

2020

# Tema 1 - Ejercicios

DWES

RODRIGO ROBLES

*Índice*

1. Protocolos de comunicaciones .....	2
2. Modelo de comunicaciones cliente/servidor y su relación con las aplicaciones web.....	4
3. Métodos de petición HTTP /HTTPS más utilizados.....	5
4. Modelo de desarrollo de aplicaciones multicapa .....	6
5. Front-end Y Back-end.....	7
6. Tipos de páginas web. ....	8
7. Componentes de una aplicación web del lado del servidor. ....	9
8. Programas ejecutados en el lado del cliente y programas ejecutados en el lado del servidor (lenguajes de programación utilizados en cada caso). ....	10
9. Lenguajes de programación utilizados en el lado servidor de una aplicación web.	11
10. Características y posibilidades de desarrollo de una plataforma XAMPP	12
11. En qué 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. ....	13
12. IDE más utilizados. ....	14
13. Servidores HTTP /HTTPS más utilizados.....	17
14. Apache HTTP vs Apache Tomcat .....	19
15. Navegadores HTTP /HTTPS más utilizados.....	20
16. Generadores de documentación HTML.....	21
17. Repositorios de Software .....	22
18. Propuesta de configuración del entorno desarrollo para la asignatura de Desarrollo web del lado servidor en este curso. ....	23
19. Propuesta de configuración del entorno de explotación para la asignatura de Desarrollo web del lado servidor en este curso. ....	24
20. Realizar un estudio sobre los siguientes conceptos y su relación con el desarrollo de aplicaciones web .....	24
21. Estudio Laravel.....	28

## 1. Protocolos de comunicaciones

Un **protocolo de comunicaciones** es un sistema de reglas que permiten que dos o más entidades (computadoras, teléfonos celulares, etc.) de un sistema de comunicación se comuniquen entre ellas para transmitir información por medio de cualquier tipo de variación de una magnitud física.

La comunicación entre servidor y cliente (aplicaciones web) se realiza a través de la red gracias a los siguientes protocolos:

### HTTP [Protocolo de Transferencia de Hipertexto]

Es el protocolo de comunicación que permite las transferencias de información en el World Wide Web (WWW).

### HTTPS [Protocolo Seguro de Transferencia de Hipertexto]

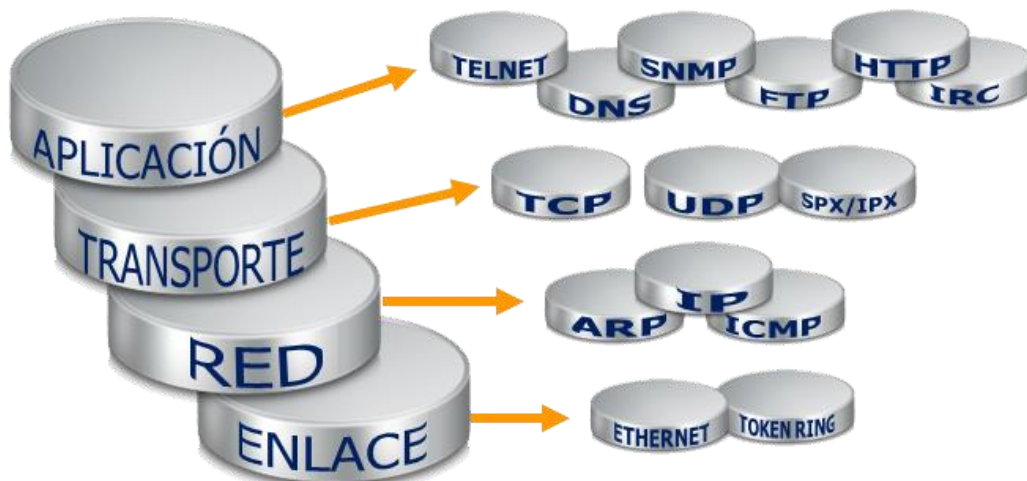
Es más seguro que el HTTP ya que encripta la información que se transmite (transferencia segura de hipertexto).

### IP [Protocolo de Internet]

Es un conjunto de números que identifican de manera única a una interfaz de un dispositivo en la red (es el “nombre” del dispositivo en la red). Se encarga del envío y recepción de datos en bloques.

### TCP (Protocolo de Control de Transmisión)

Se encarga de crear “conexiones” entre computadores para que se cree un flujo de datos (garantiza que se transmitan sin errores y en orden).



## Modelo TCP/IP

**Es un estándar de protocolos de comunicaciones.** El más usado en el mundo.

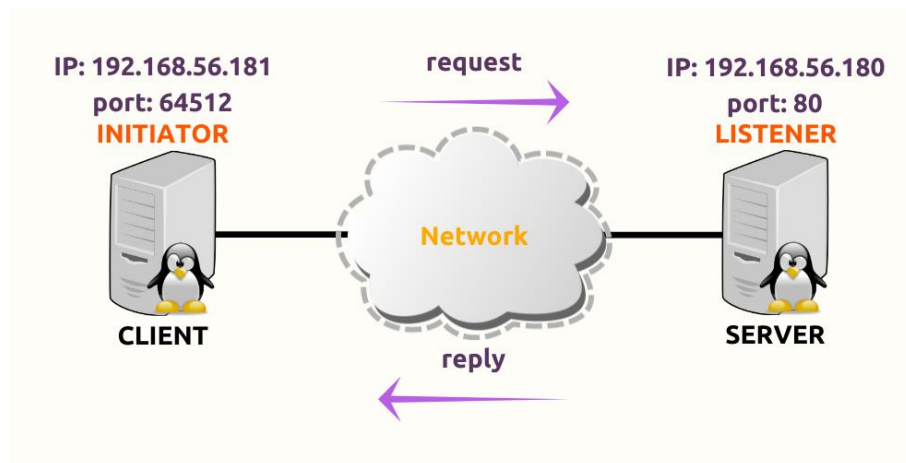
La comunicación del flujo de datos en a través de este estándar se produce gracias a los **sockets**.

Un socket queda definido por un par de direcciones IP local y remota, un protocolo de transporte [TCP] y un par de números de puerto local y remoto -> [El socket se forma al enlazarse a través de TCP dos ip:puerto].

Puerto local: Diferencia a las diferentes conexiones que tiene un mismo dispositivo

Puerto remoto:

- A. HTTP: 80
- B. HTTPS: 433



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

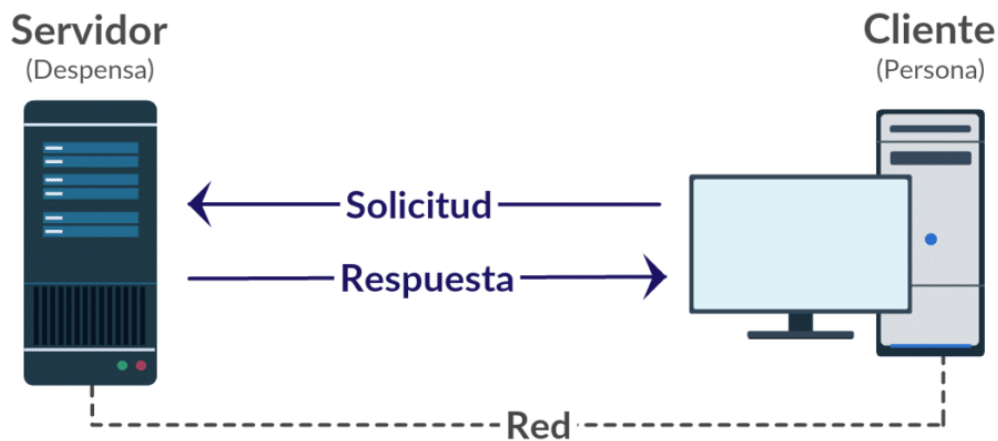
**Todas las aplicaciones web son modelo cliente/servidor, ya que se comunican mediante HTTP**, comunicación que obligatoriamente un lado de la comunicación es cliente y otro servidor.

- **Cliente (Navegador del usuario)**

Consumidor de servicios | Solicita información al servidor

- **Servidor (Servidor Web)**

Proveedor de servicios | Responde al cliente enviando flujos de información.



### 3. Métodos de petición HTTP /HTTPS más utilizados.

1. **GET:** Solicita una representación de un recurso en específico. Las peticiones que usan GET sólo deben recuperar datos. [La URL es un método GET]



2. **HEAD:** Pide una respuesta idéntica a la de una petición de GET, pero sin el cuerpo de la respuesta.

3. **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.



4. **PUT:** Reemplaza todas las representaciones actuales del recurso de destino con la carga útil de la petición (subir, cargar o actualizar).

5. **DELETE:** Borra un recurso en específico.

6. **CONNECT:** Establece un túnel hacia el servidor identificado por el recurso.

7. **OPTIONS:** Es utilizado para describir las opciones de comunicación (métodos HTTP) para el recurso de destino (URL).

8. **TRACE:** Realiza una prueba en bucle de retorno de mensaje a lo largo de la ruta al recurso de destino.

9. **PATCH:** Utilizado para aplicar modificaciones parciales a un recurso.

SAFE METHODS NO ACTION ON SERVER	{	GET	HTTP/1.1 MUST IMPLEMENT THIS METHOD
		HEAD	INSPECT RESOURCE HEADERS
MESSAGE WITH BODY SEND DATA TO SERVER	{	PUT	DEPOSIT DATA ON SERVER — INVERSE OF GET
		POST	SEND INPUT DATA FOR PROCESSING
		PATCH	PARTIALLY MODIFY A RESOURCE
		TRACE	ECHO BACK RECEIVED MESSAGE
		OPTIONS	SERVER CAPABILITIES
		DELETE	DELETE A RESOURCE — NOT GUARANTEED

#### 4. Modelo de desarrollo de aplicaciones multicapa

Dentro del modelo cliente/servidor, el diseño de software más utilizado (es la estructura más básica) es el **modelo de tres capas (división de funcionalidades por capas)**:

##### Capa de presentación

**En esta capa se crea la interfaz del usuario.** Su única función es pasarle las acciones que realice el usuario a la capa de negocio.

##### Capa de negocio

**En esta capa se gestiona la lógica de la aplicación.** Es donde se dice que hacer con los datos (donde se recibe las peticiones del usuario y desde donde se envía la respuesta adecuada).

##### Capa de persistencia

**Esta capa se encarga de guardar los datos.** Será donde se gestiona todo lo relativo a la base de datos y a la creación, edición y borrado de los datos de ésta.



**La arquitectura de software más utilizada es el modelo multicapas**, el cual utiliza el modelo de tres capas (diseño de software), pero cada capa se compone de muchas más (como una cebolla, tiene muchas capas).

### Depende de:

1. **Tamaño de los componentes:** Que elemento de la arquitectura web debe soportar más o menos carga de procesamiento.
  - a. Fat Client | Thin Server: Soporta más carga el cliente, el servidor es un mero administrador de datos.
  - b. Thin Client | Fat Server: Soporta más carga el servidor, el cliente solo presenta la información enviada por el servidor.
2. **La naturaleza del servicio ofrecido:** Capacidades ofrecidas por el servidor (servidores de ficheros, de bases de datos...).
3. **Reparto de las funciones entre el cliente y el servidor:** Según la gestión de las responsabilidades en cada una de las funciones del entorno entre cliente y servidor. Las más habituales son de 3 capas (modelo de tres capas) o 2 capas (capa de negocio y la capa de persistencia).

## 5. Front-end Y Back-end.

### Front-end

**Es la parte de la aplicación desarrollada para todo tipo de usuario** [“usuarios normales” = no manejo ni modificación], en la cual, es importante la presentación por lo que necesita de uno o varios estilos.

### Back-end

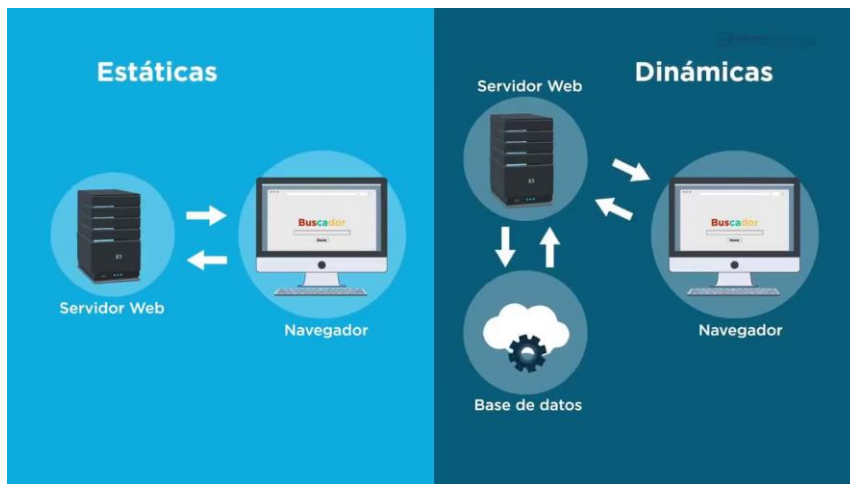
**Es la parte de la aplicación web desarrollada para “usuarios especiales”** [= manejo o/y modificación (administrador, publicador, censor...)]. En esta, no es necesario ningún estilo ya que la presentación no es relevante para dichos usuarios.





## 6. Tipos de páginas web.

- **Aplicaciones web estáticas (Páginas web estáticas):** Solo muestra información al cliente (no interacciones).
- **Aplicaciones web dinámicas (Páginas web dinámicas):** La información que se muestra se genera a partir de una petición del usuario a la página (la información depende del solicitante y del momento en el que se solicita).
- **Aplicaciones interactivas web (Aplicaciones web):** [Página web Dinámica +Compleja] Herramientas que los usuarios pueden utilizar accediendo al servidor web mediante su navegador e internet.
- **Aplicaciones web híbridas (Mashup):** [Página web Dinámica ++Compleja] Una aplicación web es usada o llamada desde otra aplicación, con el fin de reutilizar el contenido o funcionalidad, pero a través de una interfaz completamente nueva (esto se llama Web Service) [tiene persistencia de datos].



### Método de Comprobación

Para comprobar de que tipo es una aplicación web, basta con acceder a ella varias veces a través de la **URL** [Localizador de Recursos Uniforme] y si la página web no cambia en ningún momento o/y tras acceder varias veces, dicha página es estática. Si es al contrario, es decir, hay cambios, la página es dinámica.



## 7. Componentes de una aplicación web del lado del servidor.

1. **Servidor web:** Comunicación a través de la red con el navegador del usuario (cliente). Es el elemento más importante, solo con él ya se podría realizar una aplicación web, pero sin él no. -> Apache
2. **Módulo encargado de ejecutar el código (interprete).** -> Interprete de PHP
3. **Sistema Gestor de la Base de Datos [SGBD] (si es necesario):** Organizan la información con la que trabaja el servidor web. -> MySQL
4. **Código fuente escrito en lenguaje de programación.** -> PHP: Lo interpreta el servidor web a través del interprete para mostrar la información correcta de la manera correcta al cliente.



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

### Programas del lado del cliente

- **Navegador:** Es un software, aplicación o programa que permite el acceso a la Web, interpretando la información de distintos tipos de archivos y sitios web para que estos puedan ser vistos.
- **HTML y CSS**
- **Flash Player:** Es una aplicación informática englobada en la categoría de reproductor multimedia.

### Programas del lado el servidor





- **Apache:** Es un servidor web HTTP de código abierto.
- **MySQL:** Es un sistema de gestión de bases de datos.
- **PHP:** Es un lenguaje de programación, originalmente para el preprocesador de texto plano en UTF-8.
- **Nginx:** Es un servidor web/proxy inverso ligero de alto rendimiento y un proxy para protocolos de correo electrónico.

### Programas de ambos lados

- **JavaScript:** Es un lenguaje de programación interpretado.
- **XML**
- **DHTML**



## 9. Lenguajes de programación utilizados en el lado servidor de una aplicación web.

- **Python**  : Es un lenguaje multiplataforma con diversas herramientas y reduce considerablemente los procesos de desarrollo. En el año 2017 fue el lenguaje de programación más utilizado.
- **Java**  : Está orientado a objetos y es uno de los lenguajes de programación más utilizados, es fácil de utilizar, multiplataforma, seguro fiable y eficiente. En el año 2017 fue el tercer lenguaje de programación más utilizado después de C.
- **PHP**  : Es un lenguaje interpretado, se usa para generar páginas web de contenido dinámico, es robusto y es fácil de aprender. Actualmente la implantación de PHP ha incrementado considerablemente en los 2 últimos años.
- **Perl**  : Es un lenguaje interpretado, es muy dinámico y es especialmente útil para manipular cadenas de caracteres, archivos y procesos. Es el menos utilizado de los 4.

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

**XAMPP** es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos **MySQL**, el servidor web **Apache** y **los intérpretes** para lenguajes de script **PHP** y **Perl**.

### Ventajas

- **Muy fácil de instalar y las configuraciones son mínimas** o inexistentes.
- Es **multiplataforma** (se puede instalar en varios sistemas operativos) (Linux, Windows, MAC o Solaris).
- Es **software libre** (es gratis).

### Desventajas

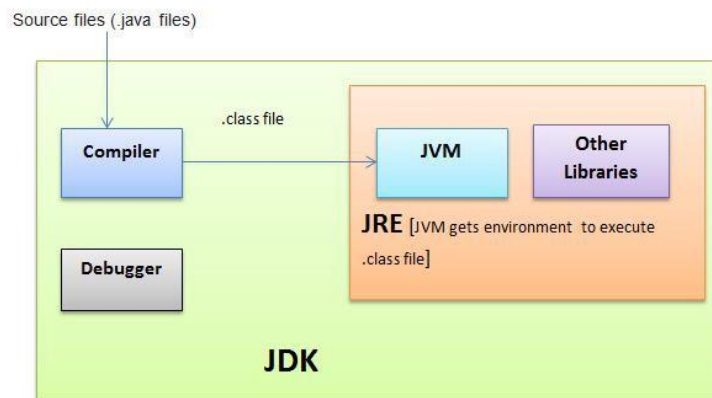
- **No soporta MySQL desde la consola** (línea de comandos).
- **No se pueden actualizar individualmente las versiones de los programas** que instala. (Es necesaria su reinstalación para actualizar sus componentes)
- **Dificultad para configurar aplicaciones de terceros.**
- **Carece de garantía** (su único propósito es la utilidad, no tiene un propósito concreto).



# 11. En qué 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.

Sólo es necesaria la instalación del **JDK en el entorno de desarrollo** y del **JVM en el entorno de explotación cuando estamos desarrollando o ejecutando**, respectivamente, **una aplicación web escrita o que contenga lenguaje Java**, para otros lenguajes distintos no es necesario. Esto se debe a que Java es un lenguaje de compilación a código intermedio, por lo cual necesita un compilador [el cual, pasa el código fuente a un código intermedio] y un ejecutable [ejecuta dicho código intermedio] (JVM).

- **Máquina virtual de Java (JVM):** es una plataforma específica, capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial (el bytecode Java), el cual es generado por el compilador del lenguaje Java. **[EJECUTABLE]**
- **Java Development Kit (JDK):** se trata de un paquete de software provee herramientas de desarrollo para la creación de programas en Java. **[HERRAMIENTAS DE DESARROLLO JAVA]**
- **Java Runtime Environment (JRE):** es una máquina virtual de Java y su función es hacer de intermediario entre una aplicación programada en Java y el sistema operativo que se esté usando. De este modo, cualquier aplicación puede funcionar en cualquier sistema operativo que disponga del JRE. **[CONECTOR ENTRE APL. JAVA Y S. OPERATIVO]**



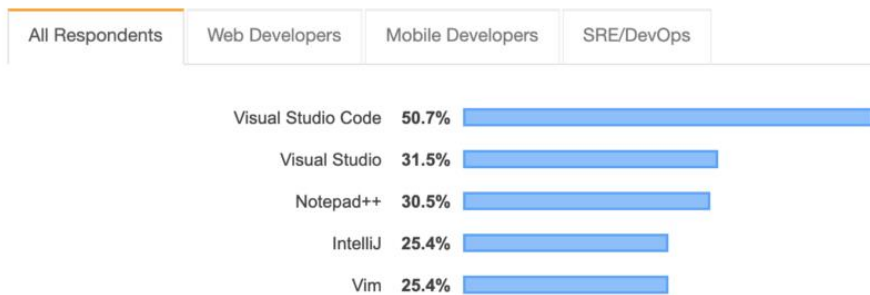
## 12. IDE más utilizados.

Según esta encuesta realizada en 2019 a los usuarios StackOverflow, estos son los 5 IDE más utilizados.



### Development Environments and Tools

#### Most Popular Development Environments



### Visual Studio Code



Visual Studio Code es un potente editor de código disponible para Windows, macOS y Linux. Lanzado en 2015 por Microsoft, VS Code está optimizado para construir y depurar aplicaciones web y de nube modernas.

#### Características Notables

- Soporte para más de 30 lenguajes de programación y también los lenguajes principales de Microsoft como ASP.NET, C#, etc...
- Su pequeño tamaño asegura una instalación y uso rápidos.
- También se encuentran disponibles versiones portátiles.

### Visual Studio



Visual Studio es un IDE premium (el precio depende de la licencia) para desarrollar interfaces gráficas de usuario (GUI), formularios de Windows, juegos, servicios y aplicaciones web.

#### Características Notables

- Una biblioteca masiva de extensiones que siempre está creciendo.
- IntelliSense (función de autocompletado).
- Panel de control personalizable y ventanas acoplables.
- Flujo de trabajo sencillo y jerarquía de archivos.
- Herramientas de automatización.
- Fácil refactorización e inserción de fragmentos de código.
- Soporte de pantalla dividida.
- Lista de errores que permite depurar durante la construcción.

### Notepad++



Notepad++ es un editor de código fuente ligero, portable y gratuito que admite más de 50 lenguajes de programación, scripting y marcado. **No es un IDE, pero es de uso habitual y como se le pueden añadir muchas características puede cumplir como uno.**

#### Características Notables

- Admite el plegado de sintaxis y el resaltado de sintaxis.
- Autocompletar palabras, funciones y mucho más.
- Extensible mediante plugins.

### IntelliJ



IntelliJ IDEA es un completo entorno de desarrollo integrado para los lenguajes Java, Kotlin, Scala, JavaScript, entre otros.

#### Características Notables

- El IDE más rápido y con mejor rendimiento.
- Muy completo en funciones.
- Atajos de teclado en prácticamente todo, con la posibilidad de crear los tuyos.
- La mejor finalización de código.
- Navegación de archivos es muy sencilla (buscador de archivos, no es necesario buscar manualmente).
- Terminal de símbolo de sistema integrado.



- Gran variedad de posibilidades de configuración (opciones de compilación, de ejecución...).

### Vim



Vim es una versión mejorada del editor de código Vi, para los principales sistemas operativos, siendo particularmente popular entre los usuarios de Linux.

#### Características Notables

- Compatible con operaciones completamente basadas en teclado.
- Puede usarse en un entorno de desarrollo remoto usando un terminal mediante SSH.
- Es de código abierto y de uso gratuito.
- Se puede personalizar al contenido del corazón usando. vimrc dotfile y VimScript.
- Mejora la productividad y funciona muy bien con archivos grandes.
- Gran cantidad de complementos compatibles con Vim, lo que amplía su funcionalidad principal.

### NetBeans



NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. **[Es el IDE que vamos a usar diariamente]**

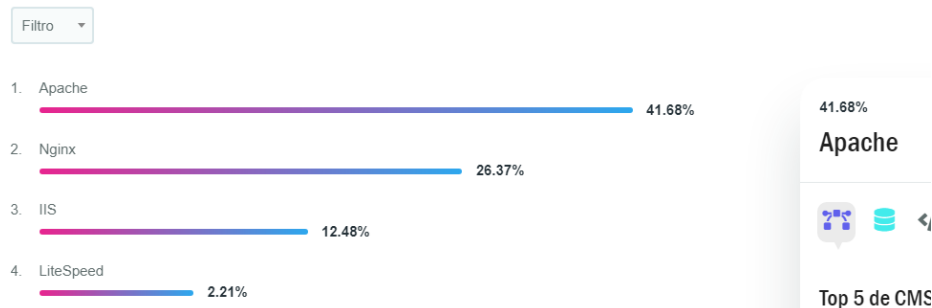
#### Características Notables

- Entorno multilenguaje y multiplataforma
- Puede usarse para crear aplicaciones web y de escritorio, así como para trabajar en Android gracias a plugins.
- Es de código abierto y de uso gratuito.
- Se puede personalizar su interfaz.

## 13. Servidores HTTP /HTTPS más utilizados.

### Cuota del mercado Global de Web Servers 2020

A continuación se muestran los servidores web con la mayoría de los usuarios en todo el mundo



#### DATOS

### Apache HTTP



Apache HTTP Server

Es el web server de referencia para Internet. Nació en 1996 y hasta el día de hoy sigue vigente. Es el más usado.

#### **Ventajas:**

- Código abierto.
- Software gratuito.
- Multiplataforma (Windows, Linux y Unix).

#### **Desventajas:**

- Bajo rendimiento cuando se reciben miles de requests.

### Nginx



Es un servidor web de código abierto y gratuito (aunque también existe una versión comercial) que se destaca por su alto rendimiento. Está diseñado para ofrecer un bajo uso de memoria y alta concurrencia. Segundo más usado.

#### **Ventajas**

Software multiplataforma.

- Consume menos recursos que la mayoría de los servicios que hacen su misma función.
- Alto rendimiento soportando mayor carga y respondiendo.
- Puede ser usado como proxy inverso.

- Podemos integrarlo junto con Apache, de forma que Nginx procese contenido estático y Apache contenido dinámico.

### Desventajas

- No se integra con PHP de forma nativa.
- Es necesario usar Fast CGI para poder integrarlo

## Microsoft 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.

### Ventajas

- Proporcionan capacidades de servidor web integrado.
- Es confiable, seguro y administrable en internet
- Desarrolla y es compatible con las aplicaciones beneficiándose con un único entorno de alojamiento de aplicaciones integrado con compatibilidad total.
- Al momento de la instalación permite elegir sobre que servidor web va a correr (Apache o IIS)

### Desventajas

- Este servidor no es multiplataforma, solo funciona bajo Windows.
- Posee limitaciones en las versiones que no son de la familia "Server".
- Posee vulnerabilidades.

## LiteSpeed



LiteSpeed es un servidor web desarrollado y comercializado por LiteSpeedTech en 2003.

### Ventajas

- Es muy veloz y eficiente, compite directamente con Nginx en cuestión de rendimiento.
- Utiliza el mismo formato de configuración que Apache HTTP Server y es compatible con la mayoría de las funciones de Apache.
- Completa integración con cPanel, lo que significa que es una mejor opción para servicios de web hosting que los servidores Apache que utilizan cPanel como panel de control. Mucho menos uso de recursos.

- Soporta por completo la sintaxis de los archivos htaccess haciendo que la migración entre Apache y LiteSpeed sea menos dolorosa que con otros servidores.
- Es mejor opción más económica pagar la licencia del servidor debido a su menor uso de recursos comparados con la competencia que realizar una actualización de hardware.
- Tiene una versión gratuita llamada [OpenLiteSpeed](#).

### Desventajas

- Es un servidor web de pago.
- Su versión gratis carece de soporte y de muchas características de su versión original.

## Apache Tomcat



Lanzado en 1999 Apache Tomcat es un servidor desarrollado por Apache Software Foundation y habitualmente se lo confunde con el servidor HTTP Apache pensando que es una adaptación.

Apache Tomcat funciona como un contenedor de servlets multiplataforma (escrito en Java). Apache Tomcat entrega contenido que cambia según quién es el cliente, si el cliente ha iniciado sesión y qué ha hecho en interacciones anteriores con el servidor.











## 14. [Apache HTTP vs Apache Tomcat](#)

Tomcat vs. Apache web server: The key differences		
	Apache HTTP Server	Apache Tomcat
Year Released	1995	1999
License	Apache License 2.0	Apache License 2.0
Categorization	File Server	Servlet Engine
Competitors	Nginx, Lighttpd, Microsoft IIS	JBoss, Jetty, Payara, Wildfly, WebSphere Liberty
Prerequisites	modern Linux or Windows operating system	a local JDK installation
Deployment type	any individual file	WAR and JAR files
Deployment folder	htdocs	webapps
Website	<a href="https://httpd.apache.org/">https://httpd.apache.org/</a>	<a href="http://tomcat.apache.org/">http://tomcat.apache.org/</a>
Logo	feather	adult male cat
Default HTTP port	80	8080

## 15. Navegadores HTTP /HTTPS más utilizados.

### Browser Statistics September 2020

Here you see the market shares of the most used browsers for the current month.

Rank	Browser	
1	 Google Chrome	52.56% 
2	 Safari	32.13% 
3	 Firefox	9.24% 
4	 Mozilla	3.61% 
5	 Internet Explorer	2.45% 

### Chrome



Es un navegador web de código cerrado desarrollado por Google, aunque derivado de proyectos de código abierto (como el motor de renderizado Blink). Está disponible gratuitamente.

#### Características:

- RLZ identifier: Una cadena codificada enviada junto con todas las consultas a Google.
- Posee un identificador (ID) único («clientID») para identificar al usuario en los registros de accesos.
- Cuenta con una marca de tiempo de cuándo fue instalado el navegador.
- Páginas de error alojadas en servidores de Google, cuando no se encuentra un servidor.

### Safari



Safari es un navegador web de código cerrado desarrollado por Apple Inc. Está disponible para macOS, iPadOS e iOS. Está escrito sobre el framework WebKit, que incluye a WebCore, el motor de renderizado, y JavaScriptCore, el intérprete de JavaScript.

### Mozilla Firefox



Es un navegador web libre y de código abierto, desarrollado para multiplataforma coordinado por la Corporación Mozilla y la Fundación Mozilla. Usa el motor Gecko para renderizar páginas web, el cual implementa actuales y futuros estándares web.

#### Características:

- Multiplataforma.
- Navegación con pestañas.
- Restauración de sesión.
- Corrector ortográfico.

### Internet Explorer



Fue una serie de navegadores web gráficos desarrollado por Microsoft para el sistema operativo Microsoft Windows desde 1995.

## 16. Generadores de documentación HTML

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

Alguna de las **ventajas** que se pueden obtener usando generadores de documentación son:

- La ayuda que recibe el programador para que todas sus aplicaciones tengan un formato de **documentación similar**.
- Es fácil estandarizar la documentación **sobre qué hace** una función, **qué parámetros recibe**, **que valores devuelve**, etc....

```
/**
 * @param int $disp_mode
 * @param $disp_option
 *
 * @internal param $total
 * @return int|null
 */
function setDisplayMode($disp_mode=0, $disp_option)
```

**Generadores de documentación:**

- [phpDocumentor](#) es un generador de documentación de código abierto escrito en PHP.
- [PHPDoc](#) es una adaptación de JavaDoc para PHP que define un estándar oficial para comentar código php.
- [ApiGen](#) es la herramienta para crear documentación API profesional a partir del código fuente PHP, similar a phpDocumentor / phpDoc.
- [JavaDoc](#) es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java.

## 17. Repositorios de Software

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** diseñado por Linus Torvalds. **Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.**

### CVS



Concurrent Versions System, **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, en un único archivo para cada fichero correspondiente), **que forman un proyecto** (de programa) **y permite que distintos desarrolladores** (potencialmente situados a gran distancia) **colaboren.**

## Subversion



Apache Subversion es una herramienta de control de versiones de código abierto basada en un repositorio cuyo funcionamiento se asemeja enormemente al de un sistema de ficheros. Utiliza el concepto de revisión para guardar los cambios producidos en el repositorio. Entre dos revisiones sólo guarda el conjunto de modificaciones (delta), optimizando así al máximo el uso de espacio en disco. SVN permite al usuario crear, copiar y borrar carpetas con la misma flexibilidad con la que lo haría si estuviese en su disco duro local.

### 18. Propuesta de configuración del entorno desarrollo para la asignatura de Desarrollo web del lado servidor en este curso.

NAJ-W7ED	NAJ-USED
CARACTERISTICAS	CARACTERISTICAS
Sistema Operativo windows 10 Pro 64 bits Memoria RAM 4 GB <u>Particiones</u> CAPACIDAD TIPO 150 GB NTFS 350 GB NTFS 4 GB <u>Nombre de la máquina NAJ-W7ED</u> Usuarios y contraseñas miadmin / paso (administrador del sistema) usuario / paso (usuario maquina) Configuración de red IP 192.168.3.XXX GATEWAY 192.168.3.1 NETMASK 255.255.255.0 DNS 192.168.20.20 / 8.8.8.8 RED 192.168.3.0/24	Sistema Operativo Ubuntu Server 20.04.1 LTS Memoria RAM 2 GB <u>Particiones</u> CAPACIDAD 150 GB TIPO ext4 RUTA DE MONTAJE / CAPACIDAD 350 GB TIPO ext4 RUTA DE MONTAJE /var <u>Nombre de la máquina NAJ-USED</u> Usuarios y contraseñas miadmin / paso (administrador de sistema) operadorweb / paso (administrador sitios web) adminisDB / paso (administrador BBDD) Configuración de red IP 192.168.3.XXX GATEWAY 192.168.3.1 NETMASK 255.255.255.0 DNS 192.168.20.20 / 8.8.8.8 RED 192.168.3.0/24
SOFTWARE USADO	SOFTWARE USADO
IDE Netbeans 12 Navegador Chrome 84.0.4147.94 ClientesFTP Filezilla 3.37 Cliente SSH Putty 0.74 Herramientas Multimedia Gimp 2.10 Ofimatica Libre Office 6.3.2 Software PhpDoc MySQL Workbench	Sistema operativo Ubuntu Server 20.04.1 LTS Servidor administración remota SSH 7.6 Servidor de transferencia de ficheros SFTP (SSH) Servidor Web Apache 2.5 SGBD MySQL 8,0 Módulos PHP 7.4



## 19. Propuesta de configuración del entorno de explotación para la asignatura de Desarrollo web del lado servidor en este curso.

NAJ-W7EE	NAJ-USEE
CARACTERISTICAS	CARACTERISTICAS
Sistema Operativo windows 10 Pro 64 bits Memoria RAM 4 GB <u>Particiones</u> CAPACIDAD TIPO 150 GB NTFS 350 GB NTFS 4 GB <u>Nombre de la máquina NAJ-W7EE</u> Usuarios y contraseñas miadmin / paso (administrador del sistema) usuario / paso (usuario maquina) Configuración de red IP 192.168.3.XXX GATEWAY 192.168.3.1 NETMASK 255.255.255.0 DNS 192.168.20.20 / 8.8.8.8 RED 192.168.3.0/24	Sistema Operativo Ubuntu Server 20.04.1 LTS Memoria RAM 2 GB <u>Particiones</u> CAPACIDAD 150 GB TIPO ext4 RUTA DE MONTAJE / CAPACIDAD 350 GB TIPO ext4 RUTA DE MONTAJE /var <u>Nombre de la máquina NAJ-USEE</u> Usuarios y contraseñas miadmin / paso (administrador de sistema) operadorweb / paso (administrador sitios web) adminisDB / paso (administrador BBDD) Configuración de red IP 192.168.3.XXX GATEWAY 192.168.3.1 NETMASK 255.255.255.0 DNS 192.168.20.20 / 8.8.8.8 RED 192.168.3.0/24
SOFTWARE USADO	SOFTWARE USADO
Navegador Chrome 84.0.4147.94 Cookies JavaScript	Sistema operativo Ubuntu Server 20.04.1 LTS Servidor administración remota SSH 7.6 Servidor de transferencia de ficheros SFTP (SSH) Servidor Web Apache 2.5 SGBD MySQL 8,0 Módulos PHP 7.4

## 20. Realizar un estudio sobre los siguientes conceptos y su relación con el desarrollo de aplicaciones web

### CMS (Sistema de gestión de Contenidos)

Un **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. (WordPress, Joomla, Prestashop...)

### Funciones Principales:

- **Creación de un sitio web**

Un CMS **permite la creación de un sitio web sin necesidad de conocer ninguna de las diversas tecnologías implicadas** o, a lo sumo, conocimientos básicos relacionados con la edición de páginas web.

**En un gestor de contenidos, diseño y contenido son independientes.** Esto significa que se puede cambiar el diseño del sitio web sin que afecte al contenido, que se sigue mostrando adaptado a las características del nuevo diseño.

**Una de las primeras tareas al crear un sitio web con un gestor de contenidos suele ser definir, seleccionar o diseñar su apariencia visual.** En WordPress, por ejemplo, eliges un “tema” y te proporciona una serie de plantillas con espacios reservados a las cuales podemos introducir el contenido.

- **Gestión y mantenimiento del sitio web**

En la mayoría de los casos, un sitio web no es un ente estático, sino que crece y evoluciona con el tiempo, ya sea para incluir nuevas páginas web, con más contenidos, o para modificar los existentes.

**A través del gestor de contenidos, realizar esta actualización o mantenimiento de los contenidos tan solo requiere localizar la plantilla o tipo de contenido que se quiera crear o modificar, editar el contenido según corresponda y guardarlo,** casi como si se tratara de un documento en un procesador de texto.

Por el contrario, si no se utilizara un Website CMS, en función de la complejidad de los cambios, podría ser necesario contratar un profesional para realizar esos pequeños cambios directamente sobre el código HTML, CSS o JavaScript de las páginas.

Asimismo, **el gestor ofrece diversas utilidades para organizar las páginas web y buscar fácilmente los contenidos del sitio web**, a través de un interface sencillo e intuitivo, sin necesidad de localizar los ficheros físicos en el disco duro ni conocer la estructura real en que están almacenados.

**La misma consideración se extiende para el resto de recursos** que utilice el sitio web (imágenes, documentos, vídeos, etc.): **el CMS los gestiona y almacena por nosotros**, que solo tendremos que buscarlos y seleccionarlos cuando los necesitemos.

- **Administración del sitio web y del propio CMS**

**El propio gestor de contenidos, como cualquier otra aplicación informática, también dispone de sus propias opciones de configuración y administración**, relacionados con su funcionamiento. Por ejemplo, el idioma de la interface de usuario, los

formatos de fecha o, una característica muy habitual de los CMS que veremos más adelante, qué usuarios pueden gestionar el sitio web y con qué perfiles.

**Asimismo, también veremos que los gestores de contenido permiten ampliar su funcionalidad a través de plugins o complementos que, a su vez, debe configurarse y administrarse, también a través de la misma interface del gestor.** Por lo general, la configuración por defecto del gestor de contenidos es válida para la mayoría de los sitios web, sobre todo de pequeño tamaño.

Sin embargo, en el momento en que se añaden nuestras funcionalidades al sitio web mediante plugins, o se pretende optimizar algún aspecto en concreto del sitio web (por ejemplo, para el posicionamiento SEO), entonces hay que revisar cuidadosamente la configuración para ajustarla lo mejor posible.



## ERP (Sistema de planificación de recursos empresariales)

Un [ERP](#) es un software que permite a las empresas controlar todos los flujos de información que se generan en cada ámbito de la organización. Los sistemas ERP integran los departamentos (donde antes teníamos un programa especializado para cada uno, ahora, podemos asegurar la existencia de una única base de datos centralizada donde se gestione la información en tiempo real y con eficiencia – [cada departamento, un módulo]).

**El objetivo último de cualquier sistema ERP es el de ayudar a las empresas en sus tareas de administración y toma de decisiones, automatizando todos sus procesos.** Gracias a ello podemos obtener datos en tiempo real, mejorar tareas de back office, controlar flujos de trabajo y minimizar errores.

### Características:

- **Administración de la cadena de suministro** – Se puede seguir fácilmente el proceso de suministro. Desde la producción a la distribución.

- **Análisis de información** – Como toda la información se almacena en una única base de datos, es fácil para el ERP analizarla.
- **Automatización** – Los ERP siempre automatizarán tareas como la creación de informes, por ejemplo.
- **Contabilidad** – Gestión de finanzas y facturación. Normalmente se compagina con compras y ventas.
- **CRM** – Gestión de relación con el cliente, comunicación directa, seguimiento de oportunidades, etc.
- **Generación de informes** – Como gestionan tantos datos, todos los ERP realizan informes.
- **Gestión de compras y ventas** – Opciones para controlar los gastos y entradas desde un solo lugar.
- **Gestión de proyectos** – Gracias a la integración con finanzas, permite calcular las ganancias.
- **Gestión de stock e inventario** – Organización desde el precio y los almacenes hasta los pedidos. Especialmente si son compatibles con una aplicación como Shopify.
- **Integración** – Este integra las distintas ramas de una empresa y centraliza sus datos. Esta es absoluta a través de departamentos y unidades de negocio.
- **Operaciones en tiempo real** – Los problemas se identifican rápidamente, dándole así tiempo al vendedor de reaccionar y encontrar una respuesta rápida.
- **Pagos online** – Opciones de pago online y facilidad para conectar con aplicaciones como PayPal y Stripe, por ejemplo.
- **Personalización** – Es imposible diseñar el programa perfecto que se adapte a todas las necesidades. Por esta razón, los buenos ERP se pueden personalizar y son flexibles.
- **Un aspecto uniforme** – Distintas ramas de la empresa se agrupan bajo un mismo programa, dándole así un aspecto uniforme.
- **Recursos humanos (RRHH)** – Gestión de nóminas, contratos, ausencias, información de empleados, etc.



21. Estudio Laravel