: #FFFFFF;
        border-bottom-color: #02339A;
        border-left-color: #FFFFFF;
    }
    .textoper06 {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        font-weight: normal;
        color: #FF6600;
        vertical-align: middle;
    }
    .textoper07 {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 10px;
        font-weight: normal;
        color: #666666;
        vertical-align: middle;
    }
    .error {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 9px;
        font-weight: normal;
        color: #FF0000;
        vertical-align: middle;
    }
    A:hover {
        COLOR: #FF9900
    ; text-decoration: none
    }
    A:link {
    }
    A:hover {
        COLOR: #FF9900

```



```
; text-decoration: none
}
A:link {
text-decoration: none
;color: #00349A
}
.textoper08 {

font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 10px;
font-weight: bold;
color: #FF6600;
vertical-align: middle;
}
.textoper09 {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 11px;
font-weight: normal;
color: #000000;
text-align: justify;
}
.textoper010 {

font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 11px;
font-weight: normal;
color: #0232A1;
text-align: justify;
}
.margenex {
padding-left: 10px;
background-color: #EFEFEF;
}
}
```

```

<?php require_once('conecta.php');

mysql_select_db($database_conexion, $conexion);
if (isset($_REQUEST['ordenado'])){
if ($_REQUEST['ordenado']=="1") {

        $query_Recordset1 = "SELECT * FROM informacion Order By codigo_inf";
} elseif ($_REQUEST['ordenado']=="2") {

        $query_Recordset1 = "SELECT * FROM informacion Order By tipo_inf";
} elseif ($_REQUEST['ordenado']=="3") {

        $query_Recordset1 = "SELECT * FROM informacion Order By periodico_inf";
} elseif ($_REQUEST['ordenado']=="4") {

        $query_Recordset1 = "SELECT * FROM informacion Order By fecha_inf";
} elseif ($_REQUEST['ordenado']=="5") {

        $query_Recordset1 = "SELECT * FROM informacion Order By seccion_inf";
} elseif ($_REQUEST['ordenado']=="6") {

        $query_Recordset1 = "SELECT * FROM informacion Order By titulo_inf";
}
}

if (isset($_REQUEST['ordenado']) or $_REQUEST['ordenado']=="0") {

        $query_Recordset1 = "SELECT * FROM informacion Order By codigo_inf";
}

        // paginado

$currentPage = $_SERVER["PHP_SELF"];

if (isset($_REQUEST['maxRows_Recordset1'])){
$maxRows_Recordset1 = 10;
}
else
{
$maxRows_Recordset1 = $_REQUEST['maxRows_Recordset1'];
}

$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
$pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1, $startRow_Recordset1, $maxRows_Recordset1);
$recordset1 = mysql_query($query_limit_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
$totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
$totalRows_Recordset1 = mysql_num_rows($recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;
$numeroPaginas= ceil($totalRows_Recordset1/$maxRows_Recordset1);

$queryString_Recordset1 = "";
if (empty($_SERVER['QUERY_STRING'])) {
$params = explode("&", $_SERVER['QUERY_STRING']);
$newParams = array();
foreach ($params as $param) {
if (strpos($param, "pageNum_Recordset1") == false &&
strpos($param, "totalRows_Recordset1") == false) {
array_push($newParams, $param);
}
}
if (count($newParams) != 0) {
$queryString_Recordset1 = "&". htmlentities(implode("&", $newParams));
}
}
$queryString_Recordset1 = sprintf("&totalRows_Recordset1=%d%s", $totalRows_Recordset1, $queryString_Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Principal</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="STYLESHEET" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</tr>
</table></td>

```

```

</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="100%" align="right"></td>
</tr>
</table>
</TD>
</TR>
<TR>
<TD ALIGN="RIGHT"><table width="100%" border="0" cellspacing="0" cellpadding="0" height="100%">
<tr class="margencuadro">
<td height="100%" align="center" valign="middle" ><table width="75%" border="0">
<tr>
<td class="textoper03">Mantenimiento </td>
<td align="right" class="textoper03"></td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">Click en el t&iacutetulo para Modificar el archivo o en Grabar Nuevo para generar
uno nuevo.</td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr>
<td colspan="3" class="texto12"></td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">(<?php echo $NumeroPaginas?> PAGINAS)</b><a name="top"></a></td>
<td width="25%" class="textoper04"><b>NOTICIAS POR PAGINA </b> </td>
<td width="20%" class="textnum" align="left">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td class="bordeinf" valign="top" height="25"><font face="Arial, Helvetica, sans-serif" size="1"> <font face="Arial, Helvetica, sans-
serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0, $queryString_Recordset1); ?>">I&lt; Primero</a>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0, $pageNum_Recordset1 - 1), $queryString_Recordset1);
?>">I&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, min($totalPages_Recordset1,
$pageNum_Recordset1 + 1), $queryString_Recordset1); ?>"> Siguiente &gt;&gt;</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;&gt;</a></font></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 40 </a></td>
<td class="bordeinf" valign="top">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr valign="bottom" align="center" >
<td height="15" colspan="3" class="lineasup">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr valign="bottom" align="center" >
<td height="15" colspan="3"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr class="textoper01">
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=1">Codigo</a> </td>
<td width="15%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=2">Tipo</a> </td>
<td width="25%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=6">Titulo</a> </td>
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=3">Periodico</a> </td>
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=4">Fecha</a> </td>
<td width="20%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=5">Secci&oacute;n</a> </td>
</tr>
</table></td>
</tr>
</table>
<br>
<div align="center"></div>
<table border="0" cellpadding="0" width="75%">
<tr>
<td align="center">
<tr align="center">
<td align="top" class="textoper07" width="10%"><a href="javascript:ventana('<?php echo
$row_Recordset1[codigo_inf]?>')"><?php echo $row_Recordset1[codigo_inf]?> </a></td>
<td align="top" class="textoper07" width="15%"><?php echo $row_Recordset1[tipo_inf]?></td>
<td align="top" class="textoper07" width="25%"><?php echo $row_Recordset1[titulo_inf]?></td>
<td align="top" class="textoper07" width="10%"><?php echo $row_Recordset1[periodico_inf]?></td>
<td align="top" class="textoper07" width="10%"><?php echo $row_Recordset1[fecha_inf]?> </td>
<td align="top" class="textoper07" width="20%"><?php echo $row_Recordset1[seccion_inf]?></td>
</tr>

```

```

<?php }while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr valign="top">
<td colspan="3" class="lineainf" height="20">&nbsp;</td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">( <?php echo $NumeroPaginas?> PAGINAS)</b></td>
<td width="25%" class="textoper04"><b>NOTICIAS POR PAGINA </b></td>
<td width="20%" class="textnum" align="left">&nbsp;</td>
</tr>
<tr class="bordesup">
<td><font face="Arial, Helvetica, sans-serif" size="1"> <span class="bordeinf"><font face="Arial, Helvetica, sans-serif"
size="1"><font face="Arial, Helvetica, sans-serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0,
$queryString_Recordset1); ?>">&lt; Primer</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0,
$pageNum_Recordset1 - 1), $queryString_Recordset1); ?>">&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, min($totalPages_Recordset1, $pageNum_Recordset1 + 1), $queryString_Recordset1)?>"> Siguiente &gt;&gt;</a> <a
href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;>
</a></font></span></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 40 </a></td>
<td class="tablas0">&nbsp;</td>
</tr>
</table>
<script language="JavaScript" type="text/JavaScript">
var codigo_inf;
function ventana(codigo_inf){
window.open('detalle_inf.php?codigo_inf=' + codigo_inf +
",Servicio",width=800,height=500,screenX=200,screenY=200,scrollbars=no,toolbar=no,menubar=no,statusbar=yes,resizable=no,locationbar=
no,hotkeys=no");
}
</script></td>
</tr>
</table></TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>
<map name="Map">
<area shape="rect" coords="10,4,115,32" href="listado_per.php?codigo_per=0&log=<?php echo $_REQUEST['log]?>">
<area shape="rect" coords="126,5,233,32" href="listado_sec.php?codigo_sec=0&log=<?php echo $_REQUEST['log]?>">
<area shape="rect" coords="242,5,348,32" href="admin.php?log=<?php echo $_REQUEST['log]?>">
</map>
</BODY>
</HTML>

```

```

<?php require_once('conecta.php');

mysql_select_db($database_conexion, $conexion);
if (isset($_REQUEST['ordenado'])) {
    if ($_REQUEST['ordenado']=="1") {

        $query_Recordset1 = "SELECT * FROM periodicos Order By codigo_per";

    } elseif ($_REQUEST['ordenado']=="3") {

        $query_Recordset1 = "SELECT * FROM periodicos Order By nombre_per";

    }
}
if (!isset($_REQUEST['ordenado']) or $_REQUEST['ordenado']=="0") {

    $query_Recordset1 = "SELECT * FROM periodicos Order By codigo_per";

}

// paginado

$currentPage = $_SERVER["PHP_SELF"];

if (!isset($_REQUEST['maxRows_Recordset1'])) {
    $maxRows_Recordset1 = 10;
}
else
{
    $maxRows_Recordset1 = $_REQUEST['maxRows_Recordset1'];
}

$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1, $startRow_Recordset1, $maxRows_Recordset1);
$query_Recordset1 = mysql_query($query_limit_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($query_Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
    $all_Recordset1 = mysql_query($query_Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;
$numeroPaginas = ceil($totalRows_Recordset1/$maxRows_Recordset1);

$queryString_Recordset1 = "";
if (empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_Recordset1") == false &&
            strpos($param, "totalRows_Recordset1") == false) {
            array_push($newParams, $param);
        }
    }
    if (count($newParams) != 0) {
        $queryString_Recordset1 = "&" . htmlentities(implode("&", $newParams));
    }
}
$queryString_Recordset1 = sprintf("&totalRows_Recordset1=%d%s", $totalRows_Recordset1, $queryString_Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Periodicos</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="stylesheet" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="100%" align="right"></td>
</tr>
</table>
</TD>
</TR>

```

```

<TR>
<TD ALIGN="RIGHT"><table width="100%" border="0" cellspacing="0" cellpadding="0" height="100%">
<tr class="margencuadro">
<td height="100%" align="center" valign="middle" ><table width="75%" border="0">
<tr>
<td class="textoper03">Mantenimiento de Periodicos </td>
<td align="right" class="textoper03"><a href="javascript:ventana(0)"></a></td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">Click en el t&iacut;e; para Modificar el archivo </td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr>
<td colspan="3" class="texto12"></td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">(<?php echo $NumeroPaginas?> PAGINAS)</b><a name="top"></a></td>
<td width="25%" class="textoper04"><b>PERIODICOS POR PAGINA </b> </td>
<td width="20%" class="textnum" align="left">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr >
<td class="bordeinf" valign="top" height="25"><font face="Arial, Helvetica, sans-serif" size="1"> <font face="Arial, Helvetica, sans-
serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0, $queryString_Recordset1); ?>">Primero</a>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0, $pageNum_Recordset1 - 1), $queryString_Recordset1);
?>">&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, min($totalPages_Recordset1,
$pageNum_Recordset1 + 1), $queryString_Recordset1)?>"> Siguiente &gt;&gt;</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;&lt; </a></font></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>">10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 40 </a></td>
<td class="bordeinf" valign="top">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr valign="bottom" align="center" >
<td height="15" colspan="3" class="lineasup">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr align="bottom" align="center" >
<td height="15" colspan="3"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr class="textoper01">
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>">&ordenado=1">Codigo</a> </td>
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>">&ordenado=3">Nombre Corto </a>  </td>
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>">&ordenado=3">Nombre Largo </a>  </td>
</tr>
</table></td>
</tr>
</table>
<br>
<div align="center"></div>
<table border="0" cellpadding="0" width="75%">
<?php do{ ?>
<tr align="center">
<td align="top" class="textoper07" width="10%"><a href="javascript:ventana('<?php echo
$row_Recordset1[codigo_per]?>')"><?php echo $row_Recordset1[codigo_per]?> </a></td>
<td width="10%" valign="top" class="textoper07"><?php echo $row_Recordset1[nombre_per]
?></td>
<td align="top" class="textoper07" width="10%"><?php echo $row_Recordset1[nombre_largo_per]
?></td>
</tr>
<?php }while ($row_Recordset1 = mysql_fetch_assoc($Recordset)); ?>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr valign="top">
<td colspan="3" class="lineainf" height="20">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">(<?php echo $NumeroPaginas?> PAGINAS)</b></td>
<td width="25%" class="textoper04"><b>PERIODICOS POR PAGINA </b> </td>
<td width="20%" class="textnum" align="left">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr class="bordesup">
<td><font face="Arial, Helvetica, sans-serif" size="1"> <span class="bordeinf"><font face="Arial, Helvetica, sans-serif"
size="1"><font face="Arial, Helvetica, sans-serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0,
$queryString_Recordset1); ?>">Primero</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0,
$pageNum_Recordset1 - 1), $queryString_Recordset1); ?>">&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, min($totalPages_Recordset1, $pageNum_Recordset1 + 1), $queryString_Recordset1)?>"> Siguiente &gt;&gt;</a> <a
href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;&lt; </a></font></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>">10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>">&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST[ordenado])){echo
$_REQUEST[ordenado];}else{echo "0";}?>"> 40 </a></td>

```

```
<td class="tablas0">&nbsp;</td>
</tr>
</table>
<script language="JavaScript" type="text/JavaScript">
    var codigo_per;
    function ventana(codigo_per){
    window.open('periodicos.php?codigo_per=' + codigo_per +
"; Servicio", "width=800,height=500,screenX=200,screenY=200,scrollbars=no,toolbar=no,menubar=no,statusbar=yes,resizable=no,locationbar=no,hotkeys=no");
    }
</script></td>
</tr>
</table></TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>
</BODY>
</HTML>
```

```

<?php require_once('conecta.php');

mysql_select_db($database_conexion, $conexion);
if (isset($_REQUEST['ordenado'])) {
    if ($_REQUEST['ordenado']=="1") {

        $query_Recordset1 = "SELECT * FROM secciones Order By codigo_sec";

    } elseif ($_REQUEST['ordenado']=="3") {

        $query_Recordset1 = "SELECT * FROM secciones Order By nombre_sec";

    }
}
if (!isset($_REQUEST['ordenado']) or $_REQUEST['ordenado']=="0") {

    $query_Recordset1 = "SELECT * FROM secciones Order By codigo_sec";

}

// paginado

$currentPage = $_SERVER["PHP_SELF"];

if (!isset($_REQUEST['maxRows_Recordset1'])) {
    $maxRows_Recordset1 = 10;
}
else
{
    $maxRows_Recordset1 = $_REQUEST['maxRows_Recordset1'];
}

$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1, $startRow_Recordset1, $maxRows_Recordset1);
$Recordset1 = mysql_query($query_limit_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
    $all_Recordset1 = mysql_query($query_Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;
$numeroPaginas= ceil($totalRows_Recordset1/$maxRows_Recordset1);

$queryString_Recordset1 = "";
if (empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_Recordset1") == false &&
            strpos($param, "totalRows_Recordset1") == false) {
            array_push($newParams, $param);
        }
    }
    if (count($newParams) != 0) {
        $queryString_Recordset1 = "&". htmlentities(implode("&", $newParams));
    }
}
$queryString_Recordset1 = sprintf("&totalRows_Recordset1=%d%s", $totalRows_Recordset1, $queryString_Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Secciones</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="stylesheet" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="100%" align="right"></td>
</tr>
</table>
</TD>
</TR>

```



```

<TR>
<TD ALIGN="RIGHT"><table width="100%" border="0" cellspacing="0" cellpadding="0" height="100%">
<tr class="margencuadro">
<td height="100%" align="center" valign="middle" ><table width="75%" border="0">
<tr>
<td class="textoper03">Mantenimiento de Secciones </td>
<td align="right" class="textoper03"><a href="javascript:ventana(0)"></a></td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">Click en el t&iacute;tulo para Modificar el archivo </td>
</tr>
<tr>
<td colspan="2" align="right" class="textoper07">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr>
<td colspan="3" class="texto12"></td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">( <?php echo $NumeroPaginas?> PAGINAS)</b><a name="top"></a></td>
<td width="25%" class="textoper04"><b>SECCIONES POR PAGINA </b></td>
<td width="20%" class="textnum" align="left">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr >
<td class="bordeinf" valign="top" height="25"><font face="Arial, Helvetica, sans-serif" size="1"> <font face="Arial, Helvetica, sans-
serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0, $queryString_Recordset1); ?>">Primero</a>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0, $pageNum_Recordset1 - 1), $queryString_Recordset1);
?>">&lt;&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, min($totalPages_Recordset1,
$pageNum_Recordset1 + 1), $queryString_Recordset1)?>"> Siguiente &gt;&gt;</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;&gt;</a></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 40 </a></td>
<td class="bordeinf" valign="top">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr valign="bottom" align="center" >
<td height="15" colspan="3" class="lineasup">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr valign="bottom" align="center" >
<td height="15" colspan="3"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr class="textoper01">
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=1">Codigo</a> </td>
<td width="10%" align="center" valign="top" class="tablas1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, $pageNum_Recordset1, $queryString_Recordset1)?>&ordenado=3">Nombre Secci&oacute;n</a>  </td>
</tr>
</table></td>
</tr>
</table>
<br>
<div align="center"></div>
<table border="0" cellpadding="0" width="75%">
<?php do{ ?>
<tr align="center">
<td align="top" class="textoper07" width="10%"><a href="javascript:ventana('<?php echo
$row_Recordset1[codigo_sec]?>')"><?php echo $row_Recordset1[codigo_sec]?> </a></td>
<td width="10%" align="top" class="textoper07"><?php echo $row_Recordset1[nombre_sec]
?></td>
</tr>
<?php while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
<table width="75%" border="0" cellspacing="0" cellpadding="0" height="25">
<tr valign="top">
<td colspan="3" class="lineainf" height="20">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td width="55%" class="textoper04"><b class="textnum"><?php echo $totalRows_Recordset1?> </b> <b
class="tablas0">ARTICULOS</b> <b class="textpag">( <?php echo $NumeroPaginas?> PAGINAS)</b></td>
<td width="25%" class="textoper04"><b>SECCIONES POR PAGINA </b></td>
<td width="20%" class="textnum" align="left">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr class="bordesup">
<td><font face="Arial, Helvetica, sans-serif" size="1"> <span class="bordeinf"><font face="Arial, Helvetica, sans-serif"
size="1"><font face="Arial, Helvetica, sans-serif" size="1"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0,
$queryString_Recordset1); ?>">Primero</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, max(0,
$pageNum_Recordset1 - 1), $queryString_Recordset1); ?>">&lt;&lt; Anterior</a> <a href="<?php printf("%s?pageNum_Recordset1=%d%s",
$currentPage, min($totalPages_Recordset1, $pageNum_Recordset1 + 1), $queryString_Recordset1)?>"> Siguiente &gt;&gt;</a> <a
href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, $totalPages_Recordset1, $queryString_Recordset1); ?>">Ultimo &gt;&gt;
</a></font></td>
<td class="textoper07" valign="top"><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=10&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 10 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=20&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 20 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=30&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 30 </a><a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$pageNum_Recordset1, $queryString_Recordset1); ?>&maxRows_Recordset1=40&ordenado=<?php if (isset($_REQUEST['ordenado']))(echo
$_REQUEST['ordenado']);else(echo "0");?>"> 40 </a></td>
<td class="tablas0">&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
<script language="JavaScript" type="text/JavaScript">
var codigo_sec;

```

Listado_sec.php

```
function ventana(codigo_sec){
    window.open('secciones.php?codigo_sec=' + codigo_sec +
    "Servicio", "width=800,height=500,screenX=200,screenY=200,scrollbars=no,toolbar=no,menubar=no,statusbar=yes,resizable=no,locationbar=
    no,hotkeys=no");
}
</script></td>
</tr>
</table></TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>
</BODY>
</HTML>
```

```

<?php require_once('conecta.php');

$msg="";
$secure=abs(rand(1000000000000,1000000000000000));

if (isset($_REQUEST['log'])) {
    $user=$_REQUEST['user'];
    $pass=$_REQUEST['password'];
    $session['user']=$user;

    if ($user=="administrador") {
        mysql_select_db($database_conexion, $conexion);
        $query_rsFunc = "SELECT * FROM administrador Where user_admin= " . $user . " and passw_admin
= " . $passw . """;
        $rsFunc = mysql_query($query_rsFunc, $conexion) or die(mysql_error());
        $row_rsFunc = mysql_fetch_assoc($rsFunc);

        do {
            $getUser=$row_rsFunc['codigo_admin'];
        }while ($row_rsFunc = mysql_fetch_assoc($rsFunc));

        if ($getUser == "") {
            $msg= "El nombre de usuario o clave son incorrectos. Vuelva a intentar";
        } else {
            header("Location: http://localhost/biblioteca/listado.php?log=". $secure);
        }
    } else {
        mysql_select_db($database_conexion, $conexion);
        $query_rsFunc = "SELECT * FROM administrador Where user_admin= " . $user . " and passw_admin
= " . $passw . """;
        $rsFunc = mysql_query($query_rsFunc, $conexion) or die(mysql_error());
        $row_rsFunc = mysql_fetch_assoc($rsFunc);

        do {
            $getUser=$row_rsFunc['codigo_admin'];
        }while ($row_rsFunc = mysql_fetch_assoc($rsFunc));

        if ($getUser ==0) {
            $msg= "El nombre de usuario o clave son incorrectos. Vuelva a intentar.";
        } else {
            header("Location: http://localhost/biblioteca/admin.php?log=" . $secure );
        }
    }
}
?>

<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Principal</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="STYLESHEET" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="50%" align="right" valign="middle"><form name="form1" method="post" action="" >
<table width="90%" border="0" cellspacing="0" cellpadding="0" align="center">
<tr valign="top">
<td height="30" colspan="5" class="textoper03"><font face="Arial, Helvetica, sans-serif"><b></b></font><font face="Arial,
Helvetica, sans-serif"><b></b></font><font class="texto1">Ingreso al Mantenimiento</font></td>
</tr>
<tr>
<td colspan="5" class="lineasup">&nbsp;</td>
</tr>
<tr>
<td width="16%" colspan="3" class="textoper01">Nombre de Usuario : </td>
<td colspan="2" valign="middle"><input type="text" name="user" size="12" class="campos">
</td>
</tr>
<tr>
<td colspan="3" class="textoper01">Clave : </td>
<td width="14%" valign="middle"><input type="password" name="password" size="12" class="campos" onKeyPress = logon2()>
</td>
<td width="29%" align="right" valign="middle"><input type="button" name="Submit3" value="Ingresar .:" class="botones" onClick
= validar()>
</td>
</tr>
</table>
</tr>
<tr>
<td width="16%">&nbsp;</td>

```

```
<td colspan="4" valign="top" class="error"><div align="center" class="texto10"> <?php echo $msg ?> </div></td>
</tr>
<tr>
<td colspan="5" class="lineainf">&nbsp;</td>
</tr>
<tr>
<td height="30" colspan="5" class="textoper07">Por favor ingrese el nombre de usuario y clave para el mantenimiento.</td>
</tr>
</table>
<input type="hidden" name="log">
</form>
<script language="JavaScript">
function foco(){
document.form1.user.focus();
}
function validar() {
if (document.form1.user.value==" " | document.form1.password.value==" ")
{return alert("Uno de los campos obligatorios est&aacute; vacio.");}
else {
document.form1.log.value = 1;
document.form1.submit();
}
}
function logon2()
{if (event.keyCode==13)
if (document.form1.user.value==" " | document.form1.password.value==" ")
{return alert("Uno de los campos obligatorios est&aacute; vacio.");}
else {
document.form1.log.value = 1;
document.form1.submit();
}}
</script></td>
<td width="50%" align="right"></td>
</tr>
<tr>
<td align="right"></td>
<td>&nbsp;</td>
</tr>
</table>
</TD>
</TR>
<TR>
<TD ALIGN="RIGHT">&nbsp;</TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>

</BODY>
</HTML>
```

```

<?php require_once('conecta.php');
mysql_select_db($database_conexion, $conexion);
$query_Recordt = "SELECT max(codigo_per) as codigo FROM periodicos";
$Recordt = mysql_query($query_Recordt, $conexion) or die(mysql_error());
$row_Recordt = mysql_fetch_assoc($Recordt);
$codigo=$row_Recordt['codigo']+1;

$query_Recordset1 = "SELECT * FROM periodicos WHERE codigo_per=" . $_REQUEST['codigo_per'] . "" ;
$Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

$msg="";
if (isset($_REQUEST['log'])){
    if (isset($_REQUEST['graba'])){
        if ($_REQUEST['graba']==1){

            $insertSQL = "INSERT INTO periodicos (nombre_per, nombre_largo_per) VALUES (" .
$_REQUEST['nombre_per'] . "," . $_REQUEST['nombre_largo_per'] . ")";
mysql_select_db($database_conexion, $conexion);
$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

            $msg="El periodico fue creado.\n";

        }

        if (isset($_REQUEST['Modificar'])){

            $insertSQL = "UPDATE periodicos
            SET nombre_per = " . $_REQUEST['nombre_per'] . " ,
            nombre_largo_per = " . $_REQUEST['nombre_largo_per'] . "
            WHERE
            codigo_per=" . $_REQUEST['codigo_per'];
mysql_select_db($database_conexion, $conexion);
$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

            $msg="El periodico fue actualizado.\n";

        }

        if (isset($_REQUEST['Eliminar'])){

            $insertSQL = "DELETE FROM periodicos
            WHERE
            codigo_per=" . $_REQUEST['codigo_per'];
mysql_select_db($database_conexion, $conexion);
$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

            $msg="El periodico fue eliminado.\n";

        }

    }

    $query_Recordset1 = "SELECT * FROM periodicos WHERE codigo_per=" . $_REQUEST['codigo_per'] . "" ;
    $Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
    $row_Recordset1 = mysql_fetch_assoc($Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - Periodicos</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="STYLESHEET" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="70%" align="right" valign="top">
<form action="periodicos.php" method="post" name="form1">
<table width="97%" align="right">
<tr align="right">
<td height="40" colspan="3" class="textoper06"><?php echo $msg?></td>
</tr>
<tr>
<td height="40" colspan="3" class="textoper03">Mantenimiento de Periodicos</td>
</tr>
<tr>
<td colspan="3" class="lineasup">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Codigo: </td>
<td height="20"><font face="Arial, Helvetica, sans-serif">
<input name="codigo_per" type="TEXT" class="campos" id="codigo_per" value="<?php echo $row_Recordset1['codigo_per']?>"
size="41" maxlength="100">

```

```

        <input name="log" type="hidden" id="log" value="1">
        <input name="graba" type="hidden" id="graba" value="0">
    </font></td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Nombre Corto :
<td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<input name="nombre_per" type="text" class="campos" id="nombre_per" value="<?php echo $row_Recordset1[nombre_per]?">"
size="41">
</font>
</td>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Nombre Largo :
<td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<input name="nombre_largo_per" type="text" class="campos" id="nombre_largo_per" value="<?php echo
$row_Recordset1[nombre_largo_per]?">" size="41">
</font>
</td>
<tr>
<td colspan="3" align="right" class="lineainf">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td align="right" width="18">&nbsp;&nbsp;&nbsp;</td>
<td align="left" width="215" valign="bottom">&nbsp;&nbsp;&nbsp;</td>
<td width="508" height="40" align="right" valign="bottom"> <font face="Arial, Helvetica, sans-serif">
<input name="Nuevo" type="button" class="botones" id="Nuevo" value="Nuevo" onClick=crear()>
<input name="Modificar" type="SUBMIT" class="botones" id="Modificar" value="Modificar">
<input name="Eliminar" type="SUBMIT" class="botones" id="Eliminar" value="Eliminar">
</font></td>
</tr>
</table>
</form>
<script language="JavaScript">
function crear(){
var obj= document.form1;
obj.codigo_per.value = <?php echo $codigo?>;

if (document.form1.Nuevo.value=="Nuevo")
{
document.form1.Nuevo.value="Grabar";
obj.codigo_per.value = <?php echo $codigo?>;
obj.nombre_per.value = "";
obj.nombre_largo_per.value = "";
obj.graba.value = "1";
}
else if (document.form1.Nuevo.value=="Grabar")
{obj.submit()}
}
</script>

</td>
<td width="30%" align="right"></td>
</tr>
<tr>
<td align="right">
<td>&nbsp;&nbsp;&nbsp;</td>
<td>&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
</TD>
</TR>
<TR>
<TD ALIGN="RIGHT">&nbsp;&nbsp;&nbsp;</TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>

</BODY>
</HTML>

```

```

<?php require_once('conecta.php');
mysql_select_db($database_conexion, $conexion);
$query_Recordt = "SELECT max(codigo_sec) as codigo FROM secciones";
$Recordt = mysql_query($query_Recordt, $conexion) or die(mysql_error());
$row_Recordt = mysql_fetch_assoc($Recordt);
$codigo=$row_Recordt['codigo']+1;

$query_Recordset1 = "SELECT * FROM secciones WHERE codigo_sec=" . $_REQUEST['codigo_sec'] . "" ;
$Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

$msg="";
if (isset($_REQUEST['log'])){
    if (isset($_REQUEST['graba'])){
        if ($_REQUEST['graba']==1){

            $insertSQL = "INSERT INTO secciones (nombre_sec) VALUES (" . $_REQUEST['nombre_sec'] . ")";
            mysql_select_db($database_conexion, $conexion);
            $Result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

            $msg="La sección fue creada.\n";

        }

    }
    if (isset($_REQUEST['Modificar'])){
        $insertSQL ="UPDATE secciones
        SET nombre_sec = " . $_REQUEST['nombre_sec'] . "
        WHERE
        codigo_sec=" . $_REQUEST['codigo_sec'];
        mysql_select_db($database_conexion, $conexion);
        $Result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

        $msg="La sección fue actualizada.\n";

    }

    if (isset($_REQUEST['Eliminar'])){
        $insertSQL ="DELETE FROM secciones
        WHERE
        codigo_sec=" . $_REQUEST['codigo_sec'];
        mysql_select_db($database_conexion, $conexion);
        $Result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

        $msg="La sección fue eliminada.\n";

    }

}

$query_Recordset1 = "SELECT * FROM secciones WHERE codigo_sec=" . $_REQUEST['codigo_sec'] . "" ;
$Recordset1 = mysql_query($query_Recordset1, $conexion) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

?>
<HTML>
<HEAD>
<TITLE>Biblioteca Virtual - secciones</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<LINK REL="STYLESHEET" HREF="is2style.css" TYPE="text/css">
</HEAD>
<BODY bgcolor="#D6AC83" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" text="#003300" vlink="#996600">
<table width="100%" height="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
<td valign="top"><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td><table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="25%"></td>
<td width="50%" valign="top"></td>
<td width="25%"></td>
</tr>
</table></td>
</tr>
<tr>
<td></td>
</tr>
</table></td>
</tr>
<tr>
<td valign="top"><TABLE border="0" cellspacing="0" cellpadding="0" width="100%">
<TR>
<TD width="100%" valign="top" nowrap>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="70%" align="right" valign="top">
<form action="secciones.php" method="post" name="form1">
<table width="97%" align="right">
<tr align="right">
<td height="40" colspan="3" class="textoper06"><?php echo $msg?></td>
</tr>
<tr>
<td height="40" colspan="3" class="textoper03">Mantenimiento de secciones</td>
</tr>
<tr>
<td colspan="3" class="lineasup">&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Codigo: </td>
<td height="20"><font face="Arial, Helvetica, sans-serif">
<input name="codigo_sec" type="TEXT" class="campos" id="codigo_sec" value="<?php echo $row_Recordset1['codigo_sec']?>"
size="41" maxlength="100">
<input name="log" type="hidden" id="log" value="1">
<input name="graba" type="hidden" id="graba" value="0">

```

```

</font></td>
</tr>
<tr>
<td height="20" colspan="2" align="right" class="texto01">Nombre :
<td height="20" align="left"><font face="Arial, Helvetica, sans-serif">
<input name="nombre_sec" type="text" class="campos" id="nombre_sec" value="<?php echo $row_Recordset1[nombre_sec]?"
size="41">
</font>
</tr>
<tr>
<td colspan="3" align="right" class="lineainf">&nbsp;  </td>
</tr>
<tr>
<td align="right" width="18">&nbsp;  </td>
<td align="left" width="215" valign="bottom">&nbsp;  </td>
<td width="508" height="40" align="right" valign="bottom"> <font face="Arial, Helvetica, sans-serif">
<input name="Nuevo" type="button" class="botones" id="Nuevo" value="Nuevo" onClick=crear(>
<input name="Modificar" type="SUBMIT" class="botones" id="Modificar" value="Modificar">
<input name="Eliminar" type="SUBMIT" class="botones" id="Eliminar" value="Eliminar">
</font></td>
</tr>
</table>
</form>
<script language="JavaScript">
function crear(){
var obj= document.form1;
obj.codigo_sec.value = <?php echo $codigo?>;

if (document.form1.Nuevo.value=="Nuevo")
{
document.form1.Nuevo.value="Grabar";
obj.codigo_sec.value = <?php echo $codigo?>;
obj.nombre_sec.value = "";
obj.graba.value = "1";
}
else if (document.form1.Nuevo.value=="Grabar")
{obj.submit()}
}
</script>

</td>
<td width="30%" align="right"></td>
</tr>
<tr>
<td align="right">
</td>
<td>&nbsp;  </td>
</tr>
</table>
</TD>
</TR>
<TR>
<TD ALIGN="RIGHT">&nbsp;  </TD>
</TR>
</TABLE>
</td>
</tr>
<tr>
<td valign="bottom"><?php include("is2foot.inc") ?></td>
</tr>
</table>
</BODY>
</HTML>

```


DISEÑO DE TESIS