

16 DE MAYO DE 2024



CYNEPHILE

CARLOS FERRER MARTÍNEZ
DESARROLLO DE APLICACIONES WEB
2º DAW

ÍNDICE

RESUMEN DEL PROYECTO.	2
JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO.....	2
DESARROLLO DEL PROYECTO	3
CONCLUSIONES	16
LÍNEAS FUTURAS DE TRABAJO	17
BIBLIOGRAFÍA	18
ANEXO 1: GUÍA DE ESTILOS DESARROLLADA	19
ANEXO 2: MOCK-UP DE LA APP	19

RESUMEN DEL PROYECTO.

El objetivo de la aplicación Cynephile es proporcionar una herramienta más para que los usuarios puedan consultar los datos de sus películas y series favoritas.

Consiste en una aplicación compuesta por varias secciones, una de ellas la principal que nos muestra todo el contenido multimedia disponible para consultar además de apartados específicos para consultar de manera individual las series y las películas que se encuentran en la base de datos, el detalle y favoritos.

Abstract

The Cynephile app is a tool for users to easily access information about their favorite movies and series. It has sections for exploring movies and series, with an intuitive interface for searching and finding multimedia content.

JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO

Con el avance de las tecnologías cada vez más personas utilizan aplicaciones para la consulta de datos de sus películas y series preferidas, el objetivo de la propuesta es crear una aplicación referente en el mundo del cine, empezando como un proyecto pequeño de consulta de datos para posteriormente convertirlo en una de las principales plataformas de visualización de contenido del mercado, además Cynephile se dirige a todas aquellas personas amantes del mundo del cine y las series que encontrarán en ésta aplicación su mayor aliado a la hora de consultar y disfrutar de sus contenidos favoritos.

DESARROLLO DEL PROYECTO

a) Análisis de mercado y posible modelo de negocio.

Los principales referentes del mercado son las aplicaciones Filmaffinity e IMDb en cuanto a consulta de datos se refiere, si nos vamos a aplicaciones de visualización de contenido encontraremos que las más grandes son Netflix, HBO, Amazon Prime y Disney Plus.

Nuestra propuesta es hacer una combinación de los dos tipos de aplicaciones, como sabemos cada plataforma tiene su contenido para visualizar, sin embargo no se pueden consultar todos los datos disponibles, nuestra aplicación pretende mostrar todos los datos de contenido multimedia que hay en el mercado y además tener su propio catálogo de películas y series disponibles para consultar.

La suscripción mensual será menor que la de la competencia, con la incorporación de anuncios para rentabilizarla y permitir la rebaja del precio.

b) Metodologías utilizadas.

Para el desarrollo de este proyecto se ha utilizado la metodología ágil, basado en ciclos de entrega cortos con el objetivo de que la entrega del proyecto sea más eficiente.

La elección de esta metodología es debida a varios motivos:

1. Se optimiza el flujo de trabajo.
2. Aumento de la productividad del equipo.
3. Mayor satisfacción del cliente.

Este enfoque además es adecuado para mi aplicación debido a que es un proyecto en el que no dispongo de mucho tiempo para realizarlo, por lo tanto necesito un flujo de trabajo dinámico.

Dentro de las metodologías ágiles encontramos el método Kanban que es el aplicado en este caso, consiste en realizar un seguimiento visual del proyecto en

un panel de tareas (se ha utilizado un tablero trello) mediante la utilización de tarjetas de señalización divididas en tres bloques, un bloque de tareas pendientes donde se nos muestran aquellas tareas que faltan por realizar, en el siguiente bloque encontramos las tareas que se encuentran actualmente en proceso, por último tenemos el bloque de tareas finalizadas en el cual se nos presentan aquellas tareas que han sido realizadas con éxito.

La organización de las tareas y subtareas de trabajo de forma clara y visual ayuda a que tengamos claro cual es el flujo de trabajo a seguir y que las tareas se puedan realizar en el orden correcto y de la manera establecida.

Zendesk. (s.f.). Metodología Ágil: ¿Qué es y cómo se implementa en tu empresa? [Blog de Zendesk](#)

c) Descripción de los componentes de la aplicación.

En primer lugar se realizó el mock up del proyecto, esto se hace para que el desarrollador tenga una idea aproximada de como deben quedar todos los componentes a la hora del desarrollo de la aplicación.

Para la realización del mock up se utilizó la plataforma Miro. Miro cuenta con funciones propias completas y listas para usar que permiten a equipos de todos los tamaños desarrollar su visión con un toque creativo y colaborativo.

Esto significa dar soporte a los flujos de trabajo en todos los ámbitos del proyecto.

Además miro contiene una biblioteca de componentes para la maquetación tanto de aplicaciones de desarrollo web como aplicaciones de desarrollo móvil, los cuales son muy variados y fáciles de usar.

Al realizarlo plasmé las cuatro pantallas principales, una que permite al usuario loggearse y acceder al resto de pantallas de la aplicación, una página inicial que nos muestra un jumbotron con dos carruseles, uno para películas y otro para series. Las siguientes páginas que vemos son una con todas las series y otra con todas las películas disponibles.

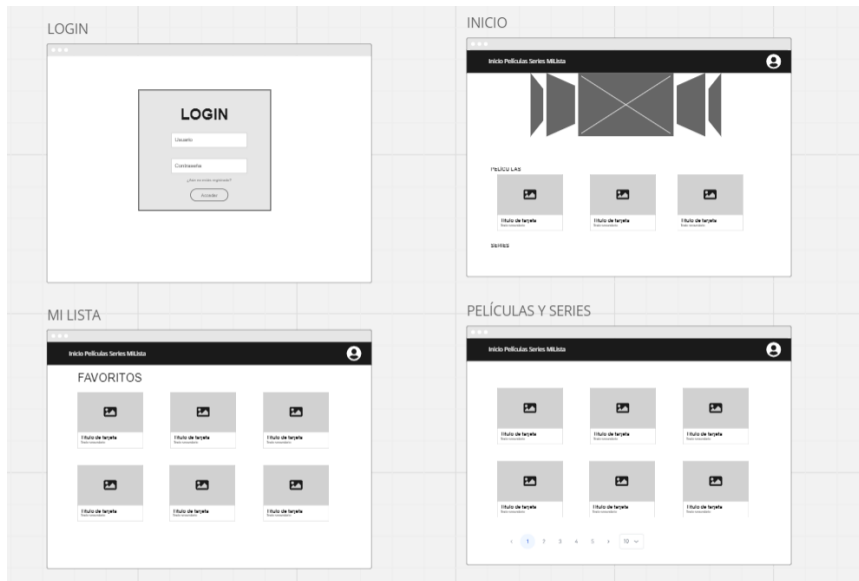


Figura 1: Mock up.

Arquitectura utilizada.

Para el desarrollo del proyecto se ha utilizado el patrón MVC (modelo-vista-controlador).

1. **Modelo:** Representa los datos y la lógica de negocio de la aplicación. Es responsable de gestionar el acceso a los datos, su validación y su procesamiento. El modelo no depende de la vista ni del controlador.
2. **Vista:** Es la interfaz de usuario de la aplicación. Muestra la información al usuario y maneja la interacción con él. La vista obtiene los datos necesarios del modelo, pero no realiza ningún procesamiento de datos.
3. **Controlador:** Actúa como intermediario entre el modelo y la vista. Responde a las solicitudes del usuario, interpreta las acciones realizadas en la vista y actualiza el modelo en consecuencia. El controlador es el encargado de manejar la lógica de negocio y coordinar la interacción entre el modelo y la vista.

Frontend.

El frontend de la aplicación se ha desarrollado con Angular 17. Angular es uno de los frameworks de desarrollo del lado del cliente más utilizados actualmente y junto con React es el más popular, encontrándose en España que la mayoría

de ofertas de trabajo piden el conocimiento de esta tecnología si se va a desarrollar desde el lado del cliente.

Entre sus principales ventajas nos permite la división de la aplicación mediante módulos y componentes, sobre todo en versiones posteriores a la 17, actualmente se pretende eliminar el uso de módulos y para ello se emplean los componentes standalone que son componentes que están diseñados para funcionar de manera independiente, sin depender de la estructura de una aplicación Angular más grande. Estos componentes pueden ser desarrollados y probados por separado, y luego integrados en cualquier proyecto Angular o incluso en proyectos que no utilicen Angular en absoluto.

La principal característica de los componentes standalone es su capacidad para encapsular funcionalidades específicas o elementos de la interfaz de usuario en unidades autónomas y reutilizables. Esto los hace ideales para compartir y mantener consistencia en el diseño y la funcionalidad en múltiples partes de una aplicación o incluso en diferentes aplicaciones.

Aunque para la realización del proyecto se ha intentado no emplearlos (únicamente se ha empleado uno para exportar componentes de una librería UI) considero debido al desempeño que realizo en las prácticas que no se debería de eliminar el empleo de módulos, esto es debido a que cuando queremos que una aplicación sea escalable la mejor opción es el empleo de módulos independientes que unidos entre si nos permiten crear una aplicación de grandes dimensiones bien estructurada y modulada, considero que dividirla únicamente por componentes aún no tiene una estructura clara que nos permita hacer un proyecto de gran tamaño.

En mi caso, al ser una aplicación pequeña he podido seguir utilizando componentes, sin embargo en un futuro si quisiera escalar la aplicación debería de convertirlo a módulos.

OpenAI. (2022). ChatGPT [Modelo de lenguaje]. <https://openai.com/chatgpt>

Librería UI

Para agilizar el desarrollo de la aplicación del lado del cliente se ha utilizado la librería de interfaz de usuario primeNG.

PrimeNG es una colección de componentes de interfaz de usuario (UI) de código abierto para Angular. Estos componentes están diseñados para ayudar a los desarrolladores a crear interfaces de usuario atractivas y funcionales de manera más eficiente. Algunas de las ventajas de PrimeNG incluyen:

1. **Amplia variedad de componentes:** PrimeNG ofrece una amplia gama de componentes UI listos para usar, que incluyen botones, tablas, gráficos, menús desplegables, cuadros de diálogo, calendarios, y muchos más. Esto permite a los desarrolladores construir interfaces de usuario completas y ricas en funcionalidades sin tener que crear cada componente desde cero.
2. **Facilidad de uso:** Los componentes de PrimeNG están diseñados para ser fáciles de usar y personalizar. Ofrecen una API intuitiva y bien documentada que facilita su integración en proyectos Angular existentes y su adaptación a las necesidades específicas del proyecto.
3. **Responsividad y compatibilidad con dispositivos móviles:** Los componentes de PrimeNG están optimizados para funcionar en una variedad de dispositivos y tamaños de pantalla, lo que garantiza una experiencia de usuario consistente y agradable en dispositivos móviles y de escritorio.
4. **Tema personalizable:** PrimeNG ofrece soporte para temas personalizables, lo que permite a los desarrolladores adaptar la apariencia de los componentes según el diseño y la marca de su aplicación. También proporciona una variedad de temas predefinidos listos para usar que pueden ser aplicados fácilmente a los componentes.
5. **Rendimiento optimizado:** Los componentes de PrimeNG están optimizados para ofrecer un rendimiento rápido y eficiente, lo que garantiza una experiencia de usuario fluida incluso en aplicaciones web complejas y de alto tráfico.

6. **Soporte activo y comunidad:** PrimeNG cuenta con una comunidad activa de desarrolladores que ofrecen soporte, comparten recursos y contribuyen al desarrollo continuo del proyecto. También cuenta con una documentación completa y ejemplos de código que facilitan el aprendizaje y la implementación de los componentes.

Backend.

Para el back se ha utilizado el framework Symfony el cual emplea PHP como lenguaje de programación y utiliza el MVC para el desarrollo de las aplicaciones.

Es una herramienta robusta y altamente flexible que permite a los desarrolladores construir aplicaciones web complejas y de alto rendimiento de manera eficiente. Algunas de las ventajas de Symfony incluyen:

1. **Arquitectura modular y escalable:** Symfony sigue un enfoque basado en componentes, lo que significa que está compuesto por un conjunto de librerías independientes que pueden ser utilizadas de manera modular según las necesidades del proyecto. Esto permite una arquitectura escalable y flexible, donde los desarrolladores pueden agregar o quitar funcionalidades fácilmente según sea necesario.
2. **Reutilización de código:** Symfony promueve la reutilización de código a través de sus componentes y bundles, lo que permite a los desarrolladores aprovechar soluciones probadas y evitar la duplicación de esfuerzos en el desarrollo de nuevas funcionalidades.
3. **Documentación extensa y comunidad activa:** Symfony cuenta con una documentación detallada y una comunidad activa de desarrolladores que ofrecen soporte y comparten recursos útiles. Esto facilita el aprendizaje del framework y la resolución de problemas durante el desarrollo de proyectos.
4. **Alto rendimiento y seguridad:** Symfony está diseñado para ofrecer un alto rendimiento y seguridad en aplicaciones web, gracias a su

arquitectura optimizada, técnicas de caching, herramientas de seguridad integradas y buenas prácticas recomendadas por la comunidad.

5. **Flexibilidad y personalización:** Symfony proporciona una gran flexibilidad y capacidad de personalización, permitiendo a los desarrolladores adaptar el framework según las necesidades específicas del proyecto y utilizar las herramientas y librerías que consideren más apropiadas.
6. **Integración con otras tecnologías:** Symfony se integra fácilmente con otras tecnologías y herramientas, lo que facilita la creación de aplicaciones complejas que requieren integración con bases de datos, servicios web, APIs externas, sistemas de cache, entre otros.

POSTMAN

Postman es una plataforma que nos permite comprobar que nuestras peticiones a la API funcionen correctamente, la he usado para comprobar mis endpoints.

BBDD

Para la base de datos se ha utilizado un sistema relacional, MySQL:

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto ampliamente utilizado en aplicaciones web y empresariales. Algunas de las ventajas de MySQL incluyen:

1. **Costo:** MySQL es de código abierto y gratuito para su uso, lo que lo hace atractivo para empresas y proyectos con presupuestos limitados. Además, ofrece ediciones comerciales con soporte adicional y características avanzadas para aquellos que lo necesiten.
2. **Rendimiento:** MySQL está diseñado para ofrecer un rendimiento óptimo, incluso en entornos de alto volumen y tráfico intenso. Su arquitectura interna está optimizada para la velocidad y la eficiencia, lo que permite tiempos de respuesta rápidos y un procesamiento eficiente de consultas.
3. **Escalabilidad:** MySQL es altamente escalable y puede manejar grandes volúmenes de datos y cargas de trabajo crecientes. Ofrece opciones para

la replicación, la partición de tablas y la distribución de carga, lo que permite escalar horizontal y verticalmente según sea necesario.

- 4. Compatibilidad:** MySQL es compatible con una amplia variedad de plataformas y sistemas operativos, incluyendo Windows, Linux, macOS y diversas distribuciones de Unix. También es compatible con numerosos lenguajes de programación y frameworks de desarrollo, lo que facilita su integración en aplicaciones existentes.
- 5. Seguridad:** MySQL ofrece varias características de seguridad para proteger los datos almacenados en la base de datos. Esto incluye autenticación de usuarios, control de acceso basado en roles, encriptación de datos en reposo y en tránsito, y registro de auditoría para el seguimiento de actividades.
- 6. Amplia comunidad y soporte:** MySQL cuenta con una gran comunidad de usuarios y desarrolladores que ofrecen soporte, comparten recursos y contribuyen al desarrollo continuo del proyecto. Además, hay una abundancia de documentación, tutoriales y recursos disponibles en línea para ayudar a los usuarios a aprovechar al máximo MySQL.

OpenAI. (2022). ChatGPT [Modelo de lenguaje]. <https://openai.com/chatgpt>

Docker

Docker es una plataforma de software que permite la creación, el despliegue y la gestión de aplicaciones en contenedores. Los contenedores son entornos ligeros y portátiles que contienen todo lo necesario para que una aplicación se ejecute, incluidas las bibliotecas, las herramientas del sistema, el código y las dependencias. Docker utiliza tecnologías de virtualización a nivel de sistema operativo para garantizar que las aplicaciones funcionen de manera consistente en diferentes entornos, lo que facilita el desarrollo, la implementación y la escalabilidad de aplicaciones. Además, Docker facilita la creación de entornos reproducibles y aislados, lo que mejora la seguridad y la confiabilidad de las aplicaciones.

En nuestro caso levantamos el back con symfony mediante la utilización de contenedores de docker.

d) Dificultades encontradas en la realización.

Las principales dificultades encontradas a la hora de realizar el proyecto han sido:

1. La versión de Angular 17, esto es debido a que este framework pretende eliminar el trabajo por módulos, y trabajar únicamente con componentes Standalone, cosa que aunque a priori puede parecer más rápido, si la aplicación sigue escalando se puede volver un problema.
2. La obtención del token de autenticación. Esta ha sido la parte que más me ha costado en la realización del proyecto, no porque sea difícil en sino porque hubo un pequeño error en un comando lanzado, un error difícil de detectar y a la vez muy fácil de solucionar si sabes de lo que se trata, sin embargo al no conocer la causa tuve que repetir el proceso de obtención del token sin éxito. A continuación muestro el problema:

```
# php bin/console lexik:jwt:generate-keypair
```

[OK] Done!

```
# php bin/console debug:router
```

Name	Method	Scheme	Host	Path
_preview_error	ANY	ANY	ANY	/_error/{code}.{_format}
app_media	GET	ANY	ANY	/api/getAllMedia
app_movies	GET	ANY	ANY	/api/getAllMovies
app_series	GET	ANY	ANY	/api/getAllSeries
app_media_by_id	GET	ANY	ANY	/api/getMediaById/{id}
api_login_check	ANY	ANY	ANY	/api/login_check

Figura 2: Ruta de la api login check.

Como podemos observar en la imagen anterior, para obtener el token he seguido los pasos de instalacion de LexikJWTAuthenticationBundle el cual nos permite obtener el token mediante una serie de comandos.

La documentación de Lexik es muy buena y precisa por lo que es muy fácil de seguir y el resultado es bastante satisfactorio sin embargo el problema fue a la hora de lanzar el comando para obtener el token de autenticación. A continuación vemos una imagen del problema surgido.

```
# curl -X POST -H "Content-Type: application/json" https://localhost:8000/api/login_check --data {"username":"carlos@gmail.com"},"password":"secret"}
curl: (35) OpenSSL SSL connect: SSL_ERROR_SYSCALL in connection to localhost:8000
```

En la figura anterior lanzo el comando que es necesario para el sistema operativo windows y la obtención del token de autenticación, sin embargo vemos que salta el error Open SSL ERROR SYSCALL, lo cual me hace buscar exhaustivamente por todo internet, sin embargo en ningún foro ni si quiera chat gpt me da una solución que funcione.

Los sitios web que implementan SSL tienen HTTPS en lugar de HTTP. HTTPS es el protocolo de transferencia de hipertexto para enviar datos de forma segura a través de internet, este está encriptado para aumentar la seguridad de la transferencia de datos.

Figura 3. Obtención del token.

Con la información recabada me doy cuenta de que estoy usando el protocolo de transferencia de hipertexto seguro, sin embargo el que necesito para mi aplicación es HTTP. Esto es debido a que yo no tengo lo que se conoce como “certificado SSL” que son los que permiten que los sitios web utilicen HTTPS.

Un certificado SSL es un archivo de datos alojado en el servidor de origen de un sitio web.d del servidor. La clave privada se mantiene secreta y segura. Los certificados SSL hacen posible la encriptación SSL/TLS y contienen la clave pública del sitio web, y la identidad del sitio web, junto con información relacionada.

El SSL, también conocido como TLS, es un protocolo para encriptar el tráfico de Internet y verificar la identidad del servidor. Los sitios web con una dirección web HTTPS utiliza SSL.

Cloudflare. (s.f.). ¿Qué es HTTPS? Cloudflare Learning Center.

Con la información anterior volví a lanzar el comando esta vez con el protocolo http, adjunto el comando:

```
curl -X POST -H "Content-Type: application/json" \
http://localhost:8000/api/login_check --data \
{"username\":\"carlos@gmail.com\",\"password\":\"secret\"}
```

Con el comando anterior y la figura 3 mostrada anteriormente vemos que ya se obtiene el token de autenticación, de tal manera que ya solo nos queda crear un login que permita al usuario obtenerlo.

3. Problema con el formulario de búsqueda:

Cuando nos encontramos en las rutas /init, /movies, o /series, el formulario de búsqueda se comportaba de manera correcta, sin embargo el problema viene cuando estamos en un detalle y decidíamos buscar otro detalle, de esa manera no funcionaba, puesto que siempre cogía la misma url con la misma id, en lugar de la nueva.

La solución: una condición que comprueba si estamos en detalle en ese momento, si estamos en detalle entonces navegamos al detalle con la nueva id, luego recargamos la página, de esa manera funciona correctamente.

```

redirectToDetailPage(id: string) {
  // Si estamos en detalle cogemos la nueva id y con then recargamos la pagina
  if (this.router.url.includes('/detail')) {
    this.router.navigate(['/detail', id]).then(() => {
      window.location.reload();
    });
  } else {
    this.router.navigate(['/detail', id]).then(() => {
      window.location.reload();
    });
  }
}

```

Figura 4. Solución al problema de la recarga de la app.

Otro problema fue el tiempo, debido a que en las prácticas estamos ocho horas diarias y en mi caso la exigencia es bastante elevada solo le podía dedicar horas al proyecto los fines de semana, en los cuales se trabajó todos y cada uno de ellos sin embargo si el proyecto es muy ambicioso te das cuenta de que no puedes realizarlo como te gustaría a menos de que dispongas de más horas.

Si solo fuese la aplicación no habría problema, las cosas se complican cuando tienes que realizar una guía de estilos, una memoria del proyecto, sincronizar Github con Trello, creación de ramas etc.

Por supuesto cualquier proyecto de desarrollo web requiere de todos esos procedimientos sin embargo es mucha carga de trabajo para una persona que en un plazo de tres meses tiene que trabajar y además en sus ratos libres realizarlo.

e) Resultados obtenidos

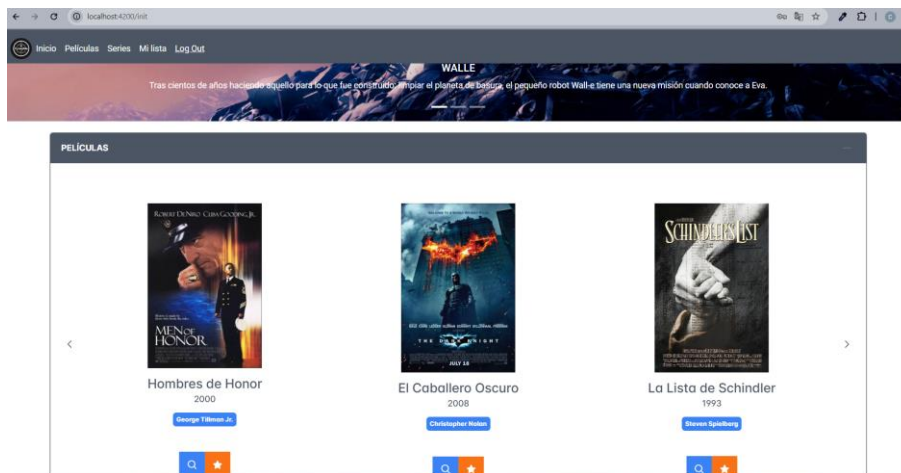


Figura 5. Vista general de la página de inicio de la aplicación.

Aunque como he explicado anteriormente el tiempo ha sido una de las principales desventajas a la hora de realizar el proyecto, estoy satisfecho con el resultado obtenido, debido a que me he querido centrar más en la parte del frontend que es a la que me he dedicado durante las prácticas, y he trabajado conceptos como los interceptores, los tokens de autenticación, los spinners de carga, los servicios, además de utilizar una de las librerías de UI más potentes como es primeNG, la cual facilita y reduce mucho el trabajo, derivando en un resultado final muy atractivo para la vista del usuario.

Me hubiese gustado dedicarle más tiempo al back y a la base de datos, sin embargo he tenido que dar prioridad al front para que el proyecto quedase lo más estético posible a la hora de la presentación.

CONCLUSIONES

El trabajo final de grado es una muy buena oportunidad para poner en práctica todo lo aprendido durante los dos años del módulo, sin embargo considero que el tiempo es clave para obtener el mejor de los resultados, mi punto de vista es que deberíamos centrarnos en el tfg las dos semanas en las que nos dedicamos a hacer el trabajo en equipo.

Por supuesto entiendo de la importancia del trabajo en equipo sobre todo en el mundo laboral, pero, a la hora de aprender es mejor que cada uno realice el trabajo individualmente, debido a que en todos los equipos siempre hay una o más personas que saben más que el resto y son los que se encargan de hacer la mayor parte del trabajo.

Por todo lo mencionado anteriormente considero que ese proyecto en grupo sería mejor eliminarlo y dedicarse desde ese momento al trabajo final de grado, ganando de esa manera dos semanas de dedicarse exclusivamente al proyecto personal, derivando seguro en un resultado más satisfactorio.

LÍNEAS FUTURAS DE TRABAJO

Las cosas que me gustaría mejorar de cara al futuro en la app Cynephile son las siguientes:

1. Mejora de la Base de Datos y Creación de Relaciones para Comentarios:
 - Implementar una estructura de base de datos más robusta mediante la inclusión de una tabla de comentarios. Esta tabla permitirá a los usuarios compartir sus opiniones y comentarios sobre películas y series dentro de la plataforma.
 - Establecer relaciones adecuadas entre la tabla de comentarios y las tablas existentes de películas/series y usuarios, utilizando claves foráneas. Esto facilitará la gestión y navegación de los comentarios dentro del sistema.
2. Ampliación con una Sección de Documentales:
 - Expandir la oferta de contenido de la plataforma mediante la inclusión de una sección dedicada a documentales. Esta nueva sección brindará a los usuarios la oportunidad de explorar y descubrir una amplia variedad de documentales sobre diversos temas.
 - Integrar la sección de documentales de manera cohesiva en la interfaz de usuario existente, proporcionando una experiencia de navegación fluida y coherente para los usuarios.
3. Migración de Datos de Películas Favoritas a la Base de Datos:
 - Mejorar la gestión de datos de usuario mediante la migración de la información sobre películas favoritas desde el almacenamiento local del navegador hacia la base de datos centralizada.
 - Crear una tabla en la base de datos para almacenar las relaciones entre los usuarios y las películas favoritas, garantizando así una gestión más segura y escalable de esta información.
4. Implementación de Certificados SSL para Seguridad Adicional:
 - Reforzar la seguridad de la plataforma mediante la implementación de certificados SSL (Secure Sockets Layer) para todas las comunicaciones entre el servidor y los clientes.

- Obtener un certificado SSL de una autoridad de certificación confiable y configurarlo en el servidor web para habilitar el acceso a la plataforma a través de HTTPS, proporcionando así una capa adicional de protección para los datos de los usuarios.

Estas líneas futuras de trabajo representan oportunidades clave para mejorar y expandir la funcionalidad de la plataforma, garantizando una experiencia más enriquecedora y segura para los usuarios a medida que la aplicación evoluciona.

BIBLIOGRAFÍA

- 1- Zendesk. (s.f.). Metodología Ágil: ¿Qué es y cómo se implementa en tu empresa? Recuperado de <https://www.zendesk.com.mx/blog/metodologia-agil-que-es/>
- 2- LexikJWTAuthenticationBundle. (s.f.). LexikJWTAuthenticationBundle Documentation - Installation. Recuperado de <https://github.com/lexik/LexikJWTAuthenticationBundle/blob/3.x/Resources/doc/index.rst#installation>
- 3- Cloudflare. (s.f.). ¿Qué es HTTPS? Recuperado de <https://www.cloudflare.com/es-es/learning/ssl/what-is-https/>
- 4- ChatGPT. (s.f.). Recuperado de <https://chatgpt.com/>

ANEXO 1: GUÍA DE ESTILOS DESARROLLADA

En este anexo se presenta la Guía de Estilos desarrollada como parte de este Trabajo de Fin de Grado. Esta guía proporciona directrices detalladas para la presentación visual y el formato de documentos, incluyendo aspectos como la tipografía, los logos, colores entre otros.

La Guía de Estilos se diseñó con el objetivo de mantener la coherencia y la profesionalidad en la presentación de documentos relacionados con el tema de estudio. Su aplicación se recomienda para todos los documentos relacionados con el proyecto de investigación.

Puedes acceder a la Guía de Estilos completa en el siguiente enlace:
<https://github.com/CarlosFerrer07/CYNEPHILE/tree/main/guiaEstilos>

ANEXO 2: MOCK-UP DE LA APP

En este anexo se presenta el mock-up desarrollado para el Trabajo de Fin de Grado. En el mock-up encontrarás una representación similar al resultado final de la app de Cynephile.

Se ha desarrollado con la app Miro, la cual tiene una amplia biblioteca de componentes para desarrollar tus ideas.

Puedes acceder al Mock-up con el siguiente enlace:

https://miro.com/app/board/uXjVKbbqkuQ=?share_link_id=6566989174

68