**DBMS Mini-Project**

KOTHA ROHIT REDDY

2020A7PS1890H

*Streaming Platform Password Sharing Web Application*

**Project Title**: OTT Share App

**Github link:** https://github.com/rohitkotha10/StreamShare-DBMS.git

**Description**: This project is a web-based application. The technologies used will be MySQL connected with nodejs server with expressjs in the backend. Frontend will be developed with HTML, CSS or a framework such as reactjs.

OTT Share is an app built for users for sharing their streaming credentials with friends and family. Sharing Rooms will be created by the admin and the Room will contain the Streaming Credentials Shared only with authorized users of the group. Members in the group may either be paying members or nonpaying members. Each Room will have a unique RoomID, an admin, and the platform for sharing (ex: Netflix, Hotstar, etc.). Note that each room will be associated with only one account. The plan amount will be split equally among all the paying members of the Room. The Room also contains other basic details such as plan duration, payment due date, number of members, number of screens at a time and so on. The dashboard will be mostly similar for both user and admin with a few added functionalities for admin to manage the users.

Many platforms have an option of having multiple profiles, but it is only limited. This app has no limit for number of members and hence more ways to split the money or more members to share the OTT experience.

**ER Diagram**

Diagram

Description automatically generated

**Relational Schemas:**

The Tables created are:

(Note: User Type means whether a paying or non paying member)

1. User(user\_email, user\_password, age, user\_name)
2. Platform(platform\_name, plan\_type, plan\_price, screens)
3. Room(room\_name, admin\_email, capacity, member\_cnt, platform\_name, plan\_type, stream\_account, stream\_password)  
   admin\_email, platform\_name, and plan\_type are foreign keys
4. Requests(user\_email, room\_name, message, user\_type)   
   user\_email and room\_name are foreign keys
5. User\_Rooms(user\_email, room\_name)   
   both foreign keys

**Technologies used:**

The Backend uses express and mysql, frontend was done using react and material ui. The SQL Queries were executed by connecting the backend nodejs express with mysql. The post and get requests were created accordingly to perform CRUD operations from frontend.

Build Instructions can be found in the read me file which can be found in the project directory. You will need to install nodejs, mysql and create the tables in mysql before starting the webapp

**Routing and Page Structure:**

Each page is associated with a separate link. These links are created with the help of react-router-dom v6 and the routes and the components in each of the routes are mentioned in routes.js.

**Login and Registration Pages:**

**A screenshot of a computer

Description automatically generated with medium confidence**

A screenshot of a computer

Description automatically generated with medium confidence

**Dashboard and its components:**

**Graphical user interface, application

Description automatically generated**