SOFTWARE REQUIREMENTS SPECIFICATION

For

Car Rental System

Prepared by:-
Sanjay.S 22EC139
Saran.K 22EC145
Shanjith.V 22EC150

1. Introduction

1.1.Purpose

The Car Rental System is designed to provide an efficient and user-friendly platform for managing the rental of cars. The primary goal is to streamline the process of renting and returning vehicles while maintaining a reliable and scalable system. The project encompasses both a Site Reliability Engineering (SRE) framework for system robustness and a Database Management System (DBMS) for data storage and retrieval. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams.

1.2.Document Conventions

- ➢ Entire document should be justified.
- ➢ Convention for Main title
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 14
- ➢ Convention for Sub title
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 12
- ➢ Convention for body
  - Font face: Times New Roman
  - Font Size: 12

1.3.Scope of Development Project

The car rental management system development project aims to create a user-friendly platform for easy vehicle reservations, transparent pricing, and efficient billing. Key features include a flexible reservation system, dynamic pricing, and prioritized user security with encryption and secure payment gateways. User roles for administrators, staff, and customers will ensure smooth management of users, vehicles, and reservations. The system provides detailed vehicle information, including maintenance history. Mobile responsiveness is crucial for accessibility. The project excludes hardware procurement, extensive third-party integrations, complex regulatory compliance, marketing, and formal user training. Constraints include a defined timeline, budget, specified technology (Java), and adherence to legal requirements. Deliverables include a prototype, comprehensive documentation, source code access, deployment plan, and potential training materials. Regular reviews allow for adaptability, and the project's flexibility supports easy implementation and future feature additions.

1.4.Definitions, Acronyms and Abbreviations

JAVA -> platform independence
SQL-> Structured query
Language ER-> Entity
Relationship
UML -> Unified Modeling Language
IDE-> Integrated Development
Environment SRS-> Software Requirement
Specification

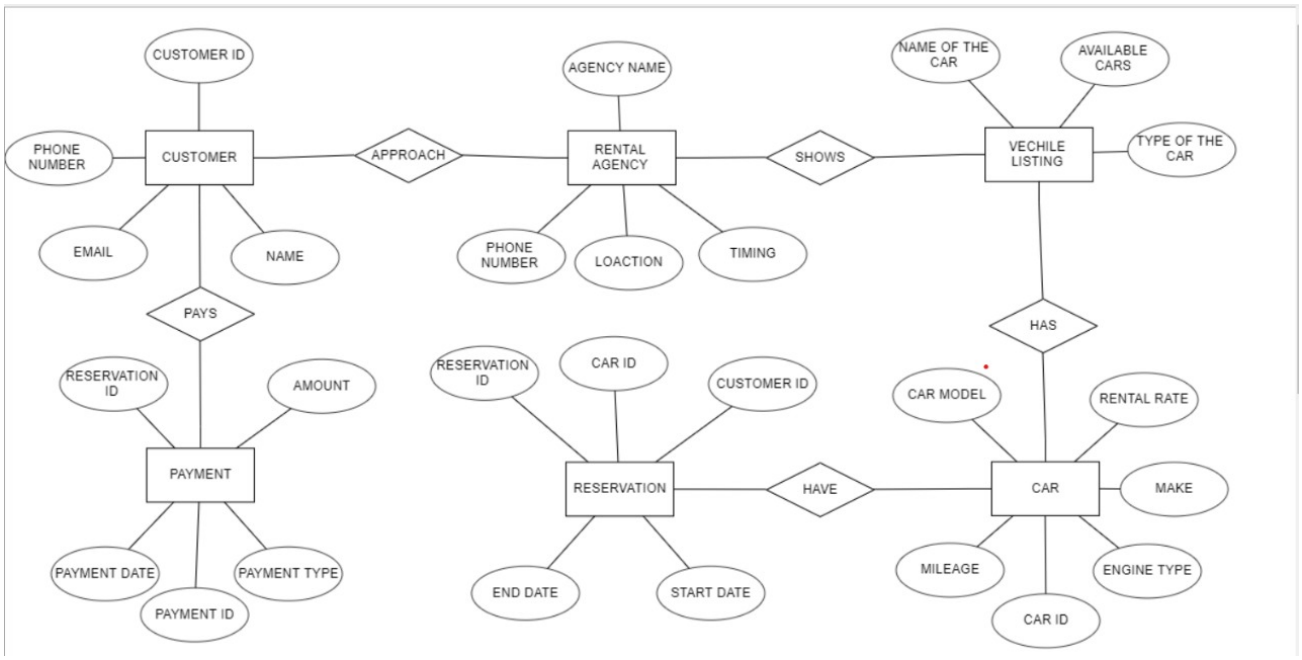
2. Overall Descriptions

2.1.Product Perspective

Use Case Diagram of Car Rental System.The users of the system can request issue/renew/return of Car for which they would have to follow certain criteria.

2.2.Product Function

Entity Relationship Diagram of Car Rental System



The Car Rental System is designed to streamline and automate the process of managing vehicle rentals, providing real-time online information about available cars and user details. The primary objective of this project is to replace manual workflows with a software solution that efficiently handles tasks such as car rentals, returns, fine calculation, and report generation, tailored to the specific needs of users. The system empowers the car rental administrator, acting as the main controller, to oversee members and effectively manage the fleet of vehicles.

Key functionalities include the ability to record and track car rentals, process returns, calculate and manage fines, and generate reports for comprehensive record-keeping. The administrator (car rental manager) utilizes the system to control user accounts, manage the fleet of cars, and retrieve member information from the database as needed. Members with valid accounts can access their personal account information through the system.

This Car Rental System aims to enhance operational efficiency, reduce manual intervention, and provide a user-friendly interface for both administrators and members. By leveraging technology, the system ensures accurate record-keeping, facilitates easy access to information, and contributes to an overall streamlined and efficient car rental management process.

2.3.User Classes and Characteristics

Administrators:

Characteristics:

Full system access and control.

Ability to add, edit, or remove vehicles from the catalog.

Manage user accounts, including staff and customer profiles.

Access to comprehensive reporting and analytics for business insights.

Staff Members:
Characteristics:
Restricted access compared to administrators.
Ability to process and confirm reservations.
Update vehicle availability status based on returns and maintenance.
Generate invoices and handle billing-related tasks.

Customers:
Characteristics:
User accounts with personal details for seamless reservations.
Browse and search vehicle listings.
Reserve vehicles for specific durations with the option to modify or cancel.
View past reservations, invoices, and provide feedback.
Access to support and help resources.

System Guests:
Characteristics:
Limited access without user accounts.
View basic information about available vehicles.
Explore general features of the application without the ability to reserve or modify bookings.

Administrator-Staff Interaction:
Administrators assign roles and permissions to staff members.
Staff members update vehicle statuses, handle reservations, and manage billing.
Regular communication channels for updates and coordination.

Staff-Customer Interaction:
Staff members assist customers with reservation details, modifications, and cancellations.
Handle customer inquiries, feedback, and support requests.
Provide necessary assistance during vehicle pickup and return.

Administrator-Customer Interaction:
Administrators monitor customer activity and feedback.
Address escalated customer issues and inquiries.
Oversee the overall customer experience and system performance.

Technical Proficiency:
Administrators and staff members require a higher level of technical proficiency to manage the system effectively.
Customers should be able to navigate the user interface with basic technical skills.

Frequency of Use:
Administrators and staff members use the system frequently for daily operations.
Customers may use the system periodically for reservations and account management.

Purpose of Use:
Administrators and staff use the system for administrative and managerial tasks.
Customers use the system primarily for vehicle reservations, billing, and accessing account information.

Security Awareness:
Administrators and staff members should have a heightened awareness of security measures, especially concerning user data and payment processing.
Customers should be aware of basic security practices when using the platform for transactions.

2.4.Operating Environment
The Car Rental System is designed to operate seamlessly within the Windows environment, offering a web-based platform accessible across popular browsers such as Microsoft Internet Explorer, Google Chrome, and Mozilla Firefox. Compatibility is ensured with Internet Explorer 6.0, and the majority of features are designed to work smoothly with Mozilla Firefox and Opera 7.0 or higher versions. The product requires only an internet connection, making it easily accessible for users to manage vehicle reservations and related tasks online. The web-based nature of the system enhances convenience, allowing users to interact with the application through their preferred browsers without the need for additional software installations.

2.5.Assumptions and Dependencies
The assumptions are:-

- ➢ The coding should be error free
- ➢ The system should be user-friendly so that it is easy to use for the users
- ➢ The system should have more storage capacity and provide fast access to the database
- ➢ The system should provide search facility and support quick transactions
- ➢ The Car Rental System is running 24 hours a day
- ➢ Users may access from any computer that has Internet browsing capabilities and an Internet connection
- ➢ Users must have their correct usernames and passwords to enter into their online accountsand do actions

The dependencies are:-

- ➢ The specific hardware and software due to which the product will be run
- ➢ On the basis of listing requirements and specification the project will be developed andrun
- ➢ The end users (admin) should have proper understanding of the product
- ➢ The system should have the general report stored
- ➢ The information of all the users must be stored in a database that is accessible by the Car Rental System
- ➢ Any update regarding the Car is to be recorded to the database and thedata entered should be correct

2.6.Requirement
Software Configuration:-
This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database.
Operating System: Windows NT, windows 98, Windows XP

Language: Java Runtime Environment, Net beans 7.0.1 (front end)
Database: MS SQL Server (back end)

Hardware Configuration:-
Processor: Pentium(R)Dual-core CPU Hard Disk: 40GB
RAM: 256 MB or more

2.7.Data Requirement

In the Car Rental System, users submit queries for actions like creating an account, choosing rental cars, and managing transactions. These queries are processed by the system's database. The system's output includes responses to user queries, providing details about their car rental activities. Users can receive information about their accounts, including transaction timestamps, dates, and a list of currently rented cars.
User queries involve actions such as account creation, car selection, and transaction updates. The system processes these queries, interacting with the database to provide accurate and real-time information.
The output not only addresses user queries but also gives users a clear view of their accounts. This includes transaction timestamps and a list of cars currently associated with their accounts. This seamless interaction between user queries and database responses ensures a straightforward and effective car rental experience.

3. External Interface Requirement

3.1.User Interface:

The application shall provide an intuitive and user-friendly web-based interface accessible via popular browsers, including Microsoft Internet Explorer, Google Chrome, and Mozilla Firefox.
Users should be able to easily navigate through vehicle listings, make reservations, view details, and access billing information.
The interface should be responsive and compatible with various screen sizes for an optimal user experience.

3.2.Compatibility:

The application must be compatible with Microsoft Internet Explorer 6.0 and above, Google Chrome, and Mozilla Firefox.
Key features should function seamlessly across different browsers, ensuring a consistent experience for users regardless of their preferred browser.

3.3.Internet Connectivity:

As an online product, the application requires a stable internet connection for users to access and utilize its functionalities.

3.4.Vehicle Listing:

External interfaces should facilitate the integration of real-time or periodically updated vehicle details, ensuring the accuracy of the vehicle listing.

3.5.Payment Gateways:

The application should support multiple payment methods, including credit/debit cards, digital wallets, and other online payment options.
Integration with secure and reputable payment gateways is essential to ensure the confidentiality and integrity of financial transactions.

3.6.Access Control:

Define external interfaces for user authentication and access control mechanisms.
Integration with external identity management systems may be necessary to manage user roles and permissions effectively.

3.7.Communication with External Systems:
External interfaces should be established for communication with third-party services or databases to retrieve and update relevant information, such as vehicle details, pricing, and availability.
3.8.Security Protocols:
External interfaces should adhere to industry-standard security protocols to safeguard user data, especially during payment processing.
Integration with external security services or tools may be necessary to enhance overall system security.
3.9.Error Handling and Logging:
External interfaces should support error handling mechanisms to provide informative messages to users.

4. System Features
The users of the system should be provided the surety that their account is secure. This is possible by providing:-

- ➢ User authentication and validation of members using their unique member ID
- ➢ Proper accountability which includes not allowing a member to see other member's account. Only administrator will see and manage all member accounts

5. Other Non-functional Requirements
5.1.Safety Requirement
The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost. Proper UPS/ inverter facility should be there in case of power supply failure.

5.2.Security Requirement
- ➢ System will use secured database
- ➢ Normal users can just read information but they cannot edit or modify anything exceptttheir personal and some other information.
- ➢ System will have different types of users and every user has access constraints
- ➢ Proper user authentication should be provided
- ➢ No one should be able to hack users' password
- ➢ There should be separate accounts for admin and members such that no member canaccess the database and only admin has the rights to update the database.

5.3.Requirement attributes
- ➢ There may be multiple admins creating the project, all of them will have the right tocreate changes to the system. But the members or other users cannot do changes
- ➢ The project should be open source
- ➢ The Quality of the database is maintained in such a way so that it can be very userfriendly to all the users of the database
- ➢ The user be able to easily download and install the system

5.4.Business Rules
A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data.This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

- ➢ The user be able to easily download and install the system

5.5.User Requirement

Users want a simple and user-friendly car rental system that allows easy reservations, clear pricing, and efficient billing. They need a flexible reservation system, dynamic pricing, and top-notch security with data encryption and secure payment options. Users also expect different access levels for administrators, staff, and customers to manage accounts, vehicles, and reservations effectively. The system should provide detailed vehicle information, be mobile-friendly, and offer the flexibility to add new features as needed. Users appreciate the choice of Java for development due to its performance, compatibility, available tools, and cost-effectiveness. The admin provides certain facilities to the users in the form of:-

- ➢ Backup and Recovery
- ➢ Forgot Password
- ➢ Data migration i.e. whenever user registers for the first time then the data is stored in theserver
- ➢ Data replication i.e. if the data is lost in one branch, it is still stored with the server
- ➢ Auto Recovery i.e. frequently auto saving the information
- ➢ Maintaining files i.e. File Organization
- ➢ The server must be maintained regularly and it has to be updated from time to time

6. Other Requirements

6.1.Data and Category Requirement

There are different categories of users namely teaching staff, Admin etc. Depending upon the category of user the access rights are decided.It means if the user is an administrator then he can be able to modify the data, delete, append etc. Similarly there will be different categories of cars available. According to the categories of cars their relevant data should be displayed. The categories and the data related to each category should be coded in the particular format.

6.2.Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

- ➢ Administrator: A login id representing a user with user administration privileges to thesoftware
- ➢ User: A general login id assigned to most users
- ➢ Client: Intended users for the software
- ➢ SQL: Structured Query Language; used to retrieve information from a database
- ➢ SQL Server: A server used to store data in an organized format
- ➢ Layer: Represents a section of the project
- ➢ User Interface Layer: The section of the assignment referring to what the user interactswith directly
- ➢ Application Logic Layer: The section of the assignment referring to the Web Server. This is where all computations are completed
- ➢ Data Storage Layer: The section of the assignment referring to where all data is recorded
- ➢ Use Case: A broad level diagram of the project showing a basic overview
- ➢ Class diagram: It is a type of static structure diagram that describes the structure of a

system by showing the system's cases, their attributes, and the relationships between the classes

- ➢ Interface: Something used to communicate across different mediums
- ➢ Unique Key: Used to differentiate entries in a database

6.4.Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes' structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalisation. The relationships are depicted using a role name and multiplicities. Here 'Customer', 'Car' and 'Payment' are the most important classes which are related to other classes.