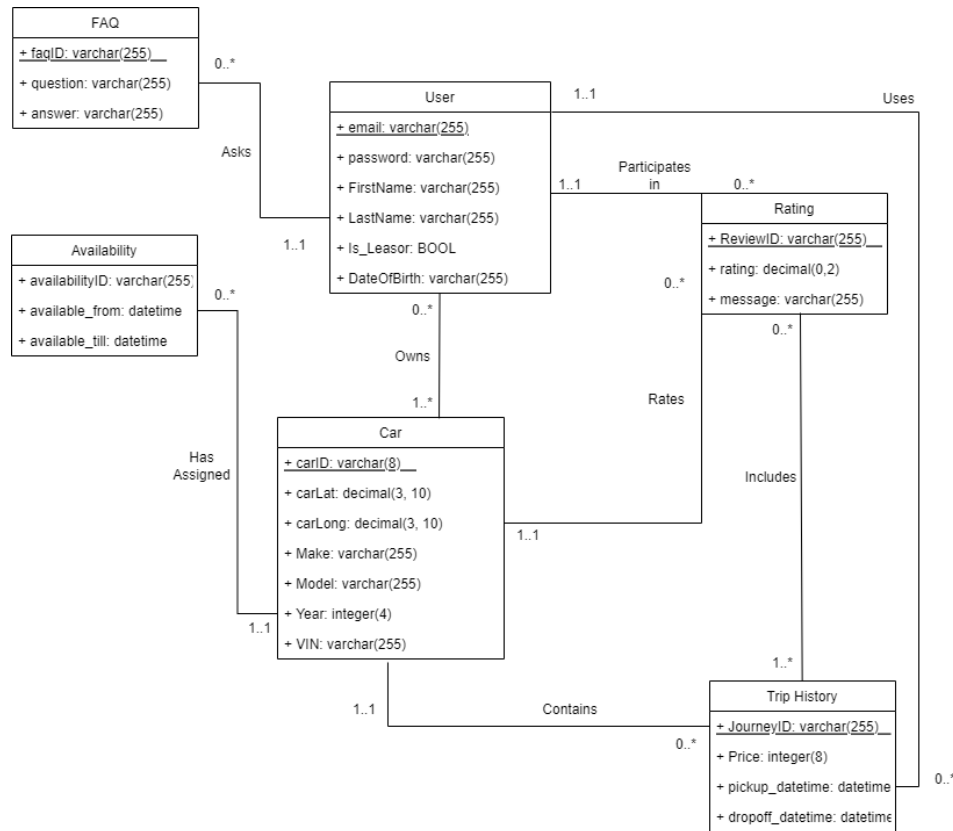


Project Track 1 Stage 2

Conceptual and Logical Database Design Documentation



Description of each entity

- **User**
 - This entity stores user login information such as email, password, date-of-birth, and other details. This table will execute SQL procedures such as CREATE for creating new rows storing user information, and the table will be updated with user information whenever a new user creates an account. To lease or put up a car for lease, the user will have to create an account.
- **Car**
 - This entity contains all the necessary details of the car such as the unique car ID, car location, owner, and other secondary car details such as make, model, year, and rating
- **Availability**
 - This entity is the core entity as it contains information about when each car will be available for the lessee to book.
- **Trip History**
 - This entity stores the history of the car, for example, the timeframes in which the car was used, the renter(s) that used the car, and the specific journey ID of that car.
- **Rating**

- Entity table that stores ratings (rating and message) for the car.
- FAQ
 - This entity consists of the list of all the questions asked by the user of the application (both the lessee and the lesser) uniquely identified by corresponding id. It also tracks if the question is pertinent to a particular journey.

Description of each relationship

- Has Assigned
 - This relationship specifies the month and dates when a particular car is available for a user to book and is assigned to CarDetails.
 - A car can have many available times assigned, but car availabilities are only assigned to a single car in a 1-to-1 relationship.
- Owns
 - Relationship if the user is referenced as an owner of a car. This is more from the car to user side where every car will have an owner / user, but not every user will have a car.
- Participates in
 - Users may participate in creating a rating/review of their renting experience for a car.
 - A user can do many ratings or none if they so choose; and a rating can have 1 or many users participating in it.
- Rates
 - Every car has multiple ratings given by all the past customers and an average score is displayed in the carDetails table.
- Includes
 - The lesser can include a rating for each of his journey/trip in the application once his/her trip is completed.
- Asks
 - Users can post questions on the FAQ page about the service of the application, any questions that they have and about feedbacks.
 - Each User can post any number of questions whereas each FAQ posted (uniquely identified by its Faqid is specific to that user who posted it).
- Contains
 - This relationship specifies the trip history of a particular car. A car can have many trips or no trips associated with it, but a trip is connected to a single car.
- Uses
 - An user can travel in many different cars throughout their time with the CarShare application and each trip is registered on the trip history table with all the necessary details of the trip associated with it.

Relational Schema

User (email: VARCHAR(255) [PK], password: VARCHAR(255), FirstName: VARCHAR(255), LastName: VARCHAR (255), Is_Leasor: BOOL, DateOfBirth: VARCHAR (255))

Car (carID: VARCHAR (8) [PK], ownerID: VARCHAR (255) [FK to User.email], carLat:DECIMAL(3, 10), carLong: DECIMAL(3,10), Make: VARCHAR (255), Model VARCHAR (255), Year:INT(4), VIN: VARCHAR (255))

FAQ (faqID:VARCHAR(255) [PK], userID:VARCHAR(255) [FK to User.email], question:VARCHAR (255), answer:VARCHAR (255))

Rating (ReviewID:VARCHAR(255) [PK], userID: VARCHAR(255) [FK to User.email], carID: VARCHAR(255) [FK to Car.carID], rating:DECIMAL(0,2), message:VARCHAR(255))

Availability (AvailibiltyID:VARCHAR(255) [PK], carID:VARCHAR(255) [FK to Car.carID], available_from:DATETIME, available_till:DATETIME)

TripHistory (JourneyId:VARCHAR(255) [PK], carID: VARCHAR(255) [FK to Car.carID], Rating: DECIMAL(0, 2) [FK to Rating.rating], price:INTEGER(8), pickup_datetime: DATETIME, dropoff_datetime: DATETIME)