## **Movie Database API**

Intro: This .NET Core solution provides APIs to manage a movie database. It includes functionalities to retrieve movies by genre or actor, and to generate a daily report of movie-related statistics.

**Features**:

* Retrieve detailed information about movies, including actors, director, genres, and reviews.
* Retrieve movies by actor.
* Retrieve movies by genre.
* Background worker service to generate and publish a daily report of movie-related statistics.

**API’s:**

1. **GET /movie/{id}**: Retrieves detailed information about a movie by its ID.
2. **GET /movie/actors/{id}**: Retrieves movies associated with a specific actor.
3. **GET /movie/genres/{genre}**: Retrieves movies belonging to a specific genre.

**Validations:**

* **Movie Title**: Required field.
* **Release Date**: Required field
* **Actor Name**: Required field.
* **Director Name**: Required field.
* **Genre Name**: Required field.
* **Review Text**: Required field, maximum length of 500 characters.

**Sample Payload:**

{

"Title": "The Shawshank Redemption",

"ReleaseDate": "1994-10-14",

"DirectorId": 1,

"Actors": [

{ "ActorId": 1 },

{ "ActorId": 2 }

],

"Genres": [

{ "GenreId": 1 },

{ "GenreId": 2 }

],

"Reviews": [

{ "Text": "Amazing movie!" },

{ "Text": "One of the best movies I've ever seen." }

]

}

**How to Run:**

**Clone the Repository**: git clone

**Navigate to Project Directory:** cd moviedatabase

**Restore Dependencies**: dotnet restore

**Update Database Connection String**: Update the connection string in appsettings.json with your database details.

**Run Migrations**: dotnet ef database update (Ensure Entity Framework Core tools are installed)

**Run the Application**:

**Access APIs**: Use a tool like Postman to access the APIs with the provided sample payloads