Software Requirements Specification

for

Car Dealership System

Version 1.1 approved

Prepared by M Amith Kini

15IT216

Information Technology, NITK

09/01/2018

Table of Contents

[1. Introduction 1](#_Toc503221066)

[1.1 Purpose 1](#_Toc503221067)

[1.2 Definitions, acronyms, and abbreviations 1](#_Toc503221068)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc503221069)

[1.4 Product Scope 1](#_Toc503221070)

[1.5 References 2](#_Toc503221071)

[2. Overall Description 3](#_Toc503221072)

[2.1 Product Perspective 3](#_Toc503221073)

[2.2 Product Functions 3](#_Toc503221074)

[2.3 User Classes and Characteristics 4](#_Toc503221075)

[2.4 Operating Environment 4](#_Toc503221076)

[2.5 Design and Implementation Constraints 4](#_Toc503221077)

[2.6 User Documentation 4](#_Toc503221078)

[2.7 Assumptions and Dependencies 5](#_Toc503221079)

[3. External Interface Requirements 6](#_Toc503221080)

[3.1 User Interfaces 6](#_Toc503221081)

[3.2 Hardware Interfaces 7](#_Toc503221082)

[3.3 Software Interfaces 7](#_Toc503221083)

[3.4 Communications Interfaces 7](#_Toc503221084)

[4. System Features 8](#_Toc503221085)

[4.1 User class 1 – The customer 8](#_Toc503221086)

[4.1.1 User registration 8](#_Toc503221087)

[4.1.2 User login 8](#_Toc503221088)

[4.1.3 User homepage 9](#_Toc503221089)

[4.1.4 Car search 9](#_Toc503221090)

[4.1.5 Car information 9](#_Toc503221091)

[4.1.6 Test drive 10](#_Toc503221092)

[4.1.7 Car booking 10](#_Toc503221093)

[4.2 User class 2 – The Administrator 10](#_Toc503221094)

[4.2.1 Login 10](#_Toc503221095)

[4.2.2 Dashboard 11](#_Toc503221096)

[4.2.3 Add cars 11](#_Toc503221097)

[4.2.4 Modify/Delete cars 11](#_Toc503221098)

[5. Other Non-functional Requirements 12](#_Toc503221099)

[5.1 Performance Requirements 12](#_Toc503221100)

[5.2 Security Requirements 12](#_Toc503221101)

[5.2.1 Communication security 12](#_Toc503221102)

[5.2.2 Account security 12](#_Toc503221103)

[5.3 Software Quality Attributes 12](#_Toc503221104)

[5.3.1 Maintainability 12](#_Toc503221105)

[5.3.2 Portability 12](#_Toc503221106)

[5.4 Business Rules 12](#_Toc503221107)

[Appendix A: To Be Determined List 13](#_Toc503221108)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Amith Kini | 9/1/2018 | Initial version | 1.0 |
| Amith Kini | 29/1/2018 | Added more functional requirements | 1.1 |

# Introduction

## Purpose

The purpose of this document is to give a detailed description of the requirements for the “Car Dealership System” (CDS) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

## Definitions, acronyms, and abbreviations

|  |  |
| --- | --- |
| **Term** | **Definition** |
| User | Someone who interacts with the application |
| Admin/Administrator | System administrator who is given specific permission for managing and controlling the system |
| System | The car dealership system |
| Car Dealership | The organization which provides a platform to sell cars to the customer |
| Customer | Someone who is interested in buying a good |
| CDS | Car Dealership System |
| Store manager | The person who manages the store/dealership |
| Store employees | People who are employed by the organization |

## Intended Audience and Reading Suggestions

The intended audiences of stakeholders for this specification of the CDS include:

* Employees of Car Dealership
* Users, who interact with the system
* Administrators

The users here are the selected beta testing group, who test the system thoroughly, informs the development team of existing bugs, and suggests features.

Employees looks after the test drive and booking features. They can query these results, but cannot modify any of these.

The administrator looks after the entire application. They are responsible for the contents in the application.

## Product Scope

This software system will be a Car Dealership website for dealerships that look to attract customers online and allow customers to be well educated about their choices before they enter the physical store. A user can see the list of all the cars that are sold by the car dealership.

The store manager needs to add the cars with their specification to the system. The user can interact with these cars in the system and all the details of the car can be known. The user can also compare between two cars and get to know the differences between them. If the user is interested in checking out the car in person, they can request for a test drive.

The user can also buy the car from the same system. In addition to this, the user can add accessories with the car and order it directly. The store manager needs to update the location of the delivery of the car, which the user can track.

## References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

# Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system is presented.

## Product Perspective

The system will consist of two parts: The web application and the database. The web application has two types of user interface: The customer UI and the admin UI.

The customer UI has a provision for login and register. After logging in, the customer can browse for cars and compare between two cars. In addition to this, the customer can book for a test drive or even order the car with the accessories. Apart from this, the customer can leave their reviews for a particular car that is visible to all the other users on the system.

The admin UI has access to all the data on the system. Admin and store managers, who have elevated access levels, can add, modify or delete data that are present in the system. The admin user can access data related to cars and user. Apart from this, the admin user has access to orders from the customer and also update the order status. In addition to this, the test drive slots must be visible to the admin, so that they can make necessary arrangements.

Since this is a data-centric product, it will need somewhere to store the data. For that, a database will be used. The application will communicate with the database to receive data, for the customer side and the application will be able to modify the data if an admin user is accessing it. All of the database connection will go over the internet.

Since the application is hosted on the internet, the size of each page must not exceed 10 MB. Reduction in webpage size can increase the performance of the webpage and the server.

## Product Functions

* The users will be able to search for cars. The result will be based on the criteria the user inputs. There are several search criteria and there are filters to make the search results relevant for the user. The administrator can modify these filters.
* On clicking the image of a particular car, the details of the car appear. These details will be provided by the administrator, and is stored in the database.
* The user can compare two cars by clicking on compare and then selecting another car. The comparison will highlight which car will have the better specification in each category.
* If the user wants to test drive the car, they can click on a button, which will allot them a slot. This slot can be viewed by the admin and can be changed.
* The user can review and rate the car, which is publicly visible and to other users.
* The user can order the car along with its accessories and can track the order. The admin can view the orders and update the order status.

## User Classes and Characteristics

There are three types of users that interact with the system: customers, store employees and administrator. Each of these three types of users has different use of the system so each of them has their own requirements.

The customers can use the application to search, compare and purchase the cars. They do not modify the data that is displayed to rest of the users. For searching a car, a user is given various options like the price filter, number of seats etc. so that the user can search efficiently.

The store employees use the same application, but they have a different login. The employees can track the test drive requests from the users and track the orders made by the user. The updated orders are visible to the users.

The administrator uses the same application, but they have a different login. They have all the access of store employees in addition to access to modify and add data in the application. They can adjust the filter options for the customers.

## Operating Environment

The software will be hosted in a server that can be accessed with the help of the internet. The server will be running Ubuntu 16.04 LTS which has a 5 years of support. The implementation has not yet been decided yet.

## Design and Implementation Constraints

The internet connection is a constraint for the application. Since the application fetches data from the database over the internet, it is crucial that there is an Internet connection for the application to run.

The application will be constrained by the capacity of the database. Since many users use the database at a particular time, it may be forced to queue incoming requests and therefore increase in the time that it takes to fetch the data.

Since the administrator decides the content inside the application, the data must be backed up in a different location. The content must also be appropriate and the administrator must monitor all of this.

The type of the database used is not yet decided. Also, the type of server will be decided after comparing the different implementable servers.

## User Documentation

The user documentation will be prepared after the completion of the project. The customers may not refer to this user documentation as the UX for the customer will be simple and easy to understand. The user manual will be in a Wiki format.

## Assumptions and Dependencies

Our assumption is that the application will be used on a computer or a mobile device with a modern web browser. Since it is a web application, use of an old, outdated browser could pose a compatibility and security issue.

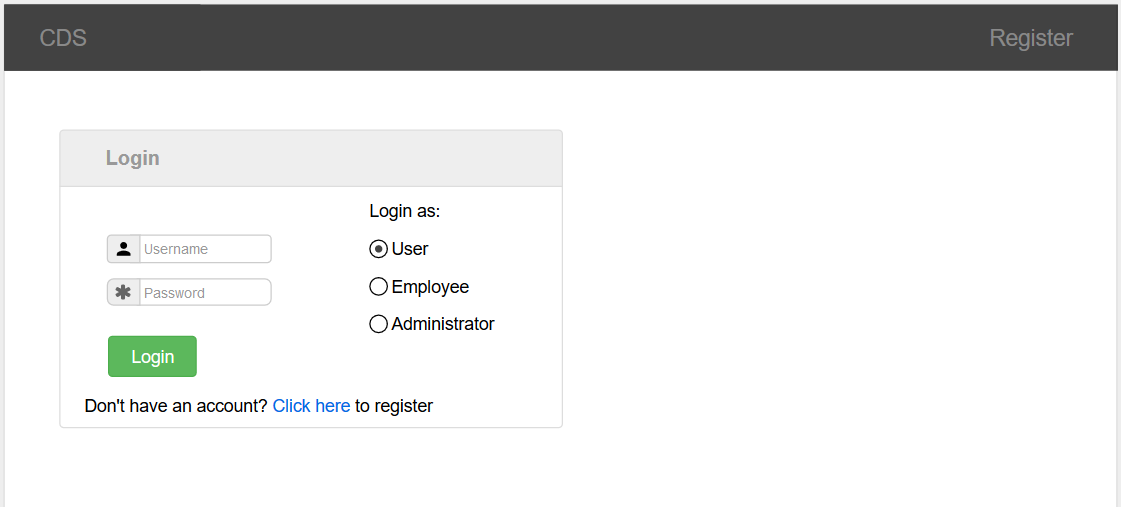
Also, being a web application, the server needs to be up all the time. Downtime in the server could mean loss of customers for the organization.

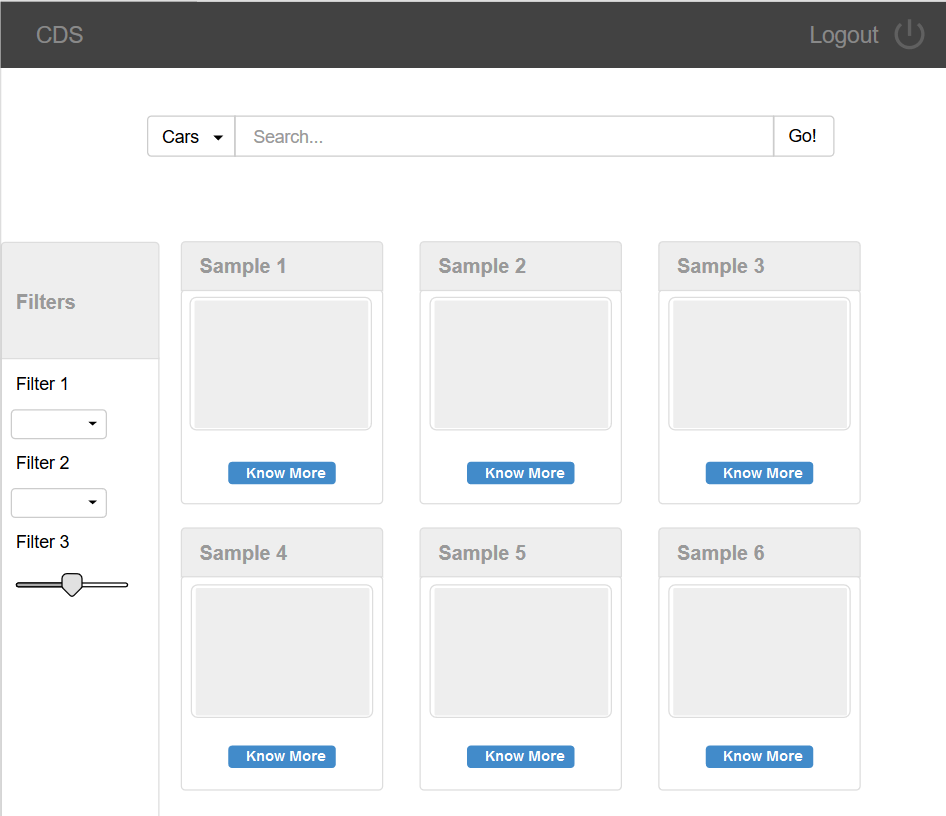
# External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

## User Interfaces

The first page that the users see when they log on the website is the landing page. The landing page is filled with offers and details about some cars. The design of the landing page changes as per the needs of the client.

If the user clicks on login, then the user is presented with a page to enter their credentials.

After the user is logged in, they are presented with a catalogue page with all the cars listed.

If the user clicks on the car, they are taken to the car details page, where all the information of the cars are shown and options for test drive or to book the car are given. If the user clicks the test drive button, a form is presented to the user to retrieve the details. If the user click on the “buy” button, the user is taken to the order confirmation page, where he makes the payment and retrieves the order details. The UI design for these functionalities are still in the works.

Similarly in the administrator side, options will be provided to check the test drive list, order lists and to update the car models. The UI design for these functionalities are still in the works.

## Hardware Interfaces

Since the web application does not have a designated hardware, it does not have any direct hardware interfaces. The hardware connection to the database servers is managed by the underlying operating system on the web application.

## Software Interfaces

The communication between the database and the web application from the users consists of only reading operation, while the communication between the database and the administrator application consists of operation concerning both reading and modifying the data. There is no dependency on the operating system, but the operating system must contain a modern browser.

## Communications Interfaces

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for the web application.

# System Features

This section includes the requirements that specify all the fundamental actions of the software system.

## User class 1 – The customer

### User registration

4.1.1.1 Description and Priority

To use the website to its fullest potential, the user should be able to register through the application. The user must provide user-name, password and e-mail address. The user can choose to provide a regularly used phone number. This has a high priority, because without this snippet, the entire application would be useless.

4.1.1.2 Stimulus/Response Sequences

On the main webpage, login button is pressed. Below the login form, there will be a link to the registration page.

4.1.1.3 Functional Requirements

REQ-1: The registration page must accept name, username, password and phone number as the input.

REQ-2: The password must be verified with a confirm password field.

REQ-3: The username must be unique for each user.

REQ-4: No field must be blank.

### User login

4.1.2.1 Description and Priority

Given that a user has registered, then the user should be able to log in to the application. The user with the help of browser password manager can save the login information. This has a high priority, because without this snippet, the entire application would be useless.

4.1.2.2 Stimulus/Response Sequences

On the main webpage, login button is pressed. The login form is then shown.

4.1.2.3 Functional Requirements

REQ-1: The login form must have radio buttons to select between user, employee and administrator.

REQ-2: The login form must query for username and password.

REQ-3: In case of an error, the error is shown explicitly.

REQ-4: [TBD] In case the user forgot the password, there must be a “forgot password” link.

### User homepage

4.1.3.1 Description and Priority

After login, the user is shown a page, which contains all the cars for sale in the showroom. This has a high priority, as any sales cannot be made without this part.

4.1.3.2 Stimulus/Response Sequences

After the user logs into the application, this page is shown.

4.1.3.3 Functional Requirements

REQ-1: All cars for sale must be shown on the page.

REQ-2: Only 25 cars can be shown at a time. Rest of the cars must be in another page.

REQ-3: The cars must be arranged in a priority order.

REQ-4: A search functionality is placed on top of the page.

### Car search

4.1.4.1 Description and Priority

To find the preferred car, the user needs to give some criteria so that a few filtered cars appear in the search results, out of many cars. This has a medium priority, as the application will still work without this feature, but it will be hard for the user to find the car.

4.1.4.2 Stimulus/Response Sequences

In the homepage, the search bar will be on the top, with a button to execute the search.

4.1.4.3 Functional Requirements

REQ-1: The search bar must be easy to access.

REQ-2: The filters must be present on the side to make the search easy.

REQ-3: In case there are no search results, it must be mentioned to the user.

REQ-4: [TBD] A predictive search bar must be used.

### Car information

4.1.5.1 Description and Priority

If the user clicks on the image of a car, the information must appear about the car and all the specifications of the car must be present.

4.1.5.2 Stimulus/Response Sequences

In the homepage, click on the image of a car to bring this page out.

4.1.5.3 Functional Requirements

REQ-1: The information must be clearly presented.

REQ-2: If some data is missing, it should be indicated as such.

REQ-3: There must be a button to buy the car

REQ-4: [TBD] A suggestion section which suggests other cars that are similar to the car which is currently displayed.

### Test drive

4.1.6.1 Description and Priority

If the user wants to test drive the car at the showroom, they can click on the button to request for the test drive. From there, they can select the date and time for the test drive.

4.1.6.2 Stimulus/Response Sequences

On the information page of a car, this button will open a form to test drive the car.

4.1.6.3 Functional Requirements

REQ-1: The form, after fill up must return a confirmation and a UID.

REQ-2: In case a test drive request failed, it must be informed to the user.

REQ-3: [TBD] There must be limited number of test slots.

### Car booking

4.1.7.1 Description and Priority

If the user wants to buy the car, they can do so by clicking a button which will lead them to the order page.

4.1.7.2 Stimulus/Response Sequences

On the information page of a car, this button will open a portal for payment.

4.1.7.3 Functional Requirements

REQ-1: After transaction, it must return a confirmation and a UID.

REQ-2: In case a transaction failed, it must inform the user.

REQ-3: [TBD] There must be limited number of cars.

### Booking status

4.1.8.1 Description and Priority

After the car is ordered, the booking status is displayed in this page. This carries a medium priority.

4.1.8.2 Stimulus/Response Sequences

On the dashboard, a link to booking status will be visible.

4.1.8.3 Functional Requirements

REQ-1: All orders must be shown, both success and failure.

REQ-2: Products bought in the list must be shown.

REQ-3: Provision to cancel the order must be provided.

## User class 2 – The Administrator

### Login

4.2.1.1 Description and Priority

Since the admin account is created at the start, there is no registration page for the administrator. They can directly login to the application. This is a high priority as the admin cannot do anything without authentication.

4.2.1.2 Stimulus/Response Sequences

On the main webpage, login button is pressed. A login form is presented.

4.2.1.3 Functional Requirements

REQ-1: The login form must have radio buttons to select between user, employee and administrator.

REQ-2: The login form must query for username and password.

REQ-3: In case of an error, the error is shown explicitly.

REQ-4: [TBD] In case the user forgot the password, there must be a “forgot password” link.

### Add/edit users

4.2.2.1 Description and Priority

Admin must be capable of adding employees as users in the application. This is a high priority.

4.2.1.2 Stimulus/Response Sequences

On the dashboard, there will be a link to add/edit users.

4.2.1.3 Functional Requirements

REQ-1: The password for each new user must be set by admin themselves.

REQ-2: Deleting user deletes all objects associated with the user, unless explicitly saved by the administrator.

REQ-3: In case of an error, the error is shown explicitly.

### Dashboard

4.2.3.1 Description and Priority

The dashboard contains the highlights of the administrator works. Some of them are cars added, test drive applications. This is a high priority, as the admin cannot control the application without this.

4.2.3.2 Stimulus/Response Sequences

After login, the dashboard appears as the main page to the admin.

4.2.3.3 Functional Requirements

REQ-1: The dashboard must contain links to all admin actions.

### Add cars

4.2.4.1 Description and Priority

To add new cars, this function is required. This is a high priority, as the admin cannot add new cars without this.

4.2.4.2 Stimulus/Response Sequences

In the dashboard, there will be a link to add cars.

4.2.4.3 Functional Requirements

REQ-1: The form must include all the required fields.

### Modify/Delete cars

4.2.5.1 Description and Priority

After the car is added, there must be provision to delete the cars. This has a medium priority, as the application will work even if this is not included.

4.2.5.2 Stimulus/Response Sequences

In the dashboard, there will be a link to modify and delete cars.

4.2.5.3 Functional Requirements

REQ-1: The form must include all the required fields.

REQ-2: The delete must be immediate, and all the linked orders must point to a generic object.

### View and edit test drives and orders

4.2.6.1 Description and Priority

All the user requested test drives and orders must be visible to the administrator. This has a medium priority.

4.2.6.2 Stimulus/Response Sequences

In the dashboard, there will be a link to enter this page.

4.2.6.3 Functional Requirements

REQ-1: The list must be ordered and must be editable by the administrator.

REQ-2: The orders must contain the product selected and payment details.

## User class 3 – Car dealer employee

### Login

4.3.1.1 Description and Priority

Since the employee account is created at the start by the administrator, there is no registration page for the administrator. They can directly login to the application. This is a high priority.

4.3.1.2 Stimulus/Response Sequences

On the main webpage, login button is pressed. A login form is presented.

4.3.1.3 Functional Requirements

REQ-1: The login form must have radio buttons to select between user, employee and administrator.

REQ-2: The login form must query for username and password.

REQ-3: In case of an error, the error is shown explicitly.

REQ-4: [TBD] In case the user forgot the password, there must be a “forgot password” link.

### View and edit test drives and orders

4.2.6.1 Description and Priority

All the user requested test drives and orders must be visible to the employee. This has a medium priority.

4.2.6.2 Stimulus/Response Sequences

In the dashboard, there will be a link to enter this page.

4.2.6.3 Functional Requirements

REQ-1: The list must be ordered and must be editable by the employee.

REQ-2: The orders must contain the product selected and payment details.

# Other Non-functional Requirements

## Performance Requirements

* The loading time of the website must be quick, as there is an expected traffic of 100 users per minutes on the server. Therefore, gzip compression of all objects is necessary.
* The search functionality must be easy and fast to use for the user.
* The image of the cars must be quick to load.
* The response time of the website must not exceed 2 seconds.
* The system must be fault tolerant.

## Security Requirements

### Communication security

The communication between the server and the client website must be encrypted so as to avoid spywares. The website will operate on HTTPS protocol for security.

### Account security

The accounts created must be secure from external threats. The database created will only be accessible from the website server. All the passwords in the database will be saved as hashes.

## Software Quality Attributes

### Maintainability

The application should be easy to extend. The code should be written in a way that it favours implementation of new functions. Test environments should be built for the application to allow testing of the applications different functions.

### Portability

The application must be portable with mobile OS like iOS and Android.

## Business Rules

There are three types of users for the application: Consumers, Employees and Administrator. These users have their own access control as mentioned in the functional requirements.

# Appendix A: To Be Determined List

* “Forgot password” section
* Predictive search bar
* Suggestion section under each car
* Controlled test slots
* Limited sales for each car
* Pre-order functionality