

# Library Management System (Kivy)

---

A modern, tabbed, multi-role library management UI built with [Kivy](#) in Python.

This project provides a clean, extensible interface for both **members** and **librarians** to interact with a library database.

This was the final project of database course

Phase 1 : semantic and logical design of database

Phase 2 : implementation using sql server and python(Kivy framework for UI)

## Features

- **Login Screen:**
  - Tabbed login for Librarian and Member roles.
- **Member Dashboard:**
  - **Search Books:** Search and view books in a table/chart format.
  - **Borrowed Books:** View currently borrowed books.
  - **Fines:** View outstanding fines.
- **Librarian Dashboard:**
  - **Search Books:** Search and view books.
  - **Search Members:** Search and view library members.
  - **User Management:** Add, edit, and remove users.
  - **Book Management:** Add, edit, and remove books and copies.
  - **Loan Management:** Edit loans, view all loans (with count), and manage fines.
  - **Reports:** View lists of all loans and fines.
- **Tabbed Navigation:**
  - Intuitive tabbed layout for quick access to all features.

---

## Screenshots

MyLibrary

Library System Login

Librarian Login

Member Login

Librarian Login

Username

Password

Login

MyLibrary

logout

Main

Dashboard

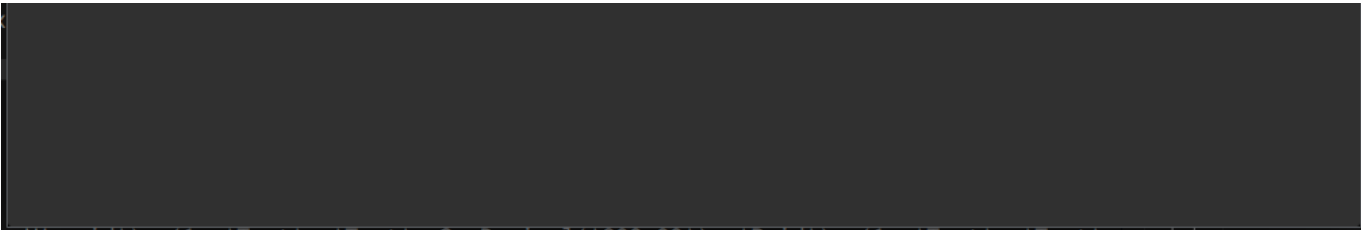
Reports

Search Books

Search books...

Search

| BookID | Title               | Publisher        | Year | Translator  | Genre     | Available |
|--------|---------------------|------------------|------|-------------|-----------|-----------|
| 1      | 1984                | Secker & Warburg | 1949 | None        | Dystopian | 3         |
| 2      | Pride and Prejudice | T. Egerton       | 1813 | None        | Romance   | 1         |
| 3      | Emma                | PublisherX       | 2020 | TranslatorX | Fiction   | 0         |
| 4      | Emma 2              | PublisherX       | 2020 | TranslatorX | Fiction   | 0         |



MyLibrary

logout

Main

Dashboard

Reports

Search Books

em

Search

| BookID | Title  | Publisher  | Year | Translator  | Genre   | Available |
|--------|--------|------------|------|-------------|---------|-----------|
| 3      | Emma   | PublisherX | 2020 | TranslatorX | Fiction | 0         |
| 4      | Emma 2 | PublisherX | 2020 | TranslatorX | Fiction | 0         |

MyLibrary

logout

Main

Dashboard

Reports

List Loans

List Fines

All Fines

TotalUnpaidFines : 1500.00 | TotalPaidFines : 800.00 | UnpaidFineCount : 2 | PaidFineCount : 1

| MemberID | Name  | LastName | LoanID | FineAmount | Status |
|----------|-------|----------|--------|------------|--------|
| 2        | Test2 | Test2    | 1      | 500.00     | Unpaid |
| 1        | Test  | Test     | 2      | 800.00     | Paid   |
| 1        | Test  | Test     | 3      | 1000.00    | Unpaid |

MyLibrary

logout

Main

Dashboard

Reports

Search Members

Add User

Edit User

Remove User

Add Book

Edit Book

Remove Book

Add Copy

Search Copies

Remove Copy

Edit User

MemberID

Set

Name

LastName

Password

Address

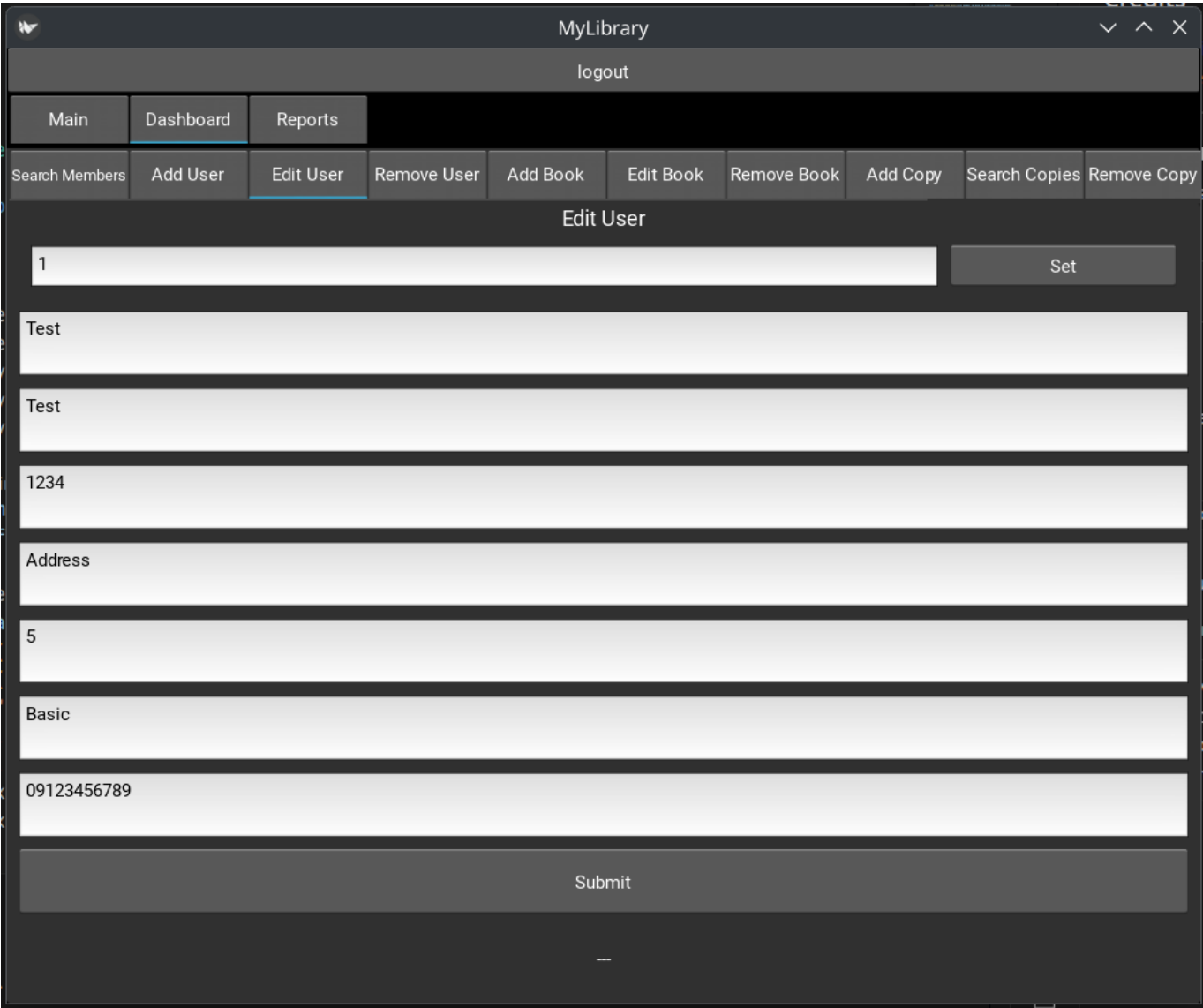
MaximumNumberOfBorrowedBooks

TypeOfMembership(VIP/Basic)

PhoneNumber

Submit

/



# Getting Started

## Prerequisites

- Python 3.8+
- [Kivy](#)
- [pyodbc](#) (for database connection)
- Microsoft SQL Server (or compatible, for backend)
- ODBC Driver 17 for SQL Server

## Installation

### 1. Clone the repository:

```
git clone https://github.com/yourusername/library-app-kivy.git
cd library-app-kivy
```

### 2. Install dependencies:

```
pip install kivy pyodbc
```

### 3. Set up the database:

- run

```
docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=Str0ngPassw0rd!' -p 1433:1433 --name sql1 -d mcr.microsoft.com/mssql/server:2022-latest
```

- Use the provided `code.sql` to create and initialize your SQL Server database.
- Update the connection string in `database.py` if needed.

### 4. Run the app:

```
python main.py
```

---

## Project Structure

### Phase 1 :

- `database.pdf` — Semantic and logical design of database

### Phase 2 :

- `main.py` — Main application logic and screen management.
- `database.py` — Database connection and query logic.
- `library.kv` — Kivy UI definitions (tabs, layouts, widgets).
- `code.sql` — SQL Server schema and stored procedures.

---

## Customization

- **UI:**  
Modify `library.kv` to change layouts, add widgets, or adjust styles.
- **Database:**  
Update `database.py` to match your schema or add new queries.
- **Logic:**  
Extend `main.py` to add new features or business rules.

---

## Troubleshooting

- **Clipboard Warnings:**  
If you see warnings about `xclip` or `xsel`, install them with:

```
sudo pacman -S xclip xsel
```

Or ignore if you don't need clipboard support.

- **Database Connection Errors:** Ensure your SQL Server is running, the database exists, and credentials are correct.
- **UI Not Updating:** Make sure your RecyclerView data is set as a list of dictionaries matching the viewclass properties.

## Team members:

<https://github.com/BehrazFS>

<https://github.com/RozhinaLatifi>