

Application Requirements Specification
For
Library Management System

Presented By

Anujna Attigadde Jagadish
Samanthaka Manogna Tadikonda
Suja Shaj

Purpose and Scope Statement

The *Library Management System* is designed to optimize the management of library resources and improve user experiences for administrators, members and guest users. By providing user-friendly functionalities, the system aims to simplify day-to-day library operations such as book catalog management, member registration, and book reservations, thereby reducing manual tasks and the likelihood of errors.

Scope

The system will support three types of user roles:

1. **Admin:** The primary system manager responsible for overseeing the overall library operations. The admin's responsibilities include managing the book catalog, maintaining the member database, handling reservations, and supervising member registration and account management.
2. **Member:** A registered user with borrowing privileges who can search the book catalog, place reservations, view borrowing history, and update personal profile information.
3. **Guest:** An unregistered visitor with limited access to the system who can browse the book catalog and register for a full member account.

Out of Scope

1. The system will not include features for tracking fines for overdue books.
2. The system will not integrate with a barcode scanner.

Requirements Narrative

Admin Role Overview:

1. **Log in with admin credentials:** Admins can securely access the system using their unique credentials.
2. **Add a new book to the catalog:** Admins can input details such as title, author, and genre to expand the book catalog.
3. **Update or delete an existing book:** Admins can modify or remove book information as needed to keep the catalog up to date.
4. **Search for a book by ID, title, author, genre:** Admins can quickly find books using different search criteria for efficient management.
5. **Register a new library member:** Admins can add new members, inputting necessary personal information.
6. **Update or delete an existing member:** Admins can modify member information or remove a member from the database.
7. **Search for a member by ID or name:** Admins can locate member records efficiently by using either ID or name.
8. **Approve book borrowing and returns:** Admins record in the system when a book is borrowed by a member and when it is returned, ensuring accurate tracking of book circulation.

Member Role Overview:

1. **Log in with member credentials:** Members can securely access their accounts.
2. **Search for a book by ID, title, author, or genre:** Members can browse the library's collection based on their preferences.
3. **Reserve a book:** Members can place reservations on available books to ensure they are held for their borrowing.
4. **View personal account details:** Members have access to their borrowing history and account information.

Guest Role Overview:

1. **Search for books by title, author, or genre:** Guests can browse the library's collection even without an account.
2. **Sign up for the library membership:** Guests can create an account to become a member and gain additional privileges, such as book reservations and borrowing.

Database Interactions:

Necessary Tables:

1. **Books Table**
 - Attributes: bookID, title, author, genre.
2. **Members Table**
 - Attributes: memberID, name, email, password.
3. **Reservations Table**
 - Attributes: ID, bookID, memberID, returnDate, borrowDate

Key Methods:

1. **Retrieve:** For browsing books and viewing member details.
2. **Update:** For modifying book or member information.
3. **Insert:** For adding new entries (books or members).
4. **Delete:** For removing entries.

Technologies To-Be Used:

1. **Backend Framework:** Spring (Java).
2. **UI Framework:** Swing.
3. **Database:** PostgreSQL.

Objectives

Login screen

LOGIN

EMAIL

PASSWORD

LOGIN

NEW USER ?

CONTINUE AS A GUEST

Signup Page

SIGNUP

NAME

EMAIL

PASSWORD

SIGNUP

LOGIN

Member Home Page

MEMBER HOME PAGE

LOGOUT

BROWSE BOOKS

VIEW PROFILE

Member Book search page

BOOK SEARCH

Search books by ID,Tilte, Author OR genre

Search

SEARCH RESULTS

Book 1 - Click here to view book details

Book 2 - Click here to view book details

Book 3 - Click here to view book details

RESERVE BOOK

BACK TO HOME

Member Profile Page

PROFILE PAGE

MEMBER DETAILS

NAME

MEMBER ID

XXXXXXX

EMAIL

EDIT PROFILE

BORROW HISTORY

BOOK ID	BOOK TITLE	BORROW DATE	RETURN DATE

BACK TO HOME

Admin Home Page View

ADMIN HOME PAGE

MANAGE BOOKS

MANAGE MEMBERS

APPROVE BORROW / RETURN

ADD MEMBER

ADD BOOK

LOGOUT

Admin Add Book View

ADD BOOK

TITLE

AUTHOR

GENRE

COUNT

ADD BOOK

BACK TO HOME

Admin Update/Delete Book

MANAGE BOOKS

Search Books by ID, Title, Author OR genre

SEARCH

SEARCH RESULTS						
	BOOK ID	TITLE	AUTHOR	GENRE	BORROWED BY	STATUS
<input checked="" type="checkbox"/>						<div>BORROW</div>
<input type="checkbox"/>						<div>RETURN</div>
<input type="checkbox"/>						

EDIT BOOK

DELETE BOOK

BACK TO HOME

Edit Book

EDIT BOOK

BOOK ID

XXXXXXXX

TITLE

AUTHOR

GENRE

SAVE CHANGES

BACK TO HOME

Admin Add Member View

ADD MEMBER

NAME

EMAIL

PASSWORD

CONFIRM PASSWORD

ADD NEW MEMBER

BACK TO HOME

Admin Update/Delete Member View

MANAGE MEMBERS

Search member by name or ID

SEARCH

SEARCH RESULTS		
MEMBER ID	NAME	EMAIL
<input checked="" type="checkbox"/>		
<input type="checkbox"/>		

VIEW MEMBER

EDIT MEMBER

DELETE MEMBER

BACK TO HOME

Edit profile

EDIT MEMBER

MEMBER ID

XXXXXXX

NAME

EMAIL

PASSWORD

CONFIRM PASSWORD

SUBMIT CHANGES

BACK TO HOME

Guest Book search View

BOOK SEARCH

Search books by ID, Title, Author OR genre

Search

SEARCH RESULTS

Book 1 - Click here to view book details

Book 2 - Click here to view book details

Book 3 - Click here to view book details

BACK TO HOME

Functional Specification

User Role Descriptions:

1. **Admin:** Manages books and members. Can add, update, and delete entries in the system. Approves borrowing requests.
2. **Member:** Searches and reserves books. Views personal borrowing history and updates personal information.
3. **Guest:** Explores the book catalog and has the option to register as a member.

Features

Admin Functions:

1. **AddBook**
 - Adds a book to the catalog via a form.
2. **UpdateDeleteBook**
 - Updates or deletes book details.
 - Includes search functionality.
3. **SearchBooks**
 - Allows searching for books by title, author, or genre.
 - Displays search results and supports drill-down into book details.
4. **BookDetails**
 - Shows detailed information about a specific book.
 - Includes borrowed member information and reviews.
5. **AddMember**
 - Registers a new member and assigns roles (admin or member).
6. **UpdateDeleteMember**
 - Updates or deletes member information.
 - Includes search functionality.
7. **AdminHome**
 - Acts as the dashboard for admin functionalities, allowing navigation to various admin tasks.
8. **ApproveBorrow**
 - Displays borrow requests for approval.
9. **ApproveReturn**
 - Displays return requests for approval.

Member Functions:

1. **MemberHome**
 - Acts as the dashboard for member functionalities.
 - Displays overdue messages and borrowed books.

2. **BrowseBooks**

- Allows members to search and browse books.

3. **MemberProfile**

- Displays personal account details.
- Includes functionality to update the profile.

4. **ReserveBook**

- Enables members to reserve books.

Guest Functions:

1. **GuestBrowseBooks**

- Allows book browsing for non-logged-in users.

2. **Register**

- Creates a new member account.

Backend Models:

1. **Book**

- Attributes: id, title, author, genre, count, reservedCount, borrowedCount.
- Represents a book in the catalog.

2. **Member**

- Attributes: id, name, email, password, role.
- Represents a user (admin or member).

3. **Reservation**

- Attributes: id, book, member, actionType, borrowedDate, dueDate, returnedDate.
- Tracks borrowing, reservation, and return actions.

4. **Review**

- Attributes: id, book, member, rating, reviewText, reviewDate.
- Represents member reviews of books.

Backend DTOs (data transfer objects)

1. **BookDTO**

- Used for guest and member book browsing (basic details).

2. **AdminBooksDTO**

- Used for admin book management (includes reserved and borrowed counts).

3. **BookDetailsDTO**

- Used for displaying detailed book information.

- Includes borrowed member and review details.

Backend Controllers

1. BookController

- Handles:
 - Add, update, delete books.
 - Search for books.
 - Fetch book details.

2. MemberController

- Handles:
 - Add, update, delete members.
 - Search for members.
 - Fetch member profile.

3. ReservationController

- Handles:
 - Borrow and return book actions.
 - Fetch overdue reservations.

Backend Repositories

1. BookRepository

- Provides methods for searching and managing books in the database.

2. MemberRepository

- Provides methods for searching and managing members in the database.

3. ReservationRepository

- Provides methods for managing reservations and fetching overdue books.

4. ReviewRepository

- Provides methods for managing book reviews.

Backend Utility

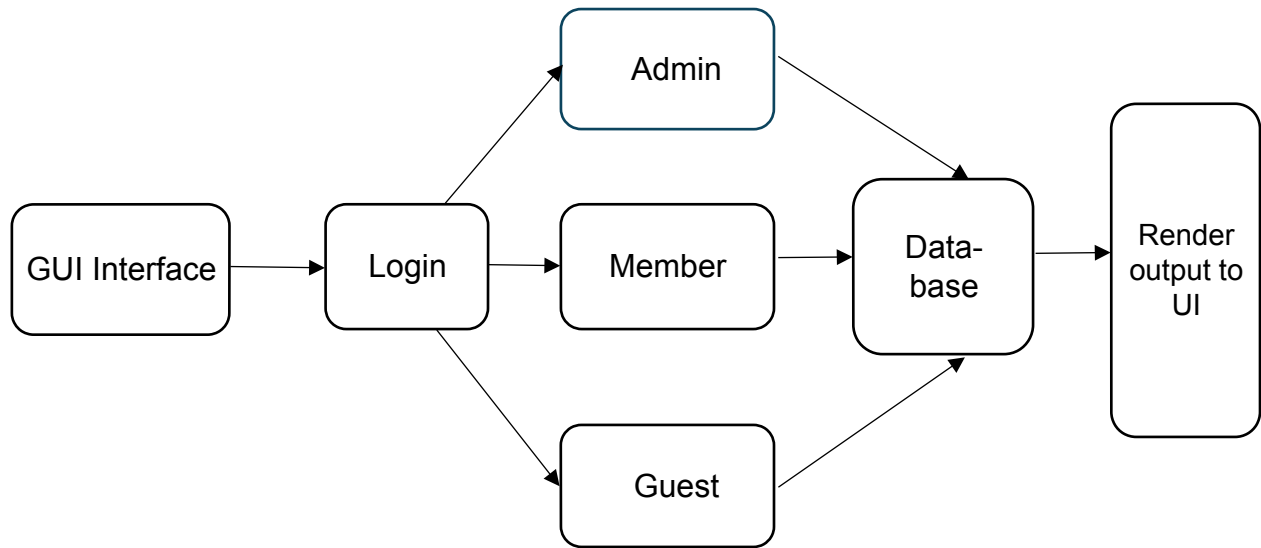
1. ActionType (Enum)

- Defines action types: RESERVE, BORROW, RETURN.

2. DatabaseConnection

- Handles database connections.

Logic Specification



1. **User Interface (UI):** Starting point for Admins, Members, and Guests.
2. **Login:** Authenticates credentials and determines user roles.
3. **Role-Based Views:**
 - **Admin View:** Access to library management features.
 - **Member View:** Access to browsing and personal account details
 - **Guest View:** Limited to browsing and registration.
4. **Database Access:** Queries and updates as per user actions.
5. **Output to UI:** Displays action results on-screen.