

TEMA 1- DESARROLLO WEB EN ENTORNO SERVIDOR

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.....	2
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.....	3
6. Página web estática – página web dinámica – aplicación web – mashup	3
7. Componentes de una aplicación web.....	4
8. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor - lenguajes de programación utilizados en cada caso.....	4
9. Lenguajes de programación utilizados en el lado servidor de una aplicación web (características y grado de implantación actual).....	4
10. Características y posibilidades de desarrollo de una plataforma XAMPP.....	4
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.....	5
12. IDE más utilizados (características y grado de implantación actual).....	5
13. Servidores HTTP /HTTPS más utilizados (características y grado de implantación actual).....	6
14. Apache HTTP vs Apache Tomcat.....	6
15. Navegadores HTTP /HTTPS más utilizados (características y grado de implantación actual)....	6
16. Generadores de documentación HTML (PHPDoc): PHPDocumentor, ApiGen,	8
17. Repositorios de software – sistemas de control de versiones: GIT , CVS, Subversion,	8
18. 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
19. Propuesta de configuración del entorno de explotación para la asignatura de Desarrollo web del lado servidor en este curso (incluyendo las versiones): xxx-USEE.....	9
20. Elegir y realizar un estudio y una presentación para la exposición del trabajo sobre una de las siguientes arquitecturas de desarrollo de Aplicaciones Web: • MEAN (con MongoDB y con MySQL) • Java EE vs Spring • Microsoft .NET • Angular 7 • Symfony • Laravel • CakePHP • CodeIgniter.....	9

EJERCICIOS

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

IP: Su función principal es el uso bidireccional en origen o destino de comunicación para transmitir datos mediante un protocolo no orientado a conexión que transfiere paquetes conmutados. La entrega de los paquetes de datos es no confiable.

TCP: Tiene la misma función que el protocolo IP con la diferencia de que con el uso de protocolo TCP, las aplicaciones pueden comunicarse en forma segura.

HTTP: Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. Permite las transferencias de información en la World Wide Web. HTTP es inseguro y está sujeto a ataques.

HTTPS: El sistema HTTPS utiliza un cifrado basado en la seguridad de textos SSL/TLS para crear un canal cifrado más apropiado para el tráfico de información sensible que el protocolo HTTP.

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

Un servidor es una aplicación que ofrece un servicio a usuarios de Internet; un cliente es el que pide ese servicio. Una aplicación consta de una parte de servidor y una de cliente, que se pueden ejecutar en el mismo o en diferentes sistemas.

Los usuarios invocan la parte cliente de la aplicación, que construye una solicitud para ese servicio y se la envía al servidor de la aplicación. El servidor es un programa que recibe una solicitud, realiza el servicio requerido y devuelve los resultados en forma de una respuesta.

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

GET: El método GET solicita una representación de un recurso específico. Las peticiones que usan el método GET sólo deben recuperar datos.

HEAD: El método HEAD pide una respuesta idéntica a la de una petición GET, pero sin el cuerpo de la respuesta.

POST: El método POST se utiliza para enviar una entidad a un recurso en específico, causando a menudo un cambio en el estado o efectos secundarios en el servidor.

PUT: El modo PUT reemplaza todas las representaciones actuales del recurso de destino con la carga útil de la petición.

DELETE: El método DELETE borra un recurso en específico.

CONNECT: El método CONNECT establece un túnel hacia el servidor identificado por el recurso.

OPTIONS: El método OPTIONS es utilizado para describir las opciones de comunicación para el recurso de destino.

TRACE: El método TRACE realiza una prueba de bucle de retorno de mensaje a lo largo de la ruta al recurso de destino.

PATCH: El método PATCH es utilizado para aplicar modificaciones parciales a un recurso.

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

El modelo de desarrollo de aplicaciones multicapa esta formado por tres capas la primera es:

la capa de presentación: La cual consta de lo que ve el usuario, presenta el sistema al usuario al cual le comunica información y captura la información del usuario. Esta capa se comunica con la de negocio.

La capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.

La capa de datos: Es donde residen los datos y es la encargada de acceder a los mismos. 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.

Front-end y back-end son términos que se refieren a la separación de intereses entre una capa de presentación y una capa de acceso a datos, respectivamente.

En diseño de software el front-end es la parte del software que interactúa con los usuarios y el back-end es la parte que procesa la entrada desde el front-end.

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

Página web estática: Son básicamente informativas y están enfocadas principalmente a mostrar una información permanente, donde el usuario se limita a obtener dicha información sin poder interactuar con la página visitada.

Página web dinámica: son aquellas páginas que se genera automáticamente en el momento que alguien solicita su visualización, por tanto, puede cambiar dependiendo de quién lo solicite o en qué momento lo haga.

Aplicación web: Es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.

Mashup: Ocurre cuando una aplicación web es usada o llamada desde otra aplicación, con el fin de reutilizar su contenido o funcionalidad.

7. Componentes de una aplicación web

Los componentes principales son:

Un servidor web para recibir las peticiones de los clientes web (normalmente navegadores) y enviarles la página que solicitan.

El módulo encargado de ejecutar el código o programa y generar la página web resultante. Este módulo debe integrarse de alguna forma con el servidor web.

Una aplicación de base de datos, que normalmente también será un servidor.

El lenguaje de programación que utilizado para desarrollar las aplicaciones.

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

Programa ejecutado en el lado del cliente: Es el ámbito en el cual un cliente ejecuta sus operaciones en una relación cliente-servidor dentro de una red informática. Lenguaje JavaScript, Java, Flash, VBScript.

Programa ejecutado del lado del servidor: Es una tecnología que consiste en el procesamiento de una petición de un usuario mediante la interpretación de un script en el servidor web para generar páginas HTML dinámicamente como respuesta. Lenguajes utilizados ASP, PHP, JSP, Python, Ruby...

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

ASP: Este modelo tecnológico es poder utilizar diversos componentes ya desarrollados como algunos controles ActiveX así como componentes del lado del servidor, tales como CDONTS.

PHP : Puede ser desplegado en la mayoría de los servidores web y en 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.

JSP: La principal ventaja de JSP frente a otros lenguajes es que el lenguaje Java es un lenguaje de propósito general que excede el mundo web y que es apto para crear clases que manejen lógica de negocio y acceso a datos de una manera prolija. Es posible ejecutar las aplicaciones en múltiples plataformas sin cambios.

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

Es una distribución de Apache sencilla y ligera que facilita enormemente a los desarrolladores crear un servidor web local para realizar pruebas.

XAMPP tiene cuatro componentes principales. Éstos son:

Apache: Apache es la aplicación de servidor web que procesa y entrega el contenido web a un ordenador.

MySQL: Gestor de bases de datos.

PHP: Lenguaje de Script del lado del servidor.

Perl: lenguaje de programación dinámico y de alto nivel que se usa ampliamente en la programación de redes, la administración de sistemas, etc.

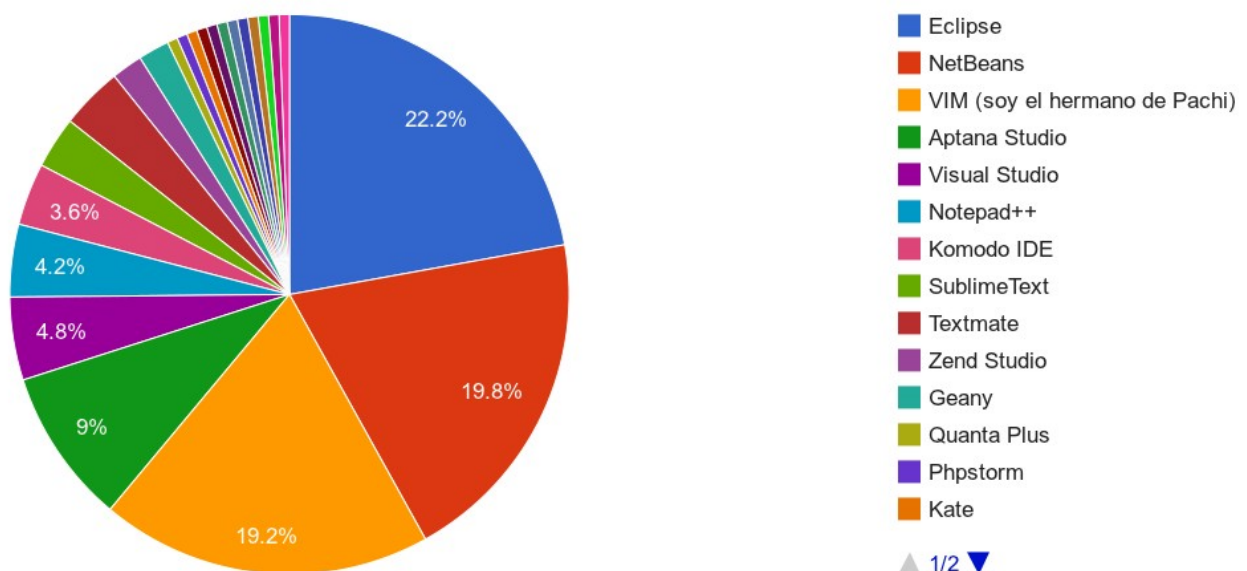
Ventajas: Xampp es una herramienta muy práctica que nos permite instalar el entorno MySQL, Apache y PHP, suficiente para empezar proyectos web o revisar alguna aplicación localmente.

Desventajas: No soporta MySQL desde la consola. Xampp trae PhpMyAdmin para administrar las bases de datos de MySQL. Xampp trae las últimas versiones de las aplicaciones que instala, sin embargo cuando pasa el tiempo y salen nuevas versiones de las mismas, no queda otra salida que reinstalar todo Xampp.

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.

En el entorno de desarrollo se instalarían JVM y JDK para poder desarrollar el programa en java y poder ejecutarlo, mientras que en el entorno de explotación solo necesitaríamos la maquina virtual de java para ejecutar el programa y realizar las pruebas.

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



Netbeans también es un entorno de programación muy utilizado por los programadores. Se trata de otro entorno multilenguaje y multiplataforma en el cual podemos desarrollar

software de calidad. Con él podemos crear aplicaciones web y de escritorio, además de contar con plugins para trabajar en Android.

Visual Studio fue diseñado por Microsoft y es uno de los mejores entornos de programación que existe siempre y cuando utilices sus lenguajes. Antiguamente tenían una versión de pago que incluía todos los lenguajes, y versiones express que eran gratuitas para un lenguaje en concreto.

JetBrain no es un entorno concreto, es una compañía que crea entornos de programación, es libre y crean entornos para multitud de lenguajes como son Java, Ruby, Python, PHP, SQL, Objective-C, C++ y JavaScript. También están desarrollando IDE's para C# y GO

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

Apache: Entre sus ventajas encontramos que es código abierto, es además software gratuito, y multiplataforma (Windows, Linux y Unix). El stack LAMP (Linux, Apache, MySQL y PHP) lo popularizó muchísimo durante el auge de las aplicaciones hechas en PHP desde el año 2000 en adelante.

Nginx: Se le conoce como Nginx (pronunciado en inglés como «engine ex»), es un servidor web de open source y gratuito (aunque también existe una versión comercial) que se destaca por su alto rendimiento. Incluye además funciones como servidor proxy reverso HTTP, balanceador de carga, así como POP3 y IMAP. Está disponible para Windows, Linux y Unix.

Micrisoft IIS: Internet Information Services, también conocido como IIS, es un tipo de servidor web creado por Microsoft específicamente para su plataforma de sistemas operativos Windows. Tuvo su origen en el viejo «Option Pack» que corría en Windows NT, pero luego dada su creciente popularidad se integraría con Windows Server 2003, Windows Server 2008 y en posteriores ediciones.

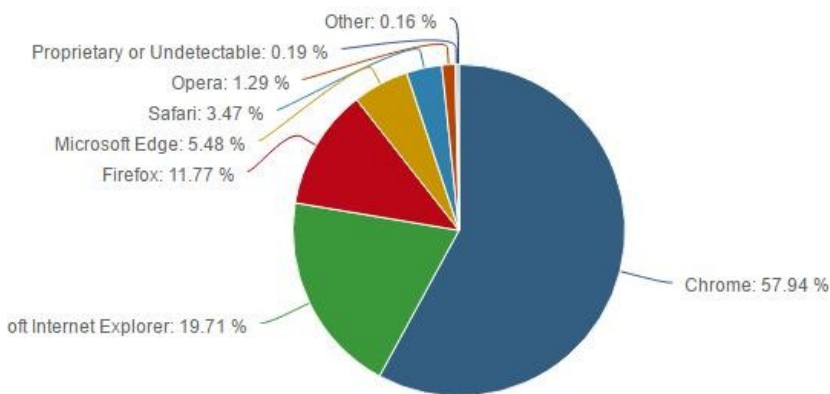
14. Apache HTTP vs Apache Tomcat

Tomcat es un contenedor de servlets. Un servlet, al final, es una clase de Java. Los archivos JSP (que son similares a PHP y los archivos ASP más antiguos) se generan en el código Java (HttpServlet), que luego el servidor compila en archivos .class y ejecuta la máquina virtual Java.

Básicamente, Apache es un servidor HTTP que sirve HTTP. Tomcat es un Servlet y Servidor JSP que sirve tecnologías Java.

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

	Versión instalada	Versión actual	Motor de renderización	Interprete de JavaScript	Empresa	Acceso a los elementos
FireFox	69.0	69.0	Gecko	SpiderMonkey	Mozilla	F12
IE	11.0	11.95	Trident	Chakra	Microsoft	F12
Opera	63.0	63.0	Blink	V8	Opera Software	Ctrl+Alt
Safari		12.2	WebKit	JS Core	Apple	Opc. desarrollo
Chrome	76	77	Blink	V8	Google	F12
Vivaldi	2.7	2.7	Blink	V8	Vivaldi Tec.	F12
Midori	0.5.11	0.5.11	WebKit	Midori	Christian Dywan	F12
Aurora	1.0.0	1.0.0	WebKit	JS Core	Throllthech	F12
Chromium	79	79	WebKit	V8	Chromium Project	F12
Orca	1.2.6	2	Gecko	SpiderMonkey	Mozilla	F12
Spider Monkey	2.49.5	2.49.5	Gecko	SpiderMonkey	SeaMonkey council	F12
Epic	71.0.3578.98	71.0.3578.98	Blink	Chrome V8	Hidden Reflex	F12
Iron	75.0.39	75.0.39	webkit	v8	Software libre	F12
flock	2.6.1	3.6.4	WebKit	Chromium	Flock Inc	Btn drch+View page source
Kmeleon	75.1		Gecko		Mozilla	F12
Maxthon	5.2.7.5000	5.2.7.6000	Blink	V8	Maxthon International	F12
Avant	Build2	Build2	Gecko, Trident, Webkit		Avant Force	Boton derecho, inspeccionar
Falkon	3.1.0	3.1.0	QtWebEngine		KDE	



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

Un generador de documentación es una herramienta de programación que genera documentación destinada a los programadores.

PHPDoc: PHPDoc es una adaptación de javadoc para php que define un estándar oficial para comentar código php.

PHPxRef: con una característica muy interesante: linkea toda llamada de un método, función, variable o constante, hacia el lugar del código donde está definido (aunque sea en otro archivo). Complementa a la perfección a Doxygen y como éste, son aplicaciones que se instalan en la PC (si usan windows, aunque también están disponibles para linux) y pueden utilizarse sin problemas, y sin tener PHP instalado.

17. Repositorios de software – sistemas de control de versiones: GIT , CVS, Subversion, ...

Repositorio de software: una colección de software ordenado, clasificado y disponible para su uso con herramientas compatibles que lo usen para descargar software y manipular dicho software.

Sistemas de control de versiones: El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo de tal manera que sea posible recuperar versiones específicas más adelante.

Git: 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.

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, que forman un proyecto y permite que distintos desarrolladores colaboren.

18. 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.

MAMP

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

20. Elegir y realizar un estudio y una presentación para la exposición del trabajo sobre una de las siguientes arquitecturas de desarrollo de Aplicaciones Web: • MEAN (con MongoDB y con MySQL) • Java EE vs Spring • Microsoft .NET • Angular 7 • Symfony • Laravel • CakePHP • CodeIgniter