Library Management System (Detailed Workflow)

System Overview

The Library Management System (LMS) is a feature-rich platform designed to efficiently manage library operations, catering to both administrators and users. Built using **MySQL**, **Node.js**, **React (Next.js)** for the web application, and **React Native** for the mobile application, the system provides an intuitive experience for managing books, users, and transactions, including notifications and fine management.

Key Features and Functionalities

1. User Module (Web and Mobile Applications)

- **Registration:**
  - Users fill out a detailed registration form, ensuring required fields and validations (e.g., matching passwords).
  - Account activation requires admin approval.
  - Notifications (via email or SMS) are sent for account activation or rejection.
- **Login:**
  - Secure login using email and password with encrypted data storage.
  - "Forgot Password" option to send a password reset link to registered emails.
- **Profile Management:**
  - Users can view and update personal details such as email, mobile number, and address.
  - Real-time account status (active, blocked, or pending activation).

---

2. Member Mobile Application (React Native)

- **Search and Browse Books:**
  - Advanced search options with filters (e.g., title, author, category).
  - Real-time book availability, including the number of available copies.
  - Detailed book information such as title, author, and description.
- **Place Orders:**
  - Members can order up to 3 books at a time.
  - Automated calculation of the return date (10 days from the issue date).
  - Order confirmation notifications via email and mobile push.
- **My Orders Section:**
  - Displays all borrowed books with details:

        - Title and author
        - Borrowed date and return date
        - Overdue fine, if applicable
    - Alerts for overdue books and reminders for returns.
- **Notifications:**
    - Real-time push notifications for:
        - Book order confirmations
        - Return reminders (3 days before the due date)
        - Overdue fine alerts and account blocking
        - Account activation or rejection updates
- **Account and Fine Status:**
    - Members can view their fine details, overdue books, and account status.
    - Accounts are automatically blocked if fines exceed the threshold (e.g., ₹500).

---

3. Librarian Dashboard (Web Application)

- **Key Metrics:**
    - Total Books
    - Total Members
    - Books Issued
    - Overdue Fines Summary
    - Notifications for pending user approvals and overdue returns
- **Admin Features:**
    - **Manage Users:**
        - Approve or reject pending registrations.
        - View and update user details.
    - **Manage Books:**
        - Add, update, delete, and categorize books.
        - Track available copies and issued books.
    - **Order Management:**
        - View pending and completed orders.
        - Process book returns and calculate fines automatically.
    - **Notifications:**
        - Send automated reminders and alerts for overdue books or unpaid fines.
        - Bulk notifications for books nearing their return dates.

---

4. Notifications

The system integrates **email** and **mobile notifications** to streamline communication between members and administrators. Key notifications include:

- **Account Creation Approval:**
  - Notification sent when an admin approves a new user account.
- **Book Order Confirmation:**
  - Confirmation email and push notification sent when a user places a book order.
- **Return Reminders:**
  - Alerts sent 3 days before the book's due date to ensure timely returns.
- **Overdue Fine Reminders:**
  - Notifications for overdue books, including the fine amount and payment instructions.
- **Account Blocking Alerts:**
  - Alerts sent when fines exceed the threshold (e.g., ₹500), notifying users that their account is blocked until the fine is paid.

---

5. Fine Calculation

The system automates fine calculation for overdue books and manages account statuses effectively.

- **Rules:**
  - Borrowing period: **10 days**.
  - Fine for overdue books: **₹50 per day**.
  - Accounts are **blocked automatically** if the fine exceeds the threshold (e.g., ₹500).
- **Process:**
  - Users are notified daily about overdue books and the accumulating fine.
  - Once the fine threshold is crossed, users receive an account blocking alert and payment instructions.
  - After payment, the admin unblocks the account and updates the user's status.
- **Examples:**
  - A user borrows a book on **1st Jan**. The return date is **10th Jan**.
    - If returned on **15th Jan**, the fine is:
      ₹50/day × 5 days = **₹250**.
    - If the user delays further and the fine reaches **₹500**, the system blocks their account and sends a notification.

Technology Stack

- **Backend:** MySQL, Node.js
- **Frontend Web Application:** React (Next.js)
- **Mobile Application:** React Native
- **Notifications:** Email (via SMTP) and mobile push notifications (via Firebase or similar).

Workflow

1. User Registration and Login

1. Users register through the web or mobile app.
2. Admin reviews and approves the registration.
3. Users are notified about their account activation or rejection.

2. Book Browsing and Orders

1. Members search and browse books via the mobile app.
2. The system displays real-time availability.
3. Members place an order, and the system calculates the return date.

3. Book Returns and Fine Management

1. Users return books to the library.
2. Admin processes the return and the system calculates fines for overdue returns.
3. Notifications are sent to users for overdue books and fines.

4. Notifications

1. System sends automated notifications for:
    - Account activation/rejection
    - Book order confirmation
    - Return reminders (3 days before due)
    - Overdue fines and account blocking

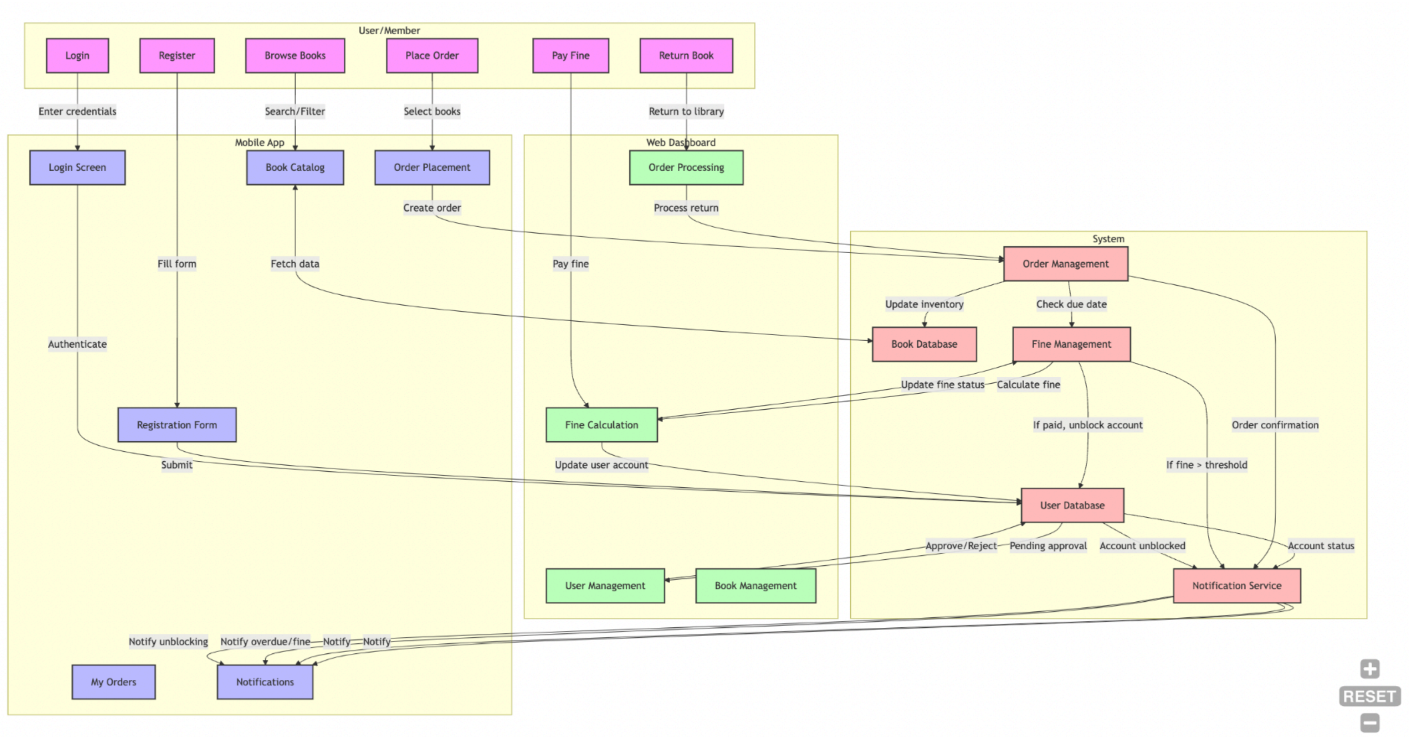
Developer Notes

- Ensure **responsive design** for the mobile app and web dashboard.
- Use **modular code** for scalability and maintenance.
- Implement **robust validation** for inputs at both frontend and backend levels.
- Conduct **extensive testing** for performance, usability, and scalability.
- Integrate **secure APIs** for seamless communication between the web and mobile applications.

This workflow ensures an intuitive user experience while maintaining the operational efficiency of library management.

Diagram Link

https://shorturl.at/6YDBE


User/Member
Login
Register
Browse Books
Place Order
Pay Fine
Return Book
Enter credentials
Search/Filter
Select books
Return to library
Mobile App
Login Screen
Book Catalog
Order Placement
Web Dashboard
Order Processing
Create order
Process return
System
Order Management
Fill form
Fetch data
Pay fine
Update inventory
Check due date
Book Database
Fine Management
Authenticate
Update fine status
Calculate fine
Registration Form
Fine Calculation
If paid, unblock account
Order confirmation
Submit
Update user account
If fine > threshold
User Database
Approve/Reject
Pending approval
Account unblocked
Account status
User Management
Book Management
Notification Service
Notify unblocking
Notify overdue/fine
Notify
Notify
My Orders
Notifications
RESET