

**UNIVERSIDAD DE LAS FUERZAS ARMADAS**

**ESPE**

**TEMA:**

Proyecto Aplicación Web Progresiva y Responsive.

**ASIGNATURA:**

Fundamentos de Sistemas Web

**ESTUDIANTE:**

Santiago Arroyo, Irving Martínez, Kenneth Cortez y Michael Villacres

**NRC 14335**

**Ing. Doris Chicaiza**

**Sangolquí - Ecuador, 07 de marzo de 2024**

**Tema:** Proyecto Aplicación Web Progresiva y Responsive.

**Objetivo:**

Desarrollar y lanzar una Aplicación Web Progresiva (PWA) y Responsive para Funachi Pizzeria, que proporcione a los clientes una experiencia optimizada en dispositivos móviles y permita el acceso offline para mejorar la accesibilidad.

**Objetivos Específicos**

* Desarrollar un sistema de gestión de pedidos para registrar cada pedido con detalle en una tabla de datos.
* Diseñar e implementar una base de datos que respalde la gestión eficiente de los pedidos, clientes y productos de Funachi Pizzeria.

**Marco Teórico:**

El desarrollo web ha evolucionado significativamente desde los primeros días de la World Wide Web. En la actualidad, la creación de páginas web efectivas y atractivas implica el uso conjunto de varias tecnologías. Entre ellas, HTML (HyperText Markup Language), CSS (Cascading Style Sheets) y JavaScript son fundamentales. Este marco teórico examinará en detalle cada una de estas tecnologías, su propósito, su evolución y cómo trabajan en conjunto para crear experiencias web dinámicas e interactivas.

**HTML: Estructura del Contenido**

**Definición y Evolución**

HTML es el lenguaje de marcado estándar utilizado para la creación y estructuración de contenido en la web. Desarrollado por el World Wide Web Consortium (W3C), HTML ha pasado por varias versiones, desde su introducción en 1991 hasta la versión más reciente, HTML5. A lo largo de su evolución, se han introducido nuevas etiquetas y características para mejorar la semántica y la capacidad de representar información. (W3C, 2022)

**Estructura Básica**

¡La estructura básica de un documento HTML incluye elementos como <!DOCTYPE>, <html>, <head>, y <body>. Estos elementos proporcionan el esqueleto necesario para organizar y presentar el contenido de una página web de manera coherente. (Duckett, 2011)

**Semántica en HTML5**

Con la introducción de HTML5, se ha hecho hincapié en la semántica, permitiendo a los desarrolladores describir más precisamente el contenido de una página. Elementos como <header>, <footer>, <article> y <nav> ofrecen una estructura semántica que mejora la accesibilidad y facilita la comprensión del contenido tanto para humanos como para motores de búsqueda.

**CSS: Estilo y Presentación**

**Definición y Evolución**

CSS, o Cascading Style Sheets, es el lenguaje utilizado para describir la presentación de un documento HTML. Su desarrollo y estandarización han sido liderados por el W3C, y se ha convertido en una herramienta esencial para separar la estructura del contenido de su presentación visual.

**Selectores y Propiedades**

Los selectores en CSS permiten apuntar a elementos específicos en el documento HTML y aplicar estilos. La cascada en Cascading Style Sheets se refiere a cómo se resuelven los conflictos cuando varios estilos se aplican al mismo elemento. Además, las propiedades CSS controlan aspectos como el color, el tamaño, la tipografía y la disposición de los elementos en la página. (EStelle, 2017)

**Layouts y Flexbox**

Con el tiempo, han surgido técnicas avanzadas de diseño en CSS, como Flexbox y Grid, que ofrecen un control más granular sobre la disposición de los elementos en la página. Estas técnicas permiten diseños más flexibles y responsivos, adaptándose a diferentes tamaños de pantalla y dispositivos. (Duckett, 2011)

**JavaScript: Interactividad y Dinamismo**

**Definición y Evolución**

JavaScript es un lenguaje de programación interpretado que permite la interactividad y la manipulación dinámica del contenido en una página web. Desarrollado inicialmente por Netscape, JavaScript se ha convertido en un componente central del desarrollo web, impulsado por estándares como ECMAScript. (MDN Web Docs, 2022)

**DOM y Eventos**

El Document Object Model (DOM) es una representación en memoria de la estructura de un documento HTML, permitiendo a JavaScript interactuar con los elementos de la página. Los eventos, como clics de mouse o cambios en el contenido, desencadenan funciones JavaScript que responden a la interacción del usuario.

**Frameworks y Bibliotecas**

El ecosistema de desarrollo web en JavaScript ha experimentado un crecimiento significativo con la aparición de frameworks como Angular, React y Vue.js, así como bibliotecas como jQuery. Estas herramientas facilitan la creación de aplicaciones web más complejas y proporcionan estructuras organizativas para el código JavaScript.

**Integración y Buenas Prácticas**

**Desarrollo Frontend y Backend**

El desarrollo web a menudo se divide en frontend y backend. Mientras que HTML, CSS y JavaScript son esenciales para el frontend, el backend maneja la lógica del servidor, la base de datos y otras operaciones del lado del servidor. La comunicación entre el frontend y el backend se logra a través de protocolos como HTTP y API RESTful.

**Herramientas de Desarrollo**

Diversas herramientas y entornos de desarrollo facilitan la creación, depuración y optimización de páginas web. Editores de código como Visual Studio Code, sistemas de control de versiones como Git, y plataformas de construcción como Webpack son comunes en el flujo de trabajo de desarrollo web moderno.

**Optimización y Rendimiento**

La optimización del rendimiento es crucial para garantizar una experiencia de usuario fluida. Técnicas como la carga diferida de recursos, la minimización de archivos CSS y JavaScript, y el uso de Content Delivery Networks (CDN) ayudan a acelerar el tiempo de carga de las páginas.

**Tendencias Futuras y Conclusiones**

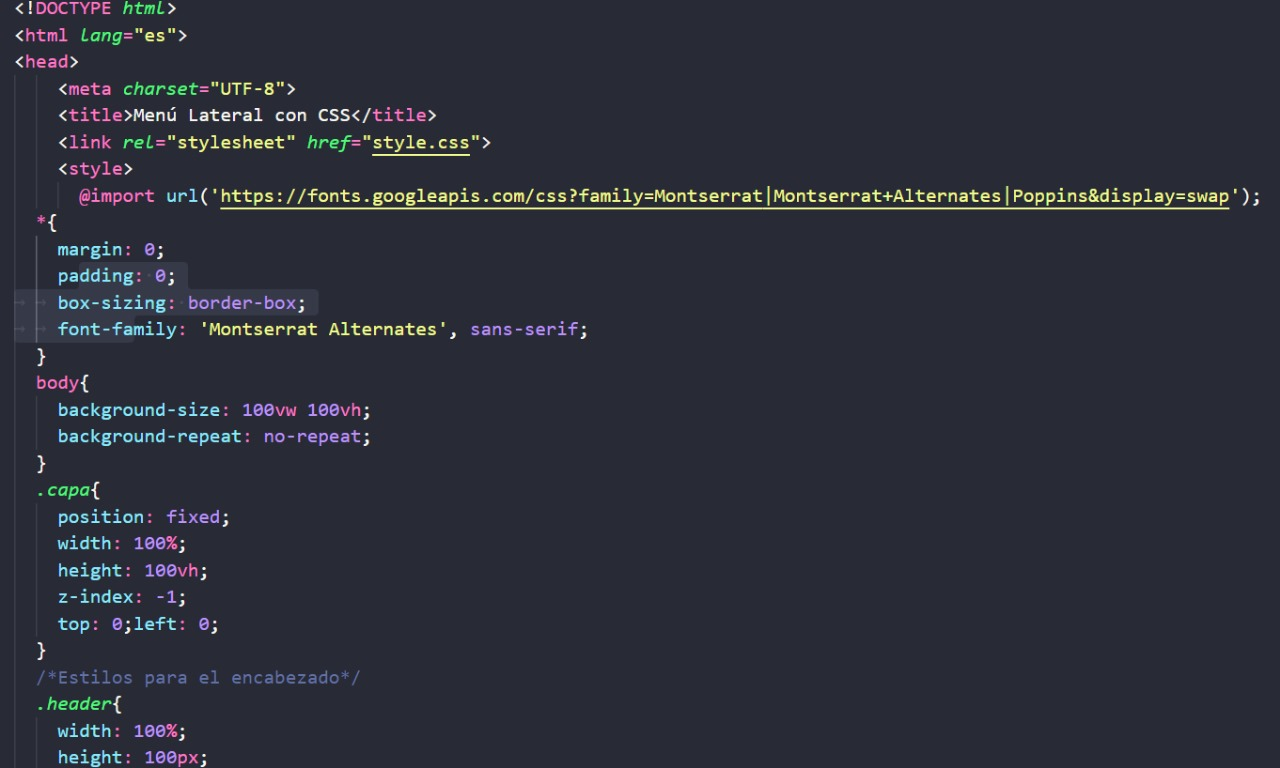
**Avances Tecnológicos**

El desarrollo web seguirá evolucionando con avances tecnológicos emergentes. Tecnologías como WebAssembly, Progressive Web Apps (PWA) y nuevas versiones de HTML, CSS y JavaScript influirán en el panorama del desarrollo web.

**Desarrollo Web con HTML, CSS y JavaScript, en el proyecto Funachi:**

El código creado proporciona una visión detallada de las mejores prácticas de desarrollo web, utilizando HTML, CSS y JavaScript de manera eficiente. A continuación, se presenta un análisis estructurado de los elementos y funciones:

**1. Estructura Básica HTML:**



La base del código inicia con la declaración del tipo de documento y la etiqueta raíz <html>. Se detallan componentes esenciales en las secciones <head> y <body>.

**1.1. Sección <head>:**

**Codificación de Caracteres (<meta charset="UTF-8">):** Establece la codificación UTF-8 para garantizar la correcta representación de caracteres.

**Título de la Página (<title>):** Define el título de la página en la pestaña del navegador, mejorando la identificación del sitio.

**Enlace a Archivo de Estilo CSS (<link rel="stylesheet" href="style.css">):** Referencia a un archivo externo para mantener la presentación visual por separado.

**Inclusión de Fuentes desde Google Fonts**: (@import url('https://fonts.googleapis.com/css?family=Montserrat|Montserrat+Alternates|Poppins&display=swap');): Mejora la estética tipográfica con fuentes personalizadas.

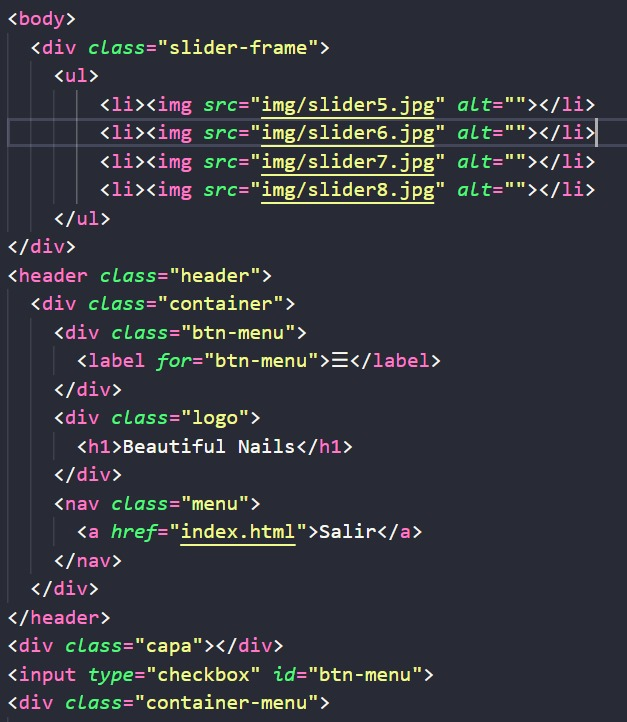
**1.2. Sección <body>:**

**Encabezado (<header>):** Punto focal con elementos importantes como el logotipo o el nombre del sitio.

**Menú Lateral:** Estructura de menú que facilita la navegación y mejora la accesibilidad.

**Tabla de Clientes (<table id="clientes-table">):** Representación visual de datos organizados de manera clara y concisa.

**Formulario de Cambio de Código (<form id="cambio-codigo-form">):** Formulario interactivo para cambiar el código de seguridad con validación en JavaScript.



**2. Enlaces y Fuentes Externas (CSS y JS):**

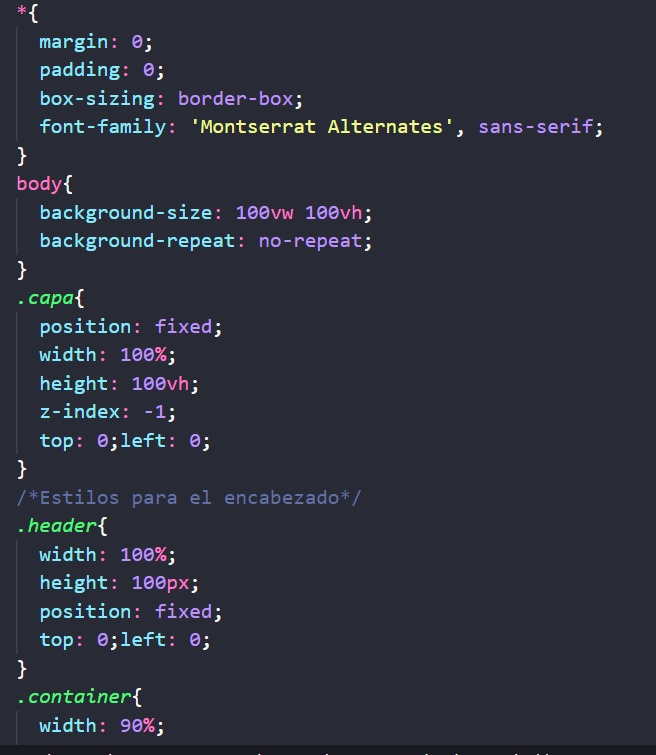
Se destaca el modularidad mediante la incorporación de archivos externos para organizar y facilitar actualizaciones.

**Enlace a Archivo CSS Externo (<link rel="stylesheet" href="style.css">):** Mejora la mantenibilidad y coherencia visual.

**Inclusión de Fuentes Externas de Google (`@import url(...)'):** Referencia a fuentes personalizadas para mejorar la estética tipográfica.

**Incorporación de jQuery (<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>):** Simplificación de la manipulación del DOM y manejo de eventos.

3. Estilos CSS:



La presentación visual se define con estilos CSS estructurados y coherentes.

**@font-face para Fuentes Personalizadas:** Incorpora fuentes distintivas.

**Estilos para Elementos Específicos (header, #clientes-table, #cambio-codigo-form, etc.):** Garantiza una presentación visual coherente.

**JavaScript:**



**1. Función JavaScript para Transiciones:**

La dinámica de la página mejora con funciones JavaScript y jQuery.

**Manejo de Ventana Emergente (<div id="popup" class="popup">):** Ventana emergente para confirmaciones visuales.

**Solicitud AJAX para Eliminación ($.ajax({...}):):** Eliminación de reservas sin recargar la página.

**Control de Tiempo con setTimeout:** Mejora la experiencia del usuario con transiciones suaves.

**2. Validaciones de Entrada en JavaScript:**

Se implementa la función validarCodigo() para garantizar la integridad de los datos del usuario.

**Verificación de Coincidencia de Códigos (if (codigo !== codigoConfirmacion)):** Mensaje de error en caso de discrepancias.

Estos elementos y técnicas proporcionan una sólida estructura para el desarrollo web, mejorando la calidad, seguridad y mantenibilidad de la aplicación.

**Conclusiones:**

**Gestión de Pedidos Efectiva:**

El sistema de gestión de pedidos implementado con HTML y JavaScript ofrece una interfaz intuitiva que facilita la registración detallada de pedidos. La estructura de la tabla de datos permite una organización eficiente y mejora la experiencia tanto para los clientes como para el personal de Funachi Pizzeria.

**Base de Datos para Eficiencia Operativa:**

La implementación de una base de datos respalda de manera efectiva la gestión integral de pedidos, clientes y productos. Esto se traduce en una mejora significativa de la eficiencia operativa al proporcionar acceso rápido y organizado a la información esencial para el negocio.

**Experiencia de Usuario Mejorada:**

La aplicación PWA y responsive cumple con el objetivo de mejorar la accesibilidad y satisfacción del cliente. Al ofrecer una experiencia optimizada en dispositivos móviles y la capacidad de acceso offline, Funachi Pizzeria asegura que sus clientes disfruten de un servicio ágil y adaptado a sus necesidades, incluso en situaciones de conectividad limitada.

**Recomendaciones:**

**Refuerzo de Seguridad y Validación:**

Implementar protocolos de seguridad adicionales, como cifrado HTTPS, para proteger la integridad de los datos. Además, realizar validaciones más exhaustivas en el lado del servidor y del cliente para prevenir posibles vulnerabilidades y garantizar la fiabilidad de la información.

**Notificaciones Push para Información Instantánea:**

Explorar la integración de notificaciones push para mantener a los clientes informados sobre el estado actual de sus pedidos. Este enfoque proactivo mejora la comunicación y la transparencia, brindando una experiencia más completa y satisfactoria.

**Mejora Continua de Responsividad:**

Continuar optimizando la responsividad de la aplicación para adaptarse a una variedad de dispositivos y tamaños de pantalla. Utilizar pruebas en dispositivos móviles y herramientas de desarrollo para garantizar una experiencia consistente, asegurando que la aplicación sea amigable y accesible para todos los usuarios.

# Referencias

Duckett, J. (2011). *duckett*. Obtenido de Html & Css: https://wtf.tw/ref/duckett.pdf

EStelle, M. y. (2017). *O Reilly*. Obtenido de The Definitive Guide. O'Reilly Media.: https://books.google.com.ec/books?id=w2U6DwAAQBAJ&printsec=frontcover&hl=es&source=gbs\_ge\_summary\_r&cad=0#v=onepage&q&f=false

MDN Web Docs. (2022). *MDN Web Docs*. Obtenido de JavaScript: https://developer.mozilla.org/en-US/docs/Web/JavaScript

W3C. (2022). *W3C*. Obtenido de HTML5 - A vocabulary and associated APIs for HTML and XHTML: https://www.w3.org/TR/html52/