



Middlesex
University

CST2550 Coursework 1

Library System

M00935155

Date Of Submission: 12/01/24

Introduction

This project is a fully functional Library System tailored for a librarian's use. It allows the librarian to add new members, issue books to members, and handle book returns alongside fine calculations.

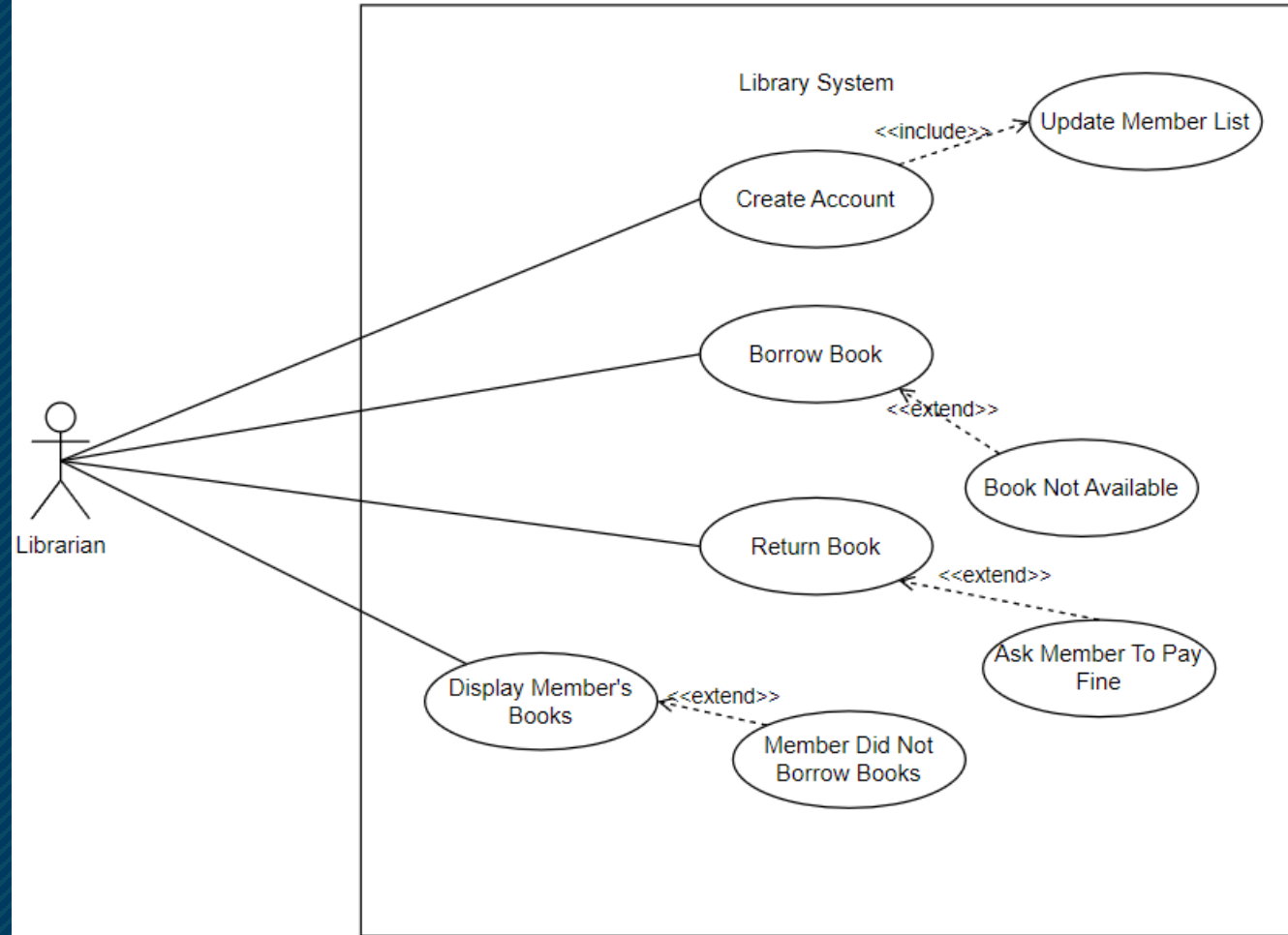
This presentation will demonstrate the project by showcasing its design and implementation and the testing approach used to ensure the system is robust.

Design

The use case diagram showcases the functionalities of the library system.

The librarian oversees tasks such as:

- Member registrations
- Book lending and returns.
- Fines Retrieval
- Displaying all borrowed book of a member

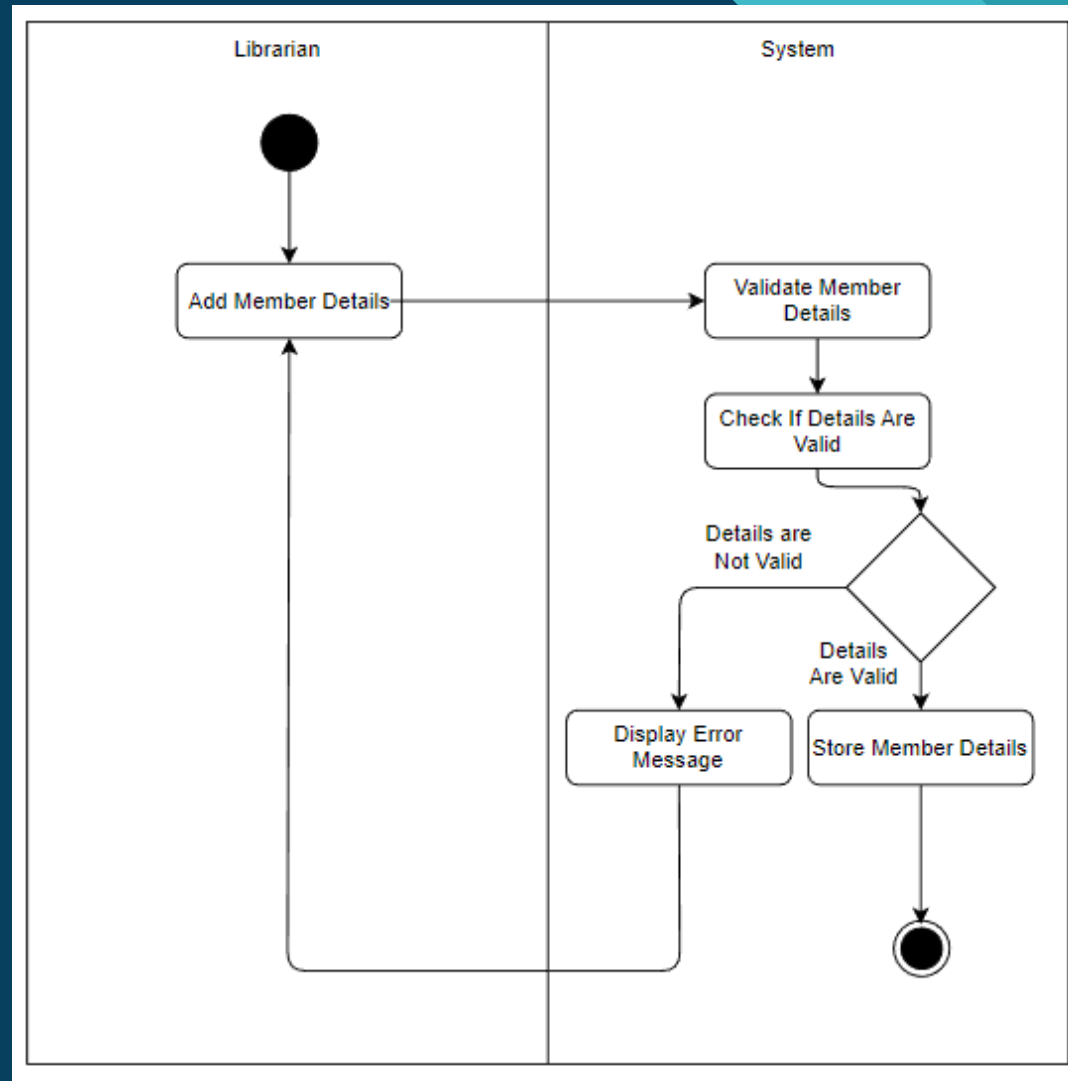


Use Case Diagram

Activity Diagram

The Create Account activity diagram handles the following:

- Add Member Details
- Validate the Details
- Display Error Messages
- Store Member Details

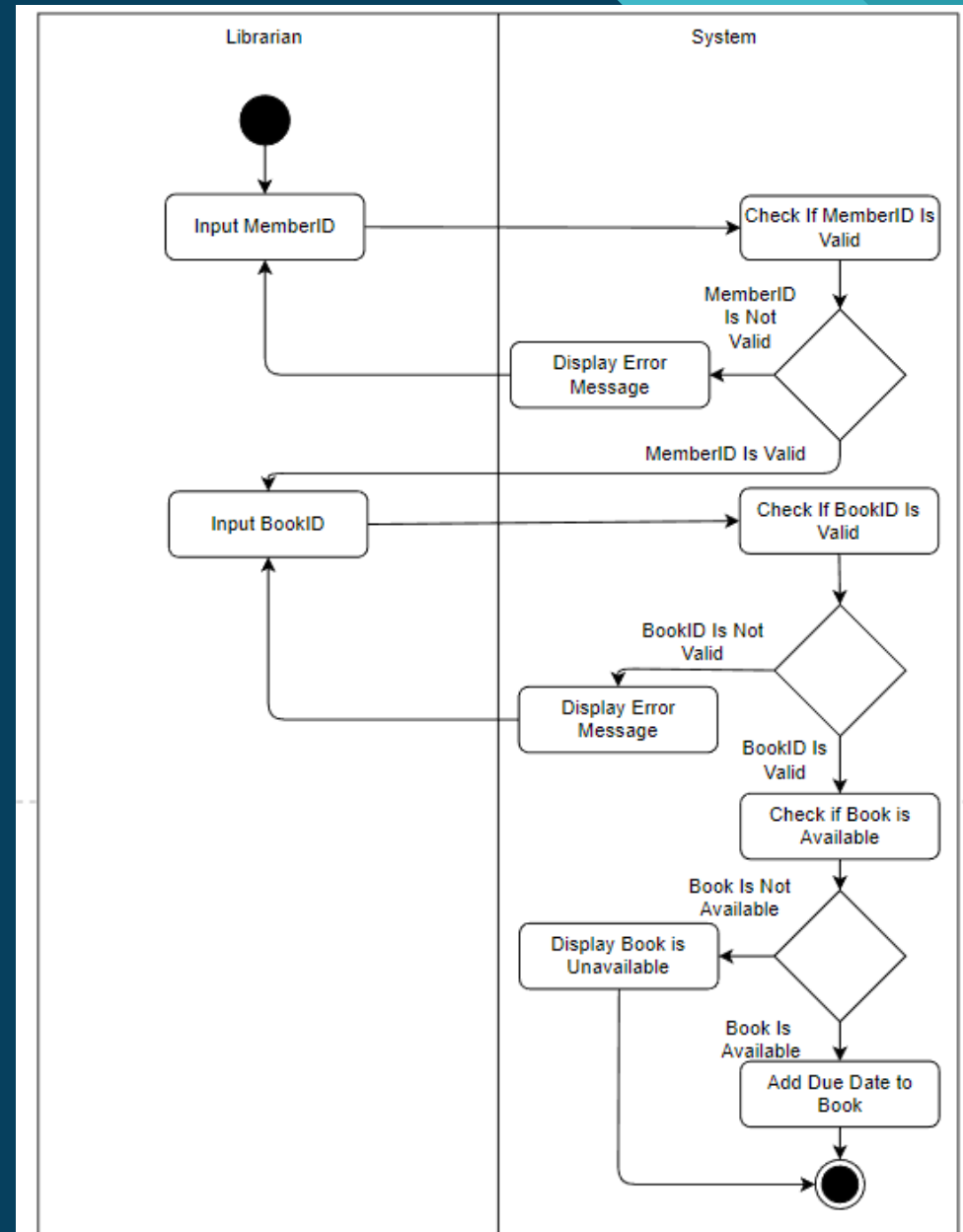


Activity Diagram - Create Account

Activity Diagram

The Borrow Book activity diagram handles the following:

- Accepts a member's ID
- Validates member's detail
- Accepts a book's ID
- Validates book's ID
- Check book's availability
- Display Error Messages
- Calculate Due Date of book

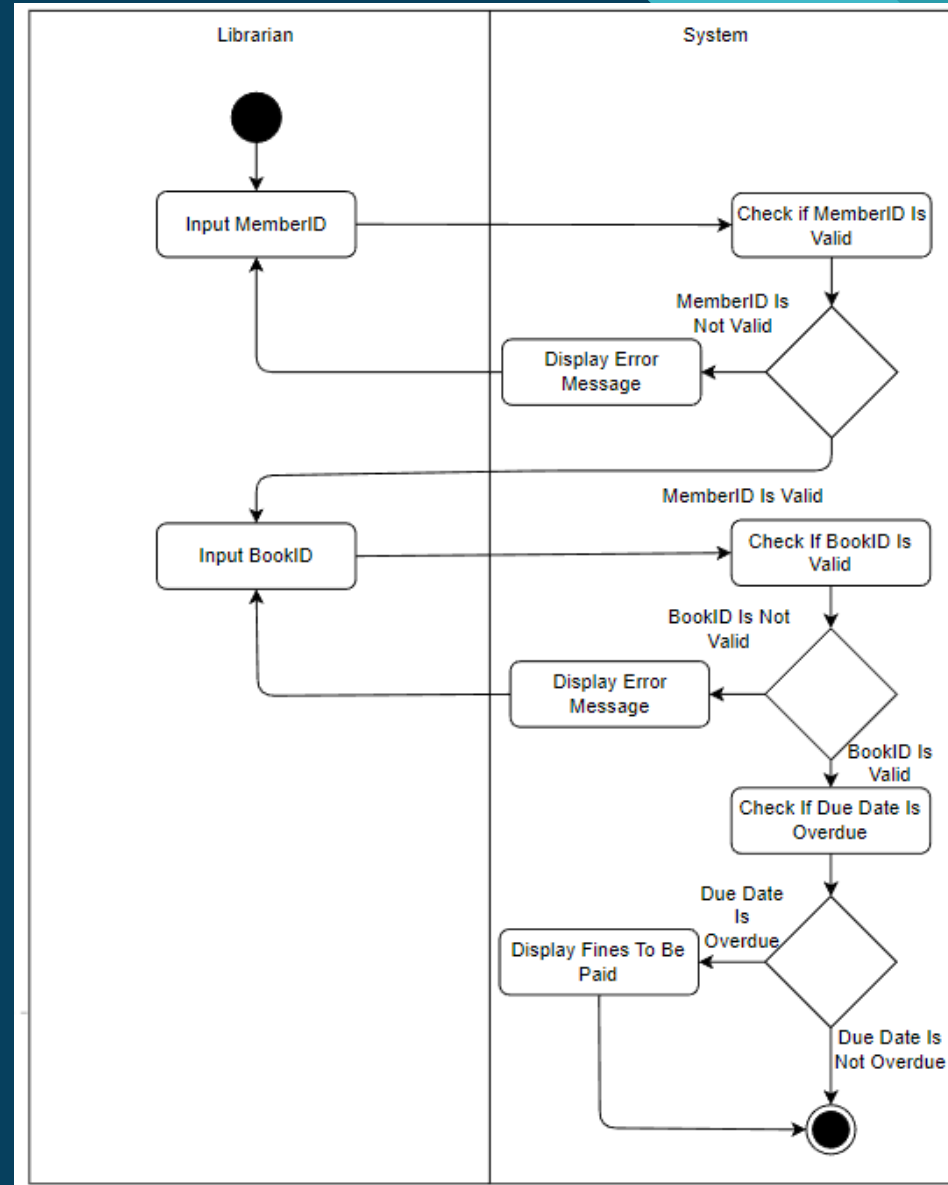


Activity Diagram - Borrow Book

Activity Diagram

The Return Book activity diagram handles the following:

- Accepts a member's ID
- Validate the Details of the Member
- Accepts a book's ID
- Validates book's ID
- Display Error Messages
- Check if book is overdue
- Display Fines if Any

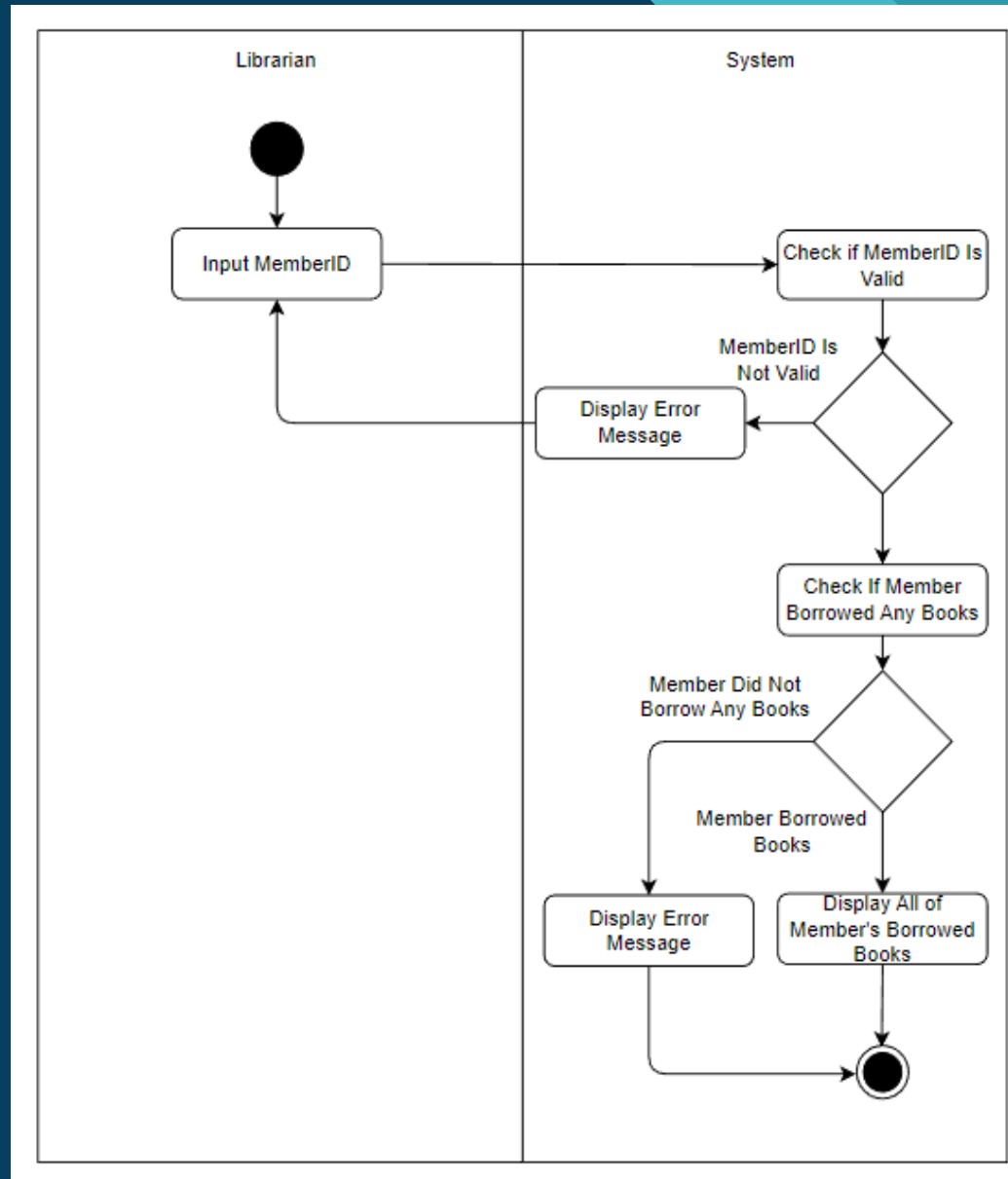


Activity Diagram - Return Book

Activity Diagram

The Display Member Book's activity diagram handles the following:

- Accepts A member's MemberID
- Validate the MemberID
- Display Error Messages
- Verify that the member borrowed any books
- Display all of the member's book



Activity Diagram - Display Member's Book

Implementation

To create the software, the UML class diagram was used to initially design different classes for Person, Member, Librarian and Book.

The Person class was the first one to be coded since it is the parent class of both Librarian and Member.

The getters and setters alongside the constructors of each class were the starting point of the program after which more focus was put into reading the CSV file and extracting data from it.

Input validation was next thing which was worked on which then allowed the Librarian Menu to be created.

After the above, most of the methods of the different classes were being completed.

Date related functions and methods were amongst the last things to be implemented.

Makefile

The makefile was used to manage the compilation and linking of the different .cpp and .h files.

Instead of manually compiling the file, the makefile allows for an automatic build process which is faster and less time consuming.

The makefile makes use of compiler flags which during compilation will display warning messages if any issues are found.

Furthermore, it can detect which files have had changes in them and only compiles those files instead of compiling every file.

Additionally, the makefile can be used to remove compiled file by setting a “clean” target.

Moreover, automatic variables allowed for a cleaner looking script which uses less text.

Finally, it also sets the compiler which will be used to compile the program.

Version Control

Version control was managed locally by Git and remoted using Bitbucket to store the repository online


























Version control was used to track all changes made to the code.


























Branching was also use to experiment with features without altering already working code.

Reverting to previous commits was useful in case of unexpected issues

The remote repository acted as a backup of the code in case something happened to the local repository

	Author	Commit	Message	Date
●	HM Hemant Mooneea	28f6932	Added validation for address	2023-12-23
●	HM Hemant Mooneea	cdd7eed	Added validation for Name	2023-12-23
●	HM Hemant Mooneea	6cec737	Added code to create a new Librarian Object	2023-12-23
●	HM Hemant Mooneea	1d054aa	Added default constructors for the Book, Member and ...	2023-12-23
●	HM Hemant Mooneea	c7eca2b	Made setbooksborrowed accept pointers to books in th...	2023-12-23
●	HM Hemant Mooneea	8543b52	Added setter for due date	2023-12-22
●	HM Hemant Mooneea	7090a77	Fixed issues with getters and setters for librarian and m...	2023-12-22
●	HM Hemant Mooneea	7a03561	Added code for Setters and Getters	2023-12-22
●	HM Hemant Mooneea	89f4097	Added all the methods for Books in library_system.cpp	2023-12-22
●	HM Hemant Mooneea	9370219	Added all the methods for Member in library_system.cpp	2023-12-20
●	HM Hemant Mooneea	b60a536	Added all the methods for Librarian in library_system.cpp	2023-12-20
●	HM Hemant Mooneea	f61f84e	Added Date structure to header file	2023-12-20
●	HM Hemant Mooneea	280e736	Can now set and get information of the librarian	2023-12-19
●	HM Hemant Mooneea	1a4c373	Added setters and getters for person	2023-12-19
●	HM Hemant Mooneea	ce0ea2d	Fixed some issues in the header file	2023-12-19
●	HM Hemant Mooneea	c8bebc2	Added all the classes for the header file	2023-12-19
●	HM Hemant Mooneea	405b995	Added classes for Book and Member	2023-12-19
●	HM Hemant Mooneea	024a2cd	created header file for person	2023-12-16

	Author	Commit	Message	Date
	 Hemant Mooneea	5808500	Fully commented the code	6 days ago
	 Hemant Mooneea	e1bba62	Removed extra function declarations	6 days ago
	 Hemant Mooneea	1644ff3	Started Commenting the code + added a check to see i...	6 days ago
	 Hemant Mooneea	f6b5f03	Added fine calculation upon returning the book late	2024-01-04
	 Hemant Mooneea	75aadbb	Added a function which returns the difference of days b...	2024-01-04
	 Hemant Mooneea	1c80311	Member can now return books, fines are NOT impleme...	2024-01-03
	 Hemant Mooneea	261c48a	Due Date is now displayed when Displaying all the boo...	2024-01-03
	 Hemant Mooneea	ce1d3ab	Issue book and Display Book added	2024-01-02
	 Hemant Mooneea	d2c1328	Added function to obtain Dates	2024-01-01
	 Hemant Mooneea	3136ce5	Started Work on issuebook method	2024-01-01
	 Hemant Mooneea	e18dacd	Added Functions to check if bookID and memberID exi...	2024-01-01
	 Hemant Mooneea	41f3d09	Removed createMember function and instead moved it...	2023-12-31
	 Hemant Mooneea	ffcd408	Made the vectors storing books and members global fo...	2023-12-31
	 Hemant Mooneea	a63708c	Fixed an issue with readBookFiles whereby books havin...	2023-12-31
	 Hemant Mooneea	6ac2396	Added findIndexBook function which will return the ind...	2023-12-31
	 Hemant Mooneea	e4963a6	Reworked the checkMember function into findIndexMe...	2023-12-31
	 Hemant Mooneea	f88a857	Added a function to check if a member exists or not	2023-12-31
	 Hemant Mooneea	03e9924	Librarian can now add members	2023-12-31
	 Hemant Mooneea	d029be4	Partial completion of the createmember function	2023-12-26
	 Hemant Mooneea	49d6eaa	Added a menu function for librarian to use	2023-12-26
	 Hemant Mooneea	430b015	Added function which extracts all the data of the book ...	2023-12-25
	 Hemant Mooneea	3460f31	Added Functions to locate CSV file + check if it exists	2023-12-24
	 Hemant Mooneea	76e67da	Added validation for email	2023-12-24
	 Hemant Mooneea	5ae9b68	Added validation for salary and staffID	2023-12-24
	 Hemant Mooneea	7cde250	Removed unnecessary parameters from validations	2023-12-23

Author	Commit	Message	Date
 Hemant Mooneea	67c5b34	Added comments for all class methods	5 minutes ago
 Hemant Mooneea	9de6a99	Added additional test cases	13 hours ago
 Hemant Mooneea	e49c66c	makefile can now compile test.cpp	23 hours ago
 Hemant Mooneea	0eb0558	Changed error message which was output when name ...	yesterday
 Hemant Mooneea	a832792	Removed redundant code in book.cpp	yesterday
 Hemant Mooneea	72e49e9	Added File description on top of each new code files	yesterday
 Hemant Mooneea	379a188	Separated old code and new code	2 days ago
 Hemant Mooneea	dbb4423	Librarian menu is now using if else instead of switch case	2 days ago
 Hemant Mooneea	c1f66cc	Commented out some redundant code	2 days ago
 Hemant Mooneea	a6d0041	makefile can now compile all header and cpp files	2 days ago
 Hemant Mooneea	1be4f5e	added comment	2 days ago
 Hemant Mooneea	7206462	separated every class to its own .h file and adjusted ma...	2 days ago
 Hemant Mooneea	eec83b5	Fixed an issue where compiling the make file would giv...	2 days ago
 Hemant Mooneea	471dc97	Updated the dates on the file header of each file	2 days ago
 Hemant Mooneea	6e7cd8c	added regex library	2 days ago
 Hemant Mooneea	ac22c59	Removed unused variable + added static_cast<int> to	2 days ago
 Hemant Mooneea	ea078e8	Added cxx and cxxflags to the makefile	2 days ago
 Hemant Mooneea	3a33e1e	Added makefile and catch.hpp	2 days ago
 Hemant Mooneea	84e02df	displayBooks now displays bookID of a book	3 days ago
 Hemant Mooneea	3ae5558	Removed some comments which were redundants fro...	3 days ago
 Hemant Mooneea	75db138	Changed all std::cin to std::getline to allow spaces withi...	3 days ago
 Hemant Mooneea	46f9d99	Rewrote all function description comments to fit the co...	3 days ago
 Hemant Mooneea	8427a60	Added file information(file name, author, date) at the to...	3 days ago
 Hemant Mooneea	9c0fabd	Removed non class functions from library_system.cpp a...	3 days ago
 Hemant Mooneea	43d6c61	Moved all library declarations in library_system.h	6 days ago

Testing Approach

Automated and Unit testing were used to verify that the program performed as required.

By making use of the Catch2 library, test cases were carried out where individual parts of the program were tested separately and the process fully automated.

Test cases included:

Ensuring that constructors, setters and getters worked as intended for each class

Ensuring that functions which validated input characters worked properly

Ensuring that functions which handled dates calculations worked

Conclusion

To conclude, this project entails a library system designed for a librarian to manage the library by performing tasks such as adding new members, lending and borrowing books and displaying all of the books of a specific member

Limitations and Future Approach

Limitations

- Use of global variable vectors to store objects of Books and Members which is not good practice
- Naming convention were sometimes ignored and inconsistent in the UML class diagram
- Some details about the books were ignored which made them redundant
- No specification for Date data type lead to use of a Struct data type to store date info
- If the program is closed, all data is loss and has to be input once more
- Librarian methods such as getStaffID() and getSalary() are unused throughout the whole code, based on the specifications

Future Approach

- Use of alternative storage options which are not global such as a librarian method which would store data
- Use of a consistent and appropriate naming convention for all of the code of the program
- Add support for all of the book's details(e.g. page number, genre, etc)
- Details about the Date data type should have been mentioned instead of leaving it up for assumption
- Use of a database to store member and book information
- Use of updated specifications to allow more use of the Librarian methods.