

26/02/21

③ Library Management Statement.

Problem Statement:

Traditional library management is time consuming and prone to errors, as it relies on manual record-keeping for books, members and transactions. This often leads to misplaced books, difficulty in tracking issued/returned items, and delays in notifying members about due dates. The proposed library management statement system aims to automate these tasks, ensuring efficient book management, faster transactions and timely communication with members.

SRS:

1. Introduction

1.1 Purpose of the Document

This document defines the requirements for the LMS. It is intended to serve as a reference for developers, testers, librarians and stakeholders to ensure that the system supports smooth borrowing / return processes.

1.2 Scope of the document

The LMS is a web-based application designed to automate day-to-day library tasks. It manages book catalogs, user memberships, borrowing / returns operations, fines and

reporting.

1.3 Overview

The CMS will provide a centralised platform for managing books, users and transactions. It will allow students to search and borrow books online, librarians to handle check-ins & check-outs efficiently.

2. General Description

2.1 Product Functions

- Maintain a digital catalog of books and resources
- Allow students to search, reserve and borrow books.
- Track issued, returned and overdue books.
- Manage user membership and accounts.
- Calculate and manage late fines automatically.
- Generate reports for library usage.

2.3 User characteristics

• Students / Members: Borrow / Return books, check availability.

• Librarians: Manage catalog, issue / return books.

• Admins: Configure system settings.

3. Functional Requirements

- FR 3.1 Catalog management: The system shall register new members with personal details.
- FR 3.2 The system shall allow librarians to add, update, or delete book records.
- FR 3.3 Students shall be able to search for books by title, author, or category.
- FR 3.4 The system shall send reminders for due dates and overdue books.
- FR 3.5 The system shall generate daily/weekly/monthly reports.

4. Interface Requirements

4.1 User interfaces

A clean web dashboard for librarians to manage books and transactions.

4.2 Hardware Interfaces

Supports barcode / RFID scanners for fast issue / return.

5. Performance Requirements

- Must handle up to 200 concurrent users.
- Search results must load within 2 secs.
- System uptime must be at least 99.5% / month.

Automatic backups must run daily.

6. Design constraints

- Developed using open-source tech.
- Must support responsive design for mobile.

- Must comply with privacy rules

7. Non functional Attributes

7.1 Security: Data encrypted during transmission

7.2 Usability: Intuitive UI for both students and librarians.

7.3 Maintainability: Modular architecture for easy updates.

7.4 Reliability: Handles unexpected shutdowns without data loss.

8. Preliminary schedule and Budget.

8.1 schedule

Phase 1 of the project Timeline:

Requirements Analysis: 2 weeks

UI/UX design: 2 weeks

Development: 8 weeks

Testing & QA: 1 week.

8.2 Budget

Total estimated cost: \$28,000.