

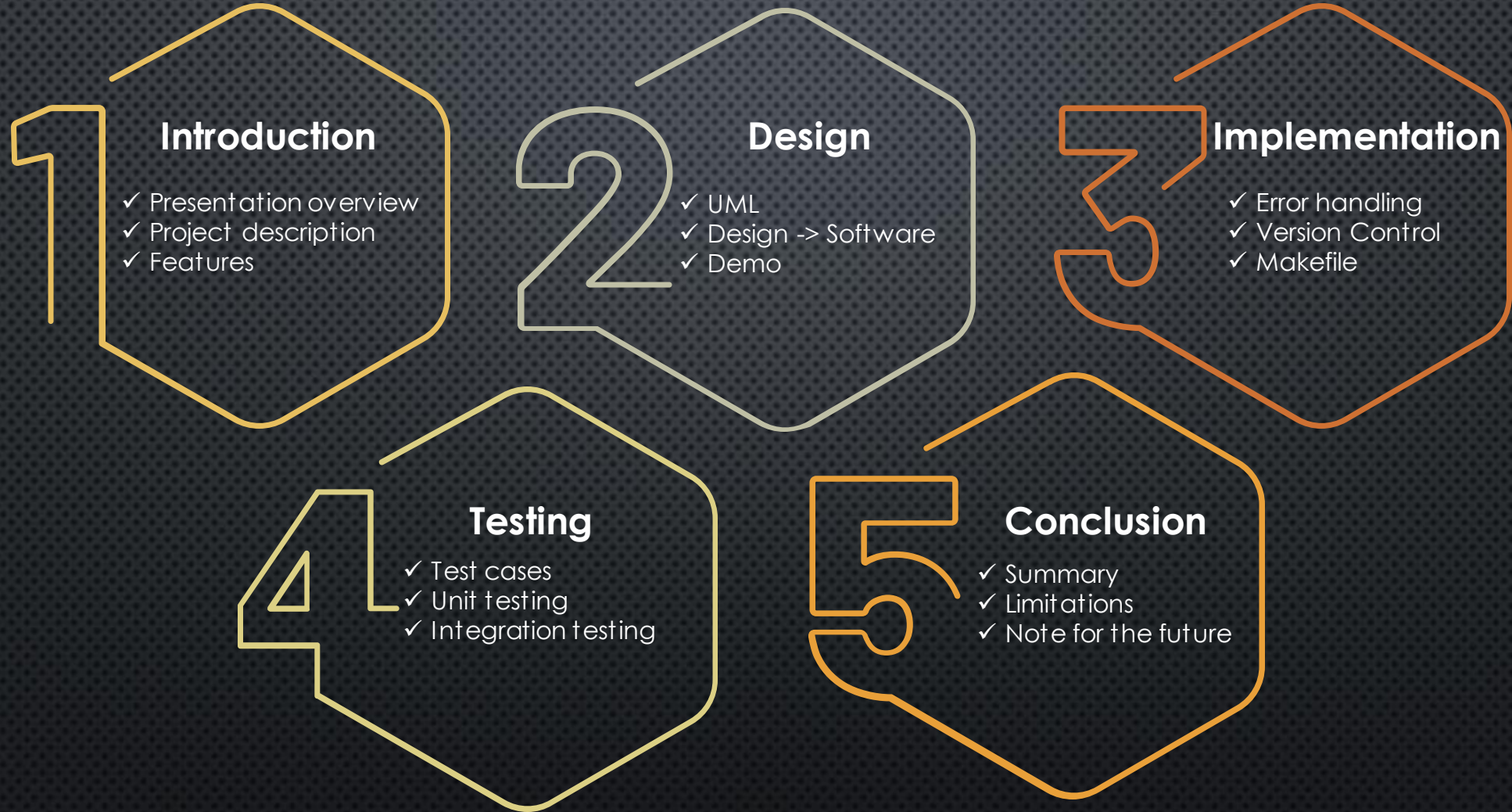


C++ LIBRARY MANAGEMENT SYSTEM

DUMITRU NIRCA

M00888146

TABLE OF CONTENTS



INTRODUCTION

THIS C++ PROJECT REPRESENTS AN EFFICIENT LIBRARY MANAGEMENT SYSTEM, PROVIDING USERS WITH A SIMPLE INTERFACE TO MANAGE BOOKS, MEMBERS AND TRANSACTIONS.

KEY FEATURES

- BOOK AND MEMBER MANAGEMENT
- BOOK ISSUING AND RETURNING
- CUSTOMIZABLE FINE AND DUE DATES
- FINE CALCULATION
- COMMAND-LINE INTERFACE

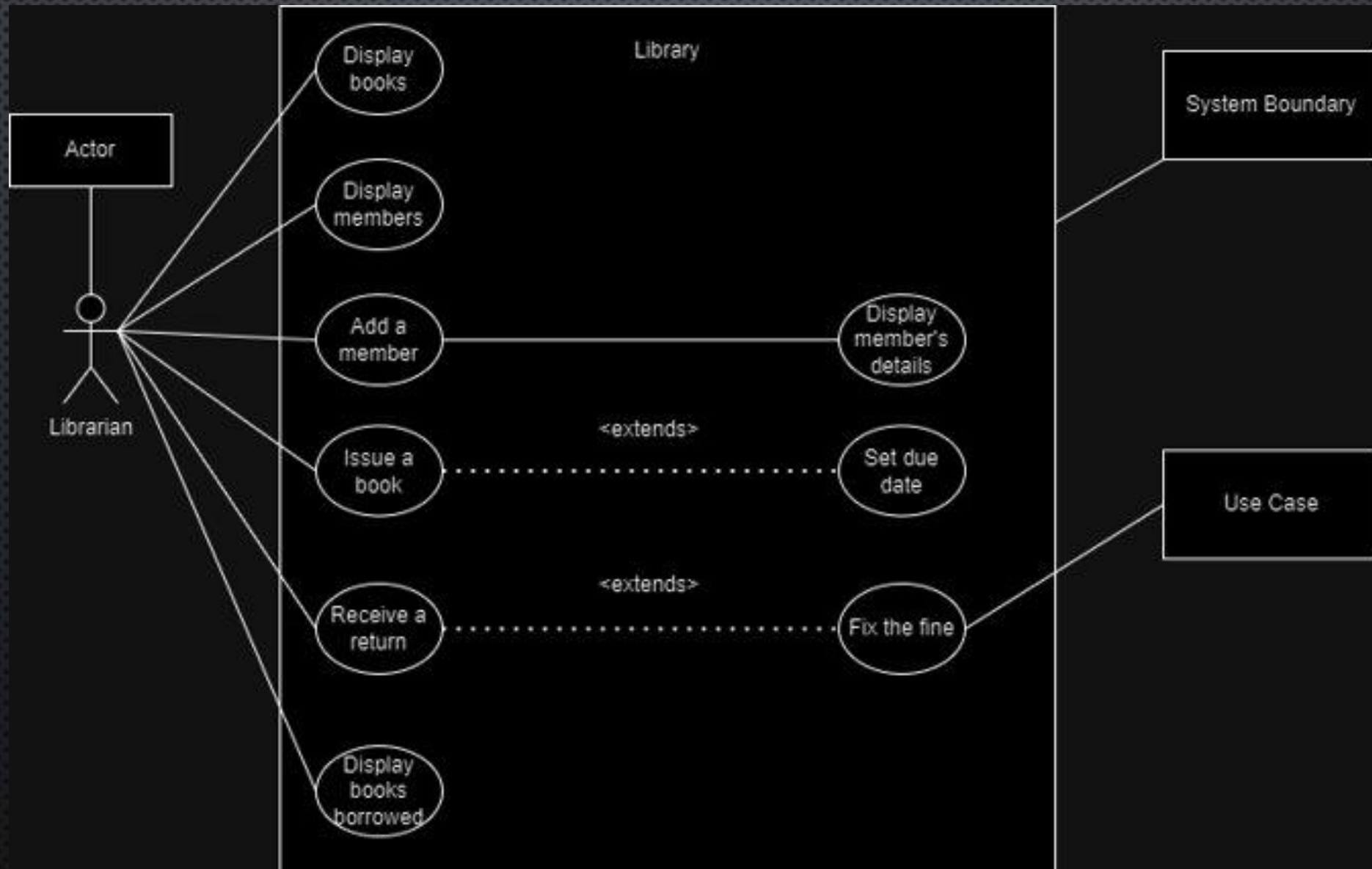


DESIGN

UML STREAMLINED DESIGN-TO-SOFTWARE TRANSLATION,
ENSURING ALIGNMENT WITH DESIGN SPECIFICATIONS.

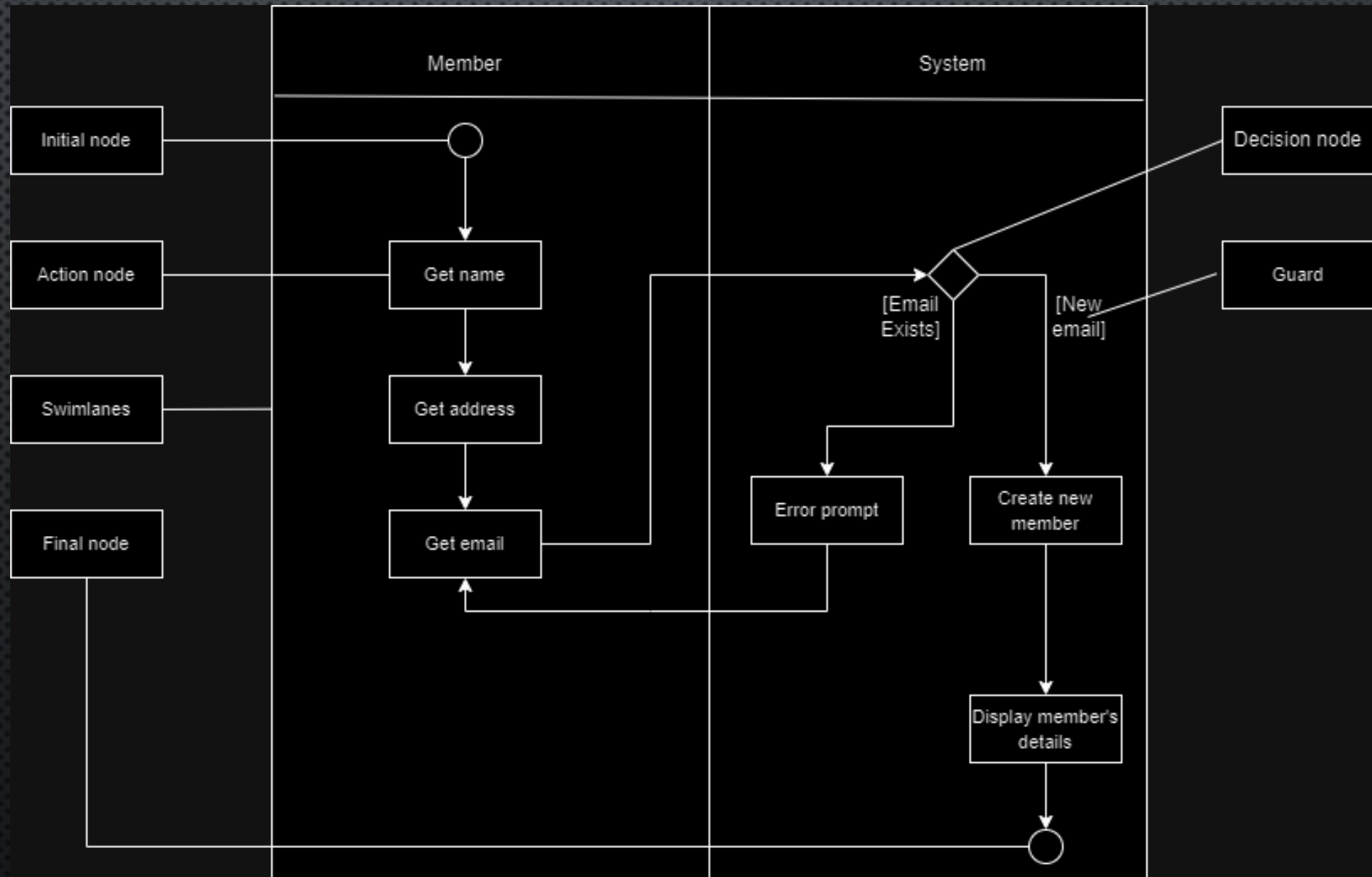


ACTIVITY DIAGRAM



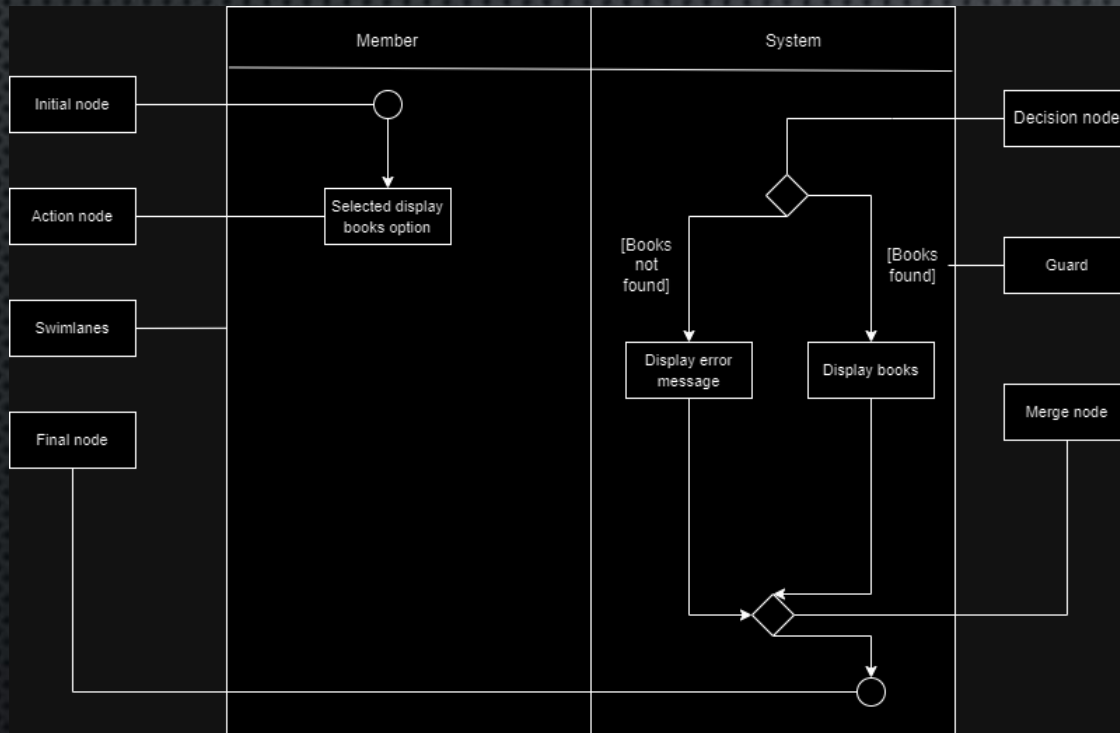
USE CASE DIAGRAM

ADD MEMBER

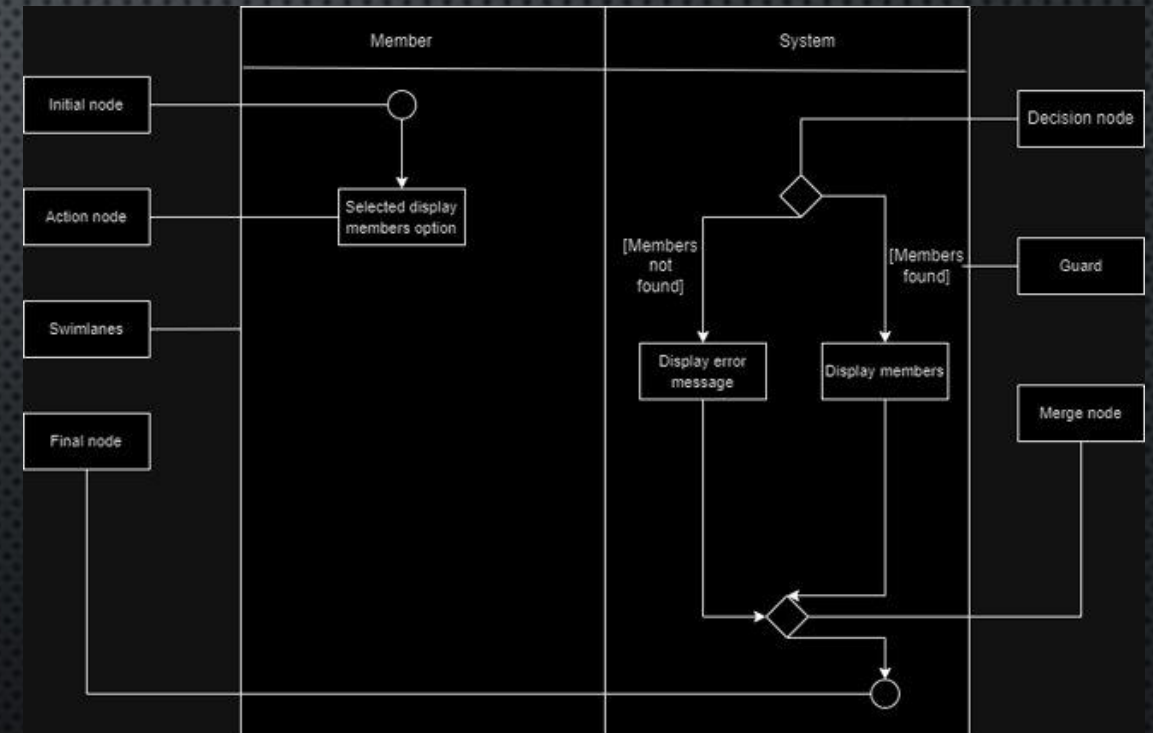


USE CASE DIAGRAMS

Display books

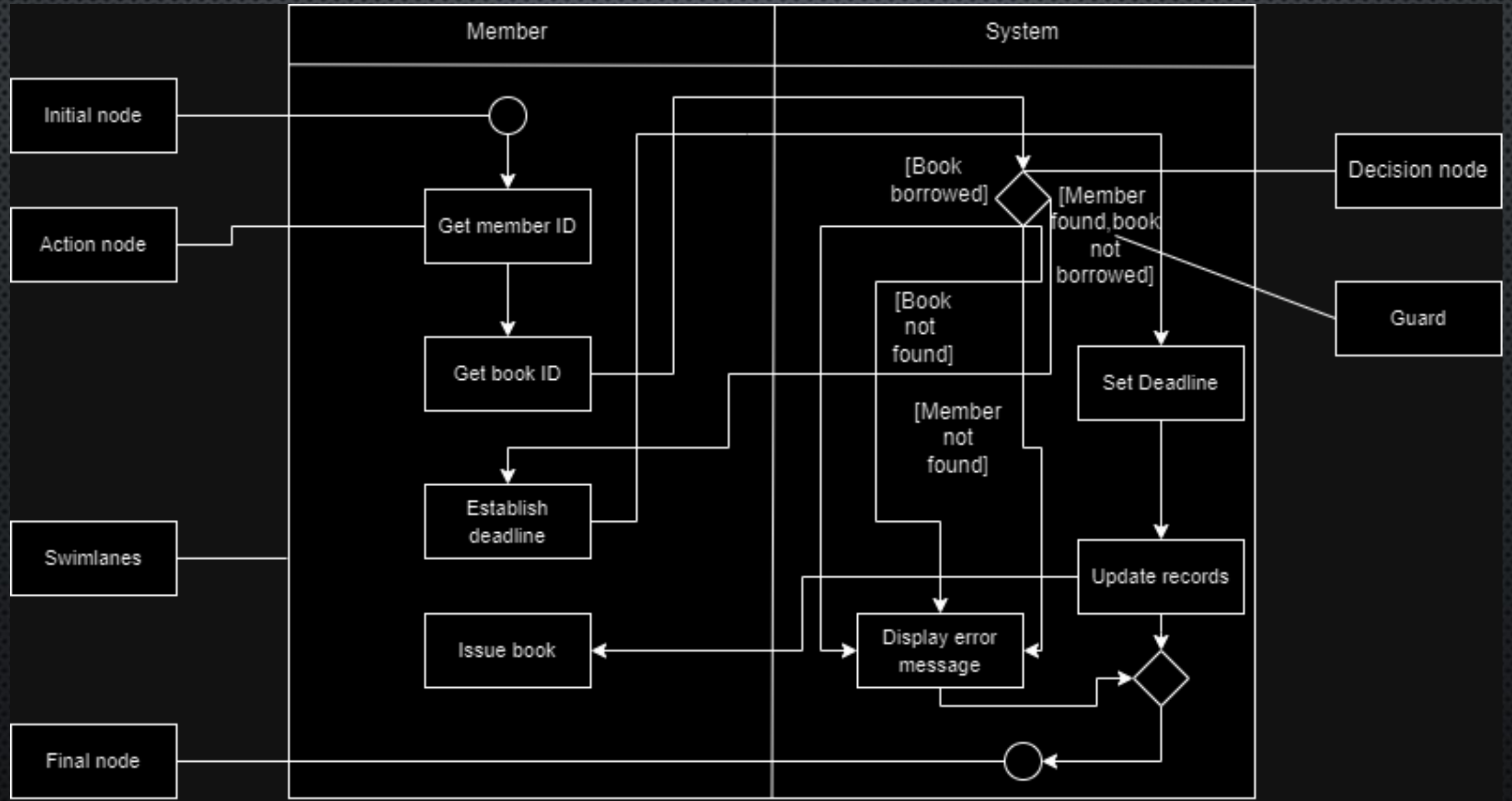


Display members



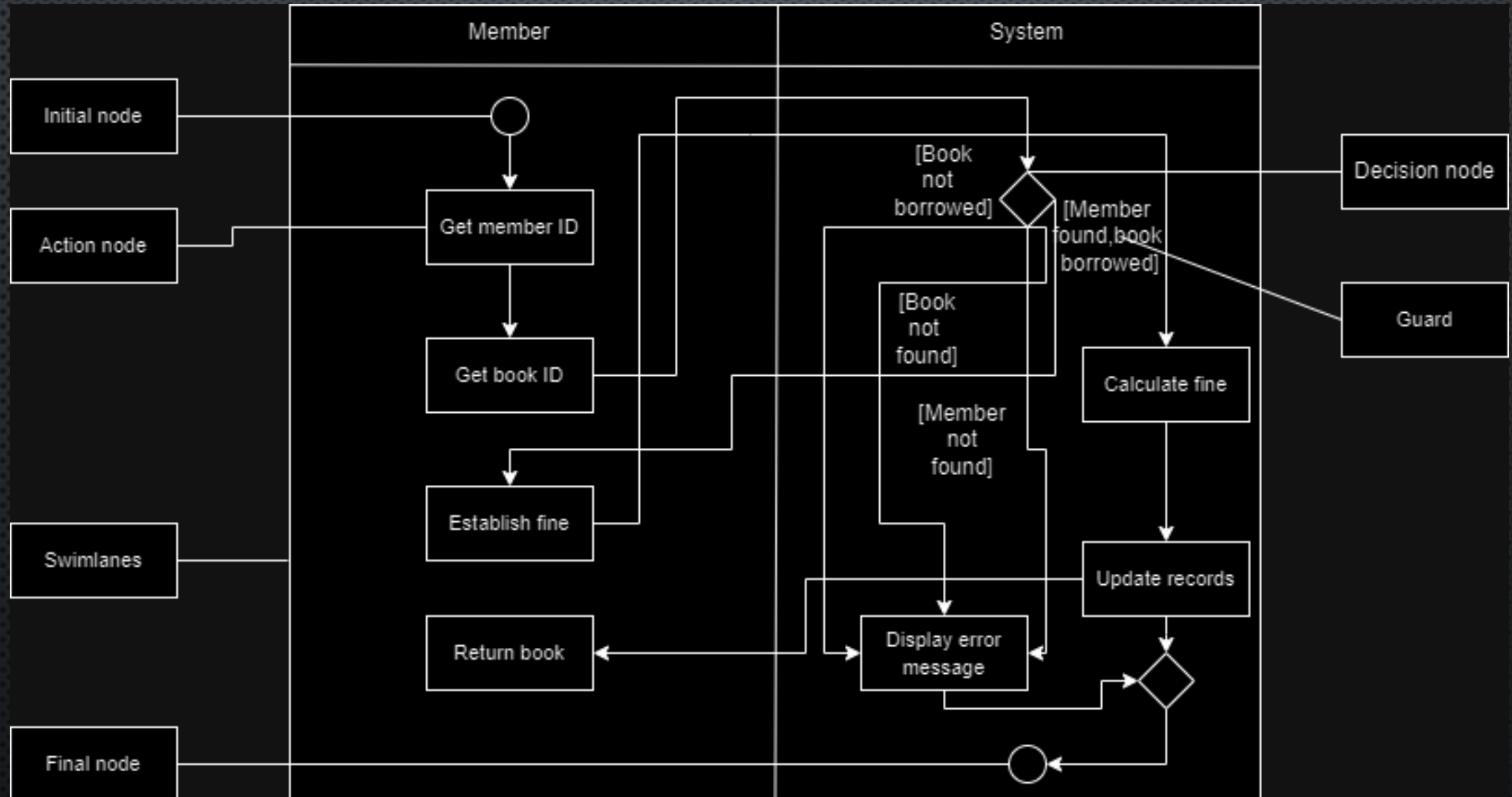
USE CASE DIAGRAM

ISSUE BOOK



USE CASE DIAGRAM

RETURN BOOK



TRANSLATING DESIGN INTO SOFTWARE

DURING IMPLEMENTATION, I TRANSLATED CLASS DIAGRAMS INTO FUNCTIONAL CLASSES, INTRODUCED A 'DATE' CLASS FOR BOOKKEEPING, MANAGED GLOBAL VARIABLES IN A DEDICATED FILE, AND DEVELOPED A 'LIBRARY' CLASS TO SERVE AS AN API. UTILIZING A CUSTOM NAMESPACE PREVENTED CONFLICTS.



VERSION CONTROL

GIT WAS EMPLOYED TO DOCUMENT CHANGES THROUGH COMMITS, MANAGE DEVELOPMENT STAGES WITH BRANCHES, AND OFFER FLEXIBILITY FOR REVERTING CHANGES WHEN NEEDED.




MAKEFILE

The Makefile automated software building, handled compilation for main and test programs, and managed project cleanup, ensuring an organized structure.




COMMIT HISTORY


Created and Tested Date Class

 Masturbino69 committed 3 weeks ago

6417742  


 Commits on Dec 18, 2023

Merge branch 'master' into Librarian

 Masturbino69 committed 3 weeks ago


ef8d451  

Renamed Librarian.cpp~ to Librarian.cpp

 Masturbino69 committed 3 weeks ago


945c462  

Created Librarian Subclass, Tested Default and Parameterized Constructors

 Masturbino69 committed 3 weeks ago


2c6e039  

Implemented Person Base Class and Tested it

 Masturbino69 committed 3 weeks ago
















07ccbd4  

Initial commit

 Masturbino69 committed 3 weeks ago


49abd7c  


COMMIT HISTORY

Modified Librarian Class  Masturbino69 committed 2 weeks ago	fb531d9		
Modified Librarian Class  Masturbino69 committed 2 weeks ago	84fc724		
Added countDays function in Date Class  Masturbino69 committed 2 weeks ago	5945ac9		
Decided to stick with normal pointers  Masturbino69 committed 2 weeks ago	7bcf213		
 Commits on Dec 23, 2023			
Implementing Book Class  Masturbino69 committed 2 weeks ago	87c9bfa		
 Commits on Dec 22, 2023			
Implemented Member Functions  Masturbino69 committed 2 weeks ago	3f9e2c9		
 Commits on Dec 21, 2023			
Implemented Smart Pointers in Book Class  Masturbino69 committed 2 weeks ago	5f3e59f		
Continued implementing Book Class  Masturbino69 committed 2 weeks ago	00fc1ab		
 Commits on Dec 20, 2023			
Committing changes before merge  Masturbino69 committed 2 weeks ago	ce955d5		


COMMIT HISTORY

Finished Testing All Classes


 Masturbino69 committed last week


 Commits on Dec 28, 2023

Finished Implementing All Classes


 Masturbino69 committed last week

Finished Implementing All Classes


 Masturbino69 committed last week

 Commits on Dec 26, 2023


Delete test_main.exe.stackdump


 Dima336l committed 2 weeks ago

Delete .clang-format.


 Dima336l committed 2 weeks ago

Continued Working on Librarian Class, Did Further Testing


 Masturbino69 committed 2 weeks ago

 Commits on Dec 25, 2023


Merge branch 'master' into Librarian ...

 Masturbino69 committed 2 weeks ago


Merging master with Librarian

 Masturbino69 committed 2 weeks ago


Merge branch 'Date' ...


 Masturbino69 committed 2 weeks ago

Continued Implementing Librarian Class


 Masturbino69 committed 2 weeks ago

Optimized suboptimal code


 Masturbino69 committed 9 hours ago


 Commits on Jan 5, 2024

Continued working on library wrapper class


 Masturbino69 committed yesterday


Added Makefile

 Masturbino69 committed yesterday


 Commits on Jan 4, 2024

Diagrams modified


 Masturbino69 committed 2 days ago


 Commits on Dec 31, 2023

Finished implementing the library


 Masturbino69 committed last week

Modified input handling to not break of the loop


 Masturbino69 committed last week

 Commits on Dec 30, 2023


Implementing the menu

 Masturbino69 committed last week


Implementing setting the date in the constructor

 Masturbino69 committed last week

Added display books function, started implementing the menu

 Masturbino69 committed last week

Started working on main file, Implemented Library class to coordinate the rest of classes

 Masturbino69 committed last week

TESTING

The C++ library project underwent testing using a dual approach: unit tests targeted specific components while integration tests ensured seamless collaboration between classes.



TEST CASES

The tests verified the functionality of constructors, getters, and setters. They also validated exception handling, proper management of invalid input and method functionality, with a focus on interactions between classes in the system.



```
Dumitru@Dumitru ~/Coursework1
$ ./Test_main
=====
All tests passed (91 assertions in 5 test cases)
```

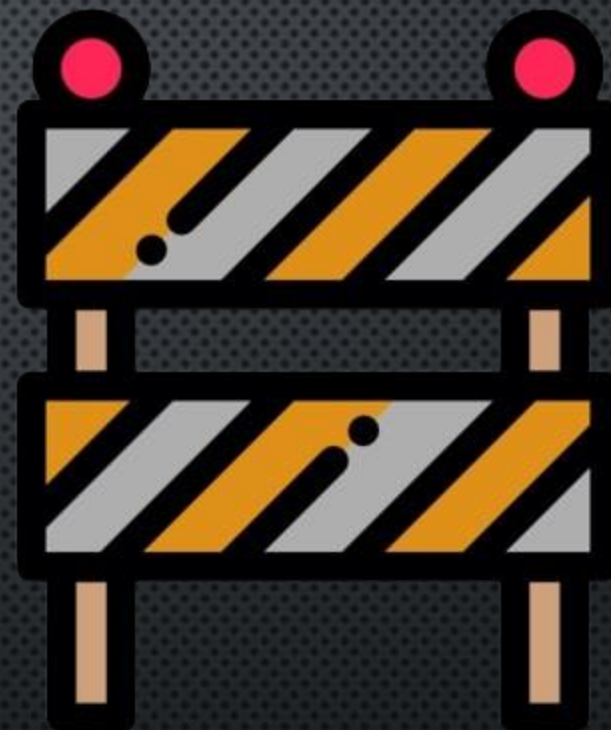

SUMMARY

The C++ library project represents a reliable system, tested with Catch2, and version controlled using Git. Emphasizing a modular design and core Object-Oriented Programming (OOP) principles ensured a clear separation of concerns. Following a structured Software Development Life Cycle contributed to the project's overall success.



LIMITATIONS

The project encounters challenges in scalability with large datasets and adapting to evolving requirements. While Catch2 is effective for testing, it may have limitations in complex scenarios. Relying on the terminal hinders non-technical users, and assuming a specific CSV format adds another limitation.



FUTURE APPROACH

For future projects, I plan to prioritize clear code documentation, implement automation for efficient testing, explore web interfaces to broaden accessibility, delve into design patterns, consider diverse data structures, and enhance commit messages for improved collaboration.



THANK YOU
FOR YOUR ATTENTION

