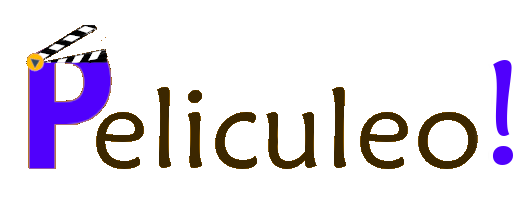
***FICHA TÉCNICA***

PELICULEO

Aplicación multiplataforma de gestión de tickets para espectadores y gestión semi-automatizada de estrenos para administradores.



Miguel Ángel Fuentes Valenzuela – 2ºDAW | IES Las Fuentezuelas

FichaTécnica **– Peliculeo!**

|  |  |
| --- | --- |
| DATOS GENERALES | |
| **Lenguajes de Programación** | JAVA (Spring), Maven, Gradle, SQL (motor: MariaDB) |
| **Entornos de Desarrollo (IDE) utilizados** | IntelliJ IDEA, Apache NetBeans, Notepad++ |
| **Modo de Despliegue** | Servidor local SpringBoot; Apache + mySQL (XAMPP stack) para servidor MySQL |
| **Motor de BBDD** | MariaDB + PHPMyAdmin |
| **Nombre de la Aplicación** | PELICULEO! |

|  |  |
| --- | --- |
| REFERENTE TÉCNICO y FUNCIONAL | |
| **Nombre** | Miguel Ángel Fuentes Valenzuela |
| **E-mail** | [mafvpersonal@gmail.com](mailto:mafvpersonal@gmail.com) – fuentesvalenzuela789@gmail.com |
| **Teléfono de contacto** | 603 58 96 82 |

Esta aplicación tiene acceso a **consultar la información que el Usuario introduce** en la BBDD. Así como a información básica (E-mail y nombre personal en Google) de su cuenta de Google, si se inicia sesión a través de dicha plataforma.

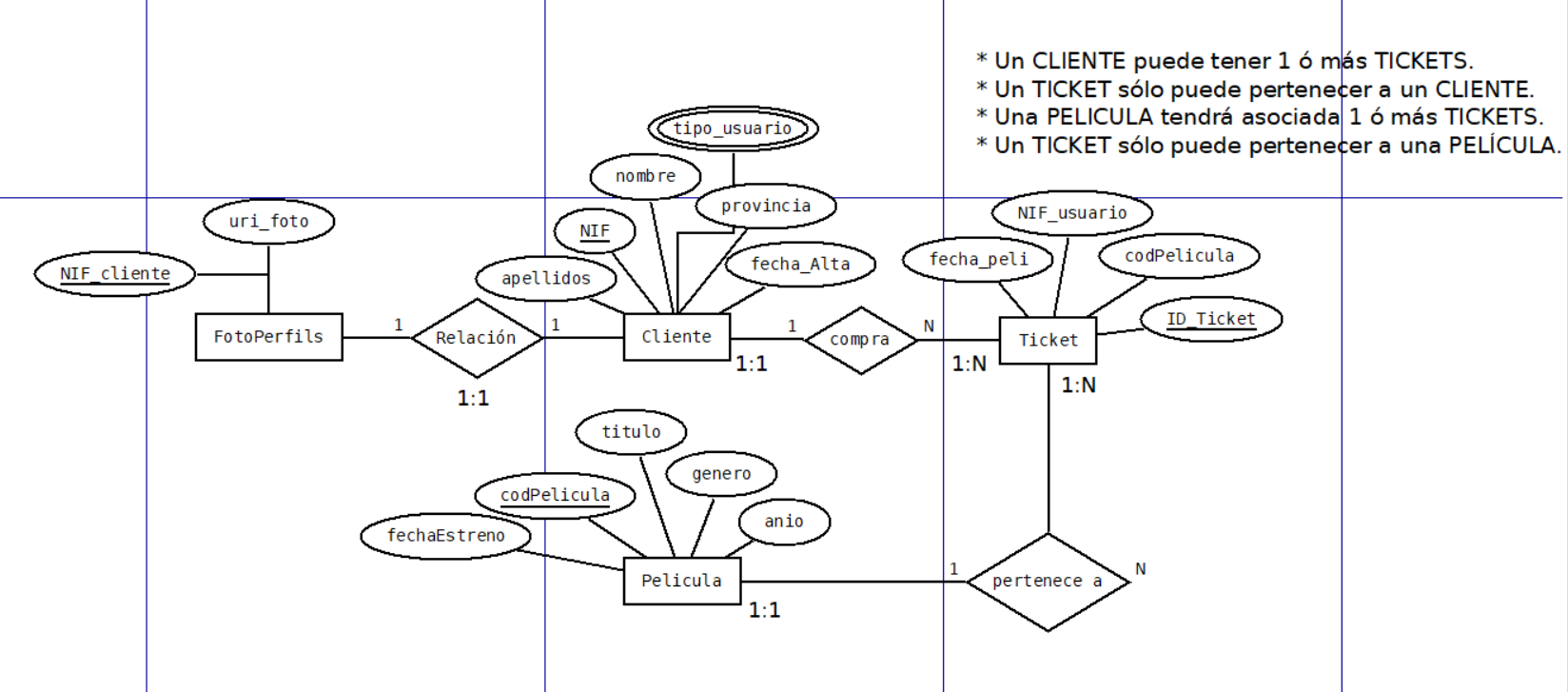
1. Estudio del problema y análisis del sistema

El proyecto consiste en crear una Aplicación Web que permita gestionar tanto películas de una Base de Datos relacional, como ‘tickets’ para los estrenos de dichas películas en un salón de cine.

Los clientes que quieran acceder a ver una película, se deberán registrar en el sistema, a través de la aplicación de móvil “MyPeliculeo”

Una vez se han registrado, tendrán acceso a las películas disponibles en cartelera, y a su vez a los tickets que ya han adquirido, con opción de descartarlos o mantenerlos.

**MODELO E--R – Peliculeo!**

****

1. Módulos del ciclo que intervienen

**Acceso a Datos** 🡪 Diseño de la base de datos y de los DAOs (clases para conectar los diferentes módulos de la app a la BBDD).

**Desarrollo de Interfaces** 🡪 Todo lo relacionado con GUIs, y la codificación interna de estas. Tanto en Android (api de Toast/Android/Glide…) como en Java Swing (api de awt/swing)

**Programación Multimedia y Disp. Móviles** 🡪 Módulo de aplicación (recyclerview con CRUD y autenticación) en Android.

**Programación de Servicios y Procesos (PSP)** 🡪 Uso de la función “sleep” en el hilo principal de las aplicaciones, para mostrar información de manera temporal al usuario (tanto en la aplicación de admón., como en la de Android).

1. Funcionalidades de la aplicación

**MySQL** Scripts de creación para tablas y esquemas

<https://drive.google.com/file/d/1GrpG6fsu0Gc1Om_HBwFySKKfMNiG89R9/view?usp=drive_link>

**MySQL**

Procedure para eliminar películas cuya fecha de estreno ya ha pasado

CREATE PROCEDURE DeleteOldMovies()

BEGIN

DELETE FROM peliculas

WHERE DATE(STR\_TO\_DATE(fechaEstreno, '%d/%m/%y, %H:%i')) < CURDATE();

END;

Trigger que se encarga de darle la fecha actual a las nuevas películas insertadas

CREATE TRIGGER before\_insert\_peliculas

BEFORE INSERT ON peliculas

FOR EACH ROW

BEGIN

IF NEW.fechaEstreno IS NULL THEN

SET NEW.fechaEstreno = DATE\_FORMAT(NOW(), '%Y-%m-%d %H:%i:%s');

END IF;

END //

1. Seguridad de la Aplicación

Para hacer la aplicación más segura, se usa un método de autenticación a nivel de Servidor (API Springboot) que usa **Spring Security** como base para interceptar las operaciones de login y registro.

Esta librería (Spring Security) incluye diferentes medidas que se aplican en momento de ejecución para evitar ataques de inyección SQL/código Java…

A nivel de código SQL, se implementan también métodos de comprobación antes de realizar cualquier operación DML (trigger before INSERT or UPDATE or DELETE), para comprobar que no se esté intentando introducir código malicioso.