

Darshan University

A Project Report on

**“**Car Showroom Management System”

Under the subject

**Software Engineering (2101CS503)**

B. Tech, Semester – VI

Computer Science & Engineering Department

|  |  |
| --- | --- |
| Submitted By | |
| Student Name: Tisha Mehta | Enrollment No.: 21010101125 |
| Academic Year  (2023-2024) | |
| Internal Guide  Prof. Devangi Kotak  Darshan University | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**DECLARATION**

We hereby declare that the SRS, submitted along with the **Software Engineering** **(2101CS503)** for entitled **“Car Showroom Management System”** submitted in partial fulfilment for the Semester-5 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Departmentto Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of **Prof**. **Devangi Kotak** and that no part of any of report has been directly copied from any students’ reports, without providing due reference.

Tisha Mehta

Student’s Signature

Date: \_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**CERTIFICATE**

This is to certify that the SRS on “Car Showroom Management System” **has** been satisfactorily prepared by **Tisha Mehta** (**21010101125**) under my guidance in the fulfillment of the course **Software Engineering (2101CS503)** work during the academic year 2023-2024.

|  |  |  |
| --- | --- | --- |
| Internal Guide  Prof. Devangi Kotak  Darshan University |  | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

**Acknowledgement**

I wish to express my sincere gratitude to my project guide Prof. **Devangi Kotak** and all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co- ordination in completing this work.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You

**Tisha Mehta**

**ABSTRACT**

The car showroom management system is a web based application designed to facilitate the booking car through online. The current system is offline system, in this purchase vehicle the customer should visit to showroom so this current system is very difficult because its fine consuming. Customer can register to this site and she can book car by login information administrator is main user of this system and she/he can add new car details. The system will provide features for login module, car module, search module etc.

**Table of Contents**

[List of Figures I](#_Toc147394721)

[List of Tables II](#_Toc147394722)

[1 Introduction 1](#_Toc147394723)

[1.1 Product perspective 1](#_Toc147394724)

[1.2 Product features 1](#_Toc147394725)

[1.2.1 There are three different users who will be using this product: 1](#_Toc147394726)

[1.2.2 The features that are required for the Customer are: 1](#_Toc147394727)

[1.2.3 The features that are required for the Admin are: 1](#_Toc147394728)

[1.3 Functional Requirement 1](#_Toc147394729)

[1.3.1 Customer 1](#_Toc147394730)

[1.3.2 Admin 2](#_Toc147394731)

[1.4 Non-Functional Requirement 2](#_Toc147394732)

[1.4.1 Usability: 2](#_Toc147394733)

[1.4.2 Accuracy: 2](#_Toc147394734)

[1.4.3 Availability: 2](#_Toc147394735)

[1.4.4 Maintainability: 2](#_Toc147394736)

[2 Design and Implementation Constraints 3](#_Toc147394737)

[2.1 Use case diagram 3](#_Toc147394738)

[2.2 Activity diagram and Swimlane diagram 4](#_Toc147394739)

[2.3 Sequence diagram 6](#_Toc147394740)

[2.4 State diagram 7](#_Toc147394741)

[2.5 Class diagram 8](#_Toc147394742)

[2.6 Data flow diagram 9](#_Toc147394743)

[2.6.1 Context diagram (level-0) 9](#_Toc147394744)

[2.6.2 DFD Level-1 10](#_Toc147394745)

[2.6.3 DFD Level-2 11](#_Toc147394746)

[3 External interface requirement (Screens) 12](#_Toc147394747)

[3.1 Screen-1: Order Creation Form 12](#_Toc147394748)

[3.2 Screen-2: Credit Card Payment 13](#_Toc147394749)

[3.3 Screen-3: Bank Transfer Payment 14](#_Toc147394750)

[4 Database design 15](#_Toc147394751)

[4.1 List of Tables 15](#_Toc147394752)

[5 Stories and Scenario 17](#_Toc147394753)

[5.1 Story-1: Order Creation 17](#_Toc147394754)

[5.1.1 Scenario# S1.1 17](#_Toc147394755)

[5.1.2 Scenario# S1.2 17](#_Toc147394756)

[5.1.3 Scenario# S1.3 17](#_Toc147394757)

[5.2 Story-2: Search 18](#_Toc147394758)

[5.2.1 Scenario# S2.1 18](#_Toc147394759)

[5.2.2 Scenario# S2.2 18](#_Toc147394760)

[5.2.3 Scenario# S2.3 18](#_Toc147394761)

[5.3 Story-3: Payment 19](#_Toc147394762)

[5.3.1 Scenario# S3.1 19](#_Toc147394763)

[5.3.2 Scenario# S3.2 19](#_Toc147394764)

[5.3.3 Scenario# S3.3 19](#_Toc147394765)

[5.4 Story-4: Tracking 20](#_Toc147394766)

[5.5 Story-5: Notification 20](#_Toc147394767)

[5.6 Story-6: Order Details 20](#_Toc147394768)

[5.7 Story-7: Add to Cart 20](#_Toc147394769)

[5.8 Story-8: Add new Car 21](#_Toc147394770)

[5.9 Story-9: Update Car 21](#_Toc147394771)

[5.10 Story-10: Feedback 21](#_Toc147394772)

[6 Test cases 22](#_Toc147394773)

[7 References 33](#_Toc147394774)

# List of Figures

[Figure 2.1‑1 Use case diagram for Car Showroom Management System 3](#_Toc147423380)

[Figure 2.2‑1 Activity diagram for Car Purchase 4](#_Toc147423381)

[Figure 2.2‑2 Swimlane diagram for Car Purchase 5](#_Toc147423382)

[Figure 2.3‑1 Sequence diagram for Car Purchase 6](#_Toc147423383)

[Figure 2.4‑1 State diagram of Customer 7](#_Toc147423384)

[Figure 2.5‑1 Class diagram for Car Showroom Management System 8](#_Toc147423385)

[Figure 2.6‑1 Context diagram for Car Showroom Management System 9](#_Toc147423386)

[Figure 2.6‑2 DFD level-1 for Library management system 10](#_Toc147423387)

[Figure 2.6‑3 DFD level-2 for Car Purchase 11](#_Toc147423388)

[Figure 3.1‑1 Screen-1: Order Creation Form 12](#_Toc147423389)

[Figure 3.2‑1 Screen-2: Credit Card Payment 13](#_Toc147423390)

[Figure 3.3‑1 Screen-3: Bank Transfer Payment 14](#_Toc147423391)

# List of Tables

[Table 3.1‑1 Screen element of Order Creation form 12](#_Toc147399994)

[Table 3.2‑1 Screen element of Credit Card Payment 13](#_Toc147399995)

[Table 3.3‑1 Screen element of Bank Transfer Payment 14](#_Toc147399996)

[Table 4.1‑1 Table: Customer 15](#_Toc147399997)

[Table 4.1‑2 Table: Car 15](#_Toc147399998)

[Table 4.1‑3 Table: Transaction 15](#_Toc147399999)

[Table 4.1‑4 Table: Payment 15](#_Toc147400000)

Table 4.1-5 Table: OrderDetails ………………………………………………………………………………………………………………… 16

# Introduction

## Product perspective

The aim of this document is to gather and analyze and give an in-depth insight of the complete can showroom by defining the problem statement in details. The detailed requirement of the car showroom is provide in this document.

## Product features

### There are three different users who will be using this product:

* Customer
* User

### The features that are required for the Customer are:

* Customer login into the application username & password.
* Can view the price of car.
* Can view the car taken is available.
* Can view the different model of car available in the showroom.
* Can add to cart.
* Can purchase the car.
* Can payment the car.

### The features that are required for the Admin are:

* Admin login into the application username & password.
* Offer successful login main page appears.
* Admin add the new car details.
* Admin update the car details.
* Admin delete the car details.
* Deactivate the car details short time period.

## Functional Requirement

### Customer

* Login: login option is providing login; also new signup option is provided for new user to sign up.
* Viewability: The screen will display all car details, exact cost, features, model details etc.
* Search: this search screen option must be provided to search the car based on name or model name.
* Add to cart (selection): this option customer can select car based on its model or name.
* Order: user should be able to create new car purchase within the system.
* Notification: after booked car user will receive booked car all information via mail.
* Bill payment: user pay the payment via credit card, debit card etc.
* Cancel order: user cancel the order.
* Tracking: they provide accurate time, accurate location information for processing further.
* Feedback: user will be able to some feedback.

### Admin

* Add product: the system will provide to admin add new details like new model etc.
* Delete product: the system will provide to admin permently delete the car details.
* Update product: the system will provide to admin update the car details like color, new model, new features etc.
* Deactivate product: the system will provide to admin short time period the product is unavailable.
* Bill generates: the system will provide to handle the bill process and make the bill.
* Stock management: how many stock available in showroom. The practice ordering, storing, tracking and controlling inventory.

## Non-Functional Requirement

### Usability:

* The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

### Accuracy:

* The data stored about the books and the fines calculated should be correct, consistent, and reliable.

### Availability:

* The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

### Maintainability:

* The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

# Design and Implementation Constraints

## Use case diagram

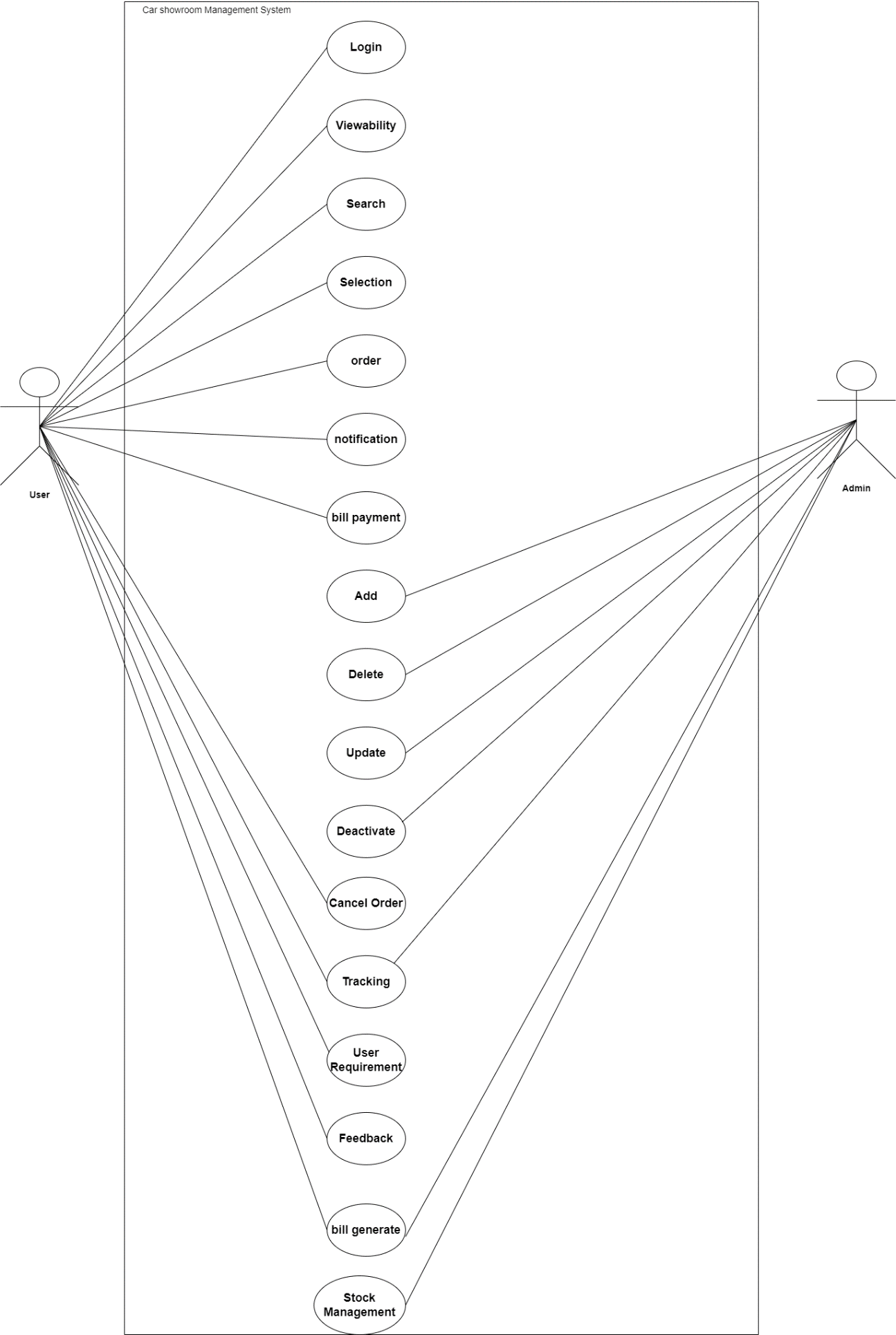


Figure ‑ Use case diagram for Car Showroom Management System

## Activity diagram and Swimlane diagram

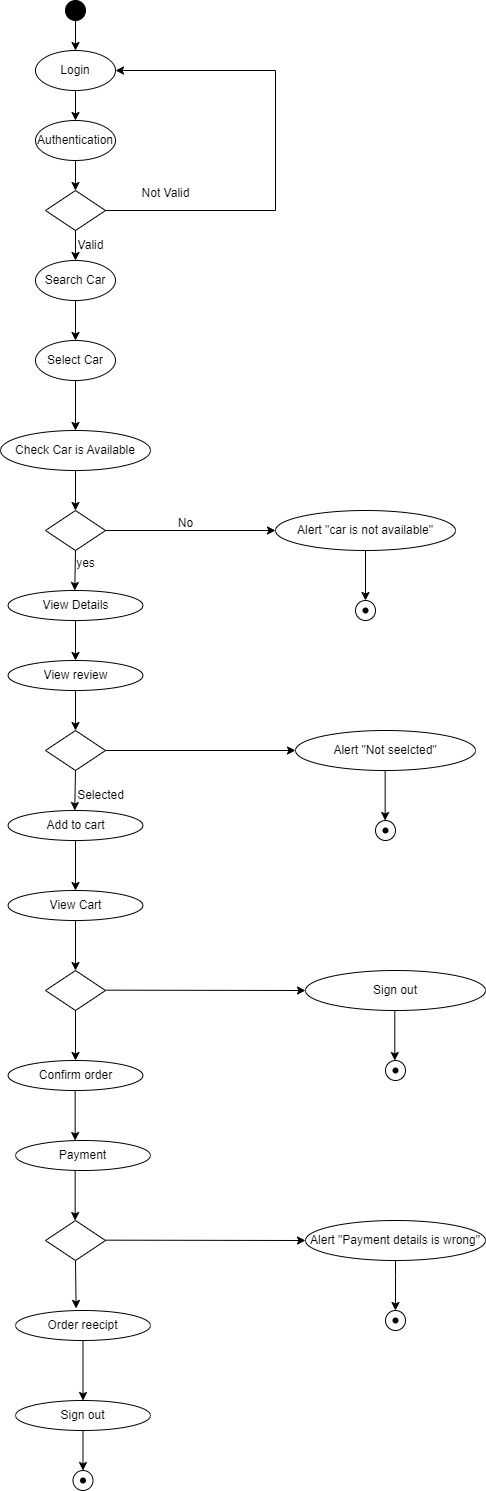


Figure ‑ Activity diagram for Car Purchase

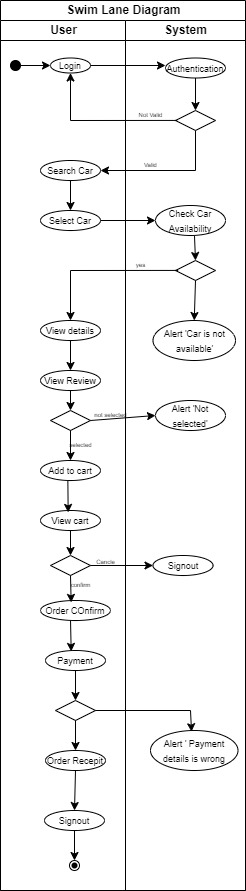


Figure ‑ Swimlane diagram for Car Purchase

## Sequence diagram

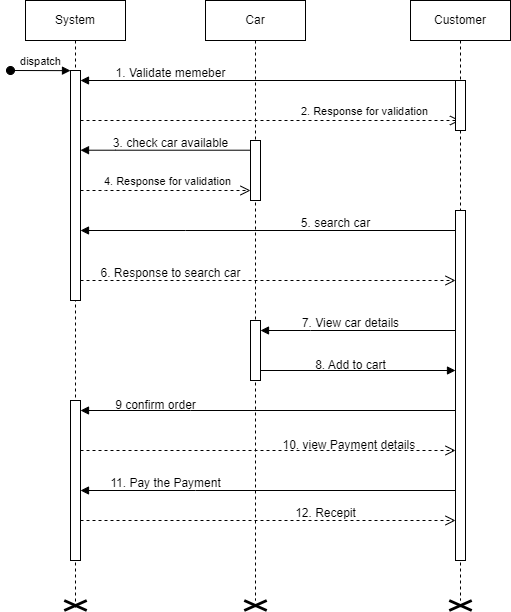


Figure ‑ Sequence diagram for Car Purchase

## State diagram

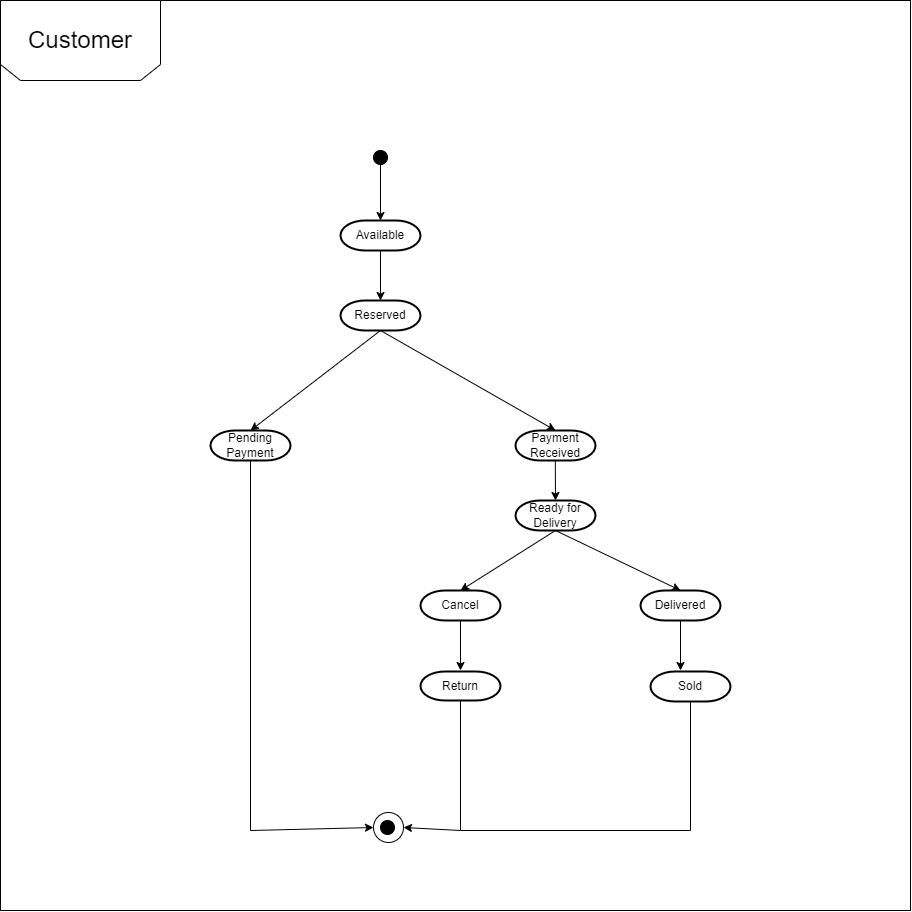


Figure ‑ State diagram of Customer

## Class diagram

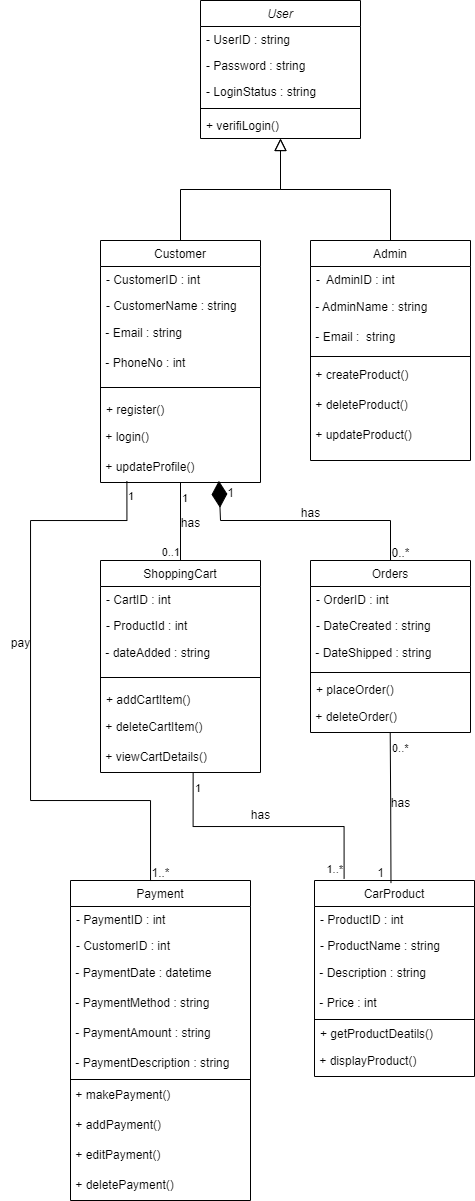


Figure ‑ Class diagram for Car Showroom Management System

## Data flow diagram

### Context diagram (level-0)

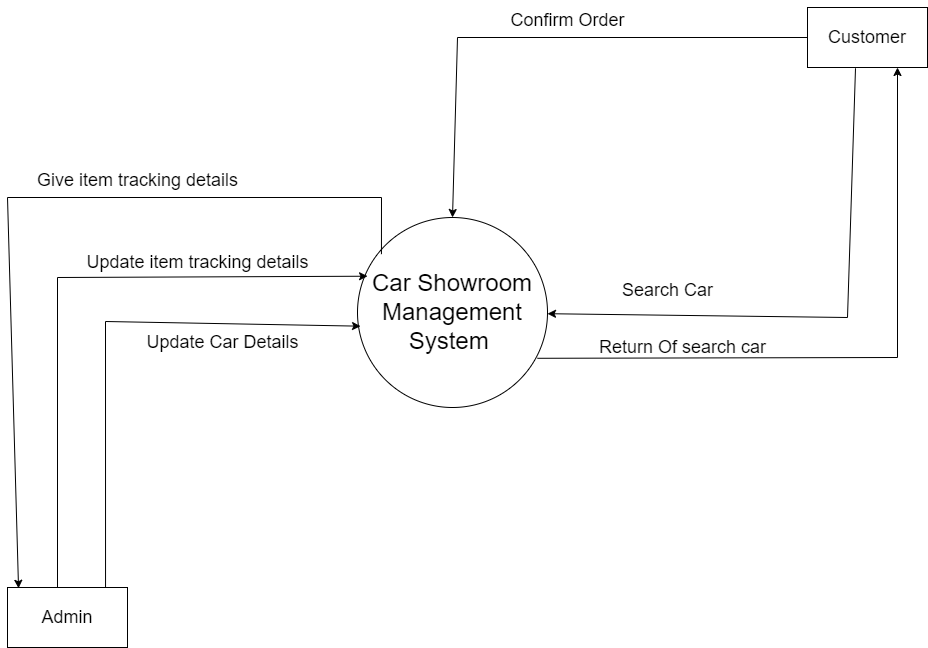


Figure ‑ Context diagram for Car Showroom Management System

### DFD Level-1

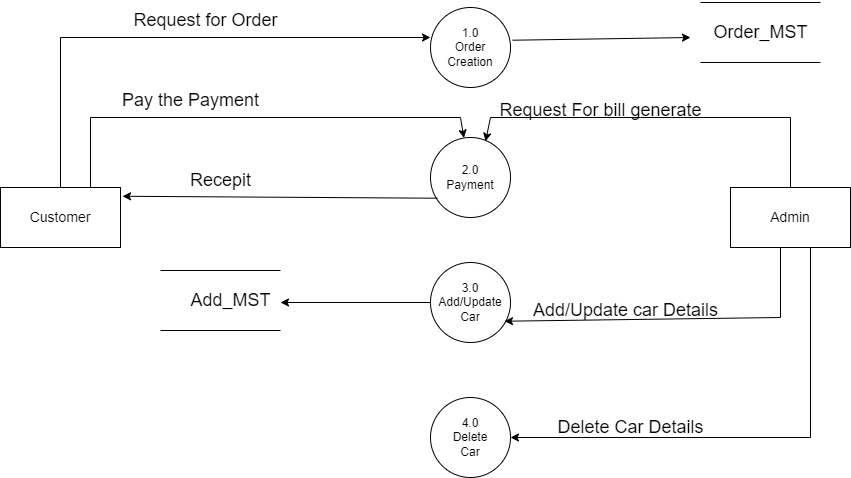


Figure ‑ DFD level-1 for Library management system

### DFD Level-2

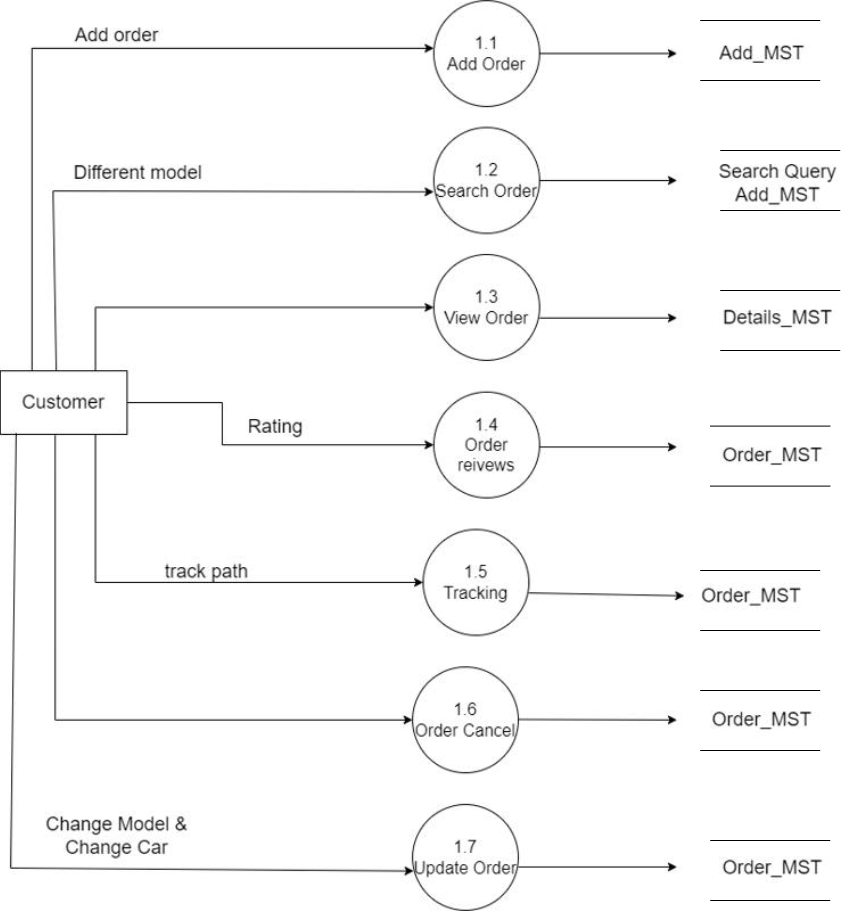


Figure ‑ DFD level-2 for Car Purchase

# External interface requirement (Screens)

## Screen-1: Order Creation Form

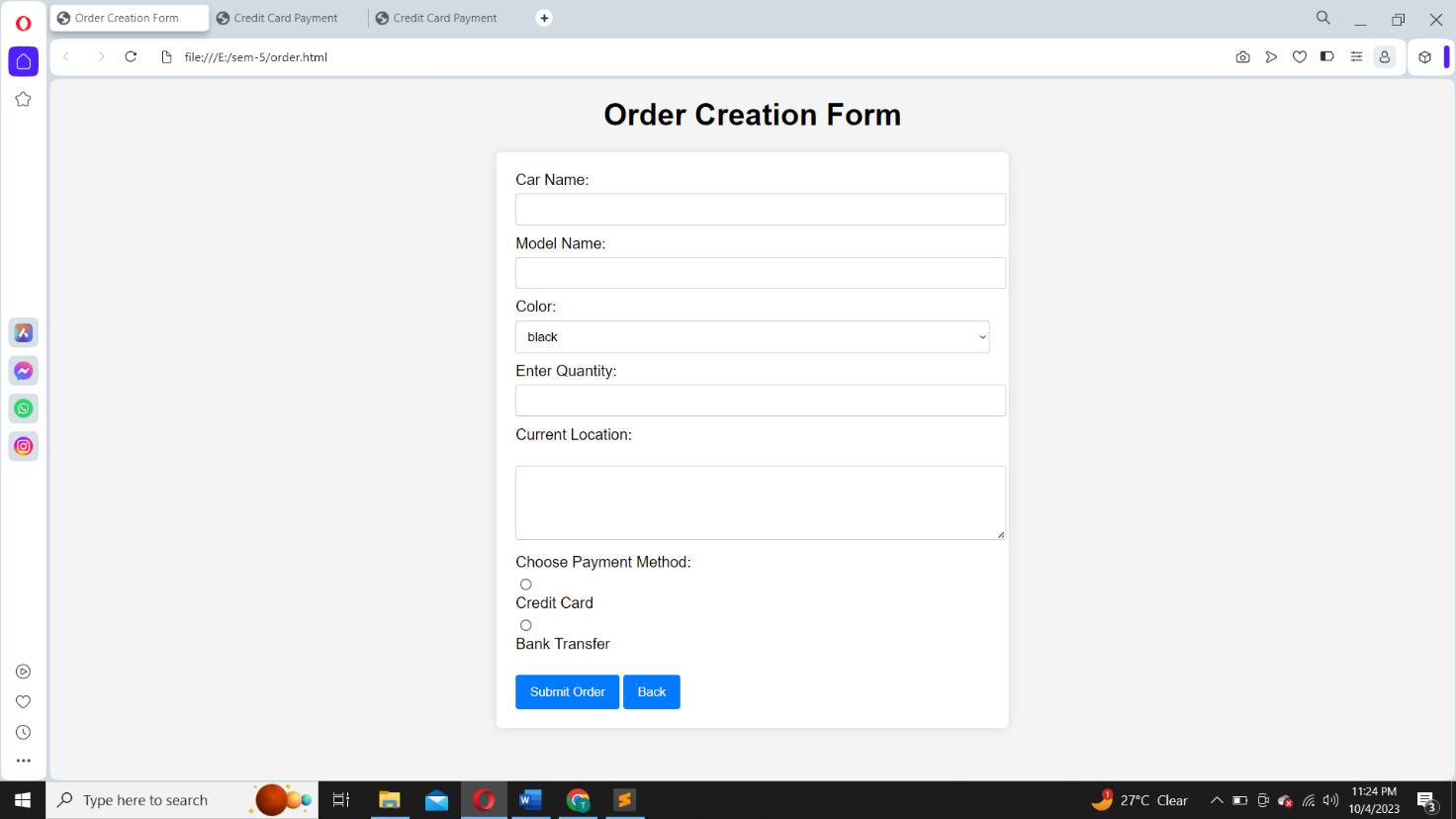


Figure ‑ Screen-1: Order Creation Form

**Purpose:** The purpose of an order creation screen is to enable users to select and purchase products or services by providing necessary information and customization options.

Table ‑ Screen element of Order Creation form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Car Name | Textbox | M | 1 | Validate the car name for proper formatting and length. |
| 2 | Model Name | Textbox | M | 1 | Validate the model name for proper formatting and length |
| 3 | Colour | Dropdown | M | 1 | User Click dropdown and select car colour |
| 4 | Entre Quantity | Textbox | M | 1 | User increase and decrease the quantity |
| 5 | Current Location | Text Area | M | 1 | Validate the Current Location for proper formatting and length |
| 6 | Payment Method | radio | M | 1 | Choose any one payment method |
| 7 | Back | Button | O | 1 | It navigates to previous slide |
| 8 | Submit Order | Button | O | 1 | It navigates to payment screen |

## Screen-2: Credit Card Payment

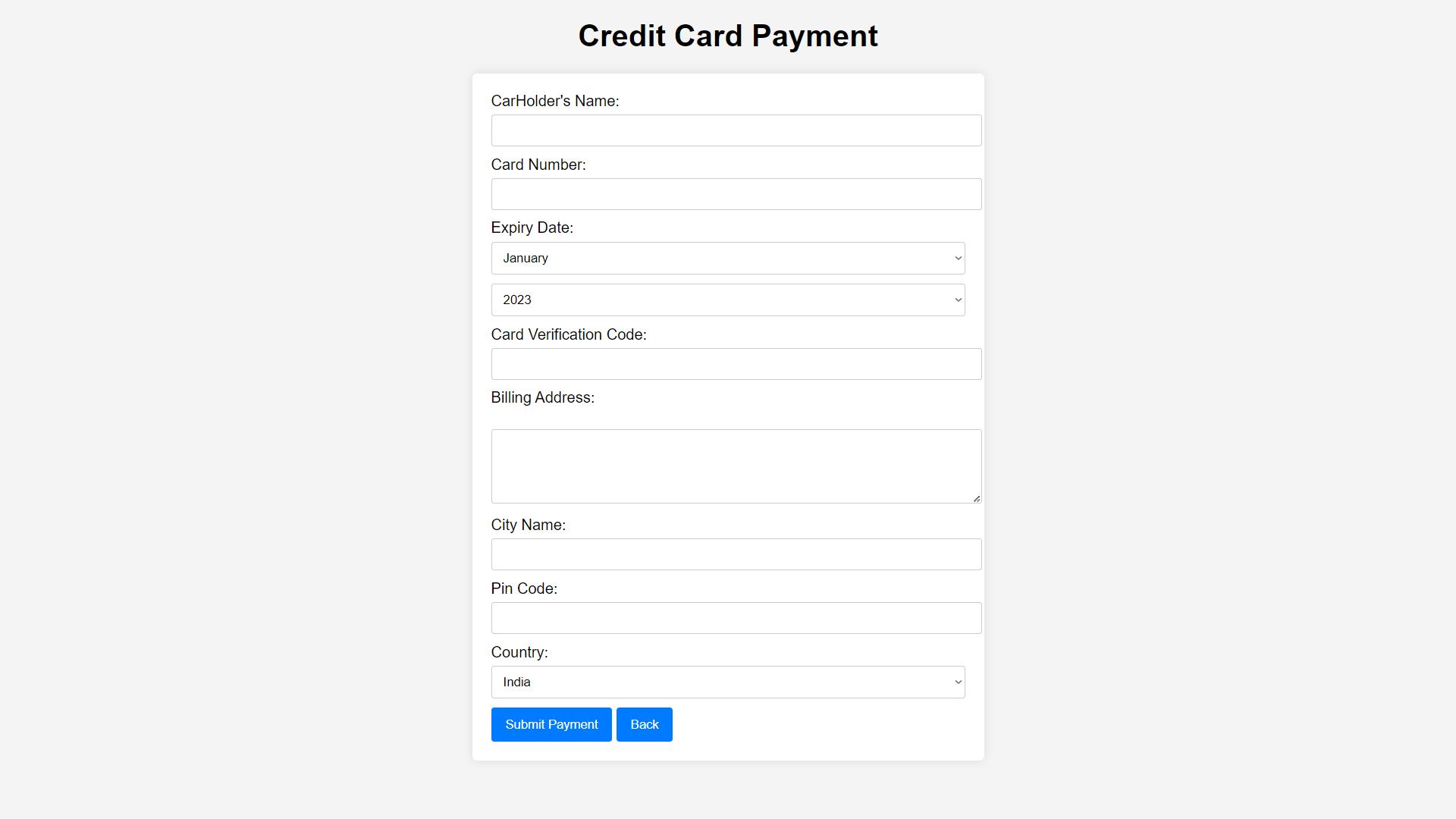


Figure ‑ Screen-2: Credit Card Payment

**Purpose:** securely collect and verify payment information for processing transactions.

Table ‑ Screen element of Credit Card Payment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Card Holder’s Name | Textbox | M | 1 | Validate the card holder’s name for proper formatting and length  Do not store the car holder’s name after the transaction is completed. |
| 2 | Card Number | Textbox | M | 1 | Implement strict validate to ensure that only valid credit card number are accepted.  Do not store full credit card number |
| 3 | Expiry Date | Dropdown for year and month | M | 1 | Validate the expiry date ensure it’s valid date in the future.  Indicate required format (MM/YYYY) |
| 4 | CVV | Textbox | M | 1 | Implement validation for the CVV to ensure it matches the required (3 digit)  Verify that the CVV is correctly entered before process |
| 5 | Billing Address | Textbox | M | 1 | Validate the billing address the car holder’s info. |
| 6 | Submit Payment | Button | O | 1 | Display the message “your payment is successfully” |
| 7 | Back | Button | O | 1 | It navigate the previous screen |

## Screen-3: Bank Transfer Payment

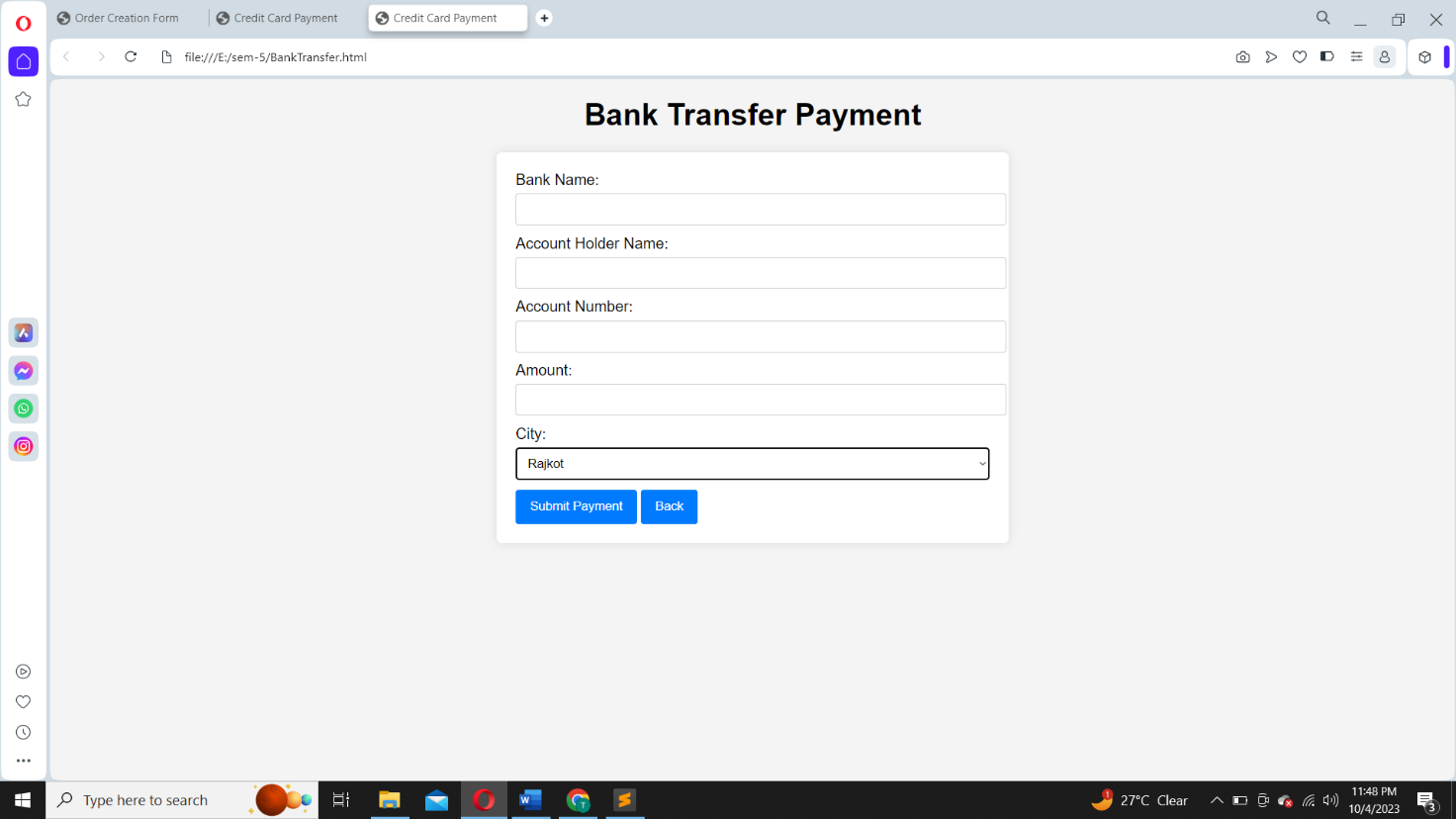


Figure ‑ Screen-3: Bank Transfer Payment

**Purpose:** transfer funds from one account to another for a specified transaction or financial exchange.

Table ‑ Screen element of Bank Transfer Payment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Bank Name | Textbox | M | 1 | Entre the full legal bank name of the beneficial bank , avoiding special character ,symbol |
| 2 | Account Holder’s Name | Textbox | M | 1 | Provide the complete legal name of the account holder as it appears on the bank account without special character. |
| 3 | Account Number | Textbox | M | 1 | Entre the valid account number associated with beneficially bank account for accurate process. |
| 4 | Amount | Textbox | M | 1 | Entre the excate payment amount ensure alignment with the order details as accurate process |
| 5 | City | Dropdown | M | 1 | Provide the city where the bank is located for accurate bank transfer process. |
| 6 | Submit Payment | Button | O | 1 | Display the message “your payment is successfully” |
| 7 | Back | Button | O | 1 | It navigates the previous screen |

# Database design

## List of Tables

* Customer
* Car
* Transaction
* Payment
* OrderDetails

Table ‑ Table: Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| CustomerID | int | NN | PK (Auto Increment) |  |
| FirstName | varchar(100) | NN |  |  |
| LastName | varchar(100) | NN |  |  |
| Email | Varchar(50) | AN |  |  |
| MobileNumber | varchar(50) | AN |  |  |
| Gender | Varchar(11) | AN |  |  |
| IDProof | Varchar(50) | NN |  |  |

Table ‑ Table: Car

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| CarID | int | NN | PK (Auto Increment) |  |
| CarName | varchar(100) | NN |  |  |
| ModelName | Varchar(100) | NN |  |  |
| PublishYear | DateTime | AN |  |  |
| VIN | Int | NN |  |  |
| Price | int | NN |  |  |
| Colour | Varchar(50) | NN |  |  |
| EngineSize | Varchar(50) | NN |  |  |
| Milage | Varchar(50) | AN |  |  |

Table ‑ Table: Transaction

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| TransactionID | Int | NN | PK (Auto Increment) |  |
| CarID | Int | NN | FK | Reference of Car table |
| CustomerID | Int | NN | FK | Reference of Customer table |
| TrasactionDate | Varchar(50) | NN |  |  |
| TrasactionAmount | Varchar(50) | NN |  |  |

Table ‑ Table: Payment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| PaymentID | int | NN | PK (Auto Increment) |  |
| TrasactionID | Int | NN | FK | Reference of transaction table |
| PaymentDate | DateTime | NN |  |  |
| PaymentMethod | Varchar(100) | NN |  |  |
| CardNumber | varchar(100) | NN |  |  |
| ExpiryDate | Varchar(100) | NN |  |  |

Table ‑5 Table: OrderDetails

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| OrderID | int | NN | PK (Auto Increment) |  |
| CustomerID | Int | NN | FK | Reference of Customer table |
| CarID | Int | NN | FK | Reference of Car table |
| OrderDate | Varchar(100) | NN |  |  |
| TotalAmount | varchar(100) | NN |  |  |
| PaymentStatus | Varchar(100) | NN |  |  |
| DeliveryAddress | Varchar(100) | NN |  |  |
| Status | Varchar(100) | NN |  |  |

# Stories and Scenario

## Story-1: Order Creation

|  |  |  |
| --- | --- | --- |
| *Story # S1* | : | As a customer involved in Car Order creation,  I want to place an order easily and accurately  So that a I can ensure error-free procurement process for the organization. |
| Priority | **:** | High |
| Estimate | **:** | XXL |
| Reason | **:** | Developing an efficient order creation feature is complex due to the need for multiple validation checks, pricing calculations, and inventory management in the procurement System |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.1* | : | Successful order creation |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The Customer selects a car model, chooses the desired colour, specifies optional features, and configures the car to their liking.  **Then** The system successfully creates the order, generate an order number,  And display a confirmation message with the order details. |

### Scenario# S1.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.2* | : | Invalid Order Quantity |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** The customer is on the “create new order” page  **When:**  The customer enters a negative quantity in the input field, which is not a valid input as order quantity cannot be negative  **Then** the system detected the negative quantity, display an error message (e.g., "Invalid Quantity: Please enter a valid quantity (greater than zero)"),  And prevent the order from being created. |

### Scenario# S1.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.3* | : | Missing Delivery Address |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given**: The customer is on the “Create New Order” Page.  **When**: The customer selects a model, entre a valid quantity, but leaves the delivery address field blank.  **Then:** The system detects the missing delivery address, display an error message (e.g., “Delivery Address required: please provide a valid delivery address”), and prevent the order from being created until a valid address is provided. |

## Story-2: Search

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a customer searching for car  I want to use a search field based on car name, model, colour  So that I could search particular car I want |
| Priority | **:** | High |
| Estimate | **:** | L |
| Reason | **:** | Implementing search functionality for order estimation to efficiently manage the car purchase management system |

### Scenario# S2.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.1* | : | Basic search |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The Customer initiates a basic car search by entering their desired criteria, such as make, model  **Then** The system displays a list of matching cars based on the provided criteria, allowing the customer to browse through the options |

### Scenario# S2.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.2* | : | Advance search |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The Customer initiates an advance car search by entering their desired criteria, such as make, model, year, price range, milage and additional features.  **Then** The system displays a refined list of matching cars that meet the detailed requirements, allowing the customer to explore these options. |

### Scenario# S2.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.3* | : | invalid search |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The Customer attempts to initiate an advanced car search but provides invalid or incomplete search criteria, such as missing make or model information, unrealistic price ranges, or contradictory features.  **Then:** The system detects the invalid search parameters and provides an error message or notification, informing the Customer about the issue. |

## Story-3: Payment

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As a customer,  I want to different payment option,  So that I can have different payment option for doing payment. |
| Priority | **:** | High |
| Estimate | **:** | XL |
| Reason | **:** | Payment option are a compulsory because a customer be comfortable with only one payment method. |

### Scenario# S3.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.1* | : | Credit card payment for car purchase |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The customer selects “credit card” as the payment method  **Then:** The system prompts the customer enter their credit card information, including card number, expiration date, CVV and billing address. |

### Scenario# S3.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.2* | : | Bank transfer |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The bank transfer is successfully completed.  **Then:** The system verifies the receipt of the payment by periodically checking the provided bank account. |

### Scenario# S3.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S3.3* | : | Invalid payment information |
| Prerequisite | **:** | The Customer is logged into the system |
| Acceptance Criteria | **:** | **Given:** A car buyer representative access the car website.  **When:** The customer enters invalid information, such as an incorrect card number, expiration date etc.  **Then:** The system detects the invalid information and display an error message indicating that the payment details are incorrect. |

## Story-4: Tracking

|  |  |  |
| --- | --- | --- |
| *Story # S4* | : | As a customer,  I want to track the status and locations of my shipments,  So that I can ensure timely delivers, manage expectations, and respond to any issues effectively. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Shipment tracking is a critical aspect of car purchase, ensuring that user have real-time visibility into their shipments progress. |

## Story-5: Notification

|  |  |  |
| --- | --- | --- |
| *Story # S5* | : | As a customer involved in car purchase activities,  I want to receive timely notification and updates regarding my shipments and orders  So that I can stay informed about the progress and any relevant changes in the car purchase process. |
| Priority | **:** | High |
| Estimate | **:** | L |
| Reason | **:** | Notification play a critical role in ensuring smooth and transparent communication throughout the car purchase process, helping customer manage their operation effectively. |

## Story-6: Order Details

|  |  |  |
| --- | --- | --- |
| *Story # S6* | : | As a customer purchasing car,  I want to provide order details  So that a secure and accurate order for specific car can be placed. |
| Priority | **:** | High |
| Estimate | **:** | XXL |
| Reason | **:** | Creating an order details page for car purchase is complex task, as it involves multiple variable such as car model, colour, financing options, and delivery preference in the car showroom management system |

## Story-7: Add to Cart

|  |  |  |
| --- | --- | --- |
| *Story # S7* | : | As a customer shopping for a car,  I want to add a car to my cart  So that I can securely reserve it for potential purchase. |
| Priority | **:** | Medium |
| Estimate | **:** | XL |
| Reason | **:** | Implementing an “add to cart” functionality for car purchase is a complex task because it involves managing various aspects like car model, optional features, pricing, financing option, and availability in the car showroom management system |

## Story-8: Add new Car

|  |  |  |
| --- | --- | --- |
| *Story # S8* | : | As an administrator managing a car showroom,  I want to add a new car to the inventory  So that I can securely list and manage the details of a new car in our showroom’s system . |
| Priority | **:** | High |
| Estimate | **:** | XXL |
| Reason | **:** | It involves gathering and inputting various details such as car make, model, year, colour, pricing, features, and availability in the car showroom management system. |

## Story-9: Update Car

|  |  |  |
| --- | --- | --- |
| *Story # S9* | : | As an administrator managing a car showroom,  I want to update car details in the inventory  So that I can securely modify and keep the car information up-to-date in car showroom system. |
| Priority | **:** | Medium |
| Estimate | **:** | XL |
| Reason | **:** | It involves ensuring involves verifying and altering various aspects such as car make, model, year, colour, pricing, features, and availability in the Car Showroom Management System while ensuring data accuracy and consistency. |

## Story-10: Feedback

|  |  |  |
| --- | --- | --- |
| *Story # S10* | : | As a customer visiting the car showroom,  I want to provide feedback about your showroom experience  So that I can share my thoughts and suggestions to help improve the car showroom management system. |
| Priority | **:** | Medium |
| Estimate | **:** | L |
| Reason | **:** | Collecting and managing feedback in the car showroom management system is an important task, as it involves capturing analyzing customer opinions, rating, and comments to enhance the overall customer experience and make informed improvements. |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | CarShowroom Management system | Test Designed by: | Tisha Mehta |
| Module Name: | Home Screen | Test Designed date: | 27/08/2023 |
| Release Version: | 1.0 | Test Executed by: | **R.B. Gondaliya** |
|  |  | Test Execution date: | 1/15/2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition The car purchase application is installed and running. | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | UI Display and Navigation Tests | Functional | navigate one screen to another screen | TC\_001 |
| TC\_002 | Search Functionality | GUI | display related to the search car | TC\_002 |
| TC\_003 | Add to cart | Functional | Verify that a product can be successfully added to the cart. | TC\_003 |

|  |  |
| --- | --- |
| **Test Case Title** | UI Display and Navigation Tests |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Open the car purchase application. | The application should open without errors. | If the application opens and displays the expected content and navigation options on the home screen, the test is considered successful. | pass |  |  |  |
| 2 | Display and Layout | show a list of car models with their names, images, and brief descriptions. | The home screen displays a scrollable list of car models with images, names, and descriptions. | pass |  |  |  |
| 3 | Navigation | The user should be redirected to the About page of the application. | The user is successfully redirected to the About page. | pass | The navigation from the home page to the About page worked as expected. The expected and actual results match. |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Search Functionality |
| **Test Type** | GUI |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Verify that a basic car search | The search results page should display a list of sedan cars available for sale. | The search results page displays a list of sedan cars available for sale. | pass | The basic car search functionality worked as expected. The expected and actual results match. |  |  |
| 2 | Verify that an advanced car search with specific criteria | The search results page should display a list of sedan cars with an automatic transmission, model year 2018 or newer, and priced at or below $40,000. | The search results page displays a list of sedan cars with automatic transmission, model year 2018 or newer, and priced at or below $40,000. | pass | The advanced car search functionality worked as expected. The expected and actual results match. |  |  |
| 3 | Verify that performing a search with invalid criteria returns appropriate feedback to the user. | The search results page should display a message indicating that no results were found for the invalid search query. | The search results page displays a message indicating that no results were found for the invalid search query. | pass | The system correctly provided feedback that no results were found for the invalid search query. |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Add to cart |
| **Test Type** | GUI |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Adding a Product to Cart | display the added product with the correct name, price, and quantity. | displays the added product with the correct name, price, and quantity. | pass | add to cart one car functionality worked as expected. The expected and actual results match. |  |  |
| 2 | Adding Multiple Products to Cart | The cart page should display multiple added products with correct names, prices, and quantities. | The cart page displays multiple added products with the correct names, prices, and quantities. | pass | add to cart multiple car functionality worked as expected. The expected and actual results match. |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | Car Showroom Management system | Test Designed by: | Tisha Mehta |
| Module Name: | Add car as admin side | Test Designed date: | 27/08/2023 |
| Release Version: | 1.0 | Test Executed by: | **R.B. Gondaliya** |
|  |  | Test Execution date: | 1/15/2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition The car purchase application is installed and running. | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Valid Car Addition | Functional | Verify that a car can be added with valid and complete information. | TC\_001 |
| TC\_002 | Invalid Information Car Addition | GUI | Test the system's response to invalid inputs | TC\_002 |
| TC\_003 | Duplicate Car Addition | GUI | Verify that the system prevents adding a duplicate car with the same make, model, year, and VIN. | TC\_003 |

|  |  |
| --- | --- |
| **Test Case Title** | Valid Car Addition |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Verify that a car can be added successfully with valid information. | After clicking the "Add" button, the system should display a success message indicating that the car has been added. The added car's details should be visible on the admin interface. | The system displays a success message, confirming that the car has been added. The added car's details are visible on the admin interface. | Pass | The test case confirms that the basic functionality of adding a car is working as expected. The added car's details are correctly stored and displayed, and the success message confirms successful addition. |  |  |
| 2 | Verify that a car can be added with optional features and that these features are correctly stored. | After clicking the "Add" button, the system should display a success message indicating that the car with optional features has been added. The added car's details, including the optional features, should be visible on the admin interface. | Actual Result: The system displays a success message, confirming that the car with optional features has been added. The added car's details, including the provided optional features, are visible on the admin interface. | pass | The test case verifies that the system correctly handles and stores optional features associated with the added car. The success message and the displayed details confirm that the process is successful |  |  |
| 3 | Verify that multiple cars can be added consecutively without issues | After each "Add" button click, the system should display a success message indicating that each car has been added. The details of all added cars should be visible on the admin interface. | The system displays a success message for each car added, confirming their successful addition. The details of all added cars are visible on the admin interface. | pass | The test case demonstrates that the system can handle the sequential addition of multiple cars without errors. The displayed success messages and the presence of all added cars' details confirm the successful addition process. |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Invalid Information Car Addition |
| **Test Type** | GUI |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Verify that adding a car with an invalid year format is handled appropriately | The system should display an error message indicating that the year format is invalid and prevent the car from being added. | The system displays an error message indicating that the year format is invalid, and the car is not added | pass | The test case successfully demonstrates that the system correctly identifies an invalid year format and prevents the car from being added. The displayed error message confirms that the validation process is working as expected. | e.g., "abcd" |  |
| 2 | Verify that adding a car with an invalid VIN format is handled appropriately. | The system should display an error message indicating that the VIN format is invalid and prevent the car from being added | The system displays an error message indicating that the VIN format is invalid, and the car is not added. | pass | The test case successfully demonstrates that the system correctly identifies an invalid VIN format and prevents the car from being added. The displayed error message confirms that the validation process is working as expected. | e.g., "12345" |  |
| 3 | Verify that adding a car with essential fields left blank is handled appropriately | The system should display error messages for the missing required fields and prevent the car from being added. | The system displays error messages for the missing required fields, and the car is not added. | pass | The test case successfully demonstrates that the system correctly identifies missing essential fields and prevents the car from being added. The displayed error messages confirm that the validation process is working as expected. | e.g. model, year |  |

|  |  |
| --- | --- |
| **Test Case Title** | Duplicate Car Addition |
| **Test Type** | GUI |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Verify that the system allows adding a car with a different VIN while having the same model and year as an existing car. | The system should successfully add the car in Attempt to add a car with the same model and year but a different VIN. | The system adds the car with a different VIN as expected. The details of both added cars should be present and correct. | pass | This test case ensures that the system accurately identifies duplicate cars based on the combination of model and year, while still allowing unique cars with different VINs. |  |  |
| 2 | Verify that the system considers make and model as case-insensitive when checking for duplicate cars. | The system should successfully add the car in Attempt to add a car with the same make and model but with different letter casing, as make and model are considered case-insensitive when checking for duplicates. | The system adds the car with different letter casing for make and model as expected. The details of both added cars should be present and correct. | pass | This test case ensures that the system correctly identifies duplicate cars regardless of the letter casing used for the make and model. It helps validate that the system's comparison logic is case-insensitive for these fields. |  |  |
| 3 | Verify that the system checks for duplicates even after successfully adding a car. | After successfully adding the car with valid details, attempting to add the same car details again should result in an error message indicating that the car is already exists. | The system displays an error message as expected when attempting to add a duplicate car, The details of the added car remain correct. | pass | This test case ensures that the system consistently checks for duplicate cars even after successfully adding one. It helps validate that the duplicate check functionality is working reliably. |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | Car Showroom Management system | Test Designed by: | Tisha Mehta |
| Module Name: | order creation | Test Designed date: | 27/08/2023 |
| Release Version: | 1.0 | Test Executed by: | **R.B. Gondaliya** |
|  |  | Test Execution date: | 1/15/2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pre-condition The car purchase application is installed and running. | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Enter valid quantity of car | Functional | Verify that the user can enter a valid quantity of grain | TC\_001 |
| TC\_002 | Access "Create New Order" from dashboard | Functional | Verify that the user can access the order creation form from the dashboard | TC\_002 |
| TC\_003 | Enter delivery address | Functional | Verify that the user can enter the delivery address in text input fields | TC\_003 |

|  |  |
| --- | --- |
| **Test Case Title** | Enter valid quantity of car |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the order creation form | The form is accessible | Form accessed successfully | Pass |  |  |  |
| 2 | Enter a valid quantity of grain | The quantity is accepted | Quantity accepted successfully | Pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Access "Create New Order" from dashboard |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the dashboard | The dashboard is accessible | Dashboard accessed successfully | Pass |  |  |  |
| 2 | Select "Create New Order" | The order creation form is displayed | Form displayed successfully | Pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Enter delivery address |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | The car purchase application is installed and running.  The user is logged in. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the order creation form | The form is accessible | Form accessed successfully | Pass |  |  |  |
| 2 | Enter a valid delivery address | The address is entered | Address entered successfully | Pass |  |  |  |
| 3 | Submit the form | The form is successfully submitted | Form submitted successfully | Pass |  |  |  |

# References

* <https://www.kia.com/in/our-vehicles/sonet/showroom.html>
* <https://www.w3schools.com/html/>
* <https://www.javatpoint.com/>