

# IMDB

In this homework we want to implement imdb. Other features can have bonuses up to 20%, but implementing all features is a must.

We have 3 types of User Role:

- Member
- Admin
- Editor

User Management

- ability for users to update their profiles, reset passwords, etc.

Admins:

- Have full access to all data and admin tools
- Can add, edit, and delete all movies, people, users, reviews, etc.
- Handle user management like banning or deleting accounts
- Review and approve/reject edits from editors
- Set configuration options and rules for the site

Editors:

- Can only suggest edits to existing data, not add or delete
- Suggest edits to things like movie details, trivia, biographies
- Add missing data like release dates, full cast/crew for older titles
- Flag inappropriate content for admin review
- Edit suggestions are queued for admin approval before going live
- Cannot access admin tools or make major changes without approval

Members:

- Rate and review movies out of 5 or 10 stars
- Write reviews with text, mark spoilers, rate reviews as helpful/not helpful
- Add movies to customizable personal lists like "Watchlist", "Favorites", "Classics to See" etc.
- Get recommendations for new movies based on past ratings and favorites
- Follow people like actors, directors, writers to get updates when they have new projects
- Follow/friend other members and see their reviews, ratings and lists
- Comment on and discuss movies in forums with other members
- Contribute trivia, goofs, quotes, soundtrack info for movies through editor program
- Report offensive or inappropriate content to admins
- Get personalized recommendations based on taste profile
- See charts of highest rated, most popular, trending movies

## Movies

- Add movies with titles, plot summaries, posters, trailers etc
  - Categorize movies by genre, release year, language
  - Link movies to actors, directors, writers etc.
  - User ratings and reviews
  - Recommendation engine to suggest movies to users
- 
- User -> Stores registered users with attributes like name, email, password, date of birth, etc.
  - Movie -> Stores movie info like title, release date, cast, plot summary, ratings, etc.
  - Person -> Stores data on actors, directors, writers, etc. with bio info.
  - Review -> User reviews and ratings for movies.
  - List -> Custom lists created by users for tracking movies.
  - Genre -> Movie genres like comedy, action, drama, etc.
  - Role -> Links actors/directors to their roles in movies.

## Key Classes:

- UserManagement -> Handles user registration, authentication, permissions for admins/editors/members.
- MovieDB -> Provides APIs to search, retrieve and update Movie entities.
- PersonDB -> APIs to access Person data.
- ReviewController -> Logic to manage Review entities like submitting, editing, moderating.
- ListManager -> Logic to create and manage user Lists.
- RecommendationEngine -> Suggests movies based on user profiles and taste.
- SearchComponent -> component for searching movies, people, lists, etc.
- ProfileComponent -> Renders user profile pages with reviews, lists, friends.

## Advanced Search

- Boolean operators, phrase searching, filters like genre, year, rating range.
- Person search by name, or movies associated with a person.
- Recommend similar movies based on keywords, cast, crew, user ratings.

Good Luck 🍀