**Software Requirements Specification (SRS) for \*\*Car Showroom Management System\*\* using MySQL**

Table of Contents

1. \*\*Introduction\*\*

1. Purpose

2. Scope

3. Definitions, Acronyms, and Abbreviations

4. Overview

2. \*\*Overall Description\*\*

1. Product Perspective

2. Product Features

3. User Classes and Characteristics

4. Operating Environment

5. Design and Implementation Constraints

6. Assumptions and Dependencies

3. \*\*System Features\*\*

1. User Management

2. Car Inventory Management

3. Sales Management

4. Customer Management

5. Reporting and Analytics

4. \*\*External Interface Requirements\*\*

1. User Interfaces

2. Hardware Interfaces

3. Software Interfaces

4. Communication Interfaces

5. \*\*System Requirements\*\*

1. Functional Requirements

2. Non-Functional Requirements

6. \*\*Database Design\*\*

1. Database Tables

2. ER Diagram

---

### 1. \*\*Introduction\*\*

#### 1.1 Purpose

This document defines the software requirements for the Car Showroom Management System using MySQL as the database system. The system aims to automate the management of a car showroom, including inventory, customer details, sales, and reporting. It serves to reduce manual efforts, enhance data accuracy, and improve business efficiency.

#### 1.2 Scope

The Car Showroom Management System will manage various aspects of a car showroom, including:

- Car inventory

- Customer information

- Sales transactions

- Employee management

- Reporting for sales and inventory analysis

This system will be designed using MySQL for the backend, ensuring data integrity and security. It will be accessible by authorized users (sales representatives, managers, and administrators).

#### 1.3 Definitions, Acronyms, and Abbreviations

- \*\*SRS\*\*: Software Requirements Specification

- \*\*DBMS\*\*: Database Management System

- \*\*SQL\*\*: Structured Query Language

#### 1.4 Overview

This SRS document outlines the detailed requirements for the Car Showroom Management System. It includes the functional and non-functional requirements, as well as an overview of the database structure.

---

### 2. \*\*Overall Description\*\*

#### 2.1 Product Perspective

The system will be a standalone software designed to manage various operations in a car showroom. It will have a MySQL-based backend to store and retrieve data efficiently. The front-end can be web-based or desktop-based, interacting with the database to perform CRUD (Create, Read, Update, Delete) operations.

#### 2.2 Product Features

The Car Showroom Management System will offer the following features:

- \*\*Car Inventory Management\*\*: Add, update, and remove car records, track the availability of cars, and manage car specifications.

- \*\*Customer Management\*\*: Manage customer information, including contact details and purchase history.

- \*\*Sales Management\*\*: Record and track sales transactions, including generating invoices.

- \*\*User Management\*\*: Manage user roles (e.g., admin, sales representative) and access rights.

- \*\*Reporting\*\*: Generate reports on sales, inventory, and customer activity.

#### 2.3 User Classes and Characteristics

- \*\*Administrator\*\*: Responsible for managing the entire system, including users, cars, and reports.

- \*\*Sales Representative\*\*: Responsible for managing customer interactions and sales transactions.

- \*\*Manager\*\*: Access to sales reports and inventory analysis.

#### 2.4 Operating Environment

- \*\*Operating System\*\*: Windows, Linux, or macOS

- \*\*Database\*\*: MySQL

- \*\*Development Tools\*\*: Any tool that supports MySQL (e.g., PHPMyAdmin, MySQL Workbench)

- \*\*Server\*\*: Apache/Nginx (for web-based systems)

#### 2.5 Design and Implementation Constraints

- The system will be implemented using MySQL as the DBMS, ensuring high performance and scalability.

- The system will be developed with the ability to handle concurrent user access.

- Data integrity and security must be ensured.

#### 2.6 Assumptions and Dependencies

- The users of the system will have basic knowledge of operating the software.

- Internet connectivity may be required for the web-based application.

---

### 3. \*\*System Features\*\*

#### 3.1 User Management

- \*\*Description\*\*: This module allows the administrator to manage user accounts and their roles.

- \*\*Functionalities\*\*:

- Add, update, and delete users.

- Assign roles (admin, sales representative, manager).

#### 3.2 Car Inventory Management

- \*\*Description\*\*: This module manages the inventory of cars in the showroom.

- \*\*Functionalities\*\*:

- Add new cars with details such as model, price, color, manufacturer.

- Update car details.

- Track car availability (sold, in stock).

#### 3.3 Sales Management

- \*\*Description\*\*: This module handles the sales process.

- \*\*Functionalities\*\*:

- Record sales transactions with customer details.

- Generate invoices.

- Update car status upon sale.

#### 3.4 Customer Management

- \*\*Description\*\*: This module manages customer details.

- \*\*Functionalities\*\*:

- Add and update customer contact information.

- View customer purchase history.

#### 3.5 Reporting and Analytics

- \*\*Description\*\*: This module generates reports for sales and inventory analysis.

- \*\*Functionalities\*\*:

- Generate daily, monthly, and yearly sales reports.

- View car inventory reports.

- Analyze customer purchase trends.

---

### 4. \*\*External Interface Requirements\*\*

#### 4.1 User Interfaces

The user interface will be designed to ensure ease of use:

- A dashboard for users to access relevant features based on their role.

- Form-based inputs for adding and updating data (cars, customers, sales).

- Table views for viewing data.

#### 4.2 Hardware Interfaces

The system will interface with hardware devices such as:

- Printers (for invoice generation).

- Bar code scanners (optional for car stock management).

#### 4.3 Software Interfaces

- \*\*MySQL Database\*\*: The backend database will store all data related to cars, customers, sales, and users.

- \*\*Web Application\*\*: A web-based interface (e.g., built using PHP, Java, or Python) will connect to the MySQL database for interacting with the system.

#### 4.4 Communication Interfaces

- \*\*HTTP/HTTPS\*\*: If a web-based system, the communication between client and server will occur over HTTP/HTTPS protocols.

---

### 5. \*\*System Requirements\*\*

#### 5.1 Functional Requirements

1. \*\*User Login/Registration\*\*:

- Users must be able to log in using their credentials.

- New users can be registered by the administrator.

2. \*\*Manage Cars\*\*:

- The system should allow users to add, update, and delete car records.

- Users can search for cars by model, manufacturer, or availability.

3. \*\*Manage Sales\*\*:

- Sales representatives can record new sales transactions.

- The system generates and stores an invoice for each sale.

4. \*\*Manage Customers\*\*:

- Users can add or update customer details.

- Users can view customer purchase history.

5. \*\*Generate Reports\*\*:

- The system can generate sales and inventory reports.

#### 5.2 Non-Functional Requirements

1. \*\*Performance\*\*: The system should handle multiple user requests efficiently without slowing down.

2. \*\*Security\*\*: Only authorized users should be able to access the system. Sensitive information such as user credentials must be encrypted.

3. \*\*Usability\*\*: The user interface should be simple and intuitive.

4. \*\*Reliability\*\*: The system should be available 99% of the time.

5. \*\*Scalability\*\*: The system should be scalable to handle increased data as the showroom grows.

---

### 6. \*\*Database Design\*\*

#### 6.1 Database Tables

1. \*\*Users Table\*\*:

- `user\_id` (Primary Key)

- `username`

- `password`

- `role` (admin, sales, manager)

2. \*\*Cars Table\*\*:

- `car\_id` (Primary Key)

- `model`

- `manufacturer`

- `price`

- `color`

- `status` (in stock, sold)

3. \*\*Customers Table\*\*:

- `customer\_id` (Primary Key)

- `name`

- `email`

- `phone`

- `address`

4. \*\*Sales Table\*\*:

- `sale\_id` (Primary Key)

- `car\_id` (Foreign Key)

- `customer\_id` (Foreign Key)

- `sale\_date`

- `total\_price`

#### 6.2 ER Diagram

The ER Diagram shows the relationships between entities such as Users, Cars, Customers, and Sales.

---

This concludes the Software Requirements Specification (SRS) for the Car Showroom Management System using MySQL.