

Anteproyecto

Real Fight

Barrera Carmona Alejandro
Proyecto Integrado Ciclo Superior DAW

Descripción del proyecto

Real Fight es una aplicación web innovadora y completa diseñada para revolucionar el mundo del combate, brinda una plataforma unica en la que encontraras el mejor equipo para los deportes de contacto en cuanto a calidad y diseño. En Real Fight podrás adquirir todo tipo de productos dependiendo de tu estilo, tenemos equipo de Grappling/BJJ, Kickboxing/MuayThai, Boxeo y MMA.

Real Fight es una tienda diseñada para dar máxima calidad a los luchadores.

Esta construido con PHP, JS, HTML y CSS.

Requisitos de información

1. Un usuario que no está registrado debe ser capaz de:
 - a. Ver los productos pero no comprarlos.
 - b. Registrarse
2. Un usuario registrado debe ser capaz de:
 - a. Comprar los productos.
 - b. Añadir reseñas en los productos.
 - c. Poder borrar su perfil y/o descargar los datos de cuenta
3. Un usuario autenticado como administrador debe ser capaz de:
 - a. Ver todos los usuarios registrados en el sistema.
 - b. Añadir, modificar o eliminar las novedades de nuestro sistema.
 - c. Ver un dashboard con información relevante de la aplicación:
 - i. El producto con mas ventas.
 - ii. La seccion de lucha mas visitada (Grappling o Striking).
 - iv. Ticket de la venta junto al correo electrónico del cliente.

Requisitos no funcionales

1. Las contraseñas del sistema deberán estar cifradas mediante SHA-256.
2. Cuando se elimine algún producto o diseño del sistema se pedirá confirmación.

Descripción General del Proyecto

El proyecto consiste en una tienda en línea para **RealFight**, una promotora de MMA, donde los usuarios pueden explorar, seleccionar y comprar productos relacionados con deportes de combate. La tienda ofrece diversas categorías, como **Striking**, **Grappling** y **Equipamiento**, y proporciona una experiencia de usuario intuitiva y moderna. Los usuarios pueden:

- Navegar por diferentes categorías de productos y ver los detalles de cada artículo.
- Añadir productos a un carrito de compras y revisar el carrito en cualquier página de la tienda.
- Completar el proceso de compra mediante un formulario de tramitación de pago que recopila su información personal, de envío y de pago.

Solución Tecnológica

Para el desarrollo de esta aplicación, hemos optado por una arquitectura cliente-servidor utilizando **PHP**, **HTML**, **CSS**, y **JavaScript**. A continuación, se detallan los lenguajes y tecnologías empleadas:

1. Front-End (Cliente):

- **HTML5 y CSS3:** Para estructurar y estilizar la interfaz de usuario de la tienda. La estética será moderna y juvenil, utilizando una paleta de colores oscuros (negro, blanco y gris).
- **JavaScript:** Se utilizará para la interacción del usuario, por ejemplo, para gestionar el carrito de compras en tiempo real, realizar validaciones de formulario en el cliente y mejorar la experiencia de usuario.
- **Font Awesome:** Para iconos como el del carrito, haciendo que la interfaz sea más visualmente atractiva y clara.

2. Back-End (Servidor):

- **PHP:** Servirá como lenguaje principal para procesar las solicitudes del cliente, manejar la lógica de negocios y comunicarse con la base de datos. PHP se encargará de gestionar las sesiones de los usuarios, el procesamiento de los formularios de pedido y la lógica para almacenar la información de cada compra.
- **MySQL:** Almacena la información relevante de productos, usuarios y pedidos en una base de datos relacional. MySQL es ideal para este tipo de aplicaciones por su compatibilidad con PHP y su eficiencia en el manejo de datos estructurados.

3. Arquitectura de la Aplicación:

- **Arquitectura Cliente-Servidor:** El cliente (navegador web) envía solicitudes al servidor, donde se encuentra el código PHP. El servidor procesa las solicitudes, interactúa con la base de datos MySQL cuando sea necesario y devuelve el contenido dinámico al cliente.
- **Modelo Vista Controlador (MVC):** Aunque será una implementación sencilla, PHP separará la lógica del negocio de la presentación. Este enfoque organiza mejor el código y facilita el mantenimiento.

Flujo del Proyecto y Esquema Cliente/Servidor

1. Navegación en la Tienda (Cliente)

- El usuario puede acceder a la tienda a través de una **página de inicio** o **menú principal**, donde se presentan las diferentes categorías de productos.
- Cada categoría tiene una página de productos individuales con la opción de agregar artículos al carrito. Al hacer clic en un producto, el usuario es redirigido a una página de detalles del producto.

2. Gestión del Carrito (Cliente)

- Los productos seleccionados se agregan a un carrito de compras gestionado con JavaScript en el cliente, permitiendo que los usuarios vean el contenido del carrito en tiempo real.

3. Tramitación de Pago (Cliente y Servidor)

- Al hacer clic en el icono del carrito (visible en todas las páginas excepto en `checkout.php`), el usuario es redirigido a la página de **tramitación de pago**.
- En `checkout.php`, el usuario completa un formulario de pedido con sus datos personales, dirección y datos de pago. Este formulario se valida en el cliente y luego se envía al servidor para su procesamiento.

4. Procesamiento del Pedido (Servidor)

- En el servidor, PHP recibe la información enviada desde `checkout.php`, valida los datos, los formatea y guarda la información del pedido en la base de datos.
- Al completarse el pedido, el servidor envía una respuesta al cliente, confirmando el éxito de la compra y mostrando un resumen del pedido.

5. Estructura Cliente-Servidor

- **Cliente:** Se encarga de la interfaz visual y de la lógica del carrito de compras en tiempo real. Las páginas están estructuradas en HTML y estilizadas en CSS, con interacción mediante JavaScript.
- **Servidor:** Procesa las solicitudes, valida los datos y los gestiona con PHP, accediendo a la base de datos para registrar los pedidos.

- **Base de Datos (MySQL):** Contiene toda la información estructurada, como productos, usuarios y pedidos. PHP interactúa con MySQL para insertar y obtener datos en función de las solicitudes del cliente.