HACETTEPE UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

BBM 487 SOFTWARE ENGINEERING LAB

SOFTWARE DESIGN DOCUMENT

Özdeş Öztürk	Ekin Kandemir	Kadir Bulut
21228635	21228434	21228137

LIBRARY BOOK LOAN SYSTEM
Software Design Description

1. Revision History

Version	*Date*	*Author*	*Change Description*
1.0	23.04.2017	Özdeş Öztürk, Ekin Kandemir, Kadir Bulut	This document consists of the deeper design details of the Library Book Loan System that we have been constructing.

2. INTRODUCTION

2.1 Purpose and Scope

The purpose of this document is giving software design details of our library system and explain the most significant points about these details. The main purpose is to explain the technical properties of the system, as well as the decisions made during the design process of the system to someone who may be using our system in the future. So we prepared this document considering the possible confusions that a future customer might have. The document also contains the design of the use cases of the system and deeper explanation about the use cases that we have already implemented so far. We also provide an explanation about the design of the interface of our system and explain the already implemented use cases on this interface.

2.2 Document Overview

As we already shared the planned requirements and planned design of the components of our system in system wide requirements specification document. In this document, we will give a deeper explanation about the design of our system by taking that document as a reference and show the interface of our system with the screens that a user may encounter, we will be providing sequential diagrams for a better understanding so that the document will provide a meaningful information for anyone who may need to read and understand our system in the future.

2.3 System Overview

In Library Book Loan System, the admin is capable of searching for a book, updating information of a user, updating information of staff, updating a book, deleting a book and adding a book. These are the use cases we have implemented so far as well as login and logout operation. These operations are developed in a way that enables user to have a smooth transaction between each process and the system is quite user friendly. Each operation can be done quite quick and in a systematic way. The admin will be able to search for a book and reach the information of a book that contains its name, author, publisher and quantity. A new book can be added to the system or deleted from the book list. The user can display list of books and list

of users and there is separation between these lists of the users since admin can view staff and member lists separately. When a new book is added, if every single information is the same as a previously existing book in the system, there will not be an error message as expected, but the quantity of the book will be updated instead. Admin will also have the authority to delete and add new member or staff to the system as well as deleting them when/if needed. We will be giving deeper information about the use cases when we explain our interface.

2.4 Definitions, Acronyms, and Abbreviations

Term/Acronym	***Definition***
Staff	Staff is anyone who is the library employee and in further development of our project, staff will have the capability of adding, deleting or updating books as well as dealing with the borrowed books from the members.
Library Member	The member is the least authorized user of the system who is a student or scholar who is using the system to display books.
Administrator	Administrator is the most authorized user of the system which has the capability of adding, deleting updating information about users of the system as well as books. The administrator is also responsible from the general mechanism of the system.

2.5 References

Even though there are slight changes made on these documents, our references for the given details of this document are the project plan document, software requirements document and architectural notebook document that we previously shared via github.

3. Design Constraints and Decisions

The most important property of the system is there is not a specific and complicated need in order to make this project work. So there will not be strict dependencies on other system. It will be working accurately in most basic systems. We used(and will keep using) php programming language for development and MySQL database is being used to hold and manipulate the data. Our application is planned as a web application so the only need is having internet connection.

About constraints, currently, there is no limitation of the number of staff, library members or books. So there will not be any restrictions about the quantity of anything. On the other hand, the system should always direct the user and inform them about the current operation. When user wants to delete or add a book or user, they should be sure that the operation was done successfully afterwards. Therefore, the system should always display windows to ask for confirmation and if the confirmation is taken from the user, then the system should also display a screen informing the user about the success of the related operation. So clearly stated confirmation messages is quite significant for our system since we want the system to direct the user accurately and make it quite clear for the user that an operation has been successful or unsuccessful.

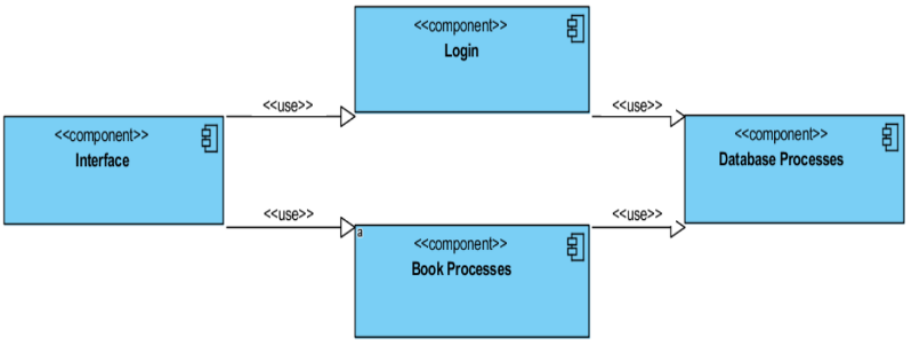
As we also stated in System Wide Requirements document, whenever the user wants to apply an operation, the answer time of the system should be as quick as possible and considering the implemented parts of our project, we can say that we have achieved that goal. The system works quite fast for any operation that the user wants to do. On the other hand, beside the response time, we also wanted to have a quite smooth and plain interface which will not distract the user. Our design is exactly what we claimed in System Wide Requirements Document and it does not contain any kind of an image or design that has a potential to distract the user.

To give a brief information about different users of the system, admin will have the capability of doing every single operation in the system considering he is the most authorized user of the system. To act as an admin, the user has to login with the related name and password, as defined in the system. The library employees will also need to login with their accurate names and passwords, and they will have the authority to delete, add, update books. Unregistered users who are the members(scholar or student) of the system can only do the search operation for the books.

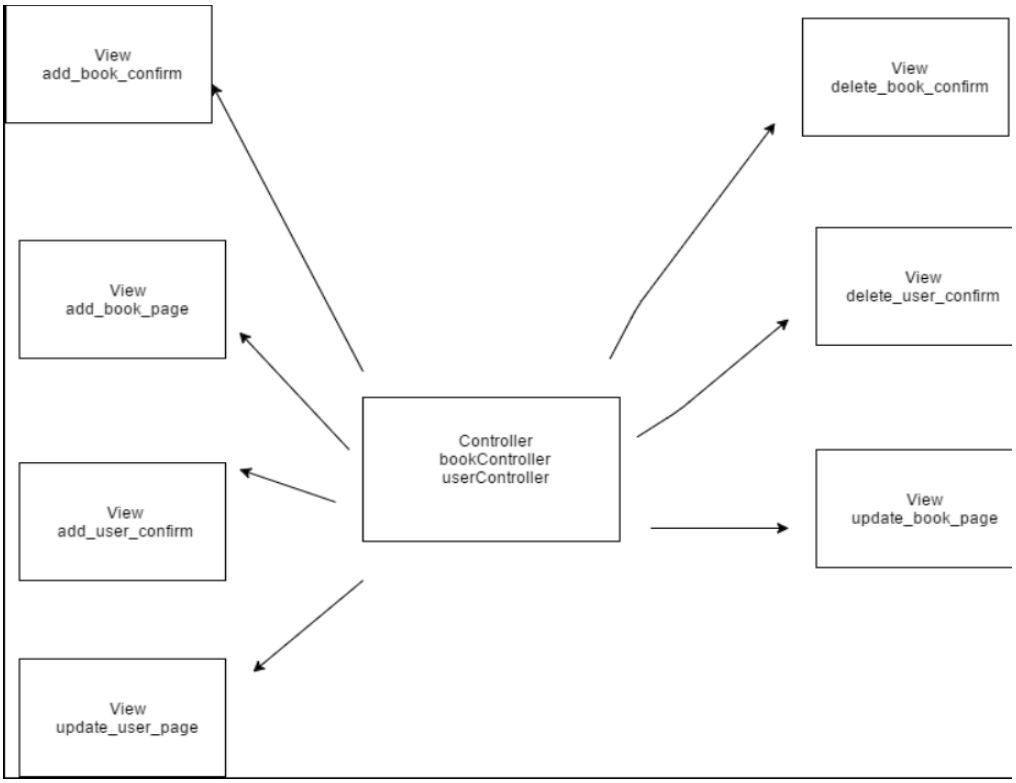
4. Design Details

4.1 Software Components

Here is the coarse component diagram of the system:

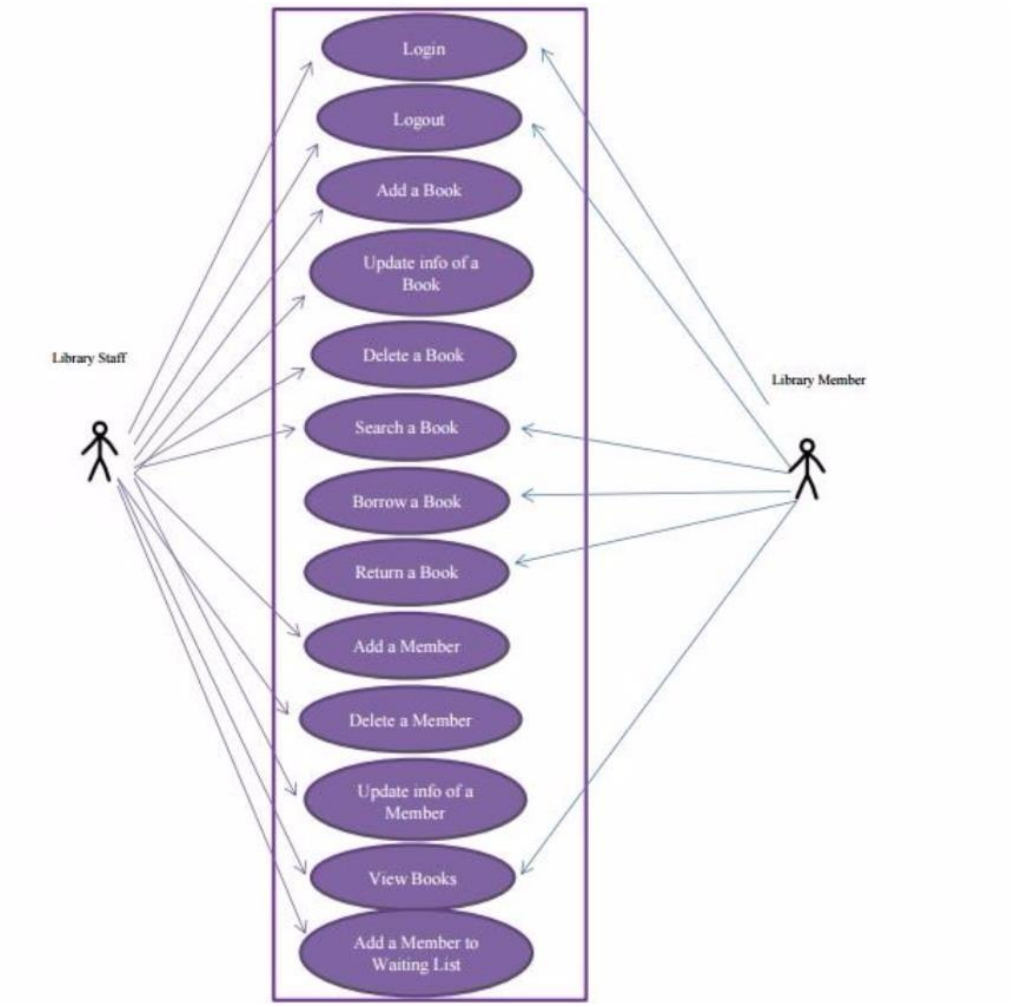


The coarse class diagram is as follows:



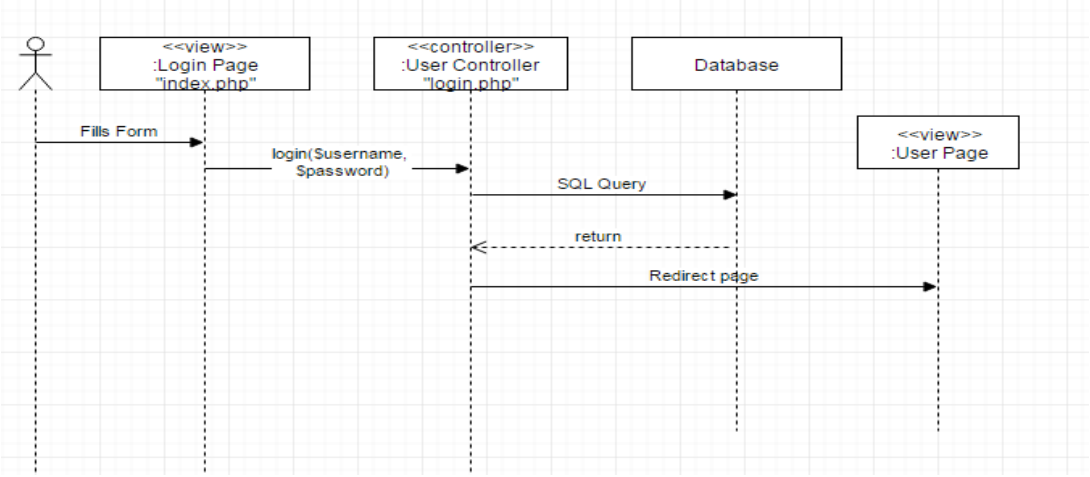
4.2 Software Behavior

The use case diagram we constructed and showed in System-Wide Requirements Specification document is as follows:

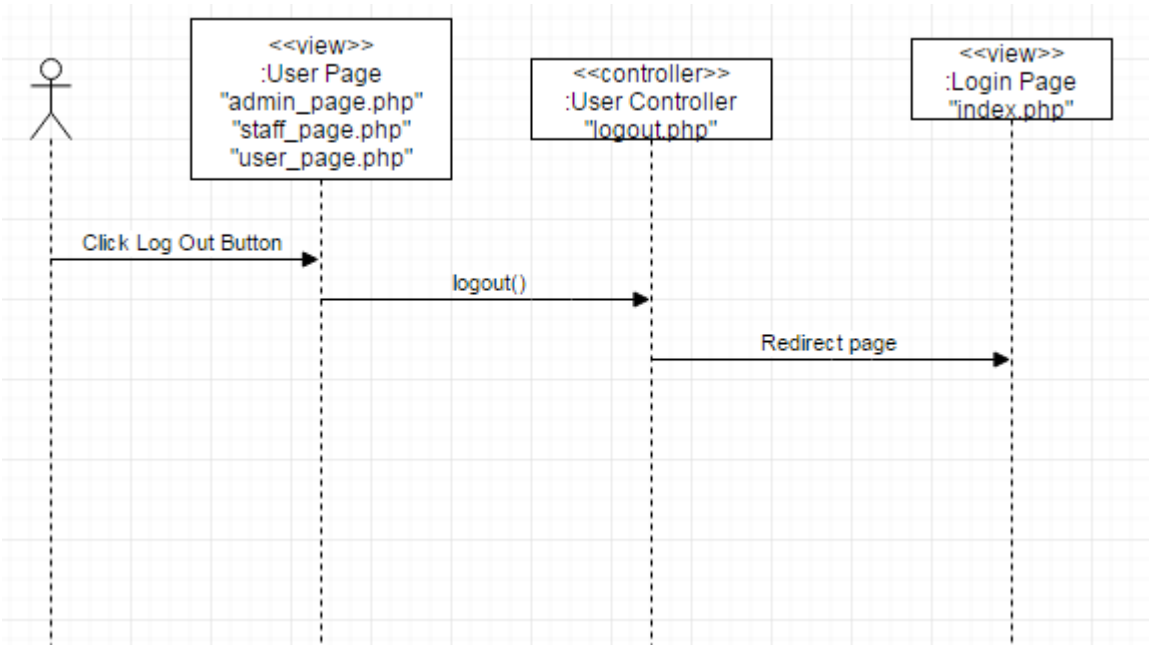


The sequential diagrams for the use cases we have already implemented are as follows:

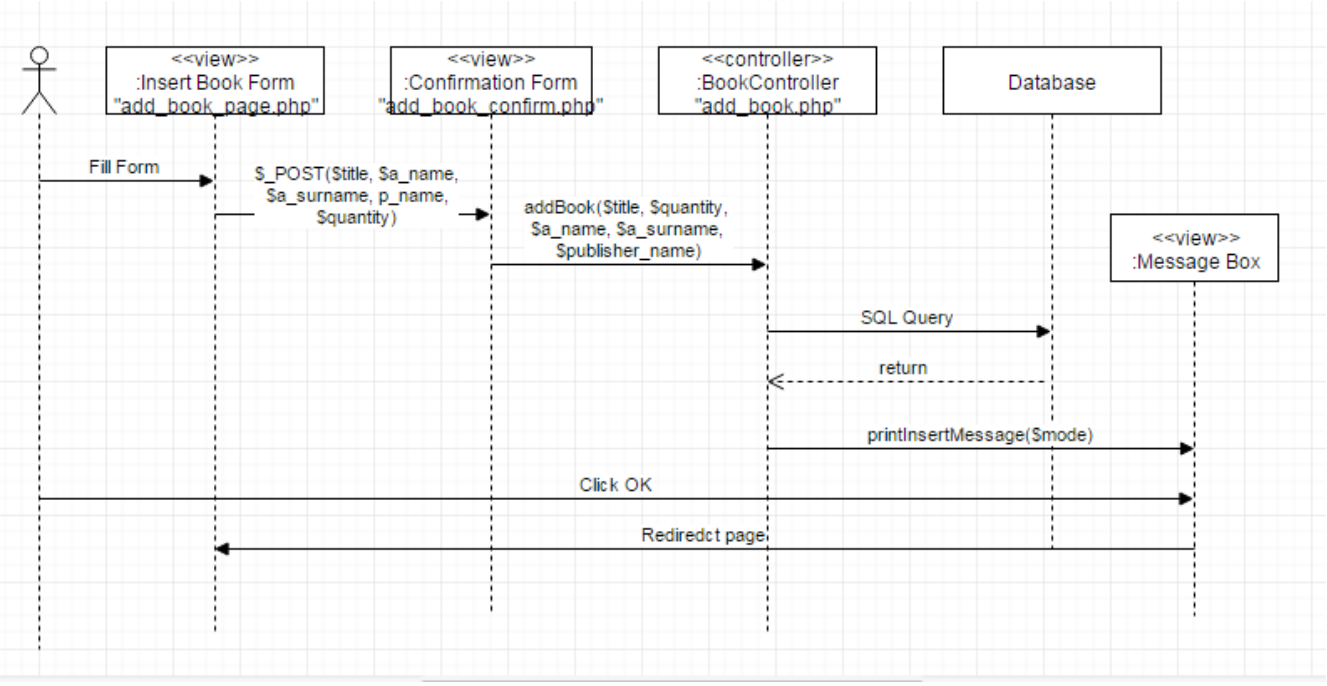
Sequential Diagram for Login is as follows:



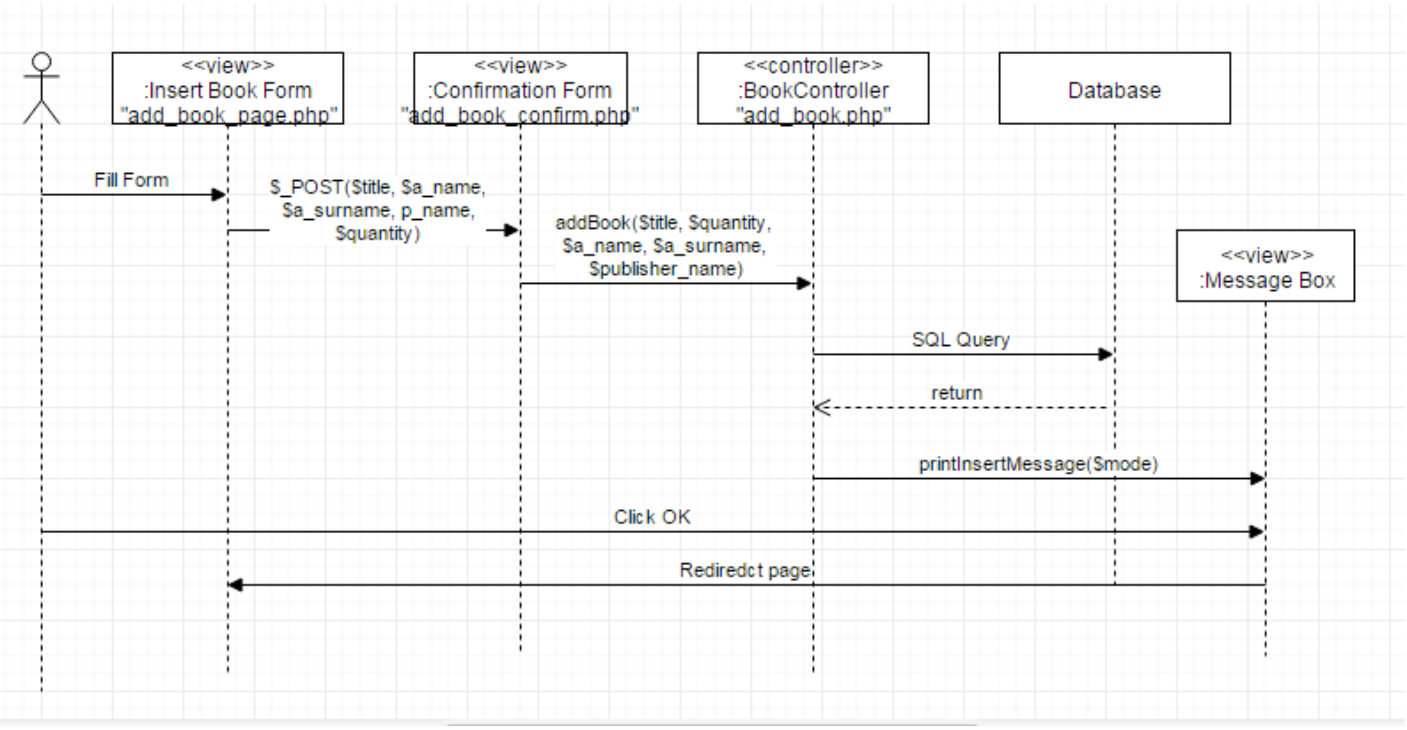
Sequential Diagram for Logout is as follows:



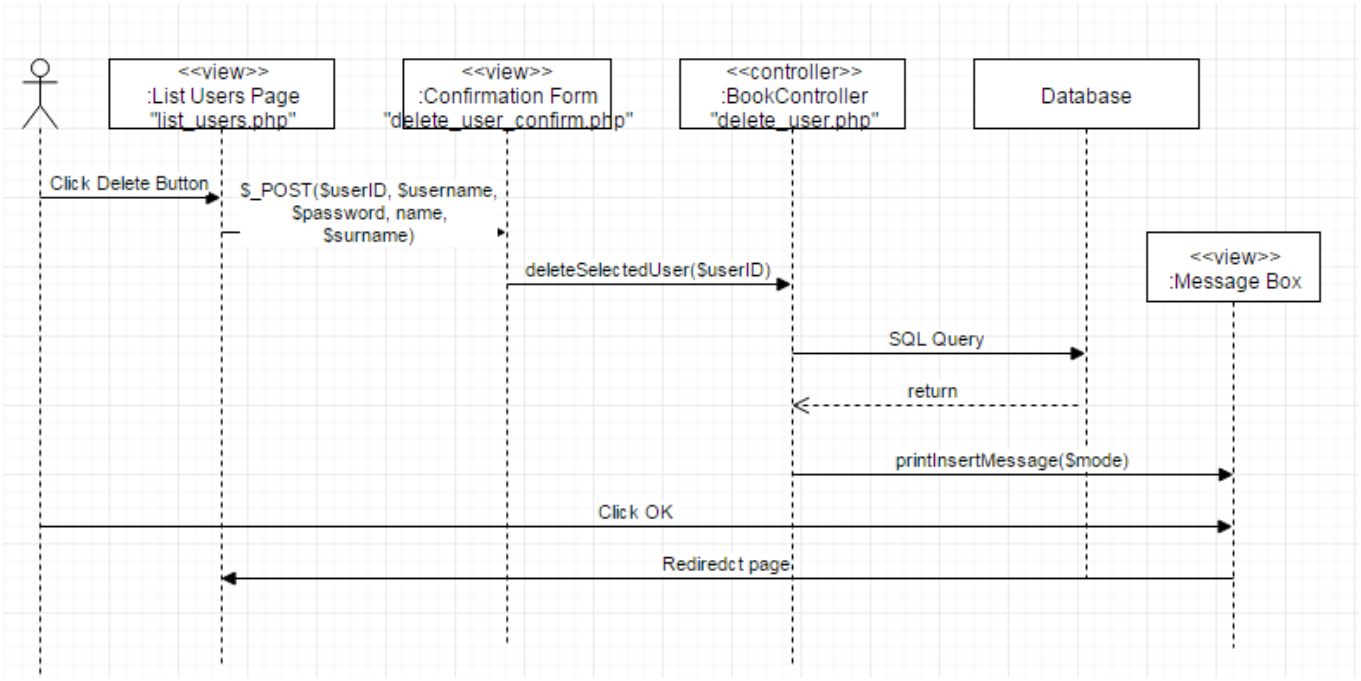
Sequential Diagram for Adding Book is as follows:



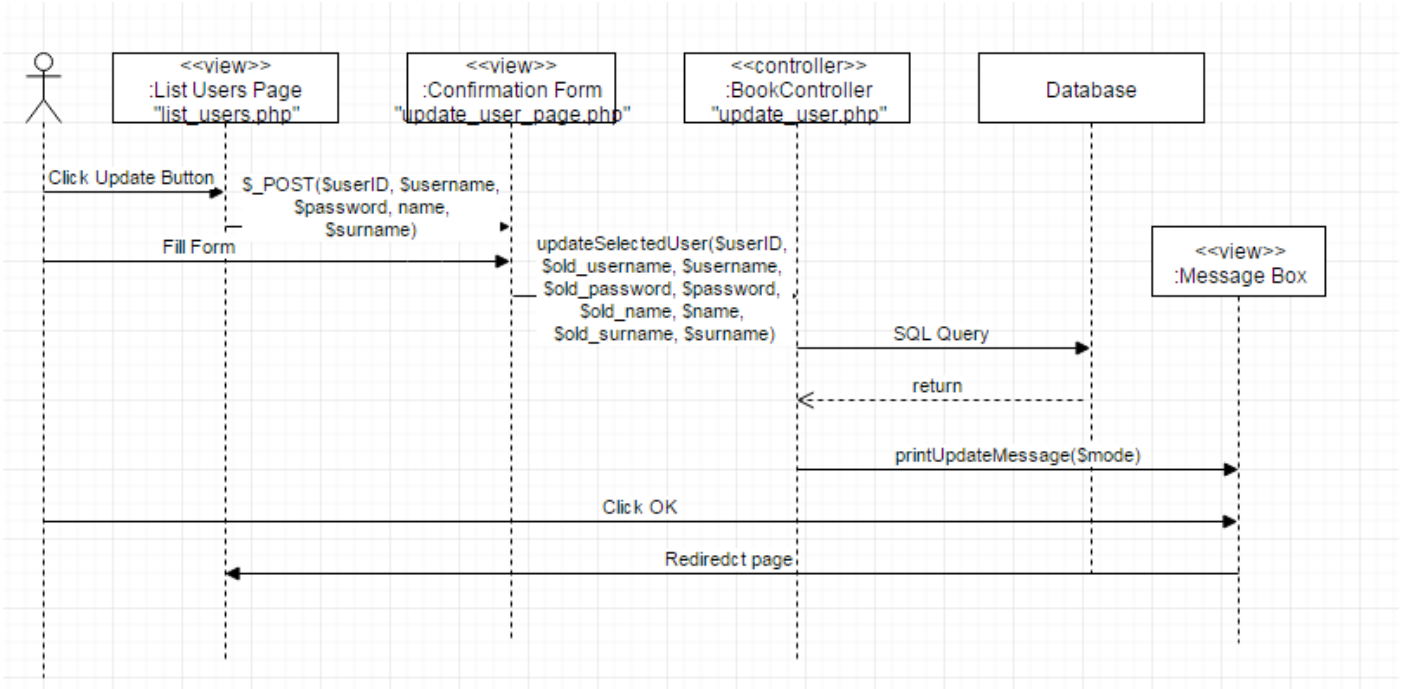
Sequential Diagram for Adding a User is as follows:



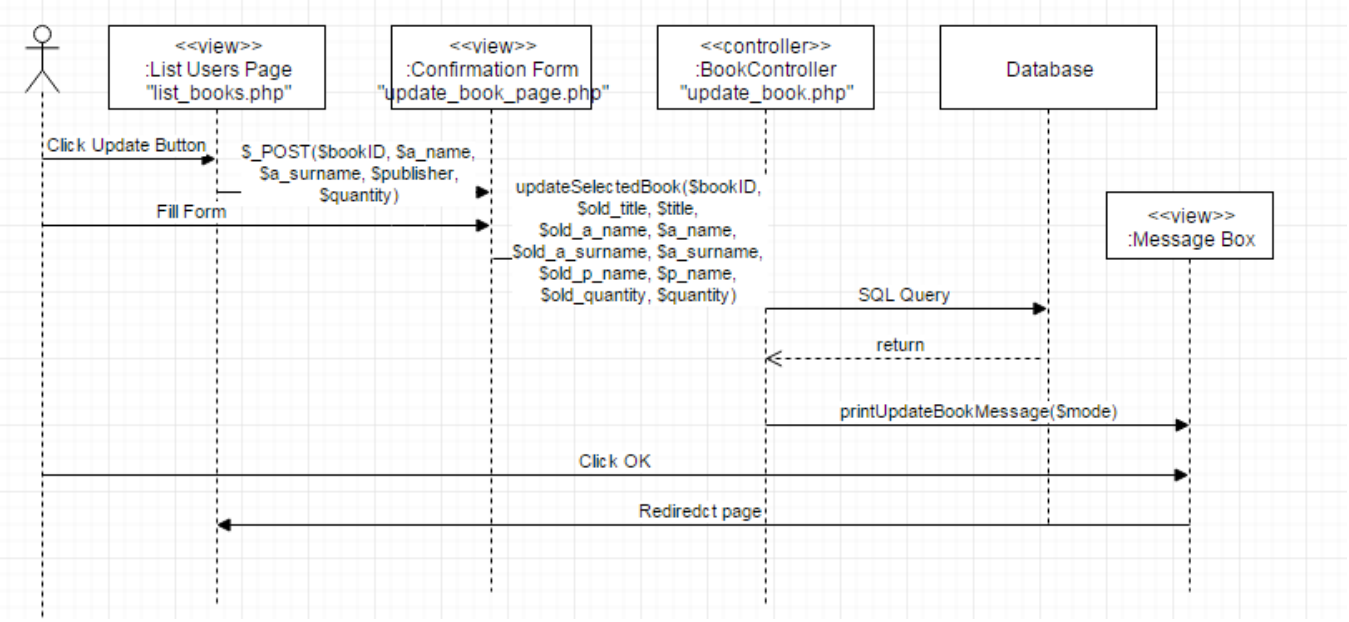
Sequential Diagram for Deleting a User is as follows:



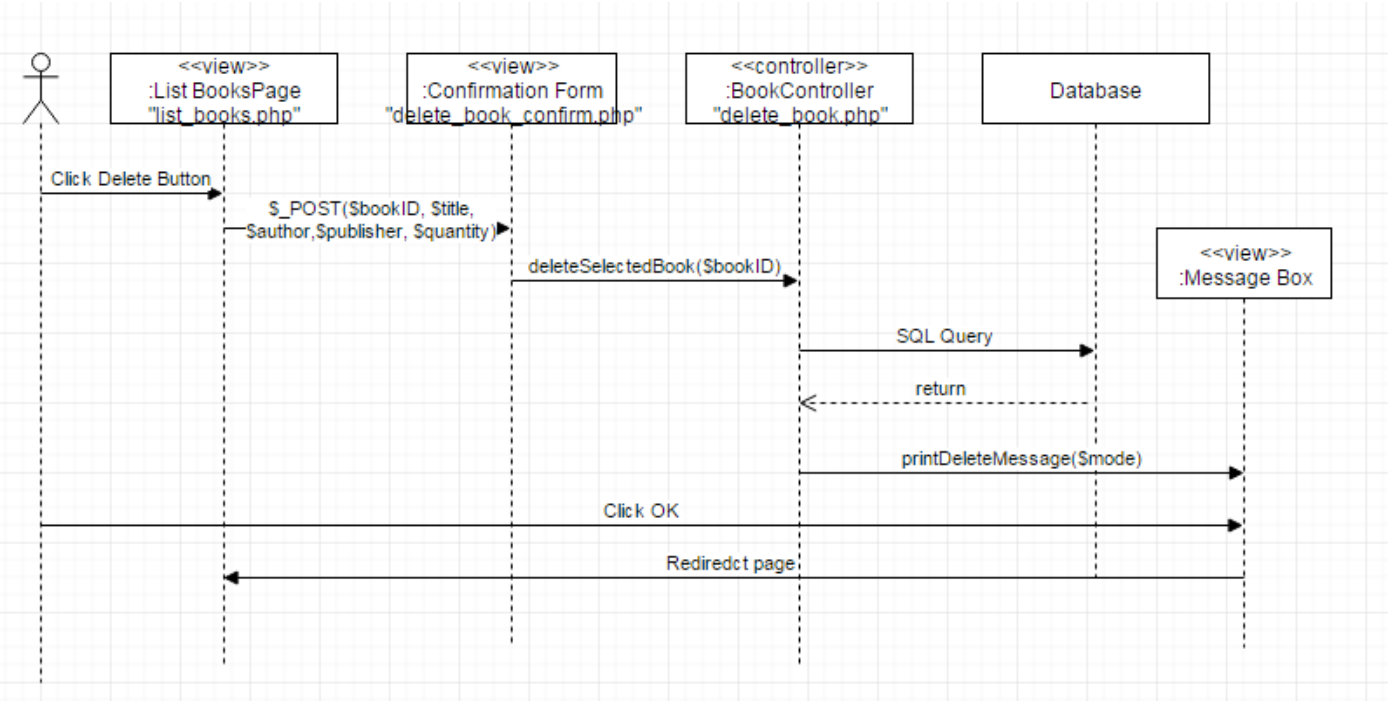
Sequential Diagram for Updating a User is as follows:



Sequential Diagram for Updating a Book is as follows:

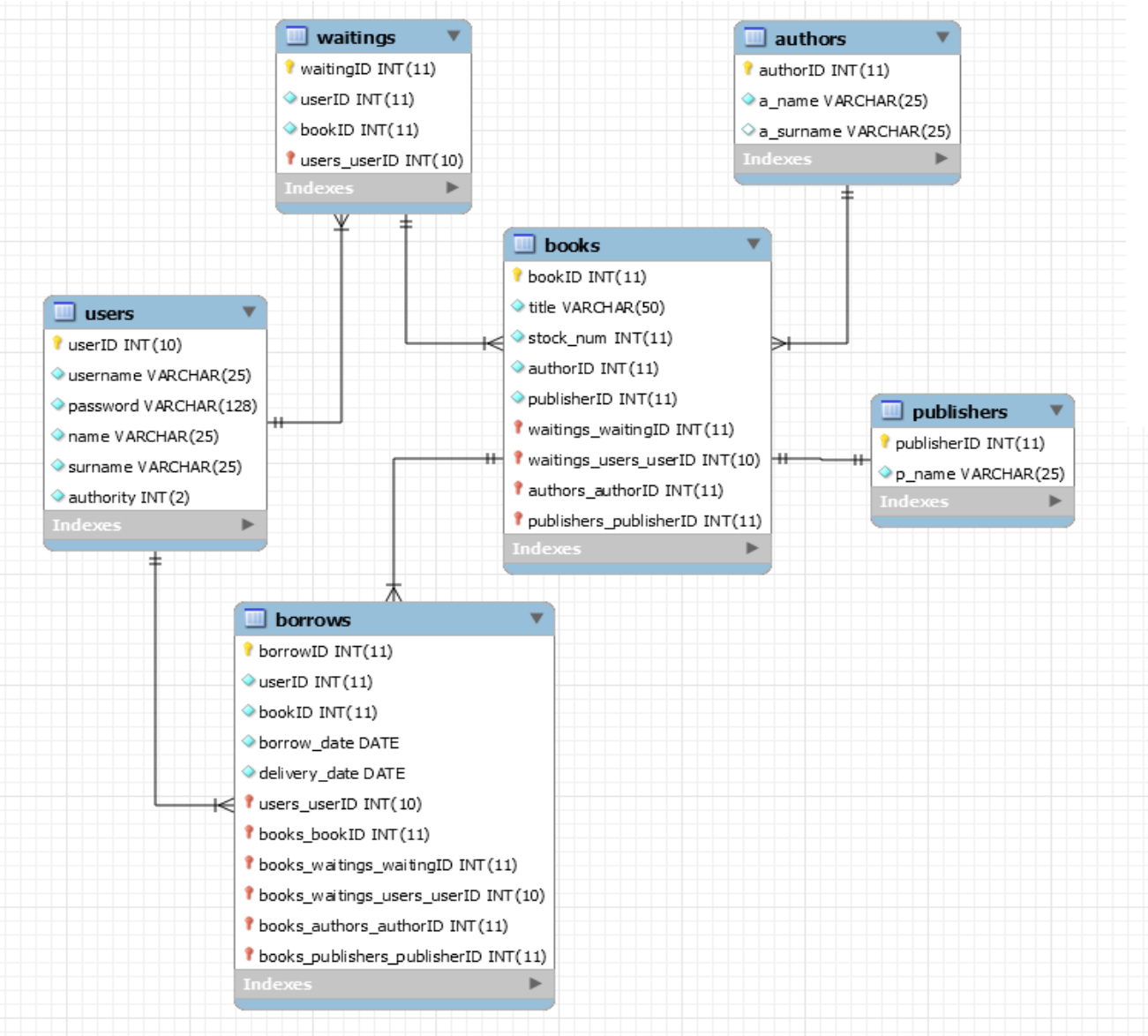


Sequential Diagram for Deleting a Book is as follows:



4.3 Data Model (E-R Diagram)

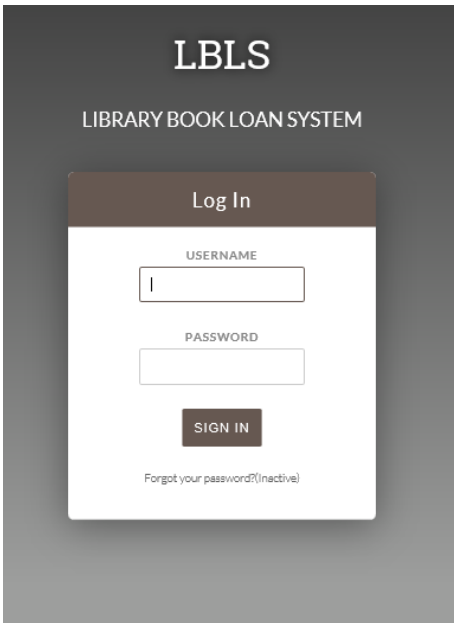
Our updated E-R Diagram that contains and shows our data models as well as the relationships between them is as follows:



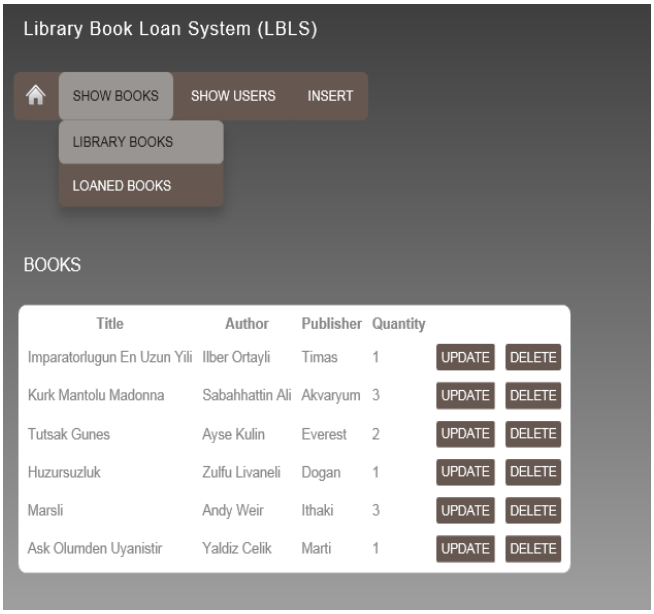
4.4 User Interface Design

We already showed planned design of our interface in software requirements document but we already mentioned that those were just the same in terms of the content. So by now, we have come up with the idea of making a new design about the interface. The following images show our system interface for various use cases.

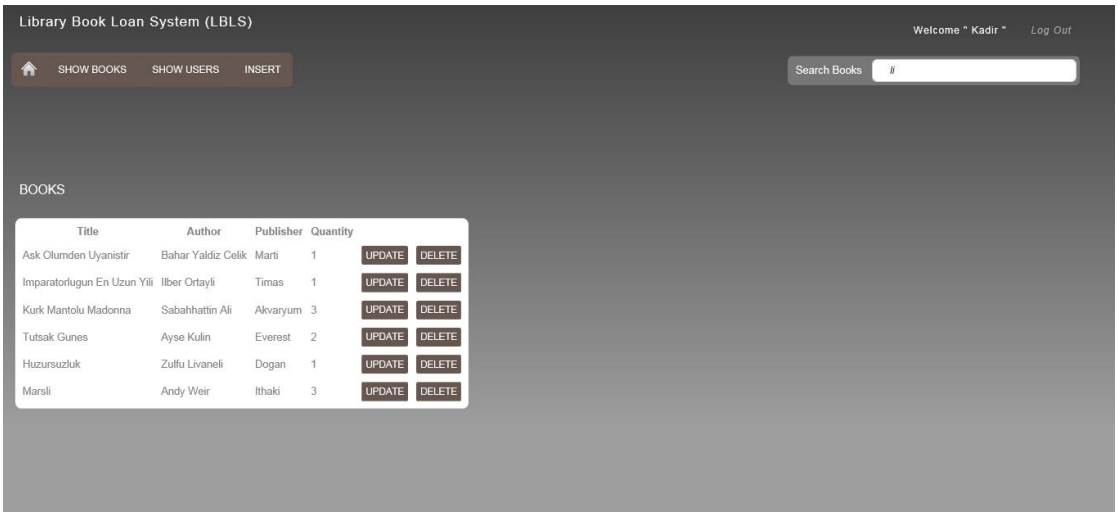
System Login Screen



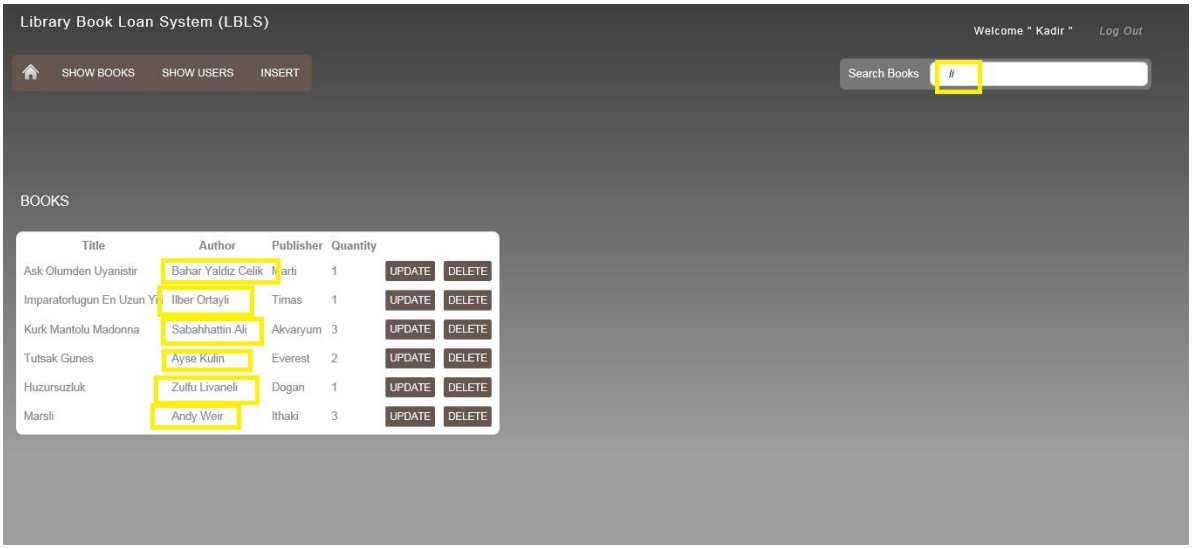
Admin views the list of books:



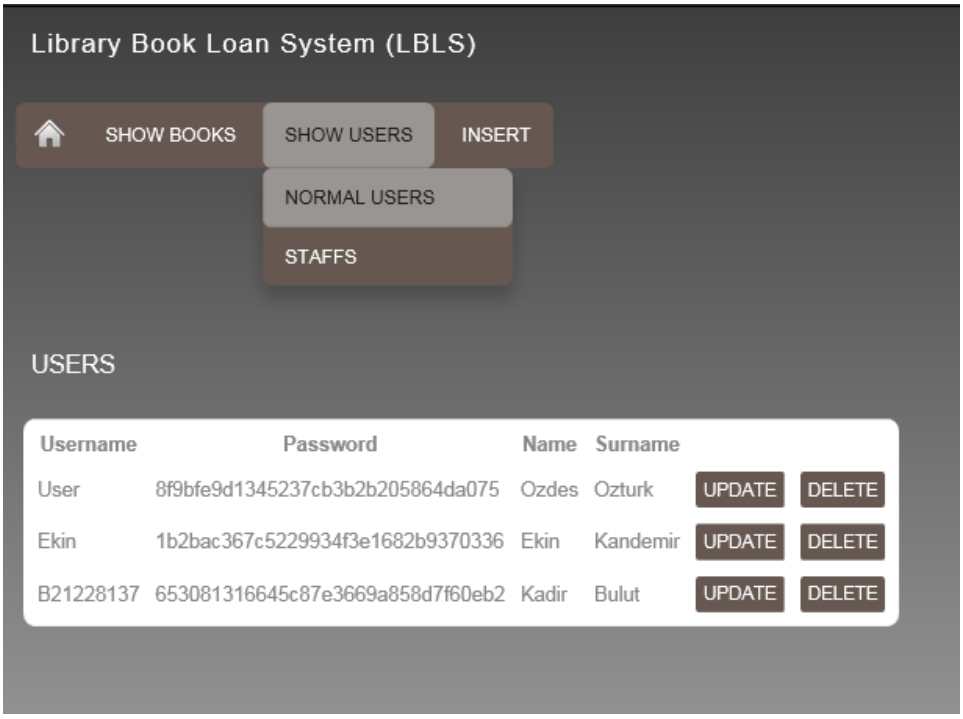
The search can be done, when the text is entered, the system will disable the results containing that text:



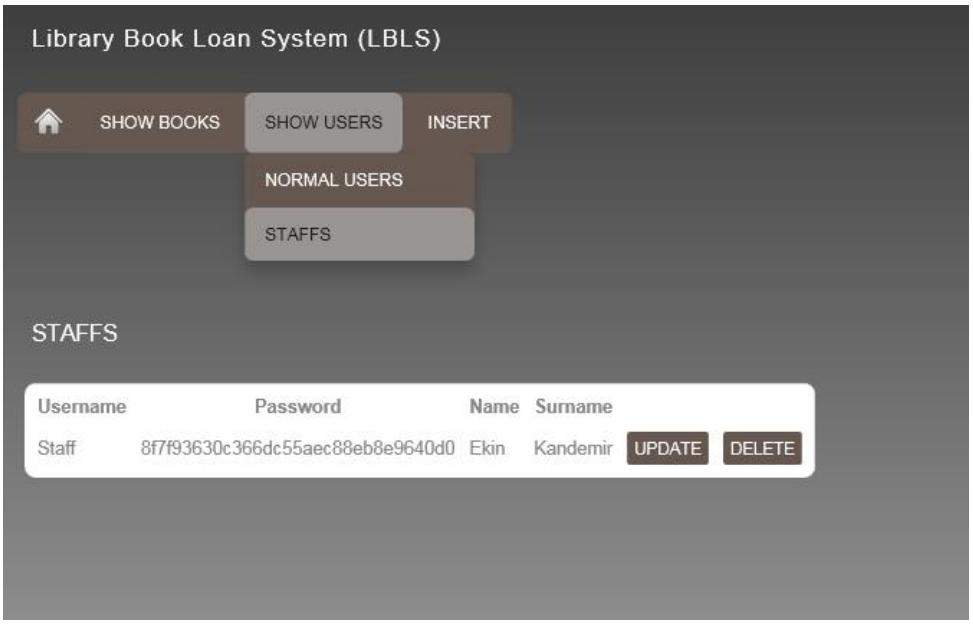
Then the system accurately displays these books containing “li”



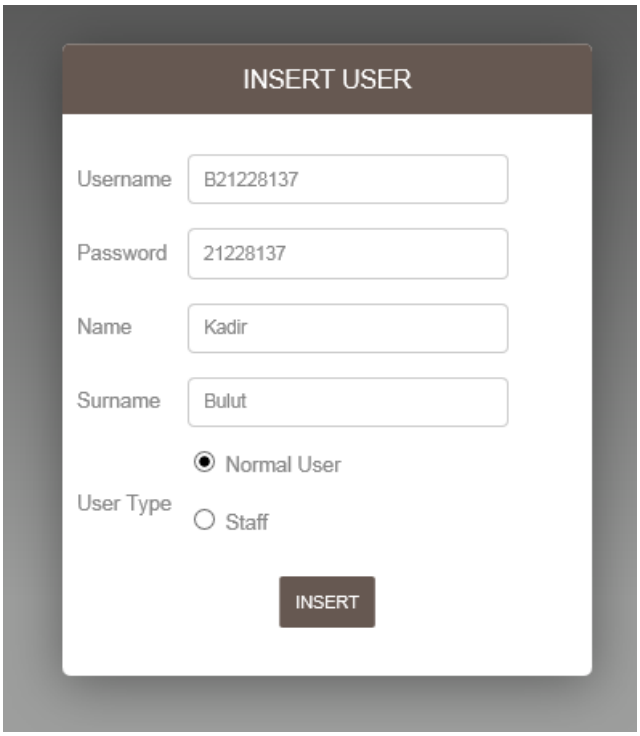
Admin can display the library members' list.



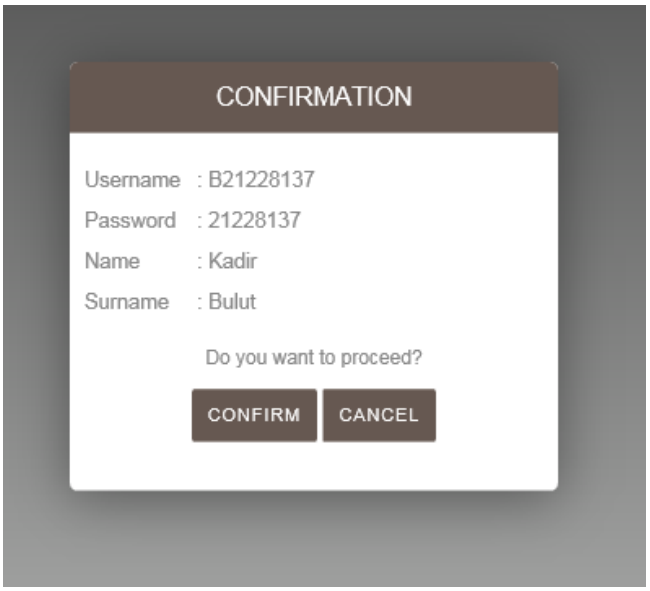
Admin can also view library staff's list.



A new member can be added to the system.

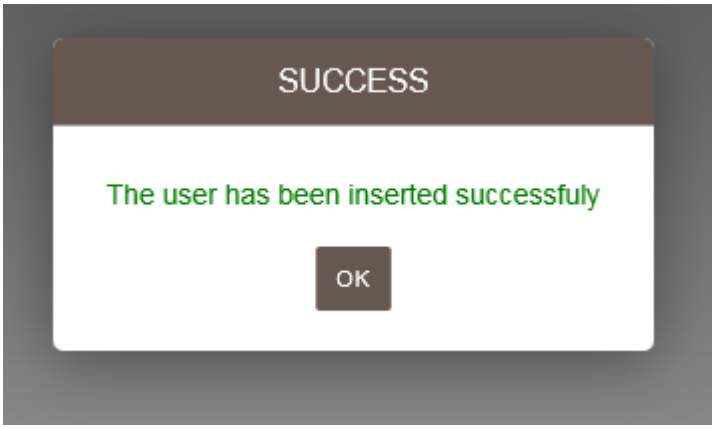


The system asks you for a confirmation by displaying the information that you have just entered and if you confirm, the system will also inform that you successfully inserted the user.

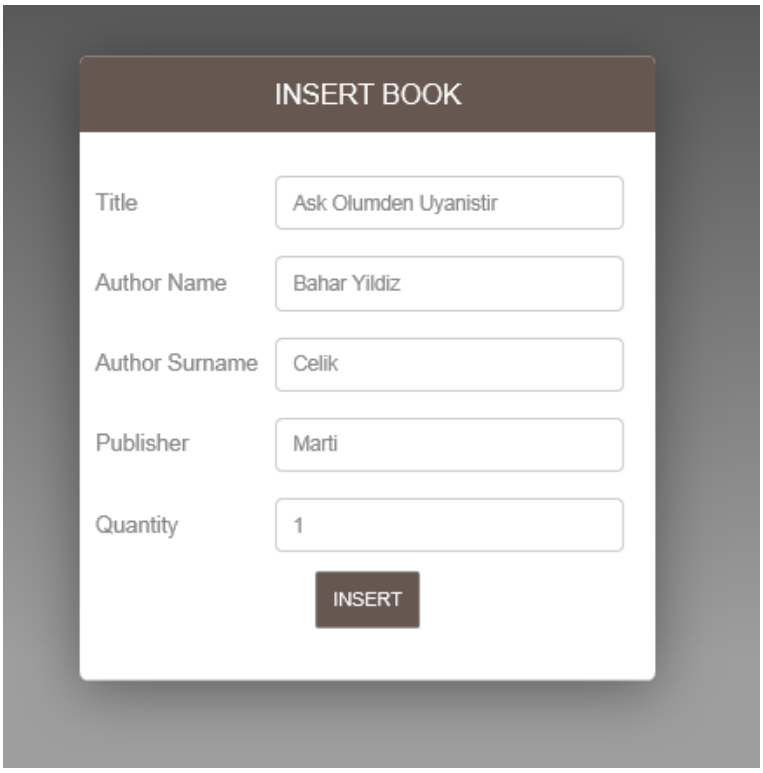


After a successful insertion and confirmation, the system displays a message informing the

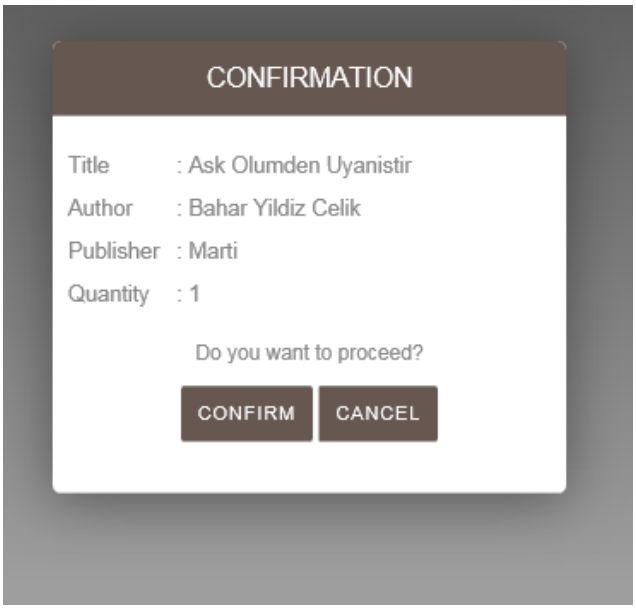
admin that the user has been successfully inserted.



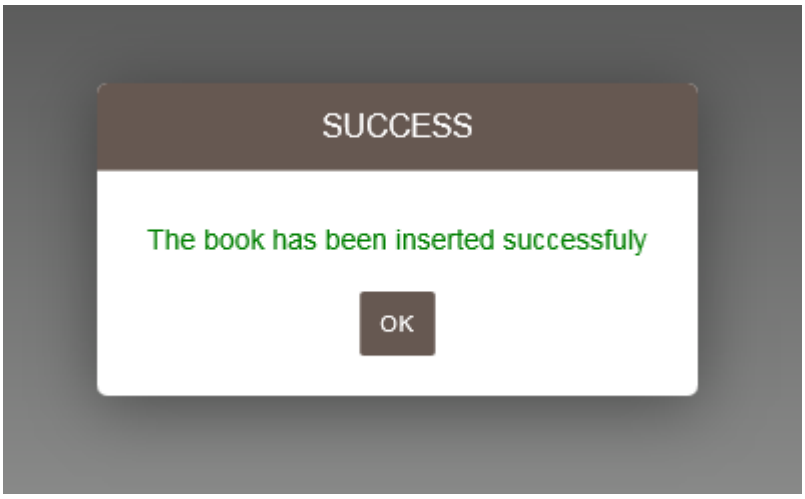
A new book can be added to the system.



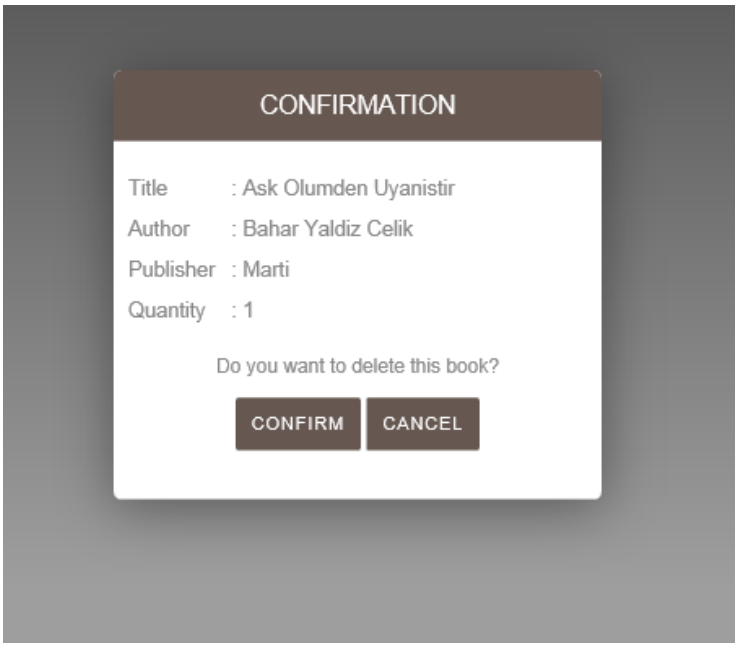
Again, the system asks you for a confirmation by displaying the information that you have just entered and if you confirm, the system will also inform that you successfully inserted the book.



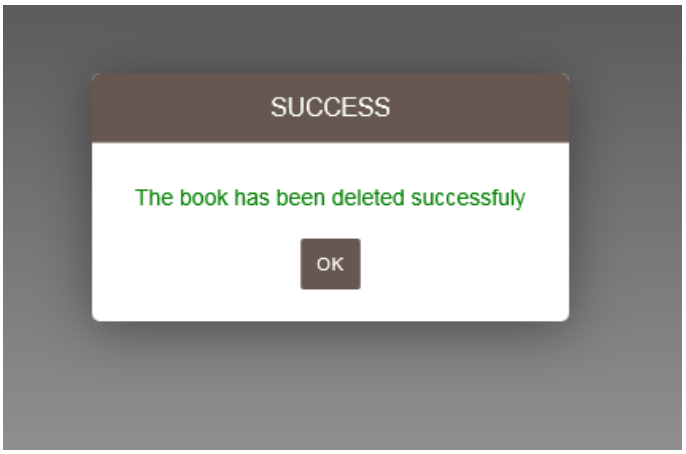
After a successful insertion and confirmation, the system displays a message informing the admin that the book has been successfully inserted.



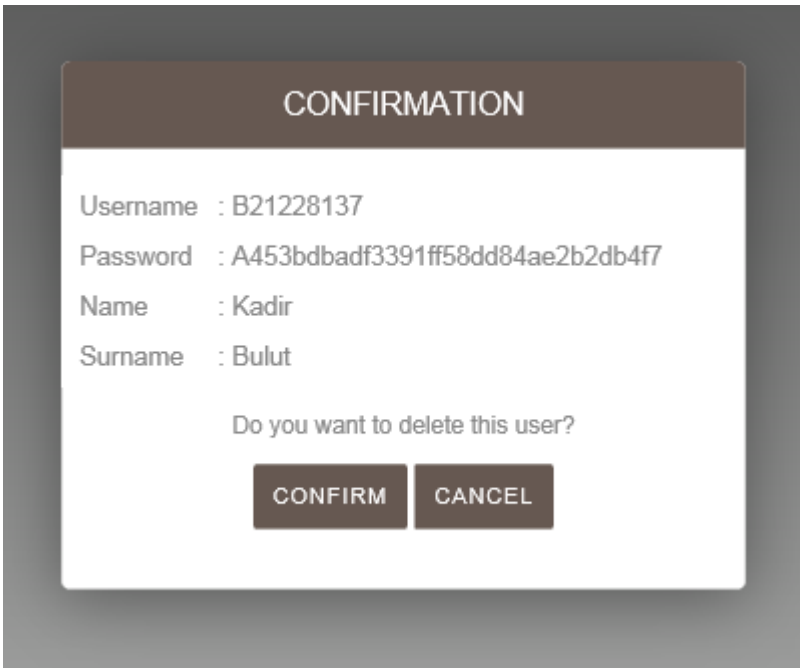
System also asks confirmation when there is an attempt to delete a book by also displaying the information of the book.



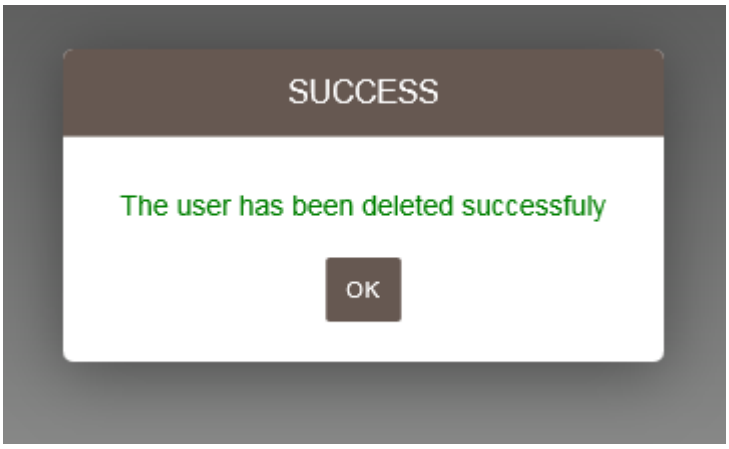
If the user confirms, the system displays a message confirming the user that the book is successfully deleted.



Admin can also delete a user when needed and system will ask for confirmation for this operation.



If the admin confirms, the system will again display a message informing the admin the user is successfully deleted.



5. Requirements Traceability

Until now, we could manage to solve the problems and implement a big part of the use cases that we claimed that we will implement in software system requirements document. As claimed, now the admin in the system has the capability of adding, updating and deleting a book as well as a user (it can be a library member or library employee), searching for a book, login and logout. We are quite satisfied with our performance right now since our system has 12 use cases and we

already implemented 9 of them. So we will keep up with the work and complete the remaining use cases as soon as possible so that we will be able to start our test process in order to build a 100% accurate project.