Library Management System

A PROJECT REPORT
for
Mini Project-I (K24MCA18P)
Session (2024-25)

Submitted by

Manish Pandey
(202410116100115)
Kartik Agarwal
(202410116100096)

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Mr. Arpit Dogra
Assistant Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206

December 2024

CERTIFICATE

Certified that **Manish Pandey (202410116100115), Kartik Agarwal (202410116100096),** has/ have carried out the project work having “**Library Management System**” (**Mini Project-I, K24MCA18P**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Mr. Arpit Dogra	**Dr. Arun Kr. Tripathi**
Assistant Professor	**Dean**
Department of Computer Applications	**Department of Computer Applications**
KIET Group of Institutions, Ghaziabad	**KIET Group of Institutions, Ghaziabad**

Library Management System

Manish Pandey
Kartik Agarwal

ABSTRACT

The Library Management System is a project designed to develop a computerized system that efficiently manages the daily operations of a library. This system offers advanced features that are typically absent in conventional library management systems. Key functionalities include user logins for students and teachers, as well as an admin login for comprehensive system monitoring.

The system also incorporates an online notice board where students and teachers can share information about workshops, seminars, and events happening in the college or nearby institutions. The librarian, upon verification, can approve and post these announcements.

Students, upon logging in, can view the list of books they have issued, along with the corresponding issue and return dates. They can also submit book requests to the librarian by filling out a dedicated book request form. The librarian, using the admin account, can generate various reports such as student reports, issue reports, and book inventory reports.

This project is designed to assist both students and library staff in maintaining the library efficiently, reducing manual effort, and enhancing the overall library experience.

ACKNOWLEDGEMENT

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Mr. Arpit Dogra** her guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi**, Professor and Dean, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Manish Pandey

(202410116100115)

Kartik Agarwal

(202410116100096)

TABLE OF CONTENTS

LIST OF FIGURES

Introduction

1.1 Introduction

Introduction

Libraries play a vital role in providing access to knowledge and resources for students, researchers, and the general public. However, managing a library efficiently can be challenging, especially as the volume of books and users increases. To address this, a Library Management System (LMS) has been developed to streamline operations, reduce manual effort, and enhance user experience.

The Library Management System is a web-based application designed to manage library resources effectively. It facilitates key functions such as book inventory management, user registration, book issuance, returns, and fine calculations. The system ensures that both library staff and users can access the required services with ease and efficiency.

Objective

The primary objective of the Library Management System is to simplify and automate library operations, ensuring accuracy, time efficiency, and user satisfaction. This system aims to:

- Provide an intuitive interface for managing book inventories and user data.
- Enable seamless tracking of issued and returned books.
- Generate reports for administrative purposes.
- Enhance user convenience through a digital platform.

Scope of the Project

The system is intended for use by:

- **Librarians**: To manage book records, track borrowing activities, and generate reports.
- **Users**: To search for books, check availability, and manage their borrowing history.
- **Administrators**: To oversee library operations, user management, and overall system performance.

Technologies Used

To develop the Library Management System, the following technologies have been utilized:

1. **HTML (HyperText Markup Language):**
    - Used to create the structure of web pages.
    - Provides the foundation for the user interface components.
2. **CSS (Cascading Style Sheets):**
    - Used to style and design the application interface.
    - Enhances user experience through responsive and visually appealing designs.

3. **JavaScript:**
   - Implements client-side scripting for interactive elements.
   - Validates user input and enhances functionality such as dynamic search and real-time notifications.
4. **PHP (Hypertext Preprocessor):**
   - Used for server-side scripting and business logic implementation.
   - Manages database interactions and processes user requests.
5. **Database (e.g., MySQL):**
   - Stores and manages data for books, users, and transactions.
   - Ensures data integrity and supports efficient querying.

Features of the System

1. **User Management:**
   - Registration and authentication for users and staff.
   - Role-based access control.
2. **Book Management:**
   - Addition, deletion, and updating of book records.
   - Categorization and search functionality.
3. **Transaction Management:**
   - Issue and return books.
   - Automated fine calculation for overdue returns.
4. **Reporting:**
   - Generation of reports on book inventory, user activity, and transactions.
5. **Notifications:**
   - Alerts for due dates and system updates.

1.2 Problem In Existing Systems

1. **No Login Facility:**

- The current system lacks any login functionality for teachers or students. This makes it challenging to offer personalized or restricted access to library resources.

2. **No Online Book Reservation:**

- There is no option to reserve books online. Students and teachers need to visit the library in person to check the availability and reserve books, leading to inconvenience and delays.

3. **No Online Notice Board:**
   - The existing system does not have an online notice board to communicate information about workshops, seminars, or other events happening in the college or nearby institutions. As a result, students miss out on timely updates and opportunities.
4. **No Facility for Lecture Notes:**
   - Teachers cannot upload lecture notes or reference materials for students, making it difficult for students to access essential resources outside classroom hours.
5. **No Report Generation:**
   - The system does not provide any tools to generate reports, such as student borrowing histories or book issue records. Librarians have to rely on manual methods, which are time-consuming and prone to errors.
6. **No Book Request or Suggestion Mechanism:**
   - Students and teachers cannot submit requests for specific books or provide suggestions to improve the library services. This lack of feedback mechanism hampers the library's ability to cater to user needs effectively.

1.3 Proposed System

1. **Login Facility for Students and Teachers:**
   - The new system will introduce dedicated login functionality for students and teachers.
     - **Student Login:** Allows students to access their borrowing history, reserve books, and provide feedback.
     - **Teacher Login:** Enables teachers to upload lecture notes, view available resources, and reserve books.
2. **Online Book Reservation:**
   - The proposed system will enable users to search for books and reserve them online.
     - This feature will reduce the hassle of physically checking for book availability and improve resource management.
3. **Online Notice Board:**
   - An online notice board will display important updates, such as:
     - Workshops, seminars, and events in the college and nearby institutions.

        - Book fairs, guest lectures, and academic deadlines.
    - This ensures students and faculty members stay informed about relevant opportunities.

4. **Facility to Upload Lecture Notes:**
    - Teachers can upload lecture notes, presentations, and additional reading materials to the system.
        - Students can easily access and download these resources, enhancing the learning experience.
5. **Automated Report Generation:**
    - Librarians will have access to tools for generating reports such as:
        - **Student Reports:** Detailed borrowing history, overdue records, and fines.
        - **Book Issue Reports:** Data on issued, returned, and reserved books.
    - These reports will streamline library operations and improve record-keeping.
6. **Book Request and Suggestion Mechanism:**
    - Students and teachers can request specific books that are not currently available in the library.
    - They can also provide suggestions to improve library services, such as adding new resources or modifying policies.
    - These requests and suggestions will be reviewed by the librarian or library committee for actionable insights.

Advantages of the Proposed System

- **Enhanced Accessibility:** Students and teachers can access library resources anytime and anywhere.
- **Improved Efficiency:** Online systems for reservations, notices, and report generation reduce manual workload.
- **Personalization:** User-specific features like login and borrowing history provide a tailored experience.
- **Resource Optimization:** Tools like online reservations and feedback mechanisms ensure better resource allocation.
- **Better Communication:** The online notice board and suggestion system improve interaction between the library and its users.

1.4 Functional Requirements

1. **User Management:**
   - Registration and authentication for users and staff.
   - Role-based access control.
2. **Book Management:**
   - Addition, deletion, and updating of book records.
   - Categorization and search functionality.
3. **Transaction Management:**
   - Issue and return books.
   - Automated fine calculation for overdue returns.
4. **Reporting:**
   - Generation of reports on book inventory, user activity, and transactions.
5. **Notifications:**
   - Alerts for due dates and system updates.

1.5 5 Non-functional requirements

Non-functional requirements (NFRs) define the quality attributes, system performance, and operational constraints for a system. For the Library Management System, the following NFRs are critical for ensuring its usability, security, and scalability:

1. **Performance**
   - **Response Time:** The system must respond to user requests (e.g., searches, book transactions) within 3 seconds to ensure a smooth user experience.
   - **Scalability:** The system should be able to handle a large number of concurrent users and scale horizontally during peak usage periods, such as examination times.
   - **Availability:** The system should have 99.9% uptime, ensuring minimal disruptions in service, especially during crucial operational hours.
2. **Usability**
   - **User-Friendly Interface:** The system should have an intuitive and easy-to-navigate interface for both users and administrators.

- **Accessibility:** The platform must comply with accessibility standards (e.g., WCAG 2.1) to ensure that users with disabilities can use the system.
- **Multi-device Compatibility:** The application must be responsive and function seamlessly across desktop, tablet, and mobile devices.

3. **Security**
   - **Data Protection:** All sensitive user data, including personal details and book transaction records, must be encrypted using industry-standard encryption protocols (e.g., SSL/TLS).
   - **Authentication:** Strong authentication mechanisms such as multi-factor authentication (MFA) for user logins and admin access to ensure system security.
   - **Authorization:** Role-based access control (RBAC) should be implemented to ensure that users and admins have access only to the resources and functions they are authorized to use.
   - **Compliance:** Ensure compliance with global data privacy regulations (e.g., GDPR, CCPA).
4. **Reliability**
   - **Error Handling:** The system should gracefully handle errors, providing meaningful error messages to users without exposing sensitive system information.
   - **Backup and Recovery:** Implement automated backup mechanisms at regular intervals to avoid data loss and ensure quick recovery in case of failure.
   - **Fault Tolerance:** The system must be able to recover from failures quickly and continue functioning with minimal downtime, especially for critical services like book issuance and returns.
5. **Maintainability**
   - **Code Quality:** The system must be built with clean, modular, and well-documented code to ease maintenance and future upgrades.
   - **Logging and Monitoring:** Implement comprehensive logging and monitoring systems to track system performance, user activities, and potential issues in real-time.
   - **Updates and Patches:** The system should be easily upgradable to incorporate new features, security patches, and bug fixes without disrupting operations.

Feasibility Analysis

2.1. Feasibility Analysis for a Library Management System

the Library Management System evaluates its viability across technical, operational, and economic dimensions to ensure successful implementation and adoption.

1. **Technical Feasibility**
    - The required technologies (HTML, CSS, JavaScript, PHP, MySQL) are widely used and well-supported, making them reliable for system development.
    - Development resources, such as skilled developers and system architects, are readily available.
    - The infrastructure required, including web servers and database management systems, is cost-effective and scalable.
2. **Operational Feasibility**
    - The system addresses the core operational needs of libraries by automating book management, user registration, and transaction tracking.
    - Librarians and users can quickly adopt the system due to its intuitive user interface and clear functionality.
    - Training requirements for staff are minimal, reducing the time needed for operational readiness.
3. **Economic Feasibility**
    - The development and maintenance costs are within budget constraints for most institutions.
    - Automation of manual tasks reduces operational costs in the long term, providing a high return on investment.
    - The system's scalability ensures that future expansions or increased user loads do not incur significant additional costs.
4. **Legal Feasibility**
    - The system complies with data privacy and security regulations, ensuring legal operation and user trust.
    - Role-based access control and data encryption minimize risks of data breaches or unauthorized access.

Objectives of Library Management System

1. **Automate Library Operations:**
   - Eliminate manual processes for managing book inventories, user records, and transactions.
2. **Enhance User Accessibility:**
   - Provide an intuitive and user-friendly platform for students, researchers, and staff to access library services.
3. **Improve Efficiency:**
   - Minimize time and effort for librarians in tracking books, managing user records, and generating reports.
4. **Ensure Data Integrity:**
   - Maintain accurate and up-to-date records of books, users, and transactions.
5. **Facilitate Real-time Updates:**
   - Enable real-time updates for book availability, issued books, and overdue notifications.
6. **Promote Resource Sharing:**
   - Allow inter-library book sharing and resource tracking for better utilization of resources.
7. **Support Decision-making:**
   - Generate detailed analytical reports to assist administrators in decision-making and planning.

Hardware and Software Requirements

Hardware Requirements

1. **Processor**: Minimum Intel i3 or equivalent (Recommended: Intel i5 or above).
2. **RAM**: Minimum 4 GB (Recommended: 8 GB or higher).
3. **Storage**:
    - 10 GB free disk space for development and database storage.
    - Additional space for XAMPP installation (~1 GB).
4. **Display**: Monitor with at least 1366x768 resolution.
5. **Network**: Internet connection (for downloading libraries and updates, if required).
6. **Peripherals**: Keyboard, mouse, and optional printer (for issuing/return receipts).

Software Requirements

1. **Operating System**:
    - Windows 10/11 (64-bit) or Linux distributions like Ubuntu (20.04 or later).
    - macOS (latest version compatible with XAMPP).
2. **XAMPP (Web Server)**:
    - Version: Latest stable version (supports PHP 8.0+).
    - Components: Apache (web server), MySQL (database), and PHP (server-side scripting).
3. **Development Tools**:
    - **Code Editor**:
        - Visual Studio Code (recommended) or Sublime Text/Notepad++.
        - Plugins for PHP, HTML, CSS, and JavaScript support.
    - **Version Control**: Git (optional, for collaboration and versioning).
    - **Browser**: Google Chrome, Firefox, or Microsoft Edge (for testing).
4. **Languages and Frameworks**:
    - **Frontend**: HTML5, CSS3, JavaScript (with optional libraries like jQuery).
    - **Backend**: PHP 8.0 or higher.
    - **Database**: MySQL (InnoDB engine recommended).

5. **Libraries and Dependencies**:
    - CSS Framework (optional): Bootstrap 5 (for responsive design).
    - JavaScript Libraries: jQuery (optional), Vanilla JS (preferred for simplicity).
    - PHP: PDO/MySQLi for database interaction.
6. **Testing Tools**:
    - Browser Developer Tools (integrated into modern browsers).
    - XAMPP Control Panel for managing the server.
7. **Other Requirements**:
    - Email server (optional) for notifications like book return reminders (using PHPMailer or similar library).

    PDF generation library (optional) like FPDF or TCPDF for generating reports

Project Flow

The project flow outlines the stages involved in the design, development, and deployment of the system. Below is the structured flow:

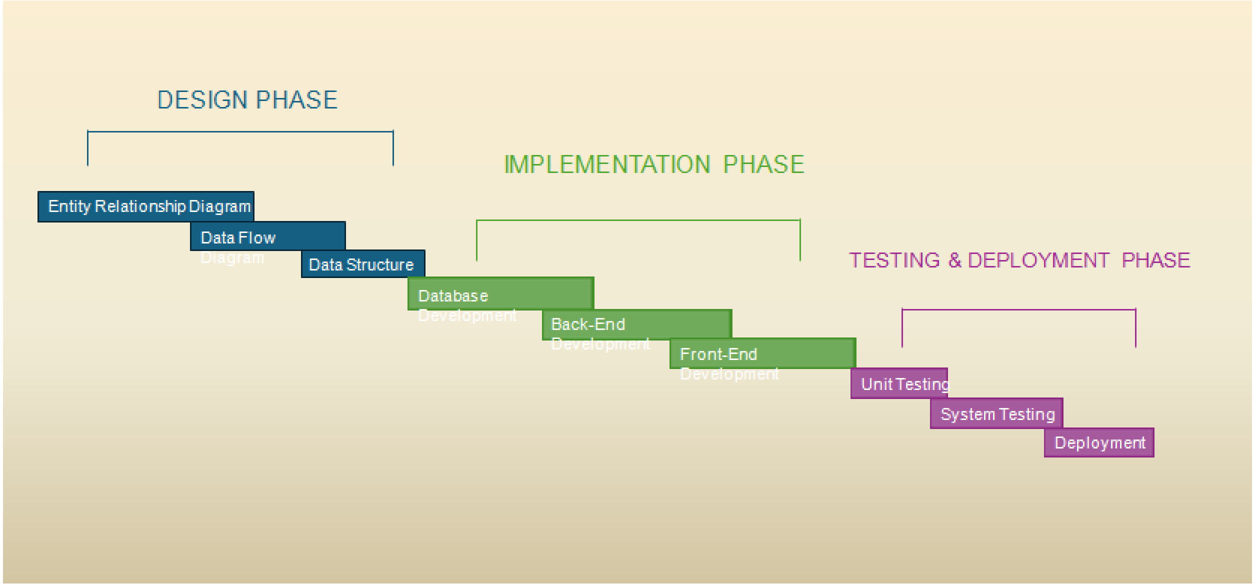


Fig. 5.1 project flow

1. Requirements Gathering

- **Tasks**:
  - Collect requirements from stakeholders (librarians, students, and administrators).
  - Identify core features like book catalog management, issue/return tracking, user management, and overdue notifications.
  - Document functional and non-functional requirements.

2. System Design

- **Architecture Design**:
  - Choose a monolithic or modular architecture for scalability.
- **Database Design**:
  - Design database schema for books, users, transactions (issue/return), and overdue logs.
- **UI/UX Design**:
  - Create wireframes for interfaces, including user dashboards, search functionality, and admin panels.

3. Development

- **Front-End Development**:
    - Build user-friendly interfaces for students and admin using HTML, CSS, and JavaScript.
- **Back-End Development**:
    - Develop server-side logic using PHP and connect the application with MySQL for database management.

4. Integration

- **Barcode Integration**:
    - Add functionality for scanning barcodes to track book issues and returns.
- **Notification Systems**:
    - Integrate email or SMS APIs for overdue reminders and updates.

5. Testing

- **Unit Testing**:
    - Validate individual modules like book addition, user registration, and search functionality.
- **Integration Testing**:
    - Ensure seamless interaction between modules (e.g., book issue and overdue management).
- **User Acceptance Testing (UAT)**:
    - Get feedback from librarians and users to ensure the system meets their needs.

6. Deployment

- **Tasks**:
    - Host the system locally (XAMPP) or deploy it on a cloud platform (e.g., AWS, Azure).
    - Set up the domain (if required), SSL, and monitoring tools for live operation.

7. Post-Launch Support

- **Tasks**:
    - Monitor system performance and resolve any issues.
    - Collect feedback for updates and enhancements.

5.2 Modules

A **Tour and Travel System** typically consists of several modules that handle different functionalities required to manage and facilitate travel bookings, packages, user management, and customer service. Here are the essential modules of such a system:

1. . User Management Module

- **Purpose**: Manages user registration, login, and profile details.
- **Features**:
  - User registration (email, phone, library ID).
  - Login and authentication (username/password).
  - Role-based access (Student, Librarian, Admin).
- **Technologies**: PHP (backend), MySQL (database), HTML, CSS, JavaScript (frontend).

2. Book Catalog Management Module

- **Purpose**: Handles adding, updating, and searching books.
- **Features**:
  - Admin can add/edit/delete books (title, author, ISBN, category, etc.).
  - Users can search and filter books by title, author, or category.
  - Display book availability status (Available/Issued).
- **Technologies**: PHP, MySQL, JavaScript (AJAX for dynamic search).

3. Issue and Return Module

- **Purpose**: Manages book lending and return processes.
- **Features**:
  - Track book issues (user ID, book ID, issue date, due date).
  - Update the status when a book is returned.
  - Overdue tracking with fine calculation.
- **Technologies**: PHP, MySQL, JavaScript.

4. Overdue Notification Module

- **Purpose**: Notifies users about due or overdue books.
- **Features**:
  - Automatic reminders via email or SMS.
  - Overdue fine calculation and notification.
  - Integration with notification APIs (e.g., Twilio, SendGrid).
- **Technologies**: PHP, MySQL, API Integration (Email/SMS).

5. Reports and Analytics Module

- **Purpose**: Generates reports and tracks library usage statistics.
- **Features**:
  - Generate reports (e.g., most borrowed books, overdue fines collected).
  - View analytics like user activity, book borrowing trends.
  - Export reports to PDF or Excel.
- **Technologies**: PHP, MySQL, JavaScript.

6. Admin Dashboard Module

- **Purpose**: Central hub for library management.
- **Features**:
  - View and manage users, books, and transactions.
  - Add/edit/delete categories and authors.
  - Monitor system analytics (e.g., book availability, overdue fines).
- **Technologies**: PHP, MySQL, HTML, CSS, JavaScript (charts and graphs).

7. Search and Recommendation Module

- **Purpose**: Enhances user experience with efficient search and recommendations.
- **Features**:
  - Advanced search options (filters by author, category, year).
  - Recommend books based on borrowing history or popular trends.

- **Technologies**: PHP, MySQL, JavaScript.

5.3 Entity-Relationship (ER) Diagram

Here’s how an **ER diagram for a Library Management System** could look conceptually. Below are the key entities and their relationships tailored for a library context:

Entities:

1. **Book**
   - Attributes: BookID, Title, Author, Publisher, Genre, ISBN, Edition
2. **Member**
   - Attributes: MemberID, Name, ContactInfo, Address
3. **Admin**
   - Attributes: LibrarianID, Name, ContactInfo, Email
4. **Borrowing Transaction**
   - Attributes: TransactionID, BorrowDate, DueDate, ReturnDate, FineAmount
5. **Payment**
   - Attributes: PaymentID, Amount, PaymentDate, Mode
6. **Category**
   - Attributes: CategoryID, CategoryName

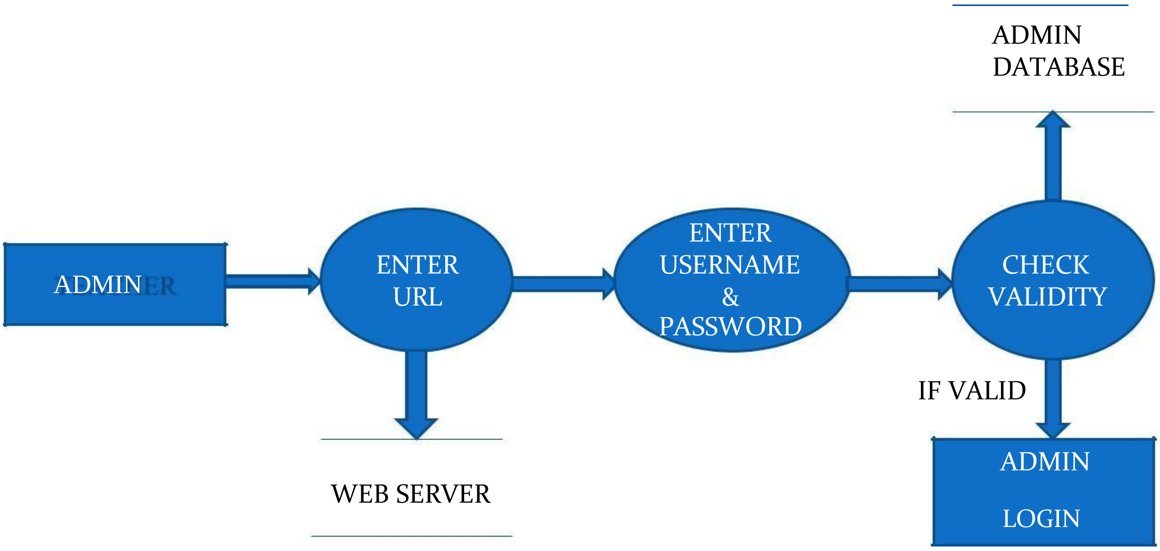
Relationships:

- **Book** is categorized into **Category** (One-to-Many).
- **Member** borrows **Book** through **Borrowing Transaction** (Many-to-Many with an associative entity).
- **Admin** manages the **Borrowing Transaction** (One-to-Many).
- **Member** makes **Payment** for fines (One-to-Many).

Would you like this structured into a formal textual schema, or should I create an image of the ER diagram?

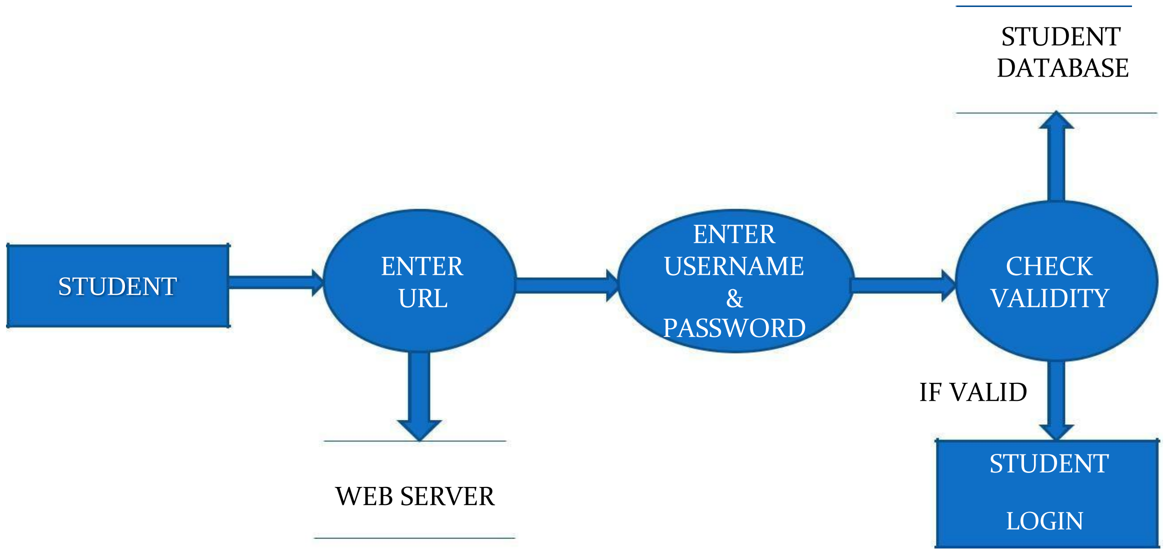
5.4 Data Flow Diagram (DFD) for a Tour and Travel System

5.4.1 DATA FLOW DIAGRAM FOR TEACHER LOGIN



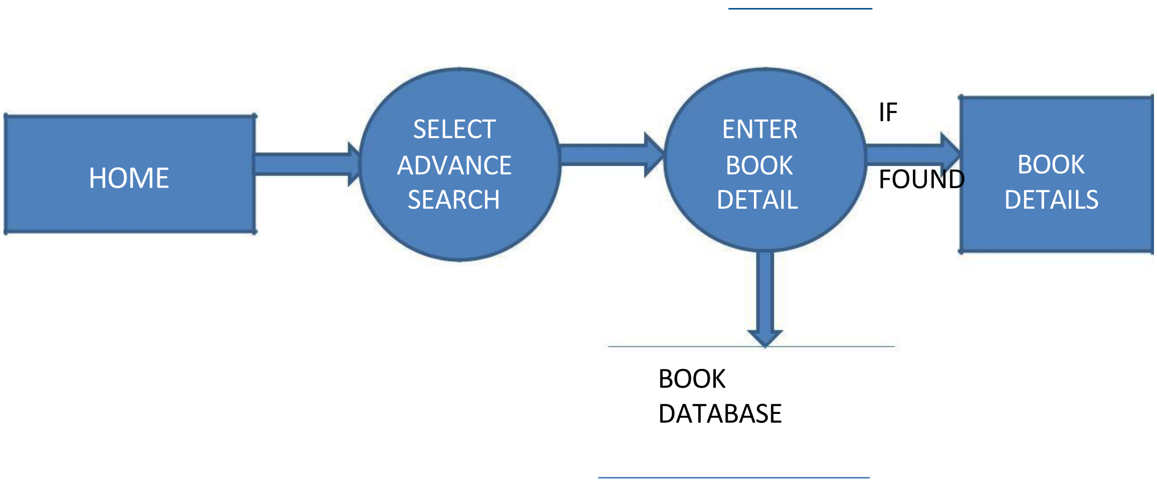
After entering to the home page of the website, teacher can choose the ADMIN LOGIN option where they are asked to enter username & password, and if he/she is a valid user then a teacher login page will be displayed.

5.4.2 DATA FLOW DIAGRAM FOR STUDENT LOGIN



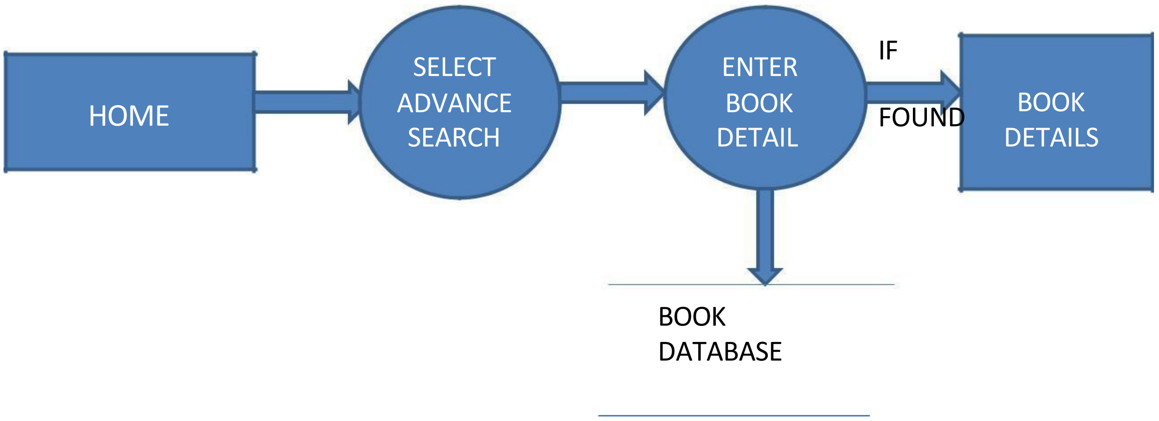
After entering to the home page of the website , student can choose the STUDENT LOGIN option where they are asked to enter username & password , and if he/she is a valid user then a student login page will be displayed.

5.4.3 DATA FLOW DIAGRAM FOR BOOK ISSUE



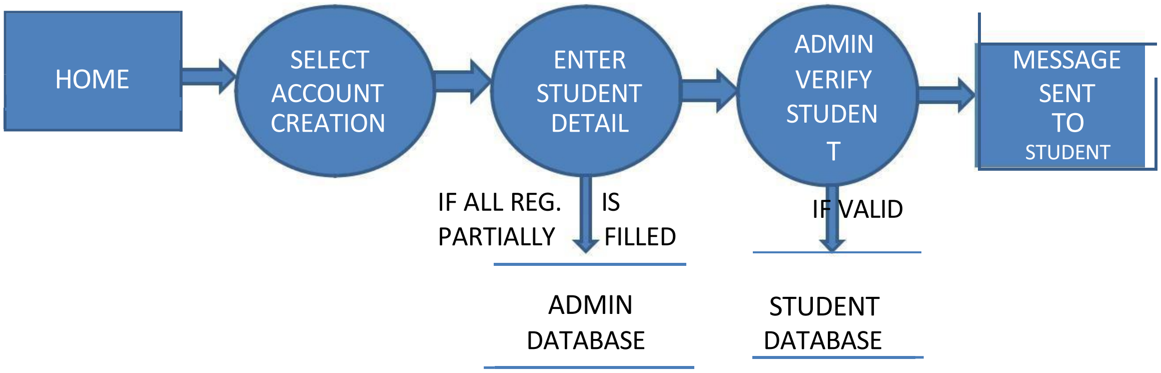
After the home page login there will be an option of the book search where after entering book detail like author name, publication, book name etc book details will be displayed.
It is a 2nd level Data Flow Diagram where after entering STUDENT LOGIN page he/she can select a book issue option where after entering the book detail, he/she can select the book issue option and if the maximum no of books issued limit is not crossed then a request will be sent to the librarian who will approve the book issue.

5.4.4 DATA FLOW DIAGRAM FOR BOOK SEARCH



After the home page login there will be an option of the book search where after entering book detail like author name, publication, book name etc book details will be displayed.

5.4.5 DATA FLOW DIAGRAM FOR ACCOUNT CREATION

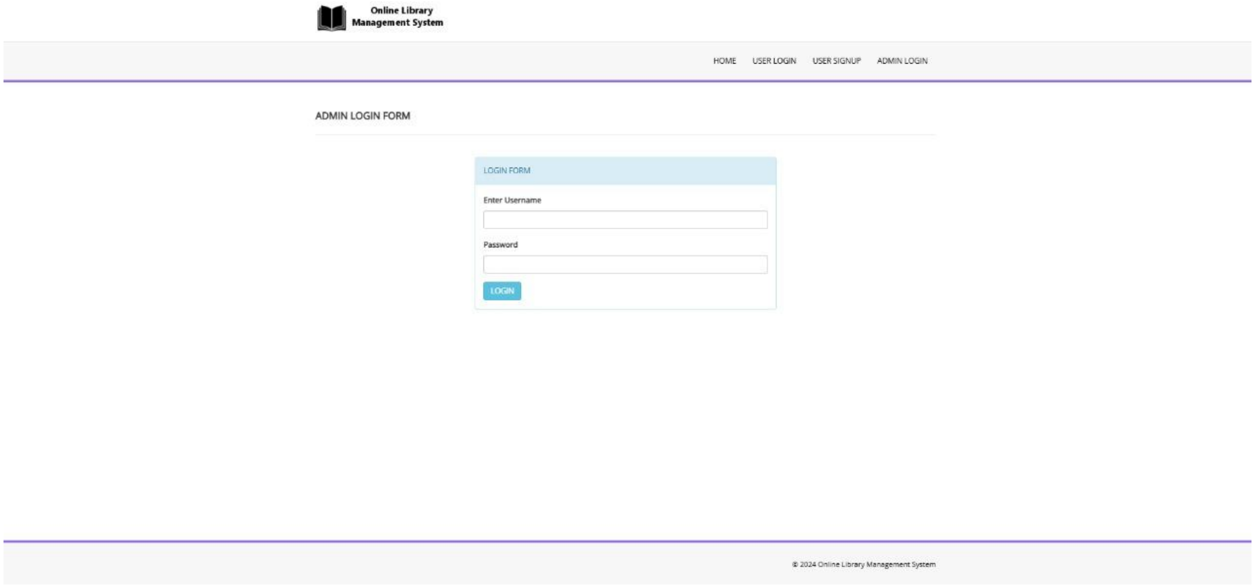


After the home page login there will be an option of CREATE AN ACCOUNT where after entering student detail ,if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the library.

Project Outcome

The Library Management System simplifies the management and accessibility of library resources. It provides users with an intuitive platform to search for books, reserve materials, and manage their borrowing activities seamlessly. Librarians and administrators benefit from efficient catalog management, real-time inventory tracking, and analytics to monitor resource usage and user behavior. The system enhances user satisfaction through personalized recommendations, automated notifications, and a streamlined interface for communication. By integrating payment gateways for overdue fines, loyalty programs for frequent users, and feedback mechanisms, it ensures a comprehensive and scalable solution. Ultimately, the system boosts operational efficiency, enhances user engagement, and supports the effective utilization of library resources.

Home Page



ADMIN LOGIN FORM

LOGIN FORM

Enter Username

Password

© 2024 Online Library Management System

Fig 6.1

Admin Dashboard

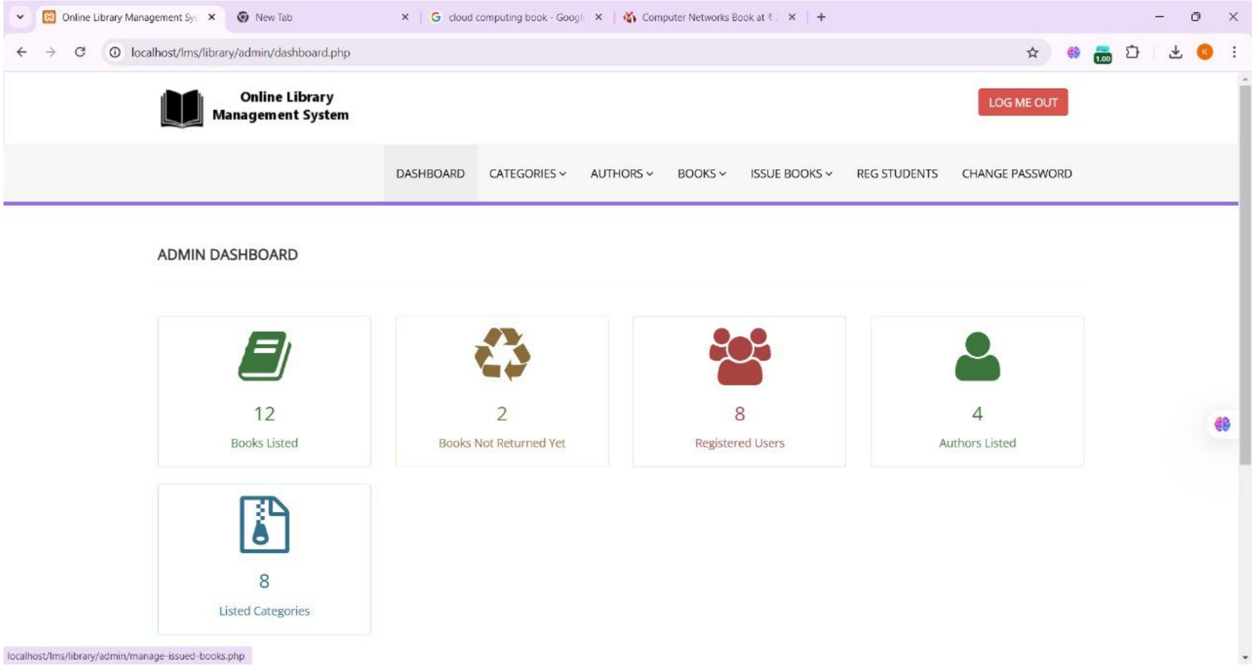


Fig 6.2

User Dashboard

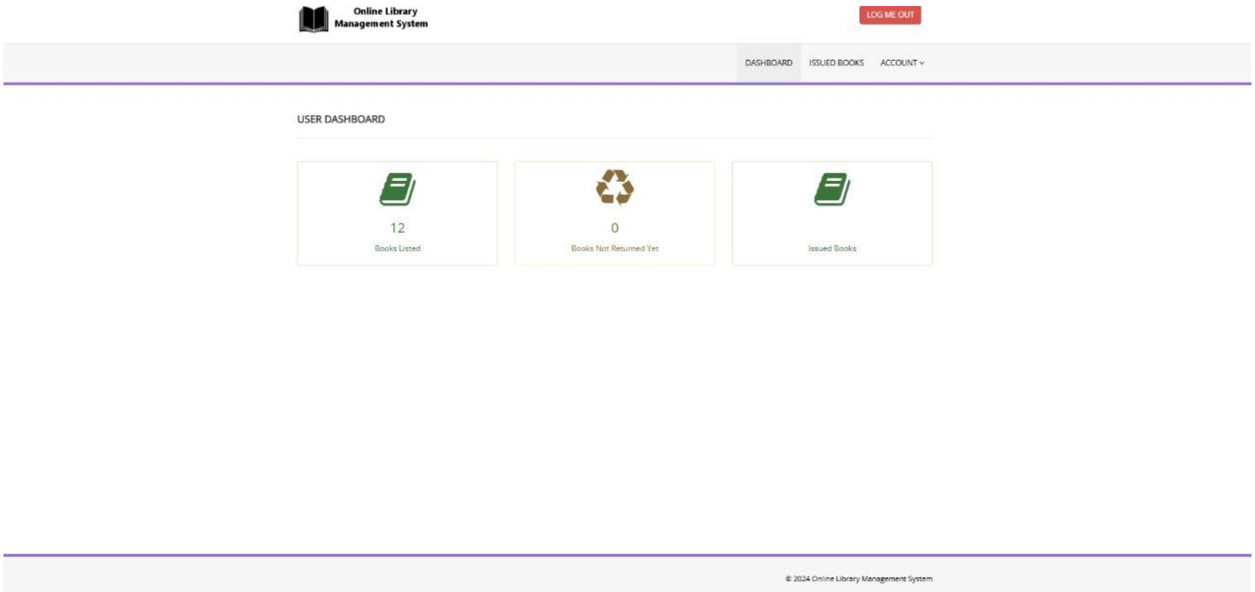


Fig 6.3

User Signup

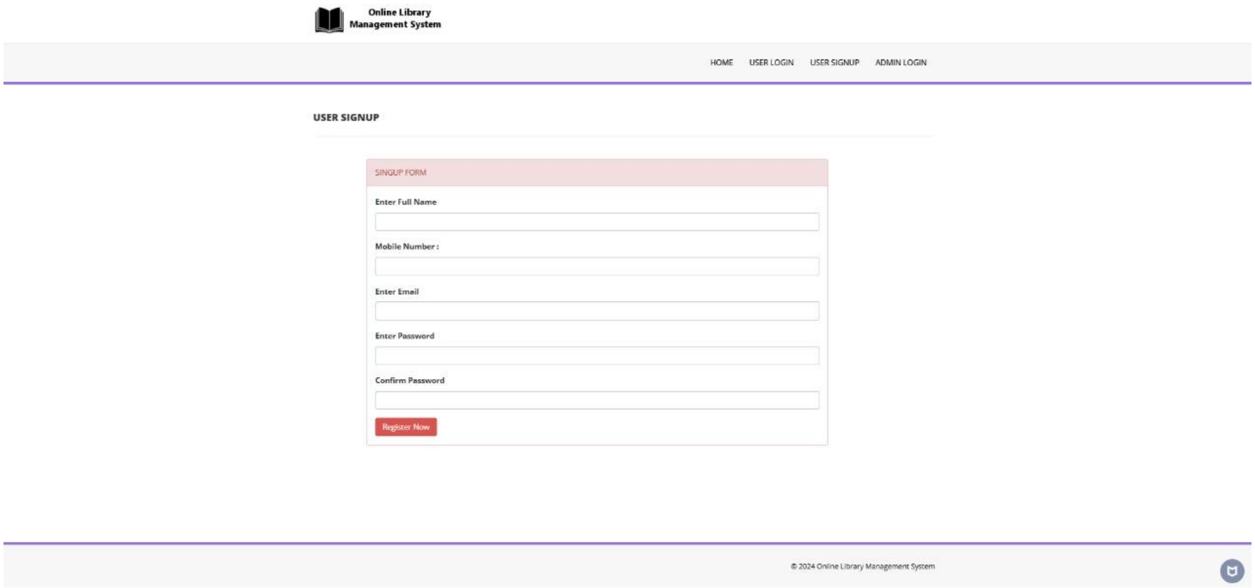


Fig 6.4

REFERENCES/ Bibliography

• 2002, Bretthauer, D. (2002) presents actual status and updates on open source software for libraries. Catherine, E. (2002) provides an overview of present state of ILS development. Breeding, M. • (2002) provides the information about Koha, Learning Access ILS, and Avanti Micro LCS Integrated Library system.

• Boss, R. W. (2005), in his article provides criteria and on the basis this criteria he has evaluated 12 open source library management systems,

• Breeding, M. (2007), in his article, provides up-to-date information about Koha Evergreen and learning access ILS, integrated library system. The author gives comprehensive information about latest developments in software since 2002.

• DeVoe, K. (2007) provides a brief overview of nine open source integrated library.

• Breeding Marshall (2008) provides thorough information on Koha, Evergreen and OPALS, New Gen Lib. He also provides information on trends in open source ILS adoption. Boss, R. W. (2008) identified 12 integrated library management systems with some current development activity underway as early 2008. Balnaves E.

• Breeding, M. (2009) focuses on questions regarding to what extent open source ILS products can be considered viable alternatives. He looks open source ILS viability from four perspectives: market acceptance, support options, product development and functionality.