

Sistema de Gestión de Cines (Backend)

Este repositorio contiene el código fuente del backend para un sistema de gestión de cines. La API RESTful permite la administración de cines, salas, películas, funciones, usuarios y la generación de reportes clave.

1. Tecnologías Utilizadas

- **Lenguajes:** JavaScript (Node.js)
- **Frameworks/Librerías:**
 - **Express.js:** Framework para la creación de la API RESTful.
 - **MongoDB:** Base de datos NoSQL para el almacenamiento de datos.
 - **bcrypt.js:** Para la encriptación y manejo seguro de contraseñas.
 - **jsonwebtoken (JWT):** Para la autenticación y autorización de usuarios.
 - **dotenv:** Para la gestión de variables de entorno (claves secretas, URLs de bases de datos, etc.).
 - **express-validator:** Para la validación de los datos en las solicitudes.
- **Base de Datos:** MongoDB

2. Manual de Instrucciones para la Ejecución

Para poner en marcha la aplicación, sigue los siguientes pasos:

Requisitos Previos

- Node.js (versión 14 o superior)
- MongoDB (instancia local o en la nube)
- Un gestor de paquetes como npm o yarn

Pasos de Configuración y Ejecución

1. **Clona el repositorio:**
git clone
https://github.com/tu-usuario/nombre-del-repo.git
cd nombre-del-repo
2. **Instala las dependencias:**
npm install

3. Crea el archivo .env:

Copia el contenido del archivo .env.example y renómbralo a .env. Completa las variables de entorno con tus datos de configuración.

URL_MONGODB=mongodb://localhost:27017/nombre-de-tu-db

DB_NAME=nombre-de-tu-db

APP_HOSTNAME=localhost

APP_PORT=3000

JWT_SECRET=tu_clave_secreta_super_segura

4. Inicia el servidor:

```
node app.js
```

El servidor se ejecutará en el puerto especificado en tu archivo .env.

3. Documentación de Endpoints

Todos los endpoints están prefijados por /api/. A continuación, se detallan las rutas, los métodos y la funcionalidad de cada una.

3.1. Usuarios (/api/users)

Método	Ruta	Descripción	Requiere Autenticación
POST	/register	Registra un nuevo usuario con los datos proporcionados.	No
POST	/login	Autentica un usuario y devuelve un token JWT para futuras solicitudes.	No
GET	/find	Obtiene la lista completa de todos	Sí (Rol admin)

		los usuarios.	
GET	/:id	Obtiene la información de un usuario específico por su ID.	Sí
PUT	/update/:id	Actualiza los datos de un usuario.	Sí (Rol admin o el propio usuario)
DELETE	/delete/:id	Elimina un usuario de la base de datos.	Sí (Rol admin)

3.2. Cines (/api/cine)

Método	Ruta	Descripción	Requiere Autenticación
POST	/register	Registra un nuevo cine.	Sí (Rol admin)