SABARAGAMUWA UNIVERSITY OF SRI LANKA

LIBRARY MANAGEMENT SYSTEM REPORT

Project Report

Bachelor Of Science
In
Physical Sciences & Technology

Prepared by,

21APP5704-Fahadh Muhammadh

Prepared for,

Computer Lab Module

Faculty of Applied Sciences

Department of Physical Science Technology

Contents

1. Problem Statement and Target Users

Small libraries commonly rely on manual, paper-based record keeping to track books, lending activity and overdue returns. This approach often results in lost records, difficulty identifying overdue books, and time-consuming searches for member or book details. As collections grow, the manual system becomes inefficient, error-prone and difficult to maintain.

To address these issues, the Library Management System (LMS) provides a **desktop software solution** that centralizes book management, member tracking and borrowing/return operations. The system streamlines workflows, reduces errors and offers a user-friendly interface suitable for small institutions.

The primary target users include:

- **Librarians** who manage daily operations such as issuing books, updating records and searching the catalog.
- **Library administrators** responsible for maintaining system access, adding staff users and overseeing library activities.
- **Students or community members** who use the library services and require accurate borrowing management.

The application is designed as a **Windows Forms desktop client**, allowing it to run on existing computers without requiring internet connectivity or web hosting. This makes it cost-effective and ideal for small-scale organizations such as school libraries, local community reading rooms, and small academic departments.

2. Application Design Overview

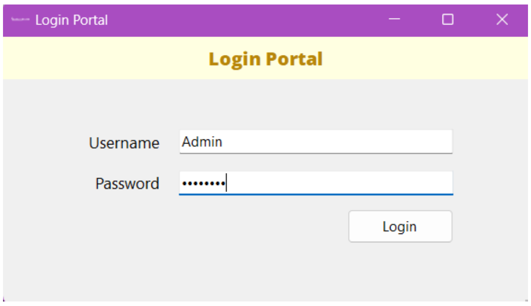
The design of the LMS emphasizes simplicity, clarity and ease of use. The system begins with a login interface to ensure secure access, followed by the main dashboard where users interact with books, members and borrowing functions.

2.1. User Interface Structure

The UI is built using WinForms with a clean and minimal layout. The LoginForm includes:

- Username and password fields
- A masked password input
- A login button and header label
- Simple alignment using anchor points for resizing

Once authenticated, the user proceeds to the main dashboard.



Username Admin

Password ••••••••

Figure 1 : Login Portal

The dashboard (in future development) will include:

- Book management section
- Member management section
- Borrow/Return module
- Summary boxes showing library statistics

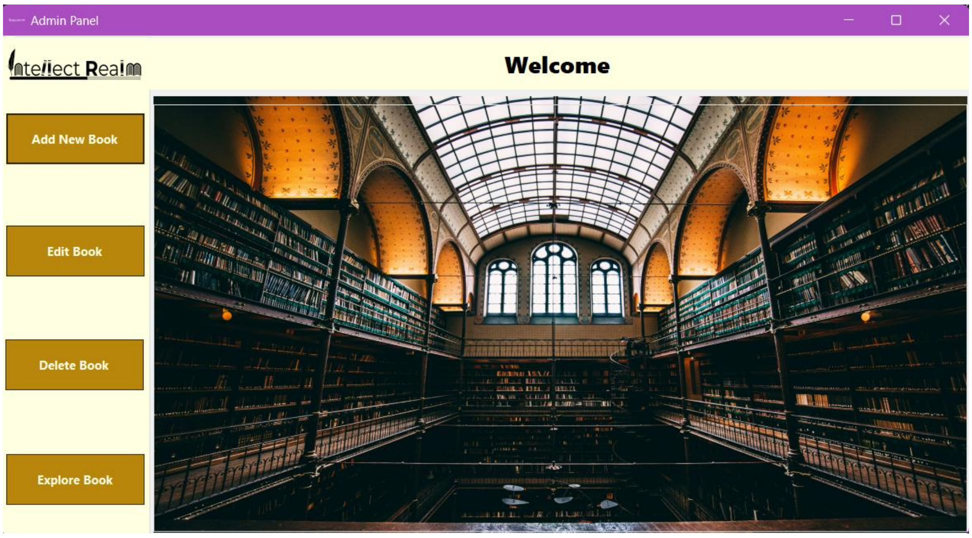
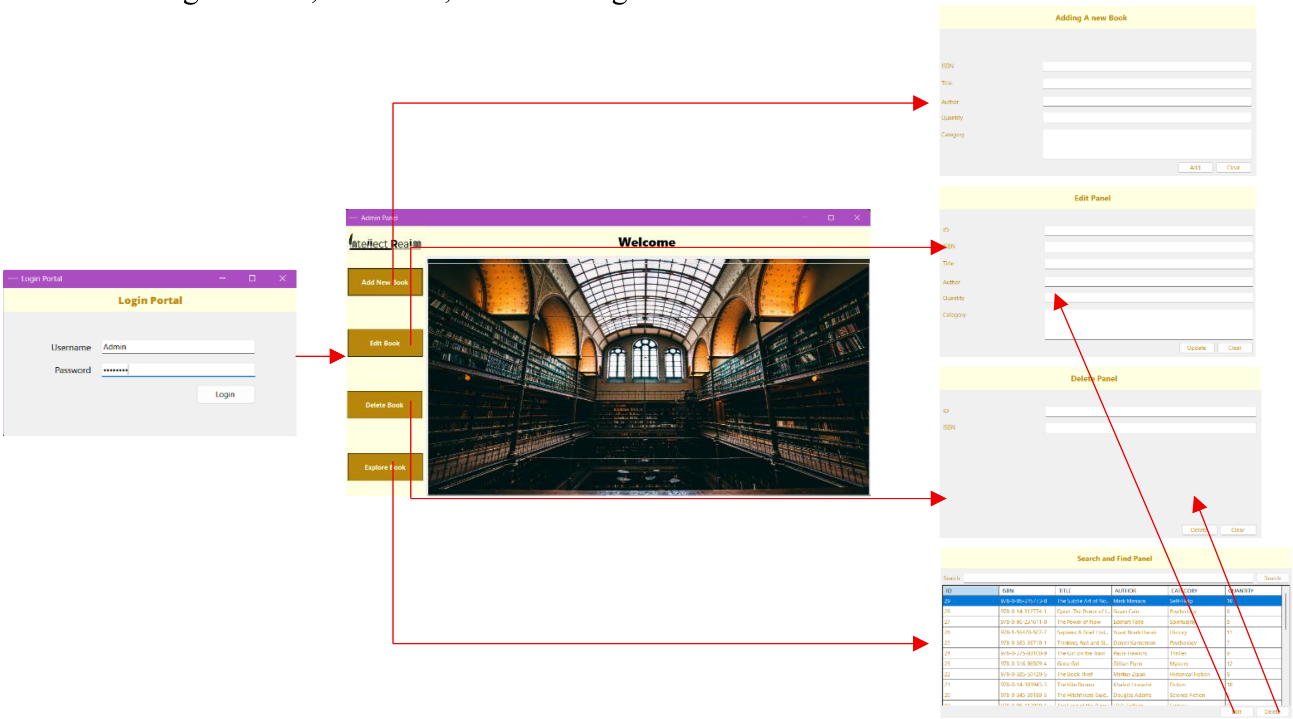


Figure 2 : Dashboard

2.2. Workflow Overview

The high-level workflow consists of the following sequence:

- User opens the application
- Login form loads
- System validates credentials
- Successful login grants access to dashboard
- User manages books, members, or borrowing activities



ID	ISBN	TITLE	AUTHOR	CATEGORY	QUANTITY
29	978-0-06-245773-8	The Subtle Art of No...	Mark Manson	Self-Help	10
28	978-0-14-312774-1	Quiet: The Power of I...	Susan Cain	Psychology	6
27	978-0-06-231611-0	The Power of Now	Eckhart Tolle	Spirituality	8
26	978-1-59420-507-7	Sapiens: A Brief Hist...	Yuval Noah Harari	History	11
25	978-0-385-33710-1	Thinking, Fast and Sl...	Daniel Kahneman	Psychology	7
24	978-0-375-03100-9	The Girl on the Train	Paula Hawkins	Thriller	9
23	978-0-316-06509-4	Gone Girl	Gillian Flynn	Mystery	12
22	978-0-385-50420-5	The Book Thief	Markus Zusak	Historical Fiction	8
21	978-0-14-303943-3	The Kite Runner	Khaled Hosseini	Fiction	10
20	978-0-345-39180-3	The Hitchhikers Guid...	Douglas Adams	Science Fiction	6

Figure 3 : workflow diagram

The goal of this design is to keep workflows linear and intuitive for non-technical users.

3. Implementation Details and Challenges

The current implementation focuses on establishing the foundational components of the system. The project is structured as a Windows Forms application running on **.NET 8**, ensuring compatibility with modern Windows systems.

3.1. Key Implementation Areas

3.1.1. Authentication

- Login validation is handled within the LoginForm.
- Credentials are currently hard-coded (Admin / Fhd@2948) for prototype purposes.
- A DialogResult.OK return value confirms authentication success.

3.1.2. User Interface

- Built using WinForms Designer tools.
- Password masking applied using PasswordChar = '•'.
- Anchoring ensures UI stability under resizing.

3.1.3. Resource Management

- Custom application icon embedded in .resx.
- Standard disposal pattern implemented via Designer-generated code.

3.2. Challenges Encountered and Solutions

3.2.1. Implementing a basic authentication mechanism

The challenge was to create a simple yet functional security checkpoint. This was solved using a modal login form that checks credentials on button click and returns an authenticated result.

3.2.2. Ensuring a clear and user-friendly UI

Keeping the layout centered and properly anchored solved issues related to alignment and resizing.

3.2.3. Password security and masking

Although the system uses a prototype-level hard-coded password, visual protection was implemented by masking the input field using the password character property.

3.2.4. Embedding resources without external dependencies

Icons and images were embedded directly into .resx files to simplify deployment.

3.2.5. Lack of persistent data storage

Currently, no database is integrated. This limits real operational use but is acceptable at the prototype phase. It is identified as a key improvement area for future iterations.

4. Conclusion and Future Improvements

The current version of the Library Management System successfully demonstrates the foundation of a desktop-based library solution. It provides a structured login mechanism, a clean interface, and a workflow suitable for small-scale libraries transitioning from manual systems.

However, the prototype lacks persistent storage and secure authentication, making it ideal as an academic assignment or portfolio project but not yet ready for production use. The underlying architecture is well-suited for expansion, and future enhancements will significantly improve its capabilities.

Recommended Future Enhancements

- **Database Integration (SQLite or SQL Server)**
  Introduce persistent storage for books, members and transaction records.
- **Secure Authentication System**
  Replace hard-coded credentials with hashed, database-stored passwords.
- **Role-Based Access Control**
  Differentiate between Admin, Librarian and Member privileges.
- **Full CRUD Modules**
  Add interfaces to add, update, search and delete books and members.
- **Borrow/Return Module**
  Implement issue tracking, overdue detection and fine calculation.
- **Reporting Tools**
  Export to PDF/CSV and view activity logs.
- **Enhanced UX**
  Improve accessibility, add themes, validate inputs, and provide error-free navigation.