**REPORT PROJECT SERVER SIDE INTERNET PROGRAMMING**

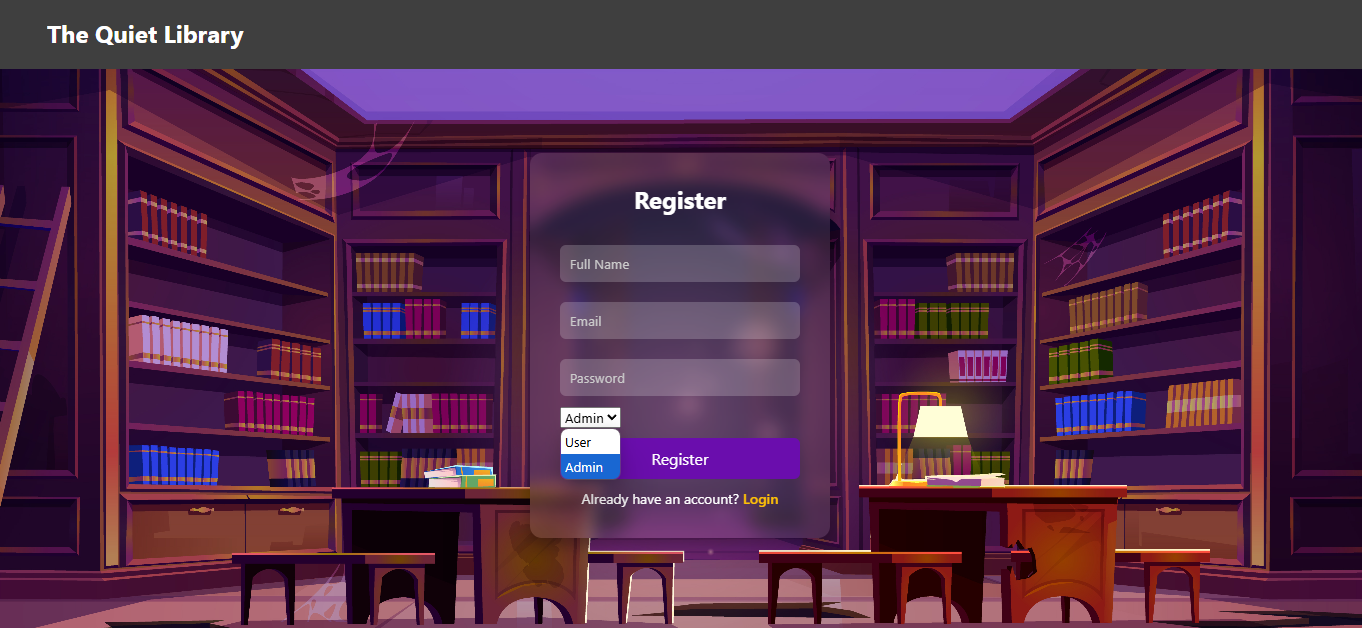


**NABILA LATASYA (001202400042)**

**CLASS INFORMATION TECHNOLOGY 4**

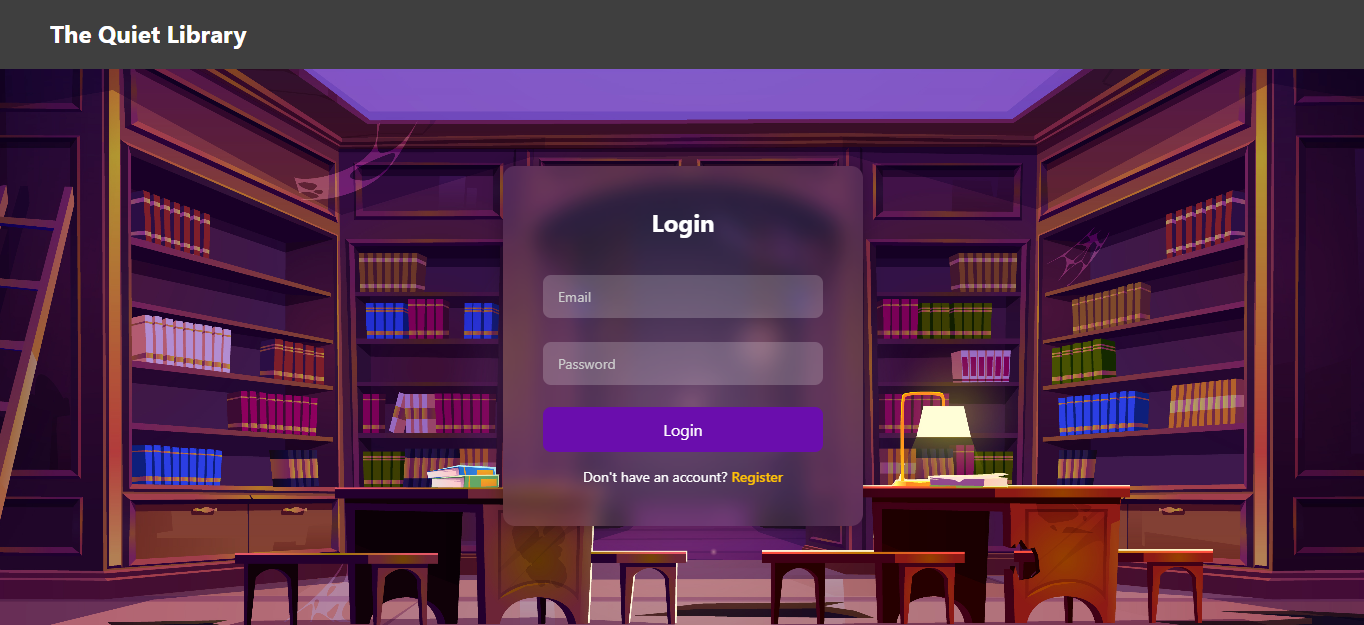
A.1 OVERVIEW

This form allows the admin to create new accounts within the system. It includes fields for username, password, email, and a role selector (Admin/User). The **role selection** is only visible when accessed by an **admin**, allowing them to assign the appropriate access level. If accessed for user self-registration, the role option is hidden and automatically set to **User**, ensuring role security.



**Picture A.1**

This page is used as the entry point for both admins and users. The form consists of **email** and **password** fields, which are required to authenticate the user. Once the login is successful, the system automatically identifies the user's role based on their account data. Admins will be redirected to the admin dashboard with access to management features, while regular users will be taken to the user interface. No role selection is needed, ensuring a clean and simple login experience.

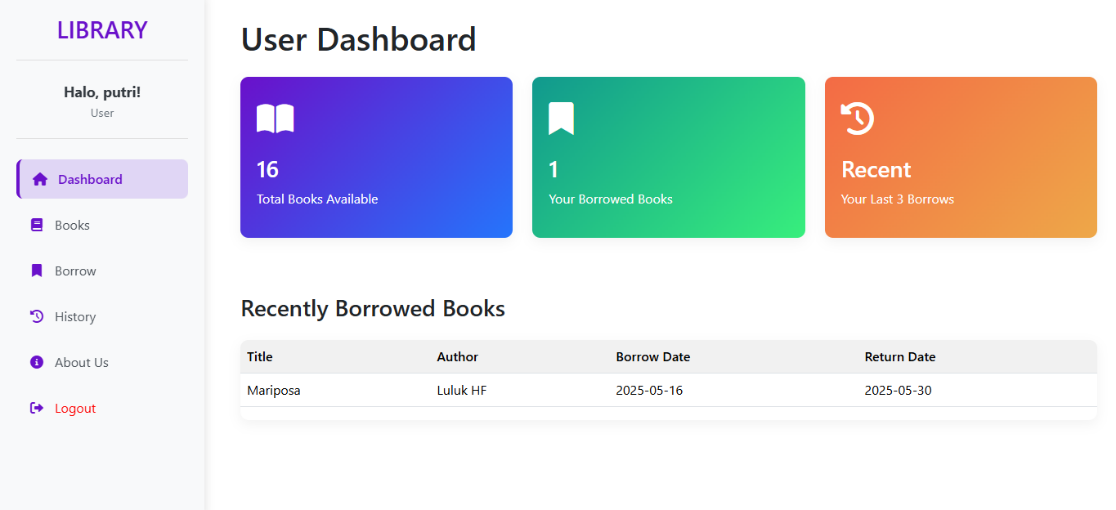


**Picture A.2**

**Login as User**

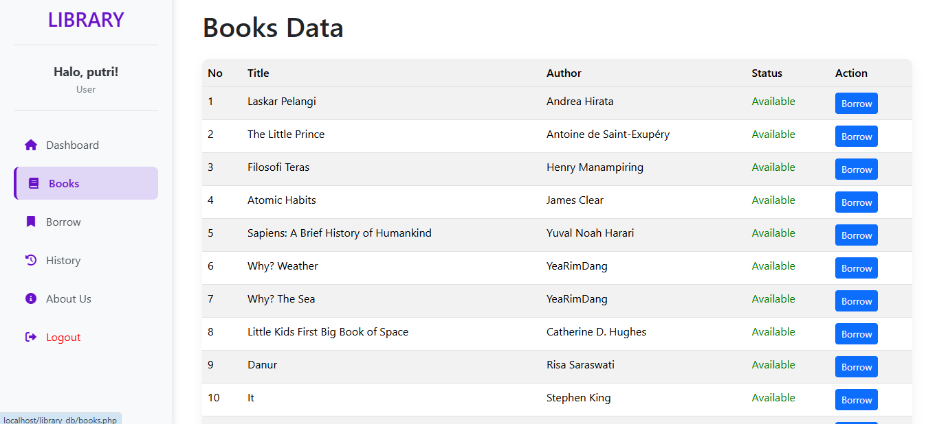
This section explains the login process from the user’s perspective. Users access the login page and enter their email and password.

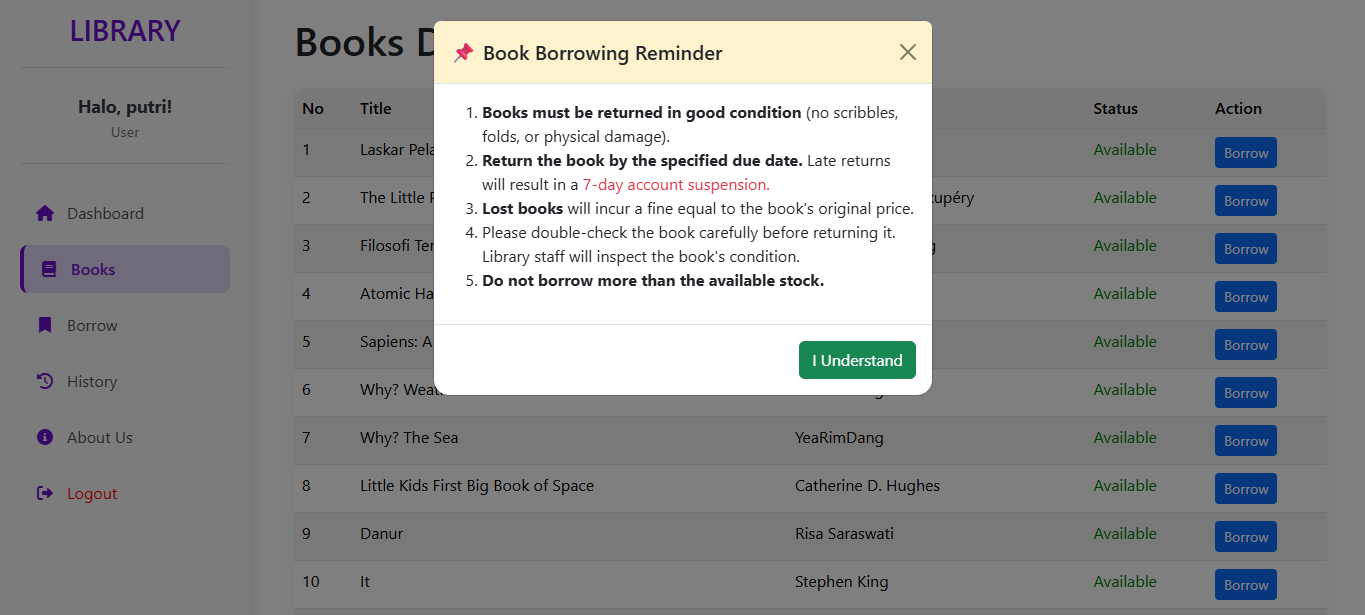
This page is the main interface shown to users after a successful login. It provides an overview of the library system, including the total number of books available, the number of books currently borrowed by the user, and a summary of their three most recent borrow activities. Below the overview section, users can see a table listing the details of their recently borrowed books, such as the title, author, borrow date, and return date. The sidebar on the left allows users to navigate through the system, including access to the books list, borrowing section, history, and other available features.



**Picture A.3**

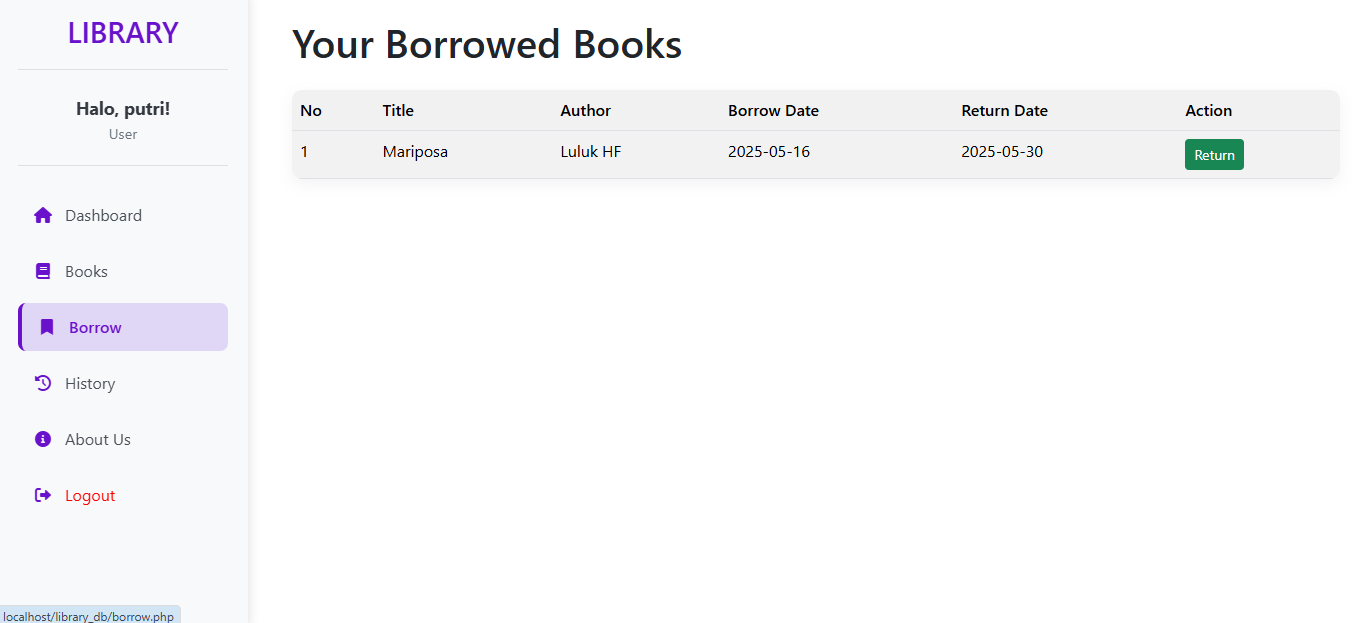
This page displays a list of all books in the library, including their titles, authors, and availability status. If a book is available, a "Borrow" button appears beside it. When the user clicks this button, a reminder notification pops up to inform them about the borrowing conditions, such as return deadlines or penalties for late returns. The user must click "I Understand" to confirm the action. Once confirmed, the system will proceed to record the borrowing and update the book’s status. This confirmation step ensures that users are aware of the borrowing rules before completing the process.





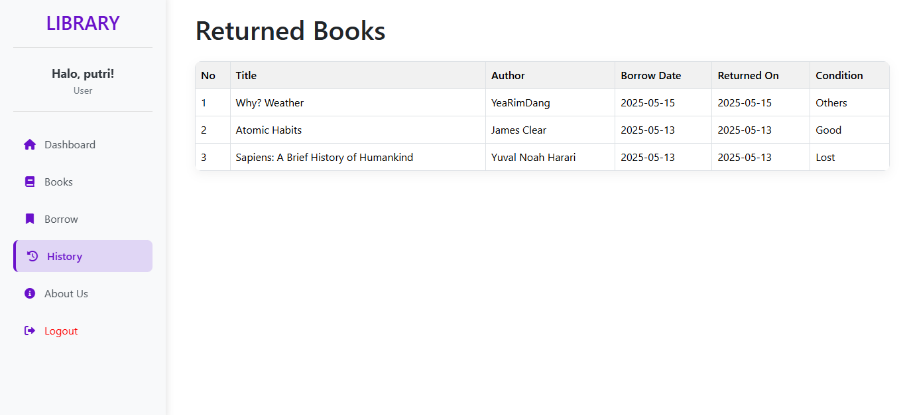
**Picture A.4 & A.5**

This page displays a list of books currently borrowed by the user. For each book, the page shows the title, the date it was borrowed, and the due date for return, which is set to 14 days from the borrowing date. Each entry includes a "Return" button. When the user clicks this button, a prompt appears asking them to select the condition of the book upon return (e.g., good, damaged, or lost). This feature helps the system track the status of each book and apply necessary actions or penalties based on its returned condition.



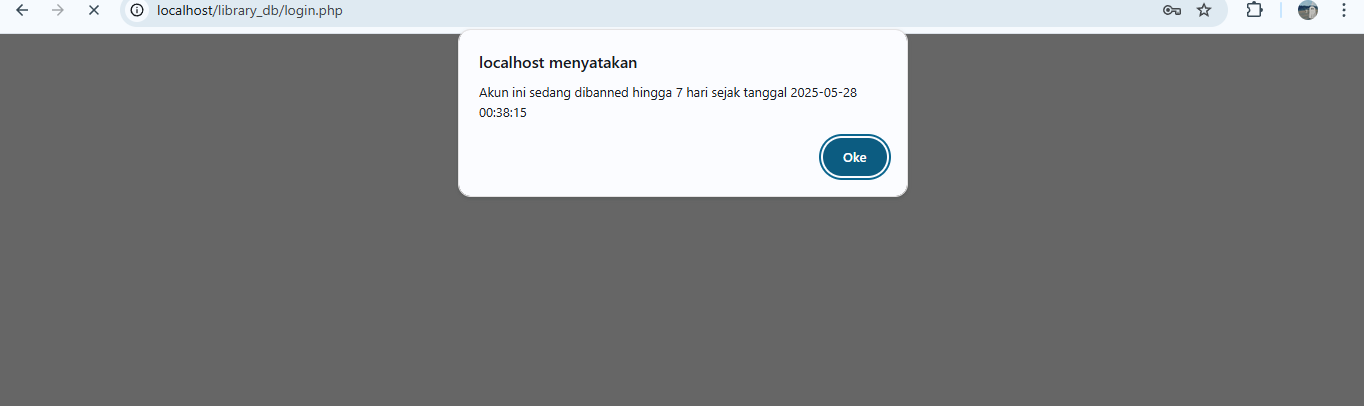
**Picture A.6**

This page displays the full borrowing history of the user. For each book, it shows the title, the date it was borrowed, the date it was returned, and the condition of the book upon return (e.g., good, damaged, or lost). The return date reflects the actual day the user completed the return process. This page helps users track their borrowing behavior and provides a clear record of all past transactions for accountability.



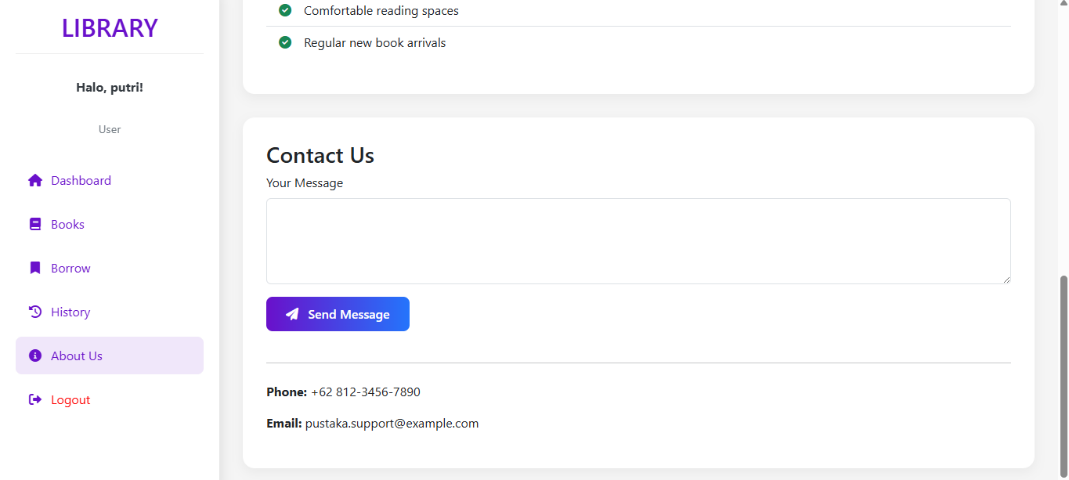
**Picture A.7**

This is the view when a user is late in returning a book; they cannot log in, and the system shows since when the restriction started :



**Picture A.8**

This page introduces users to the library and its core values. It features several sections, including **Our Services**, **Our Mission**, **Library Features**, and **Contact Us**, providing a clear overview of what the library offers and aims to achieve. At the bottom of the page, users can submit their feedback in the form of **comments, suggestions, or compliments** through an input form. Additionally, the page includes the library’s **phone number** and **email address**, allowing users to reach out directly for further inquiries or support.

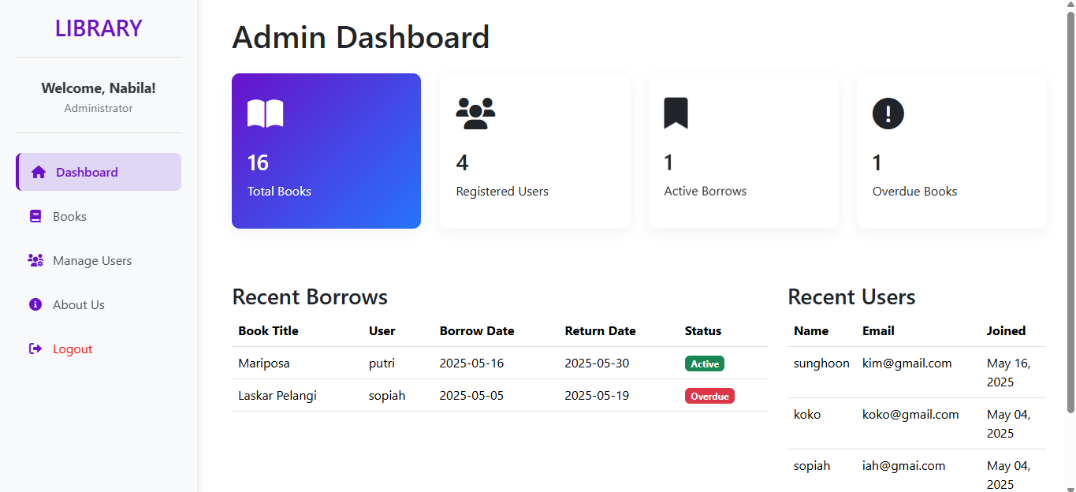


**Picture A.9**

**Login As Admin**

This section explains the login process for admin users. Admins access the same login page as regular users by entering their **email** and **password**. Once logged in, the system detects their role and redirects them to the **admin dashboard**, where they can manage users, books, and monitor system activity. No role selection is required during login, as the system automatically identifies admin accounts based on stored data.

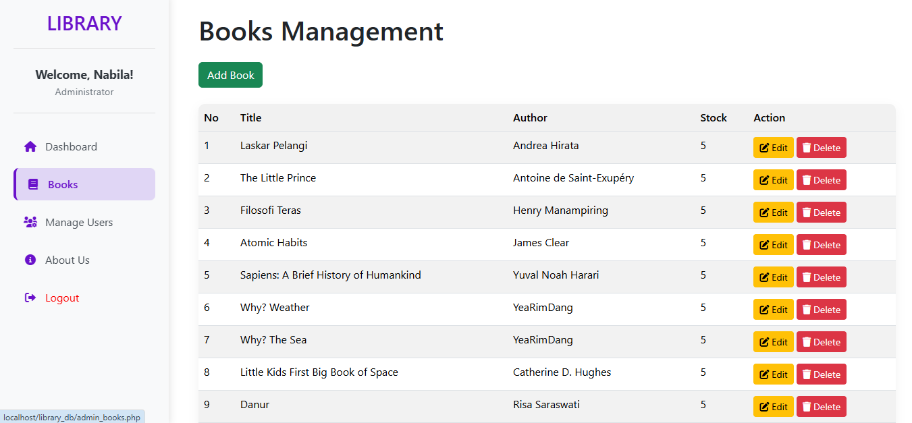
This page serves as the central hub for administrators. It displays key statistics such as the total number of books, registered users, active borrowings, and overdue books. Below the summary cards, the dashboard shows two sections: **Recent Borrows**, which lists recent book borrow activities along with their return status (Active or Overdue), and **Recent Users**, showing new users along with their name, email, and join date. This dashboard helps admins monitor user activity and manage the library system efficiently.

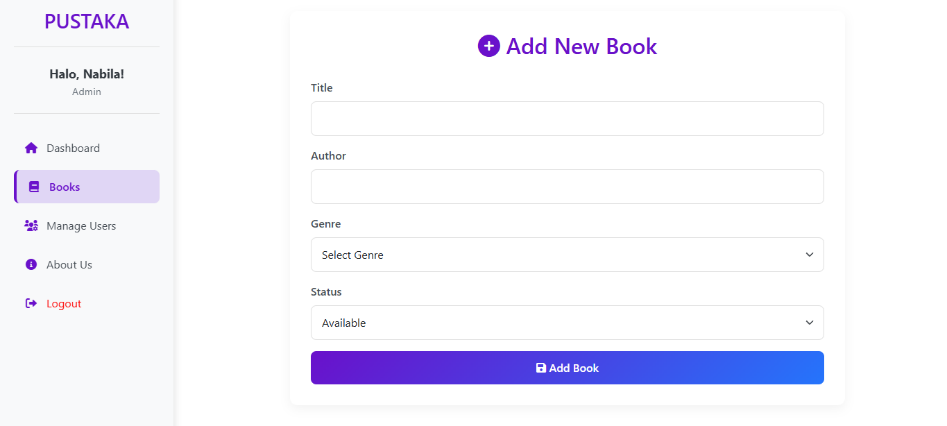


**Picture A.10**

This page is used for managing the library's book collection. At the top, there is an “Add Book” button which, when clicked, displays a form to input new book data (as shown in Figure A.12), including title, author, stock, and status.

Below the button, there is a list of existing books where the admin can edit the quantity, change the book’s status, or delete a book if needed. This page allows admins to easily update and maintain the library’s inventory

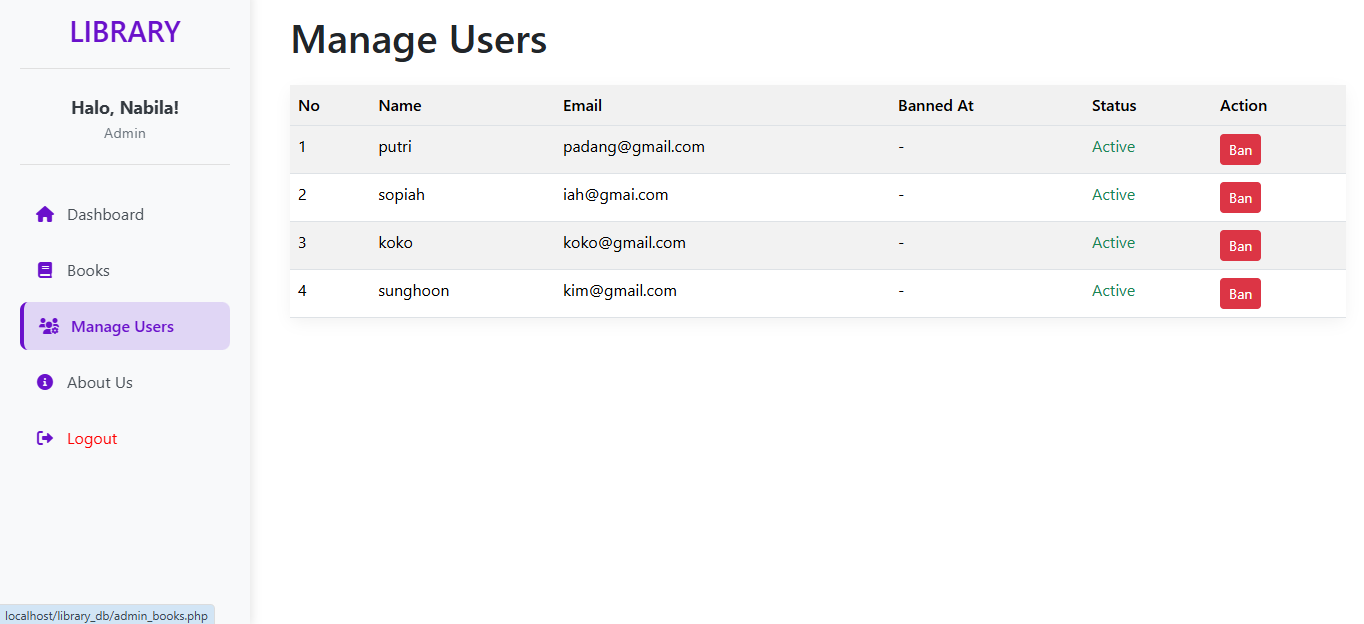




**Picture A.11 & A.12**

On the **Manage Users** page, the admin can view a list of all registered users along with their basic information, such as **name**, **email**, **status**, and **ban date** (if any). The key functionality of this page is the ability to **ban a user** by clicking the red **"Ban"** button.

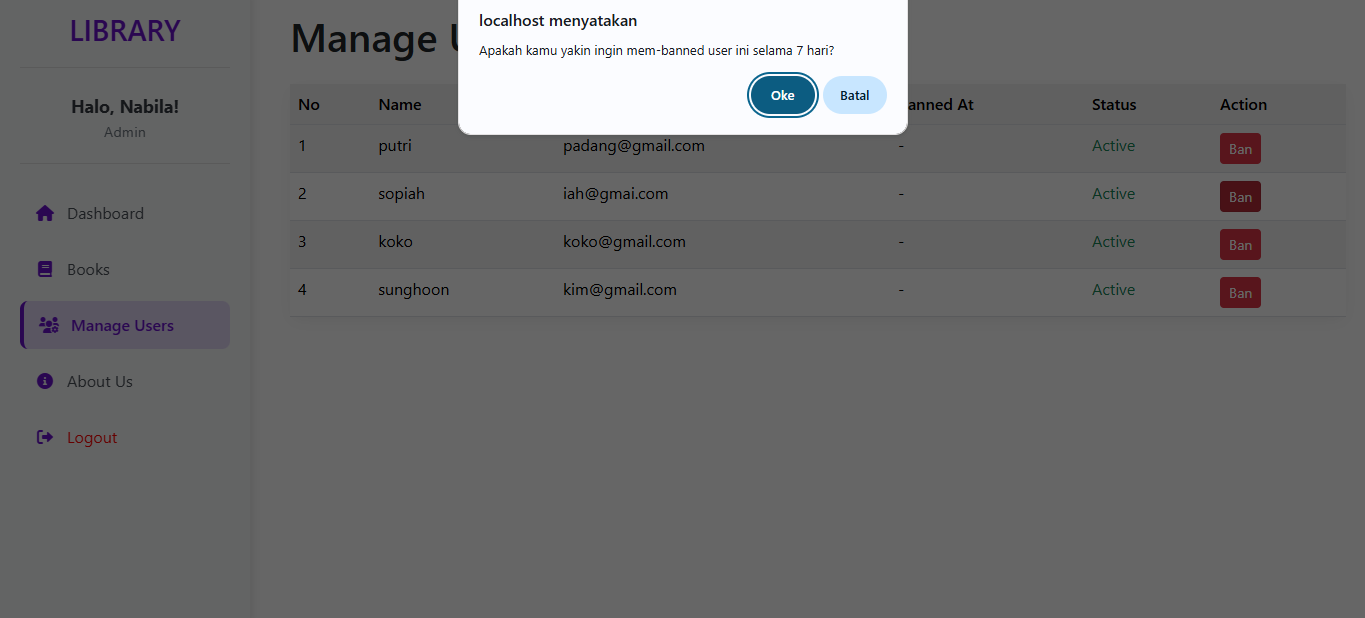
Once a user is banned, their account will be temporarily disabled for **7 days**. During this period, they will not be able to access the borrowing features. After 7 days, the system will **automatically lift the ban** and the user’s status will return to active. This feature helps the admin enforce rules and manage user behavior appropriately.



**Picture A.13**

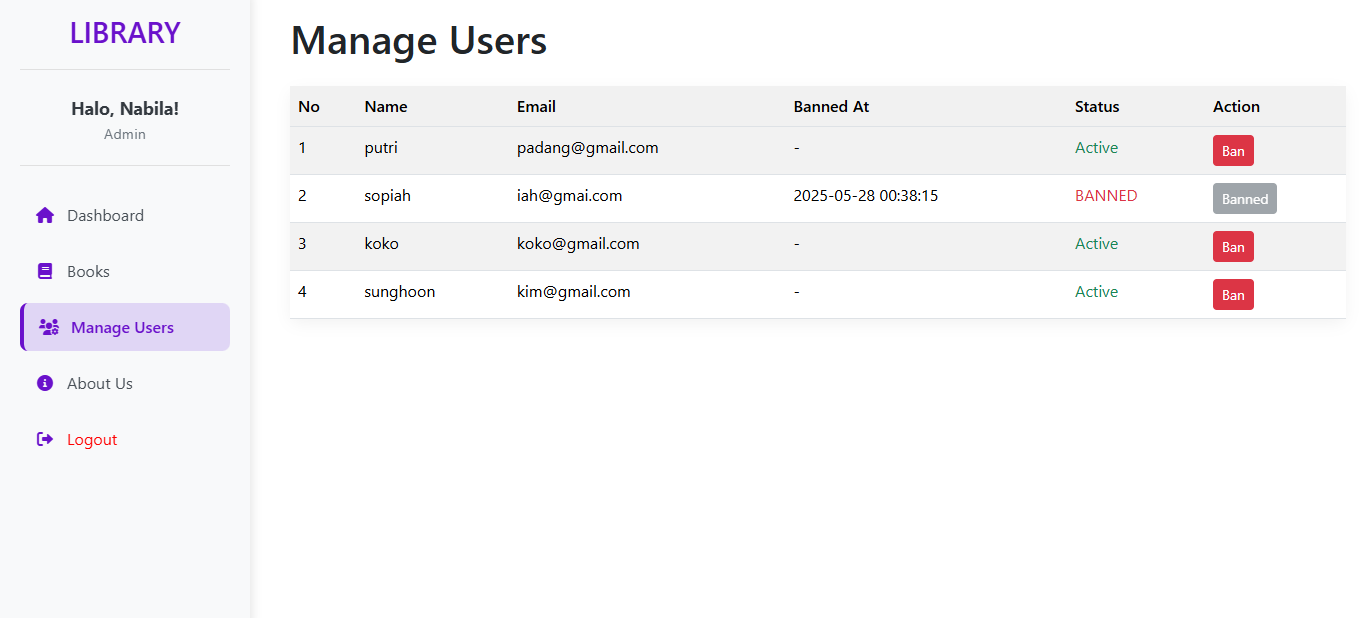
When the admin clicks the **“Ban”** button on a user’s row, a **confirmation prompt** appears with a message: **"Are you sure you want to ban this account for 7 days?"**

This note serves as a precaution to ensure the admin intends to proceed with the action. If the admin selects **"Ok"**, the banning process continues and the user will be restricted for 7 days. Otherwise, the action will be canceled. This confirmation helps prevent accidental or unintended bans.

****

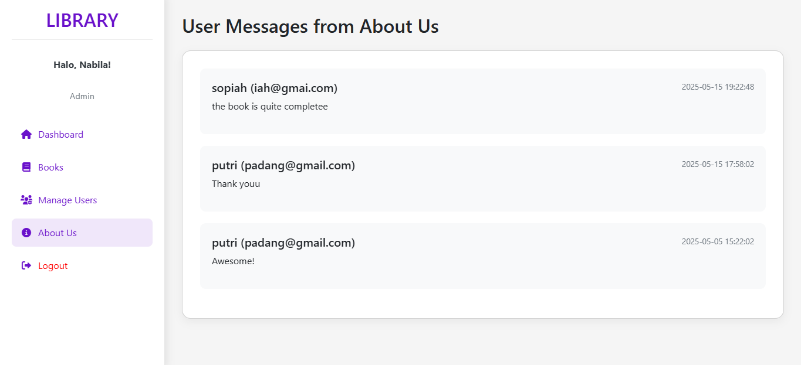
**Picture A.14**

And this is the view when the admin bans a user account :



**Picture A.15**

In the admin view, the About Us page serves to display feedback submitted by users. Admins can view comments, suggestions, criticisms, or compliments left by users through the contact section. However, admins cannot edit the content of this page or respond directly through it. This feature helps admins understand user experiences and identify areas for improvement in the library system.



**Picture A.16**

**A.2 Business Function**

* Users can borrow available books without a maximum limit.
* Books cannot be reserved in the system.
* If a user returns a book more than 14 days late, their account will be automatically banned for 7 days.
* The system records borrowing and return transactions, including dates and conditions.
* Admins can manage books, user accounts, and apply bans through the admin panel.
* User feedback such as suggestions, complaints, or praise can be submitted via the “About Us” page.
* Contact information (phone number and WhatsApp) is provided for user communication.
* Admins can view and generate reports based on borrowing history and user violations.

**A.3 Data Requirements**

* **User Account Information**: email, name, password, role, created\_at
* **Book Information**: title, author, genre, status (available/unavailable), quantity
* **Borrow Transaction**: user ID, borrow date, due date (14 days), return date
* **Borrowed Book Details**: borrow ID, list of borrowed books
* **Ban Information**: user ID, banned\_at, banned\_until, reason
* **Return Condition**: recorded when returning (e.g. good, damaged, lost)
* **Admin Privileges**: manage books, apply bans, and monitor activity
* **System Reports**: borrowing logs, book availability, and ban records
* **User Feedback**: submitted through the “About Us” page
* **Contact Info**: phone number and WhatsApp available on “About Us”

**A.4 Business Rules**

* Users can borrow any number of available books.
* Reservation is not supported in the system.
* Books must be returned within 14 days from the borrowing date.
* Returning a book more than 14 days late results in a 7-day ban.
* Lost or damaged books are recorded as violations and may lead to a ban.
* Admins can monitor, ban, and manage users and book records.
* All borrow and return transactions are logged by the system.
* Banned users cannot log in until the ban expires.
* Users can provide feedback through the “About Us” page.

**B.1 OVERVIEW**

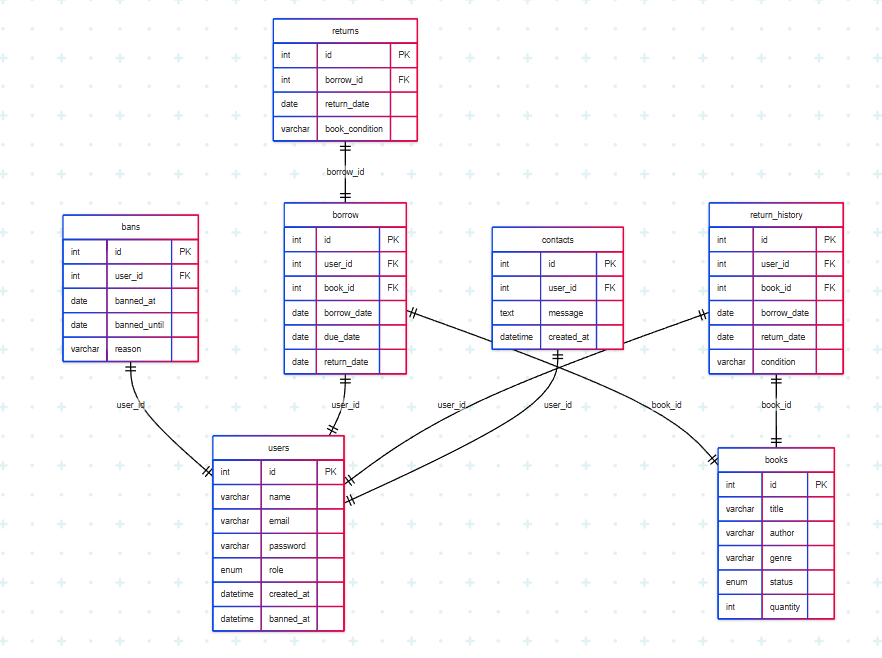
This library management system is a web-based application designed to facilitate borrowing and returning books for users while allowing administrators to manage book records and user activity. The system provides separate access for users and admins through a unified login page. Users can browse available books, borrow them directly if in stock, and view their borrowing history. A borrowing period is set to 14 days, and late returns exceeding this period will result in an automatic 7-day ban, during which the user cannot log in or borrow books.

Admins have additional access to manage users, add or update book details, view borrowing logs, and monitor system activity. In the user interface, a feedback section is provided under the "About Us" page, where users can leave suggestions, comments, or complaints. The system also includes contact information such as a phone number and WhatsApp for communication. Overall, the platform aims to deliver a simple yet functional solution for library users and administrators.

**B.2 REVISED BUSINESS RULES AND ASSUMPTIONS**

1. There is no limit to how many books a user can borrow at a time.
2. Users who are banned due to late returns (more than 14 days) are not allowed to log in, borrow books, or interact with the system.
3. The system does not support book reservations.
4. Late returns, lost, or damaged books will result in a 7-day automatic ban from the system.
5. Library staff (admin) can access and manage all user data, book data, borrowing history, and ban records.
6. The system must check book availability before allowing a user to borrow.
7. A borrow transaction may include one or multiple books.
8. A book can be borrowed multiple times by different users at different periods as long as the quantity is available.
9. The borrowed\_books table acts as a junction table that links a borrow transaction with multiple book records.
10. Each borrow and return action must be timestamped to maintain historical accuracy.
11. Feedback, suggestions, or comments from users are submitted via the "About Us" page, not a contact messages table.

**B.3 ERD (CONCEPTUAL DATA MODELLING)**



**B.4 JUSTIFICATIONS OF ERD BASED ON BUSINESS RULES & ASSUMPTIONS**

**BUSINESS RULES RELATED TO ENTITY users**

* **BR1:** A user must register with a unique email, name, and password.
* *Justification:* The users table includes name, email, and password, and enforces uniqueness on email.
* **BR2:** Each user is assigned a role (admin or user).
  + *Justification:* The role field (ENUM) in users distinguishes account privileges.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY books**

* **BR3:** Each book has title, author, genre, status, and quantity.
  + *Justification:* The books table contains all these attributes to manage book metadata and inventory.
* **BR4:** The status field tracks whether a book is available, borrowed, or reserved.
  + *Justification:* The status ENUM controls lending operations.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY borrow`**

* **BR5:** Each borrow record must store the user, book borrowed, borrow date, due date, and return date.
  + *Justification:* The borrow table includes user\_id, book\_id, borrow\_date, due\_date, and optional return\_date.
* **BR6:** Every borrow must reference one user and one book.
  + *Justification:* The foreign keys user\_id and book\_id enforce this relationship.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY returns**

* **BR7:** Book returns must include return date and condition.
  + *Justification:* The returns table links to a borrow transaction via borrow\_id, and includes return\_date and book\_condition.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY return\_history**

* **BR8:** Past returns are stored with user ID, book ID, borrow and return dates, and condition.
  + *Justification:* The return\_history table records this data for performance tracking and audit.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY bans**

* **BR9:** Users can be banned with a start and end date and a reason.
  + *Justification:* The bans table includes banned\_at, banned\_until, reason, and links to users via user\_id.
* **BR10:** Banned users cannot log in or borrow books.
  + *Justification:* The system references this table to restrict user access during login or borrowing.

**Primary Key:** id

**BUSINESS RULES RELATED TO ENTITY contacts**

* **BR11:** Users can send messages or feedback to the admin.
  + *Justification:* The contacts table stores user messages along with created\_at and user\_id.

**Primary Key:** id

**Why Is the Library Management System Built on CRUD Operations?**

The library management system you’ve developed is fundamentally based on **CRUD operations—Create, Read, Update, and Delete**. These four basic functions form the core of how the system handles user interactions, data storage, and overall functionality. Without CRUD, key features like book borrowing, user registration, and ban management wouldn’t work effectively.

**1. Create — Adding New Data**

The **Create** function is used when new records are added to the system. Examples include:

* When a user **registers for an account**, their name, email, password, and role (admin/user) are stored in the users table.
* When a user **borrows a book**, the system logs the borrowing date, due date, and related user and book in the borrow table.
* Admins can **add new books** to the catalog, including details like title, author, genre, and stock.
* If a book is returned more than 14 days late, the system automatically **creates a ban record** in the bans table.

Example :   
When a user clicks the *"Borrow Book"* button, a new entry is inserted into the borrow table.

**2. Read — Displaying and Retrieving Data**

The **Read** function allows users and admins to view existing information from the database. For example:

* Users can **browse available books** along with their titles, authors, status, and quantity.
* Admins can **monitor all borrowing and return activities** by users.
* Users can see their **borrowing history**, including any bans or overdue returns.
* Feedback messages sent by users are stored in the contacts table and can be read by the admin.

Example :  
When a user opens the *"Book List"* page, the system queries the books table and displays the results.

**3. Update — Modifying Existing Data**

The **Update** function is used to change or complete existing data. This happens when:

* A user **returns a book**, and the system fills in the return\_date in the borrow table.
* Admins **update book information**, such as changing the quantity or correcting the title or author name.
* When a returned book is damaged, the **book condition** is recorded in the returns table.

Example :  
When the admin edits a book’s details, the system updates that book’s record in the books table.

**4. Delete — Removing Data**

The **Delete** function is used to remove unnecessary or outdated data. For example:

* Admins can **delete books** that are no longer in circulation.
* Feedback messages or old records may be **deleted** to clean up the database.
* Outdated borrow or return logs may be deleted as part of maintenance.

Example :

When the admin deletes a book entry, the system removes it from the books table.

**Conclusion: The Role of CRUD in This System**

CRUD operations are essential to keep the system:

* **Structured and organized**, as all data manipulation follows a consistent pattern.
* **Easy to maintain and expand**, since new features like statistics or search filters can be added within the CRUD framework.
* **Reliable and consistent**, ensuring data such as borrow and return records are always linked correctly to users and books.
* **Flexible for multiple user roles**, allowing both users and admins to interact with the system in different ways.

In short, CRUD is what allows this library system to function as a complete and user-friendly application. Every major operation relies on creating, viewing, editing, or removing data—making CRUD the foundation of its design.

**Why Is the Library Borrowing System Built Using CRUD Operations?**

CRUD operations—Create, Read, Update, and Delete—form the essential foundation for managing data in any database-driven system. The Library Borrowing System is built upon these operations to ensure structured, consistent, and reliable data handling for both users and administrators. Here’s how each CRUD component applies in this system:

* **Create (Data Entry):**  
  New records are created when users register for accounts, administrators add new books to the catalog, borrowing transactions are logged, and bans are issued to users who violate borrowing policies.  
  Example: When a user borrows a book, a new entry is added to the borrow table. If the return is overdue, a related entry is added to the bans table.
* **Read (Data Retrieval):**  
  The system allows users and admins to access existing information. Users can browse available books, view borrowing history, and check return status. Admins can monitor user activity, ban records, and inventory.  
  Example: When a user accesses their account, the system retrieves data from the database to display borrowing history on the dashboard.
* **Update (Data Modification):**  
  Records can be edited when needed. Users can update their profile details or passwords. Admins can change book information such as status or quantity, and adjust ban periods. Return records are also updated with actual return dates.  
  Example*:* After a book is returned, the return\_date in the borrow table is updated and the book status is set back to "available".
* **Delete (Data Removal):**  
  Unneeded records can be removed to maintain the system. Admins may delete old or damaged books, clear expired bans, or remove outdated borrow records.  
  Example: An admin deletes a book from the books table if it's no longer in good condition, so users can't see or borrow it.

**Conclusion:**  
Using CRUD operations allows the Library Borrowing System to handle data in a clear, organized, and scalable manner. Each feature—such as managing users, books, borrowing activity, and bans—relies on these fundamental operations to work efficiently. This structure not only simplifies development and maintenance but also ensures that the system remains robust and adaptable as it grows.