Software Requirements Specification (SRS)

The Library Management System

Contents

1. Introduction

1.1 Purpose
This document outlines the functional and non-functional requirements for the Library Management System. It serves as a guide for the development of the system, ensuring all user needs are met effectively.

1.2 Scope
The Library Management System allows users to manage accounts, borrow and return books, search for books, and leave reviews. It provides roles for Users, Librarians, and Admins with tailored functionalities.

1.3 Definitions, Acronyms, and Abbreviations
- User: A member who borrows and reviews books.
- Librarian: Manages book inventory and assists users.
- Admin: Oversees the system and manages user accounts.

1.4 Overview
The system includes functionalities such as account creation, book borrowing and returning, and fine management. Notifications are provided for overdue books and available reserved books.

2. Functional Requirements

1. User login
2. User Registration
3. User logout
4. Different menu based on type of user
5. Account Management by Admin (The system should allow the Admin to manage users password)
6. Add book
7. Remove book
8. Search book
9. Sort books by name or rating
10. Borrow Books
11. Return books
12. Book Availability Notifications
13. Overdue Notifications (will notify users when due time is close)
14. Comments/reviews
15. History

3. Non-functional Requirements

1. Performance: The system should respond within 2 seconds for searches and operations.
2. Scalability: Support up to 50,000 books and 10,000 users.
3. Security: Encrypted passwords and secure user data handling.
4. Usability: The interface must be intuitive and user-friendly.
5. Availability: Operate 24/7 with prior maintenance notifications , history

4. Formal Use Case Descriptions

1. Borrow Book
Actors: User, System
Preconditions: The user is logged in; the book exists in the database.
Main Flow:
1. User searches for a book.
2. Selects the book and requests to borrow.
3. System checks availability and updates records.
Postconditions: Book is marked borrowed or user added to the waiting list.

2. Return Book
Actors: User, System
Preconditions: The user has borrowed the book.
Main Flow:
1. User selects a book to return.
2. System marks the book as available, removes it from the user's list, and calculates fines if overdue.
Postconditions: Book is marked as available; fines updated.
Alternative Flows:

1. **If the system detects a return attempt for a book not marked as borrowed by the user:**
   - The system notifies the user of the error and does not proceed with the return process.
2. **If the user has lost the book:**
   - The user reports the loss.
   - The system calculates and notifies the user of the replacement cost or associated penalty, which must be resolved to update the system records.

3 . Edit User Profile

Actors: User, System

Preconditions: The user is logged in.

Main Flow:

1. User navigates to the profile page and updates fields.
2. System validates changes and saves them.

Postconditions: Profile information is updated.

Alternative Flows:

1. If the user enters invalid data (e.g., incorrect email format):
   - The system highlights the erroneous fields and prompts the user to correct them.
2. If the user attempts to use an already-registered email address:
   - The system notifies the user and suggests using a different email.

4. Search and Sort Books

Actors: User, System

Preconditions:

Books exist in the database.

Main Flow:

1. User enters a search query and optionally sorts results by name or rating.
2. 2. System retrieves and displays the results.

Postconditions: Relevant books are displayed in the desired order.

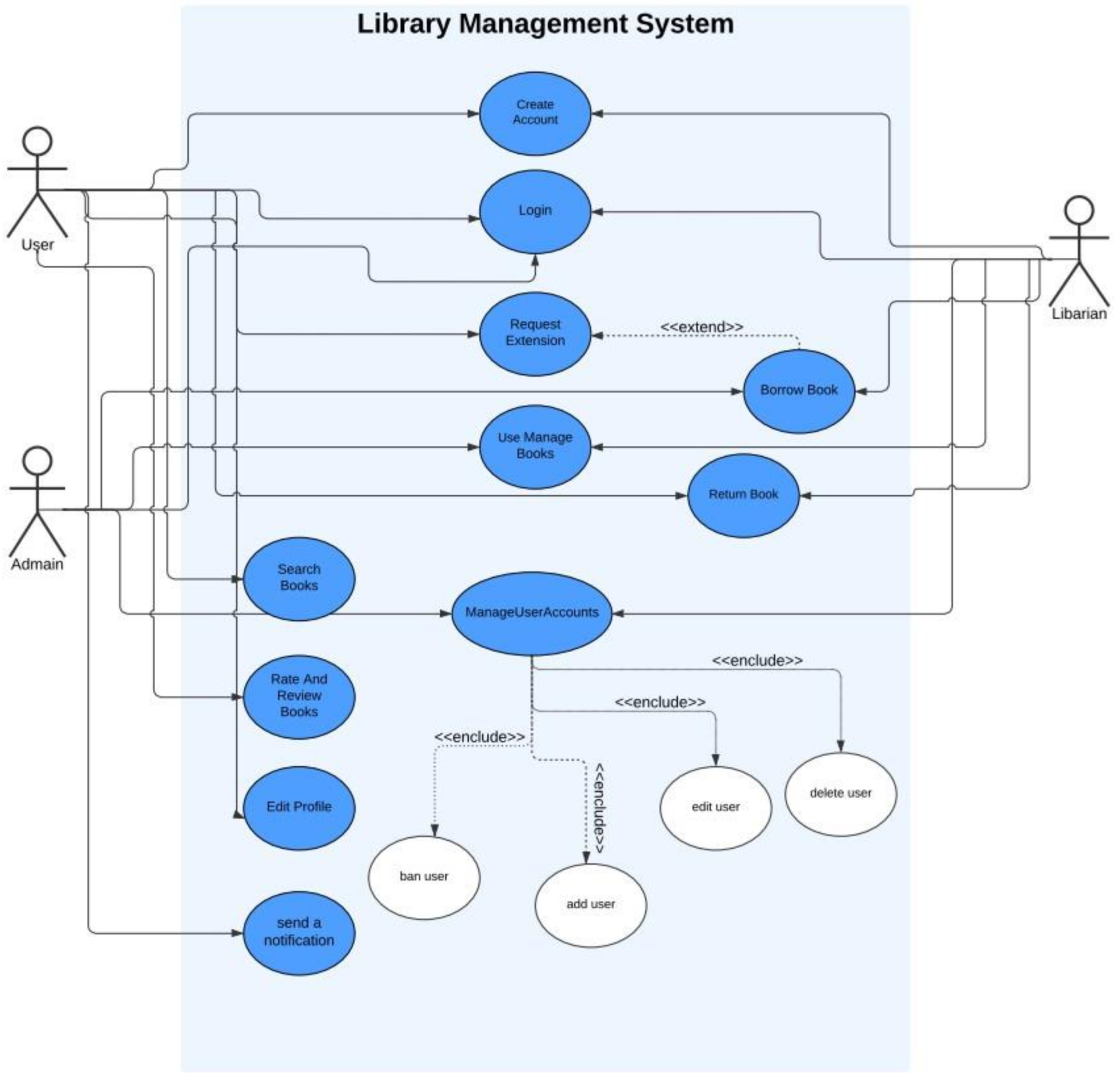
Alternative Flows:

1. If the user enters invalid data (e.g., incorrect email format):
   - The system highlights the erroneous fields and prompts the user to correct them.
2. If the user attempts to use an already-registered email address:
   - The system notifies the user and suggests using a different email.

4. Use Case Diagram

Description:

The Use Case Diagram below provides an overview of the system's interactions with its actors (User, Librarian, and Admin) and the key functionalities. It highlights the relationships between actors and use cases.



- **Actors:**
  - User: Can borrow books, return books, and perform profile-related operations.
  - Librarian: Manages books and interacts with user accounts.
  - Admin: Handles user management and system oversight.
- **Main Use Cases:**
  - Create Account
  - Login
  - Manage User Accounts
  - User Manage Book
  - Borrow Book
  - Return Book
  - Search Books
  - Rate and Review Books