HACETTEPE UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

BBM 487 SOFTWARE ENGINEERING LAB

SOFTWARE REQUIREMENTS

Özdeş Öztürk
21228635

Ekin Kandemir
21228434

Kadir Bulut
21228137

Library Book Loan System(LBLS)
System-Wide Requirements Specification

1. Introduction

This document states the basic requirements of Library Book Loan System clearly in order to define the details of the project to any user and make these details understandable. Our reference for this documentation is Eclipse Process Framework(EPF)/Open Unified Process (OpenUP)/ System Wide Requirements Specification template since the whole content and the headings are shaped according to this document. In the following sections, we will be giving functional requirements and will be stating these requirements as detailed as possible. These key details will consist on System-Wide Functional Requirements, System Qualities including the interpretation about usability, reliability, performance and supportability of the system we will be constructing, System Interfaces(both the user interface and any external system interface), Business Rules, System Constraints (as the constraints are one of the most crucial requirements that shapes the system under construction), System Compliance and the System Documentation.

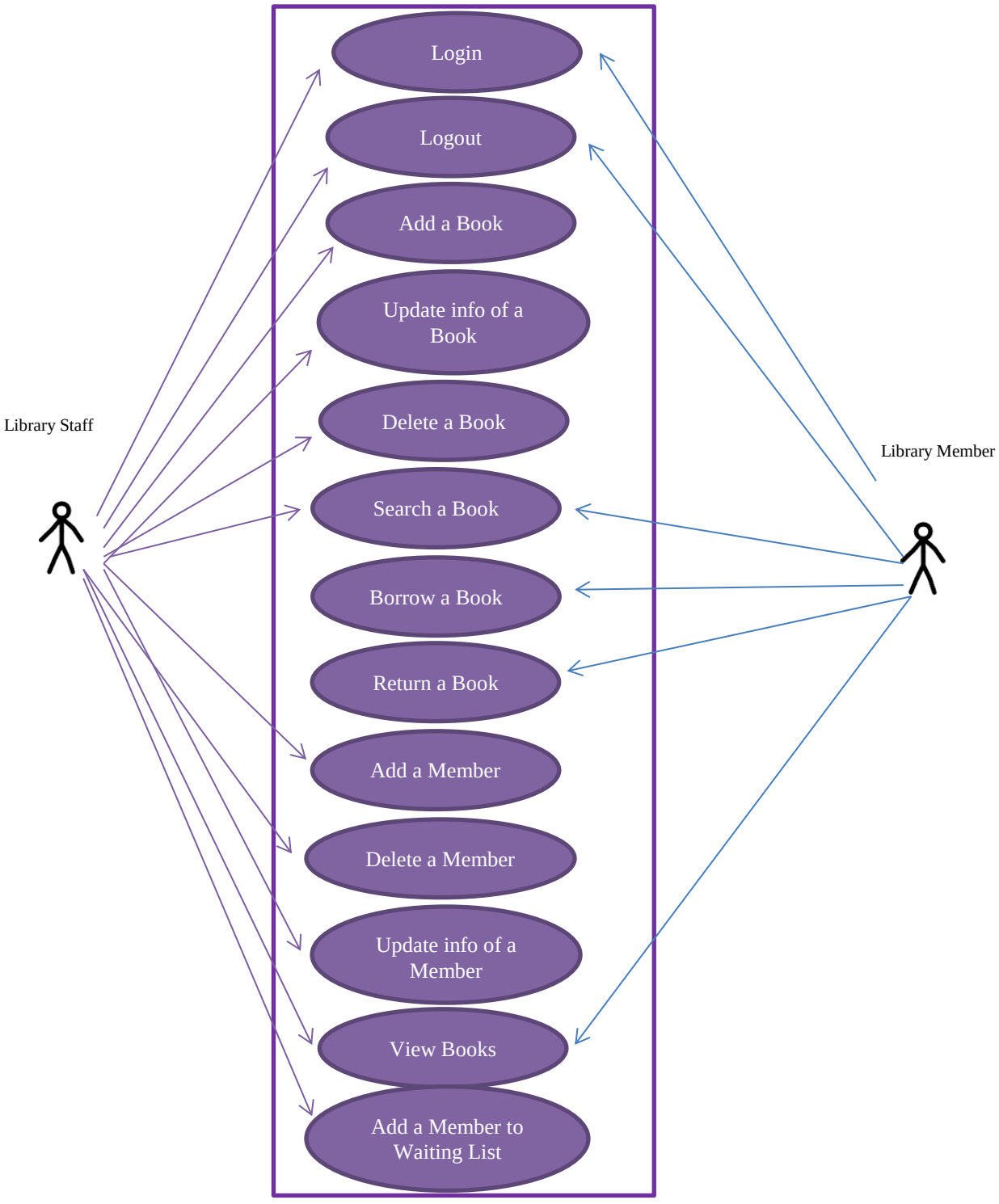
2. System-Wide Functional Requirements

Here are the functional requirements for the system that we will be constructing throughout the semester:

1. **Login:** The library staff and the library member will enter the system with the name and password.
2. **Logout:** The users who entered the system will log out.
3. **Add a Book:** When a new book arrives, the library staff will add it to the system.
4. **Update the Information about a Book:** The library staff will change the information about a particular book when needed.
5. **Delete a Book:** Library staff will remove a particular book from the system when needed.
6. **Search a Book:** Any user of the system will be able to search for the desired book by name. The system will give the user the related information.
7. **Borrow a Book:** The library member (student or scholar) will borrow books through the library staff and start the borrowing. The system will save this change.
8. **Return a Book:** When the library member returns the book, the library employee will inform the system about it so that the system makes the changes on the information about the related member and the book.
9. **Add a Member:** The library staff will add a member to the system.
10. **Delete a Member:** The library staff will remove an existing member from the system.
11. **Update the Information about a Member:** The library staff will change any kind of an information about a particular member when needed.
12. **View Books:** Borrowed books will be viewed from any user when the information about that particular member is entered.

13. **Add Name of a Member to the Book Waiting List:** If the book that a member wants to borrow is already taken from another member, that member will be in the waiting list for that particular book. Once the member who borrowed the book returns it, the first person in the waiting list will be able to borrow that book.

Considering these use cases, the following is the Use Case Diagram that we designed according to the required use cases we defined:



Related **activity diagrams of these use cases** are given in **Appendix-A**.

In addition, **corresponding data model as an E/R Diagram** is given in **Appendix-C**.

3. System Qualities

Without a doubt, the main requirements of our system will be shaped by the conditions of quality. In order to provide these qualities we have to define the set of requirements for each quality. The following sections state these basic requirements in an order:

3.1 Usability

As the developers of this project, undoubtedly, usability is the most crucial point for this system since the main aim of constructing this software project is to provide a smooth and practical system to the users. The most crucial requirement for every single user in the system is a smooth and practical interface that will direct the user without leaving any question marks on their mind. The user will be directed to the operations that they are eligible to do, so that they will make the best use of such a system. In brief, the basic requirement for usability is building a user-friendly system that is ready-to-use any time.

3.2 Reliability

Reliability is also a crucial point considering the system will contain the data of numerous members and we have to maintain the data integrity. To hold the data in a proper way, as mentioned in previous documents, we will be using a database for the purpose of data integrity. To provide the requirements and maintain reliability, we will get help from database logging which is an important part for our database solution design. With database logging, we will make it possible to recover from a failure, therefore we will maintain the data integrity. Hence, the requirements for reliability will be provided.

3.3 Performance

The system will have the capability of giving an answer to the users request as fast and as accurate as possible. A possible fault or malfunction that will affect the performance of the system will be solved by the authorities as soon as possible, either by including them in the system or remotely. So the high performance will be conserved.

3.4 Supportability

Since our project will be a web application, and web applications are meant to run any web browser since it is meant for web clients, there will not be an integration or supportability problem. In addition, providing a technical support will always be quite practical and the system will have the ability of resolving any problem from the side of authorized users' part of the system, as mentioned in the previous "Performance" section.

As the project team, during our work we realized that these qualities are related to the whole system and we have to provide these qualities for every single case, so they have to be provided for the whole system.

4. System Interfaces

4.1 User Interfaces

The graphical user interface(GUI) we designed can be seen in Appendix-B. We designed the GUI and the general content but we also state that there might be slight changes on this graphical user interface during the implementation process but we will absolutely stick with the content we provided even though during the implementation phase there might be slight changes on the general view& design.

4.1.1 Look & Feel

Since we want the program to be user-friendly, we are planning a quite plain interface that the user can make any operation smoothly and in a practical way. Our main aim is absolutely not a colorful and attractive interface since this program will be used for the users' book-related issues so there is no need to make a complicated and colorful system that will add more details to the program. The interface will consist of the buttons and textboxes that are quite basic and will respond to the users' requirements accurately which is the most important part of our project considering the purpose of use of this project.

4.1.2 Layout and Navigation Requirements

There are some set of information that has to be grouped together so that the interface will be meaningful for the user. When the library staff attempts to enter information about a new member, every single information about that member such as Name, Surname, Email, Phone Number, Date of Birth etc. should be seen together so that user will have a better understanding of what they are doing without any hesitations. Another example is for login operation. For a successful login, there has to be two textboxes, one for Name and one for password, not less, not more. So that the user will enter only required number of information and will not be misdirected by unnecessary details.

4.1.3 Consistency

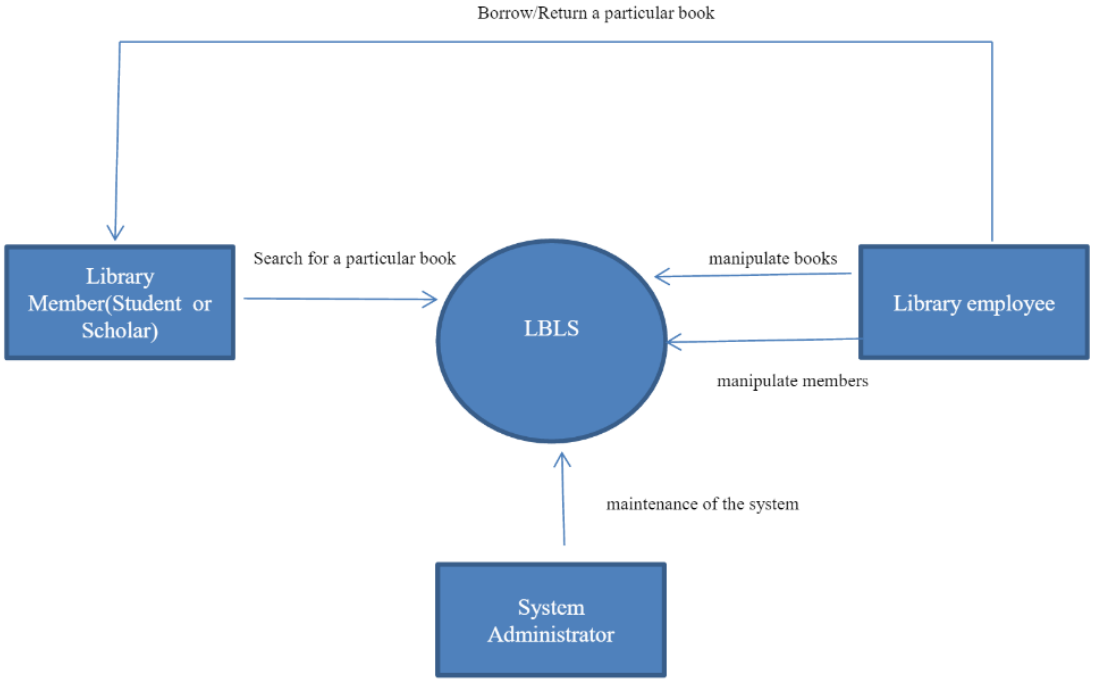
Every single step in this project is highly predictable for the user since it will be built for only professional purposes instead of fun. The user will always encounter what they accept at any stage so there will not be any problem for user to follow any kind of ambiguities.

4.1.4 User Personalization & Customization Requirements

There is absolutely a difference on the display of the system when it is entered by a member or a staff that has more authority on the system. Since a library staff has the authority of adding, deleting or updating any kind of an information about a member or book, they will encounter a different interface which a normal member will not have access to. A regular member(scholar or student) will just have the authority of basic operations in order to access an information about a particular book so that they can borrow them.

4.2 Interfaces to External Systems or Devices

As stated in Software Vision document, the following context diagram shows our diagram that defines a high level view of the system that we are planning to construct:



The graphical user interface(GUI) we designed can be seen in Appendix-B.

4.2.1 Software Interfaces

We will be using Oracle Database to hold the data. As integrated development environment (ide), we will be using Eclipse.

4.2.2 Hardware Interfaces

Our project does not include any kind of a hardware interface.

4.2.3 Communications Interfaces

Our project will not be containing any communications interface.

5. Business Rules

Since Business requirements are the critical activities of an enterprise that must be performed to meet the organizational objectives we defined some rules that states that individuals or roles can perform some particular functions under specific circumstances .

5.1 Borrow

Number of Books to Borrow

A member cannot borrow two books at the same time. If the member wants to borrow another book, they have to return the one they borrowed before.

Membership

Only the registered members can borrow a book, the library staff will not allow anyone else to borrow a book. If someone borrows a book, their personal information has to be recorded in the system.

Extend the Time Period

When the user states that they want to borrow the book again while returning, they will extend the time period for that book **unless** it is in the waiting list.

5.2 Waiting List

Extension

If a book is in the waiting list, the member will not be allowed to borrow the book again consecutively and the first member in the waiting list will be allowed to borrow the book.

5.3 Books

Adding a Book

Only library staff is authorized for adding a book. A member is never allowed to add a book to the system.

Deleting a Book

Only library staff is authorized for deleting a book. A member is never allowed to delete a book from the system.

Updating a Book

Only library staff is authorized for updating a book. A member is never allowed to update a book of the system.

5.4 Members

Adding a Member

Only library staff is authorized for adding a member. A member is never allowed to add a new member to the system.

Updating a Member

A member will never have access to the information of members in the system so they are never allowed to update information of a member.

Deleting a Member

A member will never have access to the information of members in the system so they are never allowed to delete information of a member.

6. System Constraints

Since we are constructing this project as a library book loan system, we have to be aware that there has to be a set of constraints and restrictions considering the specific requirements of the system that we are building. First of all, the system should be designed in a way that every single user can find what they are looking for. Therefore, we have to avoid creating a complicated and complex interface that will confuse the user. Another crucial point to be considered is the constraint about the system's stability. We have to consider this point because the system should be prepared to encounter any kind of an inaccurate operation or a sudden mistake. The general design should be made considering these situations so that these mistakes will not disrupt the general structure of the system. We also need this system not to be affected by technical problems or a sudden power cut.

7. System Compliance

7.1 Licensing Requirements

We will be sharing every document and source code related to our projects as well as instructions for all users, so our project will be accessible for everyone.

7.2 Legal, Copyright, and Other Notices

The content of this project is allowed to be utilized for **only** personal use.

7.3 Applicable Standards

The one and only requirement for this system was Java since the project will be implemented with Java. So the only condition for applicable standards is the support of Java Libraries.

8. System Documentation

Since we will be preparing several documents throughout this semester, we are planning to compile all these documents and will try to prepare a new document with less technical terms so that it will be easier to follow for the users. The users will be provided by a instructions manual that will enable them to have a better understanding of how to apply each operation. We will support this document with several screenshots and step by step operations.

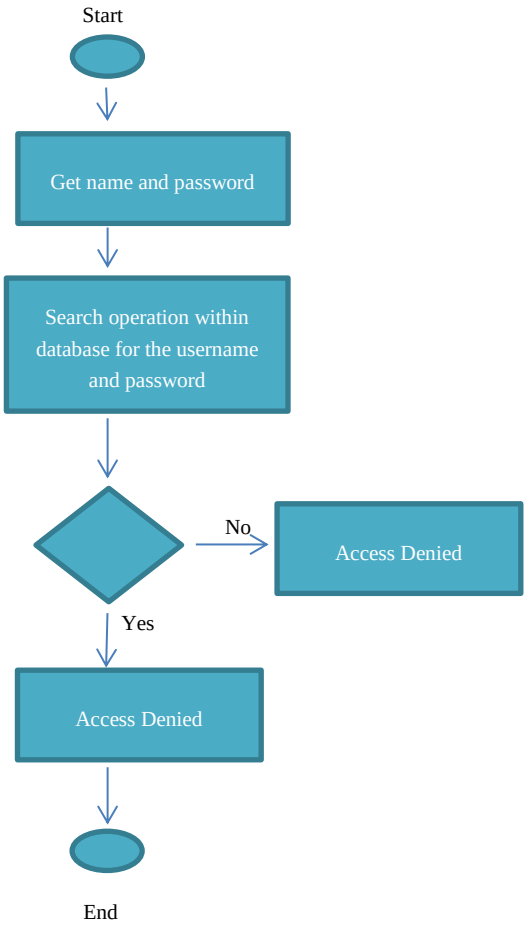
9. Appendix

9.1. Appendix A

Details and the Activity Diagrams of the given Use Cases

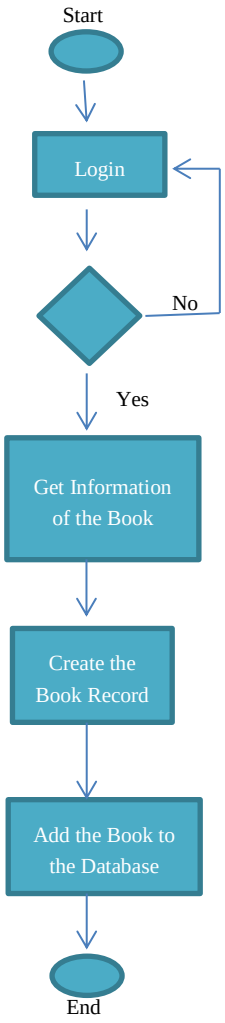
Login

The equivalence between the input and the data in the database is being compared.



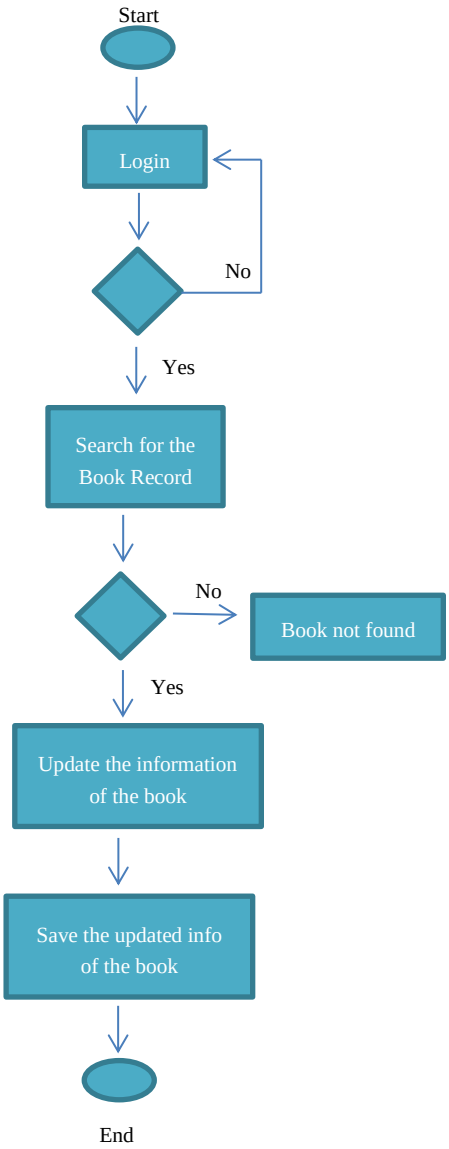
Add a Book

The book is added to the database after checking whether the login operation is successful or not.



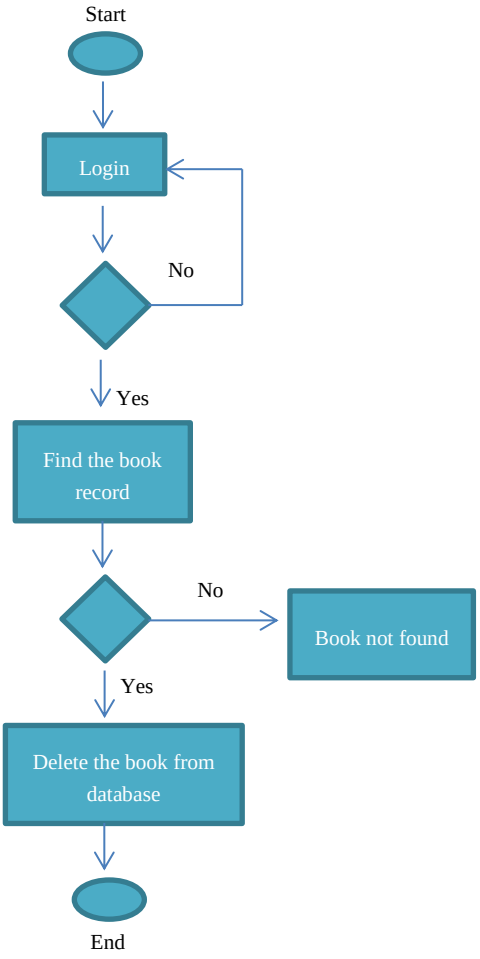
Update the Information about a Book

After checking whether the Login is successful, the book is searched and after checking whether this book exists on the system, the record is updated.



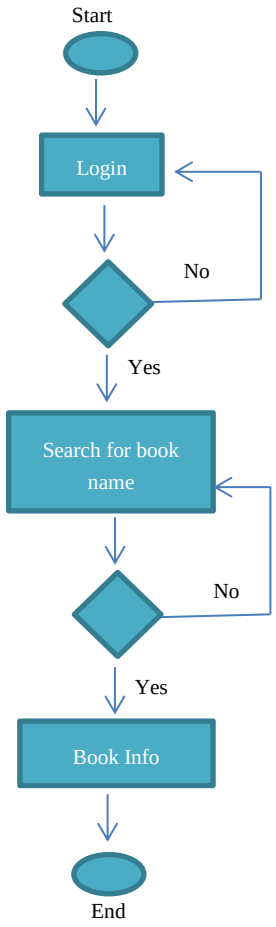
Delete a Book

After checking whether login operation is successful, the book record is searched and if the search operation is successful and related book exists, the book is removed from the database.



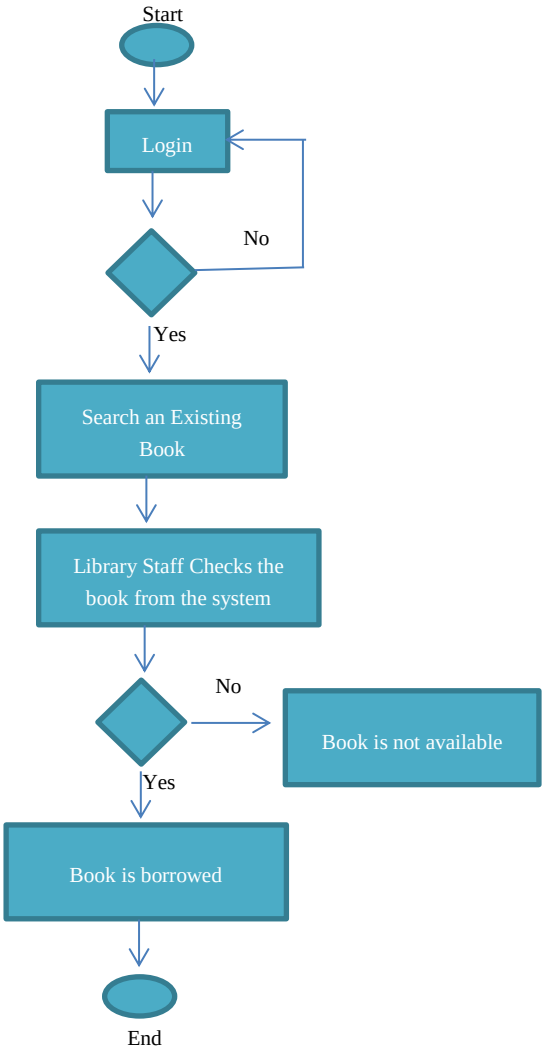
Search a Book

After checking whether the login is successful, book name is searched and if the searched book exists in the system, information about that book is displayed.



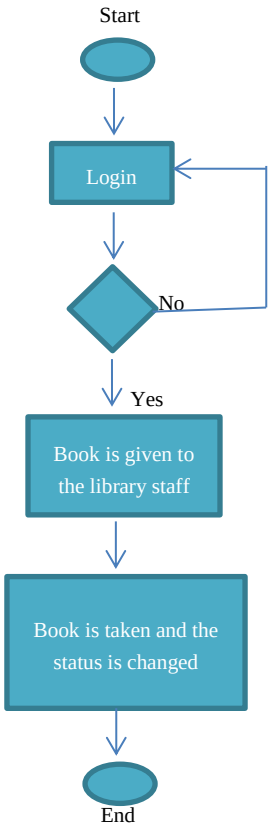
Borrow a Book

If the login operation is successful, the search is done for the particular book that the member wants to borrow. Afterwards, the library staff checks whether this book is available or not. If available, the book is given to the member who wants it and the status of the book will be changed.



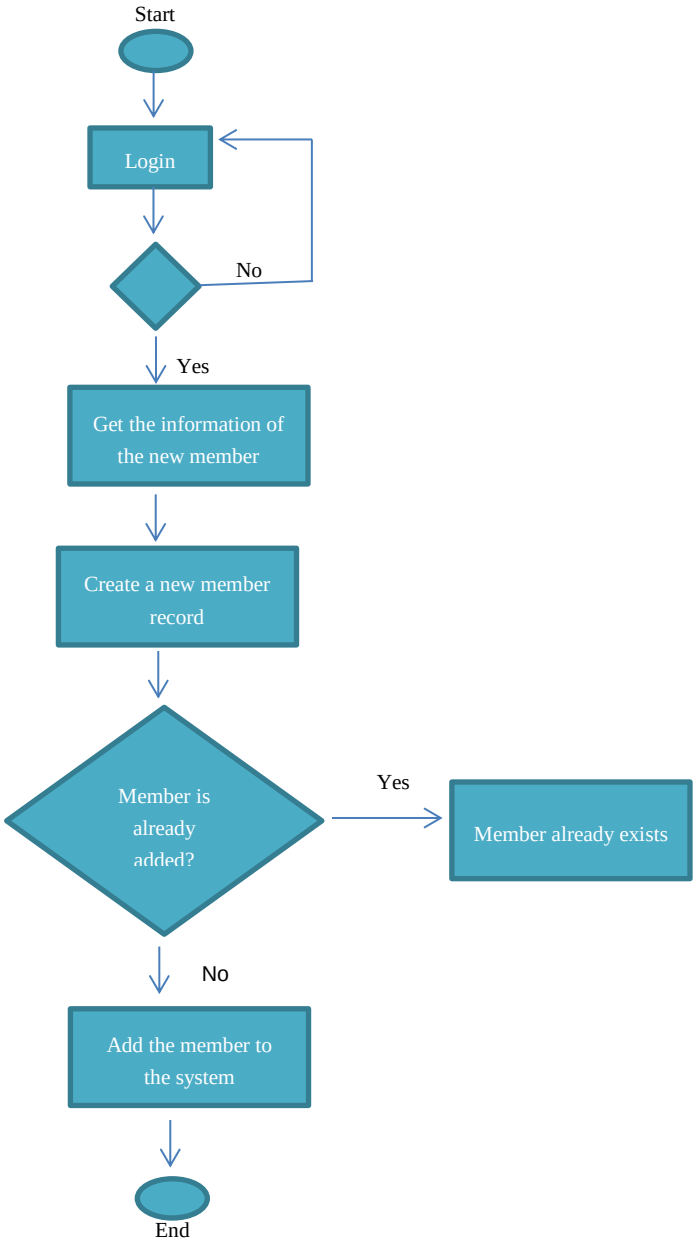
Return a Book

If login operation is successful, once the book is given to the library staff, it is taken and the necessary status change is done by the library staff.



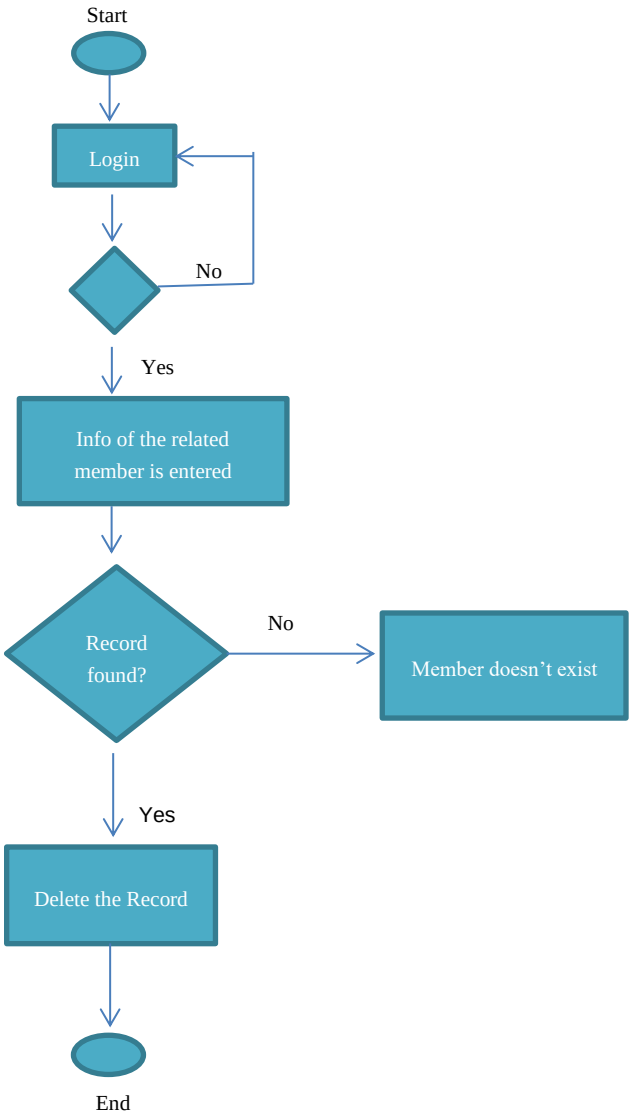
Add a Member

After a successful login, the information of the new member is entered to the system and it is checked whether a member with the same information is already in the system. If not, new member is added to the system.



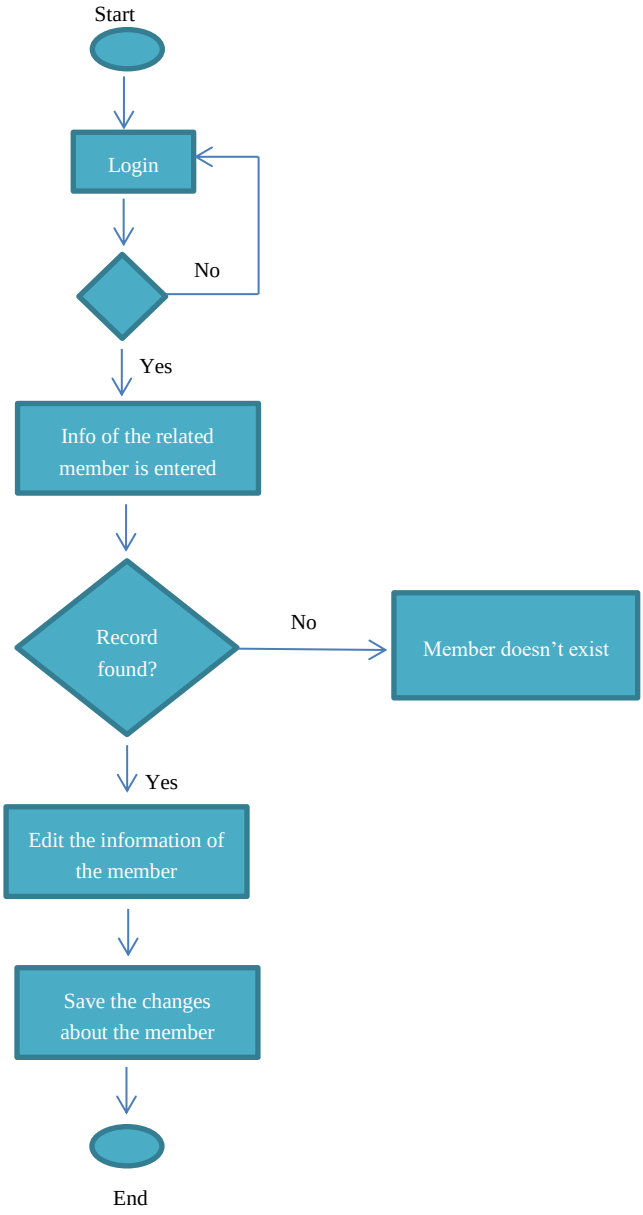
Delete a Member

After a successful login, the information of the member is entered and it is checked whether this record exists in the database. If it does, the member is deleted from the system.



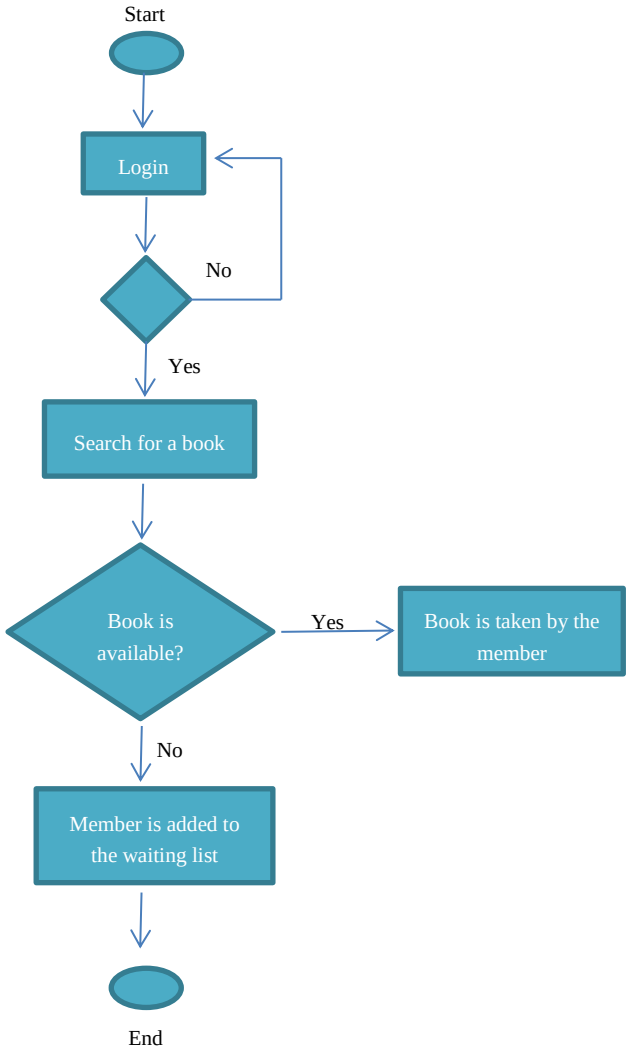
Update the Information about a Member

If the login is successful, information of the member is entered and if there is a record matching with the entered information, it is updated and saved to the database with this new information.



Add Name of a Member to the Book Waiting List

If the login is successful, book is searched and it is checked whether that book is available or nor. If it is available, the book is taken by the member and if not, that member is added to the waiting list.



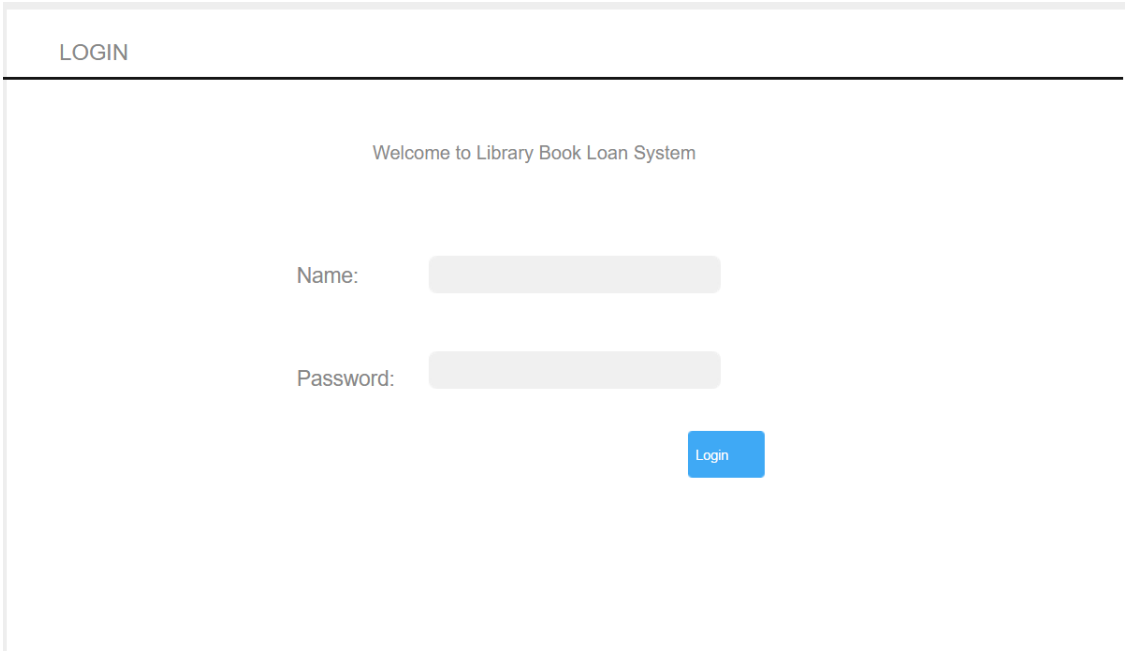
9.2. Appendix B

Graphical User Interface and Related Use Cases

The following shows our interface design. Even though the content will remain the same, there might be slight changes on the design and general view during implementation process but the interface will keep being quite smooth and easy-to-follow for every single user in the system.

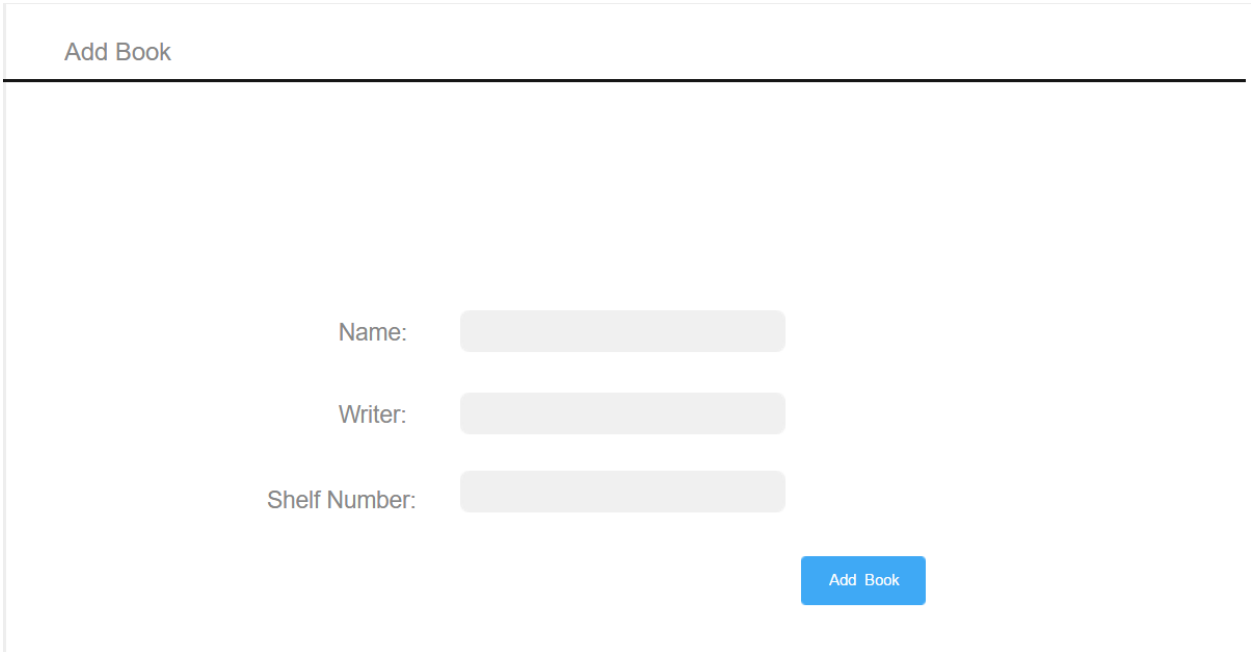
Interface Number 1 – Login

As we stated the use case and activity diagram of Login Operation in Appendix A, our interface design for this operation is as follows:



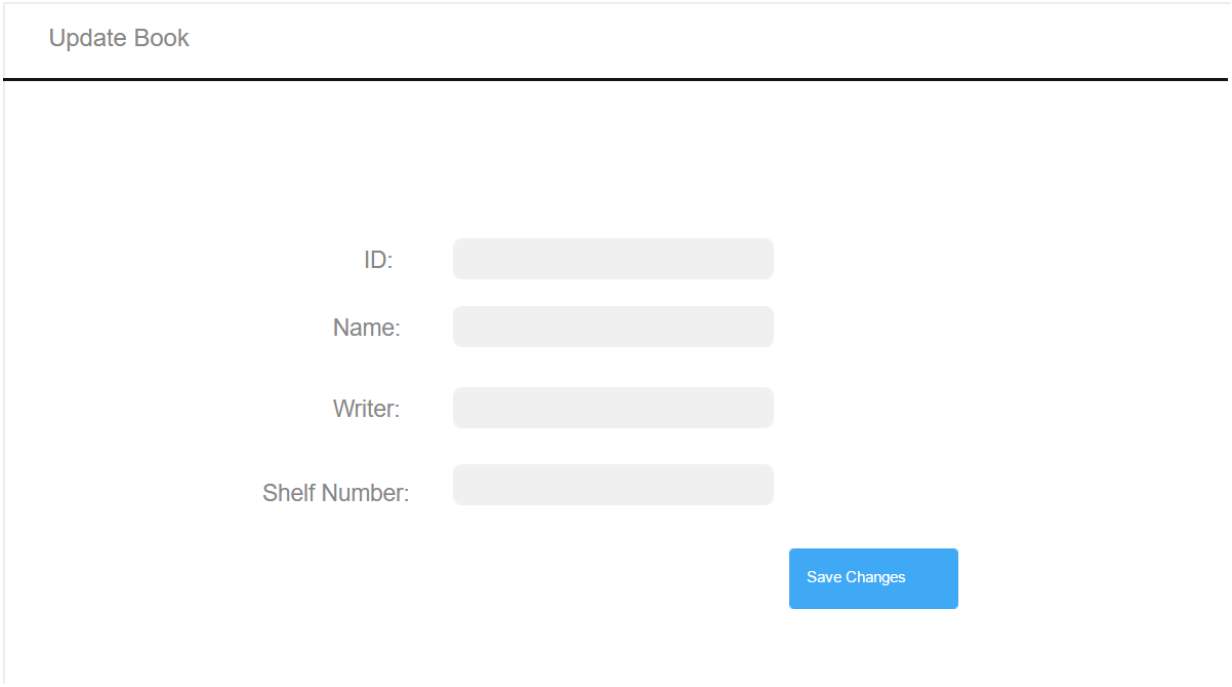
Interface Number 2- Add a Book

As we stated the use case and activity diagram of Add a Book operation in Appendix A, our interface design for this operation is as follows:



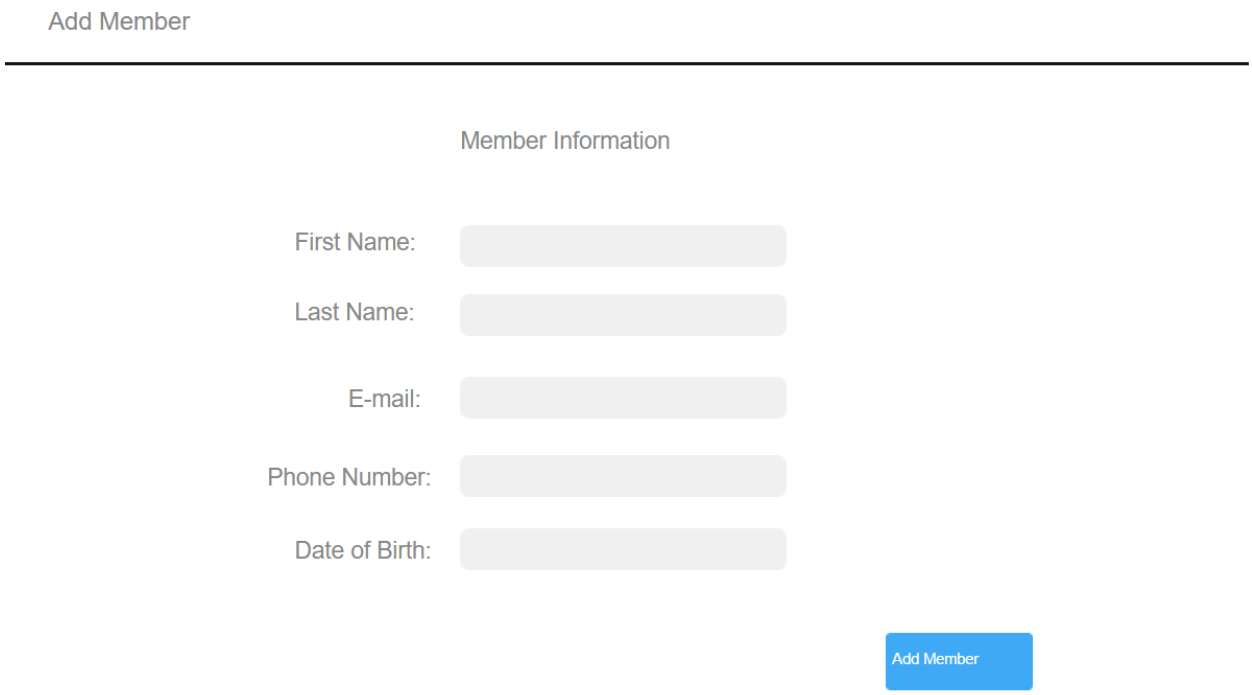
Interface Number 3- Update the Information about a Book

As we stated the use case and activity diagram of Update the Information about a Book operation in Appendix A, our interface design for this operation is as follows:



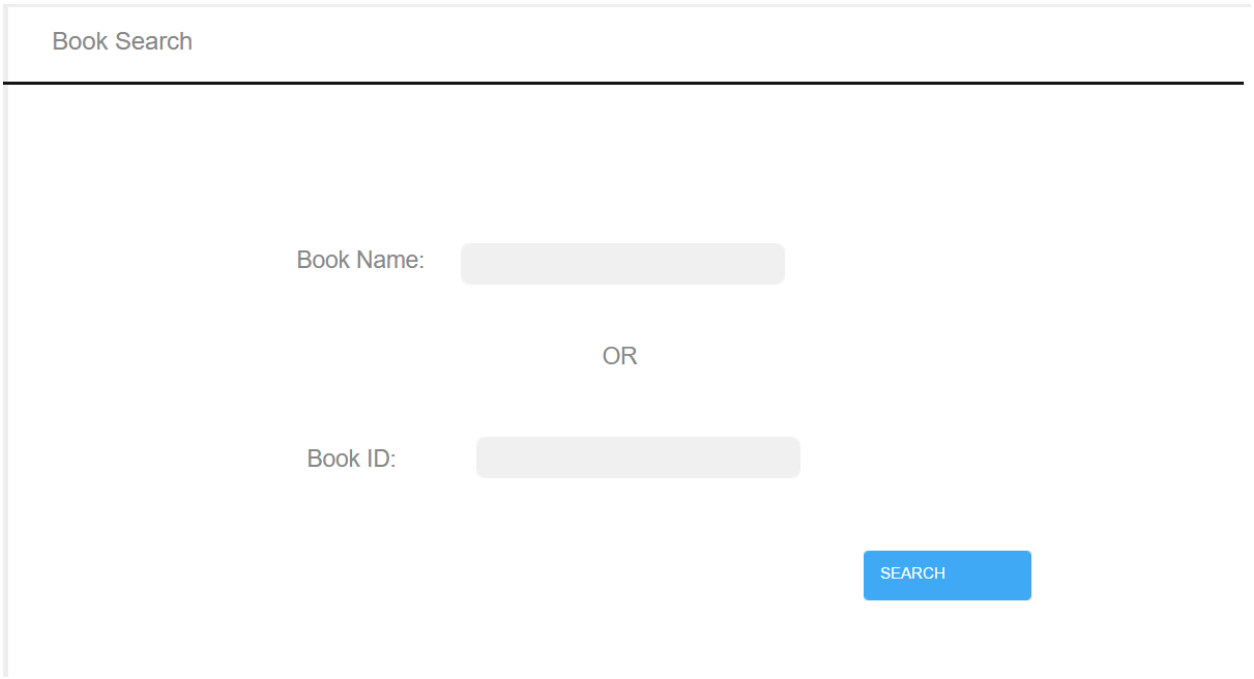
Interface Number 4- Add a Member

As we stated the use case and activity diagram of Add a Member operation in Appendix A, our interface design for this operation is as follows:



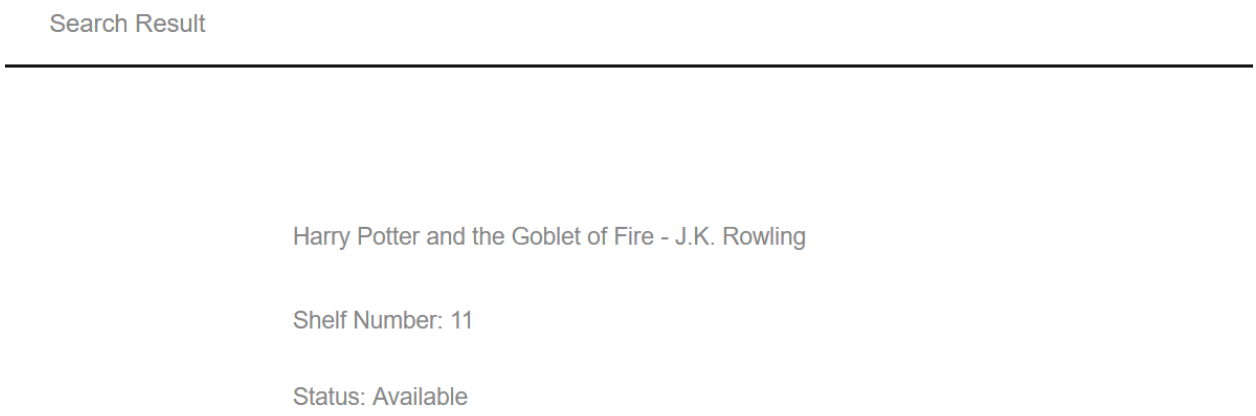
Interface Number 5- Search a Book

As we stated the use case and activity diagram of Search a Book operation in Appendix A, our interface design for this operation is as follows:



Interface Number 6- Search Result

If the corresponding book is found, the design of the result screen will be as follows:



9.3. Appendix C

E/R Diagram

The corresponding data model we constructed with an online diagram software named **Creately** is shown below as an Entity-Relationship (E/R) Model: