**Software Requirements**

**Specification**

**For**

Library Management System

**Version 01**

**Prepared by**

|  |  |
| --- | --- |
| Helena Wahba | 202220184 |
| Fadya Hazem | 202220207 |
| Huda Tarek | 202320185 |
| Ghadah Mohammed | 202220293 |

|  |  |
| --- | --- |
| Instructor: | *Mohammed Reda* |
| Course: | Software Engineering |
| Teaching Assistant: | *Mohammed karam* |
| Date: | 13/11/2024 |

# Table of Contents

[Table of Contents 2](#_Toc181811587)

[1. Introduction 1](#_Toc181811589)

[1.1 Purpose 1](#_Toc181811590)

[1.2 Document Conventions 1](#_Toc181811591)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc181811592)

[1.4 Product Scope 1](#_Toc181811593)

[2. Overall Description 2](#_Toc181811595)

[2.1 Product Perspective 2](#_Toc181811596)

[2.2 Product Functions 2](#_Toc181811597)

[2.3 User Classes and Characteristics 2](#_Toc181811598)

[2.5 Design and Implementation Constraints 3](#_Toc181811600)

[2.7 Assumptions and Dependencies 4](#_Toc181811602)

[3. External Interface Requirements 4](#_Toc181811603)

[3.1 User Interfaces 4](#_Toc181811604)

[3.2 Hardware Interfaces 5](#_Toc181811605)

[3.3 Software Interfaces 5](#_Toc181811606)

[3.4 Communications Interfaces 5](#_Toc181811607)

[4. System Features 5](#_Toc181811608)

[4.1 User Account Management](#_Toc181811609)

[4.2 Book Inventory Management](#_Toc181811609)

[4.3 Book Borrowing and Returning System](#_Toc181811609)

[4.4 Book Reservation](#_Toc181811609)

[4.5 Notification and Reminder System 5](#_Toc181811609)

[5. Other Nonfunctional Requirements 7](#_Toc181811611)

[5.1 Performance Requirements 7](#_Toc181811612)

[5.2 Safety Requirements 8](#_Toc181811613)

[5.3 Security Requirements 8](#_Toc181811614)

[5.4 Software Quality Attributes 8](#_Toc181811615)

[5.5 Business Rules 9](#_Toc181811616)

**6. Wireframe Representation**

6.1 Log in Representation

6.2 Interface Representation

6.3 My-Books Interface Representation

**7. Data Flow Diagram & UML Diagrams**

**7.1 Data Flow Diagram**

7.1.1 DFD Context

7.1.2 DFD Level 1

7.1.3 DFD Level 2

**7.2 UML Diagram**

7.2.1 Class Diagram

7.2.2 Activity Diagram

7.2.3 Use case Diagram

7.2.4 Sequence Diagram

7.2.5 State Diagram

# 1. Introduction

## 1.1 Purpose

This SRS describes the requirements for a Digital Library Management System (DLMS) designed to manage a library’s book inventory, member accounts, and borrowing/return activities. It includes version 1.0 of the software, covering all modules of the system, including user management, inventory management, transaction, processing, and reporting.

## 1.2 Document Conventions

## Bold font is used for section headings and critical terms.

## Each requirement statement is numbered and prioritized individually

## 1.3 Intended Audience and Reading Suggestions

## This document is intended for:

## Developers – for understanding the core functionalities.

## Project Managers – for a high-level overview of system requirements.

## Testers – to create test cases based on system specifications.

## Documentation Writers – to create user manuals based on this SRS.

## Suggested reading order – Introduction, then proceed to specific requirements (User, Hardware, and Software Interfaces).

## 1.4 Product Scope

## The Digital Library Management System provides a digital solution for managing book inventory, user registration, and book borrowing/returning processes. It aims to reduce the manual workload associated with library management, improve accuracy, and provide easy access to book and member information

# 2. Overall Description

## 2.1 Product Perspective

The goal of this project is to provide simplicity as well as security and efficiency to the Management of Library and also to reduce the management personnel in the library. The Digital Library Management System is a computerized system developed to help both users and the library manager keep track of all book information in the library.

## 2.2 Product Functions

* Insert books.
* Can accept or reject a new user according to the library policy.
* Can get the information (status and arrange) of any book who has borrowed.
* Add and edit book categories and arrange books by categories.
* Send lateness warnings to people who have exceeded the deadline date.
* Can record books returned by users.
* The member should be provided with the updated information about the books catalog.
* Search through books by subject, title, authors, or any information related to the book.
* Can extend the period of borrowing books according to the library policy.

## 2.3 User Classes and Characteristics

* Library members: will check the availability of the books.
  + Book borrow/return
* Administrator: The following are the sub-modules in the administration module.
  + Register user
  + Entry book details
  + Book issue

## 2.4 Design and Implementation Constraints

* This system can also run on Android devices.
* The backend database for this is SQL Server.
* This product uses a login feature to make certain functionalities available to just certain readers.
* Users must be up to 14 years
* Must be developed using responsive web design principles.
* Must support major web browsers (Chrome, Firefox, Safari, Edge)

## 

## 2.5 Assumptions and Dependencies

* Are librarians and users comfortable with computers and have enough ability to work with the product?
* Website interface must be friendly and easy-to-use.
* The search mechanism should be simple and fast

3. External Interface Requirements

**3.1 User Interfaces**

## The DLMS will have a simple graphical user interface (GUI) that allows easy navigation and management of library tasks. Key interface features include:

## Dashboard: Displays an overview of current transactions and alerts for overdue books.

## Book Catalog Screen: Allows searching, adding, updating, and deleting books.

## Transaction Screen: Facilitates borrowing and returning processes with clear prompts for each step.

## 3.2 Hardware Interfaces

## The system must support barcode scanners for book and member ID entry.

## The system will operate on standard desktop computers with Windows OS and internet connectivity.

## 3.3 Software Interfaces

## Database: The system will use MySQL for database management to store book, member, and transaction data.

## Operating System: The application will be compatible with Windows 10 or higher. And the goes for android operating system.

## 3.4 Communications Interfaces

# The system will allow email notifications for overdue books or reminders.

# The system must support HTTP/HTTPS for remote access if needed, with secure login requirements.

# 4. System Features

## 4.1 User Account Management

**4.1.1** **Functional Requirements**:

* **FR1**: Users can check book availability before borrowing.
* **FR2**: The system should record each transaction, associating it with a user account and due date.
* **FR3**: Late returns will trigger a penalty notice.
* **FR4**: Staff can override or waive late fees.

#### **4.2 Book Inventory Management**

#### **4.2.1** **Functional Requirements**:

#### **FR1:** Staff can add new books with details like title, author, ISBN, and category.

#### **FR2:** Books can be marked as available or borrowed.

#### **FR3:** The system should keep track of total and available book copies.

**4.3 Book Borrowing and Returning System**

**4.3.1. Functional Requirements:**

**FR1:** Users can check book availability before borrowing.

**FR2:** The system should record each transaction, associating it with a user account and due date.

**FR3:** Late returns will trigger a penalty notice.

**FR4:** Staff can override or waive late fees.

**4.4 Book Reservation**

**4.4.1.** **Functional Requirements:**

**FR1:** Users can reserve books and receive notifications when the book is available.

**FR2:** Reserved books will be held for 24 hours once available before being released back to inventory.

**4.5 Notification and Reminder System**

**4.5.1. Functional Requirements:**

**FR1:** Automatic notifications should be sent via email or SMS.

**FR2:** Reminders should be configurable (e.g., sent 3 days before the due date).

**5. Other Non-functional Requirements**

**5.1 Performance Requirements**

Response Time: System should respond within 2 seconds under normal load.

Throughput: Must handle at least 100 concurrent users without performance issues.

Scalability: Capable of horizontal scaling to manage user/transaction growth.

Availability: 99.9% uptime with scheduled maintenance communicated in advance.

Load Handling: Support peak loads (e.g., registration periods) without major slowdowns.

**5.2 Safety Requirements**

Data Integrity: Ensure accurate data processing without loss/corruption.

Error Handling: Graceful handling with informative, non-revealing error messages.

Backup & Recovery: Nightly automated backups; data restoration within 24 hours.

**5.3 Security Requirements**

Authentication: Secure, multi-factor methods for sensitive access

Authorization: Role-based access control (e.g., librarians vs. patrons)

Data Encryption: Encrypt sensitive data at rest and in transit

Audit Logging: Maintain logs of transactions and access for monitoring

Vulnerability Management: Regular security assessments to mitigate risks.

**5.4 Software Quality Attributes**

Usability: Intuitive interface with minimal user training.

Maintainability: Modular code design for easy updates.

Portability: Run on various platforms with minimal changes.

Interoperability: Integrate with external systems via standard APIs.

Testability: Support automated testing for efficient updates.

**5.5 Business Rules**

Membership Management: Enforce rules on types, renewals, and eligibility.

Loan Policies: Configurable policies for loan durations and limits.

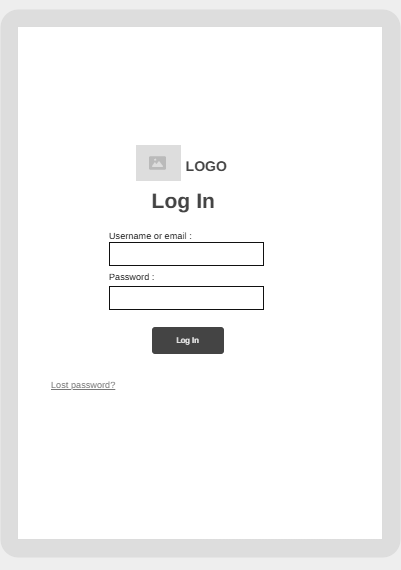
Fines/Fees: Automated fine calculation and user notifications.

Cataloging: Adhere to standards like MARC and Dublin Core.

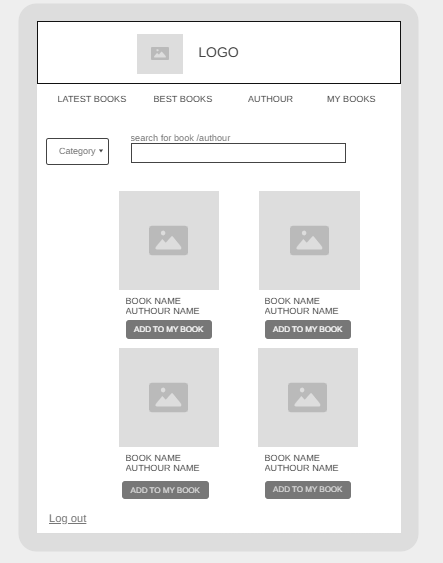
Reporting: Generate detailed reports on usage, inventory, and finances.

**6. Wireframe Representation**

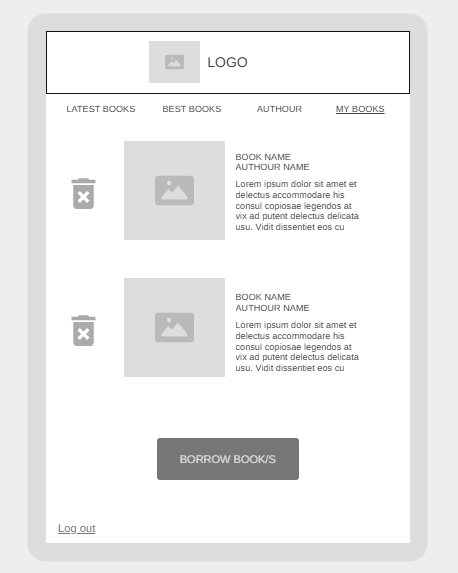
**6.1 Log in Representation**

****

**6.2 Interface Representation**

****

**6.3 MY-BOOKS Representation**

****

**7. Data Flow Diagram & UML Diagrams**

**7.1 Data Flow Diagram**

**7.1.1 DFD Context**

[**DFD LEVEL CONTEXT.eddx**](DFD%20LEVEL%20CONTEXT.eddx)

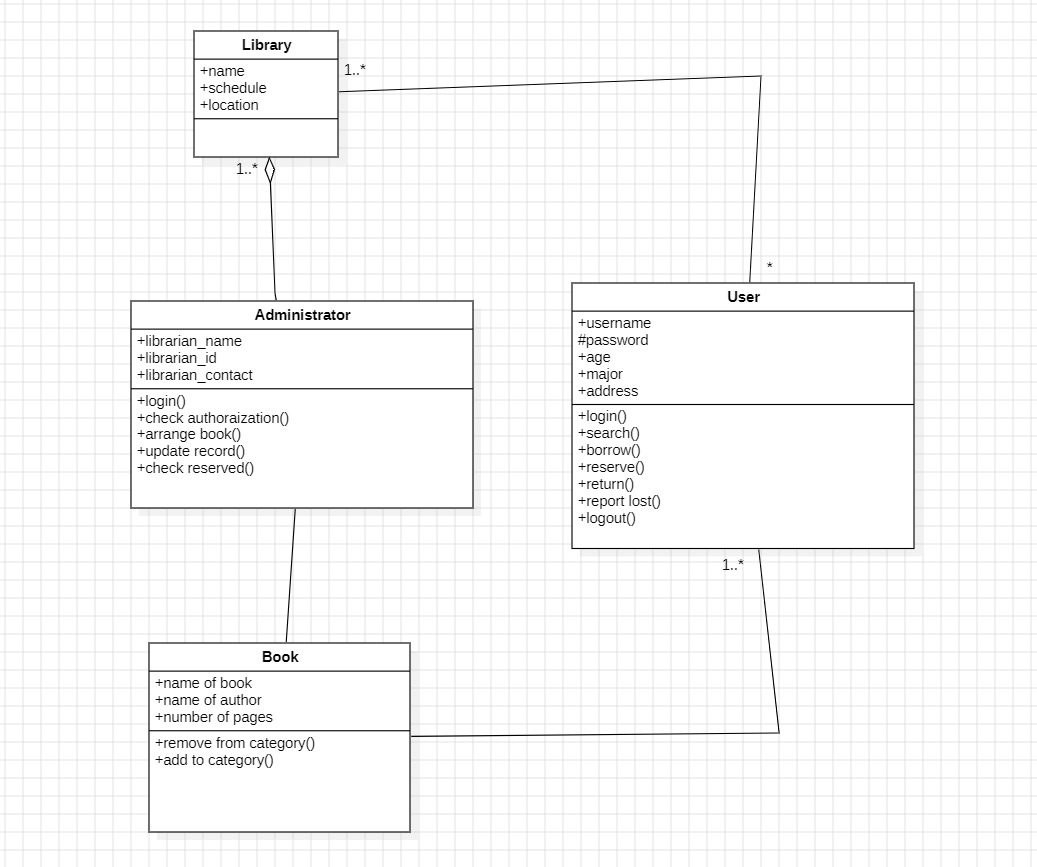
**7.1.2 DFD Level 1**

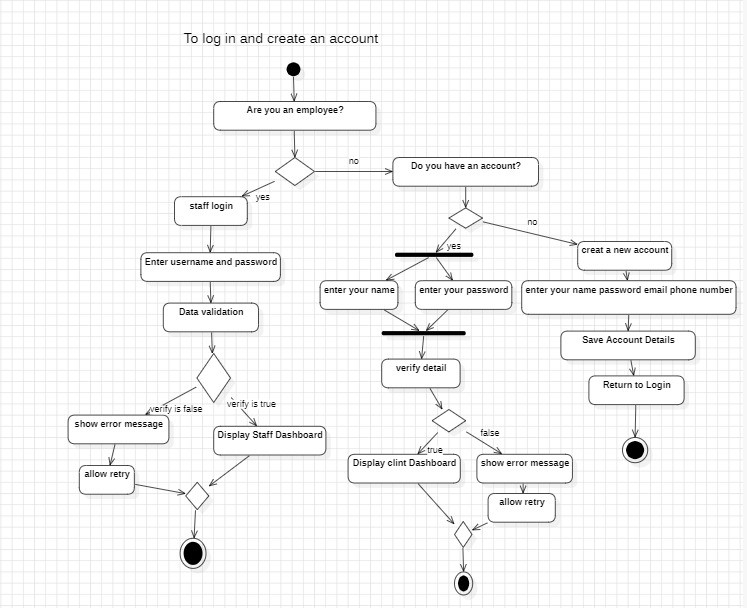
[**DFD LEVEL 1.eddx**](DFD%20LEVEL%201.eddx)

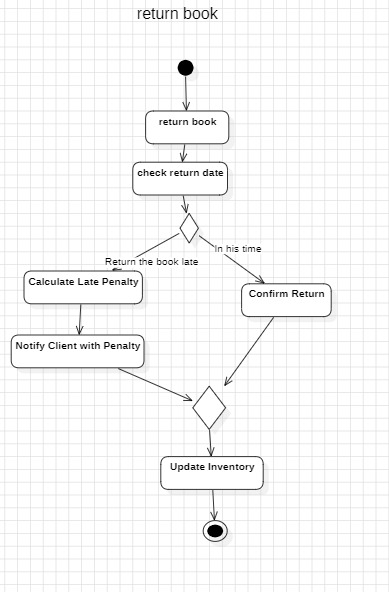
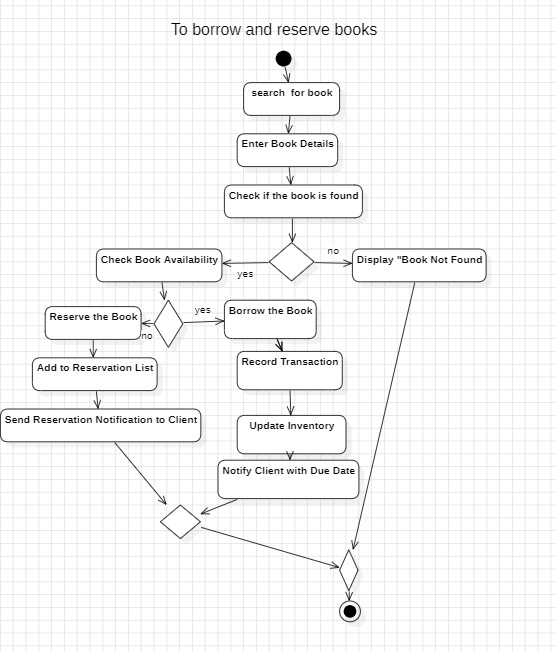
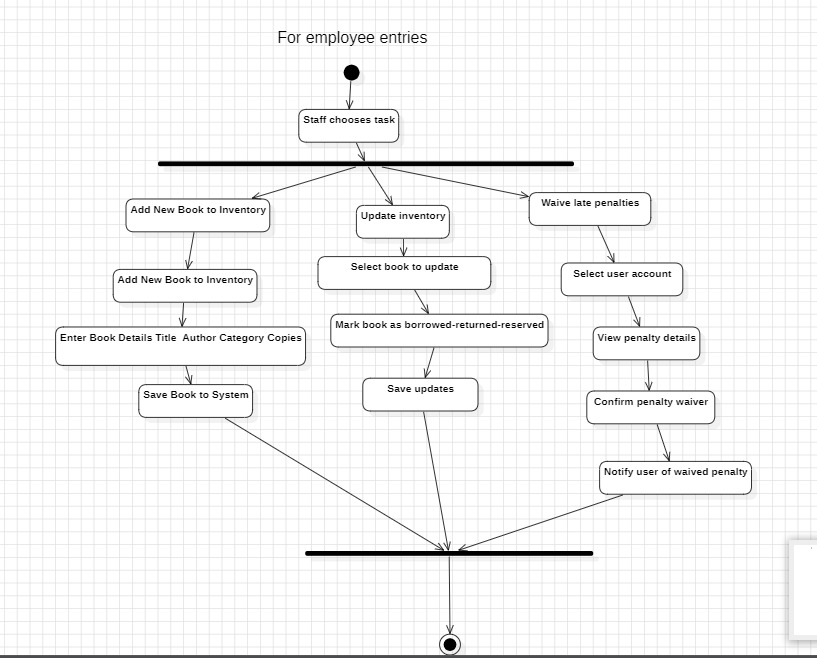
**7.1.3 DFD Level 2**

[**DFD LEVEL 2.eddx**](DFD%20LEVEL%202.eddx)

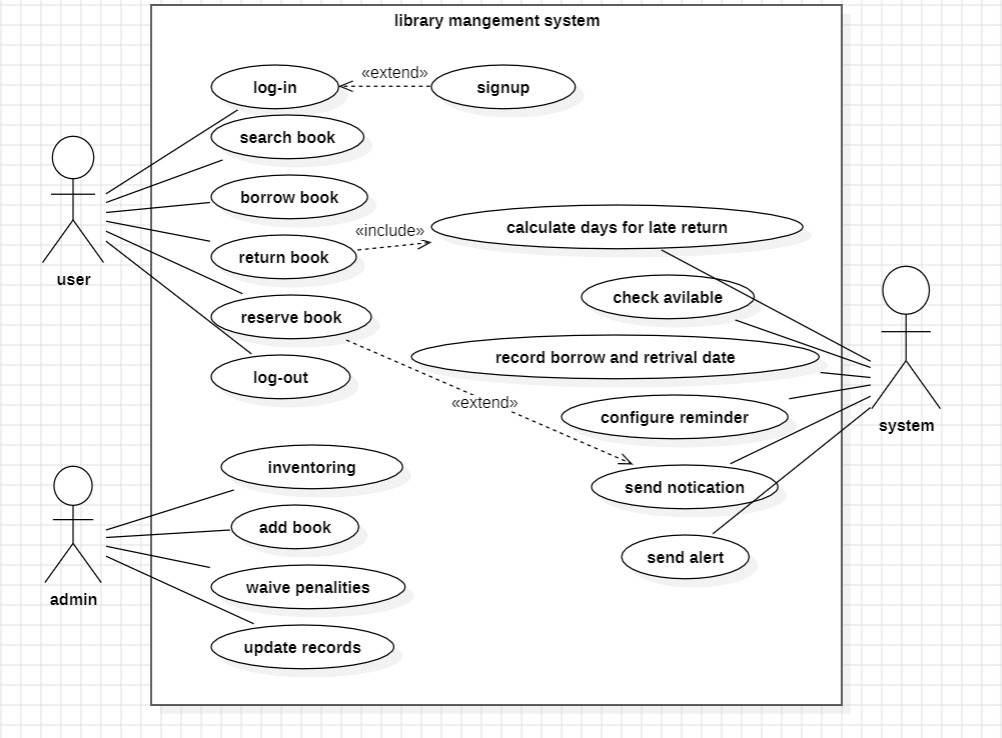
**7.2 UML Diagram**

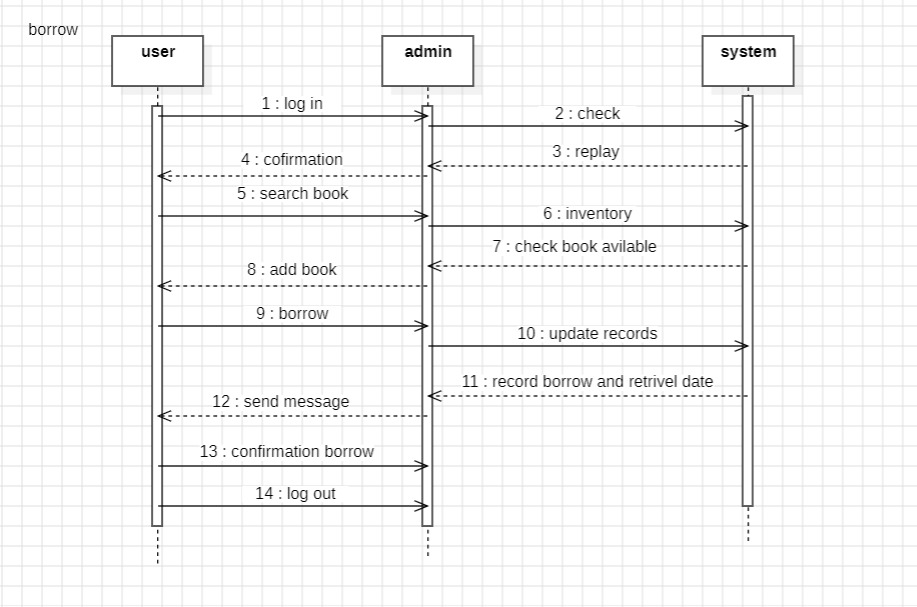
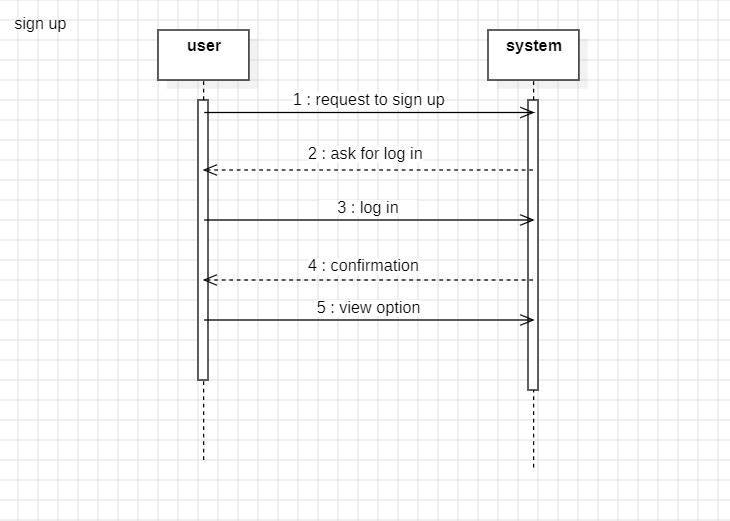
** 7.2.1 Class Diagram**

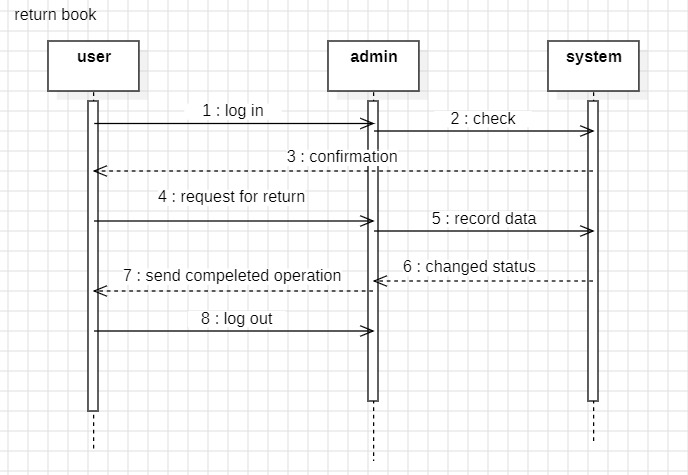
** 7.2.2 Activity Diagram**

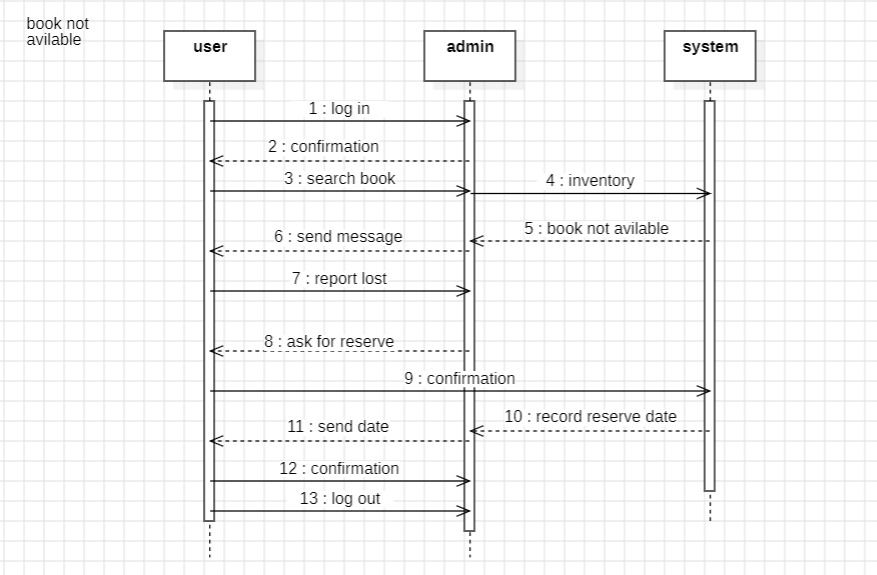
****

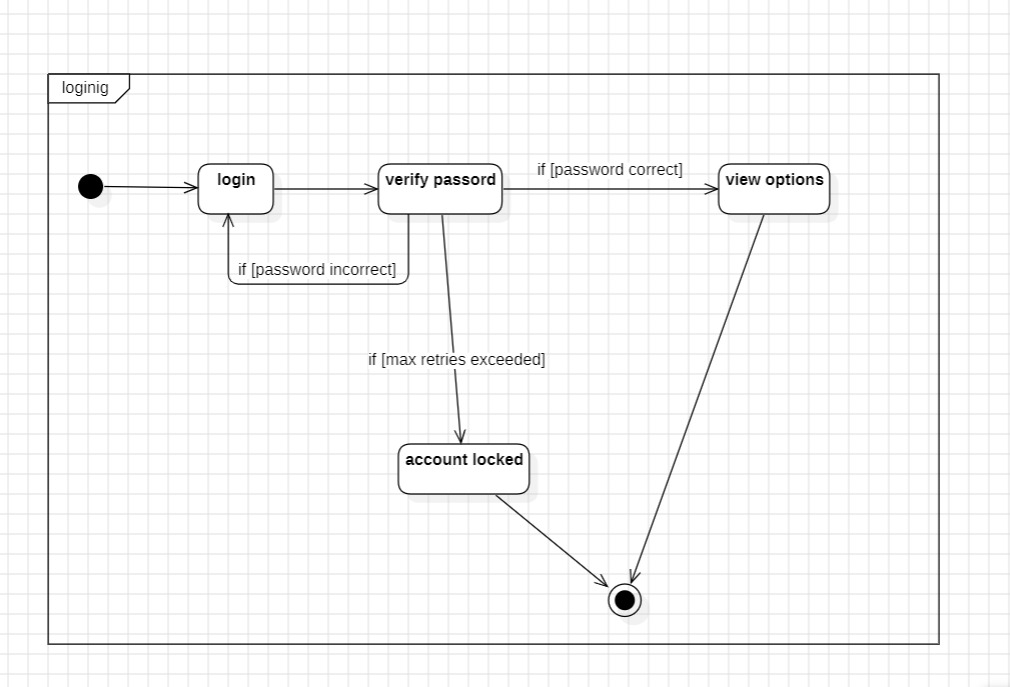
**7.2.3 Use case Diagram**

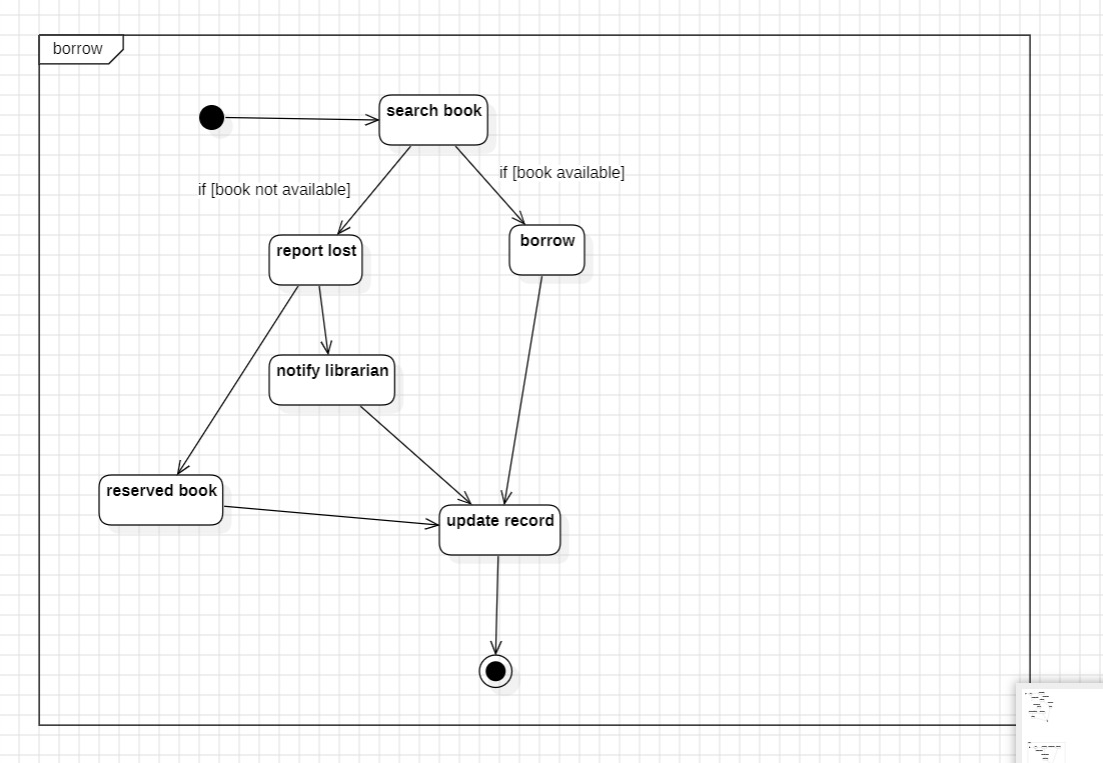
****

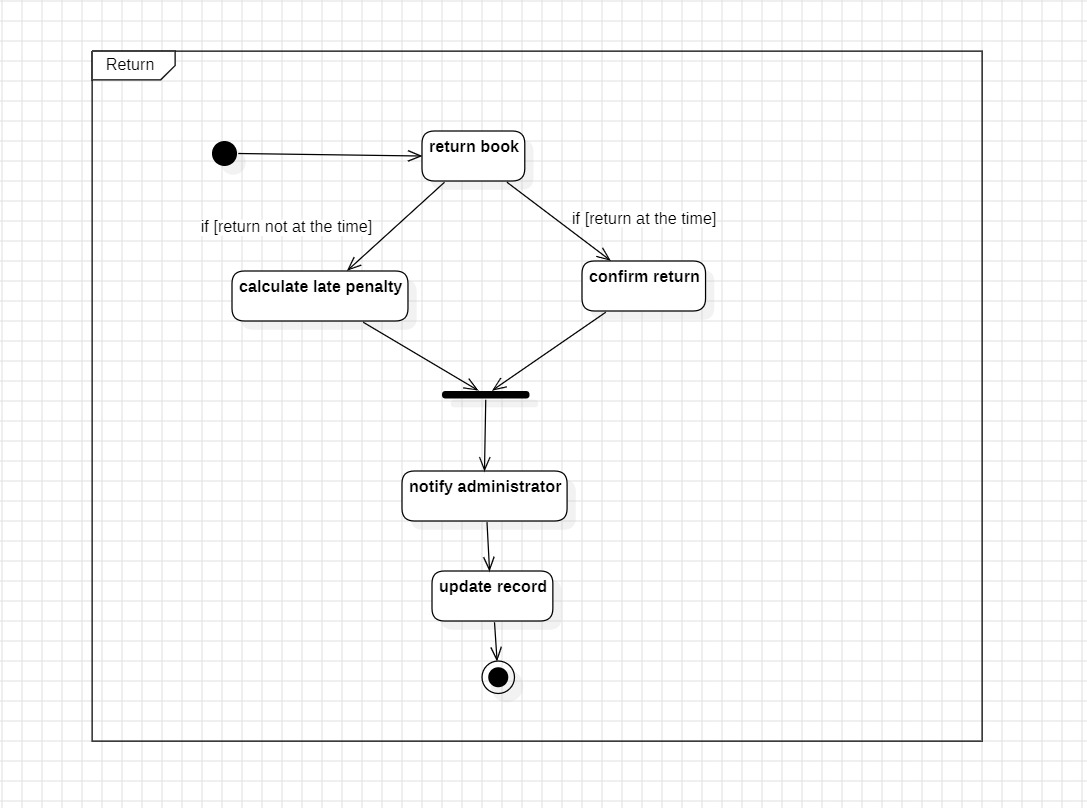
**7.2.4 Sequence Diagram**

****

**Reserve a book**

**7.2.5 State Diagram**

****

****

# Made with our blood, sweat and tears