

Darshan University

A Project Report on

**“Online Shopping System”**

Under the subject

**Software Engineering (2301CS405)**

B. Tech, Semester – IV

Computer Science & Engineering Department

|  |  |
| --- | --- |
| Submitted By | |
| Student Name: Aayushi Sata | Enrollment No.: 23010101245 |
| 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 **“Online Shopping System”** 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.

Aayushi Sata

Student’s Signature

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

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

**CERTIFICATE**

This is to certify that the SRS on **“Online Shopping System” has** been satisfactorily prepared by **Aayushi Sata** (**23010101245**) 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

**Aayushi Sata**

**ABSTRACT**

An online shopping Management system is a computerized platform that facilitates customers to browse and purchase products online. This system transforms the traditional shopping experience into a digital one. The project comes with several features that are typically not available in traditional shopping systems, such as user login, easy product search, and instant order placement. Additionally, it has an admin login feature, which allows the admin to oversee the entire system.

The system is equipped with basic functionalities like allowing the admin to add, view, update, and delete products and customer information. Customers, after logging into their accounts, can view the list of products, their prices, and availability. They can also track their orders, view order history, and place requests for out-of-stock items using a request form. The admin, upon logging into their account, can generate various reports such as customer reports, order reports, product reports, and sales reports.

Main purpose of this system is to reduce human efforts as much as possible.

**Table of Contents**

[List of Figures I](#_Toc146706805)

[List of Tables II](#_Toc146706806)

[1 Introduction 1](#_Toc146706807)

[1.1 Product perspective 1](#_Toc146706808)

[1.2 Product features 1](#_Toc146706809)

[1.2.1 There are three different users who will be using this product: 1](#_Toc146706810)

[1.2.2 The features that are required for the Librarian are: 1](#_Toc146706811)

[1.2.3 The features that are required for the Member are: 1](#_Toc146706812)

[1.3 Functional Requirement 1](#_Toc146706813)

[1.3.1 Librarian 1](#_Toc146706814)

[1.3.2 Member 2](#_Toc146706815)

[1.4 Non-Functional Requirement 2](#_Toc146706816)

[1.4.1 Usability: 2](#_Toc146706817)

[1.4.2 Accuracy: 2](#_Toc146706818)

[1.4.3 Availability: 2](#_Toc146706819)

[1.4.4 Maintainability: 2](#_Toc146706820)

[2 Design and Implementation Constraints 3](#_Toc146706821)

[2.1 Use case diagram 3](#_Toc146706822)

[2.2 Activity diagram and Swimlane diagram 4](#_Toc146706823)

[2.3 Sequence diagram 5](#_Toc146706824)

[2.4 State diagram 6](#_Toc146706825)

[2.5 Class diagram 7](#_Toc146706826)

[2.6 Data flow diagram 8](#_Toc146706827)

[2.6.1 Context diagram (level-0) 8](#_Toc146706828)

[2.6.2 DFD Level-1 8](#_Toc146706829)

[2.6.3 DFD Level-2 9](#_Toc146706830)

[3 External interface requirement (Screens) 10](#_Toc146706831)

[3.1 Screen-1: Registration Form 10](#_Toc146706832)

[3.2 Screen-2: Login Form 11](#_Toc146706833)

[3.3 Screen-3: Add borrower book 12](#_Toc146706834)

[4 Database design 13](#_Toc146706835)

[4.1 List of Tables 13](#_Toc146706836)

[5 Stories and Scenario 14](#_Toc146706837)

[5.1 Story-1: Add New Book in Library Catalogue 14](#_Toc146706838)

[5.1.1 Scenario# S1.1 14](#_Toc146706839)

[5.1.2 Scenario# S1.2 14](#_Toc146706840)

[5.1.3 Scenario# S1.3 14](#_Toc146706841)

[5.2 Story-2: Search Book 15](#_Toc146706842)

[5.3 Story-3: Manage due date for borrowed book 15](#_Toc146706843)

[5.4 Story-4: Renew book 15](#_Toc146706844)

[5.5 Story-5: Generate a report on book usage and availability 15](#_Toc146706845)

[6 Test cases 16](#_Toc146706846)

[7 References 20](#_Toc146706847)

# List of Figures

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

[Figure 2.2‑1 Activity diagram for Book Issue 4](#_Toc146706849)

[Figure 2.2‑2 Swimlane diagram for Book Issue 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 aims to transform the manual shopping system into an internet-based application, enabling users to manage their accounts, check product availability, and make purchases online. It is a multi-user platform that handles all fundamental functions of an online store, such as product cataloguing, order processing, inventory management, and customer service. This system efficiently supports the essential operations of an online shopping platform, catering to the needs of both small and large-scale e-commerce businesses.

## Product features

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

* **Admin**: Website owner who manages and oversees the entire system.
* **Seller**: A user who sells product or service on the platform.
* **Customer**: A registered user who can browse, purchase products, and manage their account.

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

* Control the platform operations to ensure smooth functionality.
* Can view all user accounts, including customers, sellers, and guests.
* Monitor and approve product listings by sellers.
* Can generate reports such as sales, inventory, and user activity reports.
* Can search for specific products or users using various criteria.
* View, edit and insert new product categories.
* Update platform content, including FAQs and policies.
* Monitor customer support activities.
* Access and edit platform settings and configurations.
* Review and respond to feedback from customers and sellers.

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

* List new products for sale on the platform.
* Set and Update pricing for products
* Update existing product listings.
* Can view and process customer orders.
* Can handle customer inquiries and provide support.
* Receive notifications about orders, inquiries, and returns.
* Can access, generate sales reports and performance analytics.
* Offer discounts and promotional deals on their products.
* Access and update their account information.
* Monitor sales performance and track revenue.

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

* Can log in and manage their profile information.
* Browse products by categories, brands, and other filters.
* Search for specific products using keywords.
* View detailed product descriptions, images, and reviews.
* Add products to the shopping cart.
* Proceed to checkout and place orders.
* Track the status of their orders.
* Provide reviews and ratings for purchased products.

## Functional Requirement

### Admin

* Login: Admin can securely log in to access the admin account.
* View User: Admin can view a list of all users registered on the platform along with their details.
* Remove User Account: Admin can delete user accounts from the platform as needed, ensuring proper data management and user privacy compliance.
* Manage ads: Admin can manage the type of advertisements customers have to see according to seller’s sponsorship.
* View Seller: Admin can view a list of all sellers registered on the platform along with all their details.
* Remove Seller: Admin can delete trader’s account if found seller breaking rules and for security purposes.
* **Customer Support:** Admin can oversee customer support operations and respond to inquiries or issues.
* Receive **Reviews and Ratings:** Admin can receive customer reviews and ratings.
* Reply to Reviews: Admin can reply to customer’s reviews and ratings.
* **Platform Customization:** Admin can customize the platform's theme, layout, and branding.

### Customer

* **Login**: Customers can securely log in to their account to access personalized features.
* **Registration: Customer can register him/herself.**
* **Search Products:** Customers can search for products using keywords.
* **Apply Filters:** Customers can apply filters (e.g., price range, brand, size) to refine search results.
* **Select Category**: Customers can browse and select products by category.
* **View Product Details:** Customers can view detailed information about each product.
* **Read Reviews/Ratings:** Customers can read reviews and ratings from other buyers.
* **Write Review/Rating:** Customers can write and submit reviews or ratings for products they have purchased.
* **Add to Cart:** Customers can add selected products to their shopping cart.
* **View Cart:** Customers can view the items currently in their shopping cart.
* **Select Payment Method:** Customers can choose from available payment methods (e.g., credit card, PayPal).
* **Provide Shipping Details**: Customers can enter shipping information such as address and preferred delivery date and gift wrap option.
* **Apply Coupon Code for Discount**: Customers can apply a coupon code during checkout to receive discounts.
* **Place Order:** Customers can confirm and place their order.
* **View Order Detail**: Customers can view details of their current and past orders.
* **Track Order:** Customers can track the status and location of their order in real-time.
* **Cancel Order:** Customers can cancel an order before it is shipped or delivered.
* **Return Product:** Customers can initiate a return request for products within the return period.
* **Customer Support:** Customers can contact customer support for assistance with orders, returns, or inquiries.
* **View Account Information:** Customers can access and update their account information, including personal details, communication preferences, and saved addresses.
* **Sort and Compare Products**: Customers can sort search results by relevance, price, popularity, or other criteria, and compare product specifications side by side.
* **Access Multi-language Support:** Customers can access the platform in their preferred language or choose from available language options for browsing and purchasing products.
* **Register complain at customer support:** If found any difficulty in receiving product , they can always use customer support to register their complain.

### Seller

* **Login:** Sellers can securely log in to their accounts.
* **Registration**: Sellers can register and create a new account on the platform.
* **Add Product:** Sellers can add new products to the inventory.
* **Remove Product**: Sellers can remove products from the inventory.
* **Update Product Information**: Sellers can update details of existing products.
* **Track Sales Information:** Sellers can view and track sales data and performance.
* **Provide Order Tracking Information:** Sellers can update customers with order tracking details.
* **Keep Track of Product Inventory:** Sellers can monitor and update product stock levels.
* **Provide Refund:** Sellers can process refunds for returned products.
* **View Orders:** Sellers can view all orders placed for their products.
* **Manage Shipping Details**: Sellers can manage shipping options and delivery methods for their products.
* **Respond to Customer Inquiries**: Sellers can communicate with customers regarding product inquiries and order issues.
* **Generate Sales Reports:** Sellers can generate detailed sales reports for performance analysis.
* **View Customer Feedback:** Sellers can view and respond to customer reviews and ratings.
* **Manage Store Profile**: Sellers can update their store information and profile details.
* **Adds discounts**: Can add special discounts on their product if they want to.
* **Sponsor themselves**: Sellers can sponsor themselves if they want to by paying certain amount to software and can request admin to advertise themselves.

## 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.

### Security:

* The system must implement secure payment processing using encryption (e.g., SSL/TLS) and be compliant with PCI DSS standards. User data, including passwords and payment information, must be stored securely and follow GDPR or applicable data protection regulations.

# Design and Implementation Constraints

## Use case diagram

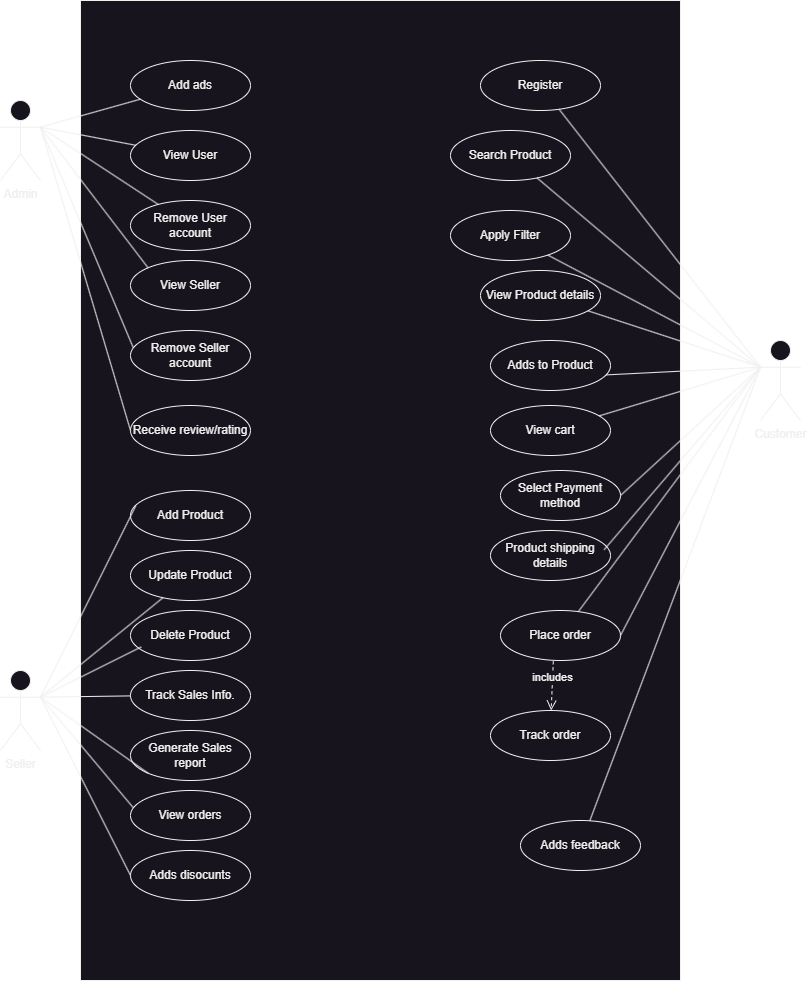


Figure ‑ Use case diagram for library management system

## Activity diagram and Swimlane diagram

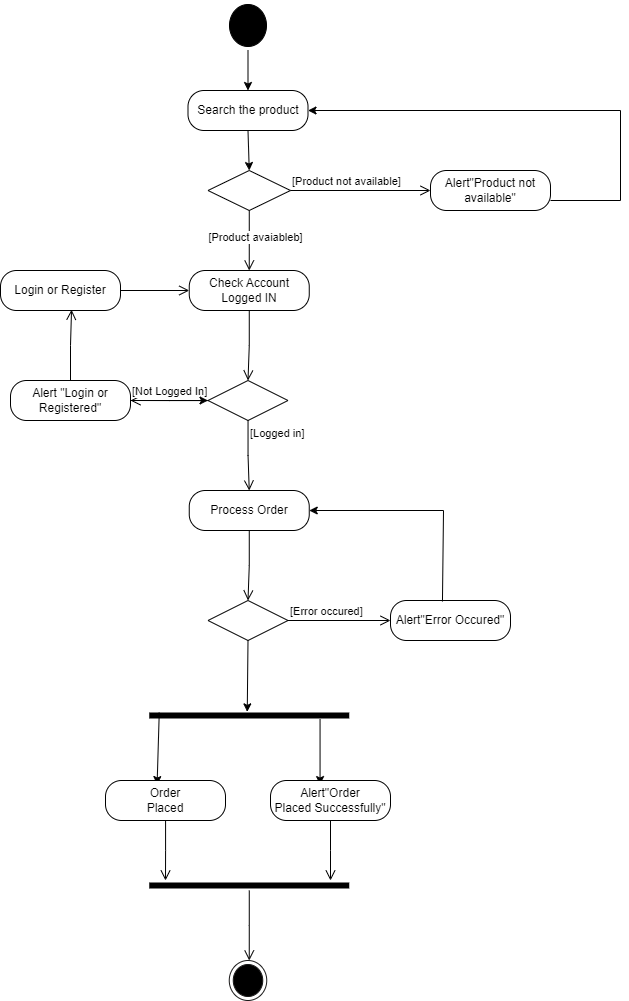
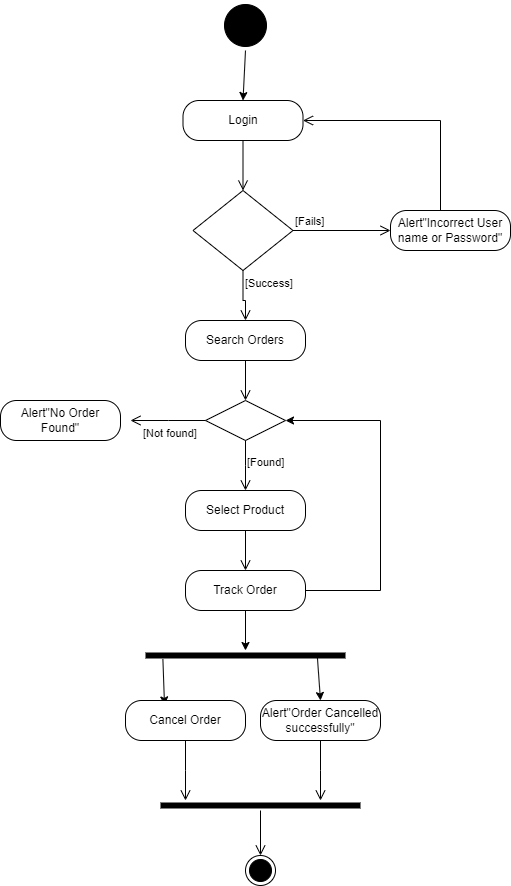
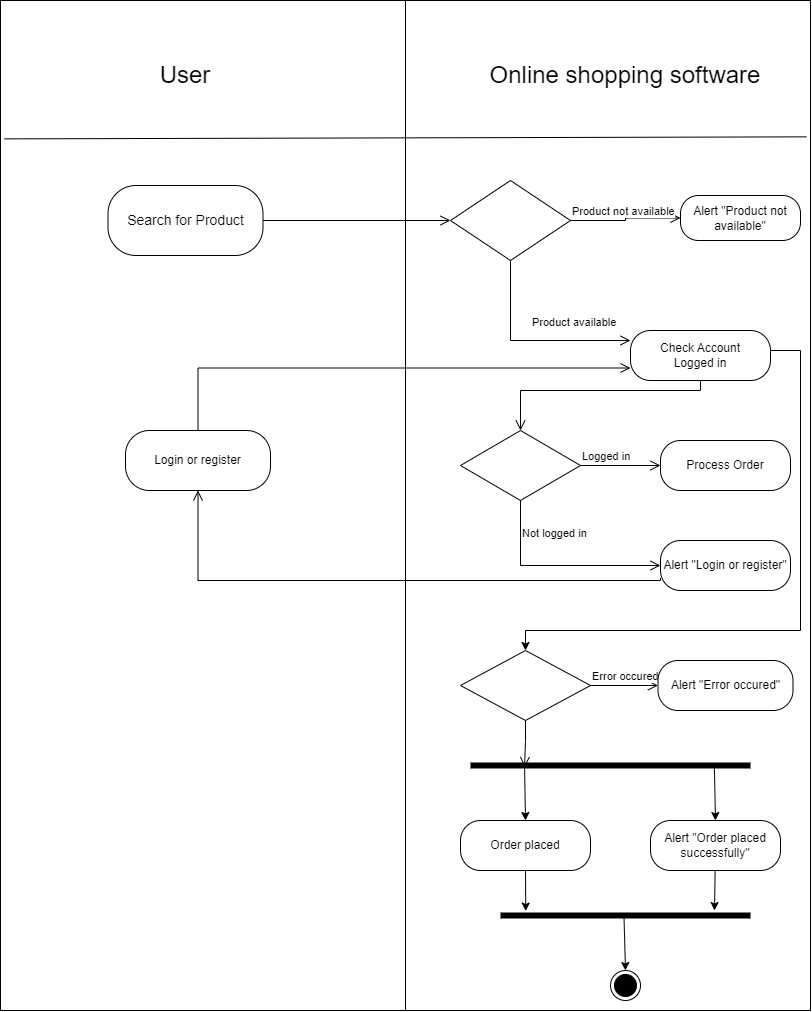


Figure ‑ Activity diagram for Product Ordering





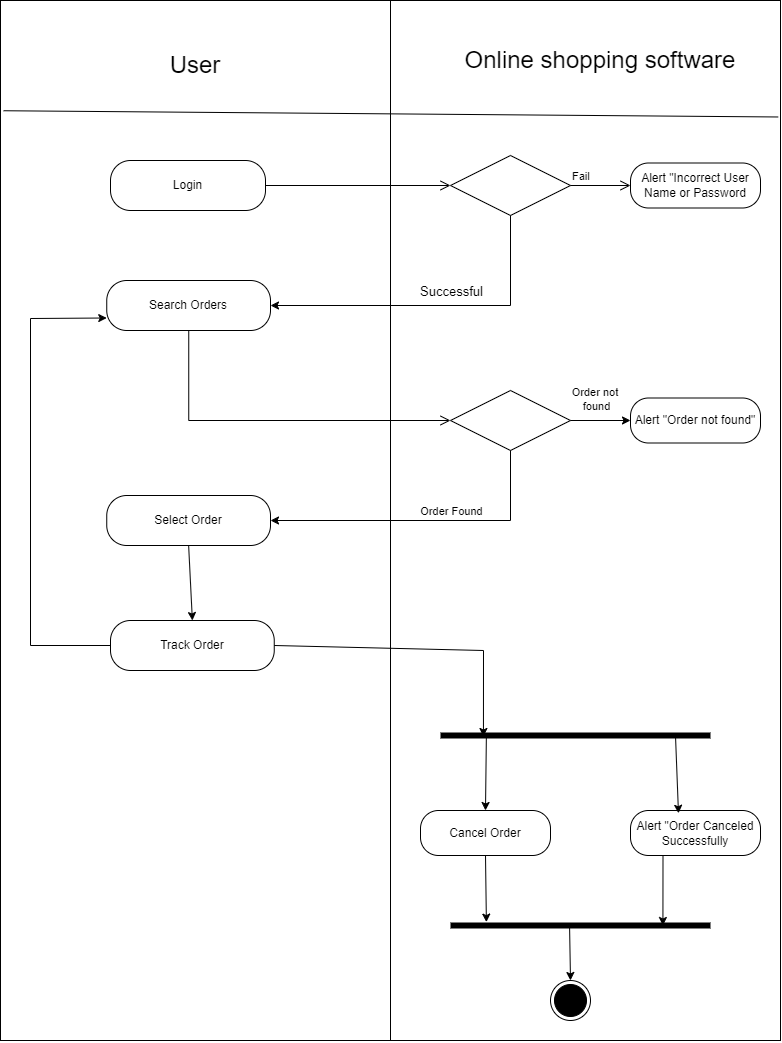


Figure ‑ Swimlane diagram for Book Issue

## Sequence diagram

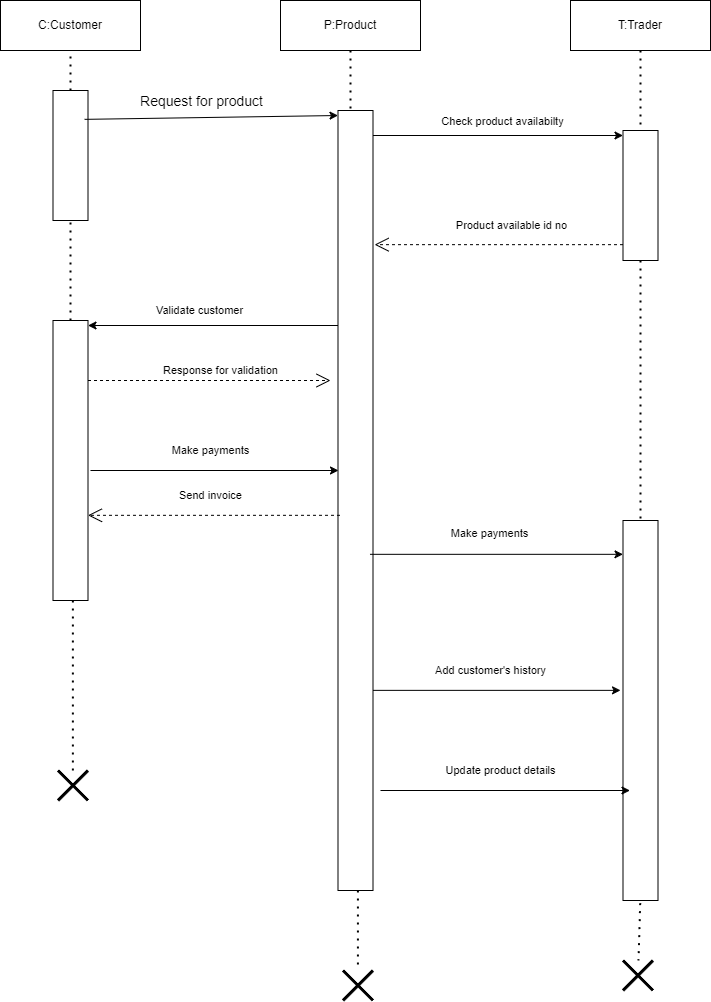
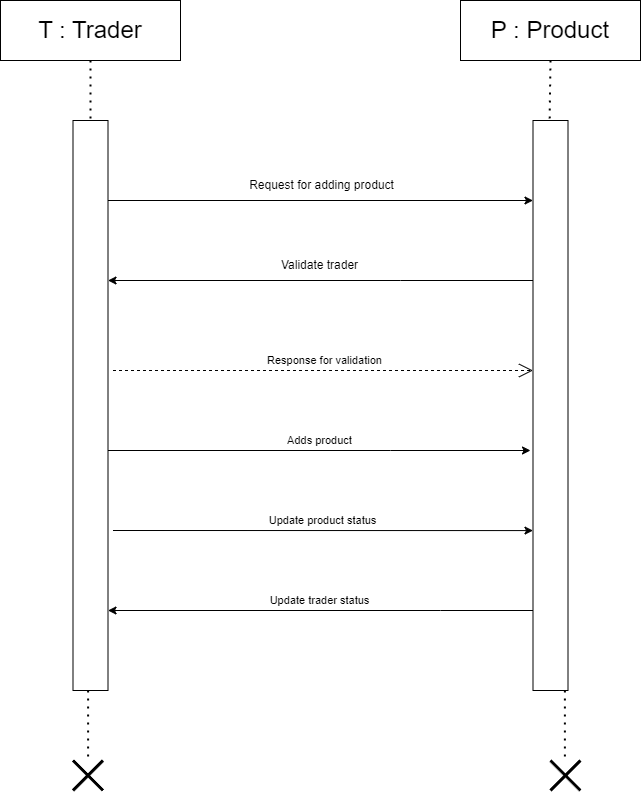


Figure ‑ Sequence diagram for Book Issue



## State diagram

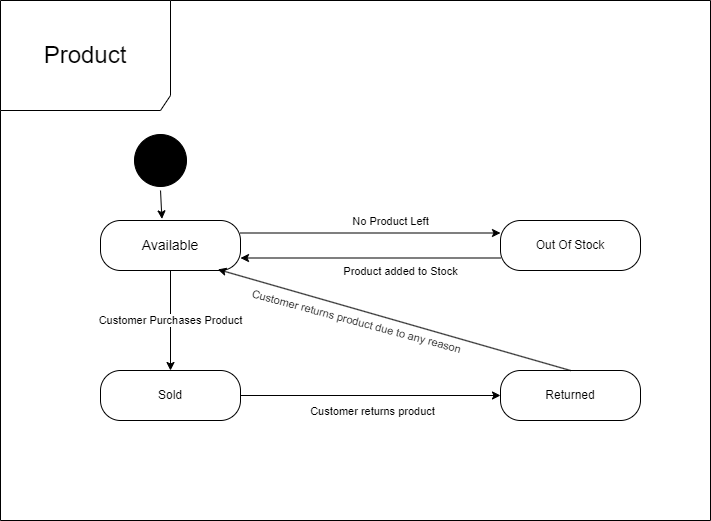


Figure ‑ State diagram of Product

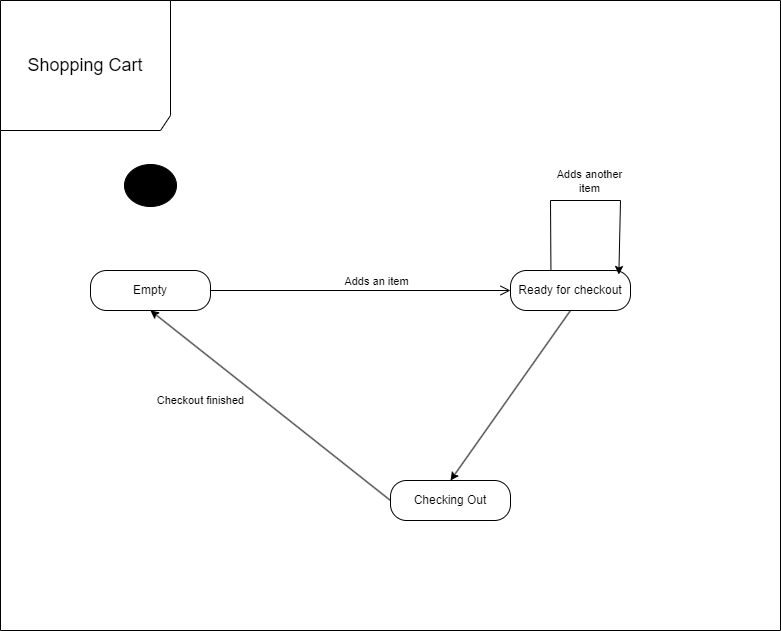


Figure ‑ State diagram for Shopping Cart

## Class diagram



Figure ‑ Class diagram for Library management system

## Data flow diagram

### Context diagram (level-0)



Figure ‑ Context diagram for Library management system

### DFD Level-1



Figure ‑ DFD level-1 for Library management system

### DFD Level-2



Figure ‑ DFD level-2 for Issue book

# External interface requirement (Screens)

## Screen-1: Registration Form



Figure ‑ Screen-1: Registration Form

**Purpose:** This form will allow the target end-users to register in the system. To register , the following information will be encoded in the system.

Table ‑ Screen element of Registration form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Username | Textbox | M | 1 | Username field should be editable and accept the Username. |
| 2 | Email | Textbox | M | 1 | Email field should be editable and accept the email with proper format. |
| 3 | Password | Password | M | 1 | Password field should be editable and accept the password and display as star or dot. |
| 4 | Retype password | Password | M | 1 | Retype Password field should be editable and accept the password and display as star or dot. |
| 5 | I agree to the terms | Checkbox | M | 1 | Checkbox for accepting terms and condition. |
| 6 | I already have a membership | Link | ------ | ------ | Link for navigate to login page for registered user. |
| 7 | Register | Button | ------ | ------ | Register is a button for store the entered data into database. |

## Screen-2: Login Form



Figure ‑ Screen-2: Login Form

**Purpose:** This form will be used by the system’s users to access records and features of the system. The users will input the correct combination of their username and password to be able to login to the system.

Table ‑ Screen element of Login form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Username | Textbox | M | 1 | Username field should be editable and accept the Username. |
| 2 | Password | Password | M | 1 | Password field should be editable and accept the password and display as star or dot. |
| 3 | Remember Me | Checkbox | M | 1 | Saving login credentials through remember me checkbox |
| 4 | I forgot my password | Link | ------ | ------ | Link for navigate to Forgot password page for allows users to recover password. |
| 5 | Register a new membership | Link | ------ | ------ | Link for navigate to membership registration. |
| 6 | Sign in | Button | ------ | ------ | Login button navigates to another page even if valid login credentials. |

## Screen-3: Add borrower book



Figure ‑ Screen-3: Add borrower book

**Purpose:** This module will allow the system administrator to add, edit, update or delete borrowers of book. The admin can add borrower information and manage it.

Table ‑ Screen element of Add borrower book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. | Screen Element | Input Type | O/M | 1/N | Description |
| 1 | Borrower ID | Textbox | M | 1 | Borrower ID field should be editable and fill autogenerated Borrower ID. |
| 2 | Student ID | Textbox | M | 1 | Student ID field should be editable and accept Student barcode id. |
| 3 | First Name | Textbox | M | 1 | First Name field should be editable and accept only string as a borrower first name. |
| 4 | Middle Name | Textbox | O | 1 | Middle Name field should be editable and accept only string as a borrower middle name. |
| 5 | Last Name | Textbox | M | 1 | Last Name field should be editable and accept only string as a borrower last name. |
| 6 | Contact | Textbox | M | 1 | Contact field should be editable and accept only 10-digit mobile number. |
| 7 | Grade Id | Textbox | O | 1 | Auto fetch detail from the database if proper student id is entered. |
| 8 | Submit | Button | ----- | ----- | Submit is a button for store the entered data into database. |
| 9 | Cancel | Button | ----- | ----- | Cancel is a button for close add borrower dialog. |

# Database design

## List of Tables

* Book
* Borrower
* Student
* Staff

Table ‑ Table: Book

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BookID | int | NN | PK (Auto Increment) |  |
| BookName | varchar(100) | NN |  |  |
| ISBN | varchar(100) | AN |  |  |
| Publication\_year | int | AN |  |  |
| Language | varchar(50) | AN |  |  |

Table ‑ Table: Borrower

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| BorrowerID | int | NN | PK (Auto Increment) |  |
| BookID | varchar(100) | NN | FK | Reference of Book Table |
| BorrowedFromDate | DateTime | AN |  |  |
| BorrowedToDate | DateTime | AN |  |  |
| ActualReturnDate | DateTime | AN |  |  |
| IssuedBy | int | NN | FK | Reference of Student and Staff Table |

Table ‑ Table: Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StaffID | int | NN | PK (Auto Increment) |  |
| StaffName | varchar(100) | AN |  |  |
| IsAdmin | Boolean | AN |  |  |
| Designation | varchar(100) | AN |  |  |

Table ‑ Table: Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Data Type | Null | Keys & Constrains | Default Value & Description |
| StudentID | int | NN | PK (Auto Increment) |  |
| StudentName | varchar(100) | AN |  |  |
| Gender | varchar(100) | AN |  |  |
| DOB | DateTime | AN |  |  |
| Department | varchar(100) | AN |  |  |
| Contact | number(10,0) | AN |  |  |

# Stories and Scenario

## Story-1: Add New Book in Library Catalogue

|  |  |  |
| --- | --- | --- |
| *Story # S1* | : | As a Librarian,  I want to add a new book in library catalogue  So that everyone can easily find and borrow it. |
| Priority | **:** | High |
| Estimate | **:** | XL |
| Reason | **:** | The addition of a new book to the library catalogue is crucial for ensuring that the library's collection is up-to-date and accessible to everyone. |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.1* | : | Adding a New Book with Valid Information |
| Prerequisite | **:** | Librarian is logged in to the Library management system. |
| Acceptance Criteria | **:** | **Given:**  The Librarian is navigated to the library catalog management page. Valid book information, including title, author, ISBN, and other relevant details is added.  **When:**  The librarian selects the "Add New Book" option  And The librarian enters valid book details  The librarian clicks the "Save" button to add the book to the catalog.  **Then t**he system successfully adds the book to the catalog and the librarian receives a confirmation message with the book's identification number. |

### Scenario# S1.2

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

### Scenario# S1.3

|  |  |  |
| --- | --- | --- |
| *Scenario# S1.3* | : | Attempting to Add a Duplicate Book |
| Prerequisite | **:** | The librarian is logged into the library management system and the librarian is on the library catalogue management page |
| Acceptance Criteria | **:** | **Given**: The book information, including title, author, ISBN, and other relevant details, is available and the book with the same ISBN is already in the catalogue.  **When**: User Clicks on “Add book” button. Enter a number of copies with the same book detail mentioned in the field.  **Then**: Generate unique book id, barcode and spine label for various book of same title. |

## Story-2: Search Book

|  |  |  |
| --- | --- | --- |
| *Story # S2* | : | As a Librarian or member,  I want to search for books by title, author, or keyword,  So that I can quickly find books that match my interests. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Implementing a search functionality is essential for enhancing the user experience, as it allows librarian and member to efficiently discover and access the library's resources. |

## Story-3: Manage due date for borrowed book

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to manage due dates for borrowed books  So that I can ensure that books are returned on time and avoid overdue fines. |
| Priority | **:** | High |
| Estimate | **:** | M |
| Reason | **:** | Proper due date management is crucial for maintaining the library's collection and ensuring that books are available for all members. |

## Story-4: Renew book

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to renew a book that I have borrowed  So that I can extend my borrowing period if needed. |
| Priority | **:** | Medium |
| Estimate | **:** | M |
| Reason | **:** | Book renewal functionality is a convenience feature for librarian, allowing them to keep a book for an extended period if no one else has requested it. |

## Story-5: Generate a report on book usage and availability

|  |  |  |
| --- | --- | --- |
| *Story # S3* | : | As Librarian,  I want to generate reports on library usage and book availability  So that I can make informed decisions about library’s books. |
| Priority | **:** | Medium |
| Estimate | **:** | L |
| Reason | **:** | Reporting functionality helps librarians track the usage of library resources and make data-driven decisions to improve services and collections. |

# 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: Web application should be accessible | | | | |
| Test Case ID | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| TC\_001 | Login to web application with valid credential | Functional | Login to Library management system web application through valid credential | TC\_001 |
| TC\_002 | Login to web application with invalid credential | Functional | Login to Library management system web application through invalid credential | TC\_002 |
| TC\_003 | Varify login page elements | GUI | varify that all elements are availabe on login page | TC\_003 |

|  |  |
| --- | --- |
| **Test Case Title** | Login to web application with valid credential |
| **Test Type** | Functional |
| **Test Priority** | High |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **BUG ID** |
| 1 | Access Web application URL | The site launched properly | Site launched successfully | Pass |  | <https://accounts.google.com/ServiceLogin> |  |
| 2 | Enter valid Username in username field | Username field should be editable and accept the Username | Username input accepted | Pass |  | Username:  Rbgondaliya@gmail.com |  |
| 3 | Enter valid Password in Password field | Password field should be editable and accept the password and display as star or dot | Password input displayed in dot and accepted | pass |  | Password: rbgondaliya |  |
| 4 | Enter valid captcha code in captch field | Captch field should editable and accept captcha and captcha is case sensitive | Captcha input accepted | Pass | Step required when human action validation perform | get captcha from image which is near by captcha field |  |
| 5 | Click on login button | User should login into site and navigated to dashboard | User navigated to dashboard and username should br display in top of the right side. | pass |  |  |  |

|  |  |
| --- | --- |
| **Test Case Title** | Login to web application with invalid credential |
| **Test Type** | Functional |
| **Test Priority** | Medium |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Verify that User is not able to Login with invalid Username and invalid Password | Should be display an error message enter wrong username or password | Display an error of wrong username and password | Pass |  |  |  |
| 2 | Verify that User is not able to Login with Valid Username and invalid Password | Should be display an error message enter wrong password | Display an error of wrong password | Pass |  |  |  |
| 3 | Verify that User is not able to Login with invalid Username and Valid Password | Should be display an error message User not found | Display an error Username not found | Pass |  |  |  |
| 4 | Verify that User is not able to Login with blank Username or Password | Set required field validation message for Username and Password | Display an error of wrong username and password | Fail | Not performa a validation function fix it |  | Bug\_002 |

|  |  |
| --- | --- |
| **Test Case Title** | Varify login page elements |
| **Test Type** | GUI |
| **Test Priority** | Medium |
| **Pre-condition** | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Launch application with the given url | The site launched properly | Site launched successfully | Pass |  | <https://accounts.google.com/ServiceLogin> |  |
| 2 | Verify that the login screen contains elements such as Username, Password, Sign in button, Remember password check box, Forgot password link, and Create an account link. | All listed control displayed properly on the page | Login page loaded successfully | Pass |  |  |  |
| 3 | Verify that cursor is focused on “Username” text box on the page load | Cursor is focused in Username textbox | Cursor focus in Username textbox | Pass |  |  |  |
| 4 | Verify that tab functionality is working properly or not | When tab pressed cursor move in next control | Cursor moving in next control | Pass |  |  |  |
| 5 | Verify that all the fields such as Username, Password has a valid placeholder | All text fields have proper placeholder | All text fields have proper placeholder | Pass |  |  |  |
| 6 | Verify that the labels float upward when the text field is in focus or filled (In case of floating label) | When field is focused or filled, label display on top of the filled | When field is focus or filled, label display on top of the filled | Pass | step required when fields with floating label |  |  |
| 7 | verify that forgot password link working properly | when click on forgot password load forgot passworg page | forgot password link not working | Fail |  |  |  |

# References

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