Low-Level Design

1. Introduction:

The Library Management System is a web-based application that simplifies the management of books and students in a library. It provides functionalities such as student registration, book search, and book details viewing. The system is primarily administered by an admin who has privileges to add, delete, and update books, issue books to students, collect fines, and manage student records.

2. System Components:

The Library Management System is composed of the following components:

- Frontend: Utilizes JSP (JavaServer Pages) for creating the user interface and enabling user interaction.

- Backend: Implements the business logic and database operations using the Spring Boot framework.

- Persistence: Utilizes Spring Data JPA for accessing and manipulating data stored in the MySQL database.

3. User Roles:

The system supports two user roles:

- Student: Can register, login, search for books, view book details, and pay fines.

- Admin: Has complete control over the system, including the ability to add, delete, and update books, issue books to students, manage fines, and maintain student records.

4. Student Management:

4.1 Registration and Login:

- Students can register by providing necessary details such as their name, email, and password.

- Upon successful registration, a new student record is created in the database.

- Students can log in using their registered email and password.

4.2 Book Search and Viewing:

- Students can search for books based on keywords that match the book's category, title, or author.

- The system displays a list of books that match the search criteria, showing details such as the book ID, title, author, and category.

- Students can view the complete details of a specific book, including its availability and any fines associated with it.

5. Book Management:

5.1 Admin Authentication:

- Admin credentials are hardcoded into the application for authentication.

- Only the admin can access the book management functionality.

5.2 Book CRUD Operations:

- The admin can add a new book by providing details such as the title, author, category, and availability status.

- The admin can delete an existing book from the system.

- The admin can update the details of a book, including the title, author, category, and availability status.

5.3 Student Management:

- The admin has access to the list of registered students in the system.

- The admin can delete a student record if necessary, provided that the student has no pending fines.

5.4 Book Issuing:

- The admin can issue books to students, with a limit of three books per student.

- The issuing date is stored in the database for each book issued.

- If a student fails to return a book within 15 days, a fine is calculated based on the number of days overdue.

- The fine amount is 10/9 rupees per day for each overdue book.

5.5 Fine Collection:

- The admin can collect fines from students who have paid the outstanding amount.

- By clicking a button next to the student's fine details, the admin can update the student's fine amount to zero.

6. Technology Stack:

The Library Management System utilizes the following technology stack:

- Frontend: JSP (JavaServer Pages)

- Backend: Spring Boot

- Persistence: Spring Data JPA

- Database: MySQL

7. Conclusion:

The Low-Level Design of the Library Management System encompasses components for student management, book management, and fine collection. The system supports user registration, login, book search, book viewing, and CRUD operations for books and student records. Admin credentials are used for system administration, including book issuing, fine management, and student record maintenance. The technology stack consists of JSP for the frontend, Spring Boot for the backend, and Spring Data JPA for data persistence using MySQL.