|  |  |  |  |
| --- | --- | --- | --- |
| **Cargo:** | Docente | | |
| **Nombre:** | M.Sc. David Fabián Cevallos Salas | | |
| **Asignatura:** | Programación Web | | |
| **Carrera:** | Marketing Digital y Comercio Electrónico | **Nivel:** | Cuarto nivel |
| **Estudiante:** |  | | |

**ACTIVIDAD PRÁCTICO EXPERIMENTAL EN EL ENTORNO ACADÉMICO**

**IMPLEMENTACIÓN DE UN SITIO WEB EMPLEANDO EL FRAMEWORK DE TRABAJO BOOTSTRAP**

**1. Objetivos**

* Aprender a utilizar el framework Bootstrap para diseñar y desarrollar un sitio web básico.
* Practicar la implementación de elementos simples como div, row, column, headers, cards, carrusel y video de YouTube en Bootstrap.

**2. Antecedentes/Escenario**

Bootstrap es un framework de trabajo front-end que facilita la creación de sitios web responsivos y modernos. Ofrece una amplia gama de componentes y estilos predefinidos que simplifican el desarrollo web.

**3. Recursos necesarios**

* Computadora con acceso a internet y un navegador web.
* Editor de texto o un entorno de desarrollo integrado (IDE) para escribir código HTML y CSS.
* Material de referencia sobre Bootstrap (documentación, tutoriales, etc.).

**4. Planteamiento del problema**

Se solicita a los estudiantes la implementación de un sitio web básico utilizando Bootstrap. El sitio debe incluir los siguientes elementos simples:

* Encabezado con título y subtítulo.
* Sección de contenido con texto e imágenes utilizando cards.
* Carrusel de imágenes.
* Video de YouTube.

**5. Pasos por realizar**

**5.1. Descargar Bootstrap:**

* Descargar la última versión de Bootstrap desde el sitio oficial.
* Incluir los archivos CSS y JavaScript de Bootstrap en el proyecto.

**5.2. Diseño de la Estructura HTML:**

* Crear un archivo HTML básico con la estructura de la página, incluyendo elementos como <head>, <body>, <header>, <section>, <footer>, etc.
* Utilizar clases de Bootstrap para definir la estructura de la página, como container, row, col, etc.

**5.3. Sección de Contenido con Cards:**

* Utilizar la clase card de Bootstrap para cada elemento de contenido.
* Incluir texto e imágenes dentro de las cards.

**5.4. Carrusel de Imágenes:**

* Utilizar el componente carousel de Bootstrap para crear un carrusel de imágenes.
* Incluir al menos tres imágenes en el carrusel.

**5.5. Video de YouTube:**

* Utilizar el componente embed-responsive de Bootstrap para incrustar un video de YouTube de manera responsiva.

**5.6. Estilización Adicional:**

* Utilizar clases de Bootstrap y estilos personalizados en CSS para mejorar el aspecto visual de la página.

**5.7. Pruebas y Optimización:**

* Probar la funcionalidad y la apariencia de la página en diferentes dispositivos.
* Optimizar el código y los estilos para mejorar el rendimiento y la experiencia del usuario.

**6. Desarrollo**

**61. Creación de la Estructura HTML:**

* Iniciar un nuevo documento HTML y agregar las etiquetas básicas como <html>, <head> y <body>.
* Importar los archivos CSS y JavaScript de Bootstrap dentro de las etiquetas <head>.

**6.2. Diseño del Encabezado:**

* Ponga un encabezado en su página web con h1.

**6.3. Sección de Contenido con Cards:**

* Crear una sección de contenido utilizando la clase container de Bootstrap.
* Utilizar la clase card para cada elemento de contenido, incluyendo texto e imágenes dentro de las cards.
* Organizar las cards en filas y columnas utilizando las clases row y col.

**Implementación del Carrusel de Imágenes:**

* Agregar un carrusel de imágenes utilizando el componente carousel de Bootstrap.
* Incluir al menos tres imágenes en el carrusel y personalizar el estilo según sea necesario.

**Inserción de Video de YouTube:**

* Utilizar el componente embed-responsive de Bootstrap para incrustar un video de YouTube de manera responsiva.
* Copiar el código de inserción proporcionado por YouTube y pegarlo dentro de la etiqueta correspondiente en el HTML.

**Estilización Adicional:**

* Aplicar estilos adicionales utilizando clases de Bootstrap y estilos personalizados en CSS para mejorar la apariencia de la página.
* Probar la página web en diferentes dispositivos para asegurarse de que sea responsiva y tenga un aspecto consistente.

**Pruebas y Optimización:**

* Probar la funcionalidad y la apariencia de la página en diferentes navegadores y dispositivos.
* Optimizar el código y los estilos para mejorar el rendimiento y la experiencia del usuario.