

TEMA 1: DESARROLLO WEB EN ENTORNO SERVIDOR

Indice

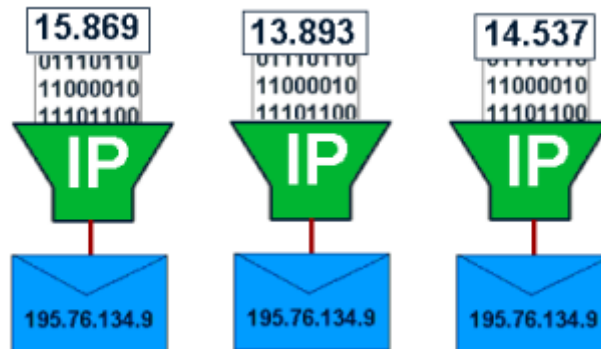
EJERCICIO 1.....	3
EJERCICIO 2.....	4
EJERCICIO 3.....	5
EJERCICIO 4.....	6
EJERCICIO 5.....	7
EJERCICIO 6.....	7
EJERCICIO 7.....	8
EJERCICIO 8.....	8
EJERCICIO 9.....	8
EJERCICIO 10.....	9
EJERCICIO 11.....	9
EJERCICIO 12.....	10
EJERCICIO 13.....	11
EJERCICIO 14.....	13
EJERCICIO 15.....	15
EJERCICIO 16.....	16
EJERCICIO 17.....	17
EJERCICIO 18.....	19

EJERCICIO 1

1. Protocolos de comunicaciones: IP, TCP, HTTP, HTTPS.

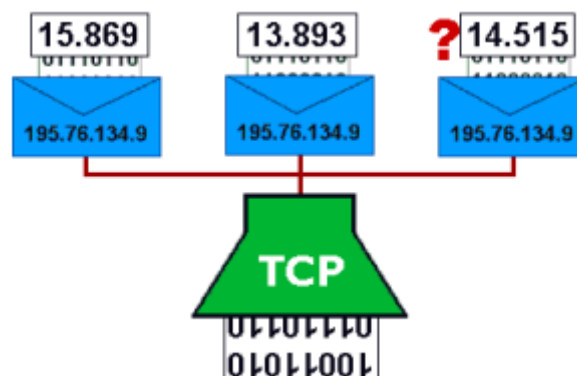
Protocolo IP

Es un protocolo de comunicación de datos digitales clasificado en la capa de red según el modelo OSI. Su función principal es el uso bidireccional en origen o destino de comunicación para transferir datos mediante un protocolo que transfiere paquetes conmutados a través de distintas redes físicas previamente enlazadas según la norma OSI de enlace de datos.



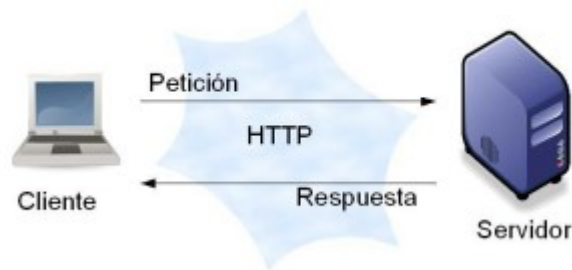
Protocolo TCP:

Protocolo de Control de Transmisión, que es el más utilizado en internet. El protocolo garantiza que los datos serán entregados en su destino sin errores y en el mismo orden en que se transmitieron. También proporciona un mecanismo para distinguir distintas aplicaciones dentro de una misma equina, a través del concepto de puerto. TCP da soporte a muchas de las aplicaciones más populares de Internet y protocolos de aplicación HTTP, SMTP, SSH y FTP.



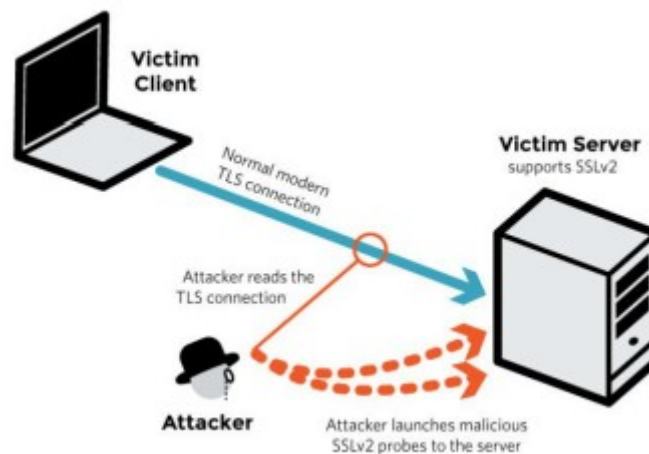
Protocolo HTTP:

El protocolo de comunicación que permite las transferencias de información en la World Wide Web. HTTP define la sintaxis y la semántica que utilizan los elementos de software de la arquitectura web (clientes, servidores, proxies) para comunicarse. HTTP es un protocolo sin estado, es decir, no guarda ninguna información sobre conexiones anteriores. El desarrollo de aplicaciones web necesita frecuentemente mantener estado. Para esto se usan las cookies, que es información que un servidor puede almacenar en el sistema cliente. Esto permite a las aplicaciones web rastrear usuarios ya que las cookies pueden guardarse en el cliente por tiempo indeterminado.



Protocolo HTTPS:

Es un protocolo de aplicación basado en HTTP, la versión segura del mismo. El sistema HTTPS utiliza un cifrado basado en SSL/TLS para crear un canal cifrado más apropiado para el tráfico de información sensible que el protocolo HTTP. De este modo se consigue que la información sensible no pueda ser usada por un atacante que haya conseguido interceptar la transferencia de datos de la conexión, ya que lo único que obtendrá será un flujo de datos cifrados que le resultará imposible de descifrar.

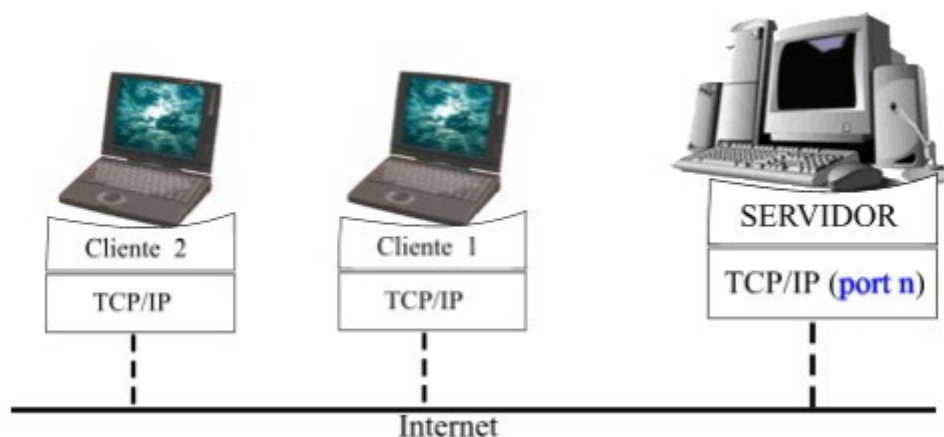


EJERCICIO 2

2. Modelo de comunicaciones cliente – servidor y su relación con las aplicaciones web

En el modelo cliente-servidor los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes

cada vez que estos son solicitados. La relación que tiene el modelo de comunicaciones cliente-servidor con las aplicaciones web, es, principalmente, que, las aplicaciones web se basan en protocolo HTTP, el cual se basa en el modelo cliente-servidor.



EJERCICIO 3

3. Estudio sobre los métodos de petición HTTP /HTTPS más utilizados

GET: Este método se utiliza cuando se necesita adquirir un archivo o recurso que se encuentran en un servidor web, este método devuelve la cabecera que contienen los metadatos del recurso solicitado y el recurso.

HEAD: Este método realiza una acción similar al método GET. El método HEAD solo solicita los metadatos de un recurso o archivo y no todo el elemento.

POST: Indica al servidor que va a recibir información del cliente. Suele usarse para enviar información desde formularios.

PUT: Crea/Carga un nuevo recurso al servidor, o en caso de que el objeto ya exista en el servidor reemplaza el recurso existente con el recurso que se carga.

Método DELETE: Este método le solicita al servidor web que se borre un recurso en específico.

Método TRACE: Este método permite monitorizar los mensajes que hay entre el cliente y el servidor web.

Método CONNECT: Se utiliza cuando se necesita utilizar un proxy para una conexión segura cifrada HTTPS o para comunicaciones vía SSL.

OPTIONS: El método OPTIONS devuelve los métodos HTTP que el servidor admite para la URL especificada. Esto se puede utilizar para comprobar la funcionalidad de un servidor web solicitando '*' en lugar de un recurso específico.

PATCH: El método PATCH aplica modificaciones parciales a un recurso.

EJERCICIO 4

4. Modelo de desarrollo de aplicaciones multicapa – comunicación entre capas - componentes – funcionalidad de cada capa.

Es un modelo de desarrollo del software cuyo objetivo principal es la separación entre las distintas que componen un software. Así es más sencillo crear diferentes interfaces sobre un mismo sistema sin requerir cambios en la capa de datos o en la lógica.

La principal ventaja es que el desarrollo se puede llevar a cabo en varios niveles, y en caso de algún cambio, que este solo afecte al nivel requerido sin tener que revisar entre el código de los otros niveles. También permite distribuir el trabajo, ya que cada grupo podrá ocuparse de un nivel.

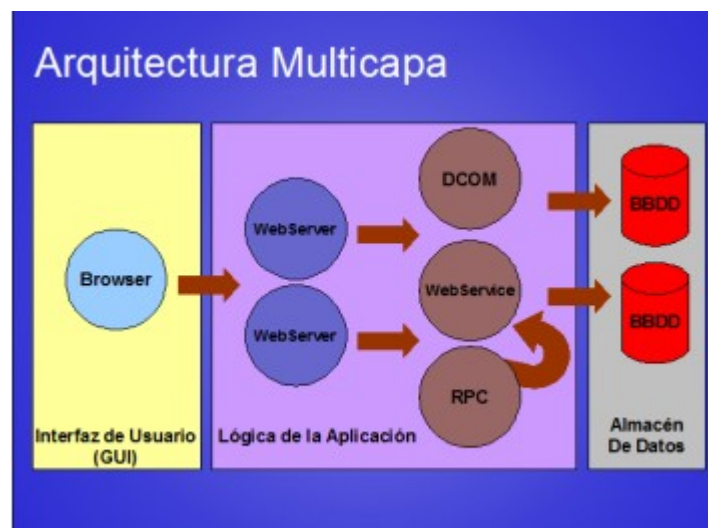
Actualmente el mas utilizado se realiza en tres capas:

Capa de presentación: es la que ve el usuario, le comunica la información y captura la información del usuario. También se conoce como interfaz gráfica y debe ser fácil de usar para el usuario. Esta capa se comunica únicamente con la capa de negocio.

Capa de negocio: en ella se encuentran los programas que se ejecutan, reciben las peticiones del usuario y envía las respuestas de negocio. Se le llama así porque es donde se establecen las normas que deben seguirse. Se conecta con la capa de presentación para recibir solicitudes y enviar resultados y con la capa de datos para solicitar el almacenamiento o recuperación de datos.

Capa de datos: es donde se encuentran los datos y se encarga de acceder a los mismos. Está compuesta por uno o varios gestores de bases de datos que almacenan los datos o recuperan los mismos desde la capa de negocio.

Todas estas capas pueden encontrarse en un mismo ordenador, aunque la capa de presentación es normal que se encuentre en multitud de ordenadores.



EJERCICIO 5

5. Modelo de división funcional front-end / back-end para aplicaciones web.

Front-end: Parte de la pagina web que interactua con el usuario

Back-end: Parte donde se programa el codigo de las paginas web

EJERCICIO 6

6. Página web estática – página web dinámica – aplicación web.

Página web estática:

Muestra información permanente, donde el usuario se limita a obtener información sin tener que interactuar con la página. Se construyen con enlaces entre las diferentes páginas que componen el sitio.

Una de las ventajas está en lo económico que resulta crearlas, pero la gran desventaja se encuentra en lo difícil que resulta su actualización.

Página web dinámica:

Las páginas web dinámicas son aquellas en las que la información presentada se genera a partir de una petición del usuario de la página. Tiene asociada una Base de Datos desde la que se permite visualizar la información contenida en ella.

Una de las ventajas de este tipo de páginas es que permiten almacenar y hacer actualizaciones de la información contenida en la misma, así como también modificaciones dinámicas de la estructura y del diseño por parte de su propietario.

Para la creación de este tipo de páginas, deberán utilizarse etiquetas HTML y algún lenguaje de programación que se ejecute “del lado del servidor”, es decir, que esté alojado en el mismo servidor en el que está alojada la página web.

Los lenguajes utilizados para la generación de este tipo de páginas son principalmente: Perl CGI, PHP, JSP y ASP.

Aplicación web:

Es el conjunto de herramientas que los usuarios pueden usar accediendo a un servidor web a través de Internet, o de un navegador.

Es una aplicación que se codifica en un lenguaje soportado por los navegadores, que confían en la ejecución del navegador.

Son populares debido a lo práctico del navegador como cliente ligero, independiente del sistema operativo, y con facilidad para actualizar y mantener aplicaciones web sin distribuir ni instalar software. Algunos ejemplos son las tiendas en línea o Wikipedia.

EJERCICIO 7

7. Componentes de una aplicación web.

Toda aplicación web, debe componerse de un navegador web, un servidor de aplicaciones y un servidor de base de datos.

El servidor de base de datos es el programa que permite guardar, codificar o mostrar los datos que se almacenan en la aplicación.

El servidor de aplicaciones es el servidor que se encuentra en una red y que ejecuta ciertas aplicaciones. Se trata de un dispositivo que proporciona servicios de aplicación a las computadoras del cliente.

El navegador web es el programa que permite el acceso a la web, interpretando la información de los distintos archivos y sitios web para que estos puedan ser visualizados.

EJERCICIO 8

8. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor - lenguajes de programación utilizados en cada caso.

Cliente: programas escritos en un lenguaje que interpreta el cliente web (navegador). Se pueden escribir tanto en ficheros aparte como incrustados en el fichero HTML.

Servidor: programas que generan respuesta. (Programas PHP, ASP, Servlet, JSP...).

EJERCICIO 9

9. Lenguajes de programación utilizados en el lado servidor de una aplicación web (características y grado de implantación actual).

ASP.net: tecnología de Microsoft destinada al servidor. Proporciona páginas web dinámicas sobre ISS (Internet Information Services) -> Servidor Windows.

JSP: las páginas JSP contienen código Java embebido en lenguaje HTML y se almacenan de forma individual en el servidor web.

PERL: Perl es un lenguaje de propósito general originalmente desarrollado para la manipulación de texto y que ahora es utilizado para un amplio rango de tareas incluyendo administración de sistemas, desarrollo web, programación en red, desarrollo de GUI y más.

PHP: Lenguaje de parte del servidor usado para el diseño del desarrollo web dinámico. Alto nivel de uso.

- Capacidad de conexión con bases de datos.
- Capaz de extender su potencial usando frameworks

JAVA: Es otra manera de incluir código a ejecutar en los clientes que visualizan una página web.

EJERCICIO 10

10. Características y posibilidades de desarrollo de una plataforma XAMPP.

Es de licencia pública.

Es multiplataforma.

Solamente requiere la descarga y ejecución.

Xampp es una herramienta muy práctica que nos permite instalar el entorno MySQL, Apache y PHP, suficiente para empezar proyectos web o revisar alguna aplicación localmente. Además, trae otros servicios como servidor de correos y servidor FTP.

EJERCICIO 11

11. En que casos es necesaria la instalación de la máquina virtual Java (JVM) y el software JDK en el entorno de desarrollo y en el entorno de explotación.

Cuando se realiza la programación en Java, es necesario la instalación de la máquina virtual de Java, tanto en el entorno de desarrollo como en el de explotación.

En cambio, en el entorno de desarrollo hay que instalar el JDD, ya que es lo que provee al sistema de herramientas de desarrollo. Pero en el entorno de explotación, tendremos que instalar el JRE, que es la herramienta necesaria para la ejecución de programas en Java.

EJERCICIO 12

12. IDE más utilizados (características y grado de implantación actual).

Visual Estudio:

- Botón de barra de herramientas y ampliación de texto
- Opciones de tamaño de texto en los editores
- Personalización de colores en los editores
- Personalización de métodos abreviados de teclado
- Función Autocompletar para métodos y parámetros

Eclipse: dispone de un Editor de texto con un analizador sintáctico. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (wizards) para creación de proyectos, clases, tests, etc., y refactorización.

Así mismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con Subversión. e integración con Hibernate.

Netbeans:

- Gestión de la interfaz de usuario (menús y barras de herramientas)
- Gestión de configuración de usuario
- Gestión de almacenamiento (guardar o cargar algún tpo de dato)
- Gestión de ventana
- Marco Asistente (soporta diálogos paso a paso)
- Librería visual de Netbeans
- Herramientas de desarrollo integrado

Pycharm:

- Asistencia y análisis de codificación, con finalización de código , sintaxis y resaltado de errores, integración de linter y arreglos rápidos
- Navegación de proyectos y códigos: vistas de proyectos especializados, vistas de estructura de archivos y saltos rápidos entre archivos, clases, métodos y usos
- Refactoring de Python: incluyendo renombrar, extraer método, introducir variable, introducir constante, tirar hacia arriba, empujar hacia abajo y otros
- Soporte para frameworks web: Django , web2py y Flask
- Depurador integrado de Python
- Prueba unitaria integrada, con cobertura de código línea por línea
- Desarrollo de Google App Engine Python
- Integración de control de versiones: interfaz de usuario unificada para Mercurial, Git, Subversión , Perforce y CVS con listas de cambios y combinación

Android Studio:

- Integración de ProGuard y funciones de firma de aplicaciones.
- Renderizado en tiempo real
- Consola de desarrollador: consejos de optimización, ayuda para la traducción, estadísticas de uso.
- Soporte para construcción basada en Gradle.
- Refactorización específica de Android y arreglos rápidos.
- Un editor de diseño enriquecido que permite a los usuarios arrastrar y soltar componentes de la interfaz de usuario
- Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versiones, y otros problemas.
- Plantillas para crear diseños comunes de Android y otros componentes.
- Soporte para programar aplicaciones para Android Wear.
- Soporte integrado para Google Cloud Platore, que permite la integración con Google Cloud Messaging y App Engine.
- Un dispositivo virtual de Android que se utliia para ejecutar y probar aplicaciones

EJERCICIO 13

13. Servidores HTTP/HTTPS más utilizados (características y grado de implantación actual).

APACHE

Características

- Servidor web de código abierto.
- Permite emplear diversos lenguajes en el lado del servidor.
- Permite conexiones seguras
- Permite el uso de bases de datos de autenticación

NGINX

Características

- Es software libre y de código abierto.
- Es multiplataforma
- Es ligero

Microsoft IIS

Características

- Proporciona un único almacén de certificados SSL para una granja de servidores y simplifica la administración de enlaces SSL.
- Permite a los administradores configurar IIS 8 para bloquear el acceso para las direcciones IP que superan la cantidad especificada de solicitudes y especificar el comportamiento cuando se bloquea una dirección IP.
- Restringe la cantidad de intentos de inicio de sesión erróneos que pueden realizarse en una cuenta FTP durante un período especificado.
- Extiende los protocolos SSL y TLS para permitir que se use un nombre de dominio virtual o un nombre de host para identificar el extremo de la red.
- Permite a los administradores web configurar IIS 8 para inicializar aplicaciones web de modo que la aplicación esté lista para la primera solicitud.
- Proporciona compatibilidad con el hardware NUMA, que permite entre 32 y 128 núcleos de CPU. Esta compatibilidad proporciona un rendimiento inicial casi óptimo en el hardware NUMA.
- Limita el consumo de CPU, memoria y ancho de banda mediante un único grupo de aplicaciones en una implementación multiempresa. IIS 8 incluye opciones de limitación adicionales.

Google Web Server

Características

- Servidor personalizado de desarrollo propio que se ejecuta en sistemas UNIX

- Es muy rápido
- Para jerarquizar sus páginas utiliza diversos factores como modelo vectorial, texto de anchors...

Lighttpd

Características

- Es ligero
- Utiliza menos cantidad de memoria y CPU
- Virtual hosting (alojar varios dominios en la misma IP)
- Soporte para PHP, Ruby, Python y otros
- Redirecciones HTTP, y reescrituras de URL



EJERCICIO 14

14. Navegadores HTTP/HTTPS más utilizados (características y grado de implantación actual).

Google Chrome

Características

- Seguridad: Ofrece seguridad contra software malicioso, “phishing”, entre otras.
- Velocidad: El objetivo principal de Chrome es la velocidad de navegación, desde su ejecución hasta la carga de aplicaciones web complejas.

- Productividad: Puedes acceder a tus marcadores, pestañas abiertas e historial desde cualquier dispositivo que sea compatible con Chrome.
- Compatibilidad: Siempre se encuentra en constante crecimiento por parte de sus desarrolladores y desarrolladores externos, esto permite la inclusión de nuevas extensiones para mejorar la compatibilidad con otras aplicaciones.
- Simplicidad: Está diseñado para ser lo más sencillo posible, esto te permite centrarte en lo que necesites hacer.
- Administrador de tareas: Chrome contiene un administrador de tareas que nos indica qué recursos se están consumiendo y en qué páginas

Mozilla Firefox

Características

- Tiempos de inicio más rápidos, aceleración de gráficos renderizados y mejoras en la velocidad de carga de las páginas.
- Firefox proporciona una navegación ininterrumpida cuando hay un fallo de los plugins de Adobe Flash, Apple QuickTime o Microsoft Silverlight.
- Explora muchos sitios web al mismo tiempo de una forma fácil y sencilla

Opera

Características

- Gestión de pestañas: ofrece múltiples opciones para visualizar las web y navegarlas. Cascada, horizontal y vertical son algunas de ellas. Mosaico vertical es una de las que se destaca.
- Configuración: múltiples opciones de configuración lo diferencian de sus pares gracias a su grado de detalles. Si bien son prácticamente las mismas que en otros navegadores van un poco más allá en detalle como aclaré.
- Fijar pestañas: reduce el tamaño del favicon y te permite abrir muchas más pestañas en la barra y hacer más cómoda la navegación.
- Opera Turbo: permite reducir la calidad de las imágenes de los sitios para navegar más rápido. La función se puede programar para que funcione todo el tiempo o cuando la carga es lenta.
- Sus extensiones: si bien no son tantas como en Firefox o Chrome son excelentes y funcionan a la perfección

Safari

Características

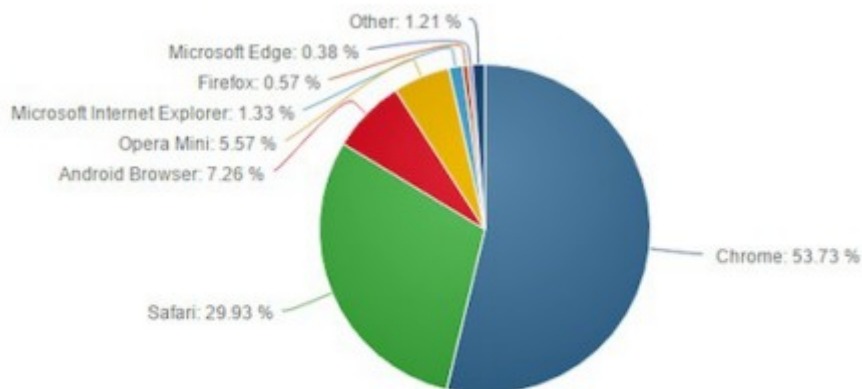
- Desarrollado por Apple Inc.

- Panel de línea de tiempo, que ofrece información acerca de todo lo que el navegador está realizando mientras navega.
- Permite ver todas las redirecciones HTTP, junto a la información completa del encabezado, incluyendo el código de estado HTTP.

Microsoft Edge

Características

- Modo de lectura (Reading View): Este modo simplifica por completo la vista de la página Web, eliminando las gráficas y diseño excesivos y deja sólo el texto con las imágenes.
- Escritura en las páginas Web: Esta función te permite “congelar” una página Web para que puedas escribir encima de ella, subrayar contenido, escribir comentarios o recortar parte de la página Web.
- Nueva pestaña/ventana: Las nuevas pestañas y ventanas traen un nuevo diseño que incluye las páginas Web que el usuario más visita, recomendaciones de apps, información del clima y noticias sugeridas.
- Extensiones: Similar a lo que ofrece Chrome y Firefox, Microsoft Edge será compatible con extensiones que permitirán añadir nuevas funciones al navegador



EJERCICIO 15

15. Generadores de documentación HTML (PHPDoc): PHPDocumentor, ApiGen,

PhpDocumentor:

Es una herramienta que hace posible generar documentación directamente desde su código fuente PHP. Con esto usted puede proporcionar a sus consumidores con más información con respecto a la funcionalidad integrada en su fuente y no sólo lo que se puede utilizar para ellos desde su interfaz de usuario. La documentación generada por phpDocuementor 2 no pretende ser un sustituto de la documentación convencional, sino más bien una documentación complementaria o de referencia.

PhpDoc:

Es una adopción de Javadoc para el mundo de PHP. PHPDoc está escrito en PHP. Le ofrece una manera de generar una documentación de API de código orientado a objetos y de procedimiento con cierto marcado en su fuente. PHPDoc es un proyecto de código abierto y se distribuye bajo la licencia de PHP. Eso significa que puedes usarlo en proyectos comerciales.

Oxygenado:

Es la herramienta estándar de facto para generar documentación a partir de fuentes C++ anotadas, pero también soporta otros lenguajes de programación populares como C, Objective-C, C#, PHP, Java, Python, IDL (Corba, Microsoft y UNO / OpenOffice favors), Fortran, VHDL, Tcl, y hasta cierto punto D.

ApiGen:

Es una distribución del repositorio en la nube de GitHub.

EJERCICIO 16

16. Repositorios de software – sistemas de control de versiones para aplicaciones web para aplicaciones sobre los Web: GIT, CVS, Subversion, ...

Git:

Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end.

CVS:

Es una aplicación informática que implementa un sistema de control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros (código fuente principalmente) que forman un proyecto (de programa) y permite que distintos desarrolladores (potencialmente situados a gran distancia) colaboren. CVS se ha hecho popular en el mundo del software libre. Sus desarrolladores difunden el sistema bajo la licencia GPL.

Subversión:

Apache Subversión (abreviado frecuentemente como SVN, por el comando `svn`) es una herramienta de control de versiones basada en un repositorio cuyo funcionamiento se asemeja enormemente al de un sistema de ficheros. Es software libre bajo una licencia de tpo Apache/BSD.

EJERCICIO 17

17. Propuesta de configuración del entorno de desarrollo para la asignatura de Desarrollo web del lado servidor en este curso (incluyendo las versiones): xxx-USED y xxx-W7ED.

Maquina LRA-USED		
Sistema Operativo	Ubuntu Server 16.04.03	
Memoria	512 MB	
Usuario y Contraseña	miadmin	paso
	operadorweb2	paso
Configuración de red	Ip	192.168.3.108
	Netmask	255.255.255.0
	Gateway	192.168.3.1
	Dns	192.168.20.20
Software	Servidor Web	Apache (con modulo PHP)
	Servidor SSH	Openssh-server
	Servicio SFTP	SFTP
	SGBD	Mysql-server

Maquina: LRA-W7ED		
Sistema Operativo	Windows 7 profesional	
Memoria	1024MB	
Usuario y contraseña	admin	admin
	usuario	paso
Configuración de red	Ip	192.168.3.128
	Netmask	255.255.255.0
	Gateway	192.168.3.1
	DNS	192.168.20.20
Software	Navegador	Chroome
	IDE	Netbeans
	Ofimatica	Libre Office
	Editor texto	Notepad++
	Otros	Filezilla, putty,

EJERCICIO 18

18. Propuesta de configuración del entorno de explotación para la asignatura de Desarrollo web del lado servidor en este curso (incluyendo las versiones): xxx-USEE.

Maquina LRA-USEE		
Sistema Operativo	Ubuntu Server 16.04.03	
Memoria	512 MB	
Usuario y Contraseña	miadmin	paso
	operadorweb2	paso
Configuración de red	Ip	192.168.3.208
	Netmask	255.255.255.0
	Gateway	192.168.3.1
	Dns	192.168.20.20
Software	Servidor Web	Apache (con modulo PHP)
	Servidor SSH	Openssh-server
	Servicio SFTP	SFTP
	SGBD	Mysql-server