PROJECT REPORT

LIBRARY MANAGEMENT SYSTEM WEBSITE

To fulfil assignment for Server-Side Internet Programming

Lecturer: Williem, M.Sc.

Compile By:

Qwyn Celine Djimondo	**(001202400205)**
Tentin Nofa Tuzika	**(001202400134)**
Tessalonika Angeline Purba	**(001202400210)**

PRESIDENT UNIVERSITY FACULTY OF COMPUTING

NORTH CIKARANG

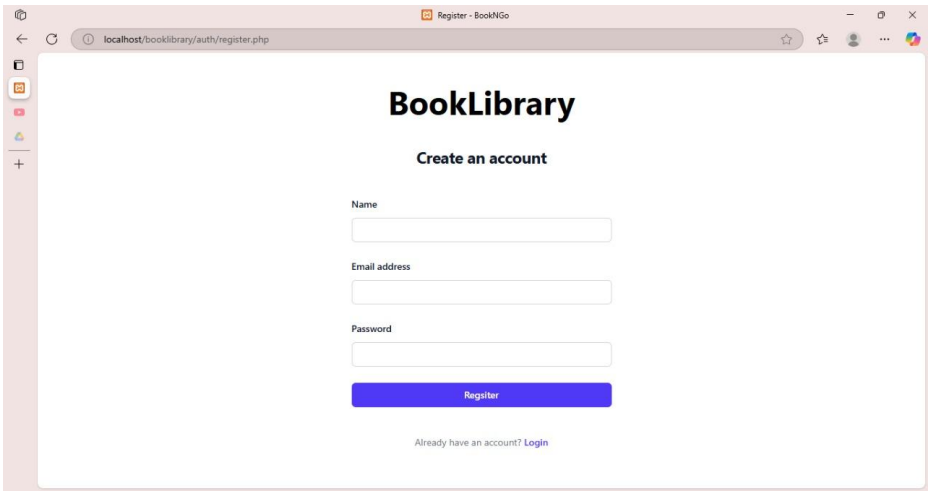
2025

I. INTRODUCTION

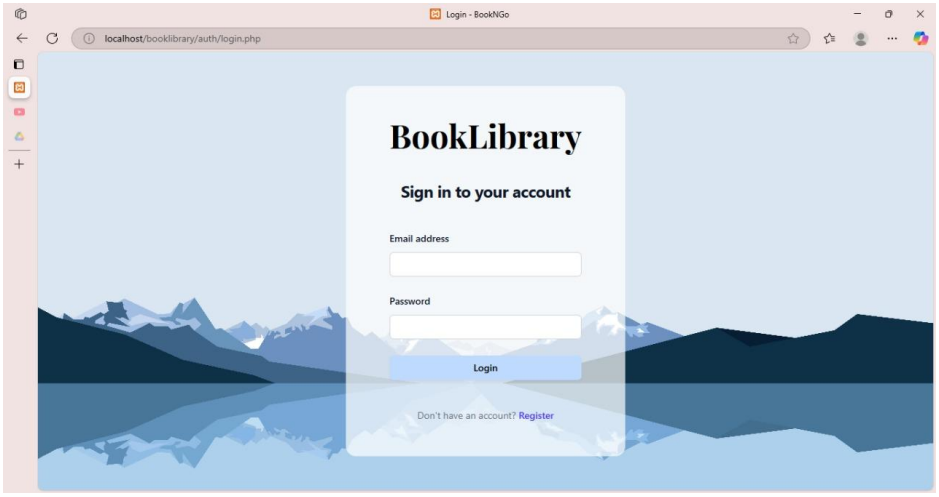
In many libraries, the traditional process of borrowing and returning books is frequent paper-based and time-consuming. Manual record-keeping leads to inefficiencies, including misplaced records, untracked overdue returns, and limited data accessibility. As libraries expand their collections and user base, there is a growing need for a more systematic and digital solution. This project aims to develop a web-based Library Management System that allows users to register, search for books, borrow and return books, and view their borrowing history. The system supports two types of users: general users and administrators. Users can sign up and log in to access the system. After logging in, they can browse available books, view book details, and rent books. They can also view their past borrowing records. A key feature of this system is that users can book (reserve) physical books online from anywhere, allowing them to conveniently secure a book before picking it up directly from the library. However, since the system is designed specifically for physical books only (not soft copies), the return process requires users to meet with an admin in person. During this meeting, both parties can fill in the return data, and the user is expected to upload proof of return, ensuring accuracy and accountable tracking of physical book transactions. Administrators also have special privileges to manage books and categories, view user data, handle reports, manage rented books, and access system activity logs. This website is built using HTML and CSS, PHP, MySQL technologies, with a responsive design focused on ease of navigation.

The application flow guides users through the following key forms and processes:

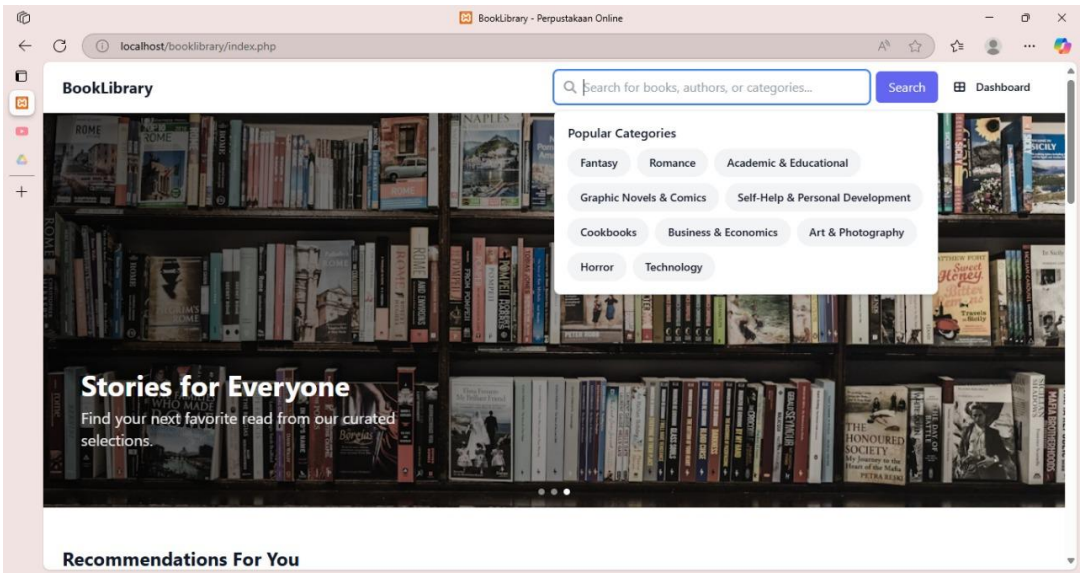
1. **Role Selection**. Before logging in, users are given the option to choose their role, namely between Users and Admins.
2. **Register.**

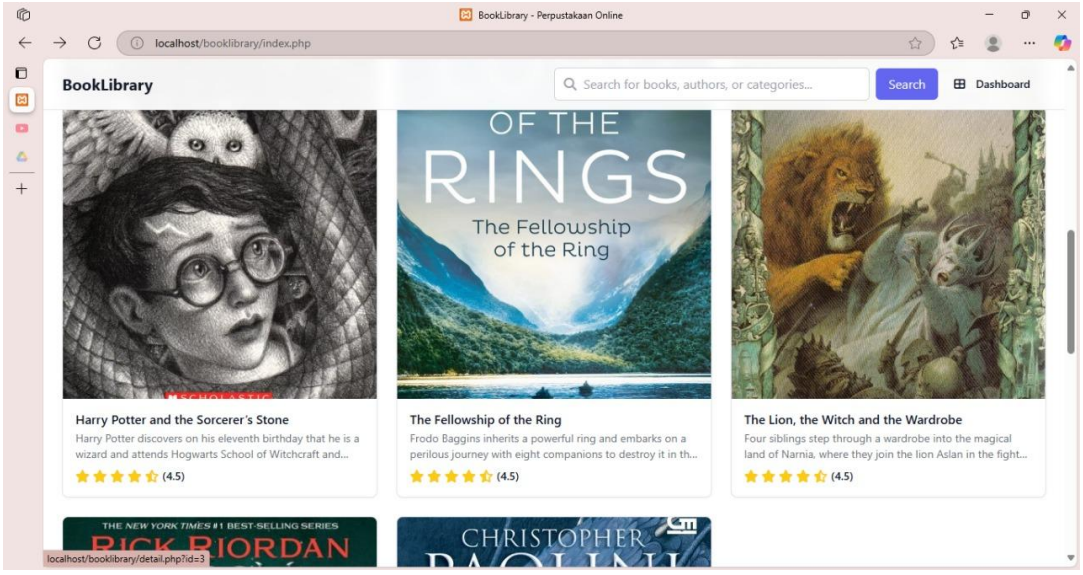


3. **Login.** Allows users and admins to log in using email and password after selecting a role.

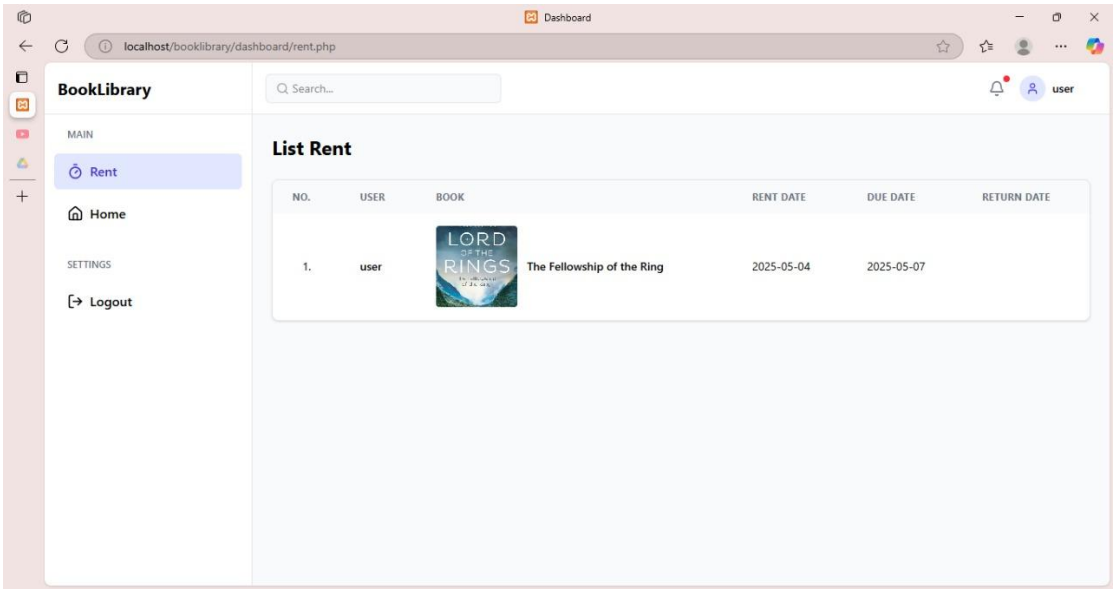


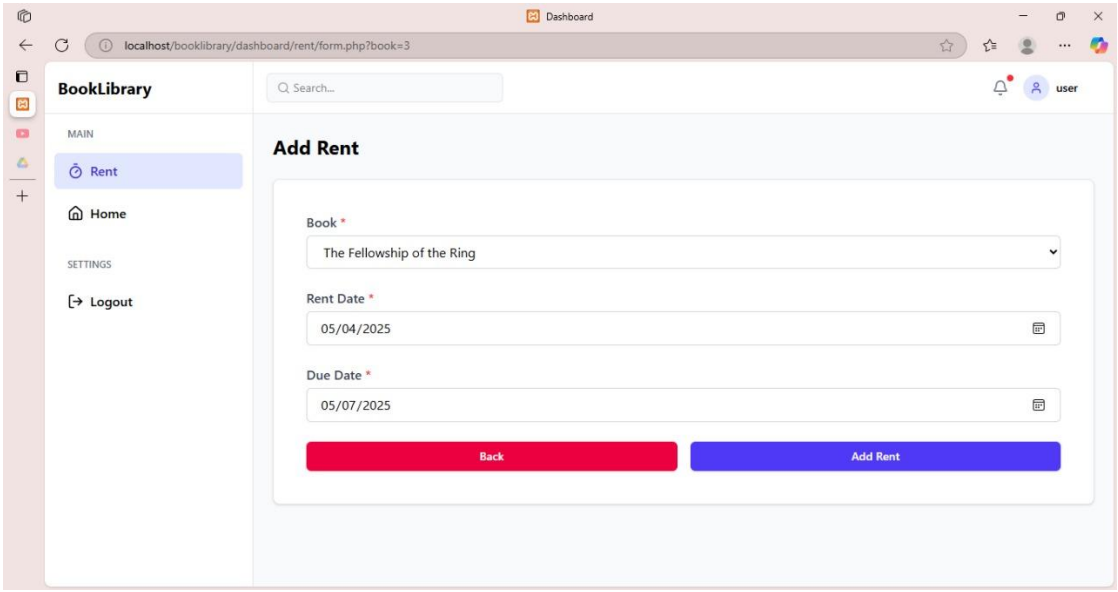
4. **Home Page**. The main landing page after the user and admin roles login, users can browse all available books and make rentals, while admins can add books, view book rental history, and report books. Users and admins each have their own dashboard page.



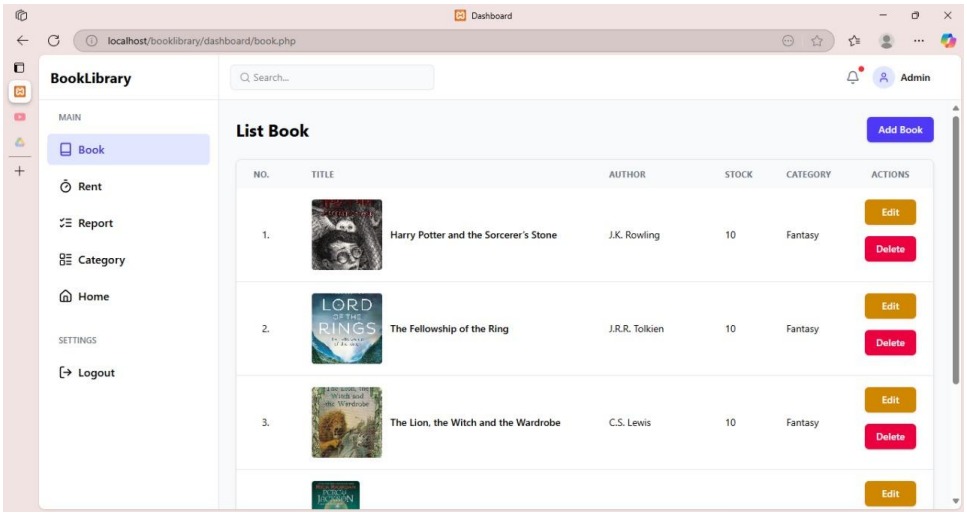


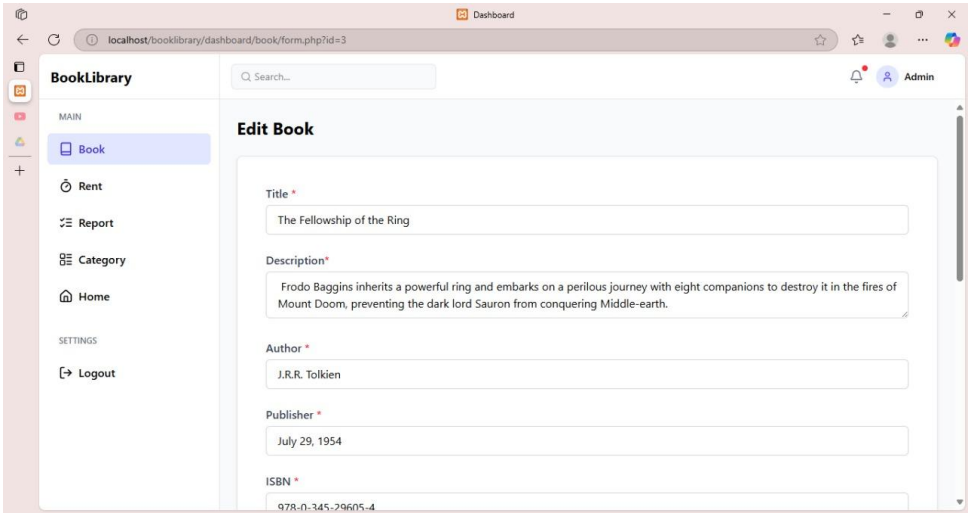
5. **Rent History**. Users can view their **complete rent history** through a dedicated section in their dashboard. This section provides a detailed record of all previously borrowed books, including the **titles** of the books, the **borrow and return dates**, and any **late fees** that may have been applied. This feature allows users to track their borrowing activity over time, manage their reading habits, and stay informed about any penalties incurred due to overdue returns.



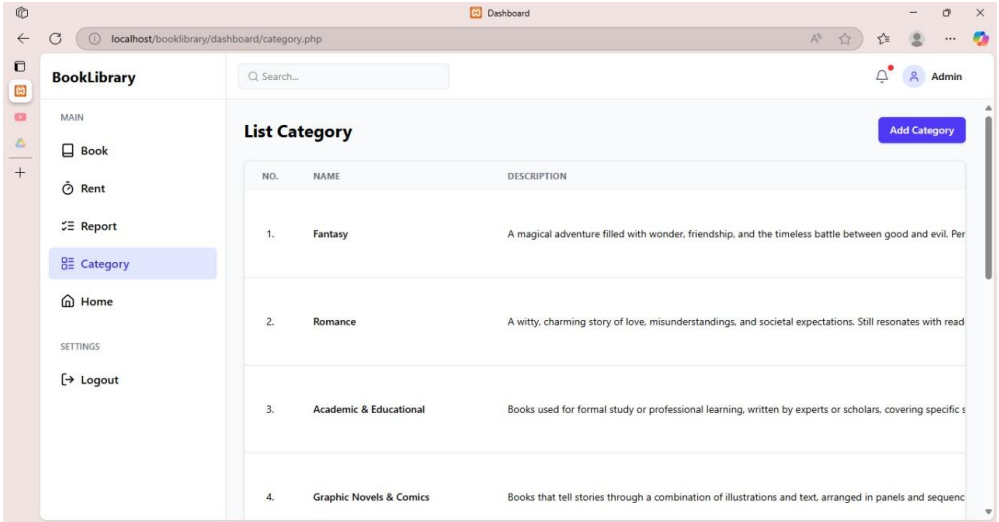


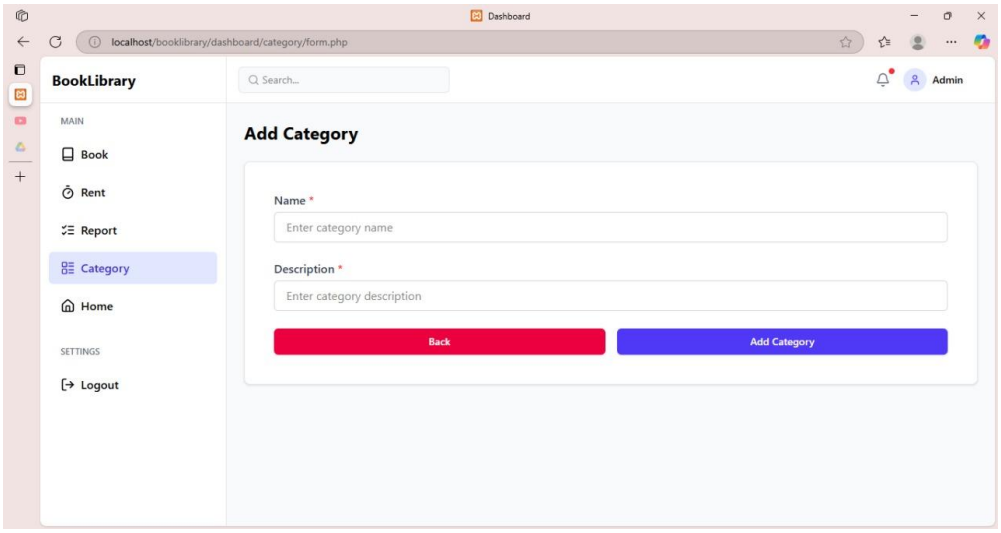
6. **Book Page.** Allows admin to add new books.



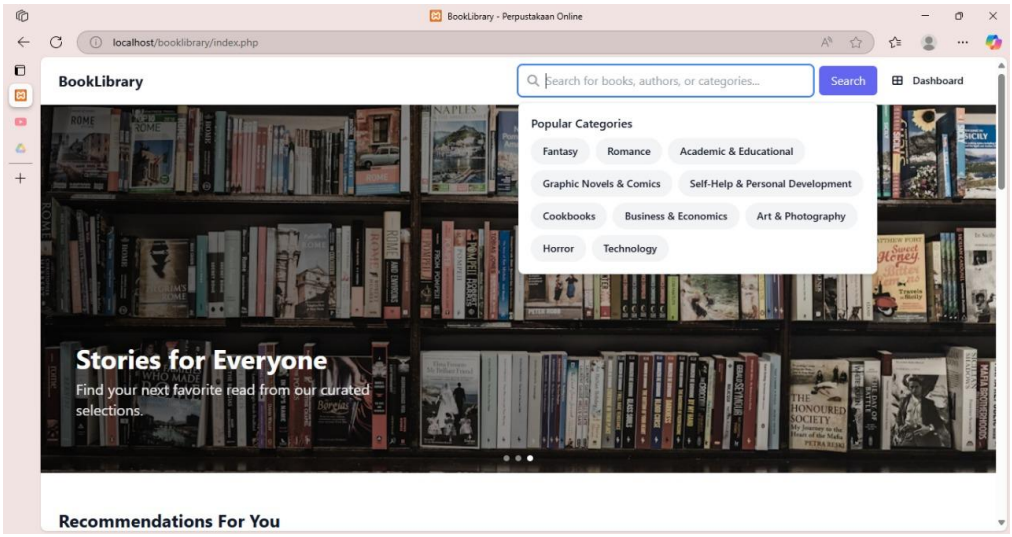


7. **Category View**. Allows users and admins to browse all book categories, available books, search for books, and add new books according to their categories.



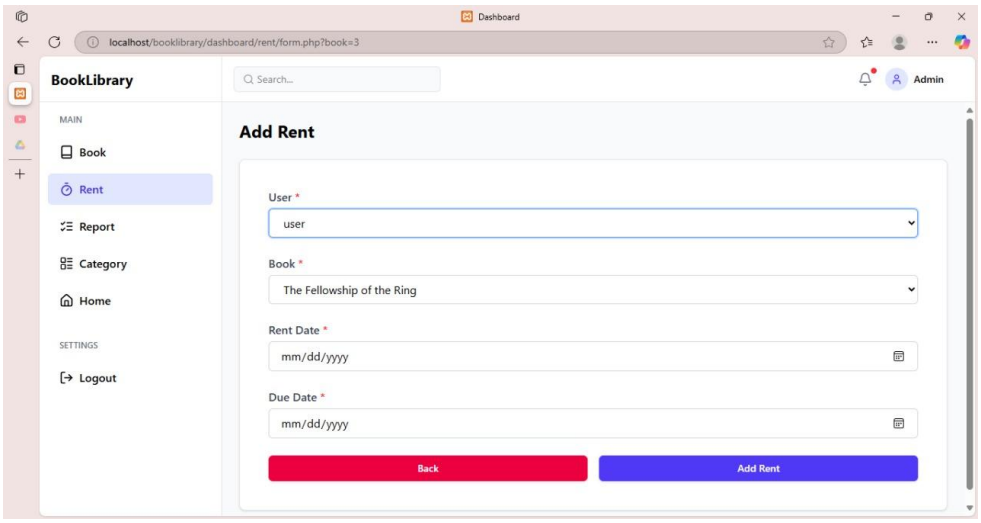


admin category view

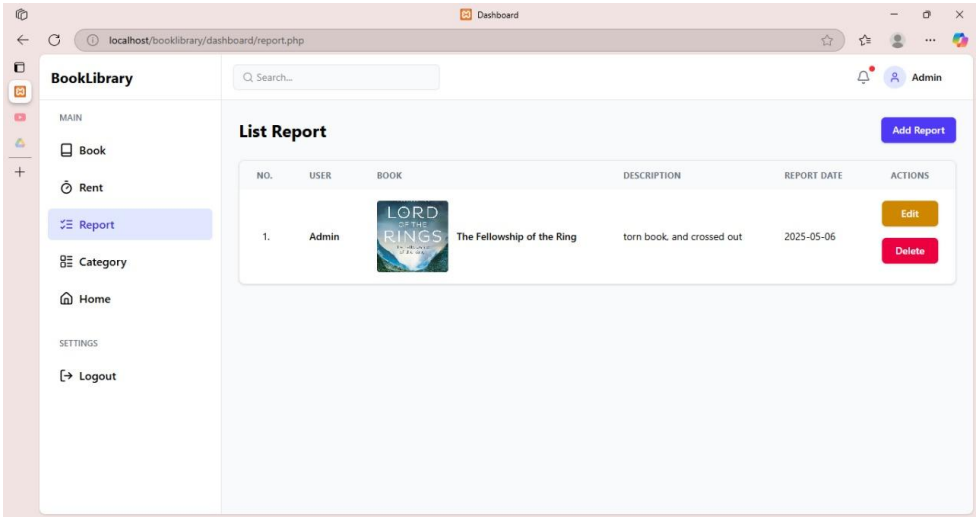


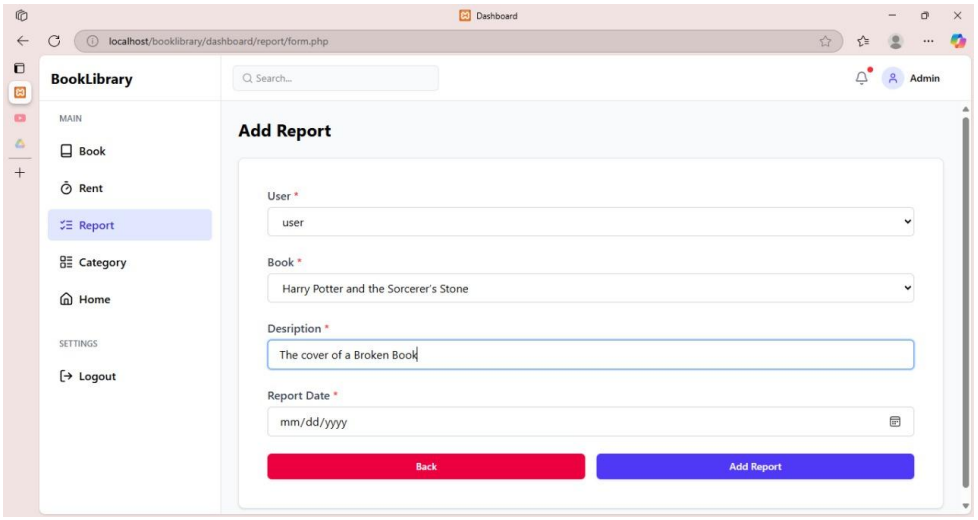
user category view

8. **Rent Page**. The rental page allows admins to help users rent available books and see what books are currently being rented.



9. **Report.** Allows admins to assist users in renting available books, such as torn books, missing pages, etc., then admins can see a list of reported books.



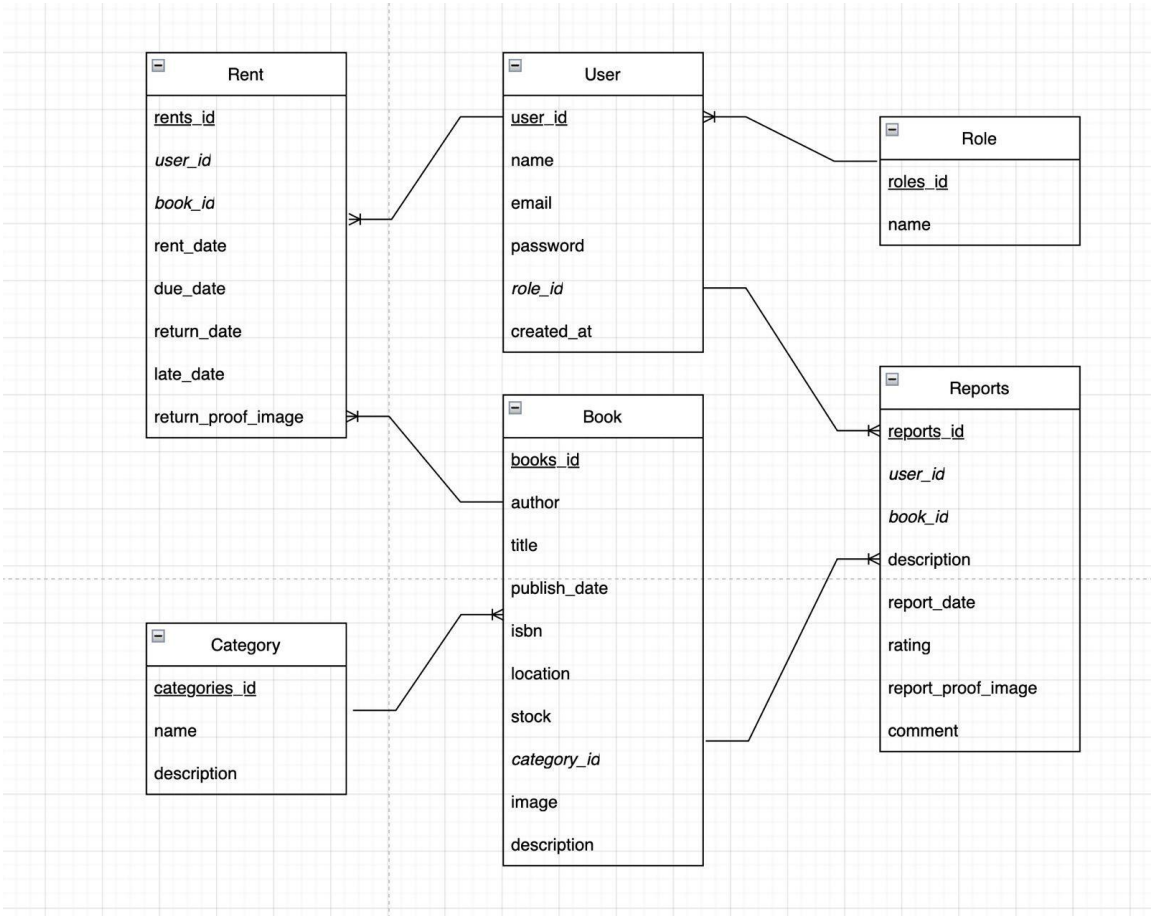


II. BUSINESS FUNCTIONS

This system supports the following main business functions which can be accessed by the admin.

1. User registration and login: Create accounts and log in securely using secure email and passwords for various roles.
2. Book list: Add, edit, and delete books
3. Search books & Categories: View available books and book categories.
4. Rents: Reservations for available book rentals.
5. Report: Report a damaged book.

III. DATA REQUIREMENTS



The system requires data entities stored in DATABASE tables.

1. Role (roles_ID, name)
2. User (user_ID, name, email, password, role_ID)

3. Book (books_ID, author, title, publish_date, isbn, location, stock, category_ID, image, description)
4. Category (categories_ID, name, description)
5. Rent (rents_ID, user_ID, book_ID, rent_date, due_date, return_date, late_date, return_proof_image)
6. Report (report_ID, user_ID, book_ID, description, report_date, rating, report_proof_image, comment)

IV. BUSINESS RULES

This system operates based on the following business rules.

1. User Roles & Access
   - Users must register with a valid and secure email and password to access features on the system.
   - One user can only have one role per account, namely User and Admin.
   - Users can only view books that are available for rental.
   - Admin can add, edit, delete and report books
2. Book
   - Admin can see the list of books.
   - Admin can add books by entering the book title, description, publisher, publication date and ISBN.
3. Category
   - Admin can see the list of existing book categories.
   - Admin can add new book categories by entering the category name and description.
4. Rent
   - Users and Admins can view a list of book rentals.
   - Admin can help Users to rent, and Users can also rent books themselves by entering the name of the book, the date of picking up and returning the book.
5. Report
   - Admin can see the list of reported books.
   - Admin can help users to report damaged books by filling in the user name, book name, description, and reporting date.

V. CRUD PROCESSES IMPLEMENTATION

1. User

Operation	Description
Create	Users can register on the website by filling out the registration form. This creates a new record in the user table.
Read	Admins can view the list of registered users, and users can view their own profiles.
Update	Admins can update user roles or account details, and users can update their own personal information.
Delete	Admins can delete user accounts when necessary.

2. Book

Operation	Description
Create	Admins can add a new book by submitting a form that includes the title, author, ISBN, stock, location, and cover image.
Read	All users can browse and view book information from the book list on the website.
Update	Admins can update existing book details, such as stock quantity, title, or description.
Delete	Admins can delete books that are no longer available from the database.

3. Category

Operation	Description

Create	Admins can create new categories by entering a category name and description.
Read	All users can view a list of available categories when browsing books.
Update	Admins can update the name or details of a category.
Delete	Admins can remove unused or outdated categories from the system.

4. **Rent / Borrowing**

Operation	**Description**
Create	A borrowing record is created when a user rents a book by submitting the rental form.
Read	Users can view their borrowing history, and admins can view all borrowing records.
Update	Admins can update return dates, rental status, or mark books as returned
Delete	Admins can delete invalid or cancelled rental records.

5. Report

Operation	**Description**
Create	Users can submit a report by filling out the report form with a description, rating, and optional proof photo.
Read	Admins can read all submitted reports from the report dashboard.
Update	Admins can update report status, add follow-up comments, or mark it as resolved.
Delete	Admins can remove resolved or irrelevant

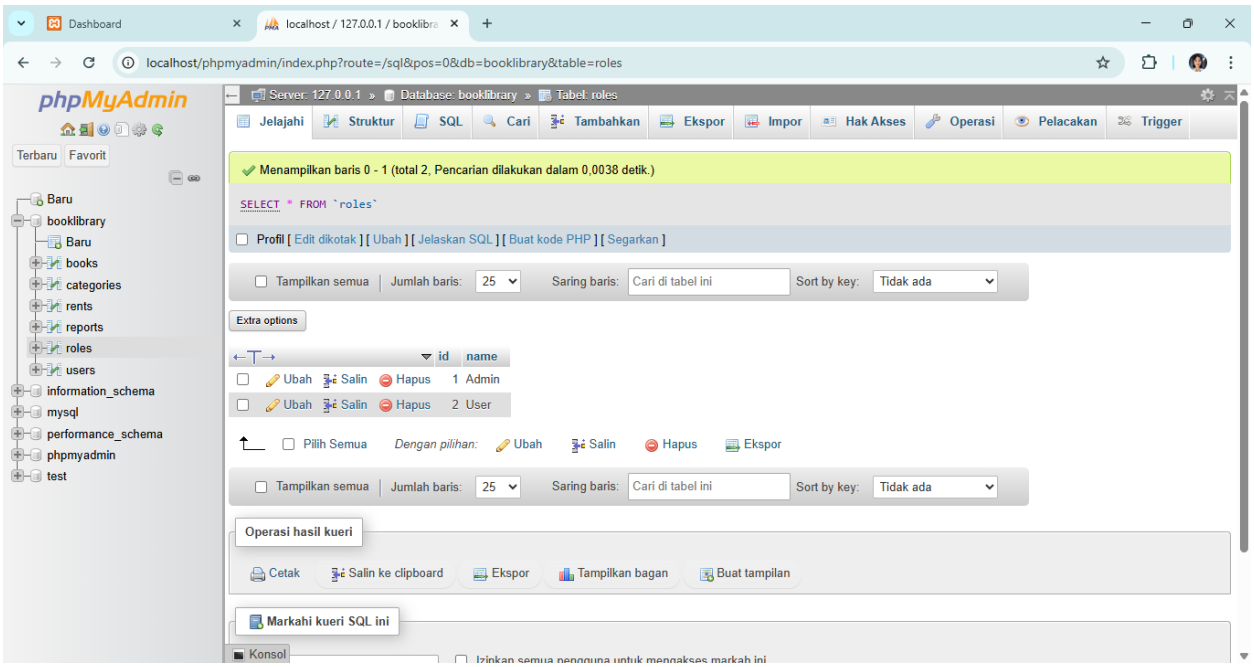
	reports from the system.

VI. DATABASE TABLE STRUCTURE

This system uses the following database tables.

Role Table

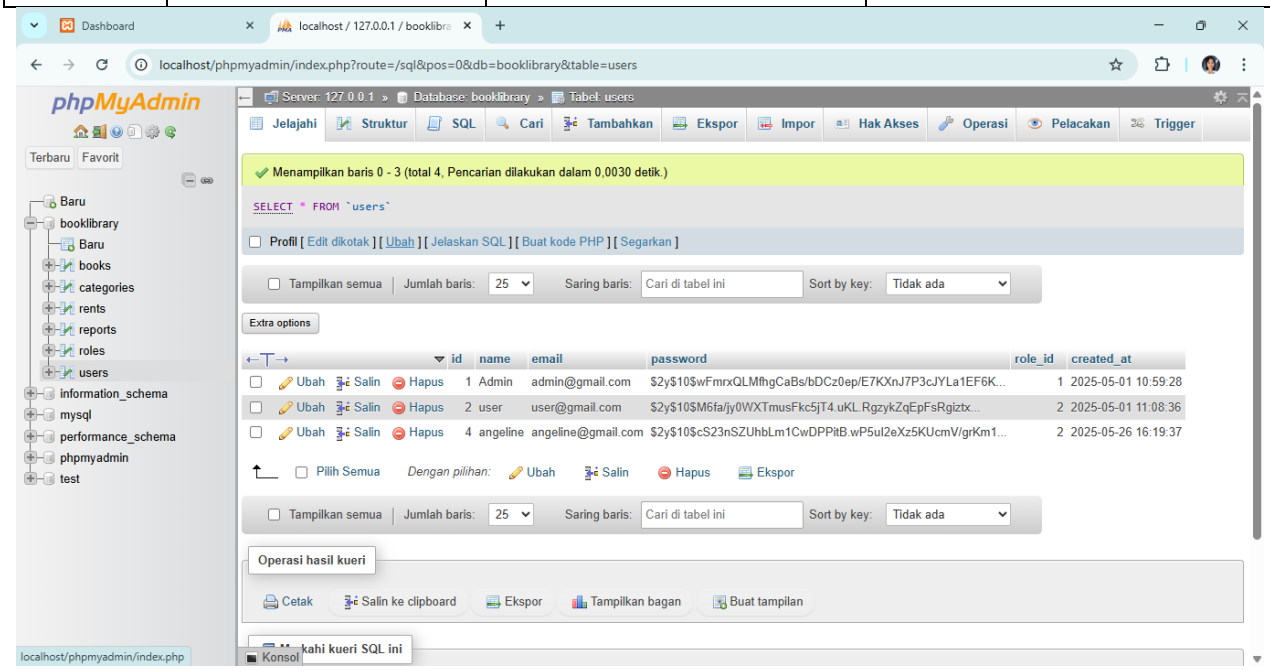
Column Name	Data Type	Constraints	Description
roles_ID	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each role.
name	VARCHAR(50)	NOT NULL	Name of the role (e.g., admin, librarian, member).



Users Table

Column Name	Data Type	Constraints	Description
user_ID	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each user.
name	VARCHAR(100)	NOT NULL	Full name of the user.

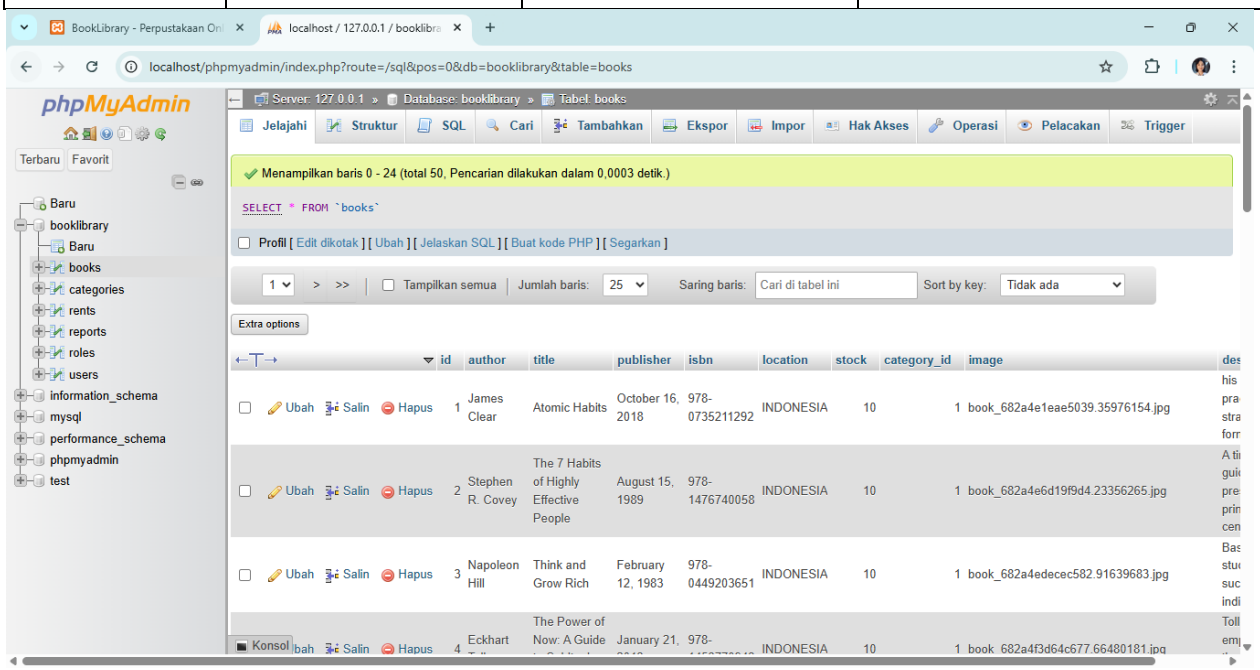
email	VARCHAR(100)	NOT NULL	User's email address (for login).
password	VARCHAR(255)	NOT NULL	Hashed password for user authentication.
role_ID	INT(11)	NOT NULL, FOREIGN KEY REFERENCES Role(roles_id)	Reference to the user's role.
created_at	TIMESTAMP	NOT NULL, DEFAULT CURRENT_TIMESTAMP	Timestamp when the user account was created.



Book Table

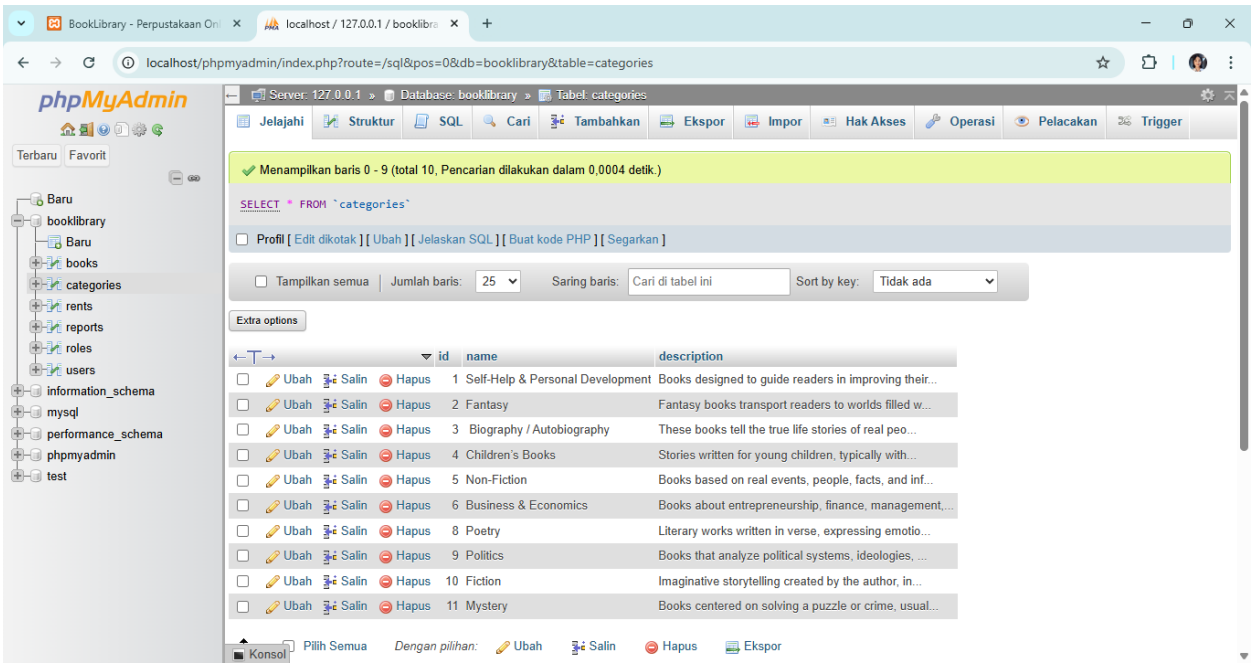
Column Name	Data Type	Constraints	Description
books_ID	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each book.
author	VARCHAR(100)	NOT NULL	Name of the book's author.
title	VARCHAR(255)	NOT NULL	Title of the book.
publish_date	DATE		Publication date of the book.
isbn	VARCHAR(20)		International Standard Book Number.

location	VARCHAR(100)		Physical location of the book in the library.
stock	INT(5)	NOT NULL, DEFAULT 1	Number of copies available.
tegory_ID	INT(11)	FOREIGN KEY REFERENCES Category(categories_id)	Reference to the book's category.
image	VARCHAR(255)		URL or path to the book cover image.
description	TEXT		Detailed description of the book.



Category Table

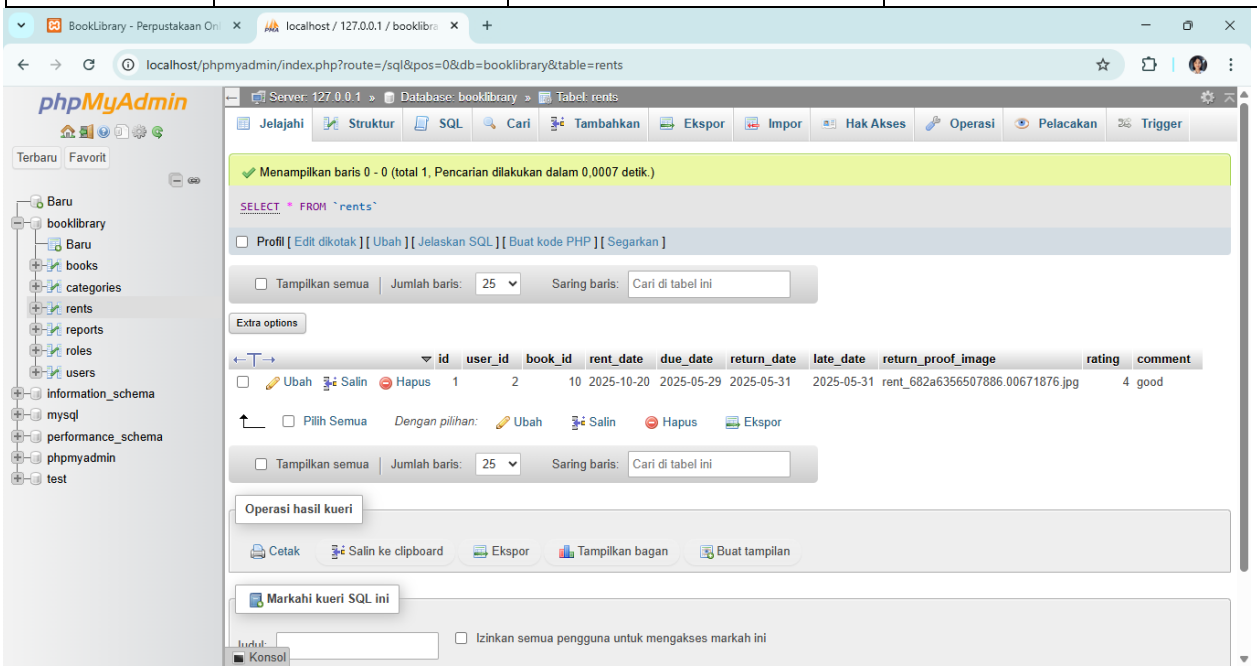
Column Name	Data Type	Constraints	Description
categories_ID	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each category.
name	VARCHAR(50)	NOT NULL	Name of the category.
description	TEXT		Description of the category.



Rent Table

Column Name	Data Type	Constraints	Description
rents_id	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each rental.
user_id	INT(11)	NOT NULL, FOREIGN KEY REFERENCES User(user_id)	Reference to the user who rented the book.
book_id	INT(11)	NOT NULL, FOREIGN KEY REFERENCES Book(books_id)	Reference to the book being rented.
rent_date	DATE	NOT NULL	Date when the book was rented.
due_date	DATE	NOT NULL	Date when the book is due to be returned.
return_date	DATE		Actual date when the book was returned (NULL if not yet returned).

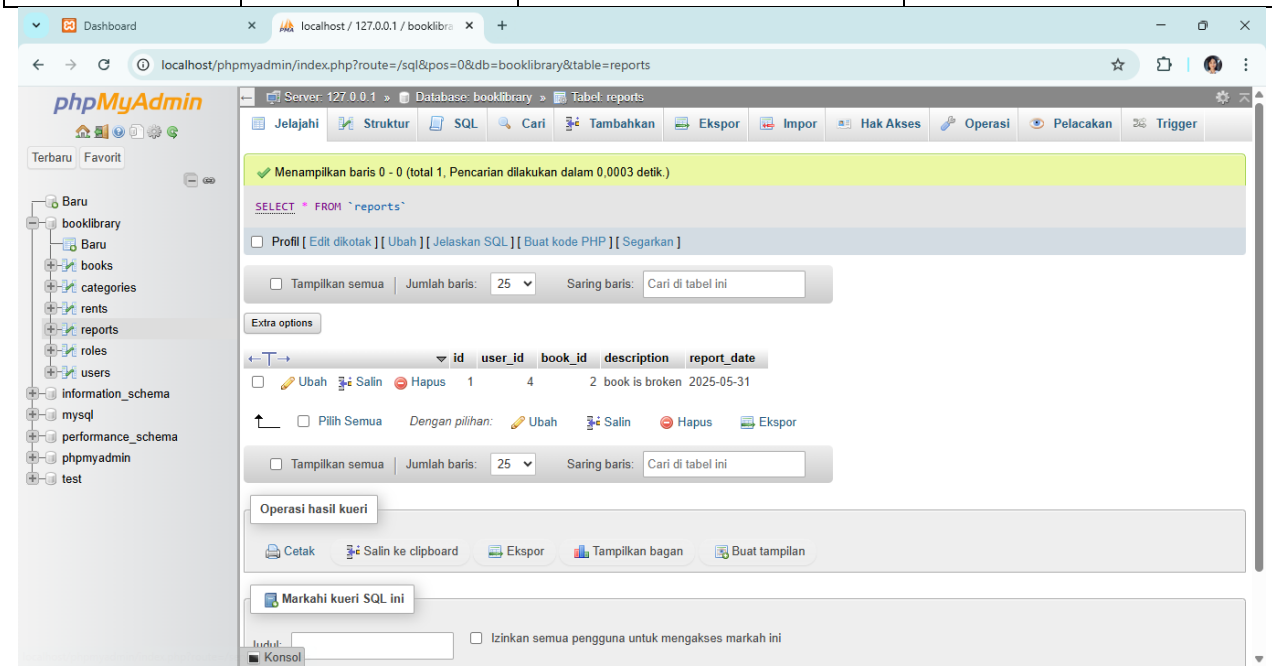
late_date	INT(11)		Number of days the book is overdue (if applicable).
return_proof_image	VARCHAR(255)		URL or path to the image of return proof (if required).



Report Table

Column Name	Data Type	Constraints	Description
reports_id	INT(11)	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each report.
user_id	INT(11)	NOT NULL, FOREIGN KEY REFERENCES User(user_id)	Reference to the user who submitted the report.
book_id	INT(11)	NOT NULL, FOREIGN KEY REFERENCES Book(books_id)	Reference to the book being reported.
description	TEXT	NOT NULL	Detailed description of the report.

report_date	TIMESTAMP	NOT NULL, DEFAULT CURRENT_TIMESTAMP	Date and time when the report was submitted.
rating	INT(1)		User rating of the book (1-5 stars).
report_proof_image	VARCHAR(255)		URL or path to any proof image for the report.
comment	TEXT		Additional comments by staff or administrators.



Entity Relationships

1. User to Role: Many-to-One (Each user has one role, each role can be assigned to many users)
2. User to Rent: One-to-Many (Each user can rent many books, each rental belongs to one user)
3. User to Reports: One-to-Many (Each user can submit many reports, each report is submitted by one user)
4. Book to Rent: One-to-Many (Each book can be rented many times, each rental involves one book)

5. Book to Reports: One-to-Many (Each book can have many reports, each report is about one book)
6. Book to Category: Many-to-One (Each book belongs to one category, each category can have many books)