Rent-A-Ride: Car Rental Service
Problem Statement
Build a computer system to support a car rental service. (Think of zipcar.com)
Our car rental company serves a small market and only has a few hundred cars. As the traditional car rental companies usually do not rent cars to people under the age of 25, this company decided to start renting cars to young people, using very flexible rental agreements (older drivers should be able to rent, as well).
The rental company will have a very small permanent staff and the system will only be available online. Drivers should be able to register with the company and approve their driving credentials (usually a driver's license and an address). The registration requires a 6-month membership fee (renewable). Once registered, drivers should be able to request a vehicle for a certain date and time to be picked up at a specific location. Furthermore, the user should be able to select one of the available vehicle types, for example, a regular car, a pickup truck, or a van. If a requested car is not available at a desired location, the system should check if a similar car is available at an alternate location. Currently, our company has 200 locations, but more may be added in the future.
Rental locations are just regular parking lots where the vehicles are available for rental. Picking up and returning vehicles is on a self-serve basis. Registered users should be able to rent vehicles by the hour, with the maximum of 72 hours (3 days). The price per hour should drop a bit as the time of the rental increases. A late return should incur a late charge.
The administrator of the system should be able to define and maintain a number of vehicle types, enter information about the individual vehicles and specify their types and other important properties. Different vehicle types should have different (per hour) rental prices, defined by the administrator. The administrator should also be able to define rental locations and assign vehicles to them. Furthermore, the administrator should be able to remove a user (customer) for serious violations of the rental agreement.

**The system outline: high-level requirements**

1. The system must allow the system administrator to define and enter into the system vehicle types, such as a small car, full-size car, truck, or a luxury car. Since vehicles are rented per hour, the administrator must be able to set an hourly rental price for each vehicle type. Furthermore, the price should be settable for hourly ranges, for example, 1-5, 6-10 hours, etc. The administrator should be able to set a late return fee and a 6-month membership price, as well.
2. The administrator should be able to enter rental locations into the system. Each rental location should have a name, address, and a vehicle capacity (the maximum number of vehicles it can hold). A number of vehicles (see below) are assigned to each rental location.
3. The system must allow the administrator to define and enter into the system individual vehicles. A vehicle should have a defined vehicle type,

and a number of properties, such as the make and model, year, registration tag, current mileage, and the time it was last serviced. Also, each vehicle's condition is specified (good, needs cleaning, needs maintenance, etc.). Each vehicle should be assigned to a rental location.

4. The administrator should be able to make changes to any of the information currently stored in the system. For example, it should be possible to change rental prices, reassign vehicles to different locations, modify vehicle properties, etc. It should be possible to remove vehicles, rental locations, etc.

5. A rental system user (a customer) should be able to register with the system. To do that, the user must establish the user name and password, and then provide his/her driver's license state and number, email address, residence address, and a credit card information to be used for payments. The user must pay the initial 6-month membership fee. The user should be able to modify this information and extend his/her membership.

6. It should be possible to browse and search rental locations and vehicles there, as well as vehicles alone.

7. The user should be able to place a reservation for a vehicle at a selected rental location. The reservation must specify a vehicle type, vehicle pickup time and the length of the rental. The system should check if the requested vehicle would be available at the requested time and place and create a reservation. If a request cannot be granted, the system should suggest a similar rental vehicle at a different location.

8. The user should be able to cancel an existing reservation up to one hour ahead of the scheduled pickup time. Otherwise, a minimum charge of one-hour rental should be applied.

9. The user should notify the system as soon as the car is returned to the rental location. The user is charged for the vehicle time starting with the reservation time and ending at the return time. If a vehicle is returned late, a late return fee may be applied in addition to the rental charge. The user may enter information about the condition of the returned vehicle. Also, the user should be able to provide comments about the vehicle and the rental service in general, if desired.

10. The user should be able to terminate the membership at any time. The membership fee is not refundable.

11. The administrator should be able to terminate the membership of a user, if necessary.

12. The system must be accessible from a common Web browser (assume Google Chrome for now).

13. The system should provide multi-user access, assuring correct concurrent behavior. The system should maintain suitable authorization information and validate access. User authentication should be implemented (by checking user id and password).

14. The system must have an easy-to-use user interface (UI) with screens designed for each part of the system's functionality and suitable for different types of users (customers, administrators, managers).

15. The system should use a persistent data store.

16. You may use any Tech stack of your choice