

# Tutorial Detallado de Desarrollo Web para Investigación Educativa

## Introducción

El desarrollo web es un campo amplio y dinámico que permite crear sitios web y aplicaciones interactivas. Este tutorial está diseñado para educadores e investigadores que desean crear y compartir contenido educativo en línea. A continuación, se detallan los conceptos básicos, las tecnologías más utilizadas y los pasos necesarios para crear y desplegar un proyecto web.

## Conceptos Básicos

### Desarrollo Web

El desarrollo web abarca todas las tareas relacionadas con la creación y mantenimiento de sitios web. Incluye aspectos como el diseño, la programación y la gestión de contenidos.

### Diseño Web

El diseño web se refiere a la creación de la apariencia visual y la experiencia del usuario en un sitio web. Incluye la disposición de elementos, la elección de colores, tipografías y otros aspectos estéticos.

### Maquetado Web

El maquetado web es el proceso de convertir un diseño gráfico en un sitio web funcional utilizando lenguajes de marcado y estilos. Es la estructura que sostiene el contenido y el diseño.

### Programación Web

La programación web implica el uso de lenguajes de programación para crear la lógica y la funcionalidad de un sitio web. Esto puede incluir la interacción con bases de datos, la gestión de formularios y la creación de aplicaciones interactivas.

## Tecnologías Básicas

### HTML (HyperText Markup Language)

HTML es el lenguaje estándar para crear y diseñar páginas web. Define la estructura del contenido utilizando etiquetas.

### Ejemplo:

```
wao > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Mi Proyecto Educativo</title>
7  </head>
8  <body>
9    <h1>Bienvenidos a mi proyecto educativo</h1>
10   <p>Este es un ejemplo de página web.</p>
11 </body>
12 </html>
13 |
```

## CSS (Cascading Style Sheets)

CSS se utiliza para describir la presentación de un documento escrito en HTML. Controla el diseño, los colores, las fuentes y la disposición de los elementos.

### Ejemplo:

```
wao > # style.css > ...
1  body {
2      font-family: Arial, sans-serif;
3      background-color: #f0f0f0;
4      margin: 0;
5      padding: 0;
6  }
7
8  h1 {
9      color: #333;
10     text-align: center;
11 }
12
13 p {
14     font-size: 16px;
15     line-height: 1.5;
16 }
17
```

## Bootstrap

Bootstrap es un framework de CSS que facilita el diseño de sitios web responsivos y atractivos. Proporciona componentes y estilos predefinidos.

### Ejemplo:

```
wao > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi Proyecto con Bootstrap</title>
7      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
8  </head>
9  <body>
10     <div class="container">
11         <h1 class="text-center">Bienvenidos a mi proyecto con Bootstrap</h1>
12         <p class="lead">Este es un ejemplo de página web utilizando Bootstrap.</p>
13     </div>
14 </body>
15 </html>
16 <!-- <div class="row">
    -->
```

## Font Awesome

Font Awesome es una biblioteca de iconos vectoriales que se pueden personalizar fácilmente con CSS.

### Ejemplo:

```
wao > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi Proyecto con Font Awesome</title>
7      <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css" rel="stylesheet">
8  </head>
9  <body>
10     <div class="container">
11         <h1 class="text-center">Bienvenidos a mi proyecto con Font Awesome</h1>
12         <p class="lead">Este es un ejemplo de página web utilizando Font Awesome.</p>
13         <i class="fas fa-book"></i>
14     </div>
15 </body>
16 </html>
17 <!-- <div class="row">
```

## JavaScript (JS)

JavaScript es un lenguaje de programación que permite crear contenido dinámico e interactivo en un sitio web.

### Ejemplo:

```
wao > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi Proyecto con JavaScript</title>
7  </head>
8  <body>
9      <h1>Bienvenidos a mi proyecto con JavaScript</h1>
10     <button onclick="saludar()">Haz clic aquí</button>
11     <script>
12         function saludar() {
13             alert('¡Hola! Bienvenido a mi proyecto.');
```

## Orden Histórico en el Desarrollo

1. **HTML:** Creado por Tim Berners-Lee en 1991, es la base de la web.
2. **CSS:** Propuesto por Håkon Wium Lie en 1994, permite separar el contenido de la presentación.
3. **JavaScript:** Desarrollado por Brendan Eich en 1995, añade interactividad a las páginas web.
4. **Bootstrap:** Lanzado por Twitter en 2011, facilita el diseño responsivo.
5. **Font Awesome:** Creado por Dave Gandy en 2012, proporciona iconos vectoriales.

## Proyecto Paso a Paso

### Creación del Proyecto

1. **Instalar Visual Studio Code:**
  - Descarga e instala [Visual Studio Code](#).
  - Sigue las instrucciones de instalación para tu sistema operativo.
2. **Crear una Carpeta para el Proyecto:**
  - Abre Visual Studio Code.
  - Haz clic en File > Open Folder y selecciona una carpeta vacía para tu proyecto. Si no tienes una carpeta, crea una nueva en tu sistema de archivos.
3. **Crear Archivos:**
  - **index.html:** Haz clic derecho en la carpeta del proyecto y selecciona New File. Nombra el archivo index.html.
  - **styles.css:** Repite el paso anterior y crea un archivo llamado styles.css.
  - **script.js:** Repite el paso anterior y crea un archivo llamado script.js.
4. **Escribir el Código:**

#### index.html:

```
wao > index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi Proyecto Educativo</title>
7      <link rel="stylesheet" href="styles.css">
8      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
9      <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css" rel="stylesheet">
10 </head>
11 <body>
12     <div class="container">
13         <h1 class="text-center">Bienvenidos a mi proyecto educativo</h1>
14         <p class="lead">Este es un ejemplo de página web utilizando HTML, CSS, Bootstrap, Font Awesome y JavaScript.</p>
15         <button onclick="saludar()" class="btn btn-primary">Haz clic aquí</button>
16         <i class="fas fa-book fa-3x"></i>
17     </div>
18     <script src="script.js"></script>
19 </body>
20 </html>
21
22
```

styles.css:

```
wao > # style.css > ...
1  body {
2      font-family: Arial, sans-serif;
3      background-color: #f0f0f0;
4      margin: 0;
5      padding: 0;
6  }
7
8  h1 {
9      color: #333;
10     text-align: center;
11 }
12
13 p {
14     font-size: 16px;
15     line-height: 1.5;
16 }
17
18
```

script.js:

```
wao > JS script.js > ...
1  function saludar() {
2      alert('¡Hola! Bienvenido a mi proyecto educativo.');
```

### Visualizar en el Navegador

#### 1. Abrir el Archivo HTML:

- Haz clic derecho en index.html en Visual Studio Code y selecciona Open with Live Server (si tienes la extensión Live Server instalada). Si no tienes la extensión, puedes instalarla desde el Marketplace de Visual Studio Code.
- Alternativamente, puedes abrir el archivo index.html directamente en tu navegador web haciendo doble clic en el archivo en tu sistema de archivos.

### Desplegar en GitHub

#### 1. Crear una Cuenta en GitHub:

- Ve a [GitHub](https://github.com) y regístrate para obtener una cuenta gratuita.
- Completa el formulario de registro con tu información personal y verifica tu cuenta a través del correo electrónico que recibirás.

#### 2. Crear un Nuevo Repositorio:

- Inicia sesión en GitHub.
- Haz clic en el ícono + en la esquina superior derecha y selecciona New repository.
- Nombra tu repositorio, por ejemplo, proyecto-educativo.
- Añade una descripción opcional.
- Selecciona Public para que otros puedan ver tu proyecto.
- Marca la opción Initialize this repository with a README para crear un archivo README.
- Haz clic en Create repository.

### 3. **Subir Archivos a GitHub:**

- En la página del repositorio, haz clic en Upload files.
- Arrastra y suelta los archivos index.html, styles.css y script.js desde tu carpeta del proyecto a la ventana de GitHub.
- Añade un mensaje de confirmación, por ejemplo, "Subida inicial de archivos del proyecto".
- Haz clic en Commit changes.

## **Desplegar en GitHub Pages**

### 1. **Configurar GitHub Pages:**

- En la página del repositorio, haz clic en Settings.
- Desplázate hacia abajo hasta la sección GitHub Pages.
- En Source, selecciona main (o master) y haz clic en Save.

### 2. **Acceder a tu Sitio Web:**

- Después de unos minutos, tu sitio web estará disponible en una URL como <https://tu-usuario.github.io/proyecto-educativo/>.
- GitHub te proporcionará la URL exacta en la sección GitHub Pages de la configuración del repositorio.

### 3. **Compartir tu Sitio Web:**

- Puedes compartir la URL con otros para que puedan ver tu proyecto educativo en línea.
- La URL se puede compartir a través de correo electrónico, redes sociales o cualquier otro medio de comunicación.

## Conclusión

Este tutorial te ha guiado a través de los conceptos básicos del desarrollo web, las tecnologías más utilizadas y los pasos detallados para crear y desplegar un proyecto web educativo. Con estas herramientas y conocimientos, estarás bien equipado para crear y compartir contenido educativo en línea.

Link del repositorio: [https://github.com/york26/Yorbik\\_Rodriguez\\_Portafolio\\_Web](https://github.com/york26/Yorbik_Rodriguez_Portafolio_Web)

Link de la pagina: [https://york26.github.io/Yorbik\\_Rodriguez\\_Portafolio\\_Web/](https://york26.github.io/Yorbik_Rodriguez_Portafolio_Web/)