



Darshan University

A Project Report on

“Gym Management System”

Under the subject

Software Engineering (2301CS405)

B. Tech, Semester – IV

Computer Science & Engineering Department

Submitted By

Student Name: Harmik Rathod

Enrollment No.: 24010101680

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 “**Gym Management System**” submitted in partial fulfilment for the Semester-4 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Department to 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.

Harmik Rathod

Student’s Signature

Date: _____



**Computer Science & Engineering
Department
Darshan University**

CERTIFICATE

This is to certify that the SRS on “**Gym Management System**” has been satisfactorily prepared by **Harmik Rathod (24010101680)** 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. Gondaliya 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

Harmik Rathod

ABSTRACT

A Gym Management System is designed to simplify and digitalize the daily operations of a gym. This system aims to provide a computerized platform for managing memberships, tracking attendance, scheduling classes, and monitoring gym resources. It acts as a modern tool to replace traditional manual processes, ensuring efficient and streamlined operations.

The system includes features like member and manager login, enabling personalized dashboards for users. Members can register, view their profiles, check attendance, and access their workout schedules. It also allows them to make payments online and view payment history. Additionally, members can request personalized training sessions or diet plans by submitting a request form.

For manager, the system offers advanced functionalities such as adding, updating, and deleting member details, managing trainers and tracking equipment usage.

The primary goal of the Gym Management System is to reduce manual workload, enhance user experience, and optimize gym efficiency while maintaining accurate records.

Table of Contents

| | |
|--|----|
| List of Figures..... | I |
| List of Tables..... | II |
| 1 Introduction..... | 1 |
| 1.1 Product perspective | 1 |
| 1.2 Product features..... | 1 |
| 1.2.1 There are three different users who will be using this product:..... | 1 |
| 1.2.2 The features that are required for the Manager are: | 1 |
| 1.2.3 The features that are required for the Trainer are: | 1 |
| 1.2.4 The features that are required for the Member are:..... | 1 |
| 1.3 Functional Requirement..... | 1 |
| 1.3.1 Manager | 1 |
| 1.3.2 Trainer | 2 |
| 1.3.3 Member..... | 2 |
| 1.3.4 Cleaning Staff..... | 2 |
| 1.4 Non-Functional Requirement..... | 2 |
| 1.4.1 Performance:..... | 2 |
| 1.4.2 Scalability:..... | 2 |
| 1.4.3 Security:..... | 2 |
| 1.4.4 Availability: | 2 |
| 1.4.5 Accessibility: | 2 |
| 1.4.6 Maintainability : | 2 |
| 1.4.7 Data Backup:..... | 2 |
| 2 Design and Implementation Constraints..... | 3 |
| 2.1 Use case diagram..... | 3 |
| 2.2 Activity diagram and Swimlane diagram..... | 4 |
| 2.3 Sequence diagram | 5 |
| 2.4 State diagram | 6 |
| 2.5 Class diagram..... | 7 |
| 2.6 Data flow diagram | 8 |
| 2.6.1 Context diagram (level-0)..... | 8 |
| 2.6.2 DFD Level-1..... | 8 |
| 2.6.3 DFD Level-2..... | 9 |
| 3 External interface requirement (Screens) | 10 |
| 3.1 Screen-1: Registration Form..... | 10 |
| 3.2 Screen-2: Login Form..... | 11 |
| 3.3 Screen-3: Add borrower book | 12 |
| 4 Database design | 13 |

| | | |
|-------|---|----|
| 4.1 | List of Tables | 13 |
| 5 | Stories and Scenario | 14 |
| 5.1 | Story-1: Add New Book in Library Catalogue | 14 |
| 5.1.1 | Scenario# S1.1 | 14 |
| 5.1.2 | Scenario# S1.2 | 14 |
| 5.1.3 | Scenario# S1.3 | 14 |
| 5.2 | Story-2: Search Book | 15 |
| 5.3 | Story-3: Manage due date for borrowed book | 15 |
| 5.4 | Story-4: Renew book | 15 |
| 5.5 | Story-5: Generate a report on book usage and availability | 15 |
| 6 | Test cases | 16 |
| 7 | References | 20 |

List of Figures

| | |
|--|----|
| Figure 2.1-1 Use case diagram for gym management system | 3 |
| Figure 2.2-1 Activity diagram for fill attendance..... | 4 |
| Figure 2.2-2 Activity diagram for trainer selection | 4 |
| Figure 2.2-3 Swimlane diagram for fill attendance | 5 |
| Figure 2.3-1 Sequence diagram for fill attendance | 5 |
| Figure 2.4-1 State diagram of Book..... | 6 |
| Figure 2.4-2 State diagram for Librarian | 6 |
| Figure 2.5-1 Class diagram for Library management system | 7 |
| Figure 2.6-1 Context diagram for Library management system | 8 |
| Figure 2.6-2 DFD level-1 for Library management system..... | 8 |
| Figure 2.6-3 DFD level-2 for Issue book | 9 |
| Figure 3.1-1 Screen-1: Registration Form | 10 |
| Figure 3.2-1 Screen-2: Login Form | 11 |
| Figure 3.3-1 Screen-3: Add borrower book..... | 12 |

List of Tables

| | |
|---|----|
| Table 3.1-1 Screen element of Registration form | 10 |
| Table 3.2-1 Screen element of Login form | 11 |
| Table 3.3-1 Screen element of Add borrower book..... | 12 |
| Table 4.1-1 Table: Book..... | 13 |
| Table 4.1-2 Table: Borrower..... | 13 |
| Table 4.1-3 Table: Staff..... | 13 |
| Table 4.1-4 Table: Student | 13 |

1 Introduction

1.1 Product perspective

This project is focused on transforming the manual gym management system into an internet-based application to enhance user experience and streamline operations. The system allows users to manage their accounts, check the availability of equipment, and view schedules for workout sessions and classes. It is a multi-user platform capable of handling all the essential functions of a gym, such as Membership Management, Equipment Tracking, Scheduling, and Billing. This system is designed to meet the fundamental requirements of a small to medium-sized gym, ensuring efficient operations and improved member satisfaction.

1.2 Product features

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

- Manager manage gym details and history of member and trainer.
- Member can be view his details view diet plan.
- Trainer can assign exercise to member.

1.2.2 The features that are required for the Manager are:

- Manage the gym's schedule, equipment usage, and avoid conflicts in booking.
- Search for specific members in the system based on their name, membership ID, etc.
- Manage and track the availability of equipment and maintenance schedules.
- Register new members and manage their profiles.
- Edit the information of existing members.
- View reports of active memberships, member attendance.
- Manage subscription plans and their renewal.
- Can check the report of the issued Books.
- Can access all the accounts of the students.

1.2.3 The features that are required for the Trainer are:

- View assigned members and their fitness goals.
- Create and update personalized workout plans for members.
- Mark attendance for assigned members during sessions.
- Track members' progress and update fitness metrics.

1.2.4 The features that are required for the Member are:

- View available gym equipment and facilities.
- Access fitness reports shared by trainers.
- View payment history and renew memberships.
- Request a specific fitness program or session with a trainer.

1.3 Functional Requirement

1.3.1 Manager

- Add Trainer: Add Training Staff in database
- Add Member: Add Member in database
- Add Cleaning Staff: Add Cleaning Staff in database
- Remove Trainer: Remove Training Staff to database
- Remove Member: Remove Member to database
- Remove Cleaning Staff: Remove Cleaning Staff to database
- Assign Trainer: Assign Trainer to Member

- Pay Salary: Pay Salary to trainer and Cleaning Staff
- Collect Fees: Manager Collect fees from member

1.3.2 Trainer

- View Member: Trainer view his member
- View Salary: View his Salary History
- Request for Leave: Trainer need to leave that request to online
- Assign Task: Assign task to member
- Add content: Add new exercise in the database
- Attendance: Mark the member attendance
- Request to Equipment: To request for new equipment

1.3.3 Member

- Registration: Register to new membership
- Pay Fees: to Pay fees
- View Diet Plan: TO check his diet plan
- View Exercise: TO check exercise to give trainer
- Gym Details: can view Gym details

1.3.4 Cleaning Staff

- View Salary: View his Salary History
- Request for Leave: Trainer need to leave that request to online
- Request to Overtime: To request to overtime
- Task Completion Update: Staff should mark task as completion

1.4 Non-Functional Requirement

1.4.1 Performance:

- The system will deliver high performance, supporting multiple user access simultaneously without significant delays or downtime.

1.4.2 Scalability:

- The project will be designed to accommodate scalability, allowing for future enhancements and feature additions post-deployment without extensive redesign.

1.4.3 Security:

- The system will ensure secure transactions for payments and protect user data through encryption and robust authentication mechanisms.

1.4.4 Availability:

- The software will be accessible anytime and anywhere, ensuring uninterrupted access for users when needed.

1.4.5 Accessibility:

- The system will feature a user-friendly interface, making it easy to use and navigate for both members and administrators.

1.4.6 Maintainability :

- The software will be designed for easy maintenance, ensuring that updates, bug fixes, and modifications can be performed efficiently after deployment.

1.4.7 Data Backup:

- The database will be securely stored in the cloud or on a reliable server to ensure regular backups and quick recovery in case of data loss.

2 Design and Implementation Constraints

2.1 Use case diagram



Figure 2.1-1 Use case diagram for gym management system

2.2 Activity diagram and Swimlane diagram

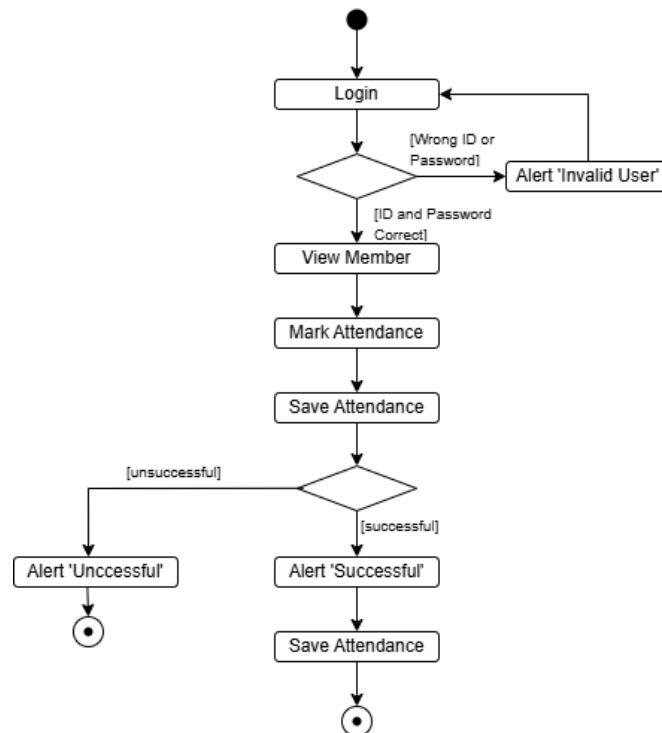


Figure 2.2-1 Activity diagram for fill attendance

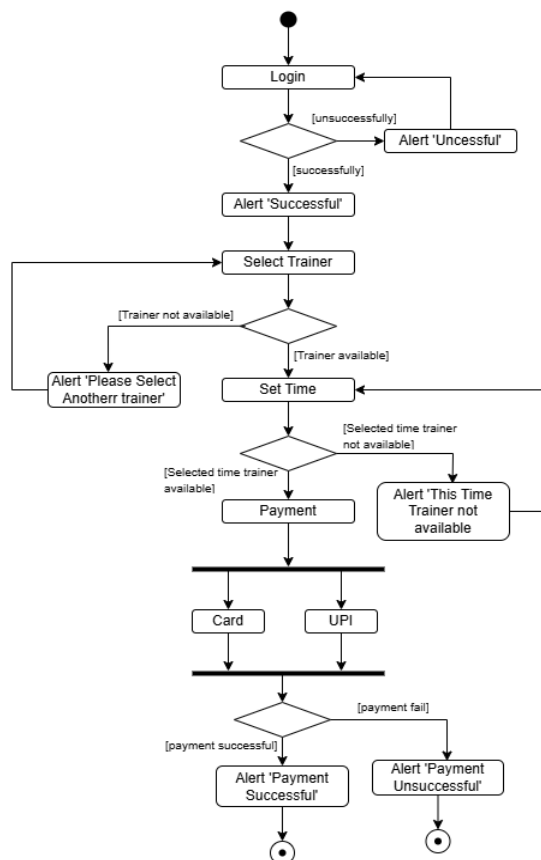


Figure 2.2-2 Activity diagram for trainer selection

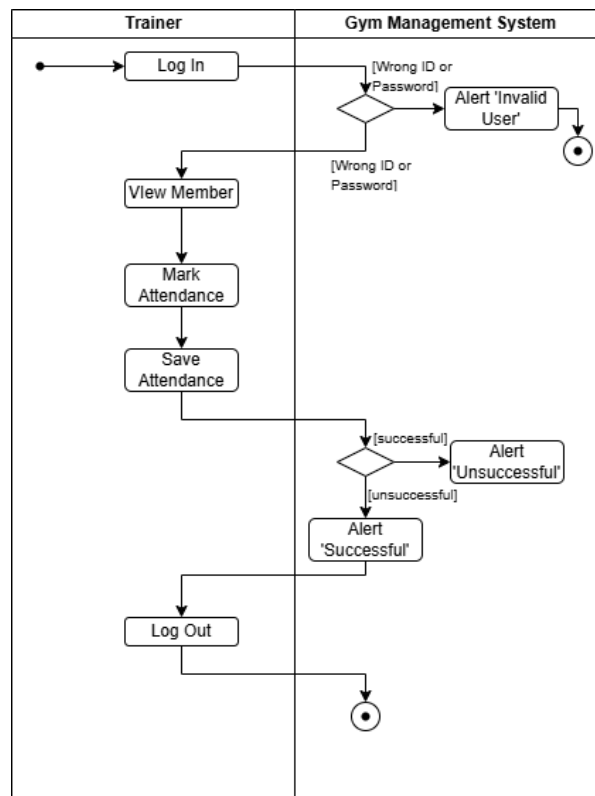


Figure 2.2-3 Swimlane diagram for fill attendance

2.3 Sequence diagram

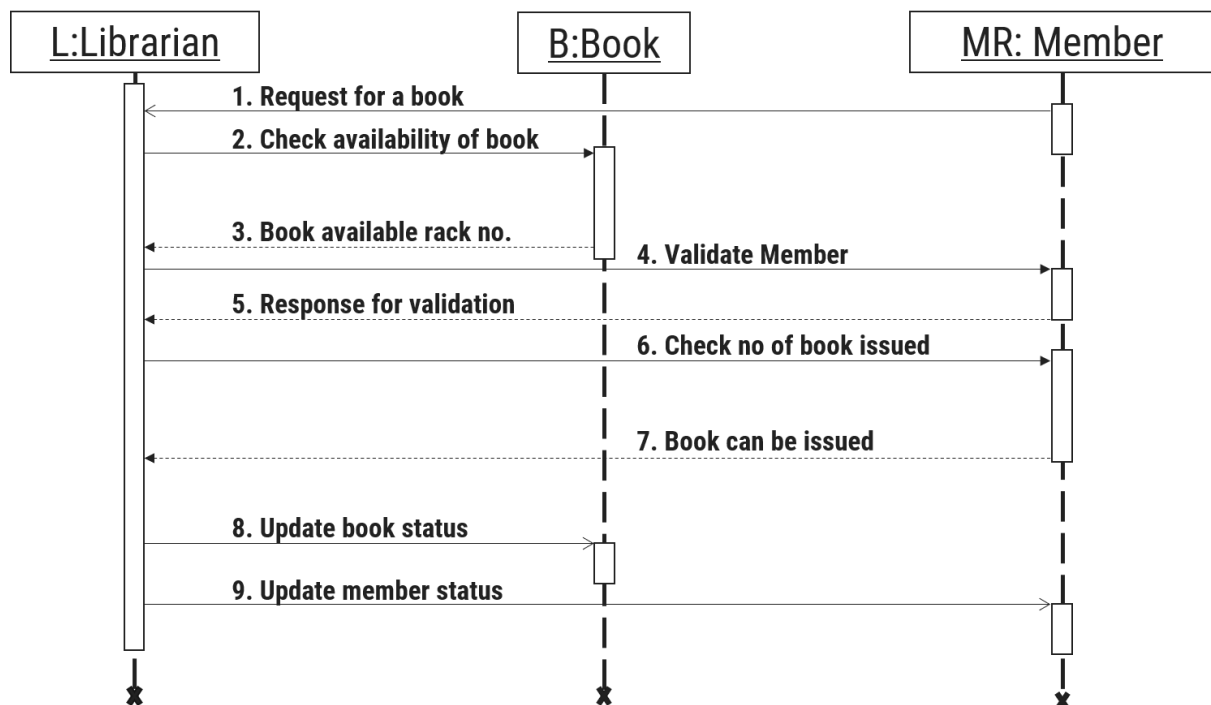


Figure 2.3-1 Sequence diagram for Book Issue

2.4 State diagram

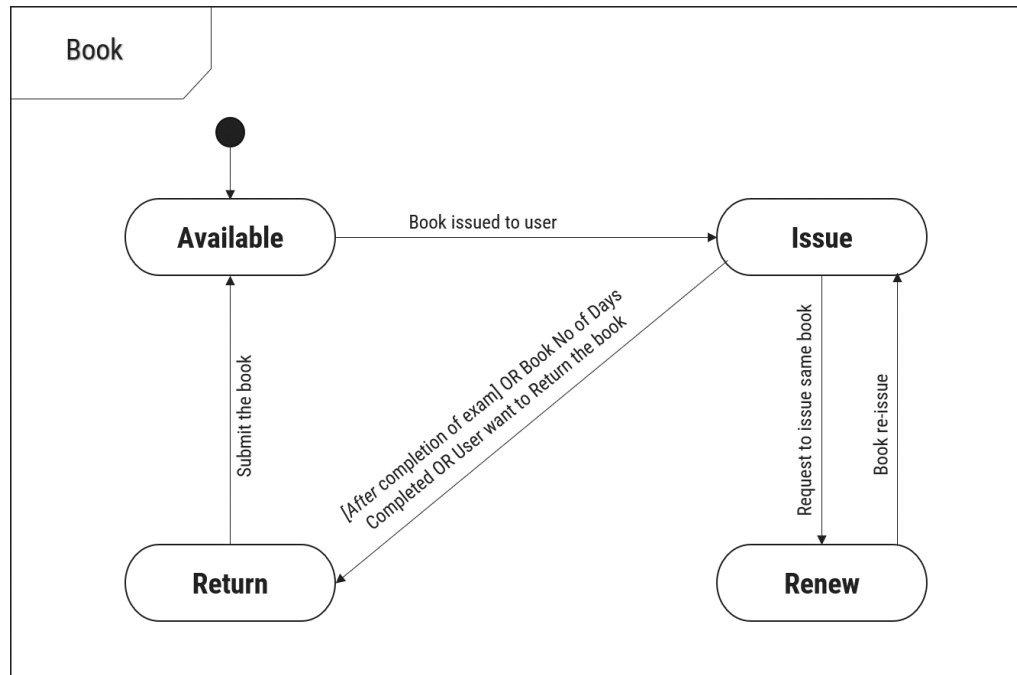


Figure 2.4-1 State diagram of Book

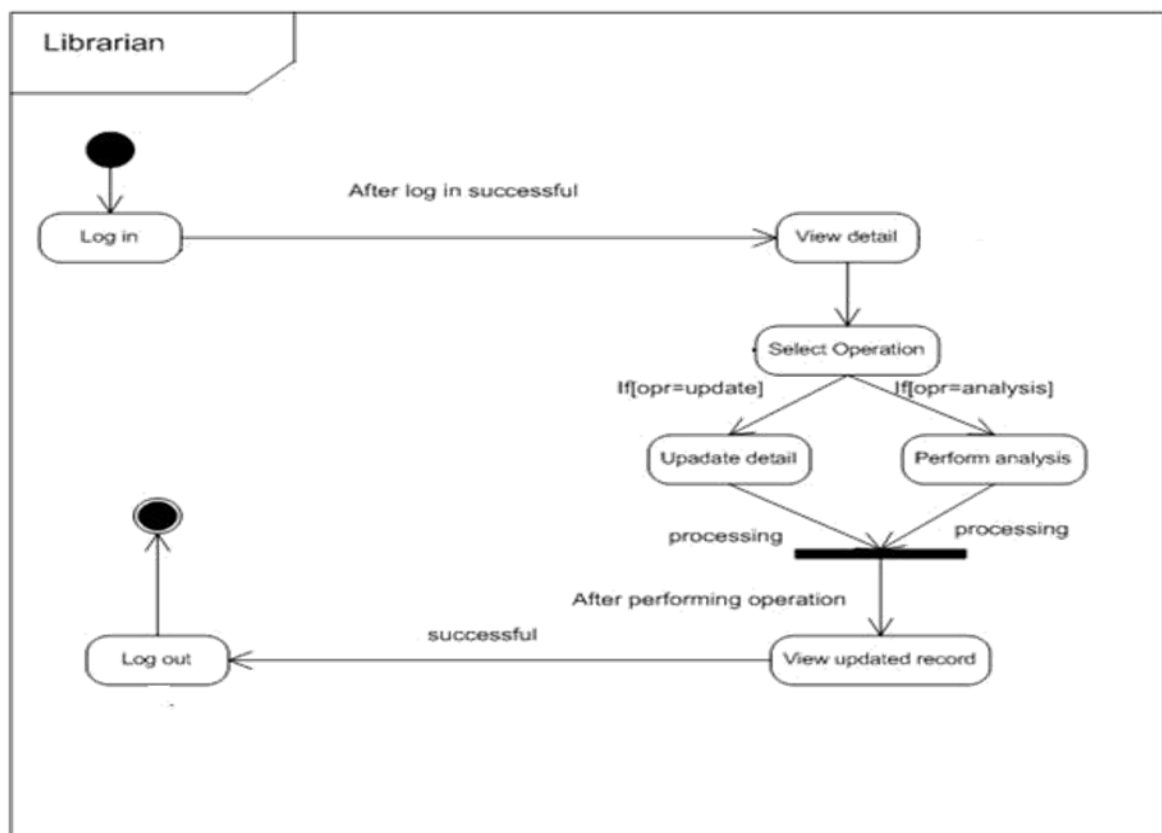


Figure 2.4-2 State diagram for Librarian

2.5 Class diagram

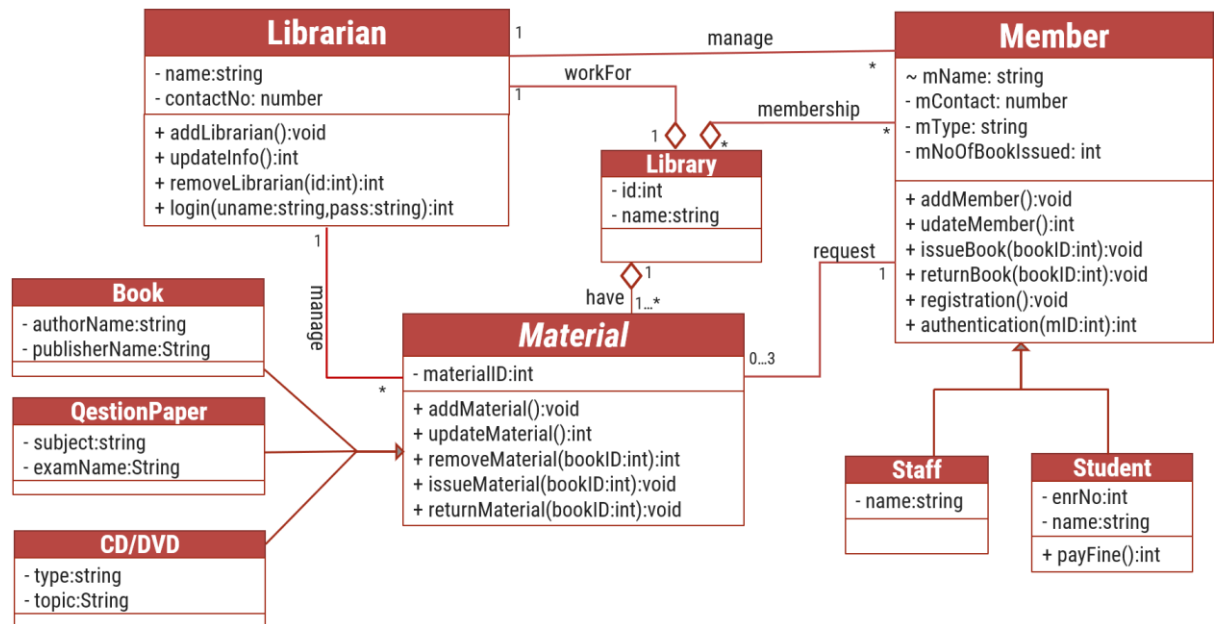


Figure 2.5-1 Class diagram for Library management system

2.6 Data flow diagram

2.6.1 Context diagram (level-0)

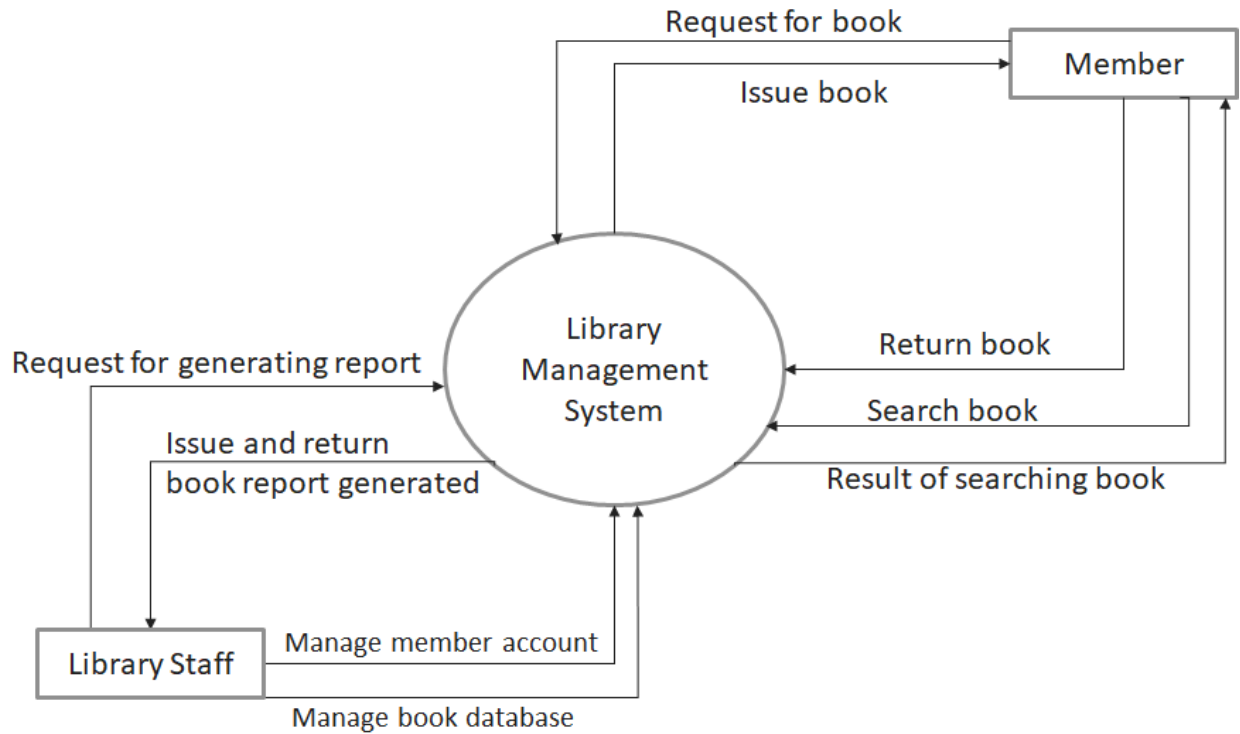


Figure 2.6-1 Context diagram for Library management system

2.6.2 DFD Level-1

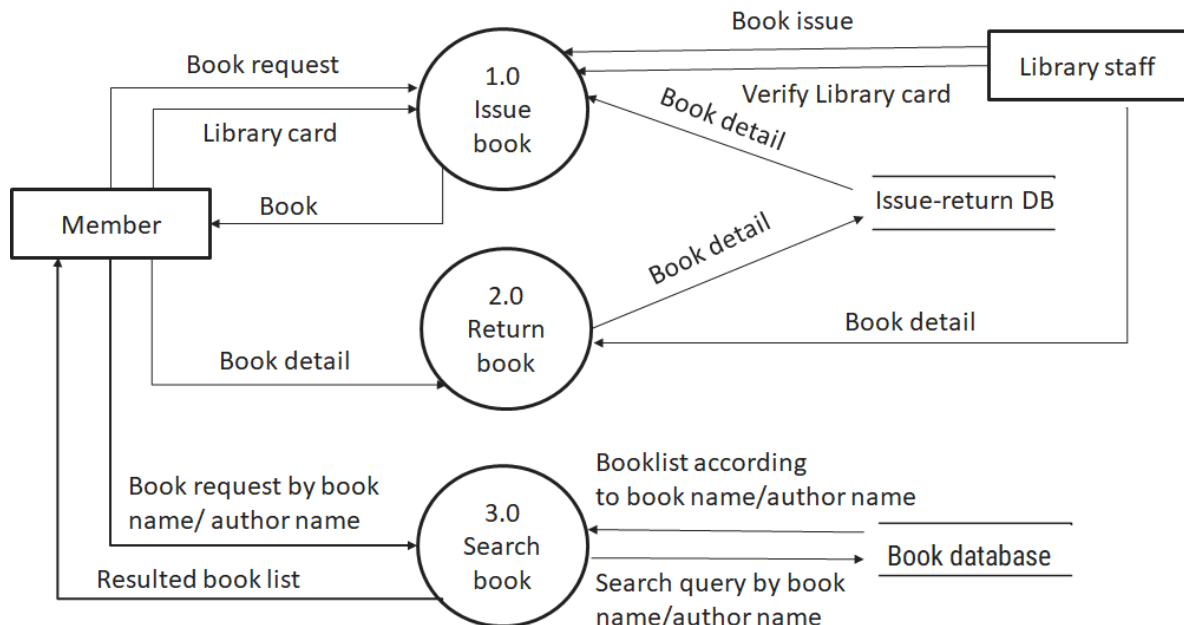


Figure 2.6-2 DFD level-1 for Library management system

2.6.3 DFD Level-2

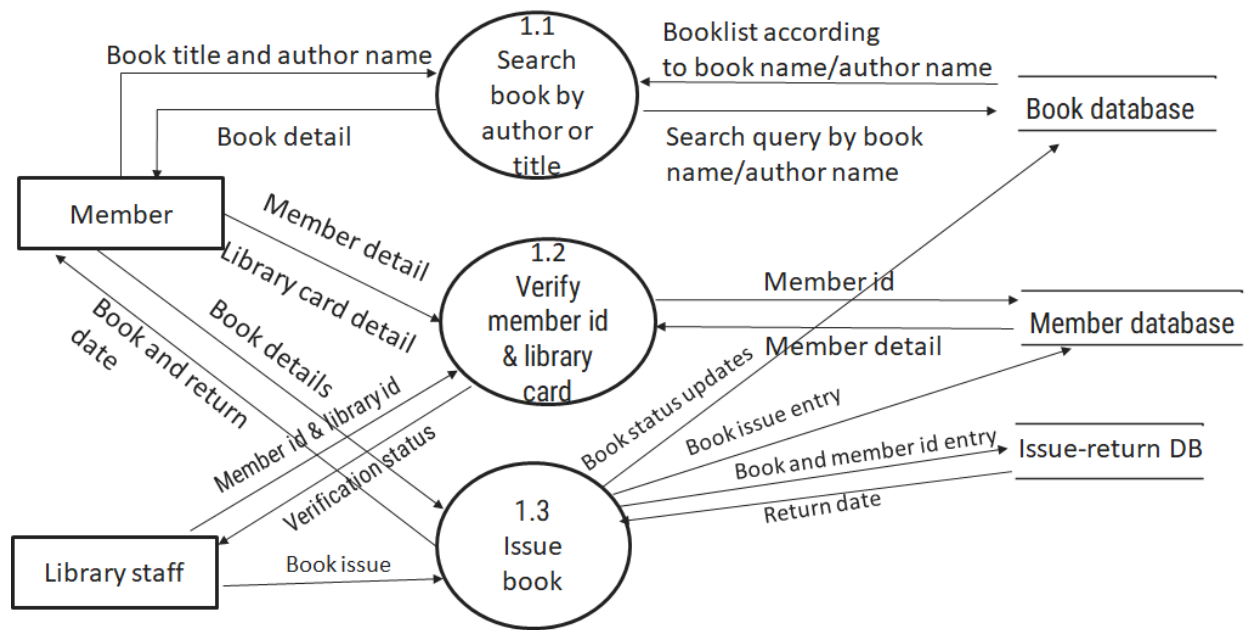


Figure 2.6-3 DFD level-2 for Issue book

3 External interface requirement (Screens)

3.1 Screen-1: Registration Form

The screenshot shows a web interface for a 'Library System'. The main heading is 'Library System' in a large, light blue font. Below it, the word 'Register' is centered. The form consists of four input fields: 'Username' with a person icon, 'Email' with an envelope icon, 'Password' with a padlock icon, and 'Retype password' with a padlock icon. Below these fields is a checkbox labeled 'I agree to the terms' and a link 'I already have a membership'. A blue button labeled 'Register' is positioned to the right of the checkbox.

Figure 3.1-1 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 3.1-1 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. |

3.2 Screen-2: Login Form

Library System

Sign in to start your session

Username 

Password 

☐ **Remember Me**

[Sign In](#)

[I forgot my password](#)

[Register a new membership](#)

Figure 3.2-1 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 3.2-1 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. |

3.3 Screen-3: Add borrower book

The screenshot shows a modal window titled "Add Borrower" with a close button (X) in the top right corner. The form inside has the following fields:

- ID**: A text input field with a placeholder "ID".
- Student ID**: A text input field with a placeholder "Student ID".
- First Name**: A text input field with a placeholder "First Name".
- Middle Name**: A text input field with a placeholder "Middle Name".
- Last Name**: A text input field with a placeholder "Last Name".
- Contact**: A text input field with a placeholder "Contact".
- Category**: A text input field with a placeholder "Category".
- Grade ID**: A text input field with a placeholder "Grade ID".

At the bottom of the form are two buttons: a blue "Submit" button and a grey "Cancel" button.

Figure 3.3-1 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 3.3-1 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. |

4 Database design

4.1 List of Tables

- Book
- Borrower
- Student
- Staff

Table 4.1-1 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 4.1-2 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 4.1-3 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 4.1-4 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 | | |

5 Stories and Scenario

5.1 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. |

5.1.1 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 | <p>Given: The Librarian is navigated to the library catalog management page. Valid book information, including title, author, ISBN, and other relevant details is added.</p> <p>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.</p> <p>Then the system successfully adds the book to the catalog and the librarian receives a confirmation message with the book's identification number.</p> |

5.1.2 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 | <p>Given: The librarian is on the library catalogue management page</p> <p>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.</p> <p>Then the system displays error messages for the incorrect or missing information and the book is not added to the catalogue.</p> |

5.1.3 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 | <p>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.</p> <p>When: User Clicks on "Add book" button. Enter a number of copies with the same book detail mentioned in the field.</p> |

Then: Generate unique book id, barcode and spine label for various book of same title.

5.2 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. |

5.3 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. |

5.4 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. |

5.5 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. |

6 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 available 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 be editable and accept captcha and captcha is case sensitive | Captcha input accepted | Pass | Step require d when human action validation | get captcha from image which is near by captcha field | |

| | | | | | | | |
|---|-----------------------|--|--|------|---------|--|--|
| | | | | | perform | | |
| 5 | Click on login button | User should login into site and navigated to dashboard | User navigated to dashboard and username should be 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 | Should be display an error message | Display an error Username not found | Pass | | | |

| | | | | | | | |
|---|---|---|---|------|--|--|---------|
| | invalid Username and Valid Password | User not found | | | | | |
| 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 perform 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 password page | forgot password link not working | Fail | | | |

7 References

- http://www.w3schools.com/html/html_intro.asp
- <https://www.w3schools.com/php/default.asp>
- <https://www.javatpoint.com/uml>