

Darshan University

A Project Report on

**“Super Market Management”**

Under the subject

**Software Engineering (2301CS405)**

B. Tech, Semester – IV

Computer Science & Engineering Department

|  |  |
| --- | --- |
| Submitted By | |
| Student Name:Smit Maru | Enrollment No.:23010101161 |
| Academic Year  (2024-2025) | |
| Internal Guide  Prof. R. B. Gondaliya  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** **(2301CS405)** for entitled **“Super Market Management”** submitted in partial fulfilment for the Semester-4 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 R. B. Gondaliya and that no part of any of report has been directly copied from any students’ reports, without providing due reference.

Smit Maru

Student’s Signature

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

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

**CERTIFICATE**

This is to certify that the SRS on **“Super Market Management ” has** been satisfactorily prepared by **Smit Maru** ( **23010101161** ) under my guidance in the fulfillment of the course **Software Engineering (2301CS405)** work during the academic year 2024-2025.

|  |  |  |
| --- | --- | --- |
| Internal Guide  Prof. R. B. Gondaliya  Darshan University |  | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

**Acknowledgement**

I wish to express my sincere gratitude to my project guide Prof. R. B. Gondaliyaand 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

**Smit Maru**

**ABSTRACT**

The Supermarket Management System is an advanced solution designed to streamline the operations of supermarkets while enhancing customer satisfaction. This system focuses on efficiently managing inventory, tracking sales, processing payments, and delivering products and services directly to customers' locations.

By integrating user-friendly interfaces and robust backend functionalities, the system provides real-time inventory updates, automated billing, and seamless order management. The primary objective is to minimize manual intervention, reduce operational costs, and ensure timely delivery of goods to customers.

The system also supports data analytics for better decision-making, helping businesses optimize stock levels, understand customer preferences, and improve overall efficiency. With its comprehensive design, the Supermarket Management System aims to revolutionize retail operations and provide a superior shopping experience.

The main purpose of this system is to minimize human efforts and automate supermarket operations effectively.

**Table of Contents**

[List of Figures I](#_Toc192281385)

[List of Tables II](#_Toc192281386)

[1 Introduction 1](#_Toc192281387)

[1.1 Product perspective 1](#_Toc192281388)

[1.2 Product features 1](#_Toc192281389)

[1.2.1 There are four different users who will be using this product: 1](#_Toc192281390)

[1.2.2 The features that are required for the Admin are: 1](#_Toc192281391)

[1.2.3 The features that are required for the Customer are: 1](#_Toc192281392)

[1.2.4 The features that are required for the Staff are: 2](#_Toc192281393)

[1.2.5 The features that are required for the Delivery boy are: 2](#_Toc192281394)

[1.3 Functional Requirement 2](#_Toc192281395)

[1.3.1 Admin 2](#_Toc192281396)

[1.3.2 Customer 2](#_Toc192281397)

[1.3.3 Staff 3](#_Toc192281398)

[1.3.4 Delivery Boy 3](#_Toc192281399)

[1.4 Non-Functional Requirement 3](#_Toc192281400)

[1.4.1 Usability: 3](#_Toc192281401)

[1.4.2 Accuracy: 3](#_Toc192281402)

[1.4.3 Availability: 3](#_Toc192281403)

[1.4.4 Maintainability: 3](#_Toc192281404)

[2 Design and Implementation Constraints 4](#_Toc192281405)

[2.1 Use case diagram 4](#_Toc192281406)

[2.2 Activity diagram and Swimlane diagram 5](#_Toc192281407)

[2.3 Sequence diagram 7](#_Toc192281408)

[2.4 State diagram 8](#_Toc192281409)

[2.5 Class diagram 9](#_Toc192281410)

[2.6 Data flow diagram 10](#_Toc192281411)

[2.6.1 Context diagram (level-0) 10](#_Toc192281412)

[2.6.2 DFD Level-1 10](#_Toc192281413)

[2.6.3 DFD Level-2 11](#_Toc192281414)

[3 External interface requirement (Screens) 12](#_Toc192281415)

[3.1 Screen-1: Manage Roles and Permission 12](#_Toc192281416)

[3.2 Screen-2: Customer Details Management 13](#_Toc192281417)

[3.3 Screen-3: Search Product 15](#_Toc192281418)

[3.4 Screen-4: Assist Customer 16](#_Toc192281419)

[3.5 Screen-5:View Delivery Address 17](#_Toc192281420)

[4 Database design 18](#_Toc192281421)

[4.1 List of Tables 18](#_Toc192281422)

[5 Stories and Scenario 20](#_Toc192281423)

[5.1 Story-1: Add New Product 20](#_Toc192281424)

[5.1.1 Scenario# S1.1 21](#_Toc192281425)

[5.1.2 Scenario# S1.2 21](#_Toc192281426)

[5.2 Story-2: Search Product 21](#_Toc192281427)

[5.2.1 Scenario#S2.1 21](#_Toc192281428)

[5.2.2 Scenario#S2.2 22](#_Toc192281429)

[5.3 Story-3 Payment 22](#_Toc192281430)

[5.3.1 Scenario#S3.1 22](#_Toc192281431)

[5.3.2 Scenario#S3.2 22](#_Toc192281432)

[6 Test cases 24](#_Toc192281433)

[**Pre-condition: User should be registered and logged in** 24](#_Toc192281434)

[Login with valid credentials 24](#_Toc192281435)

[Search for an existing product 24](#_Toc192281436)

[Make a successful payment 25](#_Toc192281437)

[Check order status update 25](#_Toc192281438)

[7 References 26](#_Toc192281439)

# List of Figures

[Figure 2.1‑1 Use case diagram for super market management system 3](#_Toc146706848)

[Figure 2.2‑1 Activity diagram for cancle order and replace order 4](#_Toc146706849)

[Figure 2.2‑2 Swimlane diagram for restore stock 4](#_Toc146706850)

[Figure 2.3‑1 Sequence diagram for Book Issue 5](#_Toc146706851)

[Figure 2.4‑1 State diagram of Book 6](#_Toc146706852)

[Figure 2.4‑2 State diagram for Librarian 6](#_Toc146706853)

[Figure 2.5‑1 Class diagram for Library management system 7](#_Toc146706854)

[Figure 2.6‑1 Context diagram for Library management system 8](#_Toc146706855)

[Figure 2.6‑2 DFD level-1 for Library management system 8](#_Toc146706856)

[Figure 2.6‑3 DFD level-2 for Issue book 9](#_Toc146706857)

[Figure 3.1‑1 Screen-1: Registration Form 10](#_Toc146706858)

[Figure 3.2‑1 Screen-2: Login Form 11](#_Toc146706859)

[Figure 3.3‑1 Screen-3: Add borrower book 12](#_Toc146706860)

# List of Tables

[Table 3.1‑1 Screen element of Registration form 10](#_Toc146706861)

[Table 3.2‑1 Screen element of Login form 11](#_Toc146706862)

[Table 3.3‑1 Screen element of Add borrower book 12](#_Toc146706863)

[Table 4.1‑1 Table: Book 13](#_Toc146706864)

[Table 4.1‑2 Table: Borrower 13](#_Toc146706865)

[Table 4.1‑3 Table: Staff 13](#_Toc146706866)

[Table 4.1‑4 Table: Student 13](#_Toc146706867)

# Introduction

## Product perspective

This project is aimed at transforming the manual supermarket operations into an internet-based application to enhance convenience and efficiency. The system allows users to browse products, place orders, and access account details, including past purchases and delivery status.

It is a multi-user version designed to handle the core functions of a supermarket, such as inventory management, billing, order processing, and delivery tracking. The system efficiently manages cataloging, stock replenishment, sales monitoring, and customer relationship management.

This application is capable of meeting the fundamental requirements of small to medium-sized supermarkets, ensuring streamlined operations and improved customer satisfaction.

## Product features

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

* Admin acts as the administrator, overseeing the operations.
* Customer purchases products from the supermarket.
* Staff manages store operations, assists with stock updates, and manages product delivery.
* Delivery delivers orders and updates delivery status.

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

* Admin can manage user roles and permissions to control access to various system features.
* Admin can manage the distribution of items to different sections or locations within the supermarket.
* Admin can monitor low-stock items and reorder products to maintain stock levels.
* Admin can create, edit, or remove product categories to organize inventory efficiently.
* Admin has the authority to manage and disburse salaries to staff based on roles and working hours.
* Admin can access transaction history, including sales, stock movements, and customer interactions, to track activities.
* Admin can maintain supplier details, manage contracts, and monitor supply chain operations.
* Admin can give commands or instructions to staff regarding stocking, pricing changes, or inventory updates.
* Admin can view, update, and manage details of staff members, including roles, working hours, and contact information.
* Admin can generate and view various reports, such as sales reports, inventory reports, and employee performance reports, to assess performance.
* Admin can access and view customer details, including purchase history, preferences, and contact information to improve customer service.

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

* Customer can search for specific products by name, brand, or category.
* Customer can add products to their shopping cart for later purchase.
* Customer can create and manage an account to store personal information, payment details, and order history.
* Customer can track the status of their orders, including shipping details and delivery times.
* Customer can make purchases by selecting products, applying discounts, and proceeding to checkout.
* Customer can choose from multiple payment options such as credit/debit cards and digital wallets.
* Customer can view their purchase history for quick reordering and reference.
* Customer can access customer support for inquiries, complaints, or product-related issues.
* Customer can request the supermarket to stock new products that are not available.
* Customer can leave reviews and ratings for products they’ve purchased.
* Customer can update their profile information, including contact details and shipping address.

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

* Staff can assist customers with product inquiries and purchases.
* Staff can update inventory details as products are sold or restocked.
* Staff can monitor and organize stock of perishable and non-perishable goods.
* Staff can help manage promotions, discounts, and in-store offers for customers.
* Staff can process returns and exchanges of products as per supermarket policies.
* Staff can report inventory needs or low-stock items to the Admin.
* Staff can help with packaging and organizing orders for delivery.
* Staff can manage the checkout process and assist customers during payment.
* Staff can ensure the cleanliness and proper arrangement of store shelves.
* Staff can assist in managing customer feedback and improving service quality.

### The features that are required for the Delivery boy are:

* Delivery boy can receive and view delivery orders assigned by the Admin or Staff.
* Delivery boy can update the delivery status (e.g., “out for delivery,” “delivered,” “failed delivery”).
* Delivery boy can verify customer identity upon delivery to ensure correct recipient.
* Delivery boy can report issues with deliveries, such as incorrect addresses or missing products.
* Delivery boy can access customer contact information for any delivery-related issues.
* Delivery boy can return undelivered products to the supermarket when necessary.
* Delivery boy can provide feedback on delivery conditions and improve efficiency.

## Functional Requirement

### Admin

* **Manage Roles and Permissions:** Admin can assign and modify user roles and permissions to control system feature access.
* **Distribute Items:** Admin can allocate items to specific sections or locations within the supermarket.
* **Monitor and Reorder Stock:** Admin can track low-stock items and reorder products to maintain inventory levels.
* **Manage Product Categories:** Admin can create, update, and delete product categories for efficient inventory organization.
* **Salary Management:** Admin can calculate and disburse salaries based on roles and working hours.
* **Transaction Monitoring:** Admin can view and analyze transaction history, including sales, stock movements, and customer interactions.
* **Supplier Management:** Admin can maintain supplier details, manage contracts, and oversee supply chain operations.
* **Issue Commands to Staff:** Admin can send instructions to staff about stock updates, pricing changes, or inventory adjustments.
* **Staff Management:** Admin can view, update, and manage staff details, such as roles, contact information, and working hours.
* **Generate Reports:** Admin can produce reports (e.g., sales, inventory, employee performance) for performance assessment.
* **Customer Details Management:** Admin can view customer details, including purchase history and preferences, for improved service.

### Customer

* **Search Products:** Customers can search for products by name, brand, or category.
* **Shopping Cart Management:** Customers can add, update, or remove items in their shopping cart.
* **Account Management:** Customers can create and manage an account for storing personal information, payment methods, and order history.
* **Order Tracking:** Customers can track the status and shipping details of their orders.
* **Purchase Products:** Customers can select items, apply discounts, and proceed to checkout.
* **Payment Options:** Customers can choose from multiple payment methods, including credit/debit cards and digital wallets.
* **View Purchase History:** Customers can review past orders for quick reordering.
* **Customer Support:** Customers can reach out for inquiries, complaints, or product-related issues.
* **Product Requests:** Customers can request the stocking of unavailable products.
* **Reviews and Ratings:** Customers can leave feedback on purchased products.

### Staff

* **Assist Customers:** Staff can help customers with product inquiries and purchases.
* **Update Inventory:** Staff can update inventory as items are sold or restocked.
* **Stock Monitoring:** Staff can organize and monitor perishable and non-perishable goods.
* **Promotions Management:** Staff can assist in managing in-store offers, promotions, and discounts.
* **Handle Returns:** Staff can process returns and exchanges as per supermarket policies.
* **Report Low Stock:** Staff can report inventory needs or low-stock items to the admin.
* **Order Preparation:** Staff can package and organize orders for delivery.
* **Checkout Assistance:** Staff can manage payments and assist customers during checkout.
* **Store Maintenance:** Staff can ensure cleanliness and proper arrangement of store shelves.
* **Feedback Management:** Staff can collect customer feedback to enhance service quality.

### Delivery Boy

* **View Delivery Orders:** Delivery personnel can access assigned orders from the admin or staff.
* **Update Delivery Status:** Delivery personnel can update statuses like “out for delivery,” “delivered,” or “failed delivery.”
* **Customer Verification:** Delivery personnel can verify customer identity upon delivery.
* **Report Issues:** Delivery personnel can report issues such as incorrect addresses or missing items.
* **Contact Customers:** Delivery personnel can access customer contact information for delivery-related issues.
* **Return Undelivered Items:** Delivery personnel can return undelivered items to the supermarket.
* **Provide Feedback:** Delivery personnel can give feedback on delivery conditions to improve operations.

## 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

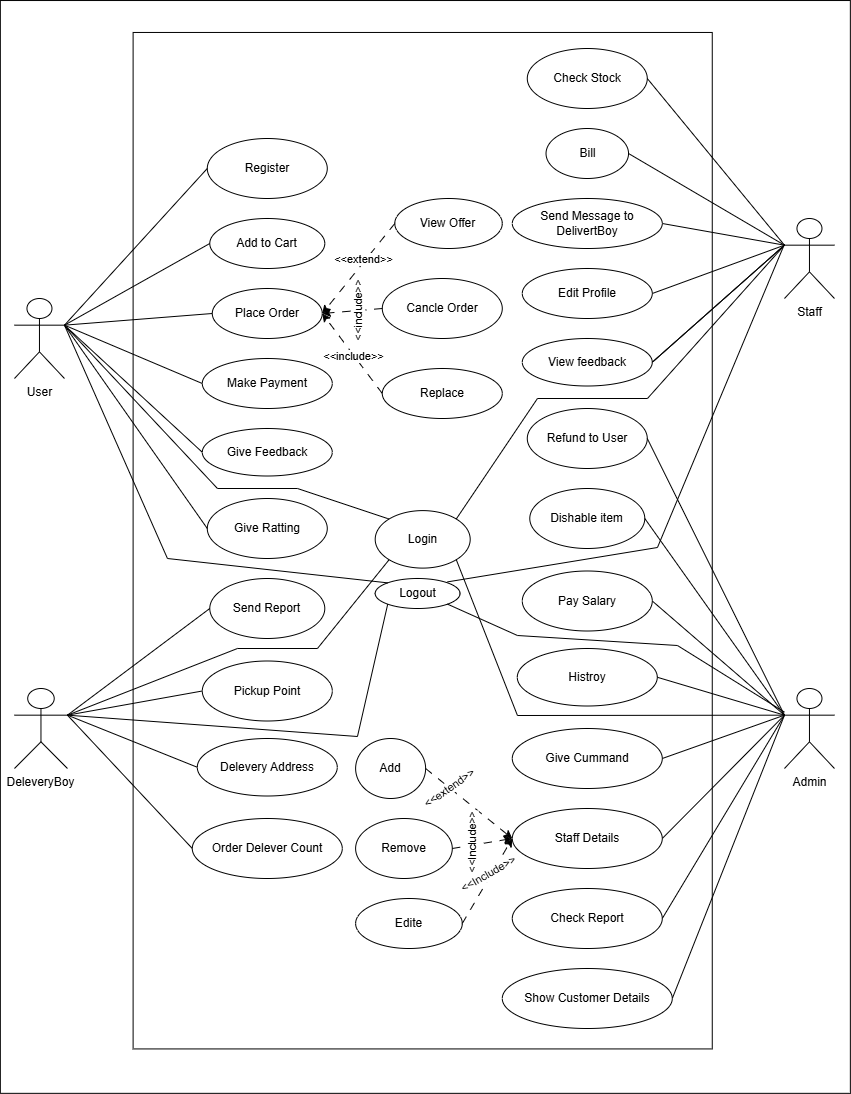


Figure 2.1‑1 Use case diagram for Super Market Management

## Activity diagram and Swimlane diagram

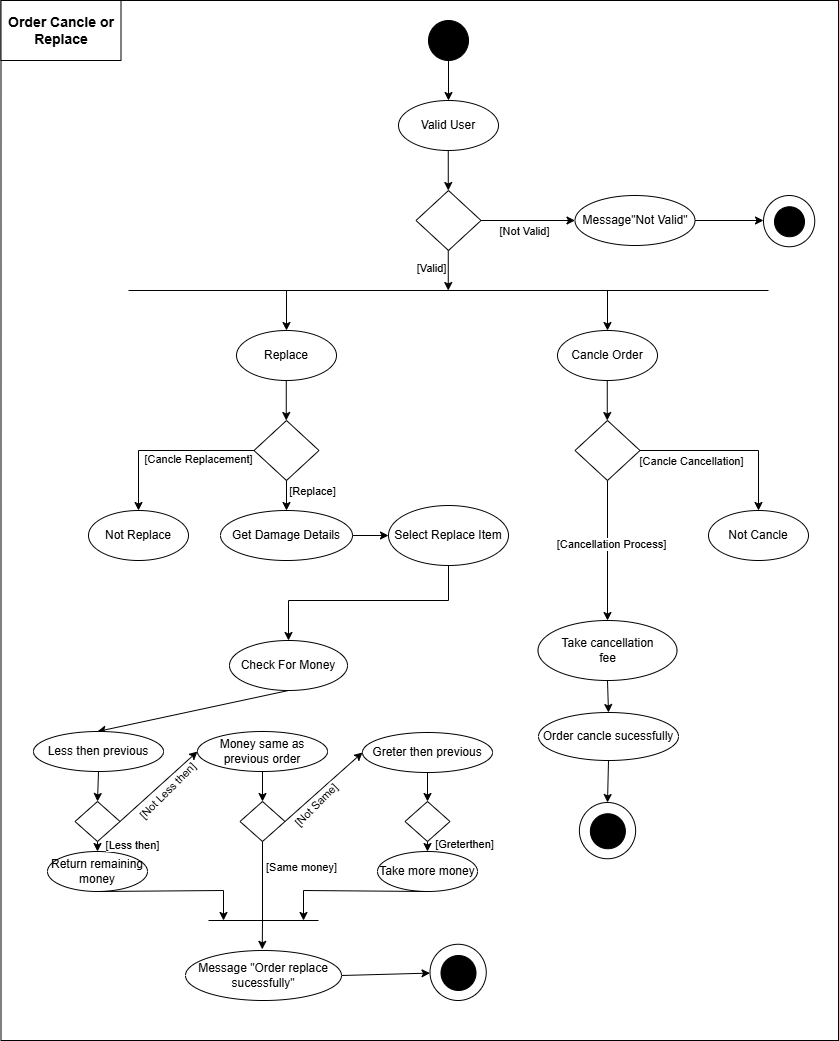


Figure 2.2‑1 Activity diagram for Order cancel or Replace

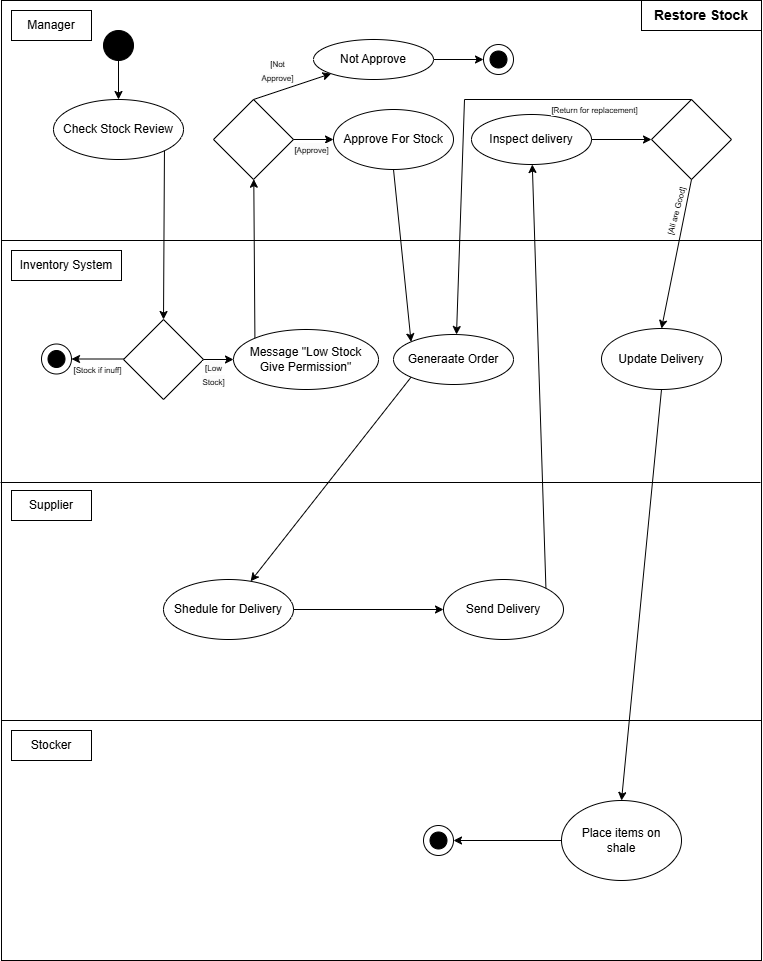


Figure 2.2‑2 Swimlane diagram for Restore Stock

## Sequence diagram

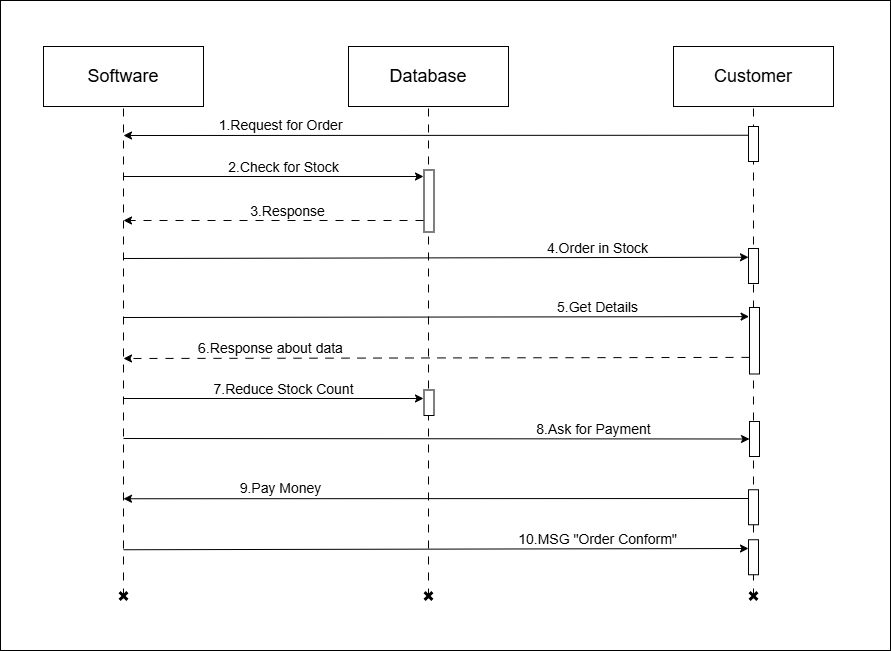


Figure 2.3‑1 Sequence diagram for Book Order

## State diagram

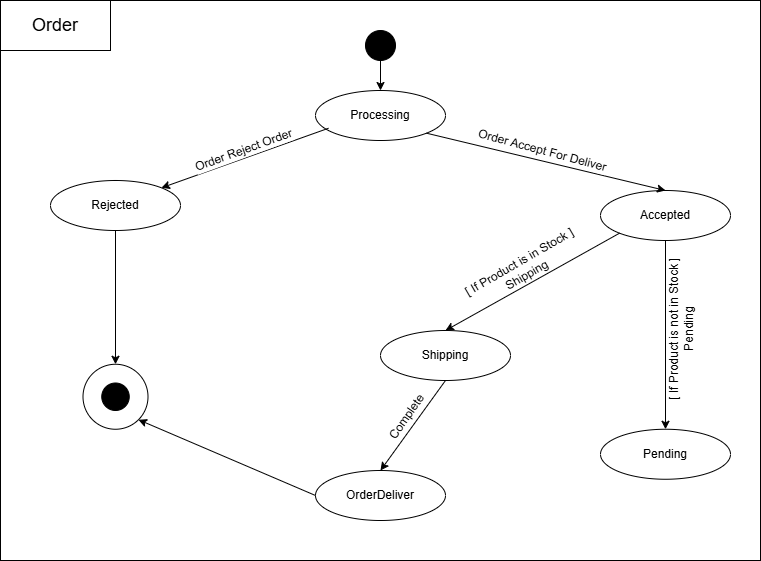


Figure 2.4‑1 State diagram of Order



Figure 2.4‑2 State diagram for Librarian

## Class diagram



Figure 2.5‑1 Class diagram for Library management system

## Data flow diagram

### Context diagram (level-0)

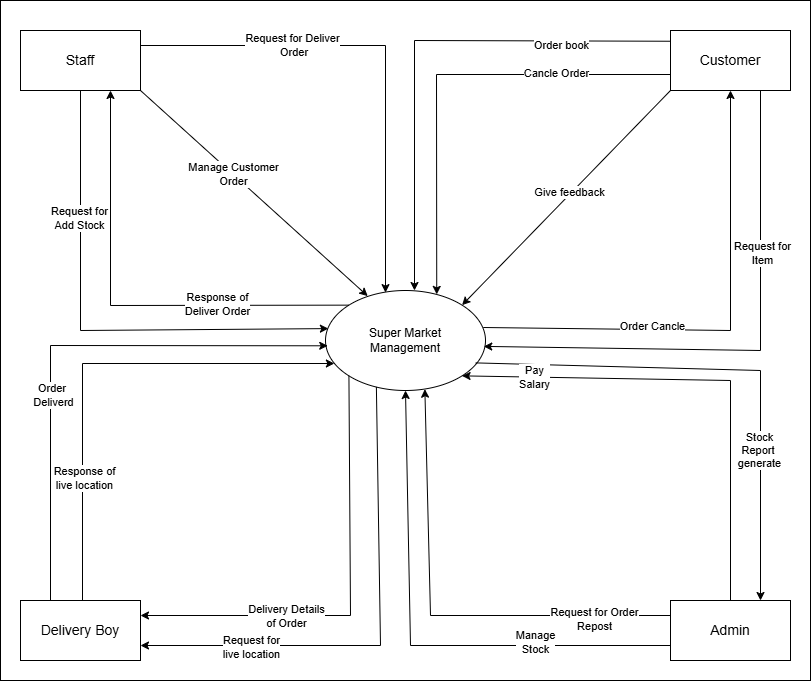


Figure 2.6‑1 Context diagram for Super Market Management

### DFD Level-1

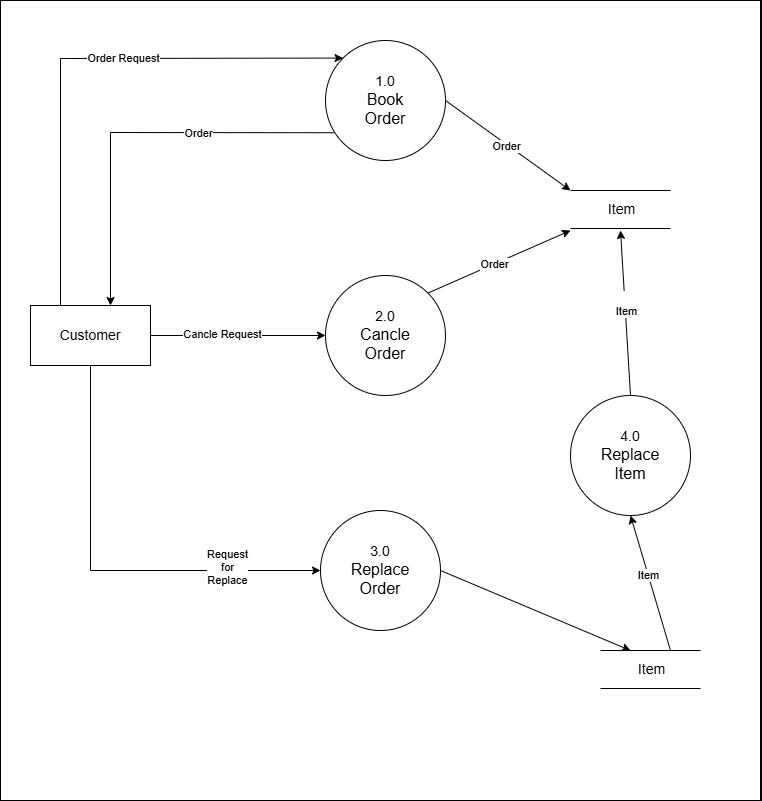


Figure 2.6‑2 DFD level-1 for Super Market Management

### DFD Level-2

Figure 2.6‑3 DFD level-2 for Issue book

# External interface requirement (Screens)

## Screen-1: Manage Roles and Permission

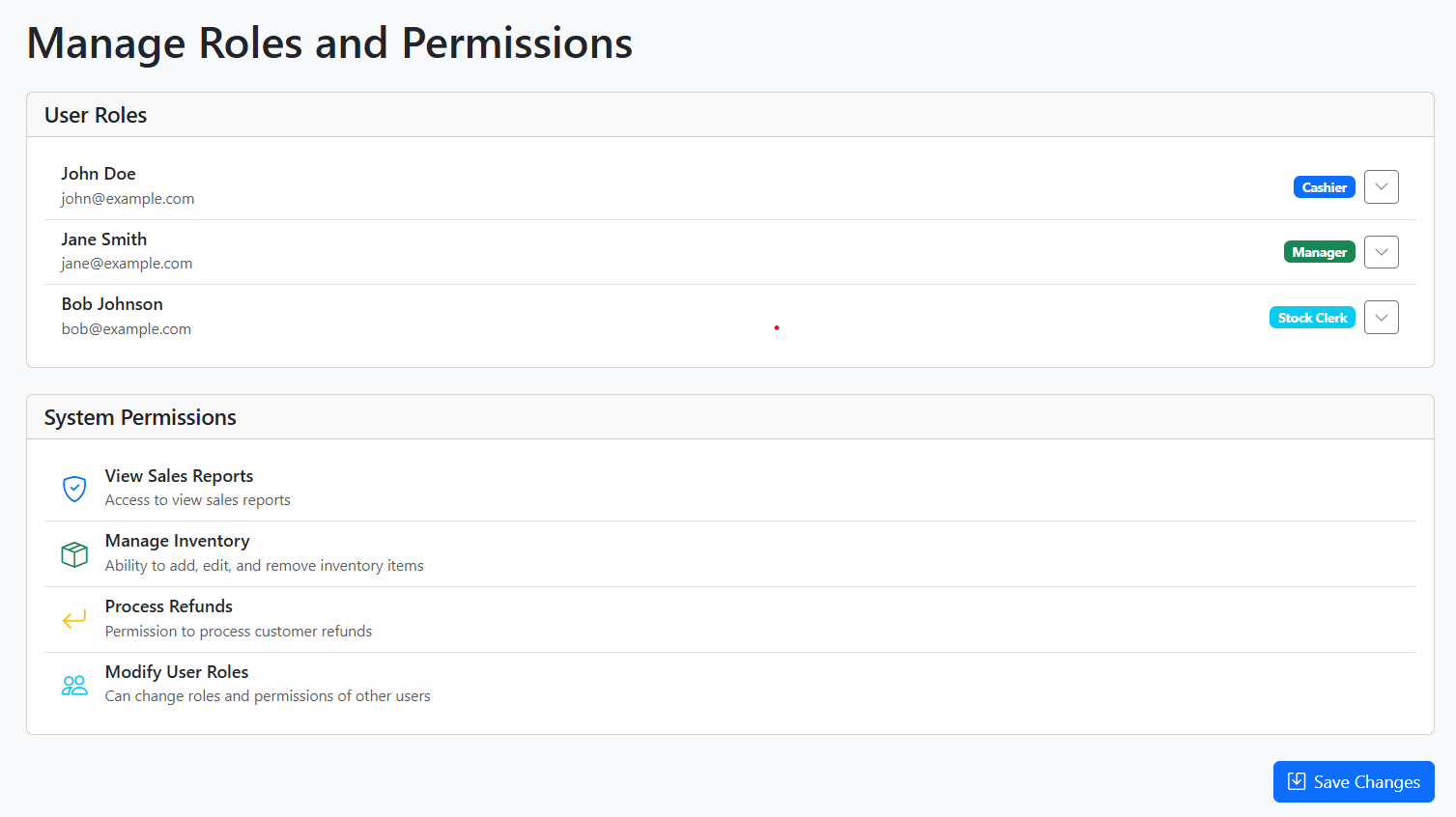


Figure 3.1‑1 Screen-1: Manage Roles and Permission

**Purpose:** This form will allow the users to their roles and responsibilities.

Table 3.1‑1 Screen element of Roles and Permission

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | |  | | --- | | User Roles List |  |  | | --- | |  | | Text | M | N | Displays a list of users with their assigned roles. |
| 2 | |  | | --- | | Role Selection Dropdown |  |  | | --- | |  | | Dropdown | M | 1 | |  | | --- | | Allows selecting a role for each user from predefined roles (e.g., Cashier, Manager, Stock Clerk). |  |  | | --- | |  | |
| 3 | |  | | --- | | System Permissions |  |  | | --- | |  | | |  | | --- | | Checkbox/List |  |  | | --- | |  | | M | N | |  | | --- | | Displays system permissions users can have, such as viewing sales reports, managing inventory, processing refunds, and modifying user roles. |  |  | | --- | |  | |
| 4 | |  | | --- | | Save Changes Button |  |  | | --- | |  | | Button | M | 1 | Saves any modifications made to user roles and permissions. |

## Screen-2: Customer Details Management

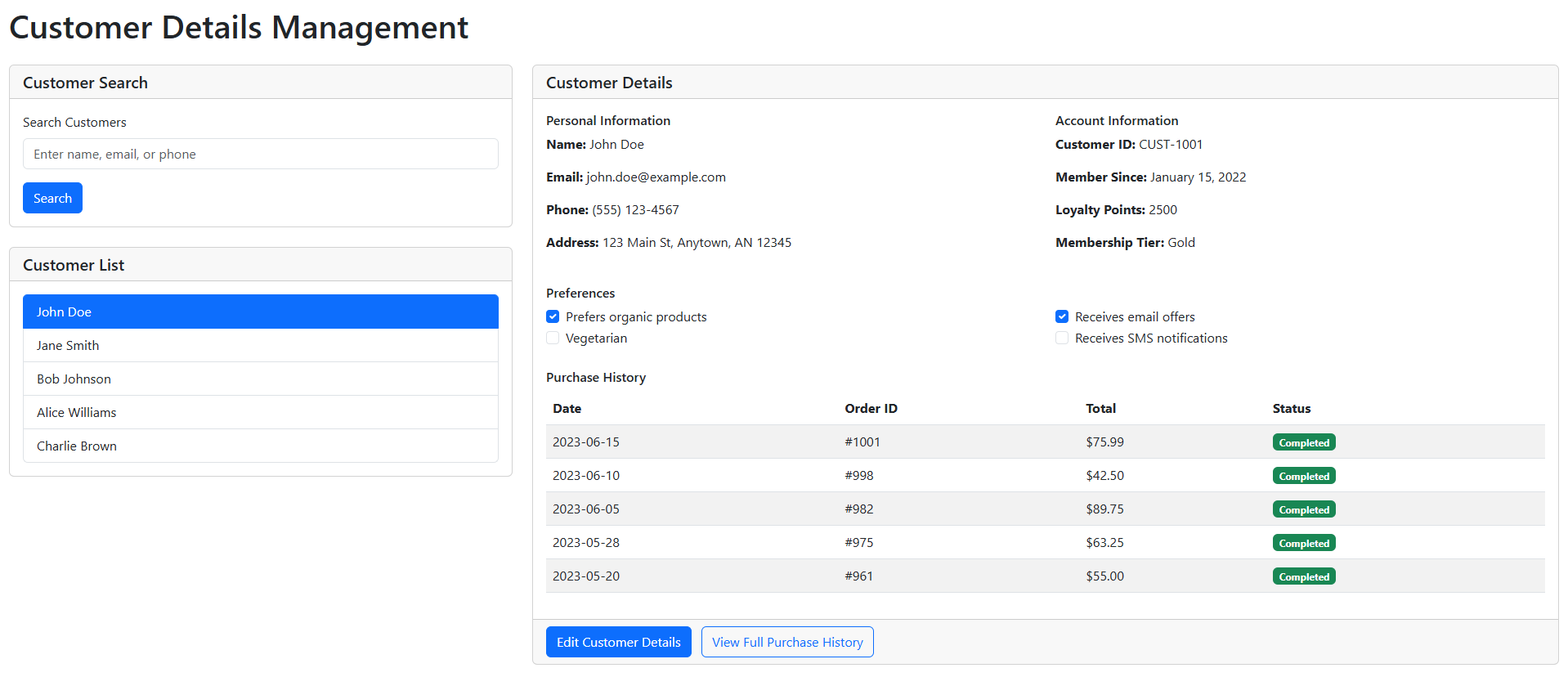


Figure 3.2‑1 Screen-2: Customer Details Management

**Purpose:** This form will be used by Admin so admin can manage it.

Table 3.2‑1 Screen element of Customer Details Management

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | |  | | --- | | Search Customers |  |  | | --- | |  | | Textbox | O | 1 | |  | | --- | | Allows users to search for customers by name, email, or phone. |  |  | | --- | |  | |
| 2 | |  | | --- | | Search Button |  |  | | --- | |  | | Button | O | 1 | |  | | --- | | Executes the search query based on the entered text. |  |  | | --- | |  | |
| 3 | |  | | --- | | Customer List |  |  | | --- | |  | | List | M | N | |  | | --- | | Displays a list of customers. Selecting a customer updates the "Customer Details" section. |  |  | | --- | |  | |
| 4 | |  | | --- | | Customer Name |  |  | | --- | |  | | Text | M | 1 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | Displays the selected customer's name. |  |  | | --- | |  | |  |  | | --- | |  | |  |  | | --- | |  | |
| 5 | |  | | --- | | Phone Number |  |  | | --- | |  | | Text | M | 1 | |  | | --- | | Displays the customer’s phone number. |  |  | | --- | |  | |
| 6 | Address | Text | M | 1 | |  | | --- | | Shows the customer's address details. |  |  | | --- | |  | |
| 7 | Email | Text | M | 1 | |  | | --- | | Shows the customer's email address. |  |  | | --- | |  | |
| 8 | |  | | --- | | Preferences | | |  | | --- | | Checkbox | | |  | | --- | | O | | |  | | --- | | N | | |  |  |  | | --- | --- | --- | | |  | | --- | | Allows selecting customer preferences like "Prefers organic products" and "Vegetarian." |  |  | | --- | |  | | |
| 9 | |  | | --- | | Customer ID |  |  | | --- | |  | | Text | M | 1 | |  | | --- | | Displays the unique identifier assigned to the customer. |  |  | | --- | |  | |
| 10 | |  | | --- | | Member Since |  |  | | --- | |  | | Text | M | 1 | |  | | --- | | Indicates the date when the customer joined. |  |  | | --- | |  | |
| 11 | |  | | --- | | Loyalty Points |  |  | | --- | |  | | Text | M | 1 | |  | | --- | | Displays the total loyalty points of the customer. |  |  | | --- | |  | |
| 12 | |  | | --- | | Membership Tier |  |  | | --- | |  | | Text | M | 1 | |  | | --- | | Indicates the customer’s membership level (e.g., Gold, Silver). |  |  | | --- | |  | |
| 13 | |  | | --- | | Email Offers Checkbox |  |  | | --- | |  | | Checkbox | O | 1 | |  | | --- | | Allows enabling/disabling email notifications for the customer. |  |  | | --- | |  | |
| 14 | |  | | --- | | SMS Notifications Checkbox |  |  | | --- | |  | | Checkbox | O | 1 | |  | | --- | | Allows enabling/disabling SMS notifications. |  |  | | --- | |  | |
| 15 | |  | | --- | | Purchase History Table |  |  | | --- | |  | | Table | M | N | |  | | --- | | Displays past transactions, including Date, Order ID, Total, and Status. |  |  | | --- | |  | |
| 16 | |  | | --- | | Edit Customer Details |  |  | | --- | |  | | Button | O | 1 | |  | | --- | | Opens a form to edit the selected customer’s details. |  |  | | --- | |  | |
| 17 | |  | | --- | | View Full Purchase History |  |  | | --- | |  | | Button | O | 1 | Shows the complete purchase history of the customer. |

## Screen-3: Search Product

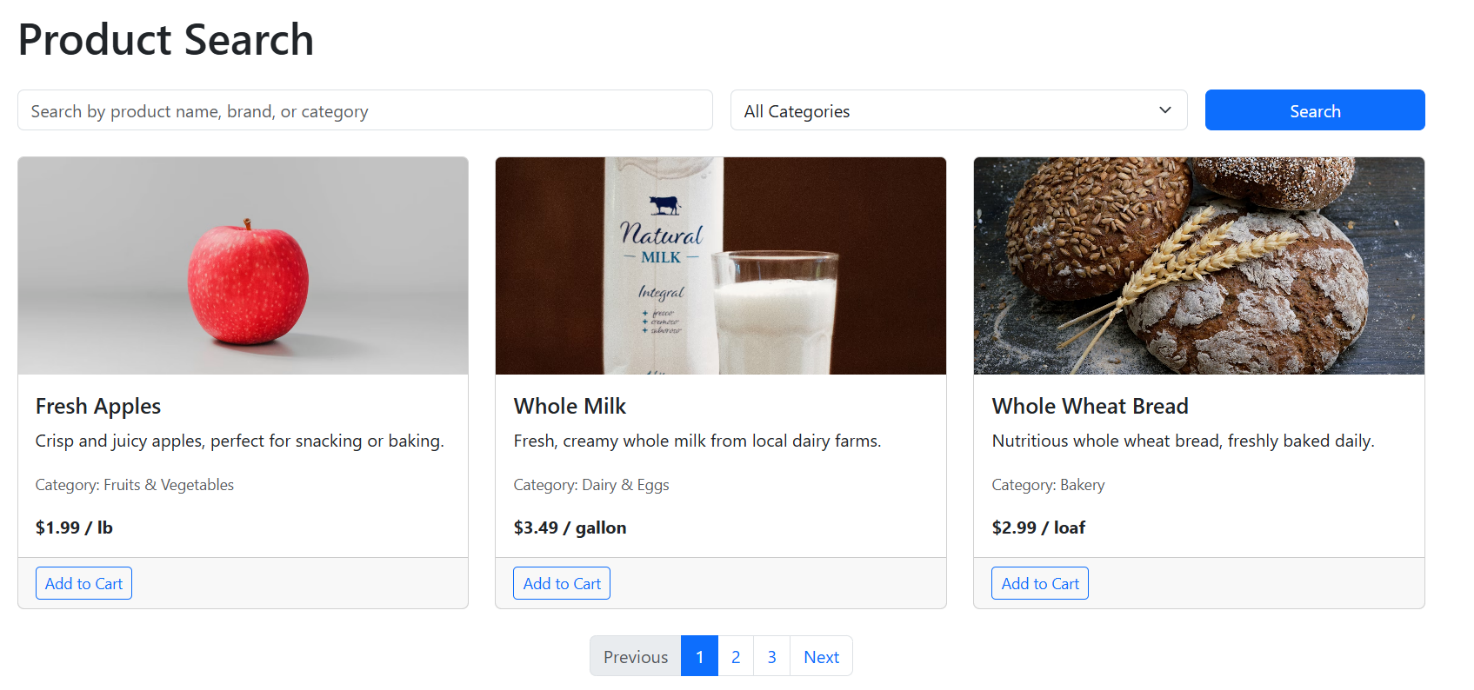


Figure 3.3‑1 Screen-3: Search Product

**Purpose:** This form will be useful for customer for search the product and find the product.

Table 3.3‑1 Screen element of Search Product Screen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Search Input | Textbox | O | 1 | Search field should be editable and accept product name, brand, or category keywords |
| 2 | Category Select | Dropdown | O | 1 | Category dropdown should allow selection from predefined categories (Fruits & Vegetables, Dairy & Eggs, Bakery, Beverages) |
| 3 | Search Button | Button | - | 1 | Button to trigger the search based on input criteria |
| 4 | Product Cards | Card Display | - | N | Display area for product information including image, title, description, category, and price |
| 5 | Product Image | Image | M | 1 | Product image should be displayed in consistent dimensions (width: 100%, height: 200px, object-fit: cover) |
| 6 | Product Title | Text | M | 1 | Product name should be displayed prominently in the card |
| 7 | Product Description | Text | M | 1 | Brief description of the product |
| 8 | Product Category | Text | M | 1 | Category label for the product |
| 9 | Product Price | Text | M | 1 | Price display with appropriate unit (per lb, gallon, loaf, etc.) |
| 10 | Add to Cart Button | Button | - | 1 | Button to add the product to shopping cart |
| 11 | Pagination | Navigation | - | 1 | Navigation controls for moving between pages of search results |

## Screen-4: Assist Customer

A screenshot of a computer

Description automatically generated

Figure 3.4-1 Screen-4: Assist Customers

**Purpose:** This module will help for assist to customers by staff.

Table 3.4‑1 Screen element of Add borrower book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Product Search Bar | Textbox | O | 1 | Search field for finding products with a search button |
| 2 | Product Table | Table | - | 1 | Display table with columns for Image, Name, Price, Stock, Actions |
| 3 | Stock Status | Label | M | 1 | Shows product availability (In Stock, Low Stock, Out of Stock) |
| 4 | Add to Cart Button | Button | - | 1 | Button to add product to customer's cart |
| 5 | Customer Name | Textbox | M | 1 | Field for entering customer's name |
| 6 | Customer Phone | Textbox | M | 1 | Field for entering customer's phone number |
| 7 | Customer Email | Textbox | M | 1 | Field for entering customer's email address |
| 8 | Notes | Textarea | O | 1 | Field for additional notes about the customer |
| 9 | Cart Table | Table | - | 1 | Shows selected products with columns for Product, Quantity, Price, Total, Actions |
| 10 | Remove Button | Button | - | N | Button to remove items from cart |
| 11 | Cart Total | Text | - | 1 | Displays total amount of items in cart |
| 12 | Proceed to Checkout | Button | - | 1 | Button to proceed with purchase |
| 13 | Check Product Availability | Button | - | 1 | Quick action to check product stock status |
| 14 | Apply Discount | Button | - | 1 | Quick action to apply discount to purchase |
| 15 | Process Refund | Button | - | 1 | Quick action to process a refund |
| 16 | Create Loyalty Account | Button | - | 1 | Quick action to create customer loyalty account |

## Screen-5:View Delivery Address

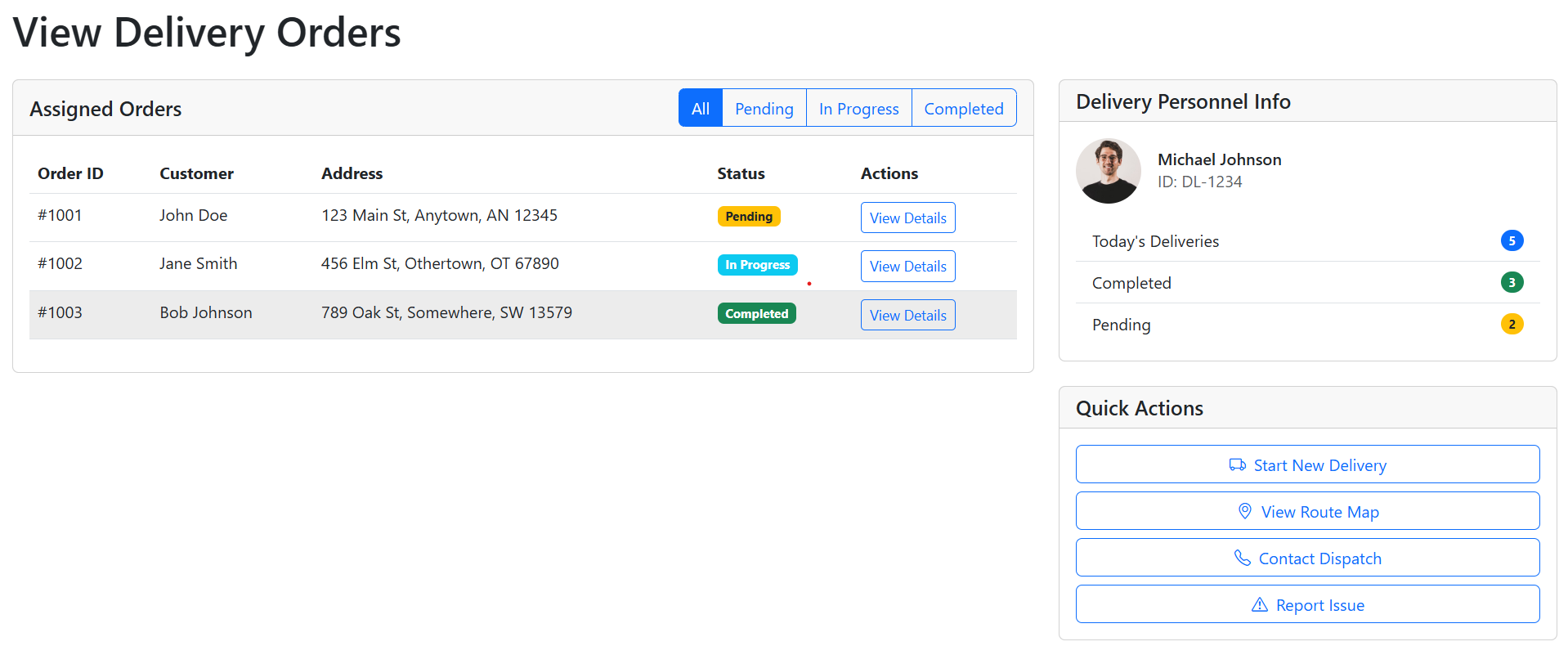


Figure 3.5-1 Screen-3: Assist Customers

**Purpose:** This module will help for assist to customers by staff.

Table 3.5‑1 Screen element of Add borrower book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Page Title | Text | M | 1 | "View Delivery Orders" heading at the top of the page |
| 2 | Order Filter Tabs | Button Group | - | 1 | Tabs to filter orders by All/Pending/In Progress/Completed |
| 3 | Orders Table | Table | - | 1 | Table showing order details with columns: Order ID, Customer, Address, Status, Actions |
| 4 | Order Status Badge | Label | M | 1 | Color-coded status indicators (Pending-yellow, In Progress-blue, Completed-green) |
| 5 | View Details Button | Button | - | N | Button in each row to view detailed order information |
| 6 | Delivery Personnel Section | Card | - | 1 | Section showing delivery person's information |
| 7 | Personnel Photo | Image | O | 1 | Circular profile photo of delivery personnel |
| 8 | Personnel Name | Text | M | 1 | Name of the delivery person |
| 9 | Personnel ID | Text | M | 1 | Unique ID of delivery person (format: DL-XXXX) |
| 10 | Delivery Stats | List | - | 1 | List showing Today's Deliveries, Completed, and Pending counts with badges |
| 11 | Quick Actions Section | Card | - | 1 | Section containing quick action buttons |
| 12 | Start New Delivery | Button | - | 1 | Button to initiate new delivery |
| 13 | View Route Map | Button | - | 1 | Button to view delivery route map |
| 14 | Contact Dispatch | Button | - | 1 | Button to contact dispatch |
| 15 | Report Issue | Button | - | 1 | Button to report delivery issues |

# Database design

## List of Tables

* User
* Products
* Orders
* OrderDetails
* Payments
* Suppliers
* Deliveries
* Inventory

Table 4.1‑1 Table: User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **UserID** | INT | NN | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for users |
| **FullName** | VARCHAR(100) | |  | | --- | | NN | | NOT NULL | Full name of the user |
| **Email** | VARCHAR(100) | NN | UNIQUE, NOT NULL | Email address |
| **Phone** | VARCHAR(15) | |  | | --- | | NN | | NOT NULL | Contact number |
| **Password** | VARCHAR(255) | NN | NOT NULL | Encrypted password |
| **Role** | ENUM('Admin', 'Staff', 'Delivery', 'Customer') | |  | | --- | | NN | | NOT NULL | Defines the user role |
| **CreatedAt** | TIMESTAMP | NN | DEFAULT CURRENT\_TIMESTAMP | Account creation timestamp |

Table 4.1‑2 Table: Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **ProductID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for products |
| **Name** | VARCHAR(255) | NO | NOT NULL | Product name |
| **Category** | VARCHAR(100) | NO | NOT NULL | Product category |
| **Price** | DECIMAL(10,2) | NO | NOT NULL | Price of the product |
| **Stock** | INT | NO | NOT NULL | Available quantity in stock |
| **SupplierID** | INT | YES | FOREIGN KEY (references Suppliers table) | Supplier reference |
| **CreatedAt** | TIMESTAMP | NO | DEFAULT CURRENT\_TIMESTAMP | Timestamp of product addition |

Table 4.1‑3 Table: Order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **OrderID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for orders |
| **CustomerID** | INT | NO | FOREIGN KEY (references Users) | Customer placing the order |
| **OrderDate** | TIMESTAMP | NO | DEFAULT CURRENT\_TIMESTAMP | Date when the order was placed |
| **TotalAmount** | DECIMAL(10,2) | NO | NOT NULL | Total cost of the order |
| **OrderStatus** | ENUM('Pending', 'Shipped', 'Delivered', 'Cancelled') | NO | NOT NULL | Status of the order |

Table 4.1‑4 Table: OrderDetails

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column Name | Data Type | NULL | Constraints | Description |
| DetailID | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for order details |
| OrderID | INT | NO | FOREIGN KEY (references Orders) | Order reference |
| ProductID | INT | NO | FOREIGN KEY (references Products) | Product being ordered |
| Quantity | INT | NO | NOT NULL | Number of units ordered |
| Price | DECIMAL(10,2) | NO | NOT NULL | Price per unit |

Table 4.1‑5 Table: Payment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **PaymentID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for payments |
| **OrderID** | INT | NO | FOREIGN KEY (references Orders) | Associated order |
| **PaymentDate** | TIMESTAMP | NO | DEFAULT CURRENT\_TIMESTAMP | Date of payment |
| **AmountPaid** | DECIMAL(10,2) | NO | NOT NULL | Amount paid |
| **PaymentMode** | ENUM('Cash', 'Card', 'Online') | NO | NOT NULL | Mode of payment |

Table 4.1‑6 Table: Supplier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **SupplierID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for suppliers |
| **Name** | VARCHAR(255) | NO | NOT NULL | Supplier name |
| **Contact** | VARCHAR(100) | YES | NULL allowed | Contact details |

Table 4.1‑7 Table: Deliveries

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **DeliveryID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for deliveries |
| **OrderID** | INT | NO | FOREIGN KEY (references Orders) | Order being delivered |
| **DeliveryBoyID** | INT | YES | FOREIGN KEY (references Users) | Assigned delivery personnel |
| **DeliveryStatus** | ENUM('Pending', 'Out for Delivery', 'Delivered') | NO | NOT NULL | Status of delivery |
| **DeliveryDate** | TIMESTAMP | YES | DEFAULT CURRENT\_TIMESTAMP | Date of delivery |

Table 4.1‑8 Table: Inventory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **NULL** | **Constraints** | **Description** |
| **InventoryID** | INT | NO | PRIMARY KEY, AUTO\_INCREMENT | Unique identifier for inventory |
| **ProductID** | INT | NO | FOREIGN KEY (references Products) | Product reference |
| **Quantity** | INT | NO | NOT NULL | Quantity in inventory |
| **LastUpdated** | TIMESTAMP | YES | DEFAULT CURRENT\_TIMESTAMP | Last stock update |

# Stories and Scenario

## Story-1: Add New Product

|  |  |  |
| --- | --- | --- |
| *Story # S1* | : | As a Admin,  I want to add a new product in the stock.  So that everyone customer can easily buy it. |
| Priority | **:** | High |
| Estimate | **:** | XL |
| Reason | **:** | The addition of a new product in the stock is crucial for ensuring that the product collection is up-to-date and accessible to every customer. |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.1* | : | Adding a New Product with Valid Information |
| Prerequisite | **:** | Admin is logged in to the super market management system. |
| Acceptance Criteria | **:** | **Given:**  The Admin is navigated to the Product management page. Valid product information, including price, brand and other relevant details is added.  **When:**  The Admin selects the "Add New Products" option  And The Admin enters valid product details  The Admin clicks the "Save" button to add the product in the stock.  **Then t**he system successfully adds the product in the stock and the admin receives a confirmation message with the product’s identification number. |

### Scenario# S1.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.2* | : | Adding a New Product with Invalid Information. |
| Prerequisite | **:** | The Admin is logged into the super market management system. |
| Acceptance Criteria | **:** | **Given:** The admin is on the product stock management page  **When:** The librarian selects the "Add New product" option and the admin enters an incomplete or incorrect product details and admin clicks the "Save" button to add the product in the stock.  **Then t**he system displays error messages for the incorrect or missing information and the product is not added in the stock. |

## Story-2: Search Product

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a Customer or Admin,  I want to search for product by name, brand, or price,  So that I can quickly find product that match my interests. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Implementing a search functionality is essential for enhancing the user experience, as it allows Admin and Customer to efficiently discover and access the products. |

### Scenario#S2.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.1* | : | Searching a Product by valid information. |
| Prerequisite | **:** | The Admin or customer is logged into the super market management system. |
| Acceptance Criteria | **:** | **Given:** The admin or customer is on the product stock management page  **When:** The admin or customer selects the "Search Product" option and the admin or customer enters an complete product details and admin or customer clicks the "Search" button to search the product in the stock.  **Then t**he system successfully display the product on the screen to the admin or customer. |

### Scenario#S2.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.2* | : | Searching a Product by invalid information. |
| Prerequisite | **:** | The Admin or customer is logged into the super market management system. |
| Acceptance Criteria | **:** | **Given:** The admin or customer is on the product stock management page  **When:** The admin or customer selects the "Search Product" option and the admin or customer enters an incomplete product details and admin or customer clicks the "Search" button to search the product in the stock.  **Then t**he system displays not such result found messages for the incorrect or missing information and the product is not dis[lay on the screen. |

## Story-3 Payment

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a Customer,  I want to Pay product bill,  So that I can quickly pay bill so that I can by this product. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Implementing a payment functionality is essential for enhancing the user experience, as it allows Customer to efficiently buy and access the products. |

### Scenario#S3.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.1* | : | Payment by valid information. |
| Prerequisite | **:** | The customer is logged into the super market management system. |
| Acceptance Criteria | **:** | **Given:** The customer is on the product payment page  **When:** The customer selects the "Payment" option and the customer enters an complete payment details and customer clicks the "Pay" button to pay bill of order.  **Then t**he system successfully display the order conformation message on the screen to the customer screen. |

### Scenario#S3.2

|  |  |  |
| --- | --- | --- |
| *Scenario# S2.2* | : | Payment by invalid information. |
| Prerequisite | **:** | The customer is logged into the super market management system. |
| Acceptance Criteria | **:** | **Given:** The customer is on the product payment page  **When:** The customer selects the "Payment" option and customer enters an incomplete payment details and customer clicks the "Pay" button to pay order bill.  **Then t**he system displays invalid information messages for the Ordre Payment. |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Project Name: | EMI Calculator | Test Designed by: | P. U. Jadeja |
| Module Name: | **Login** | **Test Designed date:** | 01-10-2023 |
| Release Version: | **1.0** | **Test Executed by:** | **R. B. Gondaliya** |
|  |  | **Test Execution date:** | 15-01-2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pre-condition: User should be registered and logged in** | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Login with valid credentials | Functional | Login to Supermarket system with valid credentials | TC\_001 |
| TC\_002 | Login with invalid credentials | Functional | Login to Supermarket system with invalid credentials | TC\_002 |
| TC\_003 | Search for an existing product | Functional | User searches for a product in inventory | TC\_003 |
| TC\_004 | Make a successful payment | Functional | User completes payment using a valid card | TC\_004 |

|  |  |
| --- | --- |
| **Test Case Title** | Login with valid credentials |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | User must have a registered account |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Open the login page | Login page should load | Page loaded | Pass |  |  |  |
| 2 | Enter valid username | Username is accepted | Accepted | Pass |  | test\_user |  |
| 3 | Enter valid password | Password is accepted | Accepted | Pass |  | \*\*\*\*\*\*\* |  |
| 4 | Click login button | User is redirected to dashboard | Redirected | Pass |  |  |  |

|  |  |
| --- | --- |
| Test Case Title | Search for an existing product |
| Test Type | Functional |
| Test Priority | Medium |
| Pre-condition | User must be logged in and products should exist in the database |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Step | Test Case Description | Expected Result | Actual Result | Status | Comment | Data | Bug ID |
| 1 | Navigate to search page | Search page should load | Loaded | Pass |  |  |  |
| 2 | Enter product name in search bar | Product should be found | Found | Pass |  | Milk |  |
| 3 | Click search button | Product list should display | Displayed | Pass |  |  |  |
| Test Step | Test Case Description | Expected Result | Actual Result | Status | Comment | Data | Bug ID |

|  |  |
| --- | --- |
| Test Case Title | Make a successful payment |
| Test Type | Functional |
| Test Priority | High |
| Pre-condition | User must have items in the cart and valid payment details |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Step | Test Case Description | Expected Result | Actual Result | Status | Comment | Data | Bug ID |
| 1 | Proceed to checkout | Checkout page should load | Loaded | Pass |  |  |  |
| 2 | Enter valid payment details | Payment details accepted | Accepted | Pass |  | Visa \*\*\*\*1234 |  |
| 3 | Click pay button | Payment should be successful | Successful | Pass |  |  |  |

|  |  |
| --- | --- |
| Test Case Title | Check order status update |
| Test Type | Functional |
| Test Priority | Medium |
| Pre-condition | User must have completed a purchase |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Go to order history | Order list should appear | Displayed | Pass | - | - | - |
| 2 | Check latest order | Status should be 'Confirmed' | Status remains 'Pending' | Fail | Delayed update | - | BUG\_001 |

# References

* http://www.w3schools.com/html/html\_intro.asp
* https://www.w3schools.com/php/default.asp
* https://www.javatpoint.com/uml