Marcus Leong
Megana Chinalachaiagari
EECS 647
Project 1 Report

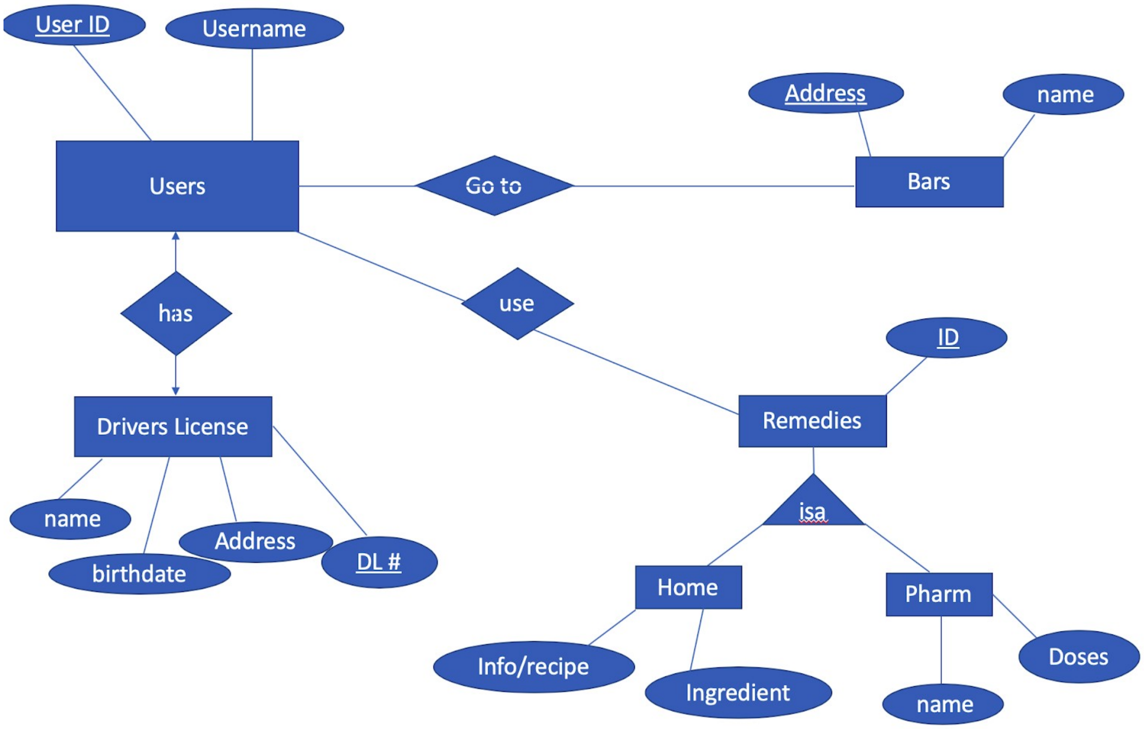
1. Introduction: what is your scenario (describe your mini-world)?

Our scenario focuses on a Bar Crawl type application. We will have the user create an account in which they will create a username, and thus be given a user ID as well (primary key). Each User will also scan into the system, a unique Driver's License. The Driver's license verifies the user's name as well as their address and birthdate. The primary key to be used will be a Driver License number. These multiple users will be able to go to multiple bars listed within the application. Each bar will be identified by its address, but also will store a name for the bar. Additionally, each user may need information on how to treat hangovers. As a result, they can submit types of remedies or view these remedies posted by other users. Remedies are broken up as either home remedies or Over the Counter "pharmacy necessary" remedies. For home remedies, information stored will include the ingredients needed and the recipe/steps to make the remedy. For OTC/Pharmacy remedies, the name of the drug and its doses will be made available.

2. Requirements analysis (be brief): data and operations on data.

Bar Crawl (people go to bars, get drinks, have hangovers, hangovers have remedies, people use remedies, populate popular bars and remedies)

3. Conceptual design: ER-diagram and constrains



4. Logical design: convert your ER-diagram to relational schemas. Normalize your relations, if needed.

User(uuid: int,username: string)
DriversLicense(number: string, name: string, address: string, birthdate: date)
Bars(address: string, name: string)
HomeRemedies(id: int, info: string, ingredients: string) PharmaceuticalRemedies(id: int, doses: string, name: string)
hasADriversLicense(DLnumber, username)
goesToBar(uuid, barAddress) usesRemedies(uuid, id)

5. Implementation:

a Backend

b Frontend

6. Sample Records/Tables

a Include description on what this table or record represents

Our project contains 4 different tables (total of 3 were required). The table names include the User table – which houses information regarding the user's userID number (the primary key) as well as their username. The next table would be the Driver's License table. The primary key of this would be the Driver's License number. We would use this table to generate queries to check if the Driver's License provided is valid for bars (users above the age of 21). The next table is the Remedies table. We plan to use queries like "do we have more pharmaceuticals or home remedies?". Finally, we have a bar table which can be used to have queries that populate the names of bars or the most popular bars etc. Provided in this section are sample queries that I have described above. We will write some of these queries out and provide a sample table or record that is produced from the queries written in section 7.

7. Sample Queries

a Queries and their Productions

In this section, we will focus on the queries mentioned in section 6. After each query will be a short description of what is expected and what is actually projected from the tables query search.

1) Query:

```
SELECT
    Name
    COUNT(NAME)
FROM
    BARS
GROUP BY NAME
HAVING COUNT (NAME) > 1
```

a. Description: generate a list of popular bars based upon the number of times varying users have mentioned this bar.

b. Sample:

2) Query:

```
SELECT RECIPE
   COUNT(RECIPE)
FROM
   REMEDIES, PHARM
GROUP BY RECIPE
HAVING COUNT(RECIPE) > 1

UNION

SELECT NAME
     COUNT(NAME)
FROM
     REMEDIES, HOME
GROUP BY NAME
HAVING COUNT(NAME) > 1
```

a. Description: This query is similar to the first in that we are generating a list of popular remedies, but projecting if they are home or pharmaceutical remedies.
b. Sample:

3) Query:

```
SELECT USERID FROM USER
```

a. Description: this will project the userIDs of all the users present in the database.
b. Sample:

4) Query (uses a join between User and DriversLicense tables):

```
SELECT USERNAME
 FROM USER, DRIVERSLICENSE
WHERE AGE < 21
```

a. Description: display the application users that are below 21 years old, and therefore do not have valid DLs for the bars
b. Sample:

5) Query (with a join between pharm and home remedy type tables)

```
SELECT *
```

```
FROM
    HOME, PHARM
```

a. Description: List all home and pharm remedies by joining the two remedy tables

b. Sample:

8. Future Changes

9. Conclusions and Final Remarks