

CLASE NRO: 01 – CONCEPTOS PREVIOS + INTRODUCCIÓN A LA PROGRAMACIÓN

Tipo de Clase: Teórica

Duración: 40 minutos

Objetivo de la clase:

El alumno al finalizar la clase debe entender claramente el protocolo HTTP y HTTPs a nivel conceptual, el ciclo de vida de la página web (desde que el cliente solicita un recurso/página web y el servidor responde la solicitud y luego, el navegador renderiza la página. Entendiendo que el código JavaScript comienza a interactuar con la página cuando esta está totalmente renderizada. Que es el evento window.onload

A todo esto, el alumno debe dominar conceptualmente la diferencia clara existente entre una página web, un sitio web, una aplicación web.

Introducción a la Programación:

1) ¿Qué es la Programación?

La programación es el proceso de analizar, diseñar, construir, testear, probar y poner en funcionamiento un programa siendo este (el programa) un conjunto de instrucciones escritas es un lenguaje que la computadora puede entender, interpretar y ejecutar paso a paso siguiendo una lógica coherente con el objetivo de resolver un problema determinado. Este programa tiene un principio y un final. En nuestro caso el principio será programa estará escrito en este bloque {}

Importancia de la Programación

La programación es fundamental en la era digital. Permite automatizar tareas, desarrollar aplicaciones, analizar datos, y crear sistemas complejos que facilitan la vida diaria.

2) Lenguajes de Programación

Es un conjunto de instrucciones escritas en un lenguaje de programación que están ordenadas de forma lógica, que se ejecutan una a la vez y que en conjunto resuelven un problema determinado.

Tipos de Lenguajes de Programación

Lenguajes de Alto Nivel: Como JavaScript, Python, y Java, que son más fáciles de entender y escribir.

Lenguajes de Bajo Nivel: Como el lenguaje ensamblador, que están más cerca del lenguaje máquina y son más difíciles de entender para los humanos.

3) Componentes Básicos de un Programa

- Variables y Tipos de Datos
- Estructuras de Control
- Condicionales
- Bucles
- Funciones
- Estructuras de Datos (Vectores)

4) Programación orientada a objetos

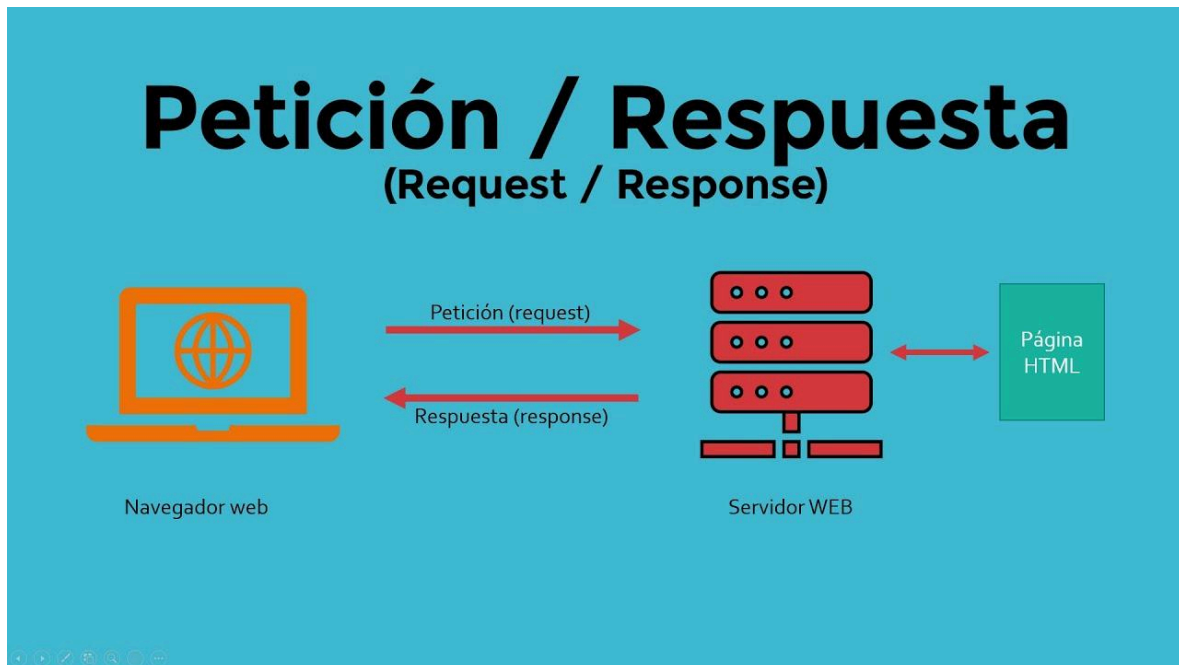
5) Depuración y manejo de errores

Depuración: proceso de encontrar y corregir errores en el código

Manejo de errores: Permite gestionar errores de forma controlada sin detener la ejecución del programa.

Introducción al protocolo HTTP y HTTPs.

- ✓ Inicio de la Solicitud: El ciclo de vida comienza cuando el usuario ingresa una URL en su navegador o hace clic en un enlace.
- ✓ Resolución DNS: El navegador traduce el nombre de dominio en la URL a una dirección IP utilizando el Sistema de Nombres de Dominio (DNS).
- ✓ Establecimiento de la Conexión: Se establece una conexión TCP (o TLS si se utiliza HTTPS) entre el cliente (navegador) y el servidor.
- ✓ Envío de la Solicitud: El navegador envía una solicitud HTTP al servidor web. Esta solicitud incluye la URL, los encabezados y, en algunos casos, datos adicionales como parámetros de consulta o contenido del formulario.
- ✓ Procesamiento de la Solicitud en el Servidor: El servidor web recibe la solicitud y la procesa. Esto implica la identificación del recurso solicitado, la ejecución de scripts si es necesario (por ejemplo, PHP, Python, etc.), y la recuperación de datos de la base de datos si es aplicable.
- ✓ Generación de la Respuesta: El servidor web genera una respuesta HTTP que incluye el código de estado (por ejemplo, 200 OK, 404 Not Found), los encabezados de respuesta y, opcionalmente, el contenido del recurso solicitado.
- ✓ Envío de la Respuesta: El servidor envía la respuesta al navegador a través de la conexión TCP/TLS establecida.
- ✓ Renderizado en el Navegador: El navegador recibe la respuesta y comienza a renderizar el contenido. Esto implica interpretar y renderizar el HTML, aplicar estilos CSS, ejecutar scripts JavaScript, y cargar recursos externos como imágenes, archivos CSS y JavaScript adicionales.
- ✓ Interacción del Usuario: Una vez que la página se ha renderizado completamente en el navegador, el usuario puede interactuar con ella haciendo clic en enlaces, enviando formularios, desplazándose por la página, etc.



Protocolo HTTP (Video de 13 minutos) donde explica funcionamiento de HTTP

VIDEO:

[Protocolo HTTP](#)

Diferencias Fundamentales de los siguientes CONCEPTOS

- o Página WEB
- o Sitio WEB
- o Aplicación WEB

Página Web:

Una página web es un único documento HTML que puede contener texto, imágenes, videos, enlaces, y otros elementos multimedia. Es una sola unidad de contenido que se puede visualizar en un navegador web. Cada página web tiene una URL única.

Ejemplo: La página de inicio de Wikipedia es una página web (<https://es.wikipedia.org/wiki/Wikipedia:Portada>).

Sitio Web:

Un sitio web es un conjunto de páginas web relacionadas y conectadas entre sí mediante enlaces. Estas páginas comparten un nombre de dominio común y generalmente están organizadas en torno a un tema o propósito específico.

DIPLOMATURA EN DISEÑO WEB FULL STACK CON JAVASCRIPT

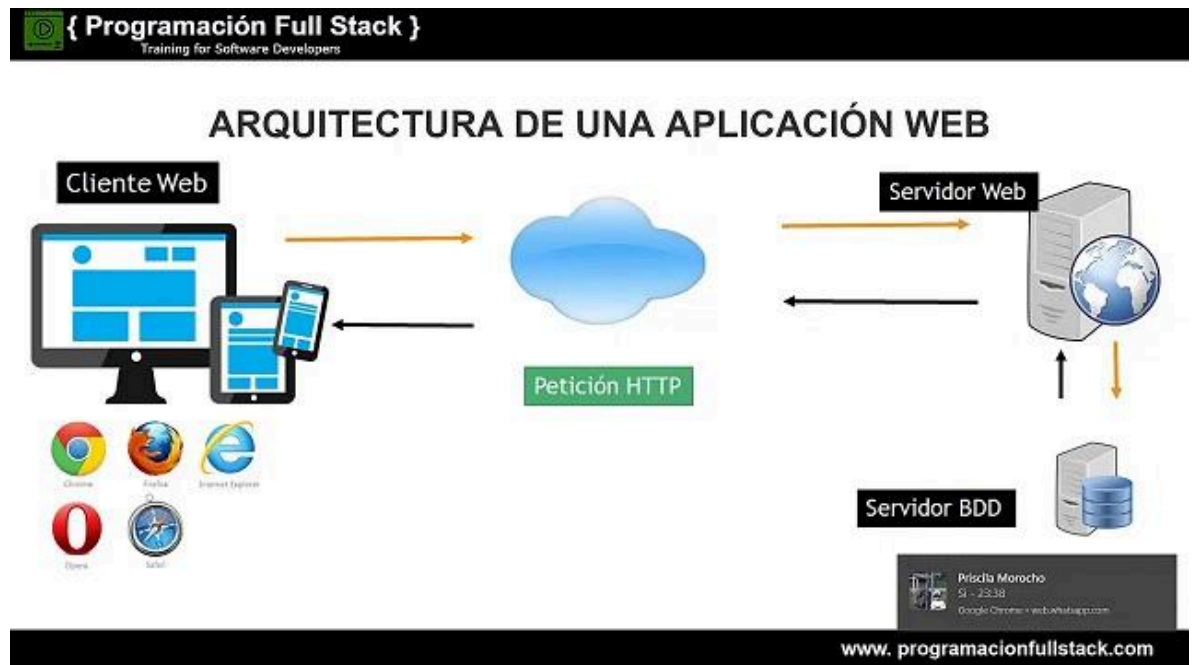
MÓDULO 02 - JAVASCRIPT

Ejemplo: Wikipedia (<https://www.wikipedia.org/>) es un sitio web que contiene numerosas páginas web sobre diferentes temas.

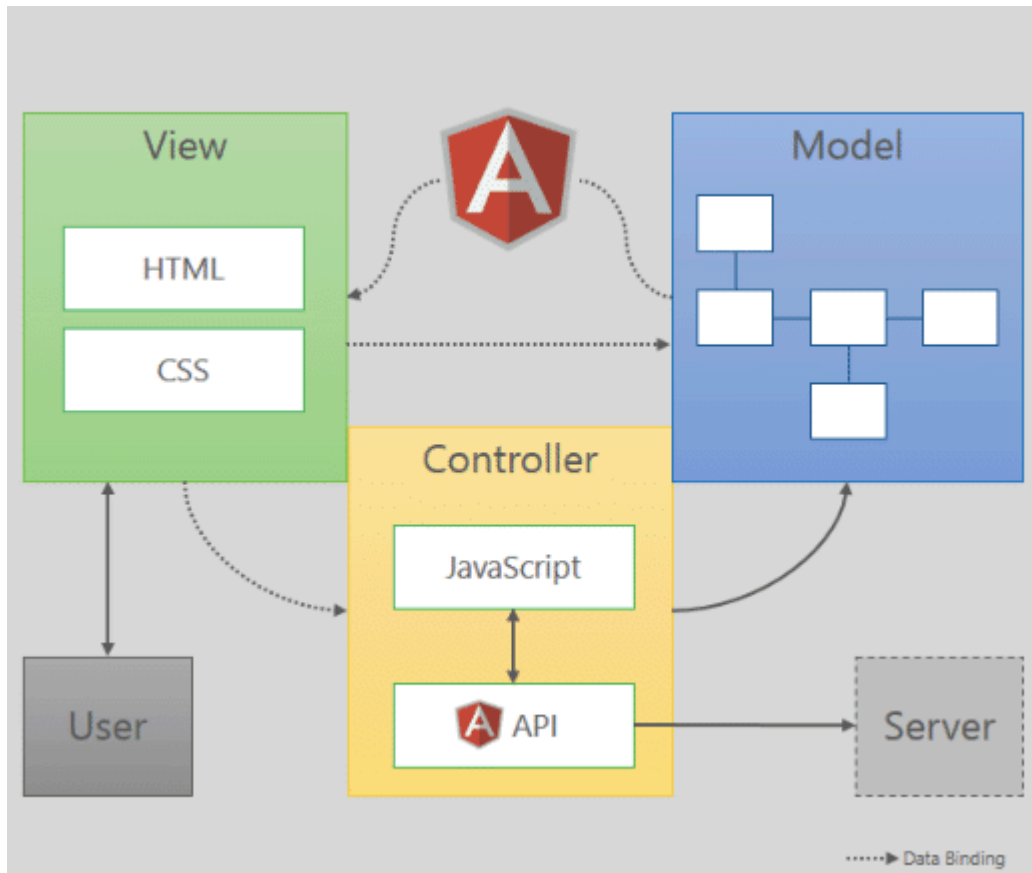
Aplicación Web:

Una aplicación web es un tipo de sitio web avanzado que ofrece funcionalidad interactiva, similar a una aplicación de escritorio o móvil. Las aplicaciones web suelen incluir características como la autenticación de usuarios, interacciones en tiempo real, acceso a bases de datos, y manejo de datos complejos. Utilizan tecnologías como HTML, CSS, JavaScript, y a menudo frameworks y bibliotecas adicionales (por ejemplo, React, Angular, Vue.js en el frontend y Node.js, Django, Ruby on Rails en el backend).

Arquitectura de una aplicación WEB



Patrón de Diseño MVC – MODELO VISTA CONTROLADOR



Modelo:

La lógica y los datos de la aplicación.

Vista:

El HTML y el CSS forman la Vista. El HTML estructura la página su maquetado y diseño responsivo

Controlador:

El JavaScript Maneja las interacciones del usuario, que cosas realiza, selecciona, elige el usuario.