



INSTITUTO TECNOLÓGICO SUPERIOR CORDILLERA

INSTITUTO TECNOLÓGICO SUPERIOR *CORDILLERA*

ESCUELA DE SISTEMAS

Proyecto de Grado, previa obtención del título de:
Tecnólogo Analista de Sistemas

TEMA

Investigación y desarrollo de una plataforma virtual sobre
los delitos informáticos en el Ecuador (Diseño y elaboración
de la plataforma virtual)

AUTOR:

Paul Ribadeneira

TUTOR:

Ing. Jaime Padilla

2011

QUITO – ECUADOR



CAPITULO I

EL PROBLEMA

1.1 Planteamiento del Problema

En nuestro país se ha dado a conocer sobre fraudes informáticos a establecimientos públicos como privados ya que dichos establecimientos ya mencionados carecen de falta de información por los cuales son blancos fáciles de hackers y crackers los cuales modifican o dañan software de mucha vitalidad para la empresa.

Los delitos informáticos se cometen mediante usos de computadoras, sistemas informáticos, u otros dispositivos de comunicación.

Tienen por objeto causar daños, provocar pérdidas, o impedir el uso de sistemas informáticos.

Se considera delito informático a actos comunes como plagio, robo, información, pornografía, todo lo relacionado con tarjetas de crédito o débito su clonación, falsificación, el espionaje informático, falsificación de documentos, fraudes con telefonía celular, el pishin, entre otros,

El delito informático tiene como origen el mismo momento que surge la tecnología informática, la facilitación de las labores que trae consigo las computadoras propician que en un momento dado el usuario se encuentre ante una situación de ocio la cual canaliza a través de las computadoras cometiendo sin darse cuenta una serie de lícitos.

Aquellas personas que utilizan tecnología de información para reproducir o divulgar algún bien producto del intelecto de diversos autores como música videos, obras literarias, o de cualquier índole que está incurriendo en la elaboración del derecho de autor.

Vemos el caso de la piratería cuando se reproducen esas películas que vemos vendiendo a los ambulantes en las calles e incluso en los trabajos universitarios cuando un estudiante busca información atribuyéndosela como propia cuando en realidad le pertenece a un tercero.



Desde que internet es una realidad en todas las partes del mundo lo que antes era un secreto a voces o el mercado de unos pocos, se ha ido convirtiendo en un verdadero problema y es que ahora con tan solo un par de clics en la información correcta cualquiera puede tener acceso a este material.

A pesar de las intensas campañas de concienciación el uso de pornografía infantil gracias a internet ha ido creciendo a niveles desmesurados, llegando a existir en la actualidad alrededor de cuatro millones de zonas virtuales donde existe la posibilidad de acceder a dicho material

Tampoco entonces nos sorprende la enorme cantidad de sitios que día a día se van creando relativos a esta temática según ciertas estimaciones se crean al día alrededor de 500 nuevas zonas virtuales destinadas a la difusión de este contenido un número indiscutiblemente alarmante.

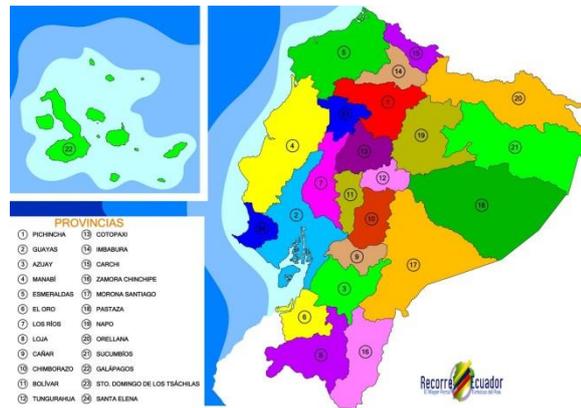
Otro de los delitos informáticos más comunes y que muchas veces comienzan con el desconocimiento de procedimientos o el descuido de un acceso indebido y que culmina con un fraude informático es el fraude con tarjetas o la estafa en el comercio electrónico de aquellos que utilizan este como medio para realizar transacciones o incluso al recibir correos electrónicos que aseguran que recibirán algún dinero y solo piden proporcionar algunos datos personales incluyendo número de tarjeta o algún donativo o más simple aun la instalación de algún virus al abrir alguno de estos correos que sin darse cuenta esta ahí se encarga de robar algún tipo de información.

1.2 Formulación del Problema

¿Con el desarrollo de una plataforma virtual sobre los delitos informáticos en el Ecuador se garantiza que estos reducirán?

1.3 Delimitación del Problema

Esta plataforma virtual se la implementará en el país de Ecuador



Nombre: Mapa Ecuador

Gráfico: Nº 1

Fuente: Internet

1.4 OBJETIVOS

1.4.1 Objetivo general

Investigar y desarrollar las leyes que se aplican en el delito informático en el Ecuador dando a conocer en un sistema orientado en la web

1.4.2 Objetivos Específicos

- Investigar delitos donde se ha utilizado la informática con fines maliciosos
- Buscar y ayudar a conocer sobre las leyes informáticas del Ecuador
- Analizar los procesos realizados por el personal indicado de la policía ecuatoriana en cada delito informático
- Desarrollar informes acerca de los delitos informáticos que estos ya se hallan dictado sentencia.
- Desarrollar una página web acerca de delitos informáticos



1.5 ALCANCES

Dadas las especificaciones y conocimientos generales de nuestro Instituto Tecnológico Superior Cordillera en investigación, se ha deducido que el sistema a implementar según los requerimientos, es una plataforma virtual dinámica, mediante esta plataforma se podrá generar reportes sobre los procesos, acciones, funciones, demostraciones de los delitos informáticos presentados.

1.5.1 Módulo de Mantenimiento de tabla:

Por ser una plataforma de control y actualización de información se maneja frecuentemente el movimiento de tablas como son las siguientes:

- Mantenimiento de tablas mediante programación con datos actualizados
- Eliminación de tablas por códigos dado el caso de redundancia de información.
- Modificación de tablas o de campos
- Registro de usuarios nuevos
- Actualizaciones de software de seguridad utilizada para la protección del mismo.

1.5.6 Módulo de Consistencia de la información:

La plataforma garantiza la seguridad y consistencia de la información ingresada con el objetivo de demostrar un verdadero análisis, permitirá administrar y controlar los siguientes procesos:

- Supervisar el control de registro de usuarios mediante un análisis
- Supervisar el control de software insertados en la plataforma virtual
- Supervisar el ingreso de nueva legislación a la biblioteca virtual
- Revisar la plataforma por medio de la web en cualquier parte del mundo



1.5.7 Módulo de Transacción

En este módulo se dará mantenimiento a la comunicación del usuario que está en el internet para que este no tenga un desfase de comunicación cuando se realicen foros acerca del tema que se halla puesto

1.6 JUSTIFICACION

Al no existir una cultura informática razón por la cual el pueblo ecuatoriano se ve atacado de fraudes informáticos y a su vez el desconocimiento de cómo enfrentar dichos fraudes.

La red de computadoras provee muchas ventajas para quienes las usan sin embargo también poseen muchos niveles de vulnerabilidad. El hecho de que la red permita compartir información

La seguridad informática toma mayor relevancia cada día, ya que el crecimiento indiscriminado de los delitos informáticos pone en riesgo la integridad, confidencialidad y disponibilidad de la información de ahí la necesidad del presente trabajo de investigación y elaboración de una plataforma virtual



CAPITULO II

MARCO TEORICO

2 ANTECEDENTES

En el Ecuador carece de una página web dinámica para poder informarse acerca de los diferentes delitos ya que en la web solo encontramos información al respecto del tema pero no dan soluciones a estos problemas.

Conociendo este problema se decide crear una biblioteca virtual para que los usuarios ecuatorianos afectados o simplemente por informarse puedan acceder a información relevante a casos similares que a ellos les pueda afectar y dando una solución coherente a su problema.

2.1 Reseña Histórica

El país por el momento no cuenta con el apoyo de ninguna plataforma virtual que informe y de a conocer sobre todos los delitos realizados por medio de tecnología, ya que es un tema que recién se está introduciendo en nuestro país y la población no sabe como poder manejarse ante estos delitos.

Esta plataforma es muy útil, nos capacita y nos da a conocer algunas formas de robos y que pasos seguir en dicha situación ya que también se darán consejos para prevenir cualquier acto delictivo.

Luego de realizar una investigación para ver, si ha existido proyectos de este tipo realizados por otras personas, no existe la creación en la web sobre modos de prevención, por consiguiente nosotros, como alumnos del instituto Cordillera, como complemento para el término de nuestros estudios superiores, para optar por el título de Tecnólogos, queremos desarrollar este proyecto final, que sabremos que tendremos el apoyo total de la comunidad, por supuesto bajo la supervisión de nuestro tutor quien nos sabrá guiar, durante el desarrollo y la creación de este proyecto informático.



La creación de la plataforma virtual sobre delitos informáticos nos ha permitido un mejor rendimiento intelectual muy óptimo en varios países que han desarrollado dicha plataforma, en la mayoría de casos han ayudado a que las personas no caigan fácilmente en mentiras ni sobornos que no existen.

Misión

Somos líderes en servicios informáticos educativos en proceso de capacitación y demostración con calidad a personas de todo tipo de edad, con un contexto pedagógico constructivista fundamentados en principios de equidad, eficiencia y pertinencia.

Visión

Los Ecuatorianos forma talentos con liderazgo, autonomía y criterio humanista, valores en las áreas administrativas, culturales, con capacidad para acceder al conocimiento a través de la investigación y el uso adecuado de la tecnología, comprometidos con el ambiente y la sociedad.

2.3 Marco referencial

Computadora

También denominada ordenador (del francés ordinateur, y este del latín ordinator), es una máquina electrónica que recibe y procesa datos para convertirlos en información útil. Una computadora es una colección de circuitos integrados y otros componentes relacionados que puede ejecutar con exactitud, rapidez y de acuerdo a lo indicado por un usuario o automáticamente por otro programa, una gran variedad de secuencias o rutinas de instrucciones que son ordenadas, organizadas y sistematizadas en función a una amplia gama de aplicaciones prácticas y precisamente determinadas, proceso al cual se le ha denominado con el nombre de programación y al que lo realiza se le llama programador. La computadora, además de la rutina o programa informático, necesita de datos específicos (a estos datos, en conjunto, se les conoce como "Input" en inglés o de entrada) que deben ser suministrados, y que son requeridos al momento de la ejecución, para proporcionar el producto final del procesamiento de datos, que recibe el nombre de "output" o de salida. La información puede ser entonces utilizada, reinterpretada, copiada, transferida, o retransmitida a otra(s) persona(s), computadora(s) o



componente(s) electrónico(s) local o remotamente usando diferentes sistemas de telecomunicación, pudiendo ser grabada, salvada o almacenada en algún tipo de dispositivo o unidad de almacenamiento.

Equilibrar prioridades

Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un equilibrio que satisfaga los deseos de todos. Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro.

RUP

Proceso Unificado de Rational

Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de ModeladoUML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

También se conoce por este nombre al software desarrollado por Rational, hoy propiedad de IBM, el cual incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el Rational Method Composer (RMC), que permite la personalización de acuerdo con las necesidades.

Originalmente se diseñó un proceso genérico y de dominio público, el Proceso Unificado, y una especificación más detallada, el Rational Unified Process, que se vendiera como producto independiente.

El RUP está basado en 6 principios clave que son los siguientes:

Demostrar valor iterativamente

Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados



Elevar el nivel de abstracción

Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks) por nombrar algunos. Esto evita que los ingenieros de software vayan directamente de los requisitos a la codificación de software a la medida del cliente, sin saber con certeza qué codificar para satisfacer de la mejor manera los requisitos y sin comenzar desde un principio pensando en la reutilización del código. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML.

Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

Adaptar el proceso

El proceso deberá adaptarse a las necesidades del cliente ya que es muy importante interactuar con él. Las características propias del proyecto u organización. El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto en un área subformal.

Colaboración entre equipos

El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.



Diagramas de UML

Diagrama de casos de uso

Los diagramas de casos de uso describen las relaciones y las dependencias entre un grupo de casos de uso y los actores participantes en el proceso. Es importante resaltar que los diagramas de casos de uso no están pensados para representar el diseño y no puede describir los elementos internos de un sistema. Los diagramas de casos de uso sirven para facilitar la comunicación con los futuros usuarios del sistema, y con el cliente, y resultan especialmente útiles para determinar las características necesarias que tendrá el sistema.

Diagrama clase

Los diagramas de clases muestran las diferentes clases que componen un sistema y cómo se relacionan unas con otras. Se dice que los diagramas de clases son diagramas «estáticos» porque muestran las clases, junto con sus métodos y atributos, así como las relaciones estáticas entre ellas: qué clases «conocen» a qué otras clases o qué clases «son parte» de otras clases, pero no muestran los métodos mediante los que se invocan entre ellas.

Diagrama de secuencia

Los diagramas de secuencia muestran el intercambio de mensajes (es decir la forma en que se invocan) en un momento dado. Los diagramas de secuencia ponen especial énfasis en el orden y el momento en que se envían los mensajes a los objetos.

En los diagramas de secuencia, los objetos están representados por líneas intermitentes verticales, con el nombre del objeto en la parte más alta. El eje de tiempo también es vertical, incrementándose hacia abajo, de forma que los mensajes son enviados de un objeto a otro en forma de flechas con los nombres de la operación y los parámetros.

Diagrama de colaboración

Los diagramas de colaboración muestran las interacciones que ocurren entre los objetos que participan en una situación determinada. En los diagramas de colaboración los

mensajes enviados de un objeto a otro se representan mediante flechas, mostrando el nombre del mensaje, los parámetros y la secuencia del mensaje. Los diagramas de colaboración están indicados para mostrar una situación o flujo programa específicos y son unos de los mejores tipos de diagramas para demostrar o explicar rápidamente un proceso dentro de la lógica del programa.

Diagrama de estado

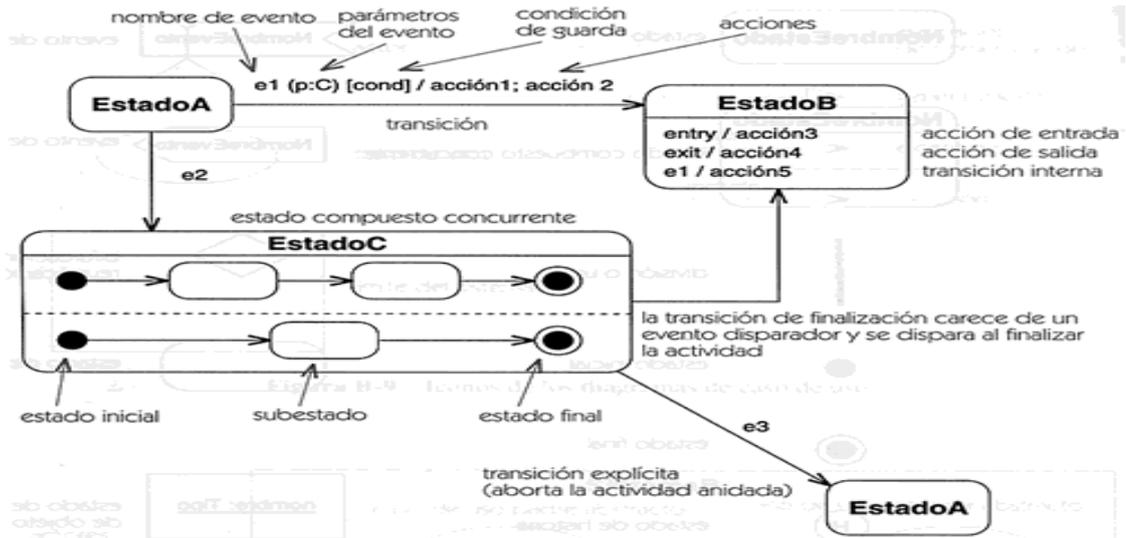


Diagrama de Estado

Grafico: N.-02

Fuente:Internet

Los diagramas de estado muestran los diferentes estados de un objeto durante su vida, y los estímulos que provocan los cambios de estado en un objeto. Los diagramas de estado ven a los objetos como máquinas de estado o autómatas finitos que pueden estar en un conjunto de estados finitos y que pueden cambiar su estado a través de un estímulo perteneciente a un conjunto finito. Por ejemplo, un objeto de tipo NetServer puede tener durante su vida uno de los siguientes estados:

- Listo
- Escuchando
- Trabajando
- Detenido

Diagrama de componentes.- Los diagramas de componentes muestran los componentes del software (ya sea las tecnologías que lo forman como Kparts, componentes CORBA, Java Beans o simplemente secciones del sistema claramente distintas) y los artilugios de que está compuesto como los archivos de código fuente, las librerías o las tablas de una base de datos. Los componentes pueden tener interfaces (es decir clases abstractas con operaciones que permiten asociaciones entre componentes. Diagrama de implementación. Los diagramas de implementación muestran las instancias existentes al ejecutarse así como sus relaciones. También se representan los nodos que identifican recursos físicos, típicamente un ordenador así como interfaces y objetos (instancias de las clases). El lenguaje de programación se lo realizara en 3 capas concatenado con c# y un modelador de base de datos los cuales son Power Designer, Toad Data Modeler.

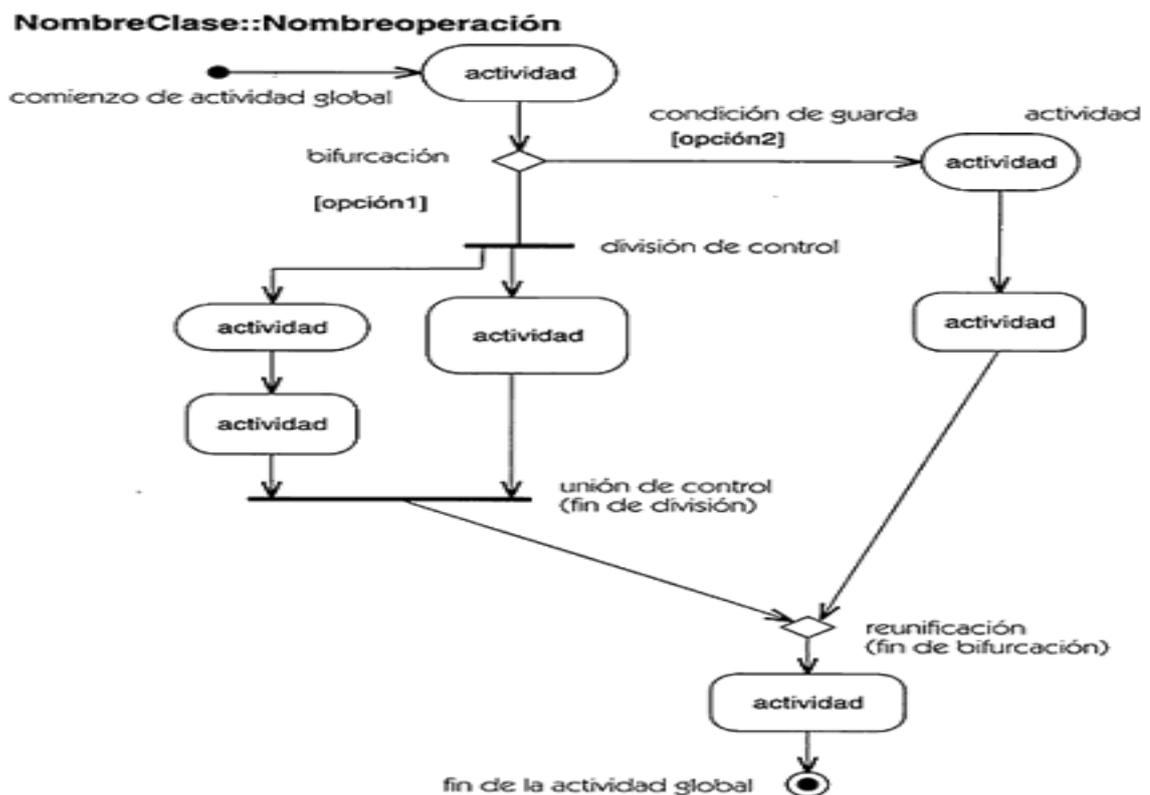


Diagrama de Actividades

Grafico: N.-03

Fuente:Internet



Protocolo de Redes

Los protocolos de red son una o más normas estándar que especifican el método para enviar y recibir datos entre varios ordenadores. Su instalación está en correspondencia con el tipo de red y el sistema operativo que la computadora tenga instalado.

No existe un único protocolo de red, y es posible que en un mismo ordenador coexistan instalados varios de ellos, pues cabe la posibilidad que un mismo ordenador pertenezca a redes distintas. La variedad de protocolos puede suponer un riesgo de seguridad: cada protocolo de red que se instala en un sistema queda disponible para todos los adaptadores de red existentes en dicho sistema, físicos (tarjetas de red o módem) o lógicos (adaptadores VPN). Si los dispositivos de red o protocolos no están correctamente configurados, se puede dar acceso no deseado a los recursos de la red. En estos casos, la regla de seguridad más sencilla es tener instalados el número de protocolos indispensable; en la actualidad y en la mayoría de los casos debería bastar con sólo TCP/IP.

Dentro de la familia de protocolos se pueden distinguir

Protocolos de transporte:

- ATP (Apple Talk Transaction Protocol)
- NetBios/NetBEUI
- TCP (Transmission Control Protocol)

Protocolos de red:

- DDP (Delivery Datagram Protocol)
- IP (Internet Protocol)
- IPX (Internet Packed Exchange)
- NetBEUI Desarrollado por IBM y Microsoft.

Protocolos de aplicación:

- AFP (Appletalk File Protocol)
- FTP (File Transfer Protocol)
- Http (Hyper Text transfer Protocol)



Dentro de los protocolos antes mencionados, los más utilizados son:

- IPX/SPX, protocolos desarrollados por Novell a principios de los años 80 los cuales sirven de interfaz entre el sistema operativo de red Netware y las distintas arquitecturas de red. El protocolo IPX es similar a IP, SPX es similar a TCP por lo tanto juntos proporcionan servicios de conexión similares a TCP/IP.
- NETBEUI/NETBIOS (Network Basic Extended User Interface / Network Basic Input/Output System) NETBIOS es un protocolo de comunicación entre ordenadores que comprende tres servicios (servicio de nombres, servicio de paquetes y servicio de sesión,

Inicialmente trabajaba sobre el protocolo NETBEUI, responsable del transporte de datos. Actualmente con la difusión de Internet, los sistemas operativos de Microsoft más recientes permiten ejecutar NETBIOS sobre el protocolo TCP/IP, prescindiendo entonces de NETBEUI.

- APPLE TALK es un protocolo propietario que se utiliza para conectar computadoras Macintosh de Apple en redes locales.
- TCP/IP (Transmission Control Protocol/Internet Protocol) este protocolo fue diseñado a finales de los años 60, permite enlazar computadoras con diferentes sistemas operativos. Es el protocolo que utiliza la red de redes Internet.

Servicios de una Red

Para que el trabajo de una red sea efectivo, debe prestar una serie de servicios a sus usuarios, como son:

1. Acceso, este servicios de acceso a la red comprenden tanto la verificación de la identidad del usuario para determinar cuáles son los recursos de la misma que puede utilizar, como servicios para permitir la conexión de usuarios de la red desde lugares remotos.
2. Ficheros, el servicio de ficheros consiste en ofrecer a la red grandes capacidades de almacenamiento para descargar o eliminar los discos de las estaciones. Esto permite almacenar tanto aplicaciones como datos en el servidor, reduciendo los



- requerimientos de las estaciones. Los ficheros deben ser cargados en las estaciones para su uso.
3. Impresión, este servicio permite compartir impresoras entre múltiples usuarios, reduciendo así el gasto. En estos casos, existen equipos servidores con capacidad para almacenar los trabajos en espera de impresión. Una variedad de servicio de impresión es la disponibilidad de servidores de fax.
 4. Correo, el correo electrónico, aplicación de red más utilizada que ha permitido claras mejoras en la comunicación frente a otros sistemas. Este servicio además de la comodidad, ha reducido los costos en la transmisión de información y la rapidez de entrega de la misma.
 5. Información, los servidores de información pueden bien servir ficheros en función de sus contenidos como pueden ser los documentos hipertexto, como es el caso de esta presentación. O bien, pueden servir información dispuesta para su proceso por las aplicaciones, como es el caso de los servidores de bases de datos.
 6. Otros, generalmente existen en las redes más modernas que poseen gran capacidad de transmisión, en ellas se permite transferir contenidos diferentes de los datos, como pueden ser imágenes o sonidos, lo cual permite aplicaciones como: estaciones integradas (voz y datos), telefonía integrada, servidores de imágenes, videoconferencia de sobremesa, etc.

Las Redes Informáticas

Una red es un sistema donde los elementos que lo componen (por lo general ordenadores) son autónomos y están conectados entre sí por medios físicos y/o lógicos y que pueden comunicarse para compartir recursos. Independientemente a esto, definir el concepto de red implica diferenciar entre el concepto de red física y red de comunicación.

Respecto a la estructura física, los modos de conexión física, los flujos de datos, etc; una red la constituyen dos o más ordenadores que comparten determinados recursos, sea hardware (impresoras, sistemas de almacenamiento...) o sea software (aplicaciones, archivos, datos...). Desde una perspectiva más comunicativa,



podemos decir que existe una red cuando se encuentran involucrados un componente humano que comunica, un componente tecnológico (ordenadores, televisión, telecomunicaciones) y un componente administrativo (institución o instituciones que mantienen los servicios). En fin, una red, más que varios ordenadores conectados, la constituyen varias personas que solicitan, proporcionan e intercambian experiencias e informaciones a través de sistemas de comunicación.

Estructura de las redes

Las redes tienen tres niveles de componentes: software de aplicaciones, software de red y hardware de red.

- El Software de Aplicaciones, programas que se comunican con los usuarios de la red y permiten compartir información (como archivos, gráficos o vídeos) y recursos (como impresoras o unidades de disco).
- El software de Red, programas que establecen protocolos para que los ordenadores se comuniquen entre sí. Dichos protocolos se aplican enviando y recibiendo grupos de datos formateados denominados paquetes.
- El Hardware de Red, formado por los componentes materiales que unen los ordenadores. Dos componentes importantes son los medios de transmisión que transportan las señales de los ordenadores (típicamente cables o fibras ópticas) y el adaptador de red, que permite acceder al medio material que conecta a los ordenadores, recibir paquetes desde el software de red y transmitir instrucciones y peticiones a otros ordenadores.

En resumen, las redes están formadas por conexiones entre grupos de ordenadores y dispositivos asociados que permiten a los usuarios la transferencia electrónica de información. En estas estructuras, los diferentes ordenadores se denominan estaciones de trabajo y se comunican entre sí a través de un cable o línea telefónica conectada a los servidores.

Dichos servidores son ordenadores como las estaciones de trabajo pero con funciones administrativas y están dedicados en exclusiva a supervisar y controlar el acceso a la red y a los recursos compartidos. Además de los ordenadores, los cables o la línea telefónica, existe en la red el módem para permitir la transferencia de información



convirtiendo las señales digitales a analógicas y viceversa, también existen en esta estructura los llamados Hubs y Switches con la función de llevar a cabo la conectividad.

Hacker

En informática, un hacker es una persona que pertenece a una de estas comunidades o subculturas distintas pero no completamente independientes:

Gente apasionada por la seguridad informática. Esto concierne principalmente a entradas remotas no autorizadas por medio de redes de comunicación como Internet ("Black hats"). Pero también incluye a aquellos que depuran y arreglan errores en los sistemas ("White hats") y a los de moral ambigua como son los "Grey hats".

Una comunidad de entusiastas programadores y diseñadores de sistemas originada en los sesenta alrededor del Instituto Tecnológico de Massachusetts (MIT), el Tech Model Railroad Club (TMRC) y el Laboratorio de Inteligencia Artificial del MIT. Esta comunidad se caracteriza por el lanzamiento del movimiento de software libre. La World Wide Web e Internet en sí misma son creaciones de hackers. El RFC 1392 amplía este significado como "persona que se disfruta de un conocimiento profundo del funcionamiento interno de un sistema, en particular de computadoras y redes informáticas"

La comunidad de aficionados a la informática doméstica, centrada en el hardware posterior a los setenta y en el software (juegos de ordenador, crackeo de software, lademoscene) de entre los ochenta/noventa.

En la actualidad se usa de forma corriente para referirse mayormente a los criminales informáticos, debido a su utilización masiva por parte de los medios de comunicación desde la década de 1980. A los criminales se le pueden sumar los llamados "script kiddies", gente que invade computadoras, usando programas escritos por otros, y que tiene muy poco conocimiento sobre cómo funcionan. Este uso parcialmente incorrecto se ha vuelto tan predominante que, en general, un gran segmento de la población no es consciente de que existen diferentes significados.



Mientras que los hackers aficionados reconocen los tres tipos de hackers y los hackers de la seguridad informática aceptan todos los usos del término, los hackers del software libre consideran la referencia a intrusión informática como un uso incorrecto de la palabra, y se refieren a los que rompen los sistemas de seguridad como "crackers" (analogía de "safecracker", que en español se traduce como "un ladrón de cajas fuertes").

Phreaker

De phone freak ("monstruo telefónico"). Son personas con conocimientos amplios tanto en teléfonos modulares (TM) como en teléfonos móviles.

Wannabe

Generalmente son aquellos a los que les interesa el tema de hacking y/o phreaking pero que por estar empezando no son reconocidos por la elite. Son aquellos que si perseveran aprendiendo y estudiando, pueden llegar a convertirse perfectamente en hackers. No por ser novato es repudiado, al igual que tampoco hay que confundirlo con un lammer.

Lammer o script-kiddies

Es un término coloquial inglés aplicado a una persona falta de madurez, sociabilidad y habilidades técnicas o inteligencia, un incompetente, que por lo general pretenden hacer hacking sin tener conocimientos de informática. Solo se dedican a buscar y descargar programas de hacking para luego ejecutarlos, como resultado de la ejecución de los programas descargados estos pueden terminar colapsando sus sistemas por lo que en general acaban destrozando la plataforma en la que trabajan.

Son aprendices que presumen ser lo que no son, aprovechando los conocimientos del hacker y poniéndolos en práctica, sin saber. En pocas palabras, no saben nada de hacking o roban programas de otros, frecuentemente recién hechos, y dicen que los crearon ellos.

White hat y black hat

Un hacker de sombrero blanco (del inglés, White hats), en jerga informática, se refiere a una ética hacker que se centra en asegurar y proteger los sistemas de Tecnologías de



información y comunicación. Estas personas suelen trabajar para empresas de seguridad informática las cuales los denominan, en ocasiones, «zapatillas o equipos tigre».

Por el contrario, un hacker de sombrero negro (del inglés, Black Hat) es el villano o chico malo, especialmente en una película de western, de ahí que en tal carácter se use un sombrero negro, en contraste con el héroe, el de sombrero blanco.

También conocidos como "crackers" muestran sus habilidades en informática rompiendo sistemas de seguridad de computadoras, colapsando servidores, entrando a zonas restringidas, infectando redes o apoderándose de ellas, entre otras muchas cosas utilizando sus destrezas en métodos hacking.

En los últimos años, los términos sombrero blanco y un sombrero negro han sido aplicados a la industria del posicionamiento en buscadores (Search Engine Optimization, SEO). Las tácticas de posicionamiento en buscadores de los hackers de sombrero negro, también llamada spamdexing, intento de redireccionar los resultados de la búsqueda a páginas de destino particular, son una moda que está en contra de los términos de servicio de los motores de búsqueda, mientras que los hackers de sombrero blanco, utilizan métodos que son generalmente aprobados por los motores de búsqueda.

Samurái

Normalmente es alguien contratado para investigar fallos de seguridad, que investiga casos de derechos de privacidad, esté amparado por la primera enmienda estadounidense o cualquier otra razón de peso que legitime acciones semejantes. Los samuráis desdeñan a los crackers y a todo tipo de vándalos electrónicos. También se dedican a hacer y decir cómo saber sobre la seguridad con sistemas en redes.

Newbie

Newbie es un término utilizado comúnmente en comunidades en línea para describir a un novato, en esta área, es el que no posee muchos conocimientos en el tema.

Sistemas informáticos: es el conjunto de partes interrelacionadas, hardware, software y de recurso humano (humanware) que permite almacenar y procesar información. El hardware incluye computadoras, que consisten en procesadores, memoria, sistemas de



Almacenamiento externo, etc. El software incluye al sistema operativo, firmware y aplicaciones, siendo especialmente importante los sistemas de gestión de bases de datos. Por último el soporte humano incluye al personal técnico que crean y mantienen el sistema (analistas, programadores, operarios, etc.) y a los usuarios que lo utilizan.

Los sistemas informáticos pasan por diferentes fases en su ciclo de vida, desde la captura de requisitos hasta el mantenimiento. En la actualidad se emplean numerosos sistemas informáticos en la administración pública.

Plataforma Virtual: Las plataformas virtuales se refieren únicamente para la tecnología utilizada para la creación y desarrollo de cursos o módulos didácticos en la Web que se usan de manera más amplia en la Web.

Con la llegada de Internet se produce un importante abaratamiento de los costos de desarrollo de programas, por lo que resulta más sencilla la creación de materiales para ser utilizados en línea. Sin embargo se siguen necesitando conocimientos avanzados de programación para crear un curso o un módulo didáctico, y por tanto estos cursos no son accesibles a todo el mundo. Desde mediados de los años 90 empiezan a surgir plataformas didácticas que permiten la creación y la gestión de cursos completos para la web sin que sean necesarios conocimientos profundos de programación o de diseño gráfico.

Servidores

En informática, un servidor es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.

Es una aplicación informática o programa que realiza algunas tareas en beneficio de otras aplicaciones llamadas clientes. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de una computadora y los servicios de aplicaciones, que realizan tareas en beneficio directo del usuario final. Este es el significado original del término. Es posible que un ordenador cumpla simultáneamente las funciones de cliente y de servidor.

Por lo cual podemos llegar a la conclusión de que un servidor también puede ser un proceso que entrega información o sirve a otro proceso. El modelo Cliente-servidor no



necesaria implica tener dos ordenadores, ya que un proceso cliente puede solicitar algo como una impresión a un proceso servidor en un mismo ordenador.

Dispositivos: Se puede definir como dispositivo un aparato, artefacto, mecanismo, artefacto, órgano o elemento de un sistema.

Los dispositivos son elementos de la PC que se requieren en todo momento para que la máquina funcione de manera correcta. Es decir, son imprescindibles al momento de usar la PC. Estos dispositivos permiten al usuario ver los resultados de los cálculos o de las manipulaciones de datos de la computadora. El dispositivo de salida más común es la unidad de visualización, que consiste en un monitor que presenta los caracteres y gráficos en una pantalla similar a la del televisor. Por lo general, los monitores tienen un tubo de rayos catódicos como el de cualquier televisor, aunque los ordenadores pequeños y portátiles utilizan hoy pantallas de cristal líquido (LCD, acrónimo de Liquid Crystal Displays) o electroluminiscentes. Otros dispositivos de salida más comunes son las impresoras, que permiten obtener una copia impresa de la información que reside en los dispositivos de almacenamiento, las tarjetas de sonido y los módem.

Delito informático a actos comunes como plagio, robo, información, pornografía, todo lo relacionado con tarjetas de crédito o débito su clonación, falsificación, el espionaje informático, falsificación de documentos, fraudes con telefonía celular, el phishing, entre otros.

Tiene como origen el mismo momento que surge la tecnología informática, la facilitación de las labores que trae consigo las computadoras propician que en un momento dado el usuario se encuentre ante una situación de ocio la cual canaliza a través de las computadoras cometiendo sin darse cuenta una serie de delitos.

Información electrónica

Aquella que se encuentra almacenada electrónicamente y a la cual se accede también electrónicamente. Esta definición incluye “formatos tangibles” como el CD-ROM y sus posibles sucesores, DVDs (Digital Video Discs --en ocasiones denominados Digital Versatile Discs) así como “no tangibles” formatos como las bases de datos electrónicas y los textos accesibles en línea.



Piratería: La piratería es una práctica de saqueo organizado o bandolerismo marítimo, probablemente tan antigua como la navegación misma. Consiste en que una embarcación privada o una estatal amotinada ataca a otra en aguas internacionales o en lugares no sometidos a la jurisdicción de ningún Estado, con el propósito de robar su carga, exigir rescate por los pasajeros, convertirlos en esclavos y muchas veces apoderarse de la nave misma. Su definición según el Derecho Internacional puede encontrarse en el artículo 101 de la Convención de las Naciones Unidas sobre el Derecho del Mar.

Junto con la actividad de los piratas que robaban por su propia cuenta por su afán de lucro, cabe mencionar los corsarios, un marino particular contratado que servía en naves privadas con patente de corso para atacar naves de un país enemigo. La distinción entre pirata y corsario es necesariamente parcial, pues corsarios como Francis Drake o la flota francesa en la Batalla de la Isla Terceira fueron considerados vulgares piratas por las autoridades españolas, ya que no existía una guerra declarada con sus naciones. Sin embargo, el disponer de una patente de corso sí ofrecía ciertas garantías de ser tratado como soldado de otro ejército y no como un simple ladrón y asesino; al mismo tiempo acarreaba ciertas obligaciones.

Internet

Es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

Uno de los servicios que más éxito ha tenido en Internet ha sido la World Wide Web (WWW, o "la Web"), hasta tal punto que es habitual la confusión entre ambos términos. La WWW es un conjunto de protocolos que permite, de forma sencilla, la consulta remota de archivos de hipertexto. Ésta fue un desarrollo posterior (1990) y utiliza Internet como medio de transmisión.

Existen, por tanto, muchos otros servicios y protocolos en Internet, aparte de la Web: el envío de correo electrónico (SMTP), la transmisión de archivos (FTP y P2P), las



conversaciones en línea (IRC), la mensajería instantánea y presencia, la transmisión de contenido y comunicación multimedia -telefonía (VoIP), televisión (IPTV)-, los boletines electrónicos (NNTP), el acceso remoto a otros dispositivos (SSH y Telnet) o los juegos en línea.

Sitios web

Un sitio web es una colección de páginas web relacionadas y comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.

Una página web es un documento HTML/XHTML accesible generalmente mediante el protocolo HTTP de Internet.

Todos los sitios web públicamente accesibles constituyen una gigantesca World Wide Web de información (un gigantesco entramado de recursos de alcance mundial).

A las páginas de un sitio web se accede frecuentemente a través de un URL raíz común llamado portada, que normalmente reside en el mismo servidor físico. Los URL organizan las páginas en una jerarquía, aunque los hiperenlaces entre ellas controlan más particularmente cómo el lector percibe la estructura general y cómo el tráfico web fluye entre las diferentes partes de los sitios.

Algunos sitios web requieren una suscripción para acceder a algunos o todos sus contenidos. Ejemplos de sitios con suscripción incluyen muchos portales de pornografía en Internet, algunos sitios de noticias, sitios de juegos, foros, servicios de correo electrónico basados en web, sitios que proporcionan datos de bolsa de valores e información económica en tiempo real, etc.

Tipos de Servidores:

En la siguiente lista hay algunos tipos de servidores:

Servidor de Archivo:

Tipo de servidor en una red de ordenadores cuya función es permitir el acceso remoto a archivos almacenados en él o directamente accesibles por este. En principio, cualquier ordenador conectado a una red con un software apropiado, puede funcionar como servidor de archivos. Desde el punto de vista del cliente de un servidor de archivos, la localización de los archivos compartidos es transparente. O sea, normalmente no hay



diferencias perceptibles si un archivo está almacenado en un servidor de archivos remoto o en el disco de la propia máquina.

Algunos protocolos comúnmente utilizados en servidores de archivos:

- SMB/CIFS (Windows, Samba en Unix)
- NFS (Unix)

Servidor de Impresiones:

Controla a una o más impresiones y acepta trabajos de impresión de otros clientes de la red, poniendo en cola los trabajos de impresión (aunque también puede cambiar la prioridad de las diferentes impresiones), y realizando la mayoría o todas las otras funciones que en un sitio de trabajo se realizará para lograr una tarea de impresión si la impresora fuera conectada directamente con el puerto de impresora del sitio de trabajo.

Proxy

Un proxy, en una red informática, es un programa o dispositivo que realiza una acción en representación de otro, esto es, si una hipotética máquina *a* solicita un recurso a una *c*, lo hará mediante una petición a *b*; *C* entonces no sabrá que la petición procedió originalmente de *a*. Su finalidad más habitual es la de servidor proxy, que sirve para interceptar las conexiones de red que un cliente hace a un servidor de destino, por varios motivos posibles como seguridad, rendimiento, anonimato, etc.

Servidor Proxy

Servidor HTTP

Este tipo de servidor opera en la Capa de aplicación de TCP/IP. El puerto de comunicación de entrada debe ser 80/http según IANA. Aunque generalmente suelen utilizar otros puertos de comunicación como el 3128, 8080 o el 8085.

Servidor HTTPS

Este tipo de servidor opera en la Capa de aplicación de TCP/IP. A diferencia de un Servidor HTTP, funciona bajo tecnologías de cifrado como SSL/TLS que proporcionan mayor seguridad y anonimato. El puerto utilizado varía, aunque debe ser 443/https.

Servicio Proxy o Proxy Web



Su funcionamiento se basa en el del Proxy HTTP y HTTPs, pero la diferencia fundamental es que la petición se realiza mediante una Aplicación Web embebida en un Servidor HTTP al que se accede mediante una dirección DNS, esto es, una página web que permite estos servicios.

Proxy Caché

Su método de funcionamiento es similar al de un proxy HTTP o HTTPs. Su función es precargar el contenido web solicitado por el usuario para acelerar la respuesta Web en futuras peticiones de la misma información de la misma máquina u otras.

Hojas Estilo Cascada

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

Por ejemplo, el elemento de HTML `<h1>` indica que un bloque de texto es un encabezamiento y que es más importante que un bloque etiquetado como `<h2>`. Versiones más antiguas de HTML permitían atributos extra dentro de la etiqueta abierta para darle formato (como el color o el tamaño de fuente). No obstante, cada etiqueta `<h1>` debía disponer de la información si se deseaba un diseño consistente para una página y, además, una persona que leía esa página con un navegador perdía totalmente el control sobre la visualización del texto.

Cuando se utiliza CSS, la etiqueta `<h1>` no debería proporcionar información sobre cómo será visualizado, solamente marca la estructura del documento. La información de estilo, separada en una hoja de estilo, especifica cómo se ha de mostrar `<h1>`: color, fuente, alineación del texto, tamaño y otras características no visuales, como definir el volumen de un sintetizador de voz, por ejemplo.



La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "style".

Visión General

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante sitio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. Este mismo sitio web de Wikipedia está desarrollado en PHP. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de sitios webs, es posible crear aplicaciones con una interfaz gráfica para el usuario, utilizando la extensión PHP-Qt o PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo; a esta versión de PHP se la llama PHP-CLI (Command Line Interface).



Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor Web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. El programa está liberado bajo la licencia GNU y actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS X.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

PHP es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C# y Visual Basic .NET como lenguajes), a ColdFusion de la empresa Adobe, a JSP/Java y a CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un entorno de desarrollo integrado comercial llamado Zend Studio. CodeGear (la división de lenguajes de programación de Borland) ha sacado al mercado un entorno de desarrollo integrado para PHP, denominado 'Delphi for PHP. También existen al menos un par de módulos para Eclipse, uno de los entornos más populares.

Historia

Fue originalmente diseñado en Perl, con base en la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador danés-canadiense Rasmus Lerdorf en el año



1994 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. El 8 de junio de 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio Form Interpreter para crear PHP/FI.

Dos programadores israelíes del Technion, Zeev Suraski y Andi Gutmans, reescribieron el analizador sintáctico (parser en inglés) en el año 1997 y crearon la base del PHP3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP3 y fue publicado oficialmente en junio de 1998.

Para 1999, Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como motor Zend. También fundaron Zend Technologies en Ramat Gan, Israel.

En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El día 13 de julio de 2007 se anunció la suspensión del soporte y desarrollo de la versión 4 de PHP, a pesar de lo anunciado se ha liberado una nueva versión con mejoras de seguridad, la 4.4.8 publicada el 13 de enero del 2008 y posteriormente la versión 4.4.9 publicada el 7 de agosto de 2008. Según esta noticia se dará soporte a fallos críticos hasta el 9 de agosto de 2008.

Qué es Dreamweaver?

Es una aplicación de estudio (basada en la forma de estudio de Adobe Flash) que está destinada a la construcción y edición de sitios y aplicaciones Web basados en estándares. Creado inicialmente por Macromedia (actualmente producido por Adobe Systems) es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium. Su principal competidor es Microsoft Expression Web y tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras. Hasta la versión MX, fue duramente criticado por su escaso soporte de los estándares de la web, ya que el código que generaba era con frecuencia sólo válido para Internet Explorer, y no validaba como HTML estándar. Esto se ha ido corrigiendo en las versiones recientes.

Se vende como parte de la suite Adobe Creative Suite.



La gran ventaja de este editor sobre otros es su gran poder de ampliación y personalización del mismo, puesto que en este programa, sus rutinas (como la de insertar un hipervínculo, una imagen o añadir un comportamiento) están hechas en Javascript-C, lo que le ofrece una gran flexibilidad en estas materias. Esto hace que los archivos del programa no sean instrucciones de C++ sino, rutinas de Javascript que hace que sea un programa muy fluido, que todo ello hace, que programadores y editores web hagan extensiones para su programa y lo ponga a su gusto.

Las versiones originales de la aplicación se utilizaban como simples editores WYSIWYG. Sin embargo, versiones más recientes soportan otras tecnologías web como CSS, JavaScript y algunos frameworks del lado servidor.

Dreamweaver ha tenido un gran éxito desde finales de los años 1990 y actualmente mantiene el 90% del mercado de editores HTML. Esta aplicación está disponible tanto para la plataforma MAC como para Windows, aunque también se puede ejecutar en plataformas basadas en UNIX utilizando programas que implementan las API's de Windows, tipo Wine.

Como editor WYSIWYG que es, Dreamweaver permite ocultar el código HTML de cara al usuario, haciendo posible que alguien no entendido pueda crear páginas y sitios web fácilmente sin necesidad de escribir código.

Algunos desarrolladores web criticaban esta propuesta ya que crean páginas HTML más largas de lo que solían ser al incluir mucho código inútil, lo cual va en detrimento de la ejecución de las páginas en el navegador web. Esto puede ser especialmente cierto ya que la aplicación facilita en exceso el diseño de las páginas mediante tablas. Además, algunos desarrolladores web han criticado Dreamweaver en el pasado porque creaba código que no cumplía con los estándares del consorcio Web (W3C).

No obstante, Adobe ha aumentado el soporte CSS y otras maneras de diseñar páginas sin tablas en versiones posteriores de la aplicación, haciendo que se reduzca el exceso de código.

Dreamweaver permite al usuario utilizar la mayoría de los navegadores Web instalados en su ordenador para previsualizar las páginas web. También dispone de herramientas de administración de sitios dirigidas a principiantes como, por ejemplo, la habilidad de



encontrar y reemplazar líneas de texto y código por cualquier tipo de parámetro especificado, hasta el sitio web completo. El panel de comportamientos también permite crear JavaScript básico sin conocimientos de código.

Con la llegada de la versión MX, Macromedia incorporó herramientas de creación de contenido dinámico en Dreamweaver. En lo fundamental de las herramientas HTML WYSIWYG, también permite la conexión a Bases de Datos como MySQL y Microsoft Access, para filtrar y mostrar el contenido utilizando tecnología de script como, por ejemplo, ASP (Active Server Pages), ASP.NET, ColdFusion, JSP (JavaServer Pages) y PHP sin necesidad de tener experiencia previa en programación.

Un aspecto de alta consideración de Dreamweaver es su arquitectura extensible. Es decir, permite el uso de "Extensiones". Las extensiones, tal y como se conocen, son pequeños programas, que cualquier desarrollador web puede escribir (normalmente en HTML y Javascript) y que cualquiera puede descargar e instalar, ofreciendo así funcionalidades añadidas a la aplicación. Dreamweaver goza del apoyo de una gran comunidad de desarrolladores de extensiones que hacen posible la disponibilidad de extensiones gratuitas y de pago para la mayoría de las tareas de desarrollo web, que van desde simple efectos rollover hasta completas cartas de compra.

También podría decirse, que para un diseño más rápido y a la vez fácil podría complementarse con fireworks en donde podría uno diseñar un menú o para otras creaciones de imágenes (gif web, gif websnap, gif adaptable, jpeg calidad superior, jpeg archivo más pequeño, gif animado websnap) para un sitio web y después exportar la imagen creada y así utilizarla como una sola, en donde ya llevara los vínculos a un dicho sitio en específico que uno le haya dado. (MRR - U.P.C.)

TOAD 2.1

Es una aplicación que no sólo permite diseñar esquemas de base de datos, sino también generar el código SQL necesario para producirlas. Con él puedes desarrollar diagramas para la mayor parte de sistemas gestores de bases de datos existentes: Access, Firebird, InterBase, MySQL, Oracle, Paradox, Postgre, Sybase y muchos más. La aplicación resulta muy útil a la hora de crear diagramas de entidad-relación, definir reglas de integridad referencial, generar scripts SQL que construyan la base de datos o detallados



informes en HTML y RTF. Además, posee una herramienta denominada 'Model Explorer' que permite navegar por todos los atributos del modelo que estemos creando. Shareware. Toad DataModeler 2.25 2.25

Características y Facilidades

Diseño – Crear estructuras de alta calidad, generarlas automáticamente, siguiéndolos estándares de Buenas Prácticas y metodologías de Diseño más avanzadas.

Documentar – Generar reportes detallados para documentar fácilmente las Bases de datos existentes.

Rediseñar – Tomar una Base de Datos, rediseñar el modelo y generar el SQL del nuevo diseño.

Migrar – Trasladar en forma simple las estructuras de una Base de Datos a otra versión o plataforma a la hora de realizar migraciones.

Sincronizar – Comparar un modelo con una Base de Datos existente, y generar automáticamente los scripts de "ALTER" (solo en Oracle) necesarios para lograr la sincronización de dos Bases de Datos

Toad Data Modeler permite

Crear estructuras de Bases de Datos en forma visual

Generar y hacer ingeniería reversa con facilidad

Documentar cambios al modelo de su base de datos

Generar SQL/DDL a partir de un modelo

Migrar fácilmente estructuras de datos de una versión o plataforma de Base de Datos a otra

Hacer cambios a un modelo creado por otra persona

Crear modelos de acuerdo a sus requerimientos de negocios incluyendo lógica en sus diagramas

Edición de sus modelos, triggers, imperativos de integridad, llaves foráneas, índices, tablas, atributos, etc. directamente en pantalla en forma interactiva.



Ajustar su interface de usuario a sus requerimientos específicos
Seguimiento de cambios con un administrador de versiones interno

Lenguaje HTML: siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

Que es el HTML?

El HTML, Hyper Text Markup Language (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomo dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX(el editor vi o ed) o el que ofrece MS Office (Word). El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o .html

Estos documentos pueden ser mostrados por los visores o "browsers" de páginas Web en Internet, como Netscape Navigator, Mosaic, Opera y Microsoft Internet Explorer. También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo, texto que vuela



desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página.

A continuación vamos a hablar un poco de historia:

Orígenes del HTML

1986. Publicación de la ISO 8879 que presenta el Standard General Markup Language, origen del HTML.

1989. Tim Berners-Lee, a la sazón en el Centro Europeo de Investigaciones Nucleares presenta su artículo Information Management: A Proposal dedicándose de lleno al desarrollo de un sistema que permitiera el acceso en línea de manera uniforme a la información disponible en muchos recursos distintos, y que pudiese funcionar en máquinas que conectadas por redes basadas en TCP/IP.

1990-1991. Tim Berners-Lee define el HTML como un subconjunto de SGML (Standard Generalized Markup Language), que más tarde se llamará nivel 0; soporta encabezados, listas y anclas. Se crea el nombre World Wide Web.

1991. Tim Berners-Lee introduce el primer visor de HTML, LineMode, que trabaja en modo texto y sólo en plataformas UNIX. El Centro Europeo de Investigaciones Nucleares realiza la apertura del primer sitio con acceso público de World Wide Web el 17 de mayo (<http://info.cern.ch>).

1992. Dan Connolly produce la primera Definición de Tipo de Documento (DTD) para el lenguaje, llamada HTML 1.0, agregando a la definición original atributos para modificar el estilo físico del texto. Se distribuye Viola, primer visor gráfico de Web y disponible sólo para X.11.

1993. Un nuevo visor que soporta un mayor nivel, Lynx, es producido por la Universidad de Kansas, si bien lee sólo texto. Aparece Mosaic, desarrollado por el Centro Nacional para Aplicaciones de Supercomputadoras, es el primer visor de Web en entorno gráfico que se hace disponible para computadoras personales, lo que lo hace inmediatamente popular. A fines de año, comienzan a aparecer los primeros artículos sobre WWW en diarios y revistas de circulación masiva. Tim Berners-Lee utiliza el



INSTITUTO TECNOLÓGICO SUPERIOR CORDILLERA

trabajo del año anterior de Connolly para presentar el borrador de la primera norma (RFC -Recommendation for Comments) de HTML para Internet.



2.1 Marco Legal

Ley N° 184 Especial de Telecomunicaciones, de 10 de agosto de 1992.

Decisión 344. Régimen Común sobre Propiedad Industrial de 21 de Octubre de 1993

Decisión 351. Régimen Común sobre Derechos de Autor y Derechos Conexos de 17 de diciembre de 1993.

Ley de Control Constitucional de 1997. que ha reglamentado la acción de habeas data.

Ley n° 83 de la Propiedad Intelectual de Ecuador de 8 de mayo de 1998.

Constitución Política de la República del Ecuador (Aprobada el 5 de junio de 1998, por la Asamblea Nacional Constituyente).

Proyecto de Ley Ecuatoriana sobre el Comercio Electrónico de 1999.

Reglamento a la Ley de la Propiedad Intelectual, de 25 de enero de 1999.

Proyecto de Ley de comercio electrónico, firmas electrónicas y mensajes de datos n° 21.315 de 6 de febrero de 2001. Comisión Especializada Permanente de lo Civil y Penal del Congreso de la República del Ecuador.

Ley de Propiedad Intelectual.

Ley de Estadísticas.

Ley 67/2002, de Comercio Electrónico, Firmas y Mensajes de Datos, de 27 de febrero de 2002. R.O. Suplemento 557 de 17 de abril del 2002.

Decreto Ejecutivo 3496. R.O. 735. Reglamento a la Ley de Comercio Electrónico de 12 de diciembre de 2002.

Ley Orgánica n° 24 de 4 de mayo de 2004, de Transparencia y Acceso a la Información Pública.

Decreto Ejecutivo n° 2471. Reglamento de la Ley Orgánica de Transparencia y Acceso a la información Pública. (Registro Oficial 507 de 19 de enero de 2005).

Decreto n° 1014 de 10 de abril de 2008, por el que se implementa el uso de Software Libre en la Administración Pública Central en sus sistemas y equipamientos informáticos.

Resolución 477-20-Conatel-2008. Consejo Nacional de Telecomunicaciones de 8 de octubre de 2008, para aprobar el modelo de acreditación como Entidad de Certificación de información y servicios relacionados.



Resolución 479-20-Conatel-2008. Consejo Nacional de Telecomunicaciones de 8 de octubre de 2008, para expedir el Reglamento para la organización y funcionamiento del Registro Público Nacional de Entidades de Certificación de información y servicios relacionados acreditadas y terceros vinculados.

Resolución 069-02-Conatel-2009. Consejo Nacional de Telecomunicaciones de 29 de enero de 2009, para aprobar la suscripción del memorando de entendimiento entre el Consejo Nacional de Telecomunicaciones de la República del Ecuador y el Ministerio del Poder Popular para las telecomunicaciones y la informática de la República Bolivariana de Venezuela.

Resolución 132-05-Conatel-2009. Consejo Nacional de Telecomunicaciones de 31 de marzo de 2009, para expedir la regulación de los centros de acceso a la información y aplicaciones disponibles en la red de internet.

DERECHOS DE PROPIEDAD INTELECTUAL

Para el desarrollo y creación del Centro de Capacitación, Tecnológica, nosotros como ejecutores del este proyecto final, tomaremos en cuenta, los artículos de la ley de Propiedad Intelectual, proporcionando el derecho y permiso al Instituto Tecnológico Superior Cordillera, para que puedan a futuro dar mejoras al tema de nuestro proyecto final, con nuevas reformas y demás actividades que competan al tema que estamos desarrollando, para beneficio de las diferentes comunidades que requieran de este tipo de ayudas.

Art. 74 de la Constitución Política de la República del Ecuador, en lo que se refiere a la Educación Superior. Consejo Nacional de Educación Superior: redacta la educación superior está conformada por universidades, escuelas politécnicas e institutos superiores técnicos y tecnológicos, cuya integración, atribuciones y obligaciones constarán en la ley; y en su sección 9ª de la Ciencia y la Tecnología redacta lo siguiente:

Art. 80 Función del Estado y de los Institutos Superiores: El Estado fomenta la Ciencia y la Tecnología, especialmente en todos los niveles Educativos, dirigidos a mejorar la productividad, la competitividad, el manejo sustentable de los recursos Naturales, y a satisfacer las necesidades básicas de la comunidad.



Dentro del artículo 80, en el numeral 4, habla de formar jóvenes capaces de utilizar y aplicar eficientemente sus saberes científicos y técnicos de nuevas alternativas de solución a las necesidades colectivas.

Ley de la Propiedad Intelectual.

En artículo primero el Estado garantiza la propiedad Intelectual, de conformidad con la ley, así:

Art.1. El Estado reconoce, regula y garantiza la propiedad intelectual adquirida de conformidad con la ley, las Decisiones de la Comisión de la Comunidad Andina y los convenios internacionales vigentes en el Ecuador.

La propiedad intelectual comprende:

1. Los derechos de autor y derechos conexos.

En el siguiente artículo contempla la igualdad de derechos para todos, así:

Art. 2. Los derechos conferidos por esta Ley se aplican por igual a nacionales y extranjeros, domiciliados o no en el Ecuador

En el siguiente artículo nos habla que, el Estado Ecuatoriano, estará bajo la supervisión de un Organismo que velará por los intereses, tanto de la nación como de su gente, así:

Art. 3. El Instituto Ecuatoriano de la Propiedad Intelectual (IEPI), es el Organismo Administrativo Competente para propiciar, promover, fomentar, prevenir, proteger y defender a nombre del Estado Ecuatoriano, los derechos de propiedad intelectual reconocidos en la presente Ley y en los tratados y convenios internacionales, sin perjuicio de las acciones civiles y penales que sobre esta materia deberán conocerse por la Función Judicial.

En el Artículo cuarto, se reconoce el derecho del Autor, así:

Art. 4. Se reconocen y garantizan los derechos de los autores y los derechos de los demás titulares sobre sus obras.

En el artículo quinto, nos habla de que el desarrollo de una obra, por cualquier autor, no importa su nacionalidad, su destino, su lugar de difusión, o su expresión, no serán alteradas, ni desprotegidas, así:

Art. 5. El derecho de autor nace y se protege por el solo hecho de la creación de la obra, independientemente de su mérito, destino o modo de expresión.



Se protegen todas las obras, interpretaciones, ejecuciones, producciones o emisión radiofónica cualquiera sea el país de origen de la obra, la nacionalidad o el domicilio del autor o titular. Esta protección también se reconoce cualquiera que sea el lugar de publicación o divulgación.

El reconocimiento de los derechos de autor y de los derechos conexos no está sometido a registro, depósito, ni al cumplimiento de formalidad alguna.

El derecho conexo nace de la necesidad de asegurar la protección de los derechos de los artistas, intérpretes o ejecutantes y de los productores de fonogramas.

En el artículo sexto, nos habla de que el autor de una obra, puede ser compatible con otros autores independientes, que puede modificar su obra, siempre que se rija y sea reconocida por la ley, así:

Art. 6. El derecho de autor es independiente, compatible y acumulable con:

La propiedad y otros derechos que tengan por objeto la cosa material a la que esté incorporada la obra;

Los derechos de propiedad industrial que puedan existir sobre la obra; y,

Los otros derechos de propiedad intelectual reconocidos por la ley.

En el artículo séptimo, nos habla de algunos tipos de términos, que dentro de esta ley, tendrán el siguiente significado y validez para su efecto, es así:

Art. 7. Para los efectos de este Título los términos señalados a continuación tendrán los siguientes significados:

Autor: Persona natural que realiza la creación intelectual.

Base de datos: Compilación de obras, hechos o datos en forma impresa, en una unidad de almacenamiento de ordenador o de cualquier otra forma.

Compilación: Agrupación en un solo cuerpo científico o literario de las distintas leyes, noticias o materias.

Copia o ejemplar: Soporte material que contiene la obra o producción, incluyendo tanto el que resulta de la fijación original como el que resulta de un acto de reproducción.

Divulgación: El acto de hacer accesible por primera vez la obra al público, con el consentimiento del autor, por cualquier medio o procedimiento conocido o por conocerse.



Obra en colaboración: La creada conjuntamente por dos o más personas naturales.

Obra colectiva: Es la creada por varios autores, por iniciativa y bajo la responsabilidad de una persona natural o jurídica, que la pública o divulga con su propio nombre, y en la que no es posible identificar a los autores o individualizar sus aportes.

LEY DE EDUCACIÓN

Ley Nº. 127 del 15 de abril de 1983

LA CÁMARA NACIONAL DE REPRESENTANTES

Considerando:

La necesidad de armonizar las normas sobre educación con los principios constantes en la Constitución Política; La utilidad de recoger las experiencias y el resultado del amplio debate en materia educativa y realizado entre los funcionarios del Ministerio de Educación y los representantes de los distintos sectores que conforman el área educativa; que es necesario, de conformidad con la estructura actual del Ministerio de Educación, dictar una ley específica consagrada únicamente a la educación, por las particularidades que tiene este sector; que es preciso establecer un marco claro y mecanismos ágiles para el desenvolvimiento de la educación nacional; y, En el ejercicio de la facultad contemplada en el artículo 66 de la Constitución

EXPIDE LA SIGUIENTE LEY DE EDUCACIÓN

Título Primero

PRINCIPIOS GENERALES

Capítulo I

OBJETO DE LA LEY

Art. 1.- La presente Ley tiene como objeto fijar los principios y fines generales que deben inspirar y orientar la educación, establecer las regulaciones básicas para el gobierno, organización y más funciones del sistema educativo y determinar las normas fundamentales que contribuyen a promover y coordinar el desarrollo integral de la educación.

Capítulo II

PRINCIPIOS Y FINES

Art. 2.- La educación se rige por los siguientes principios:



- a) La educación es deber primordial del Estado, que lo cumple a través del Ministerio de Educación y de las Universidades y Escuelas Politécnicas del país;
 - b) Todos los ecuatorianos tienen el derecho a la educación integral y la obligación de participar activamente en el proceso educativo nacional;
 - c) Es deber y derecho primario de los padres, o de quienes los representan, dar a sus hijos la educación que estimen conveniente. El Estado vigilará el cumplimiento de este deber y facilitará el ejercicio de este derecho;
 - d) El Estado garantiza la libertad de enseñanza de conformidad con la Ley;
 - e) La educación oficial es laica y gratuita en todos sus niveles. El Estado garantiza la educación particular;
 - f) La educación tiene sentido moral, histórico y social; se inspira en los principios de nacionalidad, democracia, justicia social, paz, defensa de los derechos humanos y está abierta a todas las corrientes del pensamiento universal;
 - g) El Estado garantiza la igualdad de acceso a la educación y la erradicación del analfabetismo;
 - h) La educación se rige por los principios de unidad, continuidad, secuencia, flexibilidad y permanencia;
 - i) La educación tendrá una orientación democrática, humanística, investigativa, científica y técnica, acorde con las necesidades del país; y,
1. La educación promoverá una auténtica cultura nacional, esto es, enraizada en la realidad del pueblo ecuatoriano.

Art. 3.- Son fines de la educación ecuatoriana:

- a) Preservar y fortalecer los valores propios del pueblo ecuatoriano, su identidad cultural y autenticidad dentro del ámbito latinoamericano y mundial;
- b) Desarrollar la capacidad física, intelectual, creadora y crítica del estudiante, respetando su identidad personal para que contribuya activamente a la transformación moral, política, social, cultural y económica del país;
- c) Propiciar el cabal conocimiento de la realidad nacional para lograr la integración social, cultural y económica del pueblo y superar el subdesarrollo en todos sus aspectos;



- d) Procurar el conocimiento, la defensa y el aprovechamiento óptimo de todos los recursos del país;
- e) Estimular el espíritu de investigación, la actividad creadora y responsable en el trabajo, el principio de solidaridad humana y el sentido de cooperación social;
- f) Atender preferentemente la educación preescolar, escolar, la alfabetización y la promoción social, cívica, económica y cultural de los sectores marginados; y,
- g) Impulsar la investigación y la preparación en las áreas: técnica, artística y artesanal.

Para cumplir a cabalidad con los fines de la educación, el Ministerio promoverá la participación activa y dinámica de las instituciones públicas y privadas y de la comunidad en general.

Título Segundo

ESTRUCTURA DEL SISTEMA EDUCATIVO

Capítulo I

ESTRUCTURA GENERAL

Art. 4.- (Reformado por el Art. 1 de la Ley 150, R.O. 918, 20-IV-92).- El sistema educativo nacional comprende dos subsistemas:

- a) El escolarizado; y,
- b) El no escolarizado.

En el sistema educativo nacional se garantiza la Educación Intercultural Bilingüe que, asimismo, comprenderá dos subsistemas:

- a) El escolarizado; y,
- b) El no escolarizado.

Art. 5.- El subsistema escolarizado se emplea en los establecimientos determinados por la Ley y comprende:

- a) La educación regular que se somete a las disposiciones reglamentarias sobre el límite de edad, secuencia de niveles y duración de cursos;
- b) La educación compensatoria, que tiene un régimen especial y se la ofrece a quienes no ingresan a los niveles de educación regular o no los concluyen; y,
- c) La educación especial, destinada a estudiantes excepcionales por razones de orden físico, intelectual, psicológico o social.



Art. 6.- El subsistema no escolarizado procura el mejoramiento educacional, cultural y profesional a través de programas especiales de enseñanza-aprendizaje y difusión, mediante los esfuerzos e iniciativas públicos o privados.

Art. 7.- La educación regular comprende los siguientes niveles:

a) Preprimario;

b) Primario; y, 1. Medio.

Art. 8.- La educación en el nivel preprimario tiende al desarrollo del niño y sus valores en los aspectos motriz, biológico, psicológico, ético y social, así como a su integración a la sociedad con la participación de la familia y el Estado.

Art. 9.- La educación en el nivel primario tiene por objeto la formación integral de la personalidad del niño, mediante programas regulares de enseñanza-aprendizaje y que lo habilitan para proseguir estudios en el nivel medio.

Art. 10.- La educación en el nivel medio comprende los tres ciclos: básico, diversificado y especializado.

Art. 11.- El ciclo básico inicia la formación del nivel medio en el que se promueve una cultura general básica y se desarrollan actividades de orientación, que permiten al estudiante seleccionar la especialidad en el ciclo diversificado y habilitarle para el trabajo.

Art. 12.- El ciclo diversificado procura la preparación interdisciplinaria que permita la integración del alumno a las diversas manifestaciones del trabajo y la continuación de los estudios en el ciclo post-bachillerato o en el nivel superior, atendiendo a los requerimientos del desarrollo social y económico del país y a las diferencias y aspiraciones individuales.

Las diversas modalidades se organizan de acuerdo con las necesidades del desarrollo científico, económico y cultural del país y aseguran, con sentido integral, la formación humanística y técnica.

Art. 13.- (Derogado por la Disposición Final de la Ley 2000-16, R.O. 77, 15-V-2000).

Art. 14.- La duración de los niveles pre-primarios, primarios y medio señalados en los artículos 8, 9, 11, 12 y 13 de esta Ley la fijará el Reglamento.

Art. 15.- El Ministerio de Educación y Cultura, en el ámbito de su competencia, podrá autorizar el funcionamiento experimental o piloto de unidades educativas.



Art. 16.- De conformidad con las disposiciones de la Constitución de la República y la Ley respectiva, corresponde a las universidades y escuelas politécnicas impartir la educación superior.

Art. 17.- El Ministerio de Educación establecerá mecanismos de coordinación con otros organismos del Estado o instituciones ajenas al mismo, que mantengan programas y servicios educativos.

Capítulo II

CARÁCTER GRATUITO Y OBLIGATORIO DE LA EDUCACIÓN

Art. 18.- La educación oficial es gratuita en todos los niveles.

Art. 19.- El Estado tiene el deber de facilitar el sostenimiento de la educación en todos los niveles y modalidades del sistema.

Art. 20.- La educación es obligatoria en el nivel primario y ciclo básico del nivel medio.

Art. 21.- Los establecimientos de educación particular no gratuitos se sujetarán, para el cobro de matrículas y pensiones, a las que fije el Ministerio de Educación.

Quienes no cumplan con lo dispuesto en el inciso precedente serán sancionados por el Ministerio de Educación con una multa equivalente a dos y hasta diez salarios mínimos vitales generales. En caso de reincidencia se clausurará temporal o definitivamente el establecimiento, según la gravedad de la falta en el año lectivo inmediatamente posterior.

El Ministerio de Educación, a través de las direcciones provinciales, publicará la lista de establecimientos particulares con las cantidades autorizadas para el cobro de matrículas y pensiones, al tiempo que se inicie el proceso de recepción de matrículas.

Conjuntamente con la lista publicará el presente artículo de esta Ley. La publicación se efectuará en los periódicos que se editen en cada cabecera provincial; caso de no haberlos, a través de las estaciones de las radios locales y en todo caso se exhibirá en las oficinas de las direcciones provinciales de educación.

Cualquier otro cobro, llámese bono, acción, contribución voluntaria, especial o cualesquiera otra denominación, podrá aplicarse solamente previa autorización del



Ministerio de Educación. Todos los cobros indebidos serán devueltos a los perjudicados por disposición de la institución sancionadora prevista en este artículo.

Art. 22.- Los planteles particulares que educan a niños y jóvenes de familias de escasos recursos, en los niveles indicados en el artículo anterior, recibirán del Estado, como aporte, hasta el 50% del costo alumno, cada año fiscal, de acuerdo al respectivo reglamento.

Título Tercero

ORGANIZACIÓN Y ADMINISTRACIÓN DEL SISTEMA EDUCATIVO

Capítulo I

EL MINISTERIO DE EDUCACIÓN

Art. 23.- El Ministerio de Educación y Cultura es responsable del funcionamiento del sistema educativo nacional, de la formulación y ejecución de la política cultural y deportiva y de la difusión del desarrollo científico y tecnológico.

Art. 24.- La autoridad superior del ramo es el Ministro de Educación.

Sus atribuciones y deberes en el área de la educación son:

- a) Desarrollar una política unitaria y definida, de acuerdo con los principios y fines previstos en la Constitución y en esta Ley;
- b) Aprobar los planes y programas que deben aplicarse a nivel nacional o regional y velar por su cumplimiento;
- c) Crear, reorganizar, clausurar o suprimir establecimientos educacionales, de acuerdo con esta Ley y los reglamentos respectivos;
- d) Autorizar o negar la creación de establecimientos de educación particular, suspenderlos o clausurarlos de conformidad con esta Ley y sus reglamentos; y,
- e) Las demás atribuciones que se fijan en esta Ley y en el Reglamento.



Capítulo II

EL CONSEJO NACIONAL DE EDUCACIÓN

Art. 25.- El Consejo Nacional de Educación es el organismo permanente de asesoramiento y consulta del Ministro en las políticas educativa, técnica, científica y en los asuntos sometidos a su conocimiento.

Art. 26.- El Consejo Nacional de Educación está integrado por:

- a) El Ministro de Educación o su Representante, que debe ser un funcionario del Ministerio, quien lo preside;
- b) Un Rector universitario o politécnico representante del Consejo Nacional de Universidades y Escuelas Politécnicas;
- c) Dos representantes por el Magisterio Nacional, uno por el Magisterio Fiscal y otro por el Magisterio Particular no universitario, quienes serán designados de conformidad con el respectivo reglamento; y,
- d) Un representante del Consejo Nacional de Desarrollo.

Los miembros mencionados en los literales b) y c) durarán dos años en sus funciones, pudiendo ser reelegidos después de un período.

Todos los miembros contarán con su respectivo suplente.

Art. 27.- El Consejo Nacional de Educación se regirá por esta Ley y su Reglamento.

Capítulo III

ESTRUCTURA DEL MINISTERIO

Art. 28.- (Agregado el inc. 2o. por el Art. 2 de la Ley 150, R.O. 918, 20-IV-92).- El Ministerio de Educación cuenta para su funcionamiento, además de las Subsecretarías, con las direcciones nacionales especializadas y las oficinas técnicas que se determinen en el Reglamento, de acuerdo con los requerimientos del desarrollo educativo del país. Además contará con una organización integrada por unidades de asesoramiento, de planificación y de ejecución.

La Dirección Nacional de Educación Intercultural Bilingüe, especializada en culturas y lenguas aborígenes, funcionará como una organización técnica, administrativa y financiera descentralizada, tendrá su propia estructura orgánico-funcional, que



garantizará la participación, en todos los niveles e instancias de la administración educativa, de los pueblos indígenas, en función de su representatividad.

Art. 29.- Las Direcciones Provinciales de Educación son las responsables de la organización y de la aplicación del sistema educativo en la actividad docente y discente con relación a los niveles preprimario, primario y medio, en su respectiva jurisdicción.

Art. 30.- El Ministerio regulará, supervisará y coordinará las actividades de las instituciones, empresas especializadas y más organismos descentralizados o no, públicos, relacionados con el cumplimiento de los objetivos y fines determinados en la presente Ley.

Art. 31.- Los establecimientos educativos son centros de formación humana y promoción cultural, destinados a cumplir con los fines de educación determinados en esta Ley.

Art. 32.- Los establecimientos educativos oficiales son: fiscales, municipales y de otras instituciones públicas.

Los establecimientos particulares son los promovidos, dirigidos y pertenecientes a personas naturales o jurídicas de derecho privado. Son también establecimientos educativos, para los efectos de esta Ley, los que cuentan con financiamiento parcial del Estado y se rigen por los convenios especiales.

Todos los establecimientos educativos se someten, para su funcionamiento, a lo prescrito por esta Ley y su Reglamento.

Art. 33.- Los organismos directivos de los establecimientos educacionales y sus niveles de autoridad se establecerán en el Reglamento.

Art. 34.- Especialmente en las áreas suburbanas, rurales y fronterizas, el Ministerio organizará, obligatoriamente, programas que tienden al desarrollo integral de la comunidad.

Art. 35.- En los establecimientos educativos podrán crearse asociaciones estudiantiles con fines culturales, sociales, deportivos y de investigación científica. Su funcionamiento se sujetará al Reglamento.

En cada uno de dichos establecimientos podrá funcionar un Comité de Padres de Familia cuyas atribuciones y deberes serán los determinados en el Reglamento.



El establecimiento que de cualquier manera impida o coarte a los padres de familia el ejercicio de este derecho, será sancionado de acuerdo con la Ley.

Capítulo IV

PLANIFICACIÓN

Art. 36.- Los planes y programas educativos deben ser formulados científicamente de conformidad con las orientaciones de la política educativa y las necesidades del desarrollo de la realidad nacional, para lo cual deberá contar con la participación de los sectores que conforman la acción educativa, con sujeción al Reglamento.

Capítulo V

DE LA ORIENTACIÓN Y BIENESTAR ESTUDIANTILES

Art. 37.- El Ministerio de Educación planificará y ejecutará obligatoriamente la orientación dentro de un criterio integral, considerándola consubstancial al proceso educativo y como una acción interdisciplinaria con responsabilidad participativa de la comunidad educativa, conducida por profesionales especializados y desarrollada al interior del currículum.

Los servicios de bienestar estudiantil serán aplicados de igual manera con criterio especializado para facilitar un óptimo aprovechamiento de los recursos humanos, económicos y materiales en el proceso educativo, dentro de un contexto social.

Capítulo VI

SUPERVISIÓN EDUCATIVA

Art. 38.- La supervisión es una función especializada encargada de velar por el cumplimiento de los fines y de las normas de educación y de promover el mejoramiento de la enseñanza y del desarrollo de las comunidades, mediante una acción sistemática y permanente.

Art. 39.- La supervisión, siendo principalmente un servicio de asesoramiento profesional y de control, tomará además en cuenta las opiniones de la comunidad, a fin de que el proceso educativo se desenvuelva en función de las necesidades de las mismas.



Art. 40.- Para el desempeño del cargo de supervisor es indispensable poseer preparación especializada, residir por lo menos en los días laborables en el área del lugar de trabajo que le ha sido asignada, además de reunir los requisitos que determina el reglamento.

Capítulo VII

INFRAESTRUCTURA EDUCATIVA

Art. 41.- La planificación técnica, ejecución, control, fiscalización y mantenimiento de las construcciones destinadas a establecimientos educativos fiscales, estarán a cargo de la Dirección Nacional de Construcciones Escolares.

Notas:

-La Dirección Nacional de Construcciones Escolares DINACE ha sido suprimida mediante D.E. 683 (R.O. 149-S, 16-III-99), pasando sus bienes, derechos, obligaciones, funciones y atribuciones al Ministerio de Educación, Cultura, Deportes y Recreación.

Nota:

El Art. 16 lit. e) del Estatuto del Régimen Jurídico y Administrativo de la Función Ejecutiva, cambió la denominación del Ministerio de Educación Pública, Ministerio de Educación, Cultura, Deportes y Recreación a Ministerio de Educación y Cultura (D.E. No. 66, R.O. 11, 30-I-2003), cambio de denominación que posteriormente fue ratificado por Art. 2 del D.E. No.1179, R.O. 239, 24-XII-2003.

Art. 42.- La planificación técnica se hará en base de la investigación de las necesidades nacionales y de la determinación de prioridades y características de la infraestructura educativa dentro del mapa escolar.

Art. 43.- (Sustituido por el Art. 3 de la Ley 150, R.O. 918, 20-IV-92).- En las construcciones escolares de las zonas rurales se tendrá en cuenta las características socio-culturales y arquitectónicas de la comunidad. Las necesidades de vivienda para los maestros y los servicios para la promoción social y cultural podrán llevarse a cabo con la participación de los Organismos del Estado y de la comunidad.

Art. 44.- Todo proyecto de desarrollo urbano o rural debe contemplar, para su aprobación, la reserva de áreas para servicios educativos y la construcción de locales para la educación obligatoria.



Título Cuarto

EL RÉGIMEN ESCOLAR

Capítulo I

EL AÑO LECTIVO, EL INGRESO Y LA EVALUACIÓN

Art. 45.- El año lectivo comprenderá ciento ochenta y cinco días laborables, incluidos los períodos de exámenes. Los calendarios, períodos y días de descanso para los regímenes de Sierra y Oriente, Costa y Galápagos, serán establecidos por el Ministerio de Educación.

Art. 46.- El Ministerio de Educación posibilitará la atención a la población escolar que requiera del cambio de régimen escolar evitando la pérdida de tiempo. Para el efecto tenderá a crear, al menos en todas las capitales de provincias, establecimientos, tanto de nivel primario como de nivel medio, que funcionen según el calendario de los establecimientos del otro régimen escolar.

Art. 47.- Los planteles educativos podrán adoptar el sistema de una o dos jornadas de trabajo diario, previa la justificación de su conveniencia y la autorización del Ministro. En cualquier tipo de jornada, los establecimientos educativos se sujetarán a los períodos de clase determinados en el Reglamento.

Art. 48.- La admisión de alumnos en los diversos niveles y modalidades del sistema educativo se regirá por esta Ley y el Reglamento.

Art. 49.- La evaluación en todos los niveles y etapas de la educación será permanente e integral.

Art. 50.- Los establecimientos educativos nocturnos estarán sujetos a un régimen adecuado para el cumplimiento de su finalidad, adoptando las regulaciones necesarias para que estos planteles cumplan con los objetivos y fines educativos contemplados en la Ley y su Reglamento.



Título Quinto

PERSONAL DOCENTE Y ADMINISTRATIVO

Capítulo I

EL MAGISTERIO NACIONAL

Art. 52.- El Magisterio Nacional está formado por los profesionales de la educación y por aquellos que cumplan labores docentes o que desempeñen funciones técnico-administrativas especializadas en el sistema educativo.

Art. 53.- Quienes poseen título de profesionales de la educación tienen derecho prioritario para ser nombrados en funciones del ramo educativo.

Art. 54.- La Ley y los Reglamentos que establecen las categorías del escalafón y remuneraciones del Magisterio regularán el ejercicio docente. En los establecimientos particulares la remuneración no podrá ser menor al sueldo básico del Magisterio para quienes laboran a tiempo completo; y los que trabajen a tiempo parcial recibirán al menos la alícuota correspondiente.

Art. 55.- El Ministerio de Educación procurará que los profesores en el área rural residan en sus lugares de trabajo, para lo cual se preferirá, en la designación de profesores, a personas del lugar que cumplieren los requisitos legales.

ESCALAFÓN

Art. 56.- El escalafón del Magisterio Nacional es el sistema legal del ejercicio profesional de sus miembros.

Art. 57.- Los deberes y derechos del personal que ejerce sus funciones en los establecimientos oficiales son los determinados en la Ley de Escalafón y Sueldos del Magisterio Nacional.

Capítulo II

DE LA INVESTIGACIÓN PEDAGÓGICA, FORMACIÓN Y PERFECCIONAMIENTO DOCENTES

Art. 58.- La investigación pedagógica, la formación, la capacitación y el mejoramiento docentes son funciones permanentes del Ministerio de Educación destinadas a lograr la



actualización del Magisterio para asegurar un eficiente desempeño en el cumplimiento de los fines de la educación nacional.

Art. 59.- La formación de profesionales docentes estará a cargo de las facultades de filosofía, letras y ciencias de la educación y de los colegios e institutos normales del país. Estos últimos, los colegios e institutos, tienen la finalidad específica de preparar docentes para los niveles preprimario y primario.

En cuanto a los colegios e institutos normales el Ministerio fijará los cupos de estudiantes en relación a la demanda nacional de este tipo de profesionales y a las disponibilidades fiscales del país.

Título Sexto

DE LOS BIENES Y RECURSOS

Art. 60.- El Estado destinará los bienes y recursos necesarios para el funcionamiento de los establecimientos educativos, de acuerdo a las normas establecidas por la Constitución, la Ley, el Reglamento y las disponibilidades presupuestarias.

Art. 61.- El Ministerio de Educación adoptará las medidas que propendan a la racionalización del gasto y al eficiente aprovechamiento de sus recursos financieros y materiales.

Art. 62.- En la asignación de recursos por programas se dará prioridad a la educación del sector rural y de las zonas fronterizas.

Art. 63.- Las asignaciones e ingresos de cualquier orden, destinados a los programas de educación, no podrán ser invertidos en otro objetivo diferente al previsto.

Art. 64.- Los directivos de los planteles educativos son los responsables de velar por la administración y conservación de los recursos asignados a los mismos.

Título Séptimo

DISPOSICIONES GENERALES

Art. 65.- Las atribuciones que esta Ley concede a los diversos funcionarios de la educación serán ejercidas privativamente por ellos dentro de la organización jerárquica establecida.

Art. 66.- El Ministro de Educación delegará atribuciones o designará representantes, en la esfera de la competencia, a las personas que estime conveniente, quienes responderán



directamente por los actos realizados en el ejercicio de la respectiva delegación. Esta delegación deberá recaer preferentemente en funcionarios del Ministerio.

Art. 67.- El censo escolar debe realizarse de acuerdo a las normas técnicas establecidas reglamentariamente. La participación del personal directivo, docente y técnico administrativo en estos eventos es obligatoria.

Art. 68.- Las instituciones y entidades de los diversos sectores de la economía nacional están obligadas a financiar programas educativos para su personal en beneficio de la comunidad. Los gastos que demanden estos programas serán deducibles del pago del impuesto a la renta.

Los medios de comunicación social están obligados a conceder espacios gratuitos para la difusión de programas elaborados por el Ministerio de Educación de conformidad con el Reglamento.

Artículo Final.- Deróganse la Ley de Educación y Cultura expedida por Decreto No. 1903, de 10. de noviembre de 1977, promulgada en el Registro Oficial No. 461, de 11 de los mismos mes y año y la Reforma expedida por Decreto No. 2144, de 12 de enero de 1978, promulgada en el Registro Oficial No. 511 de 23 de los mismos mes y año así como las demás disposiciones legales que se opongan a la presente Ley, la misma que entrará en vigencia luego de su promulgación.

Dado en Quito, a los 15 días del mes de abril de mil novecientos ochenta y tres.



CAPITULO III

INVESTIGACIÓN CIENTÍFICA

3.1 TIPOS DE INVESTIGACIÓN

3.1.1 Investigación Cuantitativa

Significa que se busca reducir los datos de nuestra investigación con el fin de expresarlos y describirlos de alguna manera (conceptual, numérica o gráficamente), de tal manera que respondan a una estructura sistemática, inteligible para otras personas, y por lo tanto significativa. La reducción de datos es una clase de operación que se realiza a lo largo de todo el proceso de investigación y puede hacerse de distintas formas (conceptual, numérica o gráficamente, como se indicó), pero que en la investigación cualitativa se refiere más que nada a la categorización y dosificación de los datos.

3.1.2 Investigación Documental

Este tipo de investigación es la que se realiza, como su nombre lo indica, apoyándose en fuentes de carácter documental, esto es, en documentos de cualquier especie. En materia de seguros, impera por su importancia el elemento contable y sus comprobantes. En el Internet puede constatar la gran variedad que existe colaborando con los conceptos aprendidos en clase así como la experiencia propia en el desarrollo de sistemas por otro lado como las revistas y catálogos, métodos y teorías prácticas lo que me llevo aún entendimiento universal de los temas tratados; los catálogo y manuales técnicos definieron la estrategia a seguir el modelo físico de mi tema llegando a obtener valiosas conclusiones que al mismo tiempo rectificaron las tendencias tecnológicas que se puso en práctica en mi proyecto, de tal motivo este tipo de investigación documental, en la estructura ideológica de conceptos, prácticas y métodos relacionados con el tema investigativo que se llevó a cabo.

3.1.2 Investigación Descriptiva



La investigación descriptiva consiste en llegar a conocer las situaciones, costumbres y actitudes predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas. Su meta no se limita a la recolección de datos, sino a la predicción e identificación de las relaciones que existen entre dos o más variables. Los investigadores no son meros tabuladores, sino que recogen los datos sobre la base de una hipótesis o teoría, exponen y resumen la información de manera cuidadosa y luego analizan minuciosamente los resultados, a fin de extraer generalizaciones significativas que contribuya al conocimiento

Conocer el tipo de metodología "método científico" se entiende aquellas prácticas utilizadas, como válidas a la hora de proceder con el fin de exponer y confirmar sus teorías en la aplicación.

La Metodología que se utilizara en este proyecto es la descriptiva que se utiliza para recoger, organizar, resumir, presentar, analizar, generalizar los resultados del la observación. Esta metodología implica la recopilación y presentación sistemática de datos para dar una idea clara de una determinada situación.

La ventaja de esta metodología es fácil de comprender, de corto tiempo y económica.

3.1.3 Investigación Explicativa

Este tipo de investigación se aplico en el estudio del planteamiento del problema ya que me permitió determinar las causas que generaron la problemática (enumerar la problemática 2) y de igual manera el efecto que tuvieron las mismas en el desarrollo del proyecto; normalmente pudiendo contestar las siguientes interrogantes porque ocurrió El hecho o fenómeno y a su vez la misma relacionándola con motivos de quien lo genero a fin de poder descubrir las causas y efectos que motivaron la ejecución de los mismos. Igualmente en la aplicación de este tipo de investigación que se fundamenta en la observación directa permitió determinar los conocimientos adquiridos en forma teórica y poderlos materializar en una forma práctica a través de las soluciones planteadas a través de soluciones informáticas acordes a solucionar el problema planteado

3.2 MÉTODOS DE INVESTIGACIÓN

3.2.1 Método Deductivo



Este método me permite analizar la información de lo general a lo particular, el mismo que me permite el desarrollo del trabajo investigativo realizado, se inicia analizando los parámetros que conforman el desarrollo del proyecto, Desarrollar e implementar una plataforma basado en los delitos informáticos, a fin de que la comunidad ecuatoriana esté informada acerca de estos casos.

Parte de una premisa general para obtener las conclusiones de un caso particular. Pone el énfasis en la teoría, modelos teóricos, la explicación y abstracción, antes de recoger datos empíricos, hacer observaciones o emplear experimentos.

El método de deductivo: en la ciencia: se basa en ir encadenando conocimientos que se suponen verdaderos de manera tal que se obtienen de nuevos conocimientos; es decir, es aquel que combina principios necesarios y simples (axiomas postulados, teoremas, conceptos no definidos, definiciones, etc.) para deducir nuevas proposiciones. También se llama método analítico o indirecto cuya característica es que va de la general a lo particular.

3.2.2 Método Inductivo Este método me permite analizar la información desde lo particular a lo general, por consiguiente en el trabajo de investigación realizada, se iniciara analizando los casos que conforman el ambiente de trabajo.

La conclusión es sacada del estudio de todos los elementos que forman el objeto de investigación, es decir que solo es posible si conocemos con exactitud el número de elementos que forman el objeto de estudio y además, cuando sabemos que el conocimiento generalizado pertenece a cada uno de los elementos del objeto de investigación

Los elementos del objeto de investigación no pueden ser numerados y estudiados en su totalidad, obligando al sujeto de investigación a recurrir a tomar una muestra representativa, que permita hacer generalizaciones.

Es un método utilizado en objetos de investigación cuyos elementos son muy grandes o infinitos. Se infiere una conclusión universal observando que un mismo carácter se repite en una serie de elementos homogéneos, pertenecientes al objeto de investigación, sin que se presente ningún caso que entre en contradicción o niegue el carácter común observado. La mayor o menor probabilidad en la aplicación del método, radica en el



número de casos que se analicen, por tanto sus conclusiones no pueden ser tomadas como demostraciones de algo, sino como posibilidades de veracidad. Basta con que aparezca un solo caso que niegue la conclusión para que esta sea refutada como falsa.

3.3.3 Método analítico sintético

El método analítico sintético se ha aplicado al proyecto ya que con el mismo me permite realizar un análisis a profundidad de cada uno de los elementos que intervienen en el desarrollo de un software informático es decir una metodología diseño estructuración de base de datos etc. lo que llevado a la incorporación de un lenguaje de programación van a permitir la sistematización en la organización ; por otro lado el realizar una síntesis de la información teórica y conceptual nos permitirá reunir todos los argumentos válidos a fin de reestructurar toda la lógica didáctica y conceptual de todo elemento que interviene en el desarrollo de mi tema. Pudiendo concentrar el mayor esfuerzo en la búsqueda de información y análisis de la misma pero únicamente con la salvedad que sea aplicable estrictamente al tema en ejecución. Por consiguiente el método analítico sintético tiene su fundamentación y aplicación en cómo vamos a tratar la información teórica necesaria para la estructuración del capítulo segundo de este tema de implementación.

Sistemática

Este método nos conduce a la estructuración lógica del proyecto ya que tendrá que seguir un orden definido en la estructuración de cada capítulo lo que permitirá a cualquier lector ir verificando paso a paso el análisis de la problemática que objetivos se persigue como voy a aplicar mi investigación y finalmente en el desarrollo de un software informático que será la vinculación de las partes constitutivas del proyecto y al mismo tiempo el objetivo general que se persigue al automatizar todos los procesos y procedimientos de la entidad

3.3 Técnicas de Recolección de la Información

3.3.1 Entrevista

La entrevista como herramienta, es un diálogo entablado entre dos o más personas: el entrevistador o entrevistadores que interrogan y el o los entrevistados que contestan, es



por eso que la e incorporado como base principal para ayuda del personal técnico especializado en la materia así como personal administrativo de la empresa que de alguna manera contribuya a la orientación y asesoramiento del proyecto con el fin de que el mismo sea un proyecto sustentable.

Preguntas de la Entrevista

Pregunta

1.- ¿Apoyaría usted el desarrollo de una página web informática sobre delitos realizados en Ecuador?

Respuesta

Si apoyaría ya que tener una página web donde nos podemos enterar sobre los delitos y las técnicas que utilizan los delincuentes para robar.

Análisis

Tener un buen material de información a la mano hace que las personas se instruyan en los casos de actualidad

Pregunta

2.- ¿En donde se presenta la mayor parte del problema?

Respuesta

Se presenta el problema por el desconocimiento de acciones de seguridad que deberíamos saber para protegernos.

Análisis

La falta de interés por conocer, y escaso recurso de adquisición tiene un resultado de desconocimiento hacia dicho problema

Pregunta

3.- ¿Conoce del manejo de alguna plataforma virtual parecida a la que se va a desarrollar?

Respuesta

La verdad es muy escasa por no decir ninguna página web sobre este tema.

Análisis



Las pocas páginas web puestas en el internet contienen poca información sobre el tema

Pregunta

4.- ¿Para quién va transmitida la plataforma virtual?

Respuesta

Para todo público registrado y mas aquellas personas que necesiten información y utilización de técnicas, aplicaciones, utilizadas para manejo de información.

Análisis

-Es un tema tan grande e importante que abarca a todas las personas vinculadas en la tecnología.

Pregunta

5.- ¿Los usuarios que van a utilizar la pagina web necesitarían capacitación extra?

Respuesta

No, la página está diseñada para un fácil manejo y entendimiento para que los usuarios puedan manejarla a su conveniencia.

Análisis

El fácil manejo implica que las personas puedan investigar, aprender y defenderse sobre los delitos producidos.

Es importante recalcar la importancia de la entrevista y la ayuda que da pues la información recopilada será utilizada en los requerimientos de la plataforma virtual, en la comunicación con el usuario con el fin de informar y comunicar.



CAPITULO IV

DESARROLLO DE LA PROPUESTA

4.1. Diagnóstico Situacional

Actualmente el país cuenta con páginas web con muy poca información sobre delitos informáticos, el cual no es muy favorable, es por ello que tanto, los administradores, usuarios registrados, deben estar permanente actualizando información.

Por tratarse de un tema de tesis comunitario que se trata de la implementación de la biblioteca virtual dirigida hacia la red en el cual el usuario vera nuestra biblioteca virtual en un centro de cómputo o en su propio hogar

El control de la pagina web es muy fundamental en nuestro país, ya que cada persona podrá adquirir conocimiento de cada tema, vemos muchos vacios sobre la información que tienen las personas hacia como tener un buen manejo de la tecnología es por ello la necesidad de implementar nuestra plataforma virtual.

Respecto al software como tiene un servicio de internet se maneja paquetes de office, mensajería instantánea. Para navegar por la web básicamente la mayoría de usuarios utiliza el navegador que tiene instalado en su estación de trabajo

En Telecomunicaciones actualmente se cuenta con una red LAN, la topología es en forma de estrella. El servicio de Internet se lo comparte mediante configuraciones específicas de un router, conectado hacia un switch y este distribuye a todas las estaciones de trabajo.

En seguridad, no existen seguridades periféricas estrictas para la DMZ. El Firewall utilizado es el propio del Sistema Operativo Windows Xp, Antivirus Avast, y el acceso a ciertas páginas web se las restringe a cada una de las estaciones de trabajo por medio de la carpeta System.



4.2 ESTRUCTURA ORGANIZACIONAL

Teniendo en cuenta que la Estructura Organizacional es de vital importancia en una entidad que contenga departamentos de trabajo y estos a la vez, empleados que ocupen cargos para el rendimiento óptimo de una empresa grande, ahora bien en nuestro caso no es necesario tener este tipo de estructura por el motivo de que nuestro tema de tesis está dirigido hacia la comunidad y no para una empresa específica.

Equipo	Ubicación	Departamento
Switch D-Link	Servidor	Dep. de Sistemas
Cable UTP Cat 5	Servidor, Estaciones de Trabajo	Dep. de Sistemas
Conectores RJ45	Servidor, Estaciones de Trabajo	Dep. de Sistemas
Tarjetas de Red	Servidor, Estaciones de Trabajo, Contabilidad	Dep. de Sistemas



4.3 INFRAESTRUCTURA INFORMÁTICA

4.1.1. Hardware

Características	Ubicación	Arquitectura
Intel Core 2 Duo	1 estación de trabajo	CISC
Intel Core i5	Servidor	CISC

Tabla # 1

Hardware

Fuente: internet

4.1.2. Software

Software	Ubicación	Licencia
Microsoft Office 2007 Profesional	Estaciones de Trabajo, Servidor, Contabilidad, Ventas	Si
Windows XP Profesional SP2	Estaciones de Trabajo,	Si
Mozilla Firefox	Estaciones de Trabajo,	Libre
Paquete de Mensajería	Estaciones de Trabajo	Libre

Tabla # 2

Software

Fuente: internet

4.1.3. Comunicaciones

Equipo	Ubicación	Departamento
Switch D-Link	Servidor	Dep. de Sistemas
Cable UTP Cat 5	Servidor, Estaciones de	Dep. de Sistemas



	Trabajo	
Conectores RJ45	Servidor, Estaciones de Trabajo	Dep. de Sistemas
Tarjetas de Red	Servidor, Estaciones de Trabajo, Contabilidad	Dep. de Sistemas

Tabla # 3

Comunicaciones

Fuente: internet

4.1.4. Recurso Humano Técnico

Nombre	Descripción
Ing. Jaime Padilla	Tutor de Tesis
Ing. Robert Enríquez	Director Escuela de Sistemas ITSCO

Tabla # 4

Recurso Humano Técnico

Fuente: Propia del Autor.

4.4 ALTERNATIVA 1

Con la implementación del este sistema las características que posee esta aplicación se serán más rápidas al utilizar la metodología RUP esta nos ayudara a mejorar el tiempo haciendo que nuestro sistema tenga respuestas optimas asía el usuario se utilizara esta metodología porque se apega a nuestro modelo de base de datos.

PLATAFORMA	Web
METODOLOGIA	Rup



MODELO DEL NEGOCIO	Casos de uso, Diagrama de clases, Diagrama de secuencia, Diagrama de estado, Diagrama de colaboración , diagrama de caso de uso, diagrama de colaboración, diagrama de secuencias
FRONT END	Php, cms, css, html
BACH END	MySql
PRUBAS DE FUNCIONAMIENTO	Prueba de integración Prueba de Unidad Prueba de validación Prueba de sistema Prueba de seguridad
IMPLEMENTACION	Servidor, En una red
CAPACITACION DE USUARIO	Videos
CAPACITACION TECNICA	Tutoriales en la web Administradores
MANUALES TECNICOS	Administradores
ESTANDARES	Diccionario de datos Base de datos Tablas Relaciones Dependencia Fk Pk

Soporte: cada 4 meses

Costo: \$800

4.3.4 RECURSO HUMANO TECNICO

Nombre	Descripción
--------	-------------



Bolivar Ribadeneira	Alumno
Diego Pilatuña	Alumno
Ing. Jaime Padilla	Tutor

4.4 ALTERNATIVA

Con la implementación de este sistema utilizamos una metodología RAD unas habilidades de este sistema el tiempo de entrega del software es a corto plazo la respuesta de comunicación con el usuario es óptima, pero la metodología RUP las pruebas se hacen a la par de lo que se va realizando el software

PLATAFORMA	Web
METODOLOGIA	Rup
MODELO DEL NEGOCIO	Casos de uso, Diagrama de clases, Diagrama de secuencia, Diagrama de estado, Diagrama de colaboración
FROND END	Php, cms, css, html
BACH END	MySQL
PRUBAS DE FUNCIONAMIENTO	Prueba de integración Prueba de Unidad Prueba de validación Prueba de sistema Prueba de seguridad
IMPLEMENTACION	Servidor, En una red
CAPACITACION DE USUARIO	Videos Tutoriales en la web Administradores
CAPACITACION TECNICA	Administradores
MANUALES TECNICOS	Diccionario de datos
ESTANDARES	Base de datos

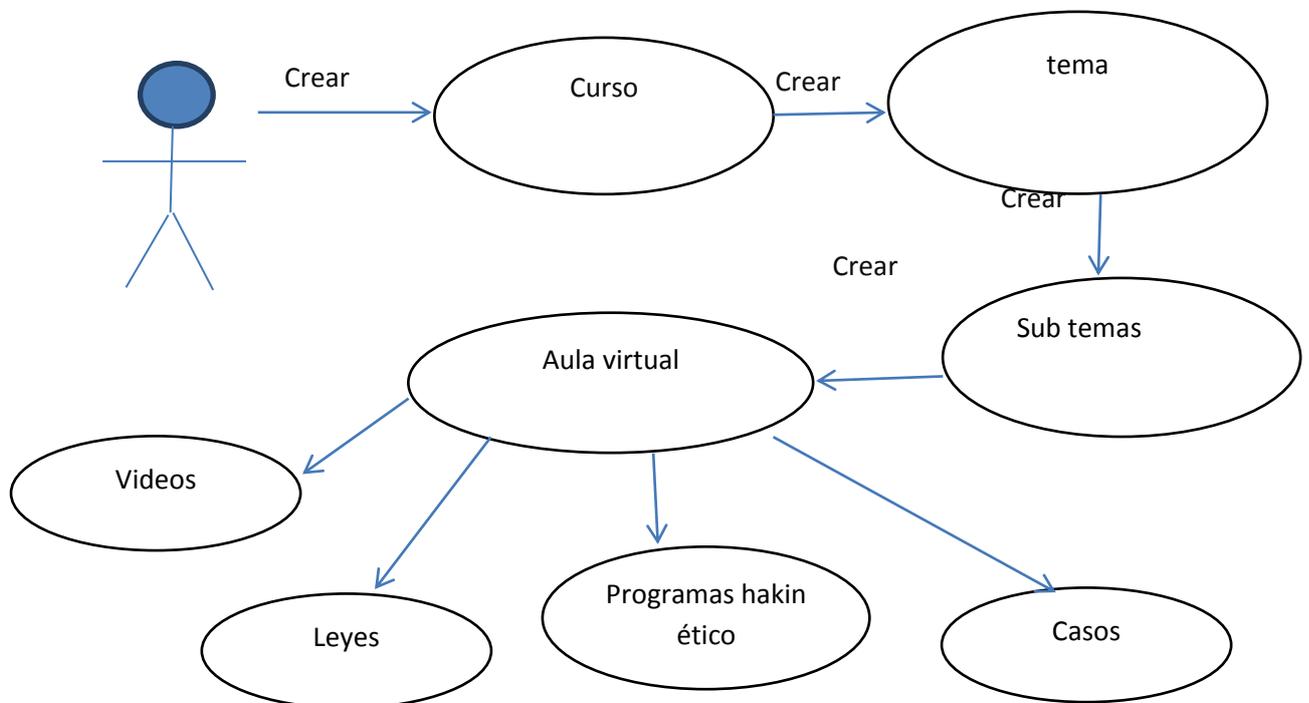
	<p>Tablas</p> <p>Relaciones</p> <p>Dependencia</p> <p>Fk</p> <p>Pk</p>
--	--

Soporte: cada 5 meses

Costo: \$850

4.5 DESCRIPCIÓN DE PROCESOS

En el diagrama observamos que tenemos un administrador que se la persona que desarrolla el proyecto, el cual tiene accesos a crear un curso que tendrá temas y subtemas que contera directamente el Aula virtual tendrá videos, leyes de informática, programas de hakin ético, casos, etc. El cual el administrador podrá ver que usuario ingreso por su fecha de inicio y fin



El administrador también tendrá el control de capacitar a los usuarios con respecto al manejo de la biblioteca virtual con ejemplos de la vida real utilizando leyes

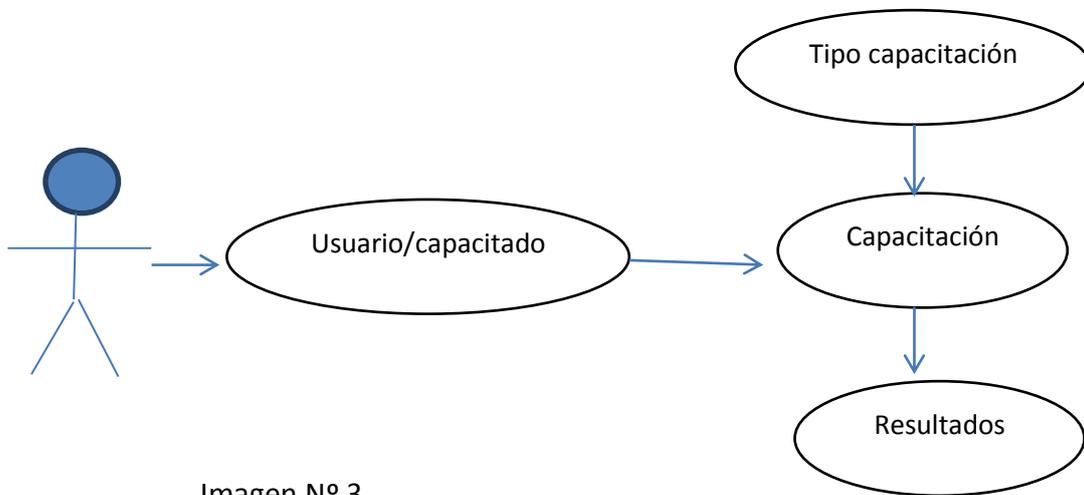


Imagen N° 3

Caso de Uso: Usuario Tipos

4.6 MODELO CONCEPTUAL

Aquí desarrollamos la base de datos ya que servirá para el desarrollo del sistema que estará contemplado tanto el control del registro así como también el aula virtual de cada uno de los casos

4.7 DESCRIPCIÓN DE METODOLOGÍA DE DESARROLLO

PROCESO RACIONAL UNIFICADO (RUP)

(RationalUnifiedProcess en Inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

También se conoce por este nombre al software desarrollado por Rational, hoy propiedad de IBM, el cual incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Está incluido en el RationalMethodComposer (RMC), que permite la personalización de acuerdo con las necesidades.



FASES DE METODO RUP

- * Establece oportunidad y alcance
- * Identifica las entidades externas o actores con las que se trata
- * Identifica los casos de uso

RUP comprende 2 aspectos importantes por los cuales se establecen las disciplinas:

Proceso: Las etapas de esta sección son:

- * Modelado de negocio
- * Requisitos
- * Análisis y Diseño
- * Implementación
- * Pruebas
- * Despliegue

Soporte: En esta parte nos encontramos con las siguientes etapas:

- * Gestión del cambio y configuraciones
- * Gestión del proyecto
- * Entorno

La estructura dinámica de RUP es la que permite que este sea un proceso de desarrollo fundamentalmente iterativo, y en esta parte se ven inmersas las 4 fases descritas anteriormente:

- * Inicio (También llamado Incepción o Concepción)
- * Elaboración
- * Desarrollo (También llamado implementación. Construcción)
- * Cierre (También llamado Translación)

4.8 MODELO FISICO

Aquí desarrollamos la base de datos ya que servirá para el desarrollo del sistema que estará contemplado tanto el control del registro así como también del aula virtual de cada uno de los casos que tenemos, que este modelo físico representa en si la realidad de la implementación



4.8.1 Evaluación y Selección de Alternativas

Con la finalidad de poder determinar y seleccionar la alternativa que más conviene a los intereses empresariales se ha definido un sistema de calificación con coeficientes matemáticos que están determinados de la siguiente manera.

Técnico 70%, Económico 20%, Garantía Técnica 5%, Soporte técnico 5%

Técnico

Las especificaciones técnicas del sistema determinan la parte más importante del todo el proceso de selección de alternativas escogiendo de esta forma la más idónea, por consiguiente para cada una de estas especificaciones técnicas se ha diseñado una matriz que contiene pesos cuantitativos de acuerdo a la importancia de cada uno de los elementos que intervienen en la mencionada alternativa estos pesos sumaran una totalidad de 100 puntos lo que corresponderá el 70% de la parte técnica y mediante de una regla de tres determinamos lo que corresponde a cada alternativa

Económico

A fin de determinar el puntaje respectivo que le corresponde al 20% consiste a la parte económica se define de la siguiente manera, 25 puntos cuyo costo sea el más económico y progresivamente se irá asignando un costo del más caro al más barato. El valor más caro estará determinado por la medida aritmética que le corresponde que sería 10 puntos y de esta manera podemos determinar el valor restante que sería de 15 puntos. De esta manera podemos sacar los valores económicos y puntajes para las alternativas.

Garantía

Este ítem tiene un puntaje de 5 puntos por lo que se determina este puntaje a la alternativa a la que presenta mayor tiempo igualmente las otras alternativas se irán determinando por el tiempo máximo y por el menor tiempo



Soporte Técnico

Está determinado por la calidad de técnicos que tiene la empresa y además por lista de clientes entre más clientes tenga quiere decir que la empresa es estable y tiene prestigio y a esta se le designara 10 puntos

REQUERIMIENTOS TECNICOS		
DESCRIPCION	PESOS(PUNTAJE)	PESO (%)
Plataforma WEB	10	7
lenguaje de programación PHP	10	7
Toad Data modeler	2	1,4
Base de datos MY SQL	2	1,4
Interfaz grafica para la pagina	5	3,5
Módulo de seguridad	8	5,6
Módulo de usuarios	8	5,6
Módulo Mantenimiento	10	7
Modulo transaccional	5	3,5
Página informativa	5	3,5
Página tema	5	3,5
Página sub tema	10	7
Página programas	10	7
Nombre dominio	5	3,5
Servicio Hospedaje	5	3,5
Total	100	70
Tiempo	10	5
Costo	25	15
Garantía	10	5
Soporte técnico	5	5
Total 2	50	30
Total final(total+total2)	150	100



REQUERIMIENTOS TECNICOS		
DESCRIPCION	PESOS(PUNTAJE)	PESO(%)
Plataforma WEB	8	7
lenguaje de programación PHP	8	7
Toad Data modeler	2	1,4
Base de datos MY SQL	2	1,4
Interfaz grafica para la pagina	5	3,5
Módulo de seguridad	8	5,6
Módulo de usuarios	8	5,6
Módulo Mantenimiento	6	7
Módulo transaccional	6	3,5
Página informativa	5	3,5
Página tema	10	3,5
Página sub tema	10	7
Página programas	10	7
Nombre dominio	6	3,5
Servicio Hospedaje	6	3,5
Total	100	70
Tiempo	10	5
Costo	15	9
Garantía	10	5
Soporte técnico	4	4
Total 2	39	23
Total final(total+total2)	139	100



REQUERIMIENTOS TÉCNICOS		
DESCRIPCIÓN	PESOS(PUNTAJE)	PESO(%)
Plataforma WEB	10	7
Lenguaje de programación PHP	10	7
Toad Data modeler	2	1,4
Base de datos MY SQL	2	1,4
Interfaz gráfica para la página	5	3,5
Módulo de seguridad	8	5,6
Módulo de usuarios	8	5,6
Módulo Mantenimiento	10	7
Módulo transaccional	5	3,5
Página informativa	5	3,5
Página tema	5	3,5
Página sub tema	10	7
Página programas	10	7
Nombre dominio	5	3,5
Servicio Hospedaje	5	3,5
Total	100	70
Tiempo	10	5
Costo	20	12
Garantía	8	4
Soporte técnico	4	4
Total 2	42	25
Total final(total+total2)	142	95

4.9 Factibilidad Técnica

De acuerdo al análisis expuesto anteriormente de cada de las alternativas propuestas para la solución del presente proyecto se puede concluir que su realización en la parte técnica es totalmente realizable la misma consideración la realizamos en el aspecto económico considerando en forma total que la factibilidad de esta es totalmente realizable desde el punto vista tecnológico.

Una vez hecho el respectivo análisis de los puntos expuestos anteriores se pudo tomar una decisión importante la cual es que la alternativa primera es la más accesible.

4.10 Descripción De Procesos

4.10.1. Login Usuarios

El proceso verifica mediante un login y password si el usuario que está tratando de ingresar al sistema se encuentra registrado en la base de datos. El proceso no se ejecutara más de 3 veces seguidas, con un mismo usuario por seguridad del Sistema.

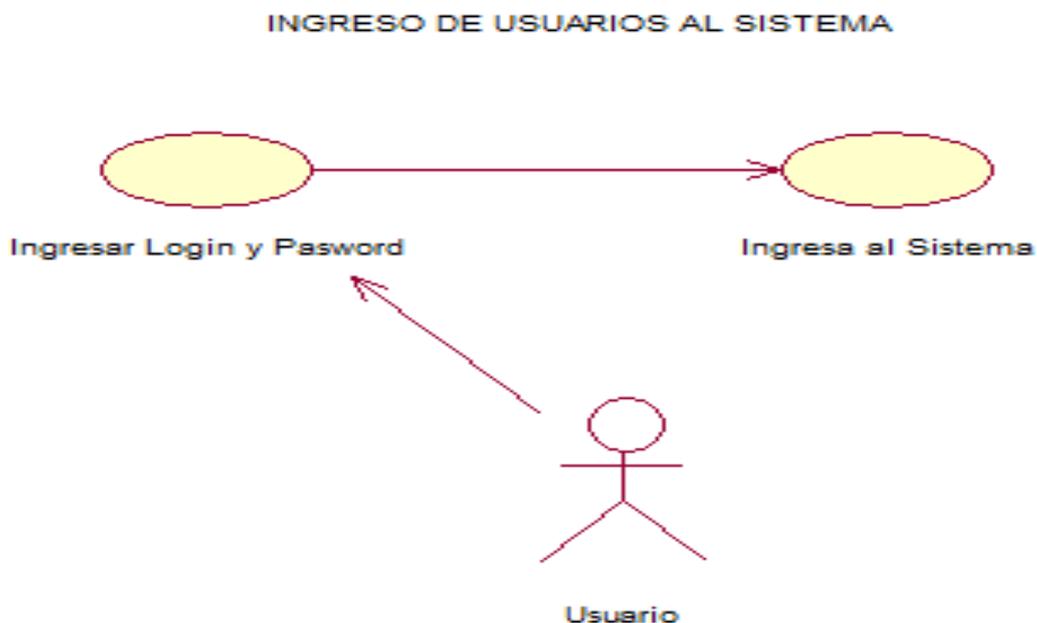


Imagen N° 4

Caso de Uso: Login de usuario

4.10.2 Mantenimiento del Sistema

El proceso de mantenimiento es exclusivamente para los usuarios con permisos, es decir para administrador o usuarios con privilegios en el Sistema.

4.10.3 Ingresar Información al Sistema

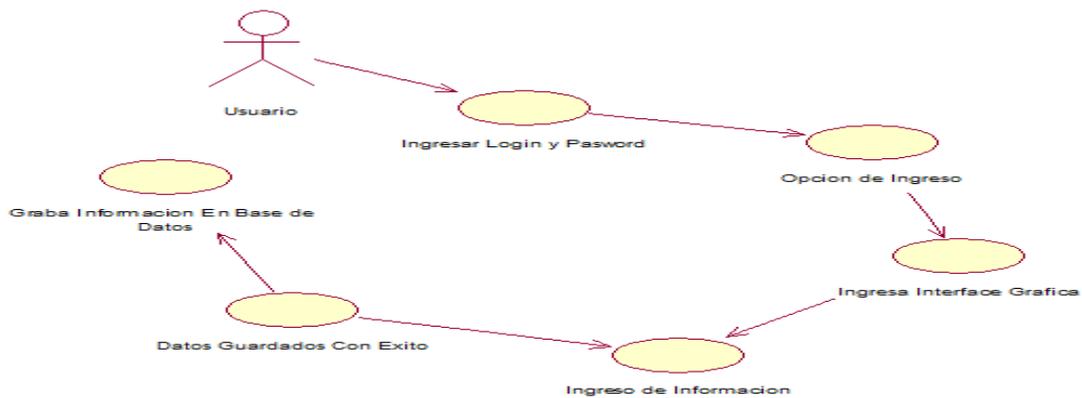


Imagen N° 5

Caso de Uso: Ingresar Información al Sistema

4.10.4 Modificar Información en Sistema

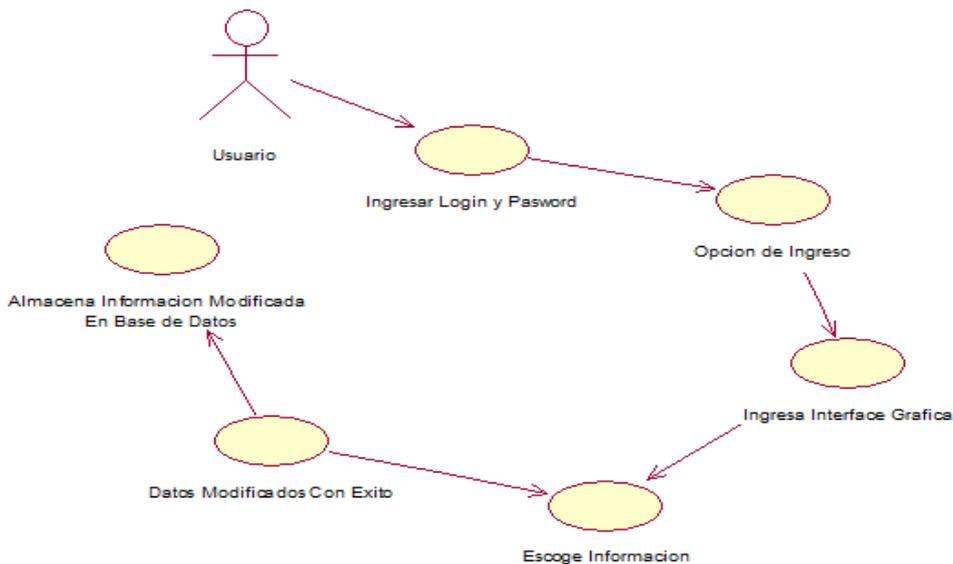


Imagen N° 6

Caso de Uso: Modificar Información al Sistema

4.10.5 Eliminar Información en Sistema

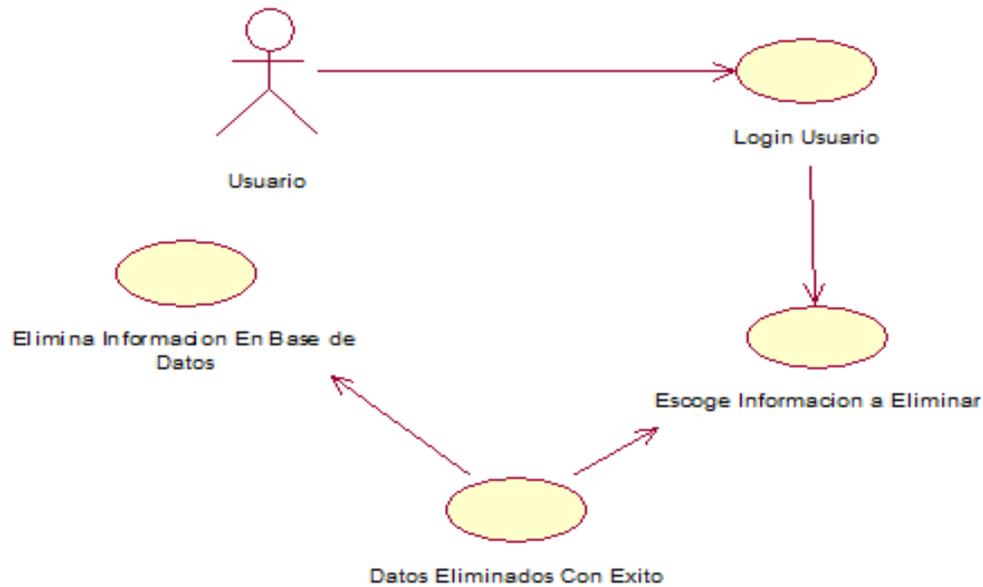


Imagen N° 9

Caso de Uso: Eliminar Información al Sistema

4.11 Diccionario de Datos

Un diccionario de datos contiene las características lógicas de los datos que se van a utilizar en el sistema que se está desarrollando, incluyendo nombre, descripción, alias, contenido y organización. Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.



En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario de datos guarda los detalles y descripción de todos estos elementos.

CONSULTA DE TABLAS

ESQUEMA TABLA	NOMBRE TABLAS	TIPO TABLAS
Dbó	docentes	base table
Dbó	escuela	base table
Dbó	jornada	base table
Dbó	jos_assets	base table
Dbó	jos_banners	base table
Dbó	jos_banner_clients	base table
Dbó	jos_banner_tracks	base table
Dbó	jos_categories	base table
Dbó	jos_contact_details	base table
Dbó	jos_content	base table
Dbó	jos_content_frontpage	base table
Dbó	jos_content_rating	base table
Dbó	jos_core_log_searches	base table
Dbó	jos_extensions	base table
Dbó	jos_languages	base table
Dbó	jos_menu	base table
Dbó	jos_menu_types	base table
Dbó	jos_messages	base table
Dbó	jos_messages_cfg	base table
Dbó	jos_modules	base table
Dbó	jos_modules_menu	base table
Dbó	jos_newsfeeds	base table



Dbó	jos_redirect_links	base table
Dbó	jos_schemas	base table
Dbó	jos_session	base table
Dbó	jos_template_styles	base table
Dbó	jos_updates	base table
Dbó	jos_update_categories	base table
Dbó	jos_update_sites	base table
Dbó	jos_update_sites_extensions	base table
Dbó	jos_usergroups	base table
Dbó	jos_users	base table
Dbó	jos_user_profiles	base table
Dbó	jos_user_usergroup_map	base table
Dbó	jos_viewlevels	base table
Dbó	jos_weblinks	base table
Dbó	materias	base table
Dbó	pregunta	base table
Dbó	respuesta	base table
Dbó	tipo_pregunta	base table

4.12 ESTANDARES DE PROGRAMACIÓN

Nomenclatura

El nombre de los objetos de base de datos no deberá exceder los 35 caracteres y no deberá terminar en “_”. En el caso que el nombre esté compuesto por más de una palabra, las mismas tendrán que separarse por el signo “_” underscore.

Los nombres de objetos de base de datos deberán estar escritos en español, ser auto-descriptivos¹ (nombres completos). El Área de involucramiento técnico asignará el nombre del proyecto, nombre de esquema y el prefijo de la aplicación a



utilizarse en la nomenclatura de los objetos de base de datos, a fin de dar identidad al proceso de seguimiento y rastreo vehicular.

Documentación

Todos los objetos de base de datos y scripts deberán estar debidamente documentados en Power Designer, en las siguientes propiedades:

Tipos de Datos

Tipo de Dato	Cuando se debe utilizar
VARCHAR2	Para campos de texto de tamaño variable de hasta 5000 caracteres. Siempre se utilizará VACHAR2 en lugar de VARCHAR.
CHAR	Para campos de texto de tamaño fijo, por ejemplo para el uso de estados (SI/NO), (ACT/INA).
NUMBER	Para campos numéricos. Siempre se especificará la precisión, por ejemplo para un número de 5 cifras enteras y 2 decimales será: NUMBER (7,2).
DATE	Para campos de fecha y de fecha y hora.
CLOB	Para campos de texto de más de 5000 caracteres.
BLOB	Para almacenamiento de archivos binarios, por ejemplo imágenes, archivos pdf, Word, etc.
XMLTYPE	Para campos en formato XML sobre los cuales se ejecutarán consultas con XPATH desde la Base de Datos. Si no se requerirá utilizar el motor de XML de la Base de Datos, es preferible utilizar un campo CLOB



Clave Primaria (Primary Key)

Nombre Campo:	“CODIGO_” [Nombre de la tabla en singular, sin prefijo de la aplicación, hasta donde alcance]
Nombre PK:	Utilizar el nombre asignado por Designer ([alias de la tabla] “_PK”)
Consideraciones:	<p>La clave primaria debe ser numérica y de un solo campo, salvo mejor criterio de Diseño.</p> <p>Como caso de excepción se permitirá la creación de claves primarias compuestas, en el caso en que la mejor representación a nivel de entidades en un diagrama de clases sea una relación de muchos a muchos y:</p> <ul style="list-style-type: none">➤ No se requiere registrar información adicional en la tabla de rompimiento➤ No se requiere relacionar la tabla de rompimiento con ninguna otra tabla.
Documentación:	Se requiere únicamente la documentación del campo en la tabla (no se requiere documentar el primary key). En el caso de utilizar una secuencia, se colocará la referencia a la misma utilizando la sección SECUENCE de Designer.



Campos

Nombre:	<p>El nombre de un campo deberá ser auto-descriptivo y estar en singular.</p> <p>Únicamente la clave primaria y los campos que provienen de un foreign key pueden comenzar con la palabra CODIGO.</p> <p>El nombre no debe incluir valores permitidos, por ejemplo: ESTADO_ACTIVO_INACTIVO, lo correcto sería: ESTADO.</p>
Tipo de Dato:	Revisar las consideraciones del punto “0Tipos de Datos”
Consideraciones:	<p>Todo campo con un conjunto de valores limitados que:</p> <p>a) Varían en el tiempo, harán referencia a la tabla o replica de la tabla correspondiente. Por ejemplo ID_VEHICULO debe tener un foreign key hacia EVENTO, etc.</p> <p>Diseño analizará los casos de excepción en los que no es posible crear foreign keys, por ejemplo:</p> <p>Tablas no replicadas.</p> <p>Tablas replicadas con tipo COMPLETE</p>



	<p>En tablas de log.</p> <p>b) Son fijos, tendrán obligatoriamente allowable values registrados en Designer de la siguiente manera:</p> <p>Value: tres letras representativas de cada palabra, por ejemplo:</p> <p>Para ACTIVO, INACTIVO, serían ACT e INA.</p> <p>Para ACEPTADO_MANUALMENTE y ACEPTADO_AUTOMATICAMENTE serían ACE_MAN y ACE_AUT.</p> <p>Para valores con menos de tres letras se colocará el nombre completo, por ejemplo: SI, NO, 0+.</p> <p>Meaning: nombre completo del valor permitido, por ejemplo ACTIVO, INACTIVO, etc.</p> <p>Caso de Excepción para MDA: en el caso de aplicaciones que cuenten con un modelo UML los allowable values deberán ser diseñados en UML a través de Enumeraciones y se cargarán en Designer como check constraints en lugar de allowable values. La documentación del meaning deberá ser incluida en el comment del check constraint, ejemplo: "ACT = Activo, INA = Inactivo"</p>
Documentación:	<p>Comentario: se debe incluir la utilidad del campo y los casos de uso, por ejemplo:</p> <p>Código de la Agencia principal del SRI, en la cual se reciben los trámites que llegan de otras oficinas</p> <p>Descripción: este campo es opcional y debe incluir aclaraciones y ejemplos respecto al contenido del</p>

	<p>campo.</p> <p>En el caso de utilizar una secuencia, se colocará la referencia a la misma utilizando la sección SECUENCE de Designer.</p>
--	---

4.13 PANTALLAS Y REPORTE (PROTOTIPO).

LOGIN

Esta es la pantalla inicial en la cual se puede observar la portada del sistema, para entrar al login hay que acceder nuestra identificación y nuestra clave.

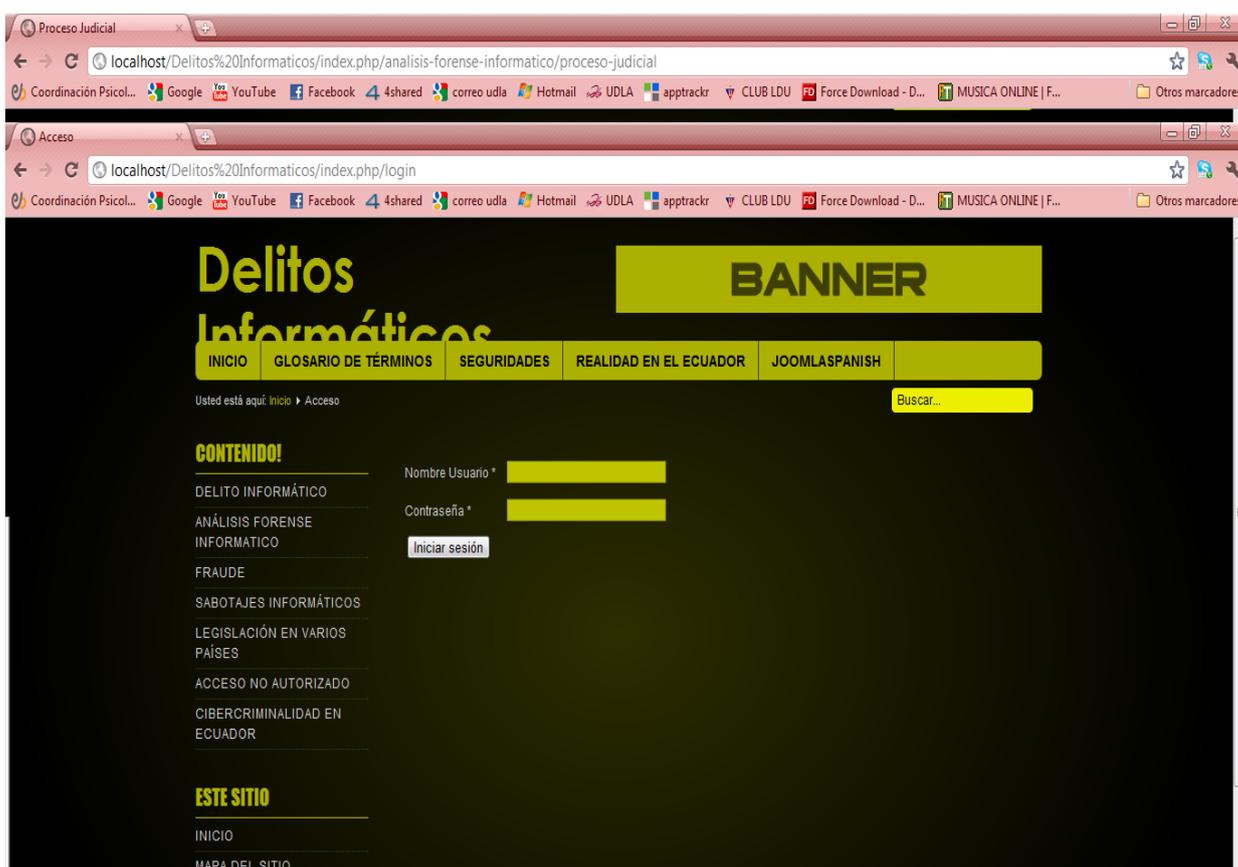


Imagen N°10



En esta pantalla se observa la portada inicial de la biblioteca virtual

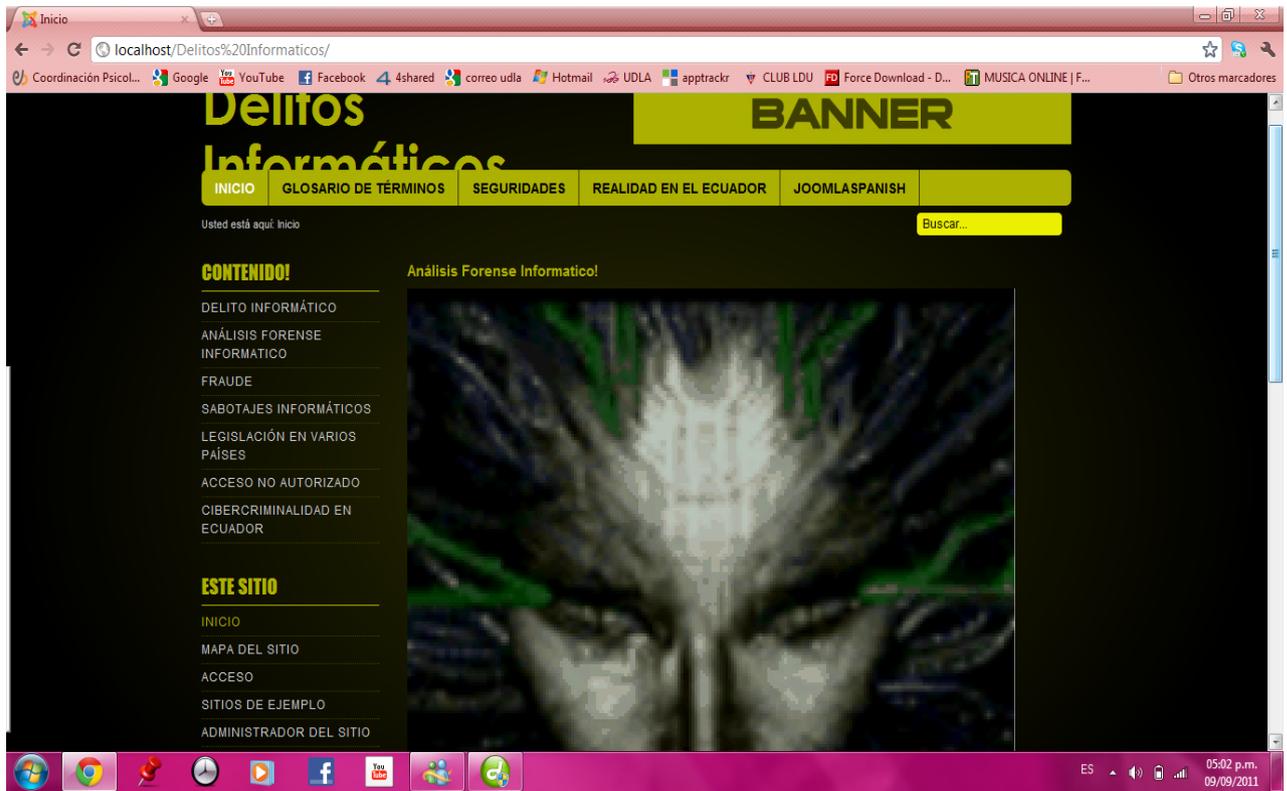
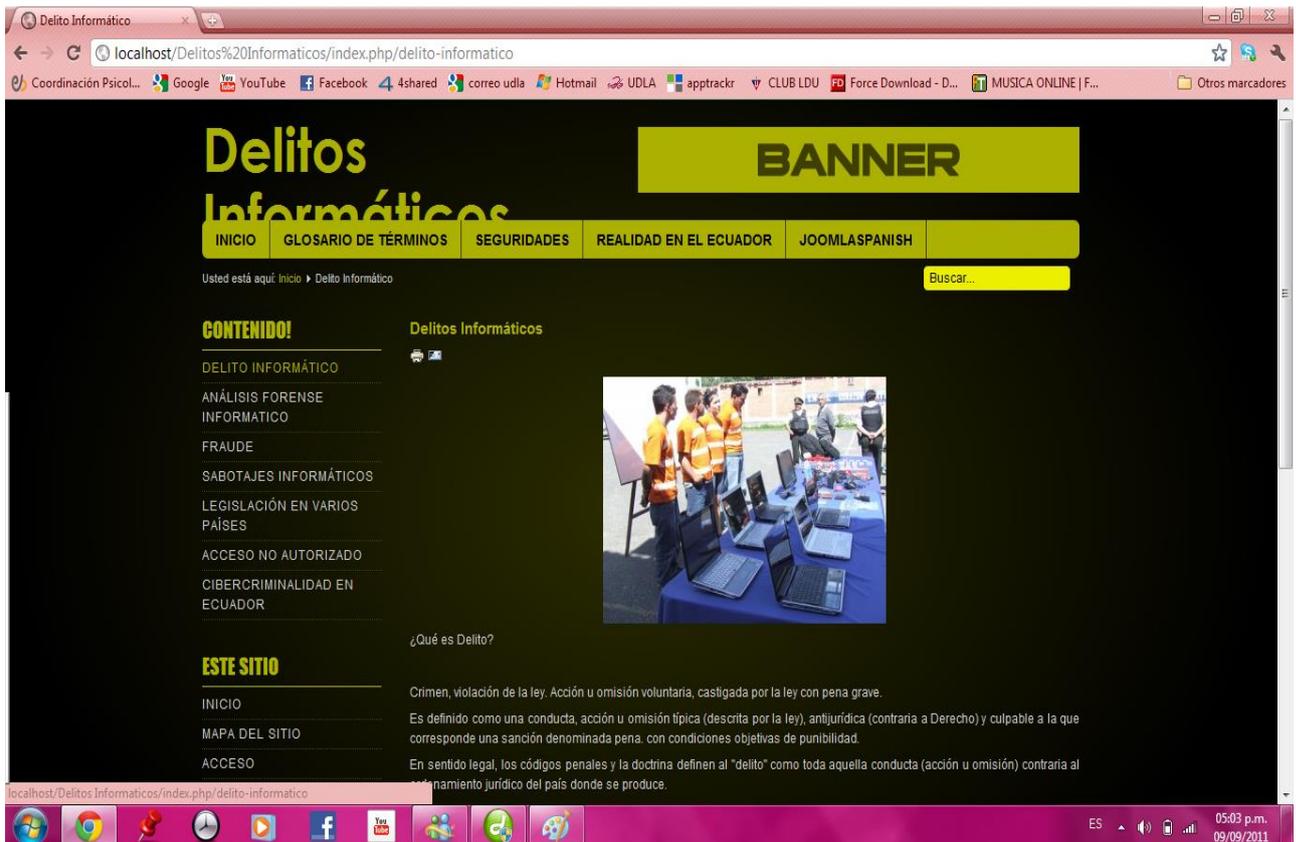


Imagen N°11

Como se puede observar en la imagen aquí podremos saber el concepto de que es un delito informático



Aquí se podrá observar que pasos se debería seguir para realizar un proceso judicial hacia las personas que cometieron un delito informático

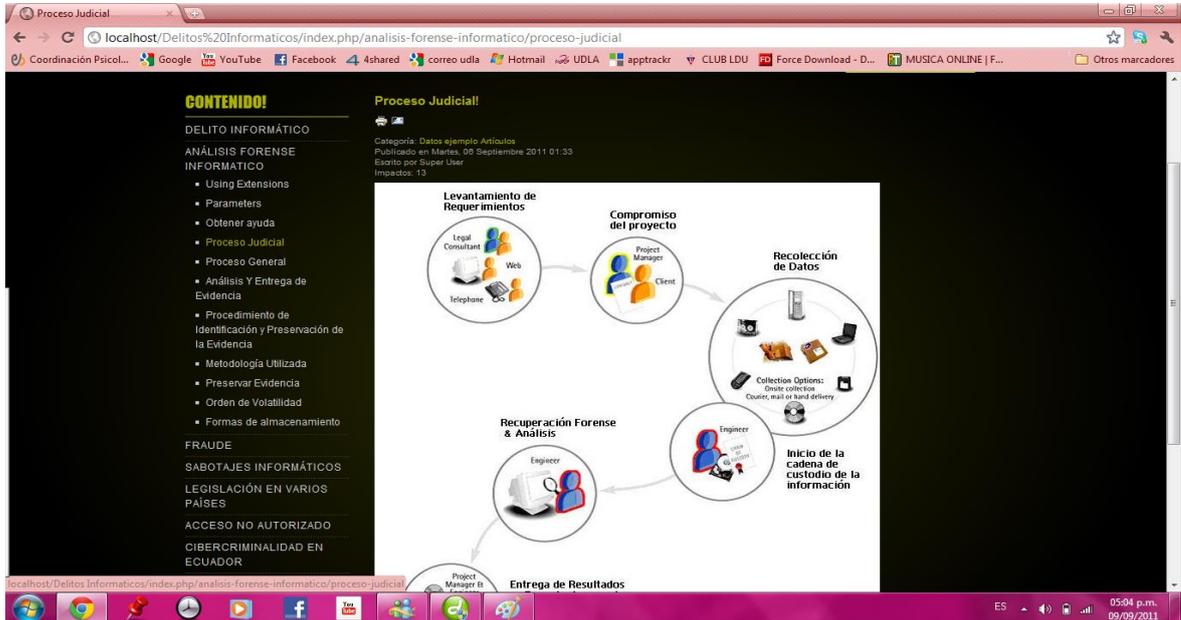
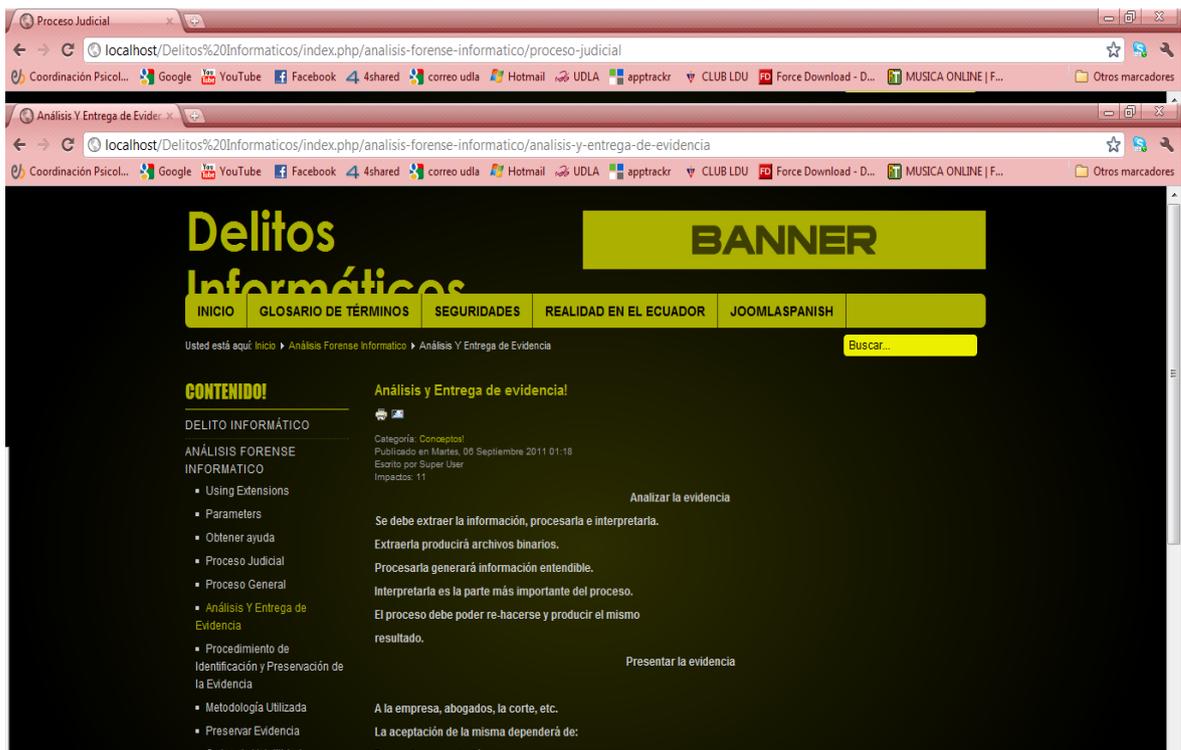


Imagen N°13

En esta página se observara el análisis de un caso informático ya una vez que esta haya acabado con el juicio también se podrá ver las posibles evidencias las cuales se las explicara como utilizaron los delincuentes informáticos para realizar los actos vandálicos



En esta parte se explicara mediante un diagrama como un delincuente informático está realizando algún fraude en cualquier parte del mundo que este se encuentre.

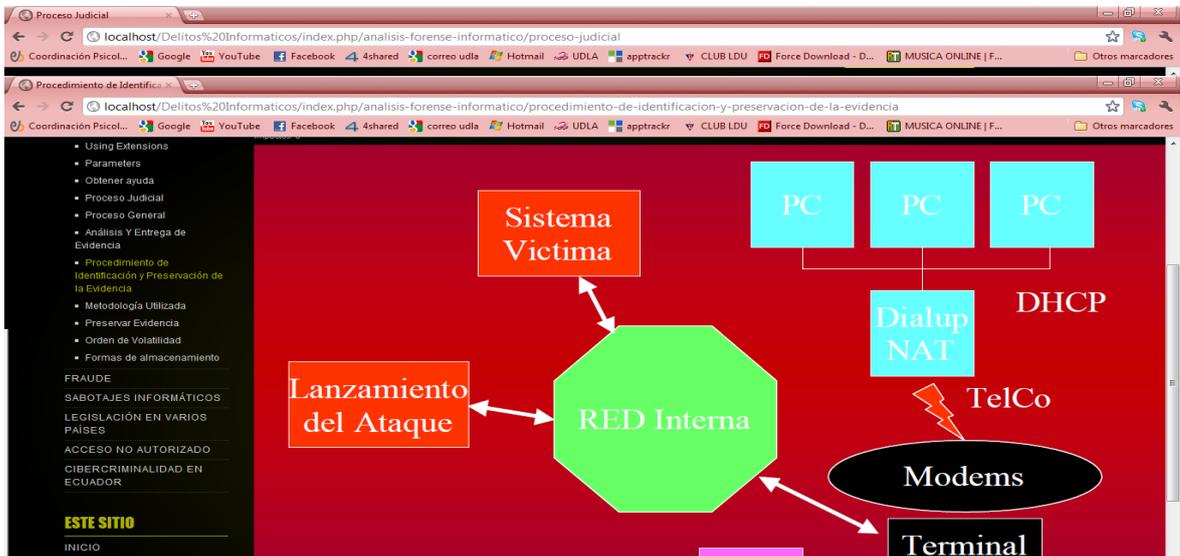


Imagen N°15

En esta página se podrá observar información acerca de cómo recuperar información perdida del disco duro y como poder recuperarla y esa ha sido manipulada



Se podrá observar en esta página como poder recuperar información de dispositivos de almacenamiento externo en caso de que estos estén dañados.

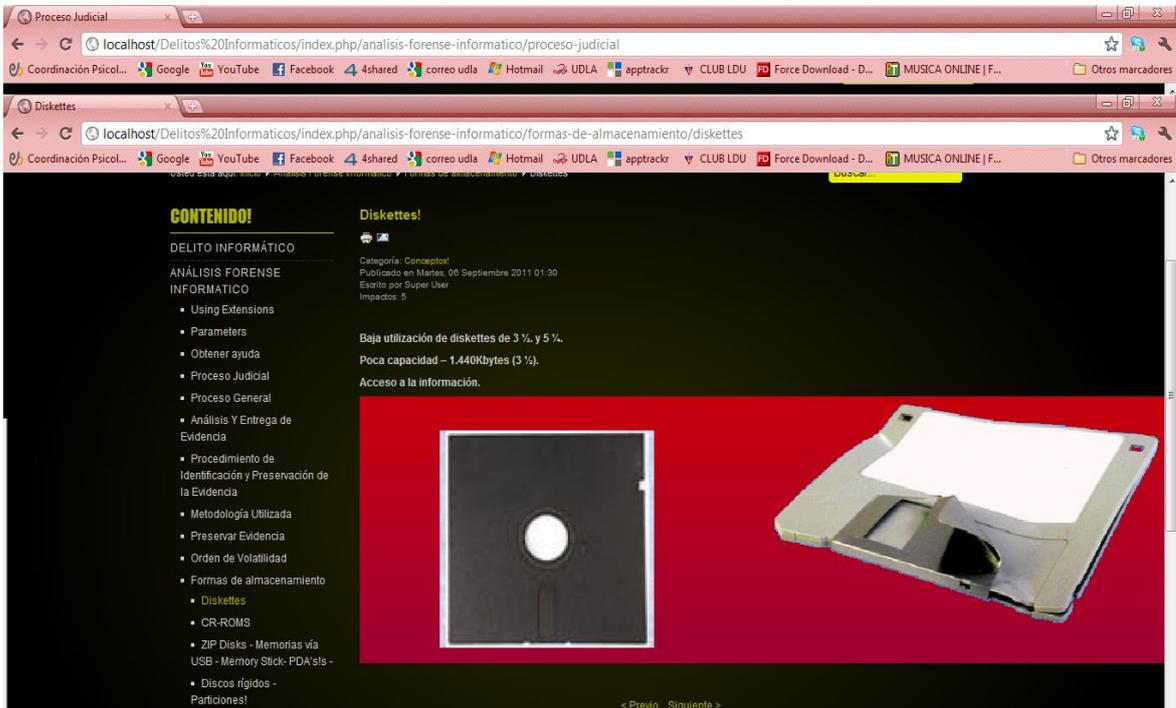
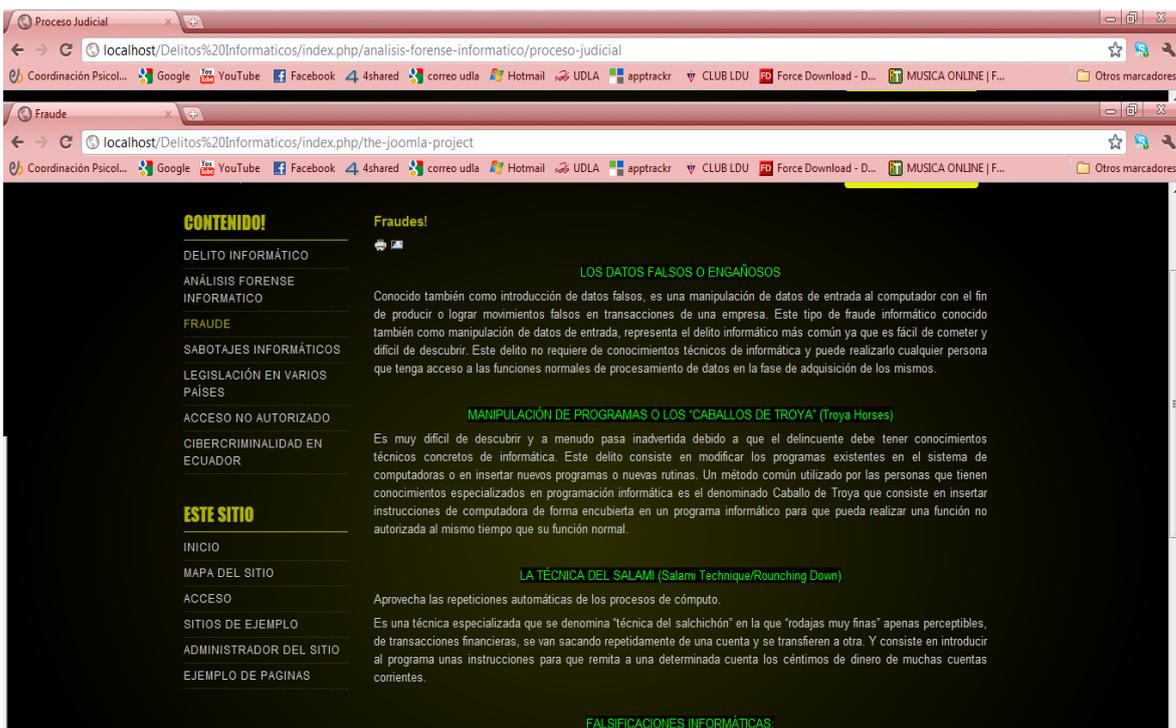


Imagen N°17

En la siguiente página se podrá informar acerca de qué tipo de fraudes informáticos existen en el Ecuador y como contrarrestarlos



También se podrá informar acerca de que tipos de virus pueden hacer daño a su sistema operativo y se dará las posibles soluciones al caso.



Imagen N°19

En esta pantalla podremos observar como un hacker informático puede ingresar a nuestro sistema violando las seguridades de nuestros computadores o cuentas de internet





PRUEBAS Y DEPURACION

4.14.1 Pruebas de Unidad

- Se concentra en la verificación de la unidad más pequeña del diseño de la página: el componente o módulo de la página.
- La verificación en el diseño se realizó determinando que exista las relaciones entre cada una de las tablas, poniendo especial énfasis en las tablas que contienen la información que almacena los datos de usuario, así como también la información de control de las entradas, y de registro considerando la importancia que tiene los reportes.
- Las pruebas de unidad se centralizan en la lógica del negocio, los procesos internos que se relacionan las acciones de Control que se consideren con la finalidad que la relación entre las tablas y almacenamientos de información pueda mantener una estructura adecuada y lógica.
- Este tipo de prueba se la realizó y el sistema mantiene su lógica, se consideró además que se mantenga la integridad de la información y sus datos, poniendo como referencia principal las relaciones, claves primarias y foráneas del modelo lógico del negocio.

4.14.2 Pruebas de investigación

Esta prueba la realice individualmente verificando que todo funciona bien, por lo tanto las relaciones existentes en el modelo establecido se verificaron la consistencia del modelo indicado las claves primarias y foráneas establecidas, a fin de poder verificar la indexación de las búsquedas o consultas realizadas tanto los alumnos, lectores, directores.

La prueba de integración es una técnica sistemática para construir la arquitectura del software, mientras, al mismo tiempo, se aplican las pruebas para descubrir errores asociados con la interfaz. La integración de los datos en la tabla, determina la robustez del modelo colocando especial énfasis en las asociaciones de las tablas que tienen relación con la información que maneja los datos.



4.14.3 Pruebas de validación

Las pruebas de validación empiezan tras la culminación de la prueba de integración, cuando se han ejercitado los componentes individuales. Se ha terminado de ensamblar el software como paquete y se han descubierto y corregido los errores de interfaz.

Este tipo de pruebas se la realizo en el modulo de seguridad establecido para poder controlar el acceso de los usuarios al sistema. También se estableció para poder verificar la integración que existe de los datos transaccionales con los datos especiales.

La prueba se concentra en las acciones visibles para el usuario y en la salida del sistema que este puede reconocer.

Este tipo de pruebas se realizo con el modulo de seguridad poniendo énfasis en los tipos de usuarios que tiene que manejar el sistema

Sin embargo, los pasos dados y las relaciones entre cada uno de las tablas se verifico la consistencia de los daos y la integridad de los mismos.

4.14.4 Pruebas del Sistema

- Al final del desarrollo de la página web se incorpora a otros elementos del sistema (hardware, personas, información) y se realiza una serie de pruebas de integración del sistema y de validación.

La carga de datos en el sistema constituye un factor determinante especialmente al realizar la migración de los datos de nuestra base de datos, por lo tanto la integración con la consola de la misma es un factor determinante a fin de poder establecer un sistema robusto, sólido y confiable, y factible en el manejo de la información, considerando que los datos del usuario y software servirá para la ejecución del reporte.

- Sin embargo, los pasos dados durante el diseño y la prueba de la plataforma virtual mejorarán en gran medida la probabilidad de tener éxito en la integración de la página web.



- La integración de los datos y las relaciones entre cada uno de las tablas se verifico la consistencia de los datos y la integridad de los mismos.

4.14 Prueba de Interfaces Gráficas de Usuario (GUI , Graphical User Interface)

- Uso de una lista de chequeo preestablecida:
Se realizo una prueba con la ejecución del manejo de los iconos de acceso al sistema y la integración con la base de datos y los softwares, las noticias, los reportes respectivos, y la integración con su respectiva plataforma.
- Entrada de datos:
El ingreso de los datos se realizara por medio de los datos del cliente el mismo que es único para cada uno de los registros, además para cada uno de los usuarios se determinará una clave de acceso individual.

4.14.1 Pruebas del Sistema

- Prueba de resistencia y consistencia
Se verifico que exista la relación y consistencia entre el registro de usuarios y el despliegue de sus privilegios lo que significa que la integración con la base de datos cumplan con su requerimiento.

4.16.2 Instalación del Sistema

Para la instalación del sistema procedemos a realizar los siguientes pasos.

Pasos Previos a la Instalación

Es importante tener en consideración una serie de pasos previos para la Instalación de paquetes como Dreamwe dreamweaver y WampServer.

- Instalar las últimas actualizaciones del Navegador.
- El instalador de Macromedia instalara tres programas dreamweaver, flash, firework.



Imagen N°21

Elegimos Dreamweaver y presionamos ENTER luego visualizara una barra de incremento de la apertura del instalador



Imagen N°22

Procesando para la instalación

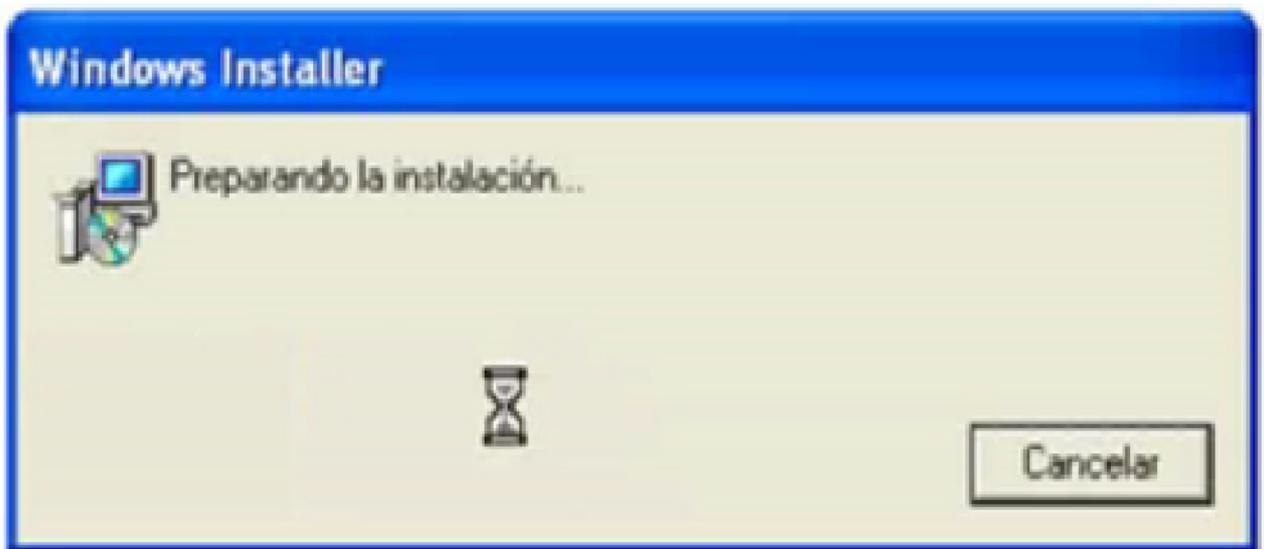
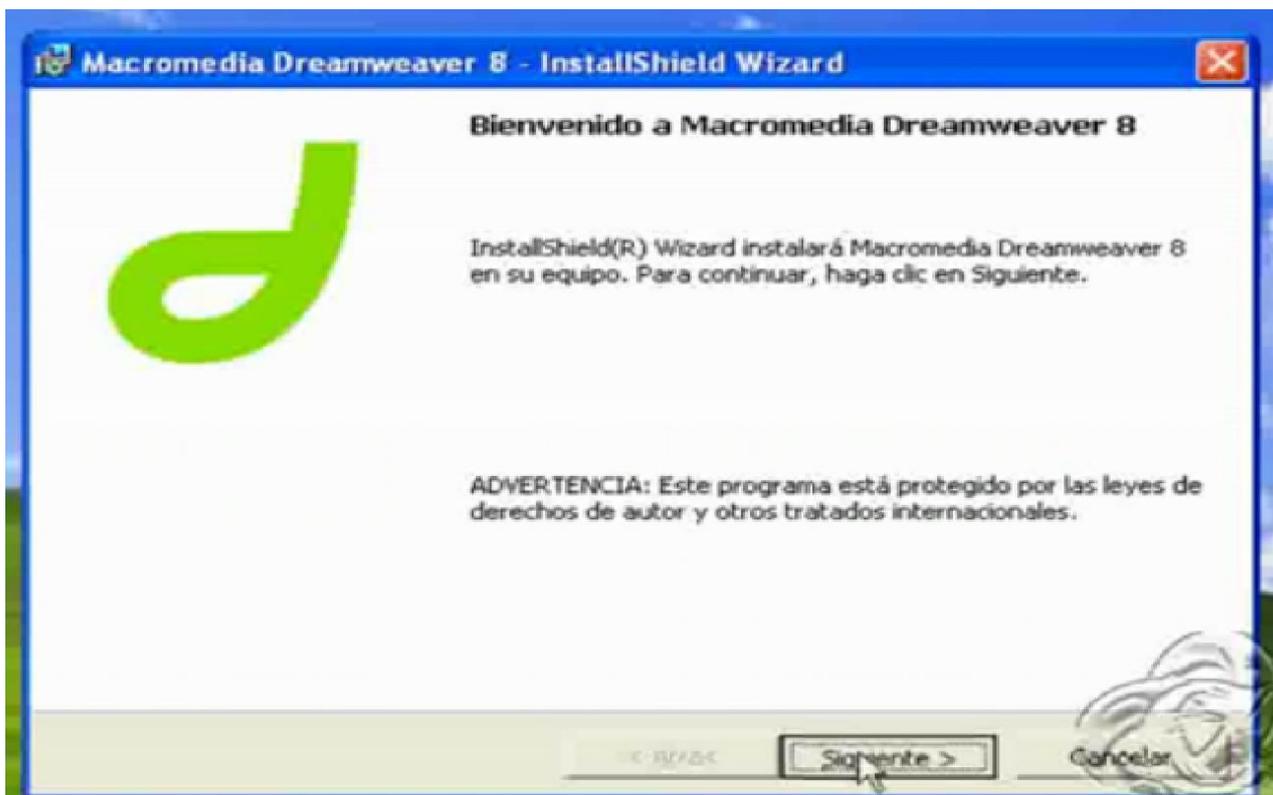


Imagen N°23

Aceptando la Instalación



Aceptando términos del contrato



Establecer los formatos que va a manejar el editor



Imagen N°26

Instalando el programa



Imagen N°27

Finalización de la instalación y apertura del mismo

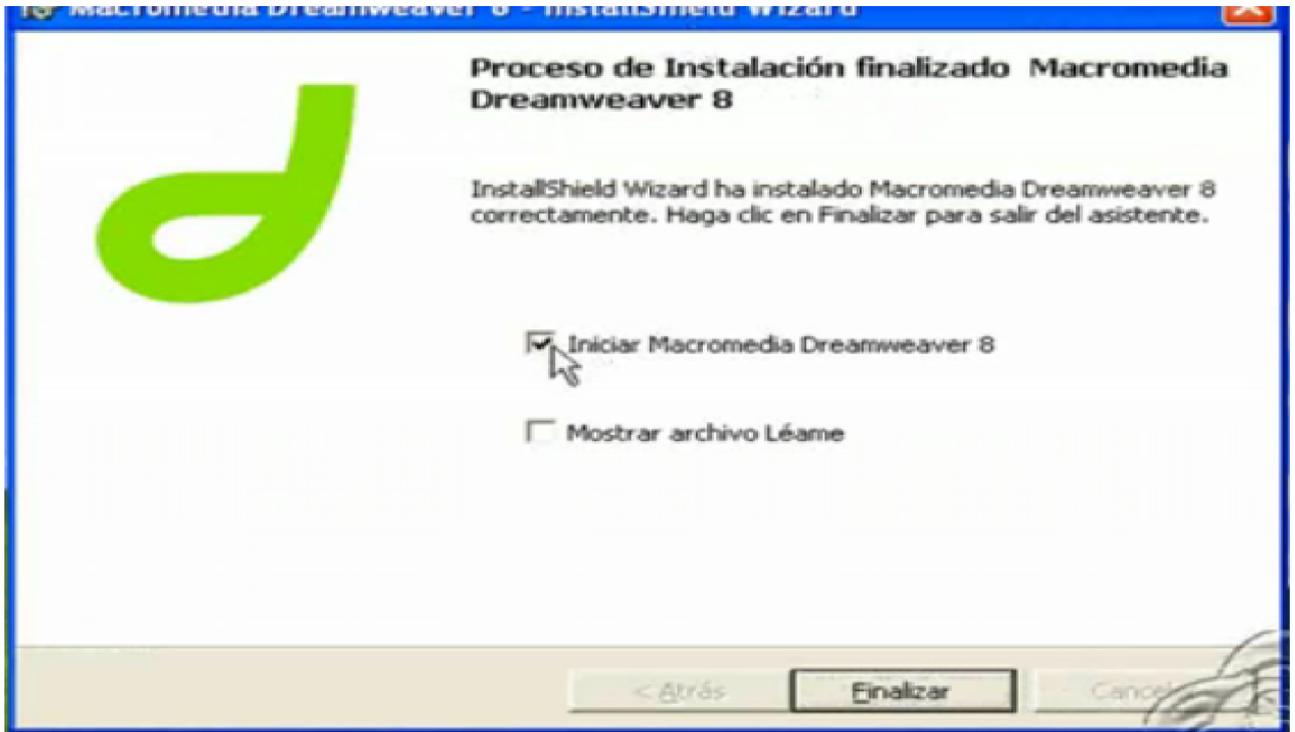
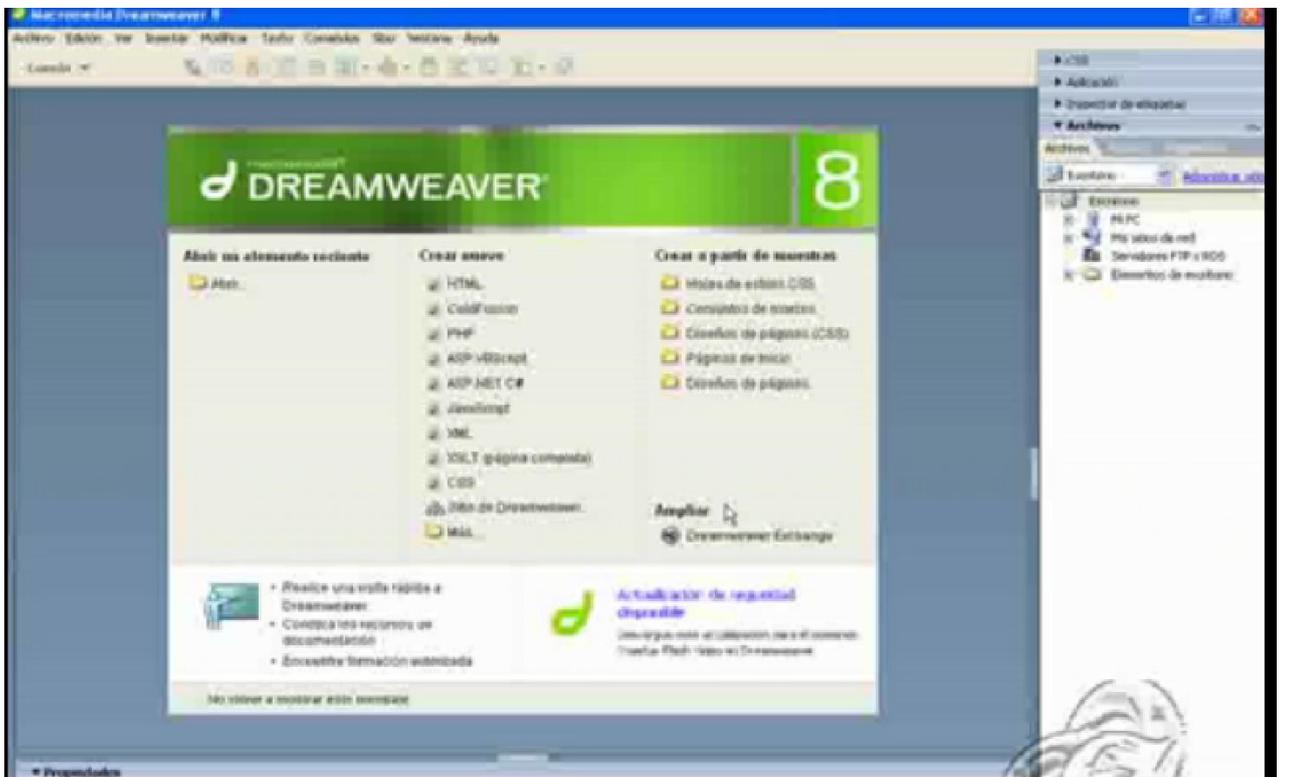


Imagen N°28

Visualización del programa



INSTALACIÓN DE WAMP SERVER



Gráfico N° 30 WampServer

En esta ocasión instalaremos uno de los programas que nos permitirán crear un servidor en nuestra casa sin necesidad de tener una conexión de internet en nuestra pc. Éste servidor casero nos servirá para poder probar y evaluar nuestros proyectos a crear.

Instalación Wamp Server

Primeramente lo ejecutamos en nuestra computadora.

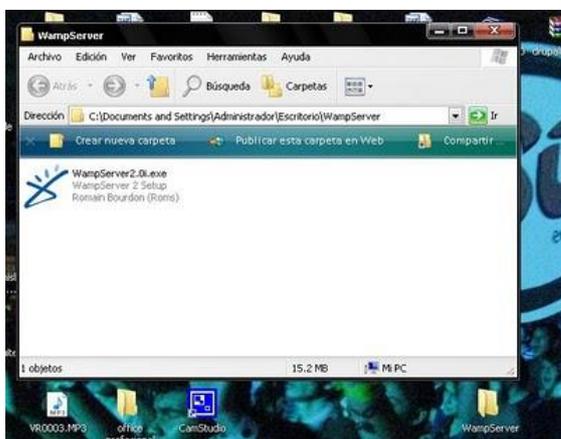


Gráfico N° 31

Ejecutamos Wamp Sever



Aquí observamos el mensaje de bienvenida y le damos clic en next para continuar la instalación.



Gráfico N° 32

Mensaje de Bienvenida de Wamp Server

Aceptamos el contrato de licencia y le damos clic en next.



Gráfico N° 33

Licencia de Wamp Server



Nos aparecerá la ubicación donde se realizara nuestra instalación c:\wamp que es por defecto y le damos clic next.



Gráfico N° 34

Lugar De la Instalación de Wamp Server

Dejamos seleccionados las 2 opciones ya que nos crearan un acceso directo en el escritorio y un acceso directo en la barra de inicio, le damos clic en next.

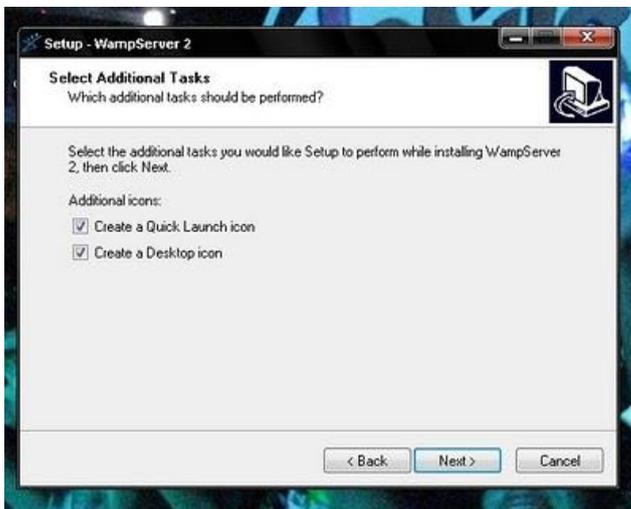


Gráfico N° 35

Selección del Icono de Wamp Server



Hacemos clic en install.

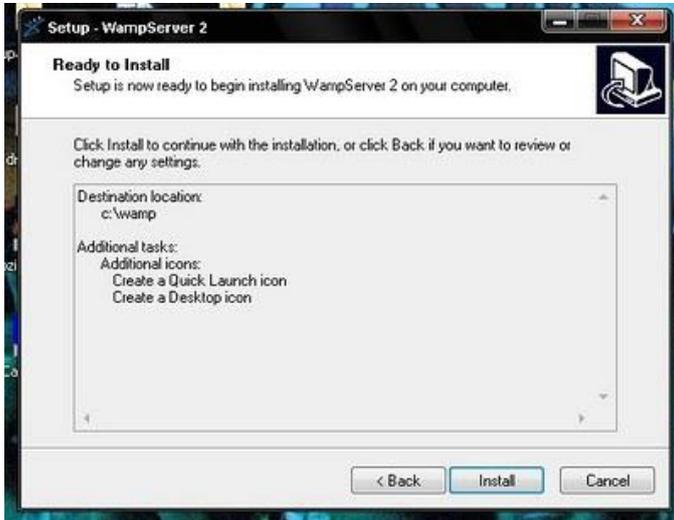


Gráfico N° 36

Instalación de Wamp Server

Observamos que se instala.

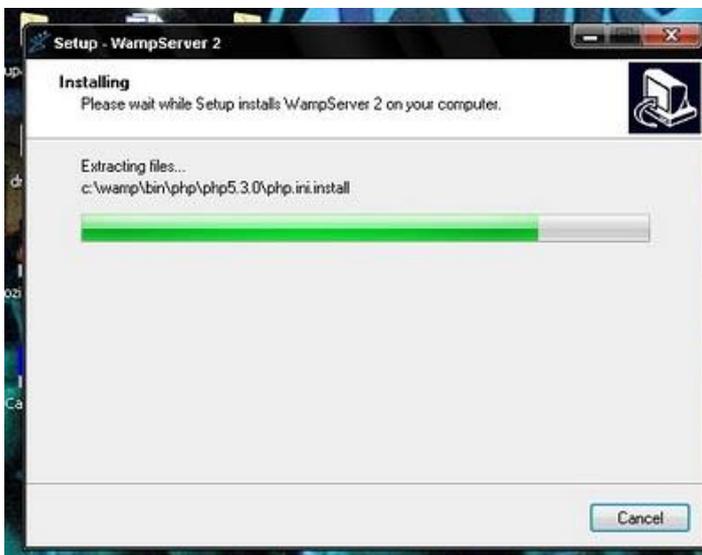


Gráfico N° 37

Proceso de Instalación de Wamp Server



Luego nos aparece una pantalla que nos indica que la instalación ha finalizado y una opción que nos permite que ejecutemos el WampServer por primera vez, lo dejamos seleccionado y hacemos clic en finish.



Gráfico N° 38

Fin de la Instalacion de Wamp Server

Aquí donde indica la flecha está el icono de inicio rápido del Wamp Server que será de gran utilidad.



Gráfico N° 39

Icono de Wamp Server



Al hacer clic en el icono vemos el menú del Wamp Server que nos será muy útil para hacer nuestro proyecto.



Gráfico N° 40

Funcionamiento de Wamp Server

4.17 Pruebas y Depuración Final en funcionamiento.

Como conectar a la base de datos (Mirar en anexos)

Registro de usuario aceptado

Direccionamiento de links

Visualización de todas las páginas

VERIFICACION DE LA BASE DE DATOS QUE NO PERMITA EL INGRESO DE BASURA.

4.18 Capacitación al Usuario Final.

Capacitación de la Plataforma Virtual

4.19.1. Datos Informáticos

Empresa: ZonaNet Telecomunicaciones

Número de Trabajadores de la Empresa: 3



Nombre Capacitador:	Paul Ribadeneira.
Método de Capacitación:	Charla con Videos por medio de mails.
Tiempo de Capacitación:	1 hora.

4.19.2. Desempeños Esperados

LO QUE EL USUARIO ALCANZARA HACER

Al término de la capacitación del manejo de la página web, el usuario (s), estará en la capacidad de manejar eficientemente los siguientes puntos:

- Entender el funcionamiento de la pagina y manejo de los software
- Ingresar al sistema con su respectiva clave de usuario
- Lograr Ingresar a foros, debates, artículos.
- Realizarlas descargas de la información.
- Efectuar el manejo de artículos.
- Realizar la apertura y cierre de cuenta.

4.19.3. Diseño Operativo

ACTIVIDAD	RECURSOS	RESULTADOS
Charla con videos: Revisión por parte del capacitador sobre el funcionamiento del Sistema.	videos, Diapositivas, pdfs s Conceptuales,	Participar con argumentos relevantes. Resolver inquietudes.
Laboratorio: Realizar practica de utilización de los software permitidos en la página web	Laboratorio Computadoras Software	Vincular al usuario con la pagina.
Prueba Diagnostica: Conocimientos adquiridos en los	Laboratorio Computadoras	Demostrar el conocimiento de



videos y la ejecución de los ejercicios aplicativos	Software	seguridad ante un delito
---	----------	--------------------------

4.19.4. Observaciones y Recomendaciones

- El usuario deberá observar todos los videos revisando los manuales de cada software antes entregado por parte del administrador de la página.
- Participación analítica y crítica en los foros.

4.20 Capacitación al Personal Técnico.

Capacitación del Sistema al Personal Técnico

4.20.1. Datos Informáticos

Empresa:	ZonaNet Telecomunicaciones
Número de Trabajadores de la Empresa:	1
Nombre Capacitador:	Paúl Ribadeneira.
Material Entregable:	Videos y foros.
Método de Capacitación:	Visual.
Tiempo de Capacitación:	1 horas.

4.20.2. Desempeños Esperados

LO QUE EL PERSONAL TÉCNICO ALCANZARÁ HACER

Al término de la capacitación del Sistema, el técnico (s), estará en la capacidad de manejar eficientemente los siguientes puntos:

- Entender el funcionamiento de la plataforma
- Ejecutar el sistema con su respectiva clave de administrador
- Lograr Ingresar al mantenimiento de la pagina, como usuarios, artículos, noticias.
- Realizar la respectiva auditoria del Sistema
- Monitorear la Base de Datos.



LO QUE EL PERSONAL TÉCNICO DEBE SABER HACER

Para poder iniciar la ejecución del Sistema, el personal técnico previamente debe conocer y saber usar los siguientes tópicos:

- Manejo de Base de Datos
- Programación en PHP.
- SQL Server 2008 R2.
- Manual de Procesos y Procedimientos
- Auditoría de Sistemas.

4.20.3. Diseño Operativo

ACTIVIDAD	RECURSOS	RESULTADOS
Charla con videos: Revisión del funcionamiento del Sistema, Base de Datos, Metodología de Desarrollo, Plataforma de Desarrollo.	Manual Técnico, Diapositivas: Mapas Conceptuales,	Participar con argumentos relevantes. Resolver inquietudes de otros compañeros.
Laboratorio: Demostración del Sistema	Laboratorio Computadoras Software	Vincular al personal técnico con el Sistema.

4.20.4. Observaciones y Recomendaciones

- El personal técnico deberá mantenerse revisando su correo ya que se enviara manuales para el funcionamiento del software.
- Participación analítica y crítica en la charla.



CAPÍTULO V

PRINCIPALES IMPACTOS

5.1 Científico.

La investigación desarrollada por el proyecto tiene una característica especial ya que aplicativos y tecnologías implementadas en nuestro medio no era muy difundido el tema del delito informático únicamente se lo conocía cuando un acto ilícito acerca de este tema se presentaba por lo tanto al tratarse de una implementación local se podrá beneficiar los usuarios que revisen nuestra biblioteca virtual

5.2 Educativo.

El desarrollo de un sistema informático involucra el conocimiento del diseño y la programación, por lo tanto al realizar un proyecto de fusión de tecnologías se realizara un aporte muy importante para el conocimiento de las personas que sufran un atentado informático.

5.3 Técnico.

Lo referente a este tema se considera las herramientas que se utilizaron para el modelamiento y desarrollo de la base de datos y en si del proyecto.



5.4 Tecnológico.

El ejecutar temas de innovación tecnológica aplicando los conocimientos recibidos en las aulas hace que cualquier proyecto sea considerado como un aporte institucional hacia la sociedad, mejor aun si lo que se resta realizando es la unión de varias tecnologías en un solo aplicativo, que es producto de la investigación tecnológica.

5.5 Empresarial.

Siendo la visión emprendedora la realización en un futuro de una empresa, se ha cristalizado el primer paso que consiste en tener un aplicativo que maneje las reglas del negocio, por lo tanto a medida que se perfecciona el mismo se continuara con una mentalidad empresarial.

5.6 Social.

La implementación de este sistema y la estructuración de una empresa en lo futuro para brindar este tipo de servicio, lograra crear conciencia en las personas para que estas no se dejen estafar, por consiguiente se estará brindando un aporte muy importante al ámbito social.

5.7 Económico.

La generación de este proyecto ayudara a los usuarios del internet especialmente a las personas que trabajan transacciones bancarias mediante el internet asiendo que estos usuarios no sean estafados y así su dinero no se pierda.



CONCLUSIONES

- Se logró diseñar una página amigable para los usuarios con información muy importante para protección y defensa sobre los delitos informáticos
- Se analizó cada uno de los procesos que realiza nuestra biblioteca virtual.
- Se aplicó conocimientos de programación y diseño
- Se consiguió darse cuenta de cómo o en base a qué se debe estructurar el software que se desarrolló.

RECOMENDACIONES

Es importante informarse sobre el uso adecuado del manejo y mantenimiento de la página, así tendrá mejor servicio y contenido importante.

1. Todo trabajo de investigación debe sustentarse con una metodología métodos y herramientas de la investigación científica por lo tanto es recomendable que las personas se preparen en este campo para realizar cualquier investigación, más aun considerando que esto es la base para poder realizar el proyecto de grado previo a la obtención del título de Tecnólogo Analista de Sistemas.
2. Seguir la misma secuencia de desarrollo de software a fin de poder llegar a obtener un sistema completo, aplicable y amigable, es el resultado de haber seguido una metodología de desarrollo y técnicas de programación que permita interactuar los datos alfanuméricos (información), con los datos y relacionarlos, a fin de poder estructurar un sistema real de Toma de decisiones.



3. Para cualquier tipo de integración de tecnologías avanzadas es indispensable conocer los diversos lenguajes de programación por lo cual es recomendable instruir a los alumnos en la nuevas herramientas de desarrollo de software que cada día se van actualizando y que nos ofrecen mayores facilidades para integrar sistemas de muy alto rendimiento.
4. Siempre será importante conocer que en el mundo del desarrollo de software existen diversos tipos de caminos o integraciones con otros sistemas y especialidades, por lo que se debería agregar como materia principal el manejo de diversas tecnologías, ya que en el mundo laboral el tecnólogo debe estar listo para programar o resolver cualquier tipo de problema.



CAPÍTULO VI

BIBLIOGRAFÍA

Título del libro: Delitos Informáticos – Ciberterrorismo Autor: Salellas Luciano
Idioma: castellano

Título Legislación del país y sus cambios año 1992 idioma español

Gordillo, Mariano Martín. (Coord.) (2001) Materiales para la educación CTS (Ciencia, Tecnología y Sociedad) del Proyecto Argo. Segundo capítulo. Grupo Norte. España

Levy, Steven Recursos: Archivo texto Formato: LibrosElectrónicos

Idioma: Inglés Colección: ProyectoGutenberg

Publicado Por: Proyecto Gutenberg Año Public: 1996 Materias: REDES DE
COMPUTADORES - MEDIDAS DE SEGURIDAD | DELITOS INFORMATICOS | HACKERS

Pascual Mayte (2006). "Comunicación e innovación en la era de Internet". Cap. 9 de En
qué mundo vivimos. Alianza Editorial. Colección Alianza Ensayo. Madrid.

Delitos informáticos, 10 años después primera edición, Marzo 2009 Editorial Ubijus

Maldonado, Tomas (1998). Crítica de la razón informática. Cap.3: "Cuerpo humano y
conocimiento digital". Paidós. Barcelona. Ficha de Cátedra.



WEBGRAFIA

<http://www.hoy.com.ec/noticias-ecuador/cinco-delitos-informaticos-cada-dia-433373.html>

<http://www.cep.org.ec/catalogo/tbl06delitolist.php>

<http://www.alfa-redi.org/rdi-articulo.shtml?x=1445>

<http://www.hoy.com.ec/noticias-ecuador/delitos-informaticos-dejan-perdidas-cuantiosas-a-los-usuarios-465398.html>

http://www.gms.com.ec/index.php?option=com_content&view=article&id=84&Itemid=153

<http://www.rosalesuriona.com/spip.php?article664>

<http://www.burodeanalis.com/2011/04/01/la-falta-de-seguridad-informatica-hace-pagar-a-la-banca/>

<http://www.grupo-netcom.com/2009/05/nueva-ley-de-delitos-informaticos-ley-26388/>

http://www.ciudadaniainformada.com/noticias-ciudadania-ecuador0/noticias-ciudadania-ecuador/browse/4/ir_a/ciudadania/article//como-evitar-ser-victima-de-un-delito-informatico.html

<http://www.metroecuador.com.ec/7230-hackear-un-delito-oculto.html>

http://asambleanacional.gov.ec/blogs/cristina_kronfle/2011/01/31/ecuador-en-vivo-26-de-enero-2011-fiscal-allana-domicilio-de-andres-zules-acosta-acusado-de-delitos-informaticos/



<http://ecuacel.mobi/noticias/leer/victimas-delito-informatico.xhtml>

http://www.youtube.com/watch?v=pmuRc_vXK7A&feature=youtube_gdata

Glosario de Términos Técnicos

Red En Estrella: En informática, red de área local en la cual cada dispositivo, denominado nodo, está conectado a un ordenador o computadora central con una configuración en forma de estrella. Normalmente, es una red que se compone de un dispositivo central y un conjunto de terminales conectados. En una red en estrella, los mensajes pasan directamente desde un nodo al Hub, el cual gestiona la redistribución de la información a los demás nodos.

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

Manual De Capacitación: Documento que tiene la información necesaria para dictar clases a personas que carecen de conocimiento.

Red (Informática): Conjunto de técnicas, conexiones físicas y programas informáticos empleados para conectar dos o más ordenadores o computadoras. Los usuarios de una red pueden compartir ficheros, impresoras y otros recursos, enviar mensajes electrónicos y ejecutar programas en otros ordenadores.

Plataforma Virtual: Las plataformas virtuales se refieren únicamente a la tecnología utilizada para la creación y desarrollo de cursos o módulos didácticos en la Web que se



usan de manera más amplia en la Web. Constituye un conjunto de estructuras, políticas, técnicas, estrategias y elementos de aprendizaje que se integran en la implementación del proceso de enseñanza-aprendizaje, dentro de las instituciones educativas.

Javascript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos,³ basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

Internet: Internet se define generalmente como la red de redes mundial. Las redes que son parte de esta red se pueden comunicar entre sí a través de un protocolo denominado

Microprocesador (Microprocessor): Es el chip más importante de una computadora. Su velocidad se mide en MHz.

RAM: Memoria de acceso aleatorio. Memoria donde la computadora almacena datos que



le permiten al procesador acceder rápidamente al sistema operativo, las aplicaciones y los datos en uso, tiene estrecha relación con la velocidad de la computadora y se mide en megabytes.

Tecnología: Término general que se aplica al proceso a través del cual los seres humanos diseñan herramientas y máquinas para incrementar su control y su comprensión del entorno material. El término proviene de las palabras griegas tecné, que significa 'arte u oficio', y logos, 'conocimiento o ciencia'; por tanto, la tecnología es el estudio o ciencia de los oficios.

Hardware: Equipo utilizado para el funcionamiento de una computadora. El hardware se refiere a los componentes de un sistema informático. La función de estos componentes suele dividirse en tres categorías principales: entrada, salida y almacenamiento.

Software: Programas de computadoras. Son las instrucciones responsables de que el hardware realice su tarea.

Online: Estado en que se encuentra una computadora cuando se conecta directamente con la red a través de un dispositivo, por ejemplo, un módem.

Antivirus: Programa que detecta la presencia de virus y puede neutralizar sus efectos.

Red En Estrella: En informática, red de área local en la cual cada dispositivo, denominado nodo, está conectado a un ordenador o computadora central con una configuración en forma de estrella. Normalmente, es una red que se compone de un dispositivo central y un conjunto de terminales conectados. En una red en estrella, los mensajes pasan directamente desde un nodo al Hub, el cual gestiona la redistribución de la información a los demás nodos.

Mainboard: Es el lugar al que están conectados directamente todos los componentes internos de la computadora.

Instaladores: Son dispositivos que guardan información o programas que van a ser utilizados en diferentes máquinas.



Firewall: Una computadora que corre un software especial utilizado para prevenir el acceso de usuarios no autorizados a la red. Todo el tráfico de la red debe pasar primero a través de la computadora del firewall.

LAN: Se trata de una red de comunicación de datos geográficamente limitada, por ejemplo, una empresa.

Conexión: Es un enlace que tiene un objeto hacia otro de tal manera que es un puente para la transmisión de datos.

ANEXOS

Manual técnico

Index

```
<?php
```

```
/**
```

```
* @version      $Id: index.php 20806 2011-02-21 19:44:59Z dextercowley $
```

```
* @package      Joomla.Site
```

```
* @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
```

```
* @license      GNU General Public License version 2 or later; see LICENSE.txt
```

```
*/
```

```
// Set flag that this is a parent file.
```

```
define('_JEXEC', 1);
```

```
define('DS', DIRECTORY_SEPARATOR);
```

```
if (file_exists(dirname(__FILE__) . '/defines.php')) {
```

```
    include_once dirname(__FILE__) . '/defines.php';
```

```
}
```



```
if (!defined('_JDEFINES')) {
    define('JPATH_BASE', dirname(__FILE__));
    require_once JPATH_BASE.'/includes/defines.php';
}

require_once JPATH_BASE.'/includes/framework.php';

// Mark afterLoad in the profiler.
JDEBUG ? $_PROFILER->mark('afterLoad') : null;

// Instantiate the application.
$app = JFactory::getApplication('site');

// Initialise the application.
$app->initialise();

// Mark afterInitialise in the profiler.
JDEBUG ? $_PROFILER->mark('afterInitialise') : null;

// Route the application.
$app->route();

// Mark afterRoute in the profiler.
JDEBUG ? $_PROFILER->mark('afterRoute') : null;

// Dispatch the application.
$app->dispatch();

// Mark afterDispatch in the profiler.
```



```
JDEBUG ? $_PROFILER->mark('afterDispatch') : null;
```

```
// Render the application.
```

```
$app->render();
```

```
// Mark afterRender in the profiler.
```

```
JDEBUG ? $_PROFILER->mark('afterRender') : null;
```

```
// Return the response.
```

```
echo $app;
```

Configuración

```
<?php
```

```
class JConfig {
```

```
    public $offline = '0';
```

```
    public $offline_message = 'Este sitio está cerrado por mantenimiento.<br /> Por favor, visítenos más tarde.';
```

```
    public $sitename = 'Delitos Informaticos';
```

```
    public $editor = 'tinymce';
```

```
    public $list_limit = '20';
```

```
    public $access = '1';
```

```
    public $debug = '0';
```

```
    public $debug_lang = '0';
```

```
    public $dbtype = 'mysqli';
```

```
    public $host = 'localhost';
```

```
    public $user = 'root';
```

```
    public $password = '';
```

```
    public $db = 'jpr';
```

```
    public $dbprefix = 'jos_';
```

```
    public $live_site = '';
```



```
public $secret = 'HITJoFEpEQBZxusS';
public $gzip = '0';
public $error_reporting = '-1';
public $helpurl = 'http://help.joomla.org/proxy/index.php?option=com_help&keyref=Help{major}{minor}:{keyref}';
public $ftp_host = '127.0.0.1';
public $ftp_port = '21';
public $ftp_user = '';
public $ftp_pass = '';
public $ftp_root = '';
public $ftp_enable = '0';
public $offset = 'UTC';
public $offset_user = 'UTC';
public $mailer = 'mail';
public $mailfrom = 'brapd@hotmail.com';
public $fromname = 'Delitos Informaticos';
public $sendmail = '/usr/sbin/sendmail';
public $smtpauth = '0';
public $smtpuser = '';
public $smtp_pass = '';
public $smtp_host = 'localhost';
public $smtp_secure = 'none';
public $smtp_port = '25';
public $caching = '0';
public $cache_handler = 'file';
public $cachetime = '15';
public $MetaDesc = '';
public $MetaKeys = '';
public $MetaTitle = '1';
```



```
public $MetaAuthor = '1';
public $sef = '1';
public $sef_rewrite = '0';
public $sef_suffix = '0';
public $unicodeslugs = '0';
public $feed_limit = '10';
public $log_path = 'C:\\wamp\\www\\Delitos Informaticos\\logs';
public $tmp_path = 'C:\\wamp\\www\\Delitos Informaticos\\tmp';
public $lifetime = '15';
public $session_handler = 'database';
}
```

Index Administrador

```
<?php
/**
 * @version      $Id: index.php 20806 2011-02-21 19:44:59Z dextercowley $
 * @package      Joomla.Administrator
 * @copyright     Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

// Set flag that this is a parent file
define('_JEXEC', 1);
define('DS', DIRECTORY_SEPARATOR);

if (file_exists(dirname(__FILE__) . '/defines.php')) {
    include_once dirname(__FILE__) . '/defines.php';
}
}
```



```
if (!defined('_JDEFINES')) {
    define('JPATH_BASE', dirname(__FILE__));
    require_once JPATH_BASE.'/includes/defines.php';
}

require_once JPATH_BASE.'/includes/framework.php';
require_once JPATH_BASE.'/includes/helper.php';
require_once JPATH_BASE.'/includes/toolbar.php';

// Mark afterLoad in the profiler.
JDEBUG ? $_PROFILER->mark('afterLoad') : null;

// Instantiate the application.
$app = JFactory::getApplication('administrator');

// Initialise the application.
$app->initialise(array(
    'language' => $app->getUserState('application.lang', 'lang')
));

// Mark afterInitialise in the profiler.
JDEBUG ? $_PROFILER->mark('afterInitialise') : null;

// Route the application.
$app->route();

// Mark afterRoute in the profiler.
JDEBUG ? $_PROFILER->mark('afterRoute') : null;

// Dispatch the application.
```



```
$app->dispatch();
```

```
// Mark afterDispatch in the profiler.
```

```
JDEBUG ? $_PROFILER->mark('afterDispatch') : null;
```

```
// Render the application.
```

```
$app->render();
```

```
// Mark afterRender in the profiler.
```

```
JDEBUG ? $_PROFILER->mark('afterRender') : null;
```

```
// Return the response.
```

```
echo $app;
```

Loader

```
<?php
```

```
/**
```

```
 * @version $Id: loader.php 20196 2011-01-09 02:40:25Z ian $
```

```
 * @package Joomla.Framework
```

```
 * @copyright Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
```

```
 * @license GNU General Public License version 2 or later; see LICENSE.txt
```

```
 */
```

```
if (!defined('DS')) {
```

```
    define('DS', DIRECTORY_SEPARATOR);
```

```
}
```

```
spl_autoload_register(array('JLoader','load'));
```



```
/**
 * @package      Joomla.Framework
 */
abstract class JLoader
{
    private static $paths = array();

    private static $classes = array();

    /**
     * Loads a class from specified directories.
     *
     * @param string    The class name to look for (dot notation).
     * @param string    Search this directory for the class.
     * @param string    String used as a prefix to denote the full path of the file (dot
notation).
     * @since 1.5
     */
    public static function import($filePath, $base = null, $key = 'libraries.')
    {
        $keyPath = $key ? $key . $filePath : $filePath;

        if (!isset(JLoader::$paths[$keyPath]))
        {
            if (!$base) {
                $base = dirname(__FILE__);
            }

            $parts = explode('.', $filePath);
```



```
$className = array_pop($parts);
switch ($className)
{
    case 'helper' :
        $className =
ucfirst(array_pop($parts)).ucfirst($className);
        break;

    default :
        $className = ucfirst($className);
        break;
}

$path = str_replace('.', DS, $filePath);

if (strpos($filePath, 'joomla.') === 0)
{
    // If we are loading a joomla class prepend the classname
with a capital J.

    $className = 'J'.$className;
    $classes = JLoader::register($className,
$base.DS.$path.'.php');

    $rs = isset($classes[strtolower($className)]);
}
else
{
    // If it is not in the joomla namespace then we have no idea
if

    // it uses our pattern for class names/files so just include
    // if the file exists or set it to false if not
```



```
$filename = $base.DS.$path.'.php';
if (is_file($filename))
{
    $rs = (bool) include_once $filename;
}
else
{
    // if the file doesn't exist fail
    $rs = false;

    // note: JLoader::register does an is_file check itself
so we don't need it above, we do it here because we
    // try to load the file directly and it may not exist
which could cause php to throw up nasty warning messages
    // at us so we set it to false here and hope that if the
programmer is good enough they'll check the return value
    // instead of hoping it'll work. remember include only
fires a warning, so $rs was going to be false with a nasty
    // warning message
}
}

JLoader::$paths[$keyPath] = $rs;
}

return JLoader::$paths[$keyPath];
}

/**
```



```
* Add a class to autoload.
*
* @param string The class name
* @param string Full path to the file that holds the class
* @return array|booleanArray of classes
* @since 1.5
*/
public static function &register($class = null, $file = null)
{
    if ($class && is_file($file))
    {
        // Force to lower case.
        $class = strtolower($class);
        JLoader::$classes[$class] = $file;
    }

    return JLoader::$classes;
}

/**
 * Load the file for a class
 *
 * @param string The class that will be loaded
 * @return boolean True on success
 * @since 1.5
 */
public static function load($class)
{
    $class = strtolower($class); //force to lower case
```



```
        if (class_exists($class)) {
            return true;
        }

        if (array_key_exists($class, JLoader::$classes))
        {
            include_once JLoader::$classes[$class];
            return true;
        }
        return false;
    }
}

/**
 * Global application exit.
 *
 * This function provides a single exit point for the framework.
 *
 * @param mixed Exit code or string. Defaults to zero.
 * @since 1.5
 */
function jexit($message = 0)
{
    exit($message);
}

/**
 * Intelligent file importer
 *
 * @param string A dot syntax path.
```



```
* @since 1.5
```

```
*/
```

```
function jimport($path)
```

```
{
```

```
    return JLoader::import($path);
```

```
}
```

Web

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<configuration>
```

```
    <system.webServer>
```

```
        <rewrite>
```

```
            <rules>
```

```
                <rule name="Imported Rule 1" stopProcessing="true">
```

```
                    <match url="^(.*)$" ignoreCase="false" />
```

```
                    <conditions logicalGrouping="MatchAny">
```

```
                        <add input="{QUERY_STRING}" pattern="base64_encode[^\(\)]*" />
```

```
                    ignoreCase="false" />
```

```
                        <add input="{QUERY_STRING}"
```

```
                            pattern="(&gt;|%3C)([^\s]*s)+cript.*(&lt;|%3E)" />
```

```
                        <add input="{QUERY_STRING}" pattern="GLOBALS(=|\\|\\%[0-9A-Z]{0,2})"
```

```
                    ignoreCase="false" />
```

```
                        <add input="{QUERY_STRING}" pattern="_REQUEST(=|\\|\\%[0-9A-Z]{0,2})"
```

```
                    ignoreCase="false" />
```

```
                </conditions>
```

```
                <action type="CustomResponse" url="index.php" statusCode="403"
```

```
                    statusReason="Forbidden" statusDescription="Forbidden" />
```

```
            </rule>
```

```
            <rule name="Imported Rule 2">
```

```
                <match url="(.*)" ignoreCase="false" />
```



```
<conditions logicalGrouping="MatchAll">
    <add input="{REQUEST_FILENAME}" matchType="IsFile" ignoreCase="false"
negate="true" />
    <add input="{REQUEST_FILENAME}" matchType="IsDirectory"
ignoreCase="false" negate="true" />
    <add input="{URL}" pattern="/^/index.php" ignoreCase="true" negate="true"
/>
    <add input="{URL}" pattern="/component/" ignoreCase="false"
negate="true" />
    <add input="{URL}" pattern="/([^.]*|\.php|html?|feed|pdf|raw)$" />
</conditions>
<action type="Rewrite" url="index.php" />
</rule>
</rules>
</rewrite>
</system.webServer>
</configuration>
```

Acceso

```
##
# @version      $Id: htaccess.txt 21101 2011-04-07 15:47:33Z dextercowley $
# @package      Joomla
# @copyright    Copyright (C) 2005 - 2011 Open Source Matters. All rights reserved.
# @license      GNU General Public License version 2 or later; see LICENSE.txt
##
```

```
##
```

```
# READ THIS COMPLETELY IF YOU CHOOSE TO USE THIS FILE!
```



```
#
# The line just below this section: 'Options +FollowSymLinks' may cause problems
# with some server configurations. It is required for use of mod_rewrite, but may already
# be set by your server administrator in a way that disallows changing it in
# your .htaccess file. If using it causes your server to error out, comment it out (add # to
# beginning of line), reload your site in your browser and test your self url's. If they work,
# it has been set by your server administrator and you do not need it set here.
##

## Can be commented out if causes errors, see notes above.
Options +FollowSymLinks

## Mod_rewrite in use.

RewriteEngine On

## Begin - Rewrite rules to block out some common exploits.
# If you experience problems on your site block out the operations listed below
# This attempts to block the most common type of exploit `attempts` to Joomla!
#
# Block out any script trying to base64_encode data within the URL.
RewriteCond %{QUERY_STRING} base64_encode(?:.*) [OR]
# Block out any script that includes a <script> tag in URL.
RewriteCond %{QUERY_STRING} (<|%)3C(?:.*)+cript.*(>|%)3E [NC,OR]
# Block out any script trying to set a PHP GLOBALS variable via URL.
RewriteCond %{QUERY_STRING} GLOBALS(=|\\|\\%[0-9A-Z]{0,2}) [OR]
# Block out any script trying to modify a _REQUEST variable via URL.
RewriteCond %{QUERY_STRING} _REQUEST(=|\\|\\%[0-9A-Z]{0,2})
# Return 403 Forbidden header and show the content of the root homepage
RewriteRule .* index.php [F]
```



```
#  
## End - Rewrite rules to block out some common exploits.  
  
## Begin - Custom redirects  
#  
# If you need to redirect some pages, or set a canonical non-www to  
# www redirect (or vice versa), place that code here. Ensure those  
# redirects use the correct RewriteRule syntax and the [R=301,L] flags.  
#  
## End - Custom redirects  
  
##  
# Uncomment following line if your webserver's URL  
# is not directly related to physical file paths.  
# Update Your Joomla! Directory (just / for root).  
##  
  
# RewriteBase /  
  
## Begin - Joomla! core SEF Section.  
#  
RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]  
#  
# If the requested path and file is not /index.php and the request  
# has not already been internally rewritten to the index.php script  
RewriteCond %{REQUEST_URI} !^/index\.php  
# and the request is for something within the component folder,  
# or for the site root, or for an extensionless URL, or the  
# requested URL ends with one of the listed extensions
```



```
RewriteCond %{REQUEST_URI} /component/|(/[^.]*|\.(php|html?|feed|pdf|vcf|raw))$
[NC]
# and the requested path and file doesn't directly match a physical file
RewriteCond %{REQUEST_FILENAME} !-f
# and the requested path and file doesn't directly match a physical folder
RewriteCond %{REQUEST_FILENAME} !-d
# internally rewrite the request to the index.php script
RewriteRule .* index.php [L]
<?php
/**
 * @version          $Id: toolbar.php 20224 2011-01-09 22:46:21Z infograf768 $
 * @copyright        Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license          GNU General Public License version 2 or later; see LICENSE.txt
 */

// No direct access
defined('_JEXEC') or die;

jimport('joomla.html.toolbar');

/**
 * Utility class for the button bar.
 *
 * @package          Joomla.Administrator
 * @subpackage       Application
 */
abstract class JToolBarHelper
{
    /**
     * Title cell.

```



* For the title and toolbar to be rendered correctly,
* this title function must be called before the starttable function and the toolbars
icons

* this is due to the nature of how the css has been used to position the title in
respect to the toolbar.

```
*  
* @param string $title The title.  
* @param string $icon The space-separated names of the image.  
* @since 1.5  
*/  
public static function title($title, $icon = 'generic.png')  
{  
    // Strip the extension.  
    $icons = explode(' ', $icon);  
    foreach($icons as &$icon) {  
        $icon = 'icon-48-'.preg_replace('#\.[^.]*$', '', $icon);  
    }  
  
    $html = '<div class="pagetitle '.implode(' ',  
$icons).'><h2>'. $title.'</h2></div>';  
  
    $app = JFactory::getApplication();  
    $app->set('JComponentTitle', $html);  
}  
  
/**  
* Writes a spacer cell.  
*  
* @param string $width The width for the cell  
* @since 1.0
```



```
*/  
public static function spacer($width = "")  
{  
    $bar = JToolBar::getInstance('toolbar');  
    // Add a spacer.  
    $bar->appendButton('Separator', 'spacer', $width);  
}  
  
/**  
 * Writes a divider between menu buttons  
 *  
 * @since 1.0  
 */  
public static function divider()  
{  
    $bar = JToolBar::getInstance('toolbar');  
    // Add a divider.  
    $bar->appendButton('Separator', 'divider');  
}  
  
/**  
 * Writes a custom option and task button for the button bar.  
 *  
 * @param string $task      The task to perform (picked up by the  
switch($task) blocks.  
 * @param string $icon      The image to display.  
 * @param string $iconOver  The image to display when moused over.  
 * @param string $alt       The alt text for the icon image.  
 * @param bool $listSelect  True if required to check that a standard list  
item is checked.
```



```
* @since    1.0
*/
public static function custom($task = "", $icon = "", $iconOver = "", $alt = "",
$listSelect = true)
{
    $bar = JToolBar::getInstance('toolbar');

    // Strip extension.
    $icon = preg_replace('#\.[^.]*$#', "", $icon);

    // Add a standard button.
    $bar->appendButton('Standard', $icon, $alt, $task, $listSelect);
}

/**
 * Writes a custom option and task button for the button bar.
 *
 * @param    string $task        The task to perform (picked up by the
switch($task) blocks.
 * @param    string $icon        The image to display.
 * @param    string $iconOver    The image to display when moused over.
 * @param    string $alt        The alt text for the icon image.
 * @param    bool  $listSelect   True if required to check that a standard list
item is checked.
 * @since    1.0
 * @deprecated
 */
public static function customX($task = "", $icon = "", $iconOver = "", $alt = "",
$listSelect = true)
{
```



```
self::custom($task, $icon, $iconOver, $alt, $listSelect);
}

/**
 * Writes a preview button for a given option (opens a popup window).
 *
 * @param string $url The name of the popup file (excluding the file
extension)
 * @param bool $updateEditors
 * @since 1.0
 */
public static function preview($url = "", $updateEditors = false)
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a preview button.
    $bar->appendButton('Popup', 'preview', 'Preview', $url.'&task=preview');
}

/**
 * Writes a preview button for a given option (opens a popup window).
 *
 * @param string $ref The name of the popup file (excluding the file
extension for an xml file).
 * @param bool $com Use the help file in the component directory.
 * @param string $override Use this URL instead of any other
 * @param string $component Name of component to get Help (null for
current component)
 * @since 1.0
 */
public static function help($ref, $com = false, $override = null, $component = null)
```



```
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a help button.
    $bar->appendButton('Help', $ref, $com, $override, $component);
}

/**
 * Writes a cancel button that will go back to the previous page without doing
 * any other operation.
 *
 * @param string $alt Alternative text.
 * @param string $href URL of the href attribute.
 * @since 1.0
 */
public static function back($alt = 'JTOOLBAR_BACK', $href =
'javascript:history.back();')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a back button.
    $bar->appendButton('Link', 'back', $alt, $href);
}

/**
 * Writes a media_manager button.
 *
 * @param string $directory The sub-drectory to upload the media to.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function media_manager($directory = "", $alt = 'JTOOLBAR_UPLOAD')
```



```
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an upload button.
    $bar->appendButton('Popup', 'upload', $alt,
        'index.php?option=com_media&tmpl=component&task=popupUpload&folder='.$directory, 800, 520);
}

/**
 * Writes a common 'default' button for a record.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function makeDefault($task = 'default', $alt = 'JTOOLBAR_DEFAULT')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a default button.
    $bar->appendButton('Standard', 'default', $alt, $task, true);
}

/**
 * Writes a common 'assign' button for a record.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function assign($task = 'assign', $alt = 'JTOOLBAR_ASSIGN')
```



```
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an assign button.
    $bar->appendButton('Standard', 'assign', $alt, $task, true);
}

/**
 * Writes the common 'new' icon for the button bar.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function addNew($task = 'add', $alt = 'JTOOLBAR_NEW')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a new button.
    $bar->appendButton('Standard', 'new', $alt, $task, false);
}

/**
 * Writes the common 'new' icon for the button bar.
 * Extended version of addNew() calling hideMainMenu() before
Joomla.submitbutton().
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 * @deprecated
 */
```



```
public static function addNewX($task = 'add', $salt = 'JTOOLBAR_NEW')
{
    self::addNew($task, $salt);
}

/**
 * Writes a common 'publish' button.
 *
 * @param string $task An override for the task.
 * @param string $salt An override for the alt text.
 * @since 1.0
 */
public static function publish($task = 'publish', $salt = 'JTOOLBAR_PUBLISH')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a publish button.
    $bar->appendButton('Standard', 'publish', $salt, $task, false);
}

/**
 * Writes a common 'publish' button for a list of records.
 *
 * @param string $task An override for the task.
 * @param string $salt An override for the alt text.
 * @since 1.0
 */
public static function publishList($task = 'publish', $salt = 'JTOOLBAR_PUBLISH')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a publish button (list).
```



```
$bar->appendButton('Standard', 'publish', $alt, $task, true);
}

/**
 * Writes a common 'unpublish' button.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function unpublish($task = 'unpublish', $alt =
'JTOOLBAR_UNPUBLISH')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an unpublish button
    $bar->appendButton('Standard', 'unpublish', $alt, $task, false);
}

/**
 * Writes a common 'unpublish' button for a list of records.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function unpublishList($task = 'unpublish', $alt =
'JTOOLBAR_UNPUBLISH')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an unpublish button (list).
```



```
$bar->appendButton('Standard', 'unpublish', $alt, $task, true);
}

/**
 * Writes a common 'archive' button for a list of records.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function archiveList($task = 'archive', $alt = 'JTOOLBAR_ARCHIVE')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an archive button.
    $bar->appendButton('Standard', 'archive', $alt, $task, true);
}

/**
 * Writes an unarchive button for a list of records.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function unarchiveList($task = 'unarchive', $alt =
'JTOOLBAR_UNARCHIVE')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an unarchive button (list).
    $bar->appendButton('Standard', 'unarchive', $alt, $task, true);
```



```
}

/**
 * Writes a common 'edit' button for a list of records.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function editList($task = 'edit', $alt = 'JTOOLBAR_EDIT')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an edit button.
    $bar->appendButton('Standard', 'edit', $alt, $task, true);
}

/**
 * Writes a common 'edit' button for a list of records.
 * Extended version of editList() calling hideMainMenu() before
 Joomla.submitbutton().
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 * @deprecated
 */
public static function editListX($task = 'edit', $alt = 'JTOOLBAR_EDIT')
{
    self::editList($task, $alt);
}
```



```
/**
 * Writes a common 'edit' button for a template html.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function editHtml($task = 'edit_source', $alt =
'JTOOLBAR_EDIT_HTML')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an edit html button.
    $bar->appendButton('Standard', 'edithtml', $alt, $task, true);
}

/**
 * Writes a common 'edit' button for a template html.
 * Extended version of editHtml() calling hideMainMenu() before
Joomla.submitbutton().
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 * @deprecated
 */
public static function editHtmlX($task = 'edit_source', $alt =
'JTOOLBAR_EDIT_HTML')
{
    self::editHtml($task, $alt);
}
```



```
}

/**
 * Writes a common 'edit' button for a template css.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function editCss($task = 'edit_css', $alt = 'JTOOLBAR_EDIT_CSS')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an edit css button (hide).
    $bar->appendButton('Standard', 'editcss', $alt, $task, true);
}

/**
 * Writes a common 'edit' button for a template css.
 * Extended version of editCss() calling hideMainMenu() before
 Joomla.submitbutton().
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 * @deprecated
 */
public static function editCssX($task = 'edit_css', $alt = 'JTOOLBAR_EDIT_CSS')
{
    self::editCss($task, $alt);
}
```



```
/**
 * Writes a common 'delete' button for a list of records.
 *
 * @param string $msg Postscript for the 'are you sure' message.
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
 */
public static function deleteList($msg = "", $task = 'remove', $alt =
'JTOOLBAR_DELETE')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a delete button.
    if ($msg) {
        $bar->appendButton('Confirm', $msg, 'delete', $alt, $task, true);
    } else {
        $bar->appendButton('Standard', 'delete', $alt, $task, true);
    }
}

/**
 * Writes a common 'delete' button for a list of records.
 * Extended version of deleteList() calling hideMainMenu() before
Joomla.submitbutton().
 *
 * @param string $msg Postscript for the 'are you sure' message.
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @since 1.0
```



```
* @deprecated
*/
public static function deleteList($msg = "", $task = 'remove', $alt =
'JTOOLBAR_DELETE')
{
    self::deleteList($msg, $task, $alt);
}

/**
 * Write a trash button that will move items to Trash Manager.
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
 * @param bool $check
 * @since 1.0
 */
public static function trash($task = 'remove', $alt = 'JTOOLBAR_TRASH', $check =
true)
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a trash button.
    $bar->appendButton('Standard', 'trash', $alt, $task, $check, false);
}

/**
 * Writes a save button for a given option.
 * Apply operation leads to a save action only (does not leave edit mode).
 *
 * @param string $task An override for the task.
 * @param string $alt An override for the alt text.
```



```
* @since    1.0
*/
public static function apply($task = 'apply', $salt = 'JTOOLBAR_APPLY')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add an apply button
    $bar->appendButton('Standard', 'apply', $salt, $task, false);
}

/**
 * Writes a save button for a given option.
 * Save operation leads to a save and then close action.
 *
 * @param    string  $task  An override for the task.
 * @param    string  $salt  An override for the alt text.
 * @since    1.0
 */
public static function save($task = 'save', $salt = 'JTOOLBAR_SAVE')
{
    $bar = JToolBar::getInstance('toolbar');
    // Add a save button.
    $bar->appendButton('Standard', 'save', $salt, $task, false);
}

/**
 * Writes a cancel button and invokes a cancel operation (eg a checkin).
 *
 * @param    string  $task  An override for the task.
 * @param    string  $salt  An override for the alt text.
 * @since    1.0
```



```
*/  
public static function cancel($task = 'cancel', $salt = 'JTOOLBAR_CANCEL')  
{  
    $bar = JToolBar::getInstance('toolbar');  
    // Add a cancel button.  
    $bar->appendButton('Standard', 'cancel', $salt, $task, false);  
}  
  
/**  
 * Writes a configuration button and invokes a cancel operation (eg a checkin).  
 *  
 * @param string $component The name of the component, eg,  
com_content.  
 * @param int $height The height of the popup.  
 * @param int $width The width of the popup.  
 * @param string $salt The name of the button.  
 * @param string $path An alternative path for the configuration xml  
relative to JPATH_SITE.  
 * @since 1.0  
*/  
public static function preferences($component, $height = '550', $width = '875',  
$salt = 'JToolBar_Options', $path = '', $onClose = '')  
{  
    $component = urlencode($component);  
    $path = urlencode($path);  
    $top = 0;  
    $left = 0;  
    $bar = JToolBar::getInstance('toolbar');  
    // Add a configuration button.
```



```
        $bar->appendButton('Popup',                'options',                $alt,
        'index.php?option=com_config&view=component&component='.$component.'
        &path='.$path.'&tmpl=component', $width, $height, $top, $left, $onClose);

    }
}

/**
 * Utility class for the submenu.
 *
 * @package      Joomla.Administrator
 */
abstract class JSubMenuHelper
{
    /**
     * Method to add a menu item to submenu.
     *
     * @param   string  $name  Name of the menu item.
     * @param   string  $link  URL of the menu item.
     * @param   bool    True if the item is active, false otherwise.
     */
    public static function addEntry($name, $link = "", $active = false)
    {
        $menu = JToolBar::getInstance('submenu');
        $menu->appendButton($name, $link, $active);
    }
}

<?php
/**
 * @version      $Id: router.php 21097 2011-04-07 15:38:03Z dextercowley $
```



```
* @package Joomla.Administrator
* @subpackage Application
* @copyright Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
* @license GNU General Public License version 2 or later; see LICENSE.txt
*/
```

```
// No direct access
```

```
defined('JPATH_BASE') or die;
```

```
/**
```

```
* Class to create and parse routes
```

```
*
```

```
* @package Joomla.Administrator
```

```
* @since 1.5
```

```
*/
```

```
class JRouterAdministrator extends JRouter
```

```
{
```

```
    /**
```

```
     * Function to convert a route to an internal URI
```

```
    */
```

```
    public function parse(&$uri)
```

```
    {
```

```
        return array();
```

```
    }
```

```
    /**
```

```
     * Function to convert an internal URI to a route
```

```
     *
```

```
     * @param string $uri The internal URL
```

```
     *
```



```
* @return    string  The absolute search engine friendly URL
* @since     1.5
*/
function build($url)
{
    // Create the URI object
    $uri = parent::build($url);

    // Get the path data
    $route = $uri->getPath();

    //Add basepath to the uri
    $uri->setPath(JURI::base(true).'/'.$route);

    return $uri;
}
}
<?php
/**
 * @version    $Id: menu.php 21097 2011-04-07 15:38:03Z dextercowley $
 * @copyright  Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license    GNU General Public License version 2 or later; see LICENSE.txt
 */

// No direct access.
defined('_JEXEC') or die;

/**
 * JMenu class.
 *
 */
```



```
* @package      Joomla.Administrator
* @subpackage    Application
* @since         1.5
*/

class JMenuAdministrator extends JMenu
{
}

<?php
/**
 * @version      $Id: helper.php 20196 2011-01-09 02:40:25Z ian $
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

// no direct access
defined('_JEXEC') or die;

/**
 * Joomla! Administrator Application helper class.
 * Provide many supporting API functions.
 *
 * @package      Joomla.Administrator
 * @subpackage    Application
 */
class JAdministratorHelper
{
    /**
     * Return the application option string [main component].
     *
     * @return      string      Option.
```



```
* @since    1.5
*/
public static function findOption()
{
    $option = strtolower(JRequest::getCmd('option'));

    $user = JFactory::getUser();
    if ($user->get('guest')) {
        $option = 'com_login';
    }

    if (empty($option)) {
        $option = 'com_cpanel';
    }

    JRequest::setVar('option', $option);
    return $option;
}
}
<?php
/**
 * @version    $Id: framework.php 21171 2011-04-18 21:50:18Z dextercowley $
 * @package    Joomla.Administrator
 * @subpackage Application
 * @copyright  Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license    GNU General Public License version 2 or later; see LICENSE.txt
 */

// No direct access.
defined('_JEXEC') or die;
```



```
/*
 * Joomla! system checks.
 */

@ini_set('magic_quotes_runtime', 0);
@ini_set('zend.ze1_compatibility_mode', '0');

/*
 * Installation check, and check on removal of the install directory.
 */
if (!file_exists(JPATH_CONFIGURATION.'/configuration.php')) ||
(filesize(JPATH_CONFIGURATION.'/configuration.php') < 10) ||
file_exists(JPATH_INSTALLATION.'/index.php')) { header('Location:
../installation/index.php');
    exit();
}

/*
 * Joomla! system startup
 */

// System includes.
require_once JPATH_LIBRARIES.'/joomla/import.php';

// Pre-Load configuration.
require_once JPATH_CONFIGURATION.'/configuration.php';

// System configuration.
$CONFIG = new JConfig();
```



```
if (@$CONFIG->error_reporting === 0) {
    error_reporting(0);
} else if (@$CONFIG->error_reporting > 0) {
    error_reporting($CONFIG->error_reporting);
    ini_set('display_errors', 1);
}

define('JDEBUG', $CONFIG->debug);

unset($CONFIG);

/*
 * Joomla! framework loading.
 */

// System profiler.
if (JDEBUG) {
    jimport('joomla.error.profiler');
    $_PROFILER = JProfiler::getInstance('Application');
}

// Joomla! library imports.
jimport('joomla.application.menu');
jimport('joomla.user.user');
jimport('joomla.environment.uri');
jimport('joomla.html.html');
jimport('joomla.html.parameter');
jimport('joomla.utilities.utility');
jimport('joomla.event.event');
```



```
jimport('joomla.event.dispatcher');
jimport('joomla.language.language');
jimport('joomla.utilities.string');
<?php
/**
 * @version      $Id: defines.php 21602 2011-06-21 17:18:18Z dextercowley $
 * @package      Joomla.Administrator
 * @subpackage   Application
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

// No direct access.
defined('_JEXEC') or die;

//Global definitions.
//Joomla framework path definitions.
$parts = explode(DS, JPATH_BASE);
array_pop($parts);

//Defines.
define('JPATH_ROOT',          implode(DS, $parts));
define('JPATH_SITE',          JPATH_ROOT);
define('JPATH_CONFIGURATION', JPATH_ROOT);
define('JPATH_ADMINISTRATOR', JPATH_ROOT.DS.'administrator');
define('JPATH_LIBRARIES',     JPATH_ROOT.DS.'libraries');
define('JPATH_PLATFORM',     JPATH_ROOT.DS.'libraries');
define('JPATH_PLUGINS',       JPATH_ROOT.DS.'plugins');
define('JPATH_INSTALLATION',  JPATH_ROOT.DS.'installation');
define('JPATH_THEMES',        JPATH_BASE.DS.'templates');
```



```
define('JPATH_CACHE',                JPATH_BASE.DS.'cache');
define('JPATH_MANIFESTS',            JPATH_ADMINISTRATOR.DS.'manifests');
<?php
/**
 * @version      $Id: application.php 21169 2011-04-18 19:52:28Z dextercowley $
 * @package      Joomla.Administrator
 * @copyright    Copyright (C) 2005 - 2011 Open Source Matters, Inc. All rights reserved.
 * @license      GNU General Public License version 2 or later; see LICENSE.txt
 */

// no direct access
defined('_JEXEC') or die;

jimport('joomla.application.component.helper');

/**
 * Joomla! Application class
 *
 * Provide many supporting API functions
 *
 * @package      Joomla.Administrator
 * @final
 * @since        1.5
 */
class JAdministrator extends JApplication
{
    /**
     * Class constructor
     *
     * @param    array    An optional associative array of configuration settings.

```



* Recognized key values include 'clientId' (this list is not meant to be comprehensive).

*

* @since 1.5

*/

```
public function __construct($config = array())
```

```
{
```

```
    $config['clientId'] = 1;
```

```
    parent::__construct($config);
```

```
    //Set the root in the URI based on the application name
```

```
    JURI::root(null, str_ireplace('/'. $this->getName(), "", JURI::base(true)));
```

```
}
```

```
/**
```

```
 * Initialise the application.
```

```
 *
```

```
 * @param array $options An optional associative array of configuration settings.
```

```
 *
```

```
 * @return void
```

```
 * @since 1.5
```

```
*/
```

```
function initialise($options = array())
```

```
{
```

```
    $config = JFactory::getConfig();
```

```
    // if a language was specified it has priority
```

```
    // otherwise use user or default language settings
```

```
    if (empty($options['language']))
```



```
{
    $user = JFactory::getUser();
    $lang = $user->getParam('admin_language');

    // Make sure that the user's language exists
    if ($lang && JLanguage::exists($lang)) {
        $options['language'] = $lang;
    } else {
        $params = JComponentHelper::getParams('com_languages');
        $client = JApplicationHelper::getClientInfo($this->getClientId());
        $options['language'] = $params->get($client->name, $config->get('language','en-GB'));
    }
}

// One last check to make sure we have something
if (!JLanguage::exists($options['language'])) {
    $lang = $config->get('language','en-GB');
    if (JLanguage::exists($lang)) {
        $options['language'] = $lang;
    } else {
        $options['language'] = 'en-GB'; // as a last ditch fail to
english
    }
}

// Execute the parent initialise method.
parent::initialise($options);
```



```
// Load Library language
$lang = JFactory::getLanguage();
$lang->load('lib_joomla', JPATH_ADMINISTRATOR);
}

/**
 * Route the application
 *
 * @return void
 * @since 1.5
 */
public function route()
{
    $uri = JURI::getInstance();

    if ($this->getCfg('force_ssl') >= 1 && strtolower($uri->getScheme()) !=
'https') {

        //forward to https
        $uri->setScheme('https');
        $this->redirect((string)$uri);
    }

    // Trigger the onAfterRoute event.
    JPluginHelper::importPlugin('system');
    $this->triggerEvent('onAfterRoute');
}

/**
 * Return a reference to the JRouter object.
 *

```



```
* @return  JRouter
* @since   1.5
*/
static public function getRouter($name = null, array $options = array())
{
    $router = parent::getRouter('administrator');
    return $router;
}

/**
 * Dispatch the application
 *
 * @param   string $component The component to dispatch.
 *
 * @return  void
 * @since   1.5
 */
public function dispatch($component = null)
{
    try
    {
        if ($component === null) {
            $component = JAdministratorHelper::findOption();
        }

        $document = JFactory::getDocument();
        $user      = JFactory::getUser();

        switch ($document->getType()) {
            case 'html':
```



```

                                $document->setMetaData('keywords',      $this-
>getCfg('MetaKeys'));

                                JHtml::_('behavior.framework', true);
                                break;

                                default:
                                break;
                                }

                                $document->setTitle($this->getCfg('sitename'). ' - ');
.JText::_('JADMINISTRATION');
                                $document->setDescription($this->getCfg('MetaDesc'));

                                $contents = JComponentHelper::renderComponent($component);
                                $document->setBuffer($contents, 'component');

                                // Trigger the onAfterDispatch event.
                                JPluginHelper::importPlugin('system');
                                $this->triggerEvent('onAfterDispatch');
                                }

                                // Mop up any uncaught exceptions.
                                catch (Exception $e)
                                {
                                    $code = $e->getCode();
                                    JError::raiseError($code ? $code : 500, $e->getMessage());
                                }
                                }

/**
 * Display the application.

```



```
*
* @return void
* @since 1.5
*/
public function render()
{
    $component = JRequest::getCmd('option', 'com_login');
    $template = $this->getTemplate(true);
    $file = JRequest::getCmd('tmpl', 'index');

    if ($component == 'com_login') {
        $file = 'login';
    }

    // Safety check for when configuration.php root_user is in use.
    $config = JFactory::getConfig();
    $rootUser = $config->get('root_user');
    if (property_exists('JConfig', 'root_user') &&
        (JFactory::getUser()->get('username') == $rootUser ||
        JFactory::getUser()->id === (string) $rootUser)) {
        JError::raiseNotice(200,
        JText::sprintf('JWARNING_REMOVE_ROOT_USER',
        'index.php?option=com_config&task=application.remove_root&'. JFactory::getToken()
        .'=1'));
    }

    $params = array(
        'template' => $template->template,
        'file' => $file.'.php',
        'directory' => JPATH_THEMES,
```



```
        'params'      => $template->params
    );

    $document = JFactory::getDocument();
    $document->parse($params);
    $this->triggerEvent('onBeforeRender');
    $data = $document->render(false, $params);
    JResponse::setBody($data);
    $this->triggerEvent('onAfterRender');

}

/**
 * Login authentication function
 *
 * @param array Array('username' => string, 'password' => string)
 * @param array Array('remember' => boolean)
 *
 * @return boolean True on success.
 * @see      JFactory::login
 * @since    1.5
 */
public function login($credentials, $options = array())
{
    //The minimum group
    $options['group'] = 'Public Backend';

    //Make sure users are not autoregistered
    $options['autoregister'] = false;
```



```
//Set the application login entry point
if (!array_key_exists('entry_url', $options)) {
    $options['entry_url'] =
JURI::base().'.index.php?option=com_users&task=login';
}

// Set the access control action to check.
$options['action'] = 'core.login.admin';

$result = parent::login($credentials, $options);

if (!JError::isError($result))
{
    $lang = JRequest::getCmd('lang');
    $lang = preg_replace('/[^A-Z-]/i', '', $lang);
    $this->setUserState('application.lang', $lang );

    JAdministrator::purgeMessages();
}

return $result;
}

/**
 * Get the template
 *
 * @return string The template name
 * @since 1.0
 */
public function getTemplate($params = false)
```



```
{
    static $template;

    if (!isset($template))
    {
        $admin_style = JFactory::getUser()->getParam('admin_style');
        // Load the template name from the database
        $db = JFactory::getDbo();
        $query = $db->getQuery(true);
        $query->select('template, params');
        $query->from('#__template_styles');
        $query->where('client_id = 1');
        if ($admin_style)
        {
            $query->where('id = '(int)$admin_style);
        }
        else
        {
            $query->where('home = 1');
        }
        $db->setQuery($query);
        $template = $db->loadObject();

        $template->template = JFactoryInput::getInstance()-
>clean($template->template, 'cmd');
        $template->params = new JRegistry($template->params);

        if (!file_exists(JPATH_THEMES.DS.$template-
>template.DS.'index.php'))
        {
```



```
        $template->params = new JRegistry();
        $template->template = 'bluestork';
    }
}
if ($params) {
    return $template;
}

return $template->template;
}

/**
 * Purge the jos_messages table of old messages
 *
 * @return void
 * @since 1.5
 */
public static function purgeMessages()
{
    $db = JFactory::getDbo();
    $user = JFactory::getUser();

    $userid = $user->get('id');

    $query = 'SELECT *'
        . ' FROM #__messages_cfg'
        . ' WHERE user_id = ' . (int) $userid
        . ' AND cfg_name = "auto_purge"'
        ;
    $db->setQuery($query);
```



```
$config = $db->loadObject();

// check if auto_purge value set
if (is_object($config) and $config->cfg_name == 'auto_purge') {
    $purge = $config->cfg_value;
} else {
    // if no value set, default is 7 days
    $purge = 7;
}

// calculation of past date

// if purge value is not 0, then allow purging of old messages
if ($purge > 0) {
    // purge old messages at day set in message configuration
    $past = JFactory::getDate(time() - $purge * 86400);
    $pastStamp = $past->toMySQL();

    $query = 'DELETE FROM #__messages'
    . ' WHERE date_time < ' . $db->Quote($pastStamp)
    . ' AND user_id_to = ' . (int) $userid
    ;
    $db->setQuery($query);
    $db->query();
}
}
```

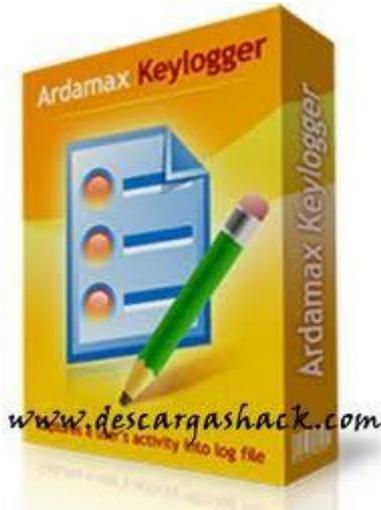
Dreamweaver



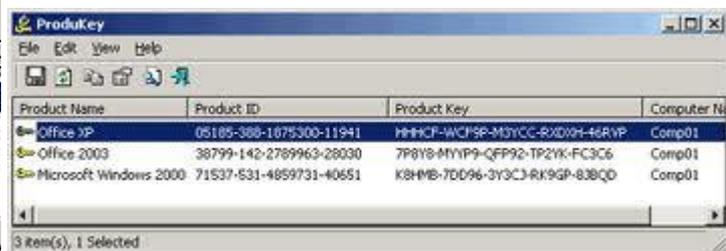
Wampserver



Ardamax



Produkey



UsbStealer

