To whomever it may concern, thank you for taking the time to read this documentation of my year 1 semester 2 project at Strathmore university. This is a Library Management System with the aim of allowing users to conveniently borrow and return books. This file contains system designs and flows as we go through the systems development. I have 4 days to complete this project, new challenge.

The system diagrams will be made using<https://draw.io>

The system itself will be made using C++ in vscode.

The final results will be posted onto my github <https://github.com/powwwy/Library-System>

To begin I shall start be explaining functionality of the system. To explain the functionality, I shall explain the various parts this system needs to work. The most important aspect of anything is the human one, therefore our main component are users. How many users do we need? As of right now I can only imagine one user that is the Library Member himself, optionally I have been given the option of a librarian. The member has to create an account. This comes with implications.

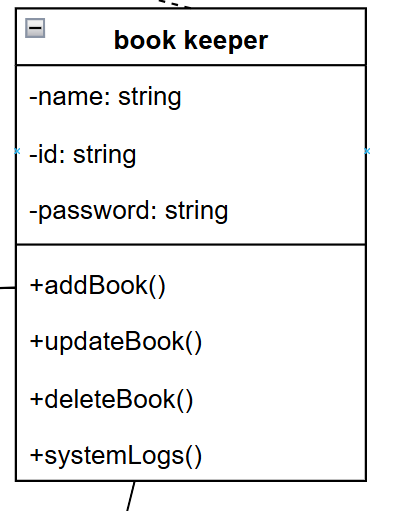
**📌 Policies & System Rules**

1. **Borrowing Rules**
   * A member can borrow a book if copies are available.
   * The due date is **14 days** after borrowing.
   * A book must be returned on time, or it will be marked **overdue**.
   * Maximum of 5 books borrowed at any 1 time and up to 3 copies if available.
2. **Return Rules**
   * If returned late, penalties may apply (e.g., warning, fee, restricted borrowing).
3. **Storage & Data Handling**
   * All book, member, and borrowing details will be stored in a **file-based system**.

booKeeper class (Inherits from person)

Attributes: keeper\_id, password, name

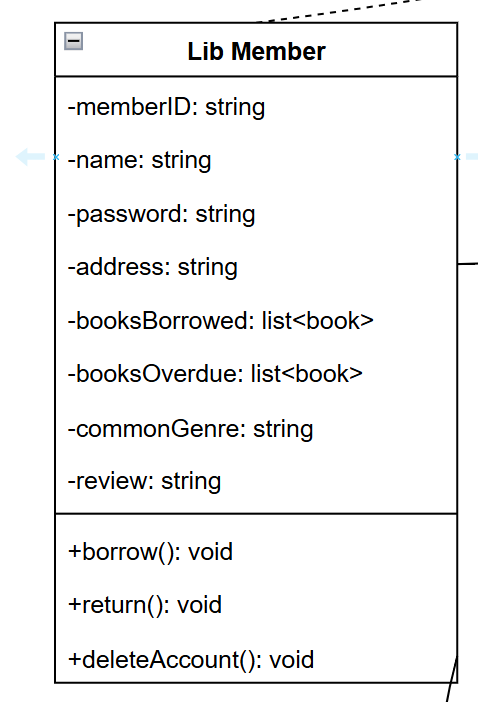
Functions: addBooks(copies), viewSystem(), removeBook()



Member class (Inherits from Person)

Attributes: member\_id, name, password, address, booksBorrowed, booksOverdue, commonGenre, review

Functions: borrow(), return(), deleteAccount()

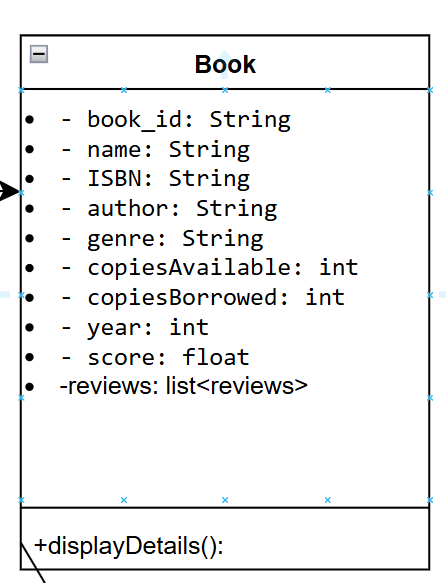


Having done the most crucial part we can move on to the subject of the system, the books. What does a book contain?

Book class

Attributes: book\_id, name, ISBN, Author, genre, copies (Borrowed and Available), year, score, reviews

Functions: displayDetails()

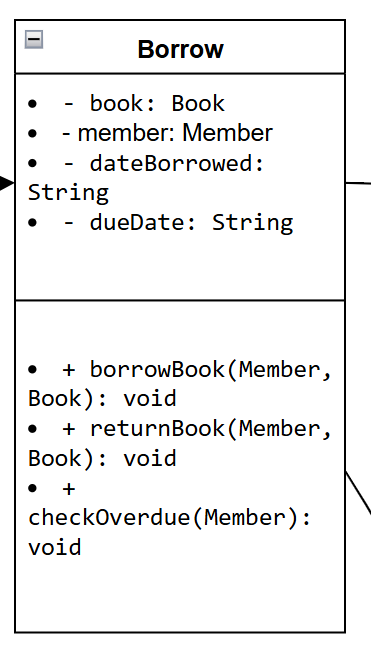


Thus the majority of the system is done. To wrap things up we need to perform some actions we can make a class for borrowing and returning. Let’s set some polices for the library as well for example books must be returned before 14 days otherwise they are marked as overdue and some penalties occur. Maximum books to be borrowed at one given time are 5.

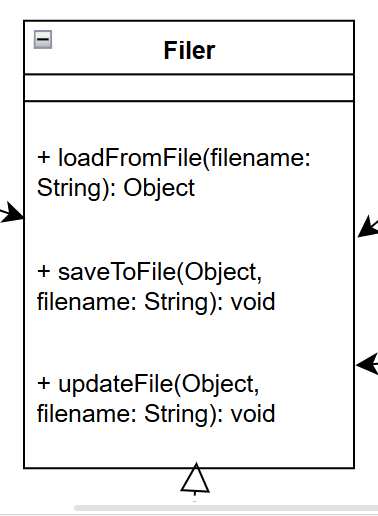
Borrowing class

Attributes: book, member, dateBorrowed

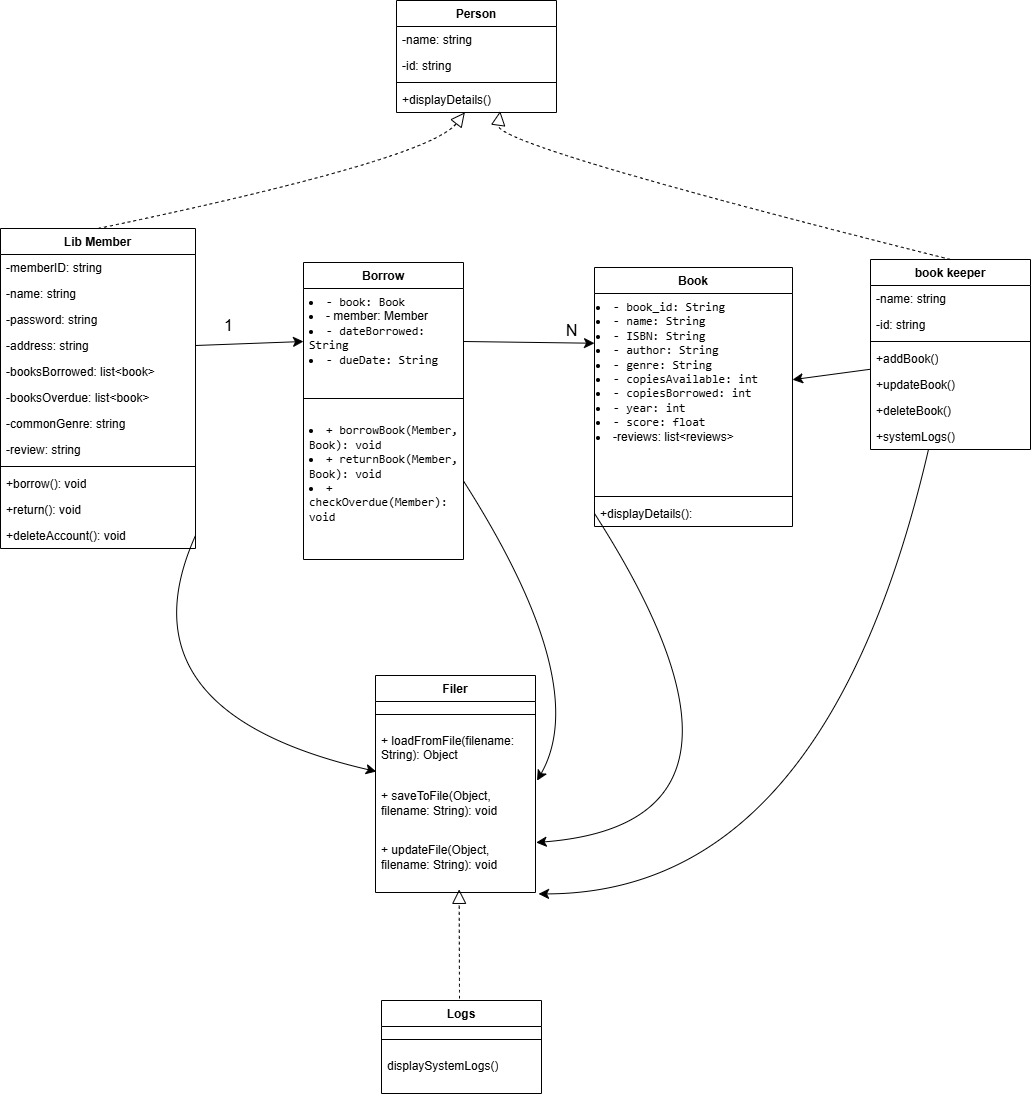
Functions: borrowBook(), returnBook()



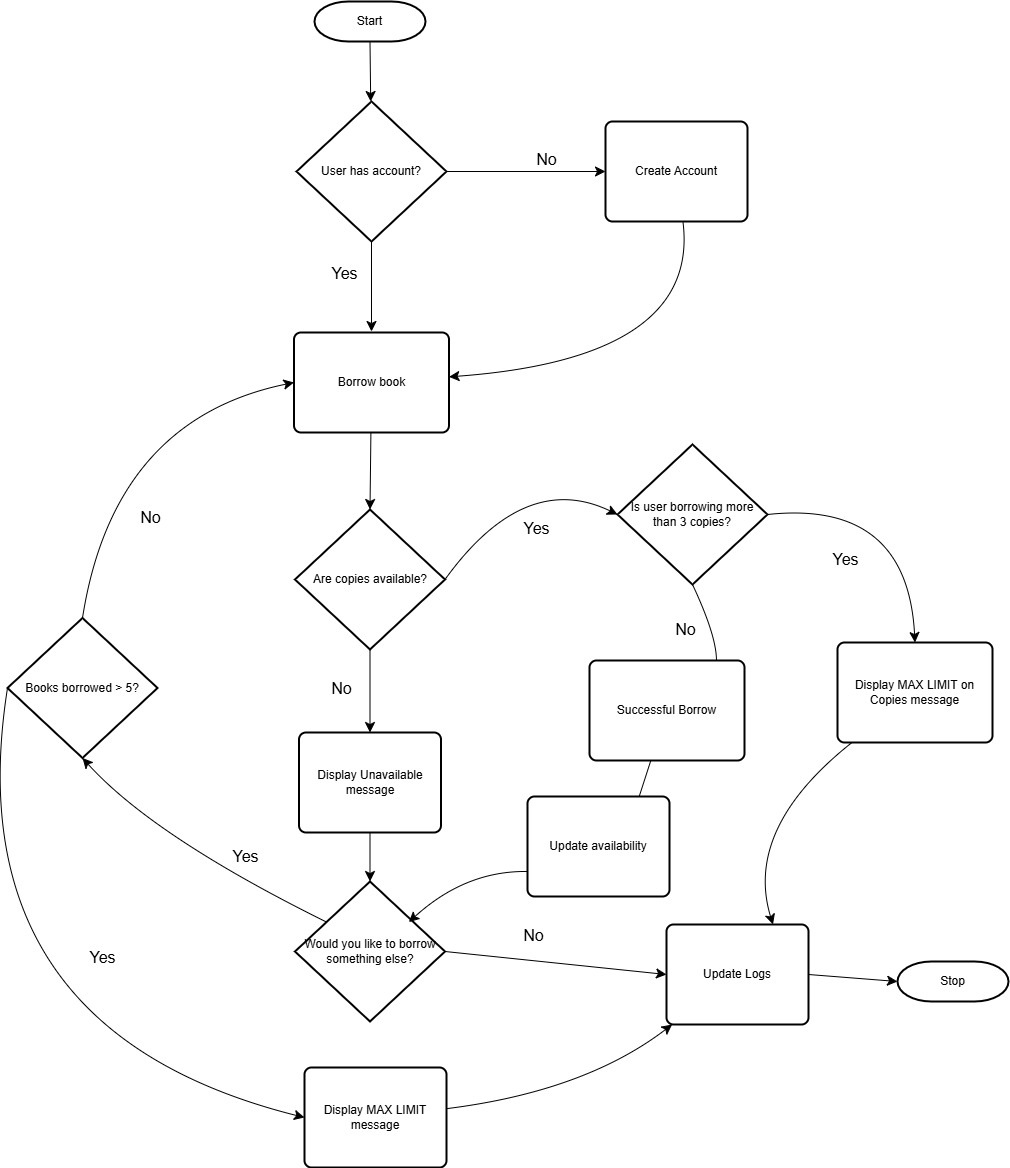
The final tie of the system is the storage of all of this data. We must set up a file storage system to hold the details of everything: users, books, dates…everything.



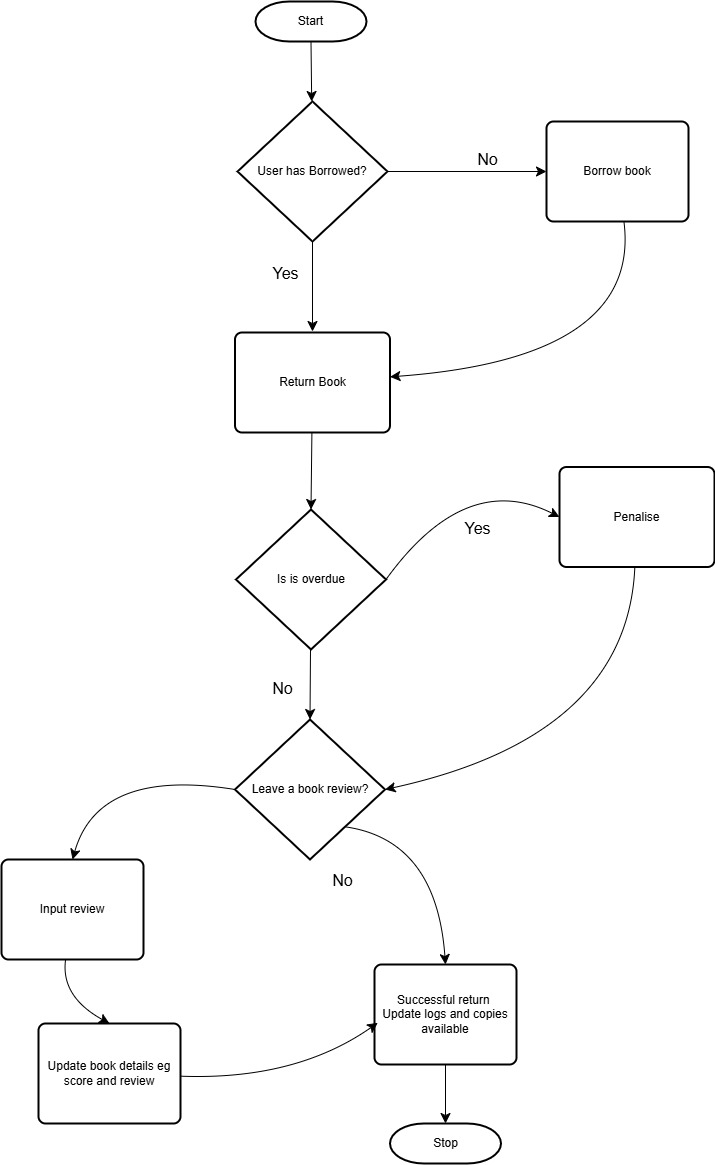
Now this is the end of this document let me know what you think. More designs after.



Full class diagram



Flowchart for Borrowing



Flow chart for returning.