**Department of Collegiate & Technical**

**Education**

**Government Polytechnic Mudhol – 164**

**Department of Computer Science & Engineering**

**LIBRARY MANAGEMENT SYSTEM**

**SOFTWARE REQUIREMENT SPECIFICATION(SRS)**

**SUBMITTED BY: GROUP 01**

**MEMBERS**:

* + - 1. Abdulmathin Golandaj
      2. Drakshayini Navi
      3. Akash Talawar
      4. Akshata Gadeppanavar
      5. Allabaksha Sanadi
      6. Araman Inamadar
      7. Bharat Hadapad
      8. Daneshwari Sunkad

**TEAM GUIDE:**

**Library Management System**

**Content :-**

1. **Introduction**
   1. Purpose
   2. Scope
   3. Definitions, Acronyms and Abbreviations
2. **Overall Description**
   1. Product Perspective
   2. Software Requirement
   3. Product Functions
   4. User Characteristics
   5. Constraints
   6. Flow Diagram
   7. Use Case diagram
   8. ER diagram
3. **Specific Requirements**
   1. Functional Requirements
   2. Non-Function Requirements
4. **System Features**
   1. System Login
   2. Add a new book
   3. Register anew user
   4. Issue a book
5. **Appendices**
   1. Glossary of Terms
   2. References
   3. Dependencies
   4. Development Tools
   5. Hardware Requirements
   6. Software Requirements
   7. Assumptions and Constraints
   8. Future Enhancements
   9. Contact Information
6. **Introduction**
   1. **Purpose**

The purpose of this Software Requirements Specification (SRS)

Document is to provide a detailed description of the Library Management System (LMS). This document outlines the functional and non-functional requirements, user interactions, and system features necessary for the successful development and implementation of the LMS. It serves as a guide for the development team and a reference for stakeholders, ensuring that the system meets the needs and expectations of its users.

* 1. **Scope**

Manually updating the library system into an android based application so that the user can know the details of the books available and maximum limit on borrowing from their computer and also through their phones.

The ILM System provides information's like details of the books, insertion of new books, deletion of lost books, limitation on issuing books, fine on keeping a book more than one month / one week from the issued date.

Also user can provide feedback for adding some new books to the library.

* 1. **Definitions, Acronyms and Abbreviations**
* 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

1. **Overall Description**
   1. **Product Perspective**

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

* 1. **Software Requirement** 
     1. Front end:
* Android developer tool
* Advance java
* HTML
* CSS
  + 1. Back end:
* MySQL
* PHP(Hypertext Preprocessor)
  1. Product Functions

1. **User Management**

* Support for multiple user roles (e.g., librarian, member, administrator).
* User registration and login functionality.
* Profile management, including updating personal details and resetting passwords.

1. **Book Management**

* Add, update, or remove books from the catalog.
* Maintain book details, such as title, author, genre, ISBN, and publisher.
* Track book availability status (available, borrowed, reserved, or damaged).

1. **Search and Catalog Browsing**

* Advanced search functionality to locate books by title, author, genre, or keywords.
* Browse the catalog by categories or genres.
* Sort and filter search results based on criteria like popularity or availability.

1. **Borrowing and Returning**

* Issue books to registered members and set return due dates.
* Allow members to return books and update the system inventory accordingly.
* Calculate and display overdue fines (if applicable).

1. **Reservation System**

* Allow users to reserve books that are currently checked out.
* Notify users when reserved books become available.
* Set an expiration period for reservations.

1. **Fine Management**

* Track overdue fines based on the library's policies.
* Provide functionality for users to view and pay their fines online.

1. **Notifications and Alerts**

* Send email or SMS notifications for due dates, overdue books, and reservation updates.
* Provide alerts for system maintenance or policy updates.

1. **Reporting and Analytics**

* Generate reports on borrowing trends, popular books, and member activity.
* Provide insights for librarians to manage inventory and policies effectively.

1. **System Administration**

* Manage user roles and permissions.
* Configure library policies, such as borrowing limits and fine rates.
* Backup and restore system data.

1. **Integration and Extensibility**

* Support integration with external systems like payment gateways or online databases.
* Provide APIs for extending system functionality.
  1. **User Characteristics**

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:

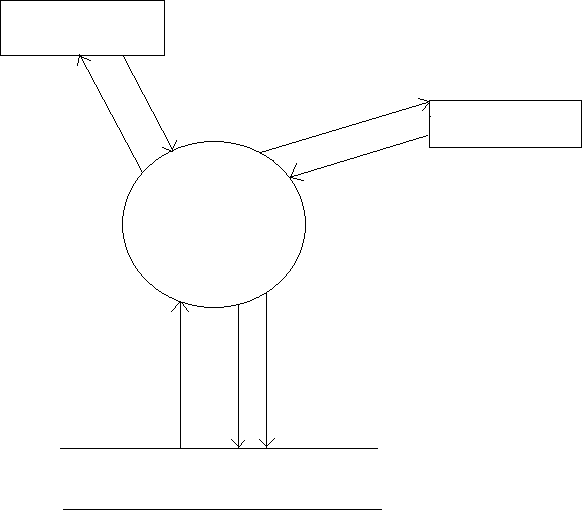
The following are the sub module in the administration module :

* + - Register user
    - Entry book details
    - Book issue

**2.5 Constraints**

Any update regarding the book from the library is to be recorded to have update & correct values, and any fine on a member should be notified as soon as possible and should be correctly calculated.

**2.6 Flow Diagram**



Library

Register/login

Update details of books

Register/login

Users

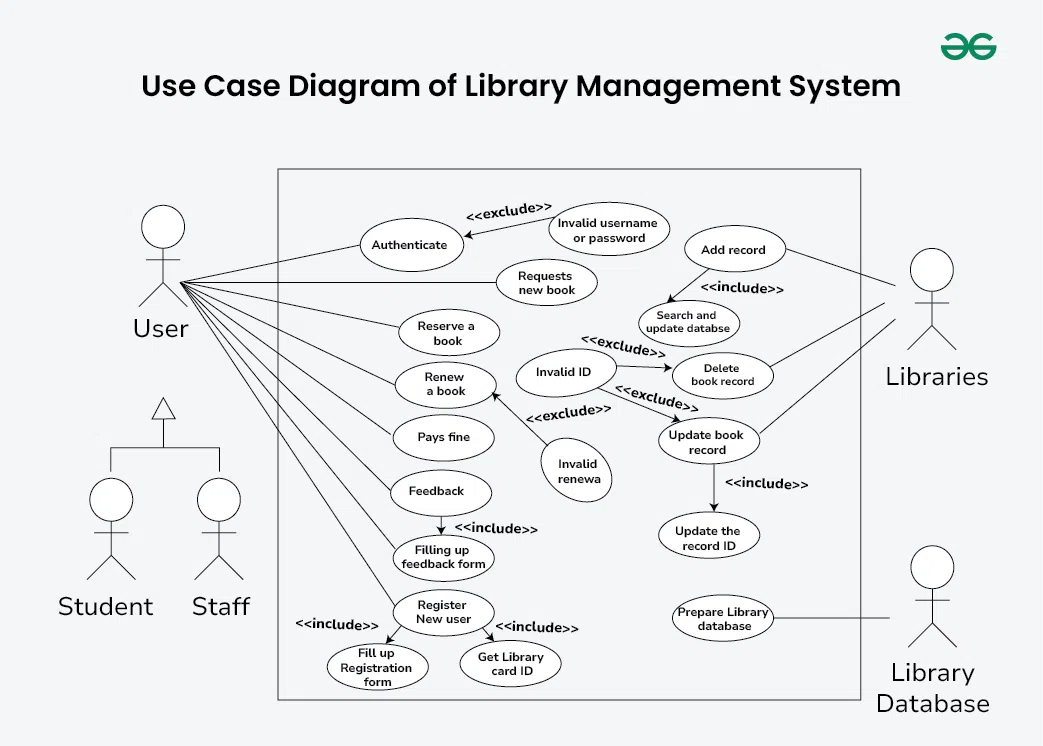
Issue, search, return, reserve books

Library Management

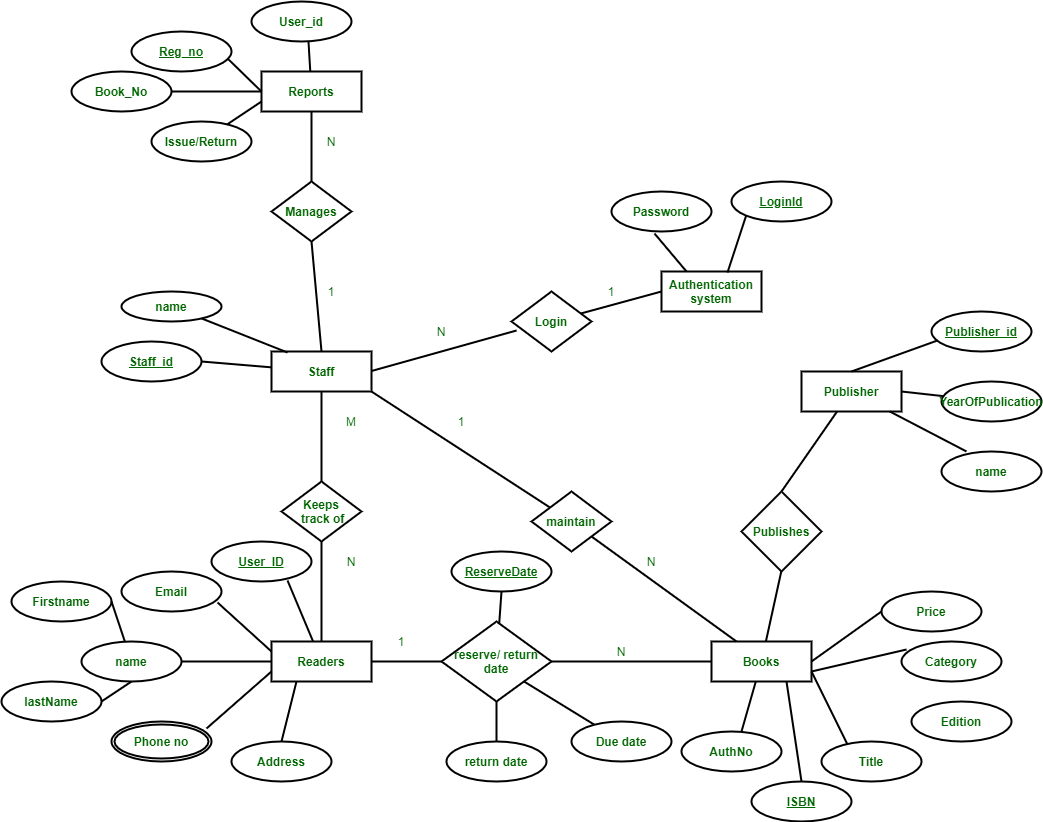
System

LIBRARY DATABASE

* 1. **Use Case diagram**



* 1. **ER diagram**



* 1. **Functional Requirements**

Functional requirements for a library management system (LMS) defines what the system can do. They include the ability to search for books, check out and return books, and store information about patrons.

Functional requirements

* **Search**: Search for books by title, author, or ISBN
* **Check out and return**: Check out and return books
* **List books**: Display a list of all books in the library
* **Store and retrieve patron information**: Store and retrieve information about library patrons, including their name and ID number
* **Maintain database**: Maintain the database of new books and the books that are borrowed by members along with their due dates
  1. **Non-Function Requirements**
* **Security**: Set access rights for security
* **Performance**: Maintain quality and usability
* **Reliability**: Consider hardware and software constraints
* **Backups**: Take database backups for safety

**4.System Features**

This section shows the features, priorities, requirements, and a few other things.

**4.1 System Login**

* Description and Priority

Here, the user's entered username and password have been authenticated.

Priority level: Very high

* Stimulus/Response Sequences

1. The user runs the system.
2. The system displays the login page.
3. The user inputs their username and password and clicks on the “login” button.
4. The system authenticates the user’s validity.

* Functional Requirements

1. • The system should only allow users with valid IDs and passwords to enter the system.
2. • The system should perform an authorization process that decides what level of access

each user can access.

1. • The user should logout after they finish using the system.

**4.2 Add a new book**

* Description and Priority

Here, the system admin can add new books to the system.

Priority level: Very high

* Stimulus/Response Sequences

1. The admin logs into his account.
2. The system displays the admin page.
3. The administrator selects the “Upload new books” button.
4. The system previews a list to fill in some details about the new book.
5. The admin fills in the details and uploads the book into the system.
6. The system displays the “Success” message or the “Error” message.

* Functional Requirements

1. • The system should allow the admin to use administration options.
2. • The system should allow the admin to upload new product details.

**4.3 Register a new user**

* Description and Priority

Here, a new user can register on the system.

Priority level: Very high

* Stimulus/Response Sequences

1. The new user should enter into the system.
2. The system displays the “Register” button.
3. The "Register" button should be selected by the user.
4. The system displays the relevant details for you to fill in.
5. The user should fill in the details asked and click on the "Register" button.
6. The system should save the user and should provide a user name.

* Functional Requirements

1. • The system should provide a “Username” and “Password” to the user.
2. • The system should open a new database for the user and save the user’s data in it.

**4.4 Issue a book**

* Description and Priority

Here, a user can borrow a book from the system.

Priority level: Very high

* Stimulus/Response Sequences

1. The user should sign into the system.
2. The system shows its “home page.”
3. The user can select to “borrow books”.
4. The system displays a list of books.
5. The user should select a book.
6. The system should show relevant details and forms to fill in.
7. The user should fill out the application or the form given.
8. The system should issue the book.

**5.Appendices**

The appendices section includes additional information and references that support the understanding and implementation of the Library Management System (LMS).

**5.1 Glossary of Terms**

User ID: A unique identifier assigned to each registered user in the system.

Book ID: A unique identifier assigned to each book in the library database.

Overdue Fine: A penalty imposed on users for keeping borrowed books beyond the

due date.

**5.2 References**

Java Development Kit (JDK) documentation for application development.

MySQL official documentation for database design and queries.

Android Developer's Guide for creating mobile applications.

**5.3 Dependencies**

The LMS depends on Android Studio IDE for front-end development.

Backend integration with MySQL database and PHP for server-side operations.

**5.4 Development Tools**

Front-End: Android Developer Tools, Advanced Java, HTML, CSS.

Back-End: MySQL, PHP for database and server operations.

**5.5 Hardware Requirements**

Minimum RAM: 4 GB.

Disk Space: 500 MB for application installation and database.

Processor: Dual-core or higher.

**5.6 Software Requirements**

Operating System: Windows, macOS, or Linux for development.

Android Version: Compatible with Android 8.0 (Oreo) or higher for end-users.

**5.7 Assumptions and Constraints**

Users should have basic knowledge of using mobile applications.

The system should handle up to 1,000 users concurrently.

**5.8 Future Enhancements**

Integration with external payment gateways for fine payments.

Extending the application to support multiple languages.

**5.9 Contact Information**

For support or queries regarding the Library Management System, contact:

Email: support@librarymanagement.com

Phone: +91 800-LIB-MGMT

This section ensures that all additional references, dependencies, and key information are readily available for developers and stakeholders.

1. • The system should update “User ID” and “Book ID” in the database