BBM 487 – SOFTWARE ENGINEERING LABORATORY

LIBRARY BOOK LOAN SYSTEM

Software Design Description

GROUP II

Özlem DEMİRTAŞ	**21327901**
Umut ÖZTÜRK	**21328394**
Rahmi Berk ŞEFKATLİ	**21427402**

Library Book Loan System

Software Design Description

1. Revision History

Version	*Date*	*Author*	*Change Description*
1.0	19/04/2017	Özlem DEMİRTAŞ	Software Design Description Document Initial Release.
1.1	21/04/2017	Umut ÖZTÜRK	Software Requirements Specification Document Release 1.1.
1.2	22/04/2017	Rahmi Berk ŞEFKATLİ	Software Requirements Specification Document Release 1.2.

2. INTRODUCTION

2.1 Purpose and Scope

The purpose of this Design Document is to present the system design at a level that can be directly traced to the specific system objective along with providing more detailed data, functional, and behavioral requirements. This Design Document will verify that the current design meets all of the explicit requirements contained in the system model as well as the implicit requirements desired by the customer.

2.2 Document Overview

The overall system design objective is to provide an efficient, modular design that will reduce the system's complexity, facilitate change, and result in an easy implementation. This will be accomplished by designing a strongly cohesion system with minimal coupling. In addition, this document will provide interface design models that are consistent, user friendly, and will provide straightforward transitions through the various system functions.

2.3 System Overview

The project titled Library Book Loan System is library management software for monitoring and controlling transactions in a library. The project “Library Book Loan System” is developed in Java, which mainly focuses on basic operations in a library like adding new member, new books, and updating new information, searching books and members and facility to borrow and return books.

“Library Book Loan System” is a windows application written for 64-bit Windows operating systems, designed to help users maintain and organize library. Our software is easy to use for both beginners and advanced users. It features a familiar and well thought-out, an attractive user interface, combined with strong searching, insertion and reporting capabilities.

2.4 Definitions, Acronyms, and Abbreviations

- **LBLS** – Library Book Loan System
- **SDD** – Software Design Description
- **JDBC**– This acronym stands for Java Database Connectivity which allow for standardized database access.
- **MySQL** - MySQL is an open source relational database management system (RDBMS) based on Structured Query Language.
- **Java Swing -** Swing is a GUI widget toolkit for Java.

2.5 References

All versions of the project were performed in accordance with the IEEE Software Engineering Standards Committee, "IEEE Std 10161998, IEEE Recommended Practice for Software.

3. Design Constraints and Decisions

- LBLS will not be dependent on any specific hardware.
- Eclipse IDE, JDBC, MySQL and Java Swing will be used to develop the system.
- It will be implemented as desktop application. For this reason, internet connection is not needed.
- It will be executed under windows operating system.
- Number of librarians and members will be limited considering the performance of the application.
- The system will respond to the member in less than two seconds from the time of the request submittal. But the system will be allowed to take more time when doing large processing jobs. (the response time can be change according to the hardware features.)
- The user interface will be simple and easy to understand and use. In this way, users can perform the various tasks easily and in an effective way.
- The error messages will be clear.
- Unregistered users will only be able to search books.
- Other operations will be done after login.
- Members and books of the library will be manipulated (add, delete and update) by only librarians.

4. Design Details

4.1 Software Components

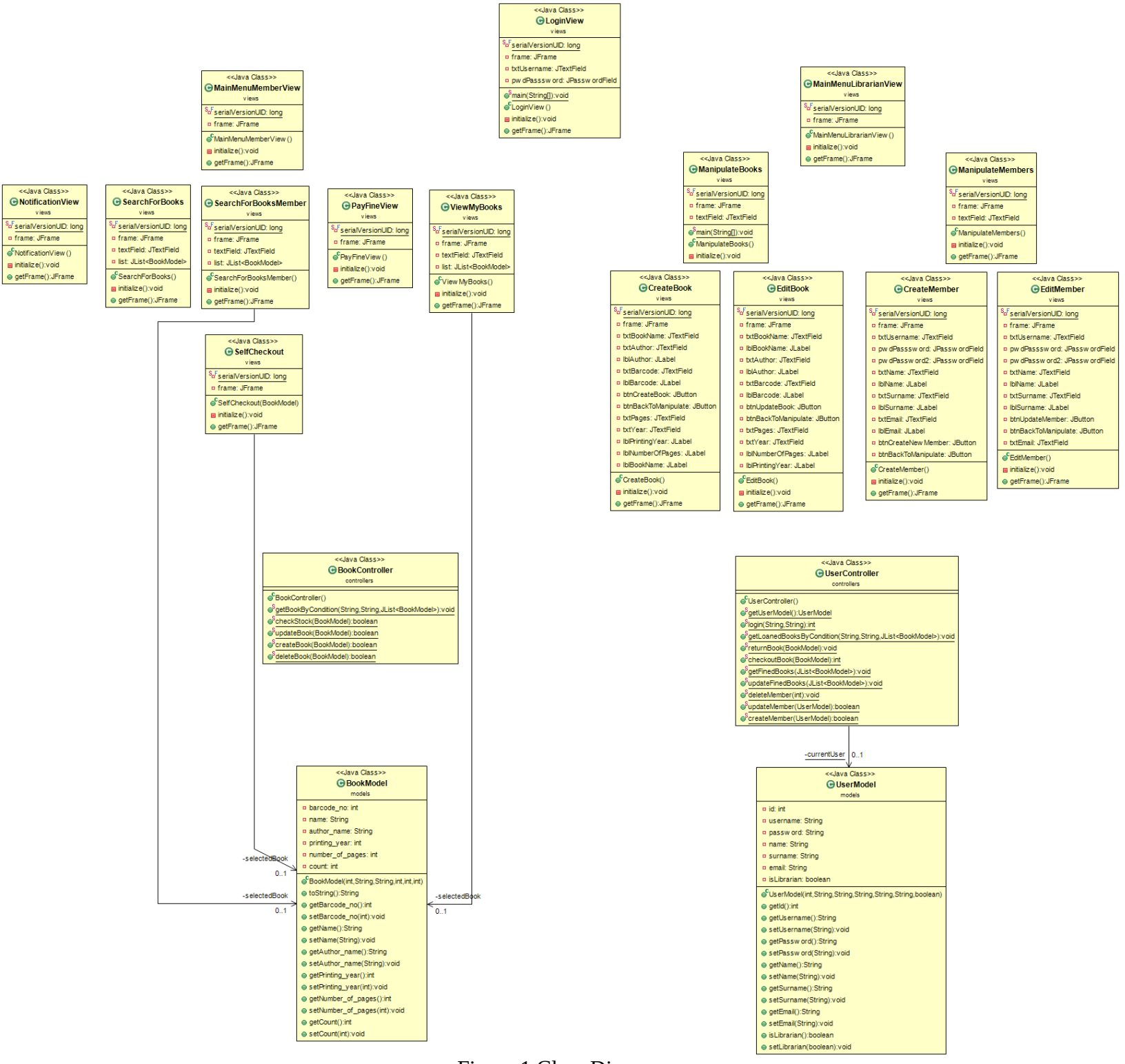

Figure 1 Class Diagram

4.2 Software Behavior

Sequence Diagrams

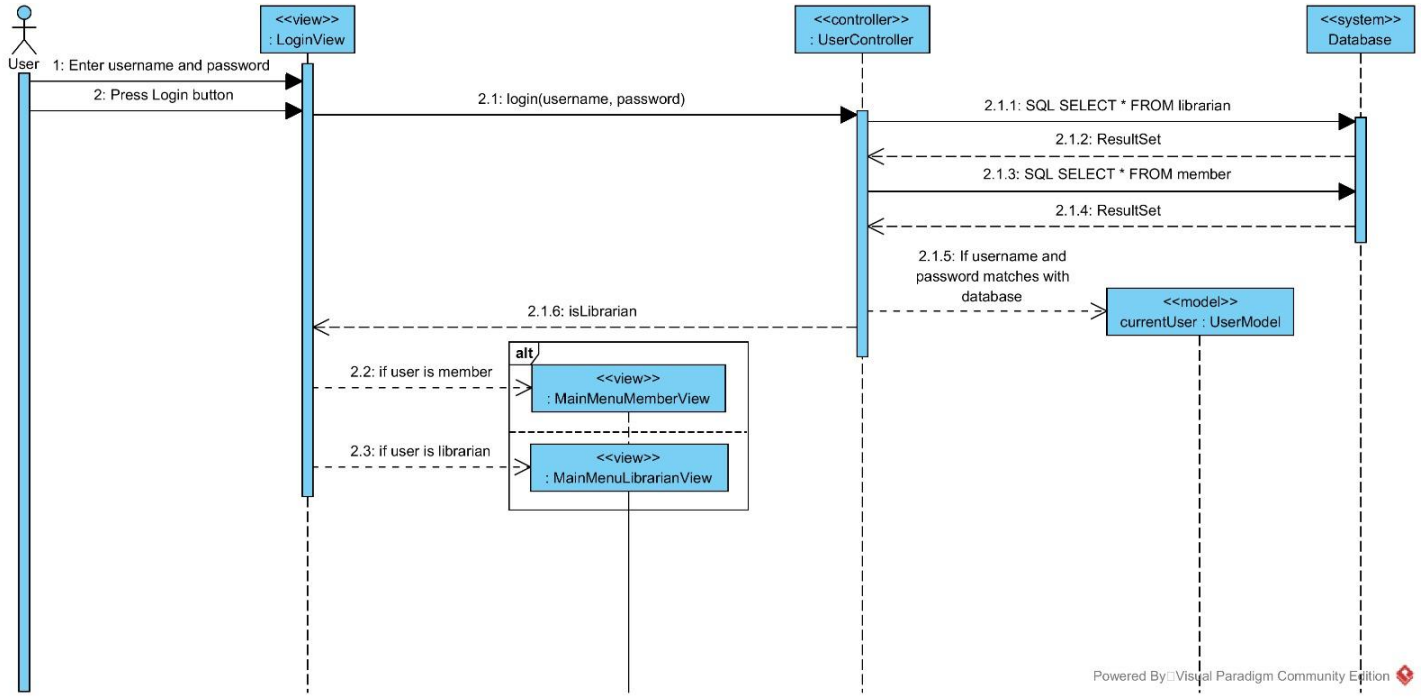

Figure 2 Login

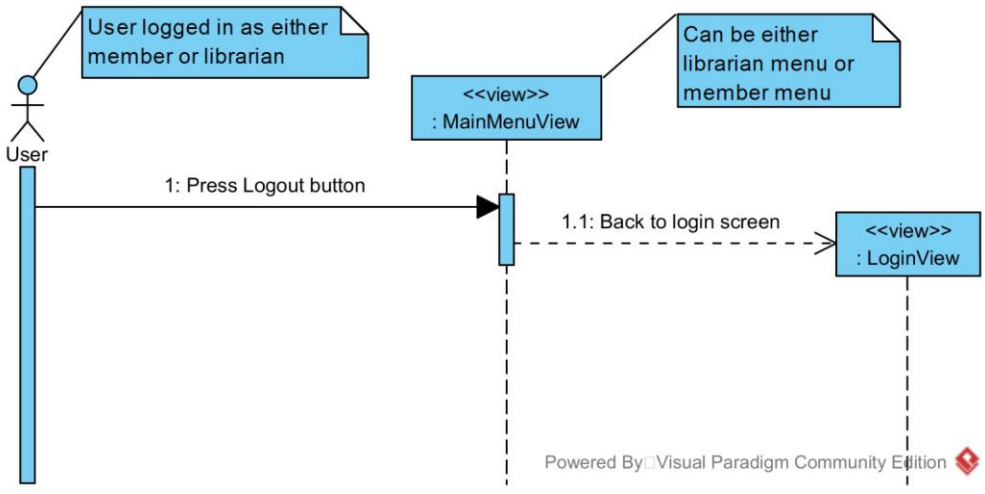

Figure 3 Logout

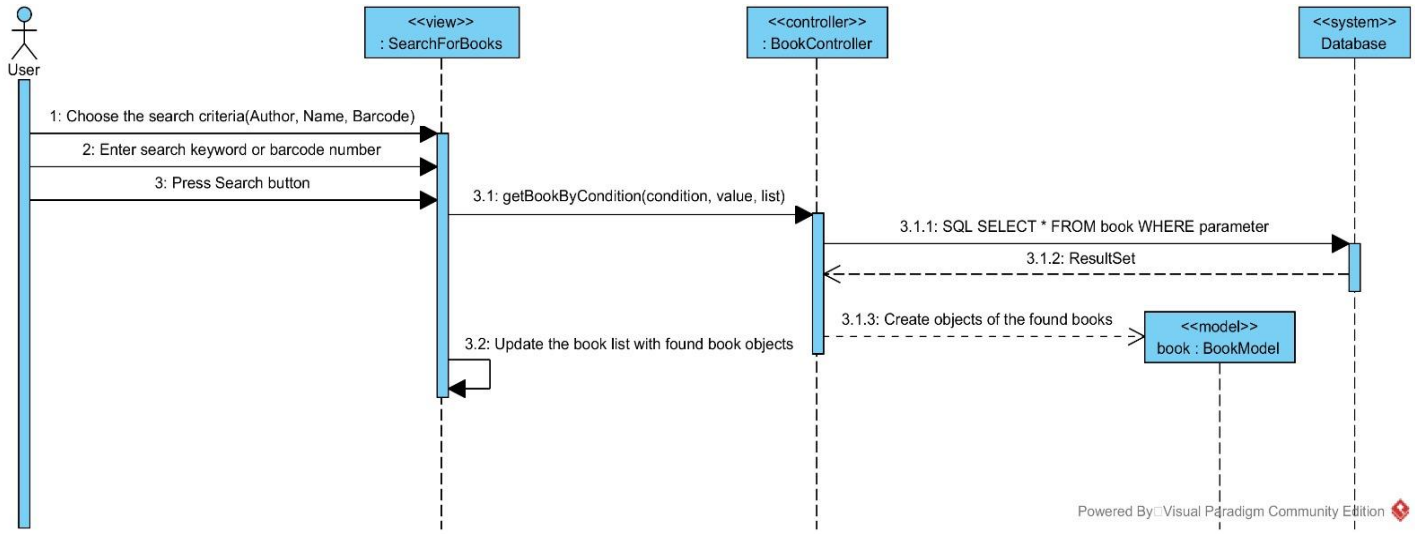

Figure 4 Search for Books

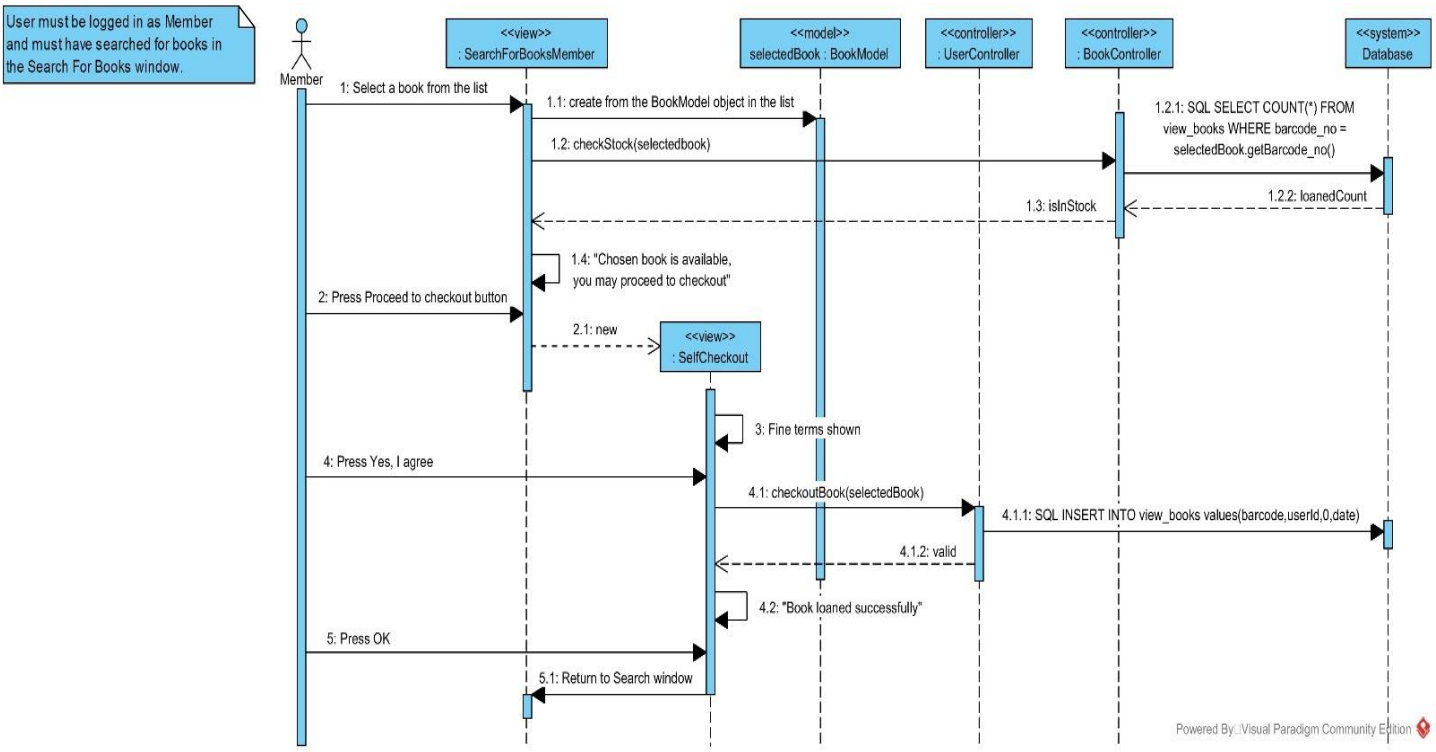

Figure 5 Self check-out

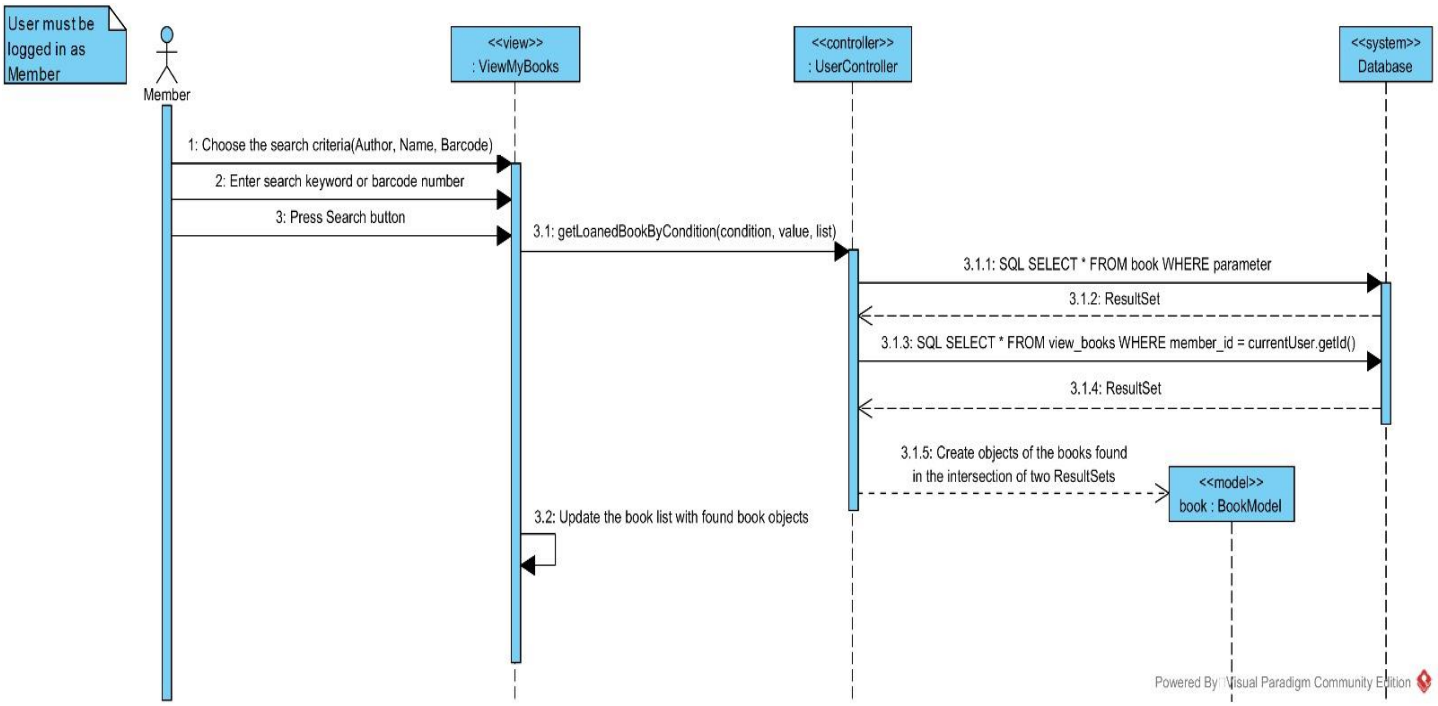

Figure 6 View My Books

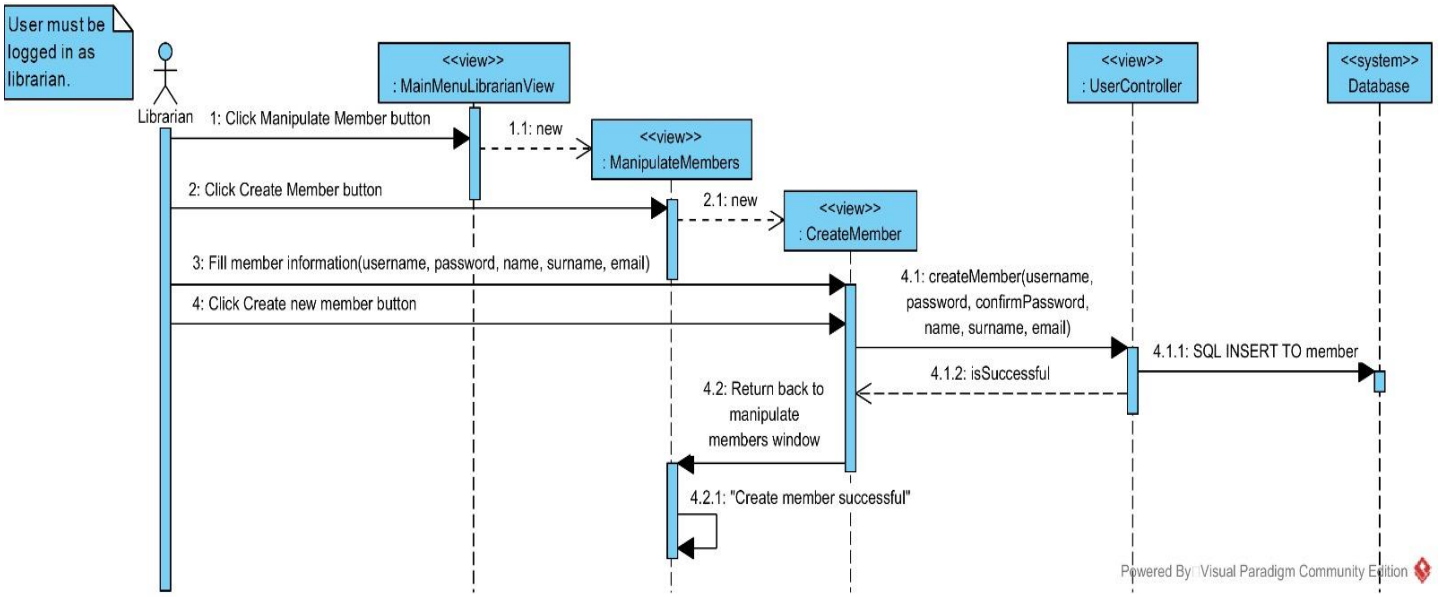

Figure 7 Create Member

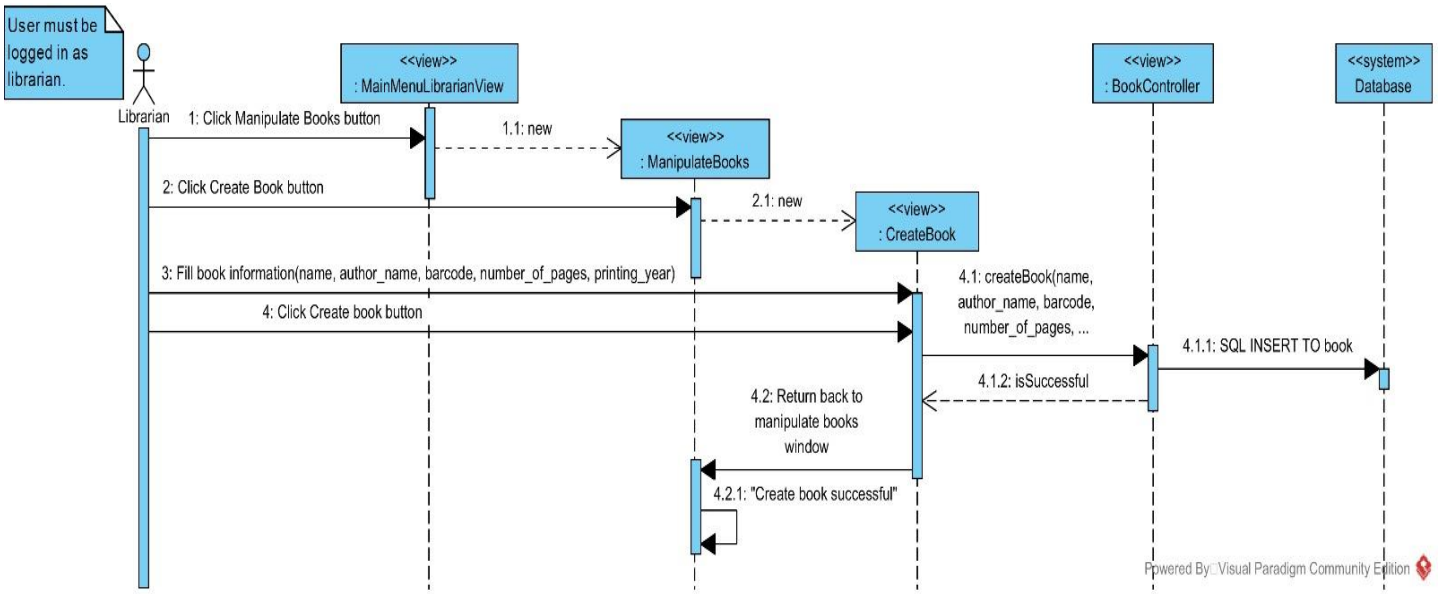

Figure 8 Create book

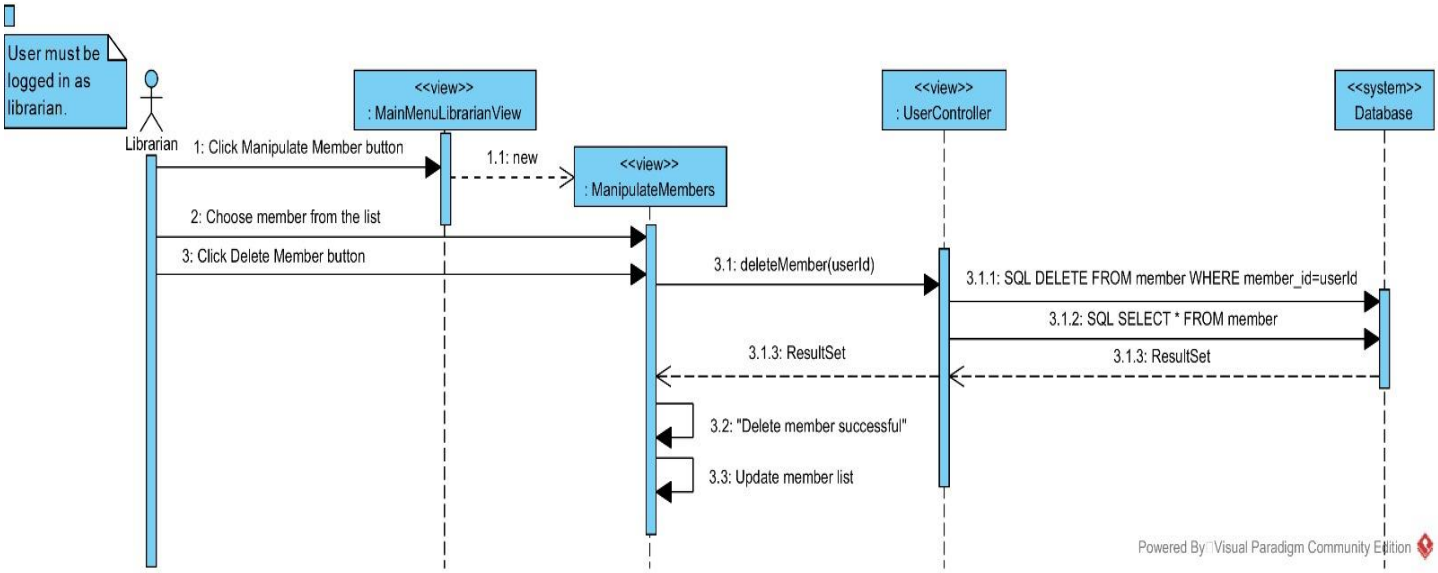

Figure 9 Delete Member

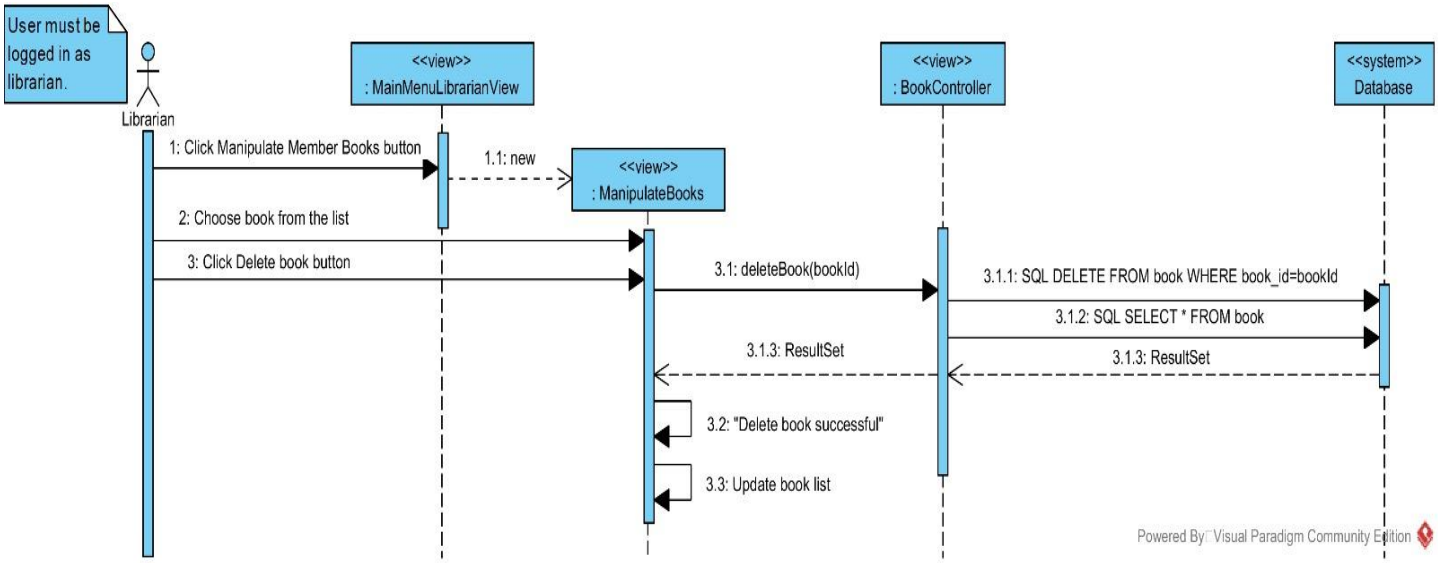

Figure 10 Delete book

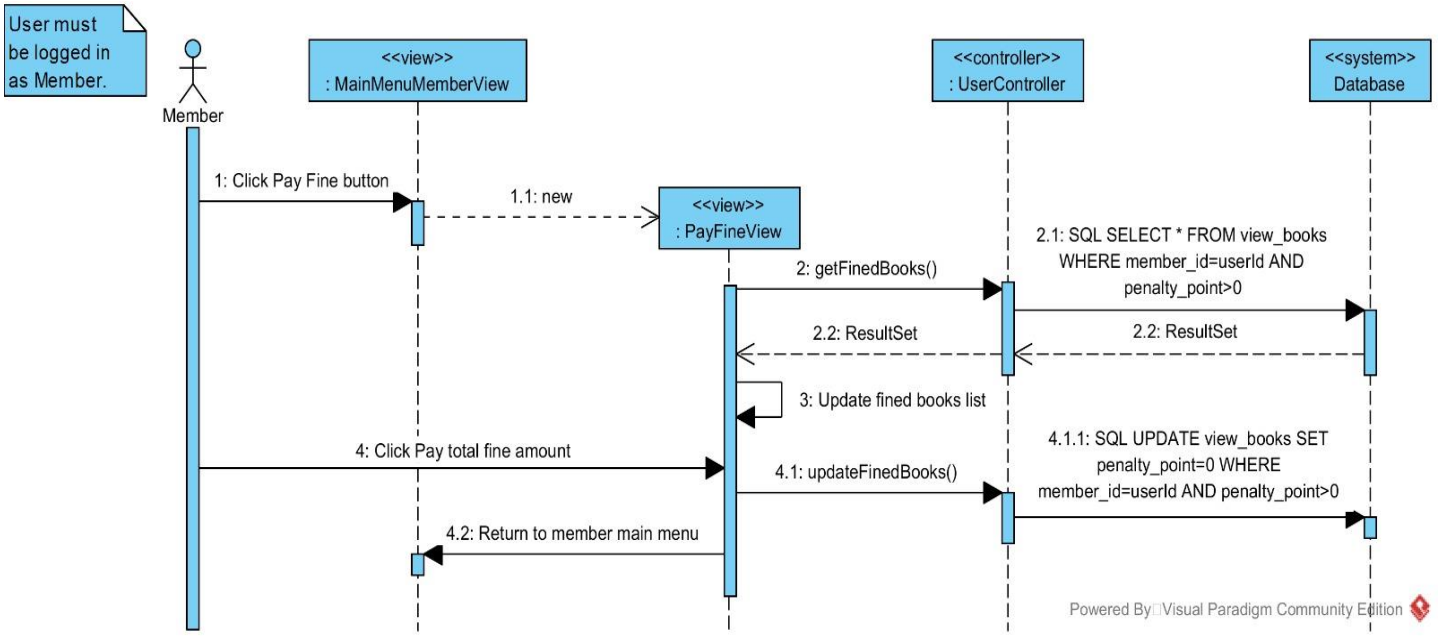

Figure 11 Pay Fine

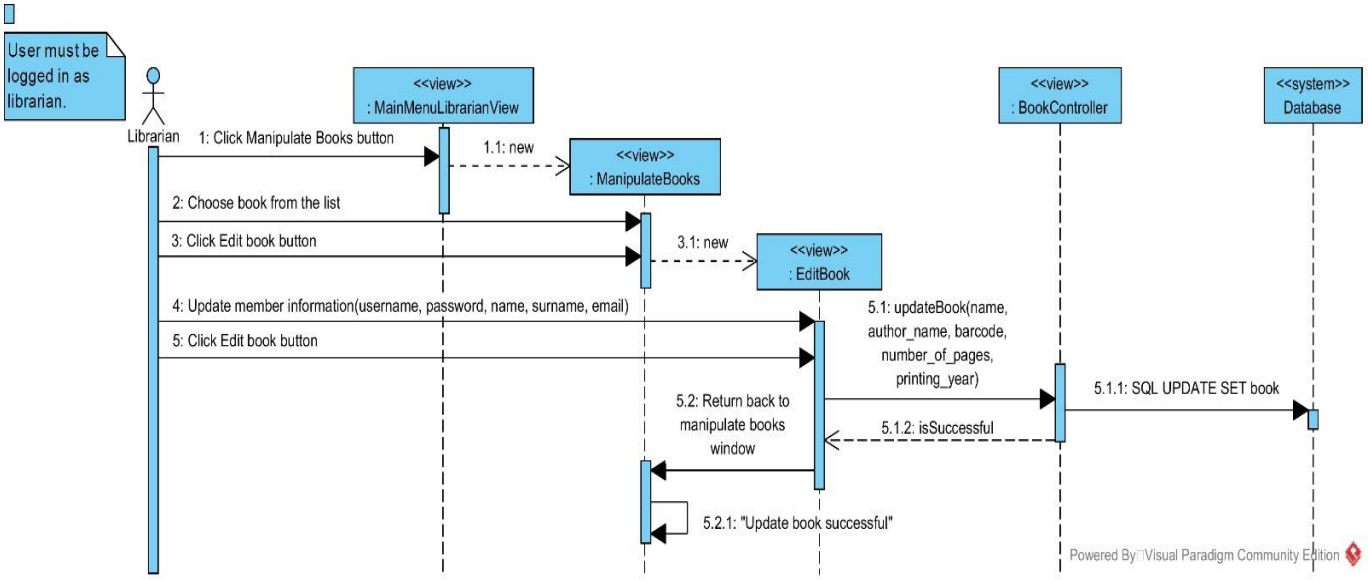

Figure 12 Edit Book

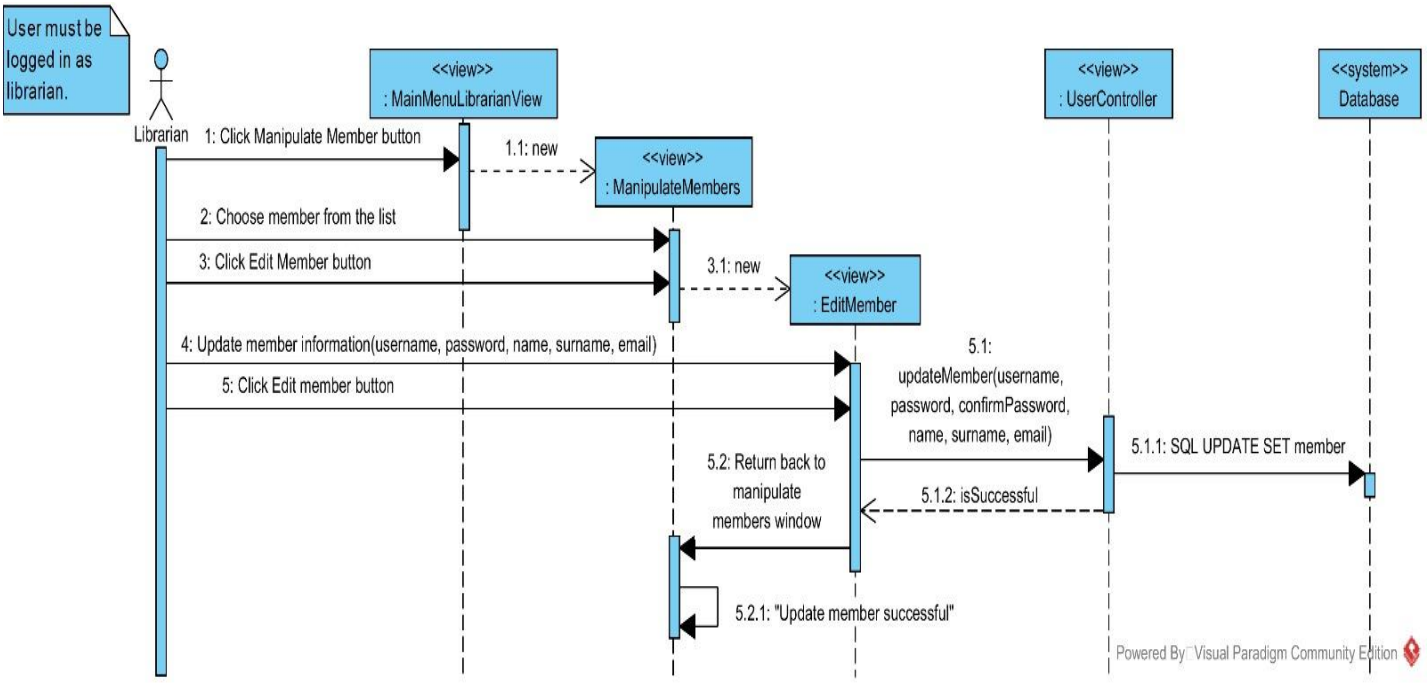

Figure 13 Edit Member

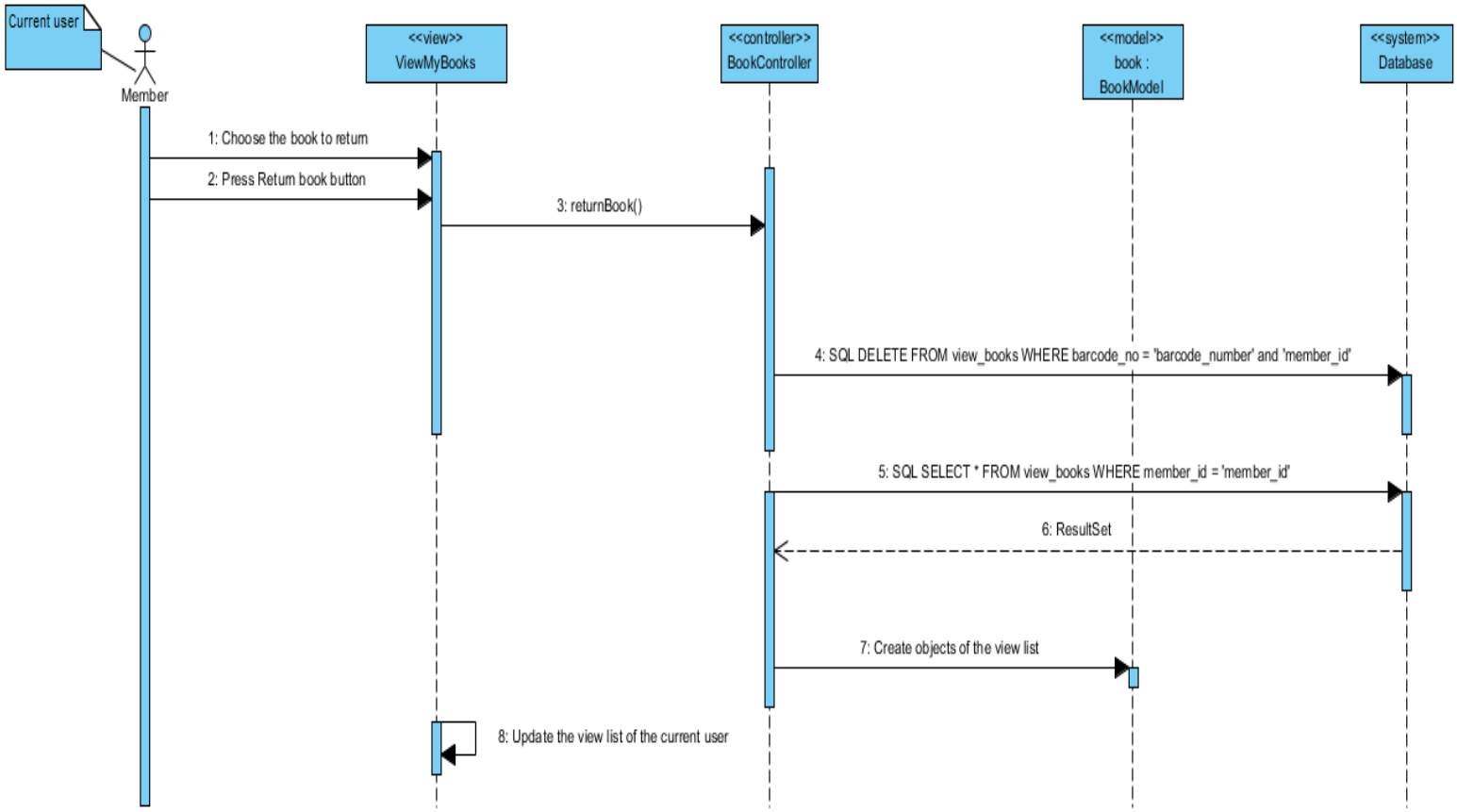

Figure 14 Self return

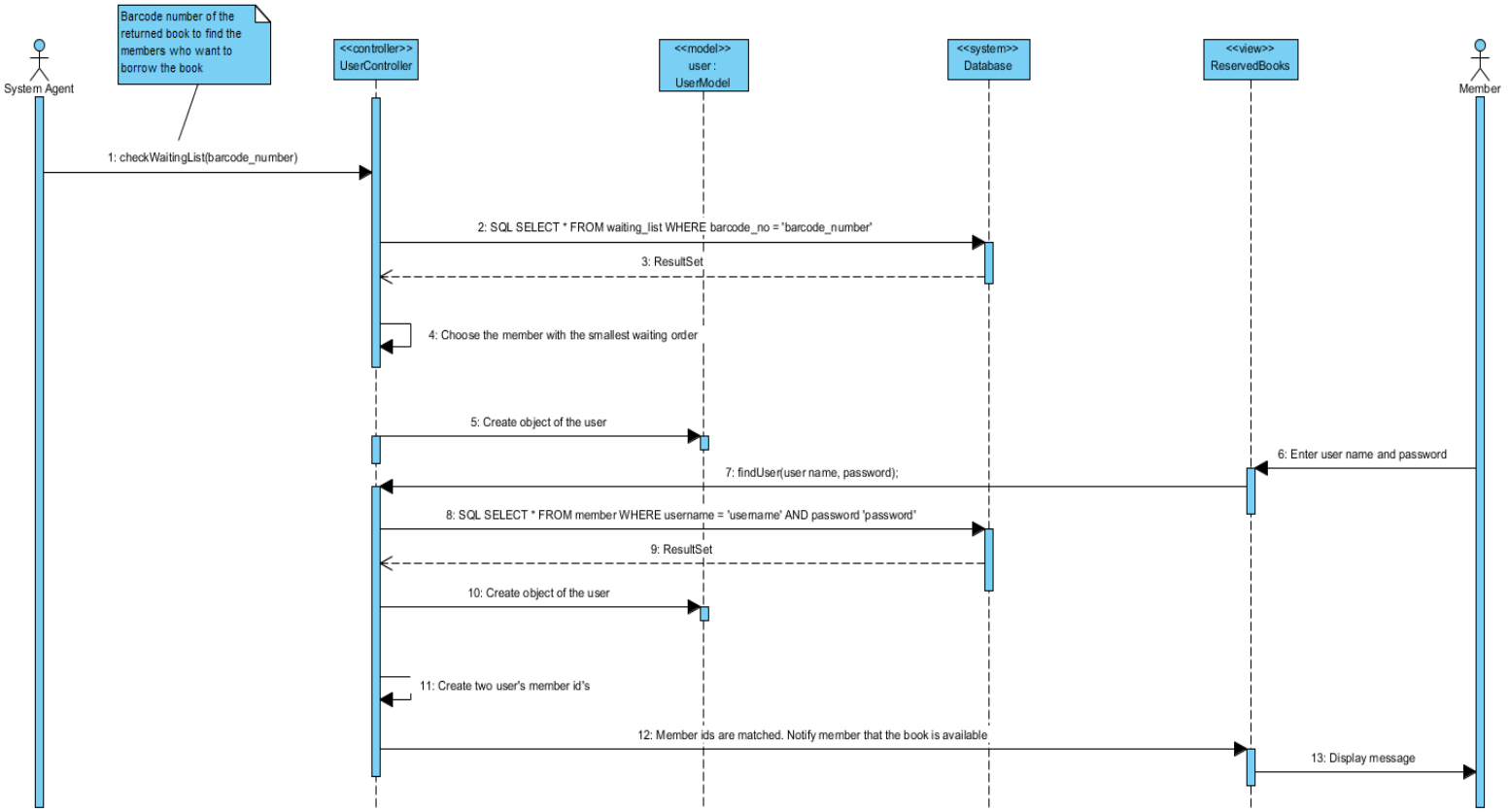

Figure 15 Notify of book being available

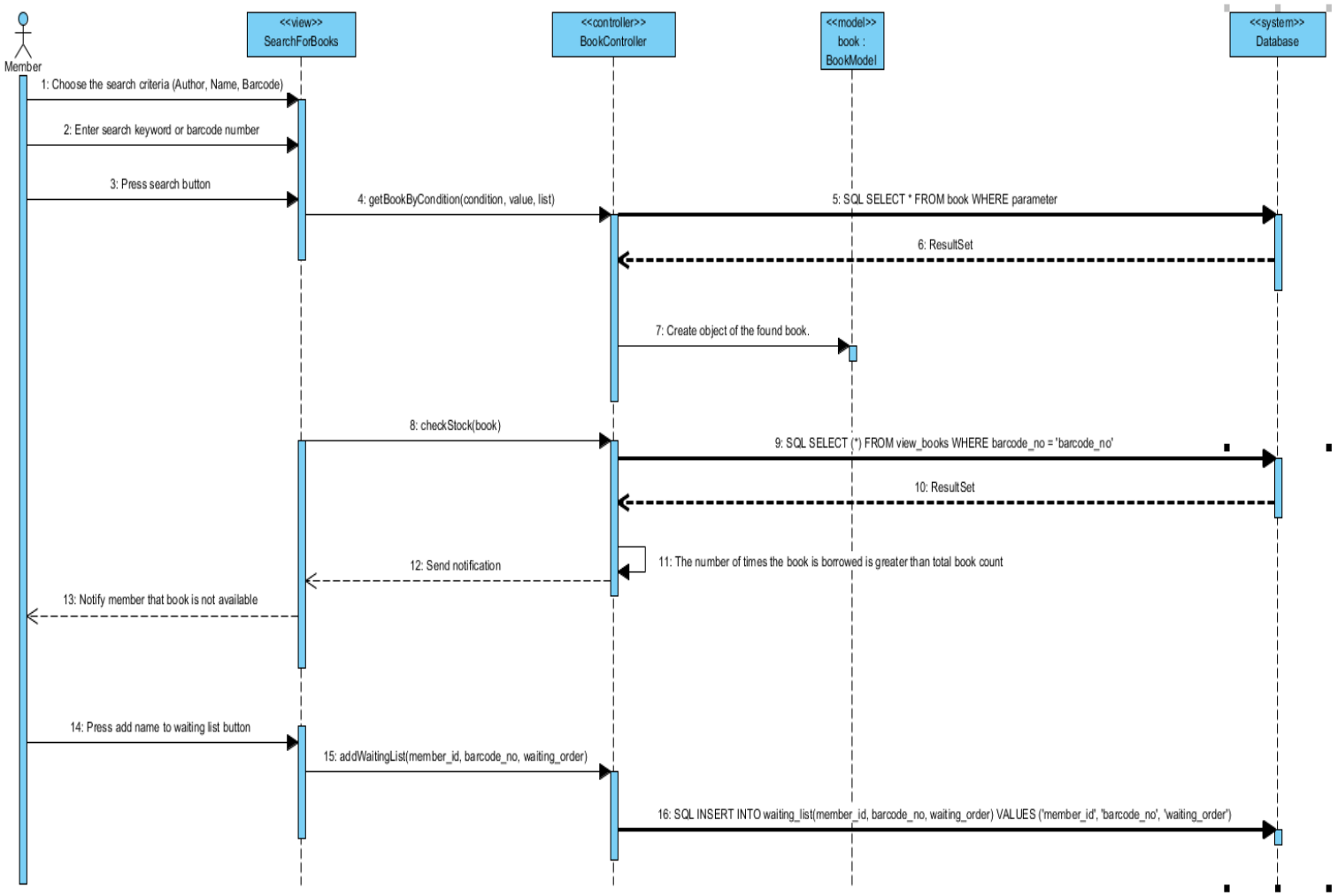

Figure 16 Add name to book waiting list

State Diagram

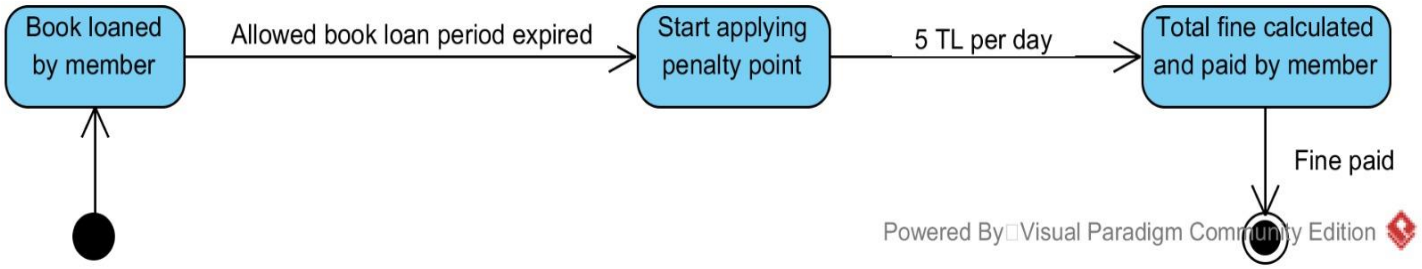

Figure 17 Pay fine

4.3 Data Model (E-R Diagram)

Relational data model is used. This model is simple and it has all the properties and capabilities required to process data with storage efficiency.

In relational data model, relations are saved in the format of tables. This format stores the relation among entities.

The tables in our project as follows

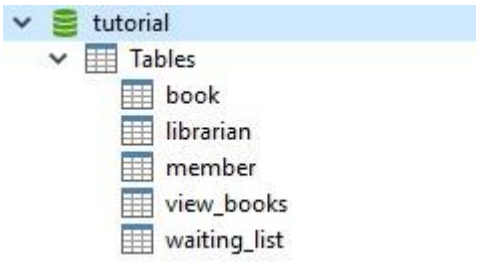

Figure 18 Tables

Attributes of tables

Name	Type	Length
barcode_no	int	11
name	varchar	20
author_name	varchar	20
printing_year	varchar	20
number_of_pages	int	11
count	int	11

Figure 19 Book table

Name	Type	Length
librarian_id	int	11
first_name	varchar	50
last_name	varchar	50
password	varchar	50
username	varchar	50

Figure 20 Librarian table

Name	Type	Length
member_id	int	11
first_name	varchar	50
last_name	varchar	50
password	varchar	50
username	varchar	20
email	varchar	50

Figure 21 Member table

Name	Type	Length
barcode_no	int	11
member_id	int	11
penalty_point	int	11
reception_time	varchar	20

Figure 22 View books table

Name	Type	Length
member_id	int	11
barcode_no	int	11
waiting_order	int	11

Figure 23 Waiting list table

The ER diagram is given below

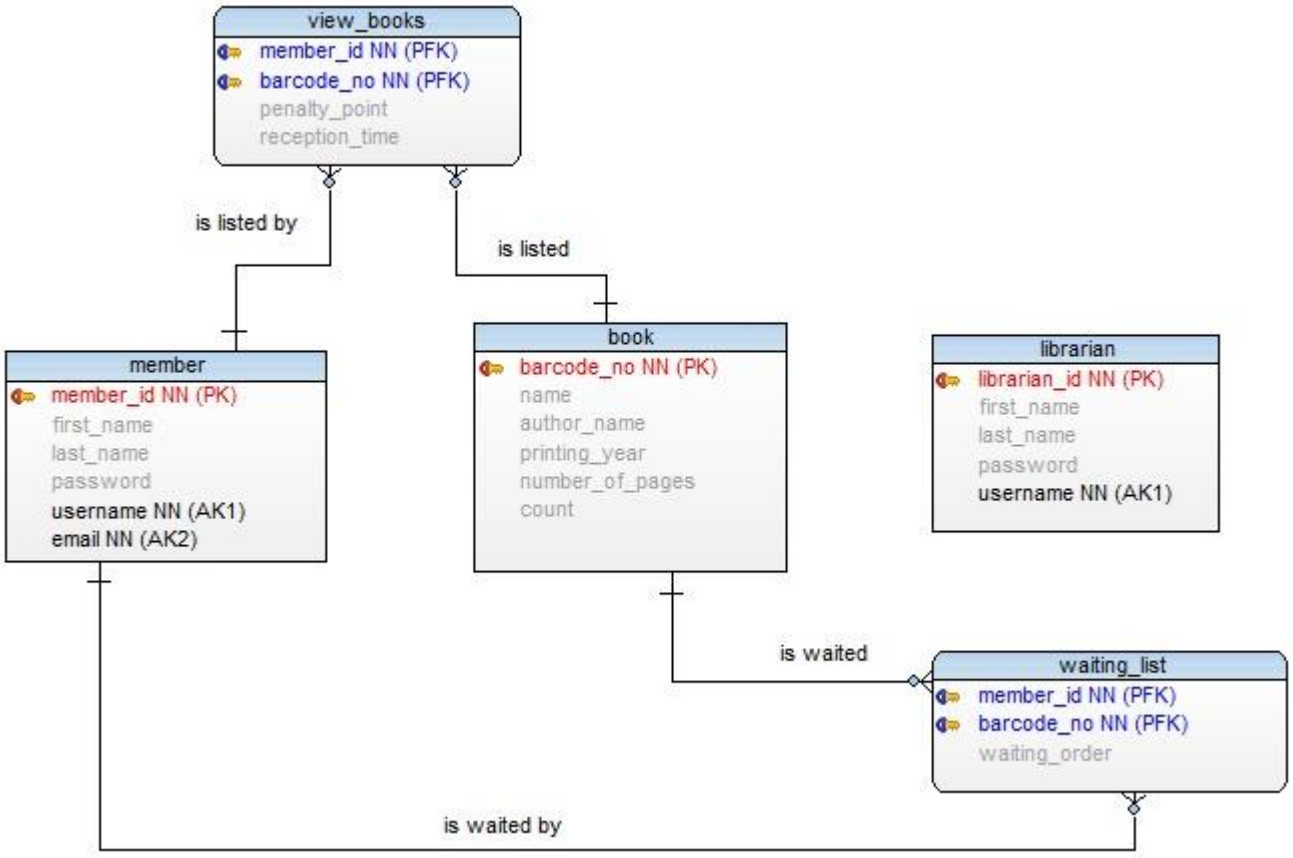

Figure 24 ER Diagram

4.4 User Interface Design

The user interface is simple and easy to understand and use. In this way, users can perform the various tasks easily and in an effective way.

All interface is represented in English.

Figure 25 Login page

Registered users enter username and password. Then, they press the Login button. If the username and password are verified, they are moved to the member page or librarian page. If the login fails, they are able to enter another username and password.

Also all users can (registered or unregistered) search for books, login is not needed.

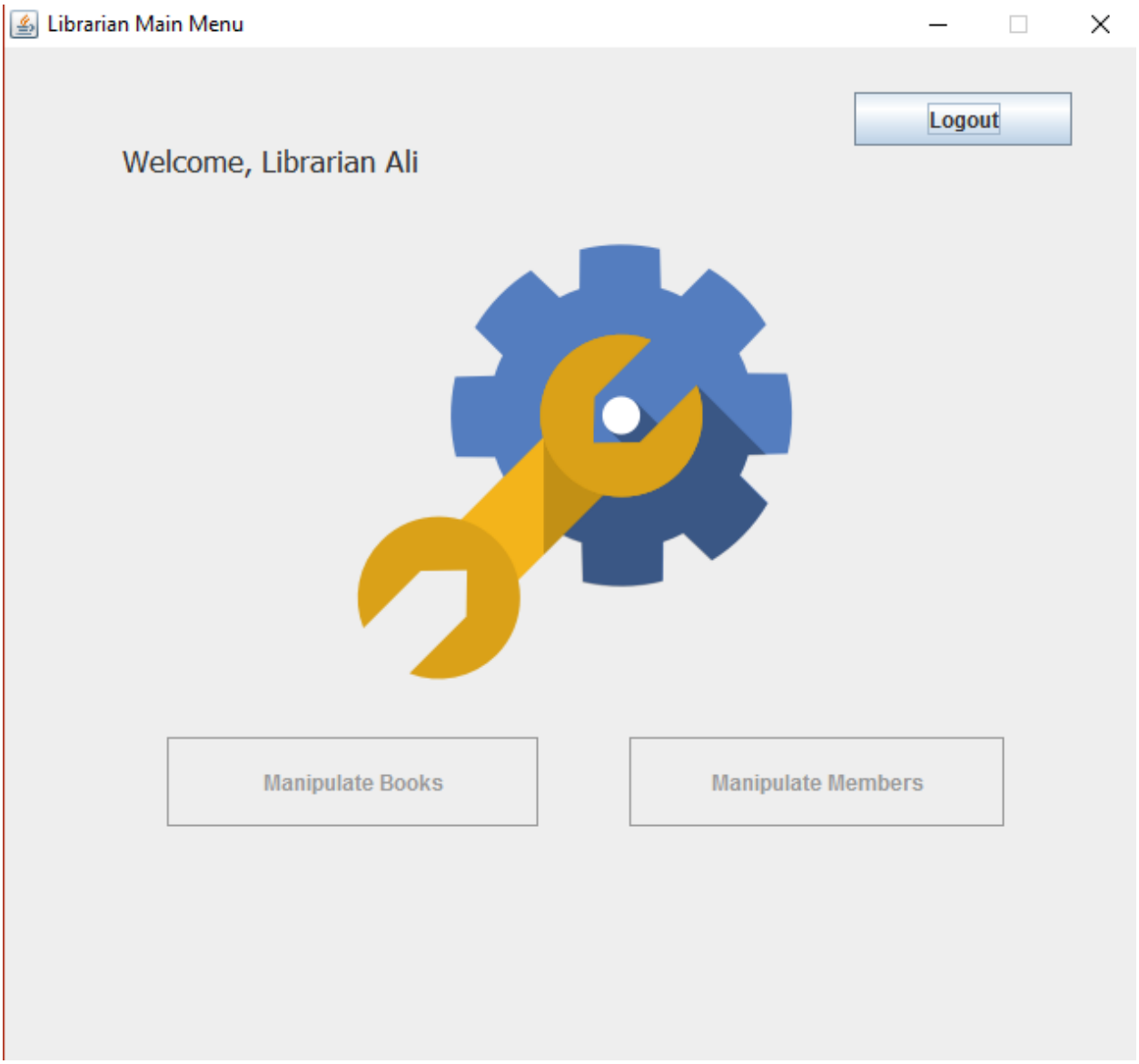

Figure 26 Librarian page

Only librarians can manipulate (add, update and delete) books and registered users (members). After login, librarian is moved to the librarian page to perform these transactions. Also, if the librarian clicks on the Logout button, he/she is redirected to Login page.

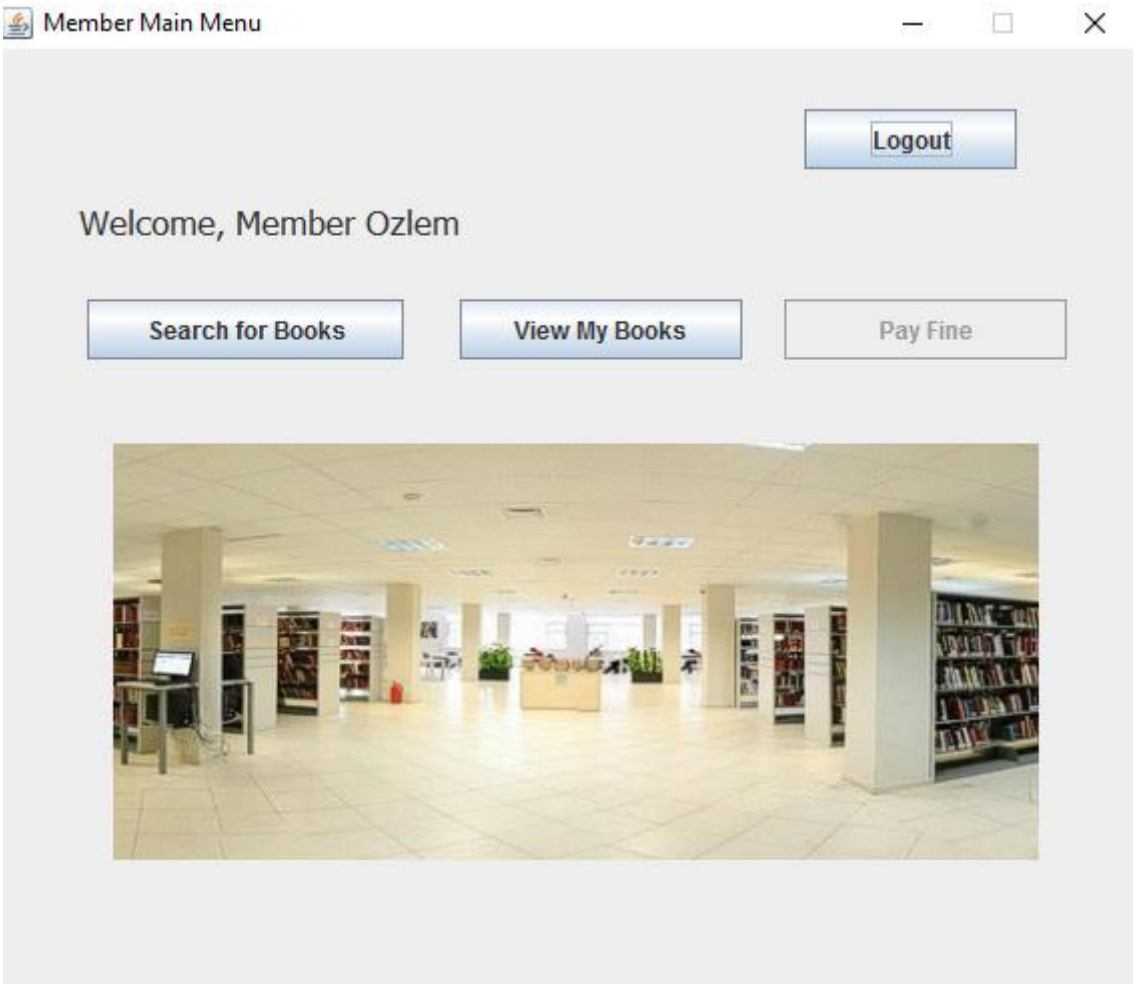

Figure 27 Member page

Members can search for a specific book, view her/his books or pay fine. After login, member is directed to member page to perform these transactions. Also, if the member clicks on the Logout button, he/she is redirected to Login page.

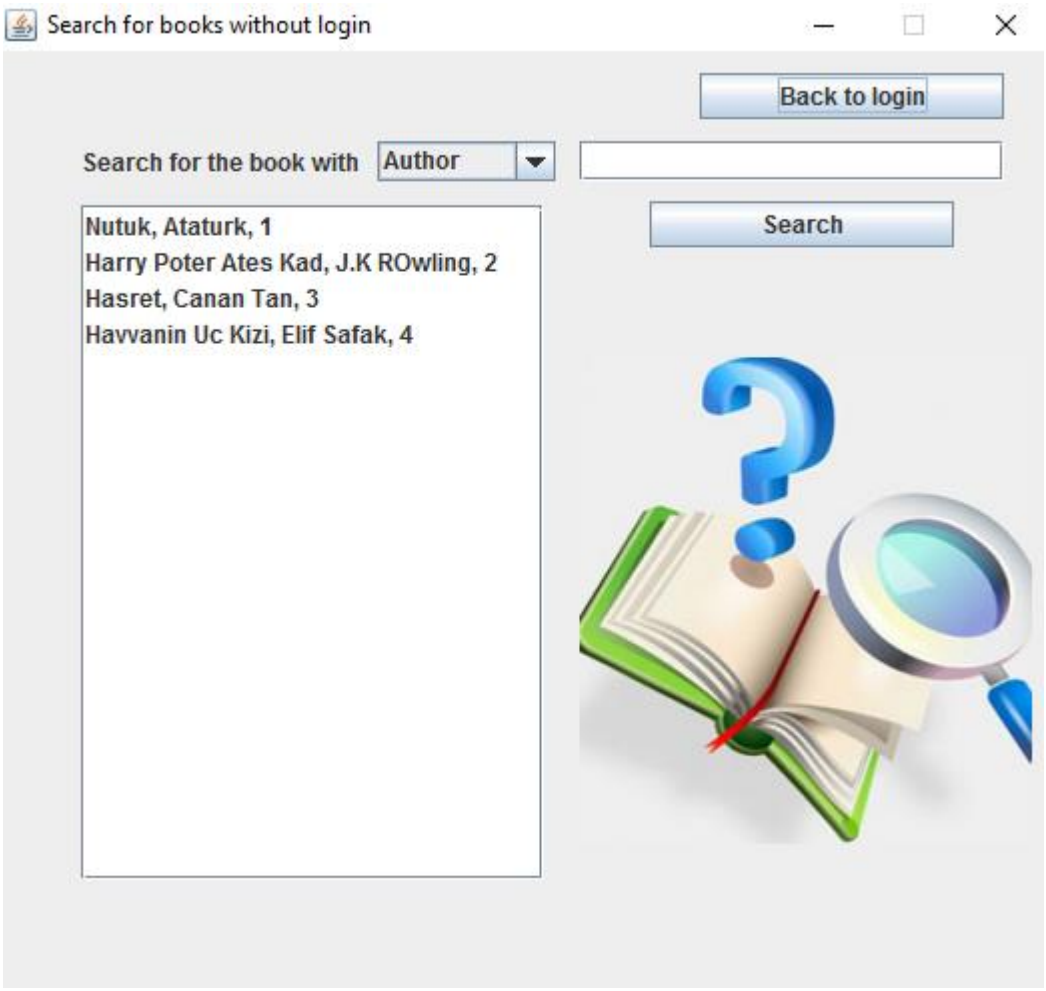

Figure 28 Search page for unregistered users

The user enters the book name, barcode or author name and clicks on search button. If the searched book is not found, user may search for another book.

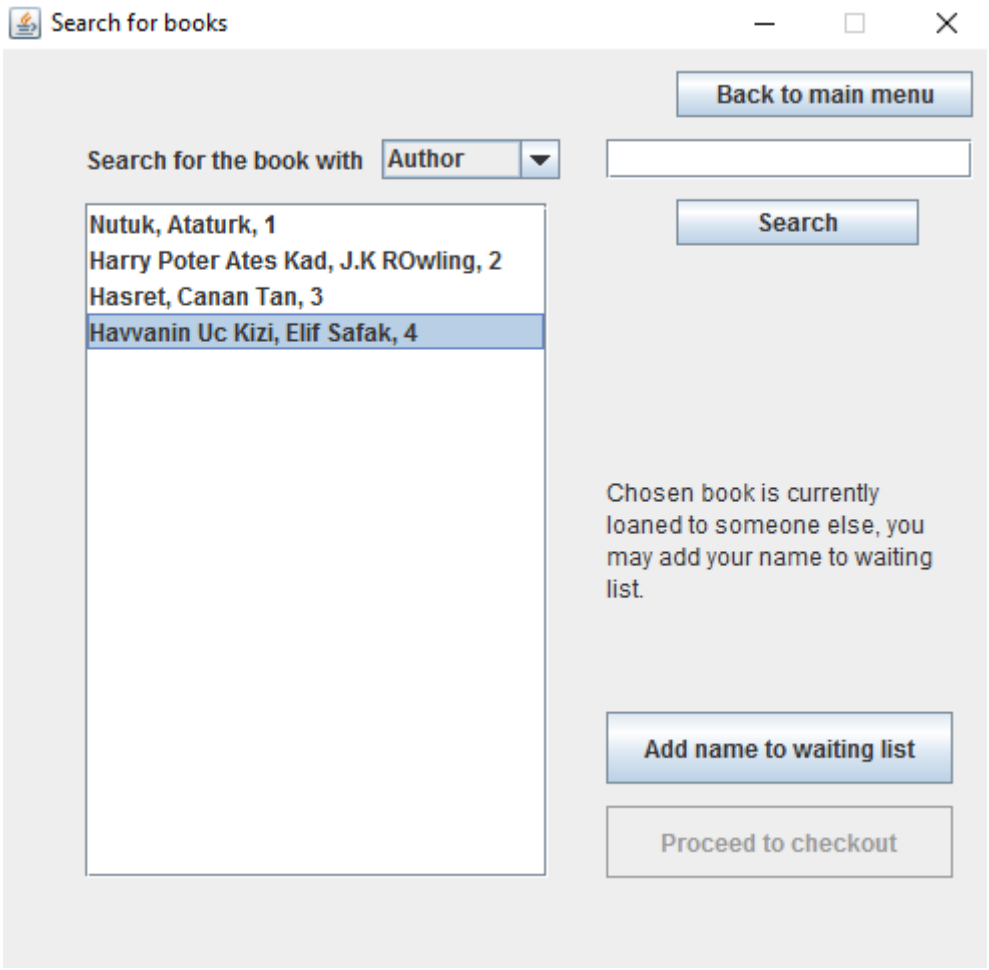

Figure 29 Search page for members

Search page for unregistered and registered users (members) is different. If the book is currently available, member can check out the book. If it is not available, he/she can add name to waiting list. If the member clicks on Back to main menu button, he/she is redirected to member page.

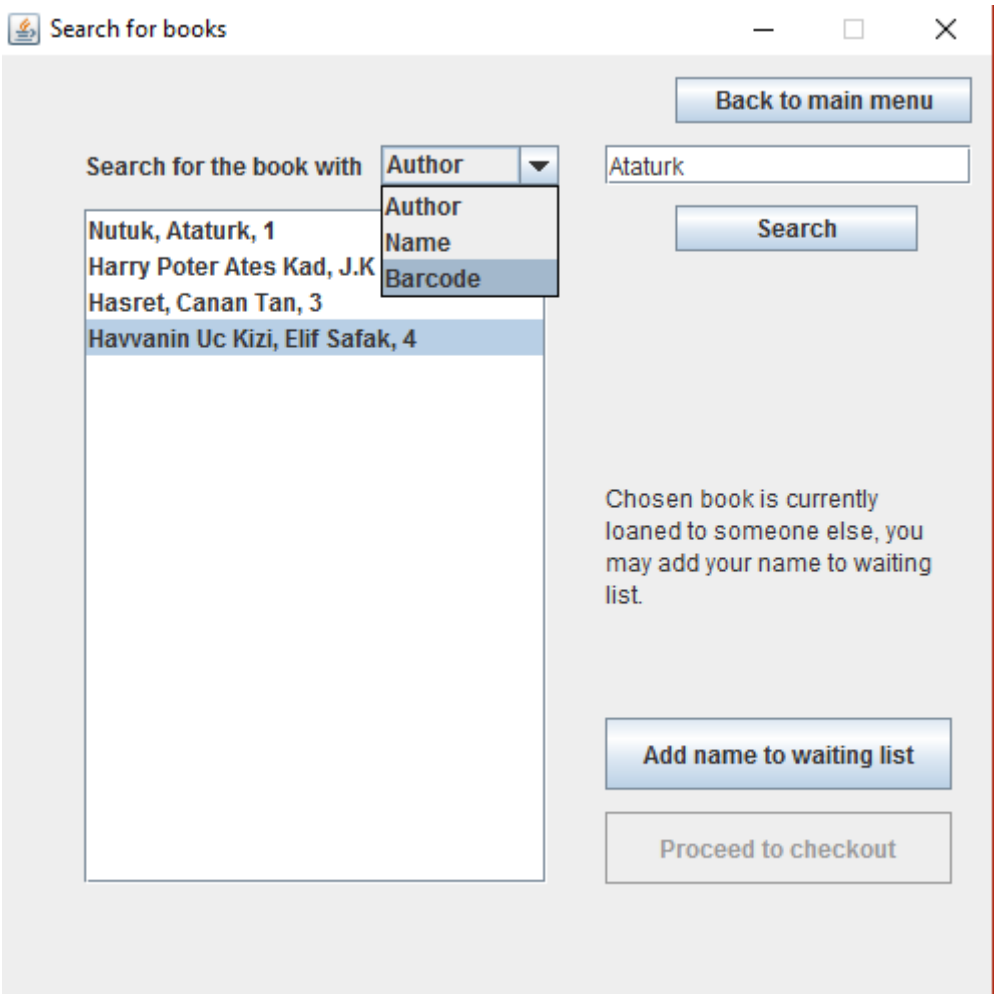

Figure 30 Search page for members

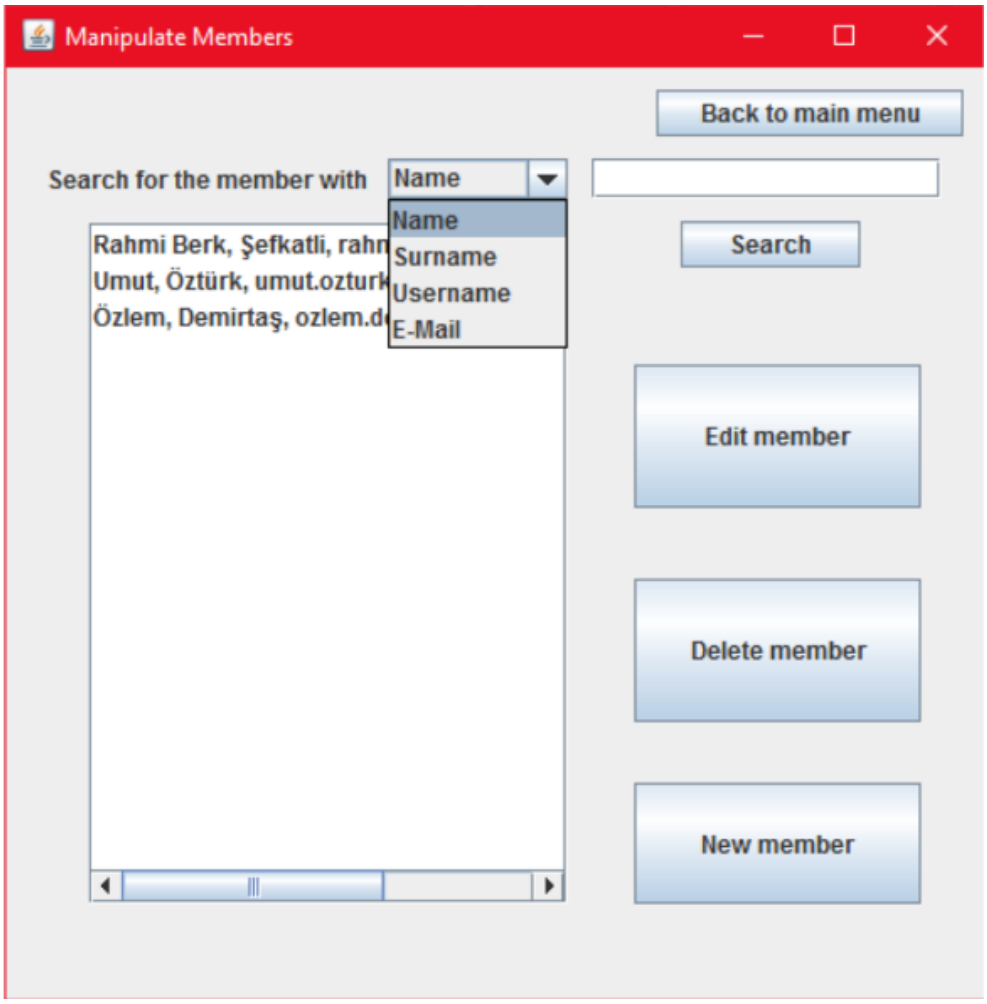

Figure 31 Manipulate members page for librarians

As shown in Figure 14, members can add a new member to library database, updates member information or remove member from database. If librarian clicks on Back to main menu button, he/she is redirected to librarian page.

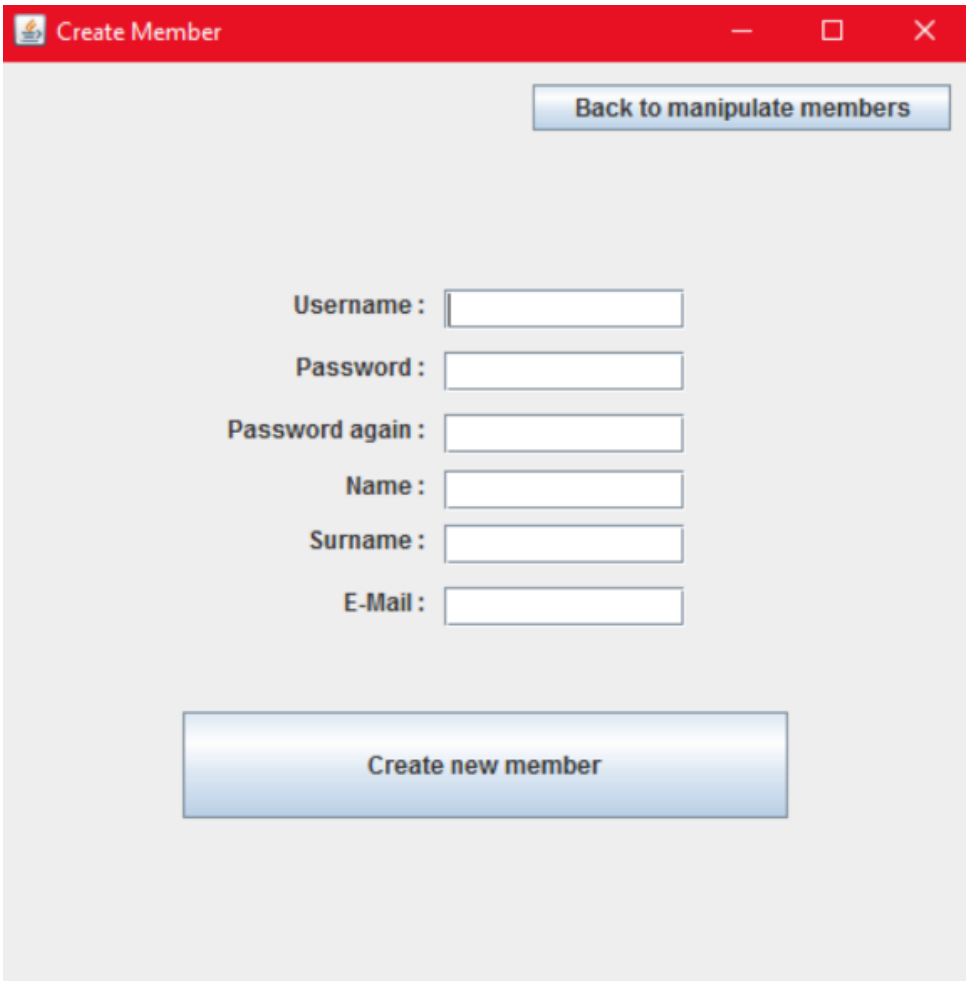

Figure 32 Create new member page for librarians

To create a new member, librarian enters the user's information (username, password, name, surname and e-mail) and click on the create new member button. If the new member record is already in the database, librarian can't add it again.

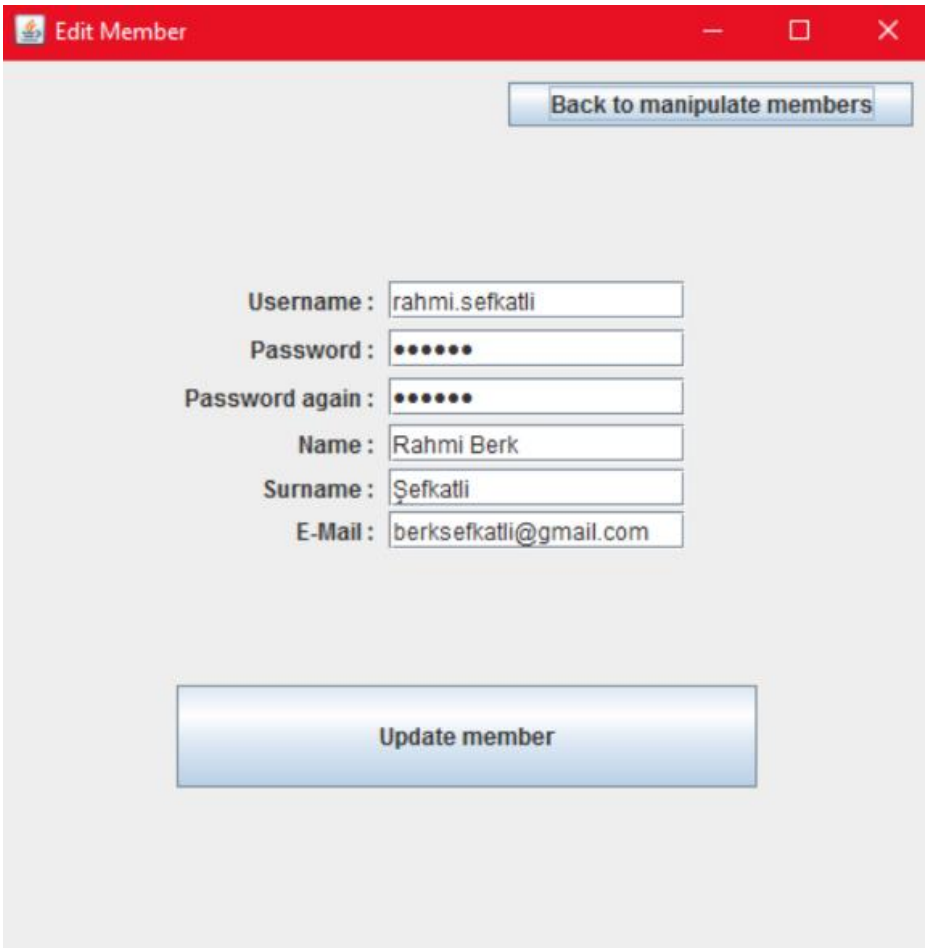

Figure 33 Update member page for librarians

The librarian enters the member's information to be updated and clicks on the update member button.

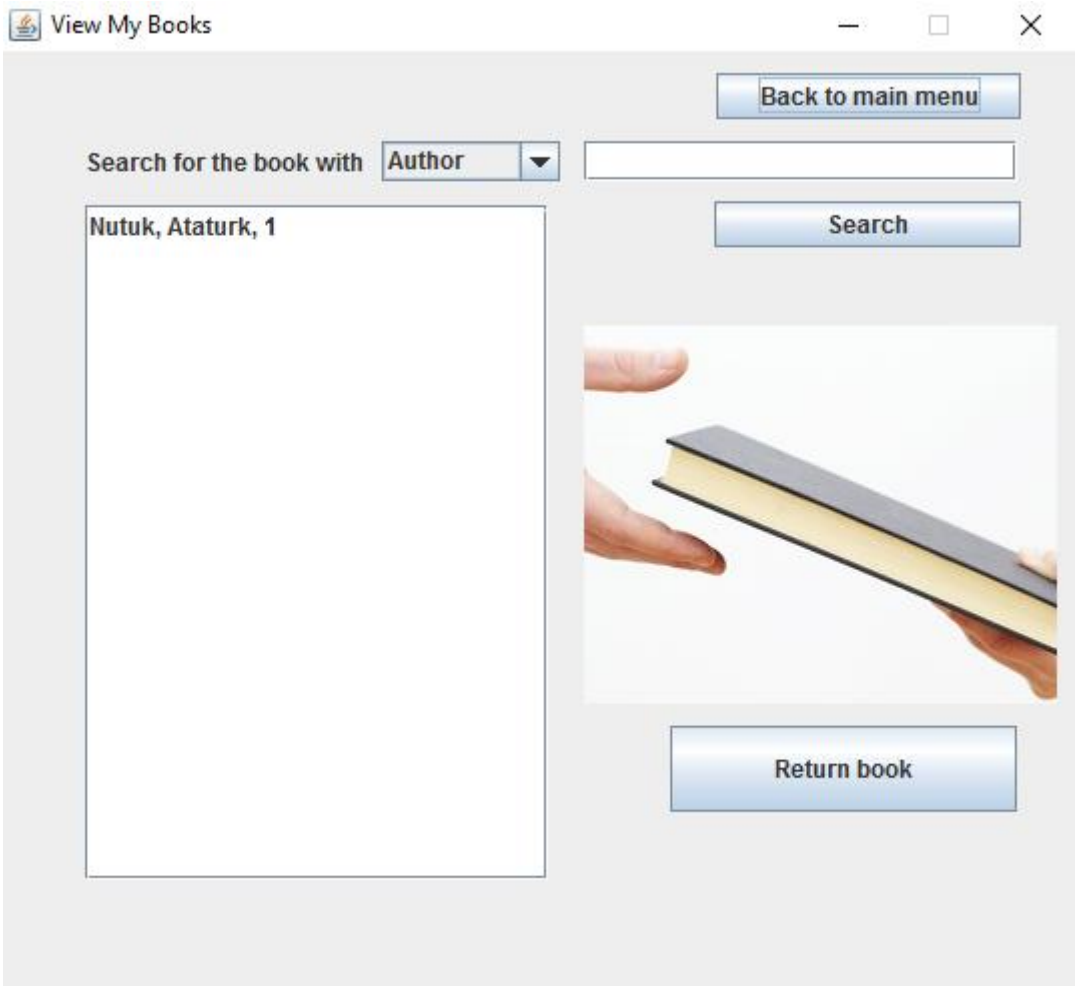

Figure 34 View my books page for members

The member clicks on the view my books button to see her/his loaned books. He/she chooses and returns the loaned book back to library by using Return book button. Also, he/she can search for a specific book in her/his loaned books.

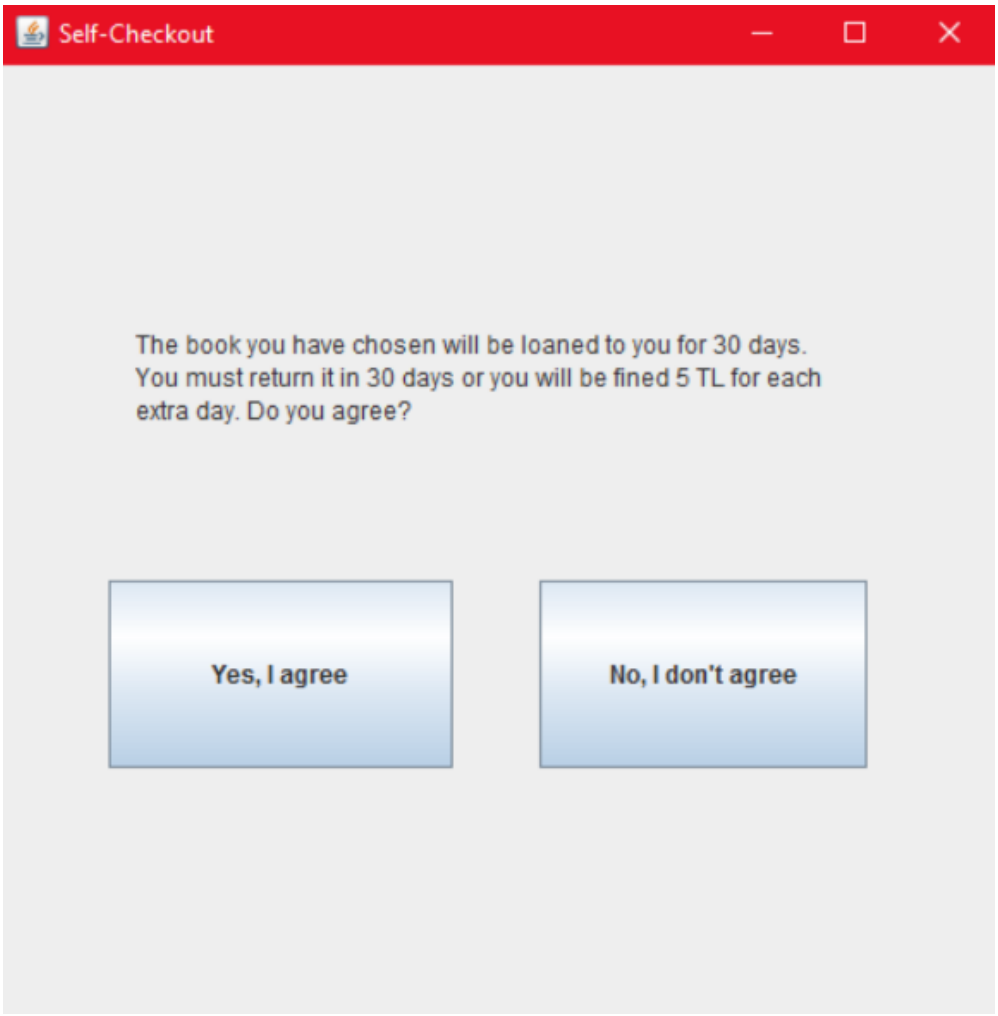

Figure 35 Self-Checkout page for members

If the member clicks on the Proceed to checkout button in search page for members (Figure 12), system gives an error about fine and asks member her/his decision. If the member agrees, the book is added to the member's loaned books list.

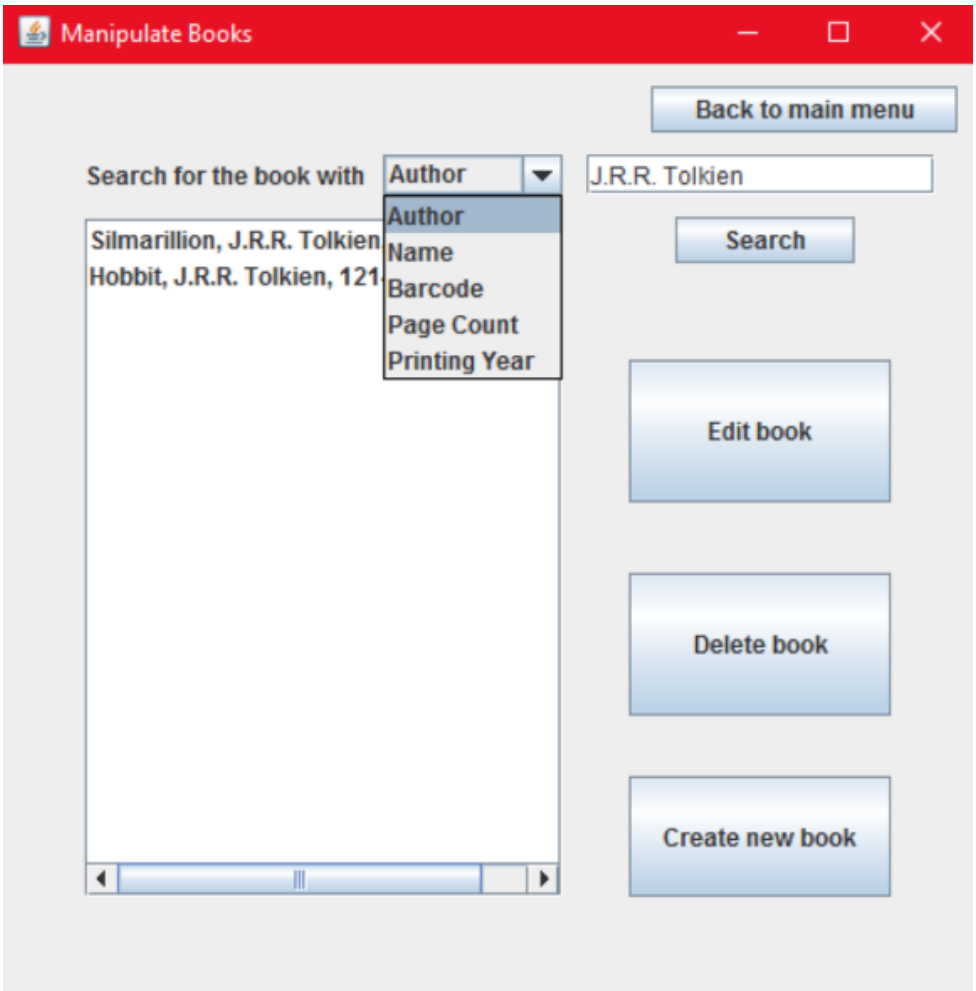

Figure 36 Manipulate Books page for librarians

As shown in Figure 19, members can add a new book to library database, updates book information or remove book from database. If librarian clicks on Back to main menu button, he/she is redirected to librarian page.

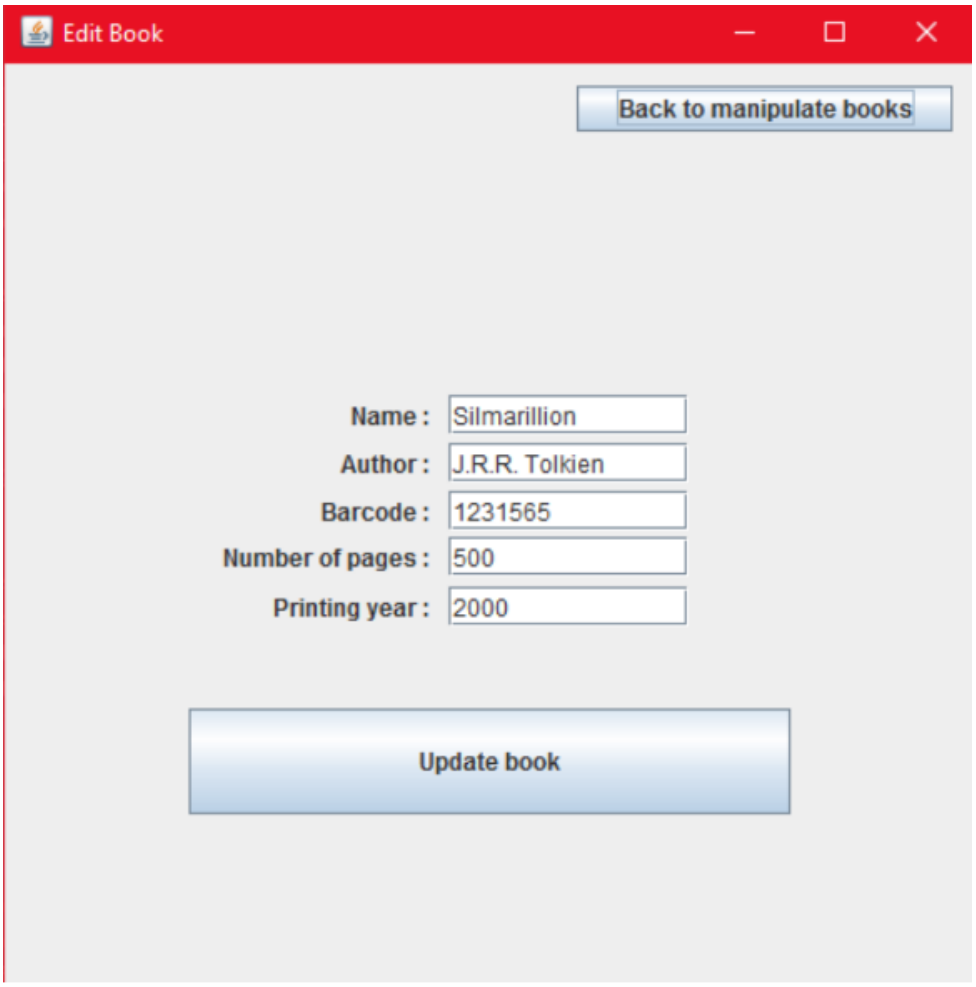

Figure 37 Edit book page for librarians

The librarian enters the book's information to be updated and clicks on the update member button.

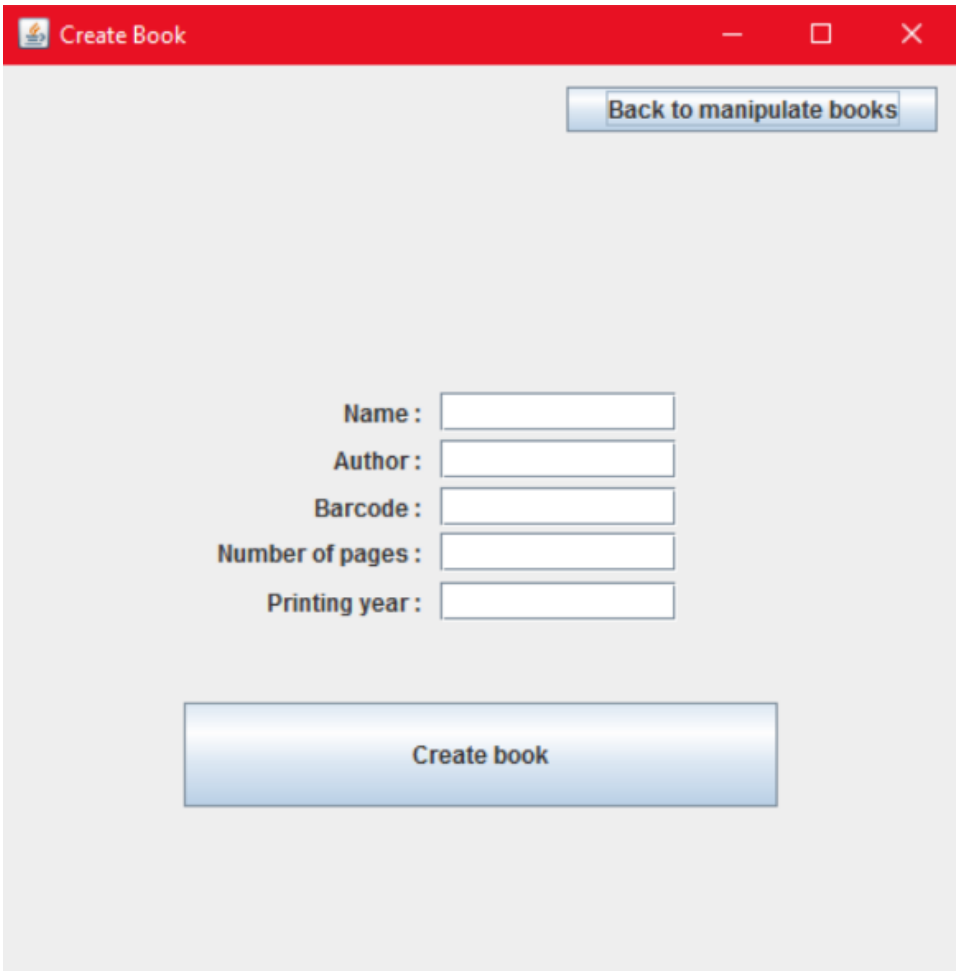

Figure 38 Create book page for librarians

To create a new book, librarian enters the book's information (name, author, barcode, number of pages and printing year) and click on the create book button. If the new book record is already in the database, librarian can't add it again.

5. Requirements Traceability

Traceability Matrix	
Requirements	**Classes**
Members and librarians login to the system.	UserModel/UserController / LoginView
Members and librarians logout from the system.	UserModel/UserController / LoginView
Librarians add, update and delete books.	BookModel/ManipulateBooks/EditBook/CreateBook
Members, librarians and unregistered users search for books.	BookModel/SearchForBooks/SearchForBooksMember/ BookController
Members view their books.	UserModel/BookModel/UserController /ViewMyBooks
Members check-out books.	BookModel/SearchForBooksMember/SelfCheckout
Members return books.	UserModel/BookModel/ViewMyBooks/
System determines which members should be fined for the late delivery of the books.	UserController/UserModel
Members pay fine.	PayFineView/UserModel/UserController
Members queue for currently not available books.	UserModel/BookModel/SearchForBooksMember
System notifies members when the books they wanted to loan becomes available.	UserController/BookModel/UserModel/NotifyView

Figure 39 Traceability Matrix