REQUIREMENT ANALYSIS DOCUMENT : UNIPA PROJECT 2 LIBRARY MANAGEMENT SYSTEM

- ## Purpose

This requirement analysis document aims to give information about Library Management System's functional and non-functional requirements, and explains how the system behaves. Stated ideas will be supported using system models such as use-case diagrams, E/R diagrams, class diagrams etc. Ultimate purpose for this document will be giving a general understanding about how program works and what it does.

- ## Audience

The audience for this RAD includes my responsibles from both my university, Izmir University of Economics and my place of internship, UNIPA. However, anyone else who wants to learn about the program is welcomed.

- ## Introduction

1. **Purpose of the system:** Library Management System is designed for being helpful to the library keeper in a library. It is not possible to manage more than one library with the system but the sytem can be adapted for different libraries.

2. **Scope of the system:** This project's scope is limited to implementing the system as it is stated in the requirements list and delivering the necessary documentations. At the end of the project, all requirements will be satisfied and program shall be tested and maintained optionally.

3. **Objectives and success criteria of the project:**In this particular project, delivering corresponding documentations and meeting the system requirements are the keys of the success criteria. Except the functional requirements, my ultimate objective will be delivering a decent library management system which can be used and adapted for other libraries, within the circumstances.

- ## Requirements

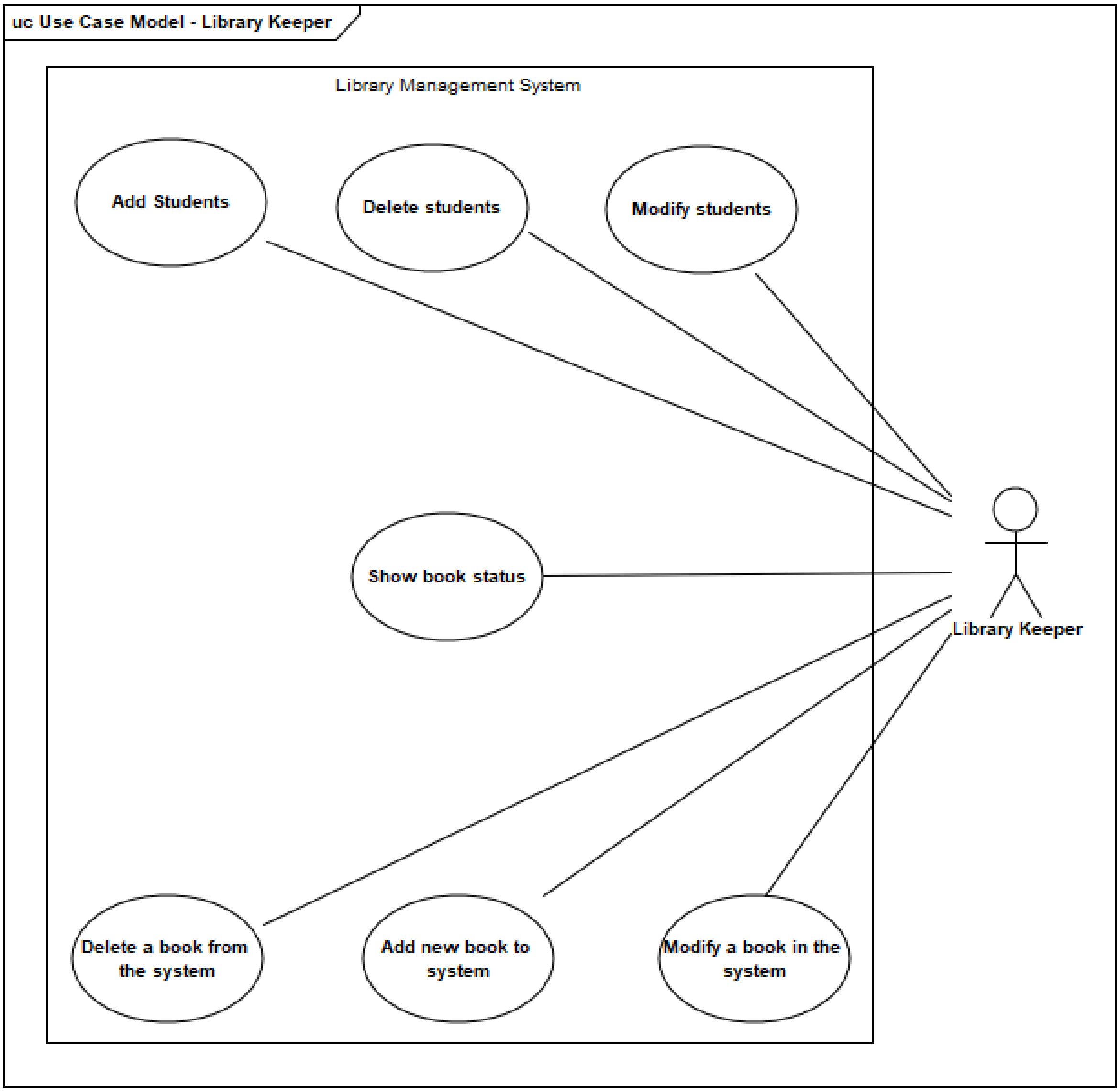
As it is shown in the below table, functional requirements rely on the needs of students and library keeper. Non-functional requirements of Library Management System concerns usability, maintainability and performance.

- **System Models**

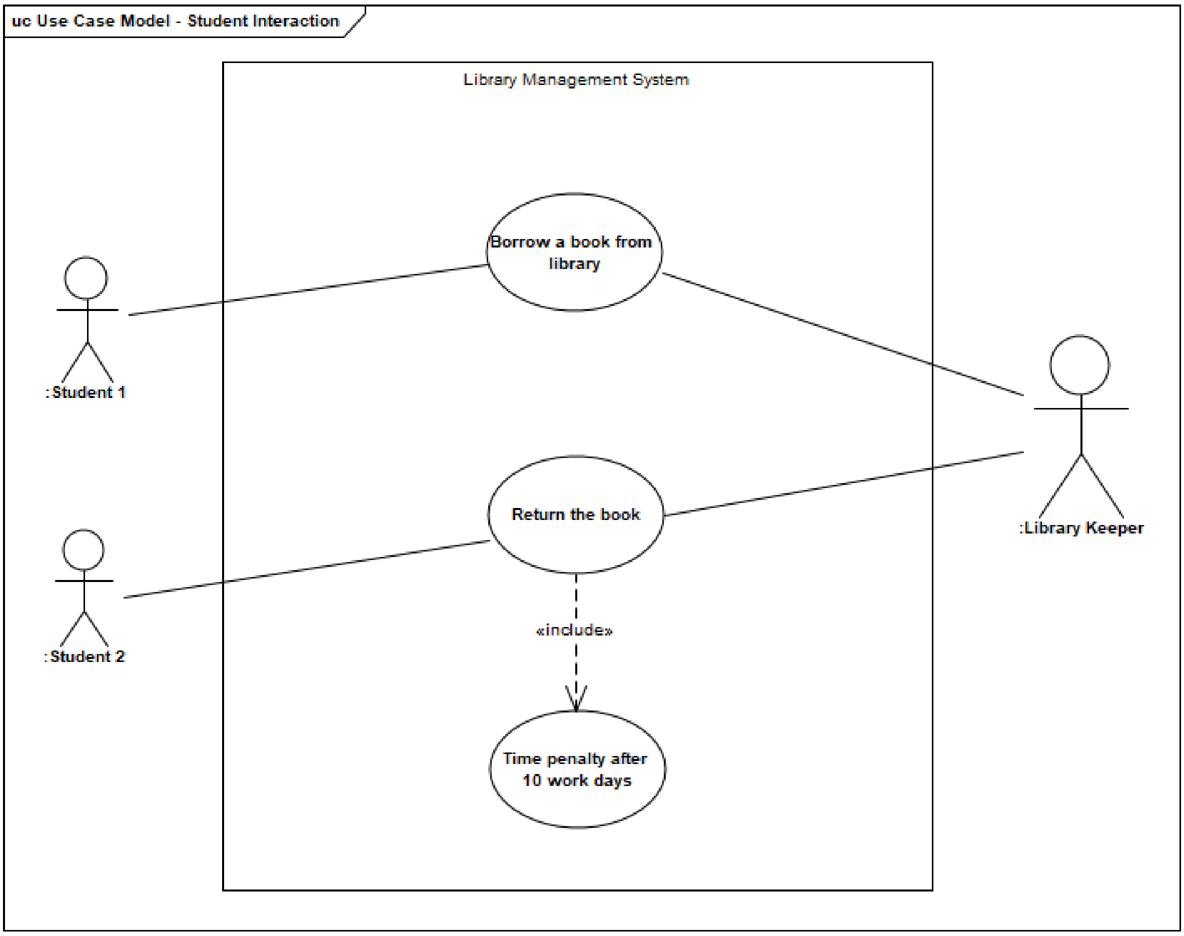
System models are implemented on Enterprise Architect. This section of the document contains Use Case Diagrams, Class Diagrams, E-R diagrams and GUI Design

Use Case Diagrams: There are two different use case diagrams, one is about the only library keeper's functionalities, the other use case focuses on library keeper-student relationship (Borrow/Return Books)

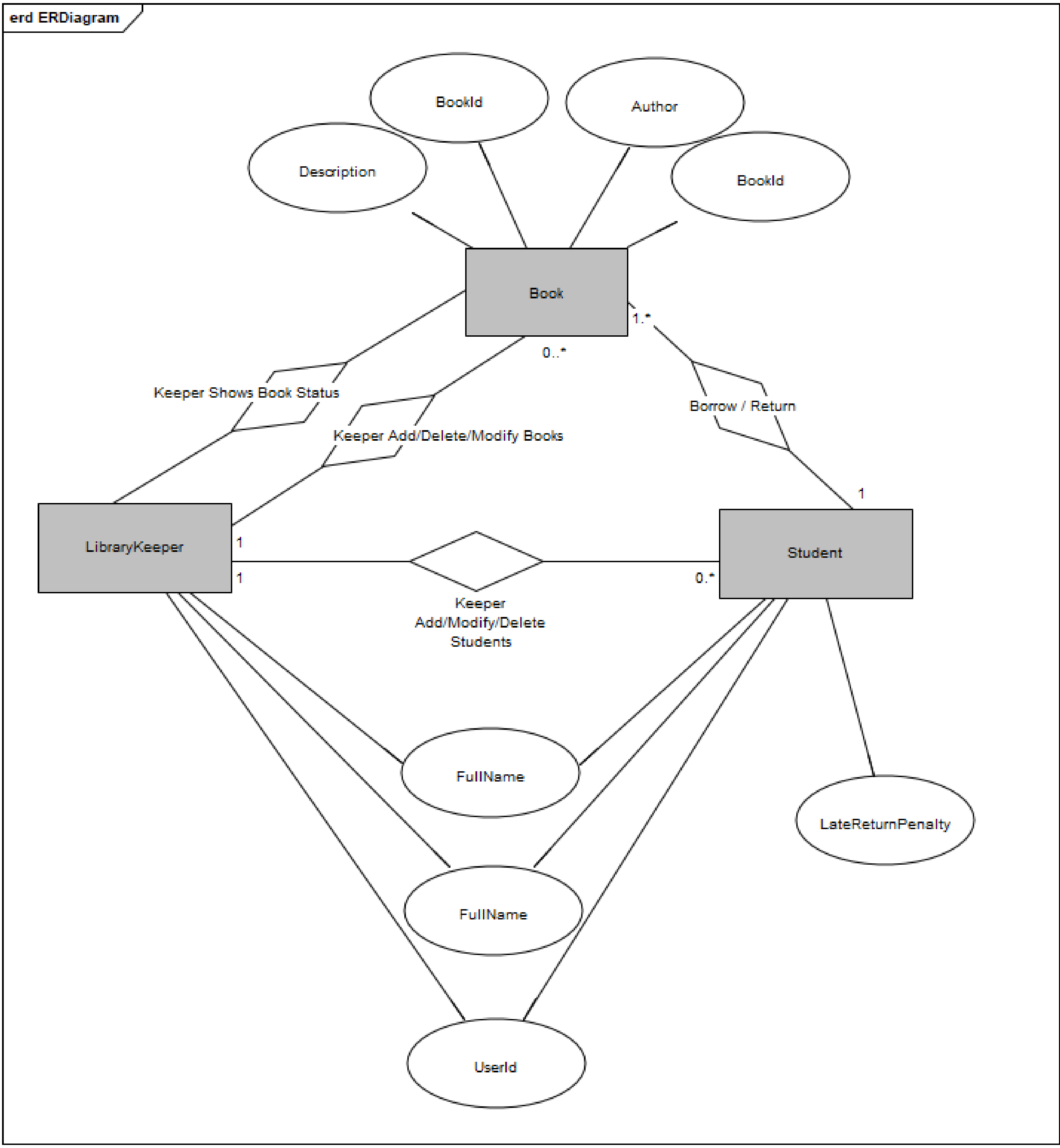
Below use case demonstrates library keepers capabilities in the system


uc Use Case Model - Library Keeper
Library Management System
Add Students
Delete students
Modify students
Show book status
Delete a book from the system
Add new book to system
Modify a book in the system
Library Keeper

Below use case diagram shows relationships between students and library keeper in library management system

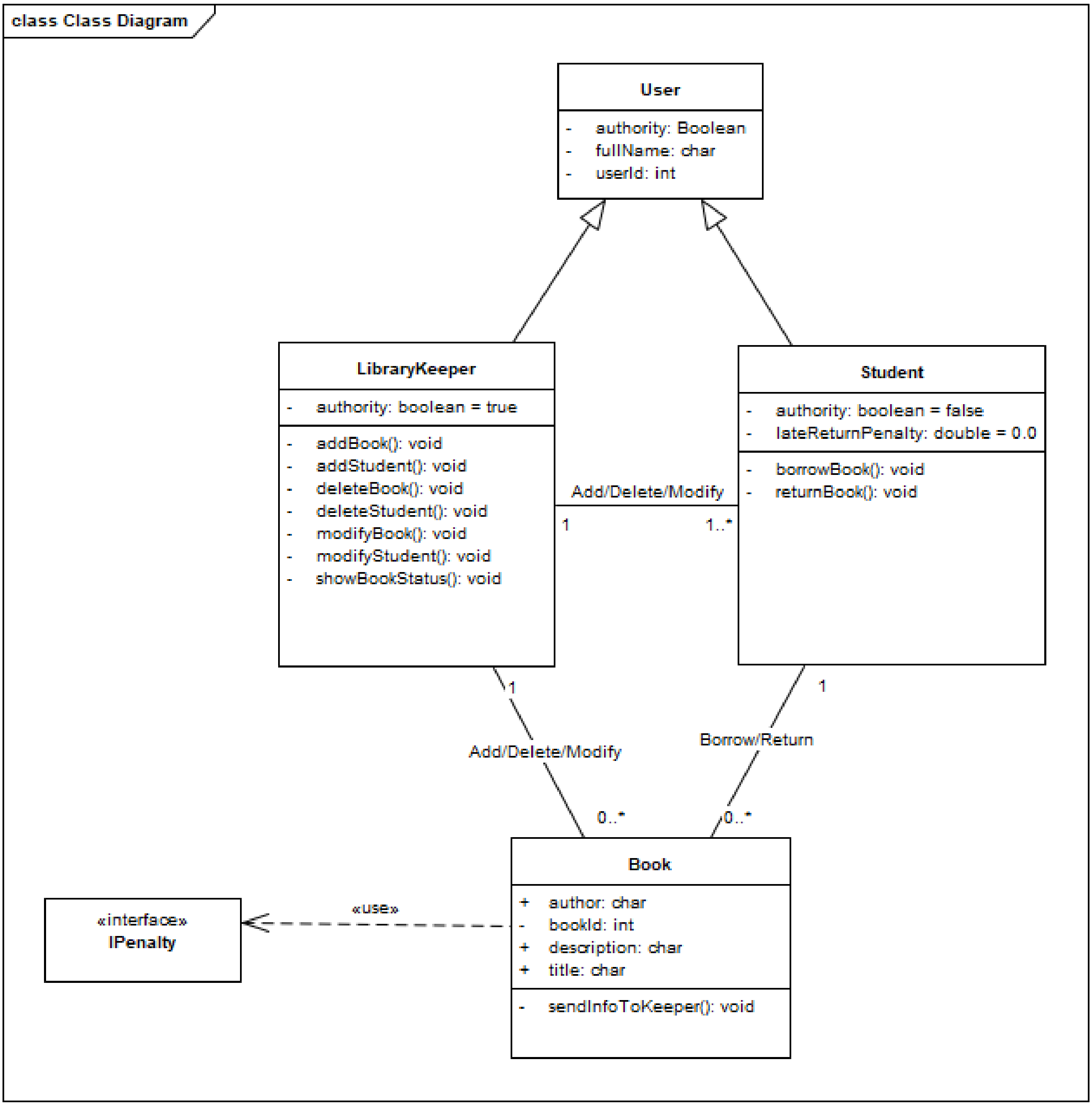

uc Use Case Model - Student Interaction
Library Management System
Borrow a book from library
Return the book
«include»
Time penalty after 10 work days
: Student 1
: Student 2
:Library Keeper

E-R Diagram: Here is the E-R diagram which shows the relationship between the entities of the system. On the other hand, it is a good reference for Library Management System's Database Design, and gives a general understanding of whole system.



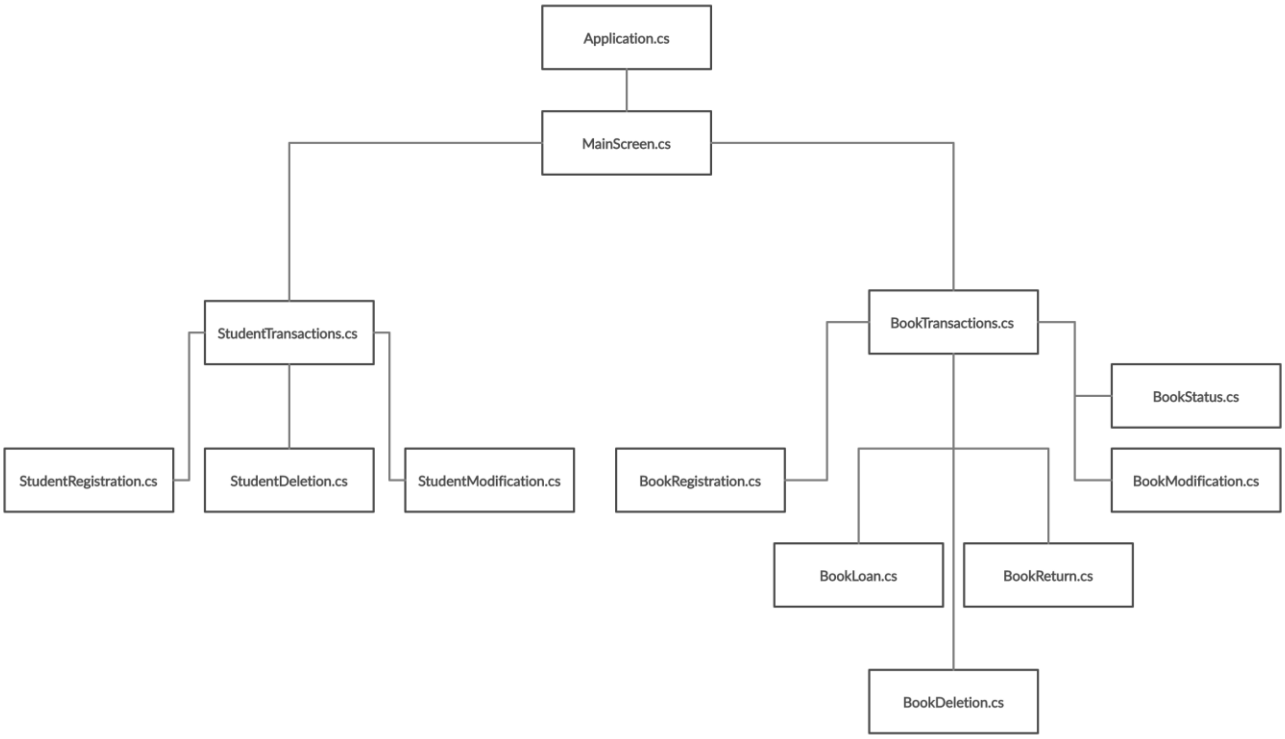
erd ERDiagram
BookId
Author
Description
BookId
Book
1.*
0..*
Keeper Shows Book Status
Keeper Add/Delete/Modify Books
Borrow / Return
1
LibraryKeeper
1
1
Student
0.*
Keeper
Add/Modify/Delete
Students
FullName
LateReturnPenalty
FullName
UserId

Class Diagram: This section contains Class Diagram of Library Management System. Having a class diagram is important for both understanding the program and programming Object-Oriented code.



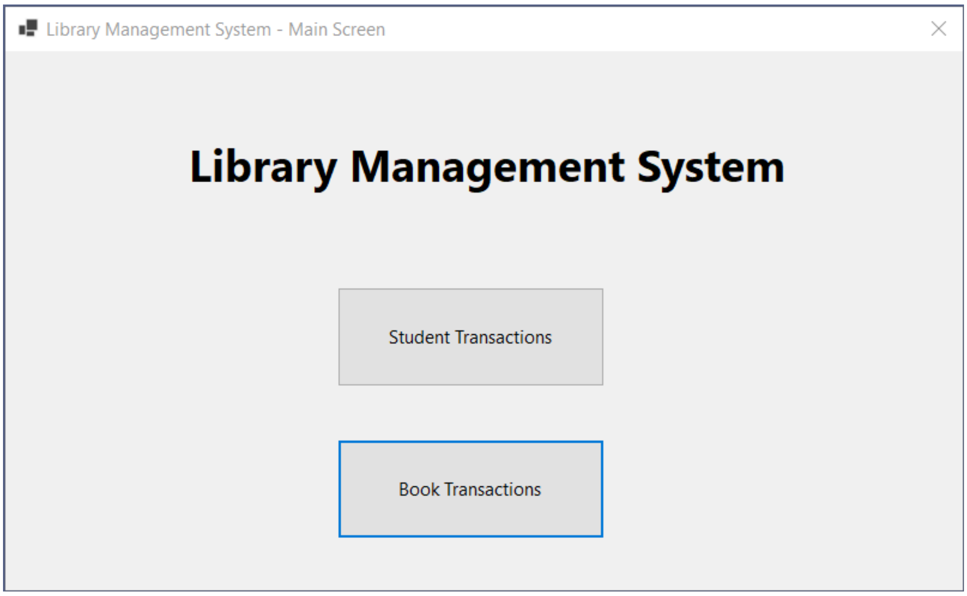
Graphical User Interface Design: GUI of the system designed for giving a simple user experience. Each window in the program is a ***Singleton object*** and if one of the Windows Forms is closed, the one up on the GUI hierarchy opens again. Application closes when MainScreen object is closed. <u>BookTransaction</u> and <u>StudentTransaction</u> screens might be change depending on the needs.

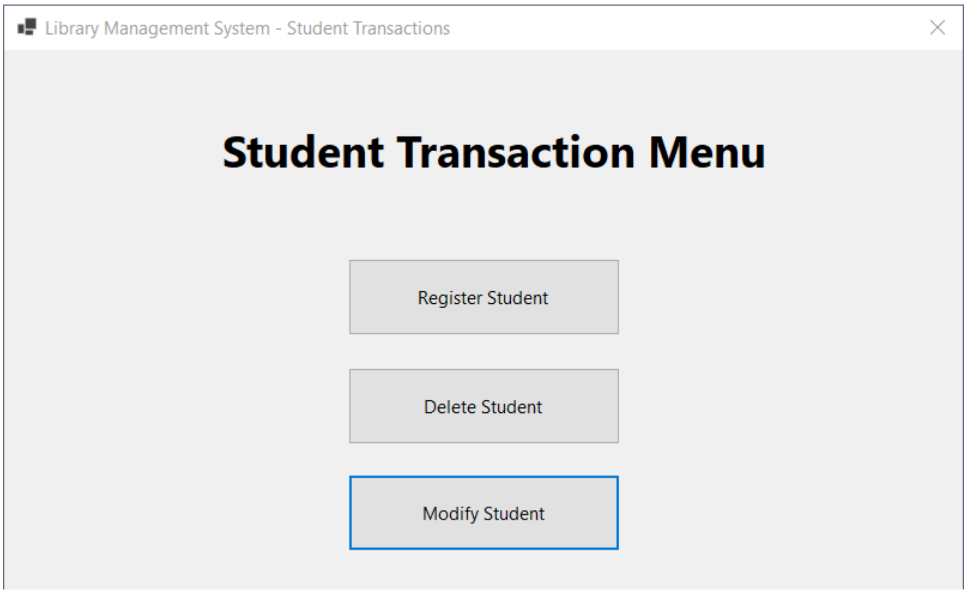
GUI Hierarchy

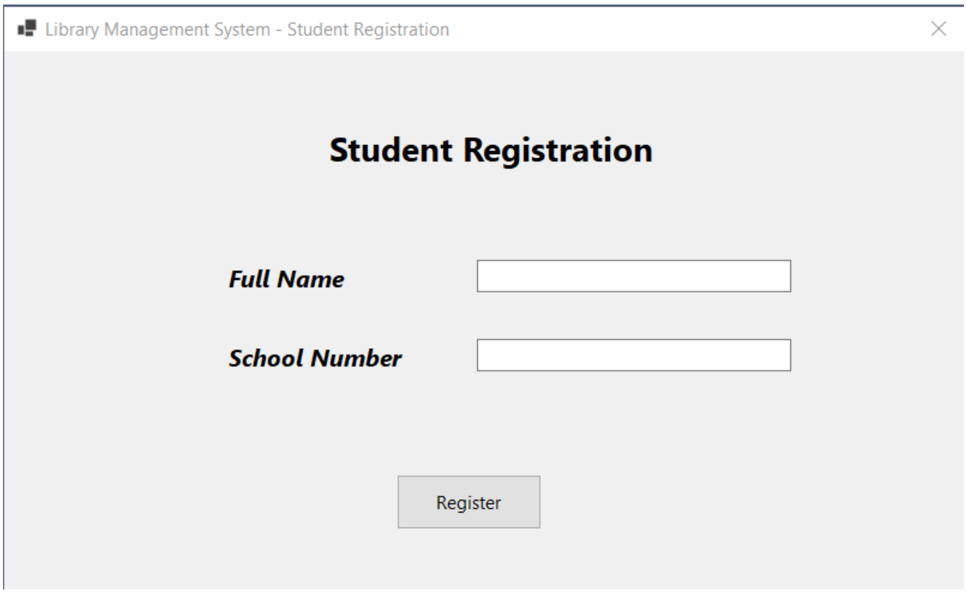


GUI Design

All Windows Forms are initialized unresizable and initialized at center screen.

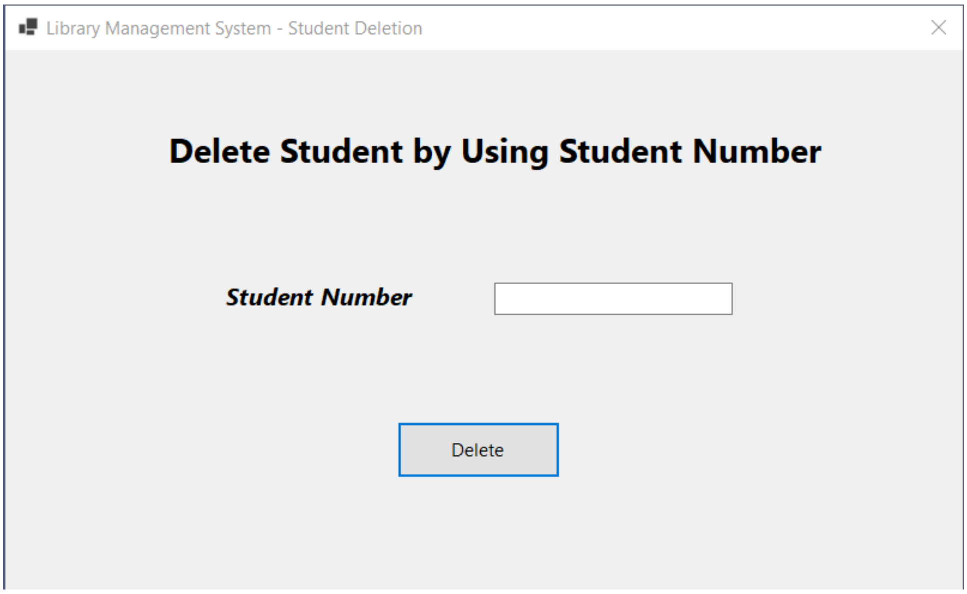

Library Management System - Main Screen
Library Management System
Student Transactions
Book Transactions


Library Management System - Student Transactions
Student Transaction Menu
Register Student
Delete Student
Modify Student



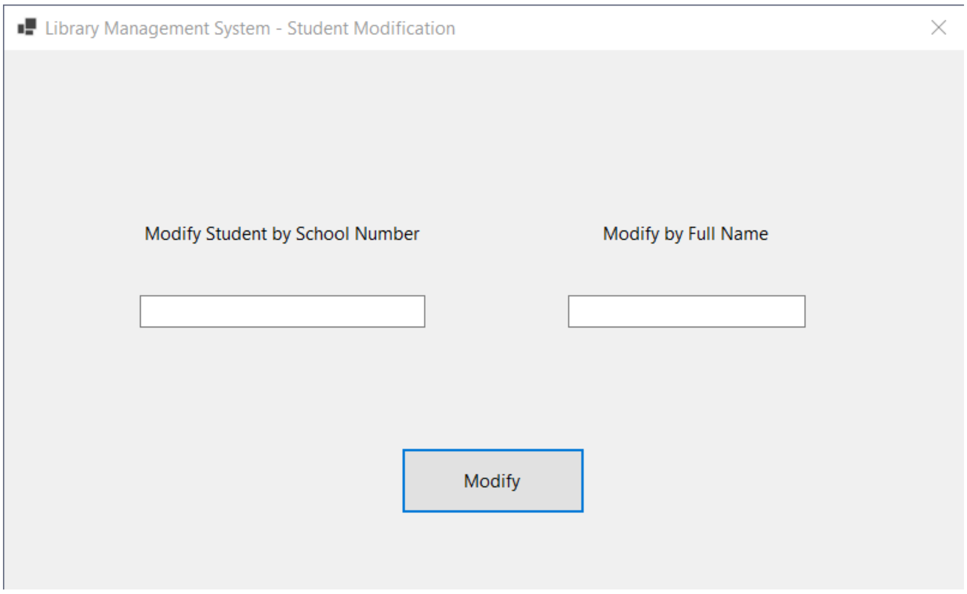
Library Management System - Student Registration

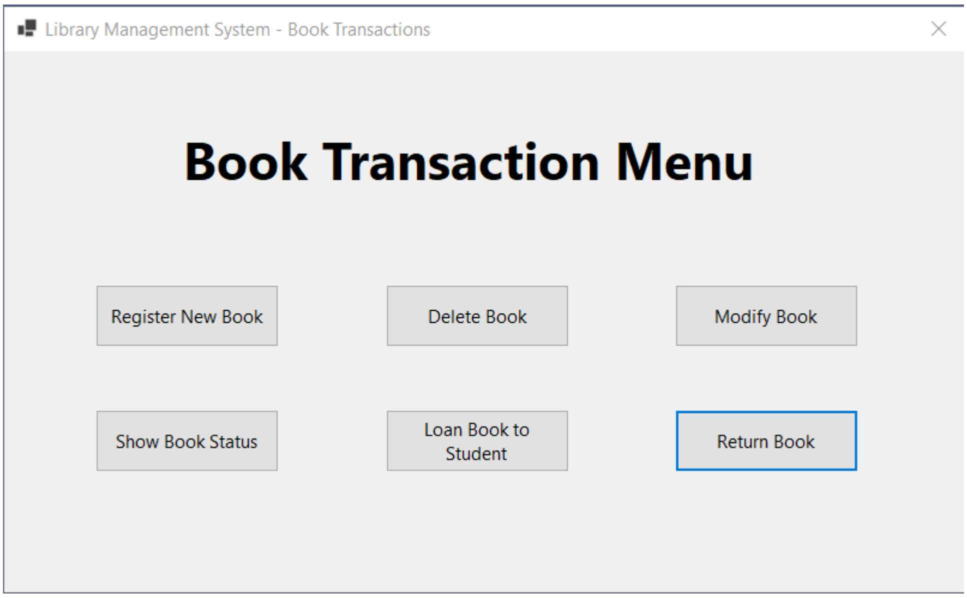
Student Registration

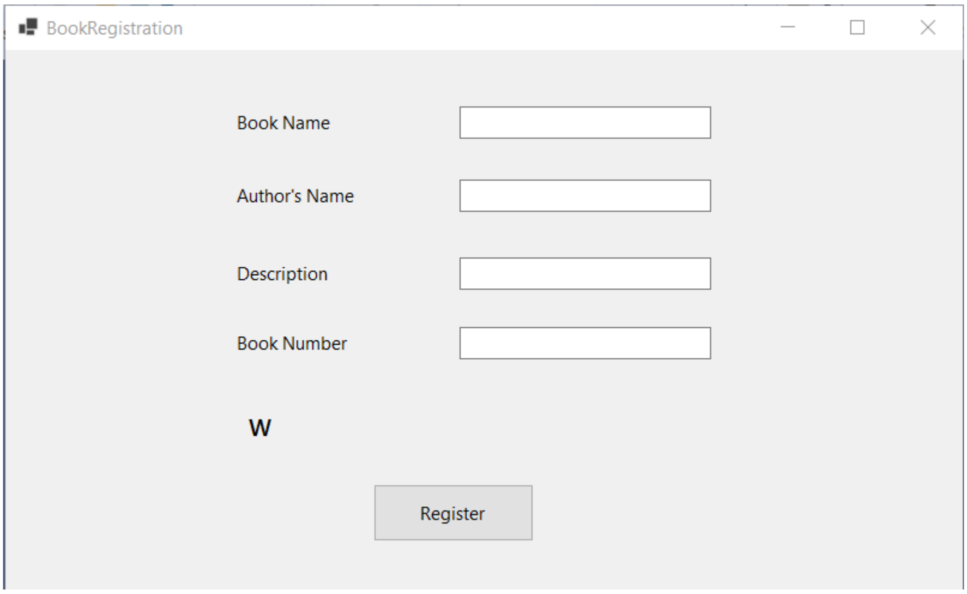


Library Management System - Student Deletion

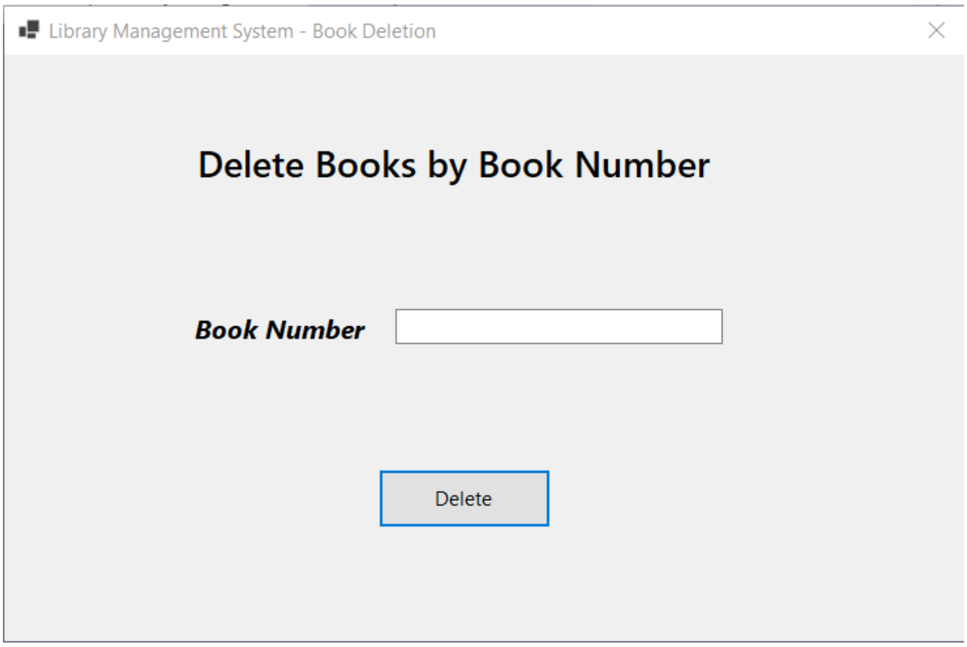
Delete Student by Using Student Number

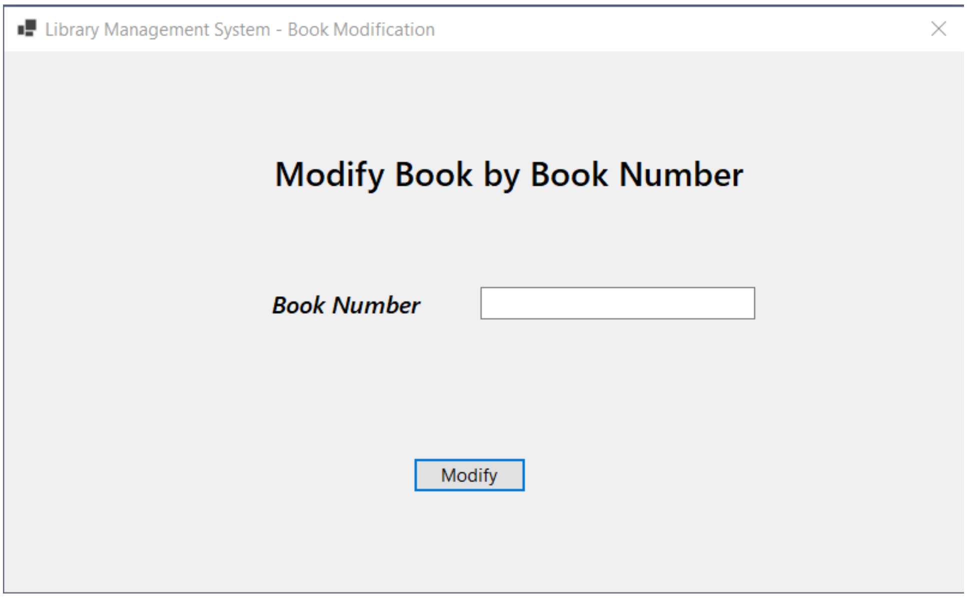

Library Management System - Student Modification
Modify Student by School Number
Modify by Full Name
Modify


Library Management System - Book Transactions
Book Transaction Menu
Register New Book
Delete Book
Modify Book
Show Book Status
Loan Book to Student
Return Book



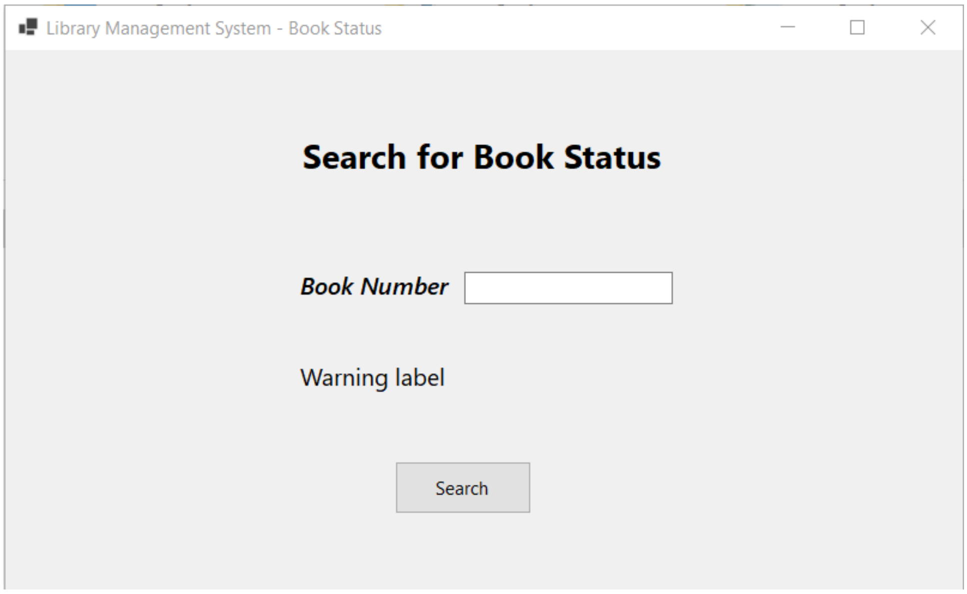
“W” stands for “warning” Label. If one of the information in a TextBox is false or empty, Label will be modified and show what is wrong to the user. Label initiliazed with no text, therefore it is invisible until the user does something wrong. I put “W” for demonstration purposes. Also there are other warning labels in other Form’s, but they initialized with no text, like it is mentioned.


Library Management System - Book Deletion
Delete Books by Book Number
Book Number
Delete



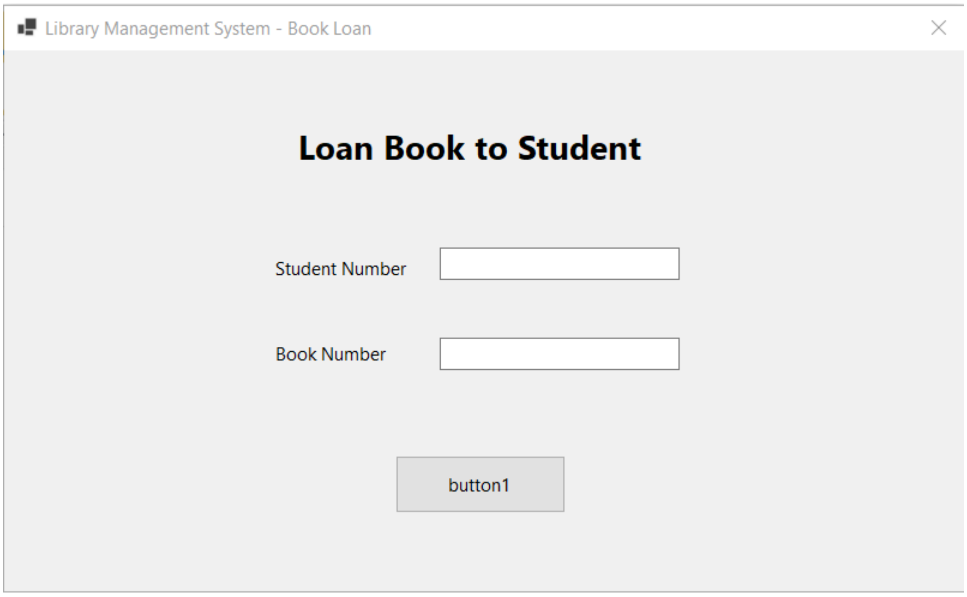
Library Management System - Book Modification

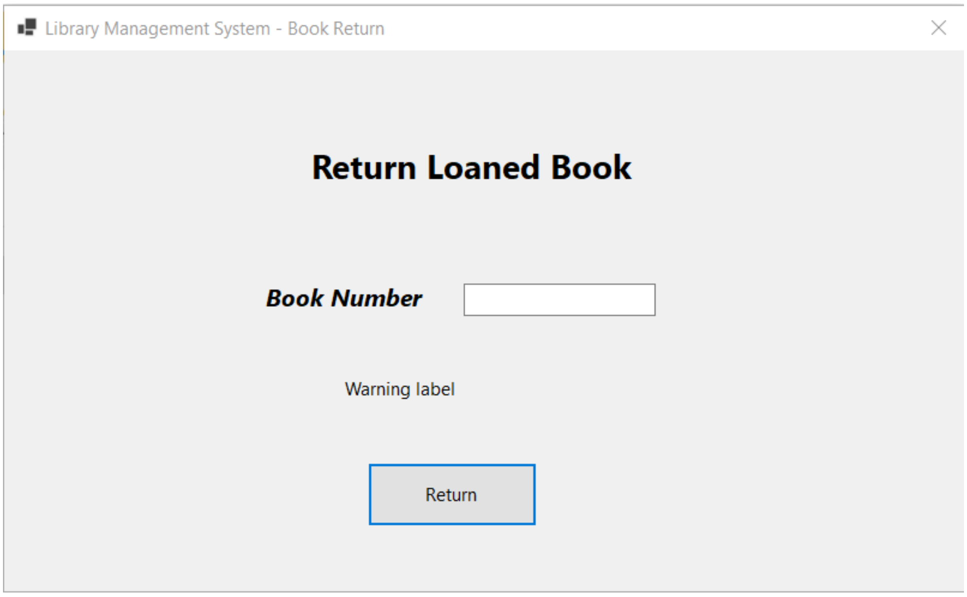
Modify Book by Book Number



Library Management System - Book Status

Search for Book Status



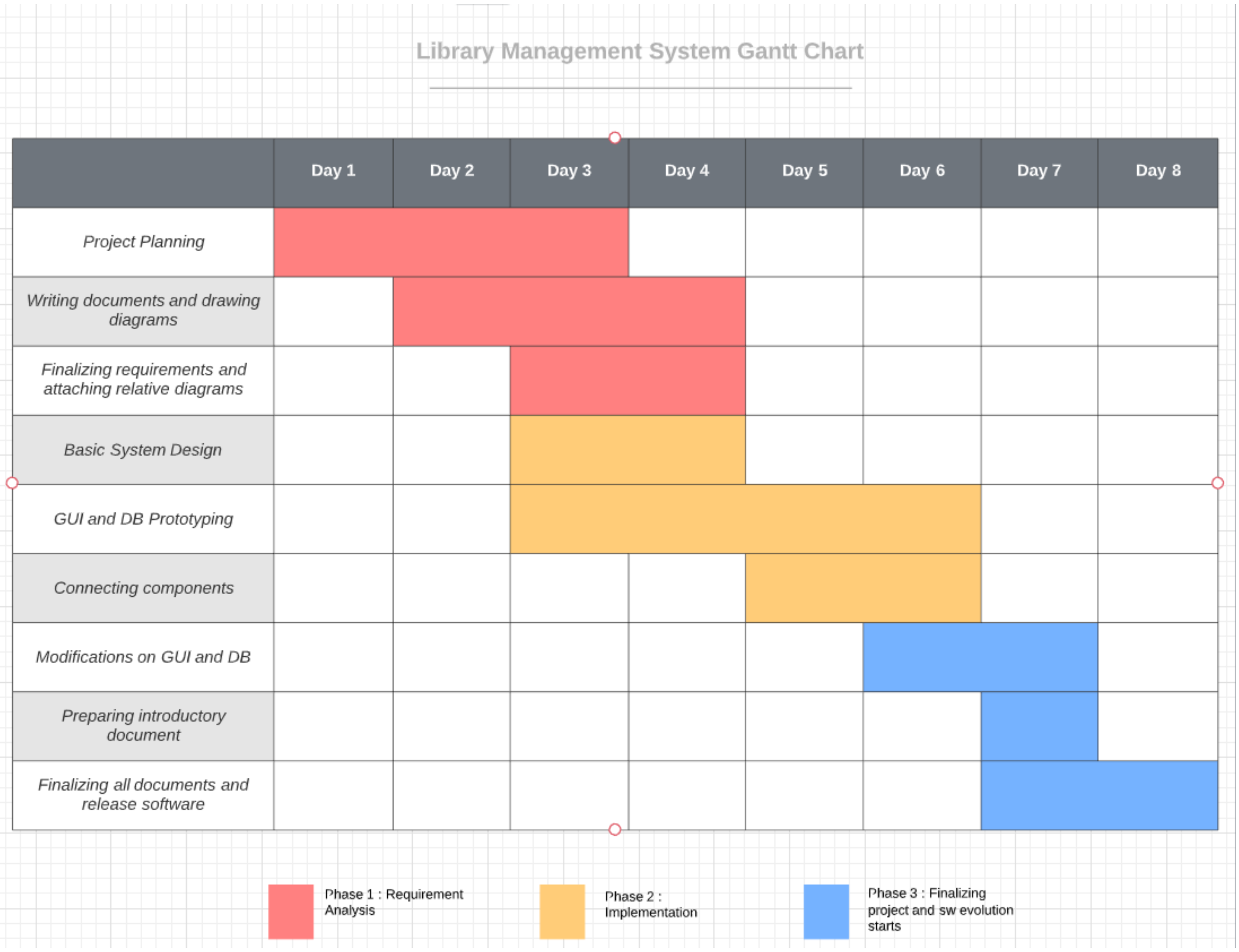


Following modifications on GUI will be denoted on introductory software program.

- **Workflow**

For such a limited time for a project, I picked incremental prototyping as my development method. In incremental prototyping, the final product is decimated into different small prototypes and developed individually. Eventually, the different prototypes are merged into a single product. This method is helpful to reduce the feedback time between the user and the application development team. (https://www.guru99.com/software-engineering-prototyping-model.html)

Below Gantt Chart indicates my work plan in eight work days.



Library Management System Gantt Chart

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
Project Planning	🟥	🟥	🟥					
Writing documents and drawing diagrams		🟥	🟥	🟥				
Finalizing requirements and attaching relative diagrams			🟥	🟥				
Basic System Design			🟧	🟧				
GUI and DB Prototyping			🟧	🟧	🟧	🟧		
Connecting components					🟧	🟧		
Modifications on GUI and DB						🟦	🟦	
Preparing introductory document							🟦	
Finalizing all documents and release software							🟦	🟦

🟥 Phase 1 : Requirement Analysis

🟧 Phase 2 : Implementation

🟦 Phase 3 : Finalizing project and sw evolution starts