

1. Тема проекта: Система бронирования билетов для кинотеатра.
2. Актуальность: сейчас кинотеатры не предоставляют услугу бронирования билетов, только покупку, что зачастую бывает неудобно, ведь не всегда понимаешь, сможешь ли точно прийти на сеанс, и покупать билет не хочется, а мест на вечерние сеансы может уже не остаться.
3. Функциональные требования:
 1. Наличие ролевой модели – есть пользователи и работники.
 2. У пользователя должна быть возможность забронировать билет и посмотреть уже забронированные.
 3. У оператора должна быть возможность добавлять новые фильмы, жанры, актеров, залы и сеансы и менять существующие фильмы.
 4. У оператора должна быть возможность просматривать количество купленных билетов за сегодняшний день.
4. Описание предметной области:

Всего есть 4 сущности: фильм, актер, жанр, сеанс, зал, билет и пользователь.

– Описание фильмов

- Фильмы (таблица movies)
 - id – суррогатный ключ
 - title – название фильма
 - release_year – дата выхода фильма
 - genre_id – связь между фильмом и жанром
- Жанры (таблица genres)
 - id – суррогатный ключ
 - genre_name – название жанра
- Актеры (таблица actors)
 - id – суррогатный ключ
 - first_name – имя актера
 - last_name – фамилия актера
- Связь many-to-many между актерами и фильмами (таблица movies_x_actors)
 - movie_id – ссылка на первичный ключ в таблице Фильмы
 - actor_id – ссылка на первичный ключ в таблице Актеры

– Описание сеансов

- Залы (таблица halls)
 - hall_name – имя зала
 - seats_count – количество мест
- Сеансы (таблица sessions)
 - id – суррогатный ключ
 - movie_id – ссылка на первичный ключ в таблице Фильмы
 - start_time – время начала
 - hall_name – имя зала, ссылка на первичный ключ в таблице Залы
 - price – цена сеанса
- Билеты (таблица tickets)
 - id – суррогатный ключ
 - session_id – ссылка на первичный ключ в таблице Сеансы

- user_name – имя пользователя, забронировавшего билет, ссылка на таблицу Пользователи.

- place – забронированное место.

- state – состояние (enum – забронировано/свободно).

- Дневной отчет (таблица daily_report)

- calc_date – дата подсчета купленных билетов.

- tickets_booked – кол-во забронированных билетов.

– Пользователи (таблица users)

- username – имя пользователя.

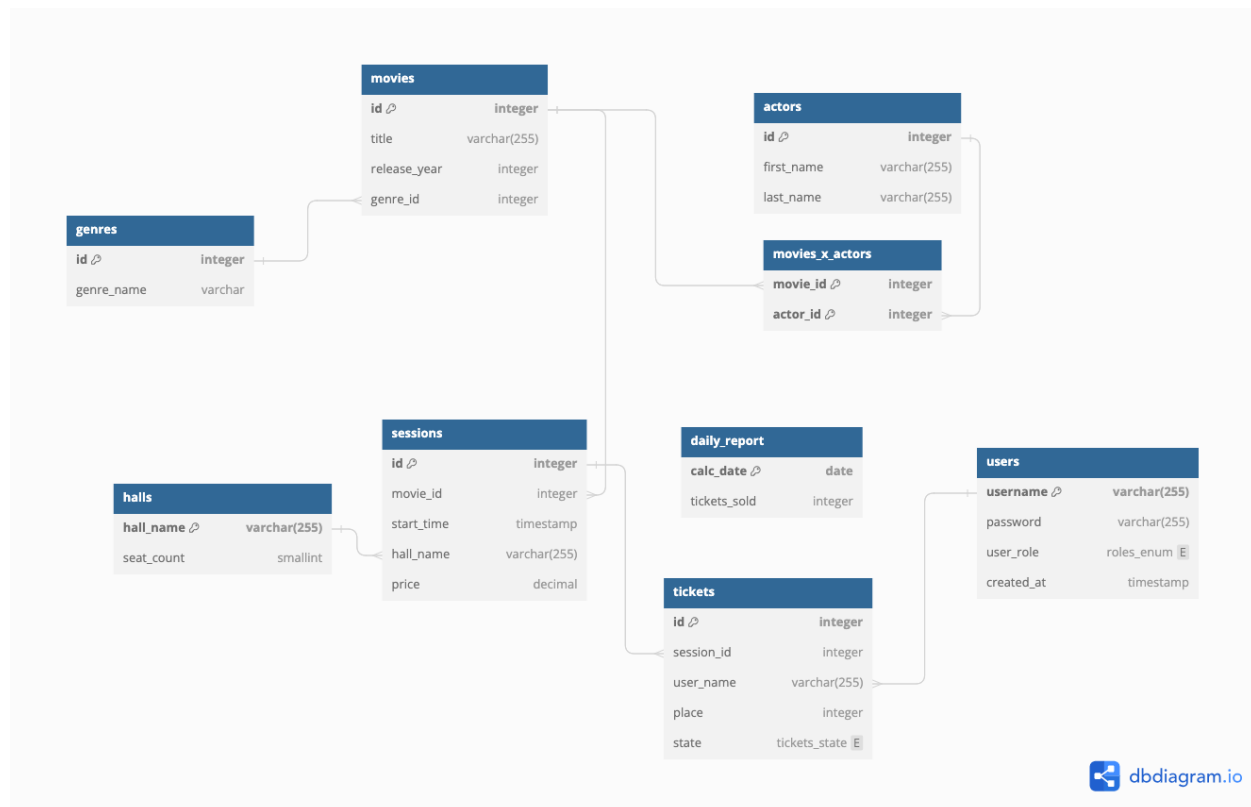
- password – хэш от пароля.

- user_role – роль пользователя (enum cinema_goer/employee).

- created_at – дата создания УЗ.

5. Схема БД:

<https://dbdiagram.io/d/657afd1a56d8064ca00812b6>



6.

1. Все атрибуты в отношениях являются простыми, все используемые домены содержат только скалярные значения. Нет повторений строк в таблицах.
2. Каждый не ключевой атрибут неприводимо зависит от первичного ключа для всех отношений.
3. Каждый не ключевой атрибут нетранзитивно зависит от первичного ключа, а все составные первичные ключи однозначно определены для каждого отношения.

БД находится в нормальной форме Бойса-Кодда.

7. Доработки, которые нужно будет еще сделать:

1. Проверка при создании сеанса, что зал свободен.
2. Выбор мест со стороны посетителя.