Programming Project / C++

Library Management & Patron Self-Service

Student 1: Burlacu Cristian-George
Student 2: Ivan Stefan

I. Task Description

This project implements a two-part Library Management & Patron Self-Service system in C++. The Librarian program allows staff to manage the book catalog and track loans, while the Patron program enables users to search, borrow, renew, and return books. Both executables work on the same plain-text data files so they remain synchronized without a database.

Student 1 – Librarian responsibilities:

- View catalog – display all books with ID, title, author, and copies available.
- Add a book – insert a new book into the catalog with its details.
- Delete a book – remove a book from the catalog (if it exists).
- Modify copies – update the number of copies for a given book.
- View loans – list all current active loans.
- Record return – mark a borrowed book as returned, updating loan records and stock.

Student 2 – Patron responsibilities:

- Search books – search the catalog by keyword and show availability.
- Borrow book – borrow a book (if available) by ID, creating a loan and reducing stock.
- View loans – list the patron's currently borrowed books and due dates.
- Renew book – extend an existing loan's due date.
- Return book – return a borrowed book, removing the loan and increasing stock.

II. Data Structures Used by the Team

- Book – string id, string title, string author, int copies
- Date – int day, int month, int year
- Loan – string bookID, Date borrowDate, Date dueDate (may include patron ID)

III. File Structure

The two programs communicate through these plain-text files:

catalog.txt

<number of books>
<book1 id> <book1 title> <book1 author> <book1 copies>
<book2 id> <book2 title> <book2 author> <book2 copies>
...

loans.txt

<loan1 bookID> <borrow day> <borrow month> <borrow year> <due day> <due month> <due year>
<loan2 bookID> <borrow day> <borrow month> <borrow year> <due day> <due month>

...

Dates are stored as separate day, month, and year fields. When a book is borrowed, a new line is added; when returned, the corresponding line is removed.

IV. Interacting with Executables

Application 1 – Librarian commands:

- ./app_1.exe view_catalog – view all books in the catalog
- ./app_1.exe add_book <bookID> <title> <author> <copies> – add a new book
- ./app_1.exe delete_book <bookID> – delete a book
- ./app_1.exe modify_copies <bookID> <new_count> – update copy count
- ./app_1.exe view_loans – view active loans
- ./app_1.exe record_return <bookID> – mark a book as returned

Application 2 – Patron commands:

- ./app_2.exe search_books <keyword> – search the catalog
- ./app_2.exe borrow_book <bookID> – borrow a book
- ./app_2.exe view_loans – view personal loans
- ./app_2.exe renew_book <bookID> – renew a loan
- ./app_2.exe return_book <bookID> – return a book

V. Optional Features (Stretch Goals)

- Overdue Fines – track due dates and calculate fines for late returns.
- Advanced Search – search by author, genre, or partial title matches.
- Reporting – list most-borrowed books, loans over a period, etc.

VI. System Data Flow Diagram

Both the Librarian and Patron programs read from and write to catalog.txt and loans.txt. When a patron borrows or returns a book, these files are updated; the Librarian application immediately reflects those changes and vice-versa.