FrameWork: ofrece librerías, estructurar código, mayor reusabilidad y a nivel web acelera el proceso de desarrollo, ofrecen escribir menos código para hacer mas funciones. DJANgo, node, AngulaJS, sympony

Librería: paquetes métodos y funciones a los que llamamos dedsde nuestro codigo, ayudan a trabajar al tener funciones específicas. Jqueri, CreateJS, Zepto, React

API: aplication program interface, es una interfaz que permite comunicarnos con las librerías con las que se tenga autorización.

Esqueleto

Patron: MVC

Diseño responsive

Web estática --> HTML y CSS

Web dinámica --> Aplicacion web

URL--> donde está alojado el servicio

URI--> en la carpeta donde se encuentre el servicio

El servidor depende más del hardware y el cliente no.

No es lo mismo abrir un navegador web desde el móvil que desde una app nativa.

Lenguaje de cliente se ejecuta en el navegador

Lenguaje de servidor ejecuta en el servidor

De marcas: estructura el esqueleto de la web

Lenguajes de scripting es lenguaje interpretado

Lenguajes de Programación es lenguaje compilado

ECMAScript: Es la especificación

JavaScript: Implementación que surge de ECMA y es de código abierto

JScript: el javaScript de Microsoft

*Caso Practico1:*

¿Que crees que está ocurriendo?:

-Se colapsa puesto que cada vez hay mas peticiones para inicio de sesión además de las peticiones de solicitud de datos

¿Como podría solucionarse?

Implementando una capa mas para la gestión de datos y así que el servidor se ocupe de menos tareas

*Caso Practico 2:*

¿Podría saturarse el servidor?¿Cómo afecta a la experiencia de usuario?

Si, terminaría pudiendo colapsarse el servidor o se ralentizaría el uso de los servicios.

¿Cómo propondrías solucionarlo?

Realizado una verificación en dos pasos, implementando un Captcha o el registro de la mac del pc que realiza la petición e impedir que dos mac iguales realicen más de una petición

¿Puede suponer un problema de seguridad? ¿Por qué?:

Si, puesto que se puede llegar a tumbar el servidor y dejarlo inoperativo durante horas.

*Configuración necesaria para trabajar*

El servidor pueda responder a nuestras peticiones

Navegador e IDE para testear y desarrollar

Control de versiones

Gestión de equipo: asignar tarea y funciones a los miembros del equipo

Como entorno de desarrollo usaremos GIT para compartir nuestros códigos con los compañeros

*Aprendizaje*: Creo que el saber utilizar tanto git como SVN me vendrá muy bien para el futuro puesto que todas las empresas se preparan para su uso si no lo utilizan ya

Las actividades se encuentran entregadas en la plataforma TEAMS cada una en su correspondiente tarea de entrega

Reflexión: es el primer tema y siempre es más pesado por ser tanta teoría, como grupo nos funciona muy bien la retransmisión por teams y nosotros seguir los pasos para quedarnos de como se hacen ciertas funciones, además de dar dinamismo a la clase, sería conveniente que las explicaciones fuesen mas altas y vocalizadas, por que con la mascarilla algunas veces cuesta entender la explicación, pero todo eso se puede mejorar fácilmente. Creo que me va a gustar esta asignatura, tendré que dedicarle horas en casa pero me gusta el ver resultados poco a poco y poder usarlos en común al final del tema.