**Library Management System**

# Project Overview

The Library Management System is a software application designed to help manage libraries efficiently. Its primary purpose is to manage tasks like tracking books, managing members, facilitating book lending, and overseeing administrative tasks. The system aims to replace traditional manual methods with a digital solution, making the library's day-to-day operations more efficient, less error-prone, and easier to manage. It promises improved organization of books, better tracking of loans and returns, and a straightforward way for library staff to handle administrative duties.

# Objective

The main objective of the Library Management System is to provide a user-friendly, reliable, and efficient way to manage library resources.

Streamlining Book Management: Organizing and maintaining the book inventory with ease.

Facilitating Member Management: Enrolling and managing members efficiently.

Simplifying Lending and Returns: Making the process of lending books and tracking returns smooth and automated.

Enhancing Administrative Control: Providing library administrators with tools to manage library operations effectively.

# System Requirements

## Functional Requirements

* **Secure Login:** Admin authentication to access the system.
* **Book Management:** Add, update, delete, and search for books, view their details like ISBN, title, author, genre, quantity, availability, and borrowed status.
* **Member Management:** Enroll new members, remove existing members, and search the member database.
* **Admin Management:** Add new admins, remove existing admins, and manage their details.
* **Lending Management:** Monitor and manage the lending process, including viewing borrowed books, assigning books to members, and accepting returns.

## Non-Functional Requirements

* **Performance:** Efficient handling of multiple simultaneous users and large data sets.
* **Usability:** Intuitive navigation and clear, responsive interfaces.

# System Design & Architecture

## Architecture Overview

The Library Management System is structured into several key modules, each dedicated to a specific aspect of library management: Member Management, Lending Management, Admin Management, and Book Management. These modules work together to provide a comprehensive, integrated system. The user interface is designed using ttkbootstrap, ensuring a responsive and visually appealing experience.

**Initial Launch Operation**

The Library Management System is initialized through the "launch.py" file. Upon execution, this file sets up the entire application environment. It performs the following operations:

**Database Creation:** Automatically creates and populates the database with default admin. This includes setting up tables for members, books, admins, and transactions, ensuring the system is ready with initial data for use.

**Frame Initialization:** Constructs all necessary frames for the application, setting up the structure for the various modules such as Book Management, Member Management, Lending Management, and Admin Management.

**GUI Activation:** Brings up the main GUI window, presenting a user-friendly interface designed with **ttkbootstrap**. The interface is responsive and intuitive, making it easy for users to navigate through different sections of the system.

When the system is first launched, the user is greeted with a login screen that requires an administrator's credentials. These credentials are verified against the admin data stored in the system's database. The login feature ensures that only authorized users can access and manage the library system, maintaining security and integrity.

This initial launch operation is establishes the foundation for the entire Library Management System, ensuring that all components are correctly initialized and ready for use.

mobilya, ekran görüntüsü, kitap, iç mekan içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Member Management and Popup**

ekran görüntüsü, yazılım, bilgisayar simgesi, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu**Member Management:** This section begins with a default set of members. It allows users to add new members to the system by entering a unique student ID and name via an "Add Member" button. Existing members can be selected from the list and removed using the "Remove Selected" button.

**Member Management Popup:** When a member is double-clicked in the member management section, a popup window opens. This popup allows users to manage the lending records associated with the selected student. Books can be searched by their ISBN, title, author, or genre. Once a book is selected, it can be returned using the "Return Book" button, updating the student's borrowing record and the book's availability.

ekran görüntüsü, yazılım, bilgisayar simgesi, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Lending Management**

metin, yazılım, bilgisayar simgesi, işletim sistemi içeren bir resim

Açıklama otomatik olarak oluşturuldu**Lending Management:** This module displays the current loans, showing which students have borrowed which books. Users can search through the records by student ID, name, or book details (ISBN, title, author, genre). Selected records can be removed using the "Remove Selected" button. Double-clicking on a record redirects users to the Member Management Popup for further actions, such as returning the book.

**Admin Management**

ekran görüntüsü, metin, yazılım, bilgisayar simgesi içeren bir resim

Açıklama otomatik olarak oluşturuldu**Admin Management:** Here, users can view all current administrators in the system and search for them by name, email, or phone number. New admins can be added by filling in the required fields: name, email, phone, and password, and then clicking the "Add Admin" button. Existing admins can be selected and removed from the system using the "Remove Selected" button.

**Book Management and Popup**

metin, ekran görüntüsü, yazılım, bilgisayar simgesi içeren bir resim

Açıklama otomatik olarak oluşturuldu**Book Management:** This section displays a list of books available in the library's pre-established database. Users can add new books by entering details such as ISBN, title, author, genre, and quantity, then clicking the "Add Book" button. Books can be removed by selecting them and clicking the "Remove Selected" button.

**Book Management Popup:** Double-clicking on a book in the book management section opens a popup window. This window displays all registered students, allowing the user to lend the selected book to a chosen student using the "Lend Book" button. As books are lent out, their available quantity decreases. The popup facilitates searching students by ID or name to quickly assign books.

A screenshot of a computer

Description automatically generated

### Database Initialization and Simulation

The databaseFill.py script can be used in the initial setup and testing phase of the Library Management System. Its primary purpose is to efficiently populate the system with a substantial and varied dataset, simulating a realistic library environment. This process is essential for conducting comprehensive system tests and ensuring the application's readiness for deployment in real-world scenarios. It automatically generates a wide array of book entries and user profiles. This includes creating books with varied titles, authors, and genres, and registering users with unique identifiers, thereby providing a rich and diverse library catalog and member database.  
  
The database sets default admin credentials if there is no admin in the system as admin/admin. This feature is useful for initial system access however, it's crucial to highlight the importance of updating these default credentials to more secure and individualized ones as soon as the initial testing phase concludes.

metin, ekran görüntüsü, diyagram, devre içeren bir resim

Açıklama otomatik olarak oluşturuldu Flowchart

# metin, ekran görüntüsü, diyagram, tasarım içeren bir resim Açıklama otomatik olarak oluşturuldu Sequence Diagram

# Project Members

Cem Erdoğdu 2003193

Hilmi Mert Acar 2004285

Kaan Özcivan 2002098

# References

We have used ttkbootstrap, sqlite3, pillow, random, string, time, threading, libraries.