



Darshan University

A Project Report on

“Hospital Management System”

Under the subject

Software Engineering (2301CS405)

B. Tech, Semester – IV

Computer Science & Engineering Department

Submitted By

Student Name: Deep Botadara

Enrollment No.: 24010101614

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 “**Hospital Management System**” submitted in partial fulfilment for the Semester-5 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.

Deep Botadara

Student’s Signature

Date: _____



**Computer Science & Engineering
Department
Darshan University**

CERTIFICATE

This is to certify that the SRS on “**Hospital Management System**” has been satisfactorily prepared by **Deep Botadara (24010101614)** 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

Deep Botadara

ABSTRACT

The Hospital Management System is a computerized solution designed to manage the administrative, medical, and financial aspects of a hospital efficiently. This system aims to replace traditional manual processes with a digital framework that streamlines hospital operations. It provides distinct modules for different user roles, including Admin, Doctor, Patient, Staff, and Medical Store Manager, each with tailored access and functionalities. The system offers features such as online patient registration, appointment scheduling, electronic medical records management, and billing.

Admin users can oversee and configure the system, generate reports, and manage hospital resources. Doctors can access patient histories, manage appointments, and create prescriptions, while patients can view their medical records and schedule visits conveniently. Staff members assist with inventory and patient admissions, and the Medical Store Manager ensures the availability and tracking of medical supplies and pharmaceuticals. Additionally, the system supports secure payment processing and insurance claim management.

The primary objective of this system is to enhance efficiency, reduce human errors, and improve the overall quality of healthcare services through automation and accessibility.

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 Four different users who will be using this product:.....	1
1.2.2 The features that are required for the Doctor are:.....	1
1.2.3 The features that are required for the Hospital Admin are:	1
1.2.4 The features that are required for the Patient are:	1
1.2.5 The features that are required for the Medical Store Manager are:	1
1.3 Functional Requirement.....	2
1.3.1 Doctor:.....	2
1.3.2 Patient:	2
1.3.3 Hospital Admin:	2
1.3.4 Medical Store Manager:.....	2
• Add Medicines: Add new medicines to the inventory.	2
• Remove Medicines: Remove expired or discontinued medicines from the inventory.....	2
• Order New Medicines: Place orders to restock medicines as required.	2
• Medical Report Analysis: Analyse medical reports to monitor medicine usage trends.	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	6
2.4 State diagram	8
2.5 Class diagram.....	9
2.6 Data flow diagram	10
2.6.1 Context diagram (level-0)	10
2.6.2 DFD Level-1.....	10
2.6.3 DFD Level-2.....	11
3 External interface requirement (Screens)	12
3.1 Screen-1: Registration Form.....	12
3.2 Screen-2: Login Form.....	13
3.3 Screen-3: Add borrower book	14
4 Database design	15
4.1 List of Tables.....	15
5 Stories and Scenario	16

5.1	Story-1: Add New Book in Library Catalogue	16
5.1.1	Scenario# S1.1	16
5.1.2	Scenario# S1.2	16
5.1.3	Scenario# S1.3	16
5.2	Story-2: Search Book	17
5.3	Story-3: Manage due date for borrowed book	17
5.4	Story-4: Renew book	17
5.5	Story-5: Generate a report on book usage and availability	17
6	Test cases	18
7	References.....	22

List of Figures

Figure 2.1-1 Use case diagram for Hospital management system.....	3
Figure 2.2-1 Activity diagram for Book Appointment	4
Figure 2.2-1 Activity diagram for Add Doctor	4
Figure 2.2-2 Swimlane diagram for Book Appointment.....	5
Figure 2.3-1 Sequence diagram for Book Issue	7
Figure 2.4-1 State diagram of Book.....	8
Figure 2.4-2 State diagram for Librarian	8
Figure 2.5-1 Class diagram for Library management system	9
Figure 2.6-1 Context diagram for Library management system	10
Figure 2.6-2 DFD level-1 for Library management system.....	10
Figure 2.6-3 DFD level-2 for Issue book	11
Figure 3.1-1 Screen-1: Registration Form	12
Figure 3.2-1 Screen-2: Login Form	13
Figure 3.3-1 Screen-3: Add borrower book.....	14

List of Tables

Table 3.1-1 Screen element of Registration form	12
Table 3.2-1 Screen element of Login form	13
Table 3.3-1 Screen element of Add borrower book.....	14
Table 4.1-1 Table: Book.....	15
Table 4.1-2 Table: Borrower.....	15
Table 4.1-3 Table: Staff.....	15
Table 4.1-4 Table: Student	15

1 Introduction

1.1 Product perspective

This project aims to modernize the manual hospital management process into an internet-based application, enabling users to access and manage patient records, appointments, billing, and inventory. It is a multi-user platform that handles core hospital functions like Patient Registration, Appointment Scheduling, Billing, and Inventory Management. It is designed to cater to the basic needs of small to medium-sized hospitals, ensuring streamlined operations and improved efficiency.

1.2 Product features

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

- **Doctor:** View Patients, assign medicines, Check Appointments, Request Leave.
- **Hospital Admin:** Add Doctor, Add Laboratories, Pay Salaries, Add Ambulance.
- **Patient:** Book appointments, modify appointments, Make Payment, Track medicines, Log in.
- **Medical Store Manager:** Add medicines, Order for new medicines, medical report analysis.

1.2.2 The features that are required for the **Doctor** are:

- View patient details, including medical history and treatment progress.
- Assign and update prescriptions and treatment plans.
- Request leaves and notify hospital administration for approval
- Access laboratory test results and provide recommendations.
- Monitor patient discharge summaries and follow-up requirements.

1.2.3 The features that are required for the **Hospital Admin** are:

- Register and track ambulance availability and assignments.
- Add and manage doctor profiles, including specialties and schedules.
- Register and update laboratory details and services offered.
- Process and manage staff salary payments.

1.2.4 The features that are required for the **Patient** are:

- Book new appointments with available doctors.
- Modify or cancel existing appointments.
- Make secure payments for consultations, treatments, and services.
- Track prescribed medicines and purchase status.
- Log in to access personal health records and manage profile details

1.2.5 The features that are required for the **Medical Store Manager** are:

- Add and update the inventory of medicines available in the store.
- Place orders for new medicines and restock inventory as needed.
- Analyse medical reports to identify frequently prescribed medicines.
- Monitor expiration dates and remove outdated stock

1.3 Functional Requirement

1.3.1 Doctor:

- View Patients: Access patient details, and ensure new entries are recorded in the database.
- Assign medicines: Prescribe or update medicines, with changes reflected in the database.
- Check appointments: View schedules and remove incorrect entries.
- Request Leave: Submit leave requests, tracked in the system for approval.

1.3.2 Patient:

- Book Appointment: Schedule appointments with doctors, ensuring availability.
- Cancel Appointment: Cancel existing appointments when needed.
- Modify Appointment: Modify scheduled appointments with new date/time.
- View Report: Access and view medical test reports.
- Make Payment: Pay for consultations, treatments, and services securely.
- Order Medicines: Order prescribed medicines from the pharmacy
- Track Medicines: Monitor the status of medicine orders and deliveries.
- Log In: Access personal health records and manage account details.

1.3.3 Hospital Admin:

- Add Doctor: Add new doctor profiles with details and specialties.
- Remove Doctor: Remove doctor profiles from the system.
- Add Medical Store: Register new medical stores and their details.
- Remove Medical Store: Delete medical store entries from the system.
- Add Laboratories: Register new laboratories and their services.
- Remove Laboratories: Remove laboratory records from the system.
- Add Ambulances: Register new ambulances with details for availability.
- Remove Ambulances: Remove ambulance records from the system.
- Pay Salaries: Process and manage staff salary payments.

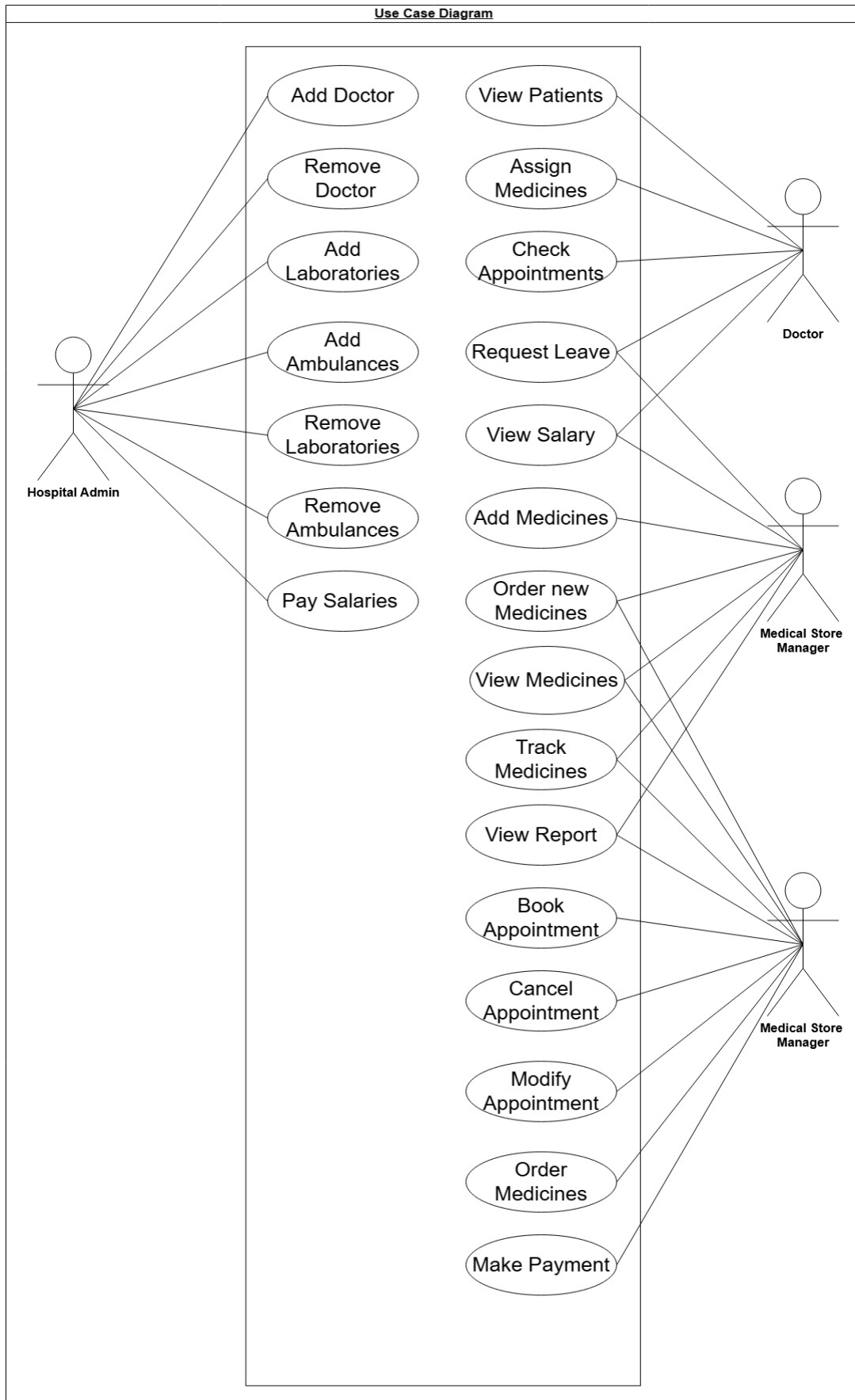
1.3.4 Medical Store Manager:

- Add Medicines: Add new medicines to the inventory.
- Remove Medicines: Remove expired or discontinued medicines from the inventory.
- Order New Medicines: Place orders to restock medicines as required.
- Medical Report Analysis: Analyse medical reports to monitor medicine usage trends.

2 Design and Implementation Constraints

2.1 Use case diagram

Figure 2.1-1 Use case diagram for library management system



2.2 Activity diagram and Swimlane diagram

HOSPITAL MANAGEMENT SYSTEM: Add Doctor

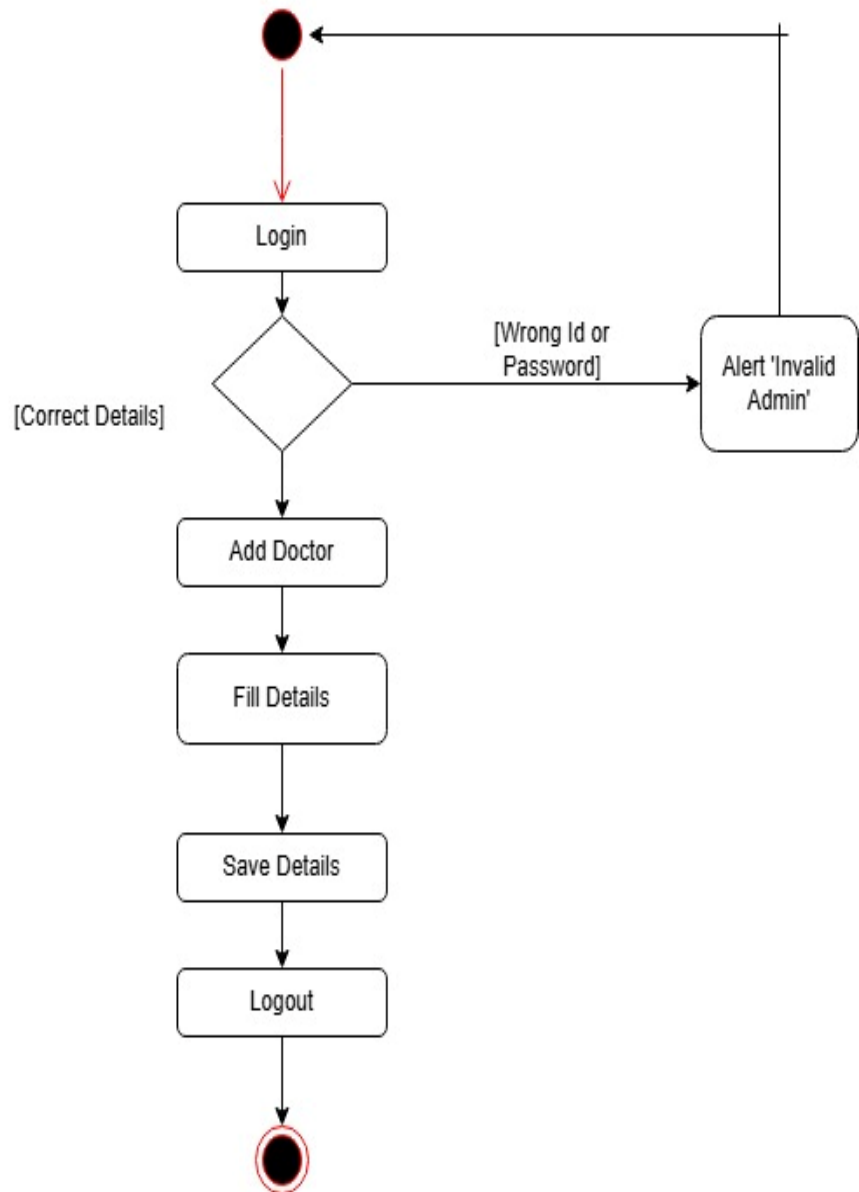
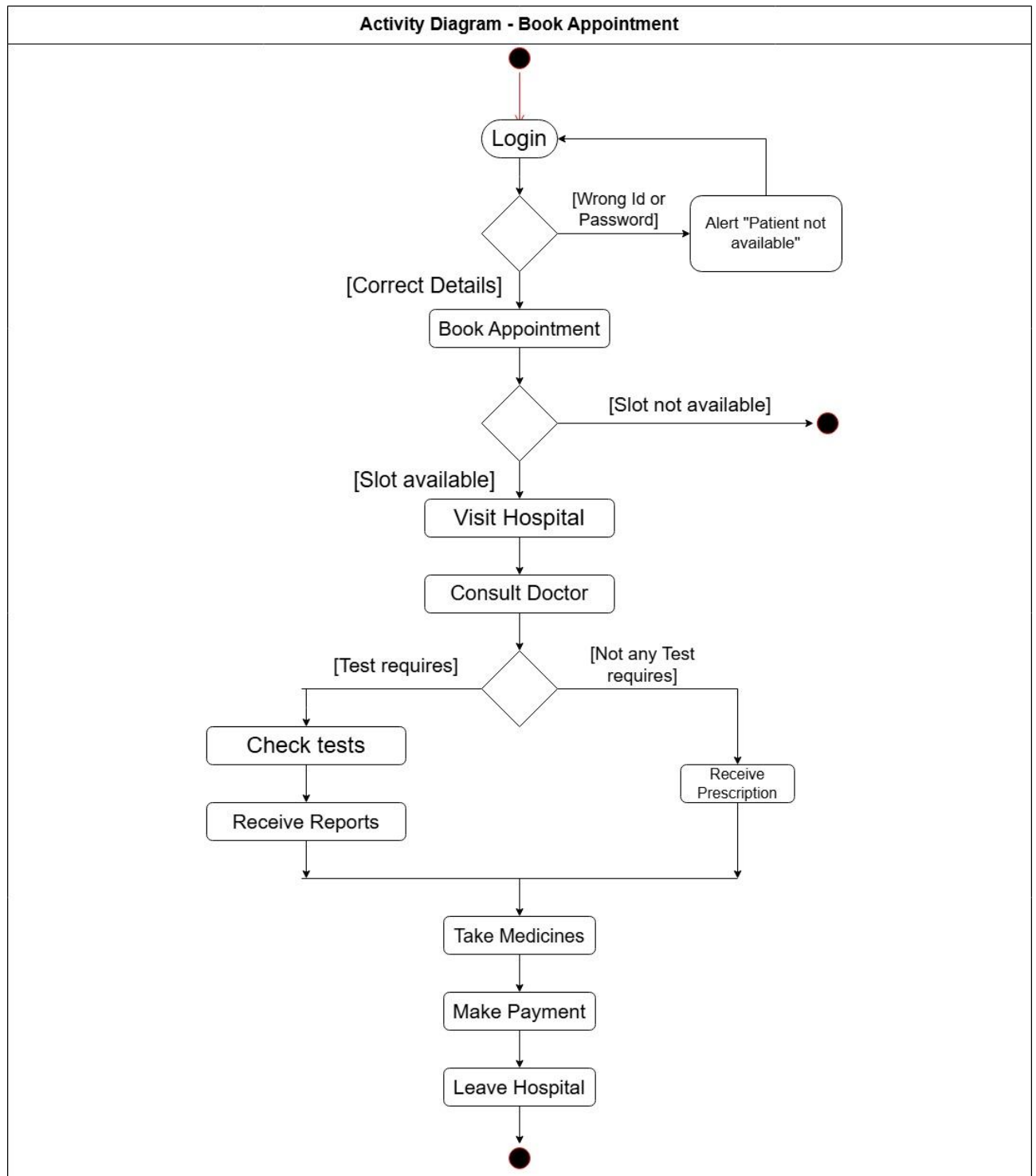


Figure 2.2-1 Activity diagram for Add Doctor



2.3 Swimlane diagram – Book Appointment

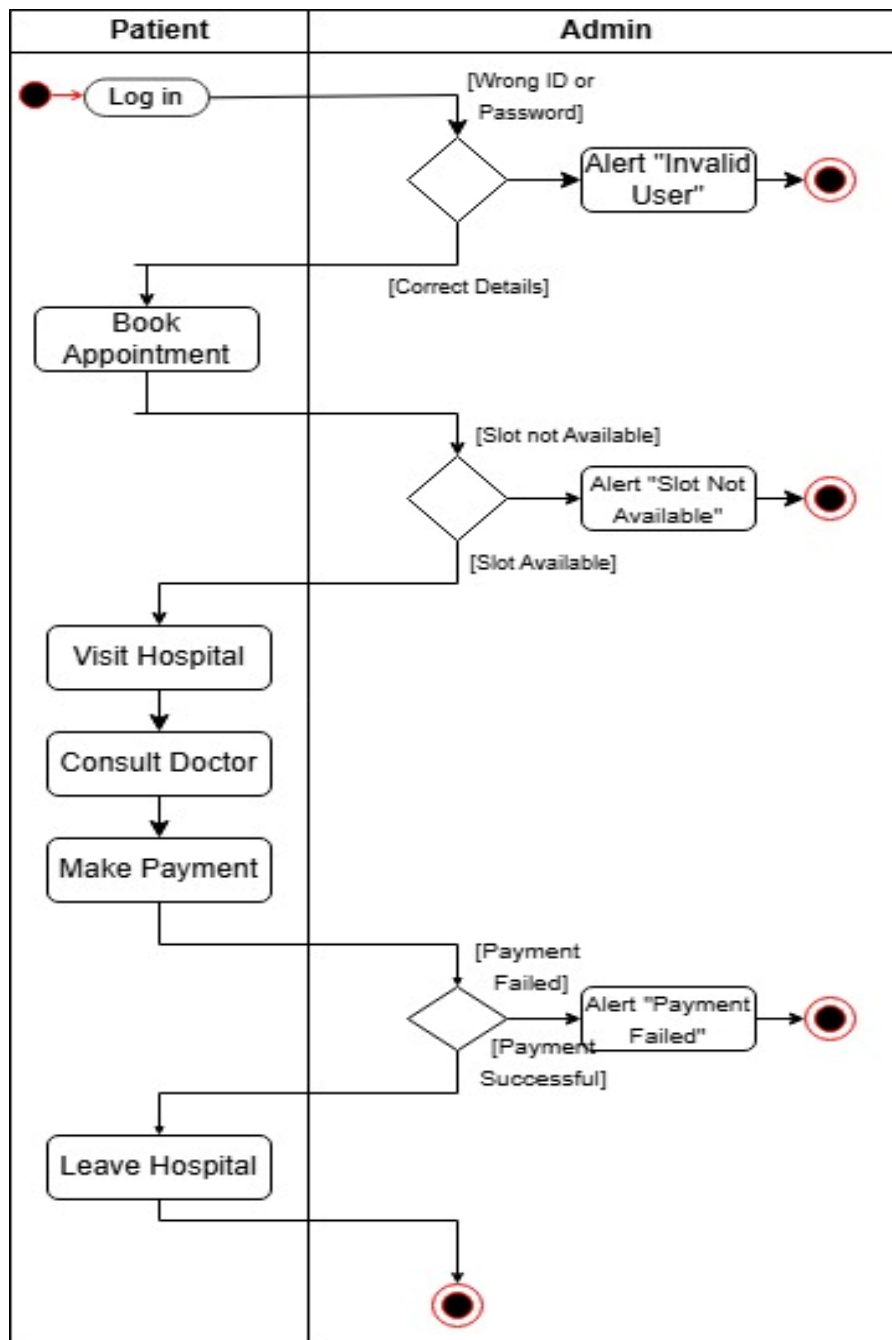


Figure 2.3-1 Swimlane diagram for Book Appointment

2.4 Sequence diagram- Book Appointment

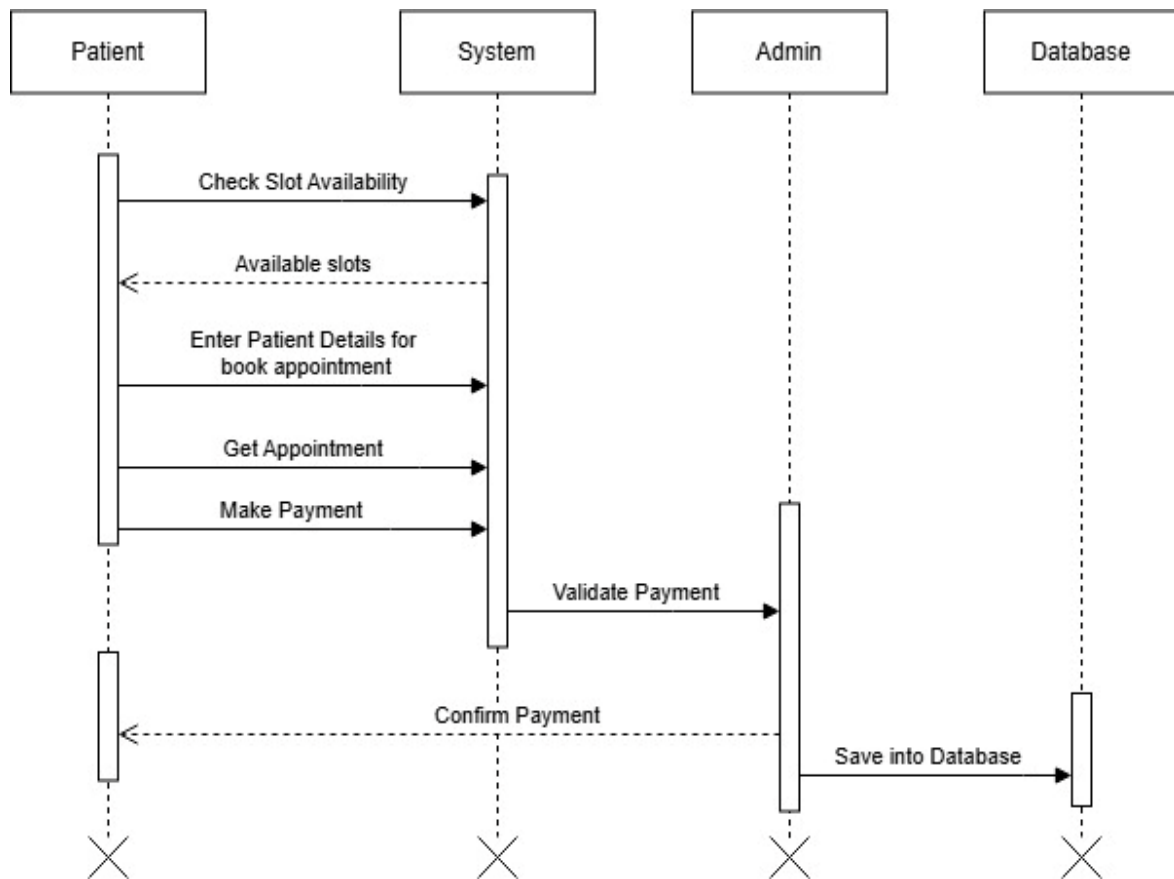


Figure 2.4-1 Sequence diagram for Book Issue

2.5 State diagram

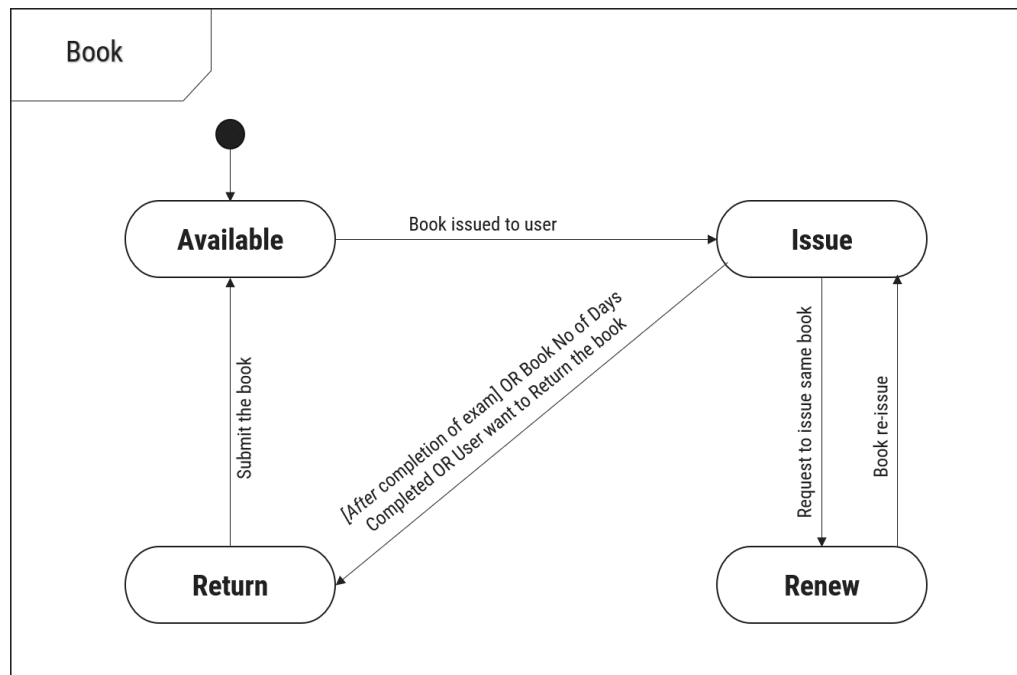


Figure 2.5-1 State diagram of Book

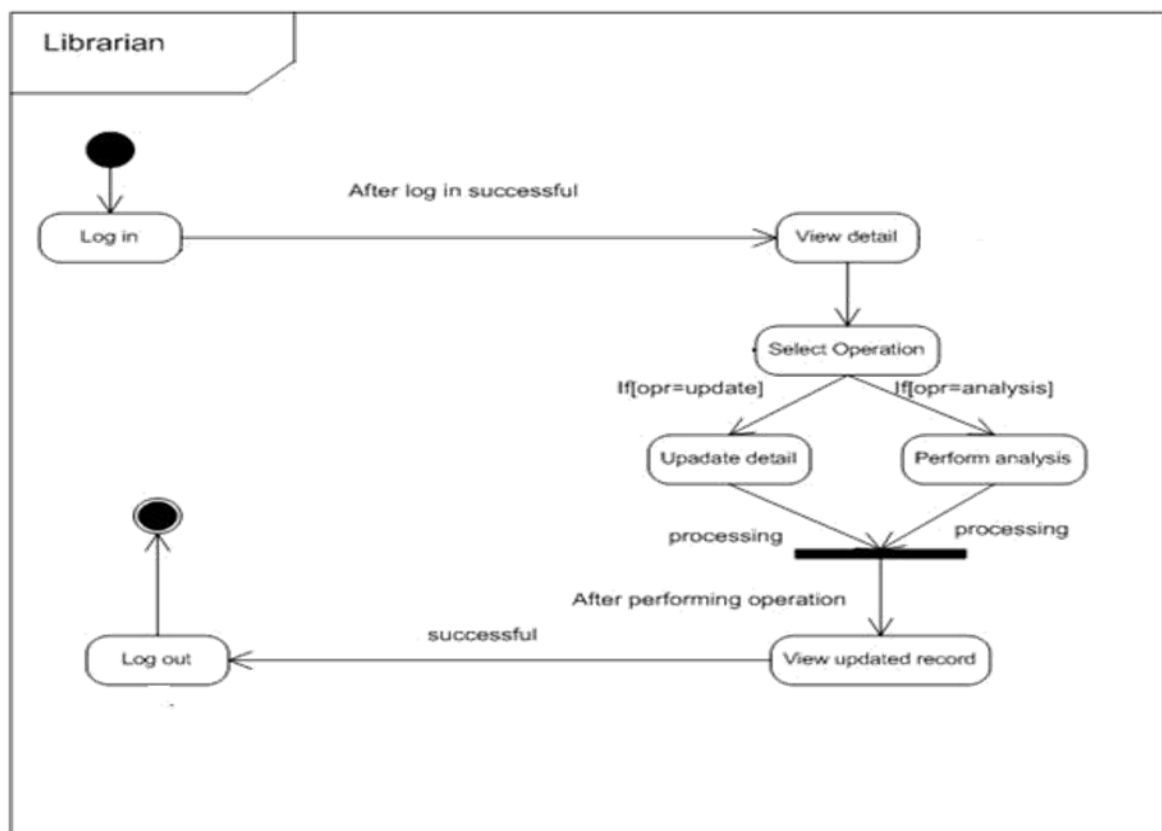


Figure 2.5-2 State diagram for Librarian

2.6 Class diagram

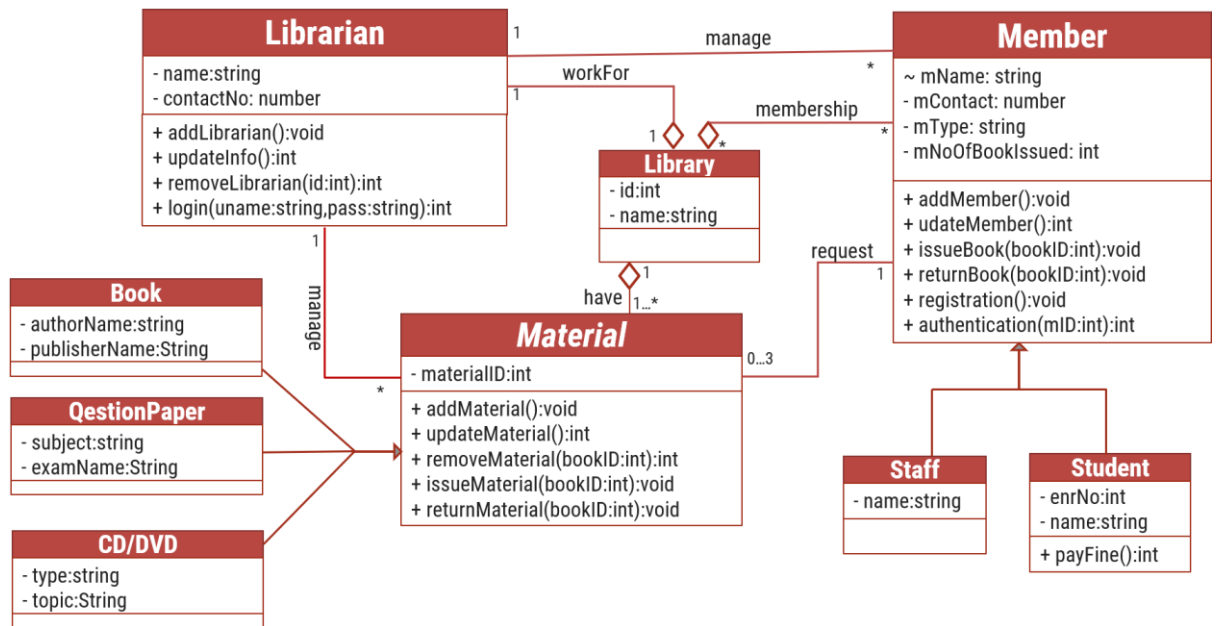


Figure 2.6-1 Class diagram for Library management system

2.7 Data flow diagram

2.7.1 Context diagram (level-0)

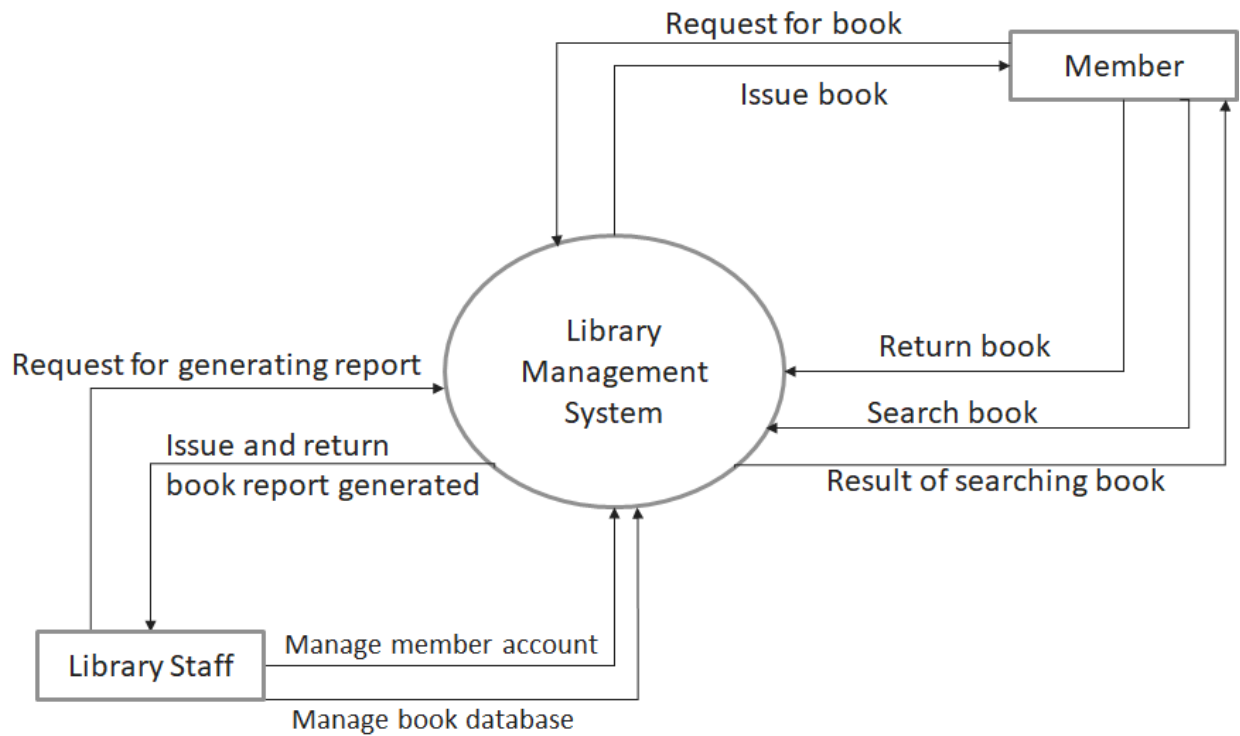


Figure 2.7-1 Context diagram for Library management system

2.7.2 DFD Level-1

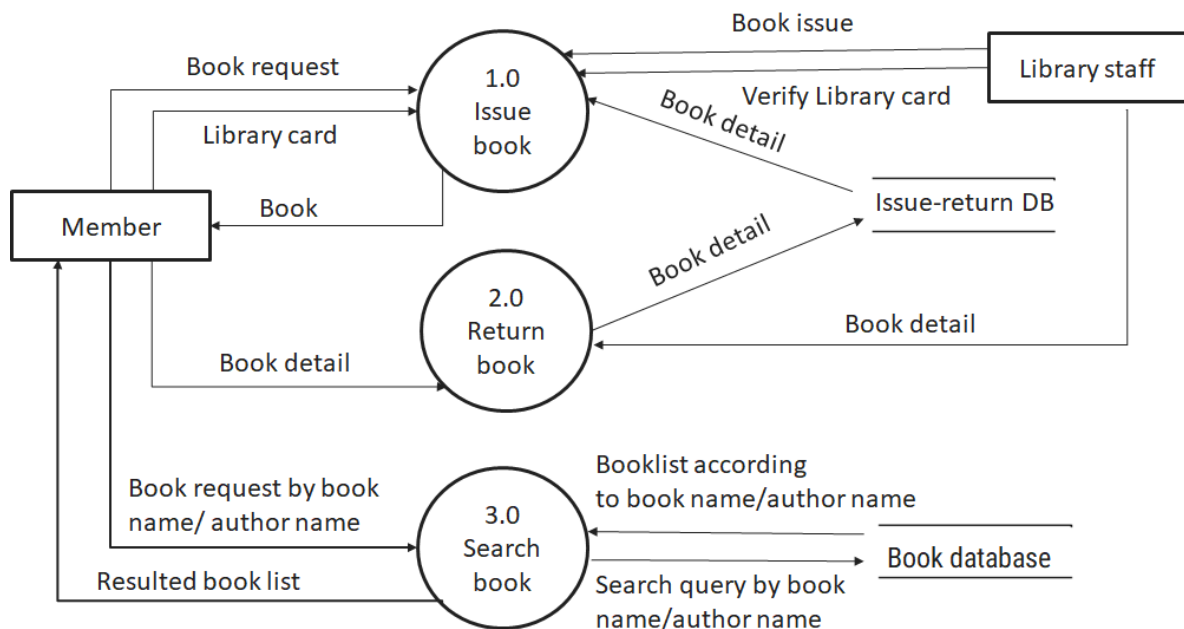


Figure 2.7-2 DFD level-1 for Library management system

2.7.3 DFD Level-2

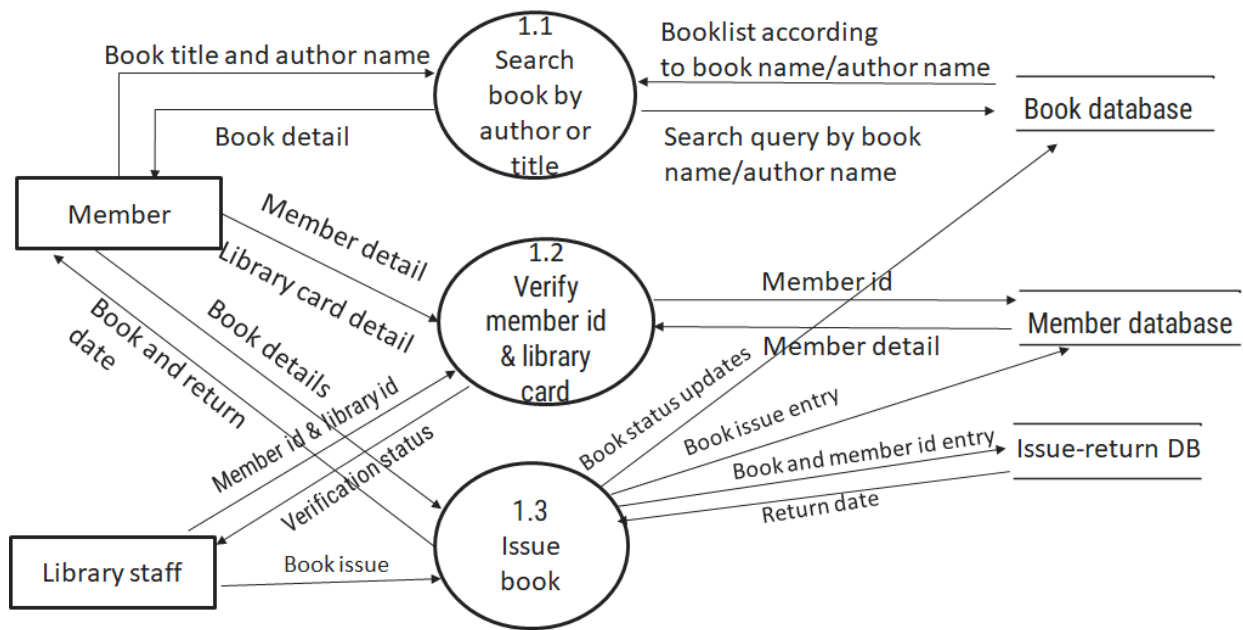


Figure 2.7-3 DFD level-2 for Issue book

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 

[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". It contains the following fields and controls:

- ID**: A text input field with a label "ID" above it.
- Student ID**: A text input field with a label "Student ID" above it.
- First Name**: A text input field with a label "First Name" above it.
- Middle Name**: A text input field with a label "Middle Name" above it.
- Last Name**: A text input field with a label "Last Name" above it.
- Contact**: A text input field with a label "Contact" above it.
- Category**: A text input field with a label "Category" above it.
- Grade ID**: A text input field with a label "Grade ID" above it.
- Buttons**: "Submit" (blue) and "Cancel" (grey) buttons at the bottom.

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>