

RideRight: The smart car rental system

A PROJECT REPORT

Submitted by

Anjali Sachan

in partial fulfillment for the award of the degree

Bachelor of Technology

IN

Computer Science & Engineering

GLA University, Mathura

November 2023

BONAFIDE CERTIFICATE

Certified that this project report **RideRight** is the bonafide work of **Anjali Sachan**
who carried out the project work under my supervision.

Signature of the HOD

Mr. Rohit Agrawal

Head Of The Department

Department : CSE

Signature of the Supervisor

Mr. Sanjay Madaan

Technical Trainer

Department : CSE

Submitted for the project viva-voce examination held on 29-November-2023

INTERNAL EXAMINER

EXTERNAL EXAMINER

CONTENTS

CHAPTER 1 Introduction

- 1.1 Motivation and Overview
- 1.2 Objective
- 1.3 Summary of Similar Application

CHAPTER 2 Software Requirement Analysis

- 2.1 Requirement Analysis
 - Functional Requirements
 - Non Functional Requirements

CHAPTER 3 Software Design

- 3.1 Data Flow Diagram
- 3.2 Use Case Diagram

Bibliography

Chapter 1: Introduction

1.1 Motivation and Overview

In the ever-evolving landscape of mobility solutions, the RideRide project stands as an innovative car rental platform that transcends traditional boundaries. While the mobile app serves as the primary interface for users, a critical component of our ecosystem is the web version built on the MERN stack. This web version, utilizing ReactJS, plays a crucial role in facilitating cross-platform access, providing structured control for administrators, and hosting essential features such as booking management and vehicle tracking.

The motivation behind RideRight goes beyond typical car rental applications. It is grounded in the integration of modern technologies like React Native, WebSocket protocol, and geolocation services, not only for the mobile app but also for the web version. The goal is to offer users a seamless, real-time car rental experience that goes beyond basic reservation functionalities, incorporating advanced features such as vehicle tracking and interactive user interfaces.

1.2 Objective

In the ever-evolving landscape of mobility solutions, the RideRide project stands as an innovative car rental platform that transcends traditional boundaries. While the mobile app serves as the primary interface for users, a critical component of our ecosystem is the web version built on the MERN stack. This web version, utilizing ReactJS, plays a crucial role in facilitating cross-platform access, providing structured control for administrators, and hosting essential features such as booking management and vehicle tracking.

The motivation behind RideRight goes beyond typical car rental applications. It is grounded in the integration of modern technologies like React Native, WebSocket protocol, and geolocation services, not only for the mobile app but also for the web version. The goal is to offer users a seamless, real-time car rental experience that goes beyond basic reservation functionalities, incorporating advanced features such as vehicle tracking and interactive user interfaces.

The primary objective of the RideRight project is to establish a robust car rental platform encompassing both the mobile app and the web version. The mobile app focuses on real-time booking and vehicle tracking, leveraging technologies like React Native, WebSocket, and geolocation services. Simultaneously, the web version, built on the MERN stack with ReactJS, extends the RideRight ecosystem by providing administrative controls, reservation management, and a structured platform for efficient fleet management.

Real-Time Booking Excellence:

RideRight is committed to delivering excellence in real-time booking, ensuring users can reserve vehicles seamlessly.

User-Friendly Interface with Expo and MERN:

RideRight provides a user-friendly interface, utilizing Expo for the mobile app and the MERN stack for the web version.

Openness and Privacy:

RideRight prioritizes an open yet secure environment, empowering users with control over their data and reservation preferences on both mobile and web platforms.

Cross-Platform Compatibility with ReactJS:

Our objective is to transcend platform limitations by using React JS for the web version, offering a consistent user experience across devices.

1.3 Summary of Similar Applications

In the competitive landscape of car rental applications, RideRight distinguishes itself by combining the strengths of existing platforms with a focus on real-time booking excellence, seamless vehicle tracking, cross-platform compatibility, user-friendly interfaces, openness, privacy, and continuous innovation. As we survey the current market, several notable applications serve as benchmarks for comparison, each with its unique features, strengths, and limitations.

Uber:

Recognized globally, Uber is a widely used ride-hailing application celebrated for its simplicity and real-time service. Concerns may arise regarding vehicle availability in certain areas

Hertz:

A prominent traditional car rental service, Hertz caters to users requiring longer-term rentals. However, concerns may arise regarding the reservation process and fleet availability

Chapter 2: Software Requirement Analysis

2.1 Requirement Analysis

The foundation of the RideRight project lies in a thorough analysis of software requirements, encompassing both the mobile app and the web version. The functional and non-functional requirements outlined below are essential for the successful development of this comprehensive car rental infrastructure.

Functional Requirements

Real-Time Booking:

- Users should reserve vehicles in real-time on both mobile and web platforms.
- Multimedia attachments, including vehicle images and specifications, should be supported seamlessly.

Vehicle Tracking :

- Right must support real-time vehicle tracking functionality using geolocation services on both mobile and web platforms.
- Users should track the location of reserved vehicles within the application on both platforms.

Cross-Platform Compatibility :

- The application should be compatible with various mobile devices, ensuring a consistent user experience.
- Cross-platform compatibility must extend to the web version built on the MERN stack.

User Authentication :

- Secure user authentication mechanisms should be implemented to protect user data on both mobile and web platforms.

Reservation Management :

- Users should manage their reservations, view booking history, and modify existing bookings on both mobile and web platforms.

Fleet Management

- RideRight must allow administrators to manage the fleet, add new vehicles, and track vehicle availability on both mobile and web platforms

Non-Functional Requirements

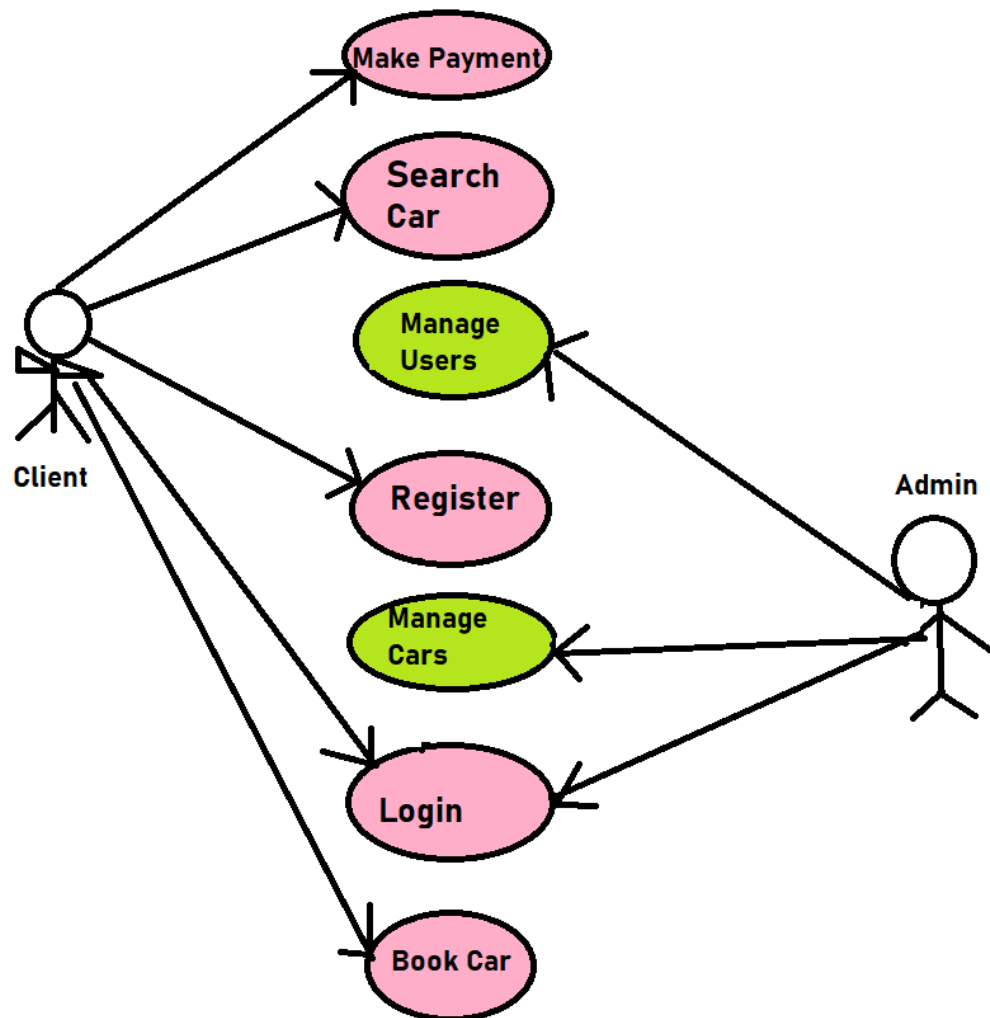
Performance :

- The application should offer smooth performance, even under high user load on both mobile and web platforms.
- Reservations and vehicle tracking should be updated with minimal latency on both platforms.

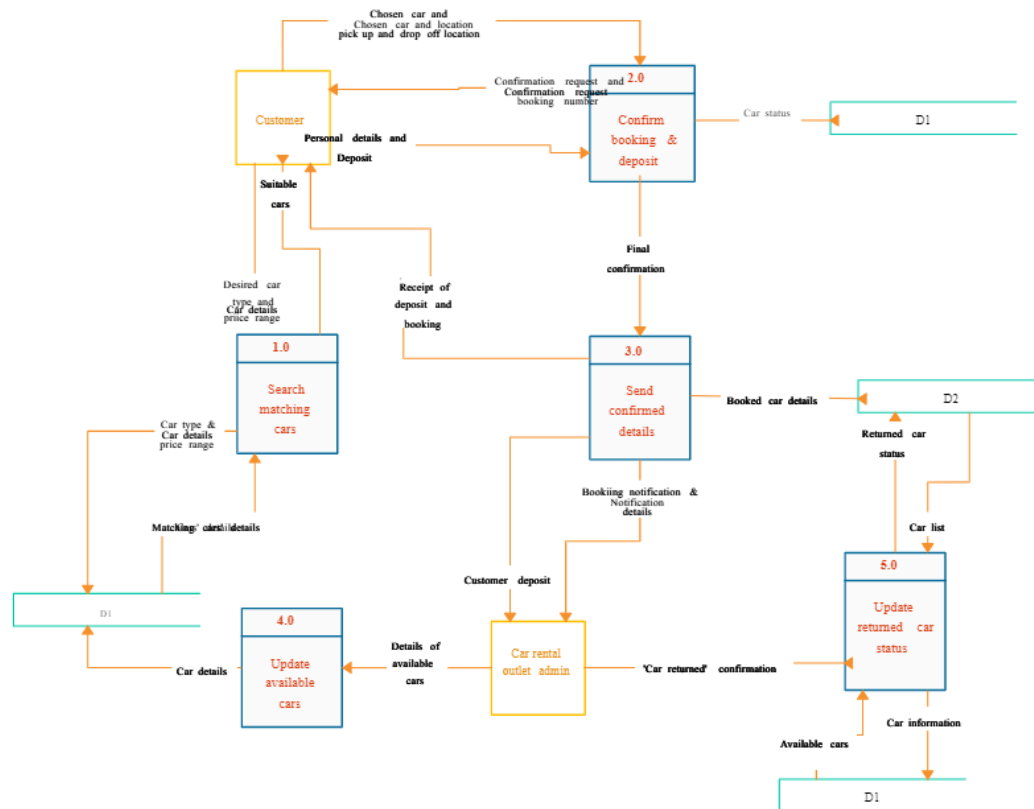
Security

Chapter 3: Software Design

3.1 Use Case:



3.2 DFD:



Bibliography:

NodeJS Docs: <https://nodejs.org/en/docs>

ReactJS Docs: <https://react.dev/learn>

MongoDB Docs: <https://www.mongodb.com/docs/>

Express.js Documentation. (2023). Retrieved from <https://expressjs.com/>

Tailwind CSS Documentation. (2023). Retrieved from <https://tailwindcss.com/docs>

Github link <https://github.com/ANJSACHAN/RideRight>

(All the related documents and video are present in the above GitHub link)