**Title: Low-Level Design - Library Management System**

1. Introduction The Library Management System is a web-based application designed to facilitate the management of books and students in a library. It allows students to register, search for books, and view book details. The system is primarily managed by an admin who can add, delete, and update books, issue books to students, collect fines, and manage student records.
2. System Components The Library Management System consists of the following components:

* Frontend: JSP (JavaServer Pages) for user interface and interaction.
* Backend: Spring Boot framework for handling business logic and database operations.
* Persistence: Data JPA for accessing the MySQL database.

1. User Roles The system supports two user roles:

* Student: Can register, login, search for books, view book details, and pay fines.
* Admin: Has full control over the system, including adding, deleting, and updating books, issuing books to students, managing fines, and managing student records.

1. Student Management 4.1 Registration and Login

* Students can register by providing necessary details such as 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 matching the search criteria, showing the book ID, title, author, and category.
* Students can view the details of a specific book, including availability and any fines associated with the book.

1. 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

* Admin can add a new book by providing details such as title, author, category, and availability status.
* Admin can delete an existing book from the system.
* Admin can update the details of a book, including title, author, category, and availability status.

5.3 Student Management

* Admin has access to the list of registered students in the system.
* Admin can delete a student record if needed, provided the student has no pending fines.

5.4 Book Issuing

* Admin can issue books to students, limited to a maximum 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

* Admin can collect fines from students who have paid the outstanding amount.
* The admin updates the student's fine amount to zero by clicking a button adjacent to the student's fine details.

1. Technology Stack

* Frontend: JSP (JavaServer Pages)
* Backend: Spring Boot
* Persistence: Data JPA
* Database: MySQL

1. Conclusion The Low-Level Design of the Library Management System includes 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 issuing books, managing fines, and student records. The technology stack includes JSP for the frontend, Spring Boot for the backend, and Data JPA for data persistence with MySQL.