ELECTRIC CAR RENTAL

PREPARED FOR
CHIPP INC.

PREPARED BY

Kongani Sumanth (AP18110010413)
Avula Prabhu Kiran (AP18110010409)
K Ruthvik Reddy (AP18110010417)

MARCH 24, 2021

Table of Contents

1. Introduction

Over the past century, the world has become accustomed to pollution from vehicles. Lots of carbon monoxide particles being released into the air from vehicles of all kinds have not only polluted our planet heavily but have put future generations at risk. Fortunately, since the turn of the past decade, most of the world is slowly moving towards a cleaner form of energy to use in vehicles, homes, etc. In the automobile industry, electric vehicles have become a major force moving forward towards achieving a net-zero carbon emission goal. Our product combines that goal of a cleaner society with the idea of a car rental service. An Electric Car Rental (**ECR**) web application is to be developed for people who seek to go to work, rent for a trip, or travel with zero tailpipe emissions, at affordable costs.

1.1 Purpose

This SRS document is meant to delineate the features of the **ECR** web application, as a software validation document for CHIPP Inc. and to serve as a guide to the developers. The Electric Car Pooling, Rental and Ride-Hailing Software specializes in providing electric carpooling services, rental services, and ride-hailing services across cities in India with electric cars. Users can avail the services by just visiting the website or mobile application, selecting the service from any locations in selected cities.

1.2 Intended Audience

This document aims to detail the various attributes and requirements pertinent to the software being developed as a solution to the **ECR** in demand. This SRS document is primarily intended for the following groups of readers:

- ❏ Developers and specialists tasked with the maintenance, further development, and the release of the software.

- ❏ Potential clientele involved directly in the design and development of the software.
- ❏ Personnel responsible for the documentation of the aforementioned software.
- ❏ Testers and Quality Assurance(**QA**)/Quality Control(**QC**) engineers contingents associated with this software development.

1.3 Product Scope

This Software Development project aims to provide a slick user experience and efficient interface for renting an electric car, by using user location. The software will provide full functionality of the car i.e its specifications, features, price, etc.

1.4 Definitions, Acronyms, and Abbreviations Used

ECR	Electric Car Rental
DBMS	Database Management System
MB	Megabyte
GB	Gigabyte
RAM	Random Access Memory
OS	Operating System
JS	JavaScript
UI	User Interface
EULA	End-User Licence Agreement

1.5 References

https://medium.com/trailblazer-of-salesforce/software-requirements-specification-srs-document-fd9ab103b18#_Toc77487652

1.6 Overview

In the Second Chapter, we will discuss the description of the software, the product's perspective, product functions, user characteristics, constraints, assumptions and dependencies pertaining to the entire project.

In the third chapter, we will discuss the external interface requirements such as user interface, admin interface, hardware and software interfaces requirements and the communication interface requirements for this project.

In the fourth chapter, we will discuss the system features such as sequencing, validation, error handling and fail safe operation scenarios of this project.

In the fifth chapter, we will discuss the non functional requirements such as performance requirements, usability requirements, reliability requirements, supportability requirements, packaging requirements and the legal requirements for this project.

2. Software Description

2.1 Product Perspective

The objective of this product is to provide an efficient electric car rental system, whose main functionality, apart from providing a seamless experience from end to end vendor to client), is to monitor and give feedback to the organization. It will help enable the vendor to see and analyze the performances of the vehicles from an organization's standpoint, upon which the vendor can add, delete and manage the vehicles in his inventory in a better manner.

The product is expected to be used in any OS with web browser compatibility.

- ❏ RAM capacity: at least 512 MB

- ❏ Any OS with Networking and Code compatibility
- ❏ A browser with cookies and JS support
- ❏ Physical storage capacity: less than 1 GB
    - ❏ The physical server requirements of the server, supporting all DBMS and Networking requirements are as follows:
    - ❏ RAM capacity: 8GB
    - ❏ Physical storage capacity: minimum 25 GB

2.2 Product Functions

The core functionality of the product itself is built upon two parts of a transaction or a deal between two parties. Before informing about the user requirements, let us study the main product requirements:

Current Vehicle Inventory: From the admin's viewpoint, the dashboard shows the entire, detailed inventory list of all the vehicles present and available for rent. Every detail pertaining to a vehicle can be accessed by clicking upon the vehicle. This means that the vehicle's type, make, price, specifications, and accessories are all present to view for the admin. Upon selecting a vehicle, the customer will also be able to see all the details pertaining to the vehicle of his choice.

User Transactions: From the main dashboard on the user's perspective, a separate page is available to view in which every transaction made by the user is present. Again, this page has each rental transaction pertaining to that user's account and upon clicking any of those transactions, the user can view the details such as the vehicle's type, make, price, period of rental, specifications and accessories, etc. Likewise, once a transaction is initiated from the user's side, the admin will receive a notification that includes all the information regarding the transaction and the vehicle.

Admin Functionality: The admin's side of the interface has all the necessary tools required to:

- ❏ View the vehicle inventory.
- ❏ Add or Delete a vehicle.
- ❏ Update a vehicle's information.
- ❏ Generate and view the invoices of all transactions.
- ❏ View and analyze feedback from customers.
- ❏ View and approve or reject customer testimonials.
- ❏ View and reply to customer grievances.
- ❏ View and Alter the records of new and existing employees in the organization.

Search Functions: Each dashboard page is provided with a search function at the beginning, which allows for searching across the page's data, both by keywords and criteria (such as car names, order dates, and so on). Results are populated directly below the search bar, thereby saving users time and effort that would otherwise require them to manually sift through lots of paper records and tallies.

Alerts and Notifications: The user and admin are continuously updated by any new information regarding their transaction by alerts and notifications that pop up. This facilitates both parties to make quick, smart decisions.

2.3 User Characteristics

Since this application is expected to be used for both internal and external operations of the business, the users are assumed to be directly connected with the organization's user department once they undergo the registration process. At smaller scales, the following use cases could be fulfilled by the same person, but for the sake of clarity and detail, we will assume that the

following roles exist in the organization, and operate on this software through one (or more) computers. The following are the intended users of the software:

Car Rental Agent: The person in charge of the physical aspects of the transactions such as vehicle pick up and delivery, communicating with the clients in person and making sure that essential paperwork and insurances (such as driver's license verification, insurance papers, and the registration card of the vehicle, etc) are in order before and after the rental period.

Manager: The person in charge of the rental outlet, this is someone who actively monitors the overall situation on a frequent basis. He is primarily concerned with Incoming transactions and current transactions. The manager is also responsible for the overall day-to-day functionality of the staff present. He is also responsible for analyzing the data of finished rentals, customer grievances, and customer testimonials.

Owner/Administrator: The overall owner of the organization, inclusive of all their outlets, this person will be concerned with all current transactions pertaining to acquiring vehicles and managing them. Since they own the enterprise, they are the ones who negotiate with other vendors for their business and add or delete available vehicles (post a successful negotiation) to the vehicle inventory page. They are responsible to note how their business is performing and make decisions to bring in new unique products or discontinue the sale of existing products, both of which they must update regularly.

2.4 Constraints

The major constraints enforced by this software are as follows:

- ❏ The software requires a memory of at least 512 MB to store and index data.

- ❏ The software must only be run on a network-compatible OS computer in a web browser.
- ❏ The server from which the software is accessible must be handled only by associates and employees of the organization, for data security reasons.
- ❏ The data being recorded into the software must be in an indexable format (such as plaintext), so as to maintain search functionality. Recording of non-indexable elements such as images and barcodes are not supported by the software.
- ❏ The software must be accessed from browsers with cookie and JS compatibility only.
- ❏ The “server” must definitely have at least 8 GB of RAM, so as to ensure operation without interference.

2.5 Assumptions and Dependencies

- ❏ The project implements an authentication mechanism since this project is a transaction type that requires the verification of both parties before it takes place.
- ❏ The project depends on a machine capable of running code, having network compatibility, and is updated enough to be able to run DBMS software such as MySQL, and connector software such as phpMyAdmin.
- ❏ While there are no explicit quantitative measures being enforced on the software in terms of speed and memory, it is generally assumed that the various functions of the software are optimized for speed and memory, since this would be a critical element of an operational business.
- ❏ Since it is a real-time updating software, it is assumed that no data backups are needed for the backend databases, not only because the volume of the data is expected to be low, but also since the software is

implicitly optimized for failsafe and interruption handling capabilities, and saves its state after every update.

3. External Interface Requirements

3.1 User Interface

Since this is a web-based application, the browser pages will be the focal access points and operation of this software for the end-users. For all the operations described above, there is a dashboard-driven interface made available to the users, which will be a composite of the following screens:

- ❏ Upon navigating to the URL serving this software site, the user is taken to the current home page, where he is presented with a navigation bar with buttons for other pages. If the user is new to the interface environment, he will need to undergo a registration process by furnishing important details such as name, phone number, email address, date of birth, a username, and other relevant details. The user will be asked to agree to the organization's terms and conditions Upon finishing the registration process, the user can now log into the website using his account so that all the rentals he purchases will be linked to his account.
- ❏ Once the user has logged in, he will be able to initiate and go through the complete process of renting an electric car. First, the user is prompted to enter a few details pertaining to the type of car he requires such as the car's class (SUV, Sedan, Hatchback, Pickup Truck, Sport, Minivan, etc), the brand of the car (TATA, Hyundai. TESLA, etc), the city they need to rent the car in (Out of 30 Major Cities and all Union Territories in India), and their price range as well as the date of booking and the date of return.

- ❏ Upon furnishing all the required details, the user will be redirected to another page where all the vehicles that fit the required criteria will be shown. The user can then select one of the vehicles.
- ❏ Upon selecting a vehicle of his choice, the user will see all the information about that vehicle such as its type, brand, price, seating capacity, specifications (color, range before a recharge is required, etc), accessories (automatic windows, air freshener, electronics charging ports, etc), and an option to change or alter the dates for the start and finish of the rental period. Upon finishing the required formalities, the user may proceed towards the payment.
- ❏ Payment can be made using both online or offline modes. Upon the verification of the payment, the user will receive the invoice along with the pickup and drop off point for the vehicle (which is one of the organization's outlets across the country). The user may go to the specified location and pick up the vehicle and start his rental period.

3.2 Admin Interface

Since this is a web-based application, the browser pages will be the focal access points and operation of this software for the end-users. For all the operations described above, there is a dashboard-driven interface made available to the administrator(s), which will be a composite of the following screens:

- ❏ Upon log in, the administrator or other employees that are granted access will be redirected to a dashboard that has all the required information pertaining to all bookings, customers, vehicles and employees. The admin can navigate through these pages via a menu provided at the side of the interface.

- ❏ The admin has access to all the users' information. Upon registration, all the users' information such as name, email, phone number and username is automatically updated into the database first, and can be viewed by the admin in the 'Registered Users' page.
- ❏ Also, the admin has access to the whole list of all the vehicles under their branch. The admin can add, or delete a vehicle, and he can also alter the details of existing vehicles. In order to add a vehicle, the admin will need to furnish information pertaining to the vehicle's make, type, model year, price per day (in INR), accessories included, specifications, color, seating capacity, and a few images for visual reference.
- ❏ The admin also has access to all the bookings made. He can access old bookings, currently active bookings and future bookings once made from the users' end. He has the ability to confirm the bookings based on the availability of the required vehicle.
- ❏ The admin also has access to customer feedback, grievances and testimonials. He may access them and reply to them and analyse them so that moving forward, the organization can make less errors. He also has access to the contact page of the organization itself. Here, he can update the existing information about the company such as its address and contact details. He also has access to send newsletters to those users who have subscribed for the same.
- ❏ The admin is also in charge of all the payments made. He will oversee the verification, booking status, and confirmation of all necessary paperwork (Such as Insurance papers, vehicle registration, driver's license, etc) in the case of virtual payments.
- ❏ Finally, the admin also has access to the information regarding every employee working in his branch. He may add or delete employees

based on their employability status, and he may also alter information of existing employees upon receiving a request and confirmation to do so.

3.3 Hardware Interface Requirements

This software does not require interaction with any dedicated hardware interfaces or components, other than generic computers, owing to the fact that this is a web-driven interface. The details of the hardware specifications of those machines have been elaborated in an earlier section(Section 2.1) of this document.

3.4 Software Interface Requirements

- ❏ This software is dependent on a small number of other tools and software, which are listed (but not in an exhaustive manner) below:
- ❏ Compatible Operating Systems:
  - ❏ Android, iOS, and Linux mobile with a web browser.
  - ❏ Window, all Linux and macOS with a web browser
- ❏ MySQL version 5.5 or above, MariaDB version 5 or above. (For the DBMS backend)
- ❏ phpMyAdmin 4.4.0 or above. (For the connector and server processes)
- ❏ HTML5, CSS3, JavaScript or above compatible browsers. (For the UI)

3.5 Communication Interface Requirements

To allow remote access of the product to the users the following communication interfaces may apply:

- ❏ TCP/IP communication channel with server machine.
- ❏ Internet access to the users to remotely access the product web page.

4. System Features

4.1 Sequencing and Validation

Tables are created in the DBMS backend for the following:

- ❏ Vehicles with vehicleID, vehicle_name, vehicle_brand, and other relevant attributes
- ❏ Users with userID, name, email, password, phone_number, address, and other relevant attributes
- ❏ Admin with adminID, email, password, and other relevant attributes
- ❏ Booking with bookingID, email, vechileID, rental_start_date, rental_end_date, status, and other relevant attributes.

These tables are sequenced and updated via the phpMyAdmin backend connector service, which has inbuilt collision avoidance, non-repudiation, and integrity verification.

4.2 Error Handling and Failsafe Operation

phpMyAdmin can be configured to provide an incremental backup facility for the tables, and also handles interrupts and outages inherently. In case of severe machine or memory failures, the backup can be restored, provided it was initially configured. In case of an overload or peak capacity operation, phpMyAdmin throttles its operations to restore safe operating conditions, although the software would be incredibly slowed down during said throttle.

5 Non-functional Requirements

5.1 Performance Requirements

- ❏ The system must not lag, because the users using it don't have down-time to wait for it to complete an action, 95% of all response time should be less than 4 seconds.
- ❏ The system must complete updating the databases, updating and managing cars, bookings, and timestamps successfully every time the user requests such a process.
- ❏ All the functions of the system must be available to the user every time the user accesses our website.
- ❏ The changes performed by the admin must be updated in real-time and should not vary unless explicitly changed by the admin.

5.2 Usability Requirements

- ❏ The system must be easy to use by both users and the admin such that they do not need to read an extensive amount of manuals.
- ❏ The system must be quickly accessible by both users and the admin.
- ❏ The system must be intuitive and simple in the way it displays all relevant data and relationships.
- ❏ The menus of the system must be easily navigable by the users and admin with buttons that are easy to understand.

5.3 Reliability Requirements

- ❏ The System must give accurate vehicle, user, and employee information inventory statuses to the manager and admin continuously. Any inaccuracies are taken care of by the regular confirming of the actual levels with the levels displayed in the system.

- ❏ The System must successfully add any details given by the user and admin and provide inventory status in relevance with the newly updated entities.
- ❏ The system must provide a password enabled login to the phpMyAdmin instance to the owner to avoid any foreign entity changing the data in the system.
- ❏ The system should provide the user updates on completion of their transaction and if the transaction should fail, it should provide the user the reason for the failure.
- ❏ The system should not update the data in any database for any failed processes

5.4 Supportability Requirements

- ❏ The software is designed such that it works even on systems having the minimum configuration.
- ❏ The system is adaptable even if additional plugins or modules are added at a later point.
- ❏ The data can be exported to the manager so as to make the system more portable.

5.5 Packaging Requirements

- ❏ The system must be able to run on the operating systems detailed in the previous sections as intended.
- ❏ The software must incorporate a license key authentication process, if needed.
- ❏ The installation must come with a manual/readme that details the use of the system, and also the instructions on how to use the program.
- ❏ This manual may be included in a docs page that comes with the software.

5.6 Legal Requirements

An EULA may be drafted that is required to be agreed upon by the user to utilize the software for its intended purpose only, so as to prevent any legal repercussions elicited by faults caused by unsupported operational mechanisms.