UNIVERSITY OF BUEA

P.O.BOX 63, BUEA SOUTHWEST
REGION CAMEROON.

REPUBLIC OF CAMEROON

PEACE-WORK-FATHERLAND

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

ONLINE BOOKSTORE MANAGEMENT SYSTEM REPORT

By

CHINGOU ZINOURAIN	**FE22A356**
OBIH DAVID CHIMAOBI	**FE22A421**
ENONGENE NGEME PETER. W	**FE22A372**

To

Course Instructure
DR. GLEN

Monday, June 2, 2025

#ZINOTECH

Abstract

Online Bookstore Management System

Purpose:
The system is designed to facilitate the management of an online bookstore, allowing users to browse, purchase, rent, and manage books digitally.

Functionalities:

- User registration and login with secure authentication
- Book browsing, searching, and categorization
- Add to cart, checkout, and rental options
- Admin dashboard for managing books and users (CRUD operations)
- User order history and rental tracking
- Google OAuth integration for secure third-party login

Technologies Used:

- **Frontend:** HTML, CSS (Bootstrap),
- **Backend:** PHP (Core PHP, Session Management)
- **Database:** MySQL (LibraryDB with users, books, loans, and orders tables)
- **Authentication:** PHP Sessions and Google OAuth 2.0
- **Tools:** XAMPP, Composer, Google API Client, VS Code

Key Outcomes:

- Developed a full-stack web application integrating PHP and MySQL
- Implemented secure login with password hashing and OAuth
- Enabled CRUD operations and real-time database interaction
- Created a functional, interactive interface for users and admins
- Demonstrated practical application of authentication, session control, and dynamic content rendering

Introduction

In the digital age, online platforms have transformed the way businesses operate and interact with customers. This project, titled **“Online Bookshop Management System”**, is a web-based application developed to simplify and modernize the management of book sales, borrowing, and inventory in a bookshop environment. The system provides a robust platform for administrators to manage book stock, monitor borrow and return activities, and keep customers informed of their obligations, while offering users a convenient interface to browse, purchase, and borrow books.

The primary motivation for developing this system was to address the inefficiencies associated with manual bookshop operations. Traditional systems often involve time-consuming processes for tracking stock levels, recording borrow details and reminding customers of return deadlines. This application leverages PHP and MySQL to provide a reliable, dynamic, and user-friendly system that automates these tasks and enhances the overall efficiency of bookshop operations.

Key features of the system include:

- Admin dashboard for inserting, updating, and deleting book records.
- Customer portal with secure login, registration, and personalized views.
- Borrow and return functionality with countdown timers.
- Automated notifications for overdue returns.
- Responsive and intuitive user interface for both admin and customers.

This report outlines the planning, design, implementation, and testing phases of the Online Bookshop Management System. Each component of the system has been carefully crafted to meet real-world business needs and offer a scalable solution for small to mid-sized bookshops.

3.1System Requirements

- This section outlines the **hardware**, **software**, and **functional requirements** needed to run and manage the Online Bookshop Management System effectively.
- **Software Requirements**
- The development and deployment of the system are based on the following software tools and platforms:

Software	Version/Description
Operating System	Windows 10 or later / Linux
Web Server	XAMPP (Apache, PHP, MySQL)
PHP	Version 8.0 or later
MySQL	Version 5.7 or later
Web Browser	Chrome / Firefox / Edge

Text Editor/IDE	VS Code / Sublime Text / Notepad++

3.2 Functional Requirements

- Admin login and session management.
- Add, update, delete books.
- Upload book cover images.
- Set and modify book return delays.
- Borrowing functionality for customers.
- Live countdowns monitor for borrowed books (admin view).
- Automatic customer notification after return deadlines.
- Real-time book stock updates.
- Secure customer registration, login, and logout.
- Purchase and return signal features for customers.

4. System Analysis

This section provides a detailed analysis of the problem the system addresses, the objectives of the system, and the overall approach taken in its design and development.

4.1 Problem Statement

Traditional bookshops or library systems often face challenges in managing their inventory, tracking borrowed books, and reminding customers about return deadlines. Manual systems are prone to human error, inefficiency, and lack of real-time monitoring. There is a growing need for a digital solution that automates these processes while being user-friendly and easily maintainable.

4.2 Objectives of the System

The main goals of the Online Bookshop Management System are:

- To digitize bookshop operations, including purchasing, borrowing, and return management.
- To provide an efficient admin panel for managing books, monitoring borrowed items, and updating book stock.
- To enhance customer experience by enabling online viewing, buying, and borrowing of books.
- To implement a live countdown system to track book return deadlines.
- To automatically notify customers when return deadlines expire.

.

5. System Design

System design is a critical phase in software development that translates requirements into a blueprint for building the system. This section outlines the architecture, functional components, and database schema of the Online Bookshop Management System.

5.1 System Architecture

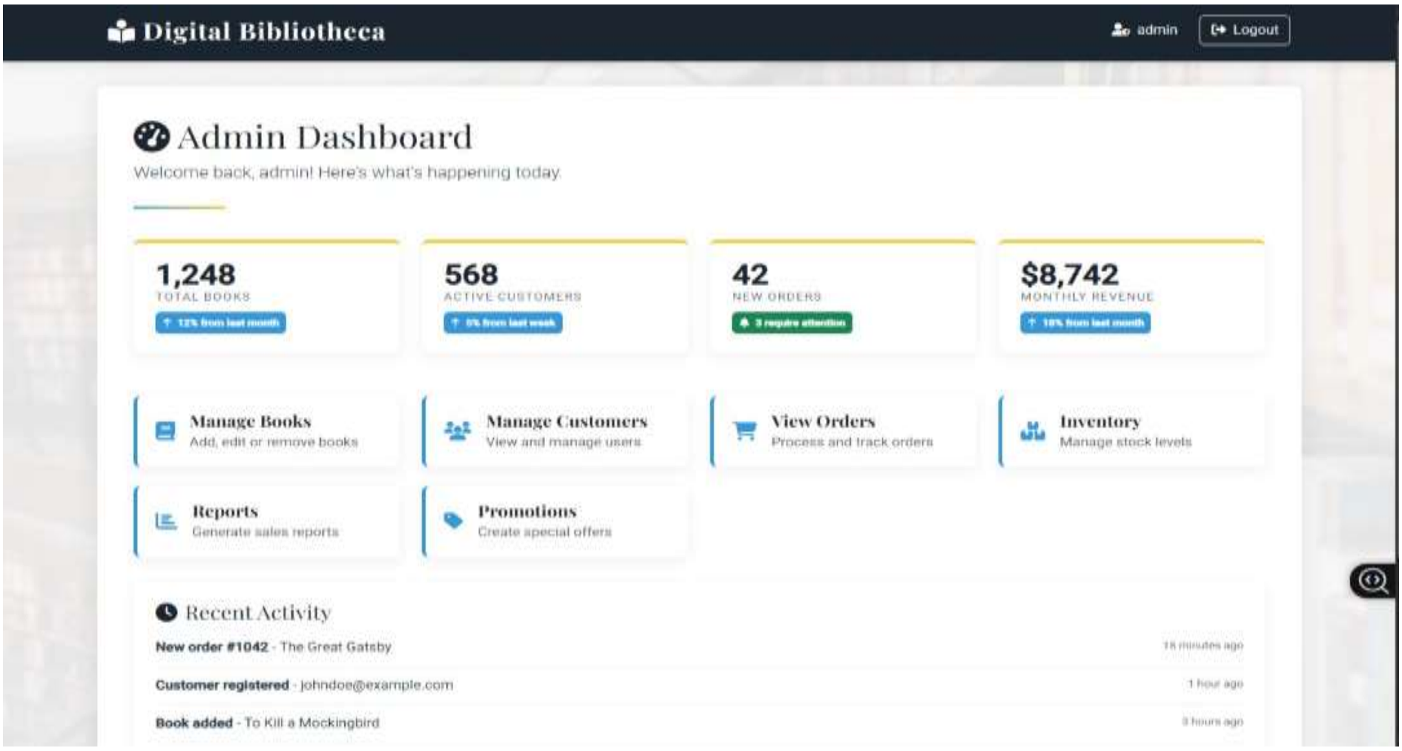
The system follows a **client-server architecture**, where:

- **Client Side** (Frontend): Implemented using HTML, and CSS (Bootstrap), providing interactive web pages for users (customers and admins).
- **Server Side** (Backend): Developed using PHP to handle business logic, session management, and database communication.
- **Database**: MySQL is used to store and manage data such as book records, customer information, transactions, and return tracking.

5.2 System Components

a) Admin Dashboard

- Secure login with session handling.
- Add, view, edit, and delete books.
- Upload book covers and manage stock.
- Set return delay (in days) per book.
- Monitor borrowed books with a live countdown timer.
- View which customers borrowed which books.
- Receive automatic return deadline alerts.



Admin Dashboard

Welcome back, admin! Here's what's happening today.

1,248 TOTAL BOOKS — ↑ 12% from last month

568 ACTIVE CUSTOMERS — ↑ 5% from last week

42 NEW ORDERS — 3 require attention

$8,742 MONTHLY REVENUE — ↑ 18% from last month

Recent Activity

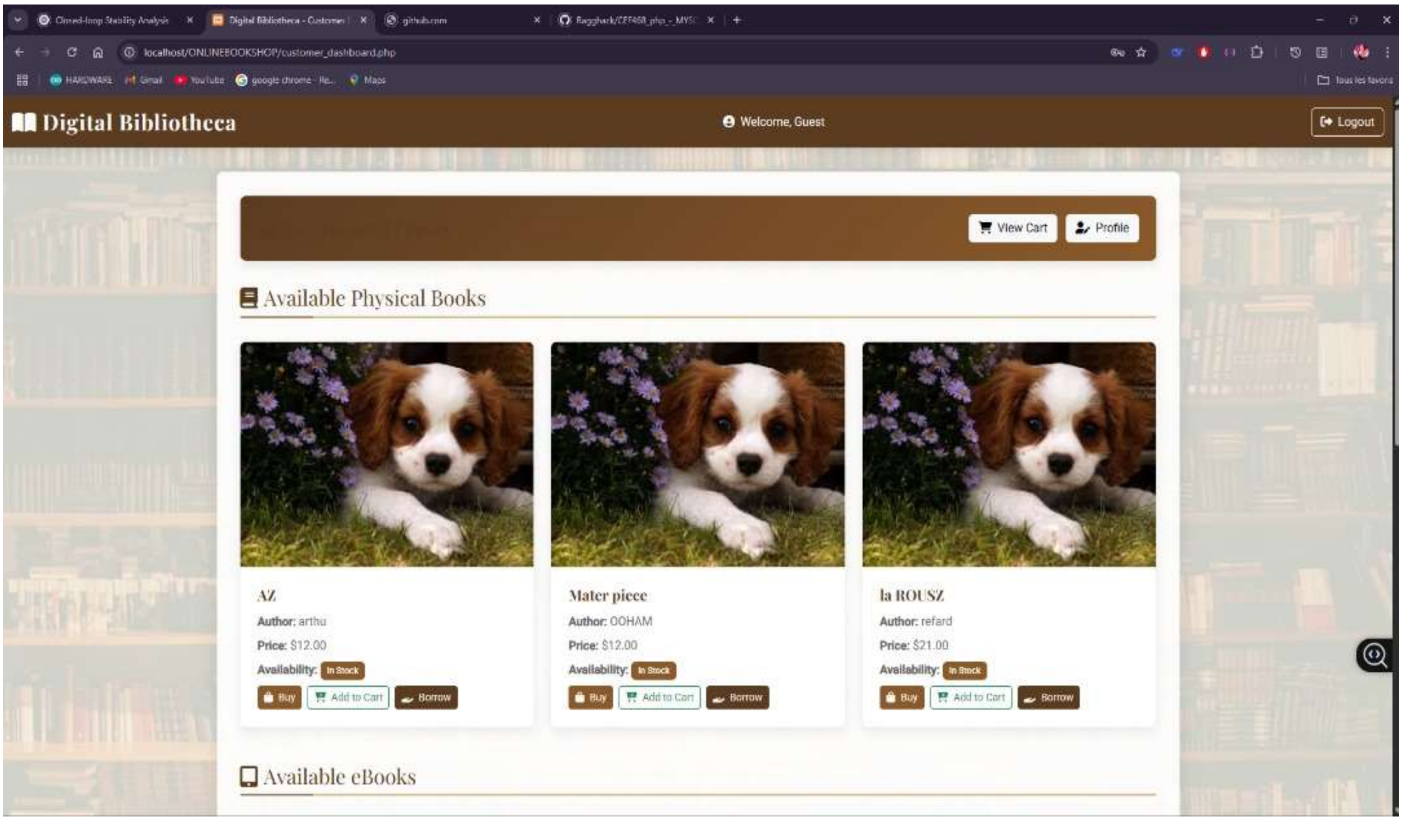
New order #1042 - The Great Gatsby — 18 minutes ago

Customer registered - johndoe@example.com — 1 hour ago

Book added - To Kill a Mockingbird — 3 hours ago

b) Customer Module

- Registration and login functionality.
- Browse available books with search and filtering.
- View book details and cover images.



Available Physical Books

AZ
Author: arthu
Price: $12.00
Availability: In Stock

Mater piece
Author: OOHAM
Price: $12.00
Availability: In Stock

la ROUSZ
Author: refard
Price: $21.00
Availability: In Stock

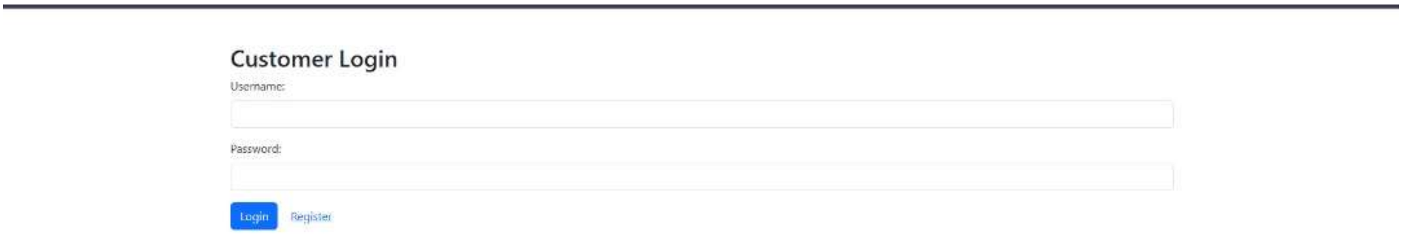
Available eBooks

- Buy or borrow books.
- Signal return of borrowed books.

#ZINOTECH

- Receive notifications if a borrowed book is overdue.

<u>Custom/Client login page</u>



Customer Login

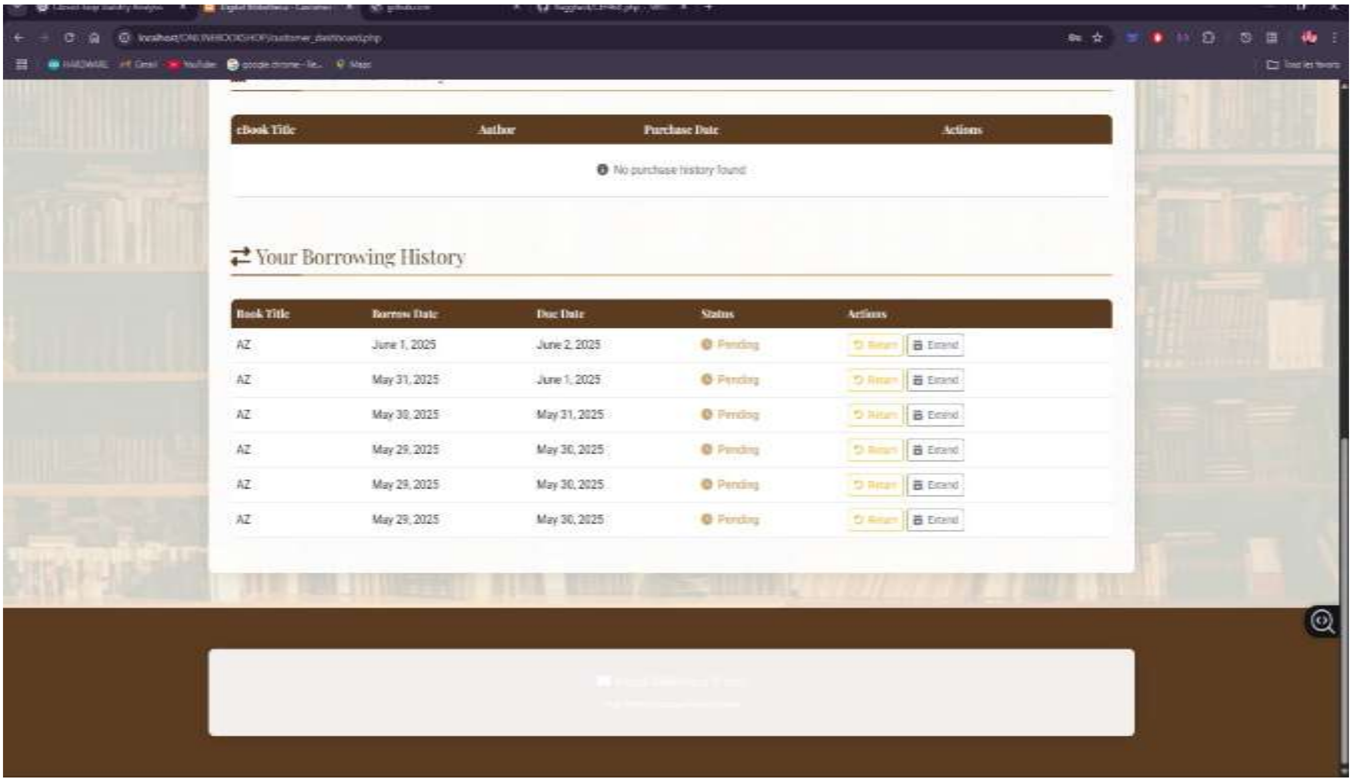
Username:

Password:

c) Notification and Timer

- A timer is associated with each borrowed book, counting down from the borrow date.
- Once the return time expires, the system flags the book as overdue.
- Customers are notified automatically via the interface.

<u>Customer Dashboard/Count Down after Borrowing</u> :
The below page is receive by the customer as an email. The customer opens the email and finds attach a link,



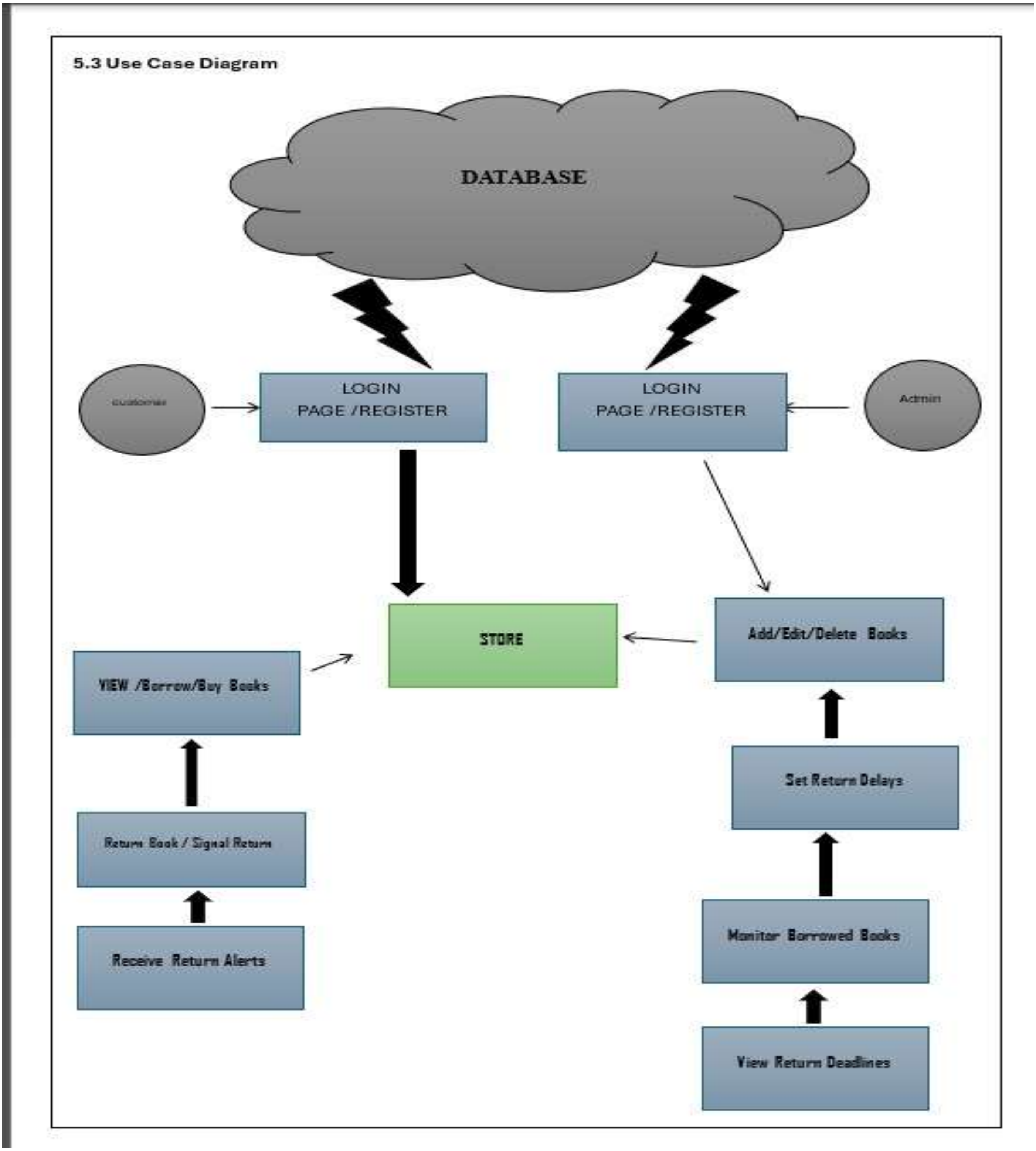
eBook Title	Author	Purchase Date	Actions
No purchase history found			

⇄ Your Borrowing History

Book Title	Borrow Date	Due Date	Status	Actions
AZ	June 1, 2025	June 2, 2025	Pending	Return Extend
AZ	May 31, 2025	June 1, 2025	Pending	Return Extend
AZ	May 30, 2025	May 31, 2025	Pending	Return Extend
AZ	May 29, 2025	May 30, 2025	Pending	Return Extend
AZ	May 29, 2025	May 30, 2025	Pending	Return Extend
AZ	May 29, 2025	May 30, 2025	Pending	Return Extend

5.3 Use Case Diagram

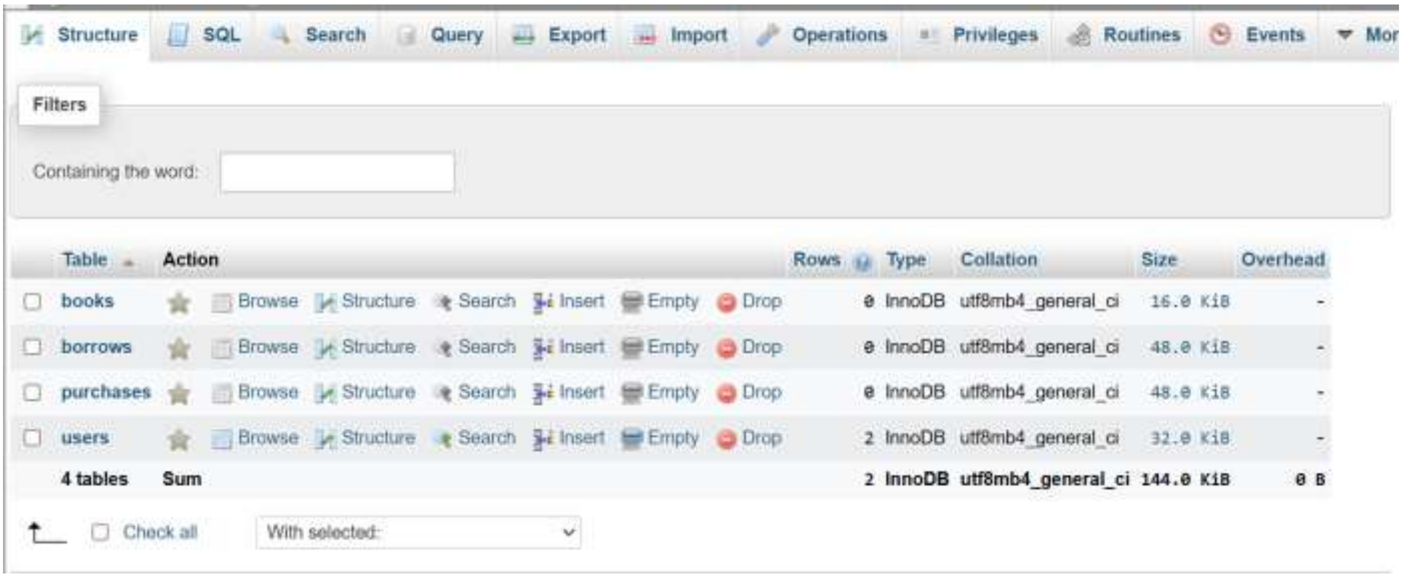
5.3 Use Case Diagram



5.4 Database Design

Tables and Key Fields:

1. **books**
    - id (Primary Key)
    - title, author, price, stock
    - return_delay (in days)
    - description, book_cover
2. **customers**
    - id (Primary Key)
    - name, email, password
3. **borrowed_books**
    - id (Primary Key)
    - book_id, customer_id
    - borrow_date, return_deadline
    - returned (boolean flag)
    - notified (boolean flag for overdue notification)
4. **purchases**
    - id (Primary Key)
    - customer_id, book_id
    - purchase_date



Containing the word:

	Table	Action							Rows	Type	Collation	Size	Overhead
□	books	★	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
□	borrows	★	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
□	purchases	★	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
□	users	★	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	utf8mb4_general_ci	32.0 KiB	-
	4 tables	Sum							2	InnoDB	utf8mb4_general_ci	144.0 KiB	0 B

6. Implementation

The implementation phase translates the system design into actual working software. This section discusses the development tools, programming languages, and logic used to implement the Online Bookshop Management System.

6.1 Tools and Technologies Used

Technology	Description
PHP	Server-side scripting language used for handling form submissions, database interactions, and backend logic.
MySQL	Relational database used to store books, customers, borrowing, and purchasing data.
HTML/CSS	Used to structure and style the web pages. Bootstrap is utilized for responsive design.
JavaScript	Enhances client-side interactivity, especially for timers and alerts.
XAMPP	Local development server for running Apache and MySQL on Windows.

6.2 Key Implementation Highlights

a) Admin Features

- The admin panel is secured using session-based login.
- Book insertion supports:
  - Dynamic return delay setting.
  - Book cover upload using $_FILES.
- Book management includes:
  - Listing all books.
  - Live countdown for borrowed books.
  - Real-time return status tracking.
  - Delete functionality to remove irrelevant books.
- SQL operations for insert, update, and delete are handled securely using mysqli prepared statements.

b) Customer Features

- Customers can register and log in securely.
- They can view a list of available books and see cover images, stock levels, and prices.
- Borrowed books are timestamped with a borrow_date, and the return_deadline is calculated using the book’s delay setting.
- A return timer is displayed in real time.
- Customers can manually signal when a book is returned.

c) Countdown Timer and Notifications

- Each borrowed book has a JavaScript-powered countdown timer that ticks down to the return-deadline.
- When the deadline passes, the system:
  - Flags the book as overdue in the backend.
  - Displays a warning to the customer in red.
  - Optionally logs the notification as sent to prevent repeated alerts.

6.4 Security Considerations

Security is a critical aspect of the Online Bookshop Management System to protect sensitive user data and ensure the integrity of transactions. Several measures were implemented throughout the development process to safeguard the system against common vulnerabilities:

6.4.1 User Authentication and Session Management

- **Secure Login:** Both admin and customer areas require login credentials to access restricted functionalities. User credentials are verified against stored database records.
- **Session Handling:** PHP sessions are used to maintain user login states securely. Sessions are started at the beginning of each page and validated to prevent unauthorized access.
- **Session Timeout:** Sessions can be configured to expire after a period of inactivity, reducing the risk of session hijacking.

6.4.2 Input Validation and Sanitization

- **Form Data Validation:** All user inputs from forms (e.g., login, book insertion, purchases) are validated both client-side and server-side to ensure correctness and prevent malformed data.
- **SQL Injection Prevention:** Prepared statements with parameterized queries (mysqli with bind_param) are used extensively to prevent SQL injection attacks.
- **Cross-Site Scripting (XSS) Prevention:** User-supplied data displayed back to the page is sanitized using PHP’s htmlspecialchars() to prevent injection of malicious scripts.

6.4.3 Access Control

- **Role-based Access:** Different access levels are enforced. Admin users have privileges to add, edit, or delete books and monitor borrowings, whereas customers can only view and interact with their own data.
- **Page Redirection:** Unauthorized users attempting to access restricted pages are redirected to the login page to prevent unauthorized data access.

6.4.4 Data Protection

- **Password Storage:** Although not detailed in this project, best practices recommend storing passwords using strong hashing algorithms (e.g., bcrypt) to protect user credentials.

- **Secure Connections:** Deployment should be done over HTTPS to encrypt data in transit, safeguarding login credentials and sensitive transactions.

6.4.5 Error Handling

- **Controlled Error Messages:** The system avoids revealing sensitive information (like database structure or server paths) in error messages, providing user-friendly notifications instead.
- **Exception Handling:** Errors such as failed database connections or invalid queries are properly caught and managed to maintain system stability.