

**University of Tabuk**  
**College of Computers & Information Technology**  
**Department of Computer Science**  
**Advanced Web Technology (CIT1303)**  
**University Library Management System**

**Final PHP Project**

**Group Members**

شذى علي البلوي  
رنيم حمدان رفاده  
اماني محمد الحويطي  
غدير عواد البلوي  
منيرة سعد الجهنبي

**Project Description**

We created a web application called the "University Library Management System". This system handles the core operations of a library, including inventory management, user authentication, and book lending. It allows an Admin to manage the entire book catalog, track current loans, and manage all user accounts (students and staff). Students can log in to view their active loans and check due dates. Non-logged-in Guests can search and view the public catalog for book availability. The application is built using a secure PHP backend and a responsive front-end design.

## Database Structure

We used MySQL to store all system data. There are three essential tables that link together using Foreign Keys to maintain data integrity:

- **users Table:** Stores student and admin information, including ID, username, password (hashed), and user\_type.
- **books Table:** Holds the library inventory, including book titles, authors, ISBN, total\_copies, and available\_copies.
- **loans Table:** Tracks all lending transactions, linking a specific user\_id to a book\_id, along with loan\_date, due\_date, and return\_date.

The screenshot shows the phpMyAdmin interface for the 'library\_db' database. The left sidebar lists databases: New, cit\_feedback\_db, information\_schema, library\_db (selected), mysql, performance\_schema, phpmyadmin, and test. The main area shows the 'Structure' tab for the 'library\_db' database. A table named 'New' is being created with 4 columns. The 'books', 'loans', and 'users' tables are listed under the database. The 'books' table has 2 rows, 'loans' has 1 row, and 'users' has 2 rows. The 'Structure' tab also includes options for Browse, Structure, Search, Insert, Empty, Drop, and Routines.

Table	Action	Rows	Type	Collation	Size	Overhead
books	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 Kib	-
loans	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_general_ci	48.0 Kib	-
users	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	48.0 Kib	-
3 tables	Sum	5	InnoDB	utf8mb4_general_ci	112.0 Kib	0 B

**Figure 1: Database structure showing all tables and relationships.**

The screenshot shows the phpMyAdmin interface for the 'library\_db' database. The left sidebar lists databases like 'New', 'ct\_feedback\_db', 'information\_schema', 'library\_db', 'mysql', 'performance\_schema', 'phpmyadmin', and 'test'. The 'users' table is selected under the 'library\_db' section. The main area displays the table structure with columns: id, username, password, user\_type, name, and email. Two rows are present: one for 'admin' (id 2) and one for 'ahmed10' (id 5). Below the table, there are buttons for Edit, Copy, Delete, and Export. A SQL query at the top shows a SELECT statement for the 'users' table. The bottom features a 'Query results operations' panel with options like Print, Copy to clipboard, Export, Display chart, and Create view.

**Figure 2: Structure view of the users table.**

The screenshot shows the phpMyAdmin interface for the 'library\_db' database. The left sidebar lists databases like 'New', 'ct\_feedback\_db', 'information\_schema', 'library\_db', 'mysql', 'performance\_schema', 'phpmyadmin', and 'test'. The 'books' table is selected under the 'library\_db' section. The main area displays the table structure with columns: book\_id, isbn, title, author, publication\_year, category, total\_copies, and available\_copies. Two rows are present: one for 'PHP Basics' (book\_id 1) and one for 'Clean Code' (book\_id 3). Below the table, there are buttons for Edit, Copy, Delete, and Export. A SQL query at the top shows a SELECT statement for the 'books' table. The bottom features a 'Query results operations' panel with options like Print, Copy to clipboard, Export, Display chart, and Create view.

**Figure 3: Structure view of the books table.**

The screenshot shows the phpMyAdmin interface for a MySQL database named 'library\_db'. The 'loans' table is selected. The top navigation bar includes links for 'Server 127.0.0.1', 'Database library\_db', and 'Table loans'. Below the navigation are tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers'. A message bar at the top indicates 'Showing rows 0 - 0 (1 total, Query took 0.0003 seconds)'. The SQL query shown is 'SELECT \* FROM `loans`'. The table structure is displayed with columns: loan\_id, book\_id, user\_id, loan\_date, due\_date, and return\_date. A single row is present with values: 3, 3, 5, 2025-11-26, 2025-12-03, and NULL. Below the table are buttons for 'Edit', 'Copy', 'Delete', 'Check all', and 'With selected'. The bottom section contains 'Query results operations' with buttons for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. There is also a 'Bookmark this SQL query' section with a 'Label' input field and a checkbox for 'Let every user access this bookmark'. The bottom right corner shows system status: 4:29 AM, ENG, WiFi, battery level, and the date 11/26/2025.

*Figure 4: Structure view of the loans table.*

## System Pages and Features

The system is split into protected portals and public access areas.

### Public Access

The system uses the Login Page to authenticate users and the Public Catalog Page (catalog.php) to allow guests to view the entire inventory.

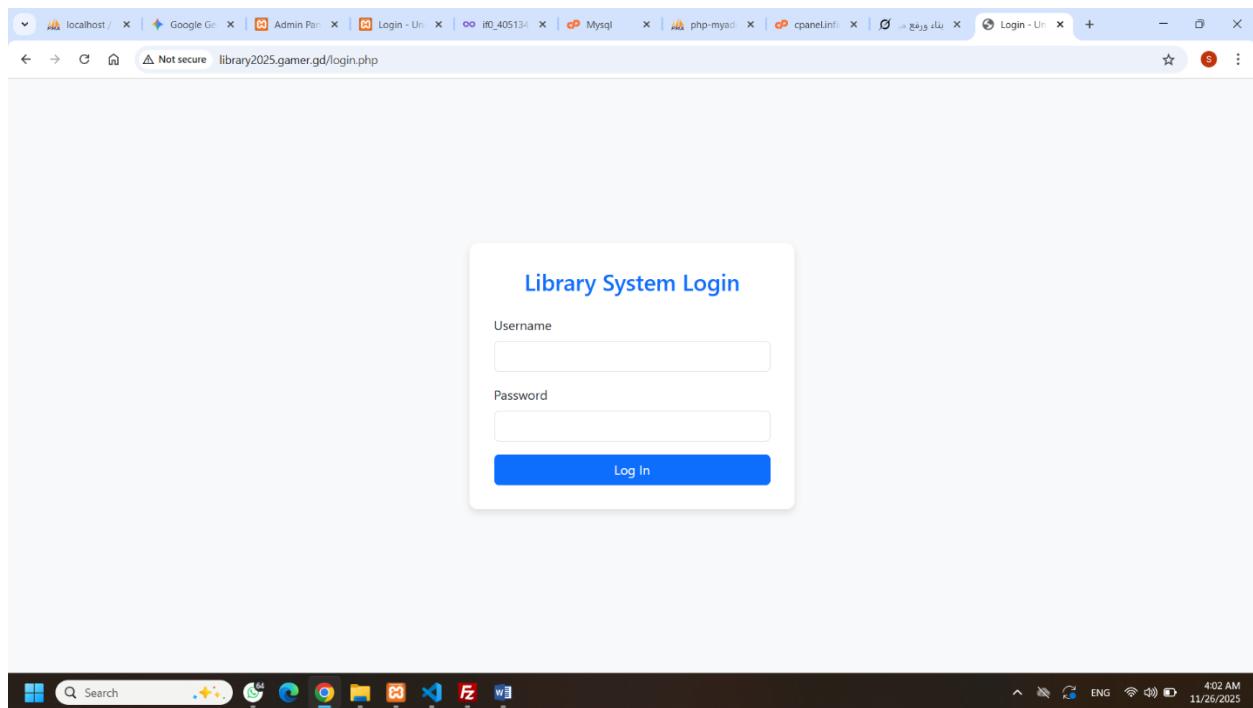


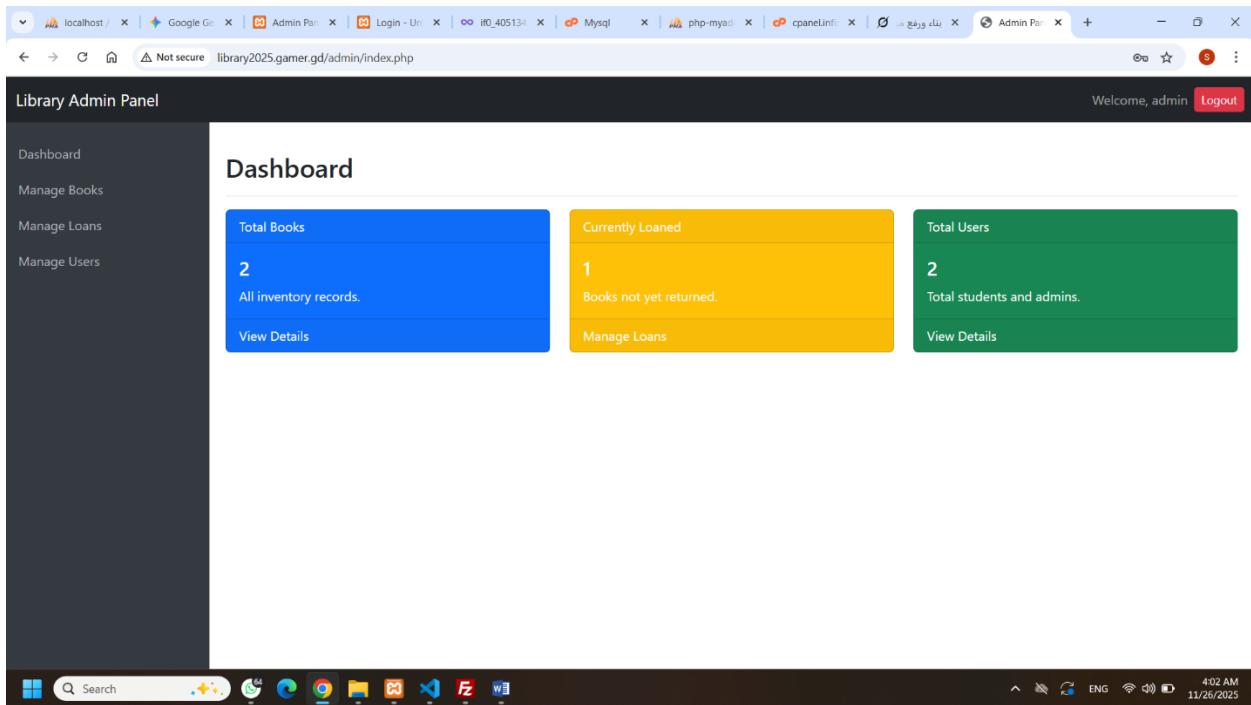
Figure 5: Login page for students and admin.

Figure 6: Public Catalog page showing all books and their availability (Guest View).

## Admin Portal

The Admin controls all CRUD operations:

- Admin Dashboard: Shows key statistics (Total Books, Current Loans, Total Users).
- Manage Books: Allows the admin to Add (C), View (R), Edit (U), and Delete (D) book records.
- Manage Loans: Allows the admin to perform Check Out (C) and Check In (U) transactions.
- Manage Users: Allows the admin to Add (C), View (R), Edit (U), and Delete (D) user accounts.



*Figure 7: Admin Dashboard with summary statistics.*

Library Admin Panel

Welcome, admin [Logout](#)

Manage Books

Add New Book

ID	Title	Author	ISBN	Total Copies	Available	Actions
3	Clean Code	Robert Martin	978-0134764324	8	7	<a href="#">Edit</a> <a href="#">Delete</a>
1	PHP Basics	SMITH	978-0134764125	5	3	<a href="#">Edit</a> <a href="#">Delete</a>

Figure 8: Manage Books page showing CRUD functionality.

Library Admin Panel

Welcome, admin [Logout](#)

Manage Loans

Check Out Book

Active Loans

Loan ID	Student	Book Title	Loan Date	Due Date	Status	Action
3	Ahmed	Clean Code	2025-11-26	2025-12-03	On Time	<a href="#">Check In</a>

Figure 9: Manage Loans page showing active transactions and Check In option.

The screenshot shows a web browser window titled "library2025.gamer.gd/admin/manage\_users.php". The title bar indicates "Not secure". The page header says "Welcome, admin" and has a "Logout" button. On the left, a sidebar menu includes "Dashboard", "Manage Books", "Manage Loans", and "Manage Users". The main content area is titled "Manage Users" and features a "Add New User" button. A table lists two users:

ID	Name	Username	Email	User Type	Actions
5	Ahmed	ahmed10	ahmed@gmail.com	Student	<a href="#">Edit</a> <a href="#">Delete</a>
2	System Administrator	admin	admin@library.com	Admin	<a href="#">Edit</a> <a href="#">Delete</a>

The bottom of the screen shows a Windows taskbar with various icons and the system tray indicating the date and time as 11/26/2025.

Figure 10: Manage Users page showing user accounts and management tools.

## Student Portal

- **Student Portal:** Allows the student to View (R) their active loans, check due dates, and see if any books are overdue.

The screenshot shows a web browser window with a tab bar containing various application icons like Google Chrome, Admin Panel, Login - Un, iO\_405134, Mysql, php-myadmin, cpanelInfo, and Student Po. The main content area has a blue header bar with 'Student Library Portal' on the left, 'Welcome, ahmed10' in the center, and 'View Catalog' and 'Logout' buttons on the right. Below the header is a section titled 'My Library Services' with a sub-section 'My Active Loans (Books Currently Checked Out)'. This section contains a table with the following data:

Title	Author	Loan Date	Due Date	Status
Clean Code	Robert Martin	2025-11-26	2025-12-03	On Time

The taskbar at the bottom of the screen shows several pinned icons and the system clock indicating 4:14 AM on 11/26/2025.

Figure 11: Student Portal showing the list of active loans.

## CRUD Operations Summary

The system implements complete Create, Read, Update, and Delete operations:

- **Create (C):** Adding new books and users (manage\_books.php, manage\_users.php), and creating a new loan transaction (Check Out in manage\_loans.php).
- **Read (R):** Displaying all data in lists and reports (manage\_books.php, manage\_loans.php), and showing user-specific data (My Active Loans).
- **Update (U):** Editing book and user details, and performing a Check In transaction (manage\_loans.php) which updates the loan status and the book's availability count.
- **Delete (D):** Removing book records and user accounts (manage\_books.php, manage\_users.php).

## Challenges and Solutions

During development, we successfully overcame critical challenges related to security and database integrity:

1. **Authorization:** We ensured that only users with admin status could access the /admin/ folder by checking the PHP Session (`$_SESSION['user_type']`) on every management page.
2. **Referential Integrity (Foreign Keys):** Deleting a user who had active loans, or deleting a book that was currently checked out, would cause a foreign key error. We solved this by implementing checks in the Delete logic to prevent the action if there is an active dependency (i.e., the user must return the book first).
3. **Cross-Page Design:** We ensured a consistent design across the Admin portal by using `layout_admin_top.php` and `layout_admin_bottom.php` as template files.

## Conclusion

The result is a professional and fully operational Library Management System. We applied core concepts learned in the Advanced Web Technology course, including secure database interaction (PDO), authorization control (PHP Sessions), robust data validation, and responsive design (Bootstrap). The project demonstrates a strong understanding of web application development principles and is ready for institutional use.