

# SportMart

Trabajo Fin de Ciclo – Desarrollo de Aplicaciones Web

Pablo Naranjo Cid

Jorge Burgos Barrera

Josep Serrano Rayó

**Universidad Europea de Madrid**



## ÍNDICE

1. [Introducción](#)
2. [Módulos formativos aplicados en el trabajo](#)
3. [Herramientas/Lenguajes utilizados](#)
4. [Componentes del equipo y aportación realizada por cada estudiante](#)
5. [Fases del proyecto](#)
6. Conclusiones y mejoras del proyecto
7. Bibliografía
8. Anexos

## INTRODUCCIÓN

En este documento, se presenta un detallado análisis del desarrollo que se ha llevado a cabo para realizar este proyecto.

El proyecto ha sido concebido con la finalidad de ofrecer todo tipo de recursos, como pueden ser accesorios, ropa, calzado..., enfocados a tres disciplinas deportivas que hoy por hoy son de las más demandadas por los usuarios, como son, el gimnasio, *running*, y Boxeo/Artes marciales.

Esta plataforma de comercio electrónico surge de la imaginación que tenían los desarrolladores que han llevado a cabo esta plataforma, en su vida personal, de tener un sitio (en el hipotético caso de hacer este proyecto realidad) en el que el usuario/deportista, tuviera unas garantías de calidad y precio todas unidas en un mismo sitio.

En un contexto donde el deporte y la actividad física cada vez tiene más importancia en la vida de las personas, se observa una creciente demanda por accesorios, ropa, calzado, que no solo sean de buena calidad y con tecnología avanzada (como pueden ser las zapatillas), sino que también se busca la tendencia en la moda deportiva.

Para comenzar, el usuario verá al entrar en nuestra aplicación, una página principal donde podrá encontrar todo tipo de artículos relacionados con los tres deportes que anteriormente hemos comentado. En esta *home*, no habrá nada filtrado por deportes, sino que las novedades que se quieren mostrar al usuario se verán reflejadas en esta página principal.

En este primer vistazo, algo que puede llamar la atención al usuario final, es que es posible cambiar de idioma. Por el momento solo hay implementado unos pocos idiomas, pero se han elegido en base a los clientes potenciales que podrían surgir, como pueden ser compradores de Francia, Portugal y el inglés como idioma global.

Se tiene un menú de navegación totalmente funcional en el que fácilmente y con un flujo bastante intuitivo, el usuario puede moverse por toda nuestra página y encontrar fácilmente el artículo que concretamente estaría buscando.

Si el usuario ha encontrado algún artículo que le puede haber llamado la atención, puede guardarlo en el carrito de la compra, donde posteriormente podría realizar el pago de todos los artículos que tiene seleccionado. Pero si no ha iniciado sesión, cuando el usuario quiera introducir en el carro de compra un producto, directamente le abrirá el menú de inicio de sesión, en el que fácilmente si no tiene cuenta, podrá acceder al registro de usuario.

Todo ello, incluyendo todo tipo de apartados como pueden ser, acerca de nosotros, contacto, políticas de privacidad, preguntas frecuentes...

El equipo también se ha preocupado en incluir las nuevas formas de pago emergentes como podría ser PayPal.

En resumidas cuentas, SportMart pretende ser un líder de su sector y un gran competidor con otras marcas las cuáles suelen sesgar sus deportes y se suelen centrar en uno solo, o como actualmente está ocurriendo, centrarse en moda deportiva o como se le conoce ahora *streetwear*.

## MÓDULOS FORMATIVOS APLICADOS EN EL TRABAJO

### Entorno Servidor

Este módulo en específico ha sido de gran ayuda en cuanto a la creación de la base de datos, la conexión con el servidor local, los *inserts* en la base de datos, y, sobre todo, las peticiones/consultas que se realizan de las tablas, para en consecuencia, luego mostrarlas y tratar con esos datos que se han recogido. A continuación, se comentan punto por punto las funcionalidades que se han realizado a través de este módulo:

#### 1. Diseño de la base de datos:

- El planteamiento y la planificación realizada al diseñar la base de datos ha sido un trabajo meticuloso, ya que, a futuro, si no se hacía bien dicha planificación, podría traer problemas al trabajar con distintas tablas a la vez. Se ha puesto mucha atención en las relaciones, claves primarias, secundarias y atributos necesarios para almacenar la información de la manera más eficiente y escalable posible. Para garantizar la integridad y consistencia de los datos, la normalización de las tablas ha jugado un papel importante a la hora del diseño de la base de datos. A continuación, un ejemplo de cómo es el diseño de la base de datos:

```
"CREATE DATABASE sportmart;"
"USE sportmart;"
"CREATE TABLE CLIENTE (
  id_cliente INT AUTO_INCREMENT,
  nombre VARCHAR(255) NOT NULL,
  apellidos VARCHAR(255) NOT NULL,
  correo VARCHAR(255) NOT NULL,
  password VARCHAR(255) NOT NULL,
  PRIMARY KEY (id_cliente)
);"
"CREATE TABLE PERFIL_CLIENTE (
  id_cliente INT,
  fecha_nac_cliente DATE,
  miembro_desde DATE,
  telefono VARCHAR(9),
  provincia VARCHAR(255),
  localidad VARCHAR(255),
  direccion_envio VARCHAR(255),
  codigo_postal VARCHAR(255),
  PRIMARY KEY (id_cliente),
  FOREIGN KEY (id_cliente) REFERENCES CLIENTE(id_cliente) ON DELETE CASCADE
);"
"CREATE TABLE PRODUCTO (
  id_producto INT AUTO_INCREMENT,
  nombre VARCHAR(255) NOT NULL,
  descripcion TEXT,
  precio DECIMAL(10, 2) NOT NULL,
  imagen VARCHAR(255),
  sexo ENUM('Hombre', 'Mujer', 'Unisex') NOT NULL,
  PRIMARY KEY (id_producto)
);"
```

```

"CREATE TABLE VARIANTE (
  id_variante INT AUTO_INCREMENT,
  id_producto INT,
  talla VARCHAR(50),
  PRIMARY KEY (id_variante),
  FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto)
);",
"CREATE TABLE CARRITO (
  id_carrito INT AUTO_INCREMENT PRIMARY KEY,
  id_cliente INT,
  id_producto INT,
  cantidad INT NOT NULL,
  talla VARCHAR(50), -- Columna para almacenar la talla del producto
  FOREIGN KEY (id_cliente) REFERENCES CLIENTE(id_cliente),
  FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto)
);",
"CREATE TABLE CATEGORIA (
  id_categoria INT AUTO_INCREMENT,
  categoria VARCHAR(150) UNIQUE,
  PRIMARY KEY (id_categoria)
);",
"CREATE TABLE PRODUCTO_CATEGORIA (
  id_producto INT,
  id_categoria INT,
  PRIMARY KEY (id_producto, id_categoria),
  FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto),
  FOREIGN KEY (id_categoria) REFERENCES CATEGORIA(id_categoria)
);",
"CREATE TABLE DEPORTE (
  id_deporte INT AUTO_INCREMENT,
  deporte VARCHAR(150) UNIQUE,
  PRIMARY KEY (id_deporte)
);",

```

```

"CREATE TABLE PRODUCTO_DEPORTE (
  id_producto INT,
  id_deporte INT,
  PRIMARY KEY (id_producto, id_deporte),
  FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto),
  FOREIGN KEY (id_deporte) REFERENCES DEPORTE(id_deporte)
);",
"CREATE TABLE PEDIDO (
  id_pedido INT AUTO_INCREMENT,
  id_cliente INT,
  direccion VARCHAR(255),
  fecha_pedido TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  total DECIMAL(10, 2) NOT NULL,
  PRIMARY KEY (id_pedido),
  FOREIGN KEY (id_cliente) REFERENCES CLIENTE(id_cliente)
);",
"CREATE TABLE DETALLES_PEDIDO (
  id_detalle INT AUTO_INCREMENT,
  id_pedido INT,
  id_producto INT NOT NULL,
  cantidad INT NOT NULL,
  precio DECIMAL(10, 2) NOT NULL,
  talla VARCHAR(50),
  PRIMARY KEY (id_detalle),
  FOREIGN KEY (id_pedido) REFERENCES PEDIDO(id_pedido) ON DELETE CASCADE,
  FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto)
);",

```

## 2. Conexión con la base de datos:

- La conexión con la base de datos se ha implementado mediante una clase a través de PHP, la cual está dedicada solo para realizar dicha conexión y también la desconexión. Esta clase ha ayudado a la hora de la configuración, ya que llama a una función, en la que antes de que se ejecute, se implementan credenciales (servidor, usuario, contraseña) que se necesitan para acceder a la base de datos.

```
class BaseDeDatos
{
    private $servidor;
    private $usuario;
    private $password;
    private $conexion;
    public function __construct($servidor = "localhost", $usuario = "root", $password = "")
    {
        $this->servidor = $servidor;
        $this->usuario = $usuario;
        $this->password = $password;
    }
    function conectar()
    {
        $this->conexion = mysqli_connect($this->servidor, $this->usuario, $this->password);
        // Lanzo excepción si la conexión no es exitosa
        if (!$this->conexion) {
            throw new Exception("Conexión fallida: " . mysqli_connect_error());
        }
        // Si no da error retorna true
        return true;
    }
    function cerrar()
    {
        // Verifica si la conexión existe antes de cerrarla
        if ($this->conexion) {
            mysqli_close($this->conexion);
        }
    }
    function getConexion()
    {
        return $this->conexion;
    }
    // Cambio de contexto a la base indicada
    function seleccionarContexto($base)
    {
        mysqli_select_db($this->conexion, $base) or die(mysqli_error($this->conexion));
    }
    // Comprueba si una consulta da resultados
    function comprobarExistencia($consulta)
    {
        $result = mysqli_query($this->conexion, $consulta) or die(mysqli_error($this->conexion));
        return (mysqli_num_rows($result) > 0) ? true : false;
    }
}
```

*Clase que realiza la conexión con la base de datos.*

- El código utilizado en esta clase permite establecer una conexión segura y proporciona métodos para verificar la existencia de los resultados en las consultas. De este modo, se asegura una interacción positiva y segura entre la plataforma y la base de datos creada. Además, implementa medidas adicionales para optimizar el rendimiento y proteger la integridad de los datos almacenados.
- ## 3. Peticiones al servidor mediante sentencias SQL:

- Las peticiones al servidor mediante sentencias SQL, han sido clave en la aplicación, ya que la página se basa en el comercio electrónico y se deben gestionar cosas como, añadir al carro de compra los artículos, mostrar dinámicamente los productos en la base de datos a través del filtrado...
- Todas las peticiones al servidor se han realizado mediante sentencias preparadas, de esta manera, se pueden evitar riesgos de seguridad como puede ser la inyección de código mediante SQL. Se ha utilizado también funciones *fetch* para hacer solicitudes y obtener recursos del servidor web, utilizando promesas.

```
"INSERT INTO CLIENTE (nombre, apellidos, correo, password) VALUES
('Carlos', 'Hernández', 'carlos@gmail.com', '\$2y\$10\$pRjIBhX6la2/zZ7VXn8eiusEc6Bimf0S0H5voQAC4ymRTKHic/QUK'),
('Ana', 'Fernández', 'ana@outlook.com', '\$2y\$10\$c6SLfnhuIvHgZ6qM3w/m/.zVLXcmrIfbwcDNuJlu.GDXdJTaHEA66'),
('Luis', 'García', 'luis@gmail.com', '\$2y\$10\$Hlmyy8SbpnFag/80SPLdp0rUYmID9n7Y4JHW2ao0PLELzdvYSj4Fi'),
('Elena', 'Torres', 'elena@outlook.com', '\$2y\$10\$tsFVuxv6tIV.cQnpKK4DB.1qNeY.1kC6MqrFr2k4x0JsJqmvXjLfy'),
('Miguel', 'Ramírez', 'miguel@gmail.com', '\$2y\$10\$I6F6mIxDPWZIC85uSQ5yeejz2xu/tHmkEg9n7k6E0khh8JXrnkyR0'),
('Lucía', 'Jiménez', 'lucia@outlook.com', '\$2y\$10\$5aLDVLRrPdq1AHbArFeq6.h/pnI1P6oHkN73QzCacmR7n9fFua4cS'),
('Javier', 'Moreno', 'javier@gmail.com', '\$2y\$10\$9mfF0PJZ/5C..kxEp3uJt.6YowSzA4izZ7a64.a5kb6PhS/cm.iDu'),
('Sofía', 'Ruiz', 'sofia@outlook.com', '\$2y\$10\$eJ5DW1Q0xyL.91/EmPBobuLVyPbQkDP8M0pWVqrt0Q.7zVWQ0oAVK'),
('Alberto', 'Molina', 'alberto@gmail.com', '\$2y\$10\$TgFnCd8652cBz0oUyrUN0.kkA6KB.ZL7xM9us62nfJTHjbZvz8Z.y'),
('Sara', 'Navarro', 'sara@outlook.com', '\$2y\$10\$Fa87prkVZ6ruH6.D/Hiu60pv6FgtLvZtdoecYZ0mpKHhL6dW9q9/q');"
```

```
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $categoria);
$stmt->execute();
$result = $stmt->get_result();
```

*Ejemplo de sentencias preparadas.*

#### 4. Sesiones de usuario:

- Se ha implementado un sistema de sesiones mediante PHP, que guarda toda la información del usuario, para así, mostrar su perfil con todos sus datos, como puede ser el nombre, teléfono, dirección de envío...
- Con este sistema, es capaz de guardar la información, por ejemplo, de los productos que tiene el usuario que ha iniciado sesión en su propio carrito.

```
$_SESSION['id_cliente'] = $id_cliente;
```

### Entorno Cliente

Este módulo mejora significativamente la experiencia de usuario, muy importante para que los usuarios quieran mantenerse en la página y guiarlos por todo el contenido sin perderse y orientarlos si, por ejemplo, no han introducido bien un campo en un *input* y otras funcionalidades implementadas.



## 1. Validación de formularios:

- De manera dinámica, se capturan los errores a la hora de que el usuario introduzca datos en los campos del formulario, como puede ser, por ejemplo, en el registro. En este caso, si el correo electrónico no tiene una @ y un punto, saltará un error en rojo, dándole *feedback* del error que está teniendo.
- También en esta valoración de formularios, se contemplan errores en la contraseña, que debe tener un número definido de caracteres, mayúsculas y minúsculas, un número y un carácter especial. De la misma manera que en el anterior, si hay algo que no se cumple, se mostrará un mensaje de error indicando al usuario el problema que está teniendo.
- Por último, en cualquier apartado que deba tener solo letras, como puede ser el nombre y apellido, está capturando que el usuario no meta números u otros caracteres y solo pueda introducir letras.
- Implementando la validación de estos formularios, se mejora la experiencia de usuario y se mantiene el tiempo que el usuario está en la página, evitando así, que se vaya y pueda irse a otra página/aplicación de la competencia.

```
function validarNombre(nombre) {
  return /\d/.test(nombre.value) ? false : true;
}
function validarTelefono(telefono) {
  return /^d{9}$/.test(telefono.value);
}
function validarCorreo(correo) {
  return /^[^\s@]+@[^\s@]+\.[^\s@]+$/ .test(correo.value);
}
function validarContraseña(contraseña) {
  return /^(?=.*[a-z])(?=.*[A-Z])(?=.*\d){8,}$/.test(contraseña.value);
}
```

*Pequeño ejemplo de funciones para validar formularios.*

## 2. Funciones fetch para comunicación con el servidor:

- Se ha implementado la función *fetch* de JavaScript, ya que facilita la comunicación con el servidor y de manera muy intuitiva, las peticiones que se le hacen. Se ha decidido utilizar *fetch* por delante de *XMLHttpRequest*, ya que está un poco más en desuso, y se consideró que *fetch*, hace lo mismo, pero mucho más legible y eficiente.

- Las peticiones se ejecutan de manera asíncrona al servidor, que su respuesta es una promesa. De esta manera se optimiza la página, y la obtención y actualización de los datos que se quieren procesar, sin necesidad de recargar la página ya que lo está haciendo en “segundo plano”.

```
fetch('php/insert_cart.php', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/x-www-form-urlencoded',
  },
  body: `id_cliente=${id_cliente}&id_producto=${id_producto}&cantidad=${cantidad}&talla=${talla}`
})
```

*Ejemplo de función fetch para la llamada a insert\_cart.php*

### 3. Botones:

- Los botones de la página han sido implementados con funcionalidades, como puede ser, añadir un producto al carrito, mostrar el inicio de sesión, mostrar el registro de usuario entre otras tantas funcionalidades.
- Se le asigna mediante JavaScript dichas funciones para mejorar la accesibilidad y fluidez en la página.

```
document.getElementById("show-regis").addEventListener("click", function() {
  if (this.textContent === "Cerrar sesión") {
    // Realizar el logout
    fetch('./php/procesar_logout.php')
      .then(response => response.json())
      .then(data => {
        if (data.success) {
          // Recargar la página después de cerrar sesión
          window.location.reload();
        }
      })
      .catch(error => console.error('Fetch error: ', error));
  } else {
    // Si el usuario no ha iniciado sesión, mostrar el formulario de registro
    document.querySelector(".popup-regis").classList.add("active2");
    document.querySelector(".popup").classList.remove("activa");
    document.querySelector(".menu-ppal").classList.remove("active")
  }
});
```

*Función para el botón de registro.*

### 4. Script con llamadas a APIs:

- Se han implementado scripts para realizar llamadas a diversas APIs, con el objetivo de mejorar la funcionalidad y la experiencia del usuario en la plataforma. A continuación, se detallan dos ejemplos: uno para la integración con Google Translate y otro para un botón de pago de PayPal.

- Google Translate:
  - Se ha implementado un script para cargar Google Translate de manera asíncrona. Esta integración permite traducir la página a varios idiomas sin necesidad de recargar, mejorando la accesibilidad y la usabilidad para usuarios de diferentes regiones.
- PayPal:
  - Se ha implementado un script para agregar un botón de PayPal, permitiendo a los usuarios realizar pagos de manera segura y eficiente. El botón se configura con estilo personalizado y se integra de forma asíncrona para no afectar el rendimiento de la página.

```

<script>
function loadGoogleTranslate() {
  var script = document.createElement('script');
  script.type = 'text/javascript';
  script.async = true;
  script.src = 'https://translate.google.com/translate_a/element.js?cb=googleTranslateElementInit';
  document.head.appendChild(script);
}

function googleTranslateElementInit() {
  new google.translate.TranslateElement({
    pageLanguage: 'es',
    includedLanguages: 'en,es,pt,it,fr,de',
    layout: google.translate.TranslateElement.InlineLayout.SIMPLE
  }, 'google_translate_element');
}

document.addEventListener("DOMContentLoaded", function() {
  loadGoogleTranslate();
});
</script>

```

*Ejemplo de llamada a API de Google.*

## Diseño de interfaces web

Con este último módulo, se ha adoptado un enfoque integral para diseñar una página web que sea tanto visualmente atractiva como funcional para cualquier tipo de usuario. Se ha empleado principalmente *Figma* como herramienta principal para llevar a cabo lo que se describe a continuación, lo que ha facilitado el proceso de diseño y ha hecho que sea más intuitivo posteriormente crear y plasmar todos los aspectos considerados. A continuación, se detallan las etapas seguidas en el proceso de diseño de nuestra plataforma:

1. Análisis de la competencia y posibles usuarios:
  - Se realiza un análisis exhaustivo de la competencia para identificar las tendencias del mercado, las fortalezas de los competidores y,

especialmente, sus deficiencias. Esto permite diferenciarse y crear una página más atractiva para los usuarios. Además, se observa lo que los competidores hacen bien y se toma como referencia aquellas prácticas exitosas.

- Se lleva a cabo un análisis detallado de los usuarios potenciales para identificar sus preferencias, comportamientos y necesidades. Al enfocarse en estas preferencias, se busca garantizar que las propuestas y requisitos superen las expectativas, ofreciendo algo que la competencia no puede. Asimismo, se intenta abarcar los deportes más demandados últimamente y trabajar con la mayor calidad del mercado.

## 2. Mockup:

- Dado el poco tiempo que se ha tenido para realizar este trabajo, se decidió hacer solo el *mockup* en vez de hacer primero el *wireframe*.
- Para la realización del *mockup*, se ha utilizado la plataforma Figma, ya que se consideró que es una herramienta muy potente y versátil, para poder hacer los diseños lo más parecido al producto final que se quiere tener.
- Se han tenido en cuenta las leyes de UX:
  - John Yablonski:
  - Efecto estático-usabilidad.
  - Ley de Hick.
  - Ley de Pragnanz.
  - Ley de Miller.
  - Efecto posición en serie.
  - Efecto Von Restorff.
  - Usabilidad y feedback al usuario.

## 3. Creación del logo:

- La creación del logo no presentó grandes dificultades, ya que la idea estaba clara desde el principio. Al ser una página de comercio electrónico, no se quiso dar demasiada importancia al logo. El objetivo era mantener un diseño sencillo y efectivo que no distrajera a los usuarios de la funcionalidad principal de la plataforma, permitiendo que la experiencia de compra fuera el centro de atención.

## 4. Sistema de diseño:

- Se implementó un sistema de diseño, de manera, como con el logo, no distrajera al usuario del objetivo final que se quiere como página de venta electrónica, y es que el usuario compre.
- Se eligieron colores “planos” en una escala de grises, dándole así importancia a las tarjetas de productos u otras secciones que se quiere que vaya a ver el usuario.
- Este enfoque asegura una apariencia coherente en todas las páginas, lo que facilita la navegación y la experiencia del usuario, además de fortalecer la identidad de la plataforma.
- Se ha valorado los colores de feedback:
  - *Success*
  - *Information*
  - *Warning*
  - *Error*

#### 5. Layout Responsive:

- Se han utilizado técnicas de diseño responsivo mediante el uso de Flexbox y Grid, permitiendo así que la web sea capaz de verse en todo tipo de dispositivos, como puede ser móvil, tablet, ordenador...
- Para ello, se han definido dos *medias queries*, para adaptar así el diseño de nuestra página web. Los *breakpoints* de cada *media query* son 768 píxeles para tabletas y 426 píxeles para teléfonos móviles.

```
@media only screen and (max-width: 768px)
```

```
@media screen and (max-width: 480px)
```

## HERRAMIENTAS Y LENGUAJES UTILIZADOS

Para la realización del proyecto, se han aprovechado herramientas y lenguajes que se han utilizado durante todo el curso, tanto en primero como en segundo. No se ha querido innovar mucho ya que el tiempo era limitado y el equipo consideró que la mejor manera para hacer un buen trabajo en ese tiempo limitado, era seguir el formato de todos los trabajos que se han hecho anteriormente en el curso. A continuación, se presentan las herramientas que han sido fundamentales para la realización de la página web:

### 1. HTML y CSS:



- Para la maquetación web, se usaron HTML y CSS, dos lenguajes muy potentes que permiten crear la estructura y el diseño de una página de manera eficiente y efectiva. Estos lenguajes han sido fundamentales en la creación de la página web, asegurando que todos los elementos se visualicen correctamente y tengan un diseño atractivo.

### 2. JavaScript:



- Además de HTML y CSS, se utilizó JavaScript para añadir interactividad a la página. JavaScript es un lenguaje de programación muy versátil que permite mejorar la experiencia del usuario, creando funciones dinámicas y respondiendo a las acciones del usuario en tiempo real. La implementación de la interactividad en el lado del cliente ha sido posible gracias a la cantidad de opciones y funciones que ofrece JavaScript.

- Este lenguaje ayudó a mejorar la experiencia de usuario mediante, por ejemplo, avisos a la hora de introducir incorrectamente ciertos datos en el registro.

### 3. PHP:



- PHP ha sido esencial en el lado del servidor para gestionar toda la lógica de la plataforma, interactuar con la base de datos, además de crearla y procesar las solicitudes del lado del cliente.
- Su uso, facilita separar al cliente con el uso de sesiones, para cuando un usuario/cliente inicia sesión, y le muestre el contenido, como puede ser el perfil, con toda su información.

### 4. Git y GitHub:



- Se decidió utilizar Git como sistema de control de versiones, ya que está prácticamente estandarizado, es la que el equipo decidió usar, también en parte porque el equipo está familiarizado con su uso.
- Ha ayudado de manera diferencial a la hora de poder trabajar cada integrante del equipo en su rama propia y en distintas funcionalidades, así se evita en cierta manera, o se reduce, los posibles conflictos de código que pueden aparecer.
- GitHub ha sido la plataforma donde se ha alojado el proyecto de forma remota.

## 5. Figma:



- Se ha utilizado Figma para el sistema de diseño y la creación de los mockups principales. Figma es una herramienta de diseño colaborativa que permite a los diseñadores trabajar juntos en tiempo real, lo cual fue esencial para coordinar el diseño visual del proyecto de manera eficiente.
- Además, Figma hizo posible que todos los miembros del equipo tuvieran acceso a los diseños. Esto mejoró la comunicación y facilitó la integración de ideas y sugerencias de todos los participantes, acelerando el proceso de diseño.

## 6. Discord:



- Para la comunicación en tiempo real y las reuniones semanales para verificar el progreso de todos los sprints, Discord fue de gran ayuda, ya que se creó un servidor específico para esta página web. Además, al contar con un chat, se podían plantear dudas y sugerencias de cambios de manera continua, que luego se trasladarían, esas dudas, a la siguiente herramienta que se va a comentar, que es Trello.

## 7. Trello:





- La gestión ágil de tareas y coordinación del equipo se llevó a cabo con aplicaciones como Trello, ya que facilitó mucho el trabajo al equipo con la organización de los sprints, tareas a realizar y asignación de esas tareas que faltaban por resolver.

#### 8. Visual Studio Code:



- Visual Studio Code fue la herramienta principal utilizada para la escritura y edición del código del proyecto. Este entorno de desarrollo integrado (IDE) eligió por su flexibilidad y las numerosas ventajas que ofrece para los desarrolladores.

#### 9. XAMPP:



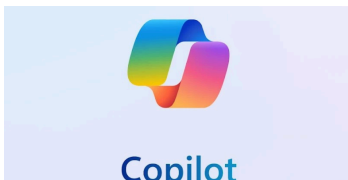
- XAMPP fue utilizado como servidor local para el desarrollo y pruebas del proyecto. Esta herramienta proporciona un entorno de desarrollo completo que incluye Apache, MySQL, PHP y Perl.
- Servidor Local: XAMPP permitió simular un servidor real en nuestros equipos, facilitando el desarrollo y las pruebas de la página web antes de su despliegue en un servidor de producción.
- Base de Datos MySQL: Se utilizó MySQL para gestionar la base de datos del proyecto, permitiendo almacenar y acceder a los datos de manera eficiente.

## 10. Chat GPT:



- Chat GPT fue utilizado para obtener asistencia en la redacción de contenido, generación de ideas y resolución de dudas técnicas durante el desarrollo del proyecto.
- Aprovechando la revolución que ha sido Chat GPT, todo el equipo ha aprovechado todo lo que ha podido esta herramienta, ya que parece que va a vivir entre nosotros mucho tiempo.

## 11. Copilot:



- Copilot también fue utilizado para obtener asistencia en la redacción de contenido, generación de ideas y resolución de dudas técnicas durante el desarrollo del proyecto.
- Aprovechando copilot para también poder generar imágenes mediante prompts, para poder dar un efecto visual mucho más bonito a la hora de utilizar los colores principales y los colores de contraste

## 12. Gimp:



- Utilizamos GIMP para la edición de fotos, principalmente para ajustar la resolución y asegurarnos de que las imágenes se adapten perfectamente a nuestras cartas sin perder la calidad que buscamos. GIMP nos permite redimensionar y optimizar las fotos manteniendo la proporción y nitidez, lo que es esencial para garantizar que las impresiones finales tengan una apariencia profesional y detallada. Gracias a sus herramientas avanzadas y su

capacidad para manejar alta resolución, podemos cumplir con nuestros estándares de calidad en cada proyecto.

### 13. Jquery:



- También hemos utilizado Jquery para que la interfaz sea mucho más buena a la hora de interactuar con ella
- Jquery es una librería de JS y la hemos utilizado para el slider de las Homes para visualizar todas las cartas y también para poder seleccionar el método de pago que queremos utilizar a la hora de hacer el pago

### 14. Redes sociales y Editores de video y texto :



- Hemos querido simular nuestro estreno al público mediante las redes sociales que aparecen en el footer
- Hemos utilizado aplicaciones de edición de videos y de imágenes para que todo parezca mucho más profesional a la hora de hacerlo público

## COMPONENTES DEL EQUIPO Y APORTACIÓN REALIZADA POR CADA ESTUDIANTE

### *Josep Serrano Rayó*

Josep ha sido el encargado principal de desarrollar las validaciones con JavaScript tanto para los formularios de pagos como de contacto en la plataforma. Además, ha diseñado el footer del sitio web y ha integrado APIs para mejorar la funcionalidad de pagos y atención al cliente. Su trabajo ha sido crucial para mejorar la calidad de los datos ingresados por los usuarios y para asegurar la integridad de las operaciones de inserción de datos en la base de datos, también ha sido el creador del figma, de los diseños de la página web y el envío de mensajes para la atención al cliente. A continuación, se detallan los aspectos específicos de su labor:

- **Validaciones en formulario**

- a. Josep implementó scripts para verificar que todos los campos obligatorios estén completados antes de permitir el envío del formulario. Esto incluye campos como el nombre del titular de la tarjeta, número de tarjeta de crédito, fecha de expiración y código de seguridad (CVV).
- b. Aseguró formatos específicos para cada campo. Por ejemplo, el número de tarjeta de crédito debe seguir el formato estándar de 16 dígitos, y la fecha de expiración debe estar en formato MM/AA.
- c. Estableció formatos específicos para cada campo en el apartado de contacto, asegurando que en el correo electrónico haya "@" y ".com".
- d. Validó que los campos no estuvieran vacíos para su envío a la base de datos y al contacto.
- e. Implementó validaciones adicionales para prevenir ataques y que la información dada no sea falsa a la hora de insertarlo en la base de datos

```

}

if (!/^[a-zA-Z0-9._-]+@[a-zA-Z0-9._-]+\.[a-zA-Z]{2,4}$/i.test(email)) {
  const error = document.createElement('p');
  error.classList.add('error');
  error.classList.add('mail');
  error.textContent = "El email es inválido o contiene caracteres no permitidos.";
  error.style.color = 'red';
  errorEmail.appendChild(error);
  return true;
}

```

```

if (cardNumber.length !== 16) {
  document.getElementById('card-number-error').innerText = 'El número de tarjeta debe tener 16 dígitos.';
  isValid = false;
} else {
  document.getElementById('card-number-error').innerText = '';
}

if (!/^\d{2}\d{2}$/i.test(cardExpiry)) {
  document.getElementById('card-expiry-error').innerText = 'La fecha de vencimiento debe estar en el formato MM/AA.';
  isValid = false;
} else {
  document.getElementById('card-expiry-error').innerText = '';
}

if (cardCvv.length !== 3) {
  document.getElementById('card-cvv-error').innerText = 'El CVV debe tener 3 dígitos.';
  isValid = false;
} else {
  document.getElementById('card-cvv-error').innerText = '';
}

if (isValid) {
  alert('Formulario enviado correctamente.');
```

// Aquí puedes agregar la lógica para enviar el formulario si todos los campos son válidos

// e.target.submit();

```

}

```

- **Diseño de Pagos**

- Validó completamente todos los formularios, para prevenir errores a la hora de insertar los datos en la base de datos y que el usuario ponga la información bien .
- Utilizó jQuery para hacer la selección de los métodos de pago, de manera que se oculten y muestren según corresponda.
- Se encargó de diseñar y asegurar la visualización de todos los objetos que están en el carrito.

### Métodos de pago

Elige un método de pago para continuar

+ Pago con Tarjeta de Crédito

Número de Tarjeta

1234 5678 9123 4567

Fecha de Vencimiento CVV

MM/AA 123

Pagar

- **Diseño del Footer :**

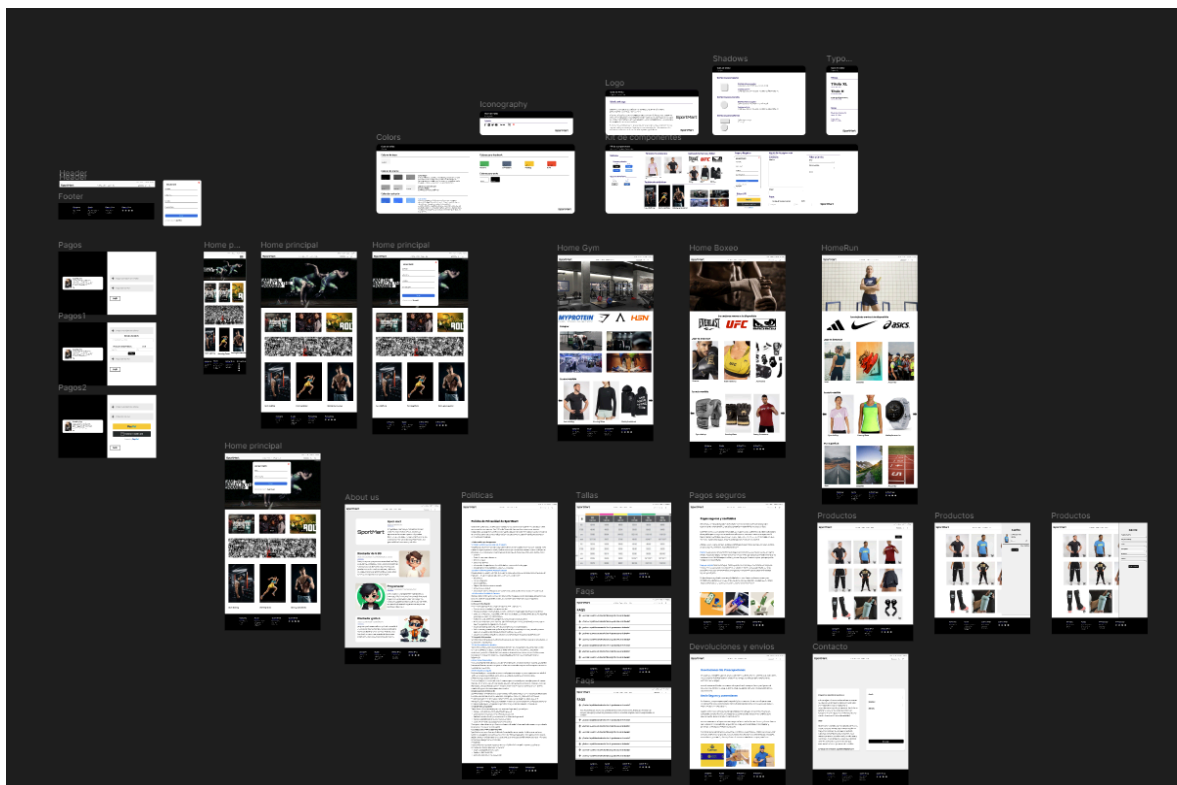
- a. Josep fue responsable del diseño del footer del sitio web, asegurando que incluyera enlaces a todas las páginas importantes de la plataforma. Esto incluye secciones como "Acerca de Nosotros", "Política de Privacidad", "Devoluciones", "Contacto", "Información" y "FAQ".
- b. Diseñó el footer para que fuera intuitivo y accesible, facilitando la navegación del usuario hacia diferentes secciones del sitio web desde cualquier página.
- c. Se encargó de todos los diseños para cada una de las páginas del footer, además de hacerlas responsivas para usuarios que accedan a la web desde portátil, móvil, etc.



```
<footer class= "footer" >
<div class="container">
<div class="row">
<div class="footer-col">
<h4>Company</h4>
<ul>
<li><a href="aboutus.php">About Us</a></li>
<li><a href="privacidad.php">Privacy Policy</a></li>
<li><a href="tallas.php">Tallas</a></li>
</ul>
</div>
<div class="footer-col">
<h4>Get Help</h4>
<ul>
<li><a href="faqs.php">FAQ</a></li>
<li><a href="PagosInfo.php">Métodos de Pago</a></li>
<li><a href="PagosYDevoluciones.php">Devoluciones</a></li>
<li><a href="contacto.php">Contacto</a></li>
</ul>
</div>
<div class="footer-col">
<h4>Online Shop</h4>
<ul>
<li><a href="home_gym.php">Menu Gym</a></li>
<li><a href="home_boxeo.php">Menu Fighting</a></li>
<li><a href="home_running.php">Menu Runinn</a></li>
</ul>
</div>
<div class="footer-col">
<h4>Follow Us</h4>
<div class="social-links">
<a href="#"><i class="fab fa-facebook-f"></i></a>
<a href="#"><i class="fab fa-twitter"></i></a>
<a href="#"><i class="fab fa-instagram"></i></a>
<a href="#"><i class="fab fa-youtube"></i></a>
</div>
</div>
</div>
</div>
</div>
```

● **Diseño del Figma :**

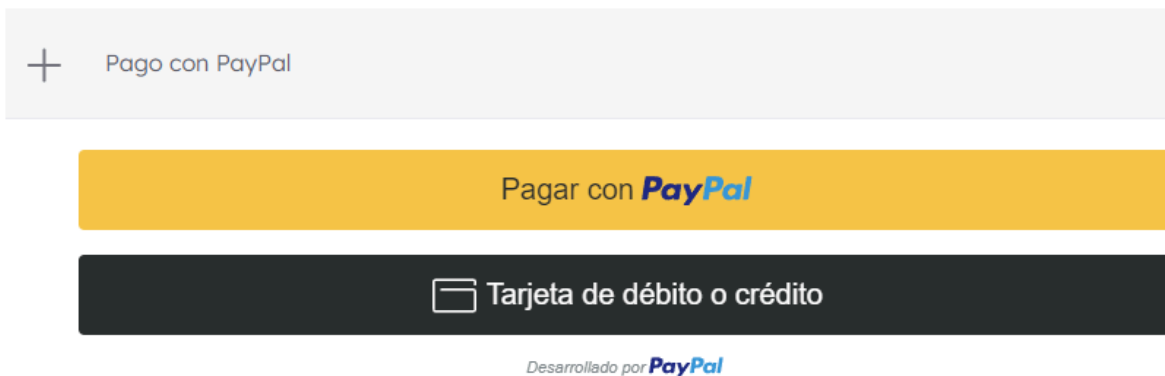
- a. Josep ha sido el principal encargado de hacer todos los diseños del figma, para tener una idea más clara de cómo se hace todo antes de pasarlo a código



- **Integración de APIs**

- a. Josep utilizó las APIs de PayPal para integrar una solución de pagos segura y eficiente en la plataforma, permitiendo a los usuarios realizar transacciones de manera sencilla y confiable.
- b. Implementó las APIs de WhatsApp para establecer un canal de comunicación más directo y profesional con los asistentes de Sportsmart, facilitando a los usuarios obtener soporte en tiempo real y resolver sus consultas de manera rápida.

```
<a href="https://api.whatsapp.com/send?phone=620365035" class="btn-wsp" target="_blank">  
  <i class="fa fa-whatsapp icono"></i>  
</a>
```



- **Simulación de las redes sociales**

- a. Josep se encargó de crear todas las redes sociales y editar los videos mediante aplicaciones de edición de videos y aplicaciones de diseño de imágenes como Gimp, Filmora y CapCut.

- **Resultados y Beneficios**

- a. Reducción de Errores de Usuario: Disminución significativa de los errores cometidos por los usuarios al completar los formularios.
- b. Mejora en la Calidad de Datos: Aumento en la calidad de los datos almacenados en la base de datos, facilitando la gestión y análisis de la información.
- c. Seguridad Mejorada: Menor riesgo de ataques y vulnerabilidades asociadas con la inserción y procesamiento de datos en la plataforma.



- d. Experiencia de Usuario Optimizada: Navegación más intuitiva y accesible gracias al diseño del footer, y mayor facilidad en la realización de pagos y en la obtención de soporte mediante las APIs integradas.
- e. Mejora Visual y eficiencia a la hora de hacer el código html css, js ya que está planificado anteriormente por el figma

312

El número de tarjeta debe tener 16 dígitos.

**Fecha de Vencimiento** **CVV**

MM/AA 123

La fecha de vencimiento debe estar en el formato MM/AA. El CVV debe tener 3 dígitos.

### ***Jorge Burgos***

Jorge ha sido el encargado de modificar la base de datos junto con Pablo y ha desarrollado las funcionalidades de login y registro, incluyendo formularios con validaciones y la comprobación de usuarios. Además, ha diseñado un menú responsive y ha implementado APIs para la funcionalidad multilingüe, haciendo la página accesible para usuarios de diferentes países. A continuación, se detallan los aspectos específicos de su labor:

- **Colaboración con la Base de Datos**

- a. Jorge trabajó junto con Pablo para modificar y optimizar la estructura de la base de datos. Su enfoque estuvo en mejorar la eficiencia y la integridad de los datos almacenados.
- b. Aseguró que la base de datos estuviera correctamente normalizada y optimizada para soportar las funcionalidades de login y registro, así como futuras expansiones.

<input type="checkbox"/>	carrito	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	4	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/>	categoria	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	cliente	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	11	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	deporte	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	detalles_pedido	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/>	pedido	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	perfil_cliente	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	11	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	producto	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	producto_categoria	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	producto_deporte	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	variante	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	198	InnoDB	utf8mb4_general_ci	32.0 KB	-
<b>11 tablas</b>		<b>Número de filas</b>						<b>362</b>		<b>InnoDB</b>	<b>utf8mb4_general_ci</b>	<b>336.0 KB</b>	<b>0 B</b>

## ● Login y Registro

- Jorge desarrolló formularios para el registro y login de usuarios, incorporando validaciones robustas para asegurar que los datos ingresados sean correctos y completos antes de ser enviados a la base de datos.
- Implementó la lógica necesaria para insertar nuevos usuarios en la base de datos durante el registro, asegurando la integridad de los datos y la prevención de duplicados.
- Desarrolló la funcionalidad para comprobar si los usuarios ya están registrados en el sistema. Esto incluye la validación de credenciales durante el proceso de login, permitiendo solo a los usuarios autorizados acceder a la página web.
- Diseño pop-ups intuitivos y visualmente atractivos para los procesos de login y registro, asegurando que se integren armoniosamente con el diseño general de la página web.

```

<!-- LOGIN -->
<div class="popup">
  <div class="close-btn"><i class="ri-close-circle-fill"></i></div>
  <p>Inicia sesión</p>
  <form id="form-login">
    <div class="form-element">
      <input required name="email-login" id="email_login" type="text">
      <label>Email</label>
    </div>
    <div class="form-element">
      <input required name="password-login" id="password_login" type="password">
      <label>Contraseña</label>
    </div>
    <span id="error-login"></span>
    <button class="btn-submit">Enviar</button>
  </form>
  <p>¿No tienes cuenta? <a href="" id="registrate" class="a2">Regístrate!</a></p>
</div>

```

```

// Comprobamos si el correo es correcto
$sql = mysqli_query($conn, "SELECT password, id_cliente, nombre FROM cliente WHERE correo = '$correo'");
if ($fila = mysqli_fetch_assoc($sql)) {
  // La función password_verify compara la contraseña ingresada con el hash almacenado
  if (password_verify($password, $fila['password'])) {
    // La contraseña es correcta
    $respuesta["success"] = true;

    // Almacenamos el id y el nombre en sesiones
    $_SESSION['id_cliente'] = $fila['id_cliente'];
    $_SESSION['nombre'] = $fila['nombre'];

    // Pasamos el nombre de usuario a la respuesta
    $respuesta["nombreUsuario"] = $fila['nombre'];
  } else {
    // La contraseña es incorrecta
    $respuesta["error"] = "contrasena";
  }
} else {
  // El correo no existe
  $respuesta["error"] = "correo";
}

```

- **Menú**

- a. Jorge se centró en el desarrollo de un menú responsive que se adapta a diferentes tamaños de pantalla. Esto incluye la implementación de un menú tipo hamburguesa para dispositivos móviles.
- b. Aseguró que el menú fuera accesible y fácil de usar en cualquier dispositivo, proporcionando una experiencia de usuario fluida y consistente.



*Menú*

```

<header>
  <button id="show-help">Ayuda</button>
  <button id="show-login" <?php echo isset($_SESSION['nombre']) ? 'disabled' : ''; ?> class="<?php echo isset($_S
  <button id="show-regis" class="bttn-regis"><?php echo $textoBotonRegistrarse; ?></button>
<div id="google_translate_element"></div>
</header>
<nav class="navbar">
  <div class="container-nav">
    <div class="logo-nav">
      
    </div>
    <div class="menu-toggle">
      <i class="ri-menu-line"></i>
    </div>
    <div class="drop-menu">
      <div class="menu-ppal">
        <a class="nav-button" href="index.php">Novedades</a>
        <a class="nav-button" href="home_running.php">Running</a>
        <a class="nav-button" href="home_gym.php">Gimnasio</a>
        <a class="nav-button" href="home_boxeo.php">Boxeo/MMA</a>
      </div>
    </div>
  </div>
</nav>

```

- **Api Multilingüe**

- Implementó APIs para cambiar el idioma de la página web, haciendo que la plataforma sea accesible para usuarios de diferentes países. Esto permite a los usuarios seleccionar su idioma preferido, mejorando la accesibilidad y la experiencia del usuario.

```

<script>
function loadGoogleTranslate() {
  var script = document.createElement('script');
  script.type = 'text/javascript';
  script.async = true;
  script.src = 'https://translate.google.com/translate_a/element.js?cb=googleTranslateElementInit';
  document.head.appendChild(script);
}

function googleTranslateElementInit() {
  new google.translate.TranslateElement({
    pageLanguage: 'es',
    includedLanguages: 'en,es,pt',
    layout: google.translate.TranslateElement.InlineLayout.SIMPLE
  }, 'google_translate_element');
}

document.addEventListener("DOMContentLoaded", function() {
  loadGoogleTranslate();
});
</script>

```

- **Resultados y Beneficios**

- a. Mejora en la Seguridad y Gestión de Usuarios: Implementación de un sistema robusto de login y registro, asegurando que solo usuarios autorizados puedan acceder a la plataforma.
- b. Optimización de la Base de Datos: Mejora en la eficiencia y estructura de la base de datos, facilitando una gestión más eficaz de la información.
- c. Experiencia de Usuario Mejorada: Navegación más intuitiva y accesible gracias al menú responsive y los pop-ups diseñados para login y registro.
- d. Accesibilidad Multilingüe: La página web es ahora accesible para un público global, permitiendo a los usuarios seleccionar su idioma preferido y mejorando la inclusividad de la plataforma.

***Pablo:***

Pablo ha sido el diseñador de la base de datos y ha realizado modificaciones continuas conforme se implementan nuevas funcionalidades en la página web. Además, se ha encargado del diseño y desarrollo de la home de running, el carrito de compras, generar de forma dinámica los productos, el slider y los filtros. A continuación, se detallan los aspectos específicos de su labor:

- **Gestión y Modificación de la BD**

- a. Pablo ha sido responsable de diseñar y mantener la base de datos, asegurando que esté siempre funcional y optimizada para soportar las nuevas implementaciones.
- b. Ha realizado modificaciones continuas en la estructura y los datos de la base de datos cada vez que se añadía una nueva funcionalidad a la página web, garantizando que todo funcionara de manera cohesiva y eficiente.

```

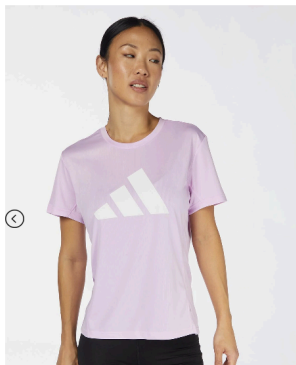
function crearBD()
{
    $bd = new BaseDeDatos();
    try {
        if ($bd->conectar()) {
            $conexion = $bd->getConexion();
            $createSql = [
                "CREATE DATABASE sportmart;",
                "USE sportmart;",
                "CREATE TABLE CLIENTE (
                    id_cliente INT AUTO_INCREMENT,
                    nombre VARCHAR(255) NOT NULL,
                    apellidos VARCHAR(255) NOT NULL,
                    correo VARCHAR(255) NOT NULL,
                    telefono VARCHAR(255) NOT NULL,
                    password VARCHAR(255) NOT NULL,
                    PRIMARY KEY (id_cliente)
                );",
                "CREATE TABLE PRODUCTO (
                    id_producto INT AUTO_INCREMENT,
                    nombre VARCHAR(255) NOT NULL,
                    descripcion TEXT,
                    precio DECIMAL(10, 2) NOT NULL,
                    imagen VARCHAR(255),
                    talla VARCHAR(255),
                    sexo ENUM('Hombre', 'Mujer') NOT NULL,
                    PRIMARY KEY (id_producto)
                );",
                "CREATE TABLE FAVORITOS (
                    id_favorito INT AUTO_INCREMENT,
                    id_cliente INT,
                    id_producto INT,
                    PRIMARY KEY (id_favorito),
                    FOREIGN KEY (id_cliente) REFERENCES CLIENTE(id_cliente),
                    FOREIGN KEY (id_producto) REFERENCES PRODUCTO(id_producto)
                );",
            ];
        }
    }
}

```

- **Desarrollo de la home de Running**

- a. Pablo se encargó de diseñar y desarrollar la página principal de la sección de running. Esto incluye la creación de un slider atractivo y dinámico utilizando CSS, JavaScript y jQuery.
- b. Gracias a su trabajo, la home de running y las demás páginas principales cuentan con un slider que mejora la presentación de los productos y ofertas, proporcionando una experiencia de usuario visualmente agradable y dinámica.

## Lo más vendido



**Camiseta Adidas**  
Camiseta ligera y transpirable para correr  
29.99€



**Garmin Forerunner 955**  
Reloj inteligente premium para carrera  
499.99€

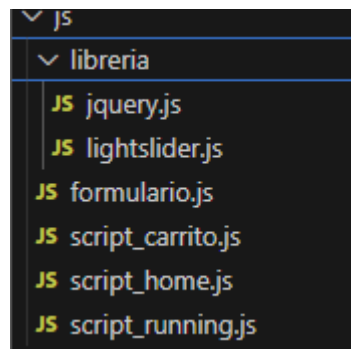


**Asics Gel-Cumulus 26**  
Comodidad y rendimiento en cada paso  
128.99€



**Gafas Oakley Radar**  
Gafas polarizadas especiales para running  
205.99€

## Tarjetas de productos



## Librerías utilizadas

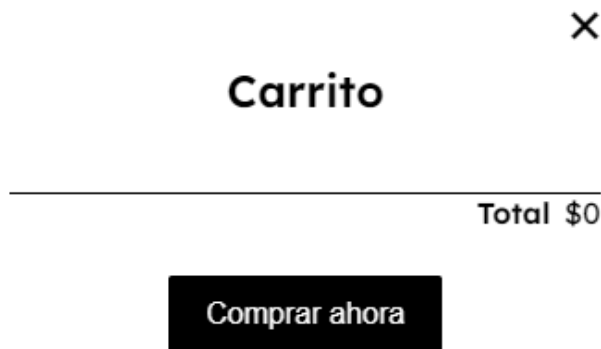
### ● Diseño y Desarrollo del Carrito de Compras

- Pablo fue el encargado del diseño del carrito de compras, asegurando que los productos añadidos se visualicen correctamente en pantalla. Utilizó SQL para mostrar toda la información relevante del carrito y aplicó estilos CSS para una mejor presentación.
- Implementó la lógica necesaria para que el carrito de compras funcione de manera fluida, permitiendo a los usuarios ver los productos añadidos, actualizar cantidades y ver el total de la compra en tiempo real.
- Gracias a ello, también es posible visualizar el carrito a la hora de pagar, proporcionando una vista clara de los productos seleccionados por el cliente.

```

function ready() {
  // Remove Items From Cart
  var removeCartButtons = document.getElementsByClassName('cart-remove');
  for (var i = 0; i < removeCartButtons.length; i++){
    var button = removeCartButtons[i];
    button.addEventListener('click', removeCartItem);
  }
  // Quantity Changes
  var quantityInputs = document.getElementsByClassName('cart-quantity')
  for (var i = 0; i < quantityInputs.length; i++) {
    var input = quantityInputs[i];
    input.addEventListener('change', quantityChanged);
  }
  // Add to Cart
  var addCart = document.getElementsByClassName('add-cart');
  for (var i = 0; i < addCart.length; i++) {
    var button = addCart[i];
    button.addEventListener('click', addToCartClicked);
  }
  // Buy Button Work
  document.getElementsByClassName('btn-buy')[0].addEventListener('click', buyButtonClicked);
}

```



*Imagen del carrito*

- **Productos**

- a. Pablo fue el encargado de la gestión de los productos, permitiendo que todos ellos puedan ser añadidos al carrito de compras mediante consultas SQL. Aseguró que la interacción con la base de datos fuera eficiente y segura.
- b. Desarrolló una interfaz de usuario intuitiva para la visualización y gestión de productos, facilitando a los usuarios la navegación y la selección de artículos para su compra.



Inicio

Productos

Filtros



Ultraboost light  
Zapatillas para correr con suela amortiguada  
89,99 €



Nike Hyperko 2  
Zapatillas de boxeo hombre  
129,99 €



Adidas Box Hog 3  
Zapatillas de boxeo mujer  
109,99 €

*Productos con filtros*

- **Detalles del Filtro de Productos:**

- a. **Criterios de Búsqueda:** Los usuarios pueden filtrar productos por diversas categorías como tipo de deporte, precio, marca, tamaño, y color. Esto permite una búsqueda más específica y rápida, facilitando la localización de productos que se ajusten a las preferencias individuales de cada usuario.
- b. **Interfaz Intuitiva:** Pablo diseñó una interfaz de usuario clara y fácil de usar para el filtro de productos. Los usuarios pueden seleccionar múltiples criterios y ver los resultados en tiempo real, lo que mejora la eficiencia de la búsqueda.
- c. **Optimización de Búsquedas:** Utilizó técnicas avanzadas de SQL y JavaScript para asegurar que las búsquedas sean rápidas y precisas, minimizando el tiempo de carga y ofreciendo resultados relevantes al instante.
- d. **Integración con la Base de Datos:** El filtro está estrechamente integrado con la base de datos para garantizar que las búsquedas reflejen la disponibilidad actual de los productos, proporcionando información actualizada y precisa.



## Filtrar y ordenar

Precio mínimo:

Precio máximo:

Sexo:

Categoría:

Deporte:

Tallas:

Aplicar filtros

- **Resultados y Beneficios**

- a. Base de Datos Optimizada y Funcional: Mantener una base de datos siempre funcional y adaptada a las nuevas necesidades de la página web, asegurando la integridad y eficiencia de los datos.
- b. Mejora en la Presentación de la Home de Running: Una página principal de running con un slider dinámico que mejora la presentación visual de los productos y promociones.
- c. Experiencia de Compra Mejorada: Un carrito de compras funcional y visualmente atractivo que facilita la experiencia de compra para los usuarios, con una gestión eficiente de productos y visualización en tiempo real.

- d. Gestión de Productos Eficiente: Una interfaz intuitiva y una gestión de productos eficaz, permitiendo a los usuarios añadir artículos al carrito de manera fácil y rápida.

## FASES DEL PROYECTO

### Estudio de mercado

Habiendo realizado el estudio de mercado previo al desarrollo de la página web, el equipo se ha dado cuenta de que faltan muchas tiendas especializadas en pocos deportes. La mayoría de las tiendas online se centran en un amplio rango de disciplinas deportivas, pero pocas ofrecen una selección especializada y de alta calidad en áreas específicas como running, gimnasio y boxeo/MMA.

En contraste con esta tendencia general, la tienda SportMart se distingue por su enfoque especializado en estos tres deportes. El equipo reconoce la importancia de ofrecer productos de calidad y especializados que cubran las necesidades específicas de los entusiastas del running, los fanáticos del gimnasio y los practicantes de boxeo y MMA.

SportMart se dedica a proporcionar una experiencia de compra única y enfocada, donde los clientes pueden encontrar todo lo que necesitan para sus actividades deportivas favoritas en un solo lugar. Con una amplia gama de productos cuidadosamente seleccionados, desde ropa y calzado hasta equipos y accesorios, SportMart se asegura de que cada artículo cumpla con los estándares más altos de calidad y rendimiento.

Para mejorar la experiencia de compra, SportMart ofrece varias facilidades de pago, incluyendo opciones seguras y convenientes como PayPal y tarjetas de crédito y débito. Esto garantiza que los clientes puedan completar sus compras de manera rápida y segura, disfrutando de una experiencia de compra sin complicaciones.

El objetivo de SportMart es ofrecer una tienda de ecommerce que no solo venda productos de alta calidad, sino que también sea un referente para aquellos que buscan mejorar su rendimiento y bienestar en el running, el gimnasio y el boxeo/MMA. FitMaster quiere ser el lugar de referencia para todos aquellos que se toman en serio su pasión por estos deportes.

## Análisis de la web

El análisis de la web revela una plataforma integral diseñada para abordar las necesidades de los entusiastas del running, gimnasio y boxeo/MMA. SportMart se estructura de manera intuitiva y eficiente para proporcionar una experiencia de compra fluida y satisfactoria.

- Página Principal

Al ingresar a la web, los usuarios son recibidos por una página de inicio que muestra las categorías principales: running, gimnasio y boxeo/MMA. La página principal también destaca productos destacados y nuevas llegadas, facilitando a los usuarios encontrar rápidamente lo que necesitan.

- Páginas para cada deporte

SportMart se especializa en tres disciplinas deportivas clave: running, gimnasio y boxeo/MMA. Cada deporte tiene su propia sección dedicada en la web, ofreciendo una experiencia personalizada para los usuarios. En cada una de estas secciones, los usuarios pueden encontrar una amplia gama de productos especializados, desde ropa y calzado técnico hasta equipos y accesorios esenciales.

- Detalles de los Productos

Cada producto en SportMart cuenta con una página detallada que incluye descripciones exhaustivas y especificaciones técnicas. Las imágenes de alta calidad proporcionan una visión clara de cada artículo. Esto asegura que los clientes tengan toda la información necesaria para tomar decisiones de compra informadas.

- Proceso de Compra

El proceso de compra en SportMart es fácil y seguro. Los usuarios pueden añadir productos a su carrito y proceder al pago de manera rápida. La plataforma soporta múltiples métodos de pago, incluyendo PayPal y tarjetas de crédito y débito, ofreciendo a los clientes flexibilidad y seguridad en sus transacciones.

- Cuentas de usuario

Principalmente está enfocado a rellenar todos los datos de contacto de ese posible cliente, así como su dirección de entrega para que se quede guardada a la hora de realizar una compra.

- Atención al Cliente

SportMart ofrece un robusto sistema de atención al cliente, incluyendo una sección de preguntas frecuentes y un formulario de contacto. Estos recursos están diseñados para resolver rápidamente cualquier duda o problema que los clientes puedan tener, asegurando una experiencia de compra satisfactoria.

### **Modelo de datos utilizado**

El modelo de datos utilizado en “SportMart” incluye varias tablas relacionadas entre sí para así asegurarse la integridad de los datos y gestionar la información de los clientes, el carrito de la compra, productos, categoría de los productos, detalles del pedido, método de pago...

A continuación se muestra la estructura de las tablas y sus relaciones:

#### **CLIENTE:**

- id\_cliente (clave primaria)
- nombre
- apellidos
- correo
- password

#### **PERFIL\_CLIENTE:**

- id\_cliente (clave primaria, referencia a CLIENTE.id\_cliente)
- fecha\_nac\_cliente
- miembro\_desde
- telefono
- provincia
- localidad
- direccion\_envio
- codigo\_postal

## **PRODUCTO:**

- id\_producto (clave primaria)
- nombre
- descripcion
- precio
- imagen
- sexo

## **VARIANTE:**

- id\_variante (clave primaria)
- id\_producto (referencia a PRODUCTO.id\_producto)
- talla

## **CARRITO:**

- id\_carrito (clave primaria)
- id\_cliente (referencia a CLIENTE.id\_cliente)
- id\_producto (referencia a PRODUCTO.id\_producto)
- cantidad
- talla

## **CATEGORÍA:**

- id\_categoria (clave primaria)
- categoría

## **PRODUCTO\_CATEGORIA:**

- id\_producto (referencia a PRODUCTO.id\_producto)
- id\_categoria (referencia a CATEGORIA.id\_categoria)
- (clave primaria compuesta: id\_producto, id\_categoria)

## **DEPORTE:**

- id\_deporte (clave primaria)
- deporte

### **PRODUCTO\_DEPORTE:**

- id\_producto (referencia a PRODUCTO.id\_producto)
- id\_deporte (referencia a DEPORTE.id\_deporte)
- (clave primaria compuesta: id\_producto, id\_deporte)

### **PEDIDO**

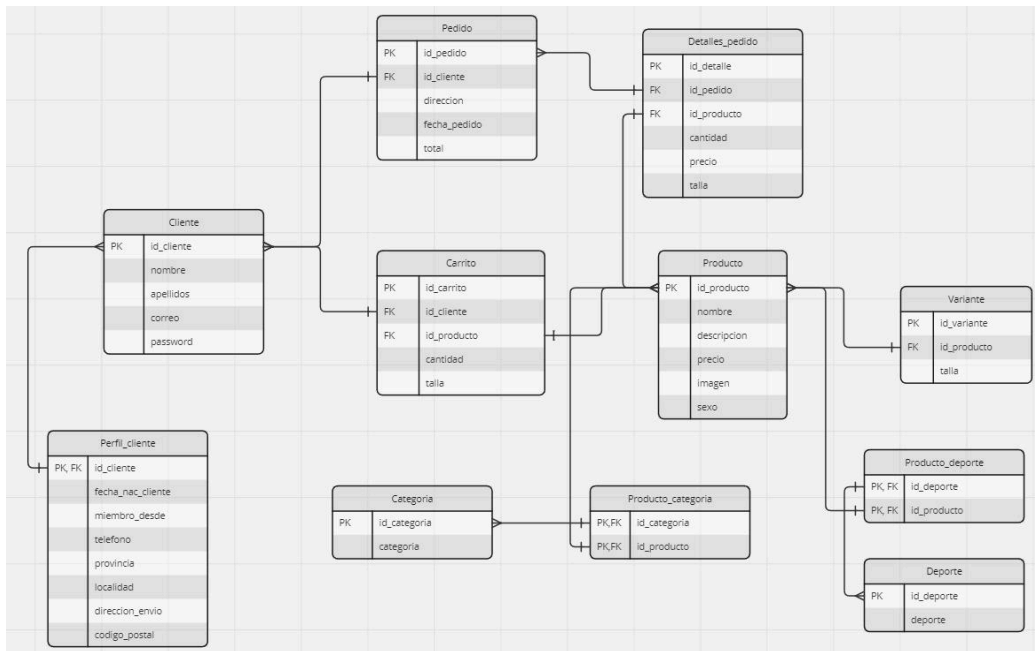
- id\_pedido (clave primaria)
- id\_cliente (referencia a CLIENTE.id\_cliente)
- direccion
- fecha\_pedido
- total

### **DETALLES\_PEDIDO:**

- id\_detalle (clave primaria)
- id\_pedido (referencia a PEDIDO.id\_pedido)
- id\_producto (referencia a PRODUCTO.id\_producto)
- cantidad
- precio
- talla



## Modelo relacional



Modelo relacional para la creación de la BBDD

## Diagrama Entidad/Relación

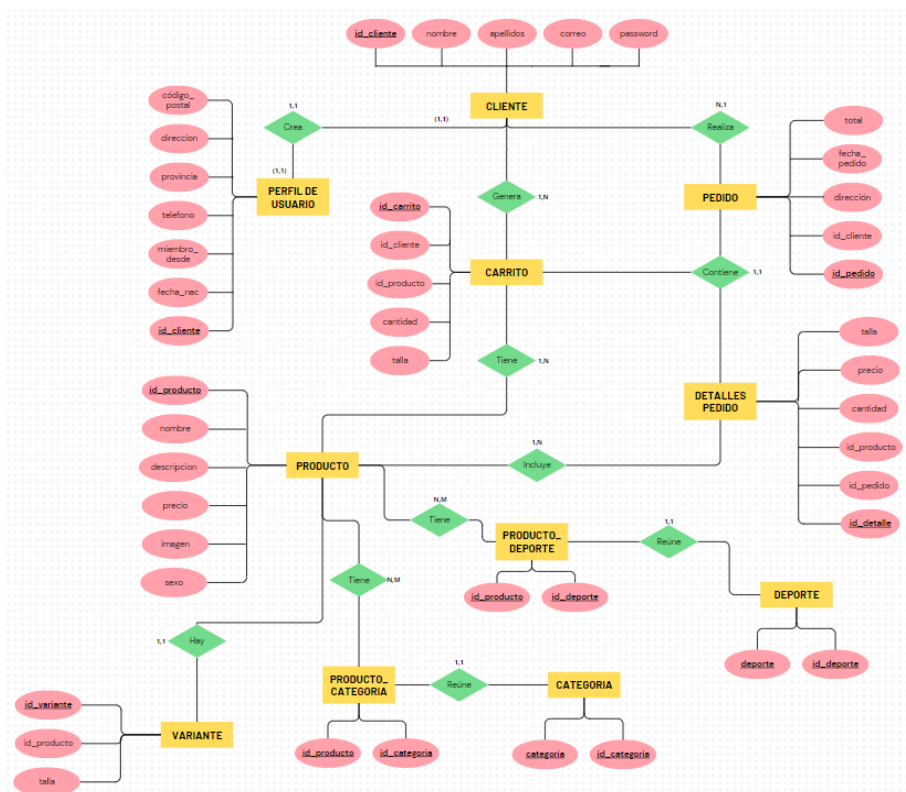


Diagrama Entidad/Relacion

## Diseño de interfaces

El equipo enfrentó un desafío al diseñar el sistema, buscando crear algo tanto funcional como estéticamente atractivo, coherente con la identidad de la marca. Se invirtió mucho tiempo y esfuerzo en este proceso, experimentando con varios sistemas de diseño antes de encontrar uno satisfactorio.

En cuanto a los colores, se optó por una paleta en tonos de grises, negros y blanco, con la intención de dirigir la atención del cliente hacia los productos en lugar de la propia aplicación. Se buscaba que el colorido de los productos fuera el foco principal de la experiencia de compra.

El logo fue otro punto importante. Se buscaba un diseño simple pero elegante que capturara la esencia de la marca. Después de varias exploraciones, se llegó a un diseño que encapsula perfectamente la visión del equipo.

El logo consiste simplemente en el nombre de la aplicación con la fuente Orbitron, optando por un enfoque minimalista para dar protagonismo a los deportes y productos en la plataforma.

# SportMart

*Logo de la marca*

**Guía de estilo**  
Colores

**Colores de base**

#F5F5F5

**Colores de marca**

#000000	#545454	#909090	<b>Color negro</b> El color negro es una elección estratégica y poderosa para nuestra tienda de deportes, y no es casualidad que lo hayamos seleccionado como nuestro color predominante. El negro es más que un color; es una declaración.
#9F9F9F	#C4C4C4	#F3F3F3	

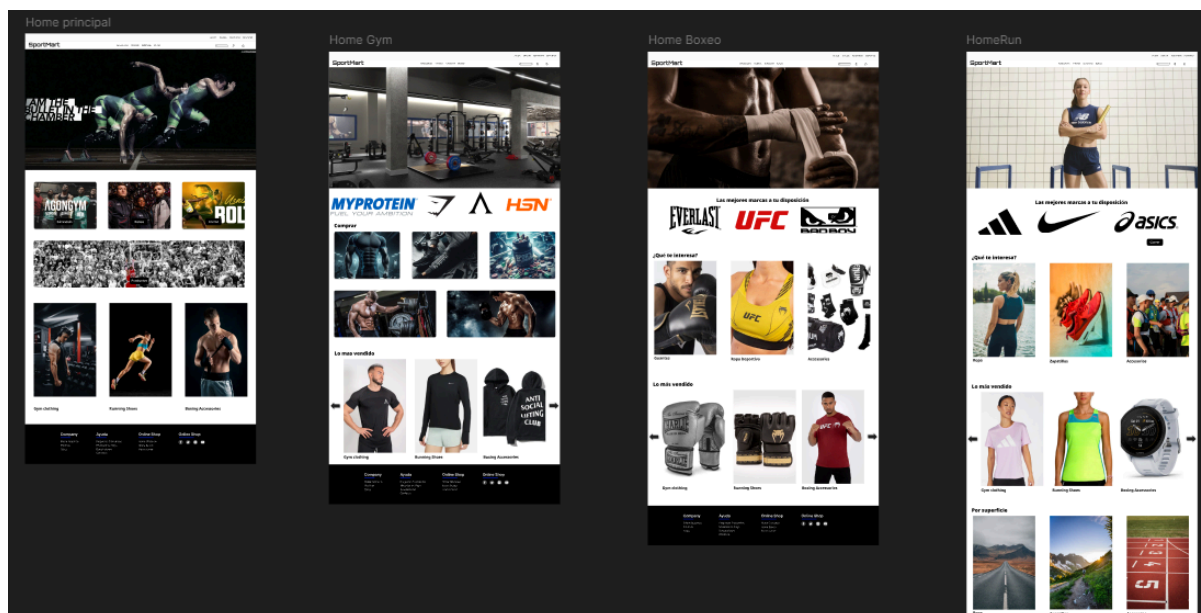
**Elegancia y Sofisticación**  
**Fuerza y Poder**  
**Confianza y Profesionalismo**

**Color de contraste**

#0171e3	#007EFF	#70B2FF	<b>Azul suave</b> Para complementar la fuerza y sofisticación del negro, hemos elegido el azul como color de contraste. El azul es un color que simboliza confianza, serenidad y profesionalismo, cualidades que también son esenciales en el ámbito deportivo. Este color añade un toque de frescura y vitalidad a nuestra tienda, creando un ambiente equilibrado y acogedor.
---------	---------	---------	--

*Paleta de colores*

Debido a la falta de tiempo, el equipo se centró principalmente en hacer mockups finales e interactivos, de esta manera se agiliza el proceso.



## Planificación del desarrollo

Para la ejecución del proyecto, se estableció una organización mediante la plataforma Trello, donde se registraron las tareas correspondientes a cada sprint, indicando su fecha límite de entrega y asignación respectiva. Se crearon múltiples secciones dentro de la plataforma, destinadas a recordar actividades de menor relevancia pero que aún así requerían atención.

## CONCLUSIONES Y MEJORAS DEL PROYECTO

En el marco del desarrollo del proyecto de ecommerce, se ha logrado crear una plataforma que refleja el compromiso del equipo con la excelencia técnica y la innovación en el campo del comercio electrónico. Desde la fase inicial de planificación hasta la implementación final, se enfrentaron diversos desafíos técnicos y conceptuales, los cuales pudieron superarse gracias a la integración de tecnologías como HTML, CSS, JavaScript, PHP y MySQL.

La plataforma ofrece una experiencia interactiva y dinámica para los usuarios, con características como la navegación intuitiva, la amplia variedad de productos y la seguridad de la información del cliente. Además, la inclusión de funciones como la gestión de carrito de compras.

Sin embargo, se reconoce que siempre hay espacio para mejoras. Si hubiera dado tiempo durante el desarrollo del proyecto, se habría valorado la opción de incluir características adicionales como un historial de pedidos para que los usuarios puedan rastrear fácilmente sus compras anteriores. Además, una sección de favoritos permitiría a los usuarios guardar productos para futuras compras, mejorando la experiencia de compra personalizada.

Además, se consideraría la implementación de un chat en tiempo real de atención al usuario para proporcionar soporte instantáneo y resolver cualquier duda o problema que los clientes puedan tener durante su visita a la plataforma. Esto ayudaría a mejorar la satisfacción del cliente y fomentar la interacción con la marca.

En resumen, aunque se está orgulloso del resultado alcanzado con la plataforma de ecommerce, siempre se está buscando formas de mejorar y ofrecer una experiencia aún mejor a los usuarios.

## BIBLIOGRAFÍA

### [Carrusel de marcas]

▶ SLIDER INFINITO AUTOMATICO únicamente con HTML & CSS desde cero en ME...

### [Api de Google]

▶ Google Translate API in JavaScript - Free Alternative

### [API de PayPal]

▶ 🤖 \*\*Acepta pagos con tarjeta\*\* [ Cómo crear BOTÓN Inteligente de PAYPAL ] Paso...

### [Guía para consultas sobre css y html]

<https://developer.mozilla.org/en-US/>

### [Guía para consultas de php]

<https://www.php.net/manual/es/index.php>

### [La mejor guía de todas]

<https://chat.openai.com/>

## ANEXOS

### Creación de la base de datos

Si la base de datos no ha sido creada previamente, al dirigirte al index.php se crea y se insertan todos los campos de prueba.

```
<?php
require_once 'conecta.php';

function existeBBDD()
{
    $bd = new BaseDeDatos();
    $flag = false;
    try {
        if ($bd->conectar()) {
            $conexion = $bd->getConexion();
            $sql = "SHOW DATABASES LIKE 'sportmart'";
            $result = mysqli_query($conexion, $sql) or die(mysqli_error($conexion));
            if (mysqli_num_rows($result) > 0) {
                $flag = true;
            }
        }
    } catch (Exception $e) {
        // echo $e->getMessage();
    }
    $bd->cerrar();
    return $flag;
}
```

```
function crearBD()
{
    $bd = new BaseDeDatos();
    try {
        if ($bd->conectar()) {
            $conexion = $bd->getConexion();
            $createSql = [...];
            $insertSql = [...];

            // Ejecutar las consultas de inserción de datos
            ejecutarSentencias($conexion, $createSql);
            ejecutarSentencias($conexion, $insertSql);
            $bd->cerrar();
        }
    } catch (Exception $e) {
        // echo $e->getMessage();
    }
}
```

*Función para crear la BBDD.*

Todo esto es posible, ya que en cada página que se vaya a dirigir el usuario, se introduce un `require_once` del archivo PHP, así como el control de las sesiones para cambiar dinámicamente el valor de los botones de inicio y registro.

```

index.php > html > head > script
You, 4 days ago | 3 authors (You and others)
1 <?php
2 require_once 'php/bbdd.php';
3 session_start();
4 // Verificar si el usuario ha iniciado sesión
5 if (isset($_SESSION['nombre'])) {
6     $textoBotonLogin = "Hola, " . $_SESSION['nombre'];
7     $textoBotonRegistrarse = "Cerrar sesión";
8 } else {
9     $textoBotonLogin = "Inicia sesión";
10    $textoBotonRegistrarse = "Regístrate";
11 }
12 ?>

```

A continuación se muestra un pequeño ejemplo de cómo quedaría estructurada la base de datos en el gestor PHPMyAdmin.

<input type="checkbox"/>	carrito	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	4	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/>	categoria	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	cliente	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	10	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	deporte	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	3	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	detalles_pedido	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/>	pedido	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	perfil_cliente	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	10	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	producto	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/>	producto_categoria	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	producto_deporte	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	44	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/>	variante	★	Examinar	Estructura	Buscar	Insertar	Vaciar	Eliminar	198	InnoDB	utf8mb4_general_ci	32.0 KB	-
<b>11 tablas</b>		<b>Número de filas</b>		<b>360</b>		<b>InnoDB</b>		<b>utf8mb4_general_ci</b>		<b>336.0 KB</b>		<b>0 B</b>	