SOFTWARE REQUIREMENTS SPECIFICATION

**For**

**Car Rental System**

**Prepared by:**

* Navin J
* Kavinkumar S
* Mahesh Kumar G

# Introduction

## Purpose

The main objective of this document is to illustrate the requirements of the project Car Rental System. The document gives the detailed description about the Car Rental System is an application that provides a platform for car rental companies to manage their operations and provide an easy and efficient way for customers to book cars online. The purpose of a Car Rental System is to provide a platform for both customers and car rental companies to manage car rentals in an efficient and user-friendly manner. The system aims to simplify the process of car rental by making it more user-friendly, efficient, and transparent. It allows customers to view available cars, book a car, make payments, and receive confirmation of their booking, all from the comfort of their home or office. This project describes the hardware and software interface requirements using ER diagrams and UML diagrams.

## 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

## Scope of Development Project

The Car Rental System is an application that provides a platform for car rental companies to manage their operations and provide an easy and efficient way for customers to book cars online.

The scope of this project encompasses the following functionalities:

* User Registration and Authentication: The system will allow new users to c and existing users to log in. It will securely store user information and handle password recovery.
* Vehicle Management: Administrators will be able to add, update, and remove vehicles from the system. They can also categorize vehicles based on type, model, and other parameters.
* Booking and Reservation: Users will be able to search for available cars, reserve a car for a specific time period, and cancel their reservation if needed.
* Payment Gateway Integration: The system will integrate with a secure payment gateway for processing payments. It will support multiple payment methods like credit card, debit card, and net banking.
* Rate and Review: Users will be able to rate and review their rental experience. These reviews will be visible to other users.
* Reporting and Analytics: The system will provide comprehensive reports on various aspects like booking history, vehicle utilization, revenue, etc., to help administrators make informed decisions.
* Notifications: The system will send notifications to users about booking confirmation, payment receipt, etc., via email or SMS.
* Customer Support: The system will include a customer support feature where users can raise queries or complaints.
* Maintenance and Support: Post-deployment, the project will also include regular maintenance and support to ensure smooth operation and to incorporate any future enhancements.

## 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

## References

* + - Books

 Software Requirements and Specifications: A Lexicon of Practice, Principles and Prejudices (ACM Press) by Michael Jackson

Software Requirements (Microsoft) Second EditionBy Karl E. Wiegers

Software Engineering: A Practitioner’s Approach Fifth Edition By Roger S. Pressman

* + - Websites

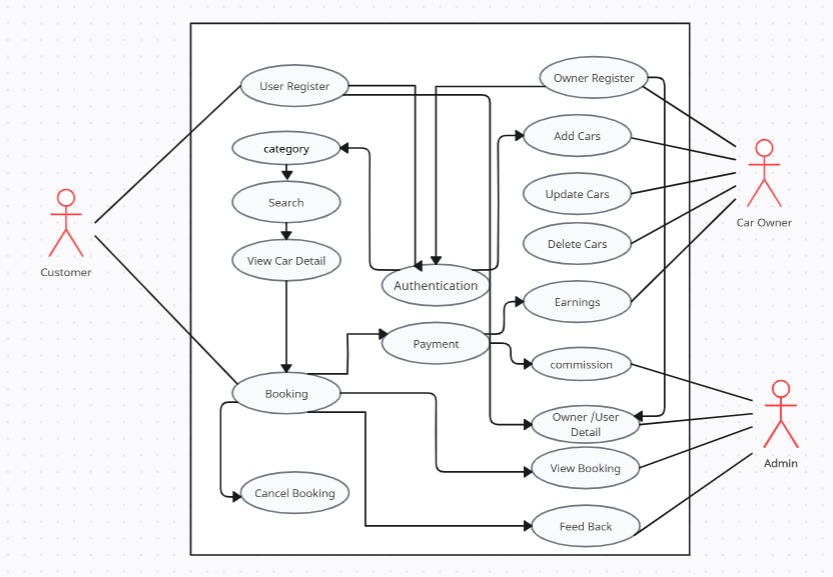
[**http://www.slideshare.net/**](http://www.slideshare.net/)

[**http://ebookily.net/doc/srs-library-management-system**](http://ebookily.net/doc/srs-library-management-system)

# Overall Descriptions

## Product Perspective

Use Case Diagram of Car Rental System



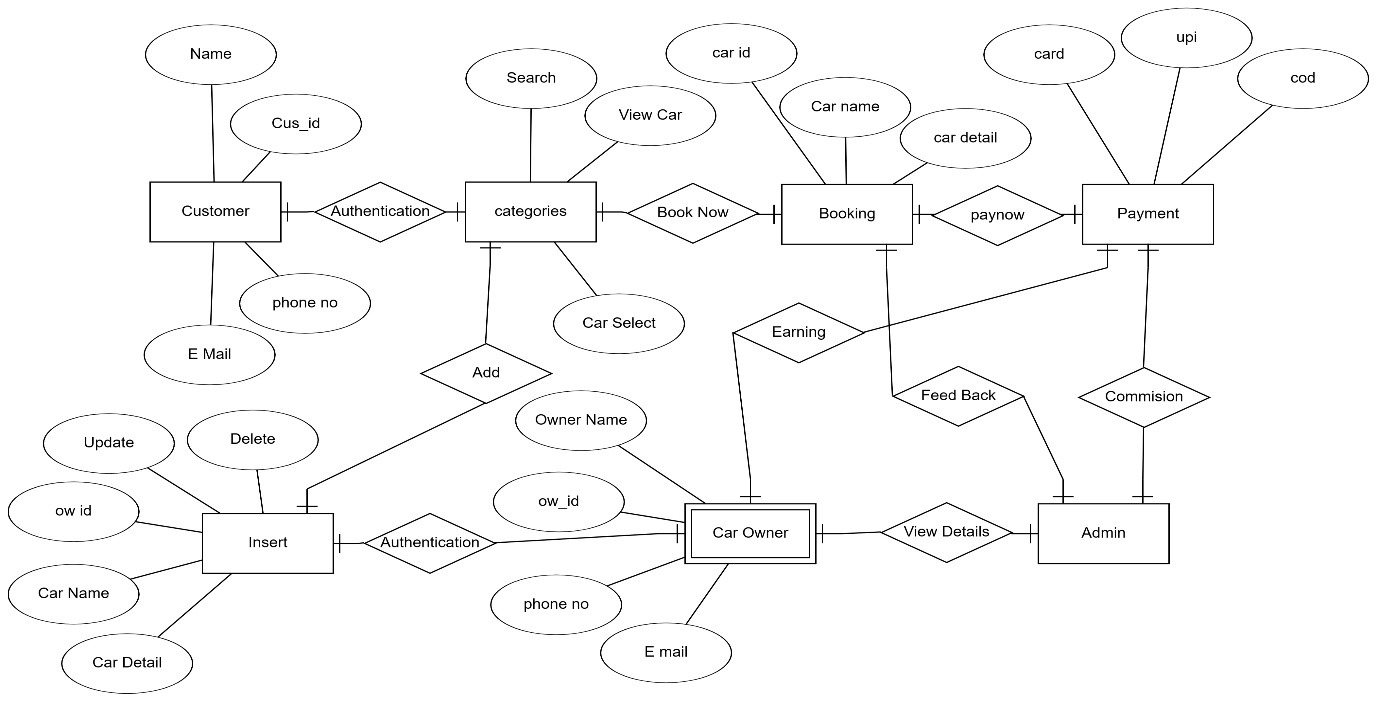
Customers can search for available cars based on their preferences, view car details, and reserve a car for a specified rental period. They can also cancel their reservation if necessary. This enables customers to easily find and rent a car that meets their needs and preferences, while also providing them with flexibility in case their plans change.

**Rental companies** can manage their car inventory by adding new cars, updating car details such as availability and rental price, or deleting cars from their inventory if necessary. This helps rental companies ensure that they have enough cars available for customers to rent and manage their car inventory effectively.

system. The users of the system can request issue/renew/return of books for which they would have to follow certain criteria.

## Product Function

Entity Relationship Diagram of Car Rental System.



The Entity Relationship Diagram (ERD) for a Car Rental System represents the system’s entities and their relationships.

Entities in the Car Rental System include:

* Customer: This entity represents the customers who will rent cars. It can have attributes like customer ID, name, contact details, etc.
* Owner: This entity represents the car owners. It can have attributes like owner ID, name, contact details, etc.
* Admin: This entity represents the system administrators. It can have attributes like admin ID, name, contact details, etc.
* Payment: This entity represents the payment details. It can have attributes like payment ID, rental ID, payment amount, payment date, etc.

Relationships between these entities could be:

* + - A customer can rent one or more cars.
    - A car can be rented by one or more customers.
    - An owner can own one or more cars.
    - A car is owned by one owner.
    - An admin can manage multiple customers, owners, and cars.
    - A rental is associated with one customer, one car, and one payment.

## User Classes and Characteristics

In a car rental system, there are typically three main user classes, each with their own characteristics:

1. Customers: These are the users who will be renting the cars. They need to be able to view available cars, make a booking, and make a payment. They may also need to be able to view their booking history and cancel bookings.

Characteristics:

- Need for a simple, intuitive interface.

- Desire for a wide range of available cars.

- Need for secure payment processing.

2. Administrators: These are the users who manage the car rental system. They need to be able to add, update, awnd remove cars from the system. They also need to manage bookings and view reports on usage and revenue.

Characteristics:

- Need for comprehensive management features.

- Desire for detailed reporting and analytics.

- Need for efficient handling of customer bookings and cancellations.

3. Car Owners: In some car rental systems, individual car owners can list their cars for rent. They need to be able to add their cars to the system, manage availability, and view bookings.

Characteristics:

- Need for easy listing and management of their cars.

- Desire for control over when their car is available for rent.

- Need for visibility into when their car is booked.

## Operating Environment

The product will be operating in windows environment. The Library Management System is a website and shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer, Google Chrome, and Mozilla Firefox. Also it will be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. The only requirement to use this online product would be the internet connection.

The hardware configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

## Assumptions and Dependencies

The assumptions are: -

* Internet Connectivity: It is assumed that users (both customers and administrators) have stable internet connectivity to access the system.
* Tech-Savvy Users: It is assumed that users are familiar with basic online operations like registration, login, online payment, etc.
* Availability of Cars: It is assumed that the listed cars are available for rent and are in good condition.
* Payment Gateway: It is assumed that the integrated payment gateway is secure and reliable for processing payments.
* Accurate Information: It is assumed that users provide accurate information during registration and booking.

The dependencies are: -

* Web Development Technologies: The system’s development depends on web development technologies like HTML, CSS, JavaScript, etc., for front-end development and suitable back-end technology.
* Database Management System: The system depends on a reliable Database Management System for storing and retrieving data.
* Payment Gateway: The system depends on a third-party payment gateway for processing payments.
* Hosting Services: The system depends on a reliable hosting service to ensure it is always accessible to users.
* Maintenance and Support: The system’s smooth operation depends on regular maintenance and support to fix bugs and implement updates.

## 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

## Data Requirement

The data requirements for a car rental system typically include:

* User Data: This includes information about the users such as name, contact details, login credentials, and booking history.
* Vehicle Data: This includes information about the vehicles available for rent such as vehicle ID, type, model, availability status, and rental price.
* Booking Data: This includes information about the bookings such as booking ID, customer ID, vehicle ID, rental start date, rental end date, and payment status.
* Payment Data: This includes information about the payments such as payment ID, booking ID, payment amount, payment method, and payment date.
* Review Data: This includes feedback and ratings provided by the customers.
* Administrative Data: This includes information required by administrators to manage the system such as reports on vehicle utilization, revenue, customer feedback, etc.

# External Interface Requirement

External Interface Requirements for a Car Rental System typically include:

* User Interface (UI): The system should have an intuitive and user-friendly interface for both customers and administrators. This includes pages for vehicle selection, booking, payment, and administration panel.
* Application Programming Interface (API): The system may need to interact with external systems such as payment gateways, GPS tracking systems, or customer relationship management (CRM) systems. This requires well-defined APIs to ensure smooth data exchange.
* Database Interface: The system will interact with a database to store and retrieve data. This requires a database interface that supports all necessary data operations.
* Hardware Interface: If the system is to be used on different devices like computers, tablets, or smartphones, it should be compatible with the hardware of these devices.
* Software Interface: The system should be compatible with different operating systems and browsers to ensure that it can be accessed by a wide range of users.
* Communication Interface: If the system needs to send notifications to users via email or SMS, it will need to interface with email servers or SMS gateways.

# System Features:

The Car Rental System can have several key features to ensure it meets the needs of both the car rental company and its customers:

* User Registration and Login: Users should be able to register and create an account in the system. They should also be able to login to their account to access the services.
* Vehicle Listing: The system should list all the available vehicles for rent. The listing can include details like vehicle type, model, price, and availability status.
* Booking System: Users should be able to book a vehicle for a specific duration. The system should update the availability status of the vehicle once it’s booked.
* Payment System: The system should integrate with a secure payment gateway to process payments. It should support multiple payment methods like credit card, debit card, and net banking.
* Rating and Review: Users should be able to rate and review their rental experience. This can help the company improve their services and can also assist other users in making informed decisions.
* Help and Support: The system should provide a help and support feature where users can find answers to common questions or contact customer support for assistance.

# Other Non-functional Requirements

## Performance Requirement

The Performance requirements for a Car Rental System typically include:

* Response Time: The system should respond quickly to user requests. For example, the time taken to load vehicle details, process a booking, or make a payment should be minimal to ensure a smooth user experience.
* Availability: The system should be available 24/7 to allow users to make bookings at any time. This requires reliable hosting and regular system maintenance.
* Scalability: The system should be able to handle an increasing number of users and bookings without performance degradation. This might involve using scalable cloud hosting services and optimizing the database design.
* Reliability: The system should function correctly and consistently, with minimal downtime or errors. This involves thorough testing and quality assurance processes.
* Data Integrity: The system should ensure the accuracy and consistency of data. For example, it should prevent double bookings of the same vehicle.
* Security: The system should securely handle user data and payments. This involves using secure protocols, encrypting sensitive data, and regularly updating the system to address security vulnerabilities.

## Safety Requirement

Safety requirements for a Car Rental System typically include:

* User Authentication: The system should verify the identity of users to prevent unauthorized access. This could involve using secure login mechanisms and two-factor authentication.
* Backup and Recovery: The system should regularly back up data to prevent data loss. It should also have a recovery mechanism in case of system failures.
* Legal Compliance: The system should comply with all relevant laws and regulations, including data protection and privacy laws.

## Security Requirement

* + - The system will use secured database.
    - Normal users can just read information, but they cannot edit or modify anything except their personal and some other information.
    - System will have different types of users and every user will have access constraints.
    - Proper user authentication should be provided.
    - No one should be able to hack users’ password.
    - There should be separate accounts for Car Owner and Customer such that no member can access the database and only admin has the right to update the database.

## Requirement attributes

## Requirement attributes provide additional information about each requirement in a Car Rental System. Here are some common requirement attributes:

## ID: A unique identifier for each requirement. This helps in tracking and referencing the requirement.

## Name: A short, descriptive name for the requirement.

## Description: A detailed description of the requirement, explaining what it entails and how it contributes to the system.

## Priority: The importance of the requirement in relation to other requirements. This helps in deciding the order of implementation.

## Status: The current status of the requirement, such as ‘new’, ‘in progress’, ‘implemented’, or ‘verified’.

## Owner: The person or team responsible for implementing the requirement.

## Complexity: An estimate of the effort needed to implement the requirement.

## 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.

## User Requirement

The users of the system are members and Employee of the Car Rental System who act as administrator to maintain the system. The members are assumed to have basic knowledge of computers and internet browsing. The administrators of the system should have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper user interface, user manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

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 the server.
    - 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

# Other Requirements

## Data and Category Requirement:

Data Requirements refer to the specific types of data the system needs to function. For a Car Rental System, this could include data about users (like name, contact details), vehicles (like vehicle ID, type, model), bookings (like booking ID, rental dates), payments (like payment ID, amount), and so on. Understanding data requirements is crucial for designing the system’s database and for implementing features that handle this data.

Category Requirements refer to how this data can be grouped or categorized. For example, users might be categorized into ‘customers’ and ‘administrators’, each with different access rights and functionalities. Vehicles might be categorized based on type, brand, or model. Understanding category requirements is important for implementing features that allow users to filter and sort data, and for designing user interfaces that present data in a structured and meaningful way.

## Appendix:

An appendix for a Car Rental System typically includes additional information or resources that support the main content of the system's documentation. This could include:

Glossary: A list of terms and their definitions used in the system¹.

* Data Dictionaries:Detailed descriptions of the data elements used in the system, including their meanings, relationships, and source.
* User Manuals: Detailed guides on how to use the system, often with step-by-step instructions.
* Technical Specifications: Detailed information about the technical aspects of the system, such as software and hardware requirements.
* API Documentation: If the system provides APIs for integration with other systems, detailed API documentation would be included.
* Test Cases and Results: Detailed information about the test cases used to validate the system and their results.
* Maintenance Guides: Information on how to maintain and troubleshoot the system.
* Legal and Compliance Information: Any legal information or details about compliance with standards or regulations.

## 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 the software
    - 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 interacts with 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

## 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 work. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicity. Here ‘Customer’, ‘Car Owner’ and ‘Admin’ are the most important classes which are related to other classes.

