**Team: SANKALPA**

**Project Title: LIBRARY MANAGEMENT SYSTEM**

**Project Start Date: 18TH SEPTEMBER,2023**

**By:**

* Aastha
* Adity
* Devansh
* Divyanshi Bhatia
* Ravi
* Sakshi

**Requirement Analysis**

## **1.Introduction**

* LMSs are software programs that assist libraries in managing their operations and collection management. We have modules for circulation, purchases, serials administration, reporting, and cataloguing.
* Automating library activities including cataloguing, circulation, acquisitions, membership management, and reporting is the major goal of a library management system. It strives to provide superior experiences for both library personnel and users while enhancing the accessibility, tracking, and consumption of library resources.

**Challenges:**

* **Inefficient Book Management:** The current manual cataloging and inventory management make it difficult to track books effectively, resulting in frequent misplacements and inconsistencies in the catalogue.
* **Time-Consuming Transactions:** The manual borrowing and returning process is time-consuming and inconvenient for both students and library staff, especially during busy academic periods.
* **Limited Accessibility:** The physical cataloging system restricts access to the library's resources, especially for remote or off-campus users.

**Solution with LMS:**

The university implements a robust LMS, allowing students and faculty to access the library's resources seamlessly. The LMS addresses the challenges by offering the following benefits:

* **Efficient Book Management:** Books are digitally catalogued and categorized based on various criteria (title, author, subject, etc.), enabling quick and accurate searches.
* **Streamlined Transactions:** Borrowing and returning books are simplified through an online portal, reducing the time and effort required for these transactions.
* **Enhanced Accessibility:** The LMS is accessible remotely, enabling users to search for and reserve books from their homes or other locations. They can also receive notifications and reminders for due dates.

### **1.1. Purpose**

This document's goal is to provide an overview of the standards and requirements for creating a library management system (LMS) for university level.

### **1.2. Scope**

This system will handle all aspects of running a library, including user management, circulation, cataloguing, and reporting.

**1.3. Users**

* Librarian
* Library Staff
* Library Member
* **Target Audience:** University / College Students.

### **1.4. Success metrics**

Creating a success matrix for a Library Management System (LMS) involves defining specific success criteria and metrics to assess the system's performance and impact. Below is an example of a success matrix for an LMS, along with some illustrative metrics and criteria:

**Success Criteria:** User Satisfaction

**Metric 1:** User Satisfaction Score

**Metric Description:** Conduct periodic surveys among library users to rate their satisfaction with the LMS on a scale of 1 to 5, with 5 indicating high satisfaction.

**Target:** Maintain an average satisfaction score of 4 or higher.

**Success Criteria:** Resource Availability

**Metric 2:** Book Availability Rate

**Metric Description:** Calculate the percentage of requested books that are available for checkout at any given time.

**Target:** Maintain a book availability rate of 95% or higher.

**Success Criteria:** User Engagement

**Metric 3:** User Engagement

**Metric Description**: Measure the average number of logins, searches, and book checkouts .

**Target:** Increase user engagement by 10% over the previous report.

**Success Criteria:** Operational Efficiency

**Metric 4:** Cataloging Efficiency

**Metric Description:** Track the time it takes to catalog and process new books into the system.

**Target:** Reduce cataloging time by 15% through process optimization.

**Success Criteria:** Digital Resource Utilization

**Metric 5:** Digital Resource Usage

**Metric Description:** Analyze the utilization of digital resources (e-books, databases) and track the most frequently accessed content.

**Target:** Increase digital resource usage by 20% over the previous year.

**Success Criteria:** User Registration and Retention

**Metric 6:** User Registration Rate

**Metric Description:** Track the number of new user registrations annually.

**Target:** Achieve a 10% increase in new user registrations.

**Metric 7:** User Retention Rate

**Metric Description:** Calculate the percentage of registered users who continue to use the LMS after the first year.

**Target:** Maintain a user retention rate of 75% or higher.

By defining these success criteria and associated metrics, we will systematically evaluate the performance of your LMS and work towards continuous improvement, ensuring that the library meets its goals and provides an exceptional service to its users and community. Adjust the metrics and targets based on your library's specific objectives and priorities.

### **1.5. Precedent of Application:**

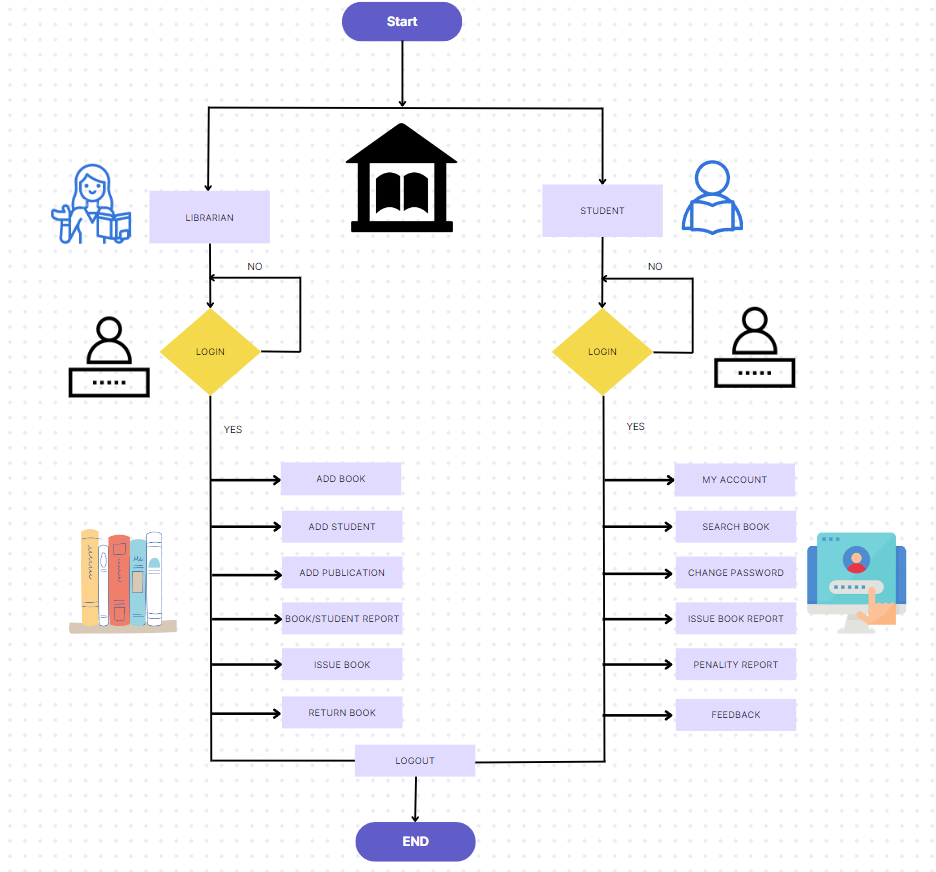
LMS applications encompass a wide range of functionalities, and there are several software applications available in this domain. Some of those are:

* **Atriuum ILS:** Atriuum is a cloud-based Integrated Library System (ILS) designed to work with any library and is built on over 30 years of library technology and library automation experience. Atriuum provides advanced features such as configurable user interfaces, customizable reports, mobile searching, and eBook resource sharing. For any further information:<https://booksys.com/atriuum>
* **Follett Destiny:** Follett Destiny is a library management solution designed for K-12 schools. It includes features for cataloging, circulation, textbook management, and resource discovery. For any further information: <https://www.follettlearning.com/education-technology/library-management-system>
* **Virtua:** Virtua is an ILS by VTLS, primarily used by academic and research libraries. It offers modules for cataloging, circulation, and resource management. For any further information: <https://librarytechnology.org/product/virtua/>
* **Alma:** Ex Libris Alma is a cloud based LMS designed for academic and research libraries. It offers comprehensive functionality for resource management, fulfilment, analytics, and more. Alma is known for its integration capabilities. For any further information: <https://exlibrisgroup.com/products/alma-library-services-platform/>
* **LibSys:** LibSys is a popular library management software used in various academic and research libraries. It provides modules for cataloging, acquisitions, circulation, serials management. For any further information: <https://www.libsys.co.in/products/rfid-deployments/library-automation>

**Functional Specifications**

## **2. System overview**

The Library Management System will be developed utilizing a client-server architecture, where the client (web application) communicates with the server and the server hosts the database and business logic.



### **2.1. Scenario**

Imagine a librarian named Mr. Kartik who works in the Central Library. Aahana, a student of Finance studying in Pune University is a regular visitor of the Central Library. Everything here worked in physical mode but after digitalization, the library authority decided to integrate the online library management system with their library to increase efficiency. Let us now investigate the different challenges that they may face while dealing with the Library Management System:

* **Adding a book:** Some new books have arrived in the stock and Mr. Kartik needs to add them in the LMS portal so that it may be visible to the users. For adding a new book, Mr. Kartik will make a new entry for that book along with all the details like author’s name, publication, ISBN number etc.
* **Renting a book:** Aahana being a keen reader, wants to rent a book from the library. She searches the catalogue and finds the book she is interested in. She applies for renting that book by clicking on “Issue Book”. Mr. Kartik gets notified about the same and accepts the request. He then makes an entry of the same in the LMS.
* **Returning the book:**  Aahana return books to the library, where Mr. Kartik manages the return process using the library management system (LMS). The LMS swiftly checks for due dates and updates book status to "returned," allowing for immediate reshelving and user notifications for any fines. The returned book is then made available for borrowing, and the transaction is recorded for statistical purposes.
* **Removing the user:** When it's necessary to remove a user from the library system, Mr. Kartik, the librarian, uses the library management system (LMS). He deactivates the user's account in the LMS, addresses outstanding fines and materials, and sends a final notification. User data is retained for records and privacy compliance

**2.2. Open issues**

* **Mobile Application Development:** Determine whether the mobile application should be developed for iOS (Apple) and/or Android platforms.
* **Barcode Scanner Compatibility:** Support various barcode types commonly used in library materials, such as ISBN barcodes.
* **Integration Issues:** Interlibrary loan requests are not consistently syncing with partner libraries.
* **Multi-Lingual Support:** Implement language localization for the LMS user interface, allowing users to select their preferred language.
* **Programming Language Selection:** Determine if the programming language can easily integrate with databases and libraries that the LMS relies on.

### **2.3. Side notes**

After researching the existing library management system, we found that the public had offered suggestions for improving the system which includes the following: -

* **User Training and Support**: Develop comprehensive training materials, including user manuals, guides, video tutorials, and FAQs, to assist users in learning how to use the LMS.
* **Security and Privacy:** Ensure that data transmission between users and the LMS, as well as data at rest, is encrypted to protect against unauthorized access.
* **Disaster Recovery and Backup:** Implement a robust data backup strategy that includes regular, automated backups of the LMS database.

## **3. Functional Requirement**

## **3.1. Features**

* **Addition/Deletion of Users**: After a user has been provided with the membership to the library, he will be automatically added to this system and provided with necessary credentials. On the other hand, if a user does not want to renew his/her membership, they will be removed from this.
* **Addition/Deletion of Books**: Our system will help the users by providing the information of newly added of newly arrived books. In case of the books which are not being read for a long time or which are no longer in demand, the system will take care that it is removed from the list.
* **Reminders/Notifications**: This system will also provide a feature of frequent reminders for returning the book on time.
* **Penalty System**: If a reader fails to return the book on time, the system will start adding fine in that user's account which will keep updating automatically.
* **ISBN support:** ISBN support will help get the details of the book such as author, publisher, publication date, number of pages, etc.
* **Digital Membership**: while getting physical membership not only seems a tedious task but is also time consuming. This feature of digital membership will help the users by providing membership within minimal time, hence saving their precious time and energy.
* **Recommendation System**: When a reader reads a book of a particular author or a particular genre, our Library management system will recommend books to that reader based on his personal choice Instead of going with the choice of the mass.
* **Reviews:** After reading a book, the reader will be allowed to provide review for the same.
* **Ratings:** Besides giving an overview of the book, our Library management system will help users by providing the rating for each book which will help the new readers
* **Pre-Booking:** If a reader wants to read some book which is currently unavailable, the user can pre- book that book. As soon as the book will be returned by the previous reader, this reader will get the opportunity to read it.



**External Interface Specification**

## **4. Interaction of software with other components**

**4.1How will the software interact with the users?**

* **Email and Notifications**: The library system will be responsible to send regular emails regarding the deadline to return a book, about the fine that the user is supposed to pay, about the newly arrived collection of books, etc.
* **Authentication Services**: The system will work with the database to tally the credentials and hence provide security to users while preventing invalid logins.
* **Customer Suppor**t: The system will assist the users by providing them support in case of any confusion or help required.

**4.2 How will the software interact with other software?**

* **Email**: Our software i.e., library management system will be working along with email to send notifications and remainders.
* **WhatsApp**: Since there may be users who do not check their email regularly, our system will also deal with sending notifications and reminders through WhatsApp.
* **API**: APIs will be used to integrate the dashboard of the library management system with the database for performing actions like searching the catalogue, checking out books, or managing user accounts (e.g., RESTful APIs)
* **Web Services**: The system will be a web-based interface which can be accessed through web browsers. Therefore, the management system will be interacting with the backend services to provide the required features like that of signing-in or searching the catalogue.

**Technical Specification**

**5. Quality Attribute**

### **5.1. Performance Expectation**

The main goal of creating a library management system is to make libraries work better and easier. It helps people find and borrow books more easily. It also makes it simpler for librarians to organize books and take care of library tasks. Ultimately, it's about making the library experience great for everyone: -

* **Response Time**: The system should respond to user requests within a reasonable timeframe, typically within a few hundred milliseconds, to ensure a smooth and responsive user experience.
* **Database Performance**: Database queries should be optimized for efficiency, ensuring fast retrieval and updating of data to support the application's functionality.
* **Search Performance**: Implement efficient and fast search capabilities within the library's collection.

### 