LIBRARY MANAGEMENT SYSTEM

AVINASH KUMAR CHAKILAM-801392754 (Database Developer)

BHANU SAISREE THANNIRU-801375971 (Backend Developer)

POOJA REDDY KADARI-801366250 (Front end Developer)

SANJANA BHANU KOCHERLA-801392962 (Tester)

SYSTEM DESCRIPTION

Performance Requirements:

- Capable of managing a significant number of books and users reliably.
- Insights and responses should be displayed within no more than 10 seconds.
- Includes error control mechanisms to handle both expected and unexpected errors, preventing data loss and minimizing downtime.
- Detects fraudulent usernames or passwords to maintain system integrity.

Security Requirements:

- Utilizes a secure database to protect sensitive information.
- Regular users can view but not edit information, except for their personal data and certain other fields.
- Implements distinct access levels for different user roles, ensuring appropriate permissions for each.
- Features strong user authentication to prevent unauthorized access.
- Admins have exclusive rights to modify the database, safeguarding against unauthorized alterations.

Software Requirements:

- Database: MySQL
- Programming Language: Java for backend development.
- Front End Development Tools: HTML, CSS, and Bootstrap for designing a user-friendly interface.
- Front End Frameworks: jQuery.
- Backend Frameworks: Spring Boot.
- Project Management Tool: Maven.
- Development Environment: Eclipse.
- Version Control:GitHub.

Hardware Requirements:

- RAM: Requires 256 MB or more.
- Hard Disk: Minimum of 1 GB space.

SCOPE

- **User and Admin Logins:** Secure access for different roles, ensuring appropriate functionalities are available to each user type.
- **Online Updates and Book Requests:** Real-time updates to the library catalog and the ability to request books online enhance the user experience.
- **Issue and Return Tracking:** Allows users to track their borrowed books and return dates, while admins can oversee these transactions and manage the library's inventory effectively.
- **Report Generation:** Admins can generate various reports to assist in the effective management of the library, providing insights into usage, inventory levels, and more.
- **Concurrent Issue Resolution:** The system supports handling multiple requests simultaneously, streamlining operations and reducing wait times.

GOAL

- **Automate Library Operations:** By digitizing the library's processes, the LMS reduces the reliance on manual work for cataloging, circulation, and reporting, thus increasing efficiency.
- **Enhance Accessibility and Information Reporting:** The system enables quick access to information and facilitates informed decision-making through comprehensive reporting tools.
- **Streamline Borrowing Processes:** By digitizing library cards and tracking book issues and returns, the LMS aims to minimize the occurrences of lost or misplaced books and improve the borrowing process for users.

USER STORIES

User Story: Add new Books

Description: The admin seeks a simplified process for adding new books to the library's database, emphasizing accuracy and efficiency. The "Add New Book" form includes fields for the book details

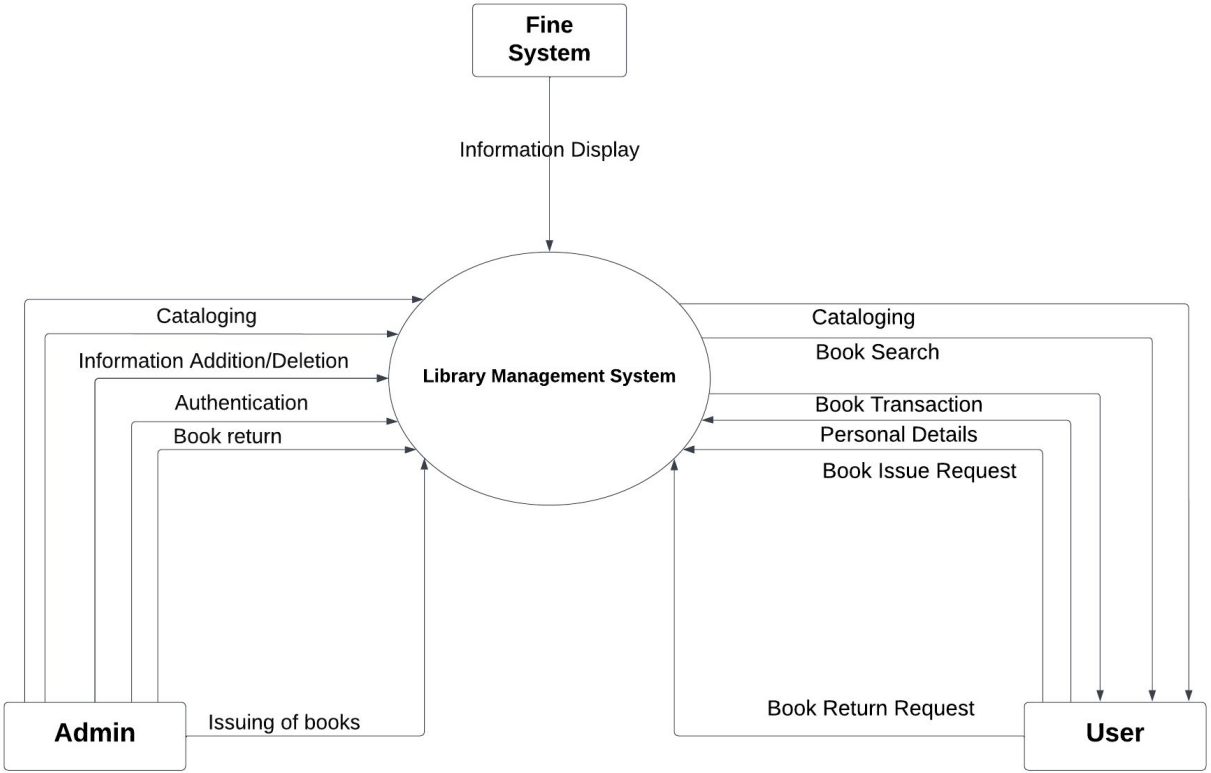
User Story : Update Books

Description: Admins can seamlessly update book details by specifying the modified information, streamlining the process of keeping the database up-to-date.

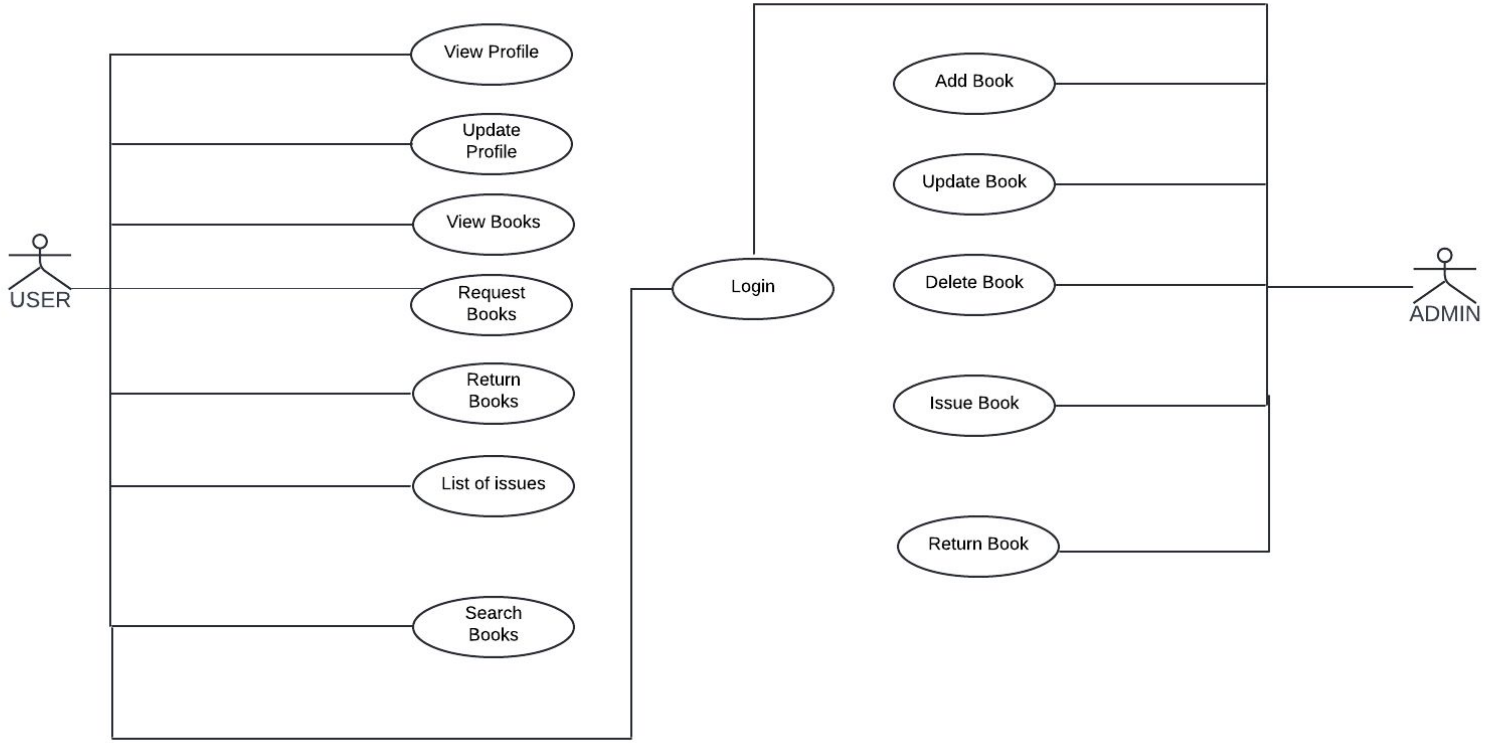
User Story : Update Profile

Description: User can access "Update Profile" feature from there dashboard. The "Update Profile" form includes fields for the updation of a few fields.

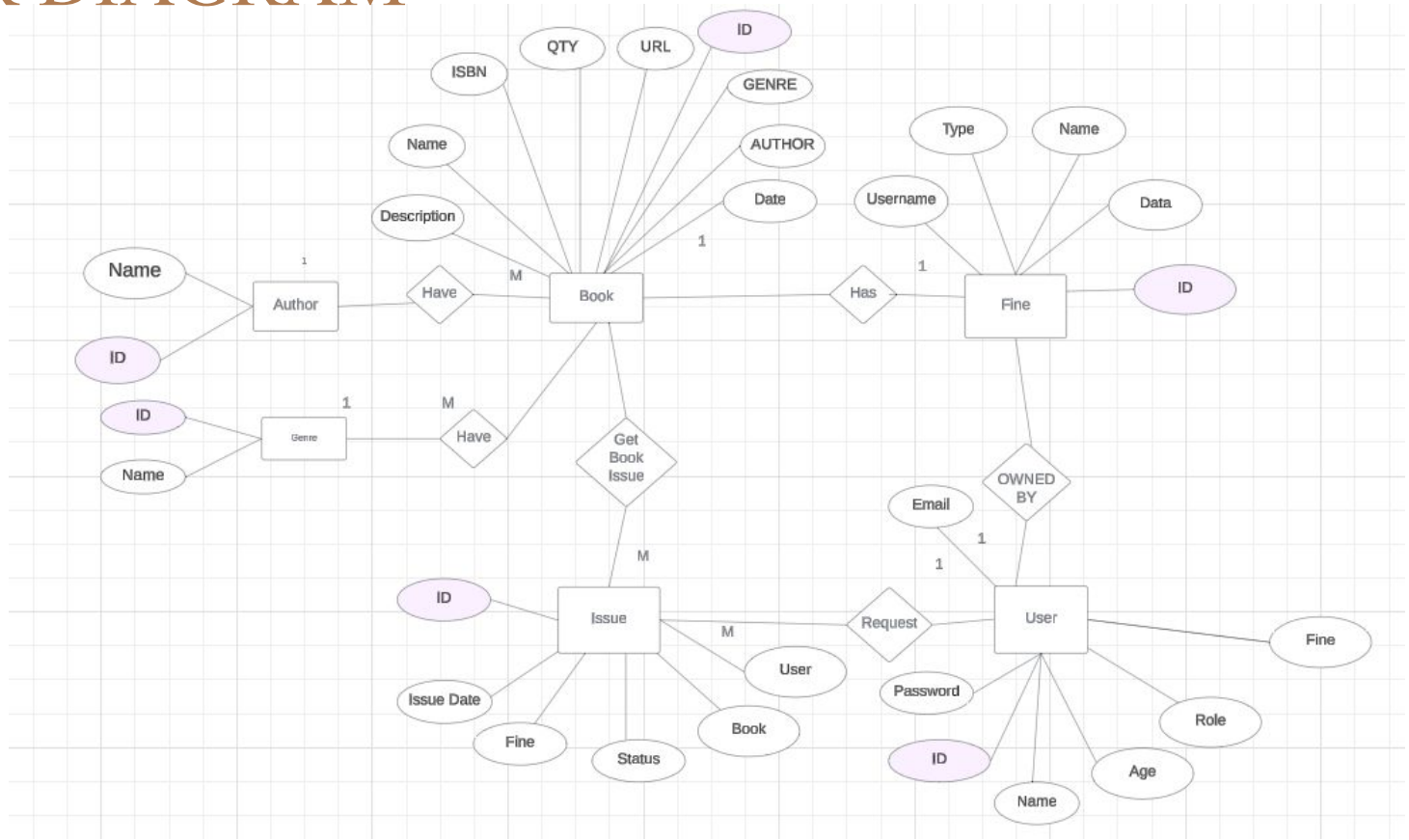
CONTEXT DIAGRAM



USE CASE DIAGRAM

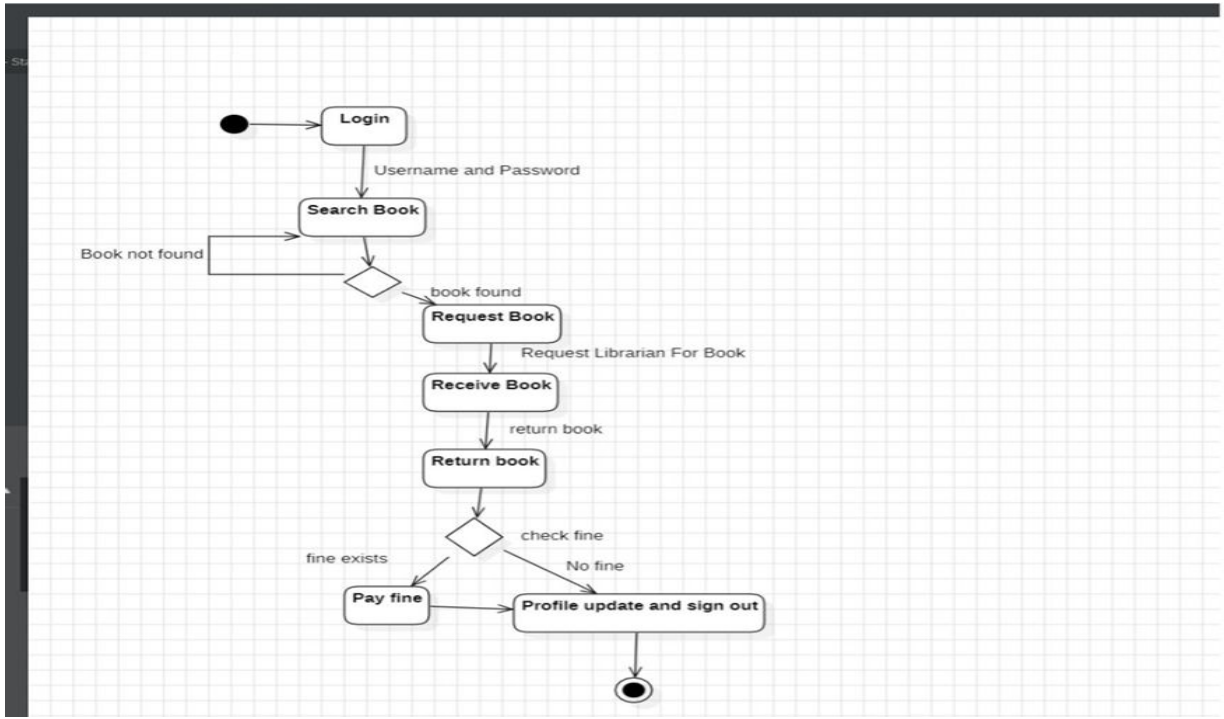


ER DIAGRAM

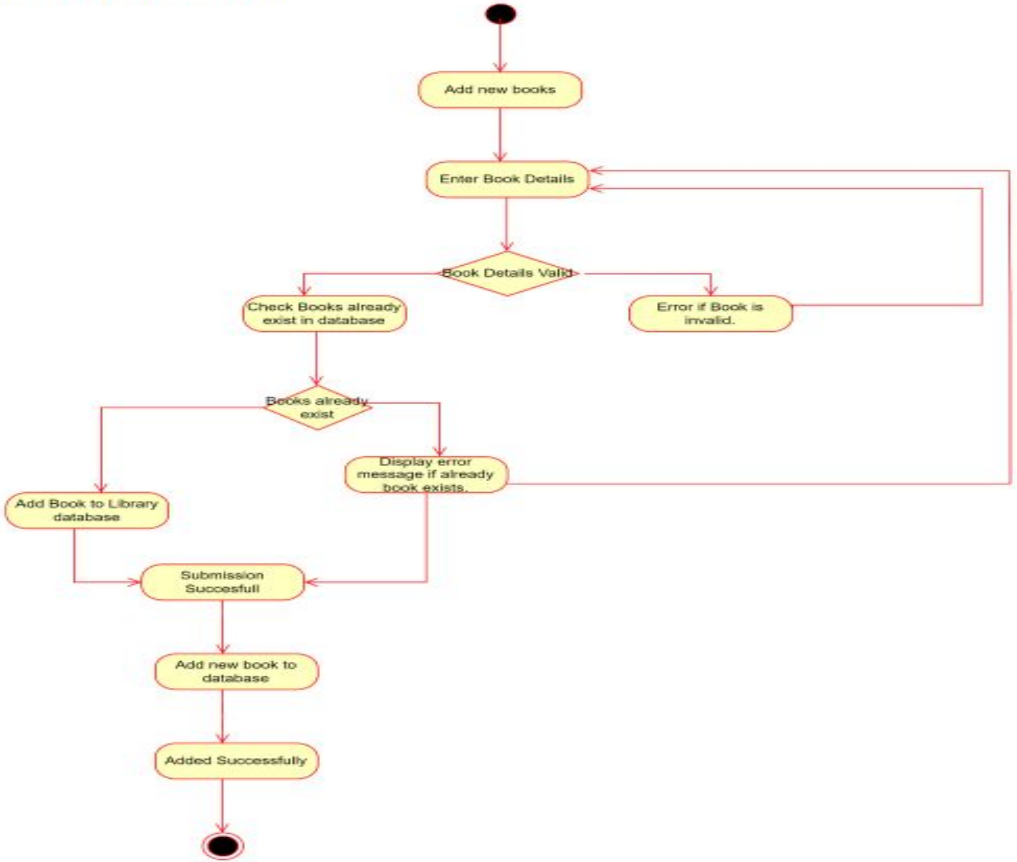

ID
QTY
URL
ISBN
GENRE
Name
AUTHOR
Type
Name
Date
Username
Data
Description
1
Name
1
M
1
Author
Have
Book
Has
Fine
ID
ID
1
M
ID
Genre
Have
Get
Book
Issue
Name
OWNED
BY
Email
1
1
M
ID
Issue
M
Request
User
Fine
User
Password
Issue Date
Book
Role
Fine
Status
ID
Age
Name

OTHER UML DIAGRAMS

State Chart Diagram:



ACTIVITY DIAGRAM:



Lessons Learned:

1. Importance of User-Centered Design

- User Feedback is Crucial: Engaging with actual users (librarians and patrons) during the design and testing phases helps tailor the system to their needs, improving usability and satisfaction.
- Intuitive UI/UX: A user-friendly interface significantly enhances user engagement and reduces the learning curve, making the system accessible to users of all tech-savviness levels.

2. The Value of Robust Testing and Validation

- Early and Often Testing: Continuous testing from the early stages of development helps identify and resolve issues before they escalate, ensuring a smoother deployment.
- Diverse Testing Scenarios: Considering various user scenarios and edge cases in testing ensures the system is robust and can handle unexpected inputs gracefully.

3.**. Security and Data Protection are Paramount**

- Proactive Security Measures: Implementing security measures from the outset is crucial to protect sensitive user data and maintain trust.
- Regular Updates and Patches: Staying abreast of security vulnerabilities and updating the system regularly helps protect against new threats.

4. Scalability and Flexibility

- Anticipating Growth: Designing the system with scalability in mind allows it to accommodate an increasing number of users and resources without significant rework.
- Adaptability to Change: Building flexibility into the system makes it easier to add new features or integrate with other systems as library needs evolve.

5. Project Management and Collaboration

- Effective Communication: Regular meetings and clear communication channels among team members and stakeholders help align the project with user needs and organizational goals.
- Agile Methodology Benefits: Adopting an agile approach, with its iterative development and feedback loops, can improve project outcomes by allowing for flexibility and adaptation to feedback.

Next Steps:

- Work on the other features in the upcoming sprints
- Testing of the application
- Work on the application based on the comments

Screencast Link

Screencast

THANK YOU