Library Management System

System Requirements Specification

Number	Name	ID
1.	Mohamed Ali Ahmed	20215063
2.	Mona Fakhri Salem	20198107

Leader Info:

Mohamed Ali

01116685808

moh7sliem@gmail.com

TA:

Eng. Ali Esma

System Requirements

- **Functional Requirements:**

1.User sign-up with role specification (admin, student, librarian).

2.User login with authentication.

3.Edit personal profile information (username, password, age, gender).

4.Admin manages user accounts (delete/ban/unban accounts, change passwords).

5.Add a book with details (name, genre, copies).

6.Remove books from storage.

7.Search books by name or genre.

8.Sort books by name or rating.

9.Borrow books (check availability).

10.Record interest in unavailable books.

11.Request to extend borrowing period (3 max, 1 week each).

12.Return borrowed books and mark them as available.

13.Rate and review books.

14.View reviews for books.

15.Calculate overdue fines ($1/day/book).

- **Non-Functional requirements:**

1.System must persist data using text files.

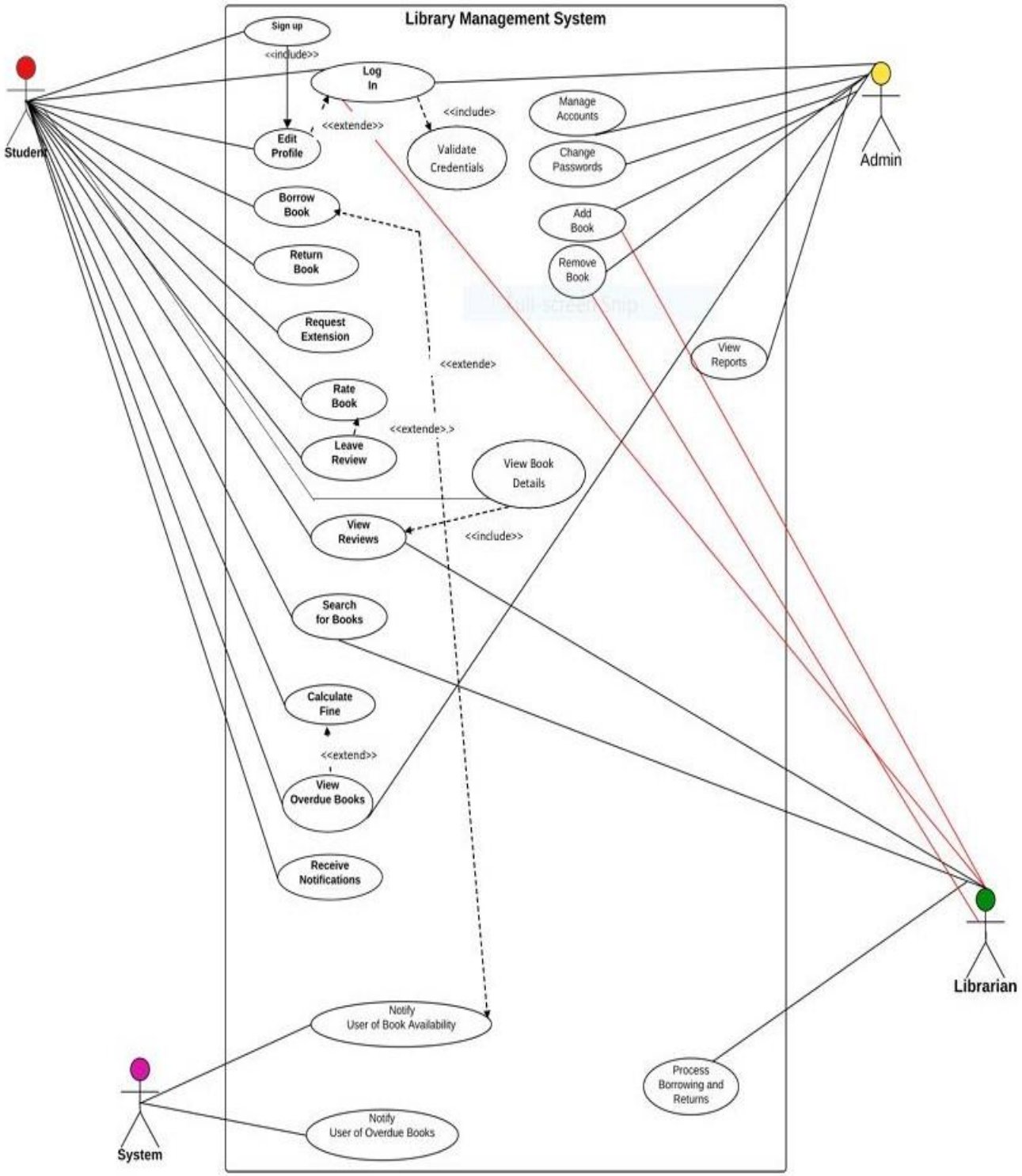
2.The application should handle up to 1000 concurrent users.

3.Response time for any operation should not exceed 2 seconds.

4.The system must provide notifications for availability and overdue books.

5.User-friendly interface for role-based menus.

Use case diagram



Write 4 formal use case descriptions of the most complex use cases in the system.

1))Use Case 1: Borrow Book

Name	Borrow Book
Actor	Student/User
Entry Condition	User must be logged in.
Exit Condition	Book is marked as borrowed.
Flow of Events	User searches for the desired book. 2. System displays availability. 3. User selects to borrow the book. 4. System confirms the borrowing request. 5. System updates the book status to "borrowed." 6. System adds the book to the user's borrowing history.
Alternative Events	If the book is not available, the system allows the user to request notification when the book is available.

Use Case 2: Manage User Accounts

Name	Manage User Accounts
Actor	Admin
Entry Condition	Admin must be logged in.
Exit Condition	User accounts are updated as per admin's actions.
Flow of Events	Admin selects the option to manage user accounts. 2. System displays a list of all user accounts. 3. Admin chooses an account to manage. 4. Admin can create, delete, ban, or unban a user account. 5. System confirms the action taken. 6. System updates the user account data accordingly.
Alternative Events	If the admin attempts to delete an account with outstanding loans, the system prompts a warning.

3)Use Case 3: Calculate Fine

Name	Calculate Fine
Actor	Student/User, Admin
Entry Condition	User must be logged in and have overdue books.
Exit Condition	User receives a detailed breakdown of their fines.
Flow of Events	User selects the option to calculate their fine. 2. System retrieves the list of borrowed books and checks for overdue status. 3. System calculates the total fine based on overdue days. 4. System presents the total fine to the user, along with a breakdown of each overdue book and its corresponding fine. 5. User can choose to pay the fine or review their borrowing history.
Alternative Events	If there are no overdue books, the system informs the user that there are no fines due.

4) Use Case 4: Return Book

Name	Return Book
Actor	Student/User, Librarian
Entry Condition	User must be logged in and have borrowed books.
Exit Condition	Book is marked as available in the system.
Flow of Events	User navigates to the borrowed books list. 2. User selects the book to return. 3. System verifies the return eligibility. 4. User confirms the return action. 5. System updates the book status to "available." 6. System removes the book from the user's borrowed list.
Alternative Events	If the book is overdue, the system calculates the fine before allowing the return.