• 2025 •

SRS DOCUMENT

Library Management System

Proposal by : **Ravindu Perera**

Software Requirements Specification (SRS)

Project Name: Library Management System
Prepared By: Ravindu Perera
Date:2025.01.24

---

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed and professional specification for the Library Management System (LMS). The LMS is designed to revolutionize library operations, offering advanced features for managing books, users, transactions, and fines, while ensuring a seamless user experience. By transitioning from manual to automated processes, the LMS reduces errors, increases efficiency, and enhances engagement for both library staff and patrons.

1.2 Scope

The LMS will serve as a standalone, feature-rich desktop application developed using JavaFX. Its scope includes:

- **Streamlined Operations:** Efficient book, user, and transaction management.
- **Automation:** Automated fine calculation and report generation.
- **Data Security:** Secure storage of records with user authentication.
- **Scalability:** Easily extendable to incorporate e-books or remote library services in the future.

1.3 Definitions, Acronyms, and Abbreviations

- **JavaFX:** Java framework for building modern, graphical user interfaces.
- **JDBC:** Java Database Connectivity for database interactions.
- **CRUD:** Create, Read, Update, Delete operations.
- **ER Diagram:** Entity-Relationship Diagram representing database schema.
- **POS:** Point of Sale system.

1.4 References

1. JavaFX Documentation: https://openjfx.io/openjfx-docs/
2. Hibernate Documentation: https://hibernate.org/orm/documentation/
3. MySQL Documentation: https://dev.mysql.com/doc/
4. Maven Repository: https://mvnrepository.com/

---

2. Overall Description

2.1 Product Perspective

The LMS is envisioned as a modern replacement for the manual processes that currently dominate library management. With its modular design and integration capabilities, the LMS provides a scalable platform for managing library resources efficiently.

2.2 Product Functions

The LMS will encompass:

1. **Book Management:**
    - Add, update, and remove books.
    - Advanced searching and filtering options.
2. **User Management:**
    - Register new users and maintain their profiles.
    - Search and manage user records effectively.
3. **Transaction Handling:**
    - Borrowing and returning books with accurate tracking.
    - Automated fine calculations for overdue items.
4. **Fine Management:**
    - Real-time fine updates and payment tracking.
5. **Reports and Analytics:**
    - Generate detailed reports with data visualization.
6. **User Authentication:**
    - Role-based access for administrators and staff.

2.3 User Classes and Characteristics

- **Library Staff:** Responsible for daily operations, managing records, and assisting patrons.
- **Administrators:** Manage system configurations and oversee operations.
- **Patrons:** Access library resources and manage their borrowing activities.

2.4 Operating Environment

- **Platform:** Cross-platform support (Windows/Linux/MacOS).
- **Languages:** Java, SQL.
- **Frameworks and Tools:** JavaFX, JFoenix, JasperReports, Maven.
- **Database:** MySQL/SQLite.

2.5 Design and Implementation Constraints

- Implementation must adhere to JavaFX guidelines.
- Responsive and modular UI design.
- Database interaction via JDBC.
- Compliance with layered architecture principles.

2.6 Assumptions and Dependencies

- The library staff will have basic familiarity with computer systems.
- Necessary software tools (JavaFX SDK, database) are pre-installed.
- Internet access for initial setup and updates.

---

3. Requirements

3.1 Functional Requirements

1. **Book Management:**
    - Add, edit, and delete books.
    - Search by multiple criteria (title, genre, author).
    - Bulk upload of books via CSV.
2. **User Management:**
    - Register users with unique IDs.
    - Maintain user history for insights.
    - Enable or disable user access.
3. **Transactions:**
    - Limit the number of active borrowings per user.
    - Support extensions for borrowing periods with conditions.
4. **Fine Management:**
    - Allow detailed view and breakdown of fines.
    - Send reminders for overdue payments.
5. **Reports and Dashboards:**
    - Visual charts for inventory and usage statistics.
    - Export reports to PDF or Excel.

3.2 Non-Functional Requirements

- **Performance:**
    - Optimized for low-latency operations.
    - Database queries must execute in under 2 seconds.
- **Scalability:**
    - Prepared for integration with e-library services.
- **Security:**
    - Use encrypted connections for database transactions.
- **Reliability:**
    - Maintain 99.9% uptime.

3.3 System Models

- **ER Diagram:** Illustrates the relationships among books, users, and transactions.
- **Use-Case Diagram:** Visual representation of user interactions.
- **Sequence Diagrams:** Detailed workflows for core processes.

---

4. Appendices

4.1 References

1. Password Encryption Techniques: https://www.javatpoint.com/how-to-encrypt-password-in-java
2. JasperReports Guide: https://community.jaspersoft.com/documentation

4.2 Acronyms

- **LMS:** Library Management System.
- **CRUD:** Create, Read, Update, Delete.
- **ER Diagram:** Entity-Relationship Diagram.

4.3 Contact Information

- **Name:** Ravindu Perera.
- **Email:** contact.ravinduperera@gmail.com
- **Phone:** 071 189 2658

---

5. Conclusion

The Library Management System (LMS) is a critical tool for modernizing library operations and addressing the inefficiencies of manual processes. By integrating advanced features such as automated fine calculations, real-time data tracking, and insightful reporting, the LMS ensures an efficient and user-friendly experience for library staff and patrons alike. With its scalable architecture and secure design, the LMS is well-positioned to adapt to future needs, such as e-library services or multi-branch integrations. This project represents a significant step toward digital transformation in library management, ensuring both operational excellence and enhanced user satisfaction.