Library Management System (LMS)

Bryant Le

Indiana University - Indianapolis

INFO- C 451: SYSTEM IMPLEMENTATION

Jafrina Jabin

March 18, 2025

TABLE OF CONTENTS

Problem Statement

A library management system is utilized to trace book transactions, which include book check-ins and check-outs, and current inventory. The current system keeps track of everything through physical media (e.g., pen and paper), which has resulted in inefficient check-ins and check-outs, librarians failing to keep locating and updating information due to lost paperwork and other acts of human error, and user inconvenience.

To rectify this issue, a digital library management system will provide an opportunity to make library operations more effective by reducing manual errors and improving the user experience. By automating how librarians manage the book database and the process of borrowing and returning books, employee physical workload will be reduced, and users will be able to locate books in real time and therefore increase system efficiency.

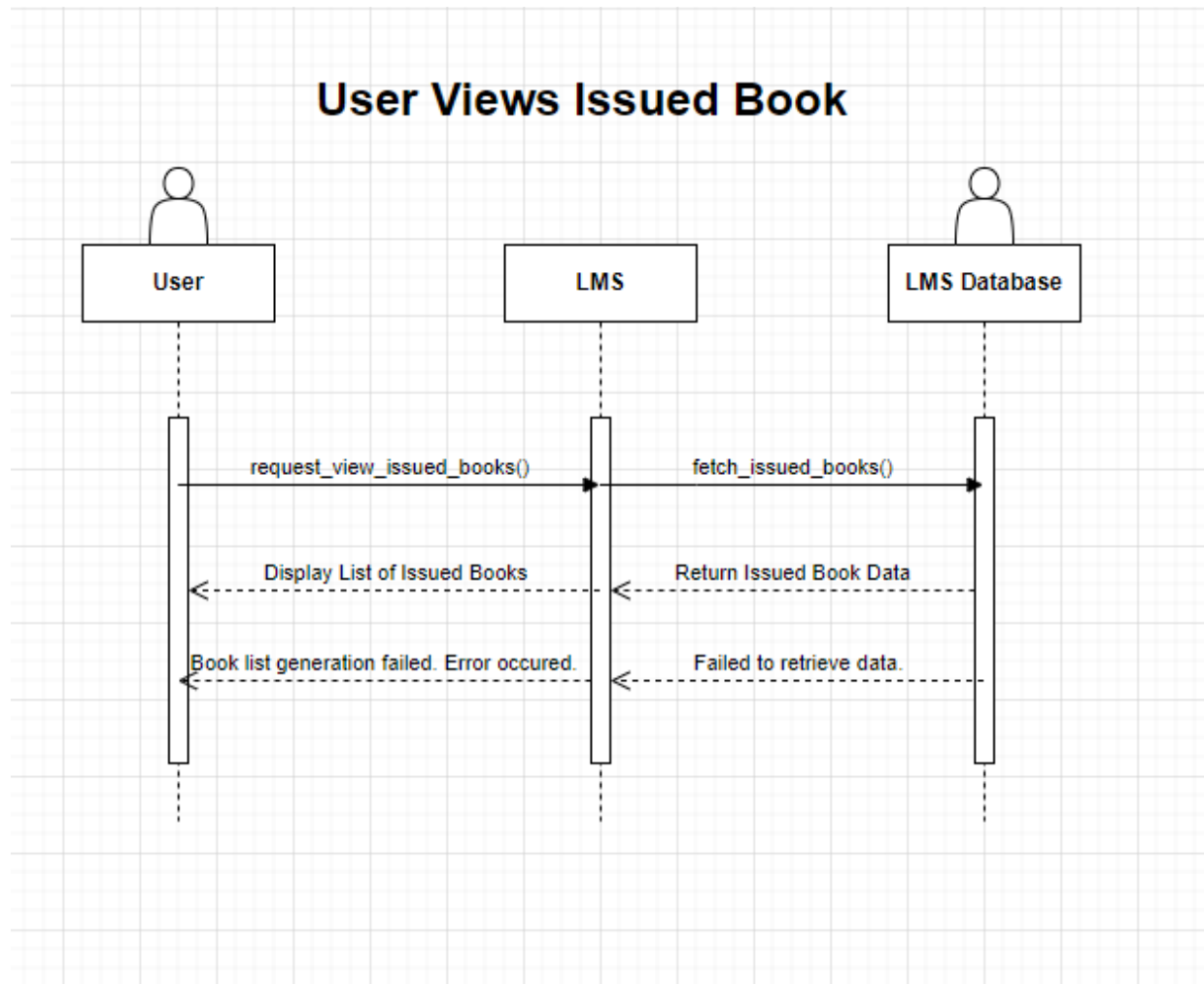**System Requirements:**
- Software
  - Operating System
    - Windows OS
  - Back-end
    - PHP
    - MySQL
  - Front-end
    - HTML
  - Framework
    - Bootstrap

- Hardware
  - Standard desktop for librarians and users
- Server Requirements
  - CPU: Quad-Core Processor (Intel or AMD)
  - RAM: 16 GB
  - Storage: 500 GB SSD (expandable)
- Network
  - 1 Gbps LAN Connection
  - Wi-Fi Access for staff and users
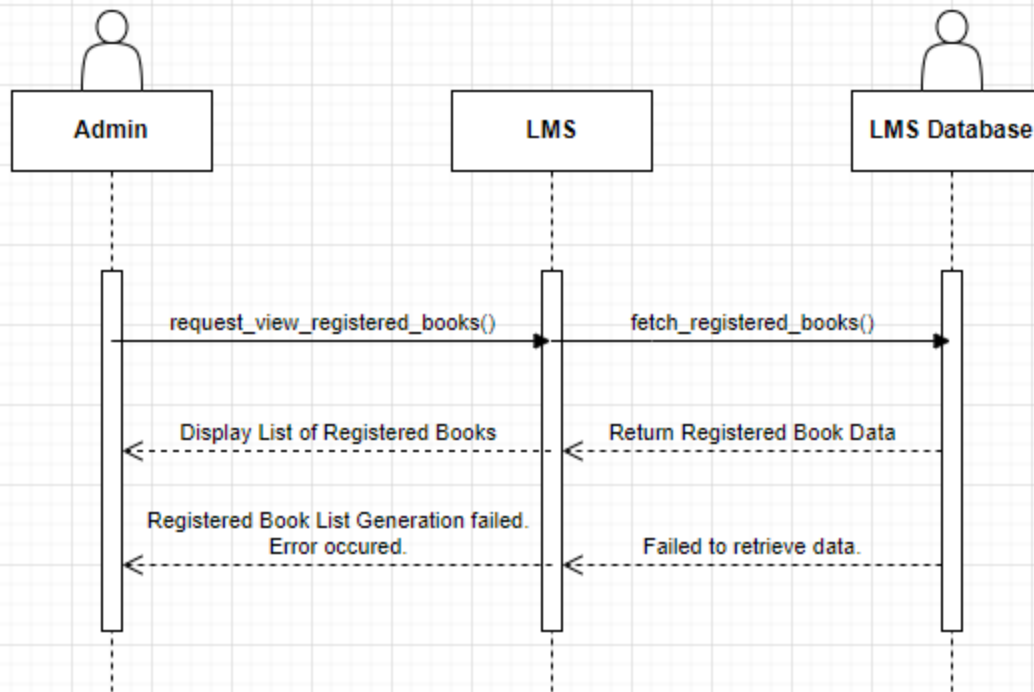 - WP2 or WP3 Recommended

Functional Requirements Specification

Requirement	Description	Priority
User Registration	Users should be able to sign up with a unique email and password.	**High**
User Login	Users should be able to log in with valid credentials.	**High**
Password Reset	Users should be able to change/reset passwords securely.	**Medium**
Admin Authentication	Admins should have separate login credentials with higher privileges.	**High**
View Profile	Users should be able to view their personal information.	**High**
Edit Profile	Users should be able to update their details.	**Medium**
Admin User Management	Admins should be able to view, update, and delete user accounts.	**High**
Add Books	Admins should be able to add books with relevant details.	**High**
Update Books	Admins should be able to update book details.	**High**
Delete Books	Admins should be able to remove books.	**Medium**
View Books	Users should be able to browse all available books.	**High**
Search Books	Users should be able to search for books by title, author, or category.	**High**
Add Authors	Admins should be able to add new authors.	**Medium**
Edit Authors	Admins should be able to update the author's details.	**Low**
Add Book Categories	Admins should be able to create and manage book categories.	**Medium**
Issue Books to Students	Admins should be able to assign books to students.	**High**
Track Issued Books	Maintain a record of issued books, including issue date.	**High**
Return Books	Admins should be able to mark books as returned.	**High**
Check Book Availability	Users should be able to check if a book is available.	**Medium**
User View	Users should be able to see the books they have borrowed.	**High**
Admin View	Admins should be able to see all the books issued and the students who borrowed them.	**High**
Overdue Notifications	Notify users and admins if a book is overdue.	**Medium**
Issuance/Return Notifications	Users should receive confirmations on book transactions.	**Low**
Manual Logout	Users should be able to log out securely.	**High**
Access Control	Regular users should not be able to access admin features.	**High**

System Sequence Diagram

User Views Issued Book

User	LMS	LMS Database

request_view_issued_books() → fetch_issued_books()

Display List of Issued Books ← Return Issued Book Data ←

Book list generation failed. Error occured. ← Failed to retrieve data. ←

Admin Views Registered Users

Admin   LMS   LMS Database

request_view_registered_books()

fetch_registered_books()

Display List of Registered Books

Return Registered Book Data

Registered Book List Generation failed.
Error occured.

Failed to retrieve data.

Activity Diagram
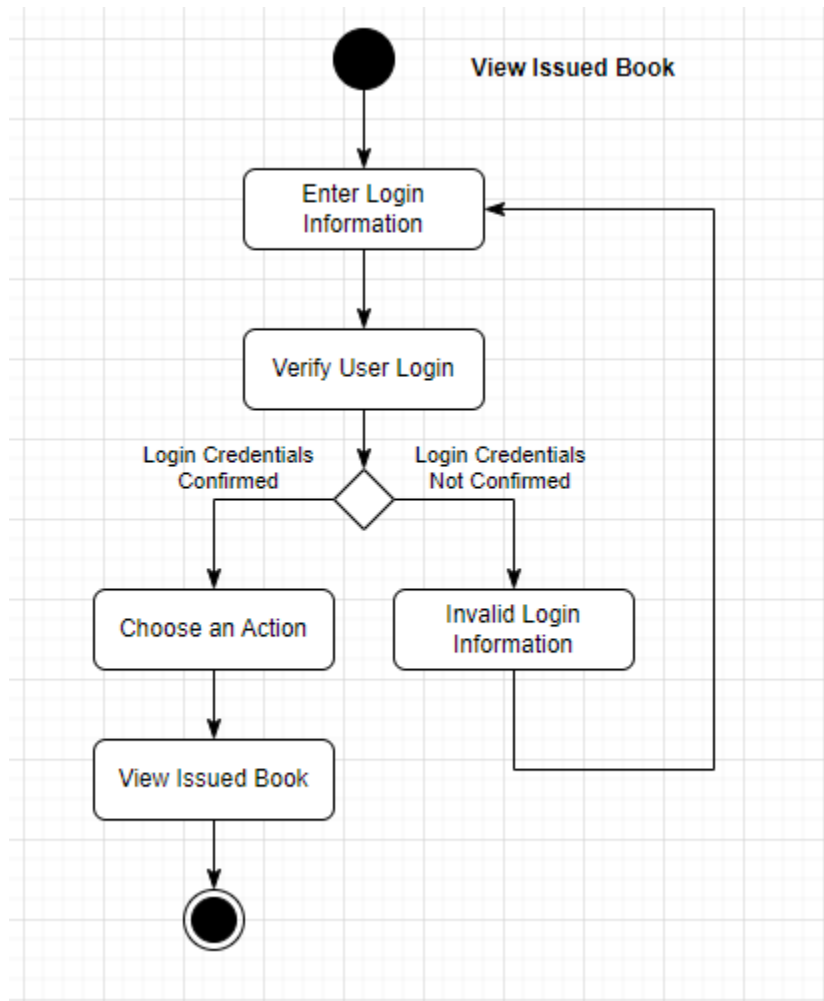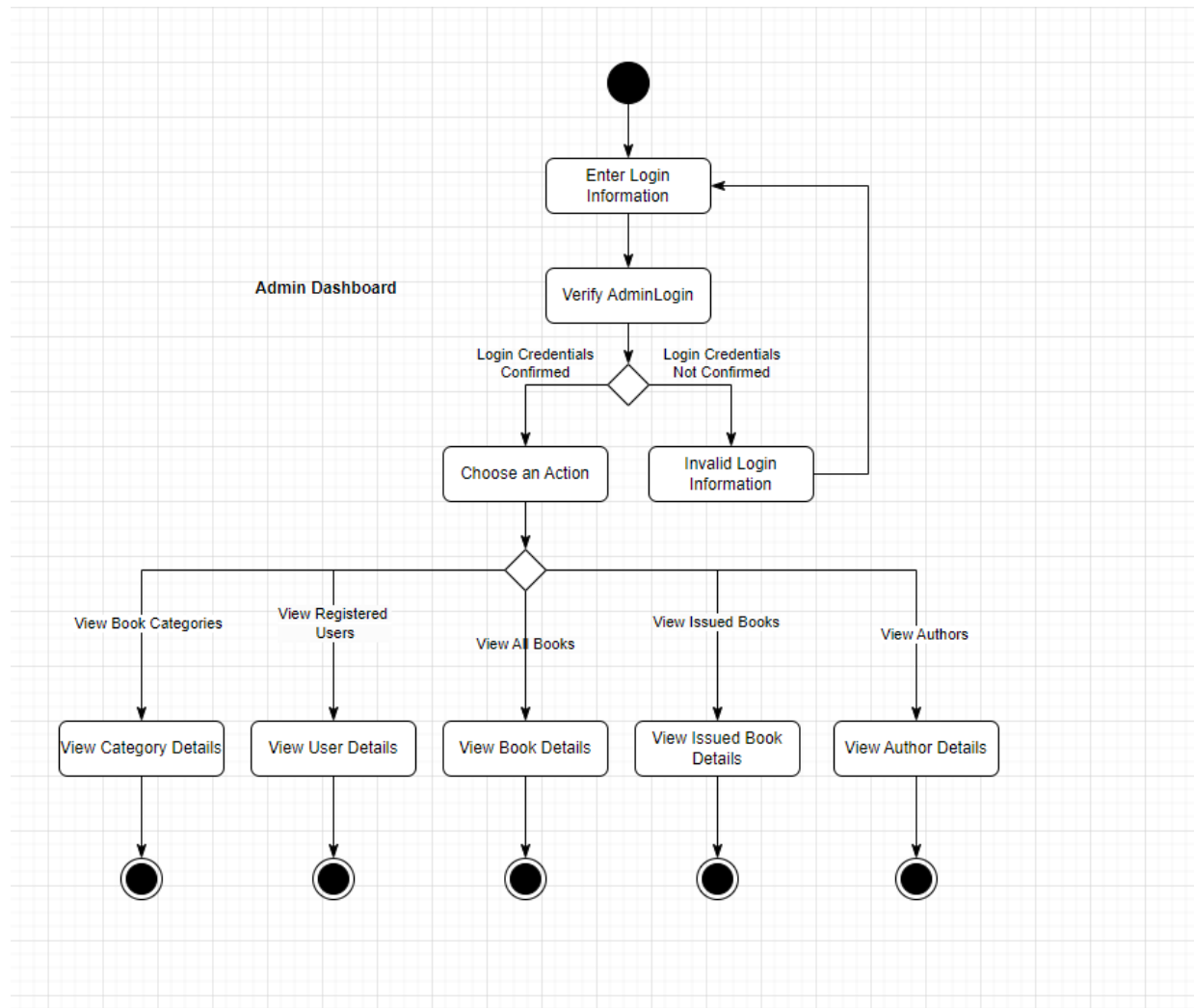
User Interface Specification

Library Management System   User Login   Admin Login   Signup

User Login Form

Email ID:

Password:

Login   Sign Up

Library Management System

User Registration Form

Full Name:

Email ID:

Password:

Mobile:

Address:

Register

Book Issued

No of book issued: 2

View Issued Books

Issued Books Details

Name	Author	Book Number
Data structure	D S Gupta	6541
The Shining	Stephen King	1111

Library Management System (LMS)　　　　　**Welcome: admin**　　　　　**Email: admin@gmail.com**　　　　　My Profile ▾　Logout

Dashboard　Books ▾　Category ▾　Authors ▾　Issue Book

Registered User
No. total Users: 6
View Registered Users

Total Book
No of books available: 8
View All Books

Book Categories
No of Book's Categories: 9
View Categories

No. of Authors
No of Authors: 10
View Authors

Book Issued
No of book issued: 10
View Issued Books

Library Management System (LMS) Welcome: admin Email: admin@gmail.com My Profile ▾ Logout

Dashboard Books ▾ Category ▾ Authors ▾ Issue Book

Registered Books

Name	Author	Price	Number
Software Engineering	M D Guptaa	3	4518
Data Structures	Chetan Bhagat	300	6541
Harry Potter and the Philosopher's Stone	J.K. Rowling	450	1234
1984	George Orwell	350	5678
The Hobbit	J.R.R. Tolkien	500	7890
The Shining	Stephen King	600	1111
Murder on the Orient Express	Agatha Christie	300	2222
The Adventures of Tom Sawyer	Mark Twain	400	3333

Library Management System (LMS) Welcome: admin Email: admin@gmail.com My Profile ▾ Logout

Dashboard Books ▾ Category ▾ Authors ▾ Issue Book

Manage Books

Name	Author	Category	ISBN No.	Price	Action
Software Engineering	102	1	4518	3	Edit Delete
Data Structures	103	2	6541	300	Edit Delete
Harry Potter and the Philosopher's Stone	105	2	1234	450	Edit Delete
1984	106	1	5678	350	Edit Delete
The Hobbit	108	4	7890	500	Edit Delete
The Shining	109	4	1111	600	Edit Delete
Murder on the Orient Express	110	5	2222	300	Edit Delete
The Adventures of Tom Sawyer	107	5	3333	400	Edit Delete

Project Plan

Most of the functionalities have been completed, but I still need to implement critical features into the program, such as editing and deleting authors, and a way to change passwords and edit user and admin details. Additionally, there are a lot of bugs in this program that need to be rectified as demonstrated in the demo. The technological stack has been finalized to match the scope of the project and my current capabilities. I was originally going to use Python for this project, but I realized it was outside of my expertise. While I am familiar with PHP, I'm not an expert at it so figuring out errors took longer than usual. That said, I will complete as much as I can before the final implementation.

- Week 9 – Implement author management and clean bugs.

  - Improve security features

- Week 10 – Clean up the UI/UX Design

- Week 11 – Have editing user details and change password features complete.

  - Have input and form validation.

- Week 12 – Work on notification feature.

- Week 13 – Pretty up the CSS

- Week 14 – Double check for bugs

- Week 15/16 – Prepare for final presentation.

References

I used this tutorial to help me understand the scope of the project, but the tutorial itself was incomplete. I had to determine bugs and errors for the rest of the project, as well as Sprucing the UI design to match my tastes. The tutorial used a previous version of Bootstrap, so I had to update the code to match Bootstrap 5.

Tutorial: https://www.geeksforgeeks.org/library-management-system/#er-model-of-library-management-system

Highlighted Changes

I did not receive feedback on any of the diagrams and project proposals at the time of submission, so I cannot highlight any changes in the proposal and project.