

Estructura Recomendada para Proyectos HTML, CSS y JavaScript

Introducción

Este documento describe la estructura recomendada para proyectos que utilizan HTML, CSS y JavaScript sin frameworks. Además, se incluye una versión optimizada para alojar el proyecto en GitHub Pages.

1. Estructura Básica del Proyecto

La estructura mínima y clara para un proyecto pequeño es la siguiente:

```
project/  
  index.html  
  assets/  
    css/  
      styles.css  
    js/  
      main.js  
    img/  
  README.md
```

Descripción

| Carpeta/Archivo | Uso |
|-----------------|---------------------------------------|
| index.html | Página principal del sitio. |
| assets/css | Todos los estilos CSS. |
| assets/js | Archivos JavaScript. |
| assets/img | Imágenes, íconos y recursos visuales. |
| README.md | Documentación básica del proyecto. |

2. Estructura Completa para Proyectos Profesionales

Para proyectos más grandes y organizados, se recomienda la siguiente estructura:

```
project/  
  src/  
    html/  
    css/  
      base/  
      components/  
      pages/  
    js/  
      components/  
      services/  
    img/  
  dist/  
  README.md  
  package.json
```

Carpetas importantes

- **src/html**: contiene las páginas HTML del sitio.
- **src/css/base**: variables globales, reset e inicialización.
- **src/css/components**: estilos reutilizables (botones, tarjetas, navbar).
- **src/css/pages**: estilos específicos de cada página.
- **src/js/components**: scripts para sliders, menús, modales, etc.
- **src/js/services**: consumo de API, manejo de datos, funciones auxiliares.
- **dist/**: versión optimizada lista para despliegue.

3. Buenas Prácticas de Organización

Reglas Generales

- Usar nombres en `kebab-case` o `snake_case`.
- Evitar espacios y mayúsculas en nombres de carpetas.
- Mantener código HTML limpio y separado en componentes.
- Agrupar archivos por función y no por tipo cuando el proyecto crezca.
- Utilizar `index.html` como punto de entrada principal.

4. Versión Optimizada para GitHub Pages

GitHub Pages permite alojar páginas estáticas fácilmente. Para una configuración óptima, la estructura recomendada es:

```
project/  
  index.html  
  assets/  
    css/  
    js/  
    img/  
  404.html  
  README.md
```

Puntos importantes para GitHub Pages

- El archivo `index.html` debe estar en la raíz.
- GitHub Pages admite una `404.html` personalizada.
- Evitar usar rutas absolutas: usar rutas relativas como:

```
src="./assets/css/styles.css"
```

- Activar GitHub Pages desde:

Settings → **Pages** → seleccionar rama `main` y carpeta raíz.

- No usar dependencias que requieran servidor (como Node.js) sin compilación previa.

Ejemplo mínimo de index.html compatible

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<link rel="stylesheet" href="assets/css/styles.css">
<title>Mi Proyecto</title>
</head>
<body>
<h1>Hola desde GitHub Pages</h1>
<script src="assets/js/main.js"></script>
</body>
</html>
```

5. Conclusión

Una buena estructura de carpetas facilita el mantenimiento, escalabilidad y publicación del proyecto. Organizar los archivos según su función permite que incluso sitios simples puedan crecer sin volverse difíciles de manejar. GitHub Pages ofrece un excelente entorno para alojar sitios estáticos siempre que se respeten sus convenciones básicas.