

Software Requirements Specification

for

Car Rental Website

Version 1.0 approved

Prepared By :

N Digvijay (PES2UG21CS310)
Nihal T M (PES2UG21CS333)
Nishanth D'Mello (PES2UG21CS343)
Rohith Hunashikatti (PES2UG21CS920)

PES University

23rd September 2023

Table of Contents

1. Introduction

- 1.1 Purpose
- 1.2 Intended Audience and Reading Suggestions
- 1.3 Product Scope

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 Operating Environment
- 2.4 Design and Implementation Constraints
- 2.5 Assumptions and Dependencies

3. External Interface Requirements

- 3.1 Communications Interfaces
- 3.2 Software Interfaces
- 3.3 User Interfaces

4. Analysis Models

- 4.1 Use-Case Diagram

5. System Features

- 5.1 User Registration and Authentication
- 5.2 Vehicle Search and Filtering

6. Other Non-Functional Requirements

- 6.1 Performance
- 6.2 Reliability
- 6.3 Security
- 6.4 Software Quality Attributes

7. Testing Requirements

- 7.1 Penetration Testing
- 7.2 Acceptance Testing

Introduction

Purpose

- The purpose of this document is to define the requirements for the development of a Car Rental Website, providing a comprehensive guide for stakeholders involved in the project.

Intended Audience and Reading Suggestions

- This document is intended for developers, designers, project managers, and other stakeholders involved in the development and implementation of the Car Rental Website. Readers are advised to familiarize themselves with the entire document for a complete understanding.

Product Scope

- The Car Rental Website aims to provide users with a user-friendly platform for renting vehicles. It includes features such as user registration, vehicle search and filtering, booking and reservation systems, and an administrative panel for managing vehicles and user accounts.

Overall Description

Product Perspective

- The Car Rental Website is a standalone system that interacts with users through a web interface. It is not dependent on external systems but may integrate with payment gateways for transaction processing.

Product Features

- User registration and authentication
- Vehicle search and filtering
- Booking and reservation system
- Admin panel for vehicle management
- Payment gateway integration

Operating Environment

- The Car Rental Website is a web-based application accessible through standard web browsers. It is compatible with major browsers such as Chrome, Firefox, and Safari.

Design and Implementation Constraints

- The system must comply with legal and regulatory requirements for online transactions and data privacy. It should be designed to accommodate at least 1000 simultaneous users without significant performance degradation.

Assumptions and Dependencies

- It is assumed that users have access to a stable internet connection and modern web browsers. The system depends on the availability and proper functioning of external payment gateways for transaction processing.

External Interface Requirements

Communication Interfaces

Frontend to Backend Communication:

Interface: RESTful API Description: React communicates with the Express.js backend using RESTful API endpoints for operations like fetching car data and making bookings.

Backend to Database Communication:

Interface: MongoDB Queries Description: Express.js communicates with the MongoDB database through MongoDB queries for tasks such as retrieving and manipulating data.

Software Interfaces

React (Frontend):

Interface: React is a JavaScript library for building user interfaces. You define components, manage state, and handle user interactions using JSX (JavaScript XML).

Node.js (Runtime Environment):

Interface: Node.js allows you to run JavaScript on the server side. You define server-side logic using the Node.js API.

Express.js (Backend):

Interface: Express.js provides a web application framework for Node.js. You define routes and handle HTTP requests using the Express API.

MongoDB (Database):

Interface: MongoDB provides a NoSQL database interface, and you interact with it using queries in the MongoDB Query Language (MQL).

User Interfaces

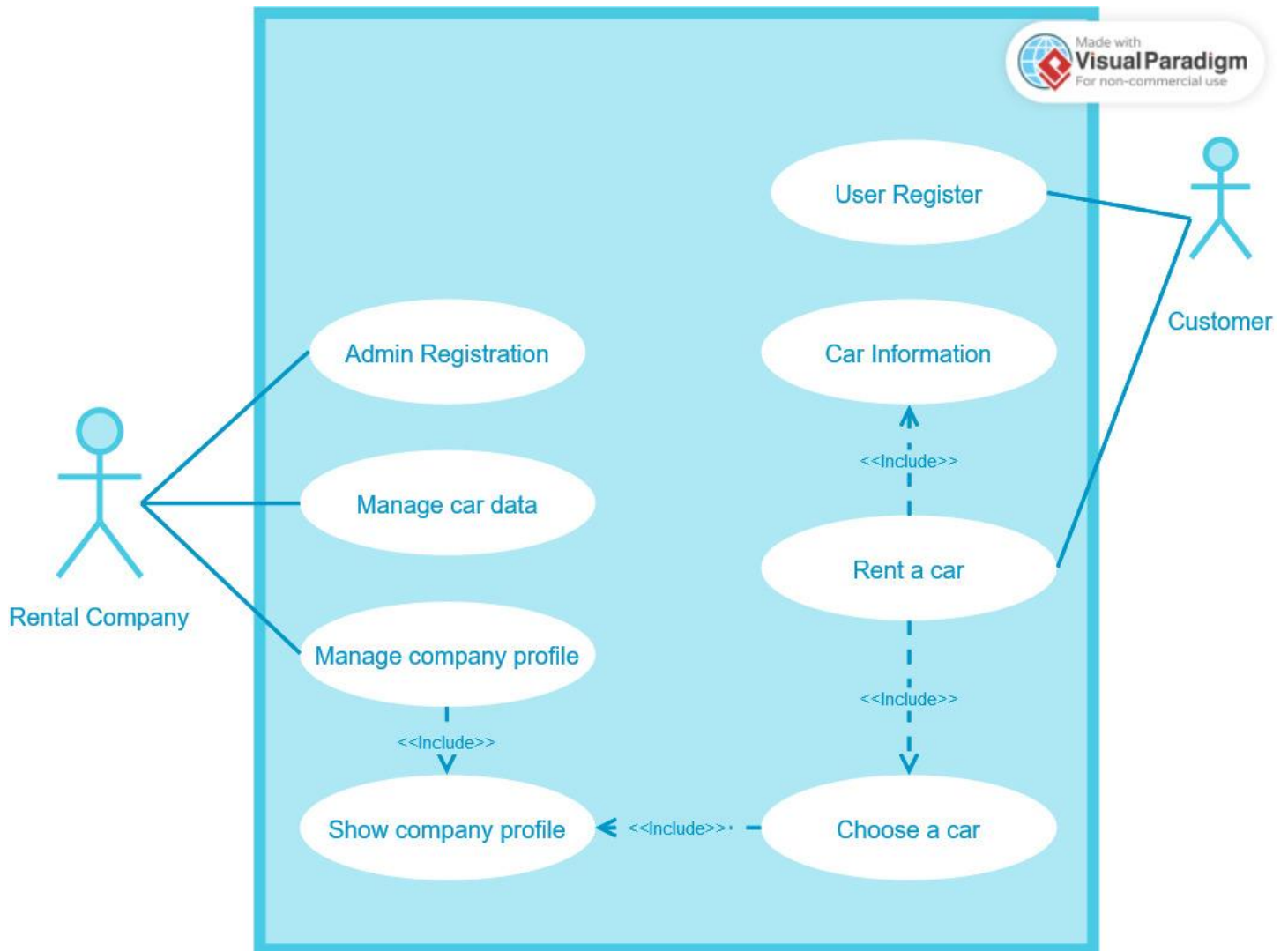
The website includes a Home page, Registration page, Contact Us page, About Us page, and a Payment Gateway.

Admin Interfaces

The admin has an Authentication page, Add Cars, Delete, and Update Page.

Analysis Models

Use Case Diagram



System Features

User Registration and Authentication

Users must be able to register with the website by providing the necessary information. Authentication mechanisms, such as email confirmation, should be implemented.

Vehicle Searching Filtering and Pagination

Users should be able to search for available vehicles based on various criteria, with advanced filtering options. Also, with the use of pagination,

Payment Gateway

Users can make payments securely on their rental car bookings

Non-Functional Requirements

Performance

- The system should handle a minimum of 1000 simultaneous users without significant performance degradation.

Reliability

- The system should ensure the security of user data, including personal information and payment details, through encryption.

Security

- The system should be available 99.9% of the time, with regular backups and recovery procedures in place.

Software Quality Attributes

- The website should be intuitive, user-friendly, and have a responsive design for various devices.

Testing Requirements

Penetration Testing

- **Purpose:** Identify and exploit vulnerabilities in the system.
- **Approach:** Simulate real-world attacks to assess the effectiveness of security defenses.

Acceptance Testing

- User acceptance testing will be performed to validate user interfaces.
- Stakeholders will review and approve the final product.