About the System:

For this project, we will be developing a minimalistic movie management system that enables registered users to efficiently add movies and provide them with ratings.
The ultimate aim is to establish a standard movie review platform where users can both access and contribute to insightful evaluations of their favorite flicks.

However, the system is currently missing a lot of the required features and hence is not operational.

The application operates primarily through API functions and does not provide a UI. Therefore, you're only required to implement the endpoints outlined in the UseCase section.

Link to GitHub repository:
https://github.com/genistat-ag/movie-database-assessment

System Description & Use Cases:

- Users should be able to log in with both the below combinations
  - username and password
  - email and password
  - No need to implement Forgot Password option.
  - Unauthenticated users have no access to any feature of the platform other than login.

- Users want to view movies
  - Only authenticated users can see their own movies.
  - Only authenticated users can see all movies.
  - A user can see a list of all available movies in the system.
  - A user can select a movie from the list and see the details about this movie.
  - Movie details are already described in the movie model.

- Users want to create and update movies
  - Only authenticated users can create movies.
  - A movie is always linked with the user who created it.
  - A movie can only be updated by its creator.

- All movies created by users are accessible for viewing by any authenticated user.

- Users want to rate movies
  - A user can give a rating between 1 to 5 for any specific movie.
  - Only authenticated users can give ratings.
  - A user can change their rating more than once. But only their own ratings.
  - For each movie, an average rating across all given ratings is calculated and displayed in the movie details.
  - When a movie review is created/updated, the avg_rating field of the specific movie should get updated automatically.
  - The updated_at field should not change when updating the avg_rating field (It should only change when a movie is edited).

- Users will be able to report movie as inappropriate
  - Only authenticated users can report movies.
  - Only SuperAdmins can request a list of all reports
    - The list should contain the state of the reports.
  - Reports are initially in an "unresolved" state. SuperAdmin can either set them to "Mark movie as inappropriate" or "Reject report".
  - State Management should be implemented with the state machine pattern.
  - If a movie is marked as inappropriate
    - Movie will be hidden from all users of the system.
    - Movie will still be visible to the creator and marked as "inappropriate".
  - If a movie is marked as "Reject report"
    - Movie will be still visible to all users of the system as expected.
    - Report will be closed.
  - SuperAdmin can at any point change his decision on a previously reviewed report.

Acceptance Criteria

The acceptance criterias are the tests we will execute on the application to verify its functionality. So for that the following criterias should be fulfilled

- Create two regular users and one super admin
- Create two movies per user
- Apply rating to all movies by all users

- Report one movie to the admin and reject it,
- Report one movie to the admin and approve it

Relationship Diagram