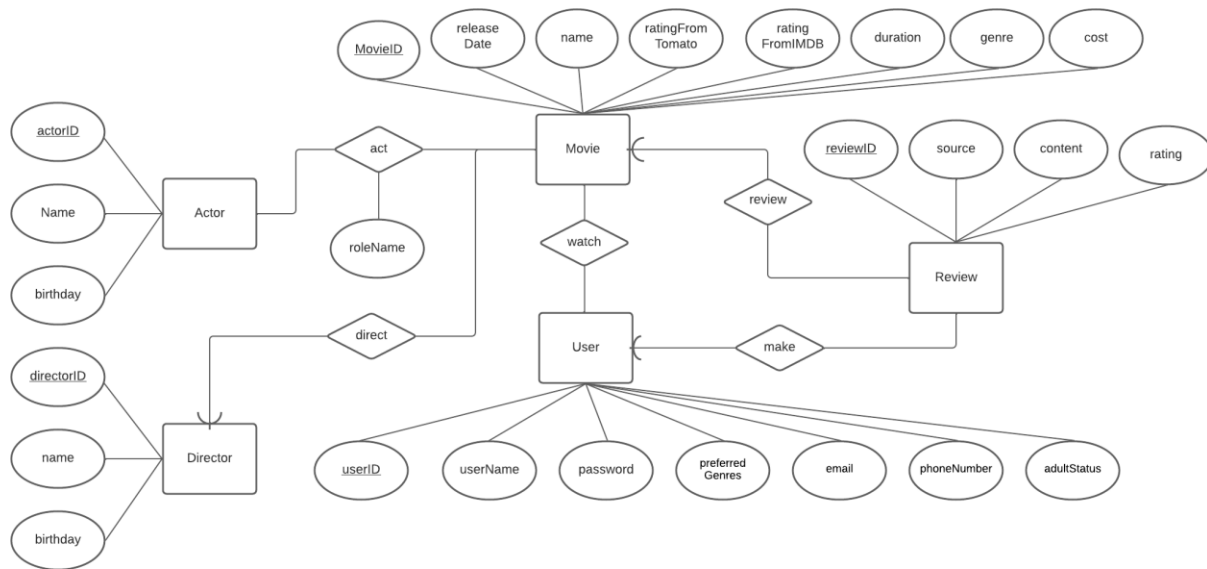


ER_Diagram:



Description:

The database should store information about Directors, Movies, Actors, Users, and Reviews.

A Director is uniquely identified by their ID. Other director attributes are name and birthday.

A Movie is uniquely identified by its movie ID. Other movie attributes are name, rating from tomatoes, rating from IMDB, release date, genre, duration, and cost.

A review is uniquely identified by review ID. Its attributes include content, source of the review, rating.

A user is uniquely identified by user ID. Other attributes are username, password, preferred genres, email, phone number, and adult status.

An actor is uniquely identified by actor ID. Other attributes include name and birthday.

A director may direct many movies, but each Movie has exactly one Director.

A movie may have multiple Actors, and an Actor may act in multiple Movies.

Every time an Actor acts in a Movie, we want to store information about what role that actor plays, specifically the attribute “RoleName”.

A review can be written by exactly one user, but one user can write many reviews.

A review can be written for exactly one movie, but one movie can have many reviews.

A user can watch many movies, and a movie can be watched by many users.

Relational Schma:

Actor (actorID: INT [PK], name: VARCHAR (20), birthday: VARCHAR (15));

Director (directorID: INT [PK], name: VARCHAR (20), birthday: VARCHAR (15));

Movie (movieID: INT [PK], name: VARCHAR (20), duration: INT, cost: INT, genre: VARCHAR (20), ratingFromIMDB: INT, ratingFromTomato: INT, releaseDate: VARCHAR (15), directedBy: INT [FK to Director. directorID]);

Act (actorID: INT [PK, FK to Actor. actorID], movieID: INT [PK, FK to Movie.movieID], roleName: VARCHAR (20));

User (userID: INT [PK], username: VARCHAR (20), Password: VARCHAR (20), preferredGenres: VARCHAR (20), email: VARCHAR (30), phoneNumber: INT, adultStatus: BOOLEAN);

Review (reviewID: INT [PK], source: VARCHAR (20), rating: INT, reviewedMovie: INT [FK to Movie.movieID], content: TEXT, writtenBy: INT [FK to User.userID]);

Watch (movieID: INT [PK, FK to Movie.movieID], userID: INT [PK, FK to User.userID]);