

The Movie App

Manual técnico



Índice

- 1. Introducción**
- 2. Análisis del problema**
 - a. Problemática
 - b. Clientes potenciales
 - c. Análisis DAFO
 - d. Monetización y beneficios
- 3. Diseño de la solución**
 - a. Tecnologías elegidas
 - b. Arquitectura
 - c. Diagrama de clases
 - d. Diagrama E/R
 - e. Consideraciones técnicas
- 4. Documentación de la solución**
- 5. Enlaces de interés**

1. Introducción

La aplicación The Movie App pretende facilitar a los usuarios la gestión de películas. Permitirá al usuario llevar control de las películas que ha visto / desea ver, dar de alta películas a mano o buscándolas, añadir comentarios y valoraciones propias a esta, ver sus detalles y exportar sus datos a ficheros externos.

2. Análisis del Problema

2.1 Problemática

La problemática a resolver es la gestión o el seguimiento de películas por parte de un usuario en concreto.

2.2 Clientes potenciales

Cómo clientes potenciales, podrá ser cualquier persona interesada en llevar un seguimiento o tener un control de cualquier película que el usuario desee.

2.3 Análisis DAFO

<i>Debilidades</i>	<i>Fortalezas</i>
<ul style="list-style-type: none">• Posibles faltas de optimización para ciertos usuarios• Posibles fallas de sistemas de seguridad	<ul style="list-style-type: none">• Variedad de opciones que brinda la aplicación.• Sencillez• No es complicada de mantener
<i>Amenazas</i>	<i>Oportunidades</i>
<ul style="list-style-type: none">• Otros desarrolladores que puedan hacer o hayan hecho aplicaciones similares	<ul style="list-style-type: none">• Ampliar funcionalidades• Optimizar la aplicación para• alcanzar un público más grande

2.4 Monetización y beneficios

La aplicación por ahora no es monetizable. Se podrían obtener beneficios a través de posibles patrocinadores, anuncios o por la implementación de funcionalidades de pago.

3. Diseño de la solución

3.1 Tecnologías

El proyecto se basa en **Java**, con la versión 1.7 para su compilación.

La parte **lógica** o backend de la aplicación está escrita en **Java** como mencionado anteriormente, desarrollada en Eclipse. Para la parte de **persistencias** se está usando **Hibernate** en su versión **6.4.1** y como gestor de base de datos se ha usado MySQL server y MySQL Workbench para su desarrollo.

Para la interfaz de usuario se ha usado **JavaFX** en su versión **21**. Junto a **Scene Builder** para desarrollar las pantallas.

3.2 Arquitectura

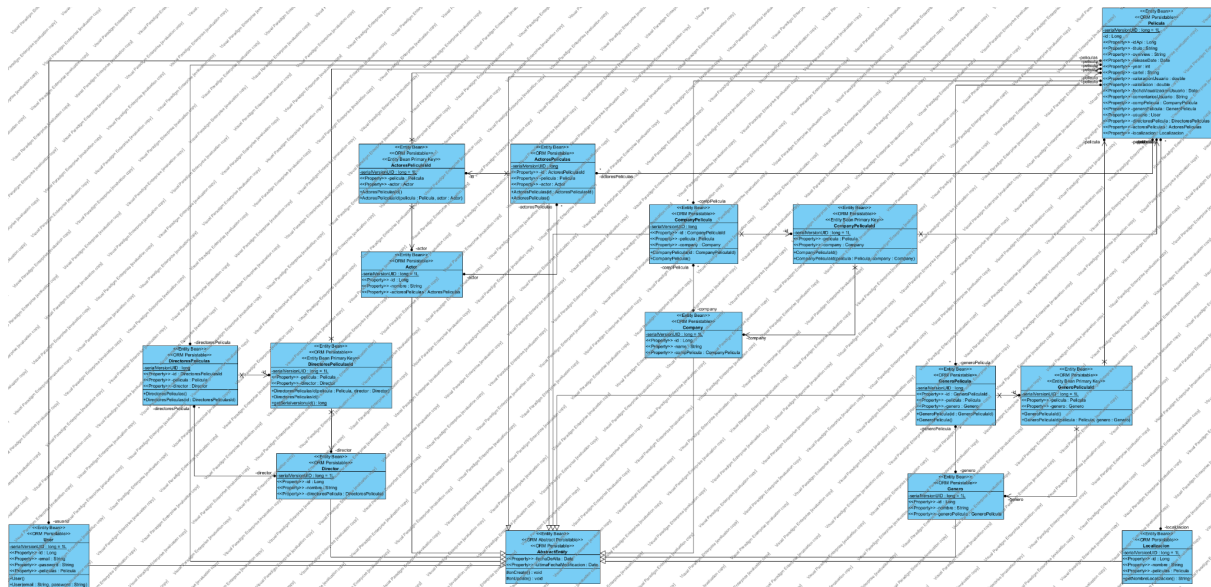
La aplicación sigue el modelo MVC (Modelo Vista Controlador).

- **Modelos:** Entidades usadas con sus respectivos DAOs. Esta capa representa los datos que son almacenados en la base de datos, y los DAOs proporcionan métodos para interactuar con esta base de datos.
- **Vista:** Cualquier pantalla o vista que ofrece JavaFX para ofrecer una interfaz al usuario.
- **Controlador:** La capa de servicios y controladores de estas vistas o pantallas actúan como el controlador de la aplicación.

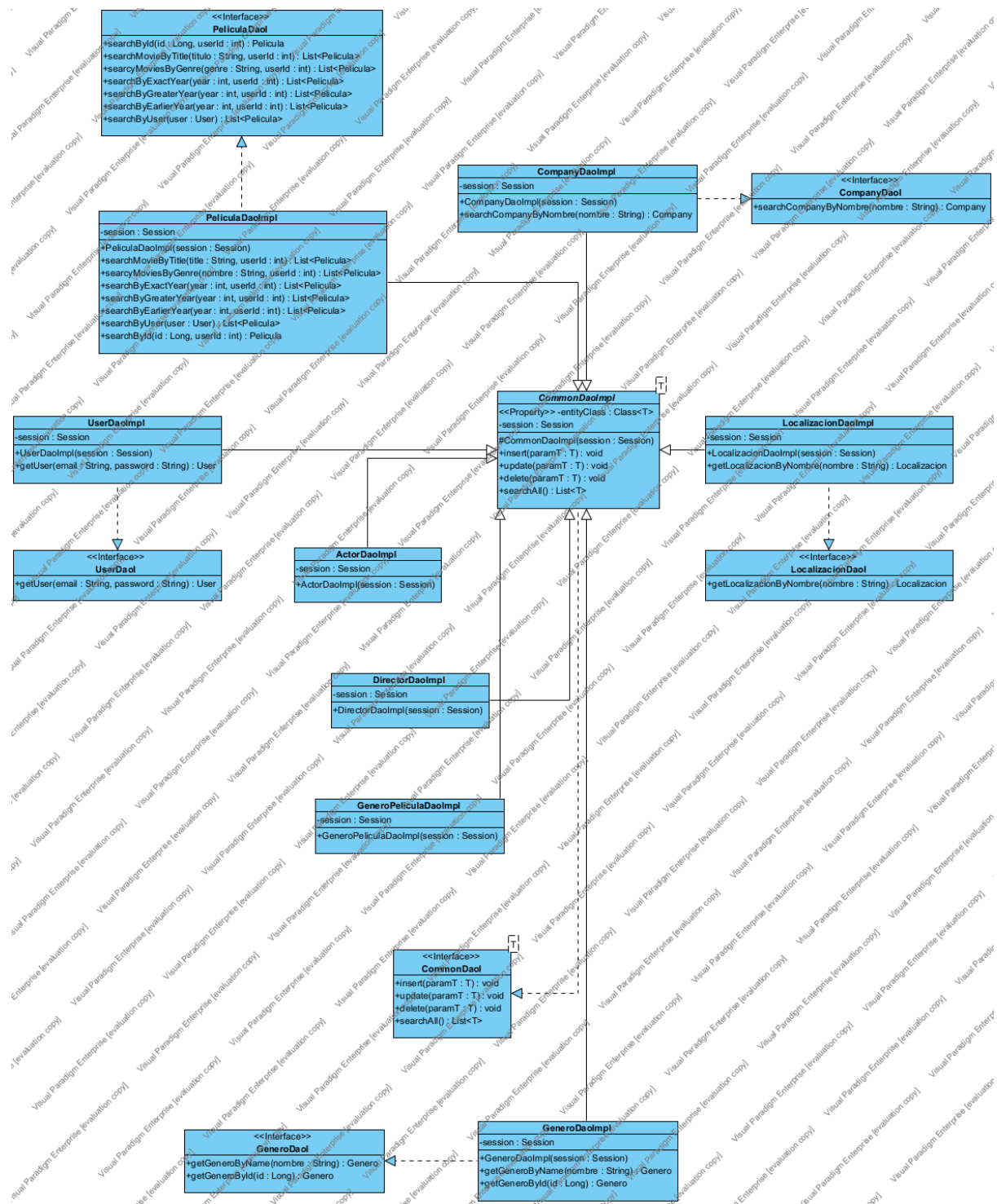
3.3 Diagrama de clases

Debido a la baja calidad de las imágenes en este documento, toda la documentación junto a imágenes con esquemas o diagramas estarán también disponibles en el repositorio de GitHub, en la carpeta **doc**. (Link en apartado 5)

3.3.1 Paquete de persistencia - entidades



3.3.1 Paquete de persistencia - dao



```

classDiagram
    class Kullanıcı {
        +AdSoyad : string
        +TcKimlikNo : string
        +EpostaAdresi : string
        +TelefonNo : string
        +KullaniciTipi : string
        +Durum : string
        +KullaniciNo : int
    }
    class Kitap {
        +KitapAdi : string
        +YazarAdi : string
        +YayinYili : int
        +Durum : string
        +KitapNo : int
    }
    class Kütüphane {
        +Ad : string
        +Durum : string
        +KütüphaneNo : int
    }
    Kullanıcı "1" -- "0" Kitap : borçludur
    Kitap "1" -- "0" Kütüphane : bulunur
    Kütüphane "1" -- "0" Kullanıcı : kullanıcılar
    Kütüphane "1" -- "0" Kitap : kitaplar
    Kütüphane "1" -- "0" Kütüphane : alt kütüphaneler
    
```

The diagram illustrates the structure of a library system. It features three main classes: **Kullanıcı** (User), **Kitap** (Book), and **Kütüphane** (Library). **Kullanıcı** has attributes for personal information (name, ID, email, phone) and library-related details (type, status, user number). **Kitap** includes book-specific attributes (title, author, year, status, book number). **Kütüphane** represents the library's physical structure (name, status, library number). Relationships are defined by associations: a user can borrow multiple books, a book is located in one or more libraries, a library has many users, a library contains many books, and a library can have sub-libraries (indicated by a dashed line).

El diagrama de base de datos para el sistema de películas incluye las siguientes tablas y relaciones:

- usuario**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), email VARCHAR(255), password VARCHAR(255).
- generos**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), nombre VARCHAR(255).
- localizaciones**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), localizacion VARCHAR(255).
- peliculas**: id BIGINT (PK), id_api BIGINT, valoracion_usuario DOUBLE, valoracion DOUBLE, year INT, fecha_de_alta DATETIME(6), fecha_visualizacion_usuario DATETIME(6), localizacion_id BIGINT, release_date DATETIME(6), ultima_fecha_modificacion DATETIME(6), user_id BIGINT, descripcion VARCHAR(10000), cartel VARCHAR(255), comentarios_usuario VARCHAR(255), titulo VARCHAR(255), visualizada TINYINT(1).
- genero_pelicula**: fecha_de_alta DATETIME(6), id_genero BIGINT (FK), id_pelicula BIGINT (FK), ultima_fecha_modificacion DATETIME(6).
- actores**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), nombre VARCHAR(255).
- actores_peliculas**: fecha_de_alta DATETIME(6), id_actor BIGINT (FK), id_pelicula BIGINT (FK), ultima_fecha_modificacion DATETIME(6).
- directores_peliculas**: fecha_de_alta DATETIME(6), id_director BIGINT (FK), id_pelicula BIGINT (FK), ultima_fecha_modificacion DATETIME(6).
- companies**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), name VARCHAR(255).
- company_pelicula**: fecha_de_alta DATETIME(6), id_company BIGINT (FK), id_pelicula BIGINT (FK), ultima_fecha_modificacion DATETIME(6).
- directores**: fecha_de_alta DATETIME(6), id BIGINT (PK), ultima_fecha_modificacion DATETIME(6), nombre VARCHAR(255).

Las relaciones principales son:

- usuario** a **peliculas**: uno a muchos.
- generos** a **genero_pelicula**: uno a muchos.
- localizaciones** a **peliculas**: uno a muchos.
- genero_pelicula** a **peliculas**: muchos a muchos.
- actores** a **actores_peliculas**: uno a muchos.
- actores_peliculas** a **peliculas**: muchos a muchos.
- directores** a **directores_peliculas**: uno a muchos.
- directores_peliculas** a **peliculas**: muchos a muchos.
- companies** a **company_pelicula**: uno a muchos.
- company_pelicula** a **peliculas**: muchos a muchos.

4. Documentación de la solución

Para obtener acceso al código fuente de la aplicación, se debe de acceder al siguiente enlace de GitHub: <https://github.com/danielgalo/The-Movie-App>

Ruta a la carpeta del proyecto: ***Java/di-the-movie-app***

5. Enlaces de interés

Librerías usadas:

- **JUnit** usado para realizar pruebas unitarias:
<https://mvnrepository.com/artifact/org.junit.jupiter/junit-jupiter-api>
- **OpenCSV**: usado para escribir archivos CSV:
<https://mvnrepository.com/artifact/com.opencsv/opencsv>
- **Hibernate** usado para capa de persistencias:
<https://mvnrepository.com/artifact/org.hibernate.orm/hibernate-core>
- **OkHttp** usado para peticiones a la API:
<https://mvnrepository.com/artifact/com.squareup.okhttp3/okhttp>
- **Fastjson2** usado para escribir archivos JSON:
<https://mvnrepository.com/artifact/com.alibaba.fastjson2/fastjson2>
- **JavaFX fxmli / JavaFX controls** para usar JavaFX:
<https://mvnrepository.com/artifact/org.openjfx/javafx-fxml>
<https://mvnrepository.com/artifact/org.openjfx/javafx-controls>

Enlace al repositorio GitHub: <https://github.com/danielgalo/The-Movie-App>