



# FilmotecaCLM

FilmotecaCLM.

Autor: Óscar Viana López.

Desarrollo de Aplicaciones Multiplataforma.

Tutor: Pablo Lledó Martínez.

## 1. Abstract:

Este proyecto presenta una aplicación que se conecta a las API de OMDB y TMDb, permitiendo a los usuarios crear cuentas, buscar películas y gestionar listas personalizadas de películas pendientes, vistas y favoritas. Los usuarios pueden valorar películas y visualizar las calificaciones de las APIs y otros usuarios registrados. Además, se proporciona la capacidad de editar perfiles, cambiando apodos (Nick) y avatares, incluso cargando nuevos avatares. El objetivo es establecer una comunidad de cinéfilos donde los usuarios no solo pueden buscar información, sino también calificarla a su gusto y expresar sus opiniones en el tablón de comentarios de cada película.

This project introduces an application that connects to the OMDB and TMDb APIs, enabling users to create accounts, search for movies, and manage customized lists of pending, watched, and favorite movies. Users can rate movies and view ratings from both the APIs and other registered users. Furthermore, the application allows profile editing, including changes to nicknames (Nick) and avatars, with the option to upload new avatars. The goal is to foster a community of movie enthusiasts where users can not only search for information but also rate it according to their preferences and share their opinions on the dedicated comments board for each movie.

## Contenido

1. Abstract: .....	2
2. Justificación del proyecto:.....	6
3. Introducción: .....	7
3.1 Interactividad y Participación del Usuario: .....	7
3.2 Recomendaciones Inteligentes:.....	7
3.3 Personalización del Perfil:.....	8
3.4 Integración con OMDb:.....	8
3.5 Implementación en Biblioteca Pública: .....	8
4. Objetivos: .....	9
4.1 Establecer Sistema de Cuentas de Usuario: .....	9
4.2 Desarrollar Buscador Eficiente: .....	9
4.3 Implementar Algoritmo de Recomendación: .....	9
4.4 Habilitar Sistema de Calificación:.....	9
4.5 Facilitar Función de Comentarios: .....	9
4.6 Diseñar Interfaz Intuitiva de Perfil: .....	10
5. Diseño:.....	11
5.1 Interfaz Gráfica: .....	11
5.1.1 Vista de usuarios: .....	11
5.1.2 Vista de administradores: .....	13
5.1.3 Manual del usuario: .....	15
5.2 Implementación de Interfaz Gráfica .....	16
5.3 Diagrama de casos de uso: .....	17
6. Base de Datos .....	19
6.1 Base de datos utilizada .....	19
6.2 Opciones descartadas .....	20

6.3 Esquema Modelo Entidad-Relacion.....	21
6.4 Tablas creadas:.....	21
6.4.1 Tabla Usuario: .....	21
6.4.2 Tabla Peticiones: .....	21
6.4.3 Tabla Avatar: .....	22
6.4.4 Tabla PelisEstado:.....	22
6.4.5 Tabla Comentario:.....	22
6.4.6 Tabla Películas: .....	22
6.4.7 Tabla Géneros: .....	22
6.4.8 Tabla Borrados: .....	22
7. Hilos y concurrencia: .....	23
8. Perfiles de Usuario .....	24
9. Desarrollo: .....	26
9.1 Metodología:.....	26
9.2 Proceso de desarrollo: .....	27
9.2.1 Fase 1.....	27
9.2.2 Fase 2:.....	29
9.2.3 Fase Final:.....	31
9.3 Herramientas y tecnologías: .....	32
9.3.1 Desarrollo de la aplicación: .....	32
9.3.2 Documentación del código: .....	32
9.3.3 Instalador: .....	32
9.3.4 Gestión de la base de datos: .....	32
9.3.5 Herramientas dentro de la aplicación:.....	33
9.3.6 Integración de APIs Externas:.....	33
9.3.7 Base de Datos y rendimiento: .....	33

9.4 Uso de Inteligencia Artificial .....	34
9.4.1 Generador de Imágenes:.....	34
9.4.2 Identificación y Corrección de Errores:.....	34
9.5 Problemas Encontrados y Decisiones Propias .....	35
9.5.1 Colocación de Avatares de una cuadrícula .....	35
9.5.2 Películas con el Mismo Director y Recomendaciones.....	35
10. Desarrollo Empresarial.....	37
11. Mejora y Actualizaciones de FilmotecaCLM: .....	38
12. Adaptación de proyecto a plataformas móviles .....	40
12.1 Rediseño de la Interfaz de Usuario .....	40
12.2 Sustitución de Funcionalidades no Compatibles .....	40
12.3 Elección de Plataforma Multiplataforma o Nativa .....	40
12.4 Adaptación del Código XAML .....	40
12.5 Optimización del Rendimiento .....	41
12.6 Integración de la Base de Datos en un Servidor .....	41
13. Conclusiones: .....	42
14. Webgrafía.....	43
14.1 Página oficial API OMDB (Open Movie DataBase).....	43
14.2 Página oficial API TMDb (The Movie Database) .....	43
14.3 Creación del Instalador: .....	43
14.4 Generador de Imágenes mediante IA:.....	43
15. Anexos:.....	44
15.1 Manual de usuario: .....	44
15.2 Desarrollo Empresarial: .....	44
15.3 Repositorio del Proyecto en GitHub: .....	44

## 2. Justificación del proyecto:

La concepción de este proyecto surgió a raíz de la inspiración proporcionada por la Biblioteca Pública del Estado de Cuenca, la cual alberga una amplia selección de películas las cuales me motivaron la idea de desarrollar una aplicación que no solo facilitara el acceso a información detallada sobre las películas, sino que también fomentara la interacción entre usuarios, permitiéndoles compartir sus opiniones y críticas acerca de sus gustos cinéfilos.

La creación de una aplicación de este tipo se fundamenta en la necesidad de enriquecer la experiencia de los usuarios al explorar y disfrutar de la oferta cinematográfica proporcionada por la Biblioteca Pública. La posibilidad de que los usuarios puedan expresar sus opiniones y valoraciones contribuirá a formar una comunidad virtual de cinéfilos, fomentando el intercambio de ideas y creando un espacio participativo en torno al séptimo arte.

En resumen, este proyecto no solo busca facilitar el acceso a información cinematográfica, sino también impulsar una experiencia más interactiva y social para los usuarios, promoviendo la participación activa y la construcción de una comunidad dedicada al cine.

### 3. Introducción:

En un entorno donde la tecnología y la experiencia cinematográfica convergen, surge la necesidad de desarrollar un buscador de películas innovador que no solo simplifique la búsqueda de contenidos en una biblioteca pública con una vasta colección, sino que también transforme la forma en que los usuarios interactúan con el séptimo arte. Este proyecto se centra en la creación de una aplicación de búsqueda y recomendación de películas, enriquecida con características interactivas y personalización, diseñada para ser implementada en una biblioteca pública con una extensa variedad de títulos.

A través de una revisión bibliográfica, se ha identificado la creciente importancia de las plataformas de búsqueda y recomendación de películas en la era digital. Las expectativas de los usuarios han evolucionado más allá de la simple visualización de contenidos, buscando experiencias más participativas y personalizadas. La aplicación se basará en el servicio OMDb para acceder a una amplia base de datos de películas, aprovechando su extensa cobertura y detalles detallados.

Las ideas clave que fundamentan este proyecto incluyen:

#### 3.1 Interactividad y Participación del Usuario:

La interacción del usuario va más allá de la simple búsqueda, permitiendo la valoración y comentario de películas. La inclusión de listas personalizadas como "Vistas", "Pendientes" y "Favoritas" busca empoderar a los usuarios para organizar y compartir sus preferencias.

#### 3.2 Recomendaciones Inteligentes:

La aplicación empleará algoritmos de recomendación para sugerir películas adaptadas al gusto individual del usuario y a las tendencias emergentes en la comunidad de usuarios de la biblioteca.

### 3.3 Personalización del Perfil:

Reconociendo la importancia de la expresión personal, la aplicación permitirá a los usuarios personalizar su perfil, cambiando avatares y nombres de usuario para una experiencia única y diferenciada.

### 3.4 Integración con OMDb:

La utilización de la API OMDb como fuente principal de datos garantiza una búsqueda exhaustiva y detallada, proporcionando información precisa sobre las películas en la biblioteca pública.

### 3.5 Implementación en Biblioteca Pública:

La aplicación está diseñada para ser implementada en una biblioteca pública, brindando a los usuarios acceso a una amplia selección de películas y mejorando su experiencia de descubrimiento.

Este proyecto busca no solo ser una herramienta de búsqueda, sino un espacio interactivo donde la comunidad de usuarios de la biblioteca pueda compartir sus pasiones cinematográficas y descubrir nuevas joyas del cine. A continuación, se detallarán los objetivos específicos y la metodología que guiará la realización de este proyecto único y dinámico.



## 4. Objetivos:

### 4.1 Establecer Sistema de Cuentas de Usuario:

Implementar un sistema de registro de cuentas de usuario que distinga entre diferentes roles, permitiendo la personalización de la experiencia según las preferencias individuales.

### 4.2 Desarrollar Buscador Eficiente:

Crear un buscador de películas eficiente que posibilite a los usuarios acceder rápidamente a la información deseada, mejorando la experiencia de búsqueda en la extensa biblioteca de la aplicación.

### 4.3 Implementar Algoritmo de Recomendación:

Desarrollar un algoritmo de recomendación de películas basado en las preferencias de los usuarios, brindando sugerencias personalizadas que se ajusten a sus gustos cinematográficos.

### 4.4 Habilitar Sistema de Calificación:

Establecer un sistema integral de calificación y clasificación de películas, permitiendo a los usuarios expresar sus opiniones y contribuir a la valoración general de las películas en la plataforma.

### 4.5 Facilitar Función de Comentarios:

Facilitar la interacción entre usuarios mediante la implementación de una función de comentarios en películas, fomentando la construcción de una comunidad cinéfila en línea y el intercambio de opiniones.

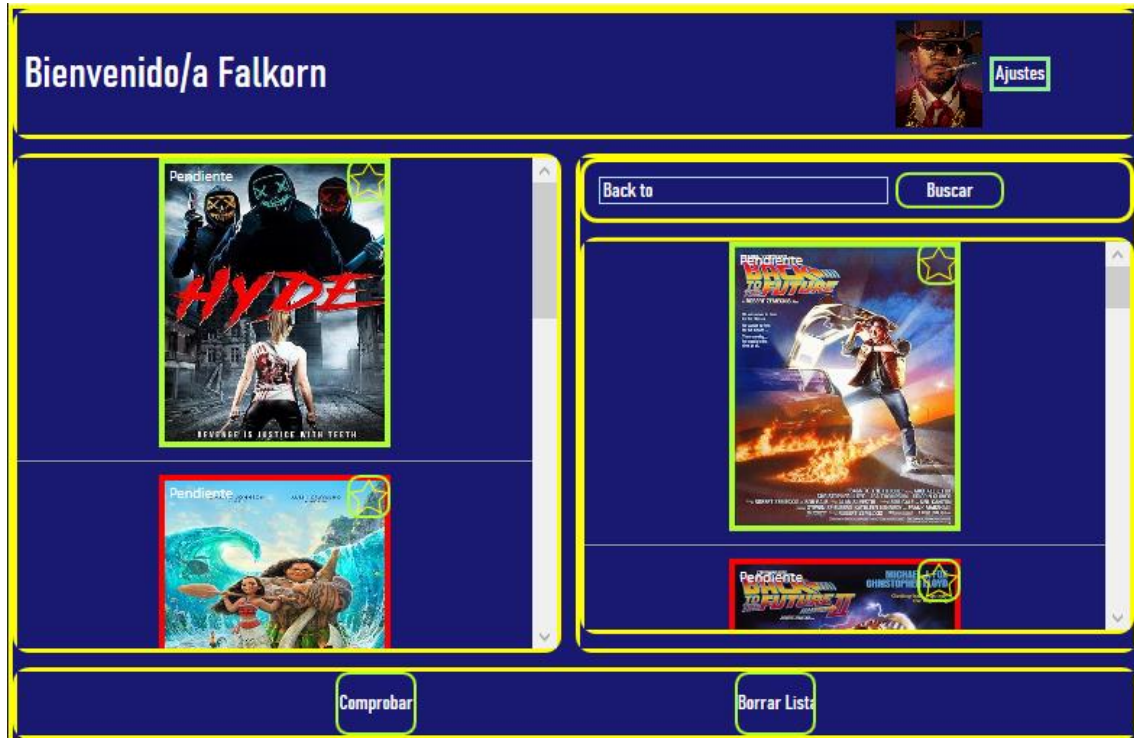
#### 4.6 Diseñar Interfaz Intuitiva de Perfil:

Crear una interfaz intuitiva para la configuración del perfil de usuario, posibilitando que los usuarios personalicen su experiencia a través de cambios en el avatar, el nombre de usuario y otras preferencias individuales.

## 5. Diseño:

### 5.1 Interfaz Gráfica:

#### 5.1.1 Vista de usuarios:



(Captura de pantalla página WPF para cualquier usuario).

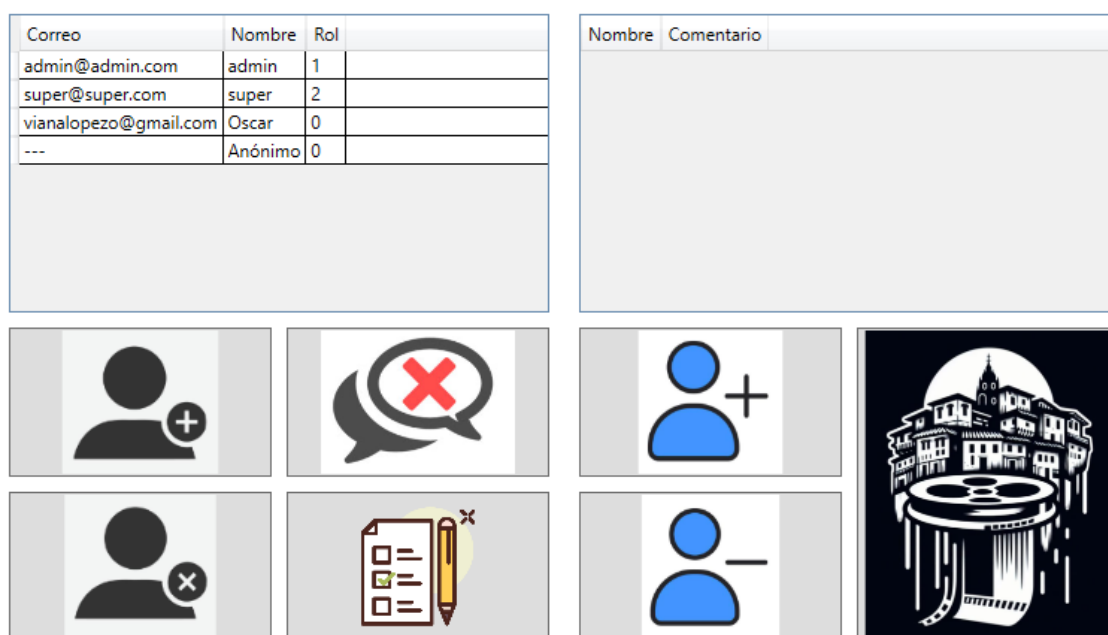
La elección estratégica de los colores MidnightBlue y Yellow como predominantes en la aplicación no solo obedece a una estética visual atractiva, sino que también se fundamenta en principios de diseño que buscan crear una armonía cautivadora y un sello distintivo para la identidad de la aplicación. La combinación de MidnightBlue, con su profundo tono azul oscuro, y Yellow, un tono vibrante y enérgico, genera un contraste equilibrado que resalta la interfaz de manera elegante y agradable a la vista. Esta paleta cromática no solo busca atraer la atención del usuario, sino también transmitir sensaciones de confianza, calidez y modernidad.

El color MidnightBlue se ha seleccionado por su tono sereno y sofisticado, evocando la tranquilidad que se desea asociar con la experiencia del usuario al explorar la aplicación. Por otro lado, el Yellow se introduce como un acento vibrante y optimista, infundiendo energía y vitalidad a la interfaz. Esta combinación crea una experiencia visual que es tanto estimulante como equilibrada, asegurando una conexión emocional positiva con el usuario.

Además, la fuente Bahnschrift Condensed ha sido elegida como la tipografía oficial de la aplicación. Su estilo moderno y condensado aporta una apariencia contemporánea y legible a los elementos textuales, mejorando la coherencia visual en todo el diseño. La elección de esta tipografía no solo busca la estética, sino también la funcionalidad, garantizando una presentación clara y eficiente de la información.

En conjunto, la combinación de colores MidnightBlue y Yellow con la tipografía Bahnschrift Condensed no solo crea una interfaz visualmente atractiva, sino que también establece un sello de identidad distintivo para la aplicación. Esta cuidadosa elección de elementos visuales contribuye a la creación de una experiencia de usuario única y memorable, reforzando la presencia de la aplicación en la mente de los usuarios.

### 5.1.2 Vista de administradores:



(Captura de pantalla página WPF únicamente para Administradores)

En lo que respecta al diseño de la interfaz gráfica destinada al administrador, se ha tomado la decisión estratégica de mantener el formato por defecto. Este enfoque se adopta con el propósito de establecer una clara diferenciación respecto al usuario genérico, proporcionando al administrador una experiencia visual y funcional única y específica para sus tareas administrativas.

La elección de elementos DataGrid se fundamenta en la necesidad de presentar información de manera estructurada y eficiente. Al emplear un color blanco grisáceo en combinación con esquinas más cuadriculadas, se logra potenciar la percepción del administrador, sugiriendo una sección dedicada a la administración de datos con un aire profesional. Este diseño busca optimizar la experiencia del administrador al proporcionar una interfaz visualmente clara y orientada a la gestión eficiente de usuarios y comentarios dentro de la aplicación.

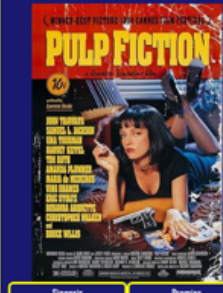
Para comprender mejor la disposición de la interfaz del administrador, consideremos la página principal. Esta se compone de dos DataGrids que presentan listados de usuarios y comentarios respectivamente. Además, se integran botones que permiten la eliminación de usuarios o comentarios seleccionados, así como la apertura de un formulario nuevo destinado a la inserción de usuarios. Un botón adicional revela una sección de la aplicación que muestra una ventana nueva con la parte de usuarios normales, manteniendo la hoja de estilos con los colores y la fuente previamente establecidos. Es importante destacar que la parte dedicada a los administradores mantiene el formato por defecto en la página XAML, asegurando así una presentación uniforme y profesional en su gestión de datos.

### 5.1.3 Manual del usuario:

## INFORMACIÓN DE LA PELÍCULA

Podremos acceder a esta página haciendo clic izquierdo sobre el póster de una de las películas. La página estará dividida en tres secciones.

### Sección Información



Título:	Pulp Fiction	Duración:	154 min
Director:	Quentin Tarantino	Genero:	Crime, Drama
Año:	1994	País:	United States

Reparto

Calificación

8.9%

50%

Sinopsis

Premios

En esta sección, podremos examinar la información detallada de la película seleccionada, junto con la calificación general proporcionada por la comunidad de usuarios en internet, así como una valoración individual y compartida entre los usuarios de la aplicación. Los usuarios tendrán la opción de otorgar su propia valoración a la película y agregarla a sus listas de Pendientes, Vistas y Favoritas.

(Un apartado del manual del usuario)

En lo que respecta al manual del usuario, se ha tomado la decisión consciente de mantener una paleta de tonos azulados. Esta elección se fundamenta en el intento de lograr una sensación de sincronía visual con la aplicación, incluso considerando que el manual se presenta en formato PDF. La coherencia cromática busca ofrecer a los usuarios una experiencia armoniosa y familiar, reflejando la identidad estilística de la aplicación.

Para estructurar de manera atractiva el contenido del manual, se ha asignado un tono azul claro al fondo de la página, proporcionando un trasfondo visual que facilita la lectura y crea un entorno agradable. El título principal, destacado como título 1, ha adoptado un tono azul oscuro, empleando la fuente Upheaval TT para acentuar su presencia con elegancia y claridad.

Los títulos de nivel 2 han sido enriquecidos con un tono azul clásico, utilizando la fuente Elephant para conferirles una apariencia distintiva y fácil de seguir. Por otro lado, el texto explicativo ha sido estratégicamente mantenido en un tono negro, ya que esta elección contribuye a una lectura más legible y cómoda. La fuente Oswald ha sido seleccionada para el texto explicativo, asegurando una presentación clara y moderna.

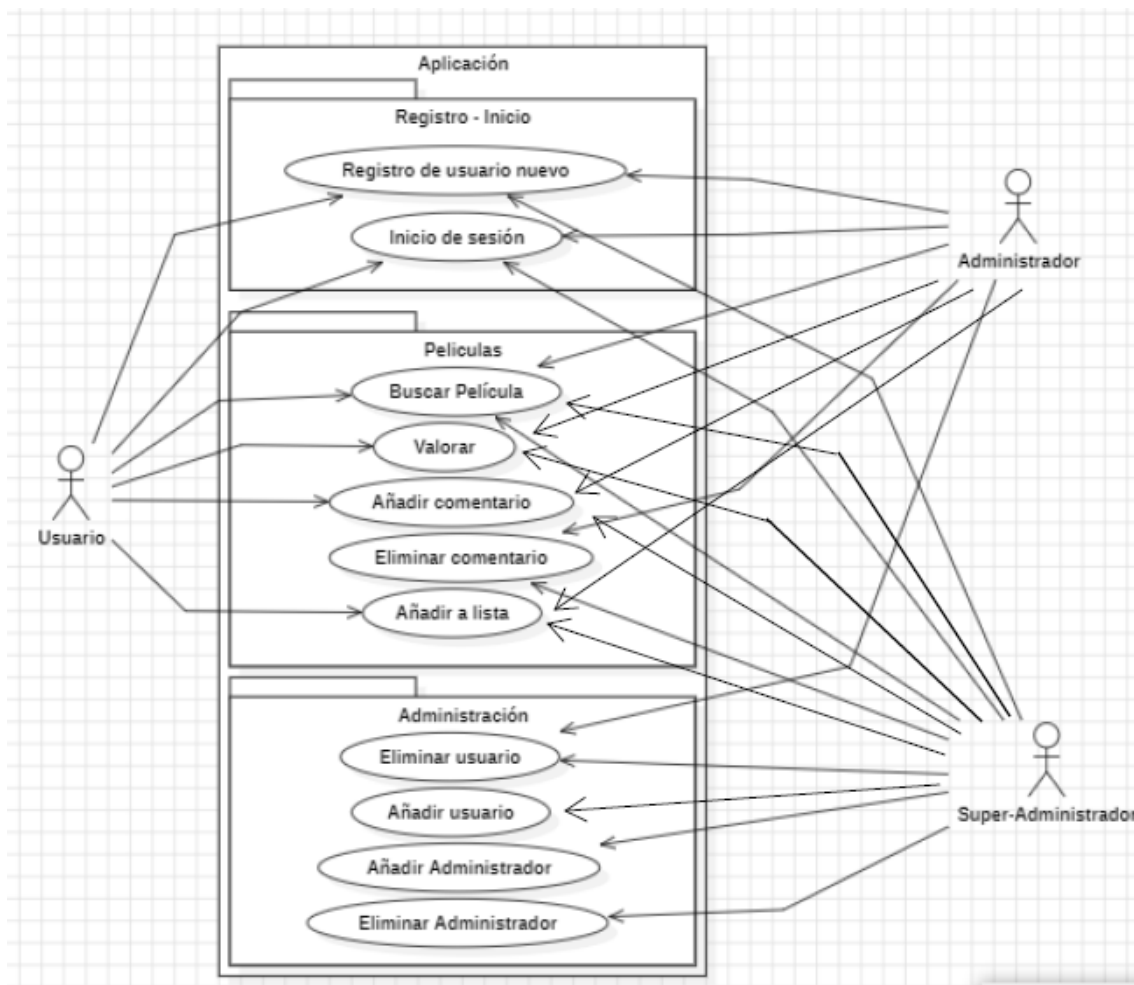
Esta cuidadosa combinación de colores y fuentes no solo persigue la estética visual, sino que también se orienta a mejorar la comprensión y accesibilidad del contenido del manual del usuario. La atención a estos detalles pretende proporcionar una guía clara y atractiva, reforzando así la experiencia completa del usuario con la aplicación.

## 5.2 Implementación de Interfaz Gráfica

La elección para desarrollar FilmotecaCLM se inclinó hacia un proyecto WPF utilizando Microsoft Visual Studio 2022 como la principal herramienta de desarrollo, aunque se contemplaron alternativas como Windows Forms y UWP (Universal Windows Platform). La selección de WPF se basó en sus sólidas capacidades gráficas, respaldadas por una excelente compatibilidad multimedia. Además, la clara separación entre la interfaz gráfica y el desarrollo en C# contribuye a la organización estructurada del código, evitando la complejidad innecesaria. Vale la pena mencionar que WPF también ofrece un robusto sistema de enlace de datos avanzado, simplificando la transferencia de información entre la lógica del código y la interfaz de usuario.



### 5.3 Diagrama de casos de uso:



(Diagrama de casos de uso)

En este esquema de casos de uso, se detallan las funciones principales de cada tipo de usuario:

#### **Usuario Normal:**

Las funciones son simples y se centran en acciones básicas como registrarse, iniciar sesión, buscar películas, asignar valoraciones, agregar películas a listas y comentar.

#### **Usuario Administrador:**

Aparte de las funciones del usuario normal, el administrador tiene la capacidad de eliminar usuarios, siempre que no sean de tipo administrador o superadministrador, y borrar comentarios.

**Usuario Super-Administrador:**

Este tipo de usuario es único y tiene todas las funciones de los usuarios anteriores. Además, cuenta con el privilegio adicional de poder agregar y eliminar administradores.

## 6. Base de Datos

### 6.1 Base de datos utilizada

FilmotecaCLM ha sido concebida como una aplicación independiente de servidores externos, y con este propósito se ha elegido implementar una base de datos embebida. En particular, se ha optado por utilizar SQLite debido a los diversos beneficios que aporta a la funcionalidad de la aplicación.

SQLite es una opción idónea por su naturaleza liviana y su capacidad para funcionar sin necesidad de un servidor centralizado. Algunos de los beneficios clave de SQLite incluyen su simplicidad de implementación, eficiencia en el manejo de datos, y su capacidad para ser integrado directamente en la aplicación, evitando así la dependencia de un servidor externo.

Esta elección se ha fundamentado en la adaptabilidad de SQLite, que permite que la aplicación funcione eficientemente como una filmoteca para una biblioteca específica. No obstante, cabe destacar que esta solución podría integrarse sin inconvenientes en cualquier ordenador personal, garantizando así una experiencia de usuario fluida y sin las complejidades asociadas a la dependencia de servidores externos.

SQLite presenta notables ventajas en comparación con otros sistemas de bases de datos, convirtiéndose en una elección óptima para FilmotecaCLM. En primer lugar, su naturaleza autónoma elimina la necesidad de un servidor externo, simplificando significativamente la implementación y la administración. Además, la arquitectura sin servidor de SQLite reduce la complejidad operativa, brindando una solución ligera y eficiente que no requiere configuraciones extensas ni mantenimiento continuo.

Otro beneficio clave es la portabilidad inherente de SQLite. Al ser una base de datos embebida, se integra fácilmente en la aplicación sin requerir instalaciones adicionales o configuraciones complicadas, lo que facilita su implementación en diversos entornos, incluyendo bibliotecas y ordenadores personales. La eficacia en la gestión de datos, la rapidez en las consultas y su capacidad para trabajar con conjuntos de datos de tamaño moderado hacen que SQLite sea especialmente adecuado para aplicaciones de tamaño medio como FilmotecaCLM.

Además, su estructura transaccional y la gestión eficiente de concurrencia contribuyen a mantener la integridad de los datos, asegurando una operación confiable y consistente. En conjunto, estas características hacen de SQLite una elección estratégica para FílmotecaCLM, garantizando un rendimiento robusto y una experiencia de usuario fluida frente a otras opciones de bases de datos más complejas y pesadas.

## 6.2 Opciones descartadas

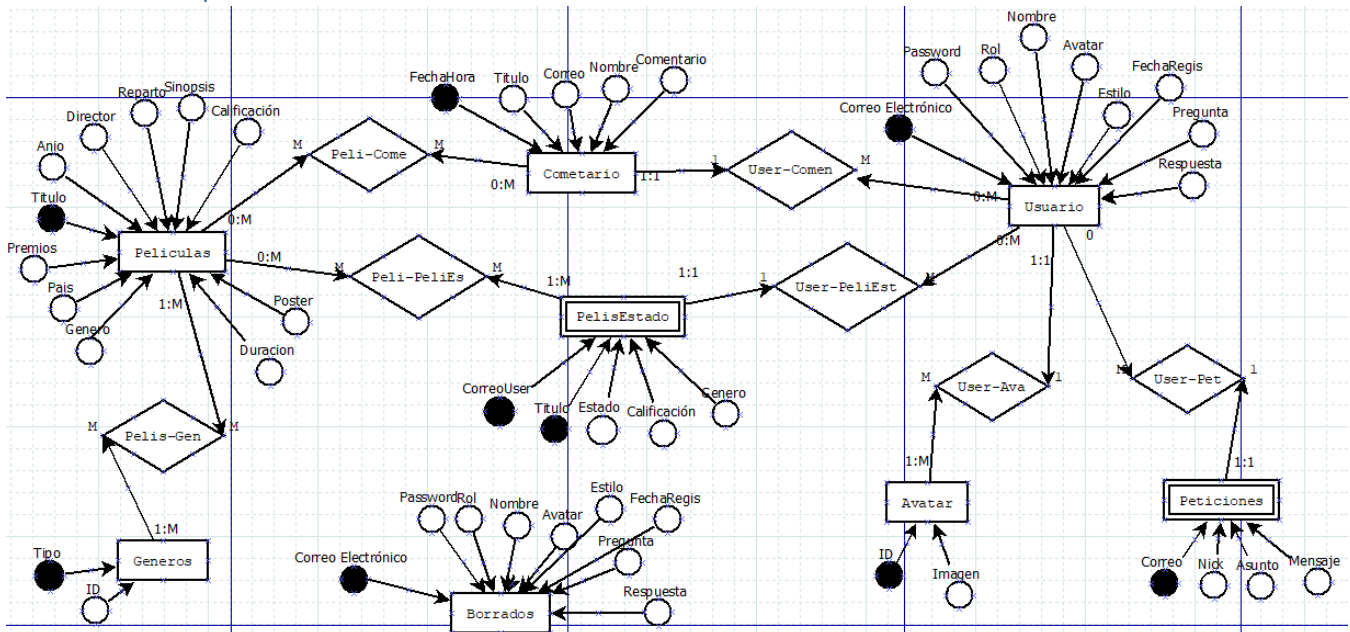
Inicialmente, se consideró la posibilidad de implementar los sistemas HSQLDB, BerkleyDB o Firebird Embedded, pero finalmente se descartaron por las siguientes razones:

**HSQLDB:** Se descartó debido a su curva de aprendizaje más tediosa y molesta en comparación con SQLite. Además, requería configuraciones demasiado específicas, lo que lo hacía menos atractivo para la aplicación, donde se valoraba la simplicidad y la facilidad de integración.

**BerkleyDB:** Fue descartado porque la estructura simple de SQLite se ajusta mejor a aplicaciones de tamaño medio, buscando una integración rápida y sencilla de la información en la base de datos. La preferencia por la simplicidad y la eficiencia hizo que **BerkleyDB** resultara menos idóneo para los objetivos de FílmotecaCLM.

**Firebird Embedded:** No se consideró como una buena opción, ya que su implementación podría introducir complejidad innecesaria en el desarrollo y mantenimiento de la aplicación. La elección de SQLite se basó en la necesidad de una solución más sencilla y autónoma para garantizar un proceso de desarrollo más fluido y eficiente.

### 6.3 Esquema Modelo Entidad-Relacion



(Modelo Entidad-Relación)

A continuación, se presenta el modelo entidad-relación de la base de datos.

En dicha base de datos, se encuentran ocho tablas, siendo la tabla "Usuario" la principal debido a la abundancia de relaciones que alberga. A continuación, se describen cada una de las tablas utilizadas:

#### 6.4 Tablas creadas:

##### 6.4.1 Tabla Usuario:

Gestiona los datos del usuario, incluyendo correo electrónico, nombre, contraseña... Utiliza el correo electrónico como clave primaria y el campo "Avatar" como clave foránea.

##### 6.4.2 Tabla Peticiones:

Representa una tabla de entidad débil encargada de administrar las peticiones realizadas por los usuarios a los administradores. La clave foránea proviene del correo electrónico en la tabla "Usuario".

#### 6.4.3 Tabla Avatar:

Contiene las imágenes utilizadas por los usuarios como avatares. La tabla tiene como clave primaria el campo "ID" y almacena el campo "Imagen".

#### 6.4.4 Tabla PelisEstado:

Entidad débil encargada de gestionar las valoraciones y listas de películas de los usuarios. Utiliza como claves foráneas el correo del usuario y el título de la película.

#### 6.4.5 Tabla Comentario:

Almacena los comentarios que los usuarios han hecho sobre películas específicas.

#### 6.4.6 Tabla Películas:

Guarda los datos de las películas obtenidos mediante la API OMDB. La clave primaria es el título de la película.

#### 6.4.7 Tabla Géneros:

Almacena los ID de los géneros utilizados para buscar películas en la API TMDB. La clave primaria es un número auto incrementable denominado "ID".

#### 6.4.8 Tabla Borrados:

Única tabla en la base de datos sin relaciones con otras tablas. Su función es almacenar información sobre usuarios vetados de la aplicación, transfiriendo datos desde la tabla "Usuario" a la tabla "Borrados".

## 7. Hilos y concurrencia:

La implementación de hilos en la aplicación ha desempeñado un papel esencial al abordar el principal desafío que enfrentaba la sección de información detallada. En esta área, la aplicación buscaba nuevas películas en la API que involucraran a un director en algún papel de desarrollo. El proceso consistía en extraer el nombre del director de la API OMDB, buscar su identificador en la API TMDb, y luego recuperar películas asociadas mediante dicho identificador. La aplicación debía encontrar cuatro películas que también estuvieran presentes en OMDB. Posteriormente, se generaban User Controls que se añadían a un grid específico en la página. Este proceso, que consumía tiempo, se ha optimizado significativamente gracias a la implementación de hilos en la interfaz de usuario WPF, garantizando una respuesta casi instantánea al cargar la página.

Adicionalmente, la integración de hilos se ha extendido al proceso de agregar nuevos usuarios por parte de los roles de administración. En este contexto, no se han presentado problemas de concurrencia al insertar más de un usuario, ya que cada hilo se completa antes de la creación del nuevo usuario. Esta implementación ha mejorado la eficiencia y la capacidad de respuesta del sistema, asegurando una experiencia de usuario más ágil y evitando posibles impaciencias durante la espera.

## 8. Perfiles de Usuario

Para este proyecto, se han definido tres categorías de usuarios distintas, cada una identificada por un rol que se representa como un valor numérico en la base de datos. El rol 0 corresponde al usuario estándar, el rol 1 al usuario administrador, y el rol 2 al usuario superadministrador.

**Rol Usuario (Rol 0):** Este rol constituye la piedra angular de la aplicación, ya que está diseñada pensando en sus necesidades. Desde el momento del inicio de sesión, los usuarios estándar pueden disfrutar de diversas funciones, como el buscador de películas, las recomendaciones automáticas de la aplicación, el comparador de películas, la modificación de datos personales (como el nick, la contraseña y el avatar), la inserción de avatares, la calificación de películas en una escala del 0 al 10, y la clasificación de las películas en listas pendientes, vistas y favoritas. Además, los usuarios tienen la capacidad de dejar comentarios sobre las películas y realizar solicitudes específicas a los usuarios con roles de administración, como la eliminación de cuentas, la gestión de comentarios inapropiados, y la notificación de problemas para su resolución.

**Rol Administrador (Rol 1):** Los usuarios con este rol pueden disfrutar de todas las funciones del usuario estándar, pero además cuentan con acceso a un área especial. Aquí, tienen la capacidad de crear y eliminar usuarios, gestionar comentarios, y atender las solicitudes de los usuarios. Aunque tienen el poder de eliminar usuarios, se les impone una restricción importante: no pueden eliminar al superadministrador, ya que este posee un nivel superior, ni pueden eliminar a otros administradores que se encuentran en el mismo nivel jerárquico.



Rol Super-Administrador (Rol 2): Este rol es único y exclusivo, asignado al bibliotecario jefe. El superadministrador posee todas las funciones de los usuarios anteriores, pero sin las limitaciones impuestas a los administradores. No hay restricciones para eliminar otros administradores ni para acceder a todas las áreas y funciones de la aplicación.

Para garantizar la seguridad y evitar posibles abusos de poder, se ha optado por la implementación de dos tipos de administradores. Esto evita que un único administrador pueda eliminar a otros administradores y adquirir el control total de la aplicación sin restricciones. Los administradores son designados por el superadministrador, lo que permite asignar responsabilidades específicas a diferentes miembros del equipo de la biblioteca, brindando un apoyo eficiente en la administración del sistema.

## 9. Desarrollo:

### 9.1 Metodología:

En lo que respecta al desarrollo del proyecto, he seguido la metodología del modelo en cascada, la cual implica establecer objetivos claros para la aplicación y luego proceder a desarrollarlos. Después de definir los objetivos, dibujaba cada página de la aplicación en papel, detallando el funcionamiento de cada sección para obtener un guion claro. Optaba por abordar un objetivo a la vez, asegurándome de completarlo antes de pasar al siguiente. Sin embargo, conforme el proyecto crecía, surgían ocasiones en las que era necesario regresar a secciones anteriores para realizar pequeños ajustes que se alinearan mejor con las partes subsiguientes.

Al concluir cada objetivo, llevaba a cabo una fase de pruebas específica y, al finalizar el proyecto en su totalidad, se ejecutaba una fase de pruebas para la aplicación completa. ¿Cuáles son los beneficios asociados a esta metodología? Principalmente, proporciona una estructura clara para la aplicación y un sistema de organización que minimiza la posibilidad de caos durante el desarrollo.

Aunque inicialmente logré los resultados deseados con esta metodología, posteriormente surgió la necesidad de agregar más funcionalidades para enriquecer y completar de manera más exhaustiva la aplicación, dando lugar a la creación de nuevos objetivos.

## 9.2 Proceso de desarrollo:

### 9.2.1 Fase 1

En la fase inicial, delineé objetivos simples para el proyecto, que incluían:

#### *9.2.1.2 Registro de Usuario, Cambio de Contraseña y Login:*

Implementé la funcionalidad básica para que los usuarios puedan registrarse, cambiar su contraseña y realizar el inicio de sesión. Se han implementado medidas para prevenir la inserción de texto malicioso y proteger contra ataques de SQL injection, como el conocido ejemplo de ' OR 1 = 1'. Esta tecnología se ha implementado en cualquier instancia en la que los usuarios tengan la capacidad de introducir texto en cualquier campo.

#### *9.2.1.3 Buscador de Películas:*

Desarrollé un sistema de búsqueda de películas que permite a los usuarios acceder fácilmente a la información deseada. Implementé un sistema de búsqueda de películas utilizando la API OMDB (Open Movie Database). Esta API me permitió acceder a información detallada sobre películas, como título, año, director, reparto, género, calificación, sinopsis y póster. A través de solicitudes HTTP con parámetros específicos, logré recuperar datos relevantes para presentarlos a los usuarios, facilitando así el acceso a la información deseada sobre películas específicas. La utilización de la API OMDB agilizó el proceso y mejoró la experiencia del usuario al proporcionar datos cinematográficos precisos y actualizados.

#### *9.2.1.4 Visualización Detallada de la Información de la Película:*

Creé una interfaz que proporciona a los usuarios detalles completos sobre una película seleccionada.

#### *9.2.1.5 Comentarios, Valoración y Listas de Películas:*

Permití a los usuarios asignar comentarios, valorar las películas y organizarlas en listas personalizadas, como "Vistas", "Pendientes" y "Favoritas". Esta acción también puede llevarse a cabo sin acceder a la información detallada de la película, utilizando los botones generados en los controles de usuario de selección de películas.

#### *9.2.1.6 Modificación del Nombre de Usuario y Avatar:*

Implementé la capacidad de los usuarios para modificar su nombre de usuario y avatar, personalizando así su experiencia en la aplicación. Además de poder elegir entre los avatares por defecto el usuario también podía insertar un avatar nuevo mediante una imagen. Este avatar puede seleccionarlo cualquier usuario. En esta sección, se ha incorporado la creación de una nueva ventana mediante la modificación de su tamaño, posición en la pantalla y título. Es importante resaltar la inclusión de UserControls en esta ventana.

#### *9.2.1.7 Visualización de Listas Pendientes, Vistas y Favoritas:*

Desarrollé una función para que los usuarios puedan visualizar fácilmente las películas en sus listas "Pendientes", "Vistas" y "Favoritas".

### 9.2.2 Fase 2:

Posteriormente, decidí incorporar nuevos objetivos:

#### *9.2.2.1 Visualización de Múltiples Películas Seleccionadas:*

Perfeccioné la experiencia del usuario al permitir la visualización de varios detalles de varias películas seleccionadas en una tabla. Se empleó el clic derecho para seleccionar o deseleccionar las películas. Cuando se eligen más de una película, se hacen visibles dos botones que permiten visualizar la información de esas películas o deseleccionar todas las películas seleccionadas.

#### *9.2.2.2 Visualizar Películas por el Mismo Director:*

Integré la funcionalidad para que los usuarios puedan ver películas dirigidas, actuadas... (en general cualquier película en la que haya colaborado de alguna forma) por el mismo director de una película seleccionada. Para lograr esto, incorporé una segunda API, específicamente la API de TMDb (The Movie Database).

La API de TMDb proporciona información adicional sobre películas, incluyendo detalles sobre el equipo de producción, reparto y otros aspectos relacionados. Al realizar solicitudes a esta API, pude obtener datos sobre películas vinculadas al mismo director de la película elegida por el usuario.

Esta integración permitió enriquecer la experiencia del usuario al proporcionar recomendaciones relacionadas con el director de una manera más completa y diversa, utilizando información adicional de la API de TMDb.

#### *9.2.2.3 Recomendación de Películas por Género Favorito:*

Desarrollé un código que ofrece recomendaciones de películas basadas en el género favorito del usuario. En caso de que el usuario no tenga un género preferido, el sistema se orienta hacia los géneros más apreciados por todos los usuarios de la aplicación. En situaciones en las que no se identifica un género común, se sugieren estrenos de los últimos dos años. Si todos los estrenos han sido visualizados, la recomendación se extiende a películas de cualquier año.

Para implementar este objetivo, se integró la API TMDb (The Movie Database). El código desarrollado establece una comunicación efectiva con ambas APIs, permitiendo así enriquecer la aplicación al aprovechar la capacidad de búsqueda por director, género y fecha proporcionada por TMDb, mientras se mantiene la funcionalidad existente de la API OMDB. Esta decisión permitió maximizar la diversidad y calidad de las recomendaciones cinematográficas ofrecidas a los usuarios.

#### *9.2.2.4 Modificación del Objetivo de Cambio de Contraseña:*

Ajusté la funcionalidad de cambio de contraseña, tomando la decisión de implementar una pregunta de seguridad personalizable en lugar de utilizar preguntas predefinidas. Esta modificación otorga a los usuarios la libertad de crear sus propias preguntas de seguridad, adaptándolas a su nivel de complejidad deseado.

Se descartó la opción de enviar correos electrónicos como parte del proceso de cambio de contraseña, ya que esta funcionalidad no cumplía con los objetivos establecidos de manera satisfactoria. La elección de la pregunta de seguridad personalizable brinda una alternativa más flexible y segura para garantizar la autenticación de los usuarios durante el proceso de recuperación de contraseña.

### 9.2.3 Fase Final:

Con todos estos objetivos desarrollados, inicié la documentación del código utilizando el programa Sandcastle. Aunque el código ya estaba comentado en el lenguaje C#, también trabajé en el desarrollo del instalador, realizando varios intentos para asegurar una instalación exitosa en cualquier ordenador.

Cabe destacar que dediqué un considerable período de tiempo perfeccionando el instalador para asegurar su funcionamiento.

### 9.3 Herramientas y tecnologías:

#### 9.3.1 Desarrollo de la aplicación:

Para llevar a cabo el desarrollo de la aplicación, he utilizado Microsoft Visual Studio 2022 como entorno de desarrollo, optando por un proyecto de Aplicación WPF de .NET. Los lenguajes de programación empleados fueron C# y XAML, proporcionando así una combinación efectiva para el desarrollo de aplicaciones Windows Presentation Foundation (WPF).

#### 9.3.2 Documentación del código:

La generación de documentación del código se realizó a través del uso del programa Sandcastle Help Builder Project. Este enfoque facilita la comprensión y mantenimiento del código, proporcionando información detallada sobre su estructura y funcionamiento.

#### 9.3.3 Instalador:

La implementación del instalador se llevó a cabo mediante el proyecto Setup Wizard, con el cual se buscó asegurar una instalación exitosa de la aplicación en cualquier ordenador. Este proceso implicó múltiples intentos y ajustes para garantizar una experiencia de usuario sin contratiempos.

#### 9.3.4 Gestión de la base de datos:

En el ámbito de la gestión de la base de datos, se eligió SQLite como sistema de gestión de bases de datos, respaldado por la herramienta DB Browser for SQLite para la visualización y comprobación de datos en la base de datos.



#### 9.3.5 Herramientas dentro de la aplicación:

La aplicación hizo uso extensivo de elementos como páginas y ventanas WPF para facilitar la navegación del usuario. Además, se implementaron User Controls para mejorar la interfaz de usuario y clases personalizadas como "BD", "Movie", "Comentario" y "Usuario", entre otras, para gestionar eficientemente la información dentro de la aplicación.

#### 9.3.6 Integración de APIs Externas:

Para enriquecer la aplicación, se integraron dos APIs externas, OMDB y TMDB. Estas APIs se gestionaron mediante clases dedicadas que facilitaron la obtención de información detallada sobre películas y su clasificación por diversos criterios.

#### 9.3.7 Base de Datos y rendimiento:

Se incorporó la tabla "Películas" en la base de datos para almacenar datos detallados de películas provenientes de las APIs. Esta elección se hizo con el objetivo de optimizar la velocidad de la aplicación al evitar tiempos de espera prolongados asociados con múltiples solicitudes a las APIs. Al tener datos locales, se mejoró significativamente la eficiencia y la velocidad de la aplicación, proporcionando una experiencia más ágil al usuario.

## 9.4 Uso de Inteligencia Artificial

En el transcurso del proyecto, se ha incorporado de manera significativa la Inteligencia Artificial (IA) como un recurso fundamental para abordar tanto posibles errores en el código como dudas relacionadas con la implementación de nuevas ideas. Esta aplicación de la IA ha impulsado la eficiencia y la calidad del desarrollo en:

### 9.4.1 Generador de Imágenes:

Se empleó inteligencia artificial en la creación del logotipo de la aplicación, visible en la portada. Se buscaba que este logotipo representara la esencia del proyecto y el entorno geográfico donde se desarrolló y, presumiblemente, se implementará la aplicación. El resultado final presenta el reconocido casco antiguo de Cuenca sobre un rollo clásico de película de ocho milímetros.

### 9.4.2 Identificación y Corrección de Errores:

La utilización de Inteligencia Artificial ha sido fundamental para abordar desafíos en el transcurso de la creación de código, desempeñando un papel esencial al ofrecer sugerencias de soluciones. Su participación ha consistido en proporcionar ideas innovadoras que han sido estudiadas y posteriormente implementadas, facilitando así la resolución de problemas en el desarrollo del código.

## 9.5 Problemas Encontrados y Decisiones Propias

Durante el desarrollo de la aplicación, se identificaron ciertos problemas que necesitaban abordarse de manera efectiva para garantizar el buen funcionamiento y la calidad del producto final. Estos desafíos exigieron una atención especial y estrategias específicas para su resolución.

### 9.5.1 Colocación de Avatares de una cuadrícula

Uno de los desafíos encontrados en la aplicación surgió específicamente en la sección de selección de avatares. El objetivo era recuperar todas las imágenes de la tabla Avatar en la base de datos y presentarlas gráficamente en un UserControl organizado de manera similar a una cuadrícula, con 5 columnas y un número variable de filas según la cantidad de imágenes disponibles. Sin embargo, tras la ejecución, se observó que gráficamente no se representaba como una cuadrícula, sino más bien como una estructura escalonada.

La solución más elegante no consistió en colocar directamente las imágenes en el Grid; en cambio, se implementó un StackPanel con orientación horizontal que contenía una serie de StackPanels con orientación vertical. Cada vez que se llenaba un StackPanel hijo con cinco avatares, se agregaba al StackPanel principal y se creaba un nuevo hijo con avatares adicionales. Este enfoque resultó en la creación de una cuadrícula perfectamente organizada, resolviendo de manera efectiva el problema inicial y mejorando la presentación visual de las imágenes de los avatares.

### 9.5.2 Películas con el Mismo Director y Recomendaciones

En ese periodo, la aplicación se limitaba al uso exclusivo de la API OMDB, de la cual se obtenían de manera satisfactoria los datos requeridos. No obstante, surgió un desafío al intentar cumplir con el objetivo de buscar películas del mismo director y recomendaciones, ya que OMDB no permitía búsquedas específicas por director, género o año.

Para superar esta limitación, se llevó a cabo un análisis de nuevas APIs disponibles en internet, y se descubrió TMDB. Aunque esta API permitía buscar por año y género, presentaba la dificultad de no proporcionar directamente las obras de un director en la consulta. La solución a este inconveniente implicó buscar en la ficha técnica del director obtenido, identificar su ID y realizar una consulta adicional utilizando dicho ID para acceder a todas las obras en las que el director hubiera participado en algún papel. Este enfoque se aplicó también a la búsqueda por género, obteniendo todos los IDs de los géneros y almacenándolos en la tabla "Genero" de la base de datos.

Posteriormente, surgió otro problema cuando algunas películas no estaban presentes en OMDB, de donde se obtenía información detallada. La solución adoptada consistió en utilizar ambas APIs simultáneamente sin eliminar ninguna de ellas para mantener el diseño previamente desarrollado. Cuando se obtenía el título de una película, se verificaba en OMDB, y si el resultado era nulo, se realizaba otra búsqueda de un nuevo título en TMDB.

## 10. Desarrollo Empresarial

Este documento se encuentra impreso de manera independiente con el propósito de optimizar la organización y estructura de la información contenida tanto en el propio documento como en este adjunto.

## 11. Mejora y Actualizaciones de FilmotecaCLM:

Ahora que nuestra aplicación está en funcionamiento y se espera que varios usuarios la utilicen, resulta lógico proceder con la actualización de la misma, considerando que hemos lanzado el proyecto de una comunidad de cinéfilos. La estrategia más coherente es continuar aprovechando este recurso. Las futuras actualizaciones podrían centrarse en mejorar la experiencia de los usuarios, incluyendo la posibilidad de agregar un campo de descripción para que puedan presentarse a sí mismos.

Además, la aplicación podría incorporar un buscador de usuarios, facilitando el envío de solicitudes de amistad y permitiendo recomendar películas directamente a una sección de recomendaciones entre amigos. En cuanto a los comentarios, se podría introducir un sistema de hilos para organizar y seguir las discusiones, así como la opción de expresar aprobación o desacuerdo mediante botones de "Me gusta" o "No me gusta". El comentario más votado podría destacarse en la parte superior de la sección.

Otra mejora sugerida sería la implementación de un sistema que permita a los usuarios crear chats o foros para debatir sobre sus películas, directores o actores favoritos. En resumen, cada nueva actualización debería introducir características innovadoras y/o abordar posibles problemas de las versiones anteriores.

En perspectiva futura, se plantea la posibilidad de agregar más administradores a la aplicación, y la respuesta es afirmativa. Se contempla la incorporación de nuevos roles con funciones específicas para ampliar la diversidad de usuarios y enriquecer la experiencia de la comunidad.

Rol Usuario Raso (Rol -1): Este rol está diseñado para usuarios que prefieren no registrarse en la aplicación. Sus funciones se limitan al uso del buscador de películas para acceder a información detallada y la capacidad de colocar comentarios, siempre bajo el nombre "Anónimo".

Rol Usuario Cineasta (Rol 2): Este usuario gozará de dos funciones adicionales: la capacidad de añadir y eliminar nuevas películas. Dado que la aplicación utiliza las API OMDB y TMDb, que abarcan una extensa base de datos con todas las producciones existentes hasta la fecha y proyectos futuros, se incorpora este rol para fomentar la participación de cineastas locales. Este usuario permitirá la inserción de nuevas películas realizadas por residentes de la provincia, como proyectos amateurs presentados en concursos de cineastas. Estas funciones también podrían ser consideradas para el rol de superadministrador y se podría evaluar su inclusión en el rol de administradores, aunque limitándolos a la función de añadir películas. La jerarquía actualizada para la aplicación sería: Superadministrador (R:3) > Administrador (R:2) > Cineasta (R:1) > Usuario (R:0) > Usuario Raso (R: -1).

La implementación de este último rol requeriría una modificación en la base de datos, incrementando en 1 los roles de superadministrador y administrador.

Esta evolución en la estructura de roles permitirá una mayor participación y contribución de la comunidad cinematográfica local, consolidando la aplicación como un espacio dinámico y colaborativo.

## 12. Adaptación de proyecto a plataformas móviles

FilмотecaCLM se ha desarrollado como un proyecto WPF mediante el uso de XAML y C#. Para adaptar esta aplicación a diferentes plataformas móviles, es imperativo tener en cuenta las divergencias fundamentales, tales como:

### 12.1 Rediseño de la Interfaz de Usuario

Dado que las pantallas de dispositivos móviles tienden a ser de tamaño reducido, es necesario reconsiderar por completo el diseño de la interfaz para optimizar la experiencia del usuario en estos dispositivos.

### 12.2 Sustitución de Funcionalidades no Compatibles

Algunas funcionalidades que son comunes en entornos de escritorio pueden no ser directamente aplicables o tener un enfoque diferente en el desarrollo móvil. Es crucial identificar y reemplazar dichas funciones para garantizar la coherencia y utilidad en dispositivos móviles.

### 12.3 Elección de Plataforma Multiplataforma o Nativa

Se debe considerar la elección entre plataformas multiplataforma como Xamarin, que permite el desarrollo para Android e iOS con un código base, o el uso de tecnologías nativas como Kotlin para Android y Swift para iOS, para maximizar la integración con las características específicas de cada plataforma.

### 12.4 Adaptación del Código XAML

El código XAML utilizado en la aplicación deberá ser revisado y ajustado para asegurar su compatibilidad con las especificaciones y requisitos de las aplicaciones móviles.



### 12.5 Optimización del Rendimiento

Debido a las limitaciones de recursos en dispositivos móviles, se debe realizar un estudio exhaustivo para optimizar el rendimiento de la aplicación, asegurándose de que funcione de manera eficiente en términos de memoria y capacidad de procesamiento.

### 12.6 Integración de la Base de Datos en un Servidor

Para cumplir con el objetivo de crear una comunidad en torno a la aplicación, se debe considerar la inserción de la base de datos en un servidor accesible desde dispositivos móviles. Esto permitirá a la comunidad acceder a información vital, como los comentarios de los usuarios y la sección de sugerencias a administradores.

Es relevante destacar que SQLite, siendo compatible con dispositivos móviles, representa una opción popular y ampliamente utilizada en el desarrollo móvil. Su capacidad de integrarse sin problemas y su eficiencia lo convierten en una elección sólida para aplicaciones que buscan adaptarse a múltiples plataformas.

## 13. Conclusiones:

Como conclusión de este trabajo, he adquirido la comprensión fundamental de la importancia de establecer objetivos sólidos antes de iniciar el desarrollo de la aplicación. Una vez redactados, llevar a cabo un esquema con pseudocódigo ha demostrado ser un paso esencial para clarificar el funcionamiento de cada elemento de la aplicación y comprender sus razones fundamentales. Además, he decidido que, en futuros proyectos con bases de datos embebidas, exploraré la opción de utilizar una base de datos de objetos, con el objetivo de optimizar significativamente el código asociado.

La metodología en cascada que he empleado se ajusta a mi preferencia como enfoque de trabajo, aunque reconozco la necesidad de agregar una cronología para determinar el tiempo necesario para alcanzar cada objetivo.

En relación con la hoja de estilos, he constatado que definirla durante la generación de elementos de la página WPF, aunque requiera planificación previa, resulta ser el enfoque más eficiente.

En una reflexión más personal, he reconocido la necesidad de imponer mayor disciplina en mi rutina de trabajo, estableciendo horarios fijos y definición de pequeños objetivos diarios para mantener un enfoque más estructurado y eficiente en la realización de tareas.

## 14. Webgrafía

### 14.1 Página oficial API OMDB (Open Movie DataBase)

<https://www.omdbapi.com/>

### 14.2 Página oficial API TMDB (The Movie Database)

<https://www.themoviedb.org/>

### 14.3 Creación del Instalador:

<https://www.youtube.com/watch?v=c2NmtvENu3s>

### 14.4 Generador de Imágenes mediante IA:

<https://www.bing.com/images/create>

## 15. Anexos:

### 15.1 Manual de usuario:

Este documento se encuentra impreso de manera independiente con el propósito de optimizar la organización y estructura de la información contenida tanto en el propio documento como en este adjunto.

### 15.2 Desarrollo Empresarial:

Este documento se encuentra impreso de manera independiente con el propósito de optimizar la organización y estructura de la información contenida tanto en el propio documento como en este adjunto.

### 15.3 Repositorio del Proyecto en GitHub:

El proyecto FilmotecaCLM está disponible en un repositorio público en GitHub, brindando acceso abierto y transparente a su código fuente, archivos y documentos asociados.

<https://github.com/Oscar-Viana/GradoDamProyectoFinal.git>