Software Design Specifications

for

Car Rental Web Application System

Prepared By:
Sumit Katiyar(IIT2019110)
Anurag Bandopadhyay(IIT2019126)
Rahul Kumar (IIT2019109)
Aditya Singh Machhaiya (IIT2019111)
Nandini Gupta (IIT2019149)

Indian Institute of Information Technology, Allahabad

Date created: 20th September, 2021

Table of Contents:

1. Introduction

1.1. Purpose

The purpose of this document is to present a detailed description of the Occupancy Monitoring System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

1.2. Document Conventions

The content of this document is written in Times New Roman with font size 12. The main titles of the body are written in Times New Roman with font size 24 while the subtitles are written in Times New Roman with font size 14. The margins are set as follows: left and right is 1” while top and bottom is 1”. The line spacing is selected as exactly 1.2 point. Bold formatting is used for headings and subheadings.

1.3. Intended Audience and Reading Suggestions

This project is a prototype for the occupancy monitoring system, and it is intended for teachers under whose guidance it has been developed or any other developers who want to develop it further or any end users or customers. This project is also useful for end users who want to monitor the occupancy of any confined area which has monitored entry and exit points.

1.4. Product Scope

The Car Rental System can be used in various cities where it is easier and a more affordable for the common people to rent a car for a particular purpose for a few days. It allows them the flexibility to select the type of car they want to drive and also the duration during which they want to avail the car. Therefore, instead of affording the entire price of the car, the user will just have to afford the rental price for the duration of which they used the car.

1.5. Definitions

- SDS: Software Development Specification
- PHP: Hypertext Preprocessor
- CSS: Cascading Style Sheets
- HTML: Hypertext Markup Language
- LDAP: Lightweight Directory Access Protocol.
- MySQL: Open-Source Relational Database Management System.

1.6. Requirements and Acknowledgements

- IEEE Software Requirements Specification Template
  https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

- IEEE. IEEE Std. 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998

2. Use Case

2.1 Actors

The Use Case of our project contains three actors: **Guest, User , Administrator.**

This is an abstraction of the specific users as they all perform distinctive actions that are required for the communications to take place on the portal.

2.1.1 User

The registered members are allowed to update their personal information if necessary with their own valid username and password through the proposed system. Each of the updated personal information will be stored and kept in company's database and categorized it as company confidential information. And, the company will not sale their customer's information to any third party to earn extra income for their company.

The registered user can view all the available cars for rent and select as per their requirements. Then he/she can select the dates from and to when they want to book the car and apply and wait for the Administrator's confirmation. In addition the User can also write a feedback, a contact us query, and even view every booking they have made in the system.

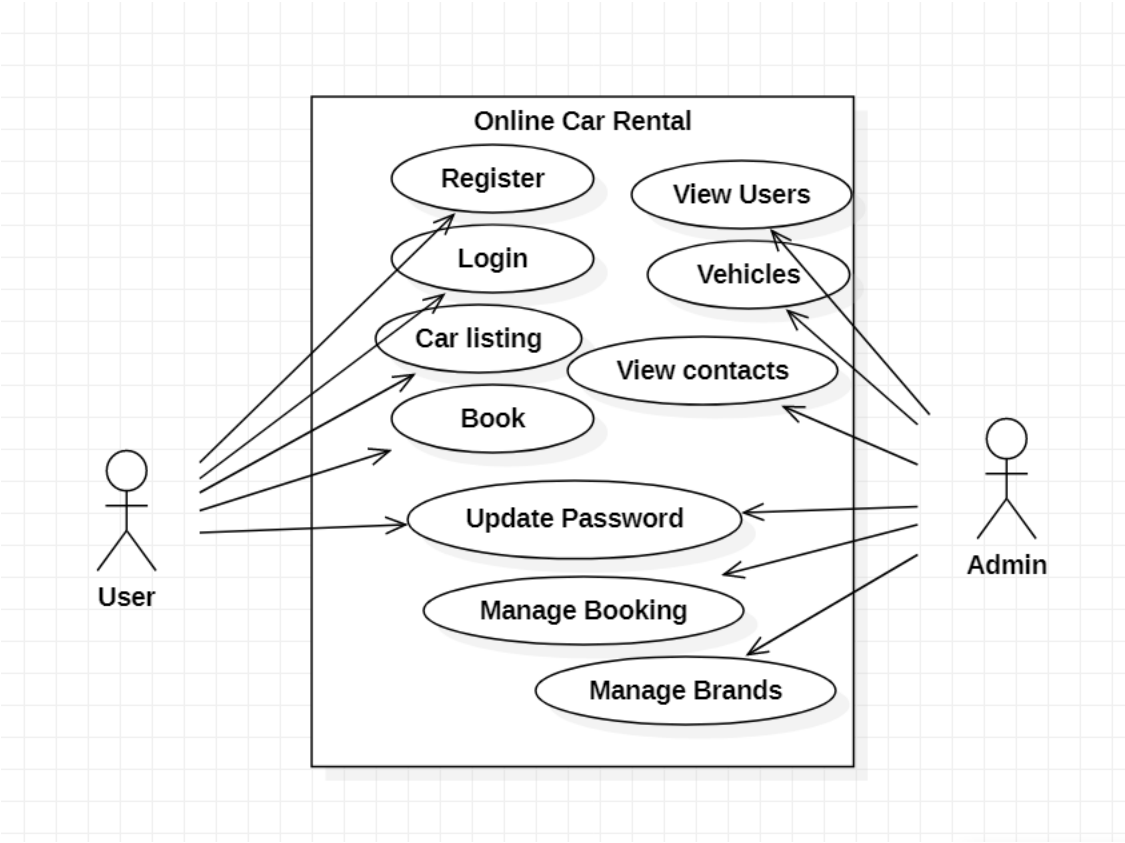
2.1.2 Guest

The unregistered members or users are restricted to view, update or delete any information about the registered members' information such as car reservation history. They are allowed to view the latest promotion of the company, the available cars, the features we offer to the users and can register themselves in the system.

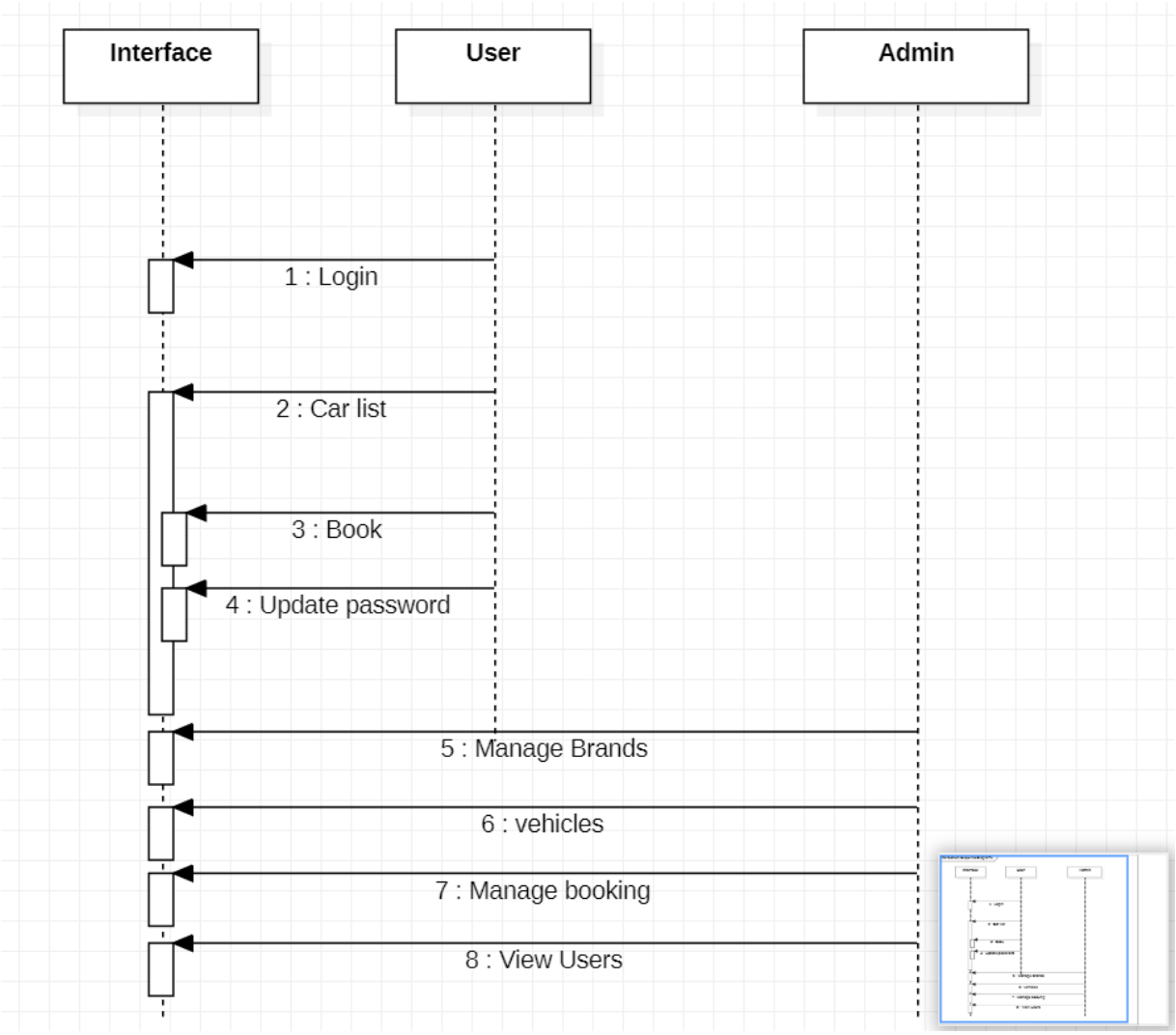
2.1.3 Administrator

The web administrator is the one who has full control and authority to control the privilege and update the proposed system from time-to-time. In addition, they also required to ensure that the different types of user access via their permitting level. They can add, update or delete various entities such as the registered users, the registered vehicles, and the feedback queries.

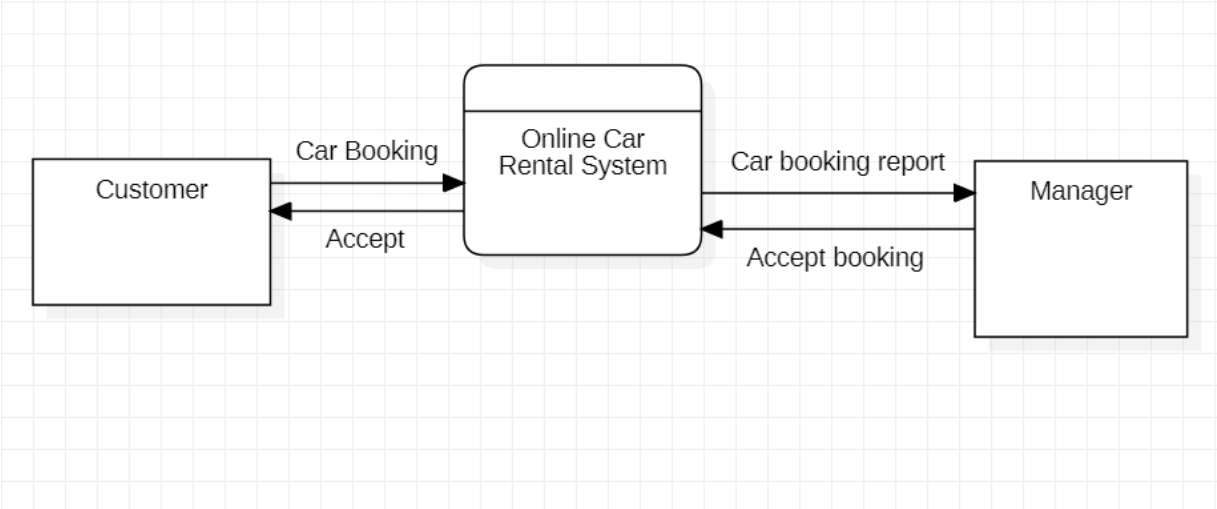
2.2 Use Case Diagram



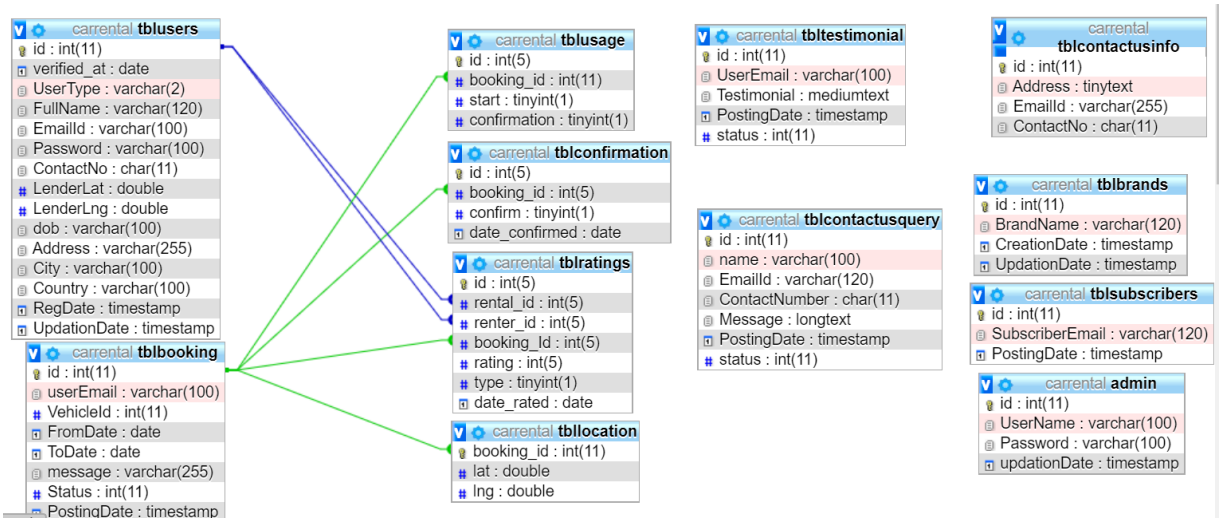
2.3 Sequence Diagram



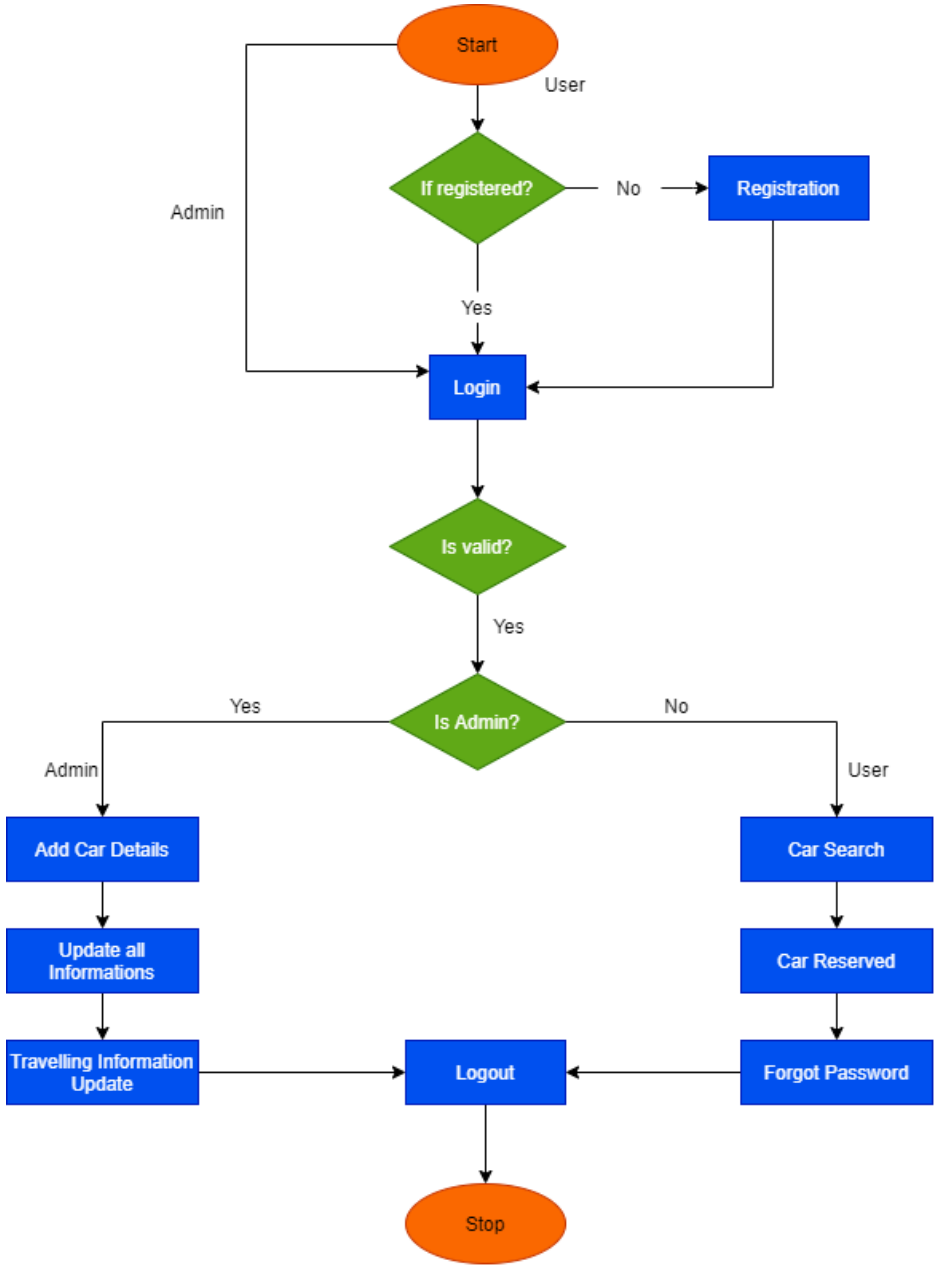
2.4 Data Flow Diagram



2.5 Entity Relationship Diagram



2.6 Flowchart



3. Logical Architecture Description

4.1 Product Functions

We describe the functional requirements by giving various use cases.

- Actor 1) Guest
  - Use Case 1) Guest can log in
  - Use Case 2) Guest can sign up
  - Use Case 3) Guest can view all the cars available for renting
  - Use Case 4) Guest can write a contact us query in order to contact the system administrator.
  - Use Case 5) Guest can view all the feedbacks made visible by the admin.
- Actor 2) Registered User(Person who wants to rent a car)

  >>The user is the end-user of our service. Users can view information of the available car, booking a car, easily get the car on rent, and also give feedback and an inquiry. Users also view the discount and other information to get the best deals.
  - Use Case 5) User can view all the available cars for renting.
  - Use Case 6) User can apply to book the car for renting for a specific period of days as per their need.
  - Use Case 7) User can view all their booking applications and their status.
  - Use Case 8) User can view his/her profile details.
  - Use Case 9) User can update his/her profile details.
  - Use Case 10) User can update credentials(email and password).
  - Use Case 11) User can write feedback to the administrator.
  - Use Case 12) User can write a contact us query in order to contact the system administrator.
- Actor 3) System Administrator

  >>Admin is basically a superuser. Admin can add a car, manage booking cars, and rent and view feedback and inquiry. Admin will keep track of each booking. Manage organization representatives. Admin is responsible for any error in the system. So, he needs to alter at any point in time. Admin should keep tracking car renting service, maintenance of cars.
    - Use Case 13) Admin can view all the users registered into the system
    - Use Case 14) Admin can view all the cars listed in the system.
    - Use Case 15) Admin can view all the brands whose cars are available in the system
    - Use Case 16) Admin can view all the feedback query written by the users.
    - Use Case 17) Admin can view all the contact us queries written by the users.
    - Use Case 18) Admin can edit or delete users from the system.
    - Use Case 19) Admin can edit or delete car details from the system.
    - Use Case 20) Admin can control which feedback is displayed in the webpage