Online Driver Booking System

Software Requirement Specification (SRS) Document

Sprint 1 Implementation

Project Timeline: 20.10.2022 to 27.10.2022

INDEX

1. Introduction: -

The introduction of the software requirement specification provides an overview of the entire software. The entire SRS with overview description purpose, scope, tools used and basic description. The aim of this document is to gather, analyze and give an in-depth insight into the complete **Online driver booking application** by defining the problem statement in detail. The detailed requirements of the **Online driver booking application** is provided in this document.

1.1 Purpose: -The purpose of this document is to show the requirements for the Online driver booking application, in which it is used to book a driver for weekly drives and many other situations. The online driver booking application is an user friendly application to book drivers online with in few clicks only.

1.2 Intended Audience: -This document is intended to be read by, Client.

1.3 Intended Use: -

- Development Team
- Maintenance Team
- Clients

Since this a general-Purpose Software any one can access it.

1.4 Scope: -This project aims to create the development of an online driver booking application. Which takes drivers information such as driving license number, name, age, address, phone number and generates a unique serial number as ID and the user information such as Aadhaar number, name, address and age. The admin will be having both users and drivers information and has report about all the rides from starting point to end point, distance in km, successfully completed rides and cancelled rides.

2. Overall Description: -

It is an online driver booking application used to book a driver for weekly drives or many other situations. The drivers can set their profile by providing their driving license number. When the user wants a driver, the user should login with their details. After that the user should provide the route by that route it will display the list of drivers for that particular route. Then the user can book a driver by entering the driver ID. If the driver is already booked then the user can choose any other drivers who are available. Based on the route selected and number of days entered by the user the driver is provided

with a booking confirmation for the driver along with the amount to be charged. The user is also asked to pay some advance for the booking. The user is also allowed to cancel a booking in case of which he will be charged 2%of the total booking amount. Once a ride is carried out and completed, a completion report is displayed and the details updated in respective files. The admin of the system can do the user and driver database maintenance through authorized access. The admin will be having both the users and the drivers information and has report about all the rides from starting point to end point, distance in km, successfully completed rides and cancelled rides. The main purpose of this project is to provide a user-friendly experience to both the users and drivers.

2.1 Assumptions and Dependency: -

- System should have Ubuntu Linux installed.
- ·System should have either 4GB or more RAM.
- The service is used preferably on a desktop or laptop.

3. System Features and Requirements: -

3.1 Functionality: -

Introduction–

This subsection contains the requirements for the online driver booking system. These requirements are organized by the features discussed in the case study provided to us. Features from case study are then refined into use case diagrams and to sequence diagrams to best capture the functional requirements of the system.

3.1.1 ODB_01->User_ Login: User login by entering password

3.1.2 ODB-02->Driver_ Login: Driver login by entering password

3.1.3 ODB-03->Admin_ Login: Admin login by entering password

3.2 User's corner:-

3.2.1 ODB_01-> Add_ Data: This function is the start point of the application as the user creates their own profile by entering user details.

3.2.2 ODB_02-> Edit_ Data: This function is used to edit/update any kind of changes into the existing records in the database itself.

3.2.3 ODB_03->Book_ Driver: To book a driver, the user should specify the route. By the route their will a list of drivers will displaying on the screen. By the list the user can select a driver and to book the driver the user should give driver's unique ID. When the driver is already

booked the user can choose another driver. The User should specify the route and no of days by that the driver can charge the money.

3.2.4 ODB_04->Show_ Top_ Drivers: This function, will display the driver's data who has completed most rides at the top.

3.2.5 ODB_05->Cancelled_ rides: In the cancelled rides, the rides which are cancelled after booking a driver are shown in this function. The admin will maintained all the cancelled rides.

3.2.6 ODB_06-> List_ To_ User: This function will copy all the data from the linked list and store it permanently on a file. The function writes each node of the linked list to the file in serial order.

3.2.7 ODB_7-> User_ To_ List: The data in the file will be converted or pushed into the linked list. This will be done at the starting of the program so the data can be used in the program. This function will return the user database records to the main function.

3.3 Driver's corner:-

3.3.1 ODB_01-> Add_ Data: This function is the start point of the application as the driver will create their profile by giving driving license no, name, age, address and phone number. The admin will specify the list of routes for the drivers by that the driver can select the route which he can hire and also admin will generate the unique ID for drivers.

3.3.2 ODB_02-> Edit_ Data: This function is used to edit/update any kind of changes into the existing records in the database itself.

3.3.3 ODB_03-> List_ To_ Driver: This function will copy all the data from the linked list and store it permanently on a file. This function will be executed when the driver completed the ride. The function writes each node of the linked list to the file in serial order.

3.3.4 ODB_04-> Driver_ To_ List: The data in the file will be converted or pushed into the linked list. This will be done at the starting of the program so the data can be used in the program. This function will return the user and drivers database records to the main function.

3.4 Admin corner:-

3.4.1 ODB_01-> User maintenance: In this function the admin can only maintain all the details of the user.

3.4.2 ODB_02-> Driver maintenance: In the function the admin, can only maintain all the details of the driver.

3.4.3 ODB_03->Delete_ Data: As the name suggests, the job of this function is to delete a certain record from the database. The admin can only delete the user and driver's data.

3.4.4 ODB_04->Display_ records: In this function, we can see both user and driver's list whose list of present.

3.4.5 ODB_05->List_ of_ routes: In the record function, the successfully completed rides in a record, the admin will maintaining all the reports in a file such as routes, starting point, end point, distance in kms.

3.4.6 ODB_06-> Ride_ Reports: In the ride reports, the successfully completed reports will be stored.

3.4.7 ODB_07-> Report_ To_ List: The data in the file will be converted or pushed into the linked list. This will be done at the starting of the program so the data can be used in the program. This function will return the user and drivers database records to the main function.

3.4.8 ODB_8-> List_ To_ Reports: This function will copy all the data from the linked list and store it permanently on a file. This function will be executed when the driver completed the ride. This function will copy the contents of linked lists and store them in the ODB database. The function writes each node of the linked list to the file in serial order.

3.5 System Requirements: -

3.5.1. Tools to be used:

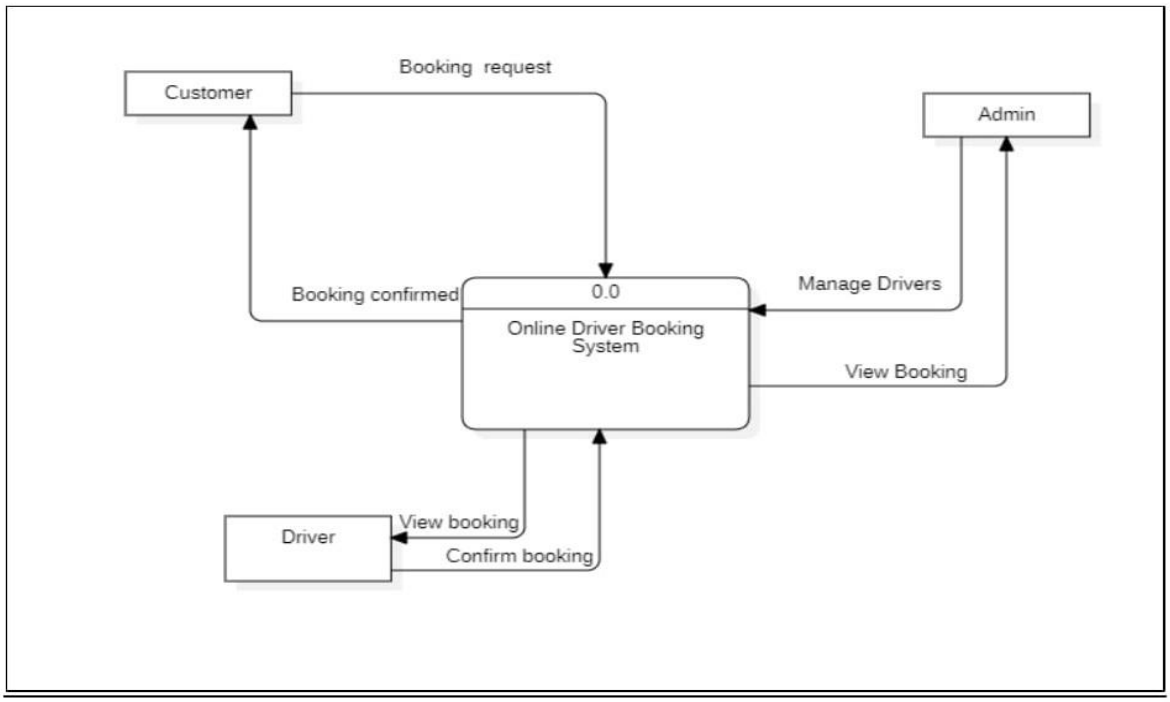
- Library
- C File Handling
- C Language
- System Programming

3.6 System Features: -

- Supportability: The system is easy to use.
- Design Constraints: The system is built using only C language.
- Usability: This application is used for to book the driver for long period of time or else for short period time. And in this application we have designed in a way that after the user specified the route the user can see all the most experienced drivers in the top of the list by selecting a driver and specify him the route no. of days then the user can directly communicate to the driver about the amount. After completing the rides the user also has an option to give rating for the particular driver for the ride by that the driver can go to the top of the list. All this can be done by just one click from the user side this application is user friendly for the customers to book a driver from anywhere.
- Reliability & Availability**:** The system is available 24/7 that is whenever the user would like to use the system, they can use it up to its functionalities.
- Performance: The system will work on the user's terminal**.**

4. Data Flow Diagram:

4.1DFD Level 0 –



4.1DFD Level 1 -