HACETTEPE UNIVERSITY COMPUTER ENGINEERING DEPARMANT
BBM 487 – SOFTWARE ENGINEERING
LABORATORY

LIBRARY BOOK LOAN SYSTEM
- System-Wide Requirements Specification -

Group 10
Berat Gökser Atlı - 21327656
Hülya Şermin Karakaş – 21591198
Sefa Feruzoğlu 21228338

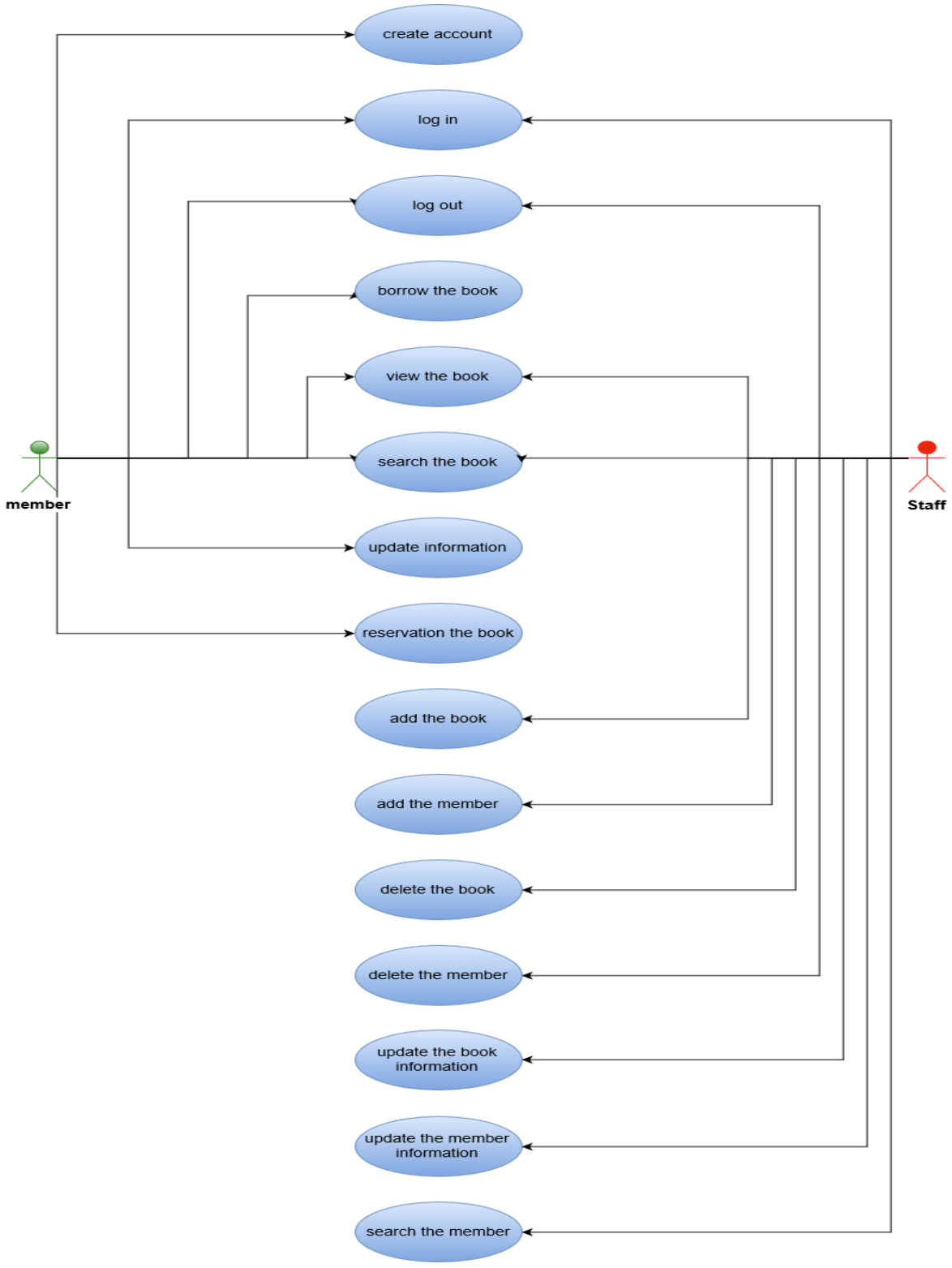
Library Book Loan System
System-Wide Requirements Specification

1. Introduction

At this stage we have analyzed the requirements of the project. Based on these system requirements, we have created "use case diagram", "activity diagram" and "E / R diagram". We designed the interfaces we use in the system.

2. System-Wide Functional Requirements

1. **Login:** The library staff and the library member will enter the system using the username and password.
2. **Logout:** Users will be able to exit the system.
3. **Create Acoount:** Users can create member accounts.
4. **Borrow the Book:** Members can borrow books from the system.
5. **View the Book:** Members will be able to see a list of all books.
6. **Search the Book:** Members will be able to search among the books in the library.
7. **Update Information:** Members will be able to update their own information.
8. **Reserve the Book:** members will be able to reserve borrowed books.
9. **Add the Book:** Staff will be able to add new books to the system.
10. **Add the Member:** Staff will be able to add new members to the system.
11. **Delete the Book:** Staff can delete the book from the system.
12. **Delete the Member:** Staff can delete the member from the system.
13. **Update the Book Information:** The staff can update the book information.
14. **Update the Member Information:** the staff can update the member information.


create account
log in
log out
borrow the book
view the book
search the book
update information
reservation the book
add the book
add the member
delete the book
delete the member
update the book information
update the member information
search the member
member
Staff

3. System Qualities

One of the most important requirements of the system is the quality requirements. We will indicate some quality requirements in the following headings:

3.1 Reliability

The user always wants a simple and usable program. One of the most important requirements is availability. It is something that will connect the user and the program. The best way to improve usability is to design an usable interface. On this count, the user will be able to use the program without any problems without question marks.

3.2 Performance

Reliability is a very important point in the project. It is very important to protect the information of the users who use the system. The database is activated at this point. We will develop the database more carefully so that no data is lost.

3.3 Supportability

The system will respond as quickly as possible to user requests. Possible problems will be resolved in the shortest time and the system performance will be kept at the highest level. In this way, we aim to increase usability.

3.4 Supportability

Because our application will be a web application, every platform will work. We will not have problems like not being supported.

4. System Interfaces

4.1 User Interfaces

You could find the graphical user interface(GUI) of our library book loan system in Appendix-B. We tried to design the very similar template of the project but we could redesign something if we need to when we start developing the software part of the project.

4.1.1 Look & Feel

While we are working on the interface, we decided to keep the design as simple as possible. We consider that, in our daily lives, the simple designs are easier to use and understand the concept of the application.

4.1.2 Layout and Navigation Requirements

Our system design starts with the login or search selection scene. The user can search a book without login to the system. If a librarian(Admin) or a member(user) tries to login to the system, admin panel or book ordering scene will be open. The admin panel contains 3 different sub panel on it which are order a book (for the members), add a book or update a book. An ordering scene contains ID, name or author of the book to order it.

4.1.3 Consistency

While we are working on interface design, we considered our past experiences and we tried to implement the system according to those experiences. Because of that, in our opinion, there are no unexpected interfaces or processes that a user cannot expect.

4.1.4 User Personalization & Customization Requirements

Like we said in the layout and navigation requirements part, the admin and the member has different interfaces and different kind of duty in the system. While a member is able to just order a book, an admin could order a book (for the members), add a book or update a book.

4.2 Interfaces to External Systems or Devices

4.2.1 Software Interfaces

All information about both actors and books will store in a database. When there are any data added, or deleted, the database will be update according to the operation.

4.2.2 Hardware Interfaces

Our project design is working on browser so it will work on any operation system and device which has browser installed on it.

4.2.3 Communications Interfaces

The hardware that will use the library book loan system needs to have an internet connection to browse the page. Other than that, we did not plan to use any other communication.

5. Business Rules

5.1 Add a Book

To add the book, admin must login to system and must have all needed information of the book.

5.2 Order a Book

To order the book, the member could login to system and order a book or could go to librarian and order the book with his/her help.

5.3 Create Account

To create an account, the user needs to be a student in that university and must give all information correctly.

5.4 Search a Book

Anyone could search a book.

5.5 Login

The system needs e mail and password to login.

5.6 Logout

The member or librarian could logout to system by clicking logout button in the interface.

6. System Constraints

- System needs to connect to the internet
- There will be some backups daily or weekly, so system needs to have enough space for those backups.
- The computer needs to have an installed browser.

7. System Compliance

7.1 Licensing Requirements

According to the system we designed, we use all softwares open source. So we don't need any licensing requirements.

7.2 Legal, Copyright, and Other Notices

We will need to show the publishers of the book as copyright in the system.

7.3 Applicable Standards

Our system will have IEEE standards about the software architecture. We are planning to follow these standards on our project's design.

8. System Documentation

We will prepare a guidance document to guide the members and librarians to how to use the system. This document will have the following information;

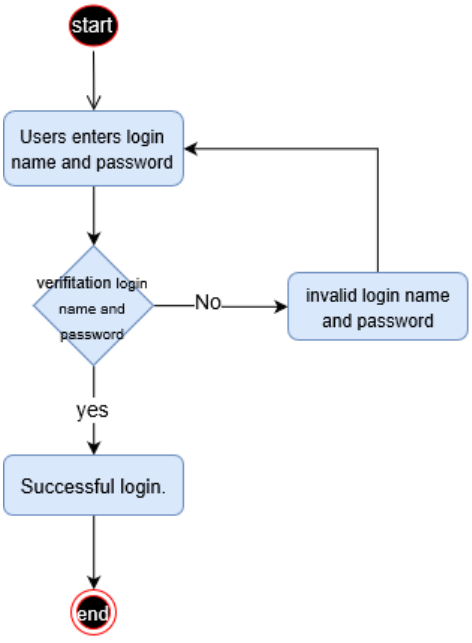
- Adding a book
- Deleting a book
- Creating an account
- Ordering a book
- Logging in

9. Appendix

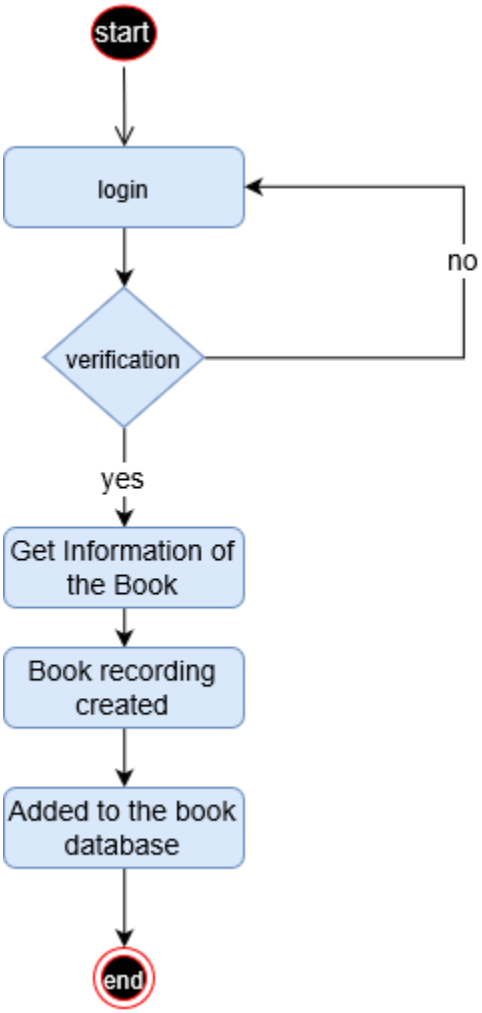
9.1. Appendix 1

Activity diagrams:

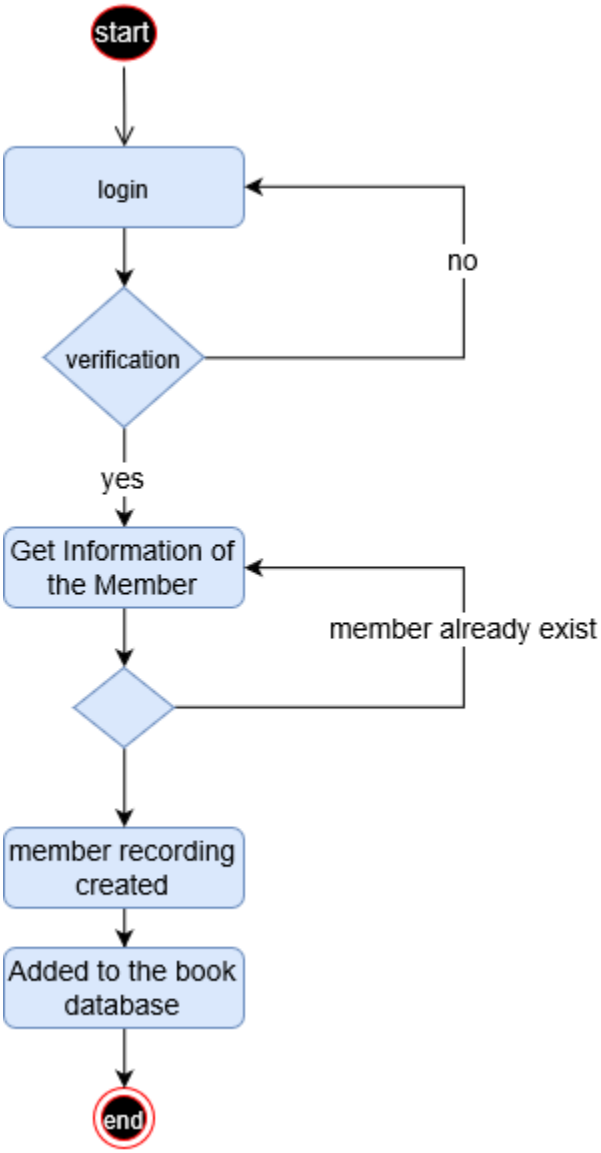
1.Login



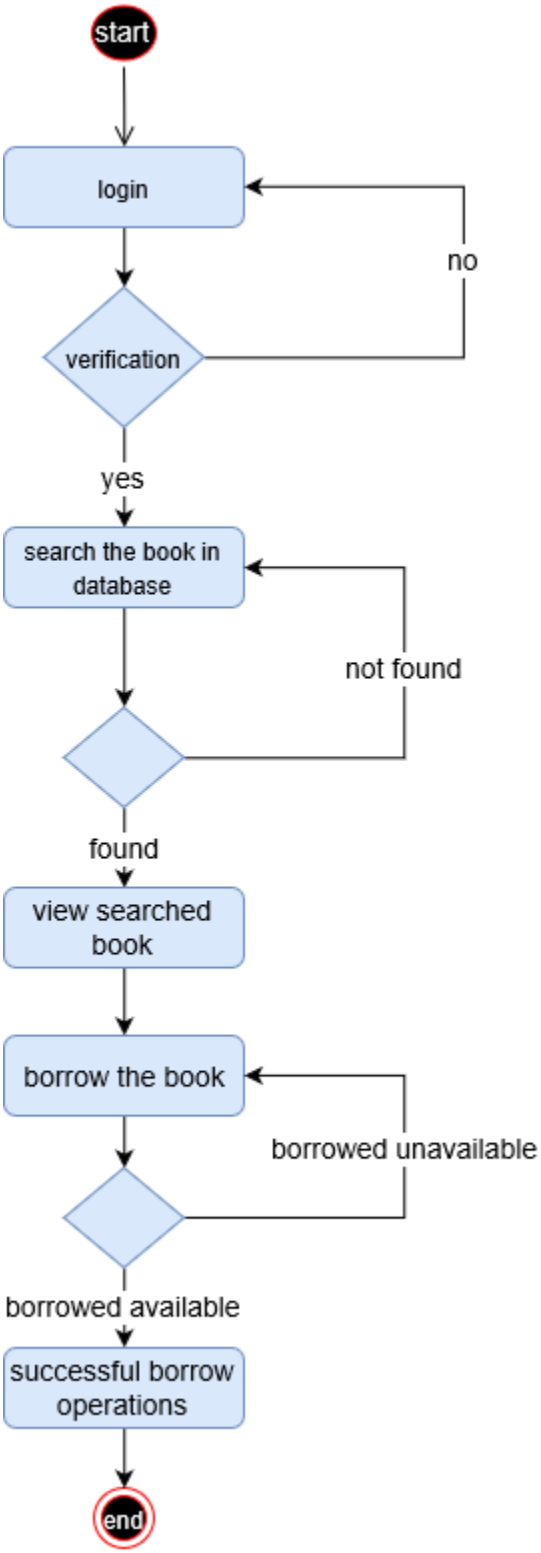
2.addBook


start
login
no
verification
yes
Get Information of the Book
Book recording created
Added to the book database
end

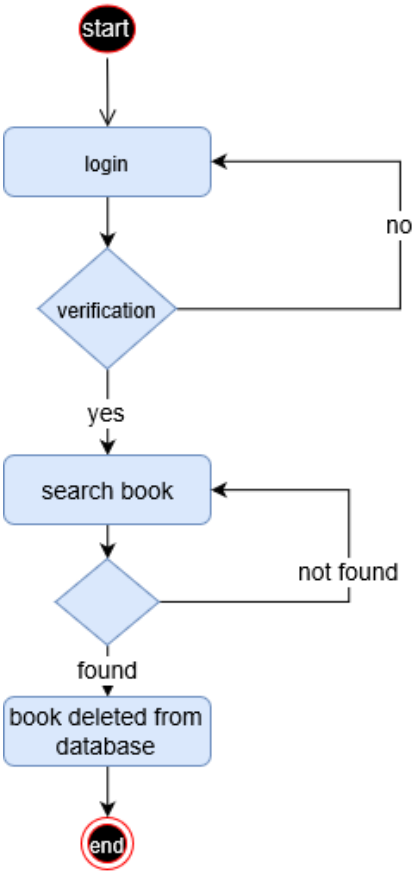
3.addMember


start
login
no
verification
yes
Get Information of the Member
member already exist
member recording created
Added to the book database
end

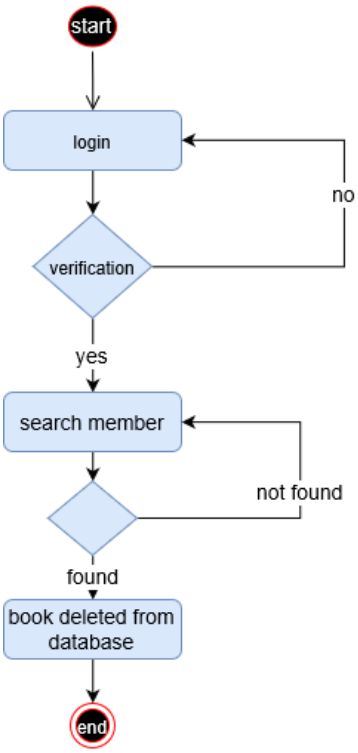
4.borrowBook



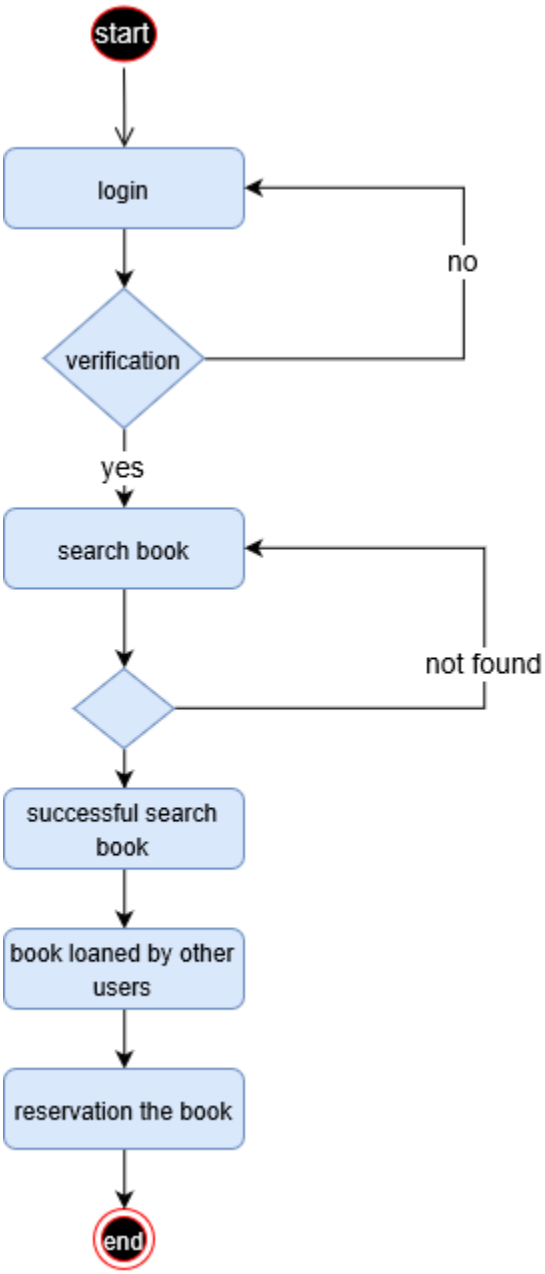
5.deleteBook



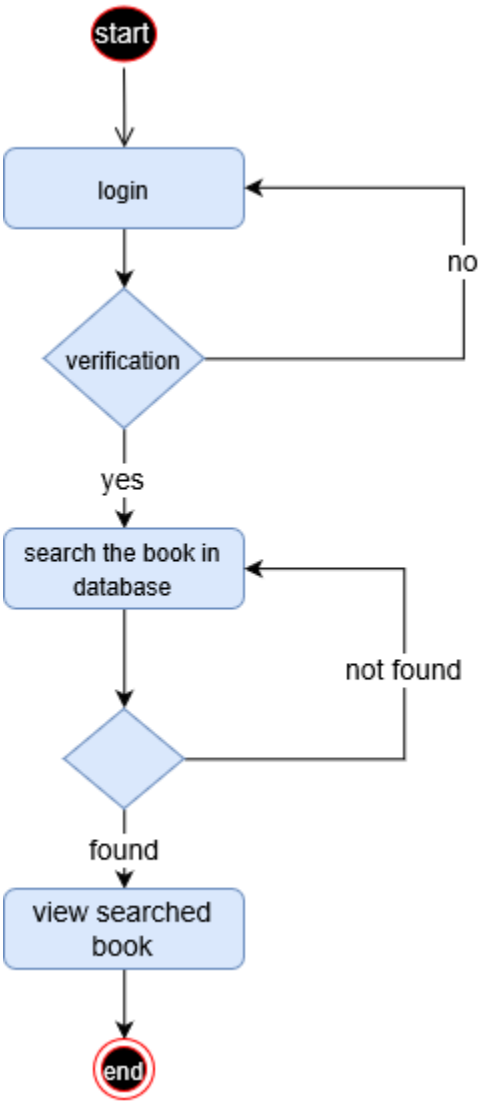
6.deleteMember



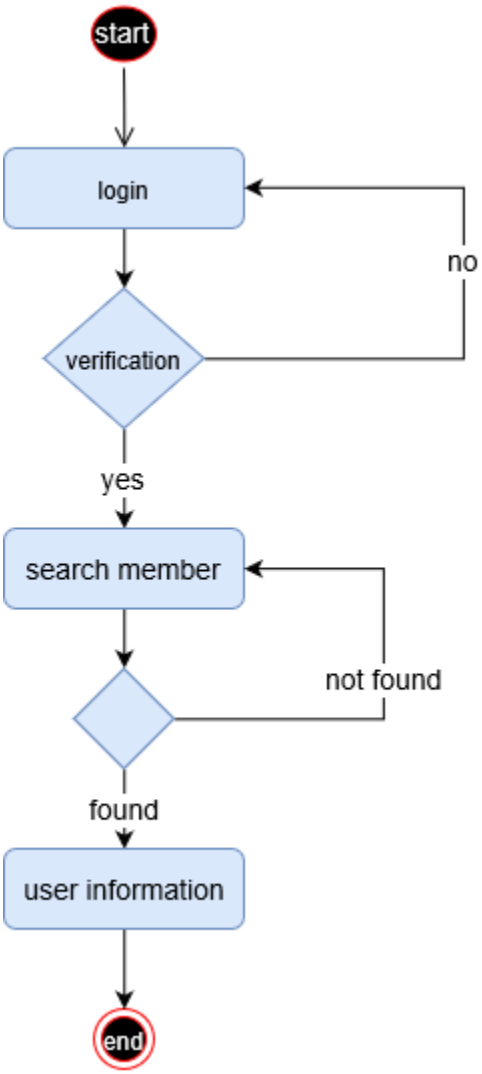
7. reservationBook



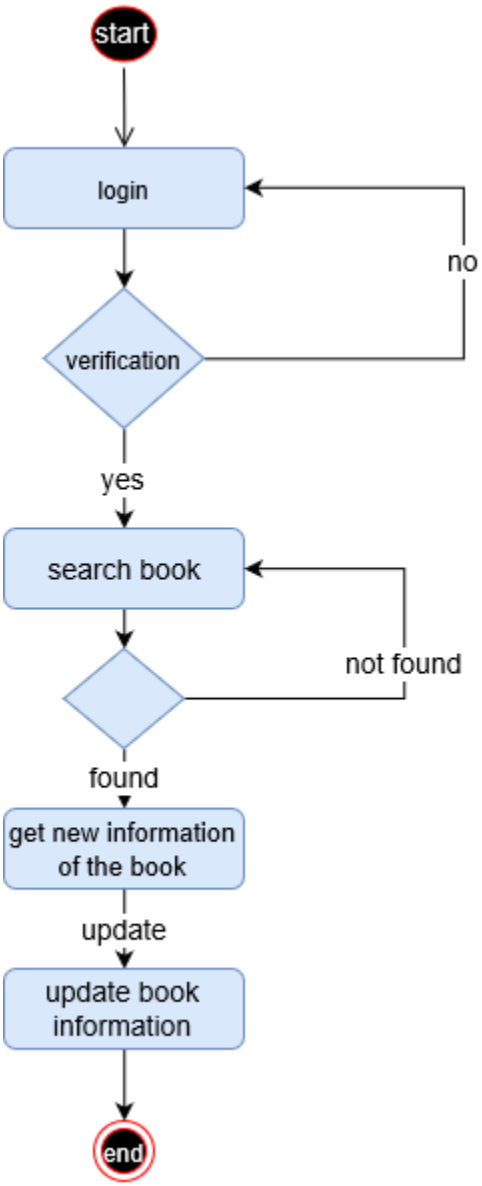
8.SearchBook



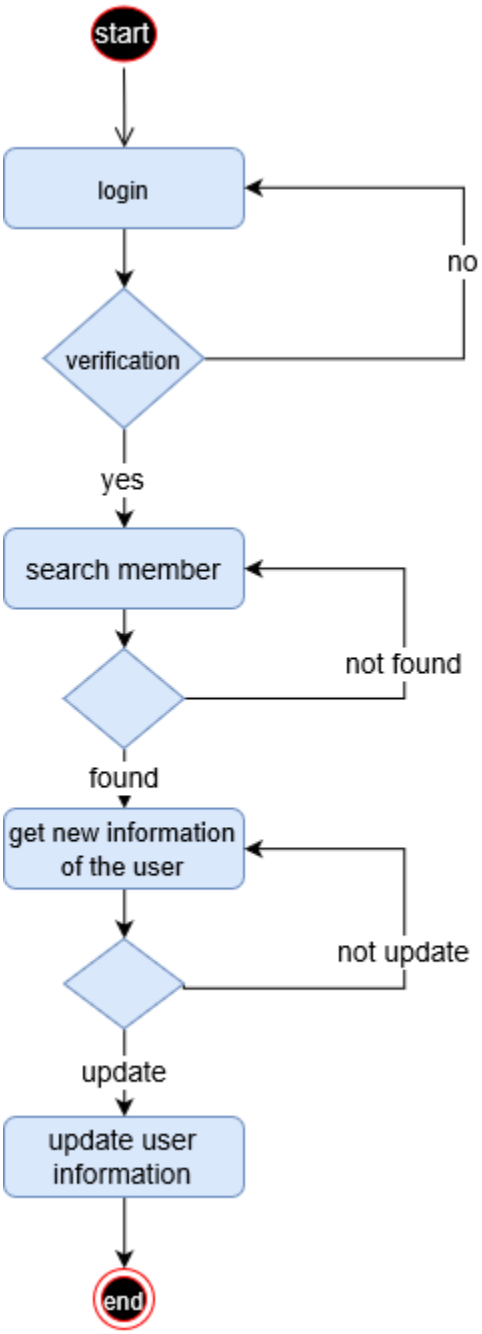
9.searchMember



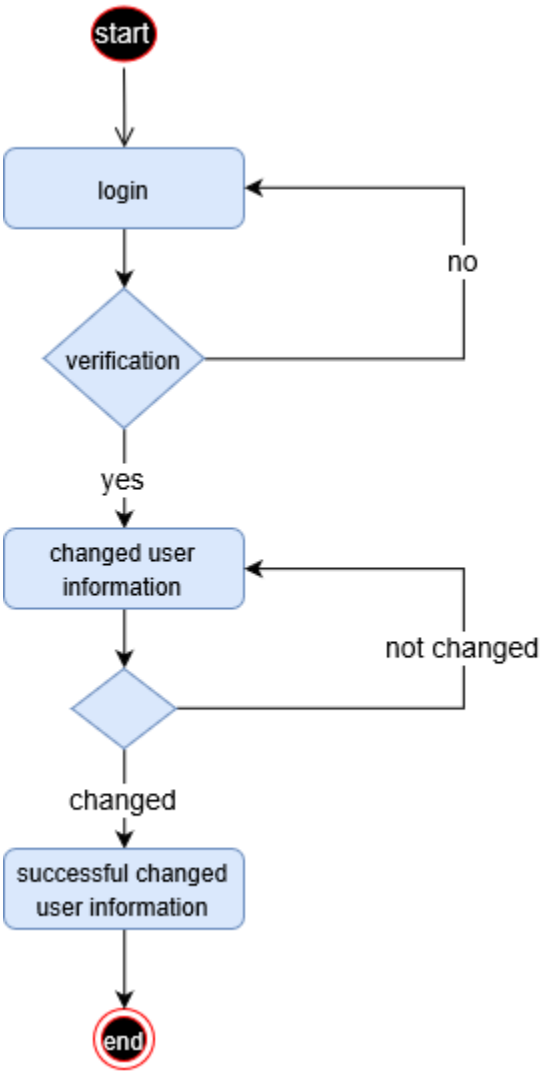
10.updateBook



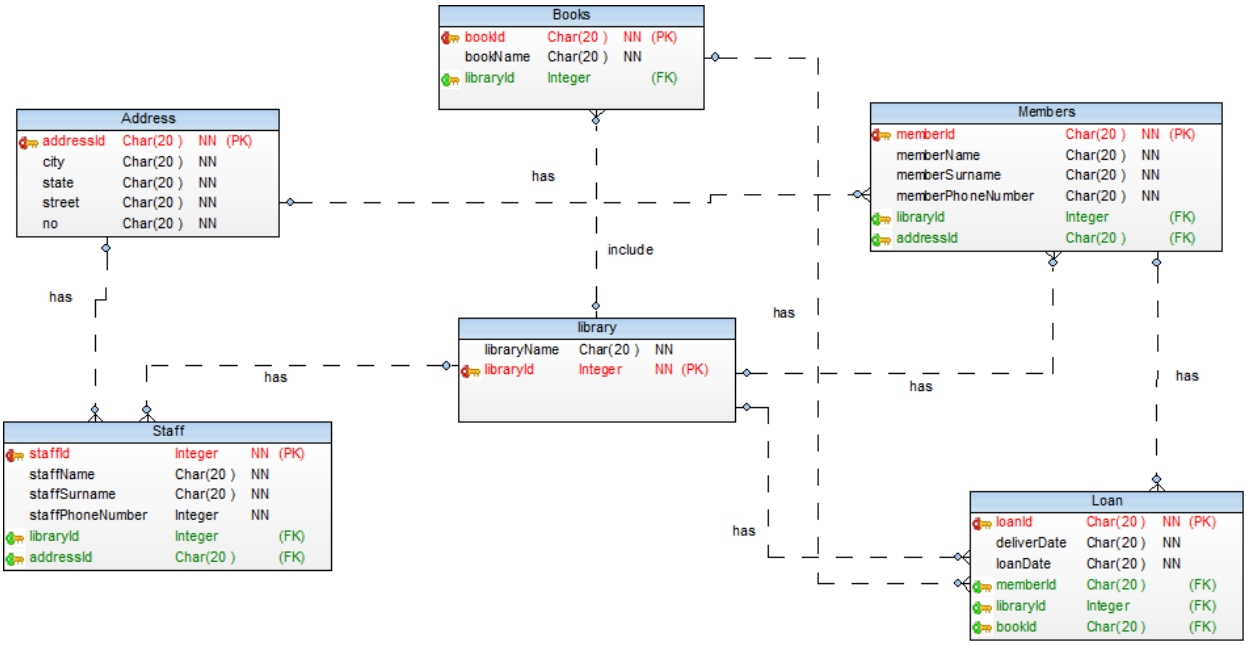
11.update Member



12.updateUserInformation



9.2. Appendix 2
E/R Diagram


Books
bookId Char(20) NN (PK)
bookName Char(20) NN
libraryId Integer (FK)
Address
addressId Char(20) NN (PK)
city Char(20) NN
state Char(20) NN
street Char(20) NN
no Char(20) NN
Members
memberId Char(20) NN (PK)
memberName Char(20) NN
memberSurname Char(20) NN
memberPhoneNumber Char(20) NN
libraryId Integer (FK)
addressId Char(20) (FK)
has
include
has
library
libraryName Char(20) NN
libraryId Integer NN (PK)
has
has
has
has
Staff
staffId Integer NN (PK)
staffName Char(20) NN
staffSurname Char(20) NN
staffPhoneNumber Integer NN
libraryId Integer (FK)
addressId Char(20) (FK)
has
Loan
loanId Char(20) NN (PK)
deliverDate Char(20) NN
loanDate Char(20) NN
memberId Char(20) (FK)
libraryId Integer (FK)
bookId Char(20) (FK)