

Ejercicios Tema 1

1. Protocolos de comunicaciones: IP, TCP, HTTP, HTTPS.....	2
2. Modelo de comunicaciones cliente – servidor y su relación con las aplicaciones web.....	2
3. Estudio sobre los métodos de petición HTTP /HTTPS más utilizados.....	3
4. Modelo de desarrollo de aplicaciones multicapa – comunicación entre capas – componentes – funcionalidad de cada capa.....	3
5. Modelo de división funcional front-end / back-end para aplicaciones web.....	4
6. Página web estática – página web dinámica – aplicación web.....	4
7. Componentes de una aplicación web.....	5
8. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor - lenguajes de programación utilizados en cada caso.....	5
9. Lenguajes de programación utilizados en el lado servidor de una aplicación web (características y grado de implantación actual).....	5
10. Características y posibilidades de desarrollo de una plataforma XAMPP.....	6
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.....	6
12. IDE más utilizados (características y grado de implantación actual).....	7
13. Servidores HTTP /HTTPS más utilizados (características y grado de implantación actual).....	7
14. Navegadores HTTP /HTTPS más utilizados (características y grado de implantación actual).....	8
15. Generadores de documentación HTML (PHPDoc): PHPDocumentor, ApiGen,8	8
16. Repositorios de software – sistemas de control de versiones para aplicaciones web para aplicaciones sobre los Web: GIT , CVS, Subversion,8	8
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.....9	9
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.....10	10

Ejercicios Tema 1

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

IP

El protocolo de Internet es un protocolo de comunicación de datos digitales, clasificado en la capa de red según el modelo internacional OSI.

TCP

Protocolo de Control de Transmisión, es uno de los protocolos fundamentales en Internet. El Protocolo garantiza que los datos serán entregados a su destino sin errores y en el mismo orden en que se transmitieron

HTTP

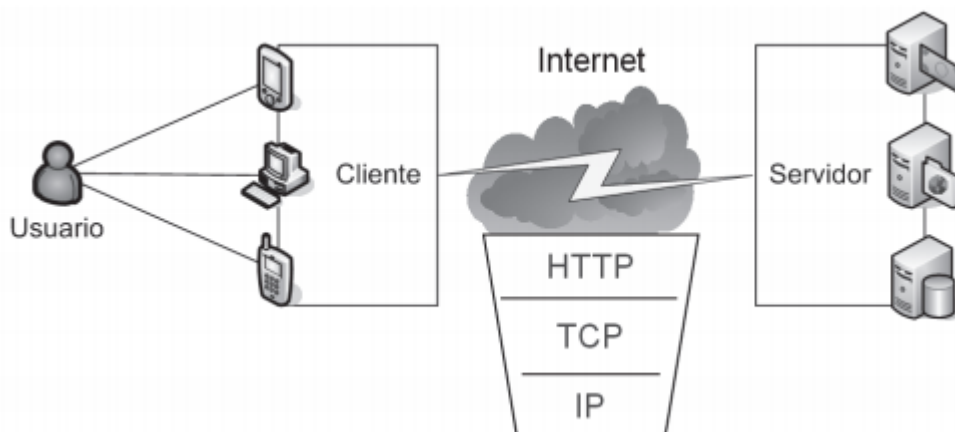
EL Protocolo de transferencia de hipertexto, es el protocolo de comunicación que permite las transferencias de información en la Word Wide Web.

HTTPS

Es un protocolo destinado a la transferencia segura de datos de hipertexto.

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

El modelo mas usado es el cliente-servidor, este modelo consiste en un cliente que es un componente consumidor de servicios y el servidor es un proceso proveedor de servicios. El agente que solicita la información se denomina cliente, mientras que el componente software que responde a esa solicitud es el que se conoce como servidor.



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

HTTP define una serie predefinida de métodos de petición que pueden utilizarse

GET

Pide una representación del recurso especificado. Por seguridad no debería ser usado por aplicaciones que causen efectos ya que se transmite a través de la URI ejemplo:

GET /images/logo.png HTTP/1.1 obtiene un recurso llamado logo.png

Ejemplo con parámetros:

/index.php?page=main&lang=es

POST

Envía los datos para que sean procesados por el recurso identificado. Los datos se incluirán en el cuerpo de la petición. Esto puede resultar en la creación de un nuevo recurso o de las actualizaciones de los recursos existentes o ambas cosas.

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

Este modelo se divide en 3 tipos de capa:

Capa de presentación:

Esta capa es la que ve el usuario. Le presenta una interfaz gráfica del recurso solicitado y sirve para recoger su interacción. Suele estar situada en el cliente

Capa de negocio:

Es la capa que conoce y gestiona las funcionalidades que esperamos del sistema o aplicación. Es donde se reciben las peticiones del usuario y desde donde se envían las respuestas apropiadas tras el procesamiento de la información proporcionada por el cliente. Esta capa puede estar tanto en el lado del cliente como en el del servidor

Capa de persistencia o de datos:

Es la capa donde residen los datos y la encargada de acceder a los mismo. Esta formada por uno o más gestores de bases de datos que realizan todo el proceso de administración de datos. Reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

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

Una aplicación Web esta dividida en 2 partes bien diferenciadas:

Parte externa o front-end, es el conjunto de páginas que ven la gran mayoría de usuarios que las usan (usuarios externos).

Parte interna o back-end, es otro conjunto de páginas dinámicas que utilizan las personas que producen el contenido y las que administran la aplicación web (usuarios internos) para crear contenido, organizarlo, decidir la apariencia externa.

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

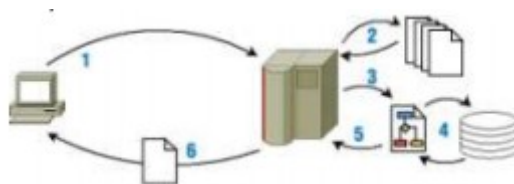
Página web estática

Estas páginas se encuentran almacenadas en su forma definitiva, tal y como se crearon, y su contenido no varía. Son utiles para mostrar información concreta y siempre la misma. La única forma en que pueden cambiar es si un programador la modifica y actualiza su contenido.

Página web dinámicas

Hay 2 tipos de ellas. Aquellas que incluyen código que ejecuta el navegador. En estas páginas el código ejecutable, es JavaScript, se incluye dentro del HTML. SE descarga con la página. Este código puede incorporar múltiples funcionalidades que pueden ir desde mostrar animaciones hasta cambiar totalmente la apariencia y el contenido de la página.

El otro tipo contenido se descarga al navegador similar a una pagina web estática. Lo que cambia es la forma a que se obtiene ese contenido. El HTML de estas páginas se forma como resultado de la ejecución de un programa, y esa ejecución tiene lugar en el servidor web.



Pasos de una ejecución de una pagina dinamica:

1. El cliente web (navegador) de tu ordenador solicita a un servidor web una página web.
2. El servidor busca esa página y la recupera.
3. En el caso de que se trate de una página web dinámica, es decir, que su contenido deba ejecutarse para obtener el HTML que se devolverá, el servidor web contacta con el módulo responsable de ejecutar el código y se lo envía.

4. Como parte del proceso de ejecución, puede ser necesario obtener información de algún repositorio
(Cualquier almacén de información digital, normalmente una base de datos), como por ejemplo consultar registros almacenados en una base de datos.
5. El resultado de la ejecución será una página en formato HTML, similar a cualquier otra página web no dinámica.
6. El servidor web envía el resultado obtenido al navegador, que la procesa y muestra en pantalla.

7. Componentes de una aplicación web.

Todas las aplicaciones de base de datos basadas en la web tienen tres componentes primarios: un navegador web (o cliente), un servidor de aplicaciones web y un servidor de bases de datos.

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

Los lenguajes de programación del lado cliente se usan para su integración en páginas web.

Un código escrito en un lenguaje de script se incorpora directamente dentro de un código HTML y se ejecuta interpretado, no compilado.

Los dos lenguajes mas usados en la actualidad son VBScript (derivado de Visual Basic) y el JavaScript (derivado de Java).

Los lenguajes de programación del lado del servidor son aquellos cuyo código, bien sea como objeto precompilado o bien como código interpretado.

Los lenguajes mas usados son PHP, ASP, Perl, Python y JSP

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

PHP (Hypertext Processor): es uno de los lenguajes más extendidos actualmente. Cuenta con una comunidad de desarrolladores muy importante debido a sus características de gratuidad, código abierto, la posibilidad de ser portado y ejecutado en diferentes plataformas, etc. Se define como un lenguaje imperativo de tipos dinámicos, con la posibilidad de utilizar construcciones orientadas a objetos. Es soportado por la gran mayoría de servidores web actuales.

ASP (Active Server Pages): se trata de una tecnología propietaria y de código cerrado para el lado del servidor de Microsoft. Su última versión es la 3.0 y data del año 2002, año a partir del cual se empezó a sustituir progresivamente por la versión ASP.Net. Aunque puede ser ejecutado en otros servidores web, ASP está diseñado especialmente para ser utilizado con IIS (Internet Information Server). El lenguaje utilizado con ASP es Visual Basic en su versión de scripting (VBScript), aunque también pueden usarse otros (JScript).

Perl: fue inicialmente concebido como un lenguaje de manipulación de cadenas de caracteres basado en un estilo de programación por bloques, como C o AWK. Perl es un lenguaje imperativo, con variables, expresiones, asignaciones, bloques de código delimitados por llaves, estructuras de control y subrutinas actualmente orientado a la generación dinámica de páginas web. Fue uno de los primeros lenguajes en ser utilizados para la programación web (para crear aplicaciones CGI). Es de código abierto y cuenta con una comunidad bastante numerosa de seguidores.

Python: es un lenguaje de scripting independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programas, desde aplicaciones Windows a servidores de red, o incluso páginas web. Es un lenguaje interpretado, lo que significa que no necesitamos compilar el código fuente para poder ejecutarlo.

Esto ofrece ventajas, como la rapidez de desarrollo, e inconvenientes, como una menor velocidad. En los últimos años el lenguaje se ha hecho muy popular gracias a la cantidad de librerías, tipos de datos y funciones incorporadas en el propio lenguaje o la sencillez y velocidad con la que se crean los programas. Además, Python es un lenguaje gratuito, es decir, no necesita de ningún software de pago adicional para su ejecución incluso para propósitos empresariales.

JSP (Java Server Pages): el funcionamiento de Java como lenguaje de script es similar al de PHP o ASP: se trata de porciones de código Java intercalado con código HTML estático. Sin embargo, la forma de interpretarlo

es diferente. Una vez que el módulo encargado de ejecutar la página JSP llega a una porción de código Java, lo transforma a un Servlet (porción de código Java cargado en la memoria de un servidor web) y lo ejecuta obteniendo el código que ha de enviarse al cliente. La diferencia con otros lenguajes es que el servlet queda cargado en memoria por si llegara otra petición a la misma página JSP, con lo que el rendimiento se ve mejorado notablemente.

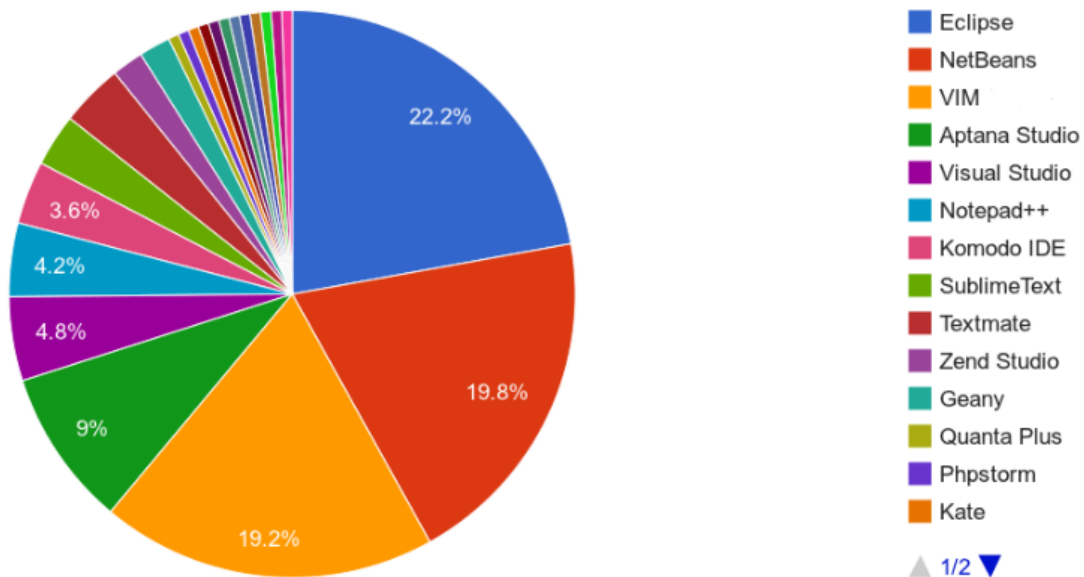
10. Características y posibilidades de desarrollo de una plataforma XAMPP

XAMPP es un paquete de instalación independiente de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL (A partir de la versión 5.6.15, XAMPP cambió la base de datos MySQL por MariaDB).

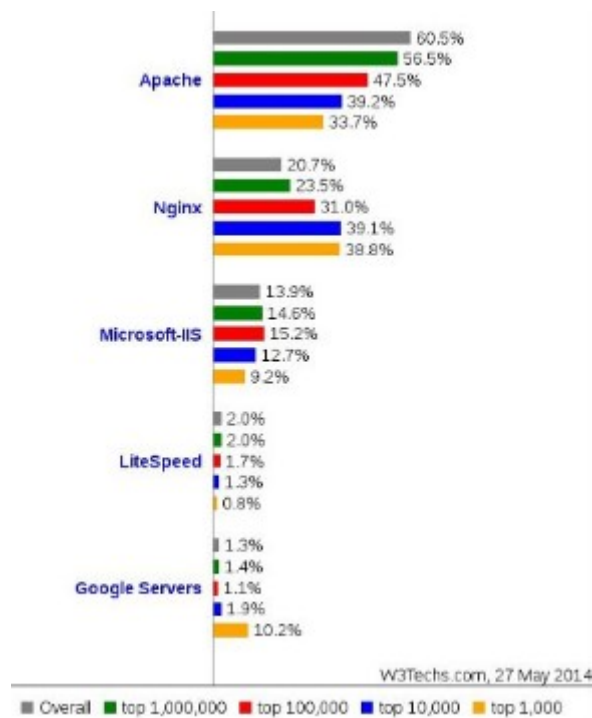
11. En que casos es necesaria la instalación de la máquina virtual Java (JVM) y el software JDK en el entorno de desarrollo y en el entrono de explotación.

La JVM es necesaria para poder utilizar las aplicaciones en java.
El JDK es el paquete para el desarrollo de código JavaScript

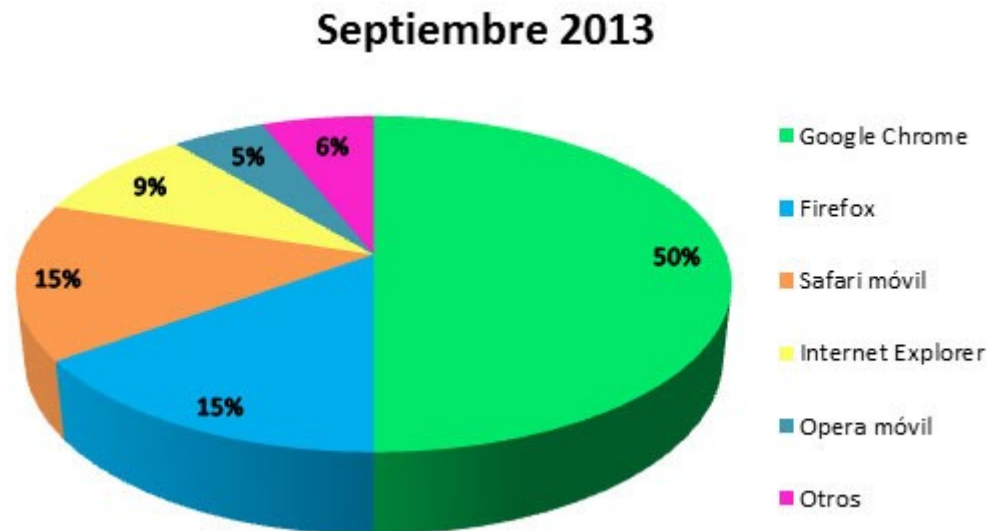
12. IDE más utilizados (características y grado de implantación actual).



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



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



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

Yo he usado [JavaDoc](#)

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

Los repositorios que he usado son [GitHub](#) y [CVS](#)

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.

Servidor de Entorno de Desarrollo

<u>S.O:</u>	Ubuntu Server 16.04lts
<u>DiscoDuro:</u>	80GB
<u>Nombre de la Maquina:</u>	RGO-USED
<u>Usuarios:</u>	root/Aaa111!!! alumno/Aaa111!!! operador/Aaa111!!!
<u>Configuración de Red:</u>	address 192.168.3.104 gateway 192.168.3.1 dns-nameservers 8.8.8.8
<u>Servicios:</u>	Servidor Web Apache2 + PHP Servidor BBDD Mysql Las conexiones las realizo atraves de OpenSSH

Cliente de Entorno de Desarrollo

<u>S.O:</u>	Windows 7 Professional 32Bits
<u>Disco Duro:</u>	100GB
<u>Nombre de la Maquina:</u>	RGO-WCED
<u>Usuarios:</u>	admin/Aaa111!!! alumno/Aaa111!!! operador/Aaa111!!!
<u>Configuración de Red:</u>	address 192.168.3.54 gateway 192.168.3.1 dns-nameservers 8.8.8.8
<u>Programas:</u>	IDE NetBeans Navegador Internet Explorer , Mozilla , Chrome , Opera , Safari . Cliente FTP FileZilla Cliente Conexión Remota Putty Editor de Texto notepad++

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.

Servidor de Entorno de Explotación

<u>S.O:</u>	Ubuntu Server 16.04lts
<u>DiscoDuro:</u>	80GB
<u>Nombre de la Maquina:</u>	RGO-USEE
<u>Usuarios:</u>	root/Aaa111!!! alumno/Aaa111!!! operador/Aaa111!!!
<u>Configuración de Red:</u>	address 192.168.3.204 gateway 192.168.3.1 dns-nameservers 8.8.8.8
<u>Servicios:</u>	Servidor Web Apache2 + PHP Servidor BBDD Mysql Las conexiones las realizo atraves de OpenSSH