

Ejercicios Tema 1

Cristina
Núñez
Sebastián



Sumario

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

Protocolos de comunicaciones: IP, TCP, HTTP, HTTPS

IP

El Protocolo IP (Internet Protocol) es uno de los pilares básicos de Internet, permite el desarrollo y transporte de paquetes de datos, también forma parte del protocolo TCP/IP.

El Protocolo IP actúa como si fuera una “distribuidora” de datos encapsulados o empaquetados, los cuales viajan por distintas trayectorias o “caminos”.

Los paquetes de información viajan a través de la Red por diferentes trayectos hasta llegar al destino.

TCP

El protocolo TCP es una forma segura de intercambio de datos al requerir de la autorización entre cliente y servidores, o emisor y receptor, antes de producirse la transferencia. Una vez ambas partes hayan autorizado la transmisión, podrá iniciarse el envío y recepción de datos.

Los datos se entregan en el mismo orden en el que se enviaron.

HTTP

El protocolo HTTP es un protocolo cliente-servidor que articula los intercambios de información entre los clientes HTTP y servidores HTTP.

El cliente realiza una petición enviando un mensaje con cierto formato al servidor y el servidor le envía un mensaje de respuesta.

HTTPS

El protocolo HTTPS está basado en el protocolo HTTP destinado a la transferencia segura de datos mediante el cifrado de estos.

Modelo de comunicaciones cliente-servidor y su relación con las aplicaciones web



Una aplicación web utiliza un protocolo HTTP o HTTPS, ambos protocolos son cliente-servidor de forma simétrica.

Estudio sobre los métodos de petición HTTP/HTTPS más utilizados

- **Get:** Solicita al servidor un objeto.
- **Post:** Envía al servidor un documento.
- **Put:** Solicita al servidor que haga accesible el documento enviado en una URI específica.
- **Delete:** Solicita al servidor que elimine un documento.
- **Trace:** Solicita una copia de la petición enviada.
- **Head:** Solicita al servidor el encabezado del recurso.
- **Options:** Solicita los métodos HTTP soportados por el servidor.
- **Connect:** Pregunta si se tiene acceso al servidor.
- **Patch:** Es utilizado para aplicar modificaciones parciales a un recurso.

Modelo de desarrollo de aplicaciones multicapa – comunicaciones entre capas - componentes – funcionalidad de cada capa

Es un modelo cuyo objetivo principal es la separación de las partes que componen un sistema de software o también una arquitectura cliente-servidor.

- **Capa de presentación:** es la capa que ve el usuario. Presenta el sistema al usuario, le comunica la información y captura la información del usuario. (Esta capa solo se comunica con la capa de negocio)
- **Capa de negocio:** lugar 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 el)
- **Capa de datos:** lugar donde residen los datos, es la encargada de acceder a los mismos. Está formada por uno o varios gestores de bases de datos.

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

Front-end: parte de la aplicación que utiliza un usuario normal.

Back-end: parte de la aplicación para usuarios especiales.



Página web estática – página web dinámica – aplicación web - mashup

Página web estática

Es una página web informática, enfocada principalmente a mostrar una información permanente sin utilizar bases de datos ni requerir programación.

Página web dinámica

Es una página web donde la información obtenida es generada a partir de una petición del cliente, de forma que puede cambiar dependiendo del momento en que realice la petición o en quien la realice.

Aplicación web

Una aplicación web es un conjunto de herramientas que pueden utilizar los usuarios accediendo a través de internet mediante un navegador.

Mashup

Es una aplicación web híbrida usada o llamada desde otra aplicación con la finalidad de reutilizar su contenido o funcionalidad.

Componentes de una aplicación web

- Servidor web
- Módulo que interprete y ejecute el código
- Sistema gestor de una base de datos
- Lenguaje de programación

Programas ejecutados en el lado del cliente y servidor – lenguajes de programación utilizados en cada caso

Programas ejecutados en el lado del cliente: Navegador (Chrome, Firefox...).

Programas ejecutados en el lado del servidor: Servidor web (Apache, Apache Tomcat...), gestor de bases de datos (MySQL).

Lenguajes de programación utilizados en el lado del cliente: HTML, JavaScript.

Lenguajes de programación utilizados en el lado del servidor: PHP.

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

PHP

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- El código escrito en PHP no es visible en el navegador web, el servidor se encarga de ejecutar el código y enviar el resultado en forma de HTML.
- Tiene una gran capacidad de conexión con la mayoría de motores de base de datos utilizados en la actualidad.
- Posibilidad de aumentar su potencial utilizando módulos(extensiones).
- Hay gran documentación en su sitio web oficial.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

Java

- Lenguaje orientado a objetos.
- Disponibilidad de un amplio conjunto de bibliotecas.
- Lenguaje interpretado y compilado.

Python

- Es un lenguaje interpretado.
- Es multiplataforma.
- Es de tipado.
- Es orientado a objetos.

Características y posibilidades de desarrollo de una plataforma XAMPP

XAMPP es un paquete el cual instala un sistema de gestión de bases de datos MySQL, el servidor web Apache y los interpretes para lenguajes de script PHP.

Este sistema es multiplataforma, existen versiones para diferentes sistemas operativos.

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

La instalación de la máquina virtual Java es necesaria cuando necesitamos interpretar código Java.

La instalación del software JDK es necesaria para precompilar el código de Java.

IDE más utilizados (características y grado de implantación actual)**Eclipse**

- Dispone de editor de texto con analizador sintáctico.
- Compilación en tiempo real.
- Tiene pruebas unitarias con JUnit, control de versiones, asistentes para la creación de proyectos...

NetBeans

- Simplifica la gestión de grandes proyectos.
- tiene herramientas para el depurado de errores.

- Posee ayuda para optimizar el código.
- Acceso a base de datos.

Visual Studio

- Creación de aplicaciones para la Web.
- Compilar aplicaciones y juegos multiplataforma.
- Conectarse a base de datos.
- Administrar el código fuente y colaborar con otras personas.
- Instalación modular.

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

Apache

- Es gratuito y de código abierto.
- Instalación y configuración sencilla.
- Extensible y adaptable mediante módulos.
- Soporte para lenguajes como PHP.

Nginx

- Servidor de archivos estáticos y dinámicos.
- Soporte de autenticación HTTP.
- Soporta índices y autoindexado.
- Balanceo de carga inteligente.
- Tolerante a fallos.
- Escalabilidad asegurada.
- Compatible con IPv4 e IPv6.

IIS

- Gestión ordenada.
- Soporte para protocolos HTTP, HTTPS...
- Soporta prácticamente cualquier lenguaje como PHP, Java...
- No es multiplataforma.

Apache HTTP vs Apache Tomcat

Apache HTTP

- Está optimizado para manejar contenido web estático y dinámico.
- Carece de capacidad para administrar Servlets Java y JSP.

Apache Tomcat

- Orientado al contenido basado en Java.
- Se encarga del JSP.
- No es eficiente como servidor HTTP tradicional.

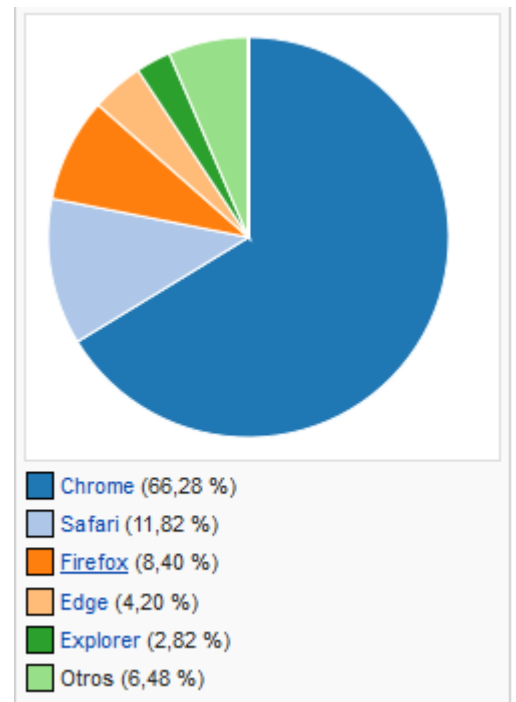
Navegadores HTTP/HTTPS más utilizados (características y grado de implantación)

Chrome

- Ofrece seguridad contra software malicioso.
- Gran velocidad de navegación.
- Puedes acceder a tus marcadores, pestañas abiertas e historial desde cualquier dispositivo.
- Se encuentra en constante crecimiento, lo cual hace posible la inclusión de nuevas extensiones para mejorar la compatibilidad con otras aplicaciones.
- Diseño simple y sencillo.
- Contiene un administrador de tareas que nos indica qué recursos se están consumiendo y en qué páginas.

Safari

- No es de los más rápidos del mercado.
- Soporta extensiones.
- Tus sitios favoritos se pueden anclar en pestañas pequeñas de tal forma que no ocupan espacio.
- Puedes consultar una página web en tu Mac y continuar leyéndola en un iPad o iPhone.
- Sincroniza todo entre dispositivos (iPhone, iPad, Mac).



Firefox

- Es multiplataforma.
- Navegación con pestañas, se pueden abrir varias pestañas simultáneamente.
- Corrector ortográfico.
- Canales RSS, permitiéndote leer titulares de noticias RSS sin tener ningún programa instalado.
- Una barra de búsqueda que integra los motores mas utilizados mundialmente (Google, Yahoo!, ...).
- Bloqueador de ventanas emergentes evitando que se desplieguen.

Generadores de documentación HTML

Un generador de documentación es una herramienta de programación que genera documentación destinada a los programadores o a usuarios finales, o a ambos, a partir de un conjunto de código fuente especialmente documentado, y en algunos casos, archivos binarios.

PHPDoc

Es una adaptación de javadoc para PHP que define un estándar oficial para comentar código PHP.

- Hace comentarios que puedan leerse en un método estándar para que los programadores definan y comenten los aspectos del código.
- Permite que los generadores de documentos externos como phpDocumentor puedan crear la documentación API en buen formato y fácil de entender.
- Permite que algunos IDE interpreten los tipos de variables y otras ambigüedades en el lenguaje de programación.

Repositorios de software – Sistemas de control de versiones

Un repositorio de software es un lugar de almacenamiento del cual pueden ser recuperados e instalados los paquetes de software en un ordenador.

Git

Es un software de control de versiones, fue creado pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran numero de archivos de código

fuelle, Git nos proporciona herramientas para desarrollar un trabajo en equipo de manera inteligente y rápida.

Configuración entorno de desarrollo

Entorno de desarrollo Windows 10:

- Navegador: Firefox 81.0.
- IDE: NetBeans 12.0.
- Servidor Web HTTP: Apache 2.4.
- Ofimática: LibreOffice 6.4.
- Cliente SSH: Filezilla 3.50.0.
- Sistema gestor de bases de datos: MySQL 8.0.
- GitHub.

Entorno de desarrollo Ubuntu Server:

- Servidor Web: Apache 2.4.
- Sistema gestor de bases de datos: MySQL 8.0.
- PHP.
- Servidor administración remota: SSH.
- Servidor de transferencia de ficheros: SFTP(SSH).

Configuración entorno de explotación

Entorno de explotación Ubuntu Server:

- Servidor Web: Apache 2.4.
- Sistema gestor de bases de datos: MySQL 8.0.
- PHP.
- Servidor administración remota: SSH.
- Servidor de transferencia de ficheros: SFTP(SSH).

Estudio y relación con el desarrollo de aplicaciones web CMS y ERP

CMS

Es un programa informático que permite crear un entorno de trabajo para la creación y administración de contenidos, principalmente en páginas web, por parte de los administradores, editores, participantes y demás usuarios, un ejemplo de CMS es WordPress.

ERP

Es un sistema de planificación de recursos empresarial. Se hace cargo de distintas operaciones internas de una empresa, desde producción a distribución o incluso recursos humanos, un ejemplo de ERP es Microsoft Dynamics.