



**CSE 3105 / CSE 3137**

**OBJECT-ORIENTED ANALYSIS AND DESIGN**

**FALL 2024**

**COURSE PROJECT: Library Management System (LMS)**

***Requirements Analysis Document***

***Group 5***

*Berke Alpaslan – 210316016*

*Mert Doğan Aygün - 210316076*

*Gül Yeşilgil - 220316055*

*Birhat Taş - 190315010*

*Tural Mammadov - 190316001*

*Aziz Berk Yıldırım - 210316055*

*29 October 2024*

## Table of Contents

1	Introduction .....	1
2	Current System .....	1
3	Proposed System .....	1
3.1	Functional Requirements.....	2
3.2	Nonfunctional Requirements.....	2
3.3	System Models.....	3
3.3.1	Scenarios .....	3
3.3.2	Use Case Model.....	5
3.3.3	Object Model .....	6
3.3.4	Dynamic Models.....	8
3.3.5	User Interface Mock-ups.....	11
4	Glossary.....	13
5	Appendix .....	14

## Table of Figures

Figure 1 Main Scenario: Book Borrowing and Returning Process Use-Case Model.....	5
Figure 2 Side Scenario 1: Book Reservation Process Use-Case Model .....	5
Figure 3 Side Scenario 2: Library Inventory and Stock Management Use-Case Model.....	5
Figure 4 Main Scenario: Book Borrowing and Returning Process Object Model .....	6
Figure 5 Side Scenario 1: Book Reservation Process Object Model.....	6
Figure 6 Side Scenario 2: Library Inventory and Stock Management Object Model .....	7
Figure 7 Main Scenario: Book Borrowing and Returning Process Dynamic Model .....	8
Figure 8 Side Scenario1: Book Reservation Process Dynamic Model.....	9
Figure 9 Side Scenario 2: Library Inventory and Stock Management Dynamic Model .....	10
Figure 10 User Login Mock-Up .....	11
Figure 11 User's Main Menu Mock-Up.....	11
Figure 12 Admin's Main Menu Mock-Up .....	12

# **1 Introduction**

The Library Management System (LMS) is designed to streamline and modernize the university library's operations, providing students and staff with an efficient way to manage library resources. This system aims to address the growing demand for a digital solution that enables users to borrow, return, and search for books with ease, while also supporting library administrators in tracking inventory and managing user accounts within a user-friendly, automated environment.

The LMS will incorporate essential functions such as book borrowing and return management, book search and reservation capabilities, real-time inventory tracking, and user notifications for overdue items. Additionally, the system will offer administrators tools for managing the collection, generating reports, and overseeing user activities. Through these features, LMS is expected to significantly enhance accuracy, efficiency, and overall user satisfaction within the library.

Aligned with the university's objective to improve digital services and make library resources more accessible, the LMS project will draw insights from similar systems in other academic institutions that have demonstrated notable improvements in operational efficiency and user satisfaction. By adapting these best practices to our university's specific needs, the LMS aims to provide a tailored and effective solution.

The primary objective of the LMS is to deliver an intuitive, reliable, and user-friendly system that elevates the library experience for students, faculty, and staff. The success of this project will be measured by its ability to handle book checkouts and returns efficiently, achieve high user adoption and satisfaction rates, reduce the time library staff spend managing inventory and accounts, and increase the timely return of books through regular, automated notifications.

## **2 Current System**

The university library currently operates on a manual system where book borrowing, returns, and inventory tracking depend on physical records. This process is time-consuming and prone to errors, with no automated notifications for overdue books, resulting in delays. The new Library Management System (LMS) will digitize these tasks, enhancing efficiency, accuracy, and accessibility for all users.

## **3 Proposed System**

The proposed system aims to replace the current manual and inefficient processes with a fully automated Library Management System. This system will allow users to search, borrow, return, and reserve books digitally. Key functionalities include real-time updates on book availability, automatic tracking of due dates, and notifications for reservations. The system will streamline library operations by reducing errors and processing times, improving the user experience, and enabling better management of library resources. The following subsections will detail the system's requirements and analysis model.

### 3.1 Functional Requirements

#### a. Book Status Management

- The system should allow library admins to check the current status of books (borrowed, reserved, available).
- It should provide an interface for updating book information (title, author, publisher, year, etc.).

#### b. Inventory Addition and Update

- When adding a new book to inventory, a form should be provided to input details such as title, author, publisher, and publication year.
- Newly added books should be stored in the system's database and automatically listed in the inventory.

#### c. Low Stock Alert

- The system should automatically notify library admins if the stock of a particular book falls below a set threshold.
- Alerts should be visible in the library admin interface, along with a list of low-stock items.

#### d. Inventory Query and Report Generation

- Library admins should be able to generate reports on inventory status and book availability.
- The system should allow searches within the inventory based on specific criteria such as author or category.

### 3.2 Nonfunctional Requirements

#### a. System Performance

- Book status queries should complete within 2 seconds, regardless of the number of books in the inventory.
- The system should immediately reflect inventory updates across all interfaces.

#### b. Usability

- The interface for library admins should be straightforward, allowing new users to adapt quickly.
- Inventory control and alert screens should be intuitive and easily accessible.

#### c. Scalability

- The system should be scalable to handle increased inventory sizes while maintaining performance.
- If additional libraries are added, each library's inventory should be manageable independently.

#### d. Security

- User accounts should be limited by permission levels, allowing only authorized staff to access inventory management.
- Users without library admin permissions should be restricted from adding or modifying book entries.

### 3.3 System Models

#### 3.3.1 Scenarios

##### a. Main Scenario: Book Borrowing and Returning Process

###### **Starting State:**

A library member accesses the system to borrow a book or to return a previously borrowed book.

###### **Steps:**

1. The member searches the library catalog to find an available book by title, author, or genre.
2. The system displays book details, availability status, and location within the library.
3. Upon selecting an available book, the member initiates a borrowing request.
4. The system validates the request and updates the book's status to "borrowed," linking it to the member's account.
5. The system sets a return date and sends a confirmation to the member with due date information.
6. For returns, the member logs into their account, selects the "return" option, and confirms the action.
7. The system updates the book's status to "available" and removes it from the member's active loans.

###### **Ending State:**

The member has successfully borrowed or returned the book, and the system updates the book's status accordingly.

##### b. Side Scenario 1: Book Reservation Process

###### **Starting State:**

A member wishes to reserve a book that is currently borrowed by another user.

###### **Steps:**

1. The member searches for the desired book and sees that it is currently unavailable.
2. The member selects the reservation option, initiating a request to reserve the book.
3. The system confirms the reservation and notifies the member that they will be alerted when the book is available.
4. Once the book is returned, the system sends an availability notification to the member.
5. The member has a limited time to borrow the reserved book before it becomes available to others.

###### **Ending State:**

The reserved book is held for the member when available, or the reservation is canceled if not borrowed in the allocated time.

**c. Side Scenario 2: Library Inventory and Stock Management**

**Starting State:**

A librarian (library admin) logs into the system to check the inventory or to add a new book.

**Steps:**

1. The librarian reviews the status of current books (borrowed, reserved, available).
2. If a new book needs to be added, the librarian (library admin) enters details (title, author, publisher, year) and saves the entry.
3. The system adds the book to the inventory, making it searchable and accessible for members.
4. The system sends automated alerts if any book's stock level is low, prompting restocking or acquisition of additional copies.

**Ending State:**

The inventory is updated with new books, and low-stock notifications help maintain availability for popular items.

### 3.3.2 Use Case Model

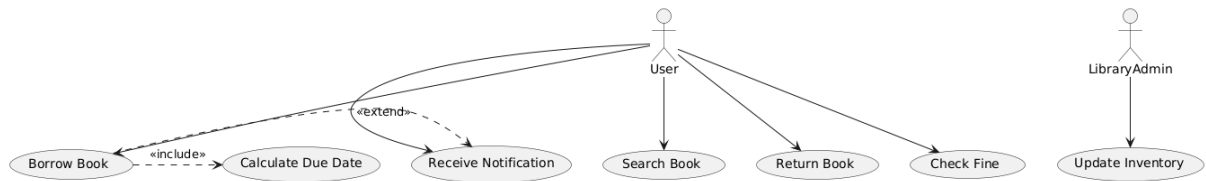


Figure 1 Main Scenario: Book Borrowing and Returning Process Use-Case Model

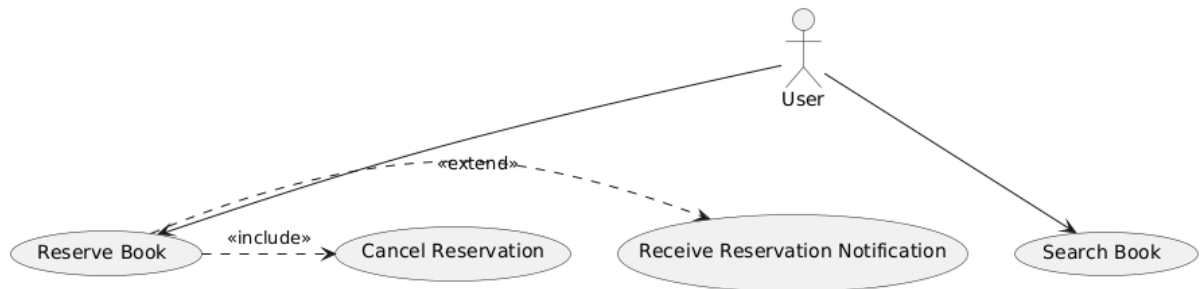


Figure 2 Side Scenario 1: Book Reservation Process Use-Case Model

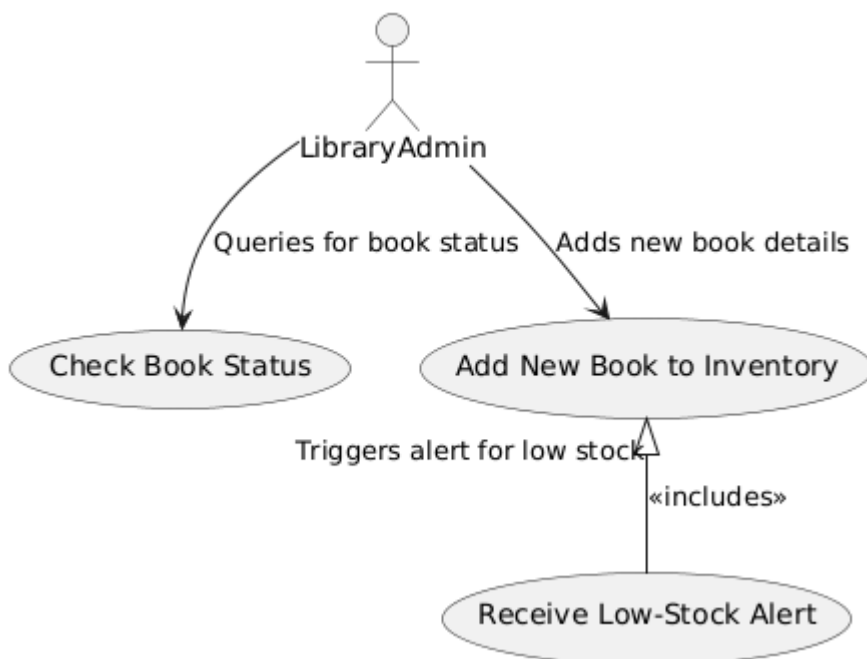


Figure 3 Side Scenario 2: Library Inventory and Stock Management Use-Case Model



### 3.3.3 Object Model

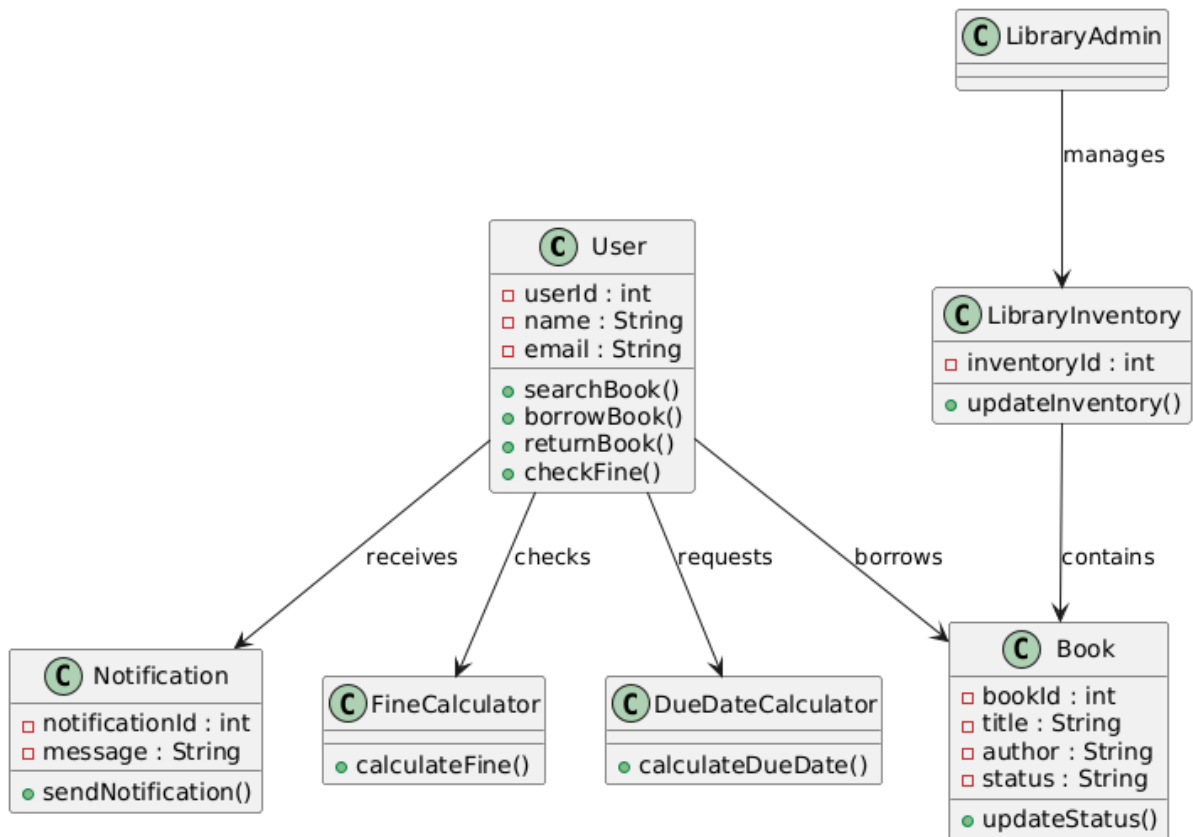


Figure 4 Main Scenario: Book Borrowing and Returning Process Object Model

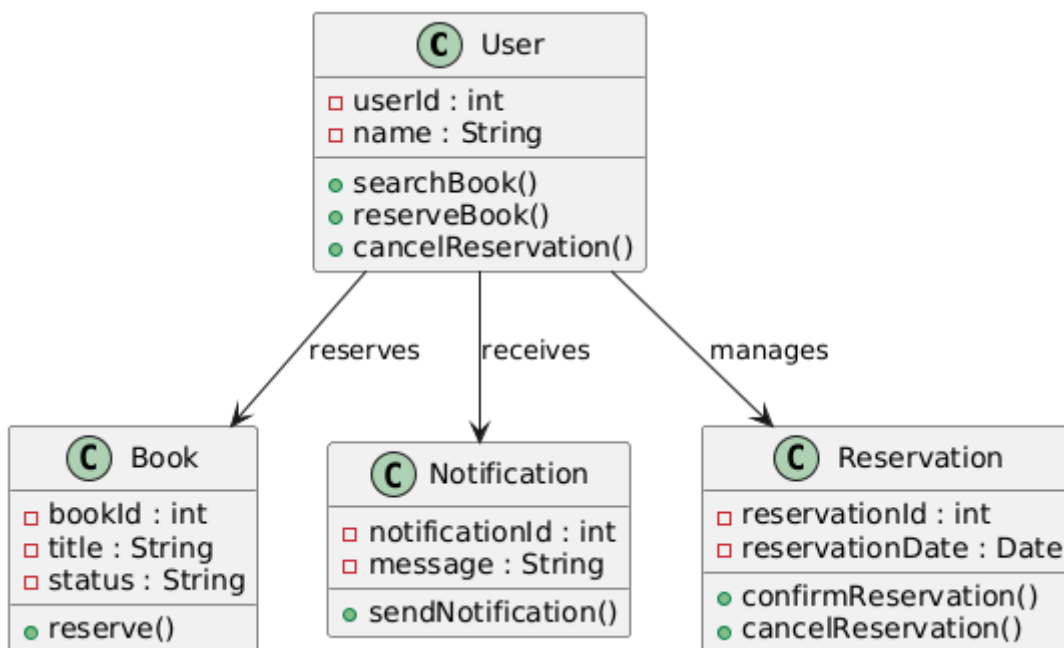


Figure 5 Side Scenario 1: Book Reservation Process Object Model

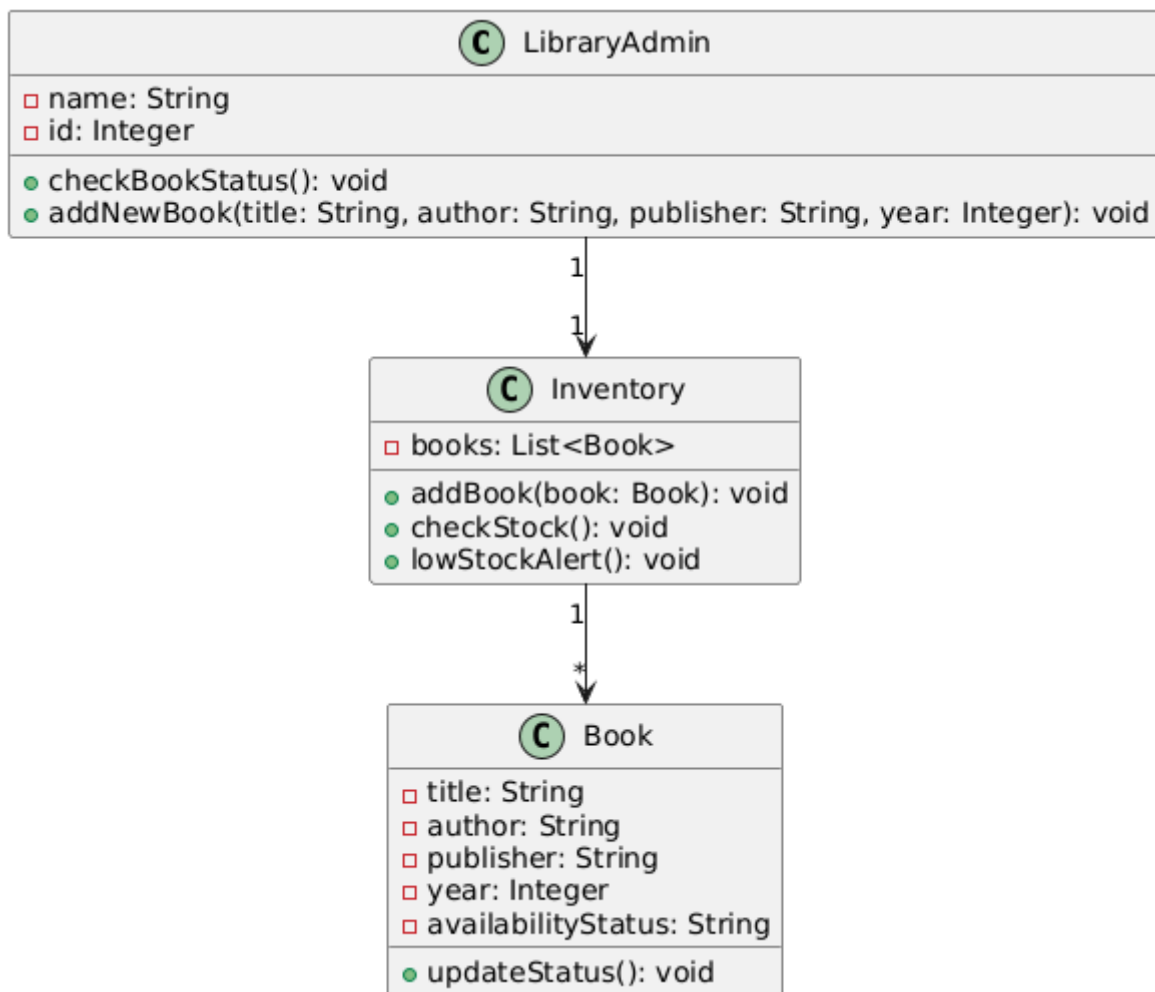


Figure 6 Side Scenario 2: Library Inventory and Stock Management Object Model

### 3.3.4 Dynamic Models

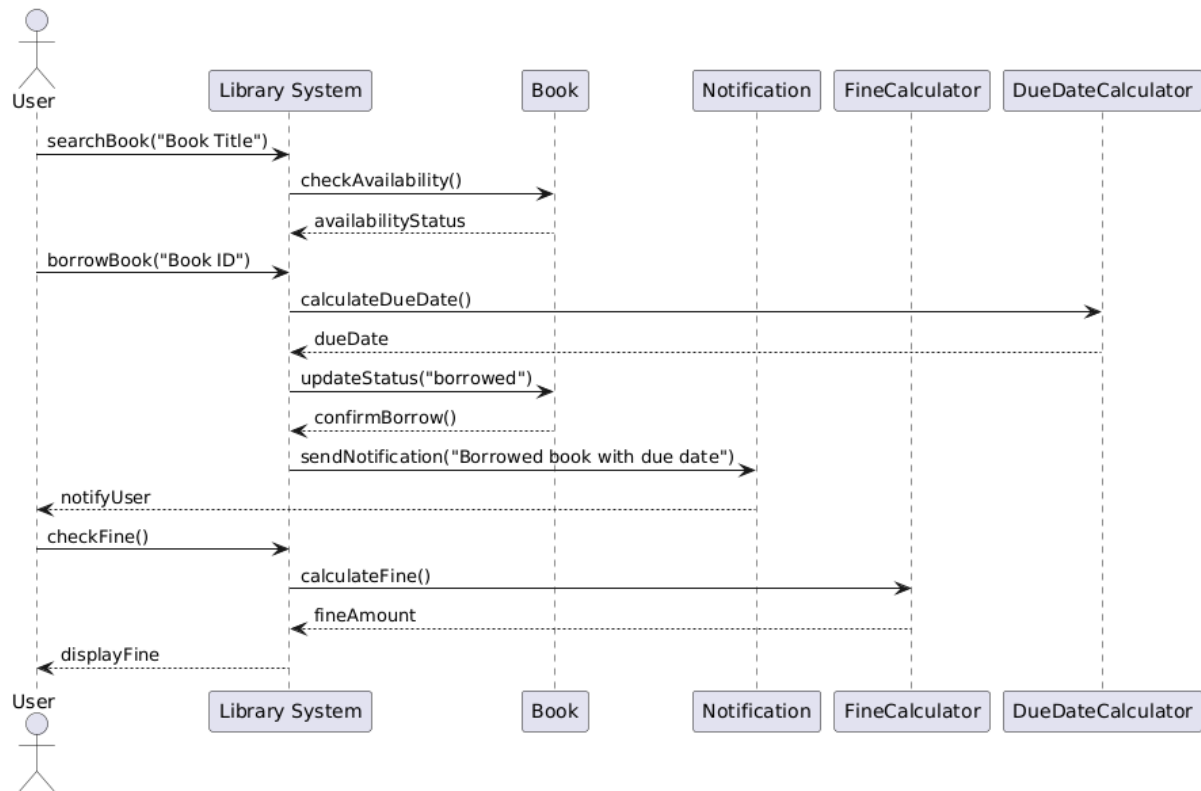


Figure 7 Main Scenario: Book Borrowing and Returning Process Dynamic Model

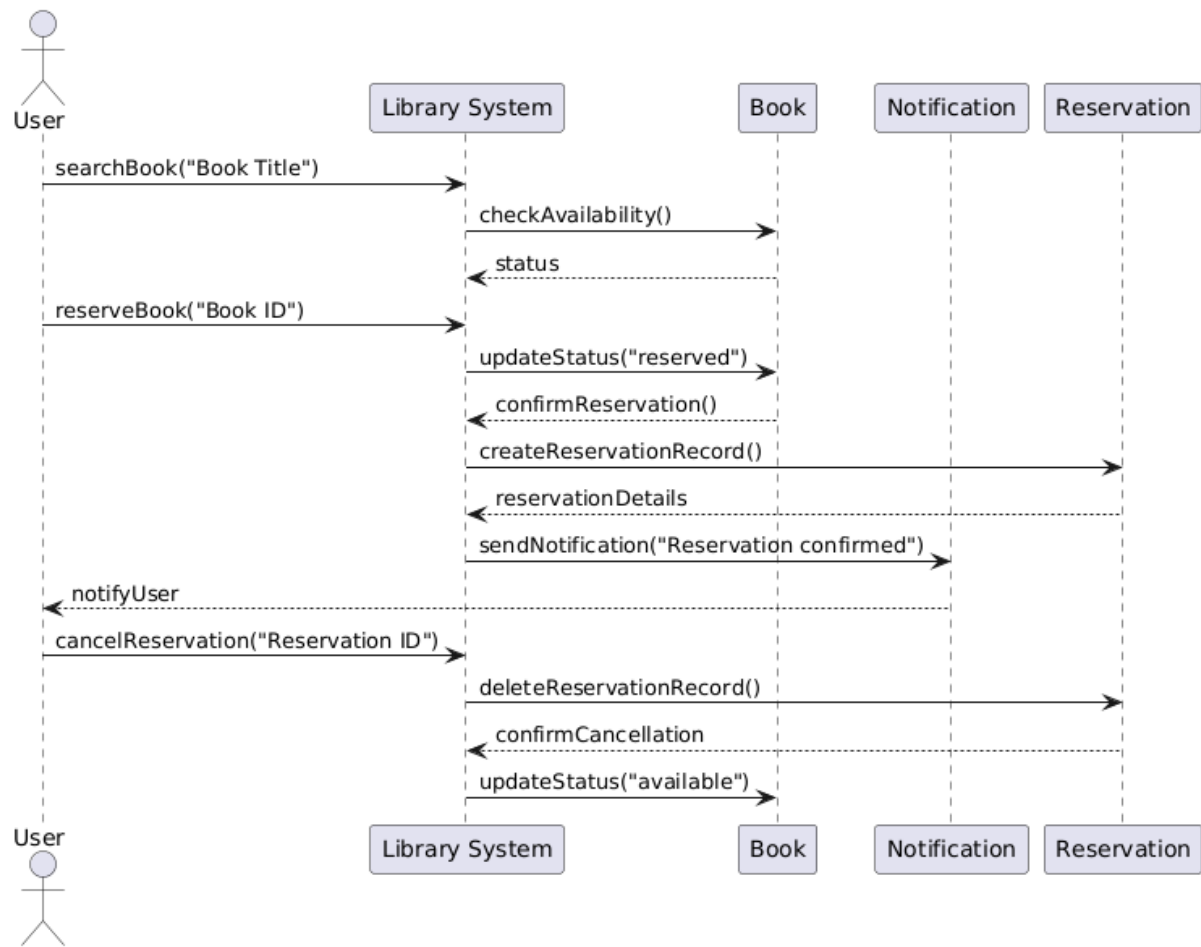
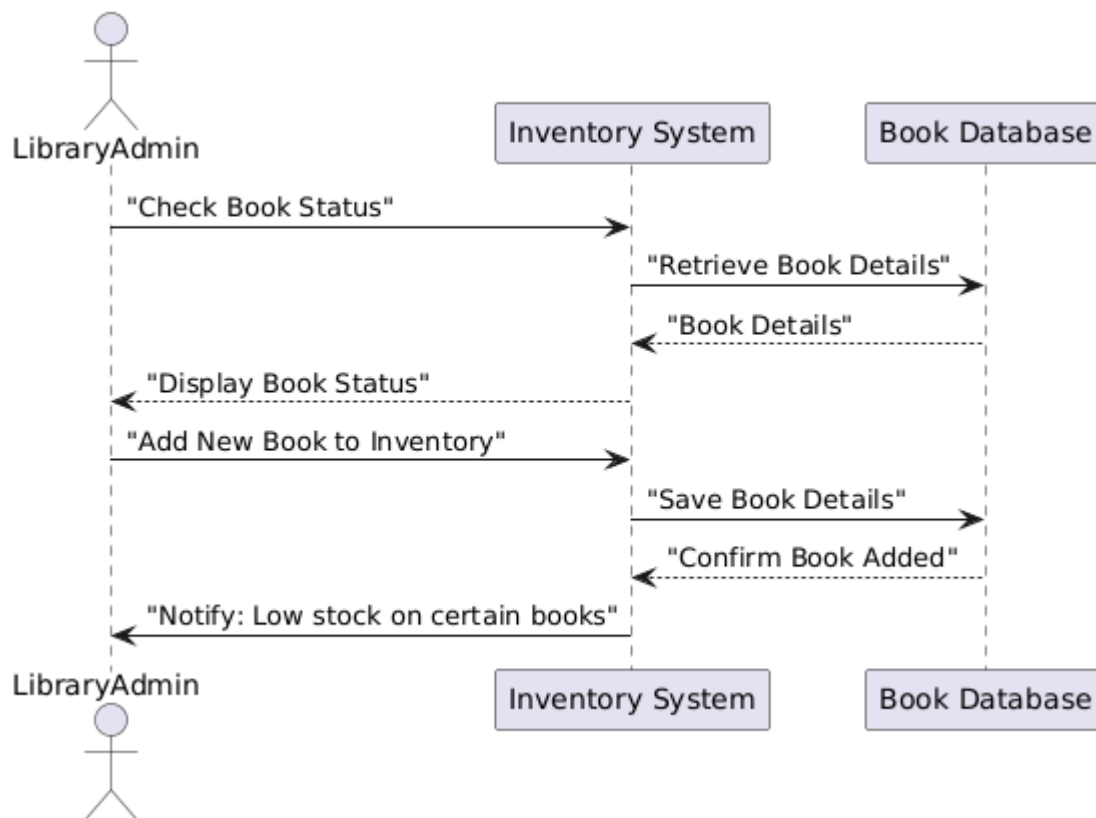


Figure 8 Side Scenario1: Book Reservation Process Dynamic Model



*Figure 9 Side Scenario 2: Library Inventory and Stock Management Dynamic Model*

### 3.3.5 User Interface Mock-ups



*Figure 10 User Login Mock-Up*



*Figure 11 User's Main Menu Mock-Up*

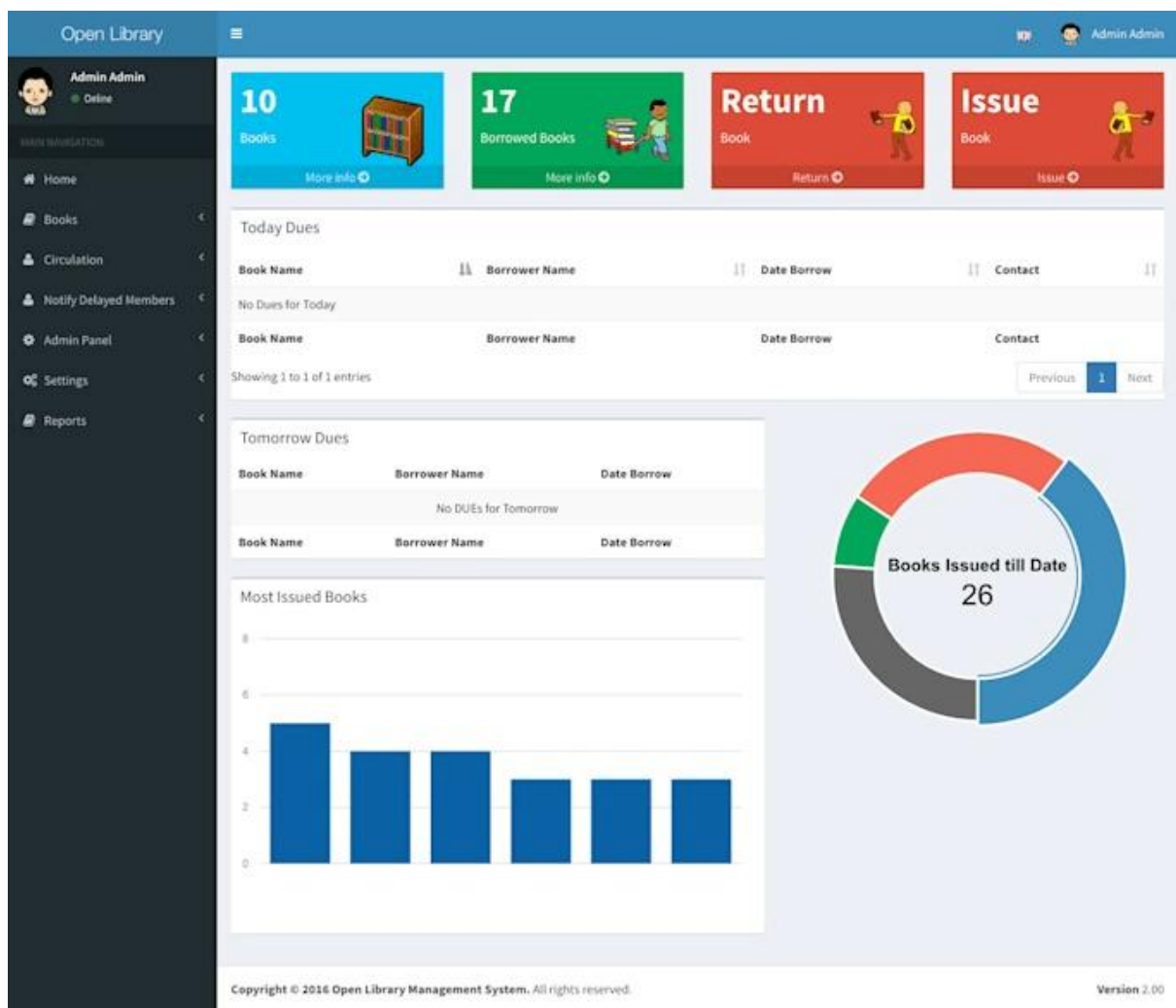


Figure 12 Admin's Main Menu Mock-Up

## 4 Glossary

- **Library Management System (LMS):**  
The primary system designed to manage various library functions, including book borrowing, returning, reservations, and inventory control.
- **Library Member (User):**  
A registered user of the LMS who is authorized to borrow and reserve books. Members include students, faculty, and other library patrons.
- **Librarian (Library Admin):**  
A system user responsible for maintaining the library's inventory, managing book stocks, and assisting members with book borrowing and return issues.
- **Borrowing Request:**  
A request submitted by a library member to borrow a book from the library's catalog. Successful requests lead to a checkout with a defined return date.
- **Return Date:**  
The due date by which a borrowed book must be returned to avoid penalties or late fees.
- **Reservation:**  
A function allowing members to hold a place for a book that is currently checked out. Reserved books are held for the requesting member upon return by the current borrower.
- **Inventory:**  
The total collection of books available in the library, managed and updated by librarians through the LMS. Inventory includes both the available stock and the currently borrowed items.
- **Stock Alert:**  
An automated system notification that informs the librarian when book stock is low, signaling the need for new acquisitions.
- **Catalog:**  
The digital list of books within the LMS, including details such as title, author, genre, and availability status.
- **Availability Status:**  
The current state of a book in the library catalog, which can be "Available," "Borrowed," "Reserved," or "Unavailable."
- **Notification:**  
Automated alerts sent by the system to inform members and librarians of important events, such as book availability for reservations or overdue reminders.
- **Overdue Notification:**  
A reminder sent to members when a borrowed book is past the return date, encouraging prompt return to avoid penalties.
- **Account Management:**  
The functions within the LMS allowing members to register, update, and manage their personal information.
- **Low-Stock Notification:**  
An alert for librarians indicating that the stock for a particular book or resource has reached a low threshold, prompting replenishment actions.



## **5 Appendix**

- Annex – I: Distribution of Work
- Annex – II: Meeting Minutes

## Distribution of Work

Our project team is composed of six members, with each person contributing equally to different project phases to maintain balanced responsibilities:

1. Requirements Analysis – ***Mert Doğan Aygün***: Responsible for gathering initial requirements, documenting system needs, and validating requirements with stakeholders.
2. System Design – ***Berke Alpaslan and Gül Yeşilgil***: Focused on structuring the system's architecture, creating UML diagrams (use-case, object, and sequence diagrams), and defining the relationships between system components.
3. Detailed Design & Dynamic Models – ***Tural Mammadov and Aziz Berk Yıldırım***: Responsible for developing dynamic models, specifying detailed interactions, and refining use cases with precise object behaviors.
4. Documentation & Quality Assurance – ***Birhat Taş***: Handles project documentation, reviews for clarity and accuracy, and ensures the final document aligns with the client's requirements.

**Meeting Minutes**

<b>Date:</b>	27/10/2024
<b>Location:</b>	Microsoft Teams
<b>Duration:</b>	120 min.
<b>Participants:</b>	Berke Alpaslan, Mert Doğan Aygün, Gül Yeşilgil, Birhat Taş, Tural Mammadov, Aziz Berk Yıldırım
<b>Content of the meeting (briefly explain the agenda, decisions, work distributions, etc.)</b>	
<p>The project requirements were discussed, and scenarios and requirements were determined. Then, task distribution was done according to these requirements and scenarios</p>	