**Software Requirements**

**Specification**

**for**

**“Car Dealership Management System”**

**Version 1.0**

Prepared by :

Sudarshan Patil 23378

Madhav Jadhav 23392

Hansraj Pawar 23387

Tushar Jadhav 23390

**Institute: Pune institute of computer Technology**

**Date**: 22-05-2023

**Table of Contents**

**Table of Contents**

1. [Introduction 2](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250020)
   1. [Purpose 2](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250019)
   2. [Intended Audience and Reading Suggestions 2](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250018)
   3. [Project Scope 3](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250017)
2. [Overall Description 4](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250016)
   1. [Product Perspective 4](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250015)
   2. [Product Features 4](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250014)
   3. [User Characteristics 4](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250013)
   4. [Principle Actors 4](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250012)
   5. General Const 5
   6. [Assumptions and Dependencies 5](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250011)
3. [Requirement Analysis 6](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250010)
   1. Funcitonal requirements 6
   2. [Non functional requirements 7](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250009)
   3. [Performance requirements 7](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250008)
   4. [Technical Issues 8](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250007)
4. [External Interface Requirements 9](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250006)
   1. [User Interfaces 9](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250005)
   2. [Hardware Interfaces 9](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250004)
   3. [Software Interfaces 10](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250003)
   4. [Communications Interfaces 10](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250002)
5. System Desgin Specificaitons 11
   1. System architecture 12
   2. [Use case diagram 14](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250001)
   3. [ER diagram 16](file:///C:\Users\MADHAV\Downloads\h11_ecommerce_DML_SRS.doc.docx#_TOC_250000)

# Introduction

### Purpose:

The purpose of a car dealership management system is to streamline and simplify the operations and processes within a car dealership. It is designed to help dealership owners and managers effectively manage their inventory, sales, customer information, and overall business operations.

The system serves as a centralized platform that enables dealership staff to track and manage the entire lifecycle of a vehicle, from acquisition to sale. It provides features such as inventory management, where dealers can easily monitor their stock of vehicles, track their specifications, and update their availability status.

Moreover, the system facilitates the sales process by capturing customer information, managing test drives, and assisting in the negotiation and documentation of sales transactions. It helps sales representatives keep track of potential customers, follow up on leads, and maintain a database of client information for future marketing and customer relationship management purposes.

### Intended Audience and Reading Suggestions

The intended audience for an SRS document for an car dealership system is the following:

* + - Project Stakeholders
    - Development Team
    - Quality Assurance/Testers
    - System Users
    - Project Managers

The SRS document should be written in a clear and concise manner that is easy to understand by all of these stakeholders. It should be well-organized and easy to navigate. The document should be complete and accurate, and it should be updated as needed.

Here are some reading suggestions for the SRS document:

* + - Clearly define the objectives and scope of the car dealership system to establish a common understanding among stakeholders.
    - Document functional requirements, specifying the system's capabilities such as inventory management, sales processes, customer relationship management, and financial transactions.
    - Include non-functional requirements, such as performance, security, scalability, and usability criteria, to ensure the system meets the desired quality attributes.
    - Use appropriate diagrams and models, such as use case diagrams, activity diagrams, and data flow diagrams, to illustrate the system's behavior and interactions.
    - Identify system actors and describe their roles and responsibilities within the car dealership system.
    - Define interfaces with external systems or third-party applications that the car dealership system needs to integrate with, such as accounting software or online marketing platforms.
    - Include constraints and assumptions, such as hardware and software dependencies, regulatory compliance, and budget limitations.

### Project Scope

* Project Goal: To create an application that will allow users to create account and order cars from the application , it also allows users to book car for test drives.
* Project Scope: The application will be accessible from any web browser and will be optimized for mobile devices. It will have the following features:
  + A user-friendly interface
  + A secure checkout processes
  + Easy account creation.
* Project Deliverables: The application will be delivered in two phases:
  + Phase 1: The application will be developed and tested.
  + Phase 2: The application will be deployed and launched.
* Project Success Criteria: The project will be a success if the following criteria are met:
  + The application meets all of the functional and non-functional requirements.
  + The application is delivered on time and within budget.
  + Users and clients give a positive feedback.
  + All the Functional and Non-Functional Requirements work properly without any problem.

# Overall Description

The car dealership management system is a comprehensive software solution designed to streamline the operations of a car dealership. It allows users to create accounts and order cars effortlessly, enhancing the overall sales process and customer experience. Customers can easily browse the inventory, access detailed information about cars, and place orders based on their preferences. The system enables dealership administrators to manage and update the inventory, while sales representatives can track order progress and communicate with customers. With integrated financial management capabilities, the system facilitates financing options and secure payment processing. By providing reporting and analytics features, the system helps dealership managers make informed decisions based on sales performance and customer data. Overall, the car dealership management system optimizes efficiency, improves customer satisfaction, and enhances the management of the dealership's operations.

* 1. **Product Perspective**

The car dealership management system is a standalone software product designed to function independently and integrate with existing dealership infrastructure, such as databases and external systems.

It interacts with various stakeholders, including dealership staff, customers, and external financial institutions, to facilitate seamless management of car sales and related processes.

* 1. **Product Features**

The proposed system should support following usecases:

* + - User-Friendly Interface: Provide an intuitive and user-friendly interface for dealership staff and customers, making it easy to navigate, search for vehicles, and perform tasks efficiently.
    - Security and Access Control: Implement robust security measures to protect sensitive data and ensure authorized access to the system. Control user permissions and roles for different levels of access.
    - Customer Relationship Management (CRM): Maintain a comprehensive customer database, track interactions, and manage leads. Provide personalized customer service and facilitate follow-ups for potential sales opportunities.
    - Inventory Management: Efficiently track and manage the dealership's inventory, including adding new vehicles, updating specifications, and monitoring availability.
    - Sales and Order Processing: Enable customers to browse and select cars, place orders, and track the progress of their purchases. Streamline sales processes, including negotiation, documentation, and invoicing.
  1. **User Characteristics**

User should be familiar with terms like login, register, buy now, etc.

* 1. **Principle Actors**

Two principle actors are customers and administrator.

* 1. **General Constraints**
     + The application must be accessible from any web browser.
     + The application must be secure.
     + The application must be scalable.
     + The application must be reliable.
     + The database should be secure.
     + The application should be bug free.
  2. **Assumptions and Dependencies**
     + Availability of stable internet connectivity for online functionality and data synchronization.
     + Availability of compatible hardware and software infrastructure to support the system.
     + Customers will provide accurate and up-to-date information during the account creation and ordering process.
     + Adequate security measures will be implemented to protect sensitive dealership and customer data.
     + Continuous updates and compatibility with evolving technologies and software frameworks.

# Requirement Analysis

Requirement gathering is an essential part of the ecommerce development process. By taking the time to gather and document the requirements, you can ensure that the system or product you create meets the needs of all stakeholders and provides a positive user experience.

* 1. **Functional requirements:**
     1. **Registration**

If a customer want’s to purchase a car then he should first register and create a account.

* + 1. **Login**

Customer logins to the system by entering valid user id and password for the purchasing.

* + 1. **Order**

The customer after registration and login can order cars and also check other cars.

* + 1. **Logout**

If user has finished checking the website then he can logout of it.

* + 1. **Some additional Functional requirements can be:**
       - Vehicle Search and Listing: Enable customers to search and browse available vehicles based on criteria such as make, model, price range, and location.
       - Sales Process Management: Support the entire sales process, including capturing customer information, generating sales quotes, managing negotiations, and tracking sales progress.
       - Customer Relationship Management (CRM): Maintain a database of customer information, including contact details, preferences, and purchase history, to facilitate personalized interactions and customer retention.

* 1. **Non-functional requirements:**
     + - Performance: The system should handle a large number of concurrent users and process transactions quickly.
       - Security: The system should implement robust security measures to protect sensitive data and prevent unauthorized access.
       - Usability: The system should have an intuitive and user-friendly interface to ensure ease of use for dealership staff and customers.
       - Reliability: The system should operate consistently without significant downtime or disruptions.
       - Scalability: The system should be scalable to accommodate the dealership's growth and increased user load.
       - Data Integrity: The system should maintain the accuracy and integrity of data throughout its lifecycle.
  2. **Performance Requirements:**
     + - Response Time: The system should respond to user interactions, such as searching for vehicles or updating inventory, within a reasonable time frame to ensure a smooth user experience.
       - Concurrent User Capacity: The system should be able to handle a significant number of concurrent users without significant degradation in performance, allowing multiple users to access and interact with the system simultaneously.
       - System Availability: The system should have high availability, minimizing downtime for maintenance or upgrades, and ensuring that it is accessible to users during regular business hours.
  3. **Technical issues:**

This system will work on client-server architecture. It will require an internet server and which will be able to run PHP application. The system communicates with the database to store and retrieve the data from the database.

# External Interface Requirements

* 1. **User Interfaces**

The user interface (UI) is the way that users interact with the car dealership management system. The UI should be simple, easy to use and understand, and it should be designed to meet the needs of the target audience. The UI should be visually appealing and engaging, and it should be responsive to user input.

Some of the key elements of the UI include:

* + - Homepage: the homepage consists of navigation bar that allows the user to navigate through different pages.
    - Company: the company names act as a link between different cars that are available of that company.
    - Registration page: this page allows the user to register to the system so that he can buy cars.
    - Login: the user should login so that he can place car order.
  1. **Hardware Interfaces**

Hardware interfaces refer to the physical connections and interactions between a software system and the hardware components it interacts with. These interfaces facilitate the communication, data transfer, and functionality between the software and various hardware devices.

In the context of a car dealership management system, hardware interfaces involve the integration and interaction with specific hardware components such as servers, workstations, printers, scanners, payment terminals, mobile devices, and networking equipment. These interfaces allow the software system to interact with and control the hardware, enabling functionalities like data entry, printing documents, scanning barcodes, processing payments, and accessing the system from mobile devices.

* 1. **Software Interfaces**

software interfaces play a crucial role in ensuring seamless integration and interoperability between different software components within the system, as well as with external systems or services.

Some examples of software interfaces in a car dealership management system may include:

* Database Interfaces: These interfaces allow the software system to interact with the underlying database management system (DBMS) to retrieve, store, and update data. Common database interfaces include SQL (Structured Query Language) for relational databases or NoSQL interfaces for non-relational databases.
* User Interfaces (UI): User interfaces define how users interact with the software system. This includes the graphical user interface (GUI) for desktop or web applications, command-line interfaces

(CLI), or mobile application interfaces.

* 1. **Communications Interfaces**

Communications interfaces refer to the methods and protocols used for communication between different systems, devices, or components within a car dealership management system. These interfaces enable the exchange of data and information across different platforms, networks, or communication channels.

Other communications interfaces that may be used with the ecommerce system include:

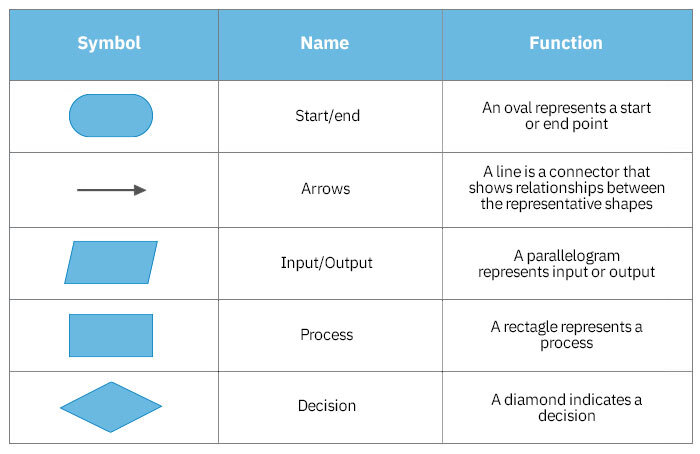
In the context of a car dealership management system, communications interfaces play a crucial role in facilitating communication and data transfer between various components. Some examples of communications interfaces in a car dealership management system include:

* Email Notifications: The system can send automated email notifications to customers regarding vehicle updates, service reminders, promotions, or special offers.
* Customer Surveys: The system can send surveys to customers via email, allowing them to provide feedback on their experience, satisfaction levels, or specific areas of improvement.
* SMS Notifications: The system can send SMS messages to customers to provide instant updates on their vehicle status, appointment reminders, or special promotions.
* Advertising Campaigns: The system can integrate with advertising platforms and tools to run targeted ad campaigns on platforms like Google Ads or social media platforms, reaching potential customers and driving traffic to the dealership's website or showroom.
* Social Media Integration: The system can integrate with social media platforms like Facebook, Twitter, or Instagram, allowing the dealership to post updates, showcase vehicles, engage with customers, and respond to inquiries or comments.
* Social Media Advertising: The system can utilize social media advertising features to create targeted ads that reach specific demographics or user interests, promoting the dealership's vehicles, offers, or events directly to potential customers on social media platforms.

1. **System Design specifications**
   1. **Architecture Design:**
      1. **Data Flow Diagram(DFD):**

A Data Flow Diagram (DFD) is a graphical representation that shows the flow of data within a system. It illustrates how data moves from one process to another, as well as the external entities and data stores involved. DFDs use symbols to represent processes, data flows, external entities, and data stores, with arrows indicating the direction of data flow. DFDs help in understanding the system's data flow, identifying inputs and outputs, and highlighting data transformations. They provide a clear and concise visualization of the system's data movement, aiding in system analysis, design, and communication between stakeholders. The basic symbols used to construct data flow diagrams are-

* + 1. **Data flow diagram symbol**



* + 1. **Context level DFD**

The context level data flow diagram (dfd) is describe the whole system. The (o) level dfd describe the all user module who operate the system. Below data flow diagram of online shopping site shows the two user can operate the system Admin and Member user.

|  |
| --- |
| Vehicle Showroom System design - Free Student Projects |
| **Fig 5.1.3: 0 – level DFD** |

* + 1. **1st Level Admin Side DFD**

The Admin side DFD describe the functionality of Admin, Admin is a owner of the website. Admin can first add category of item and then add items by category wise. and admin can manage order and payment detail.

|  |
| --- |
|  |
| **Fig 5.1.4 : Admin side data flow diagram** |

**5.1.5. User side data flow diagram:**

The user is all people who operate or visit our website. User is a customer of a website. User can first select product for buy, user must have to register in our system for purchase any item from our website. after register he can login to site and buy item by making online payment through any bank debit card or credit card.

|  |
| --- |
|  |
| **Fig 5.1.5. User side DFD** |

* 1. **Use case diagram:**

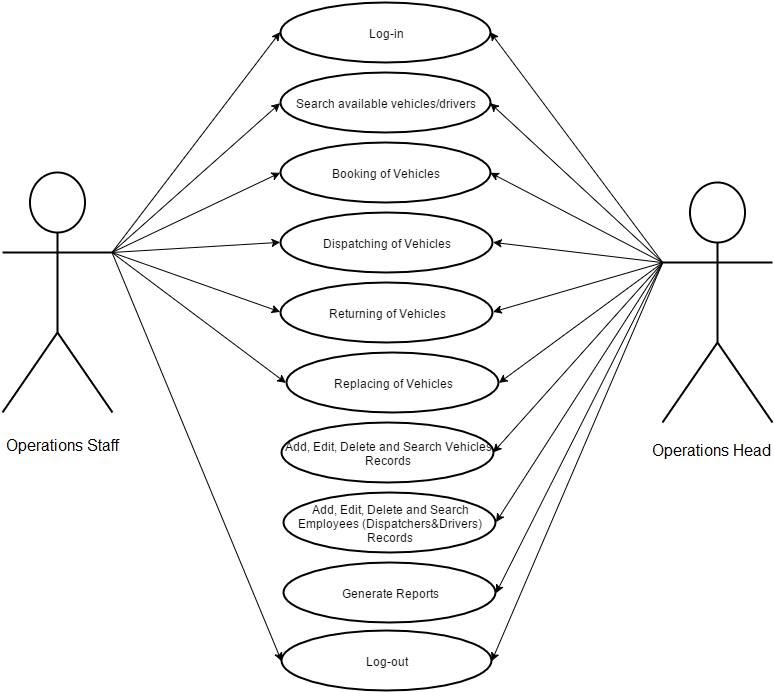
A use case diagram is a type of UML diagram that shows the interactions between actors and use cases. An actor is a person, organization, or system that interacts with the system. A use case is a description of a functionality that the system provides.

The use case diagram for an ecommerce system would show the following actors:

* Customer: A customer is a person who uses the system to purchase products.
* Merchant: A merchant is a person or organization that sells products through the system.
* System Administrator: A system administrator is a person who manages the system.

The use case diagram would also show the following use cases:

* The use case diagram is a useful tool for understanding the interactions between actors and use cases in an ecommerce system. It can be used to communicate the system's functionality to stakeholders and to help developers and testers understand how the system works.



**Fig 5.2. Use case diagram**

* 1. **ER diagram:**

An ER diagram is a type of UML diagram that shows the relationships between entities. An entity is a person, place, thing, or concept that is represented in the system.

The ER diagram for an ecommerce system would show the following entities:

* Customer: A customer is a person who uses the system to purchase products.
* Product: A product is an item that is sold through the system.
* Order: An order is a request by a customer to purchase a product.
* Payment: A payment is a transfer of money from a customer to a merchant in exchange for a product.
* Shipping: Shipping is the process of transporting a product from the merchant to the customer.

|  |
| --- |
|  |
| **Fig 5.3. ER Diagram** |