SRS Document

Introduction

- **Purpose:**
  The purpose of this document is to define the requirements for a comprehensive Library Management System. This system is designed to streamline library operations, allowing administrators, librarians, and users to efficiently manage books, borrow/return books, and perform role-specific actions. It serves as a reference for developers, testers, and stakeholders to ensure alignment with functional and non-functional objectives.
- **Scope:**
  The Library Management System aims to simplify library operations by providing features such as user account management, book cataloging, borrowing and returning processes, and fine calculations. The system supports three user roles:

1. **Admin: Manages user accounts and ensures the system's smooth operation.**
2. **Librarian: Handles book inventory and related tasks.**
3. **User: Borrows books, manages personal information, and interacts with the library's collection.**

Overall Description

- **Product Perspective:**
  The Library Management System is a standalone web-based application designed to digitize and simplify library workflows. It interacts with a centralized database to manage user accounts, book inventories, and transactional data. The system provides role-based access, ensuring that users only interact with features relevant to their roles.
- **Product Functions**
  1. **User Management: Sign up, log in, profile editing, and role-specific permissions.**

2. **Book Inventory: Add, update, remove, and search for books.**
3. **Transaction Management: Borrowing, returning, and extending borrowing periods.**
4. **Notifications: Alerts for book availability and overdue books.**
5. **Fine Calculation: Detailed reports on fines for overdue books.**
6. **Ratings and Reviews: Users can rate books and leave reviews for others to see.**

- **Assumptions and Dependencies**

1. **Users have access to a stable internet connection.**

2. **The system is compatible with modern web browsers.**

3. **Notifications are delivered via email or in-app alerts.**

Functional Requirements

1. **The system should allow users to Sign Up /Login in providing necessary details such as name, email, and password or Google account.**
2. **The system should allow users to the users should specify their roles and responsibilities like (Admin, user, librarian).**
3. **The system should display a menu showing functionalities according to each role.**
4. **The system should allow users to edit their personal profile information like username, password, age, and gender, but not their role.**
5. **The system should allow administrators to edit other accounts including deleting, banning, unbanning, and changing passwords for user and librarian accounts.**
6. **The system should allow appropriate users to add and remove books from the storage.**
7. **The system should allow users to add books by providing necessary information like name, genre, and number of copies.**
8. **The system should allow users to send a request to borrow a book. If the book is available the book should be marked as borrowed, otherwise it**

should be recorded that the user is interested in such a book.

9. **The system should allow users to request an extension for the borrowing period, provided they have no overdue books or unpaid fines. The system should allow up to three extensions, each lasting one week.**
10. **The system should allow provide a way for the user to return the book and the book should be marked as available after completion.**
11. **The system will allow the user to rate the books and leave a review for each book.**
12. **The system should allow the user to Users shall be able to search for books by name or genre. Also, they should be able to sort the results by name or by rating.**
13. **The system should allow users to view other reviews**
14. **The system should allow users to request to calculate their total fine based on how long a book is overdue and how many books are overdue.**
15. **The system will send a notification to users if a book they are interested in is now available and will notify the users if one of their books is now overdue.**

Non-Functional Requirements

1. **Performance: The system must handle a high volume of concurrent users without significant degradation in performance, Page load times should not exceed 3 seconds under normal conditions.**
2. **Availability: Users can access websites any time and everywhere and provide dozens of books.**
3. **Security: The system should encrypt users' Info, password and payment details.**
4. **Usability: The system allows system interfaces to be friendly, easy to navigate and intuitive.**
5. **Scalability: The system should scale the increasing rate of volume and users.**

Use case descriptions no 1

- **Use case name: create an account**
- **Actor: User, librarian, administrator.**
- **Preconditions: The user is not already registered in the system.**
- **Main flow:**
  1. **The user navigates to the sign-up page.**
  2. **The user enters personal details (username, password, age, gender) and selects a role.**
  3. **The system validates the entered information.**
  4. **The system creates a new account for the user.**
  5. **The system displays a confirmation message via email or authenticator app.**
  - **Postconditions: The user's account is created, and they can log in to the system.**

Use case descriptions no 2

- **Use case name: Borrow Book**
- **Actor: User**
- **Preconditions: The user is logged in**
- **Main flow:**
    1. **The system checks the availability of the book.**
    2. **If the book is available, the system marks it as borrowed.**
    3. **If the book is not available, the system marks it as user’s interest.**
- **Postcondition:**
  The book is marked as borrowed, or the user's interest is recorded.

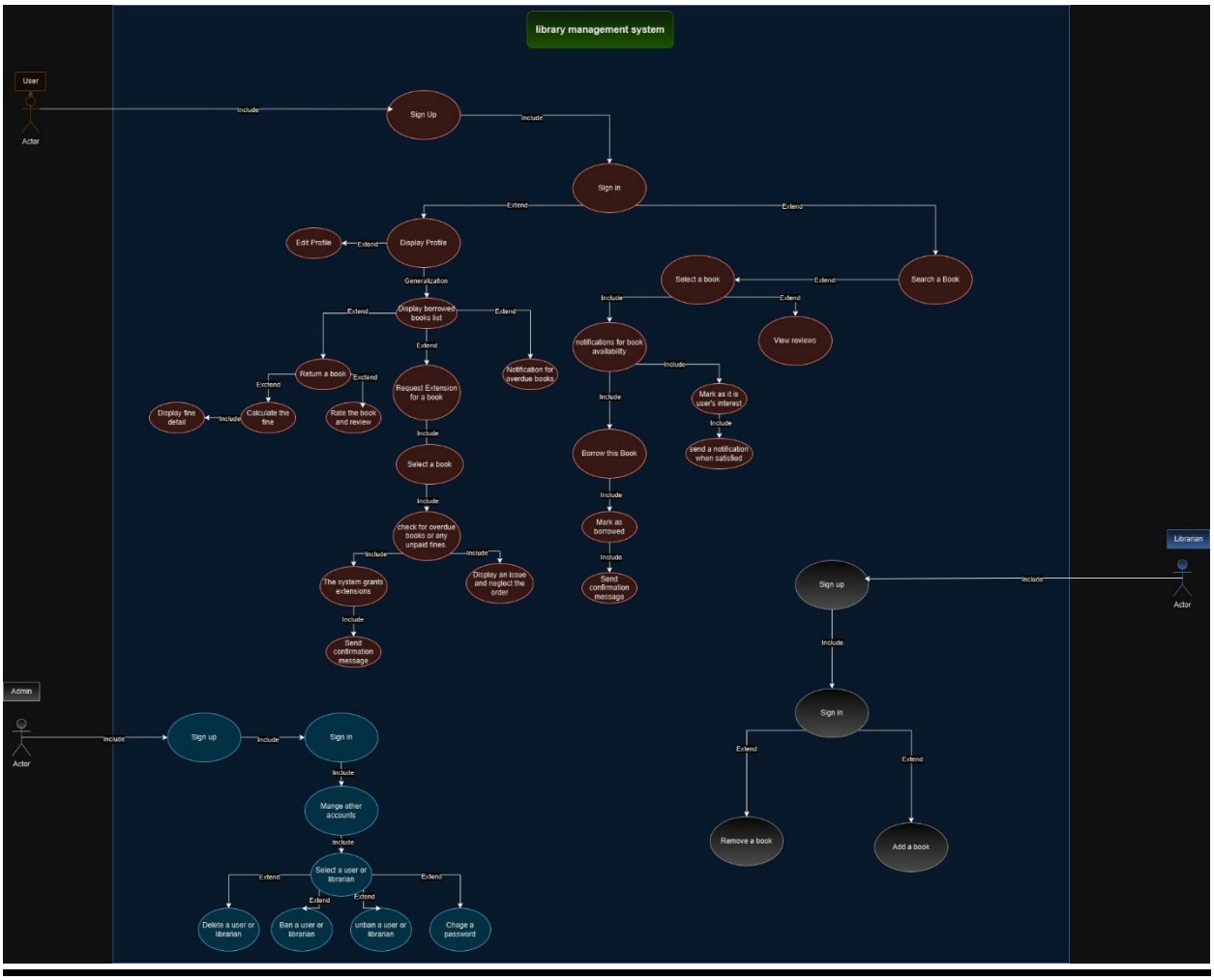
Use case descriptions no 3

- **Use case name: Request Extension**
- **Actor: User**
  Preconditions: The user is logged in and has borrowed a book.
- **Main flow:**
  1. **The user navigates to the extension request page.**
  2. **The user selects the book for which they want an extension.**
  3. **The system checks if the user has any overdue books or unpaid fines.**
  4. **If the user is eligible, the system grants the extension.**
  5. **The system updates the borrowing period and displays a confirmation message.**
- **Postcondition:**

Use case descriptions no 4

- **Use case name: Calculate Fine**
- **Actor: User**
- **Postcondition: The user is logged in and there is an overdue book**
- **Main flow:**
    1. **The user navigates to the fine calculation page.**
    2. **The system calculates the total fine based on the number of overdue books and the number of days each book is overdue.**
    3. **The system displays the total fine and the details of each overdue book.**
- **Postconditions: The user is informed of their total fine.**

UML


library management system
User
Actor
Include
Sign Up
Include
Sign in
Extend
Extend
Edit Profile
Extend
Display Profile
Generalization
Select a book
Extend
Search a Book
Include
Extend
Display borrowed books list
Extend
Extend
Extend
Return a book
Extend
Request Extension for a book
Notification for overdue books
Extend
Display fine detail
Include
Calculate the fine
Rate the book and review
Include
Select a book
Include
check for overdue books or any unpaid fines
Include
Include
The system grants extensions
Display an issue and neglect the order
Include
Send confirmation message
notifications for book availability
Include
View reviews
Include
Mark as it is user's interest
Include
Send a notification when satisfied
Include
Borrow this Book
Include
Mark as borrowed
Include
Send confirmation message
Librarian
Actor
Include
Sign up
Include
Sign in
Extend
Extend
Remove a book
Add a book
Admin
Actor
Include
Sign up
Include
Sign in
Include
Mange other accounts
Include
Select a user or librarian
Extend
Extend
Extend
Extend
Delete a user or librarian
Ban a user or librarian
unban a user or librarian
Chage a password

Glossary

- **Overdue Book: A book not returned by the due date.**
- **Borrowed Book: A book currently checked out by a user.**
- **Extension Request: A user's request to prolong the borrowing period of a book.**