**Instructions to set up database:**

1. Run sp\_movie.sql script in /SQL Scripts

**API Endpoints Added:**

Image uploading/storing for user: POST /user/:id/photo

Image retrieval for user: GET /movie/:id/photo

Search movie by genre and/or title POST /search/movie

Get list of all genres GET /genre

Delete genre DELETE /genre/:id

**API Endpoints Edited:**

GET /movie – Added calculation of average rating and total reviews

GET /movie/:id - Added calculation of average rating and total reviews

**Changes in Database:**

Type in user table was given a default value of ‘customer’

Profile\_pic\_url in user table was given a default value of ‘default.png

**Database Details**

**Table name: user**

**Description:** Stores details about all users

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| userid | INT() | Primary Key, Unique Value, Auto-Increment |
| username | VarChar(255) | Unique Value |
| email | VarChar(255) | Unique Value |
| contact | VarChar(255) | Unique Value |
| password | VarChar(255) |  |
| type | VarChar(45) |  |
| profile\_pic\_url | VarChar(45) |  |
| created\_at | Timestamp | Default: Current\_timestamp |

**Table name: genre**

**Description:** Stores details about all genres

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| genreid | INT() | Primary Key, Unique Value, Auto-Increment |
| genre | VarChar(255) | Unique Value |
| description | VarChar(1000) |  |

**Table name: movie**

**Description:** Stores details about all movies

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| movieid | INT() | Primary Key, Unique Value, Auto-Increment |
| title | VarChar(255) | Unique Value |
| description | VarChar(1000) |  |
| cast | VarChar(1000) |  |
| time | VarChar(255) |  |
| opening\_date | VarChar(255) |  |

**Table name: gmkey**

**Description:** Acts as a foreign key table between movies and genres

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| fkeyid | INT() | Primary Key, Unique Value, Auto-Increment |
| genreid | INT() | Foreign Key FK\_gmkey\_genre referencing genre table |
| movieid | INT() | Foreign Key FK\_gmkey\_movie referencing movie table |

**Table name: review**

**Description:** Stores data about reviews for each movie

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| movieid | INT() | Primary Key, Unique Value, Auto-Increment |
| userid | INT() | Foreign Key FK\_review\_user referencing user table |
| movieid | INT() | Foreign Key FK\_review\_movie referencing movie table |
| rating | INT() |  |
| review | VarChar(1000) |  |
| created\_at | TIMESTAMP | Defaut value is current\_timestamp |

**Table name: timeslot**

**Description:** Stores data about timeslots for each movie

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| timeslotid | INT() | Primary Key, Unique Value, Auto-Increment |
| movieid | INT() | Foreign Key FK\_timeslot\_movie referencing movie table |
| date | VarChar(255) |  |
| time\_start | VarChar(255) |  |
| time\_end | VarChar(255) |  |

**Table name: photo**

**Description:** Stores path information about photos for each movie. Could have added a photo column into the actual movie table, but did not want to mix advanced with the basic features.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data Type** | **Note** |
| photoid | INT() | Primary Key, Unique Value, Auto-Increment |
| movieid | INT() | Foreign Key FK\_photo\_movie referencing movie table |
| path | VarChar(255) |  |

The photo below shows a relational view of the database:

Diagram

Description automatically generated