

Universidad Nacional Autónoma de México

Facultad de Estudios Profesionales

Aragón



UNIVERSIDAD NACIONAL
AVENIDA DE
MEXICO

INFORME DE UN SITIO WEB BASADO EN EL DIPLOMADO
DE DESARROLLO E IMPLEMENTACIÓN DE SOFTWARE
LIBRE LINUX

T R A B A J O E S C R I T O
EN LA MODALIDAD DE SEMINARIOS
Y CURSOS DE ACTUALIZACIÓN Y
CAPACITACIÓN PROFESIONAL
QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACIÓN
P R E S E N T A:

GUILLERMO TENORIO RUIZ

Asesora: Ing. Silvia Vega Muytoy

México, 2006





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Agradecimientos

A mis Padres, Guillermo y Bertha

Por ser los amigos incondicionales que me apoyaron en todo momento de mi vida, con sus consejos, desvelos, experiencia, temple y actitud ante las metas.

Mi vida, mis logros, mis sueños son un reflejo de ustedes.

A Xochítl, Maydely y Danna

Por que con su respaldo y comprensión me enseñaron que la familia es mi mas fuerte soporte, ahora se que estando juntos llegaremos a donde sea.

A Iveth

Por convertir estas nuestras “tres horas” en un estilo de vida que me han formado un rostro y corazón.

A la Ing. Silvia Vega

Por su gran profesionalismo y apoyo repleto de consejos y experiencia a lo largo de este trabajo y gran parte de la carrera, convirtiéndose así no solo en mi asesora sino en un ejemplo a seguir como amiga, colega de Aragón e Ingeniera.

“No solo se enseña lo que se sabe, también se enseña lo que se es”

A mis Amigos y Profesores

Por que gracias a sus consejos, conocimiento y paciencia ante mis interminables dudas, me dieron las herramientas académicas y personales necesarias para llegar a este y más logros de mi carrera.

A la UNAM

Por ser mi hogar y templo de conocimiento...

A la Máxima casa de estudios, recinto de esperanzas y metas cumplidas, que con sus imponentes cimientos ha albergado en sus aulas a cientos de soñadores como yo, y que ha llenados a todos sus alumnos de valores, conocimiento y profesionalismo que sin duda no esperamos para poner en alto el nombre de la UNAM

“Por mi raza hablara el espíritu”

*Al Plan de Becas de la Dirección de Sistemas
DGSCA-UNAM*

Por haberme permitido ser parte de sus filas, de su historia, de sus aulas, dotándome de herramientas, amigos, maestros y sobre todo sed de conocimiento.

Por que ser “becario” es una gran dicha y nunca se deja de ser... ¡Gracias!

Introducción

CAPITULO I: Informe General del Diplomado

1. Modulo 1: Sistema Operativo Linux.....	2
1.1 Objetivo General.....	2
1.2 Objetivo Especifico.....	2
1.3 Temario.....	2
1.4 Descripción.....	2
1.5 Introducción al ambiente Linux.....	3
1.5.1 Filosofía de Linux.....	4
1.5.2 Distribuciones.....	4
1.6 Comandos Básicos.....	5
1.7 Nociones de Administración.....	6
1.7.1 Utilerías de Linux.....	6
1.7.2 Comando find.....	7
1.7.3 Comando grep.....	7
2. Modulo 2: Instalación y administración de Linux.....	8
2.1 Objetivo General.....	8
2.2 Objetivo Especifico.....	8
2.3 Temario.....	8
2.4 Descripción.....	8
2.5 Introducción.....	9
2.6 Slackware 10.....	9
2.6.1 Requerimientos.....	9
2.6.2 Perfil y actividades del administrador.....	10
2.7 Aplicación.....	11
2.7.1 Inicio de la instalación.....	11
2.7.2 Bienvenida e informa general.....	11
2.7.3 Selección de teclado.....	12
2.7.4 Identificación de root.....	12
2.7.5 Menú grafico.....	12
3. Modulo 3: Editores para el diseño de Paginas Web.....	16
3.1 Objetivo.....	16
3.2 Temario.....	16
3.3 Descripción.....	16
3.4 Introducción.....	17
3.5 HTML.....	18
3.5.1 Etiquetas.....	18
3.5.2 Formato.....	18
3.5.3 Texto.....	19
3.5.4 Listas.....	19
3.6 Rutas Relativas y Absolutas.....	20
3.7 Imágenes.....	20
3.8 Tablas.....	20
3.9 Formularios.....	21
3.10 Ejercicios y ejemplos.....	22

3.10.1	Etiquetas.....	22
3.10.2	Listas.....	23
3.10.3	Ligas.....	24
3.10.4	Tablas.....	25
3.11	Aplicación.....	26
4.	Modulo 4: Administración de servidores WWW con Linux.....	27
4.1	Objetivo General.....	27
4.2	Objetivo Especifico.....	27
4.3	Temario.....	27
4.4	Descripción.....	27
4.5	Introducción a los servidores WWW.....	28
4.5.1	Servidores WWW.....	28
4.6	Instalación de Apache Server.....	30
4.7	Configuración de Apache.....	30
4.8	Ejecución del servidor.....	31
4.9	Incorporación de Módulos.....	32
4.10	Accesos.....	32
4.11	Manejo de sitios Virtuales.....	32
4.12	Aplicación.....	33
4.12.1	CGI's.....	33
4.12.2	Host virtual.....	34
5.	Modulo 5: Programación con PHP.....	35
5.1	Objetivo General.....	35
5.2	Objetivo Especifico.....	35
5.3	Temario.....	35
5.4	Descripción.....	35
5.5	Introducción.....	36
5.6	Elementos del lenguaje.....	37
5.6.1	Sintaxis.....	37
5.6.2	Comentarios.....	37
5.6.3	Variables.....	37
5.6.4	Operadores.....	37
5.6.5	Boques y sentencias.....	37
5.7	Herramientas elementales.....	38
5.7.1	Arreglos.....	38
5.7.2	Cadenas.....	38
5.7.3	Archivos.....	38
5.8	Como diseñar una aplicación CGI.....	39
5.9	Aplicación.....	40
6.	Modulo 6: Interacción de WWW con Bases de Datos.....	42
6.1	Objetivo General.....	42
6.2	Objetivo Particular.....	42
6.3	Temario.....	42
6.4	Introducción.....	43
6.5	Arquitectura Cliente-Servidor.....	44
6.6	Manejo de Formularios Front-End.....	46
6.7	Desarrollo de Back-End con Mysql.....	46

6.8	Instalación y Configuración de la Base de Datos.....	47
6.9	Aplicación	48
7.	Modulo 7: Seguridad	50
7.1	Objetivo	50
7.2	Temario.....	50
7.3	Descripción.....	50
7.4	Introducción.....	51
7.5	Control de Accesos.....	52
7.5.1	Mecanismos de prevención.....	53
7.5.2	Mecanismos de autenticación.....	53
7.5.3	Passwords.....	53
7.6	Sistemas de Archivos.....	54
7.7	Administración Básica de seguridad.....	54
7.8	Seguridad en red	55
7.9	Políticas de Seguridad	56
8.	Programación con Java	57
8.1	Objetivo General	57
8.2	Objetivo Especifico.....	57
8.3	Temario.....	57
8.4	Descripción.....	57
8.5	Introducción.....	58
8.6	La Tecnología Java.....	58
8.6.1	Lenguaje de Programación Java.....	58
8.6.2	La Plataforma Java.....	59
8.6.3	Compilación y ejecución de un programa Java	60
8.7	Estructura del Lenguaje	60
8.8	Conceptos de programación orientada a objetos.....	61

CAPITULO II:	Proyecto de un sitio Web Basado en el Diplomado de Desarrollo e implementación de Software Libre Linux
---------------------	---------------------------------------------------------------------------------------------------------------

9. Planteamiento del problema.....	63
10. Objetivo General	64
11. Objetivo Particular.....	64
12. Justificación	65
13. Desarrollo.....	66
13.1 Opciones educativas en línea.....	66
13.1.1 E-training	66
13.1.2 Aula virtual.....	67
13.1.3 E-learning.....	67
14. Definición de elementos de trabajo	68
14.1 Servidor.....	68
14.1.1 Características Generales.....	68
14.1.2 Sistema Operativo.....	68
14.1.3 Herramientas requeridas y servicios.....	69
14.1.4 Aplicaciones.....	69
14.2 Equipo de computo en las aulas.....	70
14.2.1 Características del equipo.....	70
14.2.2 Aplicaciones necesarias.....	70
14.2.3 Políticas; uso de los equipos y área de trabajo	71
14.3 Instructor.....	71
14.3.1 Responsabilidades, derechos y obligaciones.....	71
14.4 Usuarios.....	71
14.4.1 Prohibiciones	71
14.4.2 Obligaciones	72
14.4.3 Derechos	72
14.5 Aulas o medios de aprendizaje.....	72
14.6 Desarrollo del temario.....	72
14.6.1 Nivel de aplicación y aportación a los participantes.....	73
14.6.2 Método de aprendizaje	73
14.6.3 Conocimientos previos del usuario.....	73
14.6.4 Diseño del temario.....	74
14.6.5 Método de evaluación.....	74
14.6.6 Implantación y mantenimiento del sistema	74
14.7 Aula Virtual.....	75
14.7.1 Elementos, servicios y composición del sitio Web.....	75
 Conclusiones.....	 79
 Referencias.....	 81



Introducción



Introducción

Debido a que este es un informe del Diplomado de Desarrollo e Implementación de Software Libre **LINUX**, y así mismo plataforma para exponer una de las tantas aplicaciones que tiene este tipo de sistemas en el mundo del cómputo, se decidió distribuir el informe en dos partes, cada uno cumpliendo con sus respectivas metas u objetivos:

- **Informe general del Diplomado**

En la actualidad es necesario contar con sistemas que permitan la automatización del seguimiento y control de procesos que se desarrollen en diversas empresas y oficinas. Con esta automatización se pretende tener un uso más eficiente de los recursos y un control más preciso de los mismos, lográndose con ello, compartir información entre diversas áreas y oficinas a través de la red local e incluso empleando Internet o la intranet corporativa.

El desarrollo de estos sistemas puede ser sustentado empleando software libre como LINUX, PHP, Apache-WWW-Server y MySQL, que son herramientas que han demostrado tener un alto desempeño, gran estabilidad y seguridad y, por el hecho de ser libres, permiten reducir los costos que generan las licencias de uso de software, logrando aprovechar al máximo los equipos de cómputo con que se cuenta.

Debido a que el creciente uso de las computadoras implica correr riesgos de seguridad en el acceso a la información, se ha vuelto necesario hacer conciencia de que la seguridad es una responsabilidad compartida; por lo que todos los usuarios de computadoras requieren un conocimiento esencial de los elementos de la seguridad que les ofrece Linux.

- ▶ **Objetivo General:**

El participante conocerá nuevas herramientas administrativas que le permitan desarrollar e implementar sistemas para el control de procesos de información, que funcionen de forma natural en red o por Internet, empleando herramientas de software libre que han demostrado tener una alta confiabilidad, alto desempeño y funcionalidad.

- ▶ **Objetivos Específicos:**

- El participante identificará los métodos de operación y administración del sistema operativo, utilizará los programas de manejo de usuarios, y llevará a cabo la instalación del dispositivo de respaldo, configuración de la red e instalación del software.
- El participante instalará el sistema operativo Linux en PC's y configurará la tarjeta de red, video y particiones de disco duro; así mismo manejará el ambiente del usuario.
- El participante elaborará páginas WWW, apoyado con HTML
- El participante identificará el procedimiento para instalar, configurar y administrar su propio servidor de WWW en un servidor de plataforma Linux.
- Proporcionar al participante los conocimientos necesarios que le permitan crear aplicaciones dinámicas e interactivas para el Web utilizando el lenguaje PHP.

- El alumno conocerá y desarrollará una aplicación de bases de datos que funcione a través del WWW, empleando herramientas de Software Libre.
- El participante reconocerá la importancia de la seguridad e identificará los elementos que le permitirán proteger el sistema y la información.
- **Propuesta de proyecto educativo**

Vivimos en una era de revolución tecnológica que dicta en gran medida la manera en que el ser humano interacciona con su ambiente socio-cultural. En nuestra época presente existe una diversidad de herramientas tecnológicas que nos ayudan al desempeño de las tareas cotidianas y en el trabajo, siendo las más conocidas y usadas la computadora y la Internet.

Con el crecimiento de Internet, la experiencia de la enseñanza a distancia se ha transformado completamente. Antes la enseñanza a distancia era esencialmente una experiencia aislada, en que el estudiante se enfrentaba a una cantidad enorme de materiales de estudio enviados por correo, con contactos esporádicos y rígidos con un instructor tan lejano como de difícil acceso. En este tipo de entorno, no solamente el estudiante debía superar una serie de dificultades para ponerse en contacto con el instructor, sino que además debía esperar prolongados periodos de tiempo entre el envío de su pregunta y la recepción de la respuesta. Además, la interacción se limitaba a la relación individual entre el estudiante y su instructor, ya que no existía ningún tipo de comunicación de los estudiantes entre sí.

Actualmente se dedica mayor tiempo a explorar nuevos métodos de enseñanza que ayuden a terminar con las fronteras entre los alumnos y profesores de los cursos en línea, es por ello que han surgido diversas soluciones educativas para esta nueva etapa que vive la educación. En el presente trabajo se promoverá al Aula Virtual como opción para cubrir algunas de las necesidades académicas que se presentan muy a menudo en grupos dedicados al estudio, desarrollo e implementación de las tecnologías libres, esto debido a que es un tema relativamente nuevo y difícil de dominar en los inicios y que requiere de un gran apoyo por parte de todos los participantes para lograr un mejor dominio de los temas del software libre.

En este punto cabe mencionar algunas de las ventajas y objetivos que proporciona la implementación de un sitio Web de dedicado a la enseñanza de Software Libre.

- Ofrece a los profesionales una completa oferta formativa, libre de condicionamientos externos como distancia geográfica, incompatibilidad horaria, organización familiar, etc.
- Permite seguir un ritmo de estudio personal, sin perder el apoyo de los tutores.
- Se establece un seguimiento riguroso del alumno y se lleva a cabo una evaluación formal de sus aprendizajes.
- Flexibilidad en los plazos y en la dedicación a la formación.
- Familiarización con el entorno de red, posibilitando el uso de sus recursos en el futuro.
- Inmediatez: cualquier cambio introducido en el curso: el programa, los materiales, las actividades, etc., está disponible instantáneamente para los participantes.

- La gran disponibilidad de recursos: Internet dispone de grandes masas de información, muchas de ellas de acceso libre, que con la debida orientación y selección rigurosa pueden ser de gran utilidad en el aprendizaje.
- Facilita un material de estudio que se convierte en una fuente de consulta profesional.
- Favorece la autorregulación personal que incide sobre aspectos de la propia personalidad, como el autocontrol o la disciplina.
- Facilita el intercambio con profesionales del mismo sector, o con intereses similares.

Por ultimo cabe remarcar que el propósito de la propuesta del trabajo es brindar una alternativa de enseñanza-aprendizaje basado en software libre, así como puntualizar las ventajas del mismo y proporcionar los métodos, aplicaciones y herramientas necesarias básicas para la creación de un sitio Web como apoyo didáctico a clases presénciales que funcione como **Aula Virtual** dedicada al aprendizaje y difusión del Sistema Operativo Linux.

Informe General del Diplomado

Capítulo 1



“Si suspiras al recordar aquellos días cuando los hombres eran hombres y escribían sus propios drivers. Si te sentís sin ningún proyecto interesante y te gustaría tener un verdadero sistema operativo que pudieras modificar a placer, Si te resulta frustrante tener solo Minix. Entonces este artículo es para vos”

Linus Torvalds

► **Objetivo General:**

El participante adquirirá los conocimientos generales sobre el funcionamiento del sistema operativo LINUX a través del manejo de los comandos más utilizados. Conocerá nuevas herramientas que le permitan desarrollar e implementar sistemas para el control de procesos e información, que funcionen de forma natural en red o en Internet, empleando herramientas de software libre que han demostrado tener una alta confiabilidad, alto desempeño y funcionalidad

► **Objetivo Específico:**

El participante conocerá el manejo del sistema operativo Linux, así como utilerías básicas, manipulará archivos, directorios, editores de texto y nociones básicas de administración.

► **Temario**

1. Introducción al ambiente LINUX
2. Comandos y Utilerías Básicas
3. Nociones de Administración

► **Descripción:**

A lo largo de este primer Módulo nos enfrentaremos con la parte crucial del resto del Diplomado, esto se debe a que mayoría de los usuarios estamos acostumbrados a tratar con sistemas gráficos y comerciales, así que el reconocimiento y manejo de comandos básicos será el primer paso para adentrarnos al mundo del Software libre.

Un sistema LINUX es un software muy completo y complejo que se hará mucho más robusto y potente conforme nos adentremos a él, así que la mejor manera de conocerlo es recorrerlo de lo general a lo particular, esto es sabiendo como nace, por que se creo, bajo que filosofía y con cuantas herramientas cuenta para realizar tareas específicas.

► Introducción al Ambiente Linux

Linux posee todas las características de los sistemas comerciales usados en los entornos de trabajo más exigentes por lo tanto se está adoptando con mayor frecuencia en el mercado de los grandes servidores y, está cada vez más presente en los equipos de los usuarios domésticos. Este nivel de prestaciones ha sido posible gracias a la disponibilidad de su código fuente, que permite a cualquier programador del mundo un acceso total a su corrección y mejora.

Algunas de sus principales características son:

- **Linux y sus Shells:** Cada usuario de un sistema Linux tiene su propia interfaz de usuario o Shell. Los usuarios pueden personalizar sus shells adecuándolos a sus propias necesidades específicas. En este sentido, el Shell de un usuario funciona más como un entorno operativo que el usuario puede controlar.
- **Linux es Multitarea:** La multitarea no consiste en hacer que el procesador realice más de un trabajo al mismo tiempo (un solo procesador no tiene esa capacidad), lo único que realiza es presentar las tareas de forma intercalada para que se ejecuten varias simultáneamente. Por lo tanto en Linux es posible ejecutar varios programas a la vez sin necesidad de tener que parar la ejecución de cada aplicación.
- **Linux es Multiusuario:** Para que pueda desarrollar esta labor (de compartir los recursos de una computadora) es necesario un Sistema Operativo que permita a varios usuarios acceder al mismo tiempo a través de terminales, y que distribuya los recursos disponibles entre todos. Así mismo, el sistema debería proporcionar la posibilidad de que más de un usuario pudiera trabajar con la misma versión de un mismo programa al mismo tiempo, y actualizar inmediatamente cualquier cambio que se produjese en la base de datos, quedando reflejado para todos. En conclusión, en el sistema multiusuario, varios usuarios pueden acceder a las aplicaciones y recursos del sistema Linux al mismo tiempo. Y, por supuesto, cada uno de ellos puede ejecutar varios programas a la vez (multitarea).
- **Linux es Seguro:** El concepto de seguridad es siempre difícil de abordar y de adoptar. Un sistema puede ser seguro para un determinado tipo de actividades e inseguro para otras. Si se quiere que el sistema sea seguro, se debe administrar de tal forma que se tengan controlados a los usuarios en todo momento. Para la ardua tarea de seguridad surgen nuevas herramientas constantemente, tanto para detectar intrusos como para encontrar fallos en el sistema y evitar así ataques desde el exterior.
- **Linux y las Redes de Ordenadores:** Cuando se trabaja con Linux se está ante un sistema operativo orientado al trabajo de redes.
 - Linux dispone de varios protocolos como PPP, SLIP, TCP/IP, PLIP, etc., para la transferencia de archivos entre plataformas. Tiene a su disposición multitud de aplicaciones de libre distribución que permiten navegar a través de Internet y enviar y recibir correo electrónico, hacer una videoconferencia, transferir archivos, etc. Posee gran variedad de comandos para comunicación interna entre usuarios que se encuentren ubicados en plataformas distintas (gracias a utilidades como telnet).

- Independencia de dispositivos Linux admite cualquier tipo de dispositivo (módems, impresoras) gracias a que cada que se ha instalado uno nuevo, se añade al Kernel el controlador o driver necesario con el dispositivo, haciendo que el Kernel y el driver se fusionen. Lo importante de esto es que el controlador funciona como un módulo completamente aislado del núcleo de Linux, dando así una mayor seguridad y estabilidad al sistema

También podríamos citar su arquitectura modular que permite prácticamente que el usuario se fabrique un núcleo. Estas características lo convierten, probablemente, en uno de los sistemas más avanzados que existen.

- **Filosofía Linux**

- Pequeños programas que cumplen con una función específica
- Obtiene entrada, procesa, devuelve salida. Se combina para realizar tareas complejas.
- Sistema personalizado a las necesidades del usuario.

- **Distribuciones**

Una distribución es un modo de facilitar la instalación, la configuración y el mantenimiento de un sistema GNU/Linux. Al principio, las distribuciones se limitaban a recopilar software libre, empaquetarlo en disquetes o CD-ROM y redistribuirlo o venderlo.

Ahora las grandes distribuciones son potentes empresas que compiten entre sí por incluir el último software, a veces también software propietario, con instalaciones gráficas capaces de auto-detectar el hardware y que instalan un sistema entero en unos cuantos minutos.

A continuación se presentan algunas de las diferentes distribuciones de Linux.



► Comandos Básicos

Como parte inicial de este apartado de Linux, el reconocimiento de comandos básicos es indispensable es por ello que sólo se muestran algunos de los tantos existentes.

Se llaman comandos básicos a todas aquellas instrucciones en modo texto que ayudaran a reconocer el sistema, nuestra máquina y usuarios, así como todos aquellos que faciliten la creación de tareas específicas y monitoreo de procesos que se estén realizando en tiempo real.

A continuación se presenta una lista de los comandos más usados para un primer acercamiento

COMANDO	Descripción	Ejemplo
ls	Lista el contenido del directorio	ls -l, ls -a
cp	(copy) Se encarga de copiar un archivo o directorio	cp archivo1 archivo2
rm	(remove) Borra archivos o directorios	rm -f fichero, rm -rf directorio, rm -i fichero
mkdir	(make dir) crea directorios	mkdir dir_nuevo
rmdir	(remove dir) elimina directorios vacíos	Rmdir directorio
mv	(move) renombra o mueve archivos o dir	mv dir1 dir2,
date	(fecha) Muestra la fecha actual del sistema	Date, date 10091923
history	Muestra el historial de comandos introducidos por el usuario.	history more
more	Muestra el contenido del archivo en forma paginada	more archivo
cat	Muestra todo el contenido del archivo	cat archivo
chmod	Cambia los permisos del archivo	chmod 755 archivo
mount	(montar) Monta unidades de disco duro, CD-Rom , flexible	mount /dev/hda2 /mnt/lx
umount	(desmontar) al contrario del comando anterior este desmonta las unidades	umount /dev/hda2
ftp	Establece una conexión FTP	ftp memo@dcaa.unam.mx
whois	Nombre de dominios	whois www.google.com
who	(quien) muestra los usuarios que se encuentran conectados al sistema	Who, w
mail	Comando para enviar y leer correos electrónicos	mail -v memo@dcaa.unam.mx < archivo
sort	Ordena el contenido de un archivo	sort archivo1
tail	Muestra las últimas 10 líneas de un archivo	tail -5 archivo
head	Muestra las primeras 10 líneas de un archivo	

► Nociones de Administración

Existe un gran tramo para pasar de un usuario normal del sistema a un administrador del mismo, el único camino para llegar es la práctica diaria.

Cuando a un administrador le es asignada una máquina, la primer tarea es conocer lo más que se pueda el equipo con el que va a trabajar, es por ello que continuación se muestran una serie de comandos para ese fin.

Para un mejor manejo y reconocimiento del sistema el administrador cuenta con una serie de herramientas (comandos) básicos:

COMANDO	Descripción	Ejemplo
adduser	Añade un usuario al sistema	adduser -md /home/memo memo
userdel	Elimina un usuario del sistema	userdel -r memo
usermod	Hace modificaciones a una cuenta del usuario	Usermod -s /bin/sh memo
df	(disk free) Muestra el espacio de las particiones	df -k
uname	Muestra información general del servidor	uname -a
netstat	Muestra las conexiones activas	netstat -a
ps	(Process) muestra todos los procesos del sistema	Ps -eaf
free	Muestra el estado de la memoria RAM y SWAP disponible	Free
ping	Herramienta de red para comprobar entre otras cosas si llegamos a un host remoto.	Ping www.google.com
du	(disk use) uso del disco , espacio	du -h
ifconfig	Interface config. Configuración de interfaces de red, módems, etc.	ifconfig eth0 ip netmask broadcast up

Existen una serie de herramientas que son muy poderosas en cuanto al manejo de la información que contiene el servidor; y es por ello que se le debe tomar mucho en cuenta a lo largo de este módulo, éstas son las Utilerías de Linux. Por fines educativos en esta sección solo se tomaran dos de estas herramientas, debido a su uso estándar.

- **Utilerías en Linux**

Una utilería se define como toda aquella unión o interrelación de comandos / herramientas que realizaran operaciones más complejas como monitorear, respaldar o realizar búsquedas completas. A continuación se muestran dos comandos básicos para el manejo de las utilerías

- **Comando Find**

Este comando es realmente útil y necesario manejarlo a la perfección ya que nos ayudara a realizar búsquedas con las siguientes características:

- Buscar el nombre con cierta cadena de caracteres con algún patrón;
- El enlace a ciertos archivos;
- Conocer cuales fueron los archivos usados por última vez en un cierto período de tiempo.
- Los archivos que tienen un tamaño comprendido dentro de cierto intervalo.
- Saber si los archivos son de cierto tipo (regular, directorio, enlace simbólico, etc.).
- Determinar a quien pertenecen los archivos encontrados (usuario o grupo).
- Archivos que tengan ciertos permisos de acceso;
- El patrón dentro de determinado archivo

Ejemplo:

Comando	Descripción
<code>find /var -name *.log print</code>	Busca en el directorio /var los archivos terminados en .log, imprime sus nombres en la salida.
<code>find /tmp -size +200k -print</code>	Busca archivos mayores de 200k. En los argumentos numéricos, +N es mayor que N, -N es menor que N, N es exactamente igual a N.
<code>find /var/spool/mail -atime +30 -print</code>	Busca archivos no accedidos hace más de 30 días. La opción -atime se refiere a tiempo transcurrido desde última lectura, -mtime desde última modificación de estado o permisos, -ctime de contenido.
<code>find /var/tmp -empty -exec rm {} \;</code> <code>find /var/tmp -empty -exec rm {} \;</code>	Busca archivos vacíos y los borra.

- **Comando grep**

Localiza cadenas en archivos de texto, tomando las líneas de la entrada estándar o de un archivo especificado. Este comando tan usado tiene dos variantes **fgrep** y **egrep**,

fgrep: Permite buscar múltiples objetivos que se pueden especificar en la línea de comandos o en el archivo. A cambio de esto no permite la utilización de expresiones regulares pero gracias a ello las búsquedas son más rápidas.

egrep: Incluye lo mejor de **grep** y **fgrep** y añade algunas otras características. Al igual que fgrep permite buscar múltiples objetivos y extraerlos de un archivo, como **grep** permite la búsqueda de expresiones regulares y añade algunos metacaracteres adicionales para realizar búsquedas extendidas.

MODULO II

ADMINISTRACION DE LINUX

“LINUX: la informática vuelve a ser divertida.”

▶ **Objetivo General:**

El participante manejará los métodos de operación y administración del sistema operativo, reconocerá los programas de manejo de usuarios, llevará a cabo correctamente los procedimientos de altas y bajas del sistema, podrá realizar la instalación del dispositivo de respaldo, configuración de la red y la instalación de software.

▶ **Objetivo Específico:**

Los conocimientos adquiridos permitirán al participante conocer la filosofía de Linux para administrar, eficazmente, un equipo que trabaje bajo este sistema operativo.

▶ **Temario**

1. Filosofía y estructura de Linux
2. Slackware 10
 - a. Requerimientos
 - b. Perfil y actividades del Administrador
3. Aplicación.
 - a. Inicio de la instalación

▶ **Descripción**

Como se menciona en el módulo pasado existen varias distribuciones todas ellas con un ciertas ventajas y desventajas, para este caso, se eligió la distribución de Slackware ya que es mucho más flexible que otros Linux, permite personalizar la instalación de acuerdo con las características del equipo y la tarea que se le será asignada como proporcionar algún tipo de servicio (telnet, ftp, http, etc.), además de explotar los beneficios que proporciona una licencia libre.

► Introducción

Slackware Linux es una distribución de un completo sistema multitarea de 32-bits. Actualmente, en su última versión 10.1, está basado en el kernel Linux 2.4.29 y la librería C del GNU versión 2.3.4. Contiene un programa de instalación fácil de usar, extensa documentación, y un sistema de manejo de paquetes basado en menús. Una instalación completa que incluye el Sistema de ventanas X, entornos de escritorio KDE, GNOME, XFce, entornos de desarrollo C/C++, Perl, Python, Java, utilidades de red, servidor de correo, servidor de noticias, servidor web, servidor FTP, el programa de manipulación de imágenes del GNU, navegadores web, además de muchos otros programas.

Desde su primer lanzamiento en Abril de 1993, el Proyecto Slackware Linux se ha esmerado en producir la distribución de Linux más profesional posible. Slackware obedece a los estándares de Linux publicados, como el Linux File System Standard. Siempre se ha tomado en cuenta la simplicidad y la estabilidad como meta, dando como resultado que Slackware se haya convertido en una de las distribuciones más populares, estables y amigables disponible.

La distribución de paquetes en Slackware se hace principalmente con archivos “.tar” comprimidos, si bien se puede utilizar también RPM. La interface del programa de instalación es por texto, y necesita un mayor conocimiento de Linux que la mayoría de las otras distribuciones. Esto puede ser una desventaja para usuarios principiantes, pero no representa mayor dificultad para usuarios intermedios o avanzados de Linux.

► Slackware Linux 10

• Requerimientos

Una de las principales ventajas de este sistema es el bajo requerimiento de máquina necesario para ejecutarse. Linux Slackware puede trabajar desde equipos 386 o superiores.

- Procesador 386.
- 16MB en RAM.
- 50 megabytes de espacio libre en disco duro.
- Unidad de 3,5”.
- Tarjeta de red.

Por el contrario, si de desea montarlo como un servidor los requerimientos mínimos son los siguientes:

- Procesador Pentium.
- 128 en RAM.
- 2000 megabytes de espacio libre en disco duro.
- Unidad de CD-ROM.
- Unidad de 3.5”.
- Tarjeta de red.
- Tarjeta de video.

- **Perfil y actividades del administrador**

Usualmente al abordar el tema de las actividades de un administrador, es común relacionarlas sólo al monitoreo y respaldo continuo del sistema, y no se visualiza la cantidad de trabajo que se tiene al adquirir una responsabilidad como esta, en este apartado del curso no se pretende analizar todas las actividades de un administrador, pero si al menos exponer el perfil básico que se debe cumplir.

Un administrador de sistemas LINUX es la persona responsable del mantenimiento y funcionamiento correcto del sistema Linux utilizado en la empresa o institución. Habitualmente, esto incluye todo, desde seleccionar el hardware sobre el que instalar el sistema, la instalación y configuración inicial, hasta el mantenimiento continuo del hardware y el software que conforma el sistema

La administración del sistema es vital, en especial en una máquina con sistema operativo multiusuario. Hay que resaltar que las tareas de administración son significativamente más complejas sobre un sistema multiusuario que sobre un sistema monousuario, pero aún así las versiones más recientes del sistema Linux han producido herramientas mejoradas que pueden facilitar dichas tareas.

Para tener un nivel de seguridad bueno, es necesario que alguien dedique tiempo a configurar correctamente los parámetros de seguridad del sistema, configurar el ambiente de trabajo de los usuarios, revisar y fijar los permisos de acceso de los archivos, ejecutar programas de monitoreo de seguridad, revisar las bitácoras (*logs*) del sistema, etc. No se puede pretender tener seguridad completa sino se tiene a una persona que esté dedicada de tiempo completo a este tema.

Con esto decimos que el tener un plan de trabajo en donde se determinen cuales serán las principales actividades a las que se le dedicará el tiempo para el aseguramiento de nuestro sistema será de vital importancia para tener una buena administración sobre el mismo, a continuación se enlistan algunas de las rutinas básicas:

- Dar de alta y dar de baja del sistema
- Mantenimiento de claves de usuario
- Sistema de archivos
- Instalación y mantenimiento de dispositivos
- Administración del área de Swap
- Realización de respaldos
- Uso de memoria y CPU
- Configuración y mantenimiento del uso de la red
- Seguridad
- Monitoreo y mejora del desempeño del sistema
- Contabilidad de uso de recursos
- Instalación de software
- Copias de seguridad
- Conocimiento de programación y utilerías del sistema
- Establecer Políticas de uso y administración
- Mantener Canales de Comunicación con el usuario

Al hacer este recorrido por las labores diarias de un administrador, es como se llega a comprender de manera directa las ventajas que proporciona el software libre.

► **Aplicación**

Para poder hacer un recuento de lo aprendido en este modulo fue necesario realizar una instalación completa con ciertas características, que a continuación se describen:

- 1) Instalar el Sistema Operativo Slackware 10 con las siguientes particiones:
 - /
 - /home
 - /usr
 - /opt
 - /var
- 2) Realizar la configuración de red adecuada
- 3) Iniciar el sistema de modo gráfico
- 4) Configurar idioma
- 5) Generación de usuarios y grupos
- 6) Gestión de puertos

• **Inicio de Instalación**

Una instalación completa consume aproximadamente unos 2.4 Gb (esto incluyendo todos los paquetes disponibles) esto varia dependiendo de las necesidades requeridas por el usuario.

La primera etapa de instalación requiere un gran análisis ya que esta en base a los servicios que va a dar nuestro equipo, es aquí donde se determina el tipo y número de particiones, recordando que las más comunes son las siguientes:

- /bin
- /sbin
- /etc
- /lib
- /dev
- /home
- /home/usuarios
- /base
- /web

Hay varios tipos de instalación pero por el momento el enfoque es en el uso del CD

• **Bienvenida e información general**

Una vez tomada la decisión del tipo de particiones y su tamaño procedemos a insertar el CD de nuestra versión 10 de Slackware, al hacer esto nos hará una pregunta respecto a si queremos continuar (bootear) con el sistema, lo cual contestamos con ← <enter>

#boot:

En el siguiente paso se configura el teclado (de manera momentánea) ya que esa configuración dura sólo el proceso de instalación, una vez terminado se tendrá que

editar ciertos archivos para continuar con dicha configuración. Este paso genera una lista de distintos tipos de teclados, en este caso es la forma en español (qwerty/es.map)

- **Selección de teclado adecuado**

Keyboard 1 ↵

Para continuar con el proceso se debe **“logear”** como (**root**) y hacer las respectivas particiones del Disco Duro, para eso en el **“Prom.”** Se ejecutara el comando **fdisk**

#fdisk /dev/hda

- **Identificación como root y Particionar el disco duro**

A continuación se muestran las opciones para iniciar el particionamiento del Disco Duro y el tipo de Sistema de Archivos para Slackware.

a	<i>toggle a bootable flag:</i> poner o quitar la marca de arranque de una partición.
d	<i>delete partition:</i> Eliminar una partición.
l	<i>list known part. types:</i> mostrar los tipos de particiones reconocidas.
m	<i>print this menu:</i> Mostrar la ayuda.
n	<i>add a new partition:</i> Crear una nueva partición.
p	<i>print the partition table:</i> Mostrar la tabla de particiones.
q	<i>quit w/out saving changes:</i> Salir sin salvar los cambios.
t	<i>change a partition's sysid:</i> Cambiar el tipo de partición.
u	<i>change display/entry units:</i> Cambiar unidades entre bloques y sectores.
v	<i>verify the partition table:</i> Comprobar la tabla de particiones.
w	<i>write table and exit:</i> Salir y salvar.
x	<i>extra functionality:</i> Opciones extra (sólo experto).

Se Deberá crear una partición **Linux swap** (Memoria virtual) de al menos la mitad de la memoria RAM disponible como mínimo y una partición Linux native (partición principal) de al menos **20 Gb**.

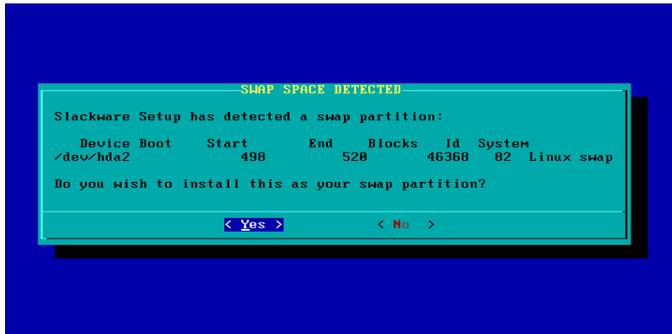
- **Entrada menú gráfico**

Arrancar el programa **setup** ↵

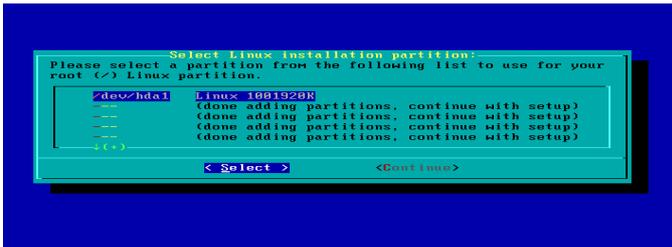
Esto arranca un programa “gráfico” con submenús de fondo azul

Help	(ayuda)
Keymap	(Configuración del teclado)
AddSwap	(este paso ya se realizo así que sólo se indica que puede formatearlo)
Target	(Se Indica en que partición estará “/”)
Source	(Selección de modo de instalación)
Select	(Qué paquetes se van a instalar)
Install	(Se instalaran todos los paquetes por default)
Configure	(configuración de periféricos)

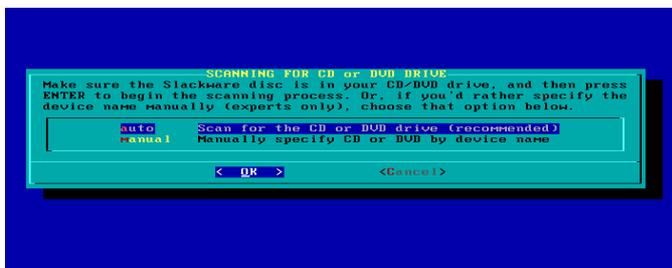
A continuación se muestra la serie de pantallas de instalación, cada una de ella con una breve descripción de ese proceso:



En este primer paso, se detecta la partición swap y pide la confirmación de formato y ubicación



A continuación se pide la ubicación y selección de la partición para el sistema operativo.



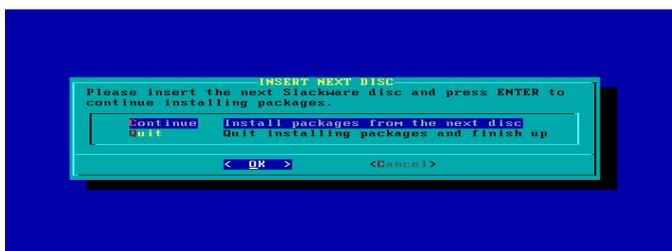
En esta ventana sólo se especifica la manera (medio) de la instalación, en este caso será por CD



Una vez preparado todo el sistema de archivos, falta indicar los paquetes que se requieren e iniciar con el proceso.



Existen diferentes formas de Instalación, todas ellas dependen del nivel del administrador, en esta instalación se hará una instalación completa (full)



Llega un momento en que se debe ingresar el segundo disco para continuar con el proceso, el mismo programa lo pide expulsando el CD1



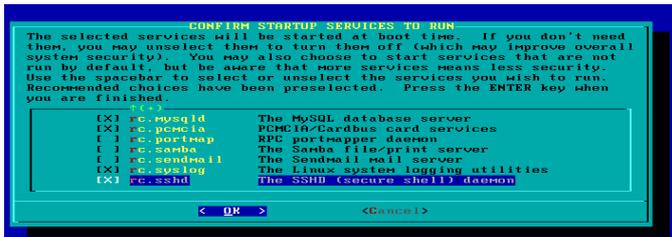
Una vez terminada la carga de los paquetes necesarios, se hace la instalación y configuración del programa de arranque (lilo)



Configuración de Mouse

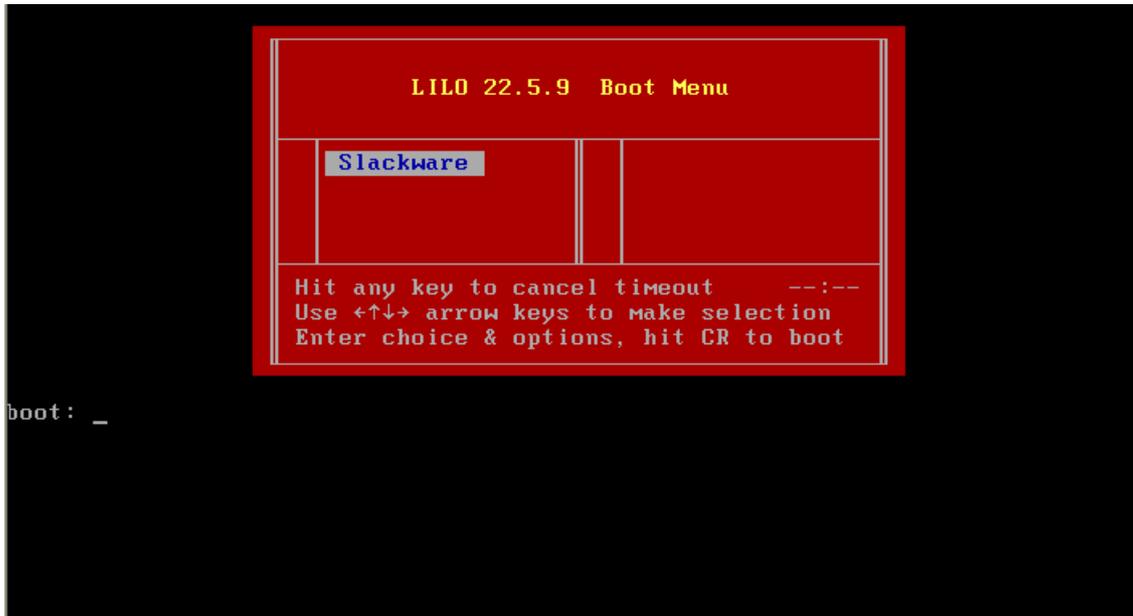


Configuración de Red



Configuración de Servicios

Con este paso se ha terminado la instalación, es necesario asignar un password a root (seguro) y reiniciar el sistema para empezar a trabajar con Linux Slackware.



La siguiente pantalla pertenece al gestor de arranque, Lilo,



A continuación se muestra el ambiente gráfico cargado y configurado por completo de Slackware 10

MODULO III

EDITORES PARA LA CREACION DE PAGINAS WEB

▶ **Objetivo:**

El participante elaborará páginas WWW, utilizando el lenguaje HTML

▶ **Temario:**

1. Etiquetas
2. Formato
3. Rutas Relativas, Rutas Absolutas
4. Imágenes
5. Tablas

▶ **Descripción:**

Este Módulo se centrara en el manejo del lenguaje HTML, de tal manera que se editen y reconozcan la mayoría de las etiquetas de manera manual, ya que actualmente existe gran variedad de editores HTML que hacen prácticamente todo sin que el usuario se de cuenta de lo que esta pasando en su código, de hecho para eso están estos editores pero el problema es que la mayoría hace uso de etiquetas basura (innecesarias), que bien se podrían eliminar.

Por otro lado cuando estas etiquetas de HTML se fusionan con PHP es mucho mejor manejarlas paso a paso que el ayudándose de un editor de código, esto debido al uso de variables.

▶ **Introducción**

Uno de los servicios por excelencia de Internet es la Web, servicio de visualización, basado en un lenguaje de marcas llamado HTML, en un protocolo para este tipo de transferencias llamado HTTP

Hoy más que nunca se está en gran contacto con páginas Web, estas se han convertido en el medio de información más flexible y común entre las empresas y escuelas, tiene tanto uso que incluso ya se cuenta con este servicio en los celulares, consolas de videojuegos, palms, etc.

Desde sus inicios y para un mejor manejo de la información se crearon una serie de especificaciones para enlazar una maquina con otra y así darle vida a este nuevo proyecto (Web), fue por ello que diversos organismos elaboraron los siguientes puntos:

1. **Protocolo:** Para poder transferir de forma adecuada los diferentes documentos de hipertexto por Internet.
2. **URL:** Ya se contaba con un protocolo que respaldara el transporte de los datos, pero era necesario contar con alguna liga o dirección que mostrara la ubicación exacta de documentos, y es aquí donde surge
3. **Lenguaje:** Para poder dar formato a un documento, se desarrolló un lenguaje específico, cuya característica primordial era el manejo de Hipertexto
4. **Browsers:** Ya se tenía todo para la publicación de los documentos, sólo falta el último paso, un programa que permitiera visualizar la información, y es aquí donde surge esta última etapa en la cadena Web.

Es así como el WWW se convierte en un sistema de información global, público e independiente, mediante el cual un usuario cualquiera puede acceder a documentos HTML almacenados en diferentes servidores ubicados en cualquier parte del mundo, pudiendo saltar de un servidor a otro de forma instantánea mediante los enlaces de hipertexto contenidos en las páginas Web.

Para generalizar los servicios que brinda la Web a continuación se muestran algunos de ellos:

- **Gráfico:** Compuesto por texto, imágenes, sonidos, animaciones
- **Global:** Se puede conectar o enlazar desde cualquier parte de mundo, desde cualquier plataforma o cualquier navegador.
- **Pública:** Toda la información publicada es de carácter público, libre para toda persona.
- **Dinámica:** En cualquier momento la información contenida en los documentos puede ser actualizada sea por el administrador o el usuario.

La Arquitectura detrás de las páginas:

Para poder visualizar una página interviene varios elementos; una máquina cliente y un host servidor, es decir, el que hace la solicitud y el que responde a ella, esta conexión o enlace se realiza por medio de Internet y bajo el protocolo http.

▶ **H T M L**

En 1989, Tim Berners-Lee y Anders Berglund, dos investigadores del Laboratorio Europe de Física de Partículas (CERN), crearon un lenguaje basado en etiquetas e hipertexto para marcar documentos técnicos con el fin de compartirlos en Internet. Así surgió una aplicación simplificada del SGML llamado HTML (Lenguaje de Marcas de Hipertexto), que se convirtió en el primer formato de información estándar en la Web.

En este caso se puede definir a HTML como un lenguaje de codificación para Web que informa a los navegadores como desplegar los textos, vínculos, gráficos entre otros medios de un documento.

Hasta hoy en día no existe una estandarización de este lenguaje ya que con el tiempo se ha ido incrementando y desarrollando de manera incontrolada esto debido a la inclusión multimedia haciendo uso de imágenes, videos, flash, applets de Java, etc.

- **Etiquetas**

Como ya se ha mencionado, HTML el lenguaje de marcas, se basa en el manejo de una serie de etiquetas especiales dentro del texto del documento, que generalmente se marcan en par, es decir abrir-cerrar instrucción, éstas son las que manipulan la manera en como se despliega la información en el navegador.

La sintaxis general es: <etiqueta> Texto </etiqueta >

Debido a su gran número de etiquetas a continuación listaré sólo algunas de ellas y sus categorías.

- **Formato**

Tal como lo se haría con cualquier procesador de palabras para elaborar algún documento, por ejemplo; un currículum, al cual se le dedica tiempo para darle una apariencia agradable y ordenada, pasa lo mismo con un documento Web.

Sólo que en este diseño o estilo se deben agregar las etiquetas necesarias para indicar que se trata de un documento en HTML y después dar el formato adecuado y cambiar las letras a negritas, subrayadas, de mayor tamaño, etc.

La forma básica de un documento HTML es el siguiente:

```
<html>
<head ><title >TITULO</title></head>
<body>
```

 Texto

```
</body>
</html>
```

- **Texto**

Para cambiar la forma del texto, se cuenta con una gran variedad de etiquetas, a continuación se enlistan algunas de las básicas.

<h1>	Da el tamaño a la letra siendo en 1 la letra mayor y 5 la menor
<p>	Etiqueta para alinear párrafos
 	Salto de línea
	Letra en negrita
<i>	Texto en cursiva
<u >	Texto subrayado
	Texto resaltado
<pre>	Pre - formato de texto
	Con esta etiqueta se puede dar color, tamaño, tipo
<!--Comentario-->	

- **Listas**

Usualmente para mostrar información de manera ordenada se recurre a puntualizar ciertos temas y para esto se utilizan las ventajas que dan las listas, para poder trabajar mejor se cuenta con tres tipos de listas:

- Listas con viñetas:

Sintaxis:

```
<ul type="tipo viñeta">  
<li>concepto  
<li>concepto  
</ul>
```

- Listas numeradas u ordenadas

Sintaxis:

```
<ol type="tipo orden">  
<li>concepto1  
<li>concepto2  
</ol>
```

- Lista de definición

Sintaxis:

```
<dl>  
<dt>concepto  
<dd>definición  
<dt>concepto2  
<dd>definicion2  
</dl>
```

▶ **Rutas Relativas y Absolutas**

Las Rutas relativas hacen referencia de alguna página o documento dentro del mismo directorio, mientras que las rutas absolutas se usan si el archivo o documento se encuentra fuera de éste.

Relativo:

```
<a href.= "página">comentario </a>  
<a href.= "indice.html">Indice</a>
```

Absoluto:

```
<a href.= "/imágenes/archivo">Imagen </a>
```

También se cuenta con otro tipo de enlace y esta es la liga de referencia o ancla, que ayudará cuando se tienen a documentos muy extensos, para su uso es necesario crear un enlace de origen – destino de la siguiente manera:

```
<a href.= "#ancla">comentario </a>  
<a name="ancla">
```

▶ **Imágenes**

El manejo de las imágenes sobre documentos HTML ayudó mucho a dar mayor presentación a los trabajos revolucionando la manera de mostrar información y es por ello que se le da importancia en esta parte del módulo. Aunque hoy en día también ya se les ha dado vida a estas imágenes haciendo uso de animaciones o películas con diferentes herramientas, las imágenes planas no pasaran de moda.

Para las últimas versiones del lenguaje HTML, se agregaron más características y usos a las imágenes incorporándoles; márgenes, color, tamaño, posición, etc.

La etiqueta para el manejo de imágenes es ``

▶ **Tablas**

Las tablas son las herramientas más útiles de que se disponen en HTML, ya que éstas ayudaran a dar una mejor presentación y formato a todo el contenido de la pagina, además de ello se le puede dar formato agregándole color, tamaño de bordes, etc.

A continuación se muestra la sintaxis de inserción de tablas

```
<table border>  
  <tr>  
    <td>dato1</td><td>dato2</td>  
  </tr>  
</table>
```

Donde `<tr>` son las filas y `<td>` las columnas

► **Formularios**

Para fines prácticos de este módulo se puede decir que el manejo de formularios es la manera más interactiva en el lenguaje de HTML, esto en el sentido de que gracias a estos, se puede mantener un contacto más personal con los usuarios del sistema recopilando información importante para los administradores.

Estos elementos son muy comunes hoy en día, ya la mayoría de las páginas en Internet cuenta con alguno de éstos, quizá para recibir comentarios o para que el usuario vote por algún producto o motivo.

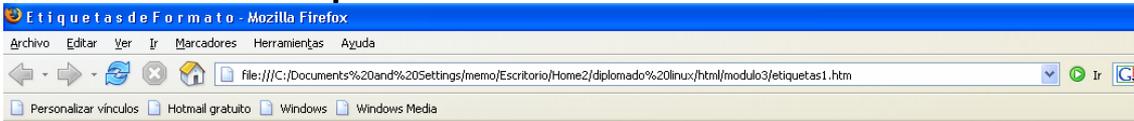
Es importante mencionar el hecho de que al implementar este tipo de elementos, ya se debe incorporar el uso de variables y métodos de envío de estos datos, para lo cual se necesita implementar programación en php, el cual se estudiara en el siguiente modulo.

A continuación se muestran los elementos con los que se cuenta para realizar nuestros formularios.

Texto	Sirve para introducir una caja de texto simple	<input type="text" value="Escriba aquí"/>
Radio	Define un conjunto de elementos de formulario de tipo circular, en los que el usuario debe optar por uno solo de ellos, que se marca con el ratón o tabulador	opcion1 <input checked="" type="radio"/> opcion2 <input type="radio"/>
Checkbox	Define una o más casillas de verificación, pudiendo marcar el usuario las que desee del conjunto total	UNIX <input checked="" type="checkbox"/> Php <input checked="" type="checkbox"/>
Button	Define un botón estándar	<input type="button" value="Aceptar"/>
Password	Define una caja de texto para contener una clave o password	<input type="password" value="....."/>
Submit	Define un botón el cual va a mandar toda la información recopilada	<input type="button" value="Aceptar"/> <input type="button" value="Aceptar"/>
Reset	Define un botón que borrará toda información obtenida antes de ser mandada al servidor.	<input type="button" value="Borrar"/> <input type="button" value="Restablecer"/>
Select	Define una entrada de datos en forma de lista desplegable	Escolaridad <input type="text" value="Primaria"/> ▼
TextArea	Inserta una caja de texto de múltiples líneas	<input type="text" value="Cuadro"/>

▶ Ejercicios y Ejemplos

• Etiquetas



Nombre: **Guillermo Tenorio Ruiz**
 Pasatiempo: Leer
 Lectura Favorita: *Filosofía*



Tomé este diplomado por la necesidad de que requiere mi proyecto, ya que estoy trabajando constantemente con software libre, y mi meta personal es llegar a ser un buen administrador de servidores. Me gustaria en un futuro estudiar a fondo la seguridad

Tomé este diplomado por la necesidad de que requiere mi proyecto, ya que estoy trabajando constantemente con software libre, y mi meta personal es llegar a ser un buen administrador de servidores. Me gustaria en un futuro estudiar a fondo la seguridad

$$E=mc^2$$



```
<HTML>

<HEAD>
<TITLE> Etiquetas de Formato </TITLE>
</HEAD>
<BODY>

<b>Nombre: Guillermo Tenorio Ruiz </b><br>
<u>Pasatiempo: Leer </u><br>
<i>Lectura Favorita: Filosofia </i><br>
<center>
<FONT color = " #0066FF" size = " 6" > <reg </FONT>
</center>
<hr width = " 30%" size = " 4" noshade >

<p align = " left">
Tomé este diplomado por la necesidad de que requiere mi proyecto, ya que
estoy trabajando constantemente con software libre, y mi meta personal
es llegar a ser un buen administrador de servidores. Me gustaria en un futuro
estudiar a fondo la seguridad .....
```

Este es un primer acercamiento al manejo de los estilos de las páginas Web, se modificara su tamaño, color, y forma a cualquier texto.

```
</p>

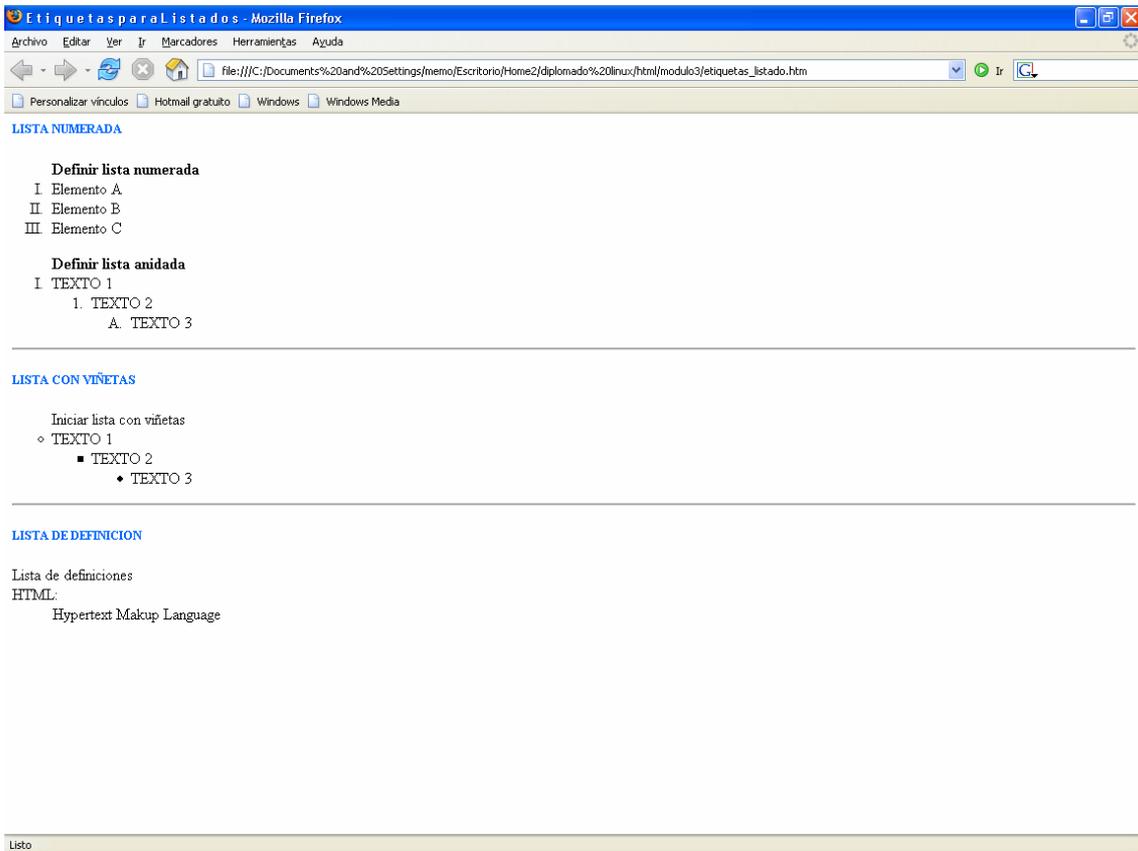
<BLOCKQUOTE>
<FONT color = " #0066FF" size = " 2" >

Tomé este diplomado por la necesidad de que requiere mi proyecto, ya que
estoy trabajando constantemente con software libre, y mi meta personal
es llegar a ser un buen administrador de servidores. Me gustaria en un futuro
estudiar a fondo la seguridad .....
```

```
</FONT>
</BLOCKQUOTE>
<H1>E=mc<sup>2</sup></H1>
```

```
</BODY>
</HTML>
```

• **Listas**



```

Archivo  Editar  Ver
<HTML>
<HEAD>
<TITLE>      E t i q u e t a s   p a r a   L i s t a d o s    </TITLE>
</HEAD>
<BODY>
<H5><FONT color = " #0066FF">LISTA NUMERADA</FONT></H5>

    <b> <ol type = " I " >Definir lista numerada </b>
    <li>Elemento A
    <li>Elemento B
    <li>Elemento C
    </ol>

    <b> <ol type = " I " >Definir lista anidada </b>
    <li>TEXTO 1
        <ol type = " 1">
            <li>TEXTO 2
                <ol type = " A " >
                    <li>TEXTO 3
                </ol>
            </ol>
        </ol>
    </ol>
<hr>
<H5><FONT color = " #0066FF">LISTA CON VIÑETAS</font></H5>

    <ul type = " circle">Iniciar lista con viñetas
    <li>TEXTO 1
        <ul type = " square">
            <li>TEXTO 2
                <ul type = " disc " >
                    <li>TEXTO 3
                </ul>
            </ul>
        </ul>
    </ul>
<hr>
<H5><FONT color = " #0066FF">LISTA DE DEFINICION </font></H5>

    <DI>Lista de definiciones
    <DT>HTML:
        <DD>Hypertext Markup Language
    </DI>
</BODY>
    
```

Ya se han visto las diferentes herramientas que ofrece el lenguaje de marcas HTML para dar formato a los textos, pero hay ocasiones que se quiere presentar en las páginas algún tipo de índice o resaltar los puntos importantes y es aquí donde entra el uso de las ligas.

En la imagen superior se muestra un ejemplo básico de las diferentes listas con algunas de sus características primordiales.

Para cada una de ellas existe una etiqueta y sus opciones, en la imagen de la izquierda se muestra el código para la generación de ligas.

• **Ligas**



[Ejemplo de liga a un archivo del sistema local](#)

[Ejemplo de liga al servicio FTP](#)

[Ejemplo de liga a un archivo de servidor WWW](#)

[Ejemplo de liga al servicio gopher](#)

[Ejemplo de liga al servicio de NEWS](#)

[Ejemplo de liga al servicio de telnet](#)

[Ejemplo de liga al servicio de mail](#)



```
<title>Ejemplos de ligas a servicios</title>
<html>
<font size=5>
<pre>
<a href="/personal.html">Ejemplo de liga a
un archivo del sistema local</a>

<a href="ftp://gauss.matem.unam.mx">Ejemplo de liga al servicio
FTP</a>

<a href="http://www.fciencias.unam.mx">Ejemplo de liga a un archivo de
servidor WWW</a>

<a href="gopher://gauss.matem.unam.mx">Ejemplo de liga al servicio
gopher</a>

<a href="news://132.248.28.3">Ejemplo de liga al servicio de NEWS</a>

<a href="telnet://132.248.28.3">Ejemplo de liga al servicio de
telnet</a>

<a href="mailto:cro@gauss.matem.unam.mx">Ejemplo de liga al servicio
de mail</a>
</pre>
</font>
</html>
```

Una de las características más conocidas de los documentos Web, como se mencionó anteriormente es el manejo de enlaces o ligas haciendo referencia a otros documentos por medio del ya mencionado hipertexto.

Aquí se muestra de manera general la forma en que se logra hacer este tipo de unión entre documentos HTML.

• **Tablas**

UNAM	Mascarones
Diplomado Software Libre	Modulo 3

Celda sobre dos columnas	
Diplomado Software Libre	Modulo 3

Celda que ocupa dos filas	Diplomado Software Libre
	Modulo 3

UNAM	Mascarones
Diplomado Software Libre	Modulo 3

```

<TABLE border height=200>
  <TR>
    <TD valign=bottom>UNAM</TD>
    <TD valign=middle>Mascarones</TD>
  </TR>
  <TR>
    <TD valign=bottom>Diplomado Software Libre</TD>
    <TD valign=middle>Modulo 3</TD>
  </TR>
</TABLE>
<hr>
<TABLE border >
  <TR>
    <TD colspan=2>Celda sobre dos columnas</TD>
  </TR>
  <TR>
    <TD>Diplomado Software Libre</TD>
    <TD>Modulo 3</TD>
  </TR>
</TABLE>
<hr>
<TABLE border >
  <TR>
    <TD rowspan=2>Celda que ocupa dos filas</TD>
    <TD>Diplomado Software Libre</TD>
  </TR>
  <TR>
    <TD>Modulo 3</TD>
  </TR>
</TABLE>

```

Hoy en día y debido al gran manejo de información y división de diferentes tópicos, el uso de las tablas es casi indispensable ya que permite entre muchas cosas redimensionar la estructura de la pagina no importando las resoluciones de los monitores.

Aquí se muestran algunos de los parámetros modificables en las celdas de una tabla, haciendo uso de diferentes etiquetas.

► Aplicación

En este módulo se tomó la decisión de realizar un sitio Web, el cual contenga todos los temas antes mencionados, con los requisitos siguientes:

- 1) Tema libre de contenido amplio
- 2) Un mínimo de 10 páginas Web
- 3) Uso de ligas, tablas, formularios, imágenes, listas.

Por lo cual se realizó un sitio dedicado a 4 escritores mundialmente famosos, de los cuales se presenta su biografía, sus obras, paginas de interés y fotos de cada autor



En el **Index** (página de inicio) se hace uso de las ligas, listas y un pequeño formulario donde se pregunta al usuario sobre el contenido del sitio

Otro aspecto importante es el manejo del estilo, tamaño y color de las letras usadas en nuestro sitio.



La distribución de información es esencial en Internet, es por ello que en este sitio se encuentra una recopilación de material (libros) que están disponibles para toda persona.

El uso de imágenes es indispensable en toda página, es por ello que en este sitio se trato de recrear la vida de los autores por medio de las fotos recopiladas.



MODULO IV

ADMINISTRACION DE SERVIDORES DE WWW CON LINUX

► **Objetivo General:**

El participante conocerá nuevas herramientas administrativas que le permitan desarrollar e implementar sistemas para el control de procesos e información, que funcione de forma natural en red o por Internet, empleando herramientas de software libre que han demostrado tener una alta confiabilidad, alto desempeño y funcionalidad

► **Objetivo Específico:**

El participante identificará el procedimiento para instalar, configurar y administrar su propio servidor de WWW en un servidor de plataforma LINUX

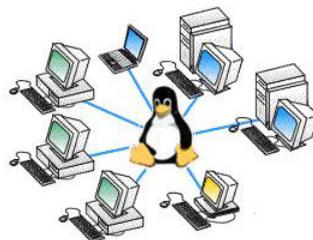
► **Temario:**

1. Introducción a los servidores WWW
2. Instalación del servidor WWW
3. Configuración del Servidor
4. Ejecución del Servidor
5. Incorporación de Módulos
6. Accesos Restringidos
7. Registro de accesos
8. Manejo de Sitios Virtuales

► **Descripción:**

Teniendo en cuenta hasta este punto la información con la que se cuenta y habiendo reconocido y configurado sobre un entorno LINUX, ha llegado el momento de enfrentarse a situaciones totalmente prácticas, esto es instalando, configurando y administrando uno de los servicios más conocidos en el mundo del software libre, que es Servidores Web.

A lo largo de este capítulo se mostrara la manera en como se levanta un servidor Web, desde la obtención del código hasta que esta corriendo y brindando servicio, así como realizar monitoreo y configuraciones avanzadas, incluso hacer uso de herramientas anexas al servidor para hacerlo más estable o seguro.



► Introducción a los servidores WWW

Este apartado está dedicado a hacer un primer reconocimiento de la importancia y uso de un servidor Web, así como identificar el funcionamiento general de WWW y decidir para el Servidor Web, que versión instalar.

En este punto es necesario definir de qué se trata un servidor Web, en la Figura 1 que se muestra a continuación se presenta un diagrama que explica este concepto.

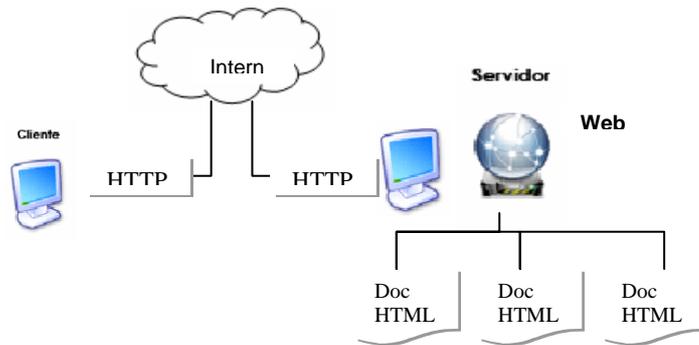


Fig 1. Esquema del servidor Web

El servidor Web es un programa que escucha las peticiones HTTP y se encarga de brindar comunicación mediante este protocolo, dependiendo de dicha petición, el servidor va a responder ya sea buscando la dirección de un documento o la ejecución de algún programa. Este servidor Web es el que brindará servicios dentro del World Wide Web en Internet.

Aunque entender como funciona el protocolo HTTP no es propio de este módulo, es necesario retomarlo a grandes rasgos, sobre todo para un mejor manejo del tema actual. Por lo tanto se analizará como funciona este protocolo.

1. El cliente HTTP abre una conexión
2. El Server manda un "acknowledge" notificando que se ha abierto una petición
3. El cliente envía su "request message" solicitando un recurso.
4. El servidor responde con "response message" que contiene el recurso solicitado y cierra la conexión.

Estos pasos se realizan todas y cada una de las veces que se visualiza una página en el navegador.

• Servidores WWW

Aquí se ha llegado a un punto interesante, ya que se ha notado como es que trabaja un servidor WWW, pero actualmente ¿Cuántos existen?, ¿Cómo saber cual es el mejor?.

A continuación se listará algunos de los Servidores más Populares:

- Internet Information Server – Microsoft
- Sun Java Web Server – Sun Microsystems
- Roxen Web Server – Open Source
- Public Domain HTTP Daemon – Zeus
- Apache Web Server – Open Source

La forma de evaluación y selección de estos servidores se basa de acuerdo a los siguientes criterios o puntos.

- Función del servidor Web
- Reconocimiento de los administradores
- Plataforma disponible
- Número de Conexiones Concurrentes
- Número de Conexiones por Segundo
- Costo Computacional por transacción
- Proyecto de crecimiento esperado
- Soporte para esa tecnología

En el curso se hará uso de Servidor Web Apache, esto debido a que es robusto, soporta un gran número de transacciones, y trabaja bajo diferentes plataformas, cuenta con un alto nivel de seguridad, etc. Actualmente es uno de los Servidores más usados a nivel mundial tanto en empresas como en universidades, como se puede observar en la Figura 2 de abajo.

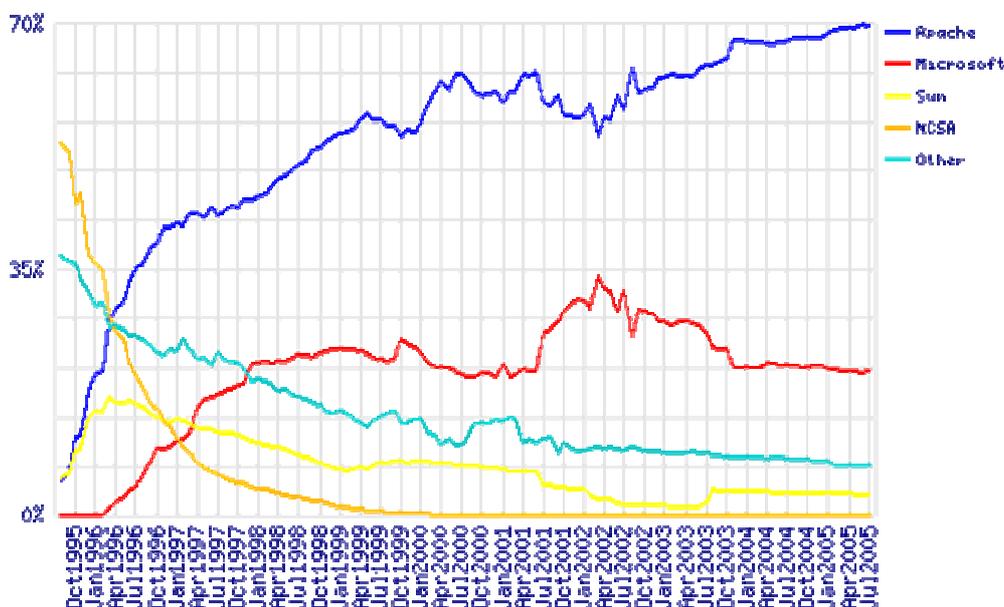


Figura 2. Comparativa de Apache con otros Servidores Web

Top Developers					
Developer	June 2005	Percent	July 2005	Percent	Change
Apache	45172895	69.70	47030635	69.60	-0.10
Microsoft	13131361	20.26	13871645	20.53	0.27
Sun	1849471	2.85	1842812	2.73	-0.12
Zeus	580844	0.90	608121	0.90	0.00

En esta gráfica se muestra un análisis reciente con fecha de Julio 1 del 2005 donde se escanea la red para saber cuáles son los servidores más usados a nivel mundial, siendo Apache el líder.

► Instalación de Apache Server

Para el sistema Linux-Slackware, a continuación se listaran los pasos para instalar: httpd-2.0.54.tar.gz, que se puede descargar gratuitamente de la página oficial de Apache.

- 1) tar -xvzf httpd-2.0.54.tar.gz
- 2) cd httpd-2.0.54/
- 3) ./configure --prefix=/usr/local/apache2
- 4) make
- 5) make install

El proceso de instalación de Apache pareciera resultar bastante fácil incluso para un novato en el área, pero recordando que Apache es un programa bastante robusto que va a crecer en la forma que se le configure para que trabaje con otros recursos de Linux.

Hasta este momento ya se ha logrado que el Servidor Web este trabajando sin ningún problema en modo "default"

► Configuración de Apache

En este apartado se mostraran las directivas más importantes de configuración, que ayudan a hacer estable esta instalación, es aquí donde se comprueba una vez más las ventajas del software libre, haciendo con las instalaciones lo que se requiere.

httpd.conf. Este es el archivo de configuración de Apache, el cual esta formado por más de 200 directivas, las cuales permiten que determinada funcionalidad pueda ser incluida o excluida, en Linux el administrador controla que directivas estarán disponibles de acuerdo a los módulos con los que se compila Apache.

Para un mejor manejo de estas directivas Apache las divide en tres grupos:

- **Global Enviroment:** Administra las directivas generales de operación
- **Main Server:** Administra las directivas del servidor principal o estándar
- **Virtual Servers :** Administra los mismos procesos que soportan diversas *ips* o nombres de dominio

A continuación se muestra una serie de directivas a las cuales se puede cambiarles su valor por default para lograr el objetivo de este apartado. Más adelante en el caso práctico se realizara nlos ejemplos respectivos

- **Server Type:** Manera de operar apache (satandalone/Inetd).
- **User:** Usuario que opera Apache.
- **Group:** Grupo ID para operar Apache.
- **Port:** Puerto por el que opera Apache.
- **ServerAdmin:** Define el correo (contacto) del administrador.
- **DocumentRoot:** Define la ruta donde se almacenan los archivos html.
- **ServerRoot:** Define la ruta de los archivos de configuración y errores.
- **MinSpareServers:** Define el número mínimo de procesos spare.
- **MaxSpareServers:** Define el número máximo de procesos.

- **Startservers:** Indica el número máximos de servidores de inicio.
- **MaxClientes:** Define el número máximo de procesos que podrán ser levantados.
- **ErrorDocument:** Actúa en caso de algún error mandando a un determinado archivo o dirección.
- **PidFile:** Define el lugar donde se almacenara el Prosses ID para el daemon padre.
- **HostNameLookup:** Habilita la resolución de nombres de DNS
- **Options:** Controla las características del servidor para un determinado documento

► Ejecución del Servidor

En este apartado se mostrara la manera de levantar y detener el Servidor Web Apache.

Una vez instalado el Servidor Web es necesario levantar este servicio y esto se logra de la siguiente manera:

```
# /usr/local/apache2/bin/apachectl start
```

Como se puede observar, el **script** encargado de arrancar el servidor es el **apachectl** a este programa se le pueden pasar varios parámetros para su control:

```
apachectl start      (iniciar servicio)
apachectl stop       (detener servicio)
apachectl restart    (reiniciar servicio)
apachectl startssl   (iniciar con soporte SSL)
apachectl configtest (verificar si esta correcta la sintaxis de httpd.conf)
```

Se debe recordar que cada que se realiza un cambio en el archivo de configuración es necesario reiniciar Apache, para que surtan efecto los cambios hechos.

El siguiente paso es hacer una comprobación del demonio de Apache, así como verificar el puerto por el que esta recibiendo las peticiones, para eso se ejecuta en una Terminal los siguientes comandos

```
netstat -antp | grep 80      (muestra servicios de tcp/ip)
ps -fea | grep httpd        (verificar que esta arriba apache)
```

▶ Incorporación de Módulos

Por cuestiones de lógica en el trabajo este apartado queda inconcluso debido a que la incorporación de módulos al **httpd.conf** depende de la instalación de otros paquetes o programas que no son indispensables en este momento.

▶ Accesos

En este apartado se configuraran las bitácoras que permiten evaluar el servidor en cuestión de seguridad o simplemente reportes.

Cuando se habla de Accesos se refiere a los **Logs** o bitacoras, Apache cuenta con dos archivos donde residen las bitácoras

- access.log: Aquí se registran todos los accesos al sitio.
- error.log: Registra los errores que se generan en el servidor.

Para un mejor manejo también se cuenta con un par de Directivas;

- ErrorLogfilename: Define el nombre del archivo para registrar errores
- LogLevel error: Define el nivel de mensajes de error (emerg/alert/crit/error/warm/notice/info/debug)
- Customlog: Define el nombre del archivo donde se almacenaran los registros de acuerdo a un formato (Logformat)

▶ Manejo de Sitios Virtuales

Existen ocasiones que se necesita configurar el Servidor Apache para que responda a diferentes peticiones, es decir, mantener múltiples sitios Web con diferentes nombres y/o IP's, a este servicio se le denomina "Virtual Host". Y es precisamente este el objetivo de éste apartado.

La directiva para configurar estos Sitios Virtuales es:

```
<VirtualHost 132.248.75.10>  
  ServerAdmin root@web.ejemplo.com  
  ServerName www.memo.com  
  DocumentRoot /home/memo/htdocs/  
</VirtualHost>
```

► Aplicación

Debido a que Apache es un Servidor Web altamente configurable la parte práctica está orientada sólo a la configuración de un servidor virtual, así como habilitar los CGI's y manejo de las bitácoras.

Para administrar de manera efectiva un servidor Web, es necesario tener registros de la actividad y el rendimiento del servidor así como de cualquier problema que haya podido ocurrir durante su operación. El servidor HTTP Apache ofrece capacidades muy amplias de registro de este tipo de información.

Para este monitoreo se hace uso de las directivas ErrorLog y LogLevel habilitándolas desde el archivo de configuración propio de apache y definiendo su ubicación en el sistema, la ruta por default es la siguiente:

```
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a
<VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog /usr/remote/apache/logs/error_log

# LogLevel: Control the number of messages logged to the error_log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn
```

• CGI's

CGI (Common Gateway Interface) define una manera de que un servidor Web interactúe con programas externos generadores de contenido, los cuales son a menudo denominados programas CGI o CGI scripts. Este es el modo más común y simple de crear contenido dinámico en un sitio Web.

La directiva ScriptAlias le dice a Apache que un directorio concreto es asignado para programas CGI. Apache asumirá que cada fichero de este directorio es un programa CGI, e intentará ejecutarlo, cuando este recurso particular sea solicitado por un cliente

```
ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"
```

Se podría usar explícitamente la directiva Options, dentro de un fichero de configuración principal del servidor, para especificar que la ejecución CGI esta permitida en un directorio concreto:

```
<Directory /usr/local/apache/htdocs/directoriocualquiera>
    Options +ExecCGI
</Directory>
```

La directiva anterior le dice a Apache que permita la ejecución de ficheros CGI. Se necesitará a su vez decirle al servidor que ficheros son CGI. La directiva AddHandler le dice al servidor que trate todos los ficheros con extensión **.cgi** o **.pl** como programas CGI:

```
AddHandler cgi-script cgi pl
```

EJEMPLO:

```
<directory "/usr/local/apache/htdocs/final">
    Options +ExecCGI
</directory>

    <directory "/usr/local/apache/htdocs/inicial">
        AuthUserFile /usr/local/apache/conf/claves/.htpasswd>
        AuthGroupFile /dev/null
        AuthName "Bienvenidos a mi sitio de Intranet"
        AuthType Basic
        require valid-user

    <directory "/usr/local/apache/htdocs/inicial">
        Options -Indexes
    </directory>

    <directory "/usr/local/apache/htdocs/inicial/practica">
        Options +Indexes
    </directory>
</directory>
```

- **Host Virtual**

Para hacer uso de esta directiva sólo basta con declararla, junto con el nombre del “Nuevo Host”, IP, la ruta de dónde se encontraran las páginas Web, como elementos básicos. Dado que se está declarando otra dirección nueva a la cual Apache va a responder a sus peticiones, ésta también cuenta con el manejo de diferentes módulos como la configuración correspondiente a cada una de ellas.

```
NameVirtualHost 132.248.75.22:8081

    <VirtualHost 132.248.75.22:8081>
        ServerAdmin guillermo@mitla.dcaa.unam.mx
        ServerName guillermo.localhost
        DocumentRoot /home/virtual

    <directory "/usr/local/apache/htdocs/cgi-bin">
        Options +ExecCGI
    </directory>

    <directory "/home/virtual">
        Options -Indexes
    </directory>

</VirtualHost>
```

MÓDULO V
PROGRAMACION PHP

▶ **Objetivo General:**

El participante conocerá nuevas herramientas administrativas que le permitirán desarrollar e implementar sistemas para el control de procesos e información, que funcione de forma natural en red o por Internet, empleando herramientas de software libre que han demostrado tener una alta confiabilidad, alto desempeño y funcionalidad.

▶ **Objetivo específico:**

Proporcionar al participante los conocimientos necesarios que le permitan crear aplicaciones dinámicas e interactivas para la Web utilizando el lenguaje PHP.

▶ **Temario:**

1. Introducción al lenguaje PHP
2. Herramientas Elementales
3. Common Gateway Interface
4. Cómo Diseñar una aplicación CGI
5. Algunas funciones Útiles
6. PHP y la Programación Orientada a Objetos.

▶ **Descripción:**

En este módulo se explorara un lenguaje de programación muy dinámico y empleado en cientos de empresas debido a su portabilidad y gran uso en la Web.

A lo largo de este capítulo se va a introducir una serie de conceptos básicos relacionados con este lenguaje. Así mismo se instalara el software necesario para el seguimiento del módulo.



► Introducción

PHP *Hypertext Preprocessor*: (Preprocesador de Hipertexto), esto es lo que usualmente se encuentra en la mayoría de las definiciones de los libros o el propio Internet, pero esto no es muy claro, con fines prácticos se va a entender a éste como un lenguaje de alto nivel embebido en páginas HTML, es decir que una página PHP es una página HTML pero con marcas especiales, las cuales son interpretadas por el servidor.

El lenguaje PHP es un lenguaje de programación de estilo clásico, con esto se quiere decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser **HTML**, **XML** o **WML**. Esta más cercano a **JavaScript** o a **C**, para aquellos que conocen estos lenguajes.

Pero a diferencia de **Java** o **JavaScript** que se ejecutan en el navegador, PHP se ejecuta en el servidor, eso permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML.

La rápida evolución que ha experimentado PHP ha hecho de él un lenguaje que nos permite hacer todo, dentro de las principales utilidades de PHP están:

- Funciones de Correo Electrónico:
- Gestión de Bases de Datos
- Gestión de Archivos
- Tratamiento de Imágenes
- Encriptación
- Comercio electrónico

► Elementos del lenguaje

- **Sintaxis**

Para iniciar con los primeros pasos de PHP es necesario indicar cuales son las partes de este código escritas en este lenguaje dentro del documento HTML, para ello se hace uso de las siguientes etiquetas:

- `<?php ?>` (etiquetas cónicas de php)
- `<? ?>` (etiquetas cortas)
- `<% %>` (estilo ASP)

Métodos de *impresión de texto*

(echo –solo texto-, print –variables, texto -)

- **Comentarios**

`/* */` Multilínea
`//` Monolínea

- **Variables**

Para el manejo de variables sólo basta con anteponer un signo de “\$” antes de la variable.

Tipos de datos:

- Enteros (número sin decimales)
- Real (Números con decimales)
- Cadenas (variables Alfanuméricas)
- Arrays

- **Operadores:**

Los operadores son componente esencial de cualquier lenguaje de programación. Con ellos se pueden asignar, unir, cambiar o comparar valores de datos, cambiar el flujo del programa, etc.

- **Bloques y sentencias:**

Todo lenguaje de programación dispone de órdenes de control de flujo, que permite al programa *tomar decisiones lógicas* según reciba unos parámetros u otros.

Las posibilidades que ofrece PHP son:

- if/else
- if/elseif/else
- switch
- do/while
- while
- for

► Herramientas Elementales

En este apartado se trabajara con algunas de las herramientas que disponen la mayoría de los lenguajes, es decir, los arreglos, el manejo de las cadenas, manejo de archivos, etc.

- **Arreglos.**

Es un tipo de variable, con la singularidad de que no contiene un único valor, sino un conjunto de valores referenciados con un índice. Aunque al principio puedan parecer poco amistosos, su uso es utilísimo e imprescindible, y de hecho se pueden encontrarlos en todos los lenguajes de programación.

La sintaxis es muy parecida a la usada con el resto de variables, con diferencias en la forma en que el arreglo es creado y lógicamente, la forma en que su valor es recuperado

Se pueden crear arreglos de forma estática usando la construcción php **array()** con la siguiente sintaxis:

```
Array array([index]=>[valor], [index2]=>[valor], ...);
```

- **Cadenas**

Una parte esencial de PHP, como lenguaje orientado al manejo de páginas Web dinámicas, es la manipulación y presentación en pantalla de cadenas de texto (Strings).

En PHP una cadena es cualquier conjunto de caracteres entrecomillados. PHP considera como cadena todo lo que encuentre entre un par de comillas, por eso todas las cadenas deben comenzar y terminar con el mismo tipo de comillas, simples o dobles

```
<?php
$nombre = "Juan";
$fruta = "manzanas";
$accion = $nombre . "come" . $fruta;

//regresaría la frase Juan come manzanas
?>
```

- **Archivos**

Habrá muchas ocasiones en las que interese acceder a ficheros, bien para recuperar sus datos, bien para escribir en ellos. Por ejemplo en contadores de hits, libros de visitas, logs, o simplemente **Web fetching** (recuperar datos de otra Web para incluirlos en la tuya).

- La función utilizada para abrir un archivo en PHP es **fopen**,

```
file_handler=fopen("nombre_de_archivo","modo");
```

- Con **fclose** se cierra el apuntador adquirido sobre un fichero previamente abierto con **fopen**

```
fclose (file_handler)
```

- Con el comando **fwrite** permite escribir en un archivo previamente abierto.

```
Fwrite (file_handler, "texto", long)
```

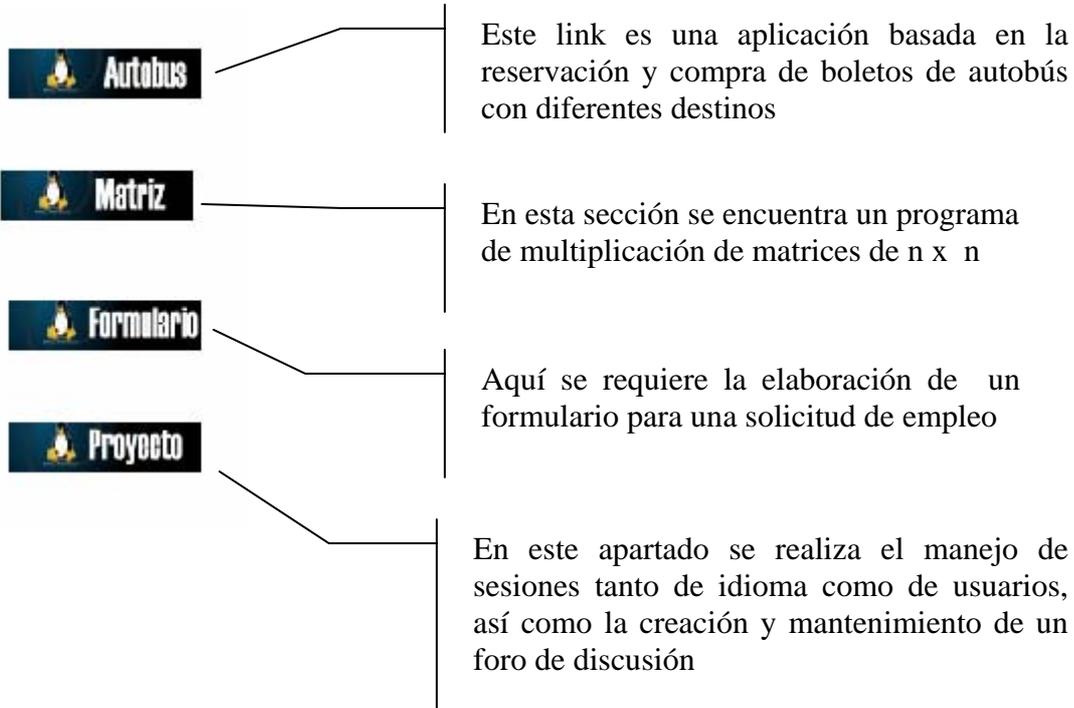
- Con el comando **unlink** se puede borrar un archivo especificado.

```
unlink (archivo)
```


► Aplicación

Para determinar los conocimientos adquiridos hasta el momento se optó por la creación de un sitio Web con diferentes aplicaciones realizadas a lo largo del curso, así como un formulario y elaboración de un foro de discusión.

El sitio cuenta con las 4 secciones siguientes:



The screenshot shows a web form with a blue background and a title bar with a bus image. The form is titled "Linea de Autobuses" and contains the following elements:

- Numero de Boletos:
- Clase de Autobus:
- Origen:
- Destino:
- Enviar
- Limpiar

Esta pequeña aplicación es un ejemplo de la implantación de un sistema de reservación de boletos, el cual calcula el costo ya sea en dólares o pesos, así como el manejo de diferentes tipos de viajes.

Multiplicación de Matrices de $n \times n$

```

    /++_x000L_X_0000T000T00T_000_000_
    00x0000000_0_000000000000000000_00
    Matriz 1:
    /++_x000L_X_0000T000T00T_000_000_
    00x0000000_0_000000000000000000_00
    
```

Filas

Columnas

```

    /++_x000L_X_0000T000T00T_000_000_
    00x0000000_0_000000000000000000_00
    Matriz 2:
    /++_x000L_X_0000T000T00T_000_000_
    00x0000000_0_000000000000000000_00
    
```

Filas

Columnas

En este ejemplo se hace un mayor uso del manejo de las variables, para lograr hacer la multiplicación de matrices.

Solicitud de Empleo

Apellidos: Tenorio Ruiz

Nombres: Guillermo

Domicilio: Isabeles 25

Telefono:

Celular: (55)

Email: guillermo@dcaa.unam

Codigo Postal:

Edad:

Fecha de Nacimiento: dd/mm/aa

Lugar: DF

Nacionalidad: Mexicana

¿Confirma que todos los datos son correctos?

Antes de enviar la solicitud por favor revise
y asegurese de que todos sus datos son correctos

Este es uno de los casos más comunes en muchos sitios de Internet; el llenado de algún formulario y la generación de una base de datos con la información almacenada.

▶ **Objetivo General**

El participante conocerá nuevas herramientas administrativas que le permitan desarrollar e implementar sistemas para el control de procesos e información, que funcionen de forma natural en red o por Internet, empleando herramientas de software libre que han demostrado tener una alta confiabilidad, alto desempeño y funcionalidad.

▶ **Objetivo Específico:**

El alumno conocerá y desarrollara una aplicación de bases de datos que funcione a través de WWW, empleando herramientas de software Libre.

▶ **Temario:**

1. Introducción
2. Manejo de Formularios como FRONT-END
3. Instalación y configuración de la Base de Datos en Linux
4. Introducción al desarrollo de CGI's
5. Desarrollo de BACK-END con MySQL
6. Desarrollo de una Aplicación Personalizada

▶ **Descripción:**

Cabe aclarar que este curso no está orientado al desarrollo y generación de instrucciones SQL, por lo tanto se limitara a crear enlaces de la base de datos a una aplicación Web.

En esta sección se va a hacer un pequeño recorrido por MySQL, por su estructura y forma de trabajar. Para ello se dedica una sección para la instalación y configuración de este manejador junto con Apache y PHP, ya que el objetivo de este curso es la generación de un proyecto completo, es decir, que integre todas las herramientas hasta ahora vistas.

► Introducción:

MySQL es actualmente el sistema de administración de bases de datos SQL más popular en Software Libre, ya que es ampliamente utilizado en los desarrollos Web gracias a su rapidez, estabilidad y a su integración con otros lenguajes de Scripting como por ejemplo PHP.

¿Por qué MySQL?

Esta utilidad se traduce en ventajas, entre las que podemos mencionar las siguientes:

- Acceso a las bases de datos de forma simultánea por varios usuarios y/o aplicaciones.
- Seguridad, en forma de permisos y privilegios, determinados usuarios tendrán permiso para consulta o modificación de determinadas tablas. Esto permite compartir datos sin que peligre la integridad de la base de datos o protegiendo determinados contenidos.
- Potencia: SQL es un lenguaje muy potente para consulta de bases de datos, usar un motor nos ahorra una enorme cantidad de trabajo.
- Portabilidad: SQL es también un lenguaje estandarizado, de modo que las consultas hechas usando SQL son fácilmente portables a otros sistemas y plataformas.

En concreto, usar **MySQL** tiene ventajas adicionales:

- Escalabilidad: es posible manipular bases de datos enormes, del orden de seis mil tablas y alrededor de cincuenta millones de registros, y hasta 32 índices por tabla.
- **MySQL** está escrito en C y C++ y probado con multitud de compiladores y dispone de APIs para muchas plataformas diferentes.
- Conectividad: es decir, permite conexiones entre diferentes máquinas con distintos sistemas operativos. Es corriente que servidores Linux o Unix, usando **MySQL**, sirvan datos para ordenadores con Windows, Linux, Solaris, etc. Para ello se usa TCP/IP, tuberías, o sockets Unix.
- Es multihilo, con lo que puede beneficiarse de sistemas multiprocesador.
- Permite manejar multitud de tipos para columnas.
- Permite manejar registros de longitud fija o variable.

► **Arquitectura Cliente-Servidor**

El término cliente-servidor es un sistema en el que una máquina cliente solicita a una segunda máquina llamada servidor que ejecute una tarea específica, el cliente suele ser una computadora personal común, conectada a una red LAN (Local Area Network) y el servidor es, por lo general, una máquina anfitriona, como un servidor de archivos PC o un servidor de archivos de LINUX.

Por lo tanto el verdadero poder de los sistemas cliente /servidor radica en la gran variedad de aplicaciones cliente y software de desarrollo, llamados también Front-End. Estos Front-End son herramientas de desarrollo, de consultas y de análisis de datos, entre otras.

Los clientes interactúan con el usuario, usualmente de forma gráfica y a su vez se comunican con los procesos auxiliares que se encargan de establecer la conexión con el servidor, de enviar el pedido, de recibir la respuesta, manejar las fallas y realizar actividades tanto de sincronización como de seguridad.

Los servidores proporcionan un servicio al cliente y devuelven los resultados, en algunos casos existen procesos auxiliares que se encargan de recibir las solicitudes del cliente, verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente.

La información es un recurso muypreciado en toda empresa, y como tal se debe procurar que sea segura, íntegra, certera y accesible. Para esto es importante cuidar el medio en el que se maneja.

Proceso de los **Front-End** en una arquitectura cliente/servidor

Las aplicaciones cliente se ven y se ejecutan igual que en cualquier otra aplicación donde el usuario tenga su PC en una estación de trabajo.

Si el software del cliente esta diseñado de manera apropiada, el único indicio de que el usuario está usando un front-end de un servidor remoto de bases de datos se da cuando se tiene que dar tanto su clave como su password para entrar en sesión con dicho servidor.

La secuencia de eventos cuando un usuario accesa al servidor de bases de datos se puede generalizar en los siguientes seis pasos.

- 1 (Cliente) El usuario crea su consulta sobre los datos (Query)
- 2 (Cliente) El front-end formatea la consulta en lenguaje SQL y la envía a través de la red hacia el DBMS
- 3 (Servidor) El servidor de Base de Datos verifica los permisos sobre los datos que desea consultar el usuario (sistema de seguridad)
- 4 (Servidor) Si se cuenta con los permisos correspondientes, el servidor de Base de Datos procesa la consulta y regresa los resultados al front-end
- 5 (Cliente) El front-end recibe la respuesta y la presenta al usuario
- 6 (Cliente) El usuario visualiza y/o manipula los datos y/o reinicia el proceso.

Los seis pasos descritos se pueden ver gráficamente en la figura 1

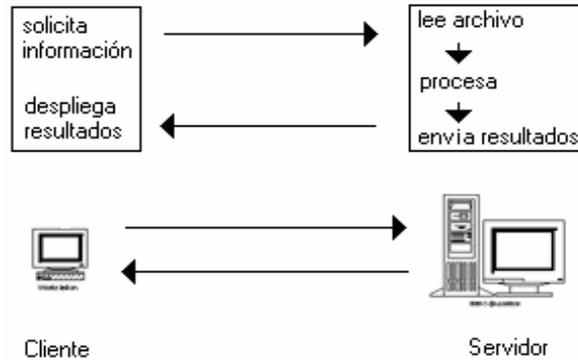


Fig.1 Elementos de conexión a la base de datos

Secuencia de eventos que surgen a partir de que un usuario accesa al servidor de base de datos

La consulta puede ser cualquier acción que el usuario haga sobre la base de datos, como actualizaciones, inserciones, borrados o simples consultas de selección.

Las aplicaciones de los front-end no pueden acceder todas las marcas de bases de datos y esto es por que existen diferentes dialectos de SQL y de la diversidad de protocolos de comunicación. Debido a que SQL no es un estándar, que cada proveedor de bases de datos le agrega extensiones de SQL que hace la incompatibilidad con otras versiones. Además que cada DBMS utiliza un protocolo de comunicaciones distinto para enlazar a los clientes con el servidor de bases de datos.

Front-end: tipo de aplicación que presenta una interfaz al usuario para recolectar datos y después realizar transacciones con uno o varios servidores de aplicación (llamados a su vez Back-end) a fin de desplegar resultados.

► Manejo de Formularios como FRONT-END

Una de las tantas ventajas que tienen las bases de datos y que nos permite una mayor interacción es acceder a ellas mediante aplicaciones Web haciendo uso de diferentes lenguajes de programación siendo el más usado PHP (entre otros).

Como ya se ha mencionado anteriormente, el manejo de formularios en la Web es cada día más común, tanto para hacer encuestas como implementaciones de sistemas muy complejos.

En esta parte del módulo tiene como objetivo crear e implementar una base de datos en MySQL accensando a ella por medio de aplicaciones PHP, con el propósito de conocer el proceso de Front-End.

En este caso cuando hay referencia a FRONT; se hace mención a una aplicación Web (interfaz) con la cual el usuario podrá interactuar de manera remota y transparente ya que no se entera de lo que ocurre dentro del servidor ni como realiza este proceso.

El siguiente concepto es END; el cual se refiere a todas las tareas que se realizan del lado del servidor como la propia conexión a la Base de Datos o la interpretación del código.

► Desarrollo de BACK-END con MySQL

De alguna manera en los módulos anteriores ya se ha desarrollado la interfaz para el usuario, pero no se ha desarrollado del lado del servidor, es aquí donde se aplica la construcción del código que ligue la interfaz con la base de datos. En la Figura 2 ejemplificaremos esta conexión:

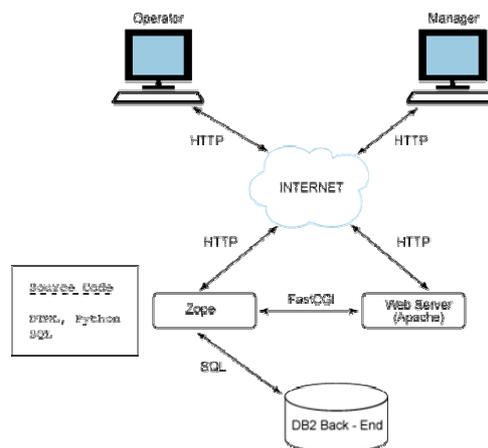


Fig. 2 Configuración y enlace con la Base de Datos

Back-end: Es un proceso de formación que se usa para designar a la unidad que está “detrás” de la computadora y como siguiendo una línea continua de procesamiento: básicamente las bases de datos y las sucesivas capas de software y hardware que regulan y coordinan el procesamiento inteligente de los datos. Hacia el front-end la inteligencia se distribuye y hace más “sensorial” mientras que hacia el back-end se hace más cerebral, tal como se entienden estas analogías humanas.

► **Instalación y configuración de la Base de Datos.**

Las distribuciones MySQL para LINUX suelen estar disponible en tres formatos distintos: Formato binario, formato RPM, formato fuente. Suelen instalarse las dos primeras porque automáticamente asignan los valores por defecto, adecuados a la implementación en la distribución para la plantilla de instalación.

Para instalar MySQL en el servidor se deberá crear una cuenta para el usuario de MySQL y el grupo como cualquier usuario del sistema que permita ejecutar en el servidor.

Ya que MySQL es actualmente distribuido para la mayoría de las versiones Linux, puede instalarse desde los paquetes binarios de tu distribución preferida, en este documento se trabajará sobre la distribución Slackware 10, es recomendable utilizar la versión más actualizada de los paquetes para no estar expuestos a vulnerabilidades, bugs u otro tipo de fallas.

Antes de seguir con la instalación de MySQL se debe tomar en cuenta que el servidor de http debe estar funcionando correctamente y configurado con las variables globales de php.

A continuación se muestra una manera sencilla de levantar el servicio de MySQL desde línea de comando.

```
# cd /etc/rc.d
# chmod 755 /etc/rc.d/mysql
# su - mysql
# mysql_install_db
```

Ahora ya se puede iniciar, de la siguiente manera:

```
# /usr/bin/mysqld_safe &
```

► Aplicación

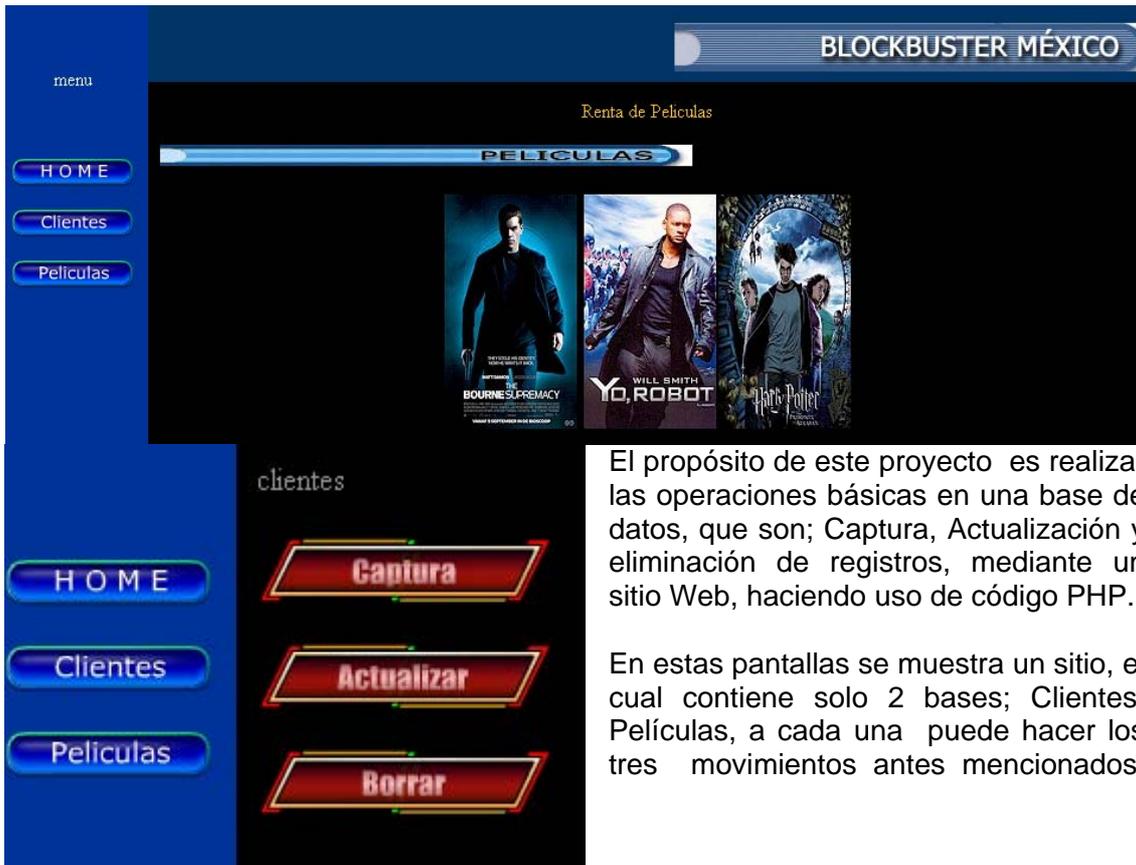
La práctica consta de una aplicación que cuente con las siguientes características:

- Planteamiento de problema
- 2 bases de datos en MySQL
- Programa que permita capturar datos en estas dos bases
- Programa que permita hacer cambios en la BD

A continuación se muestra una parte de la estructura de las tablas creadas para este proyecto:

```
-- MySQL dump 9.11
--
-- Host: localhost      Database: videos
-- -----
-- Server version      4.0.20
--
-- Table structure for table `cliente`
--
DROP TABLE IF EXISTS cliente;
CREATE TABLE cliente (
  id_cliente int(11) NOT NULL auto_increment,
  nombre text,
  direccion text,
  telefono text,
  email text,
  PRIMARY KEY (id_cliente),
  UNIQUE KEY id (id_cliente)
) TYPE=MyISAM;
--
-- Dumping data for table `cliente`
--

/*!40000 ALTER TABLE cliente DISABLE KEYS */;
LOCK TABLES cliente WRITE;
INSERT INTO cliente VALUES (1,'Guillermo Tenorio Ruiz','Isabeles
#24','56416182','neo@org.com.mx'),(2,'Claudia Iveth Astudillo Garcia','Calle 28
#68','57362643','psiquiveth@hotmail.com'),(3,'Alberto Arriaga Arredondo','Benito
Juarez # 36','56287549','alarar@hotmail.com'),(4,'Maydely Tenorio Ruiz','Coyoacan #
98','56847512','maydely@hotmail.com'),(5,'Mariana Costa Rodriguez','Mexica
#30','52137946','mar_12@yahoo.com.mx'),(13,'Marin Lopez Franco','calle 12
#89','78962546','egonlink@hotmail.com');
UNLOCK TABLES;
/*!40000 ALTER TABLE cliente ENABLE KEYS */;
```



El propósito de este proyecto es realizar las operaciones básicas en una base de datos, que son; Captura, Actualización y eliminación de registros, mediante un sitio Web, haciendo uso de código PHP.

En estas pantallas se muestra un sitio, el cual contiene solo 2 bases; Clientes, Películas, a cada una puede hacer los tres movimientos antes mencionados.

► **Objetivo:**

El participante reconocerá la importancia de la seguridad e identificará los elementos que le permitirán proteger el sistema y la información.

► **Temario**

1. Introducción
2. Control de Accesos
3. Sistemas de Archivos
4. Administración Básica de la Seguridad
5. Seguridad en Red
6. Políticas de Seguridad

► **Descripción:**

Este tema es tan amplio que podría ser otro diplomado completo, pero sólo se revisará de manera general algunos conceptos básicos de seguridad, así como el reconocimiento de algunas técnicas de ataque y de protección a equipos remotos.

Actualmente día a día se descubren cientos de vulnerabilidades y se crean otro tanto de herramientas que explotan estos huecos en diversas plataformas y Linux no podía quedar fuera de esta guerra sin cuartel, el hecho de que un sistema sea seguro o no depende del arduo trabajo por parte del administrador, en la figura 1 se muestra una arquitectura básica de protección. Este es el inicio para saber como protegerse ante eventuales intrusos y así realizar un buen trabajo como administradores.

“El único sistema seguro es aquel que está apagado y desconectado, enterrado en un refugio de concreto, rodeado por gas venenoso y custodiado por guardianes bien pagados y muy bien armados. Aún así, yo no apostaría mi vida por él”

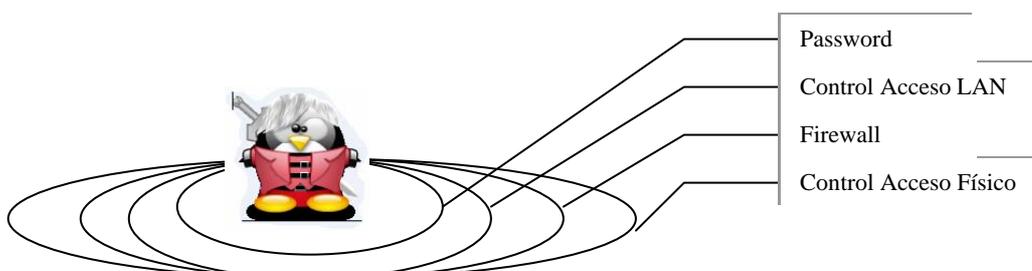


Fig 1. Controles de acceso a un sistema

► **Introducción:**

Este apartado siempre resultara muy teórico pero es indispensable para familiarizarse con este concepto de la seguridad en Cómputo.

La seguridad en cómputo representa el complemento idóneo para los administradores de centros de cómputo, que quieren fortalecer la integridad y disponibilidad de sus sistemas, especialmente en un ambiente de red local y global.

Un sistema de cómputo es seguro si se puede confiar en él, si su software se comporta como se espera que lo haga, y si la información almacenada en él se mantiene inalterada y accesible durante tanto tiempo como su dueño lo determine. La seguridad busca consolidar la confidencialidad, integridad, autenticidad y disponibilidad de la información. De igual forma, la seguridad informática busca mantener y conservar la operatividad de la organización y de sus sistemas a partir del resguardo de sus recursos.

Tomando en cuenta el concepto de Seguridad, esto lleva a determinar los Objetivos de la Seguridad, los cuales se muestran a continuación:

- **Identificación (Autenticación):** Es el proceso de identificar al cliente de la aplicación o servicio. No olvidar que los clientes pueden ser tanto personas, como otros servicios, procesos y otras computadoras
- **Confidencialidad** Consiste en asegurar que a la información sólo accede quien está autorizado para ello.
- **Integridad** Conjunto de acciones que garantizan que la información no se ha transformado durante su procesado, transporte o almacenamiento.
- **No repudio** Procedimientos para asegurar que ninguna de las partes implicadas ya identificadas (autenticadas) puede negar haber participado en una determinada transacción.
- **Autorización** Determinar a qué información puede acceder y qué tareas puede acometer, un cliente autenticado por lo tanto identificado con certeza. Este proceso determina los privilegios asociados a un perfil de usuario. Cuestión organizativa, que debe diseñar cada organización y llevar a cabo en sus sistemas particulares.
- **Auditoria** Es la posibilidad de poder rastrear los accesos realizados a la información y las operaciones hechas sobre ella por cada usuario y las circunstancias en que las hizo.
- **Disponibilidad.** Forma parte de la seguridad el poder disponer de la información cuando se necesite. Por ello se deben proteger los sistemas de forma que se mantengan en funcionamiento y se pueda acceder a la información en cualquier momento.

Si se logran mantener estos puntos se puede decir en general que los datos del sistema están protegidos y seguros.

Y a todo esto la pregunta es, ¿Contra qué o quién protegese?, y para responder a esta pregunta se analizan los siguientes conceptos.

Amenaza: La definición de amenaza en un sistema de cómputo, no difiere de la vida real, es decir, es toda aquella situación (fuera de nuestro alcance) que puede causar daño, violando los tres pilares de la información (confidencialidad, integridad o disponibilidad)

Vulnerabilidad: Simplemente se entiende este concepto como una debilidad del Sistema, y esta puede poner en riesgo los datos.

Riesgo: Este se va a medir en cuanto al potencial de pérdida o fallo, basado en los daños ocasionados ante una eventual catástrofe de datos.

Ataques: Un ataque sólo tiene un objetivo en el mundo y este es hacer que una posible víctima deje de funcionar de acuerdo con un propósito ya establecido

Exploit: Simplemente es la forma de explotar una vulnerabilidad

► **Control de Accesos**

En este apartado se reconocerá la importancia del control que se debe tener de los usuarios del Sistema, ya que en un sistema multiusuario no se puede confiar absolutamente en nadie. Es por ello que a continuación se enlistan los principios de esta sección

- El intruso probablemente es alguien conocido
- No confíes, y se cauteloso con quien requiera tu confianza
- No confíes en ti mismo, o verifica lo que haces.
- Haga que el intruso crea que será atrapado.

Si alguien que desee atacar un sistema tiene acceso físico al mismo todo el resto de medidas de seguridad implantadas se convierten en inútiles.

Para evitar todo este tipo de problemas se deben implantar mecanismos de **prevención** (control de acceso a los recursos) y de **detección** (si un mecanismo de prevención falla o no existe se debe al menos detectar los accesos no autorizados cuanto antes).

Para la **prevención** hay soluciones para todos los gustos y de todos los precios:

- analizadores de retina,
- tarjetas inteligentes,
- videocámaras,
- vigilantes jurados

En muchos casos es suficiente con controlar el acceso a las salas y cerrar siempre con llave los despachos o salas donde hay equipos informáticos y no tener cableadas las tomas de red que estén accesibles.

Para la **detección** de accesos se emplean medios técnicos, como cámaras de vigilancia de circuito cerrado o alarmas, aunque en muchos entornos es suficiente con que las personas que utilizan los sistemas se conozcan entre si y sepan quien tiene y

no tiene acceso a las distintas salas y equipos, de modo que les resulte sencillo detectar a personas desconocidas o a personas conocidas que se encuentran en sitios no adecuados.

Este control de acceso esta respaldado por dos bloques, el primero son los mecanismos de prevención y el segundo son los mecanismos de autenticación:

- **Mecanismos de Prevención**

Aumentan la seguridad de un sistema durante el funcionamiento normal de éste. Previenen la ocurrencia de violaciones a la seguridad

Ejemplos mecanismos:

- Encriptación durante la transmisión de datos.
- Passwords difíciles
- Firewalls

- **Mecanismos de autenticación**

Hacen posible identificar entidades del sistema de una forma única, posteriormente, una vez identificadas, son autenticadas (comprobar que la entidad es quién dice ser)

Basados en algo que se sabe

• Passwords, frases y números de identificación personal, NIP, Siguen siendo el sistema de autenticación más usado hoy en día.

Basados en algo que se es

• Biométricas y Comportamiento, se realiza una medición física y se compara con un perfil almacenado con anterioridad.

Basadas en algo que se tiene

• Usar un objeto físico que llevan consigo y que de alguna forma comprueba la identidad del portador; tokens, tarjetas inteligentes y pases.

- Passwords

A continuación se muestran algunas recomendaciones respecto al password, que como es bien sabido es la llave de acceso al sistema, pero sino se tiene las medidas necesarias ésta puede convertirse en la llave de acceso de muchos usuarios mal intencionados. Es por ello que es de suma importancia elegir una contraseña que no se tan fácil de romper.

- NO usar el mismo login
- NO usar nombres propios, apellidos o apodos
- NO usar nombres de familiares, amigos, etc.
- NO usar palabras contenidas en diccionarios
- NO usar información relacionada con usted
- NO usar solamente Números o letras
- NO usar menos de 7 caracteres
- Usar mayúsculas, minúsculas, números, caracteres especiales.

► Sistemas de Archivos

En este apartado se vera otra ventaja que ofrece Linux; la protección de los diferentes archivos y es que si no se tiene un control de éstos, cualquier usuario podría causar daños irrevocables.

Para brindar algo de privacidad y protección cada archivo o directorio tiene asociados permisos diferentes para el dueño, para el grupo y para los demás usuarios. En el caso de archivos los permisos que pueden darse o quitarse son: (r) lectura, (w) escritura y (x) ejecución. En el caso de directorios los permisos son: (r) para listar los archivos, (w) para escribir, crear o borrar archivos y (x) para acceder a archivos del directorio

Los permisos de un archivo pueden ser modificados por el dueño o por el administrador del sistema con el comando **chmod** que espera dos parámetros: cambio por realizar al permiso y nombre del archivo por cambiar. Los permisos se pueden especificar en octal o con una o más letras para identificar al usuario (“**u**” para el usuario, “**g**” para el grupo, “**o**” para los demás usuarios y **a** para todos), un “**+**”, un “**-**” ó un “**=**” y después letras para identificar los permisos (“**r**”, “**w**” ó “**x**”).

Por ejemplo

chmod og+x hack.sh

Da a los demás usuarios y al grupo permiso de ejecución del archivo hack.sh

Es conveniente tener claros los permisos que se pueden asignar a un fichero o directorio. Puede que algunas aplicaciones no funcionen bien si algún fichero no tiene el permiso o el propietario correctos, bien por falta de permisos o bien por exceso

► Administración Básica de la Seguridad

En este apartado se expondrán algunas de las tareas y recomendaciones básicas que se tienen como administrador de sistemas Linux y mantener a los usuarios y paquetes vigilados en todo momento:

Básicamente se partirá de los siguientes puntos:

- Políticas de cuentas y contraseñas
Como tal las políticas son las que van a definir que tan seguro será el sistema, ya que aquí se especificaran las propiedades y características de los servicios brindados
- Monitoreo del sistema
Actualmente existen cientos de herramientas para este fin, incluso el propio Linux hace uso de algunos comandos o programas para estar escaneando los procesos, pero se recomiendan los siguientes pasos:
 - Monitorear los procesos que están ejecutándose (haciendo uso de scripts externos.
 - Evitar Accesos no autorizados al sistema por medio de detectores de intrusos
 - Deshabilitar los servicios que no se estén ocupando.

- Implementación y revisión de Bitácoras
- Las bitácoras son respaldos fiables en formato texto donde se guardan los reportes de error, de acceso, de fallas, etc, es por ello que su contenido es valioso y es uno de los primeros archivos en ser atacados en caso de que el sistema haya sido comprometido por algún intruso. Dado estas circunstancias es recomendable guardar estas bitácoras en una máquina diferente al servidor principal.

► Seguridad en Red

El propio desarrollo tanto de Linux, como de la mayoría del software que lo acompaña, es de fuentes abiertas. Se puede ver y estudiar el código. Esto tiene la ventaja de que la seguridad en Linux no sea una mera apariencia, sino que el código está siendo escrutado por muchas personas distintas que rápidamente detectan los fallos y los corrigen con una velocidad asombrosa.

Si además se comprenden los mecanismos que se siguen en las conexiones en red, y se mantienen actualizados los programas, se podrá tener un nivel de seguridad y una funcionalidad aceptables.

Tampoco tienen las mismas necesidades de seguridad un equipo doméstico, con conexiones esporádicas a Internet, que un servidor conectado permanentemente y que actúe como pasarela entre una intranet e Internet.

Para describir las pautas de actuación seguras se debe conocer sólo algunos tipos de ataques a sistemas remotos.

- Pasivos
 - Sniffers: Estos programas se encargan de “olfatear” el tráfico de la red
 - Traceroute
 - Google
- Activos
 - DoS
 - Consumo de Ancho de Banda
 - Inanición de recursos
 - Defectos de Programación
 - Paquetes mas formados
 - Ataques DNS
 - SQL –Injection
 - XSS
 - Buffer OverFlows
 - Format String
 - Spoofing: Es la creación de paquetes de comunicación TCP/IP usando la dirección IP de alguien más.

► Políticas de Seguridad

Resulta importante establecer políticas de seguridad, las cuales van desde el monitoreo de la infraestructura de red, los enlaces de telecomunicaciones, la realización del respaldo de datos y hasta el reconocimiento de las propias necesidades de seguridad, para establecer los niveles de protección de los recursos.

- Especifica las características de seguridad que un sistema debe observar y proveer conjunto de reglas que deben respetarse para mantener la seguridad de la información.
- Especifica las amenazas contra las que la organización debe protegerse y cómo debe protegerse
- Depende de los objetivos y metas de la organización.
- Generalmente es expresada en un lenguaje o idioma.
- Típicamente establecida en términos de sujetos y objetos.

Las políticas deberán basarse en los siguientes pasos:

- Identificar y seleccionar lo que se debe proteger (información sensible)
- Establecer niveles de prioridad e importancia sobre esta información
- Conocer las consecuencias que traería a la compañía, en lo que se refiere a costos y productividad, la pérdida de datos sensibles
- Identificar las amenazas, así como los niveles de vulnerabilidad de la red
- Realizar un análisis de costos en la prevención y recuperación de la información, en caso de sufrir un ataque y perderla
- Implementar respuesta a incidentes y recuperación para disminuir el impacto

Este tipo de políticas permitirá desplegar una arquitectura de seguridad basada en soluciones tecnológicas, así como el desarrollo de un plan de acción para el manejo de incidentes y recuperación para disminuir el impacto, ya que previamente se habrá identificado y definido los sistemas y datos a proteger.

Es importante tomar en consideración, que las amenazas no disminuirán y las vulnerabilidades no desaparecerán en su totalidad, por lo que los niveles de inversión en el área de seguridad en cualquier empresa, deberán ir acordes a la importancia de la información en riesgo.

Así mismo, cada dispositivo que conforma la red empresarial necesita un nivel de seguridad apropiado y la administración del riesgo implica una protección multidimensional (firewalls, autenticación, antivirus, controles, políticas, procedimientos, análisis de vulnerabilidad, entre otros), y no únicamente tecnología.

Un esquema de seguridad contempla la seguridad física y lógica de una compañía. La primera se refiere a la protección contra robo o daño al personal, equipo e instalaciones de la empresa; y la segunda está relacionada con la protección a la información, a través de una arquitectura de seguridad eficiente.

MODULO VI

PROGRAMACION CON JAVA

▶ **Objetivo General**

El curso está dirigido a toda persona interesada en desarrollar aplicaciones independientes de la plataforma, con un paradigma orientado a objetos utilizando el lenguaje Java. Se requiere haber cursado lenguaje de programación estructurada con C, además cuente con conocimientos básicos de programación orientada a objetos.

▶ **Objetivo Especifico:**

El participante identificará los principales componentes del lenguaje Java, que le permitan ir incorporando los conocimientos adquiridos sobre el lenguaje, durante el curso.

▶ **Temario:**

1. La tecnología JAVA
2. Estructura del lenguaje
3. Conceptos de la programación orientada a objetos
4. Trabajando con objetos en java

▶ **Descripción:**

Con este módulo de Java que se tomó como anexo al Diplomado de Software Libre se pretende contar con otro lenguaje de programación que facilite la interacción entre el usuario y una base de datos vía Web, así como manejo de elementos interactivos gráficos. Dado que es un curso básico el enfoque será reconocer la sintaxis y métodos de este lenguaje orientado a objetos.

► Introducción

Java es un lenguaje sencillo y orientado a objetos que permite desarrollar aplicaciones en muy diversas áreas tales como: seguridad, animación, acceso a bases de datos, aplicaciones cliente-servidor, interfaces gráficas, páginas de Web interactivas, entre otras. Una de sus principales características es la creación de módulos reutilizables que funcionan sin necesidad de conocer su estructura interna, permitiendo al usuario añadir nuevos módulos. Así como la obtención de programas independientes de la plataforma en la cual fueron desarrollados.

► La tecnología JAVA

La tecnología Java consta de un lenguaje de programación y una plataforma.

El lenguaje de programación Java

Java es un lenguaje de programación de alto nivel que tiene las siguientes características:

- Orientado a objetos
- Robusto
- Seguro
- Multitarea
- Portable

La mayoría de los lenguajes de programación se caracterizan por ser interpretados o compilados, lo que determina la manera en como serán ejecutados en una computadora.

Java tiene la característica de ser al mismo tiempo compilado e interpretado, figura 1. El compilador es el encargado de convertir el código fuente de un programa en un código intermedio llamado *bytecode* que es independiente de la plataforma en que se trabaje y que es ejecutado por el intérprete de Java que forma parte de la Máquina Virtual de Java.

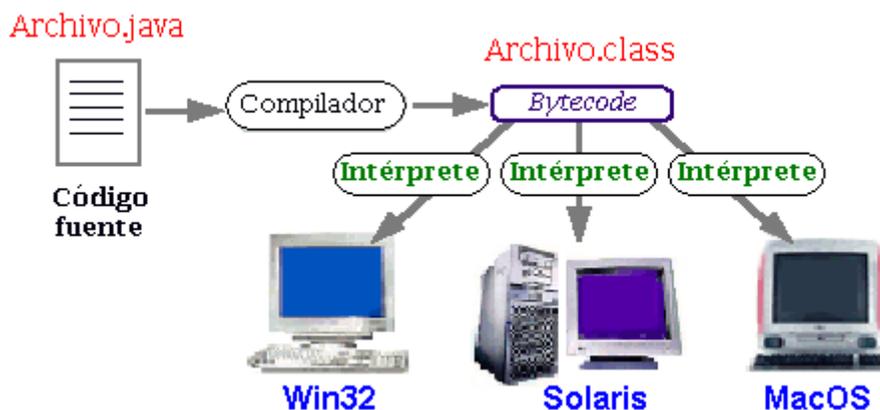


Fig1. Compilación y ejecución de programas en Java.

- **La plataforma Java**

Una plataforma es el ambiente de hardware o software en el cual se ejecutan los programas. En general, la mayoría de las plataformas pueden ser descritas como una combinación de hardware y sistema operativo. Algunas de las plataformas más populares son Windows, Solaris, Linux y MacOS.

La plataforma Java difiere de las anteriores en que ésta es una plataforma basada únicamente en software que corre por encima de las plataformas basadas en hardware.

La plataforma Java consta de dos componentes:

- **La Máquina Virtual de Java (JVM)**
- **La Interfaz de Programación de Aplicaciones de Java (API Java)**

A continuación en la figura 2 se muestra una representación de los elementos que forman parte de la plataforma Java.

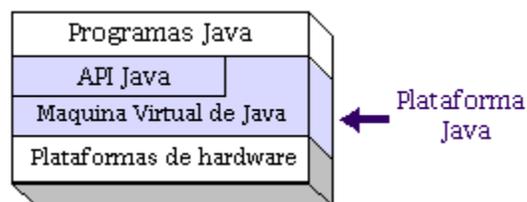


Figura 2. Elementos de la plataforma Java

- **Compilación y ejecución de programas Java**

El Kit de desarrollo de Java (JDK) contiene las herramientas y librerías necesarias para crear y ejecutar applets y aplicaciones en Java.

A continuación se listan algunas de las utilidades que se pueden encontrar en el JDK:

- **javac.** Es el compilador de Java. Se encarga de convertir el código fuente escrito en Java a *bytecode*.
- **java.** Es el intérprete de Java. Ejecuta el *bytecode* a partir de los archivos class.
- **appletviewer.** Es un visor de applets. En la mayoría de las ocasiones puede utilizarse en lugar de un Navegador Web.
- **avadoc.** Se utiliza para crear documentación en formato HTML a partir del código fuente Java y los comentarios que contiene.
- **javap.** Es un desensamblador de Java.
- **jar.** Es una herramienta utilizada para trabajar con los archivos JAR.

- **Compilación y ejecución de programas Java**

- **Obtención del J2SE SDK.**

El J2SE SDK (Java 2 Standar Edition Software Development Kit) se puede obtener directamente de forma gratuita del sitio Web de Java, <http://java.sun.com/> .

- **Instalación y configuración del J2SE SDK (MS Windows)**

El JDK normalmente se distribuye en archivos de instalación auto-extraíbles, por lo que al momento de llevar a cabo la instalación, básicamente se tiene que indicar el directorio en el cual se colocarán los archivos y directorios que se distribuyen en el JDK.

Una vez instalado el JDK se tiene que actualizar la variable de ambiente **PATH** y opcionalmente poner una nueva variable llamada **JAVA_HOME**.

Por ejemplo, si el JDK fue instalado en C:\j2sdk1.4.2, se hace lo siguiente:

```
set PATH=C:\j2sdk1.4.2\bin;%PATH%
set JAVA_HOME=C:\j2sdk1.4.2
```

Adicionalmente el uso de la variable de ambiente **CLASSPATH** le dice a la Máquina Virtual de Java y otras aplicaciones de Java en donde buscar las clases que se necesitan para ejecutar un programa, tal como el archivo *rt.jar*, que contiene una librería de clases que conforma el API de Java.

Es necesario actualizar la variable **PATH** para que el sistema operativo sepa en donde encontrar los programas que se ejecutan cuando se trabaja con Java ,básicamente el compilador y el intérprete de Java, **javac** y **java**, respectivamente.

- ▶ **Estructura del Lenguaje**

Sintaxis de Java

La sintaxis de un lenguaje define cómo se usarán las palabras claves, los operadores y las variables para construir y evaluar expresiones. La sintaxis de Java especifica como se escribirán los siguientes elementos:

- Comentarios.
- Identificadores.
- Palabras clave.
- Literales.
- Expresiones y operadores.
- Variables y tipos de datos.
- Bloques y sentencias.

► Conceptos de la programación orientada a objetos

Para destacar que al contrario de otros lenguajes como C++, JAVA no es un lenguaje modificado para poder trabajar con objetos sino que es un lenguaje creado para trabajar con objetos desde cero. De hecho, todo lo que hay en Java son objetos.

Todo puede verse como un objeto. Un objeto, desde este punto de vista, puede verse como una pieza de software que cumple con ciertas características:

- Encapsulamiento
- Herencia

Encapsulamiento significa que el objeto es auto-contenido, o sea que la misma definición del objeto incluye tanto los datos que éste usa (*atributos*) como los procedimientos (*métodos*) que actúan sobre los mismos.

Cuando se utiliza programación orientada a objetos, se definen *clases* (que definen objetos genéricos) y la forma en que los objetos interactúan entre ellos, a través de *mensajes*. Al crear un objeto de una clase dada, se dice que se crea una *instancia* de la clase, o un objeto propiamente dicho. Por ejemplo, una clase podría ser "autos", y un auto dado es una *instancia* de la clase.

La ventaja de esto es que como no hay programas que actúen modificando al objeto, éste se mantiene en cierto modo independiente del resto de la aplicación. Si es necesario modificar el objeto (por ejemplo, para darle más capacidades), esto se puede hacer sin tocar el resto de la aplicación... lo que ahorra mucho tiempo de desarrollo y debugging. En Java, inclusive, ni siquiera existen las variables globales.

Para tener una idea, se va a poner un ejemplo de una clase Java:

```
public class Muestra extends Frame {
// atributos de la clase
Button si;
Button no;
// métodos de la clase:
public Muestra () {
Label comentario = new Label("Presione un botón", Label.CENTER);
si = new Button("Sí");
no = new Button("No");
add("North", comentario);
add("East", si);
add("West", no);
}}
```

Proyecto de un Sitio Web Basado en el Diplomado de
Desarrollo e Implementación de Software Libre Linux

Capítulo 2

Informe: Aula Virtual - Curso Linux.**► Planteamiento del Problema**

Gracias a la creciente evolución, flexibilidad y aportación de plataformas basadas en Software libre Linux, gran cantidad de empresas e instituciones han decidido migrar sus sistemas comerciales a esta opción libre.

Con ello también es necesario capacitar al personal que va a administrar estas plataformas, pero aquí se presenta el primer obstáculo que provoca que muchas empresas regresen a su anterior sistema, este problema es que al nuevo usuario se le complica, entre muchas otras cosas, la manipulación o configuración de un sistema Linux debido a que su manejo no es tan amigable como al que estaba acostumbrado administrar, pero esto es totalmente un mito que debe desaparecer, ya que el único requisito para poder entrar al mundo del software Libre es un interés y preocupación especial en la integridad de los sistemas, así como desarrollar proyectos que aporten mejores herramientas de trabajo.

Es el caso de muchos administradores de sistemas que ellos no eligieron esa orientación profesional, la elección les vino dada. Al contrario que muchas otras áreas de la industria de la informática, como la ingeniería de software o el diseño de circuitos integrados, hay muy pocas, si es que hay alguna, escuela o facultad que ofrezca la administración de sistemas libres como una de sus titulaciones. Es por ello que muchos de los mejores administradores son autodidactas.

En la mayoría de los cursos de LINUX se muestra de manera general el manejo de comandos básicos o la programación de pequeños scripts, aquí el siguiente problema que hay que enfrentar todos los que se inician en este medio, resulta muy complejo entender las definiciones o tecnicismos que contiene el mismo sistema, esto se debe en parte a la preparación y estructuración del curso por parte de los instructores, y a la falta de ejemplos que expliquen de un modo explícito y a nivel usuario el funcionamiento de cada proceso del sistema.

Existe muchos medios de aprendizaje, como manuales, tutoriales, etc. que ayudan al administrador a realizar sus tareas de manera general, pero la gran mayoría de estos se muestran como recetas de cocina, que llegan a resolver el problema en el momento, pero que, lo único a lo que ayudan es a crear lagunas en el usuario, ya que éste no sabe realmente que es lo que pasa internamente. Se está tan acostumbrado a hacer las cosas tan mecánicamente que se olvida que lo más importante es conocer el funcionamiento y el por qué de los procesos.

El nivel de conocimientos que se alcance en cuanto a estos sistemas depende totalmente de lo autodidacta que sea el administrador, ya que para manipular este tipo de sistemas es necesario practicarlo constantemente.

▶ **Objetivo General**

Creación de una Aula Virtual orientada a Cursos Linux en apoyo a actividades docentes como alternativa y/o solución educativa en línea y presencial, basada en Software Libre.

▶ **Objetivos Particulares**

- Elaboración de un Sitio Web con material de apoyo al usuario Linux que contenga un temario detallado enfocado a la implementación, configuración y administración de plataformas libres.
- Crear el interés en los participantes mostrando las ventajas en el manejo e implementación de software libre
- Proporcionar un lenguaje amigable que ayude la comprensión del tema, sin que esto quite contenido a los tópicos, ya que el curso está enfocado a todo tipo de usuarios.
- Uso de herramientas como el chat, los foros y el correo electrónico para favorecer la comunicación y la participación de los usuarios, a su vez se designan tareas y actividades que facilitan la evaluación formativa de los participantes
- Elaboración de un ambiente amigable, configurable y flexible donde el participante cuente con información, noticias de las actividades y proyectos del curso.
- Creación de ejemplos y ejercicios (explícitos) en cada sección del temario
- Implementación de exámenes o trabajos de investigación para que el usuario evalúe su rendimiento y avance en el curso.

► **Justificación**

Existen diversas materias o temáticas que por si solas (ya sea por sus tecnicismos o complejidad en profundidad) son difíciles de enseñar o aprender, como es el caso de las ciencias exactas o el aprender un nuevo idioma; por mencionar algunas, pero para ambos casos se diseñaron métodos de enseñanza-aprendizaje sustentados por arduas investigaciones que facilitan dicho proceso de aprendizaje.

Es bien conocido que el aprendizaje esta en función de la relación sujeto-mundo, actualmente nos encontramos, en pleno auge de las Nuevas Tecnologías de la Información y las Comunicaciones, lo cual provoca un mejor manejo de la información para las nuevas generaciones.

La integración de las Nuevas Tecnologías Libres en el proceso de enseñanza-aprendizaje crea ambientes innovadores de aprendizaje permitiendo el desarrollo de modelos y metodologías didácticas, de prototipos y materiales didácticos y la formación de comunidades académicas. Todo esto provoca la modernización de la práctica docente y la creación de ambientes virtuales de aprendizaje; elevándose por tanto el trabajo colaborativo.

De ahí que sólo con la tecnología no basta, es importante adiestrar a los docentes para que después ellos propongan y desarrollen nuevas estrategias didácticas, cambiando su rol, para así incorporar plenamente y con ventajas las nuevas herramientas Digitales con las que cuentan. A continuación se muestran algunas de las ventajas de estas herramientas:

- Educación síncrona y asíncrona
- Favorece el trabajo cooperativo
- Trae el mundo al salón de clases y lleva el aula al ámbito global.
- Nuevos esquemas de gestión de conocimientos
- Soporte para implementar nuevos métodos de enseñanza-aprendizaje e investigación.

En ese punto han surgido diversos términos como manejo de las nuevas soluciones educativas; Aulas Virtuales, E-training, E-learning

La educación en línea esta creciendo en popularidad hoy en día, ya sea por una experiencia de aprendizaje mixta, con instrucción en un salón de clase como parte del entrenamiento o como una alternativa autodidacta.

El hecho es que este medio nos da razones de peso que favorecen a su exploración e implementación

- Permite seguir un ritmo de estudio personal, sin perder el apoyo de los tutores
- Se establece un seguimiento riguroso del alumno y se lleva a cabo una evaluación formal de sus aprendizajes.
- Flexibilidad en los plazos y en la dedicación a la formación

- Inmediatez: cualquier cambio introducido en el curso: el programa, los materiales, las actividades, etc., está disponible instantáneamente para los participantes
- La gran disponibilidad de recursos: Internet dispone de grandes masas de información, muchas de ellas de acceso libre, que con la debida orientación y selección rigurosa pueden ser de gran utilidad en el aprendizaje.
- Facilita un material de estudio que se convierte en una fuente de consulta profesional.
- Favorece la autorregulación personal que incide sobre aspectos de la propia personalidad, como el autocontrol o la disciplina
- Facilita el intercambio con profesionales del mismo sector, o con intereses similares

Con todo lo ya mencionado se busca puntualizar y anexar varios procesos en el aprendizaje, de tal manera que se visualice la misión del actual proyecto.

Este modelo no va tampoco acorde con lo que se piensa de que el alumno va a trabajar solo, sino que estará respaldado por el trabajo presencial, lo cual facilita el aprendizaje del alumno, este proceso tiene un mayor balance y una carga sobre el aprendizaje pero la enseñanza queda hacia lo que es la capacidad cognitiva del alumno para aprender, o sea sus habilidades de pensamiento. Aquí un resultado que se espera obtener es que el rol del profesor cambia, ya no es el profesor tradicional sino el asesor, el tutor y el que va a estar guiando la información

► **Desarrollo**

Teniendo en cuenta las ventajas que proporciona el Software libre como creación de nuevos sistemas novedosos, y las ventajas de una computadora, actualmente se han creado diversos sistemas de enseñanza en línea: es por ello que se puede explotar y aprovechar e integrar todos estos recursos al proceso educativo tradicional, en el proceso docente educativo y en la capacitación, así como para desarrollar material didáctico innovador.

• **Opciones educativas en línea.**

Como ya se menciona anteriormente, se cuentan con diferentes técnicas de enseñanza a distancia como opciones didácticas, a continuación se analizará cual de ellas ofrece mejores ventajas en relación a nuestro Curso para su futura implementación.

- **E-training**

E-training es un nuevo concepto en capacitación corporativa e institucional, que ofrece a las compañías la posibilidad de formar a sus empleados en temas de últimas tendencias o desarrollos técnicos necesarios para los desempeños profesionales.

Como su nombre lo indica e-training significa entrenamiento vía web o desde un sentido más completo de la palabra Capacitación vía web.

Se divide en dos grandes bloques: La descripción de "contenidos técnicos específicos" por área/puesto/función, y su aplicación instruccional dirigida a los usuarios del sistema. Con lo cuál se pretenden transformar conocimientos y mentalidades a nivel corporativo complementando las competencias técnicas y profesionales que se manejan en dichos ámbito

- Aula Virtual

Un Aula Virtual es un espacio en la red reservado para cursos de formación. En estas aulas se crea un ambiente similar al de las soluciones e-learning y se trabaja con un sistema instruccional paralelo al de e-training, pero en ambos casos su desarrollo es a menor escala.

Las aulas virtuales reducen notablemente los costos de la formación ya que se enmarcan como el desarrollo de una temática específica. Proporcionan ambientes de aprendizaje en los mismos espacios de trabajo induciendo al trabajo colaborativo entre los usuarios de la solución.

Módulo de Directorio, que le permite gestionar la información de los diferentes participantes del proceso. En él se registrarán los perfiles de los usuarios y el inventario de recursos y herramientas a los cuales se tendrán acceso.

Módulo de Temas y Actividades, en el que se incluyen la descripción general de los temas que serán abordados en los cursos, su justificación pedagógica y la descripción detallada de las actividades a realizar por parte de los usuarios en sus diferentes niveles de responsabilidad

- E-learning

Básicamente facilita la distribución de materiales formativos y herramientas de comunicación para crear un entorno para el aprendizaje. Mediante esta tecnología el alumno tiene acceso a cursos interactivos y multimedia en formato Web, apoyado con medios de comunicación que permiten la colaboración y discusión online de la materia estudiada.

e-learning responden a dos categorías: sincrónicas, imitando un salón de clases en tiempo real y con instructores y alumnos en constante comunicación, o asincrónicas lo cuál deja a los estudiantes un paquete prediseñado de soluciones con las cuales se podrá autoregular en tiempo y espacio.

La solución de e-learning pretende utilizar el Modelo Interactivo Basado en Competencias para el uso y aplicación tecnológico con propósitos educativos. Estructurado en forma modular y bajo un enfoque pedagógico que considera factores como el geográfico o contextual, esta solución integra ejercicios para el trabajo individual o colaborativo, así como actividades y ejercicios virtuales o fuera de línea, todo esto diseñado ex-profeso para brindar a los participantes rutas de aprendizaje personalizadas y servicios de valor agregado, con materiales, actividades y ejercicios intelectuales, lúdicos, relevantes y significativos para el desarrollo de competencias específicas de cada campo.

► **Definición de Elementos de trabajo**

- Servidor
- Equipo de Computo
- Instructor
- Usuarios (alumnos / invitados)
- Aulas (Aula Control, Aula taller y Aula Virtual)
- Temario

- **Servidor**

Este será la máquina principal, donde se alojará el sistema, aplicaciones y sitio Web, así como herramienta de comunicación e interacción para los alumnos del curso digital.

A continuación se presentan los requerimientos básicos que se consideran para montar el servidor:

- **Características Generales:**

Sistema Operativo: Linux Slackware

Versión de Kernel: 2.4.26

Memoria: 1 GB

Discos Duros: 120 GB

Dispositivos Periféricos:

CD-ROM

Procesador: Intel Pentium 4 3.00 GHz

Particiones en Disco:

/	10	GB
/home	30	GB
/users	50	GB
/var	8	GB
/usr	10	GB
/tmp	10	GB
Swap	2	GB

- **Sistema Operativo:**

Debido al fácil manejo en archivos de configuración y por ser una de las distribuciones de Linux estándar además de ser la plataforma con que se trabajo durante todo el Diplomado de Software Libre, el sistema operativo a usar para el Sitio Web (Aula Virtual) será **Linux Slackware 10**.

- **Herramientas requeridas y servicios**

En este punto la referencia es a todos aquellos programas o servicios que deben estar completa y correctamente configurados para la implementación de todo el sistema virtual.

A continuación se muestra una lista de las herramientas, así como las versiones sugeridas.

Servicios:

ssh	22
httpd	80
https	443
tomcat	8080
mysql	3306
postgres	5432
X11	6000

Herramientas:

httpd-2.0.5
 mysql-4.0.21
 php-4.3.11
 postgresql-7.4.1
 nessus-2.2.5

- **Aplicacion**

Este es un punto muy importante en el desarrollo del proyecto, ya que es la herramienta que va a facilitar la implementación, crecimiento y monitoreo del Aula Virtual.

Para este proyecto se hara uso de una aplicación llama **moodle** que ha sido desarrollada con software libre y exclusivamente diseñada para la enseñanza a distancia, a continuación se dara una breve descripción de esta poderosa herramienta:

Moodle es un sistema de gestión de la enseñanza, es decir, una aplicación diseñada para ayudar a los educadores a crear cursos de calidad en línea. Estos tipos de sistemas de aprendizaje a distancia a veces son también llamados ambientes de aprendizaje virtual o educación en línea.

Es una aplicación Web que puede funcionar en cualquier computadora en la que pueda correr PHP, y soporta varios tipos de manejadores de bases de datos (en especial MySQL y Postgres)

Fue creado por Martin Dougiamas, quien trabajó como administrador de [WebCT](#) en la Universidad Curtin, y se basó en trabajos sobre el constructivismo en pedagogía, que afirman que el conocimiento se construye en la mente del estudiante en lugar de ser transmitido sin cambios a partir de libros o enseñanzas. Un profesor que opera desde este punto de vista crea un ambiente centrado en el estudiante que lo ayuda a construir ese conocimiento en base a sus habilidades y conocimientos propios en lugar de simplemente publicar y transmitir la información que consideran que los estudiantes deben conocer.

Moodle ha venido evolucionando desde 1999 y nuevas versiones siguen siendo producidas. En enero de 2005, la base de usuarios registrados incluye 2.600 sitios en más de 100 países y está traducido a más de 50 idiomas. El sitio más grande reporta tener actualmente 6.000 cursos y 30.000 estudiantes.

La palabra **Moodle** era al principio un acrónimo de *Modular Object-Oriented Dynamic Learning Environment* (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular), lo que resulta fundamentalmente útil para programadores y teóricos de la educación. También es un verbo que describe el proceso de deambular perezosamente a través de algo, y hacer las cosas cuando se te ocurre hacerlas, una placentera chapuza que a menudo te lleva a la visión y la creatividad. Las dos acepciones se aplican a la manera en que se desarrolló **Moodle** y a la manera en que un estudiante o profesor podría aproximarse al estudio o enseñanza de un curso en línea. Todo el que usa **Moodle** es un *Moodler*.

- **Equipo de Computo en aulas presénciales**

Una de las condiciones para que el proyecto funcione es que las aulas de clases deben estar equipadas con computadoras actualizadas, para un mejor manejo de las aplicaciones, así como facilitar la comunicación cliente servidor.

A continuación se presentan algunos de los elementos mínimos requeridos.

- **Características del equipo**

Sistema Operativo: Windows XP Profesional SP2
Memoria: 512 MB
Disco Duro: 80 GB
Procesador: Pentium 4 2.4Ghz

- **Aplicaciones necesarias**

SSH Secure Shell 3.2
Microsoft office
Navegadores: Mozilla Fire Fox / Explorer (actualizado)
Adobe Acrobat 6
VMware Work Station

- **Políticas de uso de los equipos y área de trabajo**

- ▶ El equipo de cómputo deberá utilizarse como herramienta de uso académico y se recomienda de que exista una maquina por alumno.
- ▶ El uso del equipo de cómputo es exclusivo para los usuarios
- ▶ Cualquier persona que no esté considerada como tal y que desee hacer uso del equipo deberá reportarlo, para poder darse de alta tanto con el curso como en el sistema.
- ▶ A todos los usuarios inscritos al curso se les proporcionará un identificador de acceso personalizado (cuenta) a los servicios de cómputo. El nombre de la cuenta será proporcionada por el responsable del servidor de cursos.
- ▶ Todas las cuentas irán acompañadas de una clave de acceso (contraseña) que podrá ser cambiada a voluntad del usuario.
- ▶ Todas las cuentas deberán estar asociadas a un usuario, por lo que no se otorgarán cuentas para ser utilizadas por dos o más personas. Esto quiere decir, que Todas las cuentas son personales e intransferible

• **Instructor**

Es el responsable directo del proyecto, cuyo trabajo será actualizar y depurar toda la información que sea usada en el sitio.

- **Responsabilidades, derechos y obligaciones**

- ▶ Motivar al alumno a ser parte activa del proyecto
- ▶ Proporcionar el material necesario para cada una de las clases (realizar prácticas, exponer temas actuales y reales)
- ▶ El instructor en curso se reserva el derecho de suspender el servicio a un usuario cuando se sospeche de un abuso o uso indebido del servicio

• **Usuarios**

Se consideran alumnos a todos aquellos participantes que se encuentren inscritos en el curso y/o dados de alta en el sistema.

Los usuarios los hay de dos tipos; los alumnos propios del curso y los invitados o personas ajenas que lleguen a visitar el sitio y estén interesadas en participar y apoyar el proyecto, esto bajo la condición de la aceptación de los términos de uso del Aula virtual.

A continuación se presentan algunos ejemplos como puntos básicos e importantes a considerar de las actividades propias de los usuarios del curso (presencial y/o en línea)

- **Responsabilidades, derechos y obligaciones**

Prohibiciones

- ▶ Utilizar el equipo de cómputo para desarrollar programas o proyectos ajenos al interés académico
- ▶ Copiar software cuya licencia de uso lo prohíba.

- ▶ Alterar software instalado en el equipo de cómputo.
- ▶ Llevar a cabo acciones que interfieran o puedan interferir con la operación normal de los equipos de cómputo o de comunicación electrónica

Obligaciones:

- ▶ El participante debe ser un elemento activo, es decir, aportar ideas, realizar comentarios constructivos, apoyar en todo momento a alumnos de otros niveles, desarrollar todas y cada una de las tareas de manera limpia y personal.
- ▶ Debe concientizarse del alcance o nivel que puede lograr si dedica tiempo a sus actividades.
- ▶ El alumno debe estar comprometido a hacer buen uso de las herramientas proporcionadas con el fin de mantener el buen funcionamiento de las mismas.
- ▶ Es muy importante respetar los derechos de autor en todos los trabajos que se realicen dentro del sitio Web.

Derechos

1. Ser tratado con respeto
2. A recibir asesoría
3. A contar con equipo de cómputo que funcione adecuadamente.
4. Trabajar en un ambiente cordial y de respeto

- **Aulas o medios de aprendizaje**

En este punto se hace referencia al sitio físico y virtual donde se lleve a cabo un proceso de enseñanza-aprendizaje e intervienen nuestros dos sujetos, alumno-instructor.

Para un mejor manejo aprovechamiento didáctico y significativo se opto por el uso de tres sitios;

Aula Control: Es el salón de cómputo donde las clases se realizan de forma presencial y cotidiana.

Aula taller: Por lo regular es un salón de clases tradicional empleado para la realización de exámenes, exposiciones, y didácticas grupales.

Aula Virtual: Se trata del curso de apoyo y consulta didáctico alojado en un sitio Web

- **Temario**

- **Proceso del desarrollo del temario**

Temario desarrollado metódicamente para así lograr y alcanzar la meta en este curso, los puntos más importantes se muestran a continuación:

- Lograr que el participante se interese por el desarrollo de software libre

- Crear un ambiente de trabajo libre, el cual ayude a aportar ideas o experiencias en determinados temas.
- El participante deberá ser capaz de reconocer, configurar e instalar desde cero, la mayoría de los sistemas Linux más comunes hoy en día.
- El participante aprenderá a realizar los diferentes tipos de instalaciones que requieren las distribuciones e incluso compilar si es necesario.
- Será capaz de administrar posteriormente dicho sistema de forma básica/avanzada
- Podrá localizar en todo momento que le hace falta para completar o satisfacer una necesidad de software (aplicación)

Como se puede notar el curso resulta ser bastante ambicioso, ya que trata de abarcar diferentes temas del Software libre. Desde instalación, comandos básicos, manejo e implementación de herramientas (Web, Manejadores de Bases de Datos, parches, etc) hasta una administración completa de un servidor Linux.

Para poder realizar dicho proyecto es necesario tomar en cuenta varios puntos

- **Nivel de aplicación y aportación a los participantes**

En este curso se proporcionan los conocimientos necesarios para administrar el Sistema Operativo Linux, Al finalizar el curso el alumno estará capacitado para sacar partido a un Sistema Operativo en constante expansión, que día a día se está posicionando en el ámbito empresarial.

Dirigido a usuarios y administradores, el curso se enfoca como la base válida para todo sistema Linux

- **Método de aprendizaje; que facilite el progreso de lo aprendido**

Se partirá del método de “Enseñanza Aprendizaje Significativo”, ya que permite un manejo grupal y de evaluación constante, una de las ventajas de este método es que no requiere conocimientos previos, en este caso del Sistema Operativo Linux, pero el hecho de saber que ya ha tenido contacto con algún otro sistema operativo comercial, permite partir de un mismo punto.

- **Conocimientos previos del usuario**

Como se menciona en el anterior punto, sería deseable requerir que los alumnos no conocieran Windows, por la malformación que ello crea, pero como eso es un imposible, no se requiere nada. No se requiere conocimientos de otros sistemas, ni de procesadores de texto, y mucho menos se requieren conocimientos de lenguajes de programación.

El único requerimiento es ser un poco obstinado en querer aprender y dominar cosas.

- **Diseño del temario**

En este modulo se consideraron dos puntos característicos a este trabajo:

- Uso de temas actuales
- Temas totalmente prácticos

El Temario se repartió en cuatro grupos principales en el manejo y conocimiento del Software libre:

Introducción al sistema:

Introducción a Linux
Utilerías de Linux

Instalación:

Linux

Configuración y herramientas:

Programación en shell.
Apache
Mysql
Postgres

Administración

Administración Linux
Seguridad

- **Método de evaluación; por niveles en el sistema**

Antes de pasar de un modulo es necesario realizar una practica general, donde dependiendo de los aciertos obtenidos, se determinara el paso al siguiente nivel. Esta podría ser un mal método de evaluación pero, se toma como un filtro ya que para algunos seguir adelante será un problema, pero para los que va dirigido este curso será todo un reto

- **Implantación y mantenimiento del sistema**

La mayoría de estos puntos fueron realizados en conjunto con un grupo de alumnos, quienes determinaban gracias a sus dudas las necesidades o requerimientos de un curso como éste.

Cabe recordar que este curso es implementado y aplicado en un entorno de uso real y en producción de sistemas Linux, a diferencia de otros cursos puramente académicos, está respaldado por la experiencia real en el manejo y administración de servidores con Linux. Con ello hay un acercamiento al máximo posible, al mundo real y sus requerimientos.

▶ Aula Virtual

Elementos, servicios y composición del sitio Web

- ▶ Index (Bienvenida)
- ▶ Registro / Home
- ▶ Consulta
 - Soporte
 - Archivos
- ▶ Novedades
 - Avisos, eventos
 - Pasatiempos
 - Fotos
 - Anuario Virtual
- ▶ Glosario
- ▶ Foro de discusiones
 - Wiki
- ▶ Biblioteca
 - Revistas
 - Manuales
 - Tutoriales
 - Enlaces Educativos
- ▶ Carga y descarga de objetos de aprendizaje
- ▶ Contacto
 - Alumnos
 - Instructores
- ▶ Tareas
 - Practica
 - Auto evaluación
 - Examen de nivel

Moodle ya trae integrados diversos módulos listos para ser usados y configurados de acuerdo a las necesidades, a continuación se muestran algunos de los elementos que integran el Aula Virtual.



Search Forums [-]

>

Advanced search ?

Entrada Aleatoria del Glosario básico [-]

CVS

CVS es un Sistema Concurrente de Versiones. Normalmente se usa como un forma de almacenar el código fuente, ya que mantiene las versiones de todos los archivos de manera que no se pierda nada, y se registra el uso que hacen diferentes personas. También proporciona heramientas para combinar código si hay dos o más personas trabajando en el mismo archivo. Todo el código y todas las versiones se almacenan en un servidor central (en este caso, en Sourceforge).

[Más conceptos ...](#)

Proximos Eventos [-]

Sala de Chat
Thursday, 15 December (05:00 AM)

[Go to calendar...](#)
[New Event...](#)

Recomendaciones [-]

- [Educlip 2006: Premios a la Creatividad en la Comunicación Educativa](#)
- [Integración Moodle-LAMS disponible](#)
- [Prototipo del portátil de 100 dólares en la World Summit](#)
- [Eduteka: Google Earth en clase de Geografía](#)
- [La Open University escoge Moodle](#)
- [Número 2 de la revista DIM](#)
- [Primeras Jornadas sobre WebQuest](#)
- [Integración Sakai-LAMS](#)
- [IEM 11: uso de las TIC en Portugal](#)
- [Monográfico de Teoría de la Educación sobre sistemas de e-learning](#)

Navegador de Archivos

Localización / ▾

Sistemas Operativos	Bases de Datos	Programacion	Redes
16 Noviembre, 11:38am 4 Kb	11 Noviembre, 3:16pm 4 Kb	11 Noviembre, 3:14pm 4 Kb	11 Noviembre, 3:18pm 4 Kb
Seguridad	Laptop	DiploBD	
11 Noviembre, 3:24pm 4 Kb	5 Diciembre, 11:17pm 4 Kb	2 Diciembre, 3:01pm 4 Kb	

Steve: pasenla chevere y
diviertanse
<http://funnyfox.org>

Nombre:

E-Mail:

URL:

Mensaje:

Enviar Mensaje

Clima

Mexico City ,Mexico

8°C

Se siente 8°C



12/13/2005 04:45:00

Calendario

<< December 2005 >>

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Global events Course events

Conclusiones

-

Conclusión

Es un hecho que toda persona responsable de un sistema de cómputo debe estar en constante actualización ya sea en técnicas de desarrollo o implementación y manejo de ciertos sistemas operativos y/o programas, esto debido a que el campo de la computación es impredecible y su crecimiento es exponencial.

El Diplomado de Desarrollo e Implementación de Software libre Linux, cumple con esta visión de promover las herramientas libres más usadas actualmente por su alta aceptación y desempeño, con el objetivo de que cualquier usuario desde cualquier punto pueda aportar de su conocimiento para el beneficio propio o de terceros.

El éxito del Software Libre se debe en su mayor parte a Internet, por que esto ha permitido que las personas interesadas en los varios componentes del software libre se pongan en contacto con otras. Internet de esta manera actúa como un catalizador que acelera el desarrollo y acelera, sintetiza el conocimiento en áreas muy específicas.

Hay diferentes motivaciones que impulsan a los contribuidores y desarrolladores a trabajar en el software libre, las más importantes son:

- El deseo de crear software más robusto.
- La posibilidad de estar en control del software. Esto es importante para aplicaciones de misión crítica donde es totalmente imperante tener un control total sobre posibles problemas en cualquier punto.
- Crear aplicaciones de bajo costo.
- Re-uso del conocimiento: Esto permite que la gente tome el conocimiento que se ha sintetizado en el software. En vez de empezar siempre desde cero (que es el caso de la industria de software actual) siempre se puede empezar un proyecto desde un fundamento establecido.
- La posibilidad de adaptar el software a sus necesidades
- Aprender alguna técnica de programación.

Gracias a los puntos antes mencionados, actualmente existe una mayor aceptación de sistemas Linux, así como algunas de sus herramientas, es por ello que se le encuentra desde PC's que albergan manejadores de bases de datos o servidores de Web hasta supercomputadoras que exploran el universo.

Cada uno de los módulos tomados en el Diplomado abrió una puerta sinfín, con las bases necesarias para explotar al nivel que se requiera las aplicaciones más completas para proyectos robustos. Para este trabajo se obtuvieron muy buenos resultados creando la relación software libre y la educación a distancia.

Desde los primeros trabajos de investigación sobre el uso e implementación de las herramientas electrónicas como apoyo a actividades académicas, se han encontrado fuertes corrientes; los que apoyan a toda costa este tipo de materiales sugiriendo su uso a todos los niveles educativos, con el propósito de brindar mayor conocimiento general

del mundo que rodea a las nuevas generaciones y por el otro lado, hay un conjunto de profesionales que restringen el manejo de las herramientas sugiriendo que el uso de éstas puede provocar un retroceso en el nivel de aprendizaje de los alumnos.

Y es que con el tiempo, los medios de información brindan mayor comodidad comparada con tan solo algunos años atrás, cuando aun se buscaban respuestas en las bibliotecas más cercanas, teniendo contacto directo con aquellos libros que ahora se han ido cambiando por páginas electrónicas o enciclopedias en DVD, el problema radica en no hacer un buen uso de todos los medios disponibles.

El hecho es que independientemente de las similitudes o diferencias que se pueden encontrar entre la implementación o el rechazo de las nuevas herramientas, el desarrollo de las aplicaciones que faciliten el acceso a la información no pueden esperar y mucho menos cuando se trata de los ambientes Libres.

Es por ello que el proyecto presentado trata de acoplar perfectamente el manejo de las nuevas herramientas en el tema de la Educación fusionando las clases a distancia con las clases presenciales con el fin de explorar y explotar nuevas formas de enseñanza, en este caso un Aula Virtual.

Se puede considerar que la Educación a Distancia constituye una actividad nueva, pero que cuenta con enormes posibilidades de incremento futuro, gracias las aplicaciones tecnológicas y pedagógicas que día a día exploran la forma de hacer llegar el conocimiento de un punto a otro en una red de computadoras.

Es por ello que haciendo un uso adecuado del control y manejo presencial en las aulas y como apoyo las aplicaciones en línea como las que se expusieron en este trabajo es posible lograr un reforzamiento del conocimiento adquirido y una mayor aportación a la comunidad Linux, la cual crece precisamente debido a las ideas y necesidades que presentan los usuarios del software libre.

Gracias al Diplomado ya mencionado se ha adquirido un reconocimiento global de las capacidades y alcances de los sistemas que se pueden llegar a desarrollar, así como aprender a explotar infinidad de herramientas que nos ofrece el cómputo creciente.

El software libre en muchos casos es mejor que el software propietario, pero en algunos casos las aplicaciones libres no son tan buenas como las propietarias.

Aunque esto podría verse negativamente se tiene que pensar aquí cuantitativamente y evaluar las necesidades del proyecto y dados estos criterios evaluar si se debe adoptar una solución propietaria o una solución libre.

Referencias Bibliográficas

Administración de Red Hat Linux

Thomas Schenk et al.
Prentice Hall
España 2000

Servidor Apache

Rich Bowen & Ken Coar
Prentice Hall
España 2001

Linux Maxima Seguridad

Anonimo
Prentice Hall
Madrid, España 2000

Referencias Electrónicas

- The Apache Software Foundation. Apache HTTP SERVER PROJECT [en línea] USA. Disponible en <http://apache.org>
- The PHP Group. PHP Hypertext Preprocessor [en línea]. USA, [ref. de octubre de 2005], Disponible en: <http://www.php.net>

Metallica - San Francisco Orchestra - Nothing Else Matters.mp3

- RedHat Inc. RedHat Docs [en línea]. USA [ref. de octubre de 2005], Disponible en <http://www.redhat.com/docs/>
- Learning Workshops, Comunidad de Elearning [en línea] USA, Disponible en: <http://www.elearningworkshops.com>
- Moodle [en línea] USA, Disponible en: <http://moodle.org/>
- Campus Red, Aula Virtual FAD [en línea], Disponible en: <http://aulavirtual.fad.es/>
- CDI Curso digital en línea [en línea], Argentina Disponible en: <http://cdi.integrando.org.ar>
- Asociación para el conocimiento y la innovación de la informatica. Programación Web con PHP[en línea] España, [ref. septiembre de 2005], Disponible en: <http://ascii.eii.us.es/cursos/php/php9.html>