Database Fundamentals

Ride Sharing Application

Author:

Nuri Melih Sensoy
Winter 19/20

I. Description of the reality which will be supported by designed database and the aim of the database implementation

1. Choose the area of the reality you want to support using your database. Then describe in details all processes that are performed in this area (take into consideration their logical order). Pay the special attention at processes that generate data.

My project is about transportation. I chose to design Ride Sharing Application. Nowadays it's a popular and very promising alternative to local and intercity transportation. In ride sharing, the car is owned by one person, and they offer rides to others. Any passenger pay towards the ride cost, including fuel, road tolls, etc. My system is gonna used by peoples who wants to go somewhere and who wants to go to somewhere by its own car with lower costs.

Features:

- The mobile application should allow the users to register themselves.
- Members should be allowed to set their preferences regarding rides and travelers.
- Ride owners can create a ride by inputting their trip details(start and destination points, start time, cost)
- Travelers can search for available rides to a destination city.
- Members who are looking for a ride can contact the ride owner and put in a reservation for their seats.
- The ride owner should be able to approve or reject reservation requests.
- Available seats should be automatically updated.
- The ride owner can also add stops, which are cities on the way from their starting point to their destination.

Member

- Everyone has to register before using it.
- There are 2 types of user.
  - Ride Owner: Driver
  - Traveler : Passenger
- In the registration process every user need to provide details like;
  - First Name
  - Last Name
  - Gender
  - Email
  - Contact Number
- In the registration process Ride Owner's need to fill some additional details like
  - Driving Licence Number
  - Driving Licence Valid From : This information tells about drivers experience to the Traveler. This will help Traveler to select best ride available.
- User type will be determined by this additional details.

Car

- The ride owner needs to add details for at least one car before creating a ride.
  - Car Name
  - Car Brand
  - Car Model
  - Year
- One member can own more than one car.
- A ride will be depend on member-car pair.
- There will be a comfort level of a car on a scale from 0 to 5. This level is automatically calculated by the system based on other members feedbacks
  - Comfort Level

Member Preferences

- Preferences that matter to everyone.
- Members doesn’t have fill these the time of registiration, but must be filled before creating a ride.
- This is gonna help people for traveling with similar preferences so that everyone travels comfortably. For the Ride Owner and vice versa.
  - Is smoking allowed inside the car?
  - Are pets allowed ?
  - How talkative is the ride owner? (none, light, chatty)
  - What type of music does ride owner likes?

City

- Store a list of all of the cities available.
  - City
  - Country

Ride

- A member can create a ride by filling details.
  - Car
  - Starting City
  - Destination City
  - Date and Time of the Journey
  - The Number of Seats Available
- Each traveler has to pay towards ride expenses
  - Contribution per People
- How much luggage expected by ride owner
  - Luggage Size (light, medium, heavy)

Ride Request

- Members can look at the list of available rides or put in request for a specific trip.
- Request added defaultly as ‘submitted’.
- Only ride owner can approve or reject it.
- Available Seats in the ride will be adjusted for each approval or rejection

Stops

- Traveler might wants to get off some cities between source and destination. For such situations our system should allow ride owners to set stops.
- Travelers cost will be calculated according to their stops
- Members can make a request from and to stops.

2. Define the aim (or aims) of implementation of your database

- Provide efficient, convenient and safe access to persistent data.
- Make development process easier and shorter with flexible and smart design.
- Concurrent access.
- Recovery from crashes.

II. Data sets

Define data sets generated by particular processes (data sets should be generated separately for each process). Name every data set and define its structure (attributes for data set).

Creating an account

- Member(Member Id, First Name, Last Name, Email, Contact Number, Driving Licence Number, Driving Licence Valid From)

Creating the list of available rides

- Ride(Ride Id, Member Car Id, Created On, Travel Start Time, Source City Id, Destination City Id, Seats Offered, Contribution Per Peaple, Luggage Size Id)
- City(City Id, Description)
- Luggage_Size(Luggage Id, Description)

Creating a ride

- Car(Car Id, Name, Brand, Model, Year, Comfort Level)
- Member_Car(Member Car Id, Member Id, Car Id, Car Registration Number, Car Color, Car Vin Number)
- Member_Preferences(Member Id, IsSmokingAllowed, IsPetAllowed, Music Preference Id, Talk Preference Id)
- Ride(Ride Id, Member Car Id, Created On, Travel Start Time, Source City Id, Destination City Id, Seats Offered, Contribution Per Peaple, Luggage Size Id)

Making a request

- Request(Request Id, Requester Id, Ride Id, Stops Id, Created On, Request Status Id)
- Request_Status(Request Status Id, Description)
- City(City Id, Description)

Creating the list of submitted requests

- Request(Request Id, Requester Id, Ride Id, Stops Id, Created On, Request Status Id)
- Request_Status(Request Status Id, Description)

III. Relational model of the database

This part should include:

- the graphical presentation of the relationships between the particular tables (diagram)
- the structure of particular tables (column names and data types) presented on the diagram
- all necessary constraints defined for tables (NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY) presented on the diagram