



Libra

Developed by:

Nour Awad 202-200-626
Leila Gamal 202-201-826

Use Cases

```
## Use Case 1: User Registration

### Actor:
User

### Description:
Allows a new user to register for an account in the system.

### Preconditions:

* The user is not already registered in the system.
* The system is accessible and functional.

### Main Flow:

1. The user opens the registration page.
2. The system prompts the user to enter details (name, email, password, and phone number).
3. The user submits the registration form.
4. The system validates the entered information.
5. If validation succeeds, the system creates a new user account.
6. The system confirms the registration with a success message.

### Alternative Scenarios:

1a. Invalid Inputs:

    1a.1. The system detects missing or invalid data (e.g., invalid email format).
    1a.2. The system displays an error message and asks the user to correct the input.
2a. Duplicate Email:

    2a.1. The email provided is already associated with an existing account.
    2a.2. The system informs the user and suggests logging in instead.

### Postconditions:

* A new user account is created and saved in the system.
* The user can now log in to the system.
```

Use Case 2: Add a Book

Actor:

Admin

Description:

Allows an admin to add a new book to the library catalog.

Preconditions:

- * The admin is logged into the system.
- * The admin has the necessary permissions to add books.

Main Flow:

1. The admin logs into the system.
2. The admin navigates to the "Add Book" section.
3. The system prompts the admin to provide book details (title, author, genre, and number of copies).
4. The admin submits the details.
5. The system validates the information and adds the book to the catalog.
6. The system confirms the addition with a success message.

Alternative Scenarios:

2a. Missing Details:

- 2a.1. If required fields are not filled, the system notifies the admin and highlights the missing fields.

3a. Duplicate Book:

- 3a.1. If the book already exists, the system updates the existing record instead of creating a new one.

Post-conditions:

- * The new book is added to the library catalog and is available for borrowing.

Use Case 3: Borrow a Book

```
### Actor:
User

### Description:
Allows a member to borrow a book from the library.

### Preconditions:

* The user is logged into the system.
* The desired book exists in the library catalog.

### Main Flow:
1. The user logs into the system.
2. The user searches for the desired book.
3. The system displays the book details, including availability.
4. The user selects "Borrow Book."
5. The system validates if the user is eligible to borrow (e.g., no overdue books, below borrowing limit).
6. If eligible, the system marks the book as borrowed and updates the user's record.
7. The system displays a success message and the due date.

### Alternative Scenarios:

2a. Book Not Found:

    2a.1. The system informs the user that the book is not available.
5a. Book Unavailable:

    5a.1. The system informs the user that all copies are currently borrowed.
### Post-conditions:

* The book is marked as borrowed and associated with the user's record.
* The book's availability is updated in the catalog.
```

```
## Use Case 4: Return a Book
```

```
### Actor:
User

### Description:
Allows a member to return a borrowed book.

### Preconditions:

* The user is logged into the system.
* The book to be returned exists in the user's borrowing history.

### Main Flow:
1. The user logs into the system.
2. The user navigates to their borrowing history.
3. The system displays a list of borrowed books.
4. The user selects the book to return.
5. The system marks the book as returned and updates its availability in the catalog.
6. The system displays a success message.

### Alternative Scenarios:

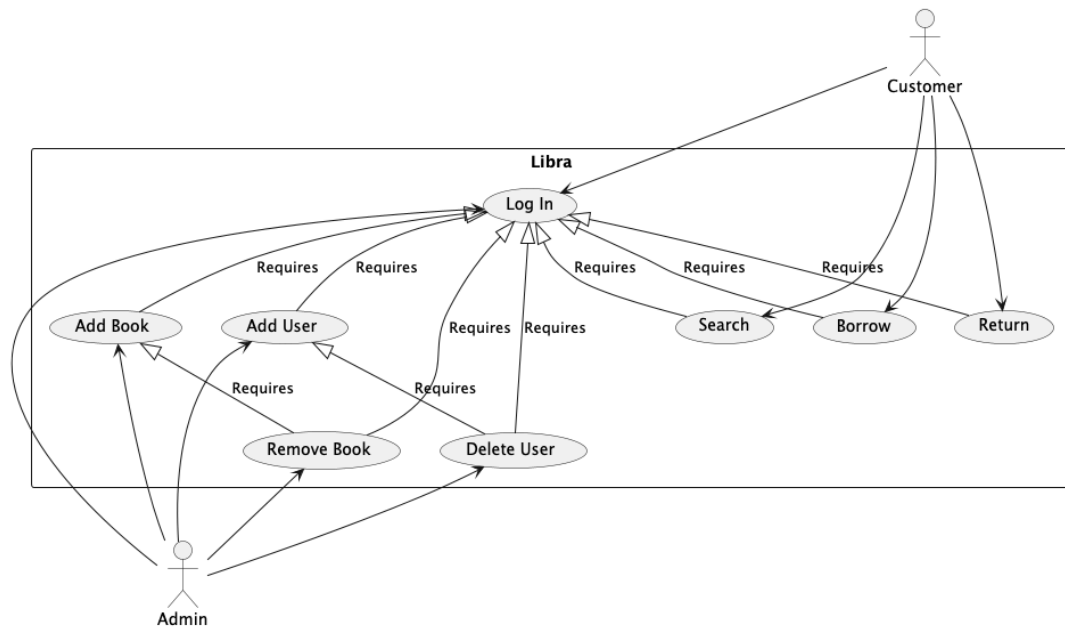
4a. Book Not Borrowed:

    4a.1. If the book is not in the user's borrowing
           history, the system displays an error.

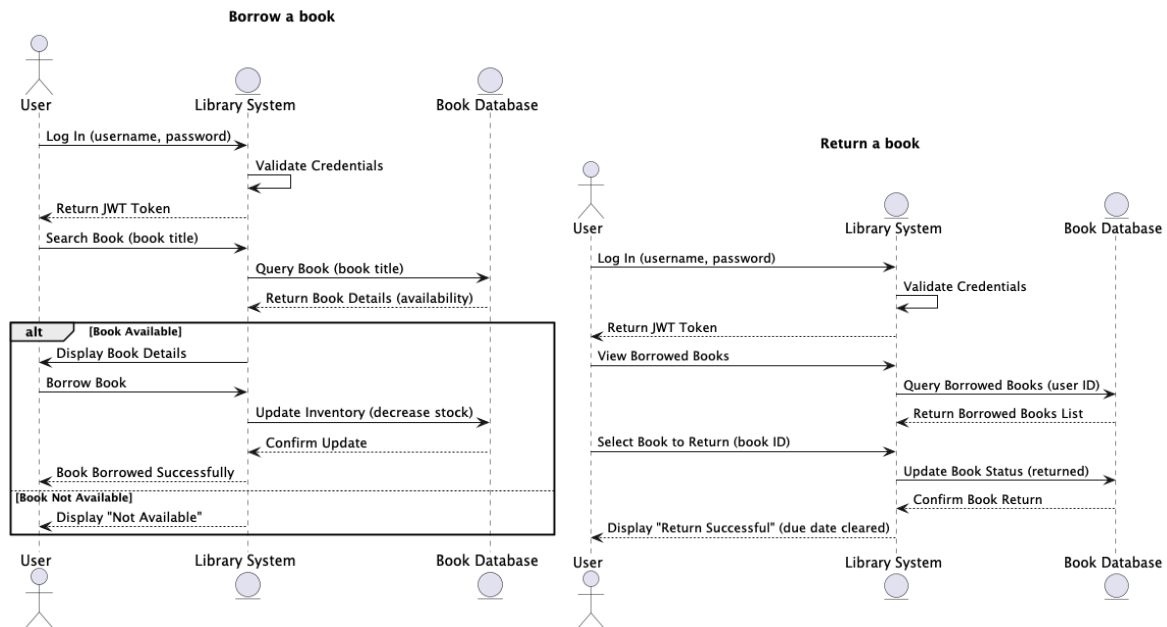
### Post-conditions:

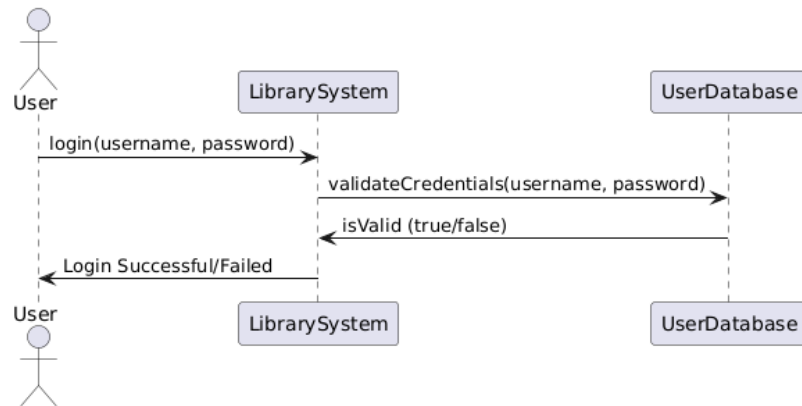
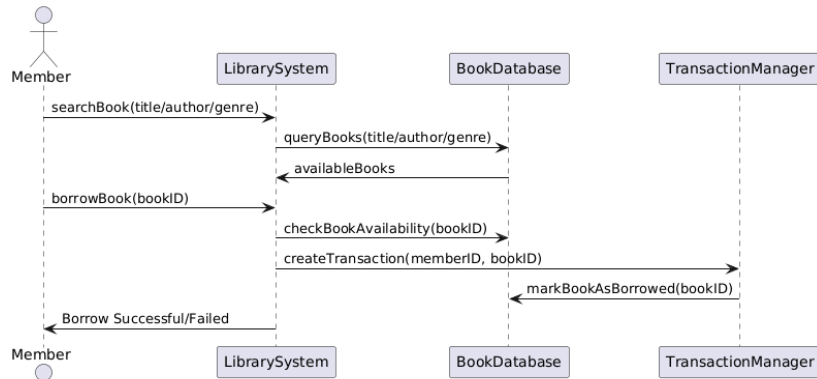
* The book is marked as returned in the system.
* The book's availability is updated in the catalog.
```

Use Case Diagram

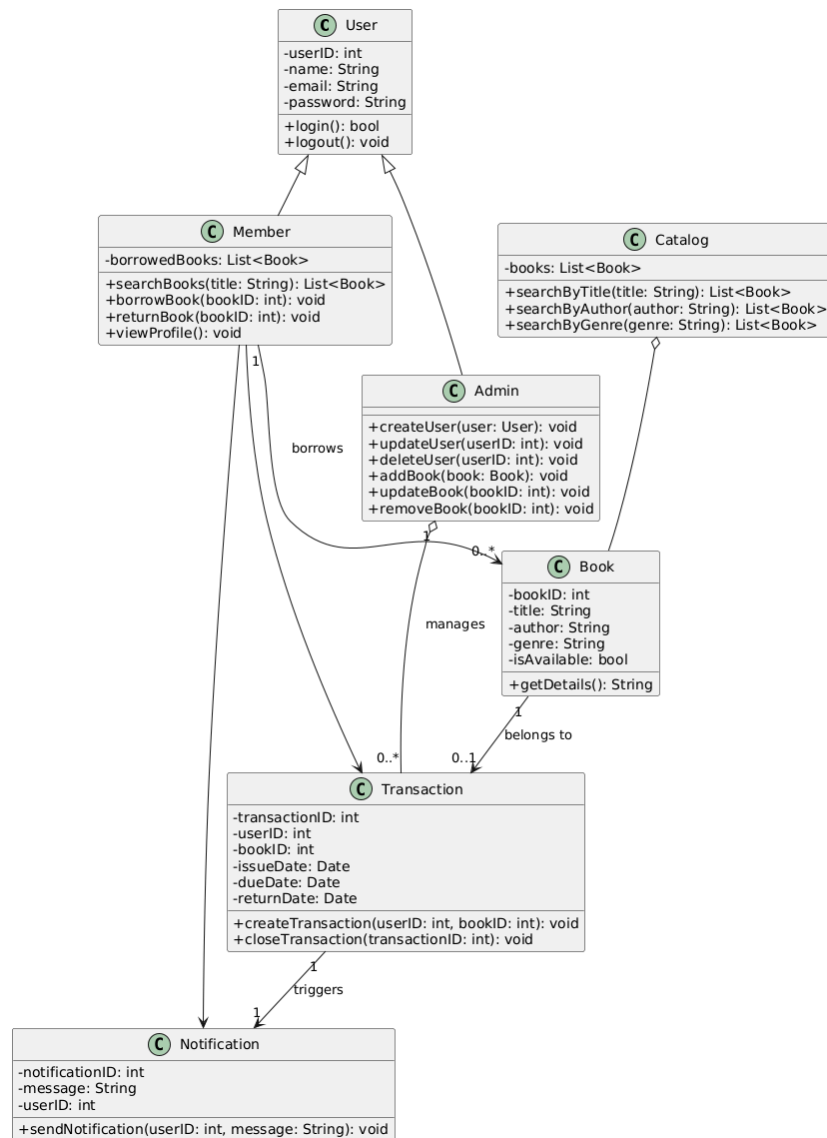


Sequence Diagram



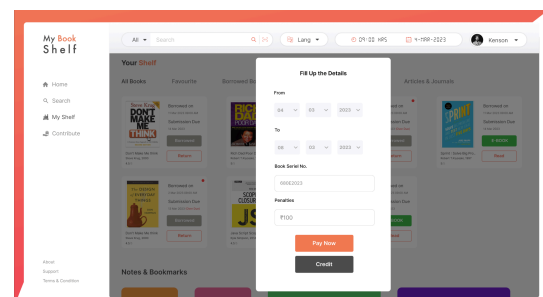


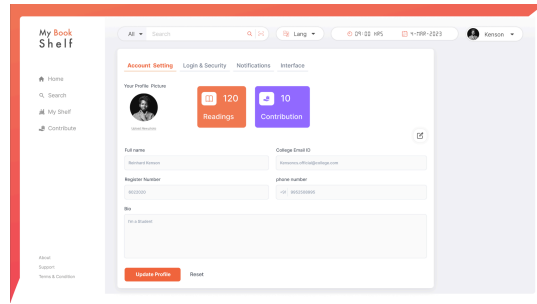
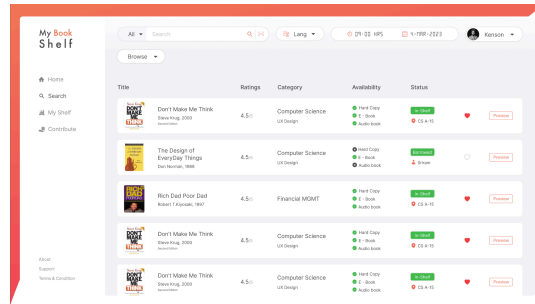
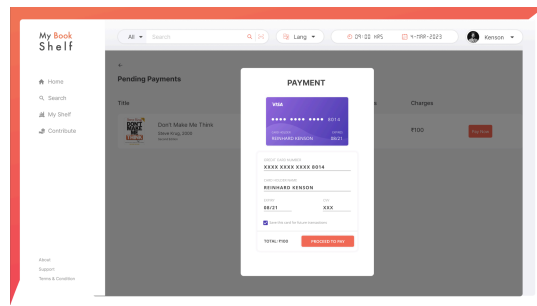
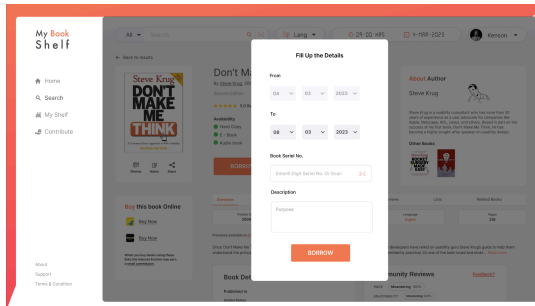
Class Diagram



Summary:

- Liela:
 - Sequence Diagram
 - Class Diagram
 - OCL
 - Use Case Diagram
 - Design Document
 - GUI Mockups:





Lessons Learnt:

- How to integrate between Thymeleaf and Spring Boot.
- Spring Boot setup.
- Database connection