Real-time/Field-Based Research Project Report

On

A Web Application on Library Management System for Students

A dissertation submitted to the Jawaharlal Nehru Technological University, Hyderabad in partial fulfillment of the requirement for the award of degree of

Bachelor of Technology in Computer Science and Engineering

Submitted by

B.Hasini (22B81A05CS)

A.Shravya (22B81A05EC)

B.Greeshma (22B81A05CR)

Department of Computer Science and Engineering

CVR COLLEGE OF ENGINEERING

(An UGC Autonomous Institution, Affiliated to JNTUH, Accredited by NBA, and NAAC)

Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),

Ranga Reddy (Dist.) - 501510, Telangana State.

2023-24

CVR COLLEGE OF ENGINEERING

(An UGC Autonomous Institution, Affiliated to JNTUH,
Accredited by NBA, and NAAC)
Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),
Ranga Reddy (Dist.) - 501510, Telangana State.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled **A Web Application on Library Management System for Students** is being submitted by B.Hasini(22B81A05CS), A.Shravya(22B81A05EC), B.Greeshma(22B81A05CR) in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering,** during the academic year 2023-2024.

Professor Incharge **Professor and Head, CSE**

(Dr. N. Subhash Chandra) **(Dr. A. Vani Vasthala)**

DECLARATION

I hereby declare that this project report titled **"A Web Application on Library Management System for Students"** submitted to the Department of Computer Science and Engineering, CVR College of Engineering, is a record of original work done by me under the guidance of Dr.N.Subhash Chandra. The information and data given in the report is authentic to the best of my knowledge. This RFP report is not submitted to any other university or institution for the award of any degree or diploma or published at any time before.

B.Hasini (22B81A05CS)

A. Shravya(22B81A05EC)

B. Greeshma(22B81A05CR)

Date: 27/5/2024

Place: Hyderabad

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose encouragement and guidance has been a source of inspiration throughout the course ofthe project.

It is a great pleasure to convey our profound sense of gratitude to our principal **Dr. K. Ramamohan Reddy, Dr. A. Vani Vathsala**, Head of CSE Department, CVR Collegeof Engineering, for having been kind enough to arrange necessary facilities for executing the project in the college.

We deem it a pleasure to acknowledge our sense of gratitude to our project professor in charge **DR.N. Subhash Chandra** under whom we have carried out the project work. Her inclusive and objective guidance and timely advice encouraged us with constant flow of energy to continue the work.

We wish a deep sense of gratitude and heartfelt thanks to management for providing excellent lab facilities and tools. Finally, we thank all those whose guidance helped usin this regard. We place highest regards to our Parents, our Friends and Well-wishers who helped a lot in making the report of this project.

ABSTRACT

This project focuses on the development of a Library Management System (LMS) utilizing Java programming language. The system aims to automate various tasks involved in managing library resources, including book cataloguing, user management, borrowing, returning, and searching functionalities. Through the utilization of object-oriented programming principles and database management techniques, our LMS offers an intuitive and efficient solution to address the complexities of library operations, enhancing accessibility and usability for both librarians and its clients. This project represents a significant advancement in library management technology, leveraging the power of Java programming language to deliver a robust, user-friendly, and future-proof solution for libraries.

By implementing features such as a user-friendly interface, comprehensive search functionalities, and robust data management capabilities, this LMS aims to streamline the workflow of libraries, enabling librarians to efficiently manage their collections and customers to easily access and utilize library resources. Additionally, the system incorporates features such as fine calculation, notifications for due dates, and comprehensive reporting capabilities to facilitate effective library management. Overall, the project seeks to provide a scalable and adaptable solution to meet the evolving needs of modern libraries, contributing to the enhancement of library services and user experiences.

Ultimately, the Design and Development of this User-Friendly Library Management System seeks to enhance the efficiency, accessibility, and user experience of libraries, fostering a more enriching and fulfilling environment for both staff and patrons.

TABLE OF CONTENTS

ABBREVATIONS

1. LMS - Library Management System
2. HTML - Hypertext Markup Language
3. CSS - Cascading Style Sheet
4. MIS - Management Information System
5. SSO - Single Sign-On
6. API - Application Programming Interface
7. SDK - Software Development Kit
8. LMS - Learning Management Systems
9. UPS - Uninterruptible Power Supply
10. JSP - JavaServer Pages
11. IDE - Integrated Development Environment
12. SQL - Structured Query Language
13. RAM - Random Access Memory
14. CPU - Central Processing Unit
15. URL - Uniform Resource Locator
16. WLAN - Wireless Local Area Network
17. OS - Operating System
18. UI - User Interface
19. UX - User Experience

CHAPTER-1
INTRODUCTION

The chapter explains in brief about the Library Management Systems in general and some key points about this project such as the motivation, problem statement, project objectives and project report organization.

A **Library Management System** is a software application used by libraries to manage their resources efficiently. It helps librarians organize, catalogue, and circulate books, journals, media, and other materials available in the library. Let's take a look at an example of a library system to understand how it works.

Libraries have been integral to the dissemination of knowledge and information for centuries. However, in today's digital age, the traditional methods of managing libraries have become increasingly inefficient. The advent of technology offers new opportunities to streamline library operations, improve accessibility, and enhance user experiences. This project aims to develop a robust Library Management System (LMS) that leverages modern technology to address the evolving needs of libraries in the 21st century.

This project is related to Library Management System (LMS). The system is produced explicitly for education, such as universities or colleges where users are students and admin. HTML (Hypertext Markup Language) main programming language, JavaScript, and CSS (Cascading Style Sheet). Use Tomcat as a web server to connect databases, MySQL as Backend. This system can improve management and maintenance levels, making it easier for the librarian to manage incoming data in the library. Most of all, the library is still doing manual management, which requires a higher workforce. With this system, the workforce can be reduced and provide better online data storage places. This project stands as a definition for transforming the traditional libraries to modern library management using the technology.

Libraries, steeped in history as bastions of knowledge, are undergoing a profound transformation propelled by the digital revolution. This Library Management System (LMS) project represents a pivotal step towards modernizing library operations, aligning them with the fast-paced demands of contemporary academia. By harnessing the power of HTML, JavaScript, CSS, Tomcat, and MySQL, the system not only optimizes resource allocation but also enhances user engagement and accessibility. The traditional manual methods, labor-intensive and prone to human error, are being superseded by automated processes that streamline cataloging, circulation, and maintenance tasks. With this innovative system, librarians can devote more time to curating collections and providing personalized support to patrons, rather than being bogged down by administrative burdens. Moreover, by facilitating online data storage and management, the project heralds a new era of efficiency and scalability in library operations. As education institutions embrace digital transformation, this project serves as a beacon of progress, symbolizing the evolution from antiquated library practices to a dynamic, tech-driven future. In essence, it's not just about managing libraries—it's about empowering knowledge dissemination in the digital age and fostering a vibrant learning community.

1.1 MOTIVATION

The present Study offers intervention of MIS (Management Information System) to the conventional Library Management. Libraries, as centers of learning are experiencing unprecedented rates of change, both from internal and external environment. The new library environment incorporates a changing user population, technology enhancements, transformation of the scholarly communication system, digital information, new approaches to management and a renewed commitment to planning and assessment throughout the organization.

One of the primary motivations behind developing our library management system is to cater to the needs of colleges that prioritize monitoring students' book borrowing activities. However, Librarians as information managers have been slow to keep pace with this change. Our web application development in the library management domain is multifacetedly motivated by issues and needs that ensure betterment of the students

experience and administration of work. Foremost among these is the call for a platform that is able to satisfy colleges on their need for effective monitoring of students' borrowing activities. This means that accessibility is highly enhanced, as the web application empowers students to connect to the library's resources from anywhere that has a network connection.

Digital transformation not only saves one's time but also enhances the visibility of resources, enabling students to seamlessly browse through the library catalog and check on real-time availability. In addition, features such as notifications and reminders contribute to the enhancement of organizational skills and accountability, ensuring students are up to date with their borrowing responsibilities. Integration of the system into existing platforms of learning enriches the experience and sense of community among students. Furthermore, data gathered through the web application offers valuable insights for making informed decisions in a way that shapes future strategies of collection development and resourcing.

The web has changed every aspect of life. The most visible change has occurred in the size, rate of change and speed of information availability and delivery. The increasing expectation of users, changes in library usage patterns brought about by, increasing use of information and communication technologies, information seeking strategy – including preference for online searching - more sophistication in the new interfaces of knowledge delivery has resulted in automating library processes, and services which underscore the adoption of library management system. Library automation which implies the application of information technologies to library operations and services with little supervision by people in developing countries is limited and have been mainly adopted in the academic libraries.

1.2 PROBLEM STATEMENT

Many libraries struggle with outdated and inefficient management systems. These old systems make it hard for librarians to manage resources and provide good service to library users. The existing systems often have confusing interfaces, limited features, and weak security. Libraries need a modern, easy-to-use Library Management System that uses advanced technology like Java programming. This would automate tasks, streamline workflows, improve accessibility, and protect library resources and user data. This project aims to create a scalable, user-friendly, and feature-rich LMS that meets the changing needs of modern libraries. The goal is to have a more efficient, accessible, and rewarding library experience for both staff and library users.

1.3 PROJECT OBJECTIVES

The objectives of our Library Management System (LMS) project encompass various aspects aimed at improving library operations, enhancing user experiences, and leveraging technology to meet the evolving needs of libraries. These objectives are outlined as follows:

Streamlining Catalog Management Processes:

The primary goal of our LMS is to streamline catalog management processes to ensure efficient organization and accessibility of library resources. This includes:

- Implementing a user-friendly interface for librarians to easily add, delete and see catalog records.
- Enabling batch processing and bulk imports for expedited cataloging of large collections.

Facilitating User Registration and Authentication:

Another key objective is to facilitate seamless user registration and authentication processes to enhance patron engagement and access to library services. This involves:

- Providing intuitive registration forms for patrons to create accounts with relevant profile information.
- Implementing secure authentication mechanisms such as username/password, email verification.
- Supporting integration with single sign-on (SSO) systems to enable users to access the library system using existing institutional credentials.

Supporting Integration and Customization:

Our LMS is designed to support seamless integration with existing library systems, as well as provide flexibility for customization to meet the specific needs and requirements of different libraries. This includes:

- Offering APIs (Application Programming Interfaces) and web services for integration with external systems such as library catalogs, digital repositories, learning management systems (LMS), and authentication systems.
- Providing software development kits (SDKs) and documentation to facilitate customizations and extensions by library developers and administrators.
- Collaborating with libraries and institutions to gather feedback and requirements for future enhancements and feature additions, ensuring continuous improvement and alignment with industry standards and best practices.

1.4 PROJECT REPORT ORGANIZATION

1. Cover Page: Include the title of the project, the name of the organization or institution, the names of team members, and the date.

2. Table of Contents: List all the sections and subsections of the report with corresponding page numbers for easy navigation.

3. Executive Summary: Provide a concise overview of the project, including its objectives, scope, methodology, key findings, and recommendations.

4. Introduction: Introduce the background and context of the project, outlining the need for a modern Library Management System and the objectives of the project.

5. Literature Review: Review existing literature, research, and best practices related to library management systems, including historical developments, current trends, technologies, challenges, and opportunities.

6. Project Objectives: Detail the specific objectives of the project, outlining the goals and outcomes to be achieved in developing the Library Management System.

7. System Architecture: Describe the high-level architecture of the Library Management System, including components, modules, and their interactions.

8. Requirements Specification: Outline the functional and non-functional requirements of the system, including user requirements, system features, usability criteria, performance metrics, and constraints.

9. Design and Implementation: Describe the design principles, methodologies, and technologies used in developing the Library Management System. Provide an overview of the system's architecture, database schema, user interface design, and implementation details of key features.

10. Testing and Quality Assurance: Discuss the testing approach, strategies, and techniques used to ensure the quality, reliability, and performance of the Library Management System.

Present test cases, test results, and any issues identified during the testing phase, along with resolutions.

11. Deployment and Maintenance: Explain the deployment process, including installation, configuration, and rollout of the Library Management System in a production environment.

Discuss ongoing maintenance activities, support mechanisms, and plans for future enhancements and updates.

12. Conclusion: Summarize the key findings, achievements, and lessons learned from the project.

Reflect on the overall success of the project in meeting its objectives and delivering value to stakeholders.

13. References: List all the sources, references, and citations used in the report, following a consistent citation style (e.g., APA, MLA).

14. Appendices: Include any additional supplementary materials, such as detailed technical documentation, code samples, user manuals, or project documentation templates.

CHAPTER-2

LITERATURE REVIEW

This chapter talks about the existing LMS's available in the market and its limitations.

2.1 EXISTING WORK

Library Management Systems have evolved significantly, pivotal in automating library operations, improving user experiences, and facilitating data-driven decision-making. This literature review highlighted the benefits, key features, challenges, and emerging trends in LMS. It is evident that LMS are essential tools for modern libraries, and their continuous development will contribute to the growth and effectiveness of library services in the digital age. The data relating to the affairs of the user and the library is critical, one of which is the proof of borrowing and transactions of the user. Manual management will cause discrepancies during data encoding.

The user's transaction history is too numerous and needs to be stored over a long period, requiring a lot of files for storage. With this system, it can increase the level of data management. Furthermore, the librarian needs to update the report at any given time, which takes a while to research the data for the report. The system provides reports automatically and faster. This system helps librarians handle a library's day-to-day transactions. This also allows librarians to keep records of borrowed, returned and non-refundable books. On the other hand, for students, they can easily find the availability of the books they need. The importance of the project is to save the user time to do the borrowing and return process of books or any transactions. In addition, if any issues regarding the user can be identified and management can take action. This system can make library management more efficient and systematic.

Increase reporting and monitoring rates. Updated records with the automatic library management system allow dynamic reporting and surveillance. This system enables the admin to know more accurate data and avoid any negligence regarding data storage.There are LMS in market where users benefit from personalized recommendations tailored to their

reading habits, enhancing their library experience. Built on a cloud-based infrastructure, our LMS offers scalability and flexibility, enabling libraries to adapt seamlessly to evolving needs without substantial infrastructure investments. With a user-centric design, our interface ensures accessibility for all demographics.

Interoperability standards facilitate seamless integration with external systems, promoting resource sharing and collaboration among libraries. Moreover, our system champions digital preservation by digitizing and archiving cultural heritage materials. Advanced cybersecurity measures safeguard user data, ensuring privacy and confidentiality. Cost-effectiveness is paramount, with our solution minimizing operational expenses while maximizing efficiency. Continuous innovation drives our commitment to staying ahead of emerging trends and challenges, providing libraries with state-of-the-art tools and services.

The process phase of the Waterfall model is used to develop this system and divided separately; the result of one step will act as an input in the next phase sequentially. In addition, the Waterfall model will help plan and schedule system development. The development process moves from planning, analyzing, designing, implementing, testing and maintenance. Library Management Systems have evolved significantly, pivotal in automating library operations, improving user experiences, and facilitating data-driven decision-making. This literature review highlighted the benefits, key features, challenges, and emerging trends in LMS. It is evident that LMS are essential tools for modern libraries, and their continuous development will contribute to the growth and effectiveness of library services in the digital age.

2.2 LIMITATIONS OF EXISTING WORK

1. Technology Stack Dependency: Many existing LMS solutions may be built on technologies that are less flexible or scalable compared to your Java-based system. This dependency on specific technology stacks could limit customization options or hinder integration with other systems and services.
2. Usability and Interface Design: Some LMS solutions may lack a user-friendly interface or intuitive design, making it challenging for librarians and users to

navigate the system efficiently. Improving usability and interface design could enhance user satisfaction and productivity within the library environment.

3. Limited Functionality: Certain LMS solutions may offer only basic functionalities for book cataloguing, user management, and circulation, without incorporating advanced features such as fine calculation, notifications, or comprehensive reporting capabilities. Your project's focus on comprehensive functionality could address these limitations and provide a more robust solution for library management.
4. 4.Scalability Issues: Existing LMS solutions may struggle to scale effectively to accommodate the growing needs of libraries with expanding collections and user bases. By leveraging object-oriented programming principles and database management techniques, your project aims to offer a scalable and adaptable solution capable of meeting the evolving demands of modern libraries.
5. Integration Complexity: Integrating the library management system with other institutional systems, such as student information systems or learning management systems, can be challenging due to compatibility issues and disparate data formats.
6. User Training Needs: Transitioning from manual or outdated systems to a new library management system often requires extensive training for librarians and staff to effectively utilize all features and functionalities.
7. Accessibility and Inclusivity: Ensuring that the library management system is accessible to users with disabilities and meets diverse user needs, including multilingual support and interface customization options, presents ongoing challenges.
8. Security Concerns: Safeguarding sensitive user data, including personal information and borrowing histories, against unauthorized access or data breaches is a paramount concern for library management systems, requiring robust security measures and compliance with data protection regulations.
9. Cost Considerations: Implementing and maintaining a comprehensive library management system can incur significant upfront and ongoing costs, including software licenses, hardware infrastructure, and staff training, which may pose financial challenges for some institutions.

10. Remote Access and Connectivity: Providing seamless access to library resources and services for remote users, including off-campus students and faculty, requires robust connectivity solutions and may pose technical challenges in areas with limited internet access or infrastructure.
11. Customization and Adaptability: Meeting the unique requirements and workflows of different types of libraries, such as academic, public, or special libraries, while maintaining a standardized and adaptable system architecture can be a complex balancing act.

CHAPTER-3

REQUIREMENTS ANALYSIS

This chapter covers all the requirements essential to complete this project which includes Software Requirements, Hardware Requirements and User Requirements.

3.1 SOFTWARE REQUIREMENTS

- Operating System: Any modern operating systems will suffice, including Windows, macOS, or Linux distributions like Ubuntu.
- Development Tools: Eclipse IDE, Notepad
- Programming Languages and Frameworks:
  -HTML
  -CSS
  -Java Programming Language
  -JavaScript
  -JSP
- Database: mySQL Workbench
- Dependencies and Libraries: JRE System Library, Jakarta, mySQL connector
- Runtime environment: Apache Tomcat 10.0

3.2 HARDWARE REQUIREMENTS

- Computer: A computer capable of running your chosen development tools and IDE comfortably. This could be a desktop or laptop computer with sufficient RAM, CPU, and storage.
- Internet Connection: While not strictly necessary for all stages of development, having a stable internet connection can be beneficial for accessing online resources, documentation, and libraries.

- Devices for Testing (if applicable): If your project involves mobile app development or requires compatibility testing on different devices, you may need access to smartphones, tablets, or other devices.
- Backup Storage: It's always a good idea to have backup storage solutions in place to prevent data loss.
- UPS (Uninterruptible Power Supply): To ensure continuous operation of critical systems in the event of power outages or fluctuations, having a UPS can provide temporary power backup to servers, networking equipment, and other essential hardware.

3.3 USER REQUIREMENTS

- This proposed system maintains record of library management system such as Student issue book, return books etc.
- User friendly and easy to access also get the information which admin requires.
- Allows User to check new books present in the library.
- Accessibility and Usability: Design an intuitive and accessible user interface that is easy to navigate for users of all abilities. Ensure compatibility with different devices and screen sizes to accommodate various users.
- Define different user roles such as user, admin.
- Search and Filter Functionality: Implement robust search and filter options that allow users to quickly find books based on various criteria such as title, author, genre, and availability status.
- Integration with Learning Management Systems (LMS): Integrate the library management system with the institution's learning management system (LMS) to seamlessly access library resources within the context of academic courses and assignments.
- Privacy and Data Security: Implement stringent privacy measures to protect users' personal information and borrowing history, complying with relevant data protection regulations to maintain user trust and confidentiality.

CHAPTER-4

SYSTEM DESIGN

This chapter conveys the interactions and functionalities of the system with the use of clear and organized flowchart diagrams.

4.1 PROPOSED METHODS ARCHITECTURE

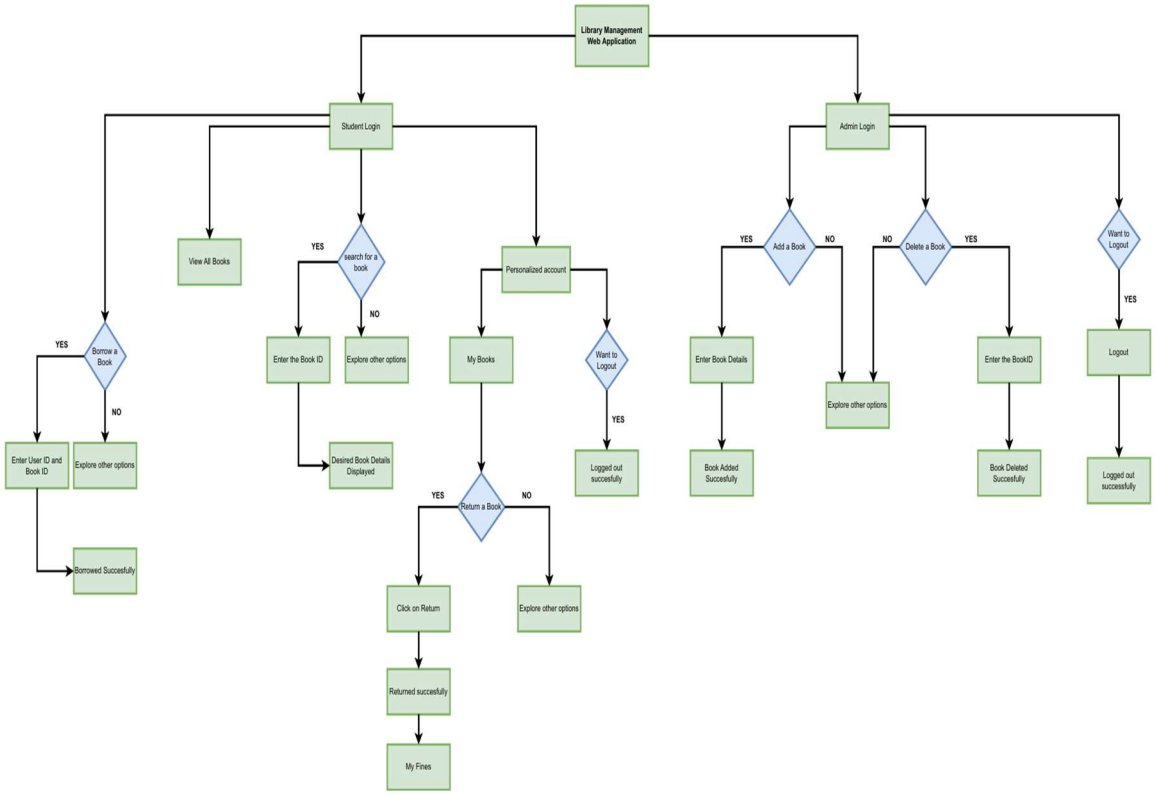


Fig 4.1 System Architecture of Stud

The figure 4.1 is a flowchart representation of this project's architecture which shows all the functionalities available at various levels of project execution.

4.2 PROPOSED METHODS

Understanding Library Needs: We start by engaging in discussions with librarians, stakeholders, and potential users to deeply understand the challenges they face and the features they require in an ideal LMS.

1. Crafting a Flexible Architecture: Our team employs object-oriented programming principles to design a system architecture that is not only scalable and adaptable but also easy to maintain and extend as library requirements evolve over time.

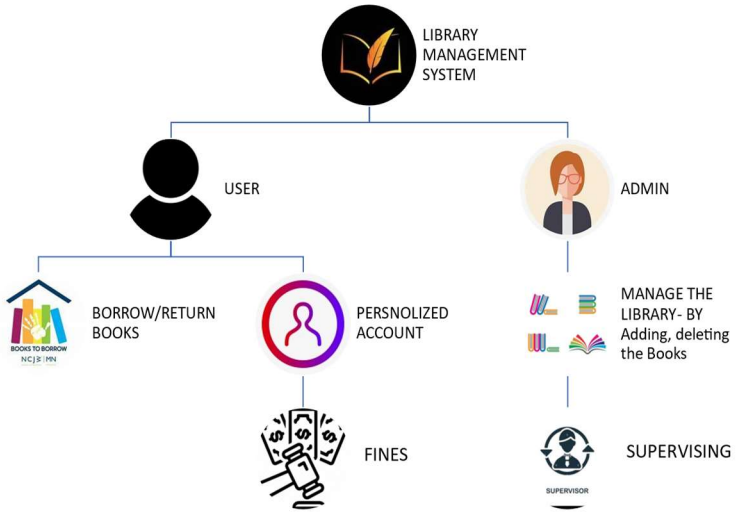


Fig 4.2.1 Project Flow of Stud

The figure 4.2.1 shows the project flow of the LMS Stud starting from the login page to personalized account system.

2. Effective Data Management: Our team employs robust database management techniques to ensure efficient storage, retrieval, and management of library data, guaranteeing seamless access to information when needed.

3. Designing an Intuitive Interface: We place great emphasis on user experience, designing an intuitive and user-friendly interface that makes navigating the LMS a breeze. Through iterative design cycles and user feedback sessions, we refine the interface to ensure optimal usability.

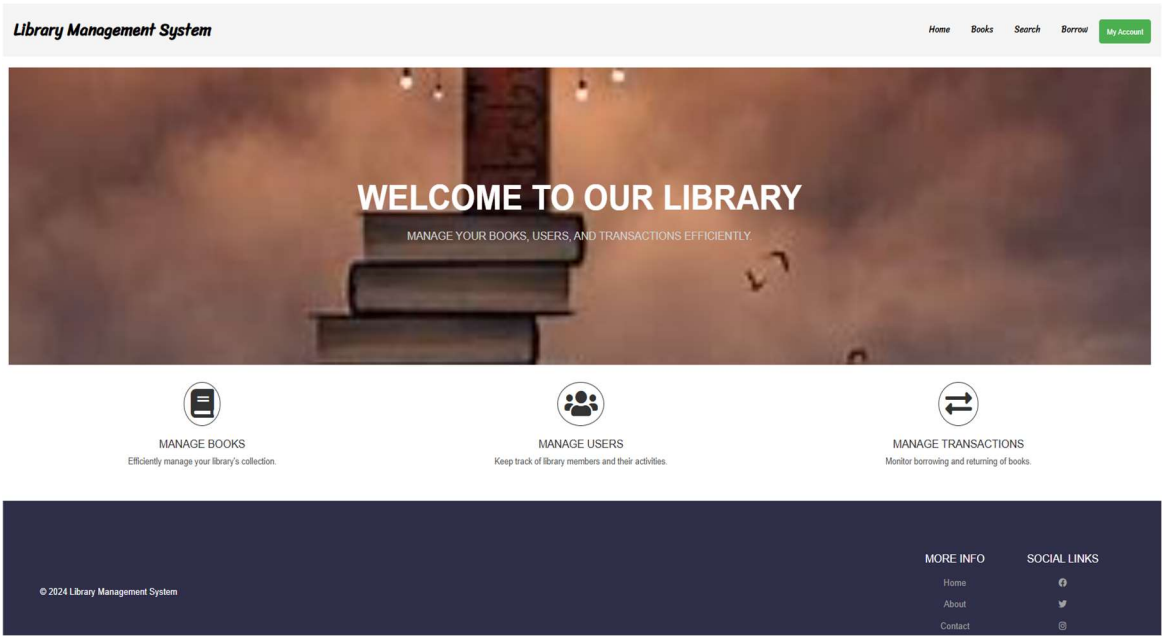


Fig 4.2.2 Users Homepage

The figure 4.2.2 give a glimpse of UI of the users homepage.

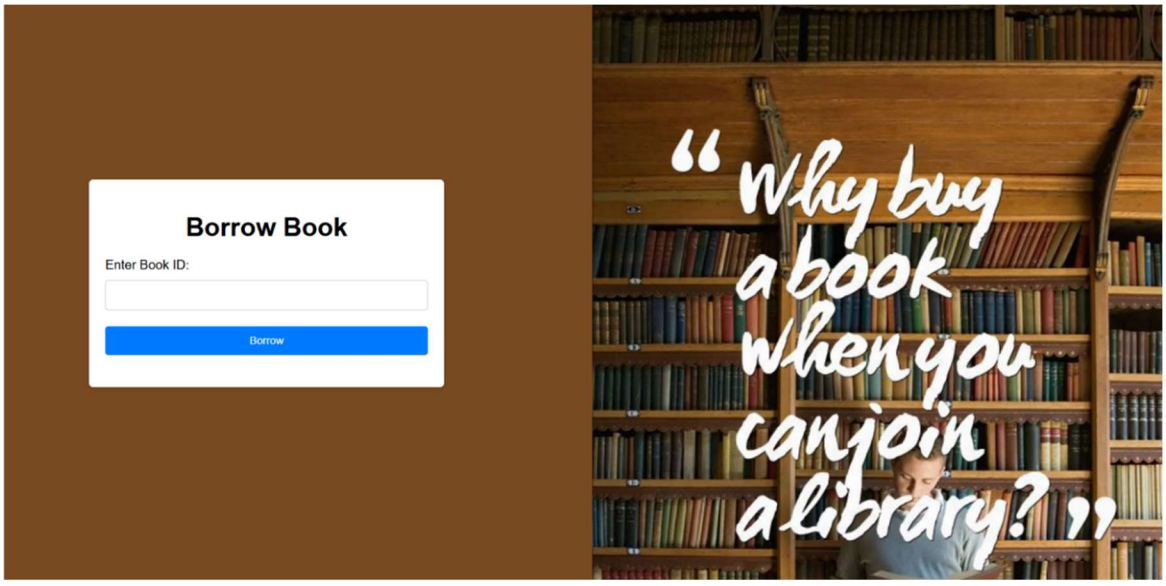


Fig 4.2.3 Borrowing a Book

The figure 4.2.3 shows a UI picture of the user functionality-book borrowing.

4. Set up Tomcat Server:

    - Developed JSP pages for dynamic content generation, along with HTML and CSS for frontend presentation.
    - Utilized Jakarta EE (formerly Java EE) for backend development, incorporating servlets and for database interactions with MySQL.
    - Configured and deployed the application on Apache Tomcat server for hosting both frontend and backend components.
    - Ensured seamless integration between JSP-based frontend and Jakarta EE backend, facilitating efficient communication with the MySQL database.

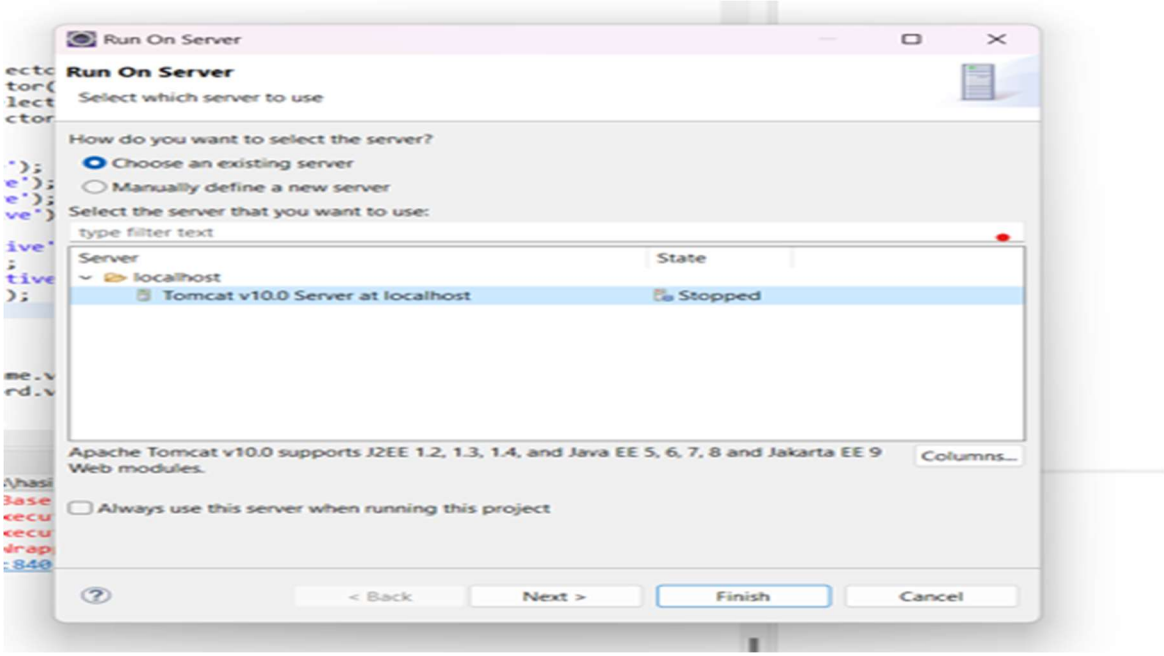


Fig 4.2.4 Running on Tomcat Server 10.0

The figure 4.2.4 illustrates the running of the project on Tomcat Server as localhost after the integration.

5. Thorough Testing and Deployment: Before deployment, we conduct rigorous testing, including unit testing, integration testing, and user acceptance testing. Following successful testing, we deploy the system and provide comprehensive training and ongoing support to ensure.

6. Implementing Core Functionality: Core functionalities such as book cataloguing, user management, circulation processes, fine calculation and reporting capabilities are implemented with precision and attention to detail, ensuring a comprehensive solution.

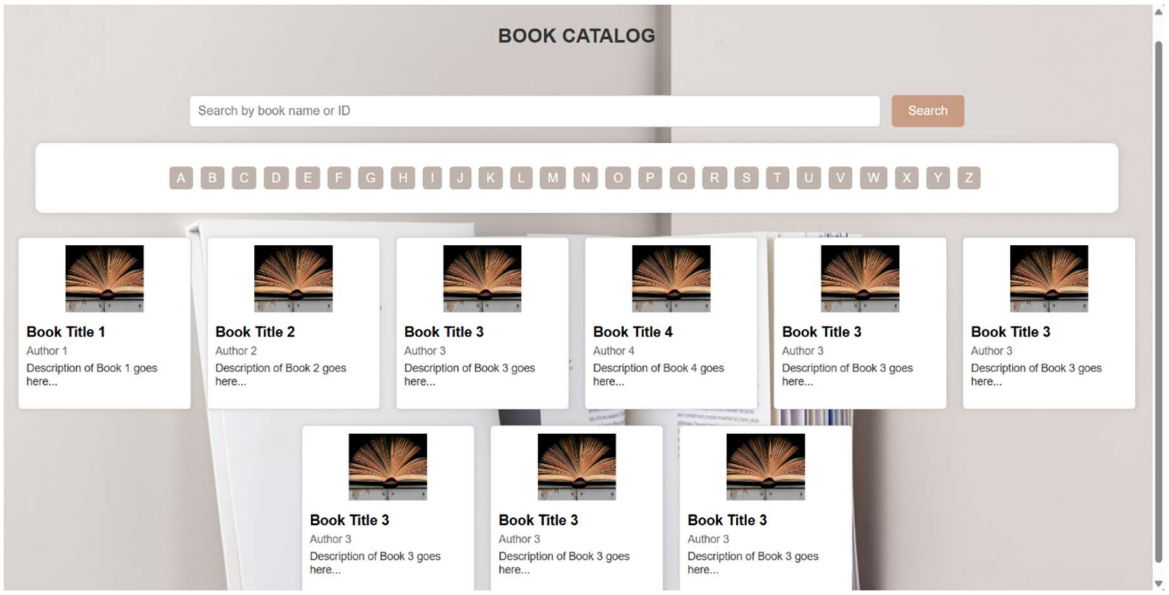


Fig 4.2.5 Book Catalog

The figure 4.2.5 depicts one of the user functionality in this project-searching for a specific book based on its book name or ID from the catalogue provided.

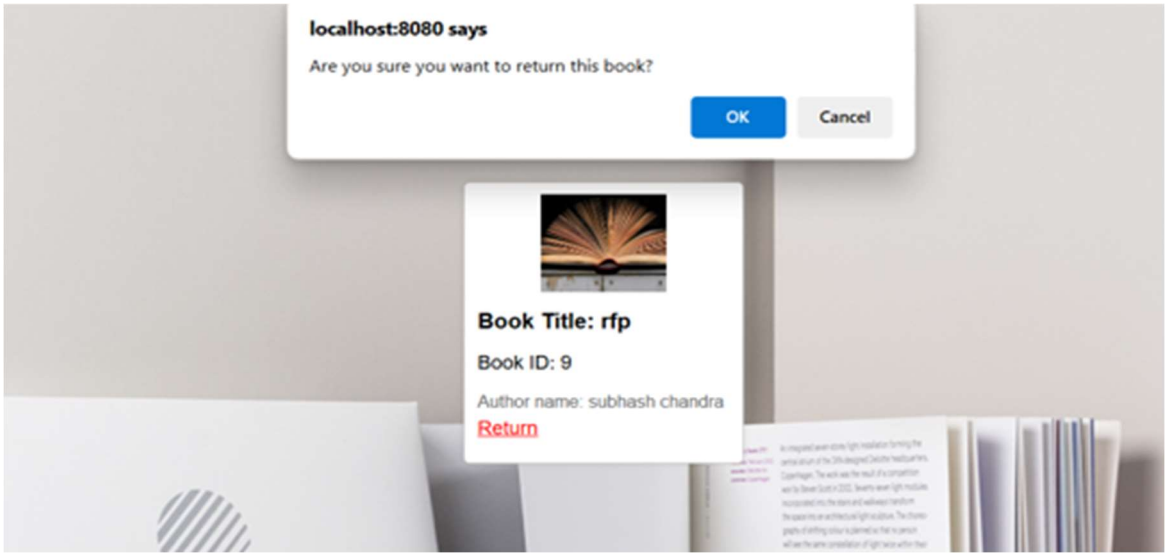


Fig 4.2.6 Returning Book

The figure 4.2.6 depicts a user functionality i.e returning a book which is implemented only on the user's personally borrowed book collection.

4.3 USE CASE DIAGRAM

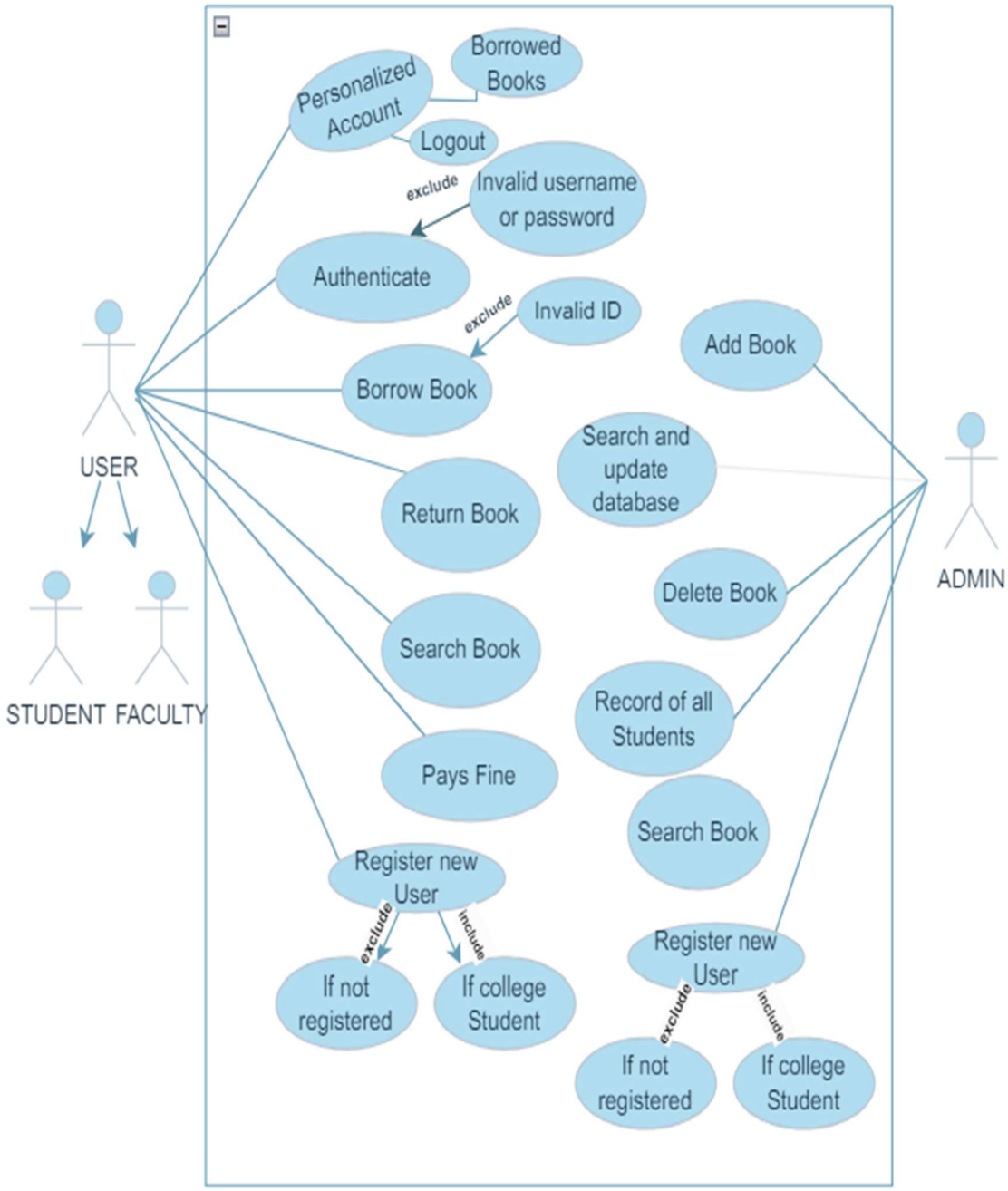


Fig 4.3 Library Management System-STUD

The figure 4.3 illustrates the Use Case diagram for the Library Management System-STUD.

4.4 DATASETS AND TECHNOLOGIES

The dataset used by this library management system consists of three major tables: users, books, and borrowings. Each table records specific information related to library users, books available, and borrowing transactions. The dataset is designed to manage and arrange library resources efficiently, allow user interaction efficiently, and maintain precise records on borrowings.

Overview of Tables

Users Table

rollno	firstname	lastname	username	password	email	usertype
05cb	akshaya	mupp	akshaya	akshaya	akshaya	student
05ce	architha	kallu	archu	arch	architha kallu@gn	student
05cf	ashritha	adluri	ashu	ash	ash@gmail.com	student
05cr	greeshma	reddy	greeshma	greeshma	greeshmab@gma	student
05cs	hasini	rao	hasinirao	hasini	hasinirao69@gm	student
05ec	shravya	adarapu	shrav	shravya	Shravya@gmail.c	student
514	pavani	mam	pavani	pavani	pavanimam@gm	admin
634	subhash	chandra	subhash chandra	subhash	subhash@gmail.c	admin
870	sampurnima	mam	sampurnima	adsj	sampurnima@gm	admin

Fig 4.4.1 Users Table

The figure 4.4.1 depicts the database table name User to store the user details.

- **Primary Key:** rollno
- **Columns:**
    1. **rollno:** Unique identifier for each user.
    2. **firstname:** First name of the user.
    3. **lastname:** Last name of the user.
    4. **username:** Username for logging into the system.
    5. **password:** Password for authentication.
    6. **email:** Email address of the user.
    7. **usertype:** Identifies whether the user is a student or faculty.

Books Table:

id	name	author	copies
1	Operating Systems	Galvin	5
10	M1	saritha	11
2	Caed	kelvin	10
3	software engineering	morgan	5
4	discrete mathematics	kenneth	10
5	dbms	shake	67
6	adsj	sampurnima	5
7	java	Nageswara Rao	2
8	coa	William Stallings	8
9	rfp	subhash chandra	6

Fig 4.4.2 Books Table

The figure 4.4.2 depicts the database table name Books to store the book details.

- **Primary Key:** book_id
- **Columns:**
    1. **book_id:** Unique identifier for each book.
    2. **book_name:** Name or title of the book.
    3. **author:** Author of the book.
    4. **no_of_copies:** Number of copies available for each book.

Borrowings Table:

borrowing_id	userid	bookid	borrow_date
10	05cs	9	24-05-2024
11	05ce	10	24-05-2024
12	05cf	8	24-05-2024
17	05cr	2	24-05-2024

Fig 4.4.3 Borrowings Table

The figure 4.4.3 depicts the database table named Borrowings which contains the book borrowing details.

- **Primary Key:** borrowing_id
- **Columns:**
    1. **borrowing_id:** Unique identifier for each borrowing transaction.
    2. **user_id:** Foreign key referencing the rollno of the user borrowing the book.
    3. **book_id:** Foreign key referencing the book_id of the borrowed book.
    4. **borrow_date:** Date when the book was borrowed.

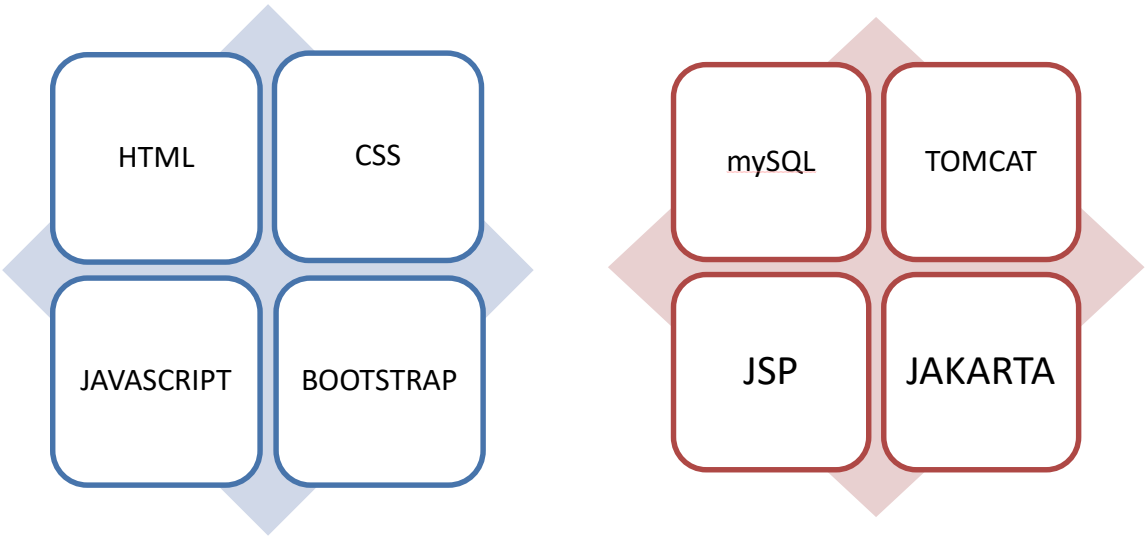
Relationships

The users table stores information on the library users, including their personal information and user type as a student or faculty. The rollno is the primary key that uniquely identifies each user.

The books table stores information on the books available in the library, including the book name, author, and the number of copies. Each book has a unique book_id as a primary key.

The borrowings table stores the borrowing transactions between users and books. It includes information on the user, user_id referencing rollno; the book borrowed, book_id; and the date of borrowing. The borrowing_id uniquely identifies each borrowing instance.

TECHNOLOGIES



Frontend:

- **JavaScript:** Used for client-side scripting to enhance interactivity and dynamic behaviour on web pages.
- **HTML:** Utilized for structuring the content and layout of web pages.
- **CSS:** Employed for styling and designing the appearance of web pages, enhancing user experience.
- **Bootstrap:** Integrated for front-end development to ensure responsive design and streamline UI development with pre-built components and styles.

Backend:

- **MySQL:** Chosen as the relational database management system (RDBMS) to store and manage data efficiently.
- **Jakarta (Java EE):** Utilized for backend development to build scalable and robust enterprise applications.
- **Tomcat**: Employed as the web server and servlet container to deploy and run Java-based web applications, providing a reliable environment for hosting dynamic content.
- **JSP (JavaServer Pages):** Used for server-side dynamic content generation, enabling the integration of Java code into HTML for building dynamic web pages.
- **MySQL Connector:** Integrated to establish a connection between the Java application and MySQL database, facilitating data retrieval, manipulation, and storage operations.

CHAPTER-5

IMPLEMENTATION

This chapter visualizes the implementation of all the key features of the Library Management System-Stud. It also covers all the testing steps performed during the course of this project execution along with the validation of all the test cases.

5.1 FRONT PAGE SCREENSHOT

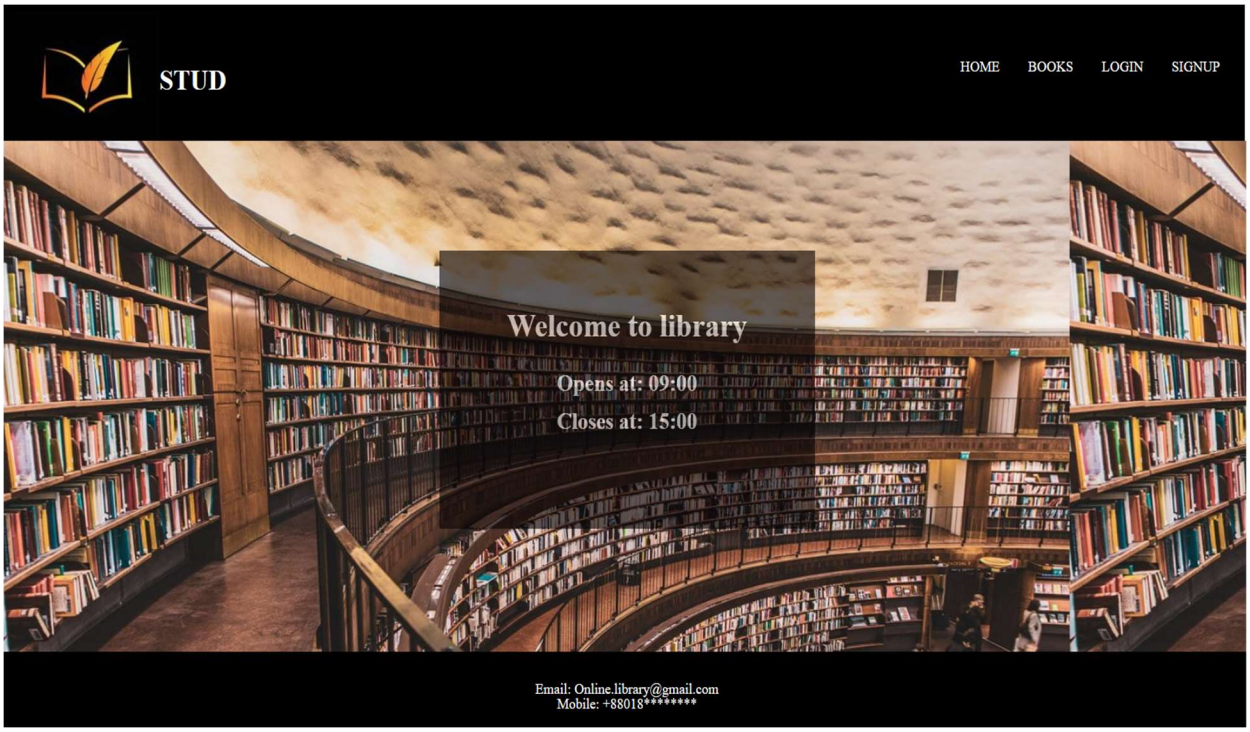


Fig 5.1 Frontpage Screenshot

The figure 5.1 is a UI picture of the opening page of the project STUD which contains options like “Login” and “signup” and “books” to view all the books.

5.2 RESULTS AND DISCUSSIONS

- **Performance**
    1. Prompt response times and efficient database operations.
    2. Optimization strategies implemented for scalability.
    3. Performance testing yielded commendable results.

- **User Experience**
    1. Positive feedback on intuitive design and ease of use.
    2. Interface praised for simplicity and clarity.
    3. Minor suggestions for improvement noted and addressed.

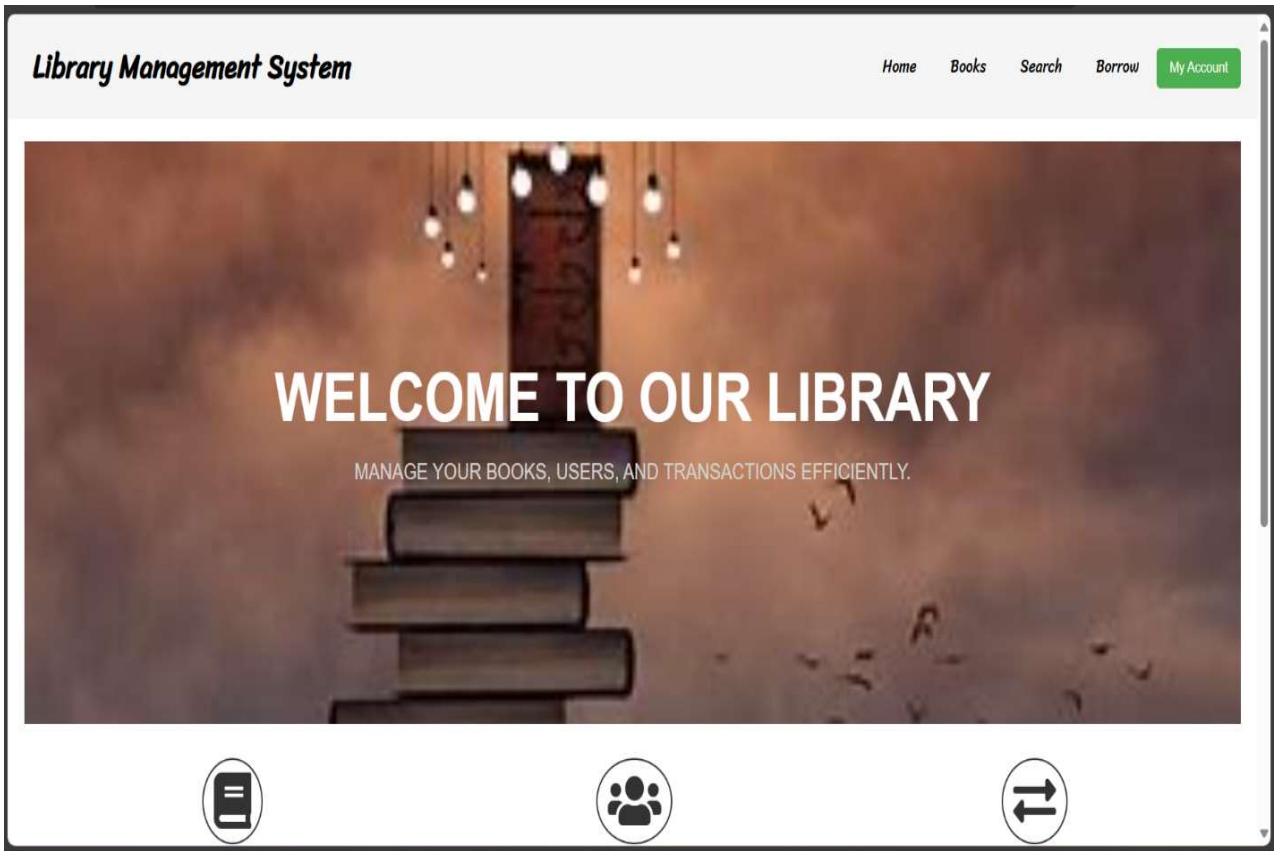


Fig 5.2.1 Student Homepage

The figure 5.2.1 is a UI of the page displayed after student login where the navigation bar contains user functionalities such as “Books” to view book catalogue, “Search” to search for a specific book based on book name, “Borrow”-designed for student users to borrow books and a personalised account to monitor your fines and return books.

- **Reliability and Robustness**
    1. System demonstrated robustness in handling errors and edge cases.
    2. Measures like data validation and exception handling ensure stability.
    3. Continuous monitoring and maintenance routines in place.

- **Comparative Analysis**
    1. System's unique features and advantages highlighted.
    2. User-friendly interface, performance efficiency, and security measures set it apart.
    3. Areas for improvement identified through comparison with existing systems.
- **Future Enhancements**
    1. Potential enhancements include advanced search algorithms and personalized recommendations.
    2. Integration with external databases and emerging technologies planned.

- **System Functionality**
    1. Comprehensive range of functionalities for streamlined library operations.
    2. Tasks include book searches, account management, borrowing, and returns.
    3. Interface designed for ease of navigation and accessibility.

STUDENT NAVIGATIONS

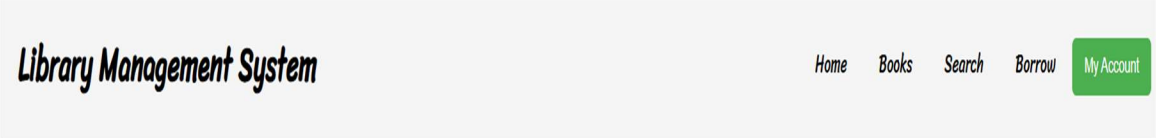


Fig 5.2.2 Student Navigation Bar

The figure 5.2.2 shows the navigation bar which appears in the student home page with student specific functionalities.

ADMIN NAVIGATIONS

Fig 5.2.3 Admin Navigation Bar

The figure 5.2.3 shows the navigation bar which appears in the admin home page with admin specific functionalities.

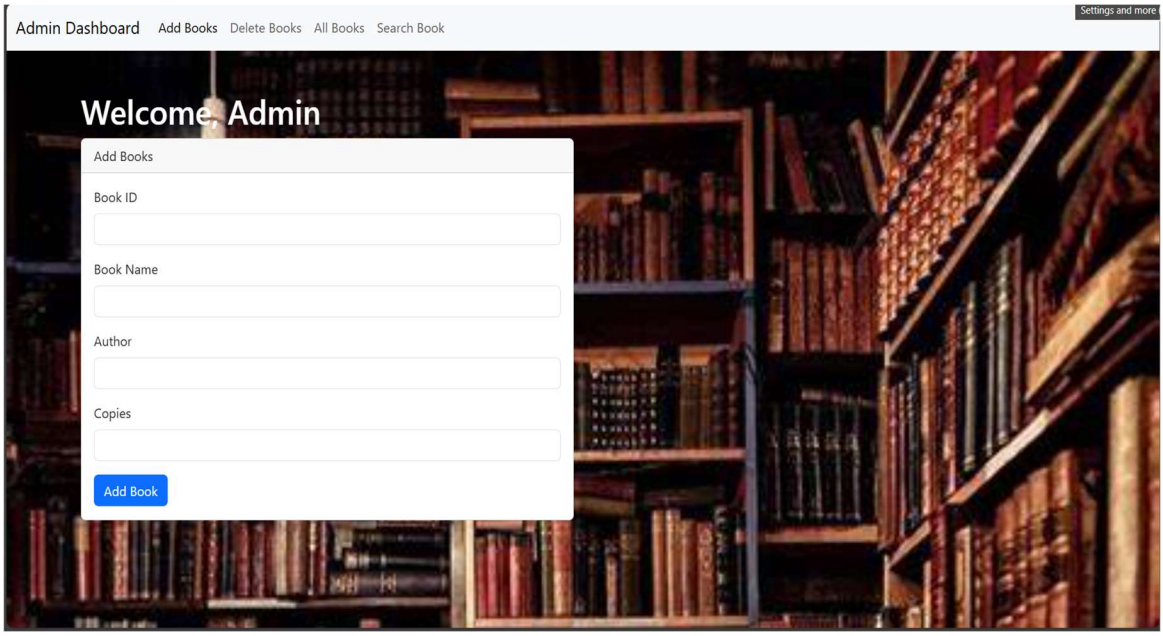


Fig 5.2.4 Adding Book

The figure 5.2.4 illustrates the UI of the user functionality-adding a book by entering all the book details such as Book ID, Book name, Author name and number of copies of the book available in the library.

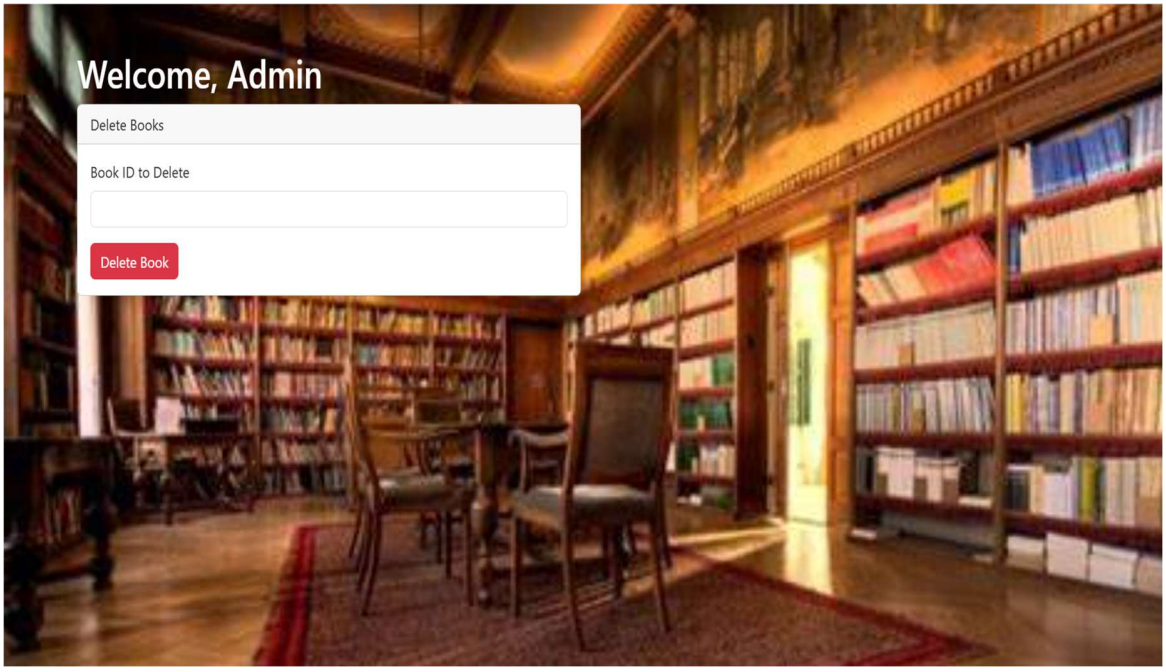


Fig 5.2.5 Deleting a Book

The figure 5.2.4 illustrates the UI of the user functionality-deleting a book by entering the book ID.

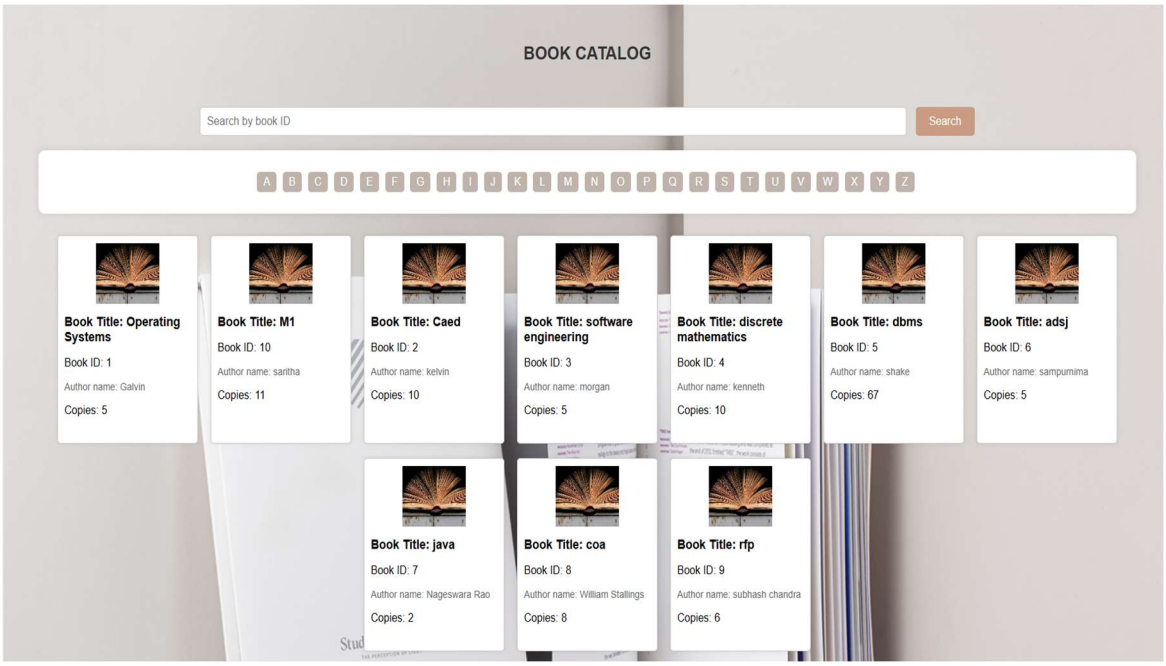


Fig 5.2.6 Searching a Book

The figure 5.2.6 depicts one of the user functionality-book searching from the book catalogue provided based on the book name or ID and the corresponding book is displayed if present.

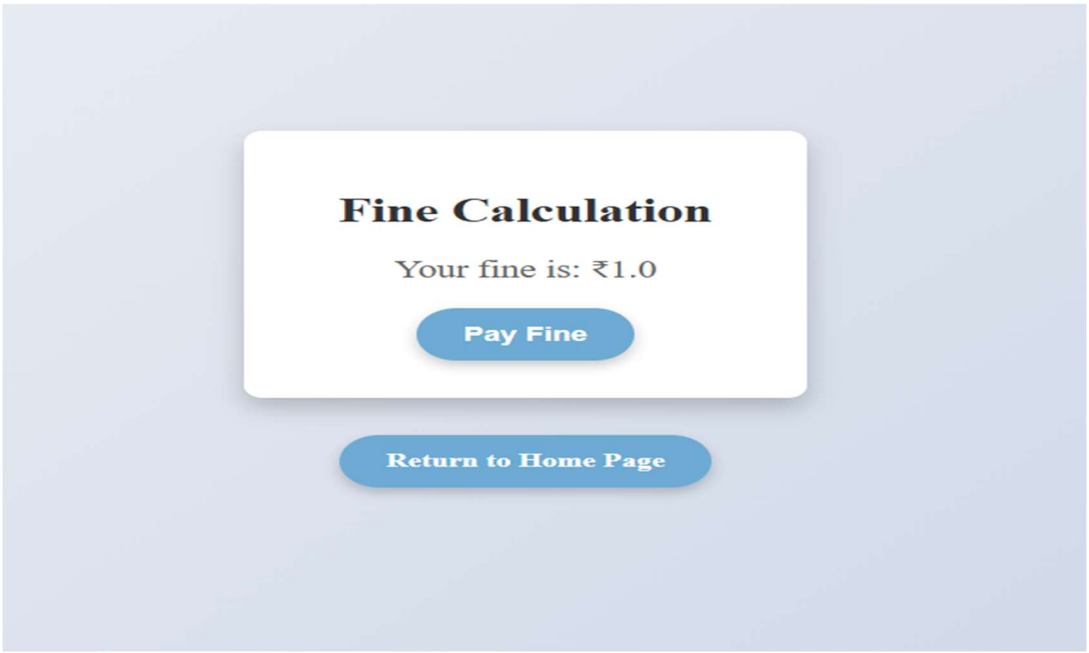


Fig 5.2.7 Fine Calculation of a Book

The figure 5.2.7 shows a UI picture of the fine calculated on a specific book if it is returned after the due date.

5.3 TESTING

Testing Steps for the Library Management System:

- **Unit Testing:**

  Step 1. Begin with unit testing to test individual components like book search, borrowing, and administrative functionalities independently to ensure they perform as expected.

  Example Test case: User will be unable to borrow a book if the book is unavailable in the library. So a disclaimer is visible saying "Failed to Borrow Book". This is clearly shown in figure 5.3.1.

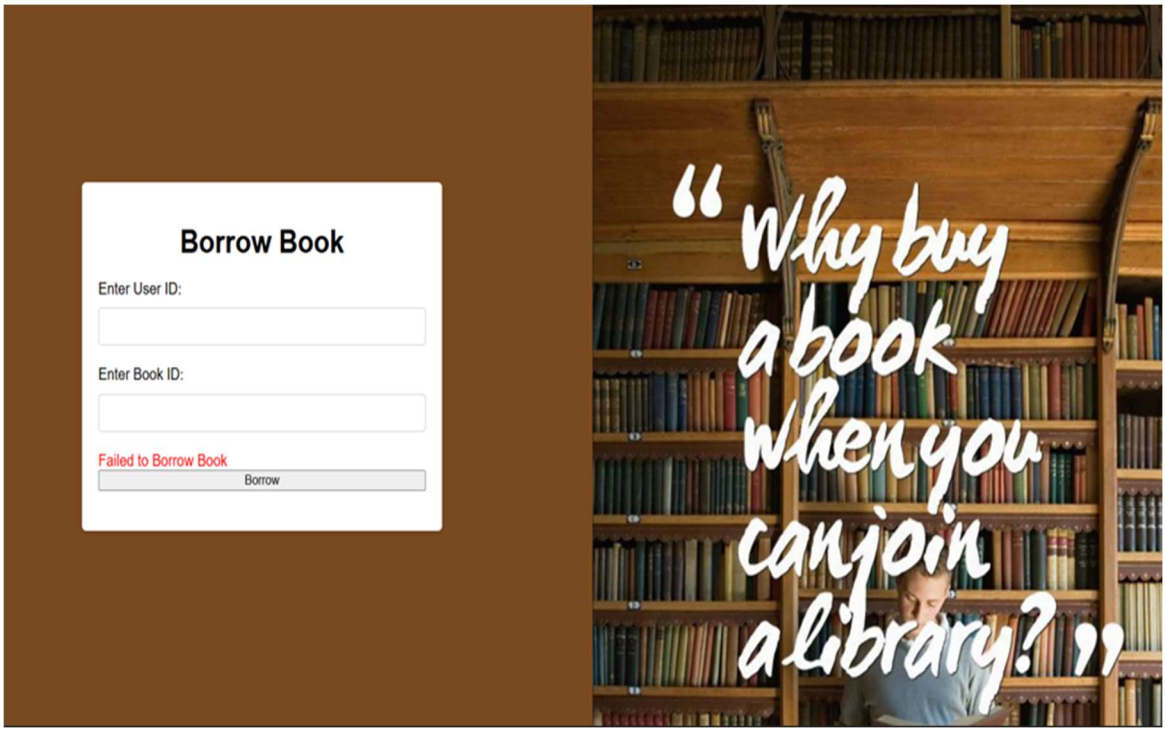


Fig 5.3.1 Borrowing a Book

- **Integration Testing:**

  Step 2: Integrate these components and conduct integration testing to verify that they work together seamlessly, with data flowing correctly between them.

- **System Testing:**

Step 3: Perform system testing by executing test cases covering various user scenarios, ensuring the system meets all specified requirements and functions correctly under different conditions.

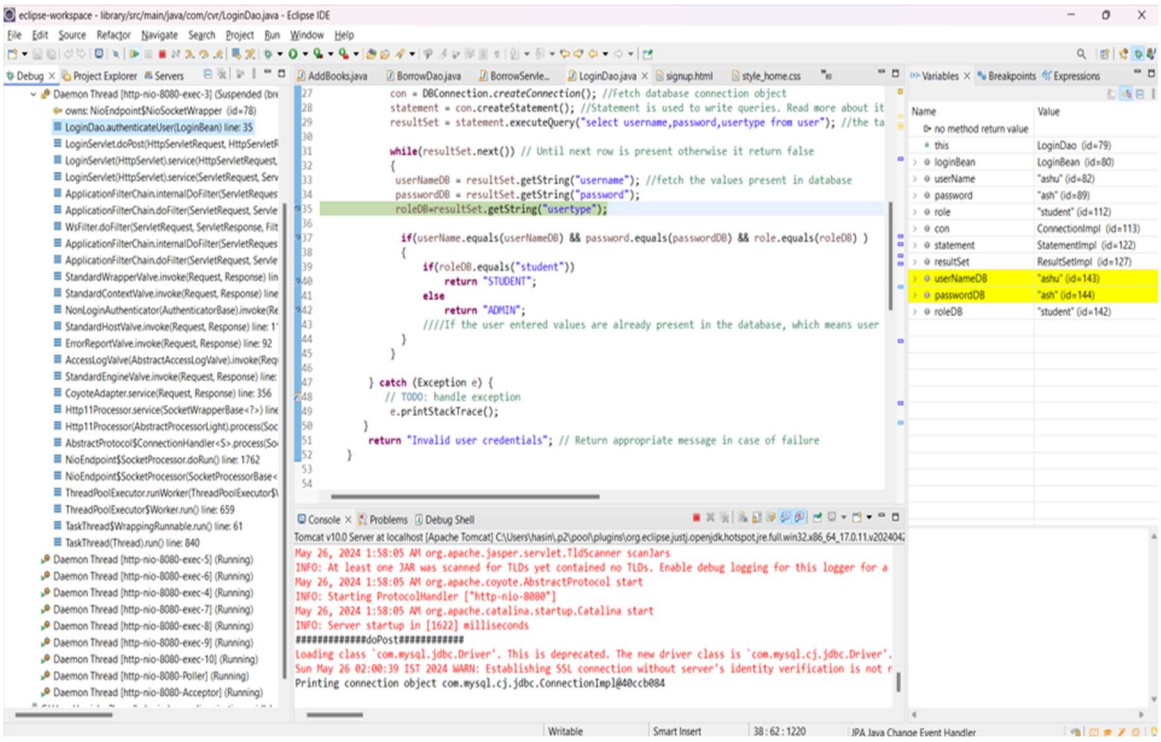


Fig 5.3.2 System Testing

The figure 5.3.2 illustrates the debugging process once the program is successfully integrated and the execution process starts on the localhost which in this case is Tomcat Server(version 10.0).

- **Functionality Testing:**

Step 4: Begin with unit testing, where individual components like book search, borrowing, and administrative functionalities are tested independently to ensure they perform as expected**.**

TESTCASE: Whenever user tries to search a book that is not available in the library, a message "Book not found" will be displayed, if found the book is displayed with the book details. This example is visualized in the below figure 5.3.3.

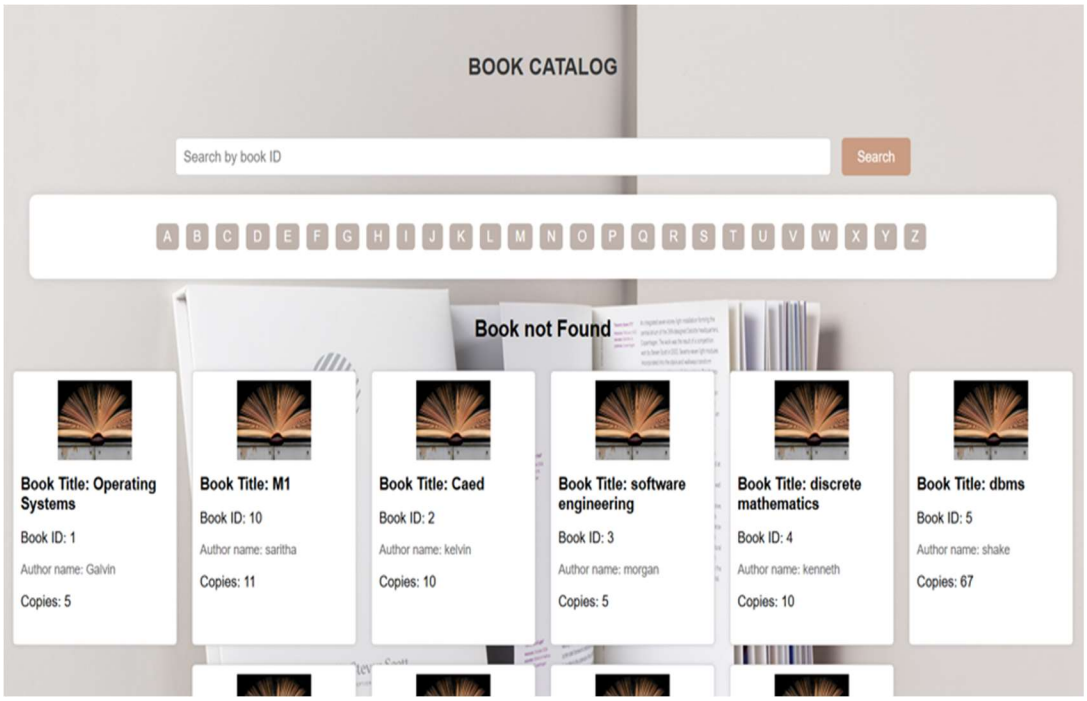


Fig 5.3.3 Book Not found Test Case

Step 5: Perform system testing by executing test cases covering various user scenarios, ensuring the system meets all specified requirements and functions correctly under different conditions.

- **Security Testing:**

  Step 6: Begin with vulnerability assessment by scanning the system for potential security vulnerabilities, including common issues like SQL injection or cross-site scripting.

- **Regression Testing:**

  Step 7: After implementing changes or bug fixes, perform regression testing to ensure that existing functionalities remain unaffected.

5.3.1 Test Cases and Results

The below table consists of all the test cases implemented and tested as a part of system testing and regression testing after the successful integration of the project. The last two columns shows the expected results and the status of the obtained result after execution i.e, the test results.

S.No	Test Module	Test Case Description	Expected Result	Test Result
1	Login Page	Select the correct user role and enter valid login credentials.	User logged in successfully and redirected to the corresponding main page based on user role selected.	Pass
2	Login Page	Select the correct user role and enter invalid login credentials.	Error message displayed.	Pass
3	Login Page	Select the incorrect user role and enter valid login credentials.	Error message displayed.	Pass
4	Login Page	Select the incorrect user role and enter invalid login credentials.	Error message displayed.	Pass
5	Registration Page	Fill in all the essential fields with valid details for registration.	User successfully registered and redirected to login page.	Pass
6	Registration Page	All the essential fields in the registration form are not filled.	Error message displayed.	Pass

7	Student home page	Click on Books	Redirected to book catalogue page.	Pass
8	Student home page	Click on Search	Redirected to search page.	Pass
9	Student home page	Enter valid book name in the search bar.	Specific book with its details displayed.	Pass
10	Student home page	Enter invalid book name in the search bar.	Error message displayed.	Pass
11	Student home page	Click on Borrow.	Redirected to Borrow page.	Pass
12	Student home page	Enter valid User ID and Book ID in book borrow form.	Book Borrowed successfully and redirected to my books page.	Pass
13	Student home page	Enter invalid User ID or Book ID in book borrow form.	Error message displayed.	Pass
14	Student home page	Enter the Book ID which is not available in the book catalogue.	Error message displayed.	Pass
15	Student home page	Click on My Account button and select My Books from the dropdown.	Only the books borrowed by the logged in user is displayed.	Pass
16	Student home page	Click on return	Book is successfully returned and redirected to fines page.	Pass
17	Student home page	Book is returned after the due date.	Corresponding fine displayed.	Pass

18	Student home page	Book is returned within the due date.	Calculated fine displayed as zero.	Pass
19	Student home page	Click on Logout.	User logged out successfully and redirected to home page.	Pass
20	Admin home page	Click on add books.	Redirected to add books page.	Pass
21	Admin home page	Enter valid book details in the add books form.	Book added successfully and redirected to book catalogue.	Pass
22	Admin home page	Enter invalid book details in the add books form.	Error message displayed.	Pass
23	Admin home page	Click on delete books.	Redirected to delete books page.	Pass
24	Admin home page	Enter valid Book ID in the delete book form.	Book deleted successfully and redirected to book catalogue.	Pass
25	Admin home page	Enter the Book ID in the delete form which is not available in the catalogue.	Error message displayed.	Pass

5.4 VALIDATION

Validation is a crucial aspect of any library management system (LMS) as it ensures that the data entered into the system is accurate, consistent, and reliable. Effective validation procedures help maintain the integrity of the library database, enhance user experience, and minimize errors in the system. In the context of an LMS, validation encompasses various aspects such as data validation, user validation, and transaction validation.

Data Validation

Data validation involves verifying the accuracy and completeness of information entered into the system. In an LMS, this includes validating details such as book titles, author names, ISBN numbers, publication dates, and other bibliographic information.

Examples of Data Validation:

Duplicate Checking: Preventing duplicate entries of books, authors, or other library resources.

Consistency Checking: Ensuring consistency between related data fields (e.g., author name matches the corresponding book record).

User Validation: User validation focuses on verifying the identity and credentials of individuals accessing the LMS.

Examples of User Validation:

Login Authentication: Verifying the username and password of library staff, administrators, and patrons.

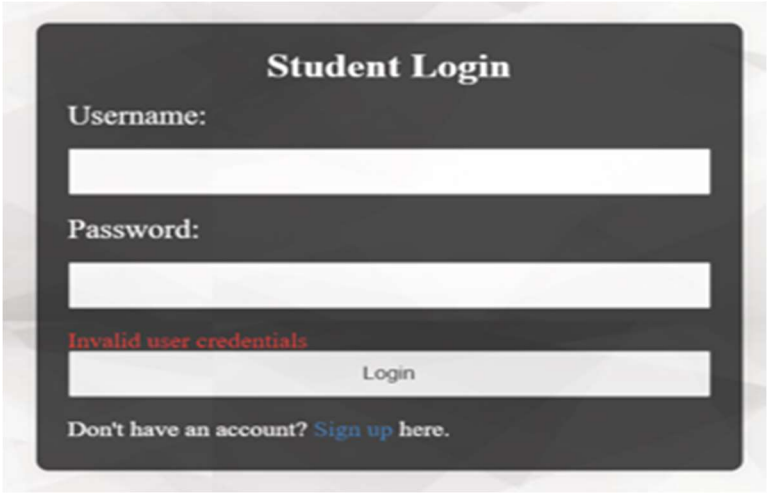


Fig 5.4.1 Wrong student credentials Test Case

The figure 5.4.1 demonstrates one of the test cases i.e an error message is displayed when invalid user details are entered.

Role-Based Access Control: Assigning different levels of access privileges based on user roles (e.g., librarian, administrator, borrower).

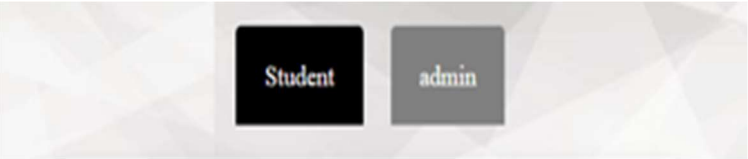


Fig 5.4.2 Role-Based Access

The figure 5.4.2 demonstrates the user role option which has to be appropriately selected during login process to ensure successful login.

Membership Validation: Ensuring that only registered members are allowed to borrow books or access certain library services. Transaction Validation Transaction validation involves validating various transactions performed within the LMS, such as book checkouts, returns, renewals, and reservations. This helps ensure that all library transactions are accurate, authorized, and properly recorded.

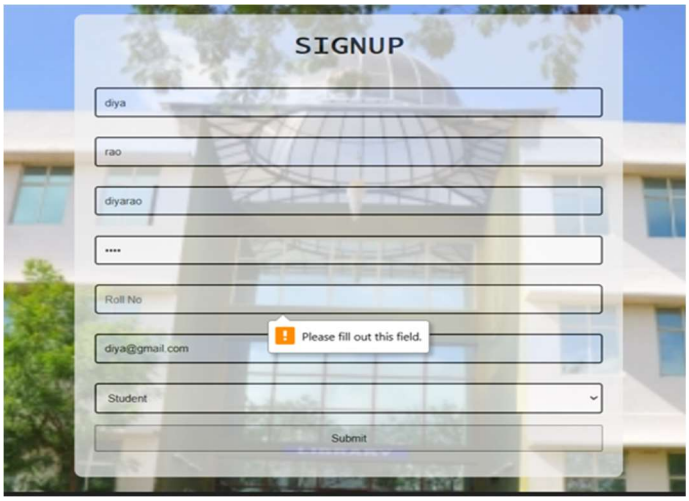


Fig 5.4.3 Necessary to fill compulsory fields

The figure 5.4.3 demonstrates one of the test cases i.e, all the essential fields has to be filled to ensure successful registration of the user.

CHAPTER-6

CONCLUSION

This chapter concludes the project report by summarizing the key concepts of the project and its execution along with a brief on the future scope for the project.

6.1 CONCLUSION

In conclusion, the Library Management System has achieved its primary objective because this system has facilitated the process of borrowing and returning books to the library. At the same time, all the processes that take place will run smoothly and systematically. Next, the system has provided a platform for the librarian to detect the late return from the students and calculate the fine automatically. Therefore, the librarian can also administer the system by managing the books available in the office, borrowers, and subordinate librarians. As a result of the development of the Library Management System, this system has overcome some of the problems librarians and students face in borrowing and returning. With the system already built, manual systems that require a logbook to record data are no longer needed. Borrowing data will be stored neatly in the database, making it easier for librarians and students to refer to the data in the future.

The system prioritizes simplicity and usability, allowing students to easily search for books, borrow them, and return them within the specified time frame. The reservation feature enables students to reserve books that are currently unavailable, ensuring fair access to popular titles.

In summary, the library management system presented in this documentation not only meets the core requirements of students and administrators but also sets a foundation for future enhancements and scalability. By leveraging technology to enhance library operations, the system contributes to a more efficient and user-centric library experience.

6.2 FUTURE SCOPE

Personalization and Recommendation:

- Implement algorithms to analyze students' reading habits, preferences, and academic interests.
- Provide personalized book recommendations based on their profiles and past borrowing history.
- Allow users to rate books, write reviews, and receive recommendations from peers.

E-books Availability:

- Integrate a digital library where students can access e-books alongside physical copies.
- Ensure seamless access to e-books across different devices through your mobile application.

Integration with Learning Management Systems (LMS):

- Enable single sign-on (SSO) integration with popular LMS platforms used by educational institutions. Sync course information, assignments, and reading lists between the LMS and your library system.
- Allow instructors to recommend specific library resources directly within the LMS interface.

Mobile Application Development:

- Develop a mobile application compatible with iOS and Android devices. Provide features such as searching for books, checking availability, placing holds, and accessing e-books on the go.
- Implement push notifications to alert users about due dates, availability of requested books, and personalized recommendations.

Data Analytics and Insights:

- Collect data on book usage, popular search queries, and borrowing patterns.

- Use analytics tools to generate insights into user behavior and preferences.
- Utilize these insights to optimize the library collection, improve recommendation algorithms, and enhance user experience.

Collaborative Tools:

- Integrate collaborative features such as group study room reservations, shared reading lists, and discussion forums.
- Enable students to create study groups, share notes, and collaborate on projects within the library system.
- Implement real-time chat or video conferencing for virtual study sessions and peer-to-peer assistance.

REFERENCES

[1] Robson, Michael, and David Stuart. "Information Management for Libraries and Information Services." Facet Publishing, 2013. - A comprehensive guide to information management principles specifically tailored to libraries and information services, offering insights into best practices and industry standards.

[2] SDLC format from geeksforgeeks.

[3] Journal of xi an university of architecture & technology by Shanmugam a P.

[4] YouTube to learn some unknown connectivity's.