

ROLL&PLAY

Desarrollo de Aplicaciones Web

IES Francisco Ayala

[Por: Juan García
Rosales]

Resumen del Proyecto

Castellano

Este proyecto consiste en el desarrollo de una plataforma web de alquiler de películas, al estilo de un videoclub tradicional pero adaptado al entorno digital.

La idea es que los usuarios puedan navegar por un catálogo de películas, alquilar las que deseen y reproducirlas durante un período de tiempo determinado.

El sistema permitirá a los usuarios registrarse, explorar el catálogo de películas y alquilar títulos de forma sencilla. Los administradores podrán gestionar el catálogo y los usuarios desde un panel de administración privado.

Se espera que el proyecto incluya un diseño adaptable a todas las resoluciones para que los usuarios puedan acceder a la plataforma desde dispositivos móviles, tablets u ordenadores sin problemas de compatibilidad.

El enfoque del proyecto es combinar la nostalgia de los antiguos videoclubs con la comodidad y accesibilidad de las plataformas de streaming modernas, ofreciendo una experiencia sencilla, intuitiva y atractiva para el usuario.

English

This project involves the development of a web-based movie rental platform, similar to a traditional video store but adapted to the digital environment.

The idea is for users to browse a catalog of movies, rent the ones they want, and play them for a set period of time.

The system will allow for the creation of user accounts, a shopping cart for each user, and, if possible, payment management.

The project is expected to include a design adaptable to all resolutions so that users can access the platform from mobile devices, tablets or computers without compatibility issues.

This project's approach is to combine the nostalgia of old-fashioned video stores with the convenience and accessibility of modern streaming platforms, offering a simple, intuitive, and engaging experience for the user.

Contenido

1.	Características generales	3
2.	Restricciones generales.....	3
3.	Aspectos a cubrir	4
4.	Estudio de las herramientas a utilizar y Justificación de las tecnologías empleada .	4
5.	Requerimientos hardware y software	6
6.	Análisis y diseño	7
6.1	Diagrama de casos de uso.....	7
6.2	Diagrama de clases.....	8
6.3	Descripción de la base de datos relacional.....	8
6.4	Modelo Entidad/Relación	9
7	Implementación.....	10
7.1	Archivos Raíz	11
7.2	app	11
7.3	Config/	12
7.4	Public/	13
7.5	Resources/	13
7.6	routes/	15
7.7	storage/.....	15
7.8	vendor/	15
8	Evaluación y prueba.....	15
8.1	Testeo de la aplicación web	15
8.2	Verificación de campos y realimentación	16
8.3	Testeo de base de datos	16
8.4	Batería de pruebas (ejemplos funcionales)	16
9	Manual de estilos	18
9.1	Sketches	18
9.2	Criterios de accesibilidad	26
9.3	Criterios de usabilidad	27
9.4	Tipografía	27
9.5	Mapa de colores de proyecto.....	28
9.6	Dispositivos/vistas para las que se ha diseñado el proyecto	28

10	. Mejoras posibles y aportaciones	29
11	. Licencias y Copyright.....	30

1. Características generales

- Registro y Autenticación de Usuarios
 - o Registro mediante nombre, apellidos, correo electrónico y contraseña
- Perfiles de Usuario
 - o Perfil básico (usuario): para alquilar películas y administrar su lista de pedidos.
 - o Perfil de administrador: para gestionar el catálogo de películas, los usuarios y controlar las suscripciones.
- Catálogo de Películas
 - o Página de aterrizaje con una serie de opciones que se pueden realizar
 - o Página de inicio con las películas destacadas.
 - o Barra de búsqueda para buscar una película en concreto
 - o Página de detalles de cada película (sinopsis, reparto, tráiler).
 - o Opciones para añadir al carrito de la compra.
 - o Página de perfil de usuario.
 - o Páginas de gestión de productos, pedidos, usuarios y categorías para administradores
- Alquiler de Películas
 - o Reproducción de películas durante un tiempo limitado. (Solo se podrá elegir el tiempo en meses)
 - o Pago mediante un portal seguro (PayPal).
 - o Historial de pedidos.
- Buscador
 - o Se podrán buscar películas por nombre
- Barra Lateral
 - o Gestión del catálogo (añadir, editar y eliminar películas) (CRUD).
 - o Gestión de usuarios y control de productos, categorías y pedidos (CRUD).

2. Restricciones generales

- Restricción de Pago: Solo es posible pagar con PayPal
- Barra lateral: Los usuarios normales no podrán acceder a todas las funciones de la barra lateral al contrario que los administradores que si podrán acceder a todas las funciones.

- Visualización de películas (tiempos de carga): Las películas solo se podrán subir mediante enlaces o mediante subidas desde algún dispositivo, pero en este segundo caso suelen tardar bastante en subirse a la página.

3. Aspectos a cubrir

- **Estructura de la página:**
 - Página de aterrizaje: Tendrá cuatro opciones destacadas para los usuarios que acaban de entrar en la página
 - Página de inicio: Mostrará una lista aleatoria de películas destacadas
 - Página de registro/inicio de sesión: Formulario de registro y e inicio de sesión.
 - Página de categorías: En el header se mostrará una lista de categorías que te llevarán a una página dedicada a cada categoría.
 - Página de detalles de película: Información detallada, botón de ver película/alquiler y tráiler y el reparto de la película.
 - Perfil de usuario: Datos personales e historial de alquileres.
 - Barra Lateral (administrador): Gestión de usuarios, pedidos, productos y categorías.
 - Página del carrito de la compra: Donde se podrán ir añadiendo productos.

4. Estudio de las herramientas a utilizar y Justificación de las tecnologías empleada

En el desarrollo de la plataforma web del videoclub se han utilizado varias herramientas modernas que ofrecen un equilibrio entre productividad, rendimiento y escalabilidad. A continuación, se justifica el por qué se han utilizado algunas herramientas frente a otras alternativas de su misma categoría:

1. **Laravel (Framework Backend)**

Laravel ha sido elegido frente a alternativas como Symfony o Slim por su estructura clara, su sistema de rutas, autenticación integrada y el ORM Eloquent, lo que acelera el desarrollo. A diferencia de Slim (microframework) o Symfony (más complejo), Laravel ofrece una solución completa y balanceada ideal para aplicaciones medianas como un videoclub.

2. **Blade y React (Frontend)**

Blade permite la generación rápida de vistas del lado del servidor, mientras que React mejora la interactividad en secciones dinámicas. React se ha preferido frente a Vue o Angular por su flexibilidad y comunidad. Esta combinación supera las limitaciones de Blade solo, ofreciendo una experiencia de usuario más fluida.

3. **Tailwind CSS (Estilado)**

Tailwind se ha utilizado en lugar de Bootstrap por su enfoque el cual permite un diseño más personalizado y liviano. Frente a otros frameworks Tailwind ofrece mayor control visual y se integra mejor con componentes React.

4. **CSS3**

Se ha utilizado también CSS para ciertos elementos de la página que son más complicados de realizar con Tailwind.

5. **MySQL (Base de datos)**

Se ha optado por MySQL por su fiabilidad, facilidad de uso y compatibilidad con Laravel. Aunque PostgreSQL es más robusto para estructuras complejas, MySQL es suficiente para la lógica relacional de un videoclub, manteniendo buen rendimiento.

6. **Carbon (Manejo de fechas en PHP)**

Carbon mejora el manejo de fechas frente a las funciones nativas de PHP, simplificando tareas comunes como calcular la duración de un alquiler o fechas de vencimiento.

7. **Composer**

Sistema de gestión de paquetes de código de PHP para manejar dependencias y librerías en el proyecto.

8. **Javascript**

También se ha utilizado un poco de javascript para reaccionar a ciertos eventos durante el funcionamiento de la página web.

Este conjunto de tecnologías ha permitido desarrollar una plataforma web moderna, eficiente y mantenible, superando las limitaciones de soluciones más antiguas o menos integradas. La elección de cada herramienta se basó en un análisis comparativo, valorando factores como la comunidad, la documentación, el rendimiento y la facilidad de aprendizaje.

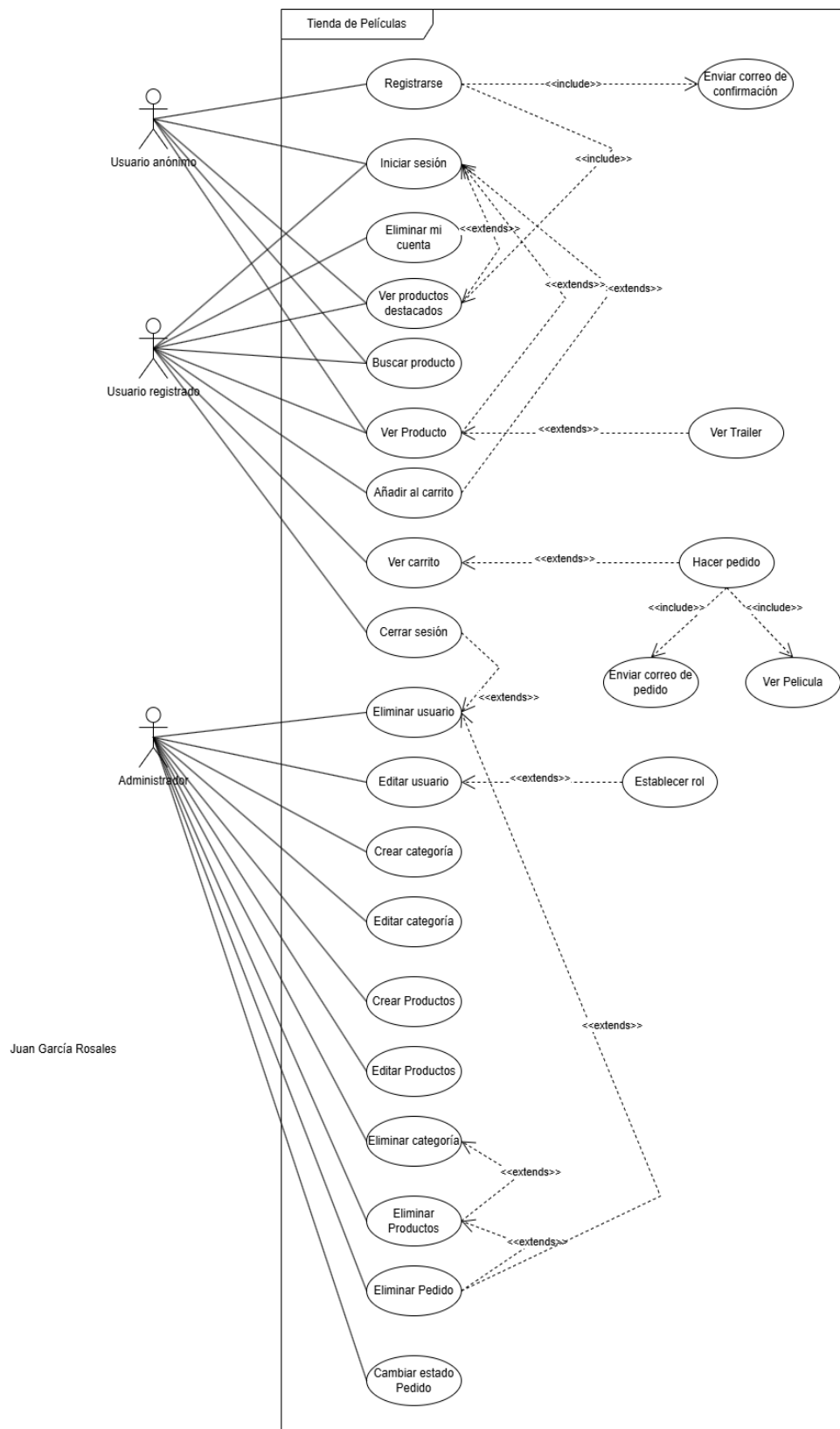
5. Requerimientos hardware y software

Para que la página web funcione correctamente será necesario acceso a internet y un navegador actualizado como Chrome, Firefox o Edge.

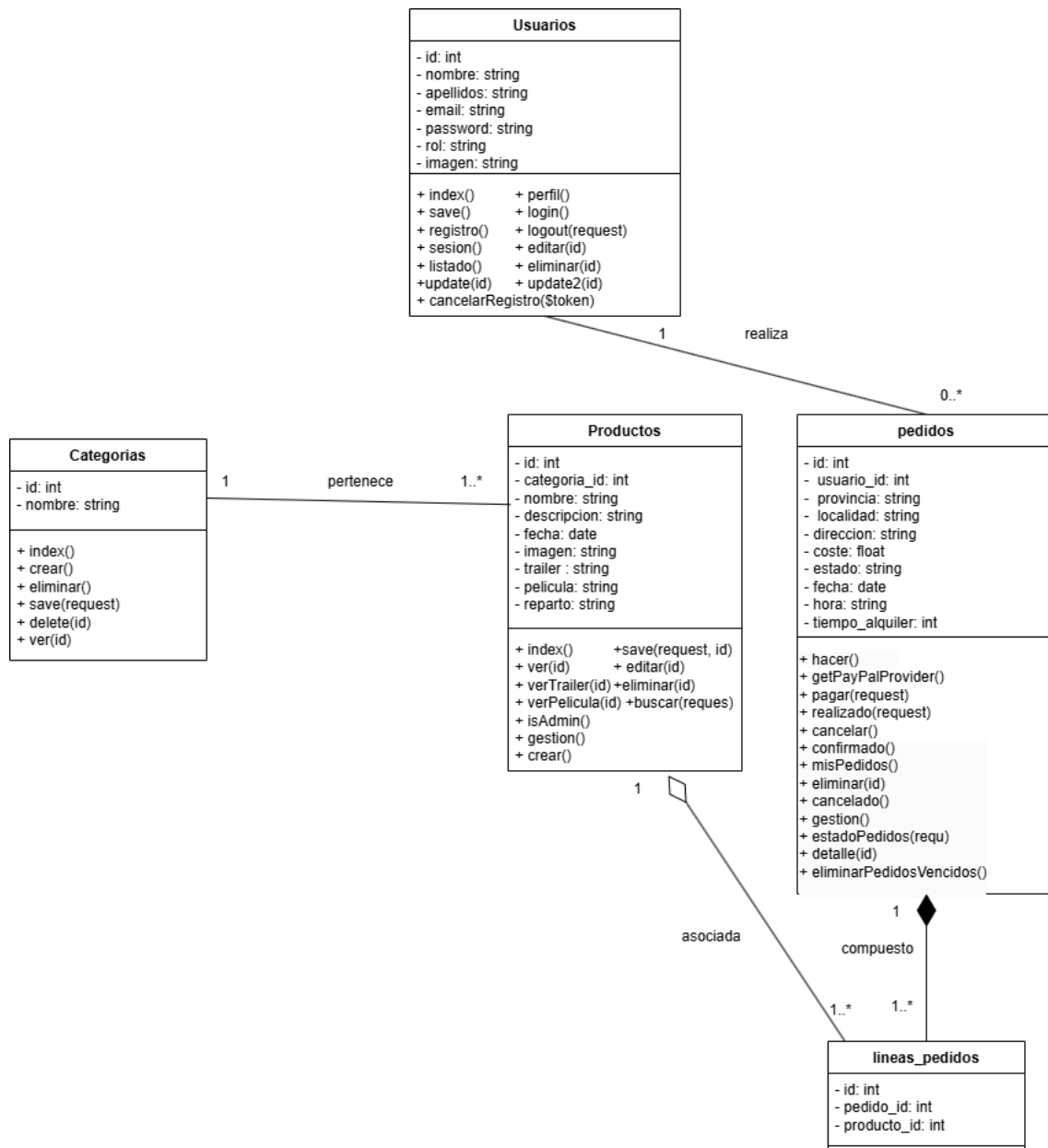
Por otra parte, si se elige ejecutar la web en remoto será necesario instalar xampp, un paquete de php actualizado a una versión por encima de la 8 y composer.

6. Análisis y diseño

6.1 Diagrama de casos de uso



6.2 Diagrama de clases



6.3 Descripción de la base de datos relacional

La base de datos constara de las siguientes tablas:

Usuarios: Esta tabla almacena la información de los usuarios que interactúan con la plataforma. Los datos de esta tabla permiten gestionar la autenticación, el acceso a la plataforma y los roles asignados a cada usuario (por ejemplo, administrador o cliente).

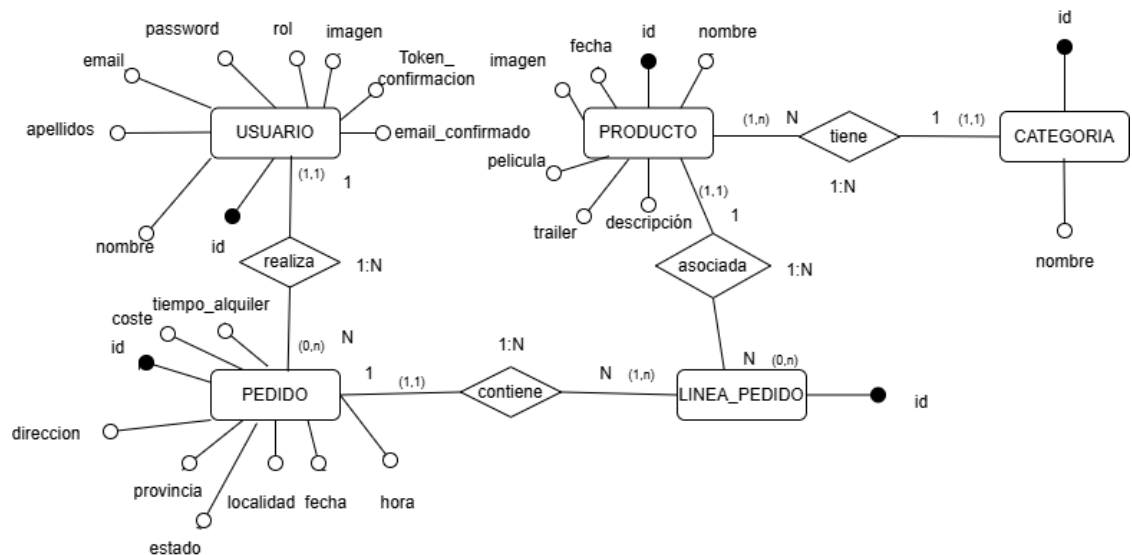
Categorías: La tabla de categorías agrupa las películas o productos en diferentes géneros o tipos, como "Acción", "Comedia", "Drama", etc. Esto permite organizar el catálogo de productos y facilitar la búsqueda o filtrado por categorías en la plataforma.

Productos: En esta tabla se almacenan los productos disponibles en el videoclub, principalmente las películas. Cada producto tiene detalles sobre su contenido, y la categoría a la que pertenece. Los productos están relacionados con las categorías para facilitar su organización.

Pedidos: Esta tabla registra los pedidos realizados por los usuarios. Contiene información sobre el estado del pedido, el precio de este, su tiempo de alquiler y el usuario que lo realizó, así como la fecha en la que se hizo.

Lineas_pedidos: Las líneas de pedidos desglosan cada producto incluido en un pedido. Incluyen que pedido es exactamente y que productos se pidieron en ese pedido.

6.4 Modelo Entidad/Relación



○ Diseño Lógico:

USUARIO (**id**, nombre, password, apellidos, email, rol, imagen, token_confirmacion, email_confirmado)

CATEGORIA (**id**, nombre)

PRODUCTO (**id**, nombre, fecha, imagen, descripción, trailer, película, categoría_id)

6.4.1.1 *Categoría_id: Clave foránea de CATEGORIA*

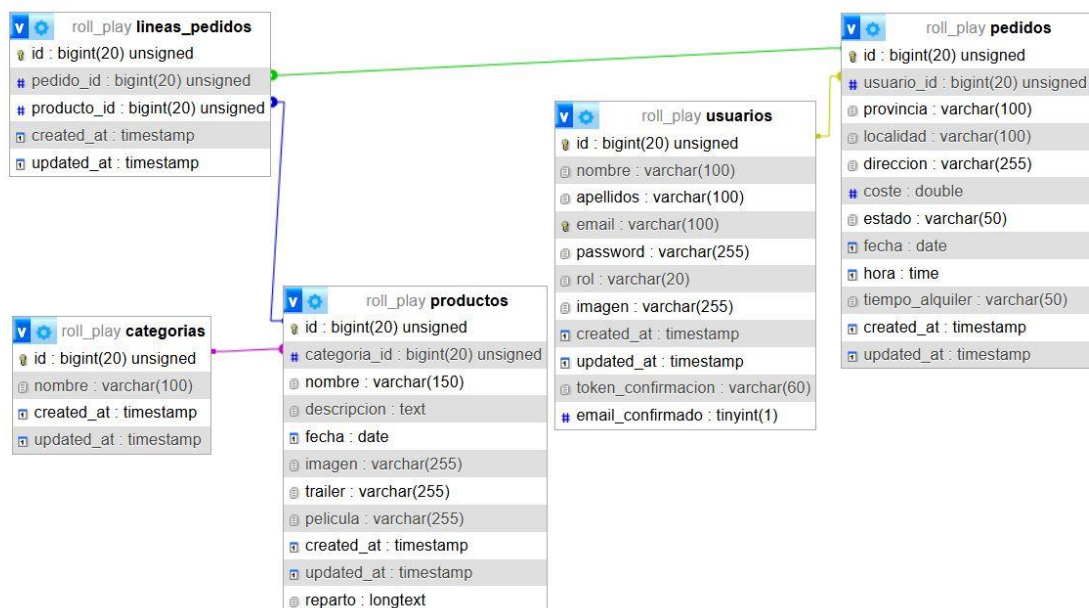
PEDIDO (**id**, provincia, dirección, fecha, hora, estado, localidad, coste, tiempo_alquiler, id_usuario)

6.4.1.2 *Id_usuario: clave foránea de USUARIO*

LINEA_PEDIDO (**id**, unidades, producto_id, pedido_id)

6.4.1.3 *producto_id: Clave foránea de PRODUCTO*

6.4.1.4 *pedido_id: Clave foránea de PEDIDO*



7 Implementación

La implementación del proyecto se ha llevado a cabo combinando tecnologías del lado del cliente y del servidor para lograr una aplicación funcional, estructurada y mantenible. A continuación, se describen los principales elementos utilizados:

7.1 Archivos Raíz

- **.env:** Contiene configuraciones sensibles como nombre de la base de datos, usuario, contraseña, URL base, claves de servicios externos. Este archivo no se debe subir al repositorio por seguridad.
- **composer.json:** Archivo con una lista de dependencias de composer la cual es necesaria para que funcione el proyecto
- **composer.lock:** Archivo con información acerca de las dependencias y subdependencias relacionadas con Composer
- **package.json:** Archivo con una lista de dependencias necesarias de Javascript, React y Tailwindcss para que funcione el proyecto
- **package.lock:** Archivo con información acerca de las dependencias y subdependencias relacionadas con Javascript, React y tailwindcss.
- **postcss.config.js / tailwind.config.js:** Configuran TailwindCSS y PostCSS para el procesamiento de estilos.
- **vite.config.js:** Configura Vite como empaquetador de frontend (JS/CSS).
- **README.md:** Documento introductorio que describe el proyecto, instalación y uso.

7.2 app

- **Http/Controllers:** Contiene los controladores que gestionan la lógica del backend.
 - **CarritoController.php:** Gestiona las funciones del carrito de compras: añadir, quitar productos, mostrar contenido.
 - **CategoriaController.php:** CRUD de categorías de productos (crear, editar, eliminar, mostrar).
 - **Controller.php:** Clase base que extienden todos los controladores. Contiene lógica común.
 - **ErrorController.php:** Muestra vistas personalizadas para errores
 - **PedidoController.php:** Se encarga de crear pedidos, confirmar, mostrar detalles y ver historial de pedidos del usuario.
 - **ProductoController.php:** Controla la gestión de películas/productos: crear, editar, ver detalles, mostrar destacados, trailers, etc.
 - **UsuarioController.php:** Controla todo lo relacionado con los usuarios: login, registro, perfil, modificación de datos, listado.
- **helpers/**

- **utils.php**: Archivo donde se definen funciones globales reutilizables, como obtener las categorías o mostrar el carrito.

- **Mail/**

- **ConfirmacionRegistro.php**: Archivo que contiene una clase con los datos que se mandaran en un email después de haber realizado el registro de usuario.

- **PedidoConfirmado.php**: Archivo con la clase que contiene los datos del pedido para mandarlo en un correo al usuario

- **Models/:**

- **categoria.php**: Modelo para acceder/modificar datos de la tabla categorías.

- **lineaPedido.php**: Cada fila de un pedido (producto, cantidad, precio).

- **pedido.php**: Representa un pedido hecho por un usuario.

- **producto.php**: Modelo para las películas/productos del catálogo.

- **usuario.php**: Modelo para los datos de usuarios registrados.

- **Rules/:**

- **DnsCheck.php**: Verifica si los dominios de los correos con los que se registra la gente son validos

7.3 Config/

Contiene la configuración de cada aspecto del framework Laravel.

- **app.php**: Nombre del proyecto, zona horaria, idioma, proveedores, etc.

- **auth.php**: Configura el sistema de autenticación.

- **cache.php**: Define cómo se almacenan temporalmente los datos.

- **database.php**: Configura el acceso a la base de datos (MySQL).

- **filesystems.php**: Define dónde se guardan archivos subidos (local o nube).

- **logging.php**: Gestión de logs del sistema (registro de errores, actividad).

- **mail.php**: Configura envío de correos electrónicos.

- **paypal.php**: Configuración para pagos con PayPal (si lo usas).

- **queue.php**: Configura el sistema de tareas en segundo plano (no siempre usado).

- **services.php**: Claves API y configuración de servicios externos como Google, PayPal, etc.

- **session.php**: Guarda las sesiones de los navegadores que entran a la base de datos

7.4 Public/

- **index.php**: Punto de entrada de la aplicación. Todo pasa por aquí antes de llegar a las rutas.

- **css/styles.css**: Archivo de estilos generado (compilado) que se aplica a toda la web.

7.5 Resources/

- **css/app.css**: Archivo fuente con los estilos de Tailwind o personalizados.

- **js/**

- **components/Example.jsx**: Componente de React que mostrara una barra lateral en toda la página web.

- **components/HoverDevCards.jsx**: Tarjetas interactivas con efecto hover.

- **bootstrap.js**: Inicializa librerías y configuraciones JS del proyecto.

- **app.jsx**: Punto de entrada de la aplicación React, donde se renderizan los componentes. En otras palabras, para poder llamar a componentes react, estos tienen que estar primer en el app.jsx

- **views/**

Vistas Blade (plantillas HTML con PHP).

- **carrito/**

- **index.blade.php**: Muestra el carrito de la compra

- **categoria/**

- Carpeta con vistas para el CRUD de categorías. Incluye páginas para listar, crear, editar y ver categorías.

- **emails/**

- **confirmacion.blade.php**: Contenido del email de confirmación que se manda después de que un usuario se registre.

- **pedido_confirmado.blade.php**: Contenido del email de confirmación de pedido que se manda después de que un usuario realice un pedido.

- **layouts/**

- **app.blade.php**: Plantilla base común que contiene el HTML general de la web (cabecera, scripts, footer), usada por otras vistas mediante @extends.

- **pedido/**

- **confirmado.blade.php**: Muestra un mensaje de éxito una vez que el pedido haya sido realizado con éxito

- **detalle.blade.php**: Mostrará los detalles de un pedido en concreto

- **hacer.blade.php**: formulario para finalizar el pedido.

- **pedidos.blade.php**: Lista de pedidos realizados por los usuarios, dependiendo de si es un administrador o un usuario normal mostrara todos los pedidos o solo algunos

- **producto/**

- **crear.blade.php**: Formulario necesario para crear o editar un producto

- **destacados.blade.php**: Actuará como una página de inicio en la que siempre se mostraran 6 productos aleatorios.

- **gestión.blade.php**: Lista de productos existentes en la página, solo podrán verlos los administradores ya que te muestran opciones para eliminar o editar el pedido.

- **película.blade.php**: Página para ver las películas

- **resultados.blade.php**: Página que mostrara los productos que coincidan con lo que se estaba buscando en el buscador de la web.

- **trailer.blade.php**: Página para ver los tráileres

- **ver.blade.php**: Mostrará los detalles de una película en concreto

- **usuario/**

- **listado.blade.php**: Mostrará una lista de todos los usuarios registrados en la página web.

- **modificar.blade.php:** Permitirá modificar los datos de un usuario ya existente mediante un formulario.

- **perfil.blade.php:** Mostrará una vista del perfil de tu usuario y te permitirá editarlo o eliminarlo. Aquí también se podrá ver el historial de pedidos

- **registro.blade.php:** Formulario de registro a la web

- **session.blade.php:** Formulario de inicio de sesión

- **home.blade.php:** Pagina de aterrizaje, te muestra una serie de acciones a realizar

7.6 routes/

- **console.php:** Define comandos personalizados de Artisan. Se usa para tareas automáticas como limpieza de datos o actualizaciones.

- **web.php:** Archivo principal de rutas del sitio. Aquí defines las URL que manejará la aplicación y qué controlador y método debe atender cada una (ej: `Route::get('/peliculas', [ProductoController::class, 'index'])`).

7.7 storage/

- **app/public:** Carpeta donde se guardan archivos subidos desde la web, como imágenes de productos o trailers. Debe estar enlazada a public/storage mediante el comando `php artisan storage:link` para que sean accesibles desde el navegador.

7.8 vendor/

Directorio **generado automáticamente por Composer** que contiene todas las librerías y dependencias necesarias del proyecto (Laravel, Carbon, DomPDF, etc.).

No debe modificarse manualmente.

8 Evaluación y prueba

8.1 Testeo de la aplicación web

Se ha hecho un testeo exhaustivo de la aplicación web para comprobar que esta funciona correctamente, algunos ejemplos de esto pueden ser:

Navegación y rutas: Se ha comprobado que al hacer click en cada uno de los enlaces estos te lleven al lugar correcto.

Interactividad React: Se ha comprobado que el menú lateral se cierre y abra correctamente y que sus rutas cargan de manera dinámica

Integración Blade + React: Se ha confirmado que los datos que se cargan de React a Blade mediante el parámetro `window.appData` se reciben correctamente.

Funcionalidad de formularios: También se ha comprobado la funcionalidad de cada uno de los formularios de la página para que estos realicen correctamente su función, como por ejemplo el formulario de registro, inicio de sesión, añadir productos, añadir categorías, etc...

8.2 Verificación de campos y realimentación

En cada uno de los campos de los formularios se han añadido validaciones para que no se puedan introducir valores erróneos y en caso de introducirlos que estos salgan en rojo e indiquen que los datos no son válidos.

8.3 Testeo de base de datos

También se ha asegurado la retención de datos de los formularios en la base de datos siempre y cuando estos datos estén bien definidos como lo indican los formularios

Además, estos datos una vez insertados en la base de datos pueden ser llamados en diferentes instancias de la página, como por ejemplo cuando se inserta una nueva película esta podrá aparecer en su respectiva categoría.

8.4 Batería de pruebas (ejemplos funcionales)

Registro de nuevos usuarios:

Se ha probado a crear un nuevo usuario con el formulario de registro y efectivamente funciona correctamente creando al usuario y asignándole por defecto como rol user. En caso de querer ser administrador necesitaras que otro admin te cambie el rol de tu cuenta.

Inicio de sesión con distintos roles (admin, cliente):

En caso de iniciar sesión como un usuario normal entonces solo te aparecerán opciones limitadas en la barra lateral como pueden ser ‘Var Carrito’, ‘mi Perfil’ o ‘mis Pedidos’.

En caso de iniciar sesión como un administrador entonces te aparecerán más opciones a poder realizar como pueden ser administrar usuario, categorías, productos o pedidos.

Creación y edición de productos:

Como administrador se podrán crear productos nuevos los cuales necesitarán un nombre, descripción, fecha de estreno, la imagen de portada, un tráiler y la película además del reparto que la conforma.

Se ha confirmado que al crear el producto este exista en la base de datos y se pueda alquilar para poder visualizar su película

Realización de pedidos desde el carrito: Cuando tú decides alquilar una película puede pulsar un botón que tiene esta que te llevará directamente al carrito en donde tendrás esta misma película que hayas elegido lista para poder alquilarla.

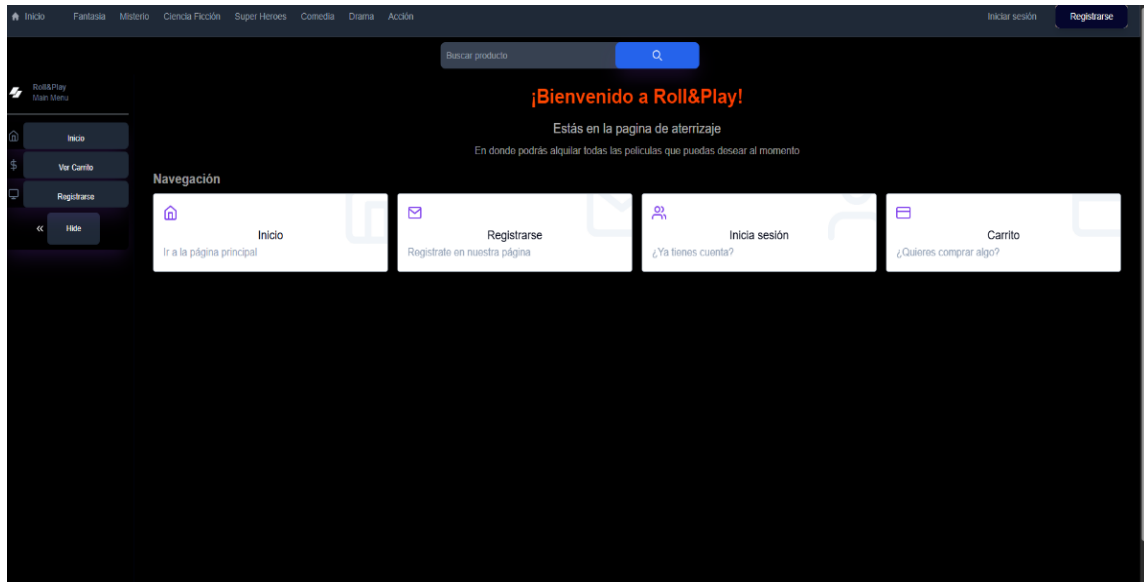
Visualización de pedidos realizados: Si eres un administrador, tienes permitido visualizar todos los pedidos realizados por cada uno de los usuarios de la página web y en caso de ser necesario puedes cancelar el pedido impidiendo así al usuario ver la película.

Filtrado de productos por categoría: En el menú de la página se muestran ciertas categorías de películas si buscas una categoría en concreto basta con pulsarla y te llevará a una página en donde solo se muestre películas de esa misma categoría.

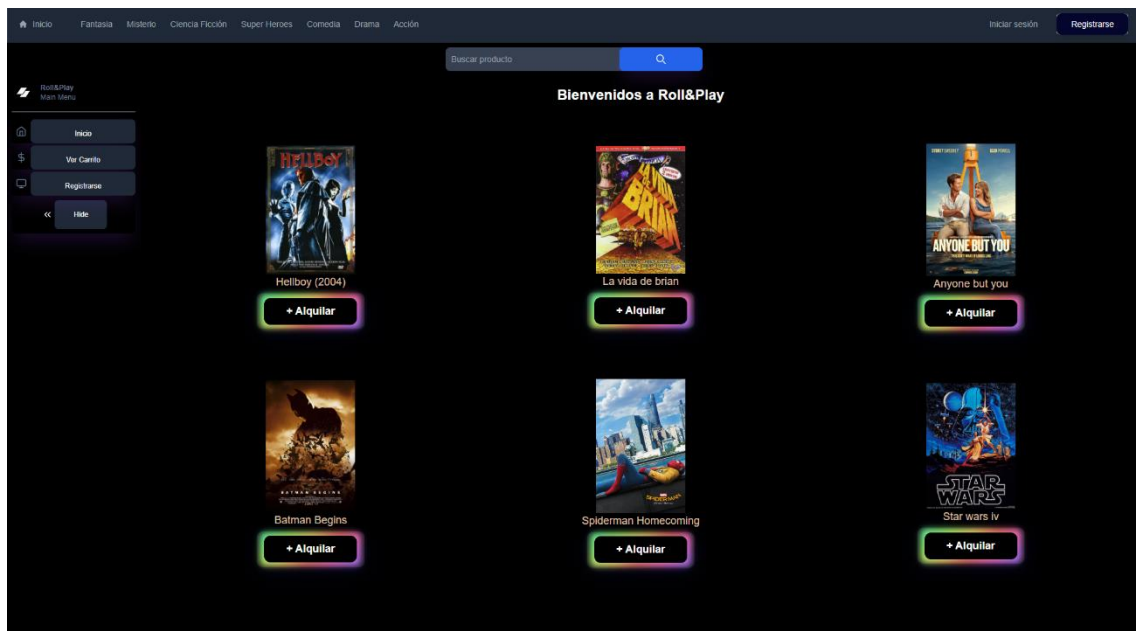
9 Manual de estilos

9.1 Sketches

Página de aterrizaje



Página de productos destacados



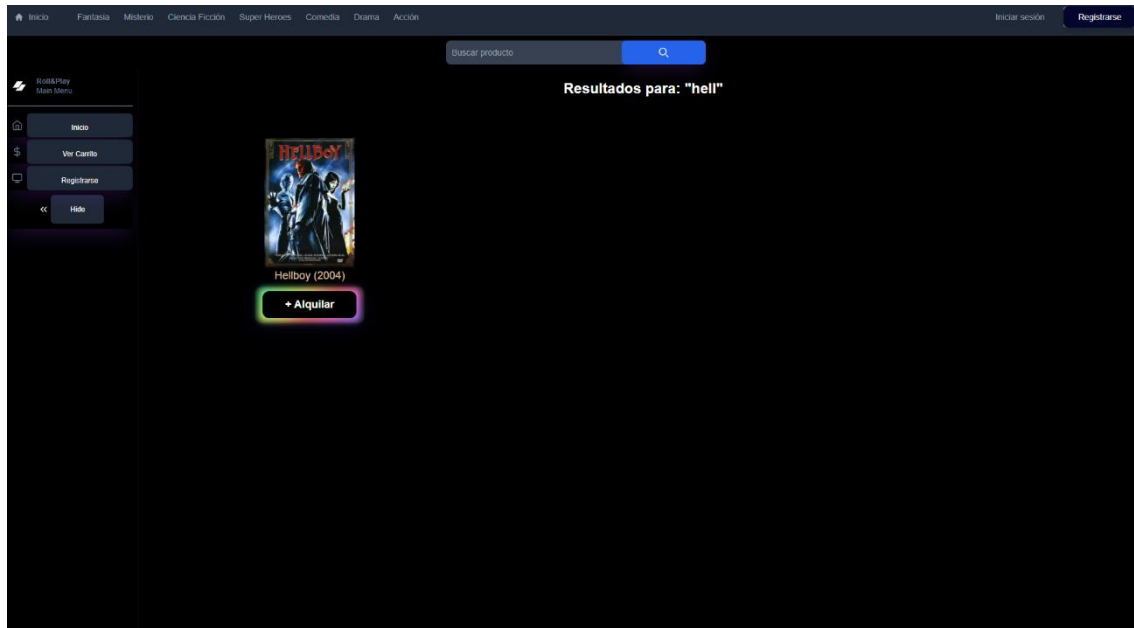
Registro

The screenshot shows a web application interface with a dark theme. At the top, there is a navigation bar with links: Inicio, Fantasia, Misterio, Ciencia Ficción, Super Heroes, Comedia, Drama, and Acción. On the right side of the navigation bar are links for 'Iniciar sesión' and 'Registrarse'. Below the navigation bar is a search bar labeled 'Buscar producto' with a magnifying glass icon. On the left side, there is a sidebar menu titled 'Roll&Play Main Menu' with buttons for 'Inicio', 'Ver Carrito', 'Registrarse', and 'Hide'. The main content area is titled 'Registrarse' and contains the following form fields: 'Nombre:' (Name), 'Apellidos:' (Last names), 'Email', and 'Password'. Each field has a corresponding input box. Below the 'Password' field is a checkbox labeled 'Mostrar contraseña' (Show password) and a 'Registrarse' button. At the bottom of the form, there is a link: '¿Ya tienes una cuenta? [Iniciar sesión](#)'.

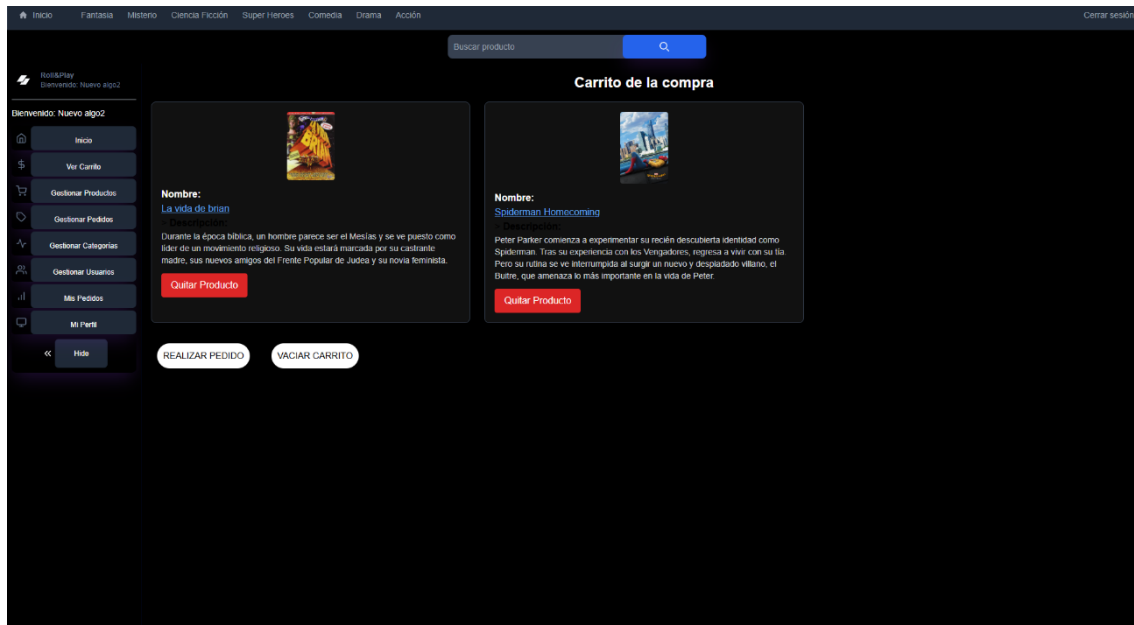
Inicio de sesión

The screenshot shows the same web application interface as the previous one, but the main content area is titled 'Entrar a la Web' (Enter the Web). The form fields are: 'Email' and 'Password', each with an input box. Below the 'Password' field is a checkbox labeled 'Mostrar contraseña' (Show password) and an 'Iniciar sesión' button. At the bottom of the form, there is a link: '¿No tienes cuenta? [Regístrate](#)'.

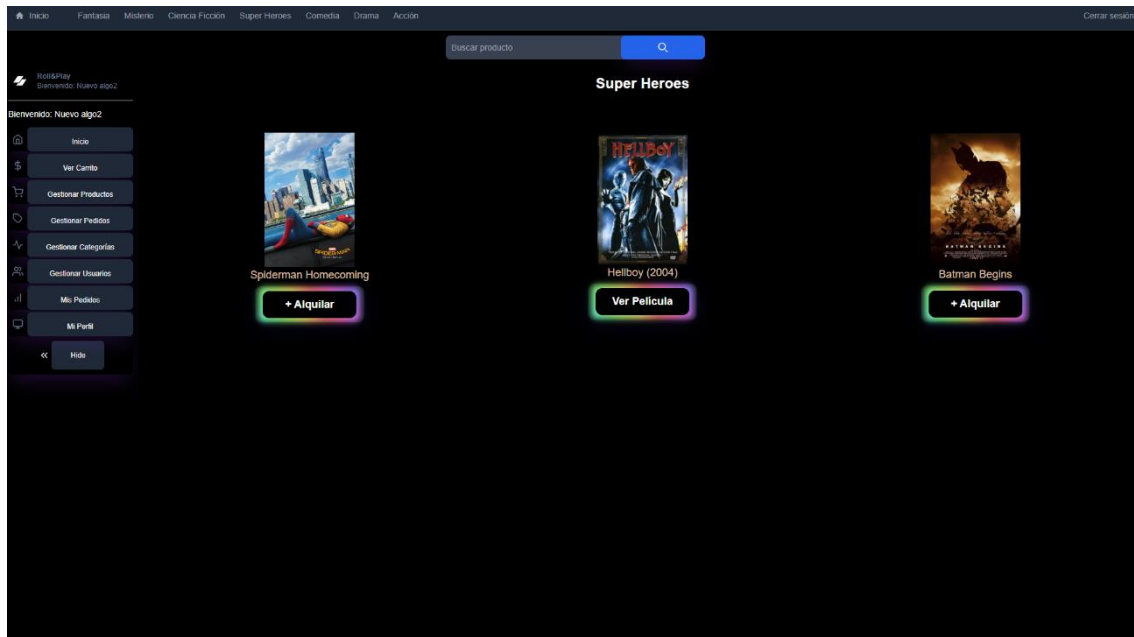
Página de búsqueda



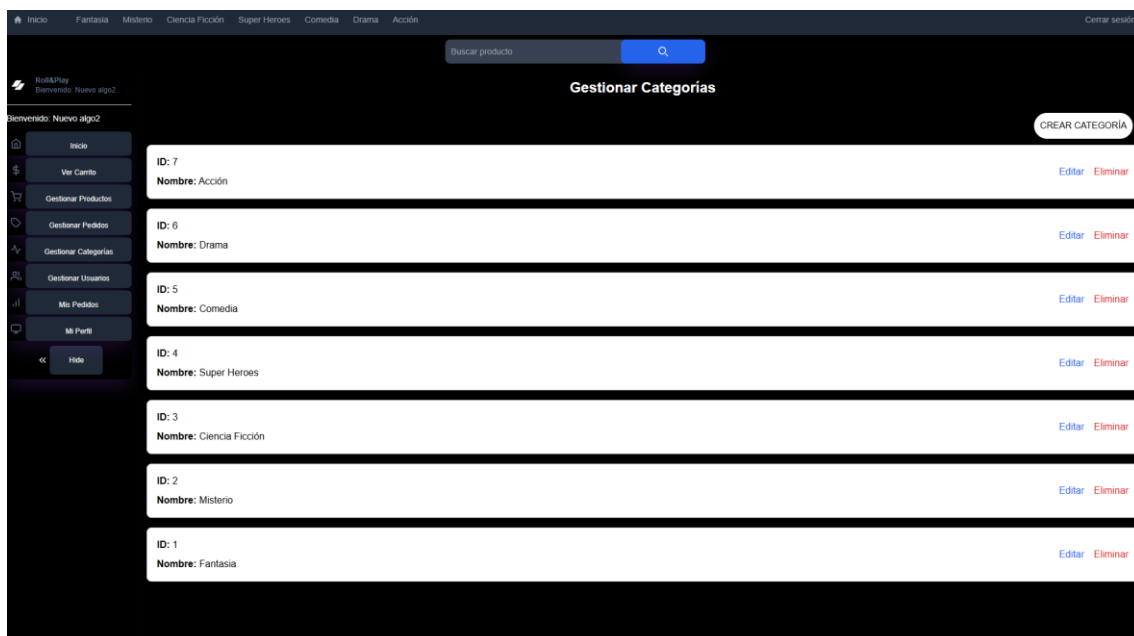
Carrito de la compra



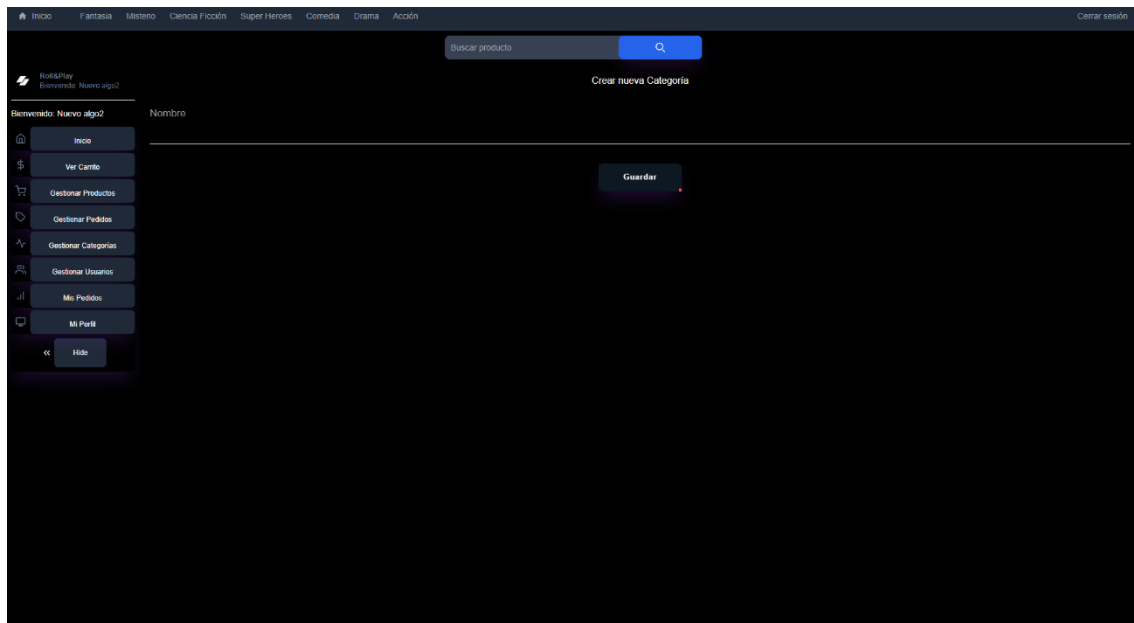
Página de categorías



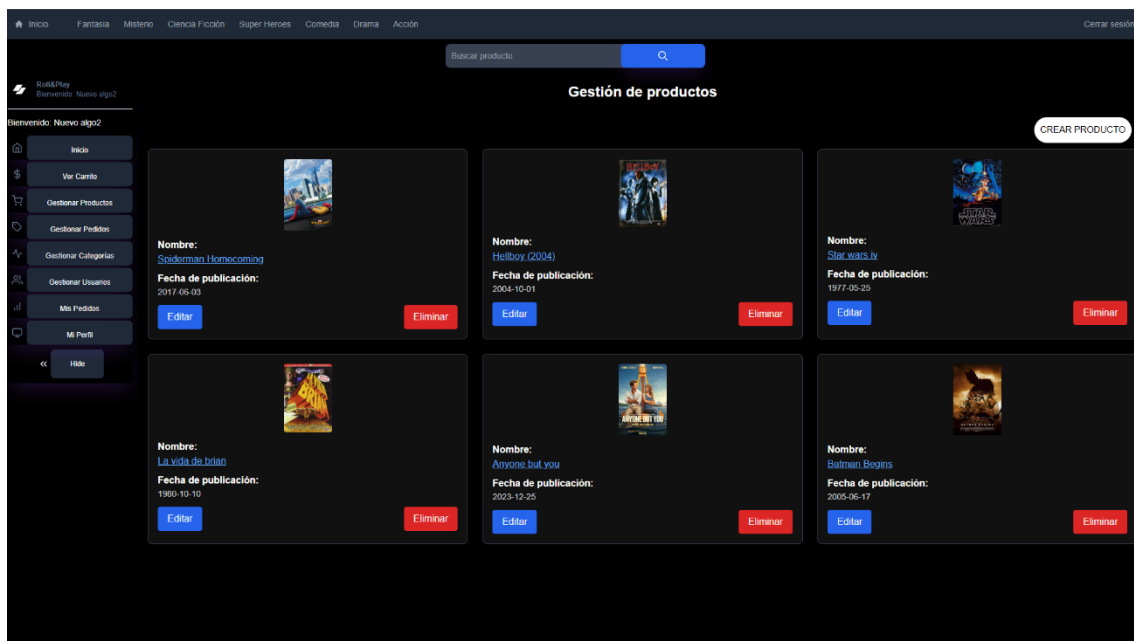
Gestión de categorías



Crear/Editar categorías



Gestión productos



Crear/Editar productos

Crear nuevo producto

Nombre

Descripción

Categoría
Fantasia

Fecha de publicación
dd/mm/yyyy

Imagen
Seleccionar archivo Ningún archivo seleccionado

Tráiler
Enlace al tráiler (YouTube o URL directa)
https://www.youtube.com/watch?v=...

Subir archivo de video (MP4)
Seleccionar archivo Ningún archivo seleccionado

Película
Enlace a la película (si es un enlace)
https://www.ejemplo.com/video.mp4

Subir archivo de video (si es un archivo MP4)

Gestión usuarios

Listado de Usuarios

ID: 1	Nombre: Nuevo	Apellidos: algo2	Email: nuevo@nuevo.com	Rol: admin	Editar Eliminar
ID: 2	Nombre: Pedro	Apellidos: calavera carrasco	Email: algo2@algo.com	Rol: user	Editar Eliminar
ID: 10	Nombre: Repetido	Apellidos: algo	Email: algo4@algo.com	Rol: admin	Editar Eliminar
ID: 11	Nombre: NuevoUsuario	Apellidos: algo	Email: algo6@algo.com	Rol: user	Editar Eliminar

Editar usuarios

Editar usuarios

Página de mis pedidos

Nº Pedido	Coste	Fecha	Estado
Nº Pedido: 30	Coste: 20	Fecha: 2025-04-29	Estado: Enviado
Nº Pedido: 29	Coste: 10	Fecha: 2025-04-29	Estado: Enviado

Página de perfil

Editar perfil de Nuevo

Nombre:

Apellidos:

Email:

Nueva contraseña (opcional):

Rol: Administrador

[Actualizar la cuenta](#)

[Eliminar Cuenta](#)

Historial de Pedidos

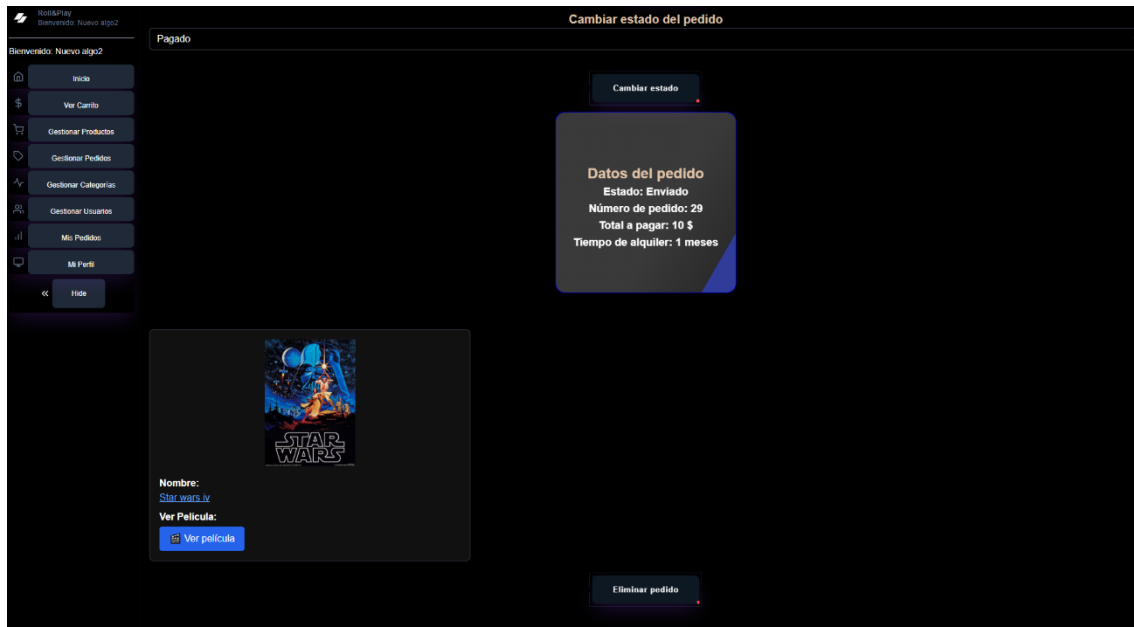
Nº Pedido	Coste	Fecha	Estado
32	40 €	2025-04-29	Enviado
31	10 €	2025-04-29	Enviado

Gestión de pedidos

Gestionar pedidos

Nº Pedido	Coste	Fecha	Estado
32	40	2025-04-29	Enviado
31	10	2025-04-29	Enviado
30	20	2025-04-29	Enviado
29	10	2025-04-29	Enviado

Página de detalles de pedidos



9.2 Criterios de accesibilidad

La accesibilidad web es fundamental para asegurar que todos los usuarios, independientemente de sus habilidades o limitaciones, puedan interactuar con la plataforma de manera efectiva. Para garantizar la accesibilidad, se han implementado los siguientes criterios:

Contrastes de color adecuados: Se han seleccionado colores con un buen contraste entre el texto y el fondo, como el blanco sobre fondo negro, para asegurar que los usuarios con discapacidades visuales puedan leer el contenido sin dificultad. El color **#B89676** y **#707070** se usan estratégicamente para no comprometer la visibilidad.

Contenido responsive: Se asegura que la web se adapta correctamente a dispositivos móviles y tabletas, con tamaños de texto y botones fácilmente legibles sin necesidad de hacer zoom.

Uso de etiquetas semánticas HTML: Se han utilizado etiquetas HTML5 semánticas (como `<header>`, `<footer>`, `<table>`, etc.) para ayudar a los usuarios de lectores de pantalla a navegar correctamente por el contenido.

9.3 Criterios de usabilidad

La usabilidad se refiere a la facilidad con que los usuarios pueden interactuar con el sistema, encontrar lo que necesitan y realizar acciones dentro de la plataforma. Para mejorar la experiencia del usuario, se han implementado los siguientes criterios:

Diseño claro y consistente: Se ha utilizado un diseño visual limpio y consistente a lo largo de todo el sitio, con un esquema de colores armonioso y una tipografía legible. Elementos clave como botones y menús tienen un diseño coherente, lo que facilita la navegación.

Interactividad intuitiva: Los elementos interactivos como botones, menús y enlaces son fácilmente identificables y responden de forma clara al ser seleccionados, asegurando que los usuarios comprendan cómo interactuar con ellos.

Optimización para dispositivos móviles: El diseño responsive asegura que los usuarios tengan una experiencia óptima tanto en dispositivos de escritorio como en móviles, adaptando los menús y el contenido según el tamaño de la pantalla.

Facilidad para completar tareas: Las tareas comunes, como la búsqueda de productos o el proceso de compra, han sido simplificadas al máximo, minimizando la cantidad de pasos requeridos y asegurando que el flujo de usuario sea directo y sin confusión.

9.4 Tipografía

Para todo el proyecto se ha utilizado la familia tipográfica **Arial, Helvetica, sans-serif**. Esta combinación ha sido elegida por su alta legibilidad, compatibilidad con la mayoría de navegadores y dispositivos, y porque es una fuente accesible que mejora la lectura para personas con dificultades visuales leves.

Se han aplicado los siguientes estilos tipográficos:

Títulos principales: Arial, tamaño grande (text-xl o text-2xl), peso bold o semibold.

Subtítulos y encabezados secundarios: Arial, tamaño medio (text-lg), peso medium.

Texto del cuerpo: Arial, tamaño base (text-base o text-sm), peso normal.

Enlaces y botones: Arial en mayúsculas o con transformación de texto (uppercase o tracking-wide), para mejorar su visibilidad.

9.5 Mapa de colores de proyecto

A continuación, se detalla el conjunto de colores utilizados en el proyecto, especificando su valor en **formato hexadecimal**, su **equivalente RGB** y su **nombre representativo**, así como los elementos donde han sido aplicados:

Hexadecimal	RGB	Nombre aproximado	Elementos donde se aplica
#000000	rgb(0, 0, 0)	Negro	Fondo general de la web, sidebar, menús
#FFFFFF	rgb(255, 255, 255)	Blanco	Texto principal, textos sobre fondos oscuros
#707070	rgb(112, 112, 112)	Gris	Fondo de tarjetas y bloques de contenido. También es usado en algunos párrafos y secciones informativas
#DC2626.	rgb(220, 38, 38))	rojo intenso	Botón del buscador, el perfil y del carrito
#2563EB	rgb(37, 99, 235)	Azul intenso	Botón de actualizar perfil
#6450E9	rgb(100, 80, 233)	morado	Botones de la página de aterrizaje

Estos colores fueron elegidos para lograr un contraste visual efectivo y una estética elegante, manteniendo buena legibilidad en todos los dispositivos. Además, el uso de tonos neutros como negro y gris se equilibra con colores cálidos y llamativos como el rojo y azul para destacar acciones y secciones clave.

9.6 Dispositivos/vistas para las que se ha diseñado el proyecto. Tecnología usada.

El proyecto ha sido diseñado con un enfoque **responsivo**, es decir, adaptable a distintos tamaños de pantalla, para asegurar una correcta visualización y experiencia de usuario tanto en **ordenadores**, como en **tabletas** y **dispositivos móviles**.

Dispositivos y vistas contempladas:

Ordenador de escritorio: se muestra el diseño completo con menús desplegados y mayor densidad de información.

Tabletas: la interfaz se adapta con menús más compactos y espaciado optimizado para pantallas medianas.

Móviles: los elementos se reordenan verticalmente, los menús se colapsan automáticamente y se prioriza la facilidad de navegación táctil.

Tecnología utilizada:

Tailwind CSS: framework de estilos CSS que permite usar clases utilitarias para diseño responsive con breakpoints (sm, md, lg, etc.), adaptando automáticamente el diseño según el ancho del dispositivo.

React: se ha usado React para crear componentes interactivos que responden al tamaño de pantalla.

JavaScript: se ha utilizado para detectar mensajes de error y hacer que desaparezcan después de un cierto tiempo.

Blade + Laravel: en la parte del servidor se han generado plantillas que cargan correctamente en cualquier resolución y dispositivo.

10. Mejoras posibles y aportaciones

Una mejora posible a realizar podría ser la mejora de la subida de videos ya que como se ha explicado anteriormente al subir un video desde tu dispositivo la subida suele tardar mucho si se trata de una película completa.

Otra mejora podría a ver sido la mejor organización de mi página de estilos css ya que las clases hay algunas que son difíciles de encontrar e incluso clases que ya no se usan en el código final y eso se podría haber gestionado mejor.

11 . Licencias y Copyright

Este proyecto ha sido desarrollado exclusivamente con fines educativos como parte de un Trabajo de Fin de Grado (TFG), sin ánimo de lucro.

Todos los derechos de autor de las imágenes, vídeos, películas, logotipos u otros contenidos utilizados en la aplicación pertenecen a sus respectivos autores y titulares legales. Su inclusión en esta plataforma tiene un propósito demostrativo para representar la funcionalidad del sistema, no comercial.

En cuanto al software empleado en el desarrollo del proyecto, se han utilizado tecnologías de código abierto y herramientas bajo licencias permisivas (como Laravel, React, TailwindCSS, etc.), respetando en todo momento sus términos de uso y distribución.

Cualquier creación original realizada dentro del proyecto (código, estilos personalizados, componentes visuales, etc.) queda bajo una licencia libre para fines educativos y de demostración, salvo que se indique lo contrario.