

Índice

1.-Protocolos de comunicaciones.....	3
-IP:.....	3
-TCP:.....	3
-HTTP:.....	3
-HTTPS:.....	3
2.-Modelo de comunicaciones cliente - servidor..	4
3.-Estudio sobre los métodos de petición HTTP /HTTPS más utilizados.....	5
-Get:.....	5
-Head:.....	5
-Post:.....	5
-Options:.....	5
-Put:.....	5
-Delete:.....	6
-Trace:.....	6
-Connect:.....	6
4.-Modelo de desarrollo de aplicaciones multicapa.....	7
-Capa de presentación:.....	7
-Capa de negocio:.....	7
-Capa de datos:.....	7
5.-Modelo de división funcional front-end / back-end.....	8
6.-Página web estática, dinámica y aplicación web.....	9
7.-Componentes de una aplicación web.....	10
-Servidor web:.....	10
-Capa de aplicaciones:.....	10
-Sistema gestor de bases de datos:.....	10
-Módulo de PHP:.....	10
8.-Programas ejecutados en el lado del cliente y en el lado del servidor.....	11
9.-Lenguajes de programación utilizados en el lado servidor.....	12
-Python:.....	12
-Java:.....	12
-PHP:.....	12
-Perl:.....	12

10.-Características de una plataforma de desarrollo.....	13
-XAMPP:.....	13
-Java EE:.....	13
-MEAN:.....	13
11.-Instalación de JVM y JDK.....	14
12.-IDE más utilizados.....	15
-Eclipse:.....	15
-NetBeans:.....	15
-Visual Studio:.....	15
13.-Servidores HTTP /HTTPS más utilizados.....	16
-Apache:.....	16
-Nginx:.....	16
-Microsoft IIS:.....	16
-Google web Server:.....	16
14.-Navegadores HTTP /HTTPS más utilizados.....	17
-Google Chrome:.....	17
-Safari:.....	17
-Firefox:.....	17
-Internet Explorer:.....	17
-Opera:.....	18
15.-Generadores de documentación HTML.....	19
16.-Repositorios de software.....	20
17.-Propuesta de configuración del entorno de desarrollo.....	21
CMH-WXED:.....	21
CMH-USED.....	21
18.-Propuesta de configuración del entorno de explotación.....	22

1.-Protocolos de comunicaciones.

-IP:

Es el protocolo de Internet, se encarga de la comunicación bidireccional entre dos sistemas para transmitir datos a través de las redes físicas, los datos son enviados en forma de paquetes.

-TCP:

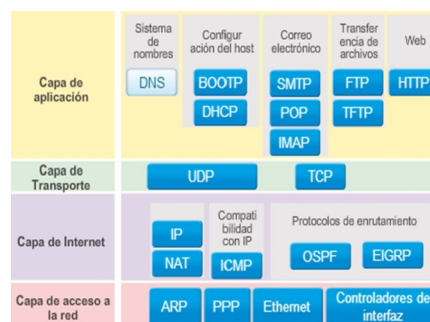
Es el protocolo de control de transmisión, éste asegura que los datos sean entregados a su destino correctamente y en el mismo orden en que fueron enviados, además permite distinguir aplicaciones en un sistema mediante los puertos.

-HTTP:

Es el protocolo de transferencia de hipertexto, permite la transferencia de información en la web de forma no segura, aunque no guarda información sobre conexiones anteriores, para lo cual se emplean las cookies. La comunicación se realiza por defecto por el puerto 80.

-HTTPS:

Es el protocolo seguro de transferencia de hipertexto, permite el intercambio de información en la web de forma segura, está basado en el protocolo HTTP. La comunicación se realiza por defecto por el puerto 443.

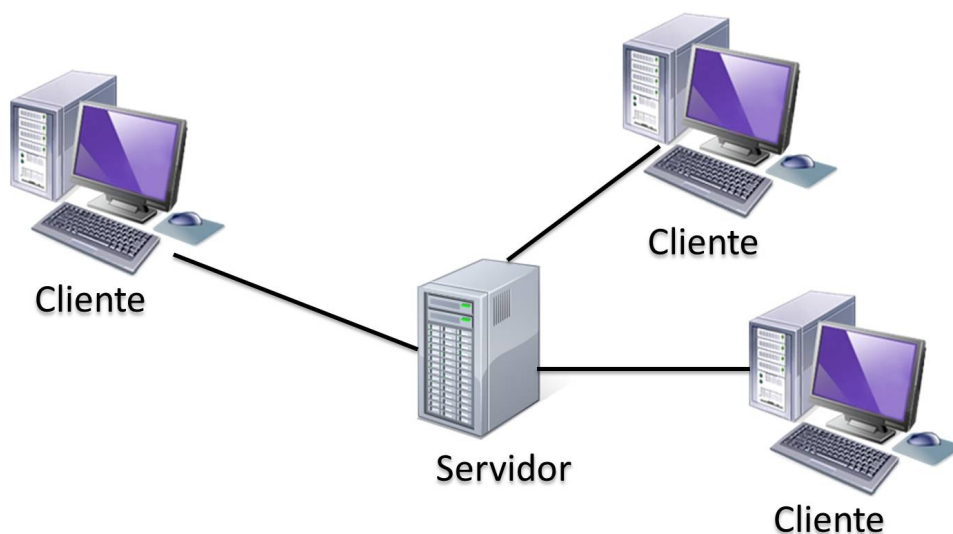


2.-Modelo de comunicaciones cliente - servidor.

Y su relación con las aplicaciones web.

Es un modelo que se basa en que las tareas se reparten entre los servidores y los clientes, éstos últimos realizan peticiones a los servidores, que le dan una respuesta. Éste concepto está íntimamente relacionado con el desarrollo web donde se separan las aplicaciones en dos entornos: el entorno cliente que es todo aquello que se va a cargar en el navegador y va a estar a disposición de los clientes y el entorno servidor que es la información, recursos y aplicaciones almacenadas en el servidor.

En la imagen se muestra un esquema general de este modelo:



La información se comparte fácilmente a través de redes, en donde los servidores almacenan datos que pueden ser compartidos por los clientes.

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

-Get:

Se emplea para adquirir un recurso ubicado en un servidor web, que devuelve los metadatos y el recurso que se precisa.

-Head:

Realiza una función similar a la anterior, pero sólo se solicitan los metadatos y no el recurso en sí.

-Post:

Se usa para enviar información a un recurso web del servidor, no para crear un elemento nuevo, es especialmente útil en el envío de formularios.

-Options:

Sirve para descubrir qué métodos HTTP soporta un servidor concreto.

-Put:

Crea un recurso nuevo en el servidor o si éste ya existe lo reemplaza.

-Delete:

Borra un determinado recurso del servidor.

-Trace:

Sirve para el diagnóstico de errores o detección de servidores intermedios en la conexión mediante la monitorización de mensajes.

-Connect:

Solicita una conexión TCP/IP, es útil para llevar a cabo una conexión HTTPS o SSL.

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.

Comunicación entre capas, componentes y funcionalidad de cada capa.

Es un tipo de desarrollo software donde se persigue la separación de las partes de las que está compuesta un sistema software en capas, éstas sólo pueden comunicarse con la capa anterior, de la que reciben información y con la capa posterior, a la que envían información:

-Capa de presentación:

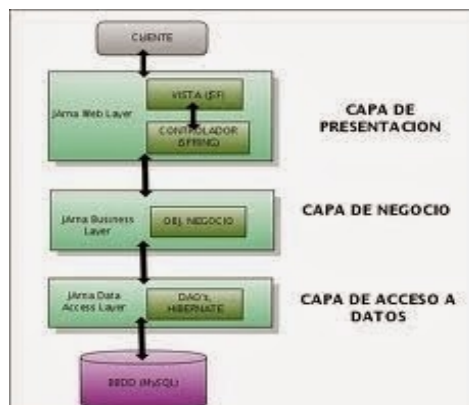
Es lo que el usuario ve, le transmite información y captura información de éste.

-Capa de negocio:

Se recibe la información del usuario y se envían los datos tras el procesamiento.

-Capa de datos:

Contiene los datos y se encarga de acceder a ellos pudiendo tener uno o más gestores de BBDD.



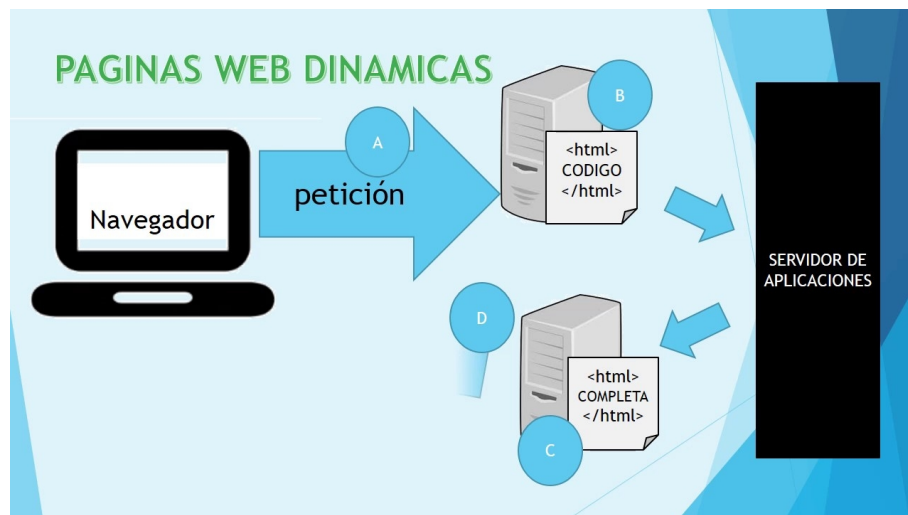
5.-Modelo de división funcional front-end / back-end.

Es una abstracción que mantiene dos roles distintos dentro de una aplicación web, por un lado el front-end que es la parte de la aplicación creada para usuarios normales que van a utilizar la aplicación y que contarán con una interfaz más amigable que en el back-end y con menos funcionalidad, por otro lado el back-end es la parte de la aplicación a la que tendrán acceso los usuarios especiales o administradores de la aplicación, que contarán con una mayor funcionalidad que los usuarios normales, aunque la interfaz gráfica suele ser menos amigable para el usuario.

- El usuario back – end
Super admin; administrador; gestor
- El usuario front – end
Registrado; autor; editor; publicador

6.-Página web estática, dinámica y aplicación web.

Una página web estática es empleada para la realización de páginas sin muchas pretensiones que únicamente desean informar de algo concreto, pero en las que no se pueden realizar acciones que alteren el contenido de la página, por el contrario, las páginas web dinámicas, también denominadas comúnmente aplicaciones web, permiten la interacción del usuario con la página, mostrando diferentes resultados dependiendo de las acciones que lleve a cabo el usuario. El uso de unas u otras páginas dependerá de lo que se pretenda hacer con ellas.



7.-Componentes de una aplicación web.

La alternativa más frecuente para realizar una aplicación web es conocida como XAMPP (Apache, MySQL y un intérprete de PHP y Perl) y está formada por:

-Servidor web:

Generalmente suele ser Apache, se encarga de la comunicación a través de la red con el navegador del usuario.

-Capa de aplicaciones:

Se encarga de dar la funcionalidad a una aplicación web o página dinámica, estos programas pueden estar escritos en muchos lenguajes ya sean compilados o interpretados, uno de los más utilizados es PHP.

-Sistema gestor de bases de datos:

Se encarga de organizar toda la información con la que se trabaja, las bases de datos más empleadas son las relaciones, especialmente MySQL.

-Módulo de PHP:

Se encarga de interpretar los comandos en PHP escritos en la aplicación para procesarlos antes de enviar la página HTML al cliente.



8.-Programas ejecutados en el lado del cliente y en el lado del servidor.

Lenguajes de programación utilizados en cada caso.

Los programas ejecutados en el lado del servidor están escritos generalmente en PHP/Perl/Python... embebidos en el HTML, después de ser ejecutados se le envía al cliente el resultado en forma de HTML, la parte en PHP nunca es mostrada al cliente en el navegador, sin embargo, los programas ejecutados en el lado del cliente suelen estar escritos en JavaScript y son ejecutados directamente en el navegador del cliente sin tener que pasar por un pretratamiento en el servidor.



9.-Lenguajes de programación utilizados en el lado servidor.

En una aplicación web con características y grado de implantación actual.

-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 de una plataforma de desarrollo.

Y posibilidades de desarrollo de una plataforma XAMPP, MEAN y Java EE.

-XAMPP:

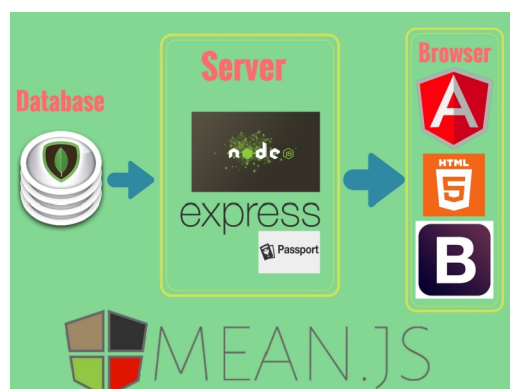
Una de sus mejores características es que es multiplataforma teniendo versiones para muchos sistemas operativos, cuenta con licencia libre y consta principalmente del servidor web Apache, la base de datos MySQL e intérpretes para PHP y Perl, es fácil de utilizar y permite interpretar páginas dinámicas. Es una de las plataformas más intuitivas y utilizadas en windows para desarrollo web.

-Java EE:

Utiliza el lenguaje de programación Java y cualquier base de datos.

-MEAN:

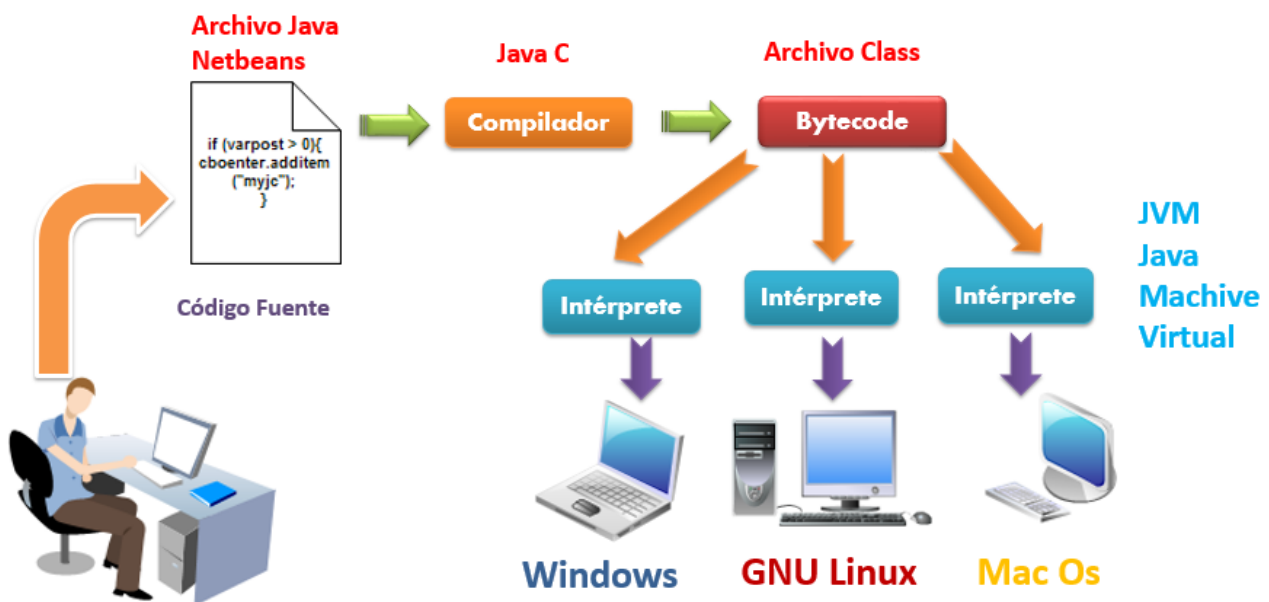
Utiliza el lenguaje de programación JavaScript y la base de datos Mongo.



11.-Instalación de JVM y JDK.

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.

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.



12.-IDE más utilizados.

Características y grado de implantación actual.

-Eclipse:

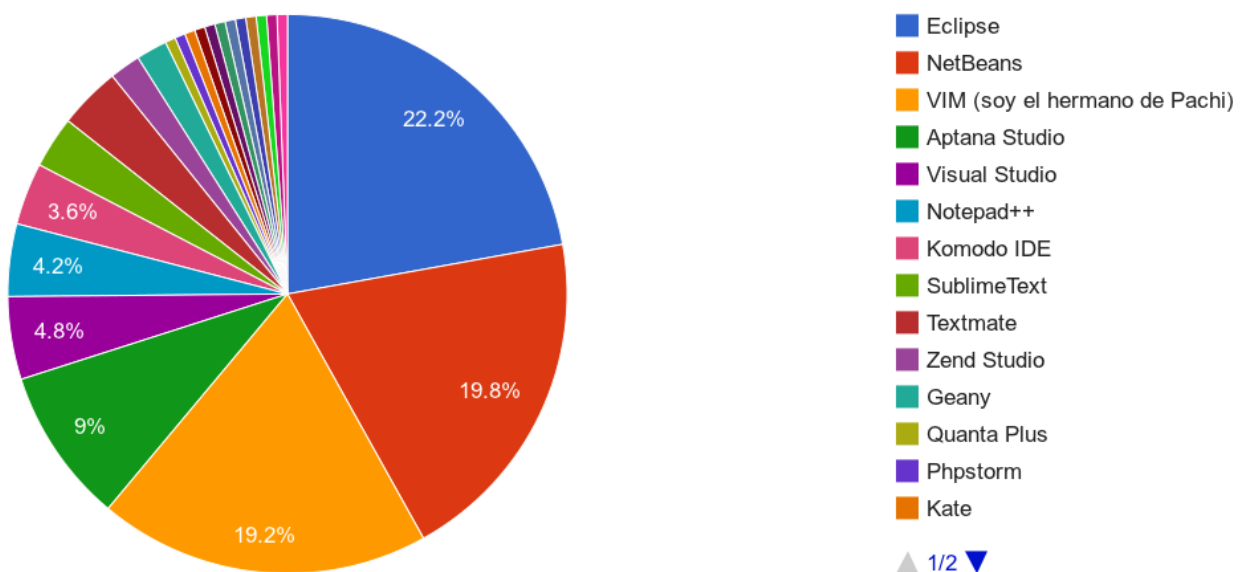
Es el IDE más conocido y utilizado, es de código abierto y multiplataforma, posee muchos plugins que hacen que sirva para casi cualquier lenguaje de programación.

-NetBeans:

Es el segundo IDE más popular, es gratuito, multilenguaje y multiplataforma, fue creado para Java, aunque soporta muchos más lenguajes.

-Visual Studio:

Es uno de los mejores IDE con muchas ayudas al programador, aunque sólo soporta lenguajes de programación de Microsoft.



13.-Servidores HTTP /HTTPS más utilizados.

Características y grado de implantación actual.

-Apache:

Es el más utilizado, su principal característica es que la mayor parte de fallos de seguridad descubiertos sólo pueden ser aprovechados por usuarios locales.

-Nginx:

Es multiplataforma, y es ampliamente utilizado, es el segundo servidor más utilizado.

-Microsoft IIS:

Incluye varios servicios, convierte el ordenador en un servidor web permitiendo que todos los ordenadores que lo posean puedan publicar sitios web de forma local o remota. Es el tercer servidor web más empleado.

-Google Web Server:

Es empleado por Google en todas sus infraestructuras.

14.-Navegadores HTTP /HTTPS más utilizados.

Características y grado de implantación actual.

-Google Chrome:

La principal característica es su gran velocidad, es seguro y estable, posee un modo incógnito, permite el uso de extensiones y es compatible con la mayoría de lenguajes de programación.

-Safari:

Es el navegador mejor adaptado para equipos Apple, aunque también se usa en Windows, no es un navegador tan rápido ni seguro como Chrome y tampoco recibe demasiadas actualizaciones.

-Firefox:

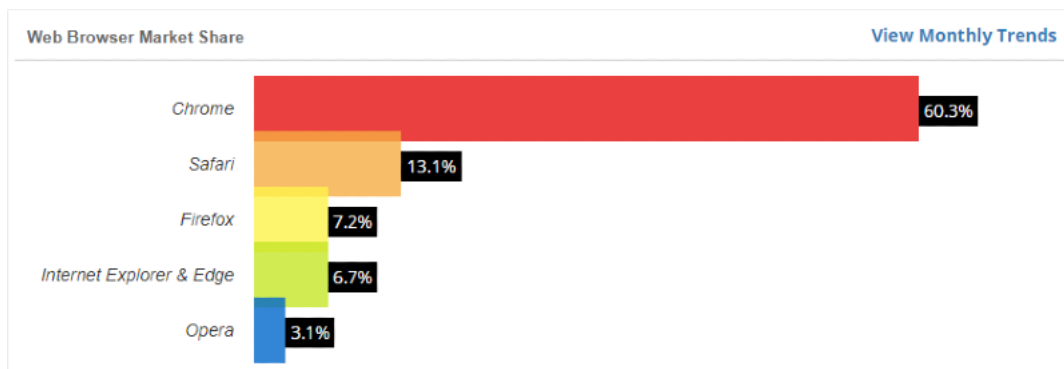
Es un navegador muy seguro y uno de los más antiguos, recibe muchas actualizaciones de decenas de programadores, es estable y personalizable aunque es más lento que Chrome y posee más incompatibilidades.

-Internet Explorer:

Fue el primer navegador en aparecer aunque actualmente no se emplea mucho debido a que es un navegador lento y de poca fiabilidad, ya que la seguridad de este navegador no ha sido muy actualizada.

-Opera:

Es uno de los navegadores más utilizados en dispositivos móviles, y es uno de los navegadores mas antiguos, es compatible con las extensiones creadas para Google Chrome, además es un navegador muy estable y rápido. En el siguiente gráfico se muestran los navegadores web más utilizados en 2018 con sus porcentajes:



15.-Generadores de documentación HTML.

PHPDoc: PHPDocumentor, ApiGen, ...

Son muy útiles a la hora de generar documentación HTML, PDF, Docbook... en base al código escrito, se basan en el estándar formal PHPDoc, son compatibles tanto con la orientación a objetos como con procedimientos, están escritos en PHP y se pueden usar tanto desde línea de comandos como mediante interfaz web.

```

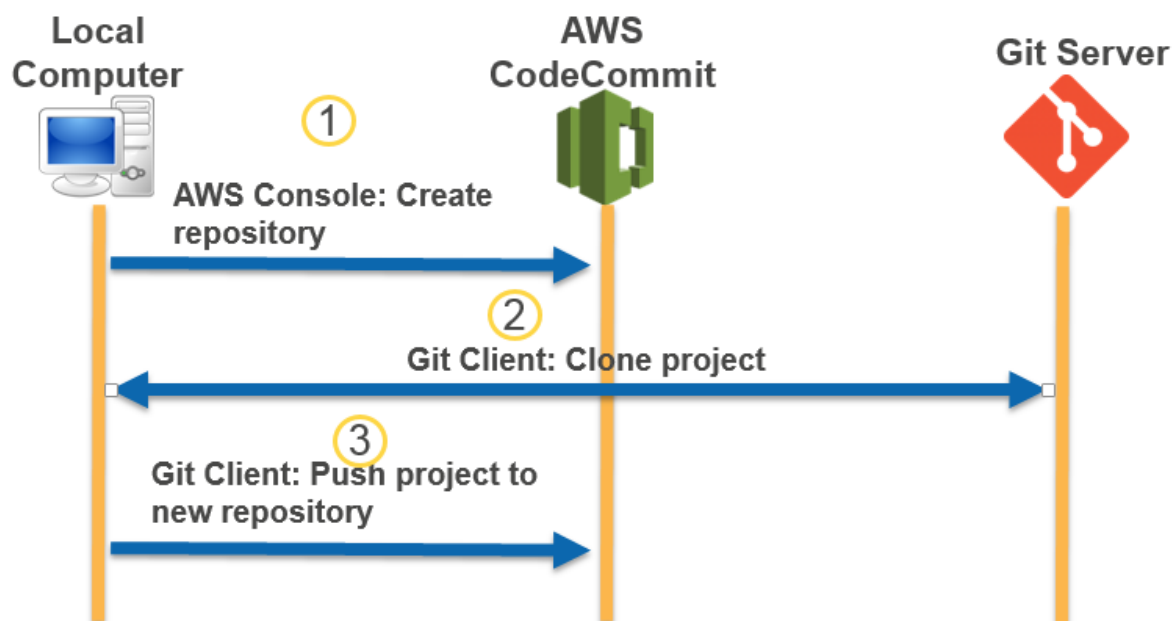
/**
 * @param int $disp_mode
 * @param $disp_option
 *
 * @internal param $total
 * @return int|null
 */
function setDisplayMode($disp_mode=0, $disp_option)
{
    if ($disp_mode == 0) {

```

16.-Repositorios de software.

Sistemas de control de versiones para aplicaciones web: GIT, CVS, Subversion, ...

Los repositorios permiten almacenar nuestras aplicaciones web en un servidor ya sea local o remoto y nos permiten implantar un sistema de control de versiones dentro de nuestras aplicaciones, para que si se produjera un error en la aplicación podamos volver atrás a una versión anterior donde no existía dicho error para determinar el por qué de éste. El repositorio más utilizado de forma remota es el GitHub aunque existen otros como CVS o Subversion. De forma local, uno de los más empleados es el GitLab.



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.

CMH-WXED:

Sistema operativo: Windows 10 education 64 bits.

Servidor administración remota: SSH.

Servidor de transferencia de ficheros: SFTP (SSH).

Repositorio: GIT Lab y GIT Hub.

Servidor web: Apache 2.4.29 (mod_php, mod_ssl,...) y XAMPP 7.2.9.

SGBD: MySQL 8.0.12.

Navegador: Chrome 69.0.3497.100, Mozilla 62.0, Edge 42.17134.1.0 y Opera 55.0.2994.61.

IDE: NetBeans 8.2.

Ofimática, multimedia, generador html, ...: Libre office 6.1.0.3.

Cliente ssh: FileZilla 3.37.0 y Putty 0.70.

CMH-USED

Sistema operativo: Ubuntu Server 18.4.1.

Servidor web: Apache 2.4.29 (mod_php, mod_ssl,...) y XAMPP 7.2.9.

SGBD: MySQL 8.0.12.

Repositorio: GIT Lab y GIT Hub.

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.

Sistema operativo: Ubuntu Server 18.4.1.

Servidor administración remota: SSH.

Servidor de transferencia de ficheros: SFTP (SSH).

Repositorio: GIT Lab y GIT Hub.

Servidor web: Apache 2.4.29 (mod_php, mod_ssl,...) y XAMPP 7.2.9.

SGBD: MySQL 8.0.12.

Cliente ssh: FileZilla 3.37.0 y Putty 0.70.

Navegador: Chrome 69.0.3497.100, Mozilla 62.0, Edge 42.17134.1.0 y Opera 55.0.2994.61.