**PROJECT PROPOSAL**

**Library Management System**

**Group members**

1.MUHAMMAD ASIF NAEEM

F2024065492

2.Rizwan

F2022065234

3.Adnan Dab

F2022065302

4.Abdul Rafay

F2022065306

**1. Introduction:**

This project involves the development of an open-source, desktop-based **Library Management System** that allows users to manage books, issue/return transactions, and user records efficiently. The front-end of the application will be developed using the **Kinder GUI framework** in Python, while the back-end logic will also be implemented in Python.

**2. Objectives:**

* Design a clean and intuitive GUI using the Kinder framework
* Enable core library operations: Add books, issue books, return books, and view available books
* Manage user records and book inventory
* Implement persistent storage using JSON or SQLite
* Promote modular, reusable, and well-documented code for open-source contributions

**3. Tools & Technologies:**

* **Programming Language:** Python 3.x
* **Frontend Framework:** Kinder
* **Backend Logic:** Python modules
* **Data Storage:** SQLite or JSON
* **Version Control:** Git and GitHub
* **Development Environment:** VS Code / PyCharm

**4. Key Features:**

* Add, update, delete book records
* View book availability and borrower details
* Issue and return functionality with date tracking
* Persistent data management (saved locally)
* Clean GUI layout with categorized views (Books, Users, Transactions)
* Responsive and user-friendly interface

**5. Target Users:**

* Small library managers (e.g., school or college libraries)
* Librarians and students for practice and learning
* Developers contributing to open-source educational tools

**6. Open Source Contributions:**

* Project will be hosted on **GitHub**
* Repository will include:
  + Clear folder structure and documentation
  + README file with setup instructions
  + Contribution guidelines and license (MIT recommended)
  + Issues and pull requests for community collaboration

**7. Tentative Timeline:**

|  |  |
| --- | --- |
| **No.** | **Task** |
| 1 | Setup project environment and GitHub repo |
| 2 | Design front-end UI with Kinder |
| 3 | Implement back-end logic for book and user management |
| 4 | Integrate frontend with backend |
| 5 | Add issue/return functionalities |
| 6 | Implement data persistence (JSON/SQLite) |
| 7 | Testing, error handling, and UI polishing |
| 8 | Documentation and final submission |

**8. Conclusion:**

The **Library Management System** project showcases the integration of a Python-based GUI (Kinder) with a functional backend to create a useful desktop application. It reflects practical skills in software development and open-source collaboration. The project lays the groundwork for potential future enhancements such as user authentication, reporting tools, or cloud storage.