

Bootcamp Programación Básica

Grupo: R6_L2_PG_B_V_JN_G1

Integrantes:

- Lesly Montaña
- Cristian Acosta
- Santiago Marchena

UNIVERSIDAD LIBRE

Barranquilla - Atlántico

2024

Introducción al Proyecto: Sportech - E-Commerce Virtual

El proyecto *Sportech* es el resultado del desarrollo de una tienda en línea de ropa deportiva, creada por un equipo de trabajo utilizando tecnologías web fundamentales como HTML, CSS y JavaScript en su mayoría. El objetivo principal del proyecto fue diseñar y construir una página de e-commerce básica que permitiera a los usuarios navegar por la página principal, viendo una serie de productos deportivos, y agregar artículos al carrito de compras para así simular el proceso de compra.

El proyecto fue realizado utilizando Visual Studio Code como entorno de desarrollo integrado (IDE) y el sistema de control de versiones Git, con la colaboración del equipo a través de GitHub. Además, se hizo uso de **Bootstrap**, un framework de CSS, para mejorar la estética y garantizar que la interfaz fuera responsiva y adaptable a diferentes dispositivos.

A lo largo del proyecto, el equipo trabajó de manera colaborativa, desde la planificación inicial, la creación de wireframes, el desarrollo del frontend con interfaces interactivas, hasta la simulación del backend para manejar las operaciones básicas del carrito de compras de manera local. A pesar de que no se implementó un backend real, se utilizó JavaScript para simular la lógica de funcionamiento y brindar una experiencia completa de usuario.

Este proyecto permitió al equipo aplicar los conocimientos adquiridos en el bootcamp de programación brindado por la universidad libre y conocimientos en desarrollo web, y también permitió mejorar sus habilidades de trabajo en equipo, gestión de código y resolución de problemas. Este es el resultado final es un portal funcional, estructurado que simula una experiencia de compra en línea, demostrando la capacidad del equipo para crear soluciones efectivas y colaborativas en un entorno real de desarrollo.

Planificación y diseño estructurado

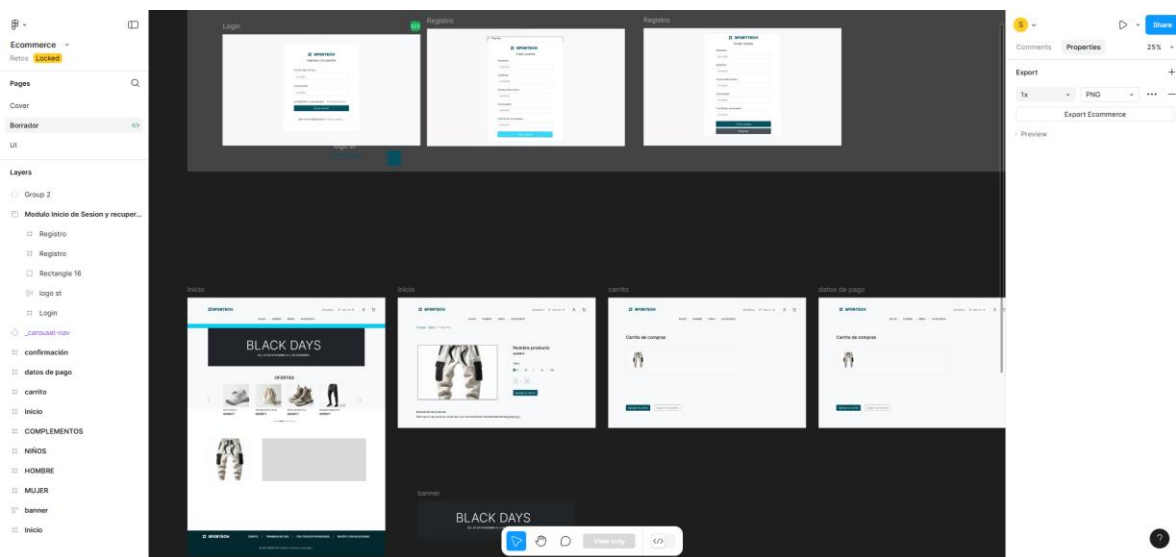
Documentación detallada de requerimientos.

El equipo comenzó el proyecto definiendo los requisitos del sistema. Se estableció que el objetivo principal era desarrollar una tienda en línea de deportes llamada *Sportech*, en la que los usuarios pudieran navegar por los productos, agregar artículos a su carrito de compras, y visualizar un resumen de los productos seleccionados. Los elementos esenciales que debía tener la página fueron:

- **Página de inicio** con una vista general de los productos.
- **Catálogo de productos** donde los usuarios pudieran ver información relevante de cada producto (nombre, imagen, precio).
- **Carrito de compras** que permita agregar y quitar productos, además de ver el total de la compra.
- **Proceso de compra básico**, aunque no se implementaron pagos reales, se simuló el proceso de agregados al carrito.

Wireframes aprobados por el equipo.

Se crearon wireframes en herramientas de diseño como **Figma** para definir la estructura visual y la disposición de los elementos en la página. Los wireframes fueron aprobados por todos los miembros del equipo y establecieron una base clara sobre cómo debería lucir la interfaz, qué secciones incluiría y cómo el usuario interactuaría con el sitio.



<https://www.figma.com/design/3xrnu1ofPbJYf2BXkV6l1N/Ecommerce?node-id=3-3&node-type=canvas&t=bdoaDlch1HsVTDHr-0>

Frontend desarrollado

Interfaces funcionales y estilizadas según los wireframes.

Se desarrolló el frontend utilizando HTML, CSS y Bootstrap. La interfaz se estructuró en varias secciones:

- Una **barra de navegación** con enlaces a las diferentes secciones del sitio.
- **Sección de productos** donde se listaban los artículos, utilizando una cuadrícula responsiva con Bootstrap para que se ajustara correctamente en diferentes dispositivos.
- **Carrito de compras** accesible en todo momento desde un icono en la barra de navegación, que se actualiza dinámicamente cuando el usuario añade o elimina productos.
- **Páginas de detalles** para cada producto, con una vista ampliada de la imagen, descripción y precio.

Se implementaron **estilos responsivos** para garantizar que la página se vea bien en diferentes dispositivos (móviles, tabletas y computadoras de escritorio).

Uso adecuado de tecnologías de frontend.

El equipo usó HTML5 para estructurar el contenido, CSS3 para los estilos y Bootstrap para la creación rápida de una interfaz responsiva y estilizada. Además, se incorporó **JavaScript** para permitir la interacción dinámica, como la actualización en tiempo real del carrito de compras, el movimiento del carrusel y la gestión de la selección de productos.

Estructuración HTML: Se crearon las páginas principales del sitio, como la página de inicio y las páginas de productos, usando HTML. Cada producto tenía su propio bloque con una imagen, descripción y precio.

Estilización con CSS: Se aplicaron estilos CSS para mejorar la estética del sitio. Se configuraron márgenes, colores y tipografía para lograr una apariencia coherente y atractiva. También se personalizaron algunos componentes de Bootstrap para ajustarlos a las necesidades del diseño.

Interactividad con JavaScript: Para la funcionalidad del carrito de compras, se implementó JavaScript. Los usuarios pueden agregar productos al carrito, ver el total de la compra y actualizar la cantidad de los productos. Se utilizó **localStorage** para guardar los elementos del carrito y mantenerlos entre sesiones.

Backend simulado

Creación de la lógica necesaria para el funcionamiento correcto del e-commerce.

Aunque el proyecto no incluyó un backend real (por limitaciones de tiempo y alcance), se simularon algunas funcionalidades de backend con **JavaScript**. Por ejemplo:

- **Gestión del carrito de compras:** se creó un script que guarda el estado del carrito de compras en el almacenamiento local del navegador para que los productos seleccionados persistieran entre las sesiones.
- **Lógica de agregar/eliminar productos del carrito:** se utilizó JavaScript para permitir que el usuario agregue productos, actualice la cantidad o elimine artículos.

Uso adecuado de la lógica aprendida

Se aplicaron conceptos de JavaScript como el manejo de eventos (click, cambios en el carrito), manipulación del DOM para actualizar la interfaz de usuario (mostrar productos en el carrito) y almacenamiento local para simular el almacenamiento de datos entre sesiones.

Lógica de Carrito de Compras: Aunque no se implementó un backend real, el equipo simuló la funcionalidad del carrito mediante JavaScript. Se implementó un sistema de gestión de productos dentro del carrito, donde los usuarios pueden agregar, eliminar y modificar la cantidad de productos seleccionados.

Persistencia de Datos: Los productos seleccionados por los usuarios se almacenaron en el **localStorage** del navegador para mantener la información incluso si el usuario recarga la página.

Integración y colaboración exitosa

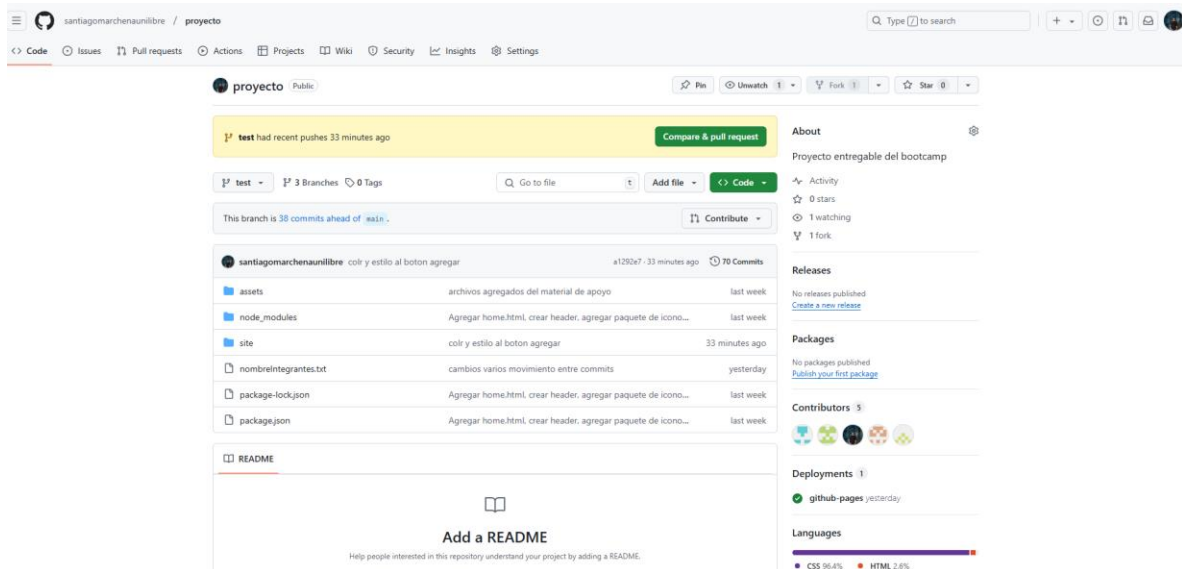
Repositorios organizados y actualizados en GitHub

El equipo usó **Git** para el control de versiones y **GitHub** para alojar el repositorio del proyecto. El repositorio fue organizado en carpetas y ramas, siguiendo una estructura clara:

- **Carpeta raíz:** Contenía los archivos principales del proyecto, como index.html, styles.css, y los scripts.
- **Carpetas:** Separación de las imágenes y recursos estáticos.

Evidencia de commits frecuentes y colaboración entre los miembros del equipo.

Cada miembro del equipo contribuyó con tareas específicas (diseño de la interfaz, lógica del carrito, pruebas de compatibilidad) y realizó **commits frecuentes** con descripciones claras del trabajo realizado. El equipo también mantuvo una **gestión de ramas** para evitar conflictos y facilitar la integración continua. Por ejemplo, se utilizó una rama principal (main) para la versión estable y ramas de características (test/json) para cada nueva funcionalidad o ajuste.



<https://github.com/santiagomarchenaunilibre/proyecto>

Creación de Repositorio: Se creó un repositorio en **GitHub** para almacenar y versionar el código del proyecto. Todos los miembros del equipo contribuyeron al repositorio utilizando Git, lo que permitió un desarrollo organizado y la integración de código sin conflictos.

Manejo de Ramas: Se utilizó una estrategia de ramas para el desarrollo del proyecto. Cada miembro del equipo trabajó en ramas separadas para sus tareas específicas (por ejemplo, diseño, funcionalidad del carrito, pruebas) y luego las integró en la rama principal (main) mediante **pull requests**.

Commits Frecuentes: Durante el proceso de desarrollo, se realizaron **commits frecuentes** para asegurar que los avances fueran registrados y permitir que los miembros del equipo pudieran revisar el progreso de manera continua.

Asignación de Roles

Cada miembro del equipo recibió responsabilidades específicas, como el diseño de la interfaz, la implementación de funcionalidades dinámicas, y la gestión del repositorio en GitHub.

Portal funcional alojado en un servidor o plataforma en línea.

El sitio fue publicado en **GitHub**, donde se cargó el código HTML, CSS y JavaScript. Esto permite al equipo compartir el proyecto con profesores encargados del bootcamp, que pueden interactuar con el código fuente de la página.

Documentación del proceso de desarrollo: Se documentaron los pasos del proceso de desarrollo, incluyendo la planificación inicial, la creación de los wireframes, las decisiones tomadas sobre la arquitectura del frontend y la integración de las funcionalidades. La documentación también cubrió cómo se usaron las tecnologías y los desafíos que se enfrentaron en el camino.

Pruebas de Funcionalidad: Se realizaron pruebas para verificar que el sitio funcionara correctamente, asegurando que las interacciones, como agregar productos al carrito y ver el total, funcionaran como se esperaba.

Pruebas de Responsividad: Se probaron diferentes tamaños de pantalla para garantizar que el sitio fuera completamente funcional y estéticamente atractivo en dispositivos móviles, tabletas y computadoras de escritorio.

Corrección de Errores: Se corrigieron varios errores durante las pruebas, como la alineación incorrecta de productos en dispositivos pequeños.

Conclusión

El proyecto *Sportech* fue una experiencia de aprendizaje para todo el equipo. A través de este proyecto, los miembros adquirieron experiencia en el desarrollo web colaborativo, el uso de tecnologías clave como HTML, CSS, JavaScript y Bootstrap, y la gestión de un proyecto utilizando herramientas como Git y GitHub. La página final es una demostración de la capacidad del equipo para crear una tienda en línea funcional y estética, a pesar de las limitaciones de tiempo y recursos.