



# **UNIVERSIDAD DEL AZUAY**

**Facultad de Ciencias de la Administración**

**Escuela de Ingeniería de Sistemas**

## **“SITIO WEB DINAMICO PARA LA PARROQUIA EL VALLE CON EL USO DE PHP-NUKE”**

**Monografía previa a la obtención del  
Título de Ingeniero de Sistemas**

**AUTOR: PEDRO VALLEJO GUERRERO**

**DIRECTOR: ING. MARCOS ORELLANA CORDERO**

**Cuenca, Ecuador**

**2006**

## **RESPONSABILIDAD DE AUTOR**

Las ideas y opiniones vertidas en la presente monografía  
son de exclusiva responsabilidad de su autor.

---

Pedro Vallejo G.

## **DEDICATORIA**

A Vicente y Targelia por su apoyo incondicional en esta etapa de mi vida, sin ellos hubiese sido imposible llegar a la cúspide de mi carrera.

Con profundo cariño a toda mi familia por todos esos detalles y por el compartir diario de nuestras existencias.

## **AGRADECIMIENTO**

A la Universidad del Azuay que me acogió en sus aulas y me formó como profesional y como persona.

Al Ingeniero Oswaldo Merchán, Director de la Escuela de Ingeniería de Sistemas.

Al Ingeniero Marcos Orellana, Director de esta Monografía, por su apoyo y sus acertados consejos.

A todos mis compañeros y amigos a lo largo de la Vida Universitaria.

## Índice de Contenidos

<b>RESUMEN</b> .....	<b>IX</b>
<b>ABSTRACT</b> .....	<b>X</b>
<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>INTERNET</b> .....	<b>2</b>
1.1. DEFINICIÓN .....	2
1.2. HISTORIA DEL INTERNET .....	2
1.3. ARPANET.....	4
1.4. CONCLUSIÓN .....	5
<b>PHP</b> .....	<b>6</b>
2.1. INTRODUCCIÓN.....	6
2.2. HISTORIA.....	6
2.3. QUÉ SE PUEDE HACER CON PHP .....	7
2.4. COMPATIBILIDAD CON SISTEMAS OPERATIVOS, SERVIDORES Y BASES DE DATOS .....	8
2.5. VENTAJAS DE USAR PHP .....	9
2.6. SINTAXIS BÁSICA .....	10
2.7. SEPARACIÓN DE INSTRUCCIONES.....	11
2.8. COMENTARIOS.....	11
2.9. TYPES.....	11
2.9.1. Arrays .....	12
2.9.1.1. Arrays Unidimensionales.....	12
2.9.1.2. Arrays Multidimensionales .....	13
2.9.2. Números en punto flotante .....	13
2.9.3. Enteros.....	13
2.9.4. Objetos.....	14
2.9.4.1. Inicialización de Objetos.....	14
2.9.5. Cadenas .....	14
2.10. Variables.....	15
2.10.1. Variables predefinidas.....	15
2.10.2. Variables de entorno.....	16
2.10.4. Ámbito de las variables.....	16
2.11. CONSTANTES .....	16
2.11.1. Sintaxis.....	17
2.11.2. Definiendo constantes.....	17
2.12. EXPRESIONES .....	17
2.13. OPERADORES.....	18
2.13.1. OPERADORES ARITMÉTICOS .....	18
2.14. ESTRUCTURAS DE CONTROL.....	19
2.14.1. <i>if</i> .....	19
2.14.2. <i>else</i> .....	19
2.14.3. <i>elseif</i> .....	19
2.14.4. <i>while</i> .....	20
2.14.5. <i>do..while</i> .....	20
2.14.6. <i>for</i> .....	20
2.14.7. <i>break</i> .....	20
2.14.8. <i>continue</i> .....	21
2.14.9. <i>switch</i> .....	21
2.14.10. Sintaxis Alternativa de Estructuras de Control .....	21
2.15. CONCLUSIÓN .....	21
<b>PHP-NUKE</b> .....	<b>22</b>
3.1. INTRODUCCIÓN.....	22
3.2. HISTORIA.....	22
3.3. LA LICENCIA DE PHP-NUKE.....	22

3.4. EVOLUCIÓN DE PHP-NUKE .....	22
3.5. REQUISITOS PARA EL USO DE PHP-NUKE .....	23
3.6. INSTALACIÓN .....	24
3.6.1. <i>Instalación en Linux</i> .....	24
3.6.1.1. Creación de la base de datos .....	24
3.6.2. <i>Instalación en Windows</i> .....	25
3.6.3. <i>Instalación de PHP-Nuke</i> .....	25
3.6.4. <i>La carpeta html</i> .....	25
3.6.5. <i>El fichero nuke.sql</i> .....	26
3.6.6. <i>El fichero config.php</i> .....	27
3.7. ADMINISTRACIÓN.....	28
3.7.1. <i>Nueva Noticia</i> .....	28
3.7.2. <i>Respaldo</i> .....	28
3.7.3. <i>Bloques</i> .....	28
3.7.4. <i>Contenido</i> .....	29
3.7.5. <i>Descargas</i> .....	29
3.7.6. <i>Editar Administradores</i> .....	29
3.7.7. <i>Enciclopedia</i> .....	30
3.7.8. <i>Efemérides</i> .....	30
3.7.9. <i>Foros</i> .....	30
3.7.10. <i>Referencias HTTP</i> .....	30
3.8. CONCLUSIÓN .....	31
<b>SERVIDOR APACHE.....</b>	<b>32</b>
4.1. INTRODUCCIÓN.....	32
4.2. HISTORIA.....	32
4.3. LICENCIA.....	32
4.4. CARACTERÍSTICAS.....	33
4.5. CONCLUSIÓN .....	34
<b>MY-SQL.....</b>	<b>35</b>
5.1. INTRODUCCIÓN A LAS BASES DE DATOS.....	35
5.2. HISTORIA.....	35
5.3. CARACTERÍSTICAS DE MYSQL .....	36
5.4. EL LENGUAJE SQL .....	37
5.4.1. <i>Cláusula SELECT</i> .....	37
5.4.2. <i>Funciones de agrupamiento</i> .....	38
5.4.3. FUNCIONES DE AGRUPAMIENTO PERMITIDAS.....	38
5.4.4. <i>Cláusula FROM</i> .....	39
5.4.5. <i>Cláusula WHERE</i> .....	39
5.4.6. <i>Cláusula GROUP BY</i> .....	39
5.4.7. <i>Cláusula HAVING</i> .....	40
5.4.8. <i>Operador UNION</i> .....	40
5.4.9. <i>Cláusula ORDER BY</i> .....	41
5.4.10. <i>Enlace de varias Tablas o ficheros</i> .....	41
5.4.11. <i>Alias de las Tablas</i> .....	41
5.4.12. <i>Orden de las tablas en la cláusula FROM</i> .....	42
5.4.13. <i>Enlace de más de dos tablas</i> .....	42
5.4.14. <i>SELECT anidadas</i> .....	42
5.5. EXPRESIONES SQL .....	43
5.6. FUNCIONES.....	43
5.7. SENTENCIAS PARA CREAR O DESTRUIR TABLAS .....	44
5.7.1. <i>Creación de una tabla</i> .....	44
5.7.2. <i>Destrucción de una tabla</i> .....	45
5.8. SENTENCIA INSERT .....	45
5.9. SENTENCIA UPDATE .....	46
5.10. SENTENCIA DELETE .....	46
5.11. TRANSACCIONES .....	47
5.12. ¿MYSQL ES UNA ALTERNATIVA ACEPTABLE?.....	47
5.13. CONCLUSIÓN .....	48

<b>INTEGRACION DE HERRAMIENTAS .....</b>	<b>49</b>
6.1. INTRODUCCIÓN.....	49
6.2. ASPECTOS BÁSICOS DE CALIDAD DE SOFTWARE .....	49
6.2.1. FACTORES DE CALIDAD .....	50
6.2.2. <i>Calidad de software</i> .....	51
6.2.3. <i>Elección de Tecnologías</i> .....	51
6.3. BENEFICIOS .....	51
6.4. DESVENTAJAS .....	52
6.5. DESIGNACIÓN DE HERRAMIENTAS SOFTWARE .....	52
6.6. CONCLUSIÓN .....	54
<b>PRESENTACION WEB.....</b>	<b>55</b>
7.1. INTRODUCCIÓN.....	55
7.2. USOS COMUNES DE LAS APLICACIONES WEB .....	55
7.3. PÁGINAS WEB ESTÁTICAS Y DINÁMICAS .....	56
7.3.1. <i>Procesamiento de páginas Web estáticas</i> .....	56
7.3.1. <i>Procesamiento de páginas Web dinámicas</i> .....	56
7.4. ERRORES EN SITIOS WEB.....	57
7.5. PRESENTACIÓN DEL SITIO .....	58
7.5.1. <i>Información del Sitio Turístico</i> .....	58
7.5.2. <i>Importancia del Sitio Turístico</i> .....	59
7.5.3. <i>Quienes pueden consultar el Sitio Web</i> .....	59
7.5.4. <i>Contenido y Administración</i> .....	60
7.6. CONCLUSIÓN .....	60
<b>CONCLUSIÓN GENERAL .....</b>	<b>61</b>
<b>REFERENCIAS .....</b>	<b>62</b>
<b>BIBLIOGRAFÍA.....</b>	<b>63</b>
<b>GLOSARIO .....</b>	<b>64</b>

## Índice de ilustraciones y cuadros

Imágen1.1 Trayectoria de Paquetes en la Red .....	3
Cuadro 2.1 Compatibilidad de PHP con Bases de Datos.....	9
Cuadro 2.2. Código Fuente PHP.....	9
Cuadro 2.3. Etiquetas soportas para PHP.....	10
Cuadro 2.4. Etiquetas soportas para PHP.....	11
Cuadro 2.5. Comentarios en PHP .....	11
Cuadro 2.6. Arrays Unidimensionales .....	12
Cuadro 2.7. Arrays Multidimensionales .....	13
Cuadro 2.8. Números en punto flotante .....	13
Cuadro 2.9. Enteros.....	13
Cuadro 2.10. Inicialización de Objetos .....	14
Cuadro 2.11. Caracteres protegidos .....	15
Cuadro 2.12. Variables.....	15
Cuadro 2.13. Definiendo constantes .....	17
Cuadro 2.14. Funciones .....	18
Cuadro 2.15. Operadores Aritméticos.....	18
Imagen 3.1. PHP-NUKE.....	23
Programación Web Masters .....	23
Imagen 3.2. El Menú de Administración .....	28
Cuadro 5.1. MySQL.....	36
Cuadro 6.1. Integración de Herramientas .....	53

## Índice de Anexos

Modelo Entidad – Relación.....	78
Diccionario de datos.....	79
Modelo de flujo de información.....	82
Pantallas del Sitio Turístico.....	83



## RESUMEN

Como requisito para obtener el título de Ingeniero de Sistemas, se hace necesario la realización de una monografía, se ha escogido el tema de un Sitio Web Dinámico con herramientas de software de desarrollo modernas, potentes, y de libre acceso.

Se ha hecho un estudio preliminar del software que se usa, de cada una de las herramientas que permiten implementar el producto desarrollado, esto como parte teórica y seguido se desarrolló la aplicación práctica. Debo anotar que el uso de Internet ha sido un pilar fundamental en los diferentes temas que se ha ido tratando.

En la recopilación de información para el portal Web se realizo visitas de campo en la cual se tomo fotografías de los diferentes sectores de la Parroquia, a su vez se hizo algunas visitas a personas que han estudiado al Valle desde diferentes perspectivas y como referencias bibliográficas se investigo en la Biblioteca local, misma que funciona desde hace 10 años.

El software desarrollado es fácil de manejar; se pretendió en su diseño e implementación que sea amigable al usuario y sencillo de administrar. Solo el Administrador puede hacer los mantenimientos y cambios que el Portal requiera, los demás usuarios pueden hacer las visitas y consultas que deseen siempre y cuando el sitio este disponible.

La monografía aquí expuesta trata en primer lugar la parte teórica, seguido están impresas las pantallas de la aplicación práctica y se acompaña de un CD con todo el material necesario.

## **Abstract**

As requirement to obtain Engineer's of Systems title, it becomes necessary the realization of a monograph, the topic of a Place Dynamic Web has been chosen with modern, potent tools of development software, and of free access.

A preliminary study of the software has been made that is used, of each one of the tools that allow to implement the developed product, this like theoretical and followed part was developed the practical application. I should write down that the use of Internet has been a fundamental pillar in the different topics that he/she has left trying.

In the summary of information for the portal Web one carries out field visits in turn in which take pictures of the different sectors of the Parish, it was made some visits to people that have studied to the Valley from different perspectives and like bibliographical references one investigates in the local, same Library that works for 10 years.

The developed software is easy to manage; it was sought in its design and implementation that it is friendly to the user and simple of administering. Alone the Administrator can make the maintenances and changes that the Portal requires, the other users can the visits and consultations that you/they want ago provided the place this available one.

The monograph here exposed it treats the theoretical, followed part in the first place they are printed the screens of the practical application and it is accompanied of a CD with the whole necessary material.

## **Introducción**

El uso de Tecnología Informática en todos los ámbitos del desarrollo humano y su constante y acelerada expansión de manera específica el Internet y las herramientas informáticas en relación a esta permiten trabajar y colocar información en el cyber espacio de temas que en ocasiones poco o nada se menciona y muchas veces se tiene información esparcida sobre ella.

El objetivo es implementar un Sitio Web con administración dinámica de información de la Parroquia “El Valle”, tomando en consideración los temas de mayor relevancia para colocar en dicho sitio, la razón fundamental para optar por este tema es el hecho de que vivo en esta Parroquia y creo que es factible el uso de tecnología informática para darla a conocer y fomentar su conocimiento a nivel local y nacional.

Se realiza investigaciones en Internet y en libros sobre las diferentes herramientas de software que permiten el desarrollo del portal Web, su evolución, tendencia, campos de aplicación y en sí sobre el diseño y concepción visual que van dando forma al portal en su conjunto a través de las páginas que se van creando. Se realiza un cuerpo de trabajo teórico analizando el Internet, el Lenguaje PHP, el desarrollador PHP Nuke, el Servidor Apache, la Base de Datos MY-SQL y a continuación se proceda a explicar la aplicación práctica con el uso de pantallas gráficas del Sitio.

# CAPÍTULO 1

## INTERNET

### 1.1. Definición

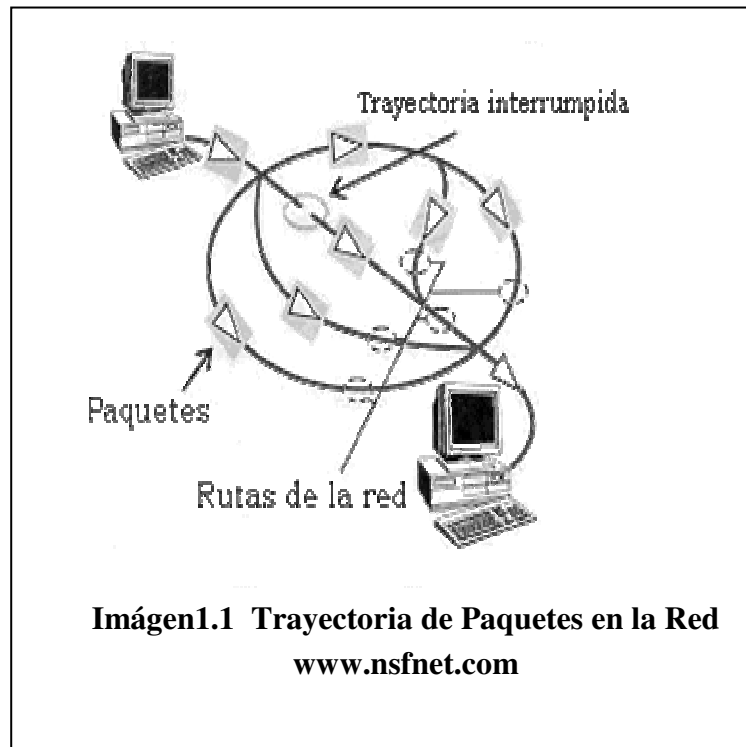
Podemos definir a Internet como una "red de redes", es decir, una red que no sólo interconecta computadoras, sino que interconecta redes de computadoras entre sí. Una red de computadoras es un conjunto de máquinas que se comunican a través de algún medio (cable coaxial, fibra óptica, radiofrecuencia, líneas telefónicas, etc.) con el objeto de compartir recursos.

De esta manera, Internet sirve de enlace entre redes más pequeñas y permite ampliar su cobertura al hacerlas parte de una "red global". Esta red global tiene la característica de que utiliza un lenguaje común que garantiza la intercomunicación de los diferentes participantes; este lenguaje común o protocolo (un protocolo es el lenguaje que utilizan las computadoras al compartir recursos) se conoce como TCP/IP.

### 1.2. Historia del Internet

En los años sesentas, cuando en los Estados Unidos se estaba buscando una forma de mantener las comunicaciones vitales del país en el posible caso de una Guerra Nuclear. Este hecho marcó profundamente su evolución, ya que aún ahora los rasgos fundamentales del proyecto se hallan presentes en lo que hoy conocemos como Internet.

“En primer lugar, el proyecto contemplaba la eliminación de cualquier "autoridad central", ya que sería el primer blanco en caso de un ataque; en este sentido, se pensó en una red descentralizada y diseñada para operar en situaciones difíciles. Cada máquina conectada debería tener el mismo status y la misma capacidad para mandar y recibir información.” (UCLA, 2000, p.30)



El envío de los datos debería descansar en un mecanismo que pudiera manejar la destrucción parcial de la Red. Se decidió entonces que los mensajes deberían de dividirse en pequeñas porciones de información o paquetes, los cuales contendrían la dirección de destino pero sin especificar una ruta específica para su arribo; por el contrario, cada paquete buscaría la manera de llegar al destinatario por las rutas disponibles y el destinatario reensamblaría los paquetes individuales para reconstruir el mensaje original. La ruta que siguieran los paquetes no era importante; lo importante era que llegaran a su destino.

Curiosamente fue en Inglaterra donde se experimentó primero con estos conceptos; y así en 1968, el Laboratorio Nacional de Física de la Gran Bretaña estableció la primera red experimental. Al año siguiente, el Pentágono de los EE.UU. decidió financiar su propio proyecto, y en 1969 se establece la primera red en la Universidad de California (UCLA) y poco después aparecen tres redes adicionales. Nació así ARPANET (Advanced Research Projects Agency NETWORK), antecedente de la actual Internet.

### 1.3. ARPANET

Gracias a ARPANET, científicos e investigadores pudieron compartir recursos informáticos en forma remota; este era una gran ayuda ya que hay que recordar que en los años 70's el tiempo de procesamiento por computadora era un recurso realmente escaso. ARPANET en sí misma también creció y ya para 1972 agrupaba a 37 redes.

La mayor parte del tráfico estaba constituido por noticias y mensajes personales, y no tanto por procesos informáticos; de hecho, cuando se desarrollaron las listas de correo electrónico (mensajes de correo que se distribuyen a un grupo de usuarios), uno de los primeros temas que abordaron con éxito fue el de la ciencia-ficción a través de una popular lista que se llamaba SF-LOVERS (Fanáticos de la ciencia-ficción).

El Protocolo utilizado en ese entonces por las máquinas conectadas a ARPANET se llamaba NCP (Network Control Protocol ó Protocolo de Control de Red), pero con el tiempo dio paso a un protocolo más sofisticado: TCP/IP, que de hecho está formado no por uno, sino por varios protocolos, siendo los más importantes el protocolo TCP (Transmission Control Protocol ó Protocolo de Control de Transmisión) y el Protocolo IP (Internet Protocol ó Protocolo de Internet). TCP convierte los mensajes en paquetes en la máquina emisora, y los reensambla en la máquina destino para obtener el mensaje original, mientras que IP es el encargado de encontrar la ruta al destino.

La naturaleza descentralizada de ARPANET y la disponibilidad sin costo de programas basados en TCP/IP permitió que ya en 1977, otro tipo de redes no necesariamente vinculadas al proyecto original, empezaran a conectarse. En 1983, el segmento militar de ARPANET decide separarse y formar su propia red que se conoció como MILNET. ARPANET, y sus "redes asociadas" empezaron a ser conocidas como Internet.

#### **1.4. Conclusión**

Se puede afirmar que el aporte de Internet al desarrollo humano y al cúmulo de conocimiento es enorme, hoy se puede tener a disposición en cuestión de segundos toda la información que se necesita, cosa que hasta hace unos años era viable solo en películas de ciencia ficción. Al igual que un sinnúmero de tecnologías como el microondas nació como un propósito militar y hoy en día esta al alcance de la comunidad civil.

La red de redes al igual que muchas ciencias debe ser asequible a todas las personas y de manera especial a los grupos humanos más desamparados, sería o es injusto hacer de Internet una exclusividad; en todo el mundo hay personas que están abogando por esta situación, Internet debe ser para unir a los pueblos y no para crear más brechas sociales.

## CAPITULO 2

### PHP

#### 2.1. Introducción

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje "Open Source" 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." (*Rafael Martínez, 1997-2002, p.75*)

#### 2.2. Historia

PHP es un lenguaje creado por una gran comunidad de personas. El sistema fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf. El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas. Dada la aceptación del primer PHP y de manera adicional, su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (Form Interpreter) y el conjunto de estas dos herramientas, sería la primera versión compacta del lenguaje: PHP/FI.

La siguiente gran contribución al lenguaje se realizó a mediados de 1997 cuando se volvió a programar el analizador sintáctico, se incluyeron nuevas funcionalidades como el soporte a nuevos protocolos de Internet y el soporte a la gran mayoría de las bases de datos comerciales. Todas estas mejoras sentaron las bases de PHP versión 3. Actualmente PHP se encuentra en su versión 4, que utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades actuales y solucionar algunos inconvenientes de la anterior versión. Algunas mejoras de esta nueva versión son su rapidez gracias a que primero se compila y luego se ejecuta, mientras que



antes se ejecutaba mientras se interpretaba el código, su mayor independencia del servidor Web.

“Actualmente el número de servidores que utilizan PHP se ha disparado, logrando situarse cerca de los 5 millones de sitios y 800.000 direcciones IP, lo que le ha convertido a PHP en una tecnología popular. Esto es debido, entre otras razones, a que PHP es el complemento ideal para que el tándem Linux-Apache sea compatible con la programación del lado del servidor de sitios Web. Gracias a la aceptación que ha logrado, y los grandes esfuerzos realizados por una creciente comunidad de colaboradores para implementarlo de la manera más óptima, se asegura que el lenguaje se convertirá en un estándar que compartirá los éxitos augurados al conjunto de sistemas desarrollados en código abierto.” (*Rafael Martínez, 1997-2002, p.115*)

### **2.3. Qué se puede hacer con PHP**

PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies.

Existen tres campos en los que scripts escritos en PHP son usados.

- Scripts en la parte del servidor. Este es el campo más tradicional y el principal campo de trabajo. Se necesitan tres cosas para que esto funcione. El parseador PHP (CGI ó módulo), un servidor Web y un navegador. Se necesita correr el servidor Web con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador, conectando con el servidor Web.
- Scripts en línea de comandos. Se puede crear un script PHP y correrlo sin ningún servidor Web ó navegador. Solamente se necesita el parseador PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en Unix ó Linux) ó el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesamiento de texto.

- Escribir aplicaciones gráficas clientes. PHP no es probablemente el mejor lenguaje para escribir aplicaciones gráficas, pero si se conoce bien se puede utilizar algunas características avanzadas en programas clientes, se utiliza PHP-GTK para escribir dichos programas. Es posible escribir aplicaciones independientes de una plataforma.

#### **2.4. Compatibilidad con Sistemas Operativos, Servidores y Bases de Datos**

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más.

PHP soporta la mayoría de servidores Web: Apache, Microsoft Internet Information Server, Personal Web Server, Netscape y iPlanet, Oreilly 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.

Con PHP se tiene la libertad de escoger el sistema operativo y el servidor de acuerdo a las necesidades y presupuesto. También provee la posibilidad de usar programación de procedimientos ó programación orientada a objetos. Aunque no todas las características estándares de la programación orientada a objetos están implementadas en la versión actual de PHP, muchas librerías y aplicaciones grandes (incluyendo la librería PEAR) están escritas íntegramente usando programación orientada a objetos.

PHP no limita los resultados a HTML. Entre las habilidades de PHP se incluyen, creación de imágenes, ficheros PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y ficheros XML. PHP puede auto generar estos ficheros y grabarlos en el sistema de ficheros en vez de presentarlos en la pantalla.

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

**Cuadro 2.1 Compatibilidad de PHP con Bases de Datos**  
**Manual de PHP Licencia General Pública GNU**

## 2.5. Ventajas de usar PHP

```
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <?php
      echo "Hola, Yo soy un PHP script!";
    ?>
  </body>
</html>
```

**Cuadro 2.2. Código Fuente PHP**  
**Manual de PHP Licencia General Pública GNU**

“No es lo mismo un script escrito en otro lenguaje de programación como Perl o C. En vez de escribir un programa con muchos comandos para crear una salida en HTML, escribimos el código HTML con cierto código PHP embebido (introducido) en el mismo, que producirá cierta salida (en nuestro ejemplo, producir un texto). El código PHP se incluye entre etiquetas especiales de comienzo y final que nos permitirán entrar y salir del modo PHP.” (*Rafael Martínez, 1997-2002, p.200*)

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. Si tuviésemos un script similar al de nuestro ejemplo en nuestro servidor, el cliente solamente recibiría el

resultado de su ejecución en el servidor, sin ninguna posibilidad de determinar que código ha producido el resultado recibido. El servidor Web puede ser incluso configurado para que procese todos los ficheros HTML con PHP.

Lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales. Aunque el desarrollo de PHP está concentrado en la programación de scripts en la parte del servidor, se puede utilizar para muchas otras cosas.

## 2.6. Sintaxis Básica

Para interpretar un archivo, PHP simplemente interpreta el texto del archivo hasta que encuentra uno de los caracteres especiales que delimitan el inicio de código PHP. El intérprete ejecuta entonces todo el código que encuentra, hasta que encuentra una etiqueta de fin de código, que le dice al intérprete que siga ignorando el código siguiente. Este mecanismo permite embeber código PHP dentro de HTML: todo lo que está fuera de las etiquetas PHP se deja tal como está, mientras que el resto se interpreta como código.

Hay cuatro conjuntos de etiquetas que pueden ser usadas para denotar bloques de código PHP. De estas cuatro, sólo 2 (`<?php. . .?>` y `<script language="php">. . .</script>`) están siempre disponibles; el resto pueden ser configuradas en el fichero de `php.ini` para ser o no aceptadas por el intérprete. Mientras que el formato corto de etiquetas (short-form tags) y el estilo ASP (ASP-style tags) pueden ser convenientes, no son portables como la versión de formato largo de etiquetas. Además, si se pretende embeber código PHP en XML o XHTML, será obligatorio el uso del formato `<?php. . .?>` para la compatibilidad con XML.

- |   |
|---|
| <ol style="list-style-type: none"><li>1. <code>&lt;?php echo("etiquetas php"); ?&gt;</code></li><li>2. <code>&lt;? echo ("etiquetas php "); ?&gt;</code></li><li>3. <code>&lt;script language="php"&gt; echo ("etiquetas php"); &lt;/script&gt;</code></li><li>4. <code>&lt;% echo ("etiquetas php "); %&gt;</code></li></ol> |
|---|

**Cuadro 2.3. Etiquetas soportadas para PHP**  
**Manual de PHP Licencia General Pública GNU**

## 2.7. Separación de instrucciones

La separación de instrucciones se hace de la misma manera que en C o Perl; terminando cada declaración con un punto y coma; la etiqueta de fin de bloque (?>) implica el fin de la declaración.

```
<?php
    echo "This is a test";
?>
<?php echo "This is a test" ?>
```

**Cuadro 2.4. Etiquetas soportadas para PHP**  
**Manual de PHP Licencia General Pública GNU**

## 2.8. Comentarios

PHP soporta el estilo de comentarios de 'C', 'C++' y de la interfaz de comandos de Unix.

```
<?php
    echo "Esta es una prueba"; // Este es un comentario al estilo C++
    /* Este es un comentario
    de multiple línea */
    echo "Esta es una prueba"; # Este es un comentario al estilo Shell de Unix
?~
```

**Cuadro 2.5. Comentarios en PHP**  
**Manual de PHP Licencia General Pública GNU**

## 2.9. Types

PHP soporta los siguientes tipos:

- array
- números en punto flotante
- entero
- objeto
- cadena

El tipo de una variable normalmente no lo indica el programador; en su lugar lo decide PHP en tiempo de ejecución dependiendo del contexto en el que se utilice esa variable. Si se quisiese obligar a que una variable se convierta a un tipo concreto, se podría forzar la variable o usar la función `settype()` para ello. Una variable se puede comportar de formas diferentes en ciertas situaciones, dependiendo de qué tipo sea en ese momento.

## 2.9.1. Arrays

Los arrays actualmente actúan tanto como tablas hash (arrays asociativos) y como arrays indexados (vectores).

### 2.9.1.1. Arrays Unidimensionales

PHP soporta tanto arrays escalares como asociativos. De hecho, no hay diferencias entre los dos. Se puede crear una array usando las funciones `list()` o `array()`, o se puede asignar el valor de cada elemento del array de manera explícita.

```
$a[0] = "abc";  
$a[1] = "def";  
$b["foo"] = 13;
```

**Cuadro 2.6. Arrays Unidimensionales**  
**Manual de PHP Licencia General Pública GNU**

También se puede crear un array simplemente añadiendo valores al array. Cuando se asigna un valor a una variable array usando corchetes vacíos, el valor se añadirá al final del array.

```
$a[] = "hola"; // $a[2] == "hola"  
$a[] = "mundo"; // $a[3] == "mundo"
```

### 2.9.1.2. Arrays Multidimensionales

Los arrays multidimensionales son bastante simples. Para cada dimensión del array, se puede añadir otro valor [clave] al final:

```
$a[1] = $f; # ejemplos de una sola dimensión
$a["foo"] = $f;
$a[1][0] = $f; # bidimensional
$a["foo"][2] = $f; # (se pueden mezclar índices numéricos y asociativos)
$a[3]["bar"] = $f; # (se pueden mezclar índices numéricos y asociativos)
$a["foo"][4]["bar"][0] = $f; # tetradimensional!
```

**Cuadro 2.7. Arrays Multidimensionales**  
**Manual de PHP Licencia General Pública GNU**

### 2.9.2. Números en punto flotante

Los números en punto flotante ("double") se pueden especificar utilizando cualquiera de las siguientes sintaxis:

```
$a = 1.234;
$a = 1.2e3;
```

**Cuadro 2.8. Números en punto flotante**  
**Manual de PHP Licencia General Pública GNU**

### 2.9.3. Enteros

Los enteros se pueden especificar usando una de las siguientes sintaxis:

```
$a = 1234; # número decimal
$a = -123; # un número negativo
$a = 0123; # número octal (equivalente al 83 decimal)
$a = 0x12; # número hexadecimal (equivalente al 18 decimal)
```

**Cuadro 2.9. Enteros**  
**Manual de PHP Licencia General Pública GNU**

## 2.9.4. Objetos

### 2.9.4.1. Inicialización de Objetos

Para inicializar un objeto se usa la sentencia `new`, de esta manera se instancia el objeto a una variable.

```
class foo {  
    function do_foo () {  
        echo "Doing foo."  
    }  
}  
  
$bar = new foo;  
$bar->do_foo();
```

**Cuadro 2.10. Inicialización de Objetos**  
**Manual de PHP Licencia General Pública GNU**

## 2.9.5. Cadenas

Las cadenas de caracteres se pueden especificar usando uno de dos tipos de delimitadores. Si la cadena está encerrada entre dobles comillas ("), las variables que estén dentro de la cadena serán expandidas (sujetas a ciertas limitaciones de interpretación). Como en C y en Perl, el carácter de barra invertida ("\") se puede usar para especificar caracteres especiales:



<b>secuencia</b>	<b>significado</b>
<code>\n</code>	Nueva línea
<code>\r</code>	Retorno de carro
<code>\t</code>	Tabulación horizontal
<code>\\</code>	Barra invertida
<code>\\$</code>	Signo del dólar
<code>\"</code>	Comillas dobles
<code>\[0-7]{1,3}</code>	la secuencia de caracteres que coincida con la expresión regular es un carácter en notación octal
<code>\x[0-9A-Fa-f]{1,2}</code>	la secuencia de caracteres que coincida con la expresión regular es un carácter en notación hexadecimal

**Cuadro 2.11. Caracteres protegidos**  
**Manual de PHP Licencia General Pública GNU**

Se puede proteger cualquier otro carácter, pero se producirá una advertencia en el nivel de depuración más alto.

## **2.10. Variables**

En PHP las variables se representan como un signo de dólar seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas.

```
$var = "Bob";
$Var = "Joe";
echo "$var, $Var"; // produce la salida "Bob, Joe"
```

**Cuadro 2.12. Variables**  
**Manual de PHP Licencia General Pública GNU**

### **2.10.1. Variables predefinidas**

PHP proporciona una gran cantidad de variables predefinidas a cualquier script que se ejecute. De todas formas, muchas de esas variables no pueden estar

completamente documentadas ya que dependen de sobre qué servidor se esté ejecutando, la versión y configuración de dicho servidor, y otros factores.

Algunas de estas variables no estarán disponibles cuando se ejecute PHP desde la línea de comandos.

### **2.10.2. Variables de entorno**

Estas variables se importan en el espacio de nombres global de PHP desde el entorno en el que se esté ejecutando el intérprete PHP. Muchas son proporcionadas por el intérprete de comandos en el que se está ejecutando PHP, y dado que a sistemas diferentes les gusta ejecutar diferentes tipos de intérpretes de comandos, es imposible hacer una lista definitiva.

### **2.10.4. Ámbito de las variables**

El ámbito de una variable es el contexto dentro del que la variable está definida. La mayor parte de las variables PHP sólo tienen un ámbito simple. Este ámbito simple también abarca los ficheros incluidos y los requeridos. Por ejemplo:

```
$a = 1;  
include "b.inc";
```

## **2.11. Constantes**

Una constante es un identificador para expresar un valor simple. Como el nombre sugiere, este valor no puede variar durante la ejecución del script. (Las constantes especiales `__FILE__` y `__LINE__` son una excepción a esto, ya que actualmente no lo son). Una constante es sensible a mayúsculas por defecto.

Por convención, los identificadores de constantes suelen declararse en mayúsculas. El nombre de una constante sigue las mismas reglas que cualquier etiqueta en PHP. Un nombre de constante válido empieza con una letra o un carácter de subrayado, seguido por cualquier número de letras, números, o subrayados. Se podrían expresar mediante la siguiente expresión regular: `[a-zA-Z_\x7f-\xff][a-zA-Z0-9_\x7f-\xff]*`

### 2.11.1. Sintaxis

Se puede definir una constante usando la función `define()`. Una vez definida, no puede ser modificada ni eliminada. Solo se puede definir como constantes valores escalares (boolean, integer, float y string).

Para obtener el valor de una constante solo es necesario especificar su nombre. A diferencia de las variables, no se tiene que especificar el prefijo `$`. También se puede utilizar la función `constant()`, para obtener el valor de una constante, en el caso de que queramos expresarla de forma dinámica usa la función `get_defined_constants()` para obtener una lista de todas las constantes definidas.

### 2.11.2. Definiendo constantes

```
<?php
define("CONSTANT", "Hello world.");
echo CONSTANT; // outputs "Hello world."
echo Constant; // outputs "Constant" and issues a notice.
?>
```

**Cuadro 2.13. Definiendo constantes**  
**Manual de PHP Licencia General Pública GNU**

## 2.12. Expresiones

Las expresiones son la piedra angular de PHP. Casi cualquier cosa que escribes es una expresión; la forma más simple y ajustada de definir una expresión es "cualquier cosa que tiene un valor". Las formas más básicas de expresiones son las constantes y las variables. Cuando escribe "`$a = 5`", está asignando '5' a `$a`. '5', obviamente, tiene el valor 5 o, en otras palabras '5' es una expresión con el valor 5 (en este caso, '5' es una constante entera).

Las funciones son un ejemplo algo más complejo de expresiones. Por ejemplo, la siguiente función:

```
function foo () {  
return 5;  
}
```

**Cuadro 2.14. Funciones**  
**Manual de PHP Licencia General Pública GNU**

Al estar familiarizado con el concepto de funciones se asume que teclear  $\$c = \text{foo}()$  es esencialmente lo mismo que escribir  $\$c = 5$ , y has acertado. Las funciones son expresiones que toman el valor que retornan. Como  $\text{foo}()$  devuelve 5, el valor de la expresión ' $\text{foo}()$ ' es 5. Normalmente las funciones no devuelven un valor fijo, sino que suele ser calculado.

## 2.13. Operadores

### 2.13.1. Operadores Aritméticos

Los operadores permitidos son de: adición, substracción, multiplicación, división y módulo.

ejemplo	nombre	resultado
$\$a + \$b$	Adición	Suma de $\$a$ y $\$b$ .
$\$a - \$b$	Substracción	Diferencia entre $\$a$ y $\$b$ .
$\$a * \$b$	Multiplicación	Producto de $\$a$ and $\$b$ .
$\$a / \$b$	División	Cociente de $\$a$ entre $\$b$ .
$\$a \% \$b$	Módulo	Resto de $\$a$ dividido entre $\$b$ .

**Cuadro 2.15. Operadores Aritméticos**  
**Manual de PHP Licencia General Pública GNU**

Otro operador condicional es el operador "?:" (o ternario), que funciona como en C y otros muchos lenguajes. (expr1) ? (expr2) : (expr3);

La expresión toma el valor expr2 si expr1 se evalúa a cierto, y expr3 si expr1 se evalúa a falso.

## **2.14. Estructuras de Control**

Todo archivo de comandos PHP se compone de una serie de sentencias. Una sentencia puede ser una asignación, una llamada a función, un bucle, una sentencia condicional e incluso una sentencia que no haga nada (una sentencia vacía). Las sentencias normalmente acaban con punto y coma. Además, las sentencias se pueden agrupar en grupos de sentencias encapsulando un grupo de sentencias con llaves.

Un grupo de sentencias es también una sentencia.

### **2.14.1. if**

La construcción if es una de las más importantes características de muchos lenguajes, incluido PHP. Permite la ejecución condicional de fragmentos de código. PHP caracteriza una estructura if que es similar a la de C:

### **2.14.2 else**

A menudo queremos ejecutar una sentencia si se cumple una cierta condición, y una sentencia distinta si la condición no se cumple. Esto es para lo que sirve else. else extiende una sentencia if para ejecutar una sentencia en caso de que la expresión en la sentencia if se evalúe como FALSE. Por ejemplo, el siguiente código mostraría a es mayor que b si \$a fuera mayor que \$b, y a NO es mayor que b en cualquier otro caso.

### **2.14.3. elseif**

elseif, como su nombre sugiere, es una combinación de if y else. Como else, extiende una sentencia if para ejecutar una sentencia diferente en caso de que la expresión if original se evalúa como FALSE. No obstante, a diferencia de else, ejecutará esa

expresión alternativa solamente si la expresión condicional `elseif` se evalúa como `TRUE`.

#### **2.14.4. while**

Los bucles `while` son los tipos de bucle más simples en PHP. Se comportan como su contrapartida en C.

#### **2.14.5. do..while**

“Los bucles `do..while` son muy similares a los bucles `while`, excepto que las condiciones se comprueban al final de cada iteración en vez de al principio. La principal diferencia frente a los bucles regulares `while` es que se garantiza la ejecución de la primera iteración de un bucle `do..while` (la condición se comprueba sólo al final de la iteración), mientras que puede no ser necesariamente ejecutada con un bucle `while` regular (la condición se comprueba al principio de cada iteración, si esta se evalúa como `FALSE` desde el principio la ejecución del bucle finalizará inmediatamente).” (*Rafael Martínez, 1997-2002, p.400*)

#### **2.14.6. for**

Los bucles `for` son los bucles más complejos en PHP. Se comportan como su contrapartida en C. La sintaxis de un bucle `for` es: `for (expr1; expr2; expr3) sentencia`. La primera expresión (`expr1`) se evalúa (ejecuta) incondicionalmente una vez al principio del bucle.

Al comienzo de cada iteración, se evalúa `expr2`. Si se evalúa como `TRUE`, el bucle continúa y las sentencias anidadas se ejecutan. Si se evalúa como `FALSE`, la ejecución del bucle finaliza. Al final de cada iteración, se evalúa (ejecuta) `expr3`.

#### **2.14.7. break**

`break` escapa de la estructuras de control iterante (bucle) actuales `for`, `while`, o `switch`. `break` acepta un parámetro opcional, el cual determina cuantas estructuras de control hay que escapar.

### **2.14.8. continue**

continue se usa dentro de la estructura del bucle para saltar el resto de la iteración actual del bucle y continuar la ejecución al comienzo de la siguiente iteración; continue acepta un parámetro opcional, el cual determina cuantos niveles (bucles) hay que saltar antes de continuar con la ejecución.

### **2.14.9. switch**

La sentencia switch es similar a una serie de sentencias IF en la misma expresión. En muchas ocasiones, se quiere comparar la misma variable (o expresión) con muchos valores diferentes, y ejecutar una parte de código distinta dependiendo de a qué valor es igual. Para ello sirve la sentencia switch.

### **2.14.10. Sintaxis Alternativa de Estructuras de Control**

PHP ofrece una sintaxis alternativa para algunas de sus estructuras de control; a saber, if, while, for, y switch. En cada caso, la forma básica de la sintaxis alternativa es cambiar abrir-llave por dos puntos (:) y cerrar-llave por endif;, endwhile;, endfor;, or endswitch;, respectivamente.

## **2.15. Conclusión**

PHP se presenta como una propuesta moderna para el desarrollo de aplicaciones a nivel de Internet, en los años que se esta trabajando con este lenguaje de quinta generación ha sentado y tiene las bases sólidas para desarrollar un sinnúmero de aplicaciones y para proyectarse a nuevas versiones con herramientas más poderosas y ampliando su campo de acción a otros Sistemas Operativos.

Al ser un lenguaje que aparece a la par del desarrollo de Linux es parte del mundo del software libre, en cualquier parte que se esté desarrollando con PHP se encontrarán personas que brinden el soporte necesario en especial a los profesionales y programadores en general que están acostumbrados a trabajar con lenguajes clásicos y tradicionales.

## CAPITULO 3

### PHP-NUKE

#### 3.1. Introducción

“PHP-Nuke es un sistema automatizado de noticias especialmente diseñado para ser usado en Intranets e Internet. El Administrador tiene el control total de su sitio Web, sus usuarios registrados, y tendrá a la mano un conjunto de herramientas poderosas para mantener una página Web activa y 100% interactiva usando bases de datos.”

*(Francisco Burzi.)*

Php Nuke permite generar páginas Web dinámicas, crear foros, espacios de noticias, encuestas; y todo tipo de utilidades Web que pueda necesitar un portal profesional.

#### 3.2. Historia

Su autor es Francisco Burzi, que es el que mantiene el código y realiza todas las modificaciones que lleva el paquete original. Se puede encontrar su trabajo en <http://www.phpnuke.org/>. Los requisitos para usar PHP-Nuke no forman parte del sistema y han de instalarse independientemente.

#### 3.3. La licencia de PHP-Nuke

PHP-Nuke se acoge a la licencia GPL, lo que nos permite obtener su código fuente y modificarlo para adaptarlo a nuestras necesidades siempre que respetemos dicha licencia. Una copia de la licencia se encuentra en <http://www.gnu.org/>.

#### 3.4. Evolución de PHP-Nuke

Hay muchos usuarios que se preguntan por qué PHP-Nuke está tan orientado a entornos Linux. La respuesta es sencilla, el proyecto comenzó siendo el sistema utilizado para un sitio llamado Linux Preview. Con el tiempo se ha ido adaptando a los usuarios Windows debido, sobre todo, a que Apache, PHP y MySQL han sido portadas a dicha plataforma.



La primera versión de PHP-Nuke vio la luz en Junio del año 2000, era la versión 1.0. La versión 2.02 apareció en Julio del mismo año. En Agosto se lanzan las versiones 2.5 y 3.0. Las versiones 3.5 y 3.6 salen en Septiembre. En Octubre saldría la revolucionaria 4.0, junto con las actualizaciones 4.2 y 4.3.

Las versiones 4.4 y 4.4.1 perfeccionaron la serie 4.x. Un año después de la primera versión, saldría la primera versión de la actual serie PHP-Nuke 5.0 en julio y agosto aparecen las actualizaciones 5.0.1, 5.1 y 5.2. En noviembre aparece la versión 5.3 que trae, como novedad más destacada, la abstracción de la base de datos.

Las versiones 5.3.1 (noviembre) y 5.4 (enero 2002) suponen el paso previo a la actual versión. La versión 5.5 es una reestructuración de PHP-Nuke, ahora el sistema es totalmente modular.

### 3.5. Requisitos para el uso de PHP-Nuke

Para poder utilizar PHP-Nuke, necesitamos un servidor que nos permita el uso de PHP y una base de datos. Se recomienda MySQL por ser la utilizada en el desarrollo de PHP-Nuke, aunque se puede utilizar PostgreSQL, mSQL, ODBC, ODBC Adabas, Sybase, Interbase. Lo más común es contar con un servidor Web Apache con soporte PHP y el servidor de bases de datos MySQL.

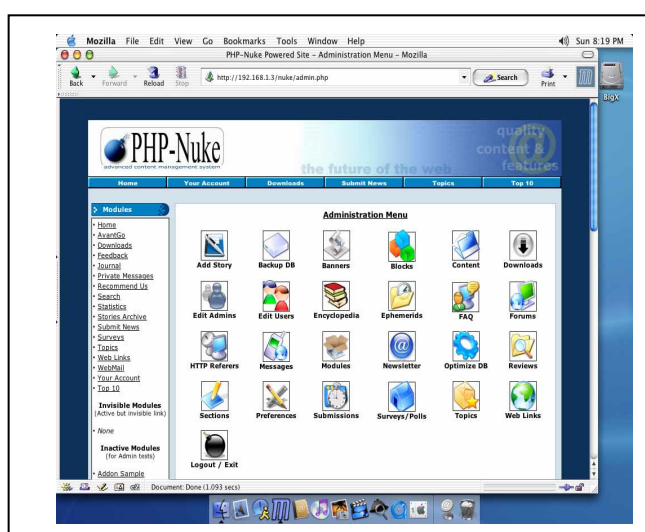


Imagen 3.1. PHP-NUKE

Programación Web Masters

## 3.6. Instalación

### 3.6.1. Instalación en Linux

Para la instalación en Linux necesitamos tener acceso a la cuenta de administración (root) y una serie de paquetes en orden para poder instalar los servicios necesarios.

**Para Red Hat y Mandrake serán necesarios:**

apache-1.20

php-4.0.0

mysql-3.23

php-mysql

Trataremos de tener las versiones en rpm más recientes posible para tener todas las funcionalidades y presentar la menor cantidad de errores posibles.

**Luego instalaremos los paquetes con:** rpm -i nombre\_del\_paquete.rpm

Lo mejor es empezar con Apache, luego MySQL y finalmente PHP y php-mysql, por razones de dependencias.

Las instalaciones en Red Hat y Mandrake sólo difieren en que los paquetes son específicos para cada distribución, aunque, en la mayoría de los casos son compatibles entre sí.

#### 3.6.1.1. Creación de la base de datos

```
[root@Servidor root]# mysqladmin create nuke -u root -p  
Password:
```

La opción -u root la usaremos en caso de utilizar un usuario diferente a root. La opción -p sirve para que nos pida la contraseña de administración de MySQL. En nuestro caso no existe y le daremos a entrar.

### **3.6.2. Instalación en Windows**

Para la instalación en Windows podemos elegir entre dos modalidades: conseguir alguno de los paquetes en los que vienen Apache, PHP y MySQL integrados o ir a las páginas de cada uno de estos programas y obtener las versiones para Windows. Trataremos aquí de instalar por los dos métodos y comenzaremos por el primero de ellos.

### **3.6.3. Instalación de PHP-Nuke**

Después de tener instalados los distintos servidores que vamos a utilizar, la instalación de PHP-Nuke se puede dividir en dos partes:

- 1 Copiar los ficheros necesarios para utilizar PHP-Nuke.
- 2 Introducir las tablas necesarias dentro de la base de datos.

El paso previo a cualquiera de los dos anteriores es descomprimir el paquete PHP-Nuke que hayamos descargado:

En Windows se debe utilizar Winzip, en sus versiones más modernas.

En Linux tenemos con las herramientas del sistema. Puede ejecutarse paso a paso o con: `gunzip PHP-Nuke-5.5.tar.gz && tar -xf PHP-Nuke-5.5.tar`

### **3.6.4. La carpeta html**

Después de descomprimir los archivos, tendremos una serie de archivos y dos directorios, html y sql. Dentro del directorio html tendremos los ficheros necesarios para el funcionamiento de la página. Los debemos copiar al directorio dónde vayan a residir los ficheros de acceso mediante Web. Lo normal es utilizar algún cliente de ftp.

### 3.6.5. El fichero nuke.sql

La configuración de la base de datos reside en el fichero nuke.sql que se encuentra en el directorio sql. Su configuración es idéntica bajo Linux o bajo Windows, aunque hay algunos matices:

Para Linux sería: `mysqladmin create nuke -u root -p`

con lo que nos pedirá la clave de administración de mysql. Luego debemos reiniciar MySQL con:

```
mysqladmin reload -u root -p
```

para poder hacer uso de la base creada.

En Windows tenemos que situarnos en el directorio bin dentro de la ruta de instalación de MySQL y la sintaxis sería:

```
mysqladmin create nuke
```

pero no hay necesidad de utilizar la clave. Luego reiniciamos el proceso de MySQL.

Para la introducción de las tablas usaremos la siguiente sintaxis para ambos sistemas:

```
mysql nuke < nuke.sql -u root -p
```

omitiremos la `-p` si estamos en Windows. Debemos tener nuke.sql en el mismo directorio en el que estemos trabajando o referirnos a él con la ruta completa.

Lo más lógico es tener un usuario que tenga permisos para manejar la base de datos, de esta manera, en caso de problemas, no exponemos los datos del administrador de MySQL. La orden sería:

```
[usuario@servidor directorio]$ mysql -u root
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 4 to server version: 3.23.41
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
mysql>
```

En la línea de comandos de MySQL escribiremos:

```
mysql> GRANT ALL ON base_datos.* TO usuario@localhost IDENTIFIED BY "clave"
```

Ahora podremos usar para el acceso a todas las tablas de la base de datos base\_datos con el usuario usuario y la clave clave.

### 3.6.6. El fichero config.php

Este fichero es clave en la configuración de PHP-Nuke. En él residen todos los parámetros que van a definir nuestro sitio, desde el acceso a la base de datos hasta el aspecto que va a tener el mismo.

Para que nuestro sitio funcione debemos fijarnos en las siguientes variables:

```
$dbhost = "localhost";  
$dbuname = "root";  
$dbpass = "";  
$dbname = "nuke";  
$prefix = "nuke";  
$user_prefix = "nuke";  
$dbtype = "MySQL";
```

El resto de opciones están accesibles desde el panel de administración de PHP-Nuke.

Veamos cada una de las variables por separado:

```
dbhost = "localhost";
```

Esta es la dirección del servidor de base de datos, lo normal es que esté en la misma máquina que PHP-Nuke, lo dejaremos tal y como está.

```
dbuname = "root";
```

Este es el usuario que tiene permiso para acceder a la base de datos, podemos poner el administrador de MySQL o el usuario al que le dimos acceso anteriormente.

```
dbpass = "";
```

La clave de acceso a la base de datos.

```
dbname = "nuke";
```

El nombre de la base de datos donde residen las tablas que va a utilizar PHP-Nuke.

```
prefix = "nuke";
```

Este es el prefijo de las tablas de la base de datos. Si hemos utilizado nuke.sql sin modificar, será 'nuke'.

```
user_prefix = "nuke";
```

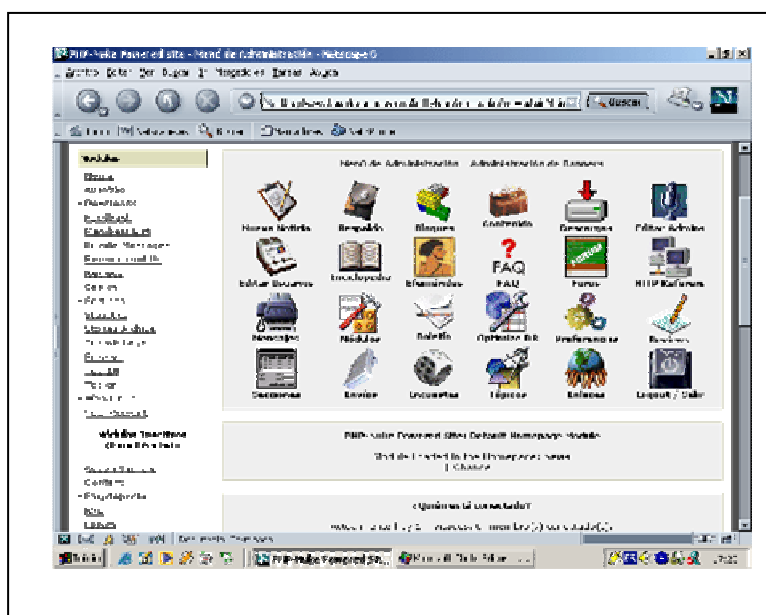
El prefijo de la tabla de usuarios de PHP-Nuke. Éste nos servirá si tenemos dos sitios hechos con PHP-Nuke y queremos compartir los usuarios.

```
dbtype = "MySQL";
```

Aquí indicamos el tipo de base de datos a utilizar. Originalmente PHP-Nuke se desarrolló sobre MySQL, en la actualidad soporta más motores de bases de datos.

### 3.7. Administración

La administración de PHP-Nuke se hace, en su mayor parte, desde un panel de control accesible desde un navegador. La parte que no se hace desde aquí es la de añadir nuevas funcionalidades puesto que requiere la copia de archivos y, ocasionalmente, la creación de tablas en la base de datos.



**Imagen 3.2. El Menú de Administración  
Programación Web Masters**

#### 3.7.1. Nueva Noticia

Cuando queramos añadir una noticia a nuestro sitio, seleccionaremos esta opción y llenaremos los campos que se nos presentan.

#### 3.7.2. Respaldo

Esta es la opción que genera una copia de seguridad de la base de datos de nuestro PHP-Nuke.

#### 3.7.3. Bloques

Los bloques son aquellas partes de nuestro sitio que podemos ver a la izquierda y derecha. Desde esta opción accederemos al panel que nos permitirá decidir qué bloques tenemos y en qué posición lo vamos a poner.

### **3.7.4. Contenido**

Esta opción permite crear páginas de contenido diverso, nos da la opción de poder crear un documento de varias páginas con el tag (etiqueta) <!--pagebreak-->. Esto permite que como administrador puedas tener tus propios contenidos, sólo tienes que agregar los tags si quieres darle formato, y escribir el texto.

### **3.7.5. Descargas**

Esta opción permite administrar las descargas, así como asignarles categorías, está dividida en varios campos:

Nombre de la categoría.

Descripción de la categoría.

Nombre de la subcategoría.

Categoría a la que será asignada la subcategoría.

Nombre del archivo tal y como será visto en el área de descargas.

URL del archivo a descargar.

Categoría o subcategoría a la que será asignada las descarga.

Descripción del archivo.

Nombre del autor del archivo.

Correo electrónico del autor del archivo.

Tamaño del archivo.

Versión del archivo a descargar.

Página del o los autores del archivo.

Votos para el archivo.

Y modificar categorías.

### **3.7.6. Editar Administradores**

Con esta opción podemos crear, editar y borrar usuarios (la cuenta God no se puede borrar). Editar administradores nos da oportunidad de asignar permisos a los administradores, sólo hay que llenar los siguientes campos:

Nombre del administrador, mismo que no podrá ser cambiado.

Alias de administrador, esto es como será visto por los usuarios normales.

Dirección de correo.

Permisos, si se asigna como súper usuario tendrá acceso a todas las opciones y no es necesario marcar ninguna más, hay que tener mucho cuidado con esto.

Contraseña o password del administrador.

Todo esto se aplica a la edición de administradores.

### **3.7.7. Enciclopedia**

Permite generar una enciclopedia basada en categorías, sólo se tiene que poner nombre a una enciclopedia y empezar a agregar los términos, si se necesita más de una página puede usar el tag `<!--pagebreak-->`.

### **3.7.8. Efemérides**

Desde esta opción podremos poner las efemérides que queremos que salgan según el día en el que estemos.

### **3.7.9. Foros**

Los foros pueden ser creados y administrados desde este panel, se puede asignar una categoría, un moderador, definir el tipo de acceso, la descripción de un foro, así como el nivel de los usuarios que ponen tópicos en el foro, los emoticonos se pueden configurar, la censura (cambiar una palabra por otra), hasta la restricción de ips para acceder al foro.

### **3.7.10. Referencias HTTP**

PHP-Nuke controla desde qué páginas enlazan nuestro sitio, desde esta opción podremos ver esa información. La cantidad de referencias que pueden salir se controlan desde el panel de Preferencias.



### **3.8. Conclusión**

La compatibilidad de herramientas de desarrollo simplifica el manejo de Sitios Web, con Administradores como NUKE es más fácil la elaboración y mantenimiento de Portales.

El ganador del desarrollo de PHP-NUKE es el usuario, que en cualquier caso verá como el software libre puede producir para él un sistema de administración de contenidos, usuarios, links y downloads de primer nivel, esto repercute en ganancias para si y la compañía en la que este laborando.

## **CAPITULO 4**

### **SERVIDOR APACHE**

#### **4.1. Introducción**

Apache es el servidor Web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

#### **4.2. Historia**

La historia de Apache se remonta a febrero de 1995, donde empieza el proyecto del grupo Apache, el cual esta basado en el servidor Apache httpd de la aplicación original de NCSA. El desarrollo de esta aplicación original se estancó por algún tiempo tras la marcha de Rob McCool por lo que varios Webmaster siguieron creando sus parches para sus servidores Web hasta que se contactaron vía email para seguir en conjunto el mantenimiento del servidor Web, fue ahí cuando formaron el grupo Apache.

Fueron Brian Behlendorf y Cliff Skolnick quienes a través de una lista de correo coordinaron el trabajo y lograron establecer un espacio compartido de libre acceso para los desarrolladores.

Fue así como fue creciendo el grupo Apache, hasta lo que es hoy. Aquella primera versión y sus sucesivas evoluciones y mejoras alcanzaron una gran implantación como software de servidor inicialmente solo para sistemas operativos UNIX y fruto de esa evolución es la versión para Windows.

#### **4.3. Licencia**

La licencia Apache es una descendiente de la licencias BSD, no es GPL. Esta licencia permite modificar el código fuente (incluso forks y productos propietarios) siempre que se les reconozca su trabajo.

#### 4.4. Características

La popularidad de este software libre grandemente reconocido en muchos ámbitos empresariales y tecnológicos es por las siguientes razones:

1. Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
2. Apache es una tecnología gratuita de código fuente abierto. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si queremos ver que es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto, sin ninguna puerta trasera.
3. Apache es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que los instalemos cuando los necesitemos. Otra cosa importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un modulo para realizar una función determinada.
4. Apache trabaja con gran cantidad de Perl, PHP y otros lenguajes de script. Perl destaca en el mundo del script y Apache utiliza su parte de Perl tanto con soporte CGI como con soporte modo perl. También trabaja con Java y páginas jsp. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
5. Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.

Se pueden extender las características de Apache hasta donde nuestra imaginación y conocimientos lleguen.

#### **4.5. Conclusión**

El Servidor Web Apache es un ejemplo, al igual que Linux de que el trabajo voluntario y cooperativo dentro de Internet es capaz de producir aplicaciones de calidad profesional de alto rendimiento que aceleran el crecimiento del Software GNU, al estar migrado a Windows crece aún más su potencialidad tomando en cuenta tan solo el número de usuarios que tiene Microsoft a nivel mundial.

Como un comentario a fe de errata, personalmente pensaba que el nombre Apache es en honor a los indios americanos, cosa que no es, más bien es un lastre de parchar o parche, ya que como se vio es un software que se lo manipulaba a criterio de muchos programadores en base a necesidades puntuales pero hoy es un paquete de categoría.

## CAPITULO 5

### MY-SQL

#### 5.1. Introducción a las Bases de Datos

“El término base de datos corresponde a un conjunto auto descriptivo de registros integrados. Es auto descriptivo porque además de los datos fuentes del usuario, contiene una descripción de su propia estructura; tal descripción es conocida como diccionario de datos.” (Oswaldo Merchán, 2003, p.3)

La Base de Datos es un depósito único de datos para toda la organización, por lo que debe ser capaz de integrar los distintos sistemas y aplicaciones, atendiendo a las necesidades de los usuarios en los niveles: operativo, táctico y estratégico.

“Demuestra el mejor rendimiento y escalabilidad junto a otras soluciones líderes en el mercado de bases de datos empresariales. MySQL tiene un rendimiento excelente en estabilidad, facilidad de afinación y conectividad. Considerando su precio (gratuito) y su facilidad de uso, MySQL se posiciona como la base de datos a elegir cuando el mejor rendimiento para uso en aplicaciones Internet.

Según la revista PCMagazine, MySQL compite casi a la par en características importantes como escalabilidad y desempeño con otras soluciones comerciales. Esto la ubica dentro del espectro de posibilidades al momento de elegir una plataforma de base de datos dentro de la empresa a esta singular y veloz base de datos.” (*YAGE, evolución digital*)

#### 5.2. Historia

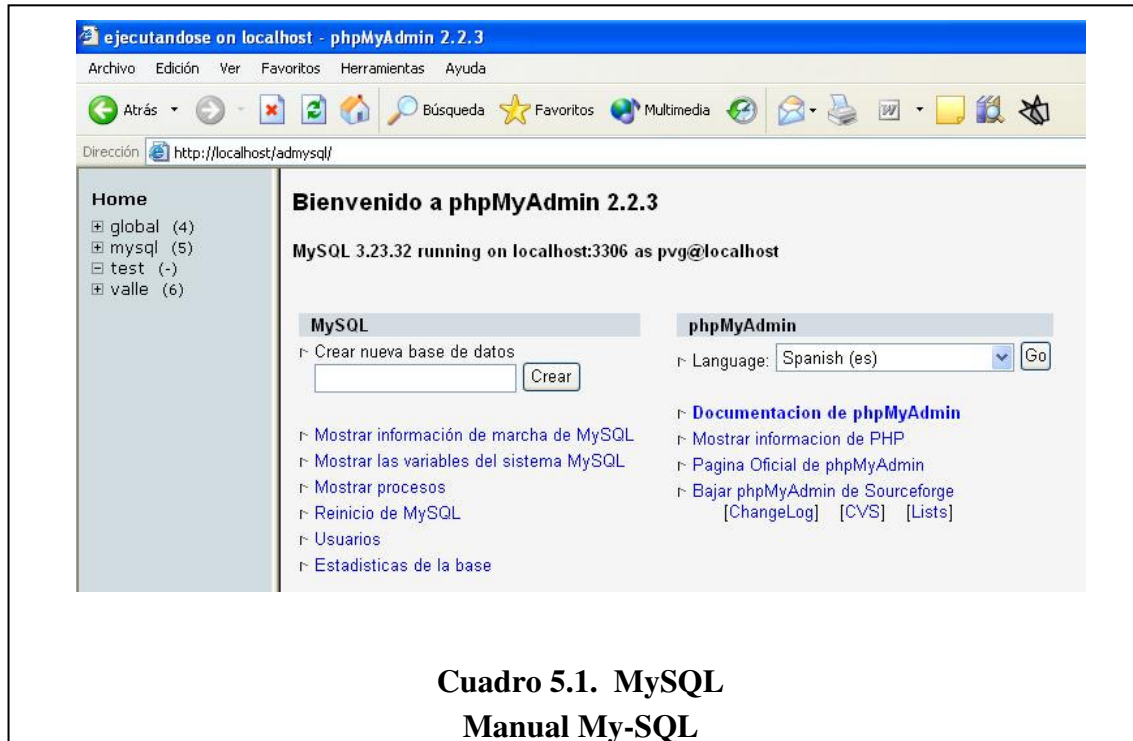
MySQL surgió como un intento de conectar el gestor mSQL a las tablas propias de MySQL AB, usando sus propias rutinas a bajo nivel. Tras unas primeras pruebas, vieron que mSQL no era lo bastante flexible para lo que necesitaban, por lo que tuvieron que desarrollar nuevas funciones. Esto resultó en una interfaz SQL a su base de datos, con una interfaz totalmente compatible a mSQL.

No se sabe con certeza de donde proviene su nombre. Por un lado dicen que sus librerías han llevado el prefijo 'my' durante los diez últimos años. Por otro lado, la hija de uno de los desarrolladores se llama My. No saben cuál de estas dos causas (aunque bien podrían tratarse de la misma), han dado lugar al nombre de este conocido gestor de bases de datos.

### 5.3. Características de MySQL

Las principales características de este gestor de bases de datos son las siguientes:

1. Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
2. Soporta gran cantidad de tipos de datos para las columnas.
3. Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
4. Gran portabilidad entre sistemas.
5. Soporta hasta 32 índices por tabla.
6. Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.



**Cuadro 5.1. MySQL  
Manual My-SQL**

## 5.4. El lenguaje SQL

SQL es una herramienta para organizar, gestionar y recuperar datos almacenados en una base de datos informática. El nombre "SQL" es una abreviatura de *Structured Query Language* (Lenguaje de consultas estructurado). Como su propio nombre indica, SQL es un lenguaje informático que se puede utilizar para interactuar con una base de datos y más concretamente con un tipo específico llamado *base de datos relacional*. MY-SQL es a la vez un lenguaje fácil de aprender y una herramienta completa para gestionar datos.

Las peticiones sobre los datos se expresan mediante sentencias, que deben escribirse de acuerdo con unas reglas sintácticas y semánticas de este lenguaje. Su aprendizaje no solo sirve para esta aplicación sino para todas las existentes en el mercado que soporten este lenguaje ya que es un lenguaje estándar por haberse visto consolidado por el Instituto Americano de Normas (ANSI) y por la Organización de Estándares Internacional (ISO).

### 5.4.1. Cláusula SELECT

La cláusula SELECT lista los datos a recuperar por la sentencia SELECT. Los elementos o datos a seleccionar pueden ser columnas de la base de datos o columnas a calcular por SQL cuando efectúa la consulta o también el asterisco (\*) para recuperar todos los campos de un fichero o tabla.

Los nombres de campos pueden ir precedidos por el nombre de la tabla o su alias. Por ejemplo ALUMNOS.MATRICULA o A.MATRICULA donde A es el alias para la tabla o fichero ALUMNOS.

El **operador DISTINCT**, si se incluye, debe preceder la primera expresión de columna. Este operador elimina las filas o registros duplicados del resultado de la consulta. Por ejemplo la sentencia:

```
SELECT DISTINCT PROVINCIA FROM ALUMNOS
```

mostrará las *distintas* provincias de las que proceden los alumnos del Centro.

### 5.4.2. Funciones de agrupamiento

Las funciones de agrupamiento pueden ser también parte de una cláusula SELECT. Devuelven un único valor de un conjunto de registros. Pueden usarse con un nombre de campo (por ejemplo, AVG(NUMERO\_DE\_HERMANOS) o en combinación con una expresión de columna más compleja (por ejemplo, AVG(NUMERO\_DE\_HERMANOS \* 1.07) ). La expresión de columna puede ir precedida por el operador DISTINCT. El operador DISTINCT eliminará los valores repetidos de una expresión de agrupamiento.

### 5.4.3. Funciones de agrupamiento permitidas

**SUM** Devuelve la suma total de los valores de una expresión de columna o campo numérica. Por ejemplo, SUM(NUMERO\_DE\_HERMANOS) devolverá el total de hermanos de los alumnos del Centro.

**AVG** Devuelve la media de los valores de una expresión de columna. Por ejemplo, AVG(NUMERO\_DE\_HERMANOS) devolverá la media de hermanos que posee el alumnado del centro.

**COUNT** Devuelve el número de valores en una expresión de columna. Por ejemplo, COUNT(MATRICULA) devolverá el número de registros con valores no nulos en ese campo. COUNT(PAIS) nos dirá el número de alumnos que en el campo país tiene alguna información y no está a blanco. Un ejemplo especial es COUNT(\*), que nos devuelve el número de registros incluyendo aquellos registros con valores nulos.

**MAX** Devuelve el valor más alto de los contenidos en una expresión de columna. Por ejemplo, MAX(MATRICULA) devolverá el número de matrícula más elevado que existe en el fichero de alumnos.

**MIN** Devuelve el valor más bajo de los contenidos en una expresión de columna. Por ejemplo, `SELECT MIN(FECHA_NACIMIENTO) FROM ALUMNOS` nos devolverá la fecha de nacimiento del alumno más joven del centro.



#### 5.4.4. Cláusula FROM

La cláusula FROM lista las tablas o ficheros que contienen los datos a recuperar por la consulta. El formato de esta cláusula es:

**FROM *nombretabla* [*alias\_tabla*] ...**

*nombretabla* puede ser una o mas nombres de tabla en el directorio de trabajo si se omite este, o en un directorio distinto si se especifica.

*alias\_tabla* es un nombre que se usa para referirse a la tabla en el resto de la sentencia SELECT para abreviar el nombre original y hacerlo más manejable, en el caso de existir más de una tabla en la consulta y, también para poder realizar consultas uniendo varias veces la misma tabla. ejm:

```
SELECT A.NOMBRE, A.APELLIDOS FROM MATRICUL M, ALUMNOS A
WHERE M.MATRICULA = A.MATRICULA AND M.GRUPO = '1A' AND
ANNO = 1995
```

#### 5.4.5. Cláusula WHERE

La cláusula WHERE dice a SQL que incluya solo ciertas filas o registros de datos en los resultados de la consulta, es decir, que tienen que cumplir los registros que se desean ver. La cláusula WHERE contiene condiciones en la forma:

**WHERE *expresión1* operador *expresión2***

*expresión1* y *expresión2* pueden ser nombres de campos, valores constantes o expresiones.

#### 5.4.6. Cláusula GROUP BY

La cláusula GROUP BY especifica una consulta sumaria. En vez de producir una fila de resultados por cada fila de datos de la base de datos, una consulta sumaria agrupa todas las filas similares y luego produce una fila sumaria de resultados para cada grupo. Seguido de la cláusula GROUP BY se especifican los nombres de uno o más campos cuyos resultados se desean agrupados. Tiene la forma: **GROUP BY *expresión\_columna* *expresión\_columna*** debe coincidir con la expresión de columna utilizada en la cláusula SELECT. Puede ser uno o más

nombres de campo de una tabla, separados por coma o una o más expresiones separadas por comas. El siguiente ejemplo nos dice cuantos alumnos están matriculados en cada grupo en el año 1995:

#### **5.4.7. Cláusula HAVING**

La cláusula HAVING dice a SQL que incluya solo ciertos grupos producidos por la cláusula GROUP BY en los resultados de la consulta. Al igual que la cláusula WHERE, utiliza una condición de búsqueda para especificar los grupos deseados. En otras palabras, especifica la condición que deben de cumplir los grupos. Sólo es válida si previamente se ha especificado la cláusula GROUP BY. La cláusula HAVING tiene la forma:

**HAVING *expresión1* operador *expresión2***

*expresión1* y *expresión2* pueden ser nombres de campos, valores constantes o expresiones y estas no deben coincidir con una expresión de columna en la cláusula SELECT.

#### **5.4.8. Operador UNION**

El operador UNION combina el resultado de dos sentencias SELECT en un único resultado. Este resultado se compone de todos los registros devueltos en ambas sentencias. Por defecto, los registros repetidos se omiten. Para no quitarlos se empleará la palabra ALL. Tiene la forma:

**SELECT *sentencia* UNION [ALL] SELECT *sentencia***

Cuando se utilice el operador UNION, la lista de selección para cada sentencia SELECT debe tener el mismo número de expresiones de columnas con el mismo tipo de datos y en el mismo orden.

#### 5.4.9. Cláusula ORDER BY

La cláusula ORDER BY ordena los resultados de la consulta en base a los datos de una o más columnas. Si se omite, los resultados saldrán ordenados por el primer campo que sea clave en el índice que se haya utilizado.

Por tanto, indica como deben clasificarse los registros que se seleccionen. Tiene la forma:

**ORDER BY {*expresión\_orden* [DESC | ASC], ... ]**

*expresión\_orden* puede ser el nombre de un campo, expresión o el número de posición que ocupa la expresión de columna en la cláusula SELECT. Por defecto se ordenan ASCendentemente (de menor a mayor). Si se deseará de mayor a menor se empleará DESC (DESCendente).

#### 5.4.10. Enlace de varias Tablas o ficheros

Los ejemplos visto hasta el momento solo extraen datos de una única tabla y pocas cosas podríamos hacer si no pudiéramos interrelacionar varios ficheros para obtener las consultas que deseáramos.

El enlace o relación de dos tablas para realizar una consulta, se conoce con el término de JOIN. Para expresar este término vamos a utilizar un ejemplo que se utiliza muy a menudo en todas las consultas de GEIWIN: Deseamos un listado alfabético con los Nombres y Apellidos de todos los alumnos que están matriculados, de forma Oficial, actualmente en el centro.

El fichero de ALUMNOS contiene todos los datos generales del alumnado, entre ellos Nombre y Apellidos, y el fichero MATRICUL los distintos datos de las diferentes matrículas de cada uno de ellos, entre ellos el año y tipo de la matrícula.

#### 5.4.11. Alias de las Tablas

Los *alias* son un instrumento para abreviar los nombres de las tablas y poder referirse a ellos en toda la sentencia. ejm:

```
SELECT APELLIDOS, NOMBRE FROM MATRICUL M, ALUMNOS A WHERE  
M.MATRICULA = A.MATRICULA AND ANNO = 1996 AND TIPO = 'O'  
ORDER BY APELLIDOS, NOMBRE
```

#### **5.4.12. Orden de las tablas en la cláusula FROM**

Especial mención merece el orden de enumeración de las tablas en la cláusula FROM. El Gestor de SQL, cuando existe un JOIN entre dos tablas, recorre la tabla que figura en primer lugar, y por cada registro que encuentra que satisface el resto de restricciones, comprueba si existe correspondencia en la otra tabla.

#### **5.4.13. Enlace de más de dos tablas**

El Gestor de SQL puede unir muchas tablas en una única sentencia, para ello debemos efectuar el *Join* entre todas ellas y de la forma más óptima posible para que el tiempo de respuesta sea mínimo.

#### **5.4.14. SELECT anidadas**

Pueden existir consultas a la Base de Datos que requieran una restricción que sea el resultado de otra consulta. Cuando esto se produce podemos actuar de dos formas:

1.- La primera efectuando la consulta que genera la restricción y posteriormente, realizar la sentencia con el resultado de la restricción calculada:

```
SELECT MAX(CURSO) FROM PLANES WHERE ESTUDIO = '@Estudio'
```

Si, por ejemplo, el valor máximo nos devuelve **3**, entonces realizaríamos la sentencia definitiva con:

```
SELECT COUNT(MATERIA) FROM PLANES WHERE ESTUDIO =  
'@Estudio' AND CURSO = '3'
```

2.- La segunda forma es más compleja pero mucho más práctica. La sentencia que genera la restricción compleja se anida dentro de la principal encerrándola entre paréntesis:

```
SELECT COUNT(MATERIA) FROM PLANES WHERE ESTUDIO =  
'@Estudio' AND CURSO = (SELECT MAX(CURSO) FROM PLANES  
WHERE ESTUDIO = '@Estudio')
```

Las sentencias SELECT anidadas se utilizan con mucha frecuencia a la hora de Actualizar o Borrar registros de una tabla que requieran condiciones o restricciones en las que están implicadas otras tablas distintas de la que se va a actualizar o borrar.

### **5.5. Expresiones SQL**

Las expresiones se utilizan en las cláusulas WHERE, HAVING y ORDER BY de las sentencias SELECT.

Las expresiones nos permiten utilizar operaciones matemáticas como también cadenas de caracteres y operadores de manipulación de fechas para construir consultas complejas.

Los elementos que componen las expresiones son:

- Nombres de campos
- Operadores numéricos
- Operadores de fechas
- Operadores lógicos
- Constantes
- Operadores de caracteres
- Operadores de relación
- Funciones

### **5.6. Funciones**

Las funciones permiten realizar con los datos operaciones adicionales a las ya vistas, pudiendo participar como operadores en las expresiones.

Una función representa un valor único que se obtiene aplicando unas determinadas operaciones a otros valores dados, que se llaman argumentos. Se especifica como una palabra predefinida seguida de los argumentos entre paréntesis y separados por comas.

El lenguaje SQL dispone de un conjunto de funciones que pueden usarse en las consultas y que aparecen aquí agrupadas por el tipo de valores que devuelven.

## **5.7. Sentencias para crear o destruir tablas**

La sentencia para crear una tabla es CREATE y para destruirla DROP.

### **5.7.1. Creación de una tabla**

La sentencia para crear una tabla tiene la forma:

```
CREATE TABLE nombre_fichero (definición_columna, ...)
```

*nombre\_fichero* puede ser únicamente el nombre del fichero o el nombre completo con su camino. Si se utiliza solo el nombre del fichero se crea en el directorio donde se encuentran los ficheros de la aplicación (C:\geiwin\datos).

*definición\_columna* esta compuesto por el nombre de la columna o campo, seguida del tipo de dato de dicha columna; los nombres de columna no deben tener espacios en blanco.

**ALPHANUMERIC** Sus valores pueden tener letras, números o caracteres de puntuación. Se precisa un parámetro para la longitud máxima que se quiere almacenar y esta no puede sobrepasar los 255 caracteres.

**BLOB** Sus valores contienen una clase de datos binarios de longitud indefinida que se suelen utilizar para imágenes.

**CURRENCY** Sus valores contienen solo números empleados en cantidades monetarias. Emplean el mismo formato que el tipo NUMBER.

**DATE** Se emplea para almacenar fechas.

**MEMO** Se emplea para texto de longitud indefinida cuando el tipo ALPHANUMERIC se queda corto.

**NUMBER** Se emplea para cantidades numéricas en general. Pueden almacenar decimales y signo. Internamente se almacenan como decimales de coma flotante de doble precisión.

SHORT Solo puede contener números enteros entre los valores -32767 y 32767.

### 5.7.2. Destrucción de una tabla

El formato para destruir o borrar una tabla es

```
DROP TABLE nombrefichero
```

*nombrefichero* puede ser únicamente el nombre del fichero o el nombre completo con su camino. Si se utiliza solo el nombre del fichero se utiliza el directorio donde se encuentran los ficheros de la aplicación (C:\geiwin\datos).

La sentencia para borrar la tabla antes creada es

```
DROP TABLE D:\LIBROS
```

### 5.8. Sentencia INSERT

La sentencia de INSERT se utiliza para añadir registros a las tablas de la base de datos. El formato de la sentencia es:

```
INSERT INTO nombrefichero [(nombrecolumna, ...)] VALUES (expr, ...)
```

*nombrefichero* puede ser únicamente el nombre del fichero o el nombre completo con su camino. Si se utiliza solo el nombre del fichero se utiliza el directorio donde se encuentran los ficheros de la aplicación (C:\geiwin\datos).

*nombrecolumna* es una lista opcional de nombres de campo en los que se insertarán valores en el mismo número y orden que se especificarán en la cláusula VALUES. Si no se especifica la lista de campos, los valores de *expr* en la cláusula VALUES deben ser tantos como campos tenga la tabla y en el mismo orden que se definieron al crear la tabla.

*expr* es una lista de expresiones o valores constantes, separados por comas, para dar valor a los distintos campos del registro que se añadirá a la tabla. Las cadenas de caracteres deberán estar encerradas entre comillas ' o " y las fechas entre llaves { }.

```
INSERT INTO nombrefichero [(nombrecolumna, ...)] FROM fichero_ascii
```

## 5.9. Sentencia UPDATE

La sentencia UPDATE se utiliza para cambiar el contenido de los registros de una tabla de la base de datos. Su formato es:

```
UPDATE nombre_fichero SET nombre_columna = expr, ...
```

```
[WHERE { condición }]
```

*nombre\_fichero* puede ser únicamente el nombre del fichero o el nombre completo con su camino. Si se utiliza solo el nombre del fichero se utiliza el directorio donde se encuentran los ficheros de la aplicación (C:\geiwin\datos).

*nombre\_columna* es el nombre de columna o campo cuyo valor se desea cambiar. En una misma sentencia UPDATE pueden actualizarse varios campos de cada registro de la tabla.

*expr* es el nuevo valor que se desea asignar al campo que le precede. La expresión puede ser un valor constante o una subconsulta. Las cadenas de caracteres deberán estar encerradas entre comillas ' o " y las fechas entre llaves { }. Las subconsultas entre paréntesis.

La cláusula WHERE sigue el mismo formato que la vista en la sentencia SELECT y determina que registros se modificarán.

## 5.10. Sentencia DELETE

La sentencia DELETE se utiliza para borrar registros de una tabla de la base de datos. El formato de la sentencia es:

```
DELETE FROM nombre_fichero [WHERE { condición }]
```

*nombre\_fichero* puede ser únicamente el nombre del fichero o el nombre completo con su camino. Si se utiliza solo el nombre del fichero se utiliza el directorio donde se encuentran los ficheros de la aplicación (C:\geiwin\datos).

La cláusula WHERE sigue el mismo formato que la vista en la sentencia SELECT y determina que registros se borrarán.



Cada sentencia DELETE borra los registros que cumplen la condición impuesta o todos si no se indica cláusula WHERE.

### **5.11. Transacciones**

Una transacción es una serie de cambios en la base de datos que deben ser tratadas como una sola. En otras palabras, que se realicen todos o que no se haga ninguno, pues de lo contrario se podrían producir inconsistencias en la base de datos.

Cuando no se tiene activada una transacción el gestor de base de datos ejecuta inmediatamente cada sentencia INSERT, UPDATE o DELETE que se le encomiende, sin posibilidad de deshacer los cambios en caso de ocurrir cualquier percance. Cuando se activa una transacción los cambios que se van realizando quedan en un estado de provisionalidad hasta que se realiza un COMMIT, el cual hará definitivos los cambios o hasta realizar un ROLLBACK que deshará todos los cambios producidos desde que se inició la transacción.

### **5.12. ¿MySQL es una alternativa aceptable?**

Debido al elevado costo de una licencia comercial de cualquier vendedor de bases de datos actual, el Open Source definitivamente nos ofrece una alternativa barata y con un desempeño totalmente aceptable. El desarrollar aplicaciones utilizando este tipo de tecnología puede ser la diferencia entre poner a correr su negocio, o gastar una considerable cantidad de su capital en soluciones comerciales.

Sin embargo esta base de datos todavía no nos ofrece todas las funcionalidades que nos ofrecen las comerciales, podemos enumerar: clustering, alta disponibilidad, procesamiento OLAP, datawarehousing, datos geo-espaciales, etc.

Tomando en cuenta los bajos costos de utilizar software Open Source, ¿debería MySQL convertirse en una opción más al momento de considerar la plataforma de base de datos? La pregunta en realidad puede contestarse de la siguiente manera: ¿Cuán importantes son las opciones no ofrecidas por esta solución, que vienen incluidas en la competencia?.

Una solución viable podría ser utilizar tecnología Open Source para prototipar aplicaciones antes de ser puestas en producción. Quizás el camino más lógico sería el utilizar una solución comercial cuando el negocio lo amerite. Para la mayor parte de casos, es más fácil “mover” un proyecto desarrollado inicialmente en MySQL a una base de datos comercial. *(YAGE, evolución digital)*

### **5.13. Conclusión**

Las Bases de Datos aportan en gran medida el desarrollo de un proyecto al constituir no sólo el contenedor o repositorio de información, sino más bien aporta desde un principio en la estructuración lógica de la información que se va a tratar misma que va a ser ingresada, modificada y consultada en diferentes tiempos y sometida a diferentes análisis y cálculos.

Al momento de emprender un proyecto y determinar que base de datos se va a manejar, se debe dar importancia a My-SQL por las ventajas que ofrece y la versatilidad para ser manejada, es en realidad una Base de Datos que tiene futuro.

## CAPITULO 6

### INTEGRACION DE HERRAMIENTAS

#### 6.1. Introducción

En este capítulo se hace un análisis de las herramientas que se va a utilizar en el desarrollo del trabajo práctico mismas que se dieron a conocer en los capítulos anteriores, es importante conocer la integración de estas herramientas, beneficios, ventajas y desventajas.

“PHP-Nuke, PHP, Apache y My-SQL en principio fueron desarrollados para plataformas Linux, hoy siguen trabajando para este Sistema Operativo, no obstante se han llevado al entorno del Sistema Operativo Windows por la creciente demanda de millones de usuarios de Windows alrededor del mundo.” (*Andrés Aguayo Maldonado, 2002, p.100*)

#### 6.2. Aspectos básicos de calidad de software

La descripción que se hace de los factores que influyen en un software de calidad se basa principalmente en las ideas presentadas por Robert Dunn, Philip Crosby y Roger S. Pressman. Sin embargo, también se han tomado algunos aportes de Bertrand Meyer y Mauricio Fernando Alba.

Robert Dunn presenta la calidad en el software tomando dos puntos de vista: la calidad en el proceso de desarrollo y la calidad en el producto final, estos dos grupos principales los agrupa en los siguiente aspectos de calidad: confiabilidad, utilizabilidad, mantenibilidad, y adaptabilidad.

Roger Pressman describe similares factores de calidad agrupados en tres grupos: calidad en operación, calidad en revisión y calidad en transición.

### **6.2.1. Factores de calidad**

Confiabilidad. Este termino es necesario sea separado en varios elementos que permiten darle al software el matiz de fiable. Sus componentes son:

Completitud

Consistencia y precisión

Solidez

Simplicidad

Calidad en los procesos de desarrollo

Seguridad y Verificabilidad, estas dos últimas que se determinan con el sistema en uso.

Usabilidad. Si bien es cierto que la confiabilidad es un factor muy importante en la calidad del software también lo es el hecho de que es necesario considerar otros factores como los que se mencionan en esta sección puesto que de nada sirve un software que funcione correcta y confiablemente si el usuario prefiere no utilizarlo.

Exactitud de los procesos

Claridad y exactitud de la documentación

Completitud

Eficiencia y verificabilidad del software

Claridad y amigabilidad de la interfaz

Mantenibilidad. Este aspecto de calidad involucra los elementos que simplifican la labor de prevención, corrección o ampliación del código del programa. Retomar un código escrito meses antes es un trabajo dispendioso y agobiante, en especial cuando las aplicaciones no cuentan con la característica a la cual aquí se hace referencia. Se pueden considerarse como atributos de este aspecto:

Exactitud y claridad en la documentación

Modularidad acoplamiento

Facilidad de lectura

Simplicidad

Portabilidad. Es la capacidad que posee un sistema de información que le permite funcionar en diferentes plataformas ya sean hardware o de software.

### **6.2.2. Calidad de software**

Se define la calidad de software como la ausencia de errores de funcionamiento, la adecuación a las necesidades del usuario, y el alcance de un desempeño apropiado (tiempo, volumen, espacio), además del cumplimiento de los estándares. Los objetivos que la calidad persigue son: La aceptación (utilización real por parte del usuario) y la Mantenibilidad (posibilidad y facilidad de corrección, ajuste y modificación durante largo tiempo).

“Para alcanzar estos objetivos, es necesaria una actitud y compromiso de todo el personal que se encuentre en el desarrollo del proyecto, y en todas y cada una de las etapas (en general, planeación, análisis, diseño, programación, pruebas, mantenimiento) correspondientes al ciclo de vida que se hubiese seleccionado para el proyecto.” (*Lucas Morea, 1997, p.218.*)

### **6.2.3. Elección de Tecnologías**

La elección de tecnologías para crear una aplicación depende de diversos factores, entre los que se encuentran el nivel de conocimiento de los diferentes lenguajes y el servidor de aplicaciones que se vaya a utilizar.

## **6.3. Beneficios**

Se cita los siguientes aspectos como beneficios al desarrollar con estas herramientas.

- Al ser herramientas nativas de Linux son sumamente livianas al trabajar individualmente y en conjunto.
- Pertenecen al creciente software GNU o software libre, no tenemos problemas con licencias de ningún tipo.
- El aprendizaje de las herramientas es rápido, Php que es extremadamente flexible, rápido y versátil por ejemplo tiene mucha concordancia con el lenguaje C en su versión más sencilla. Al trabajar con MY-SQL como plataforma se logra escalabilidad. Esto quiere decir que se pueden agregar módulos de programación sin

tener que hacer cambios mayores en las partes ya existentes, o sin tener que cambiar la programación entera.

- Una ventaja enorme es el hecho de que estas herramientas trabajan en computadores personales, esto acrecienta la popularidad de su uso.
- Están en constante estudio e innovación, en Internet se puede encontrar material de trabajo como cursos en línea, manuales, foros, entrevistas a expertos, etc.

#### **6.4. Desventajas**

Como desventajas o falencias se cita:

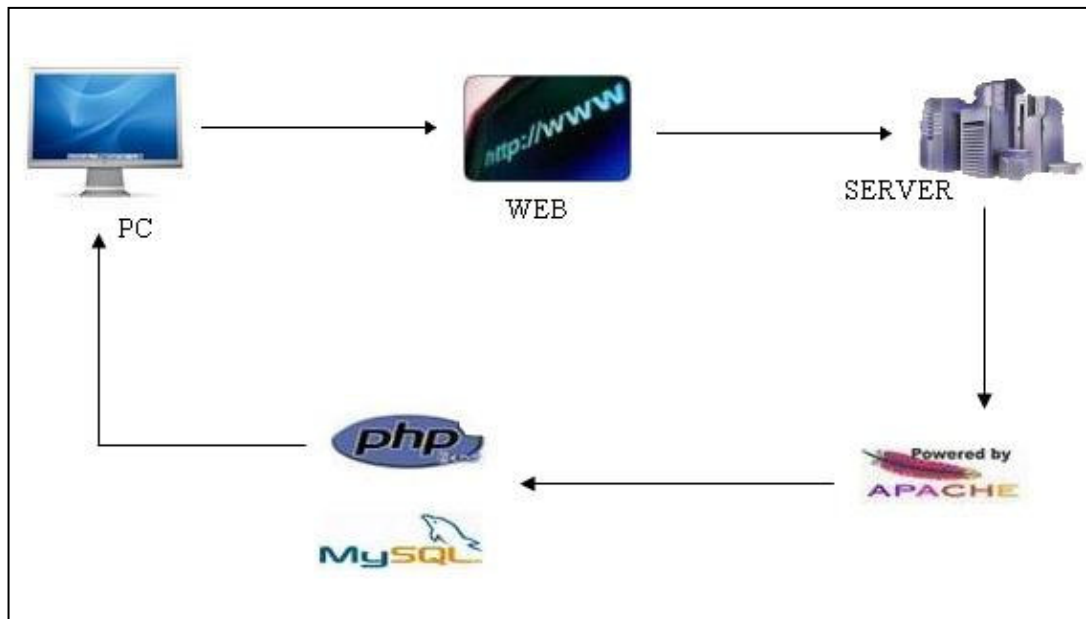
- En el caso de MySQL es una Base de Datos no muy comercial en comparación por ejemplo con el Oracle que trae un paquete de herramientas extenso para cada una de las etapas de gestión, desarrollo e implementación de un sistema de cualquier naturaleza.
- Las herramientas en mención tienen como rivales por decirlo así al gigante del software Microsoft, esto hace que muchos usuarios opten por uno u otro paquete en ocasiones llevados por el respaldo de las empresas que los representan y no por las fortalezas reales del software.
- Al momento de preparar los archivos de configuración de algunas de nuestras herramientas podemos caer en ciertas incongruencias o llegar al punto de que alguna herramienta simplemente no funcione, específicamente en el caso de Apache al configurarlo mal no permite que arranque MySQL.
- PHP-Nuke puede resultar demasiado abstracto al momento de trabajar con la interfaz de la aplicación e interactuar en las transacciones de la misma.

#### **6.5. Designación de Herramientas Software**

Se escogieron las herramientas PHP-Nuke, PHP, Apache y My-SQL por algunas razones tanto técnicas como personales.

- Las cuatro herramientas trabajan en conjunto alcanzando un alto porcentaje de compatibilidad entre sí.

- A criterio de muchos expertos el futuro del Software esta en el desarrollo en Internet e Intranet, es una estimación valedera por los resultados que se están dando, el software que aquí se esta tratando permite abrirse en ese campo.
- Son programas modernos, algunos de ellos se estudiaron en el Curso de Graduación y creí conveniente ampliarlas en esta Monografía y complementar con la aplicación práctica del Sitio Web Turístico que se analiza en el capitulo siguiente.



**Cuadro 6.1. Integración de Herramientas  
Aplicaciones Dreamweaver 2004**

## 6.6. Conclusión

La compatibilidad e integración de un grupo de herramientas de Software va en función de lo que se pretenda hacer, tomando para ello lo mejor de cada una de ellas, haciendo un análisis de sus beneficios y limitaciones, no sería factible y aún más “ético” el decidirse por el uso de tal o cual herramienta guiándonos por el gusto, inclinación o situaciones subjetivas.

“Las licencias que cubren la mayor parte del software están diseñadas para quitar la libertad de compartirlo y modificarlo. Por el contrario, las licencias de GNU pretenden garantizar la libertad de compartir y modificar software libre, para asegurar que el software es libre para todos sus usuarios. Cuando hablamos de software libre, estamos refiriéndonos a libertad, no a precio. Las Licencias Públicas Generales están diseñadas para asegurarnos de que tenga la libertad de distribuir copias de software libre (y cobrar por ese servicio si quiere), de que reciba el código fuente o que pueda conseguirlo, de que pueda modificar el software o usar fragmentos de él en nuevos programas libres, y de que sepa que puede hacer todas estas cosas.” (*Mass Ave, 199, p. 675*)



## CAPITULO 7

### PRESENTACION WEB

#### 7.1. Introducción

Al llegar a la implementación de la aplicación práctica que para el caso es el Sitio Turístico para la Parroquia El Valle, se analizaran algunos tópicos generales, importancia, componentes, quienes pueden consultar el sitio, beneficios que representa a corto y largo plazo, la información que debe contener y que en nuestro caso contiene el Portal Web.

“Una aplicación Web es un conjunto de páginas Web que interactúan entre sí, con el usuario y con diversos recursos en un servidor Web, incluidas las Bases de Datos, a su vez tienen diferentes usos tanto para los visitantes como para los ingenieros de desarrollo.” (*Sheila McGinn, 2003, p.57*)

#### 7.2. Usos comunes de las aplicaciones Web

Las aplicaciones Web pueden tener numerosos usos:

- Permitir a los usuarios localizar información de forma rápida y sencilla en un sitio Web en el que se almacena gran cantidad de contenido. Este tipo de aplicación ofrece a los visitantes la posibilidad de buscar contenido, organizarlo y navegar por él mismo, ejm: y [www.amazon.com](http://www.amazon.com).
- Recoger, guardar y analizar datos suministrados por los visitantes de los sitios. En ocasiones los datos introducidos en los formularios se enviaban como mensajes de correo electrónico para su procesamiento. Una aplicación Web permite guardar datos de los formularios directamente en una Base de Datos, además de extraer datos y crear informes para su análisis, ejm: páginas de los bancos en línea, las encuestas y los formularios con datos suministrados por el usuario.

- Actualizar sitios Web cuyo contenido cambia constantemente. Una aplicación Web evita al diseñador tener que actualizar continuamente el código del sitio. Los proveedores de contenido, como los directores de noticias, proporcionan el contenido a la aplicación Web y ésta actualiza el sitio automáticamente, ejm: [www.economist.com](http://www.economist.com) y [www.cnn.com](http://www.cnn.com).

### **7.3. Páginas Web estáticas y dinámicas**

Una página Web estática es aquella que no cambia cuando un usuario la solicita: el servidor Web envía la página al navegador Web solicitante sin modificarlo. Por el contrario el servidor modifica las páginas Web dinámicas antes de enviarlas al navegador solicitante. La naturaleza cambiante de este tipo de página es la que le da el nombre de dinámica.

#### **7.3.1. Procesamiento de páginas Web estáticas**

Un sitio Web estático consta de un conjunto de páginas y archivos HTML relacionados y alojados en un equipo que ejecuta un servidor Web. Un servidor Web es un software que suministra las páginas Web en respuesta a las peticiones de los navegadores.

La petición se genera cuando el usuario hace clic en un vínculo de una página Web, elige un marcador en un navegador o introduce un URL en el cuadro de texto dirección del navegador. El contenido final lo determina el diseñador de la página y no cambia cuando se solicita la página.

#### **7.3.1. Procesamiento de páginas Web dinámicas**

Cuando un servidor Web recibe una petición para mostrar una página Web dinámica transfiere la página a un software especial encargado de finalizar la página; este software especial se denomina servidor de aplicaciones.

El servidor de aplicaciones lee el código de la página, finaliza la página en función de las instrucciones de código y elimina el código de la página. El resultado es una página estática que el servidor de aplicaciones devuelve al servidor Web, que a su vez lo envía al navegador solicitante. Lo único que el navegador recibe cuando llega la página es código HTML puro.

#### **7.4. Errores en Sitios Web**

Los sitios Web suelen cometer errores imperdonables a la hora de llevar su contenido al público. (*Javier Cabrera, 2005, p.140*).

- El archivo flash mal utilizado suele ser pesado y consumidor de recursos. Los archivos Flash en Macintosh o PC del Diseñador pueden llegar a correr y cargar tan fluidamente que a veces asombra, pero resulta que el 90% del resto de usuarios no está equipada para soportar miles de vectores y videos en movimiento.
- Fuentes ilegibles. En ocasiones se recurre a fuentes demasiado sofisticadas, puede que el diseñador esté familiarizado pero el usuario común no las puede entender, hay que estar conciente que la mayoría de navegantes Web están acostumbrados a los tipos de letra básicos.
- Pantalla completa. Muchos de los Sitios Web que utilizan esta propiedad no son bien visto por los navegantes. Según encuestas, es más probable que un navegante cierre un WebSite que no le permite la opción de utilizar el resto de su pantalla que el visitante cierre un Banner Pop-up.
- Link a mail to... en lugares inesperados. Este es un error cometido por muchos Diseñadores. En ocasiones se olvida que la arquitectura y usabilidad es tan importante como el aspecto gráfico de un Sitio Web. Se debe especificar el link que lleva a un formulario de correo electrónico y no confundir al navegante del Web.
- Recolectar correos ilegalmente. El spam es un problema que involucra a todos y cada uno de los profesionales que realizan contenidos Web, y lamentablemente

muchos de los Sitios Web de ISPs y empresas proveedoras de casillas de correo gratuito que han ido apareciendo envían mails indeseados a los navegantes.

- Páginas pesadas y contenidos extensos. Por el afán de ofrecer cada vez más, muchos Sitios han cometido el error de ofrecer demasiado contenido a sus navegantes perdiendo así la línea de lo que en verdad quieren ofrecer o dar a conocer. Estudios demuestran que un navegante se pierde muy rápido en un Sitio Web extenso que en uno con su información concentrada y enfocada.

- Poca información del navegante. Dado los continuos problemas que se ocasionaba al navegante al pedirle demasiada información en formularios para permitirle luego acceder a las páginas se optó por no pedir más datos de los necesarios, pero se excedió en este aspecto hasta no pedir información en absoluto sobre quien es la persona que esta navegando en nuestros Sitios.

- Contactar con ¿Quiénes somos?. La finalidad de una sección de contáctenos en un WebSite es simplemente eso, un lugar donde se muestra al navegante toda la información necesaria para ponerse en contacto con la empresa. Naturalmente aquí se tendrían que colocar todos los datos de la empresa, como dirección postal, ubicación en la ciudad, teléfonos, emails, etc. Pero desgraciadamente esto no se lleva a cabo y solamente ponen a disposición del navegante una dirección email, olvidando que no solamente los "buscadores de ocio" navegan la Internet, si no también, empresarios, accionistas, socios y clientes en potencia visitan Internet día a día.

## **7.5. Presentación del Sitio**

### **7.5.1. Información del Sitio Turístico**

Para dar a conocer a la Parroquia El Valle en el ciberespacio se recopiló información a través de la Investigación de campo, fuentes bibliotecarias y consultas a personas conocedoras de este interesante sector Azuayo; se tomaron fotos actuales de muchas zonas, de las cuales se tomaron en cuenta las de mejor presentación.

En el Sitio Web se da a conocer lo siguiente:

Sitios propios del Valle, como son los Sectores de mayor renombre y lugares de importancia por su trascendencia histórica y cultural.

Proyectos que se están realizando y los que están por realizar, mismos que son realizados por la Junta Parroquial con fondos del Estado Ecuatoriano.

Noticias de la Parroquia que son de interés de la colectividad en general.

Cuenta con un Directorio en el que se da a conocer a los Profesionales en las diferentes Ciencias Sociales y Técnicas, el lugar donde residen y sus teléfonos.

En la Sección Clasificados se incorpora el ámbito y movimiento mercantil que existe, para que los Vallenenses puedan promocionar lo que desean vender y a su vez se enteren de las posibles compras que puedan hacer.

### **7.5.2. Importancia del Sitio Turístico**

Si hablamos de un mundo globalizado, los pueblos en su totalidad se deben dar a conocer con la tecnología informática. Son sus habitantes quienes deben impulsar el desarrollo de proyectos a fin de promover sus potenciales turísticos, laborales, etc. Y por otra parte comunicar lo que dentro de su jurisdicción se esta llevando a cabo.

El índice de migración es alto en la Zona del Austro, El Valle no es la excepción lamentablemente, en el momento que el Sitio Web sea colocado en Internet se logrará al menos que quienes están en el exterior estén al tanto de los avances de su lugar de origen.

### **7.5.3. Quienes pueden consultar el Sitio Web**

El Sitio puede ser consultado por cualquier persona, no se escatima ni se restringe el uso a tal o cual individuo por alguna inclinación de índole política, religiosa u otra, lo que se considera más bien es que quienes visiten el Sitio tengan las ideas básicas de lo que significa navegar en Internet, dar clic en un vínculo, moverse en una página y otra, si creen que algo no esta acorde o debería mejorar, se permite la libre crítica constructiva desde luego.

#### **7.5.4. Contenido y Administración**

El contenido del Sitio esta en una Base de Datos en My-SQL llamada valle, la información almacenada (texto y gráficos) es la que el Administrador va colocando día a día de acuerdo a las situaciones que se presenten, los usuarios pueden navegar libremente en el sitio pero solo el Administrador esta en capacidad de acceder al Sitio en el cual se hace el mantenimiento del Portal y se prueba por tanto la potencialidad Dinámica del mismo.

Al ser un Sitio Turístico, se ha concebido la idea de presentar pantallas muy decidoras por si solas de lo que se pretende mostrar, se exhibe sitios de renombre, y a su vez se permite conocer a las personas (¿Quienes somos?) que están en el día a día de un mejor lugar llamado El Valle.

En el supuesto caso de que el portal se vendiese a terceras personas o a una empresa que decida comercializarlo; se les capacitaría con lujo de detalle en cuanto a la estructura de los programas y la lógica que maneja el portal internamente, cambiaría la información para otros lugares pero la concepción del Sitio se mantiene.

#### **7.6. Conclusión**

“En el mundo del diseño Web tenemos que tener una forma demasiado impresionante para venderle al cliente la idea de tener su sitio Web de tal forma que el adore y sepa que necesita estar en la red.” (*Javier Cabrera, 2005, p.220*).

Cuando nos planteamos crear un Sitio Web debe ser con una estrategia de ventas no sólo en el sentido económico sino de promoción tan buena que logremos crear una necesidad de verla y dependiendo del caso tenerla. Se debe mostrar los beneficios y los esquemas que rompe un Sitio Web de tal forma que a pesar de que antes era algo raro o abstracto, las páginas Web de pronto se conviertan en una necesidad que es indispensable adquirir.

## **Conclusión General**

Al concluir esta Monografía cabe destacar que el aprendizaje ha sido valioso tanto en lo referente a la aplicación técnica de las herramientas de software y la metodología en la redacción de este trabajo de graduación que ha permitido plasmar conocimientos teóricos y prácticos.

En lo concerniente al Sitio Web desarrollado se puede decir que hay muchos de este tipo o similares pero en este caso la información a presentar es única, ya que por primera vez en la historia de la Parroquia El Valle se va a tener un Portal Turístico exclusivo pero abierto a ser consultado por quienes deseen.

Esta Monografía y la aplicación contenida es parte del desarrollo moderno de aplicaciones informáticas, es susceptible a cambios y a la trascendencia a nuevas versiones con nuevos métodos de análisis y al manejo de nuevas herramientas de desarrollo; a su vez las personas que quisieren hacer uso de esta Monografía están en pleno derecho de hacerlo, como constancia quedan copias escritas y electrónicas en la Universidad del Azuay.

## REFERENCIAS

### Referencias a páginas en Internet

Albanet.com. *Ciberespacio, Historia del Internet* [en línea]. 1ra edición. México DF: Albanet, 2003, 2003, [ref. de 10 de enero de 2006], disponible en <http://www.albanet.com.mx/articulos/HISTORIA.htm>

Programatium.com. *Capacitación en PHP-NUKE* [en línea]. 1ra edición. Buenos Aires, 2004, 2004, [ref. de 20 de diciembre de 2005], disponible en <http://www.programatium.com>

My-SQL. *Manual de My-SQL* [en línea]. 1ra edición. Madrid, 2002, 2002, [ref. de 10 de enero de 2006], disponible en <http://www.mysql-hispano.org>

Desarrollo Web. *Manual de PHP* [en línea]. 1ra edición. Madrid, 2000, 2000, [ref. de 5 de enero de 2006], disponible en <http://www.desarrolloWeb.com>

NSFNET. *Servidor Apache* [en línea]. 1ra edición. Madrid, 2001, 2001, [ref. de 6 de febrero de 2006], disponible en <http://www.nsfnet.com>



## BIBLIOGRAFÍA

AVE, Mass. *Licencia Pública GNU*. 1ra edición. Free Software Foundation, 1991. 800p.

BAKKEN, Stig Sæther. *Manual de PHP*. 2da edición. Ratmat Gan: Zend Technologies, 1997. 2000p.

CABRERA, Javier. *Maestros del Web*. 1ra. Edición. Madrid: MDW, 2005, 600p.

GRAY, Neil. *WEB SERVER PROGRAMING*. 1ra edición. Sydney: A.B, 1999. 500p. ISBN 0 470 85097 3

MALDONADO, Andrés. *Tecnologías de Información Turística*. 1ra edición. Málaga: Teatinos, 2002. 400p.

MCGINN, Shelia. *Dreamweaver MX primeros pasos*. 1ra edición. San Francisco: Macromedia Inc, 2003. 155p.

MERCHAN, Oswaldo. *Fundamentos de Bases de Datos*. 1ra edición. Cuenca: UDA, 2003. 117p.

MOREA, Lucas. *Ingeniería del Software*. 1ra edición. California: Sinexi S.A., 1997. 600p.

MARROBA NETWORKS. Foro PHP Nuke.

<http://miarroba.com/foros/ver.php?foroid=883865&temaid=4569032>

[consulta 5 de enero del 2006]

## **GLOSARIO**

### **Active X**

Lenguaje desarrollado por Microsoft con el fin de elaborar aplicaciones exportables a la red las cuales deben ser capaces de operar sobre cualquier plataforma a través de navegadores WWW de forma que le da dinamismo a las páginas Web.

### **Adserver**

Sistema de control de inserciones publicitarias "en línea" cuyos softwares permiten rotar distintas campañas por emplazamiento de forma que la información de contactos conseguida es recopilada así como los "click-thru" generados.

### **ADSL (Línea de Suscripción Asimétrica Digital)**

Se refiere a una tecnología para mejorar el ancho de banda de los hilos del cableado telefónico convencional que transporta hasta 16 Mbps (megabits por segundo) gracias a una serie de métodos de compresión.

### **ANSI**

American National Standards Institute - Instituto Nacional de Normas de Estados Unidos.

### **ANSI Lumen**

Norma definida por el ANSI para medir el brillo de un monitor. La medida representa el valor medio de 9 puntos en la imagen proyectada en la pantalla.

### **Apache**

Servidor HTTP de dominio público el cual está basado en el sistema operativo Linux. Fue desarrollado en 1995 y actualmente es uno de los servidores HTTP más utilizados en la red. <http://www.apache.org/>

### **Aplicación**

Programa que lleva a cabo una función específica para un usuario en Internet tales como WWW, FTP, correo electrónico y Telnet.

## **Archivo**

Unidad significativa de información la cual puede ser manipulada por el sistema operativo de un ordenador debido a que tiene una identificación única formada por un "nombre" y un "apellido". El nombre suele ser de libre elección del usuario y el apellido debe identificar el contenido o el tipo de archivo. A manera de información, los archivos word tienen el apellido .doc; los de excel tienen .xls; los ejecutables .exe, los de texto .txt y así sucesivamente.

## **ARPANet**

Advanced Research Projects Agency Network -- El precursor del Internet. Desarrollado a finales de los 60's y principios de los 70's por el Departamento de Defensa de los Estados Unidos como un experimento de una red de área amplia que sobreviviría a una guerra nuclear.

## **ASCII**

American Standard Code for Information Interchange. Es de facto el estándar del World Wide Web para el código utilizado por computadoras para representar todas las letras (mayúsculas, minúsculas, letras latinas, números, signos de puntuación, etc.). El código estándar ASCII es de 128 letras representadas por un dígito binario de 7 posiciones (7 bits), de 0000000 a 1111111.

## **Bit (Dígito Binario)**

Unidad mínima de almacenamiento de la información cuyo valor puede ser 0 ó 1; o bien verdadero o falso.

## **Byte**

Conjunto de 8 bits el cual suele representar un valor asignado a un carácter.

## **C/ C++**

Lenguajes de programación (orientado a objetos en el caso de C++) utilizados en el WWW a través de un CGI, principalmente para realizar consultas a bases de datos tipo Oracle, SQL-Server, SyBase, etc; o a herramientas locales como WAIS. Generalmente el servidor donde se encuentra el programa funciona en ambiente UNIX.

### **CGI 'Common Gateway Interface'**

Es un método para la transmisión de información hacia un compilador instalado en el servidor. Su función principal es la de añadir una mayor interacción a los documentos Web que por medio del HTML se presentan de forma estática.

### **Ciberespacio**

Término concebido por el escritor William Gibson en su novela de ciencia ficción "Neuromancer" (1984) con el propósito de describir un mundo de redes de información. Actualmente es utilizado para referirse al conjunto de información digital y a la comunicación que se realiza a través de las redes, un espacio en el cual casi todo lo que contiene información; o puede transmitirla, debe ser incluido.

### **Cibernética**

Término acuñado por un grupo de científicos dirigidos por Norbert Wiener y popularizado por su libro "Cybernetics or Control and Communication in the Animal and the Machine" de 1948. Viene del griego "cibernetes" (timonel o piloto) y es la ciencia o estudio de los mecanismos de control o regulación de los sistemas humanos y mecánicos, incluyendo las computadoras.

### **Clic**

Pulsar un determinado comando de un mouse o ratón una vez colocado el cursor del mismo sobre una determinada área de la pantalla con el fin de dar una instrucción a la computadora.

### **Código Fuente**

(Source Code) - Conjunto de instrucciones que componen el programa informático mediante el cual se elabora un sitio Web. Estos programas se escriben en determinados lenguajes como, por ejemplo, el HTML.

### **Correo Electrónico (e-mail)**

El e-mail, o correo electrónico, es uno de los servicios más usados en Internet, que permite el intercambio de mensajes entre las personas conectadas a la red de manera similar al correo tradicional. Utiliza el protocolo de comunicación TCP/IP. Básicamente es un servicio que nos permite enviar mensajes a otras personas de una

forma rápida, barata y cómoda. Gracias a la aparición de aplicaciones de correo electrónico para PC's y Mac's es posible intercambiar no sólo mensajes personales, sino también todo tipo de archivos, lo que facilita el trabajo en grupo a distancia. Es uno de los medios de comunicación de más rápido crecimiento en la historia de la humanidad.

### **CPU**

De las siglas en inglés Central Processing Unit (Unidad Central de Procesos) -- Es la parte que constituye el cerebro de cualquier computadora, es el encargado de realizar y dirigir todas las sus funciones. Contiene memoria interna, la unidad aritmética / lógica. Realiza el procesamiento de los datos y además el control de las funciones del resto de los componentes de la computadora. Gobierna el sistema y dicta la velocidad de trabajo del mismo.

Existen diferentes tipos de CPU, por ejemplo, los CPU de la familia 8086: 80286, 80386, 80486, Pentium.

### **Dato**

Unidad mínima que compone cualquier información.

### **Desarrollador de Web**

Web developer - Persona o empresa responsable de la programación de un sitio Web así la cual incluye, si se da el caso, plataformas de comercio electrónico.

### **Descarga (Download)**

Proceso en el cual la información es transferida desde un servidor de información al propio ordenador personal.

### **Descomprimir**

Acción de desempaquetar uno o más archivos que anteriormente han sido empaquetados, y habitualmente también comprimidos, en un solo archivo, con objeto de que ocupen menos espacio en disco y se precise menos tiempo para enviarlos por la red.

**Descriptación/ Descifrado**

Recuperación del contenido real de una información previamente cifrada.

**Dimensión Multi-Usuario (MUD)**

Entorno de realidad virtual, basado en texto o gráficos, en el cual los usuarios pueden conversar o interpretar diferentes roles como diversión. Los usuarios entran en el juego desde cualquier parte de Internet y solo tienen que conectarse por medio de la red al sistema donde se guarda el juego para posteriormente interactuar de manera recíproca uno con otro.

**Directorio**

Espacio lógico de una estructura jerárquica en forma de árbol el cual contiene la información almacenada en un ordenador, habitualmente contenida en archivos y es identificado mediante un nombre (ej. "Mis documentos"). Índice organizado por categorías de forma que cada categoría consta de un determinado número de páginas el cual es actualizado periódicamente (ej. Yahoo).

**Directorio Web**

Las páginas que se incluyen en la base de datos del directorio son previamente revisadas por humanos (no es automatizado como los crawlers o arañas). No se agrega la página completa, sino únicamente algunos datos tales como el título, la URL y un breve comentario redactado especialmente que explique el contenido, y se la ubica en una categoría. Un ejemplo es <http://www.yahoo.com/>

**En Línea (On Line)**

Condición de estar conectado a una red.

**Encriptación (Cifrado)**

Tratamiento de un conjunto de datos, contenidos o no en un paquete, a fin de impedir que nadie excepto el destinatario de los mismos pueda leerlos. Hay muchos tipos de cifrado de datos, que constituyen la base de la seguridad de la red.

## **FAQ**

Frequently Asked Questions (Preguntas Frecuentes) - FAQs son documentos que enlistan y responden las preguntas más comunes de un tema en particular. Existen cientos o miles de FAQs de miles de distintos temas.

## **FTP -- File Transfer Protocol**

Protocolo de transferencia de archivos. Se usan programas para FTP como son CuteFTP o LeapFTP para Windows, por ejemplo, que permiten la conexión entre dos computadoras, usando por lo general el puerto 21 para conectarse (aunque se pueden usar otros puertos).

## **Fundación para el Software Libre (FSF)**

Entidad que busca eliminar las restricciones de uso, copia, modificación y distribución del software. Apoya el desarrollo de sistemas operativos (Linux), compilador GNU C (GCC), PERL, etc.

## **GNU**

Proyecto creado en 1984 con el fin de desarrollar un sistema operativo tipo Unix según la filosofía del "software libre".

## **Google**

Buscador de Internet, uno de los más populares, y nuestro favorito. Introduce páginas Web en su base de datos por medio de robots o crawlers (googlebot).  
<http://www.google.com/>

## **Gopher**

Sistema de búsqueda de información en Internet a base de menús, que permite navegar por directorios FTP y bases de datos.

## **GPL**

General Public License -- Licencia de regulación de los derechos de autor de los programas de software libre (free software) la cual es promovida por la Free Software Foundation (FSF) en el marco de la iniciativa GNU. Permite la distribución

de copias de programas (e incluso cobrar por ello), así como modificar el código fuente de los mismos o utilizarlo en otros programas.

### **HDSL**

Línea Digital de Abonado de Alta Velocidad. Sistema de transmisión de datos de alta velocidad que utiliza dos pares trenzados.

### **Hacker**

Persona que tiene un conocimiento profundo acerca del funcionamiento de redes de forma que puede advertir los errores y fallas de seguridad del mismo. Al igual que un cracker busca acceder por diversas vías a los sistemas informáticos pero con fines de protagonismo.

### **Hipermedia**

Páginas Web que integran información en distintos tipos de formatos texto, gráficos, sonidos y vídeo; actualmente es un recurso ampliamente explotado en el WWW.

### **Hipertexto**

Cualquier documento que contiene vínculos con otros documentos de forma que al seleccionar un vínculo se despliega automáticamente el segundo documento.

### **Hipervínculo**

Vínculo existente en un documento hipertexto que apunta o enlaza a otro documento que puede ser o no otro documento hipertexto.

### **Hoja de Estilo en Cascada (CSS)**

Conjunto de instrucciones HTML que definen la apariencia de uno o más elementos de un conjunto de páginas Web con el objetivo de uniformizar su diseño.

### **HTML**

Acrónimo de HyperText Markup Language, lenguaje de marcas de hipertexto. Formato estándar de los documentos que circulan en la World Wide Web (WWW); se utiliza desde 1989. Los documentos HTML contienen dos tipos de información: la



que se muestra en la pantalla (texto, imágenes...) y los códigos (tags o etiquetas), transparentes al usuario, que indican cómo se debe mostrar esa información. Para crear un documento HTML, basta con disponer de un editor de código ASCII y teclear el texto y las etiquetas que sean precisas.

### **HTTP (Hypertext Transfer Protocol)**

HTTP es un protocolo con la ligereza y velocidad necesaria para distribuir y manejar sistemas de información hipermedia.

### **HTTPS**

Creado por Netscape Communications Corporation para designar documentos que llegan desde un servidor WWW seguro. Esta seguridad es dada por el protocolo SSL (Secure Sockets Layer) basado en la tecnología de encriptación y autenticación desarrollada por la RSA Data Security Inc.

### **Interfaz (Interface)**

Zona de contacto o conexión entre dos componentes de "hardware"; entre dos aplicaciones; o entre un usuario y una aplicación. Apariencia externa de una aplicación informática.

### **Interfaz de Programación de Aplicaciones**

API (del inglés Application Programming Interface - Interface de Programación de Aplicaciones, interfaz de programación de la aplicación) es un conjunto de especificaciones de comunicación entre componentes software que definen cómo se solicita un servicio desde un programa.

### **Interfaz Gráfica de Usuario (GUI)**

Componente de una aplicación informática que el usuario visualiza y a través de la cual opera con ella. Está formada por ventanas, botones, menús e iconos, entre otros elementos.

**Internet**

Sistema que aglutina las redes de datos de todo mundo, uniendo miles de ellas mediante el protocolo TCP/IP. El mayor conjunto que existe de información, personas, ordenadores y software funcionando de forma cooperativa.

**Internet Explorer (IE)**

Navegador Web gratuito creado por Microsoft en el año 1995 inicialmente para el sistema operativo Windows y posteriormente para Apple Macintosh.

**Key**

Serie de signos previamente convenidos que sirven como clave o fórmula para transmitir mensaje secretos o privados.

**Keyword (clave de búsqueda, palabra clave)**

Indispensables cuando se busca una información dentro de algún buscador o cuando queremos registrar una página en uno de ellos.

**Kilobyte**

Unidad de medida de la capacidad de transmisión de una línea de telecomunicación equivalente a mil bytes aunque actualmente es usado como 1024 (dos elevado a la 10) bytes.

**LAN (Local Area Network)**

Red de área local. Red de computadoras personales ubicadas dentro de un área geográfica limitada que se compone de servidores, estaciones de trabajo, sistemas operativos de redes y un enlace encargado de distribuir las comunicaciones.

**Libro Electrónico (e-book)**

Libro en formato digital que, en algunos casos requiere programas específicos para su lectura. Suele aprovechar las posibilidades del hipertexto, de los hiperenlaces y de multimedia, y puede estar disponible en la red.

### **Línea Conmutada (Dial Up)**

Conexión temporal que se establece usando un emulador de terminal y un módem; en oposición a conexión dedicada o permanente, la cual es establecida entre ordenadores por línea telefónica normal y realiza una conexión de datos a través de una línea telefónica.

### **Línea Dedicada**

Línea privada que se utiliza para conectar redes de área local de tamaño moderado a un proveedor de servicios de Internet y se caracteriza por ser una conexión permanente.

### **Linux**

Versión de libre distribución del sistema operativo UNIX el cual tiene todas las características que se pueden esperar de un moderno y flexible UNIX. Incluye multitarea real, memoria virtual, librerías compartidas, dirección y manejo propio de memoria y TCP/IP.

### **Microprocesador (Chip)**

Circuito integrado en un soporte de silicio el cual está formado por transistores y otros elementos electrónicos miniaturizados. Es uno de los elementos esenciales de un ordenador.

### **Microsoft**

Compañía creadora de los sistemas operativos Windows 95, 98, NT, 2000, XP; de los controles Active X, y del navegador IE de WWW entre otros recursos. Fundado por Bill Gates. <http://www.microsoft.com/>

### **MySQL**

My SQL es uno de los Sistemas Gestores de Bases de Datos más populares. Lo mantiene la empresa sueca MySql AB bajo la licencia GPL (GNU Public License); puede utilizarse gratuitamente y su código fuente está disponible.

**Navegador (Browser)**

Aplicación para visualizar todo tipo de información y navegar por el ciberespacio que cuentan con funcionalidades plenamente multimedia. Como ejemplo de navegadores tenemos Internet Explorer y Netscape. Éstos programas pueden también actualizarse a sus últimas versiones de forma gratuita.

**Off line**

Lo opuesto a on line, fuera de conexión.

**On line**

En línea o en tiempo real. Procesamiento de datos en el momento en que se desarrolla una acción (como obtención de señales, comunicación por módem, etc.). Significa que un programa adquiere y/o calcula datos y muestra los resultados en forma simultánea en valores numéricos y/o gráficos y/o sonidos.

**Open Source**

Es el término por el que se conoce al software distribuido y desarrollado en forma libre. Este término empezó a utilizarse en 1998 por algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (free software).

**Operador del Sistema**

Persona responsable del funcionamiento de un sistema o de una red, comúnmente denominado Sysop.

**Página Web**

Resultado en hipertexto o hipermedia que proporciona un navegador del WWW después de obtener la información solicitada. Su contenido puede ir desde un texto corto a un voluminoso conjunto de textos, gráficos estáticos o en movimiento, sonido, etc. Algunas veces el citado término es utilizado incorrectamente en orden de designar el contenido global de un sitio Web, cuando en ese caso debería decirse "sitio Web".

**PHP (Hypertext Preprocessor)**

Lenguaje de script diseñado para la creación de páginas Web activas (similares a ".ASP" en el mundo MS Windows), muy popular en entornos Unix, aunque existe también versión para sistemas Microsoft. Concebido en el tercer trimestre de 1994 por Rasmus Lerdorf, es usado principalmente para la programación de CGI's para páginas Web, destaca por su capacidad de ser embebido en el código HTML.

**Puerto**

Número que aparece tras un nombre de dominio en una URL. Dicho número va precedido del signo (dos puntos). Canal de entrada/salida de una computadora.

**Red (Network)**

Sistema de comunicación de datos que conecta entre sí sistemas informáticos situados en lugares más o menos próximos. Puede estar compuesta por diferentes combinaciones de diversos tipos de redes.

**Software**

Conjunto de programas, documentos, procesamientos y rutinas asociadas con la operación de un sistema de computadoras, es decir, la parte intangible o lógica de una computadora.

**Software Libre**

Programas desarrollados y distribuidos según la filosofía de dar al usuario la libertad de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar dichos programa (Linux es un ejemplo de esta filosofía). El software libre no es siempre software gratuito (equivocación bastante habitual que tiene su origen en que la palabra inglesa free significa tanto "libre" como "gratuito").

**URL**

Acrónimo de Uniform Resource Locator. Localizador Uniforme de Recurso. Es el sistema de direcciones en Internet. El modo estándar de escribir la dirección de un sitio específico o parte de una información en el Web.

**Vínculo (Link)**

Apuntadores hipertexto que sirven para saltar de una información a otra, o de un servidor a otro, cuando se navega por Internet; o bien la acción de realizar dicho salto.

**Vínculos Recíprocos**

Enlaces hipertexto entre dos sitios Web establecidos ya sea por razones de cortesía o por razones contractuales.

**Virtual**

Término de frecuente utilización en el mundo de las tecnologías de la información y de las comunicaciones el cual designa dispositivos o funciones simulados.

**Virus**

Programa que se duplica a sí mismo en un sistema informático incorporándose a otros programas que son utilizados por varios sistemas.

**Webmail**

Servicio que permite gestionar el correo electrónico desde un sitio Web el cual es de gran utilidad para personas que tienen que desplazarse con frecuencia y lo ofrecen habitualmente los proveedores de acceso a Internet. Entre los más utilizados están [www.hotmail.com](http://www.hotmail.com) y [www.yahoo.com](http://www.yahoo.com)

**Web master**

Administrador de Web; persona responsable de la gestión y mantenimiento de un servidor Web, principalmente desde el punto de vista técnico; por lo que no debe ser confundido con un editor de Web.

**Windows**

Sistema operativo desarrollado por la empresa Microsoft cuyas diversas versiones (3.1, 95, 98, NT, 2000, XP, ME) dominan de forma abrumadora el mercado de las

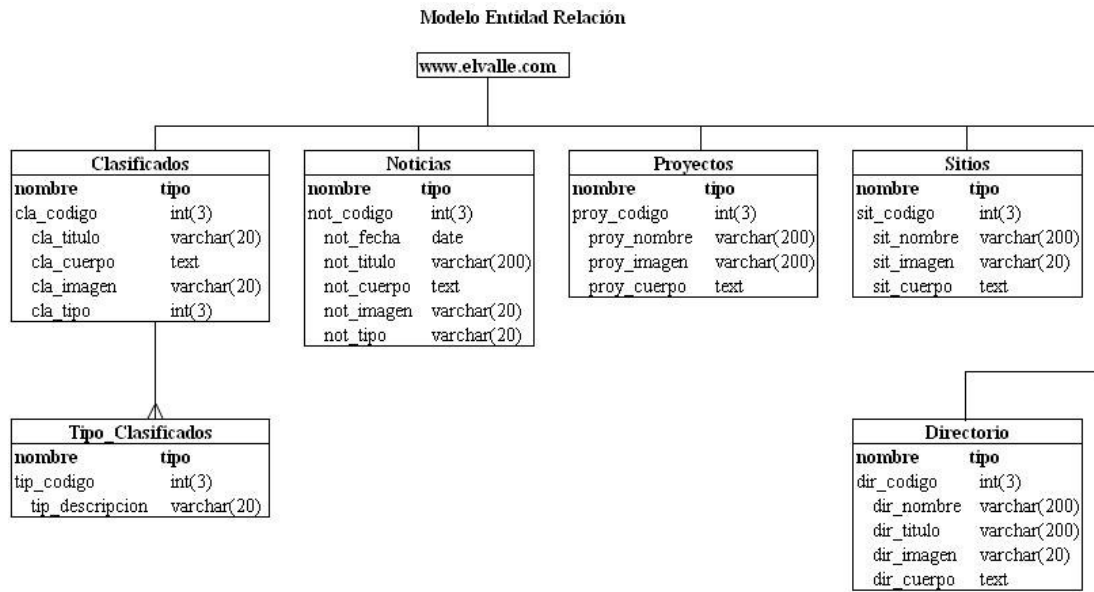
computadoras personales, aunque no se puede decir lo mismo del mercado de redes corporativas.

## **WWW**

Mecanismo proveedor de información electrónica para usuarios conectados a Internet. El acceso a cada sitio Web se canaliza a través del URL o identificador único de cada página de contenidos. Este sistema permite a los usuarios el acceso a una gran cantidad de información: leer publicaciones periódicas, buscar referencias en bibliotecas, realizar paseos virtuales por pinacotecas, compras electrónicas o audiciones de conciertos, buscar trabajo y otras muchas funciones.

# ANEXOS

## Modelo Entidad – Relación





## DICCIONARIO DE DATOS

### Tabla Clasificados

**cla\_codigo**

Código de Clasificado, único y autoincrementable.

**cla\_titulo**

Título del Clasificado.

**cla\_cuerpo**

Cuerpo a detalle explicando lo que se quiere comprar o vender.

**cla\_imagen**

Imagen descriptiva que representa el clasificado.

**cla\_tipo**

Se relaciona con el tipo de clasificado.

### Tabla Tipo\_Clasificados

**tip\_codigo**

Código del Tipo de Clasificado, único y autoincrementable.

**tip\_descripcion**

Nombre del clasificado, por ejemplo autos, casas, terrenos, etc.

### Tabla Noticias

**not\_codigo**

Código de Noticia, único y autoincrementable.

**not\_fecha**

Fecha en la cual se sube al servidor la noticia, se graba del sistema automáticamente.

**not\_titulo**

Título de la Noticia.

**not\_cuerpo**

Cuerpo a detalle explicando la Noticia.

**not\_imagen**

Imagen representativa de la Noticia.

**not\_tipo**

Tipo de Noticia: Titular o Secundaria.

**Tabla Proyecto****proy\_codigo**

Código de Proyecto, único y autoincrementable.

**proy\_nombre**

Nombre del Proyecto.

**proy\_imagen**

Imagen representativa del Proyecto en ejecución o a ejecutar

**proy\_cuerpo**

Cuerpo a detalle explicando de que se trata el Proyecto.

**Tabla Sitios****sit\_codigo**

Código del sitio, único y autoincrementable.

**sit\_nombre**

Nombre del Sitio.

**sit\_imagen**

Imagen representativa del Sitio en mención.

**sit\_cuerpo**

Cuerpo a detalle explicando el Sitio, sus cualidades, etc.

**Tabla Directorio****dir\_codigo**

Código de directorio, único y autoincrementable.

**dir\_nombre**

Nombre de la entidad, puede ser un Profesional o una Institución.

**dir\_titulo**

Título Profesional de la Persona o actividad de la entidad.

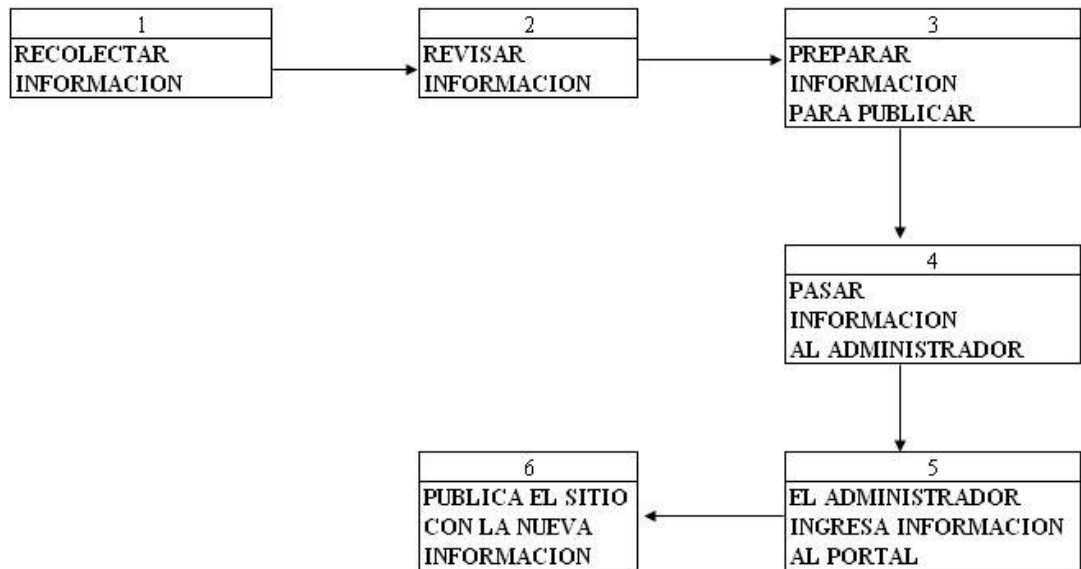
**dir\_imagen**

Imagen representativa del Profesional o entidad.

**dir\_cuerpo**

Contiene la información a detalle de la entidad, dirección de domicilio, dirección de oficina o empresa, números de teléfonos, correos electrónicos, referencias, etc.

## MODELO DE FLUJO DE INFORMACIÓN



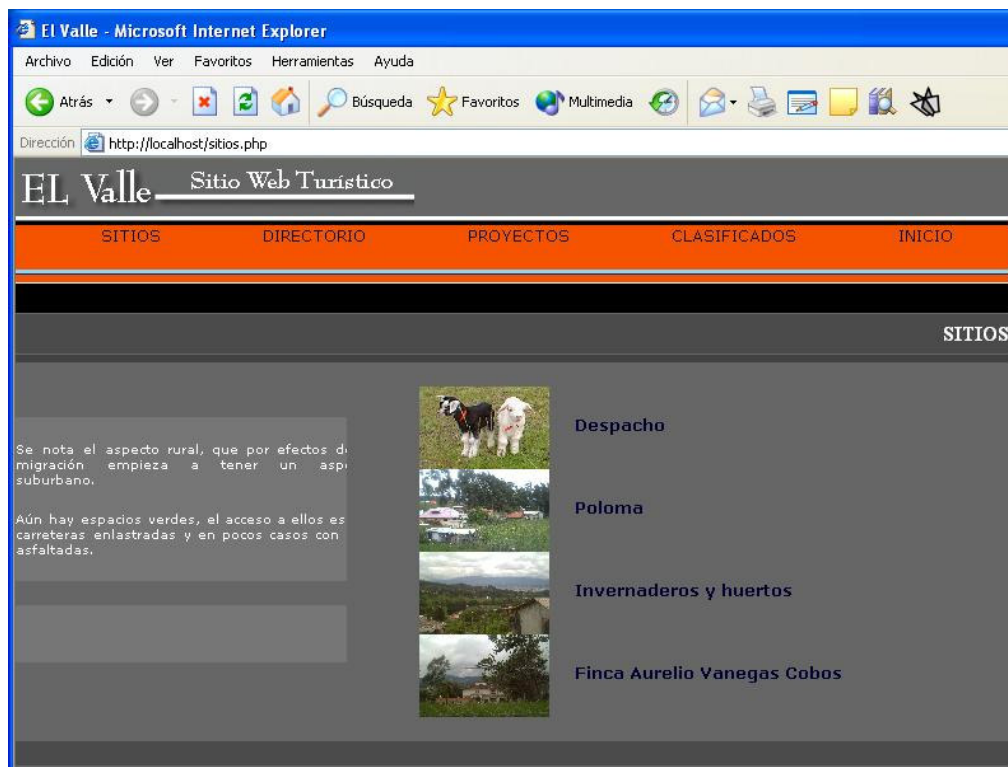
# PANTALLAS DEL SITIO TURÍSTICO

## Pantalla Principal



Esta es la pantalla principal del Proyecto, se toma como imágenes de presentación principal el frente de la Iglesia Católica y el monumento al Dr. Gabriel García Moreno que esta en el parque central de la parroquia. En la parte inferior de la pantalla están las noticias, mismas que el Administrador va cambiando de acuerdo a los acontecimientos que se presenten.

## Sitios

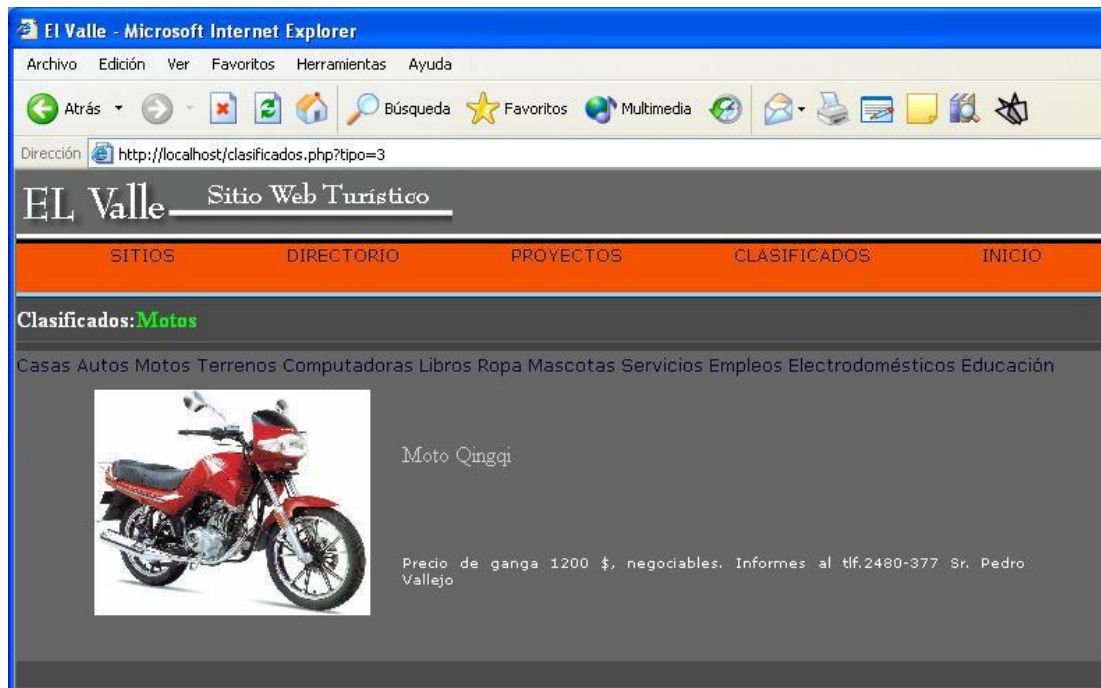


En la pagina de Sitios tenemos el nombre del lugar y una foto representativa, al dar clic en el nombre del Sitio nos lleva a una página en la cual se amplía la foto y a su vez se presenta la información respectiva.

### Enlace al dar clic en un sitio específico

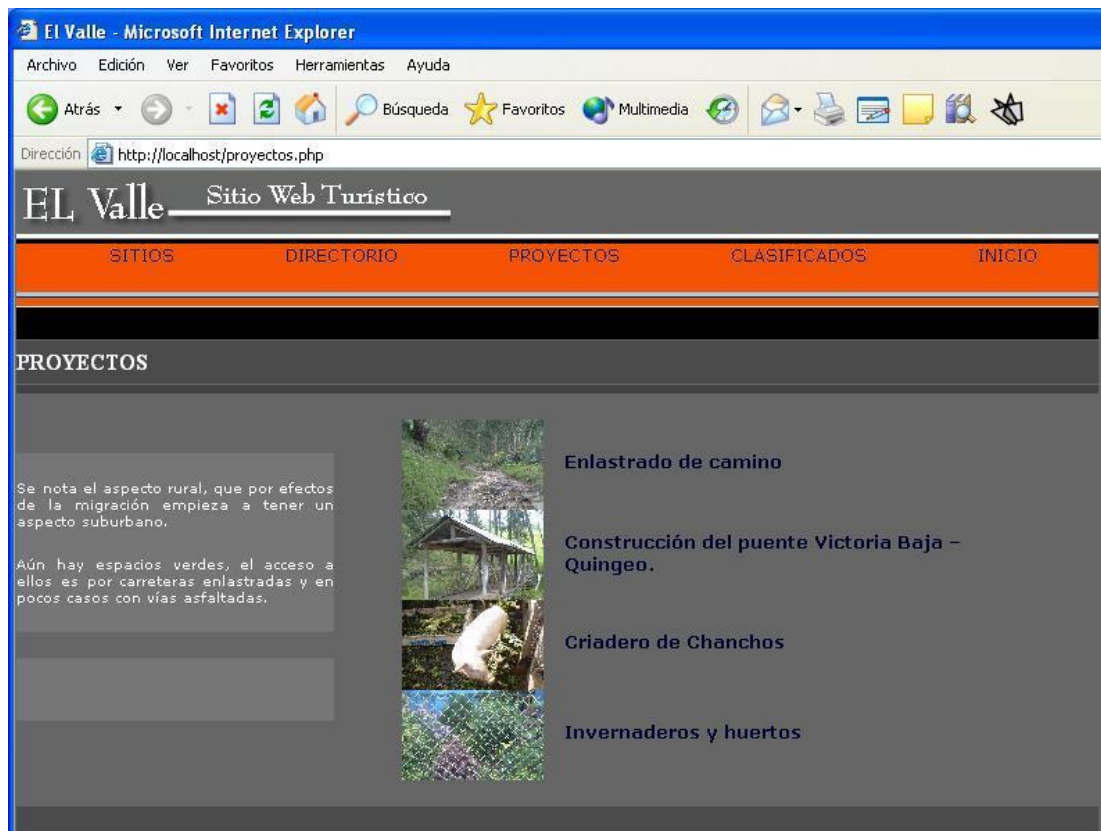


## Clasificados



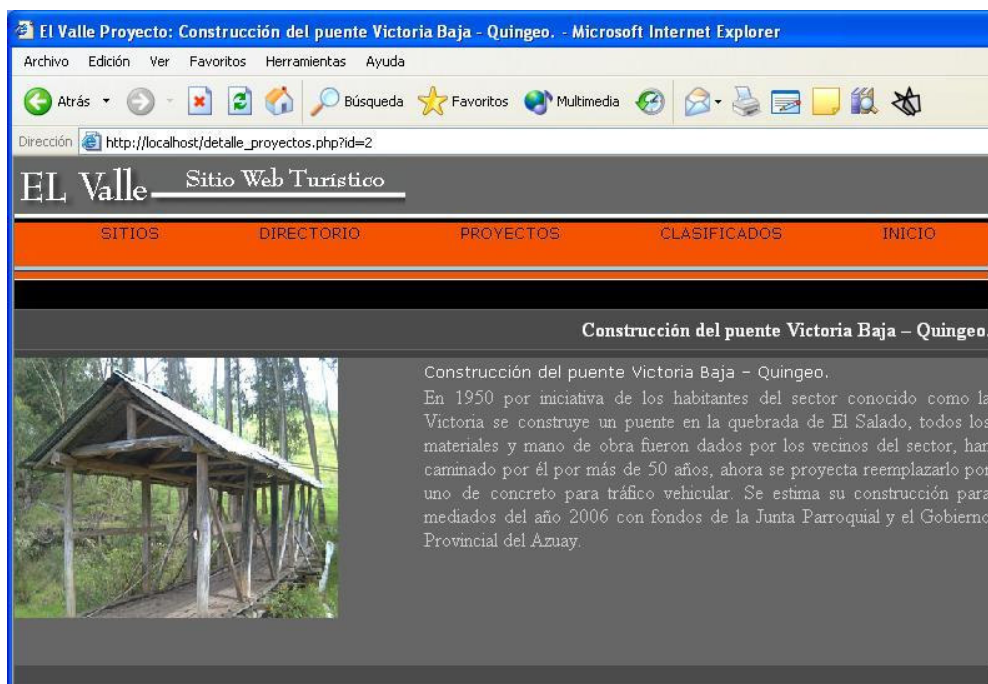
En la pagina de Clasificados tenemos el listado de clasificados, al dar clic en un clasificado nos exhibe todos los ítems correspondientes a esa categoría, de cada ítem nos muestra una foto, nombre del ítem e información respectiva.

## Proyecto



En la página de Proyectos tenemos el nombre del proyecto y una foto representativa, al dar clic en el nombre del Proyecto nos lleva a una página en la cual se amplía la foto y a su vez se presenta la información respectiva.

### Enlace al dar clic en un proyecto específico



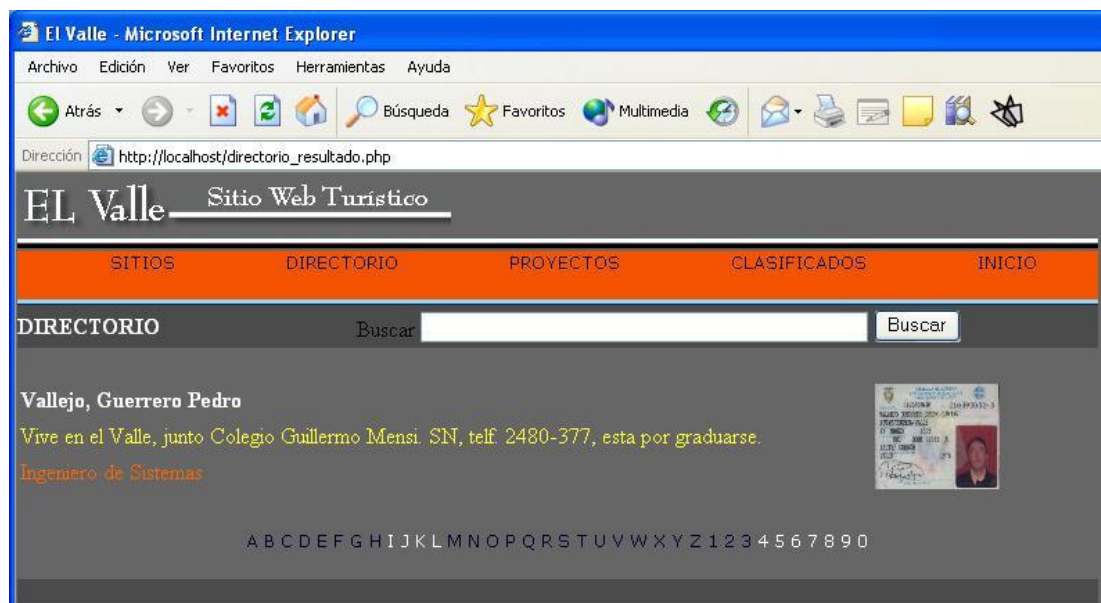


## Directorio

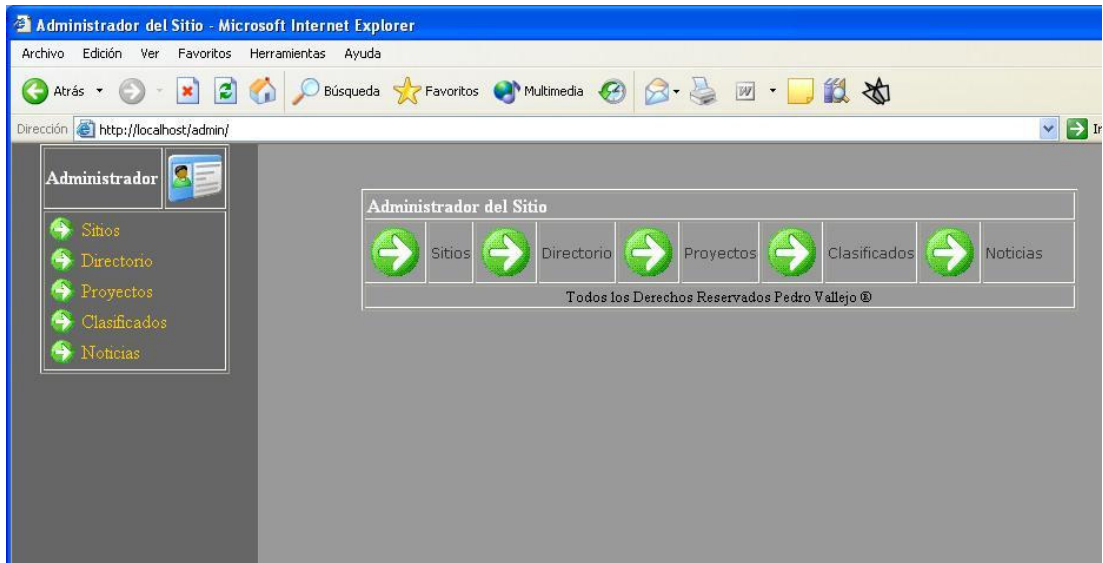


Esta página contiene información de los Profesionales o Entidades que prestan algún tipo de servicio en El Valle, a su vez tiene implementado un buscador en el cual se digita el nombre del Profesional o la Profesión.

### Página que se exhibe al dar un criterio de búsqueda



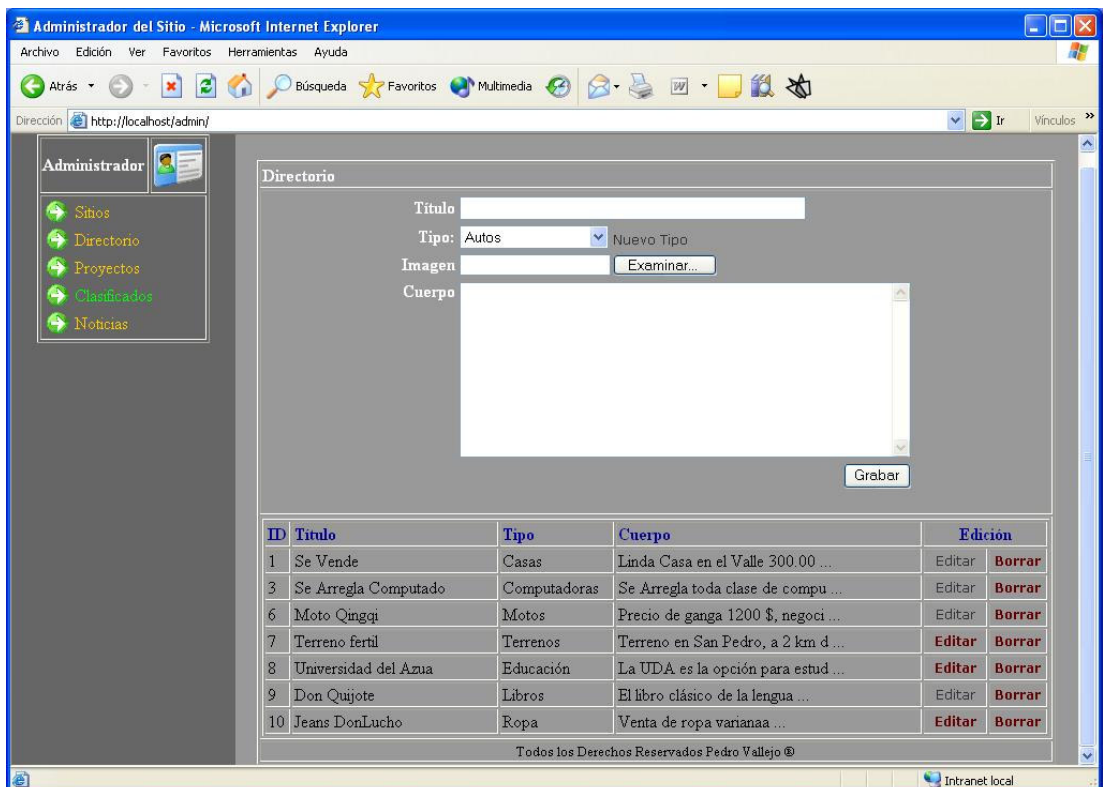
## Administrador del Sitio



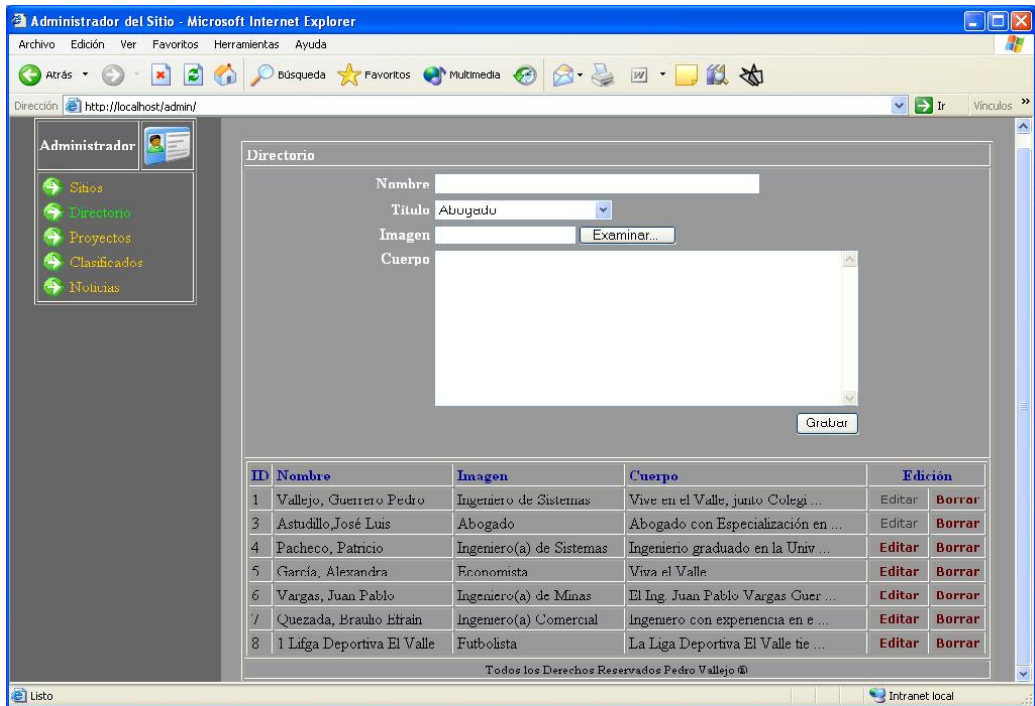
El Administrador es la única persona que puede modifica el contenido de las páginas ya sea Sitios, Directorio, Proyectos, Clasificados y Noticias.

En cada página tiene la opción de cambiar el nombre, la imagen o el contenido con el vínculo **editar** o eliminarlo con el vínculo **borrar**.

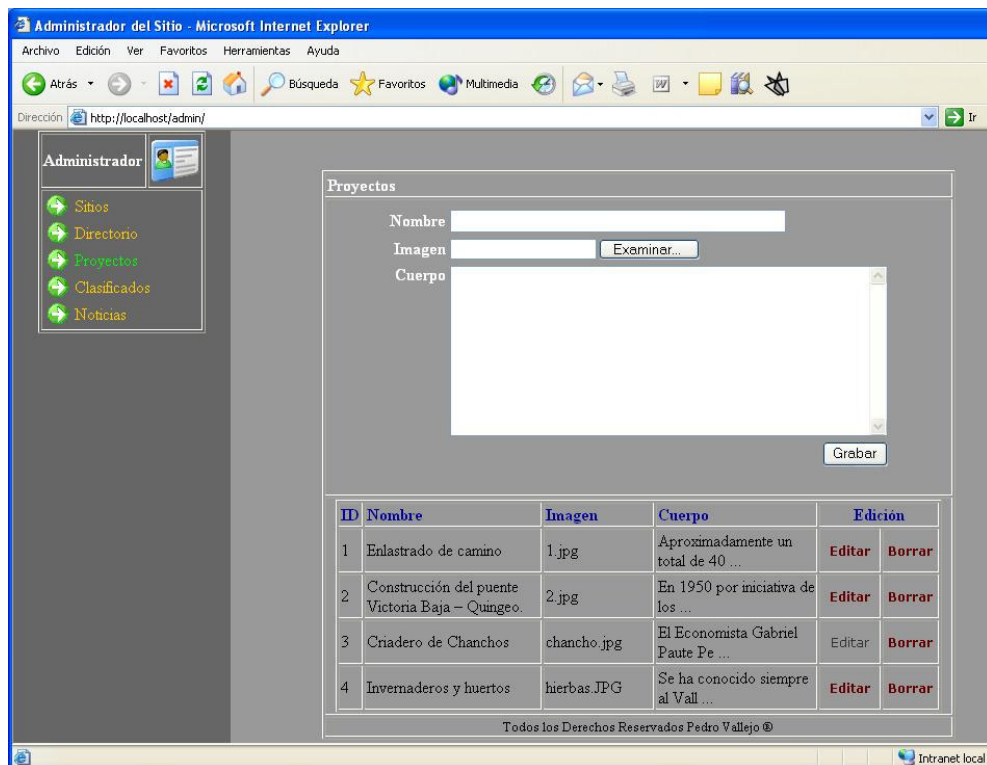
## Administrador de Clasificados



## Administrador de Directorio



## Administrador de Proyectos



## Administrador de Noticias

Administrador del Sitio - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Dirección <http://localhost/admin/> Ir Vinculos >>

**Administrador**

- Sitios
- Directorio
- Proyectos
- Clasificados
- Noticias

### Noticias

Titulo

Imagen

Cuerpo

Tipo  Titular  Secundaria

ID	Titulo	Tipo	Cuerpo	Edición
1	Construcción de Biblioteca	Titular	Para el día 30 de marzo se tie ...	Editar <b>Borrar</b>
5	Asfalto El Valle – Zhidmad	Secundaria	Con el apoyo del Consejo Provi ...	<b>Editar</b> <b>Borrar</b>

Todos los Derechos Reservados Pedro Vallejo ©

Intranet local

