

Documento de Realización del Código:

Memorial Web Interactivo

Este documento describe la estructura, las tecnologías implementadas y el funcionamiento del código base para el desarrollo de un memorial web interactivo. Su objetivo principal es servir como guía preliminar para personas externas que deseen comprender la arquitectura y la lógica fundamental del sitio.

Herramientas implementadas:

- **HTML5:** Lenguaje de marcado fundamental para la estructuración y el contenido de las páginas web.
- **CSS:** Hoja de estilos en cascada utilizada para la presentación visual, el diseño y la maquetación del sitio web.
- **JavaScript (JS):** Lenguaje de programación que añade interactividad, dinamismo y manipulación del DOM (Document Object Model) en el lado del cliente.
- **Bootstrap 5:** Framework de CSS de código abierto que proporciona una amplia gama de componentes de interfaz de usuario (UI) predefinidos, un sistema de rejilla responsivo y utilidades para facilitar el desarrollo de sitios web adaptables a diferentes dispositivos.
- **Font Awesome:** Librería de iconos vectoriales escalables que permite integrar fácilmente iconos personalizables en la interfaz del usuario.
- **Firebase:** Plataforma de desarrollo en la nube que ofrece diversos servicios backend, incluyendo una base de datos NoSQL (Cloud Firestore), autenticación y hosting, utilizada en este proyecto para la gestión de datos y la interacción con los usuarios.

Estructura y Diseño:

La estructura fundamental del memorial web interactivo se construyó utilizando HTML5, aprovechando la semántica del lenguaje para organizar el contenido de manera lógica y accesible. Para garantizar una experiencia de usuario óptima en una variedad de dispositivos (computadoras de escritorio, tabletas y dispositivos móviles), se implementó el sistema de rejilla y los componentes responsivos de Bootstrap 5.

Dentro del directorio raíz del proyecto, denominado M_Flores, se encuentran las páginas principales del sitio web en español. Adicionalmente, se han previsto carpetas vacías para la futura incorporación de assets (imágenes, archivos multimedia, etc.) o para la integración de fuentes tipográficas que no puedan ser enlazadas externamente.

Gestión de Estilos (CSS):

Se ha adoptado una estrategia inicial de una única hoja de estilos CSS para centralizar la definición de estilos visuales y facilitar la reutilización de reglas en todo el proyecto. Si el proyecto experimenta un crecimiento significativo en el futuro, se recomienda considerar una arquitectura de hojas de estilos más modular, separando estilos globales de estilos específicos para secciones o componentes particulares.

Componentes Dinámicos (JavaScript):

Para optimizar el mantenimiento y la consistencia de elementos comunes como el pie de página (footer) y el menú de navegación (menu), se ha implementado una técnica de carga dinámica mediante JavaScript. Esto implica que el contenido de estos elementos se define en archivos separados y se inserta en las páginas a través de llamadas de JavaScript. Esta estrategia asegura que cualquier modificación realizada en estos archivos se refleje automáticamente en todo el sitio web, simplificando las actualizaciones y manteniendo la coherencia visual. Las referencias a estos scripts se pueden identificar al final de cada sección de las páginas, marcadas con la etiqueta `<script>`.

Integración de Recursos Externos:

La tipografía utilizada en el sitio web y los iconos se han integrado mediante la librería Font Awesome y la API de Google Fonts. Las referencias a estas bibliotecas se encuentran en la sección `<head>` de cada página HTML, asegurando que estos recursos estén disponibles para la correcta visualización del sitio.

Implementación de Firebase:

La integración con Firebase se realizó siguiendo los siguientes pasos:

1. Creación del Proyecto en Firebase: Se configuró un nuevo proyecto en la consola de Firebase.
2. Obtención de Credenciales: Se obtuvieron las credenciales de configuración del proyecto de Firebase.
3. Integración en el Código HTML: Las credenciales de Firebase se incorporaron directamente en el código HTML para permitir la comunicación con los servicios de Firebase.
4. Conexión mediante JavaScript: Se utilizaron scripts de JavaScript para interactuar con los servicios de Firebase, específicamente con la base de datos Cloud Firestore.
5. Estructura de la Base de Datos: Se creó una base de datos con dos colecciones principales:
 - a. colaboradores: Destinada a almacenar información de personas que desean colaborar con el memorial.

- b. informantes: Diseñada para registrar datos de personas que proporcionan información relevante para el memorial.

Esta estructura de base de datos permite organizar y gestionar eficientemente la información proporcionada por los usuarios a través de formularios, facilitando la búsqueda y el contacto con ellos en función de sus solicitudes o contribuciones.