

27/8/23

Library Management System

Problem Statement -

Design and implement a Library Management System that automates book cataloging, user registration, issue/return tracking, and fine calculation, providing an efficient, secure and user friendly platform for students, librarians and administrators.

SRS Document -

1. Introduction

1.1 Purpose

The purpose of this document is to specify the functional and non functional requirements of a Library Management System that automates book cataloging, user registration, borrowing and returning processes.

1.2 Scope

The Library Management System will allow students and staff to search for books, borrow and return them efficiently. Librarians can add, remove or update book records and manage user accounts. The system will be accessible via a web interface.

1.3 Overview

The LMS will be a web-based application accessible via desktop and mobile browsers. It will include modules for:

- User registration and authentication

- Book catalogue management
- Book issue, return and renewal
- Fine calculation and payment tracking
- search and reporting features.

2. General Description

The Library Management System automates catalogue, user management and book transactions, providing secure access, fine calculation and reporting on standard hardware with internet support.

3. Functional Requirements

FR. 1 Catalog Management: Add, update, ~~delete~~ delete and search books in the database. Maintain records of authors, editions and publishers.

FR. 2 User Management: Register and update user information, track issued, returned and overdue books.

FR. 3 Circulation Management: Issue and return books with fine calculation for overdue items.

FR. 4 Reporting and Billing: Generate daily/weekly/monthly reports on circulation, fine collection summaries and transaction logs.

4. Interface Requirements

1. User Requirements: Intuitive, user-friendly interface accessible via browsers and mobile devices.

2. Hardware Interface: Compatible with standard hardware (computers, barcode scanners, printers).

3. Software Interface: Integration with online payment gateways for fine collection.

5. Performance Requirements

- The system shall respond to user actions within 2 seconds.
- Support at least 500 concurrent users during peak hours.
- Ensure consistent and accurate data access across all modules.

6. Design Constraints

- The system shall run on standard library hardware with reliable internet access.
- Utilize relational database systems, with secure frameworks.

7. Non-Functional Requirements

7.1 Security:

Implement role-based authentication and encryption to protect sensitive data.

7.2 Reliability:

Ensure high availability with fault tolerance.

7.3 Maintainability:

Use modular code to facilitate updates and feature enhancements.

7.4 Usability: Provide a clear, intuitive interface for staff and users.

7.5 Data Integrity: Ensure accurate, consistent and retrievable records.

8. Preliminary Schedule and Budget

8.1 Schedule

Phase	Timeline
Requirements Analysis	2 weeks
UI/UX	3 weeks
Development	10 weeks
Testing	4 weeks

8.2 Budget

Estimated → \$40,000