**DBMS Project Synopsis**

**Semester 5**

**Name:** Anup Nair

**MIS:** 111811052

**Class:** TY Comp Division 2 T4

Taxi management services

**DBMS Project Synopsis**

**Problem Statement:** Location based database for Taxi services (LocDB).

**Introduction:**

With the increasing demand of online services, the taxi booking services have emerged to be one of the primary sectors where databases are employed heavily. Although there is a lot of optimizations and tools being deployed for the major real-life use cases, a simple and basic version of the same design pattern is aimed at by this project. The major focus of LocDB is to simulate a smaller case scenario of a location based reactive system. The basic requirement of the entire project will rely heavily on the database design and the way the data is represented for the needed services. LocDB will provide a similar interface wherein a customer can book taxis that are in a proximity to their current location and create trips to new locations. The underlying gist of the project is to handle updates in real time and provide services by efficient retrieval of data from the database. The other enhancements can include seeing the trip history and nearby places that have been visited and so on.

**Functional requirements:**

1. The users should be able to see the nearby taxis that are available.
2. The users should be able to request for a new service from their current location and request for a taxi to a location that they specify.
3. The taxis need to be listed in order which will be used to query the nearest taxi to the user.
4. Taxis will be categorized into classes and fares will be calculated accordingly by drivers.
5. Users will be able to provide ratings for a driver that has completed a trip and this rating gets accumulated for the driver.
6. The location database will fetch data from zipcodes in the form of tiny cells that are stored locally by the application and these cells represent a proximity of location.
7. Users can view their previous trip history.
8. Taxis database to be updated in real-time based on the location of the taxi and the status of the current trip.
9. The application will store all the trips that are identified by taxis and customers.

**Relation Schema:**

**Diagram

Description automatically generated**

1. Users( user\_id, name, address, phone\_no)
2. Drivers(driver\_id, d\_name, d\_phone\_no, taxi\_id, rating)
3. Taxis(taxi\_id, driver\_id, model, color, number, class, seating\_capacity)
4. Locations(zipcode, l\_name)
5. Trips(trip\_id, driver\_id, taxi\_id, user\_id, from\_zip, to\_zip, status, fare, rating, start\_time, end\_time, duration)
6. Shifts(shift\_id, start, end)
7. Garage(garage\_id, status)
8. Books(trip\_id, user\_id)
9. Works(driver\_id, shift\_id)
10. Drives(driver\_id, taxi\_id)
11. Availability(taxi\_id, zipcode)
12. Present\_At(driver\_id, zipcode)
13. Used\_For(taxi\_id, trip\_id)
14. Contains(taxi\_id, garage\_id)

**ER Diagram**

**Diagram

Description automatically generated**