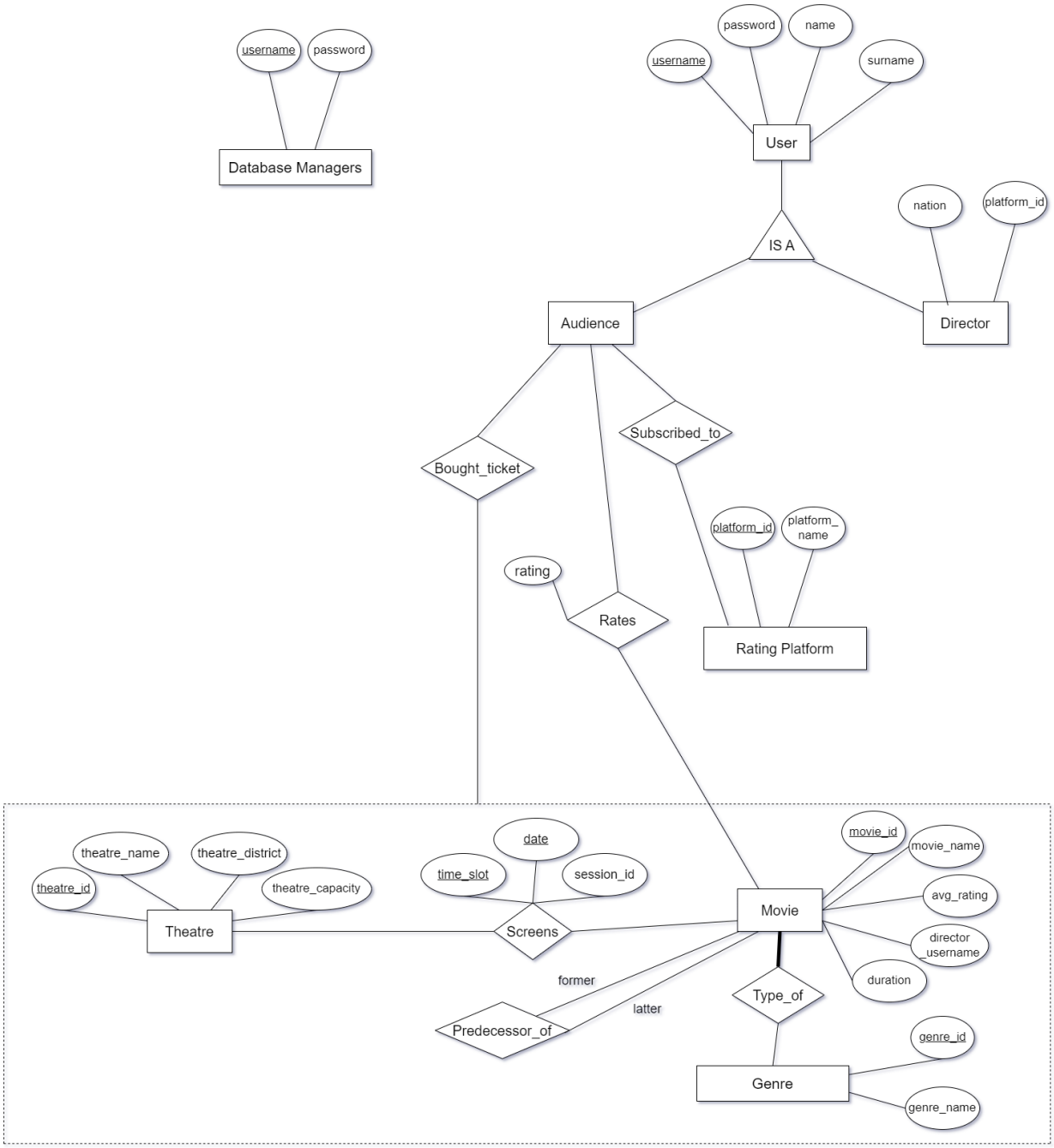
CMPE321 - Project 1: Movie DB

Muhammet Ali Topcu - 2020400147

Project Description: Purpose of the project is designing ER diagram and database schemas for a booking and rating system for movies.

Design & Implementation

Part 1 - ER Diagram:



Notes For ER Diagram Design:

- There is IS-A relationship between User entity, and Audience & Director entities. Audience and Director entities COVER User entity.

- Denoted Movie Sessions as aggregation.

- There is many-to-many relationship between Audience and MovieSessions.

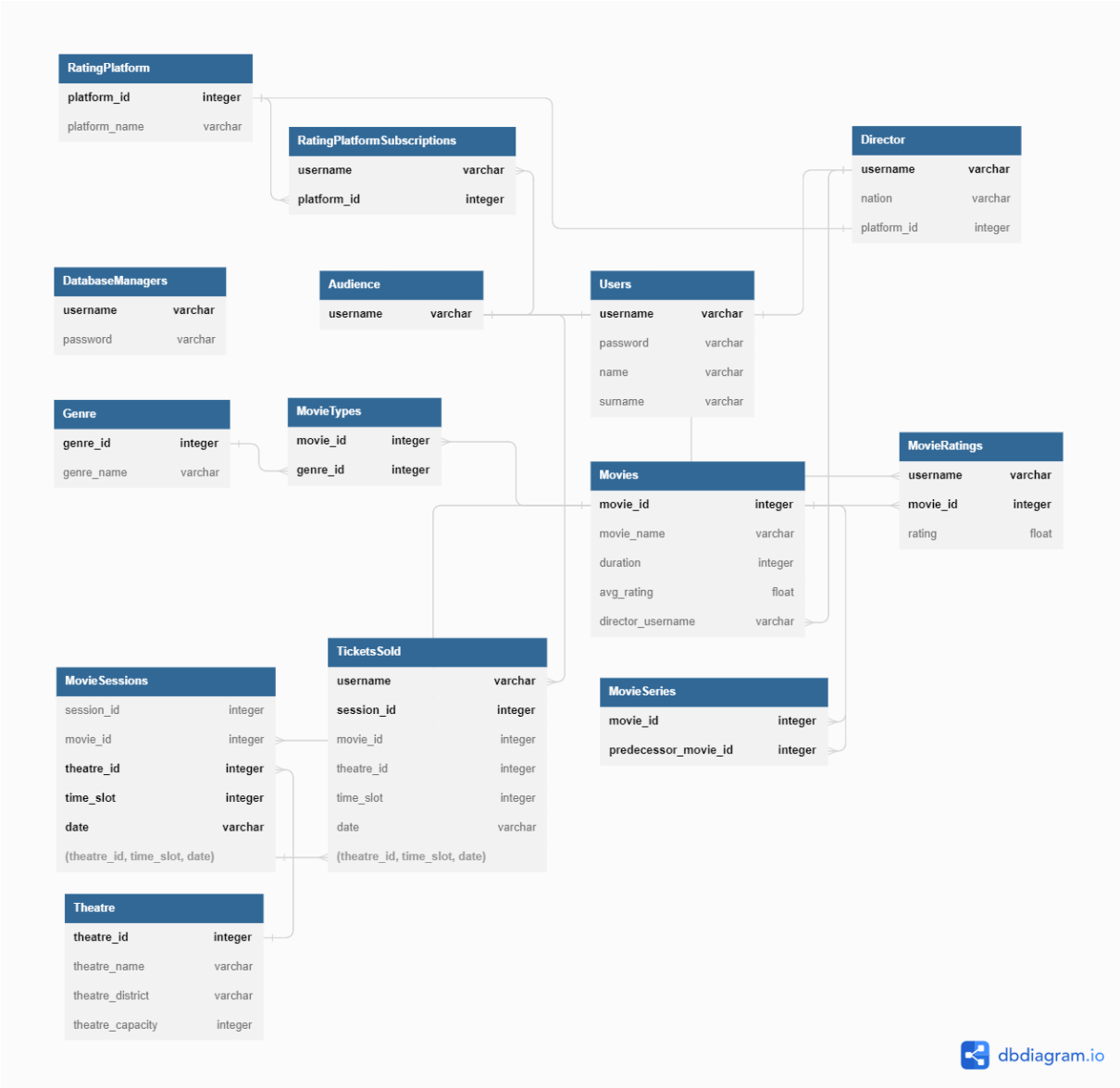
- There is many-to-many relationship between Audience and Movies for rating.

- There is many-to-many relationship between Audience and RatingPlatform.

- Movies have total participation constraint in relation Type_of with Genre since there exists a constraint "Every movie needs to have at least one genre."

- Primary key for MovieRatings is (username, movie_id) so that the constraint "A user can rate the same movie only once." is met.

Part 2 - Logical Database Design:



Schemas:

Users(username: varchar, password: varchar, name: varchar, surname: varchar)

Audience(username: varchar)

RatingPlatform(platform_id: integer, platform_name: varchar)

Director(username: varchar, nation: varchar, platform_id: integer)

DatabaseManagers(username: varchar, password: varchar)

RatingPlatformSubscriptions(username: varchar, platform_id: integer)

Movies(movie_id: integer, movie_name: varchar, duration: integer, avg_rating: float, director_username: varchar)

MovieRatings(username: varchar, movie_id: integer, rationg: float)

Genre(genre_id: integer, genre_name: varchar)

MovieTypes(movie_id: integer, genre_id: integer)

MovieSeries(movie_id: integer, predecessor_movie_id: integer)

Theatre(theatre_id: integer, theatre_name: varchar, theatre_district: varchar, theatre_capacity: integer)

MovieSessions(session_id: integer, movie_id: integer, theatre_id: integer, time_slot: integer, date: varchar)

TicketsSold(username: varchar, session_id: integer, movie_id: integer, theatre_id: integer, time_slot: integer, date: varchar)

Constraints that are not met:

- For these two constraints "No two movie sessions can overlap in terms of theatre and the time it’s screened." and "The duration of the movie is closely related to the time slots. The time slot attribute determines the starting time of the movie and the end time is determined by the duration. (If a movie starts at time slot 2 and has a duration of 2, the theatre is reserved for that movie during the following time slots: [2, 3])", we need to address during insertion of movie sessions in our application layer. For example, before inserting a movie session, we need to first check whether there is an ongoing session at that (theatre_id, time_slot, date) usign the duration of the movie that are screened at that time slot.

- For the constraint "If a movie has any predecessor movies, all predecessor movies need to be watched in order to watch that movie. (See the example below: The Minions need to be watched before Minions: The Rise of Gru).", we need to address during insertion of a new ticket to TicketsSold. We need to check whether the audience watched all predecessor movies.

- For the constraint "A user can rate a movie if they are already subscribed to the platform that the movie can be rated AND if they have bought a ticket to the movie", we need to address during insertion of a rating to MovieRatings. We need to check whether the audience watched the movie and already bought a ticket for that movie.

- For the constraint "There can be at most 4 database managers registered to the system.", we need to check the number of database managers in the system before inserting a new one in application layer.

Part 3 - Write SQL DDL statements: Check createTables.sql and dropTables.sql for the code.