

# Sprint 1: Configuración Inicial y Estructura del Proyecto

## (Días 1-4, 20 horas)

### Objetivos principales:

- Configurar el entorno de trabajo (repositorio en GitHub, ramas y flujo de trabajo con Git).
- Crear la estructura básica del sitio web (HTML y CSS).
- Asegurarse de que los alumnos comprendan el flujo de trabajo con Git y GitHub.

### Tareas:

#### 1. Preparación del Proyecto:

- **Clonar el repositorio de GitHub:** Los alumnos deben clonar el repositorio en su máquina local.
- **Configuración de Git:** Los alumnos configuran Git (nombre y correo) y realizan su primer commit de prueba para verificar la correcta configuración.
- **Creación de ramas:** Cada alumno crea una rama con su nombre o la funcionalidad que va a desarrollar (por ejemplo, `feature/nombre-alumno` o `feature/navbar`).

#### 2. Definición de la estructura HTML inicial:

- Crear la estructura básica del sitio con etiquetas semánticas (`<header>`, `<nav>`, `<main>`, `<footer>`).
- Crear las páginas principales: `index.html`, `contacto.html`, `servicios.html`, etc.

#### 3. Primeros estilos CSS:

- Definir un archivo `styles.css` para aplicar los primeros estilos generales (colores, tipografías, etc.).
- Comenzar a aplicar flexbox o grid para una estructura básica responsive.

#### 4. Primer commit y push:

- Hacer un commit de los cambios en la rama personal y enviar (**push**) los cambios a GitHub.
- Crear un **pull request (PR)** para que el profesor revise y apruebe el código.

### Entrega final del Sprint 1:

- Repositorio de GitHub actualizado con la estructura HTML y primeros estilos CSS.
  - Ramas personales creadas y pull requests enviados.
  - Revisión y aprobación del profesor de las primeras implementaciones.
-

## **Sprint 2: Diseño Visual y Responsividad (Días 5-8, 20 horas)**

### **Objetivos principales:**

- Mejorar el diseño visual con CSS.
- Asegurar que el sitio sea completamente responsive en diferentes dispositivos.
- Añadir la barra de navegación funcional.

### **Tareas:**

#### **1. Mejorar el CSS y diseño del sitio:**

- Refinar el diseño general del sitio: colores, tipografías y distribución de elementos.
- Aplicar un sistema de grid o flexbox más avanzado para mejorar el layout.

#### **2. Responsividad:**

- Asegurarse de que el diseño se vea bien en pantallas de diferentes tamaños (móvil, tablet, desktop).
- Utilizar media queries para ajustar los estilos según el tamaño de pantalla.

#### **3. Implementación de la barra de navegación:**

- Crear una barra de navegación común en todas las páginas.
- Asegurarse de que la barra sea responsiva (hamburguesa o menú desplegable para pantallas pequeñas).

#### **4. Commit, push y pull request:**

- Hacer un commit de los cambios realizados, enviar (**push**) los cambios a GitHub y crear un nuevo **pull request** para revisión.

### **Entrega final del Sprint 2:**

- Diseño visual refinado y responsive en todas las páginas.
- Barra de navegación funcional y responsive.
- Pull request con las mejoras enviado y aprobado por el profesor.

## **Sprint 3: Interactividad con JavaScript y Formularios (Días 9-12, 20 horas)**

### **Objetivos principales:**

- Implementar interactividad básica con JavaScript (menús, sliders, etc.).
- Crear formularios funcionales (por ejemplo, un formulario de contacto).
- Validación de formularios y mejoras de accesibilidad.

### **Tareas:**

#### **1. Añadir interactividad con JavaScript:**

- Implementar pequeños scripts en JavaScript para mejorar la experiencia del usuario (por ejemplo, menús desplegables, sliders o efectos al hacer scroll).
- Añadir efectos de hover y transiciones CSS/JS donde sea necesario.

#### **2. Creación de formularios:**

- Crear un formulario de contacto funcional.
- Implementar validación de campos en el frontend (nombre, email, mensaje, etc.).

#### **3. Accesibilidad:**

- Asegurarse de que todos los elementos interactivos (navegación, formularios, botones) sean accesibles mediante teclado.
- Añadir etiquetas ARIA donde sea necesario para mejorar la accesibilidad.

#### **4. Commit, push y pull request:**

- Realizar el commit de los cambios en JavaScript y formularios.
- Hacer push a GitHub y crear un pull request para revisión.

### **Entrega final del Sprint 3:**

- Interactividad básica añadida al sitio web.
  - Formularios funcionales con validación en el frontend.
  - Mejoras de accesibilidad implementadas y pull request aprobado.
-

## Sprint 4: Optimización, Pruebas Finales y Entrega (Días 13-16, 20 horas)

### Objetivos principales:

- Optimizar el rendimiento del sitio web (CSS, imágenes, JavaScript).
- Realizar pruebas en múltiples dispositivos y navegadores.
- Documentación del proyecto y despliegue.

### Tareas:

#### 1. Optimización del rendimiento:

- Comprimir y optimizar imágenes para reducir el tamaño de los archivos.
- Minificar archivos CSS y JavaScript.
- Revisar y optimizar el código para mejorar la velocidad de carga.

#### 2. Pruebas finales:

- Probar el sitio en diferentes dispositivos (móvil, tablet, desktop) y navegadores (Chrome, Firefox, Safari).
- Realizar correcciones de errores detectados durante las pruebas.

#### 3. Documentación del proyecto:

- Actualizar el archivo `README.md` con instrucciones claras sobre cómo ejecutar el proyecto.
- Añadir comentarios en el código donde sea necesario para facilitar su comprensión.

#### 4. Despliegue del proyecto:

- Si es necesario, hacer el despliegue del proyecto en GitHub Pages, Netlify o cualquier otro servicio.

#### 5. Commit final, push y pull request:

- Realizar un commit final con las optimizaciones y correcciones.
- Hacer push a GitHub y crear un **pull request final**.

### Entrega final del Sprint 4:

- Proyecto optimizado y completamente funcional.
- Pruebas en múltiples dispositivos completadas y errores corregidos.
- Pull request final aprobado y proyecto desplegado (si aplica).

---

### Resumen del Proyecto:

- **Duración total:** 16 días (80 horas).
- **Sprints:** 4 sprints de 20 horas cada uno.
- **Repositorio GitHub:** Los alumnos trabajan en sus ramas individuales, y el profesor revisa los pull requests y hace los merges en la rama principal (`main`).
- **Entrega final:** Un sitio web multipágina, responsivo y optimizado, con funcionalidad básica y una estructura bien organizada.