Forma, Círculo

Descripción generada automáticamente

Texto

Descripción generada automáticamente Imagen que contiene Texto

Descripción generada automáticamente

CYNEPHILE

16 de mayo de 2024

CARLOS FERRER mARTÍNEZ

DESARROLLO DE APLICACIONES WEB

2º DAW

**ÍNDICE**

1. **RESUMEN DEL PROYECTO.**
2. **JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO.**
3. **DESARROLLO DEL PROYECTO:**
4. **Análisis del mercado y posible modelo de negocio.**
5. **Metodologías utilizadas.**
6. **Descripción de los componentes de la aplicación.**
7. **Dificultades encontradas en la realización.**
8. **Resultados obtenidos.**
9. **CONCLUSIONES.**
10. **LÍNEAS FUTURAS DE TRABAJO.**
11. **BIBLIOGRAFÍA.**
12. **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 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.

1. **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.

1. **DESARROLLO DEL PROYECTO**
2. 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.

1. Metodologías utilizadas.

Para el desarrollo de este proyecto se ha utilizado la metodología agil, 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.

1. 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 son una con todas las series y otra con todas las películas disponibles.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

*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.

Ente 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.

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.

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.

**6. BIBLIOGRAFÍA**

[**https://www.zendesk.com.mx/blog/metodologia-agil-que-es/**](https://www.zendesk.com.mx/blog/metodologia-agil-que-es/)

Imagen que contiene Texto

Descripción generada automáticamente



Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

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