Michael Chau

Week 3

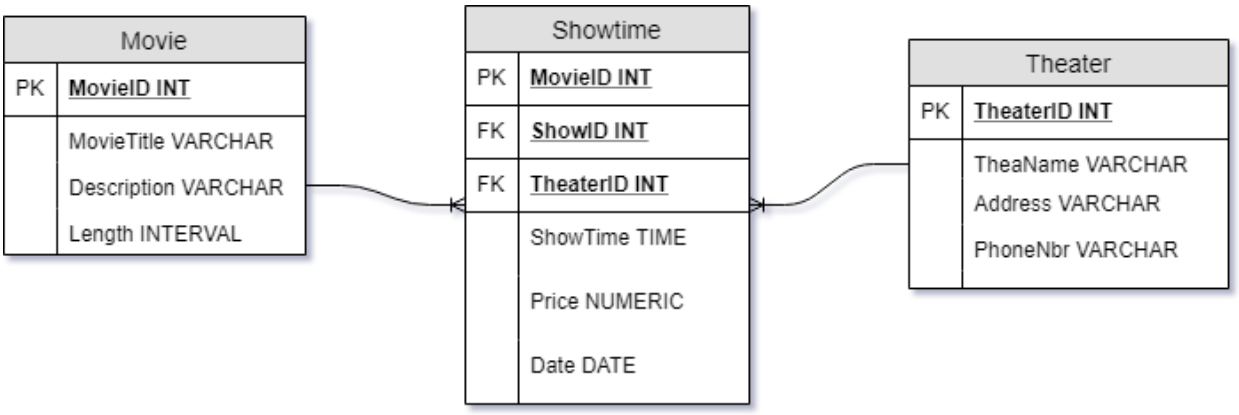
Tixter Database Redesign

Database Technology

For the Tixter web project, I have chosen to keep the use of a relational database—specifically PostgreSQL. Data will be web scraped and stored daily. I chose Postgres because it is open-sourced and widely supported for scrapped data. A document-based database would not be as great for the potential growth and scale of the web application. For the MVP, I may only have a few tables and relationships, but for stretch features, I would like to add more.

There are currently 3 tables: Movies, Showtime, and Theaters. Movies and theaters information will manually be inputted, but I hope to automatically pull that data in the future. Each individual showtime for the next 7 days for select theaters will be stored daily. Each showtime tuple will have a relationship to a movie and theater object. Showtimes will have a many-to-one relationship with movies and a many-to-one relationship with theaters.

Entity-relationship Diagram



For the redesign, I changed the primary keys showID and theaterID of the showtimes table to foreign keys. It didn't make sense to have 3 primary keys. I also took out the rating column of the movies table because it's unnecessary for the MVP.

The Data

Movie Table:

The movie table contains data on an individual movie. The data of each movie includes a title, description, and the movie's time duration. In the list of movies in Tixter, all data will be displayed for the user to see.

Showtime Table:

The showtime table contains data on each movie's showtime at the theaters for the theaters included in the web app. The table holds data for a show's ID, so each showtime will receive an ID. Each tuple will also hold a movie ID in relation to the movie for that showtime. It will also hold the theater ID of the theater the showtime is at. Another information is the time the movie is at, which includes a.m. and p.m. for the pacific time zone. If the price can be found, it will be stored as a numeric type fixed to 2 decimal places. The showtime will also include the day the show is playing.

Theater Table:

The theater table contains data on the theaters in the East Bay. Each theater will have its own row. The data stored will be its name, address, and phone number.

Stretch Feature Design

The overall goal of Tixter is to help users find a movie to watch. I believe that with more information, it can help them. For my stretch features, I would like to gather data on directors and actors that star in the movies listed. This way, a user like Actor Aaron, who is a new fan of Jordan Peele, might just choose to watch *Us* because of Peele. I would create a new table for actors and have a many-to-many relationship with the movies table.

Another piece of data for the movies table that I would include is the parental guidance rating. Günter from Germany might want to watch a family-friendly movie with his kids one day, so having that rating would be very helpful.

The ultimate stretch feature is having Tixter support all theaters and movies across the country. It is very limiting to only support East Bay residents, and just unfair for the rest of the world that can't use Tixter.