# Fundamentals of Computing II - LAB
# CSCE 1102- Project
# Milestone 1 - Design Report

**Team#: Team Email: youstinasaleh@aucegypt.edu**

**Members' Info:**

| Member Name | ID | Email |
|---|---|---|
| Abdelrahman Ali | 900232237 | abdelrahman72688@aucegypt.edu |
| Wassim Bousbia | 900234040 | Wassim@aucegypt.edu |
| Youssef Sarofeem | 900237960 | Youssef_Sarofeem@aucegypt.edu |
| Youstina Saleh | 900245564 | youstinasaleh@aucegypt.edu |

**GitHub link:**
**https://github.com/youstina-saleh/LibraryManagementSystem.git**

## Core classes

| Class Name /Short description | Data members | Member Functions |
|---|---|---|
| **Book**<br>Represents a single book in the library's collection, including all information needed to manage it. | int bookID<br>string title<br>string author<br>string category<br>int totalCopies<br>int availableCopies<br>bool isAvailable | setBookDetails()<br>getBookInfo()<br>updateAvailability(int change)<br>isInStock()<br>matchesSearch(string keyword)<br>toString() (for file export) |

| | | |
|---|---|---|
| **Transaction**<br>Represents a borrow or return event of a book by a user, including due dates and fines. | int transactionID<br>int bookID<br>int memberID<br>string borrowDate<br>string dueDate<br>string returnDate<br>double fineAmount<br>bool isReturned | createTransaction(int bookID, int memberID)<br>markReturned(string date)<br>calculateFine(string currentDate)<br>toString() (for file storage) |
| **Catalog**<br>Manages the collection of books (the library's catalog), including search, add, and remove operations. | vector<Book> booksList | addBook(Book newBook)<br>removeBook(int bookID)<br>searchBook(string keyword)<br>updateBook(int bookID, Book updatedBook)<br>getAvailableBooks()<br>loadFromFile(string filename)<br>saveToFile(string filename) |
| **BorrowRecord**<br>Tracks which members borrowed which books and their current statuses. | vector<Transaction> transactions | recordBorrow(int memberID, int bookID)<br>recordReturn(int memberID, int bookID)<br>getMemberBorrowHistory(int memberID)<br>getOverdueTransactions(string currentDate)<br>saveToFile(string filename)<br>loadFromFile(string filename) |
| **Report**<br>Generates and exports different types of reports (borrowed, overdue, most popular, etc.). | vector<Transaction> transactions<br>vector<Book> books | generateBorrowedBooksReport()<br>generateOverdueBooksReport()<br>generateMostBorrowedBooksReport()<br>exportToTxt(string filename)<br>exportToCsv(string filename) |
| **Reservation**<br>Handles book reservations and | int bookID<br>queue<int> waitingMembers | addReservation(int memberID)<br>getNextReservation()<br>isMemberInQueue(int memberID) |

| | | |
|---|---|---|
| waiting lists (optional but recommended as a core feature for the “Bonus Features” section). | | removeReservation(int memberID) |

## Users classes and functionality of each user

| Class Name /Short description | Data members | Member Functions |
|---|---|---|
| **Admin**<br>Manages user accounts system control. Only the default admin can create and manage user accounts. | - string username<br>- string password<br>- string role (set as "Admin")<br>- vector<User> users | addUser() → Create new librarian or member accounts.<br>- deleteUser() → Remove existing users.<br>- listUsers() → Display all user accounts.<br>- validatePassword() → Ensure password meets criteria (≥8 chars, includes number).<br>- saveUsersToFile() → Store user info persistently. |
| **Librarian**<br>Responsible for managing books, transactions, and reports within the system. | - string username<br>- string password<br>- string role (set as "Librarian")<br>- vector<Book> bookList | - addBook() → Add a new book record.<br>- editBook() → Modify existing book details.<br>- deleteBook() → Remove book from inventory.<br>- searchBook() → Search by title, author, or category.<br>- generateReport() → Create book usage and borrowing reports. |
| **Member** | string username<br>- string password<br>- string role (set as "Member")<br>- vector<Book> borrowedBooks | - viewBooks() → Display available books.<br>- borrowBook() → Borrow a book if copies are available.<br>- returnBook() → Return borrowed book and update status. |

| | | |
|---|---|---|
| Represents library members who can view, borrow, and return books. | - float fineAmount | - calculateFine() → Compute overdue fines based on return date.<br>- viewBorrowedBooks() → List all borrowed books with due dates. |

## Maestro class(es)

Note: the bonus parts are not included here yet

| List Name | Data members | Member Functions |
|---|---|---|
| **UserManager**<br>Handles creation, authentication, and management of users. | - vector<User*> usersList<br>- string usersFile | addUser()<br>removeUser()<br>listUsers()<br>login()<br>validatePassword()<br>loadUsersFromFile()<br>saveUsersToFile() |
| **BookManager**<br>Manages the library catalog and book records. | - vector<Book> booksList<br>- string booksFile | addBook()<br>deleteBook()<br>editBook()<br>searchBook()<br>updateBookAvailability(),<br>loadBooksFromFile()<br>saveBooksToFile() |
| **TransactionManager**<br>Tracks borrow/return operations and overdue fines. | -vector<Transaction> transactionsList<br>-string transactionsFile | borrowBook()<br>returnBook()<br>calculateFine()<br>loadTransactionsFromFile(),<br>saveTransactionsToFile() |
| **LibrarySystem**<br>The central controller | - UserManager userManager | loadDataFromFiles()<br>saveDataToFiles() |

| | | |
|---|---|---|
| coordinating all operations between managers. It loads/saves data, handles login, and directs user actions. | - BookManager bookManager<br>-TransactionManager transactionManager | loginUser() |