Library Management System

(SRS Report)

College Name: Lovely Professional University
Department: SCHOOL OF COMPUTER APPLICATION

Members: Amar Kumar Nayak(12318243)
Alen p shyju(12312257)
Ujjaval Saini(12312886)

Team Guide: Prof. Kamal Nain Sharma

Table of Content

1 INTRODUCTION

A library management system is a sophisticated software application designed to streamline and enhance the operations of a library or information resource center. In an age where digital technology is rapidly transforming the way we access and consume information, a well-structured library management system has become an essential tool for libraries to efficiently catalog, organize, and provide access to a wide array of resources. This system helps librarians and library staff manage their collections, assist patrons, and maintain the overall functionality of the library. This introduction explores the core components and benefits of a library management system, highlighting its crucial role in modernizing and improving the traditional library experience. Whether in academic institutions, public libraries, or corporate settings, the library management system plays a pivotal role in maintaining an organized and easily navigable repository of knowledge.

1.1 PURPOSE

The purpose of a library management system (LMS) is multifaceted, serving as a vital tool for libraries to streamline their operations and enhance the overall experience for both library staff and patrons. At its core, an LMS is designed to efficiently catalog, organize, and maintain an extensive array of library resources, from books and multimedia to digital collections. It simplifies inventory management, automates circulation processes, and enables precise tracking of user information, such as library card issuance and patron preferences. The LMS empowers patrons to easily search for and retrieve materials, both within the library and remotely, fostering self-service and expanding access to knowledge. Furthermore, these systems generate valuable reports and analytics that assist libraries in making informed decisions about resource acquisition, usage patterns, and budget allocation. In essence, a library management system is a versatile and indispensable tool that modernizes and optimizes library operations, ensuring that libraries remain vibrant hubs of learning and information in today's digital age.

1.2 SCOPE

A Library Management System (LMS) plays a crucial role in modernizing and enhancing the efficiency of library operations. Its scope encompasses a wide range of functions aimed at streamlining the management of library resources and improving the overall user experience. LMSs are designed to handle catalog management, including cataloging, classification, and indexing of library materials, ensuring accurate

and accessible records. They facilitate user management, enabling library patrons to register, borrow, and manage their accounts seamlessly. LMSs also automate circulation management, tracking check-in, check-out, renewals, and holds while maintaining due dates and late fee records. Additionally, they aid in acquisition and budget management, helping libraries select, order, and manage new materials while adhering to budget constraints. These systems enable users to search and discover resources efficiently, whether in physical or digital formats. They offer features for managing digital resources and may support interlibrary loans. Furthermore, LMSs generate valuable reports and analytics to inform collection development and library decision-making, contributing to the continuous improvement of library services. With their security features, integration capabilities, and tools for user services and staff management, LMSs encompass the essential functions required to modernize and streamline library operations and services.

1.3 Definition, Acronyms, Abbreviation:

A library management system (LMS) is a software application that helps libraries to manage their collections, users, and circulation services. LMSs typically include features such as:

JAVA -> platform independence •

SQL -> Structured query Language •

DFD -> Data Flow Diagram

• CFD -> Context Flow Diagram

• ER -> Entity Relationship

• IDE -> Integrated Development Environment

• SRS -> Software Requirement Specification

2. OVERALL DESCRIPTION

2.1 PRODUCT PRESPECTIVE

The proposed Library Management System will take care of the current book detail at any point of time. The book issue, book return will update the current book details automatically so that user will get the update current book details.

2.2 SOFTWARE REQUIREMENT

- Front end:
  - Android developer tool
  - Advance java
- Back end:
  - MySQL

2.3 HARDWARE REQUIREMENT

- Android version 4.4 KitKat (minimum, android user's)
- 2GB ram
- Intel i3 (minimum)
- 1.2 GHz processor
- Windows 7/8/8.1/10

2.4.1 FUNCTIONAL REQUIREMENT

- **R.1:Register**
  - Description : First the user will have to register/sign up. There are two different type of users.
  - The library manager/head : The manager have to provide details about the name of library ,address, phone number, email id.
  - Regular person/student : The user have to provide details about his/her name of address, phone number, email id.

- **R.1.1: Sign up**
  - Input: Detail about the user as mentioned in the description.
  - Output: Confirmation of registration status and a membership number and password will be generated and mailed to the user.
  - Processing: All details will be checked and if any error are found then an error message is displayed else a membership number and password will be generated.

- **R.1.2 : Login**
  - Input: Enter the membership number and password provided.
  - Output : User will be able to use the features of software.

- **R.2 : Manage books by user**

  - **R.2.1 : Books issued.**
    - Description : List of books will be displaced along with data of return.

- **R.2.2 : Search**
  - Input : Enter the name of author's name of the books to be issued.
    Output : List of books related to the keyword.

- **R.2.3 : Issues book**
  - State : Searched the book user wants to issues.
  - Input : click the book user wants.
  - Output : conformation for book issue and apology for failure in issue.
  - Processing : if selected book is available then book will be issued else error will be displayed.
  -

- **R.2.4 : Renew book**
  - State : Book is issued and is about to reach the date of return.
  - Input : Select the book to be renewed.
  - Output : conformation message.
  - Processing : If the issued book is already reserved by another user then error message will be send and if not then conformation message will be displayed.

- **R.2.5 : Return**
  - Input ; Return the book to the library.
  - Output : The issued list will be updated and the returned book will be listed out.

- **R.2.6 ; Reserve book**
  - Input ; Enter the details of the book.
  - Output : Book successfully reserved.
  - Description : If a book is issued by someone then the user can reserve it ,so that later the user can issue it.

- **R.2.6 Fine**

- Input : check for the fines.
- Output : Details about fines on different books issued by the user.
- Processing : The fine will be calculated, if it crossed the date of return and the user did not renewed if then fine will be applied by Rs 10 per day.

❖ **R.3 Manage book by librarian**

- **R.3.1 Update details of books**
  - Update the list of the books available

- **R.3.1.1 Add books**
  - Input : Enter the details of the books such as names ,author ,edition, quantity.
  - Output : confirmation of addition.

- **R.3.1.2 Remove books**
  - Input : Enter the name of the book and quantity of books.
  - Output : Update the list of the books available.

2.4.2 Non Functional Requirements

➢ **Usability**

- User-Friendly Interface: The system should have an intuitive and user-friendly interface that allows both library staff and patrons to easily navigate and perform tasks.
- Accessibility: The LMS should be accessible to individuals with disabilities, complying with relevant accessibility standards (e.g., WCAG).

➢ **Performance**

- Response Time: The system should respond to user actions within an acceptable time frame. For example, searches should return results promptly. The LMS should be highly reliable, minimizing downtime and data loss.

➢ **Security**

- Authentication: Implement secure authentication and authorization mechanisms to ensure that only authorized users can access specific features and data. Use encryption to protect sensitive data, such as user information and transaction records. Regularly back up data and have a plan in place for data recovery in case of system failures.

➢ **Data Management**

- Data Integrity: Ensure the integrity of the data by implementing mechanisms to prevent data corruption and unauthorized alterations. Data Privacy,Comply with data privacy regulations and protect the privacy of user information.

➢ **Integration**

- Integration with External Systems: The LMS should be able to integrate with external systems such as online catalogs, e-books, and databases.

➢ **Maintainability**

- Modularity: Design the system in a modular fashion, making it easier to maintain, update, and extend in the future.
- Documentation: Provide comprehensive documentation for system administrators, developers, and end-users.
- Version Control: Implement version control for software updates and changes.

➢ **Performance Testing**

- Load Testing: Conduct load testing to ensure the system can handle peak usage periods without performance issues.
- Stress Testing: Test the system's response under extreme conditions to identify potential failure points.

➢ **Scalability**

- Vertical Scalability: The system should support scaling vertically by upgrading hardware resources to meet increasing demands.
- Horizontal Scalability: The LMS should be able to scale horizontally by adding more servers or nodes to distribute the load.

➢ **Error Handling**

- Error Logging and Reporting: Implement a robust error handling system that logs and reports errors to facilitate troubleshooting and debugging.

➢ **Compliance and Standards**

- Compliance with Library Standards: Ensure that the system complies with library cataloging and classification standards such as MARC (Machine-Readable Cataloging) and Library of Congress Subject Headings (LCSH).

➢ **Geographical and Language Support**

- Multilingual Support: The system should support multiple languages to accommodate diverse user groups.
- Geographical Considerations: Consider time zone and currency support, especially for libraries serving users in different regions.

2.5 USER CHARACTERSTICS

We have 3 levels of Users

❖ **User module:**

In the user module, user will check the availability of the books.

- Issue book
- Reserve book
- Return book
- Fine details

❖ **Library module:**

- Add new book
- Remove books
- Update details of book

❖ **Administration module:**

The following are the sub module in the administration module :

- Register user
- Entry book details
- Book issue

2.6 CONSTRAINTS

Functional constraints

- The LMS must be able to support the core library functions of adding, deleting, and updating book records, as well as issuing and returning books to users.
- The LMS must be able to generate reports on library usage, such as the number of books checked out, the most popular books, and overdue books.
- The LMS must be able to be accessed by multiple users simultaneously, including library staff and patrons.

Non-functional constraints

- The LMS must be easy to use and navigate, for both library staff and patrons.
- The LMS must be secure, to protect user data and prevent unauthorized access to the system.
- The LMS must be scalable to support a growing library collection and user base.
- The LMS must be reliable and available 24/7.
- The LMS must be affordable to purchase and maintain.

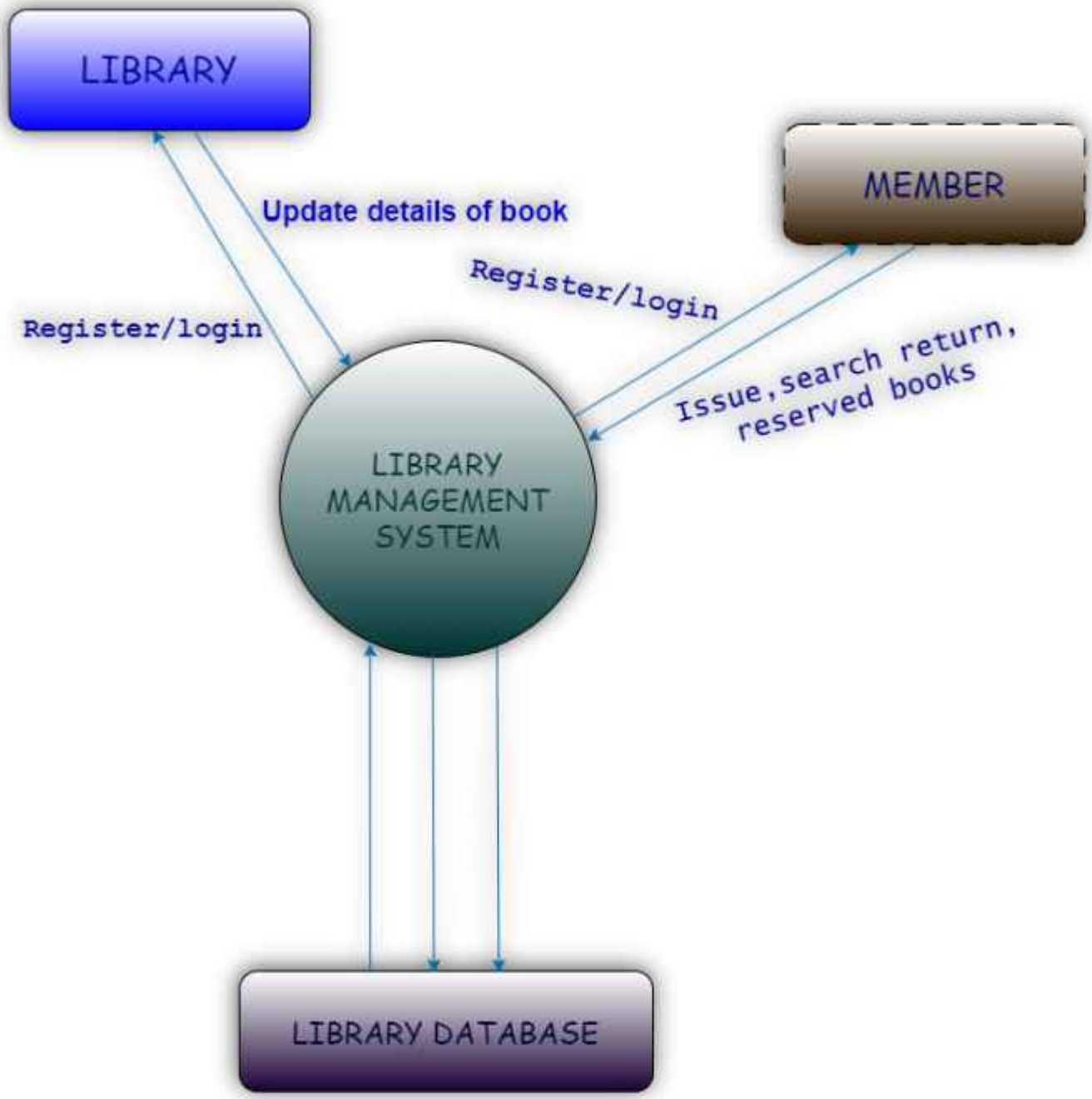
Other constraints

- The LMS must be compatible with the library's existing hardware and software infrastructure.
- The LMS must be compliant with all applicable laws and regulations.
- The LMS must meet the specific needs of the library, such as its type of collection, user base, and budget.

2.7 Assumptions and Dependencies

- The users have sufficient knowledge of computers.
- The servers on which system is to be hosted should have Internet connection and Internetserver capabilities.
- The users know the English language, as the user interface will be provided in English
- The system can access and modify the Library database
- The product needs following third party applications for the development of the project:
    - Android Studio (for development of android based applications)
    - Netbeans
    - Photoshop (for editing layouts, icons, buttons, etc)

2.8 Flow chart

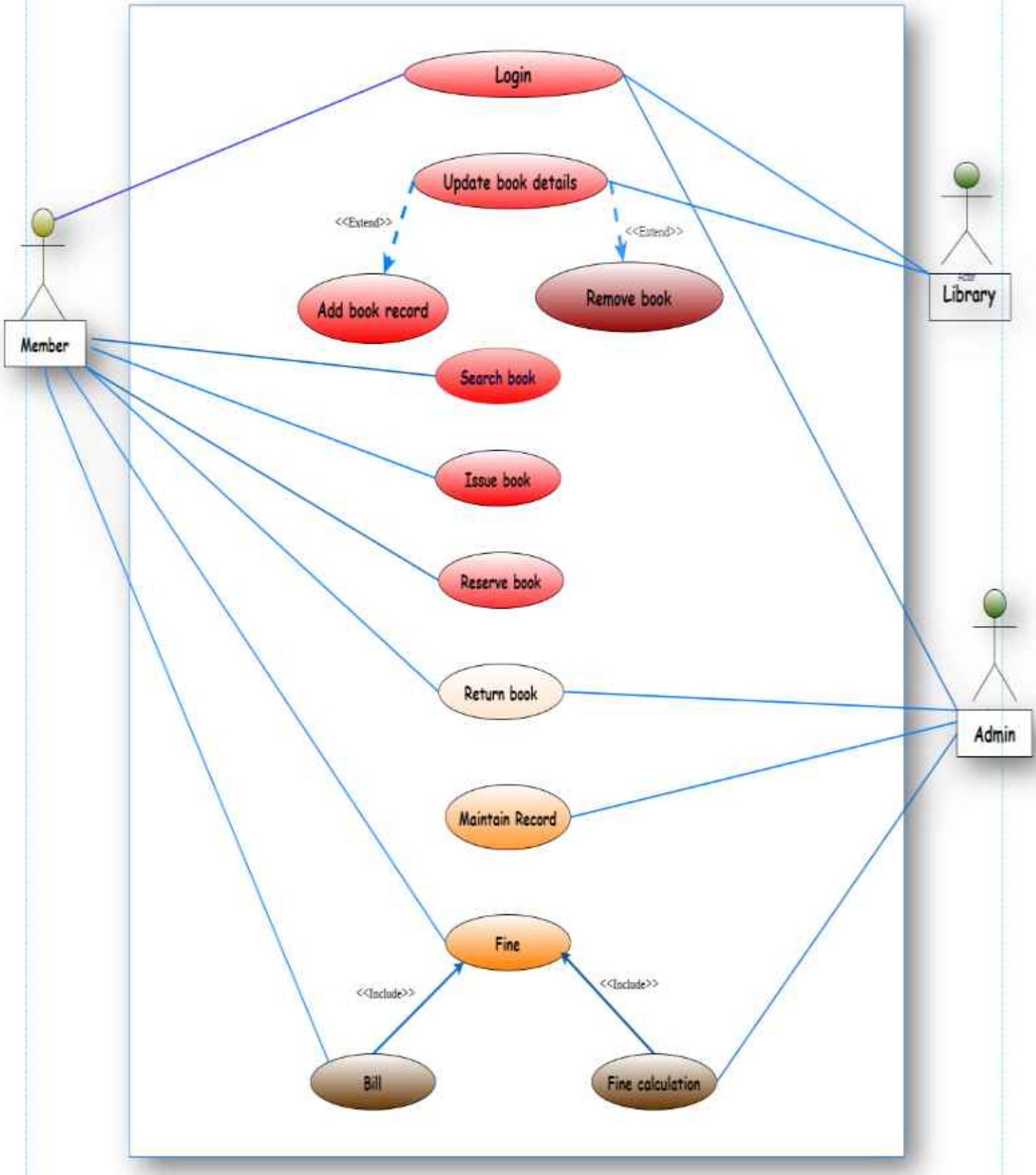


2.9 USE CASE MODEL DESCRIPTION

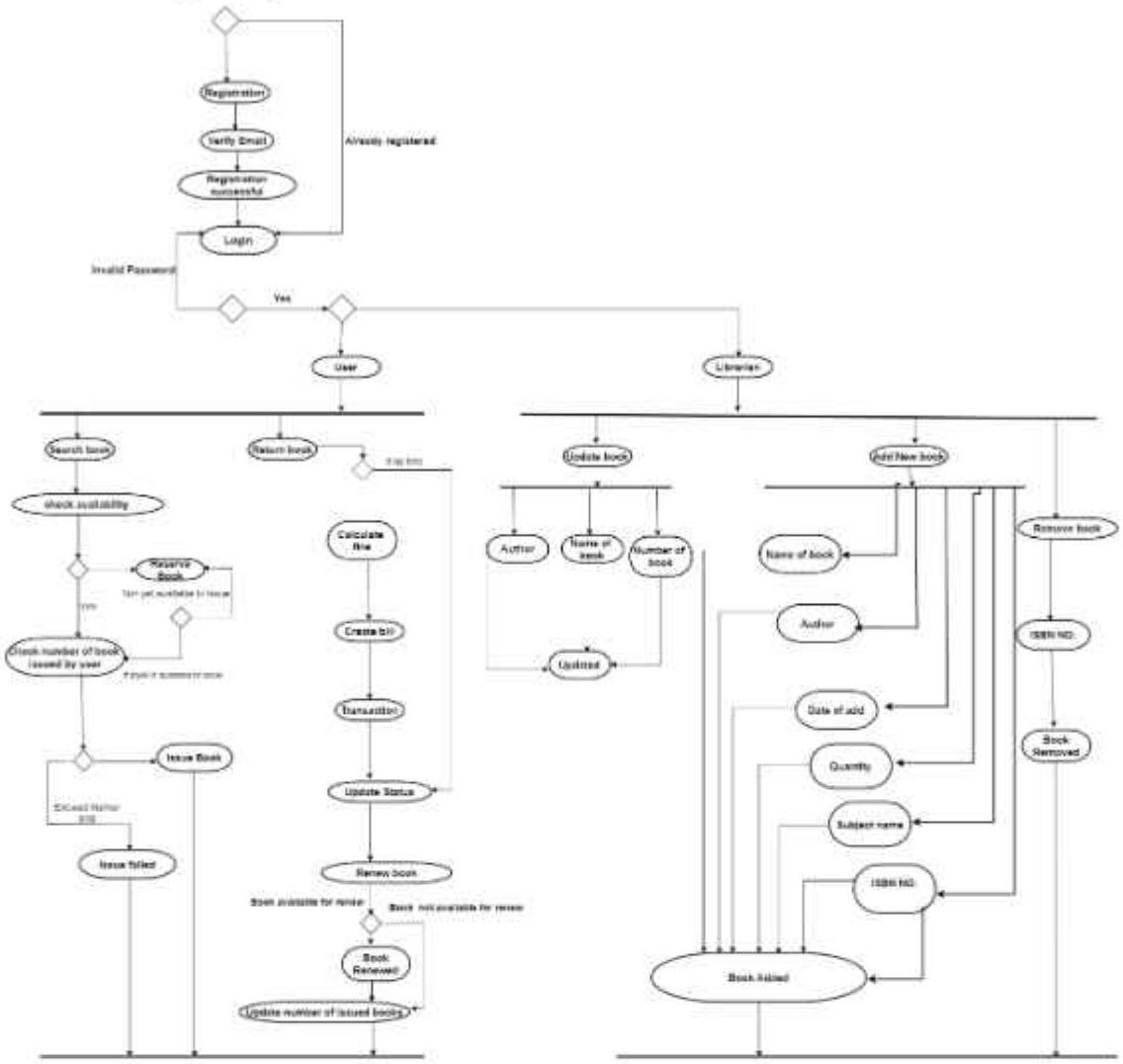
Use Case selection	Description
Use Case name	Add student and library record
Level	Sub-Functional level
Primary actor	Student, Library
Stakeholders and interest	Student: wants to register into the system. Library: wants to register into the system and update book details. Administrator: responsible for the management of the transaction of fine and also login and register details.
Pre-condition	Students and Library have submitted their registration form.
Post-condition	Record for a student/library has been added.
Main success scenario	1 Student/Library opens the application to access the services of the LMS 2 Student/Library sign-up to get registered online. 3 He/She provides correct information and secret password. 4 He/She got registered.
Alternative flow	1 Student/Library opens the application in their phone 2 He/She tries to sign-up 3 He/She fails and receives an error 4 He/She will report an error and the error will be rectified as soon as possible.
Specific requirement	□ The response time for registration is 1 minute. □ The response time for login is 1 minute

Table 1: table for use case description.

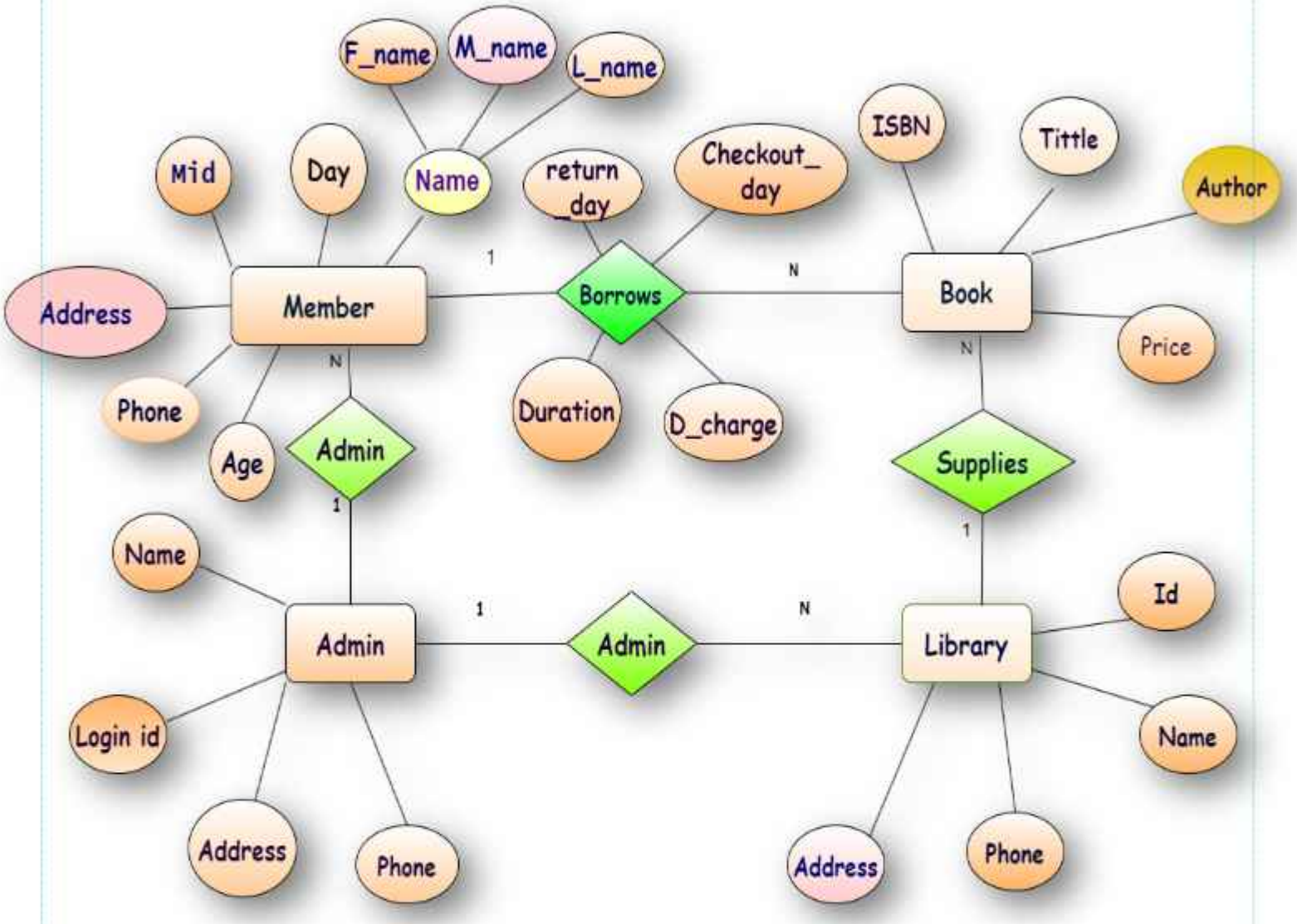
2.10 USE CASE DIAGRAM



2.11 Activity Diagram



2.12 ER Diagram



Thank YOU