**Software Engineering and Software Development**

Karl Sam

M00853212

**Project Task Description:**

In this project we will be creating a library system that is able to allow users of the library to be able to become an official member of the library and have their data saved in the library system, borrow a book of another number of books for up to three days, return a book or books that the member have decided to borrow where if they return the book past the limited days announced they will be fined one pound per day missed, and this project will also be able to display all the books borrowed.

**Project Presentation Overview:**

In this project presentation we will be going through the creation and implementation that I have used to create the recommended task small library, I will be discussing the UML diagrams explaining the components in detail with what they mean, what they do, and how they correlate with the program including how they were conjured into code with these uses of classes, pointers, functions along with testing the source code with the methods of catch2 and compiling with the use of make files using the Cygwin terminal.

**Design:**

A diagram of a user interface

Description automatically generated

**Figure 1: Use Case Diagram**

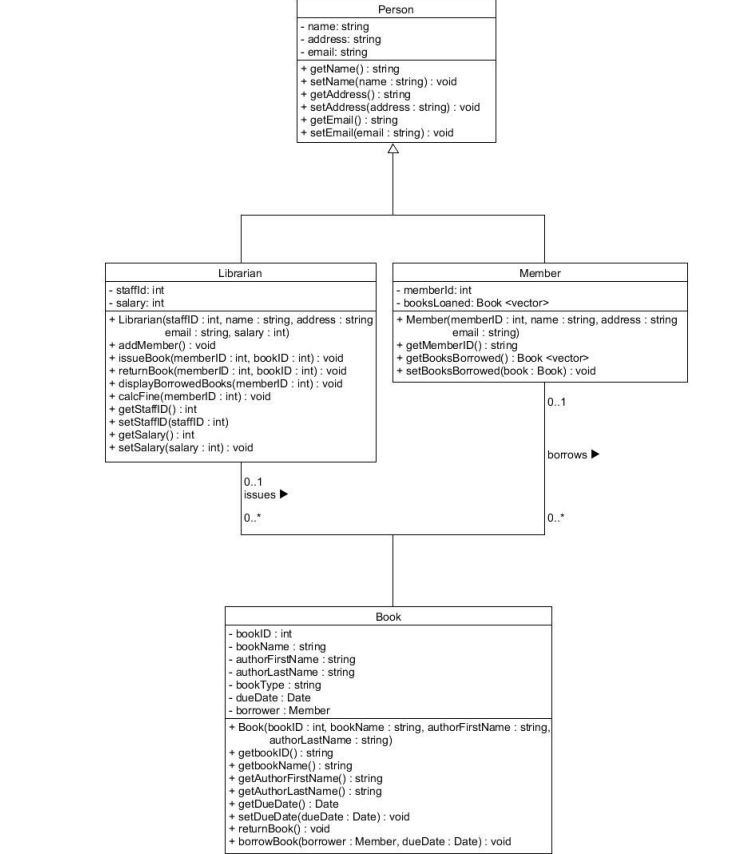
For this use case diagram, it is showing a visual presentation of the interaction between an actor and a system that being the librarian and the library system management software. The requires for this use case are many as the users of the library may want to become a member, borrow, or return a book, with an additional task calculation of a fine if the due date has passed. In this use case it shows the many paths that the librarian can go through in the system to meet said requirements. This also shows the transitions as the Librarian selects options within the library system.

**A diagram of activity diagram

Description automatically generated**

**Figure 2: Activity Diagram**

The activity diagram shows the process of using the library system management software from a user’s perspective, this is a more detailed UML diagram compared to the use case diagram because it incorporates a more detailed approach towards each decision path made that currently exists within the occurrences contained in the section of the program by the user, that being the librarian. Providing with a further insight on how the program is run with a brief hint on what the users will expect while using the system display a domino effect as one case of events leads to another until the end of that decision ends, showing the start point and finishing point in an activity diagram it is displayed as a simple node. Some decision can be made on the current state for example if a member was to hand in a book that had been expired then the calculation fine will be summed based on the number of days past.



**Figure 3: UML Class Diagram**

The Unified Modeling Language Class Diagram is a more software approach diagram that goes into depth of the structure of the system this is represented by displaying the implemented classes of the system within depth attributes and methods that each of these classes can perform along with the correlation between the objects. These are useful to software developers as it gives them a clearer approach on how to execute these tasks in creating a system. In this diagram it is showing the classes that should be included in the systems, displaying the inheritance that starts from the person which is ideally the parent while the librarian member etc. are the child, along with what each of these object’s while showing the capability of performing while the system is running along with giving the developer key instructions on how to create it with the type of data it should be and how it should be structured. It also prompts the developer on the type of behaviours each of the objects should have such as how many books are allowed to be issued and borrowed.

**Approach/Implementation:**

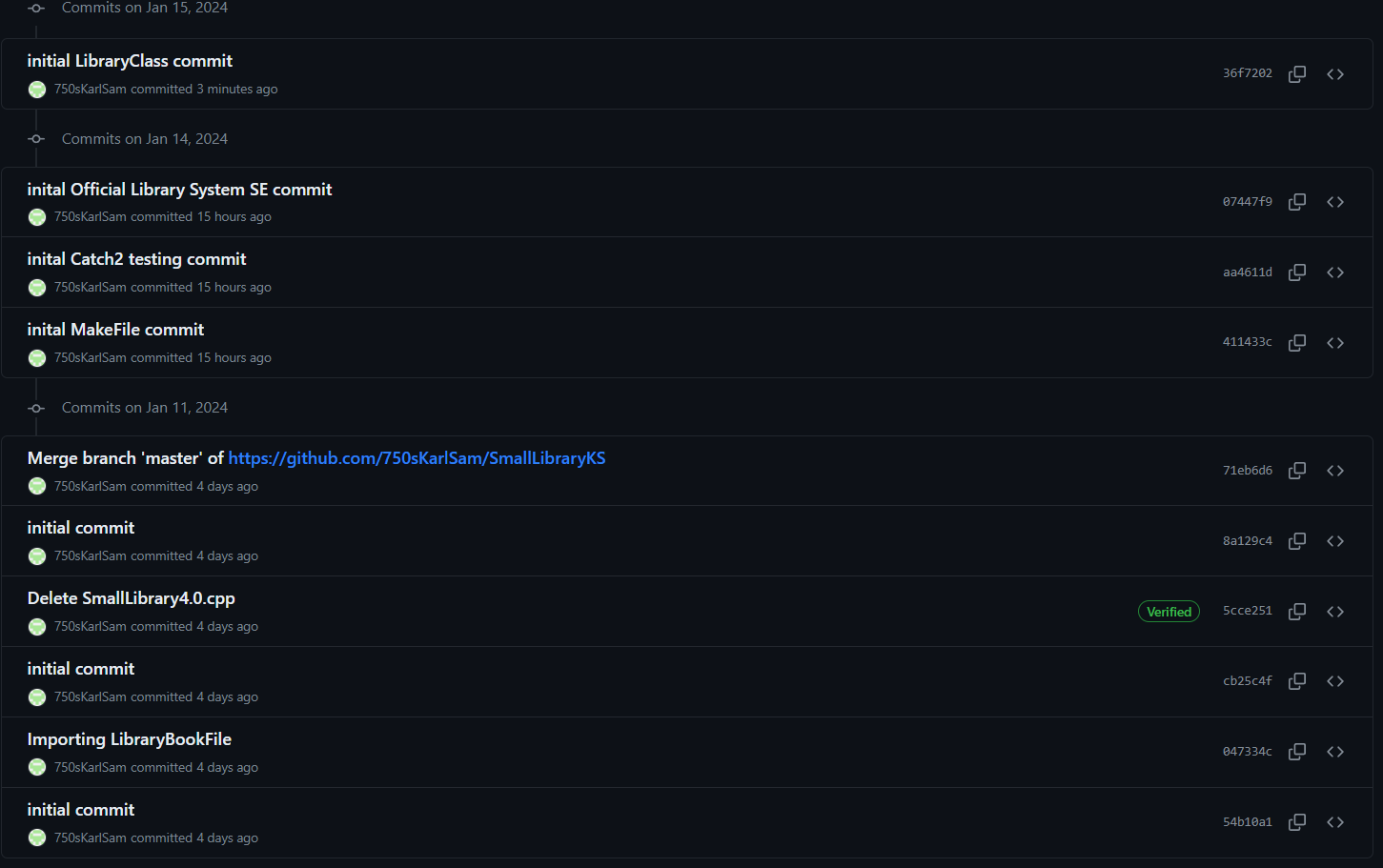
The way that I was able to translate the design into a working software with C++ is by using the following hints to conjure up the right technique to execute each function after carefully examining the UML class diagram understanding the four entities were objects it was clear that a class for each of these objects would need to be implemented into the code giving them their respected variables types and attributes which could be implemented by creating functions that allow these objects to have these types of behaviours within the program with further inspection through the diagram the use of parameters was helpful to allow data from other objects to be transferred to another object instead of creating it all entirely using this technique made it more effective when writing code given the techniques and understanding of them while discovering the right approach to executing this problem such as creating a while loop to iterate the program after an interaction of an event within the users choice would be complete. The diagram also assists with how to specify the behaviours and rules of the borrow and issue function giving us a hint on the variety of books can be borrowed per member this helped give a clearer view on how to select the correct approach to this functionality within the system.

The Makefile concept was simply used for compiling and running code. It allows the programmer to recompile specific parts of the code based on the changes made within the program by issues commands directed towards these sections. These are given specific tasks by the commands with an order, the files that are executable. The ones being updated from object files with a given rule to be enacted with a specific way to do it and fixed time, so ultimately, its choice is to compile the code every time the source files have been edited.

(Explain how and why version control was used)

Version control which in some cases are referred to as source control is the method of maintenance of changes to software code. The way that a version control works is by creating a repository in for example Github then creating a file and using the commands to compile and transfer the file into your repository with a commit. The reason as to why version control was used in this project is so that I was able to keep track of each modification that I have implemented towards the code overtime, this had been beneficial for me as a student as a few hurdles have come by my direction which had ultimately given me no choice but to retreat back to previous codes that I had updated on my GitHub repository which allowed me to get back on track with the existing code

***Screenshots of repository and commits:***



**Testing Approach:**

(Talk about how you applied testing approach)

After some research I had an idea of how to approach testing also known as unit testing is basically testing one part of the code by itself testing each function of chosen classes making sure that there’s no errors that occur during the processing time, I applied the testing approach including the catch.hpp the source code and the catch configuration main which then I could create a test case for specific classes that will be updated over time.

(Talk about test cases in detail what was being tested)

**Test Cases:**

The test cases that’s being tested where the main classes of the program for the Library System. For starters I decided to test the person class as it is the main class that is the inheritance of the following classes implemented into the source code I decided to the getter and setters within the person class to test with the following parameters. To do that you’d need to create a function called Test case with new parameter relating to the class parameter in your main code with the correct data types then, require built-in function which simply test the expression then discards it if the one of the expression fail then the functions within the person class that you want to be test in this example I choice the receivals of the name, address and email with the unit testing this will compile and process each section of the code and inform me if the testing has passed for each of them.

(Talk about the limitations of your work, what was the drawbacks, disadvantages)

**The Journey:**

The limitation of my work from the basic would be the lack of research for each concept included in this software project for the library system this would include the catch2 unit testing method understanding the syntax fully was something that I will implement with other concepts like this in future as I ended up encountering errors that did consume some time while I had other objective to complete within this project and perhaps the approach I decided to with may have limited capabilities to a certain degree within my code which resulting in the run task to give me outputs that wasn’t correctly accurate to my expectations nonetheless I was still satisfied that the results were still able to present a logical outcome for a library system where it would expectedly make sense to the user. There were a few drawbacks in the progression of the Library System as I had come across some unexpected events where some code had been erased which would cause my program to run different or have unexpected errors this too some time to figure out to my disadvantages until I ultimately was able to spot the messing code with satisfaction I might add, these were all learning curves which gave me different perspectives when coding.

(Talk about how the code is working while running the program/interacting with it to show the viewer how it works)

(Talk about how you would approach a similar project in future)

**Future Improvements:**

I believe what would be a more advisable approach for me in regards to implementing a more efficient and effective approach in future would be to perhaps plan out more thoroughly and organized structure with the techniques used to make the provided functions and even functions I would have conjure up, I believe with that approach I would have a better mapping of how to get the functions to result in a more accurate manner to the description. Additionally, I believe I should also take more time to understand and have better practice of the programming language C++ as it was an interesting experience and has made me more intrigued about the language itself and what other ideas and projects I may conjure up in future fully understanding the basic concepts in more depths would perhaps give me a bigger horizon on how to approach with maximum effort with satisfactory results. Another idea that I believe would allow me to perform better in future tasks is to utilize the GitHub and it’s repository as for first hand I’ve realized how useful it is to keep track of your code and also another storage for previous codes as it is natural for programmers to get lost in their implementations and modifications of their code, keeping tracks is something I’ve realized is essential in programming. Having a sharper and clearer mindset towards the project would show better work ethic overall resulting into deliver the best work.

(

C++

We are creating a system.

Use Case Diagram(s) = Person, Book, Members.

Class Diagram

Activity Diagram

Git Repository

Implement Classes

In this project I will be going through the Library System Management I have constructed with the use of C++ Programming Language