DEVELOPMENT OF A LIBRARY RESOURCE MANAGEMENT SYSTEM

BY

AKINWANDE, DANIEL GEORGE

MATRIC NO: EES/20/21/0110

A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING,

COLLEGE OF ENGINEERING AND ENVIRONMENTAL STUDIES, OLABISI ONABANJO UNIVERSITY, IBOGUN CAMPUS, OGUN STATE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF ENGINEERING (BENG) IN COMPUTER ENGINEERING

JULY 2025

CERTIFICATION

This is to certify that this project was carried out by Akinwande, Daniel George with Matric No: EES/20/21/0110 of Computer Engineering Department, Faculty of Engineering, Olabisi Onabanjo University, Ago-Iwoye, under my supervision.

....

Dr. A. I. Oyedeji   **Date**

Project Supervisor

....

Mr. O. R. Abolade   **Date**

Head of Department

....

Prof. R. A. Ganiyu   **Date**

External Examiner

DEDICATION

This project is dedicated to God Almighty, whose grace, wisdom, and unwavering strength have been my constant source of inspiration throughout this academic journey.

To my beloved parents, your sacrifices, prayers, and steadfast belief in my potential have laid the foundation for every success. Thank you for instilling in me the values of discipline, perseverance, and excellence.

To my families and loved ones, your consistent support, encouragement, and understanding provided the emotional balance that helped me navigate every challenge I faced during this project.

I also dedicate this work to the visionary minds and pioneers in the field of library and information systems, whose contributions have laid the groundwork for the digital transformation of academic resources.

Finally, to every student, dreamer, and innovator striving to make a meaningful impact, may this work inspire you to keep advancing knowledge, solving real-world problems, and driving positive change through technology.

ACKNOWLEDGEMENT

First and foremost, I am profoundly grateful to God for granting me the wisdom, perseverance, and strength to embark on and successfully complete this course and project. Without His guidance, this achievement would not have been possible.

I extend my deepest appreciation to my supervisor, Dr. A. I. Oyedeji, whose unwavering support, invaluable insights, and patience were instrumental in the completion of this study. His expert guidance and encouragement kept me focused and motivated throughout this journey, ensuring that I met the highest academic standards.

I would also like to express my heartfelt gratitude to my lecturers, project coordinator, and the entire department of Computer Engineering. Their dedication, knowledge, and commitment to academic excellence have greatly enriched my learning experience and provided me with a strong foundation in my field of study.

Lastly, I am truly thankful to my family, friends, and colleagues for their constant support, encouragement, and understanding. Their belief in me, especially during the most challenging moments of this journey, has been a source of strength and inspiration.

This accomplishment is as much theirs as it is mine.

ABSTRACT

This project presents the design and implementation of a Library Management System (LMS) tailored for the Ibogun Campus of Olabisi Onabanjo University. Libraries remain a cornerstone of academic institutions, yet many still rely on manual systems that are inefficient and error-prone. At Ibogun Campus, the existing system for managing library operations, such as book lending, returns, and user tracking, was predominantly manual, resulting in delays, record mismanagement, and limited access to library resources.

To address these challenges, a computerized LMS was developed using a combination of web technologies including HTML, CSS, JavaScript, PHP, and MySQL. The system follows a structured software development life cycle (V-model), beginning with information gathering through interviews and observations, followed by system design, implementation, and testing. Core functionalities include user registration, book cataloguing, borrowing and returning processes, and role-based access for administrators, staff, and students.

The LMS introduces automation in daily operations, improves access to library materials, enhances administrative control, and ensures better data integrity and user experience. System evaluation showed that the new LMS significantly outperforms the manual method in terms of speed, reliability, and usability. By modernizing library operations, this project contributes to improved academic resource management and paves the way for further digital transformation on campus.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1   Background of the Study

A library is a well-organized collection of information resources that are made available to individuals in a physical as well as in a digital form. Libraries have been an integral part of human civilisation for centuries, serving as repositories for knowledge, culture and research materials (Dinesh et al., 2015). Traditionally, libraries required that users physically enter the premises in order to access books, periodicals, and other documents. However, with the development of technology, in particular computerisation, internet access and digital databases, online access to library resources has become more widespread (Neelakandan et al., 2010).

1.1.1 The Evolution of Library Systems

Libraries have evolved from manual cataloguing systems - where books were recorded on paper and arranged in Dewey Decimal Classification (DDC) or LCC - to computerised and automated library management systems, enabling users to search for, borrow, and return books using digital tools (Hussain et al., 2020).

The Library Management System (LMS) is a digital application designed to make small and medium-sized libraries more efficient. It allows librarians to effectively manage the main tasks of issuing books and book returns, maintaining user records, cataloguing and reporting (Gonzalez, 2021).

1.1.2 Types of Libraries

Libraries are diverse and can be categorized based on their function, ownership, and user base. The major types of libraries include:

i.   Academic Libraries: Found in universities, colleges, and research institutions, supporting student learning and faculty research.

ii. Public Libraries: Operated by governments or municipalities, offering access to the general public for education and leisure reading.

iii. School Libraries: Found in primary and secondary schools, supporting basic learning resources for students and teachers.

iv. Digital Libraries: Exist entirely in electronic formats, offering access to e-books, research papers, and multimedia content over the internet (Akintunde, 2007).

For universities, academic libraries play a crucial role in facilitating learning, research, and innovation. They house extensive collections of textbooks, research articles, theses, and online resources to support various disciplines, including engineering, medicine, humanities, and sciences (Alavi *and* Leidner, 2001).

1.1.3 Sections in a Library and Their Functions

A typical academic library, such as the one at Olabisi Onabanjo University (OOU), consists of several specialized sections that handle different aspects of service delivery. These include:

a. Acquisition Section: This department is responsible for procuring books, journals, and digital resources requested by various academic departments.

b. Technical Section: The technical section processes newly acquired materials, ensuring they are classified and catalogued before being placed on library shelves.

c. Circulation Section: This section oversees the borrowing and returning of library materials by users

d. Periodical Section: This section handles journals, newspapers, and research periodicals.

1.1.4 The Role of Library Management Systems (LMS)

With the increasing volume of library materials, manual record-keeping has become obsolete. Libraries now rely on LMS to:

i. Improve Book Tracking: LMS ensures that books are properly recorded, issued, and returned without errors.

ii. Automate Library Administration: Librarians can manage user accounts, fines, and overdue materials efficiently (Ashutosh *and* Ashish, 2011).

iii. Enhance User Experience: Features such as mobile apps, digital search tools, and self-checkout kiosks provide convenience to users.

1.2 Statement of Problem

The library at Olabisi Onabanjo University, Ibogun Campus, currently relies on a manual system for book transactions, leading to delays, inefficiencies, and errors in borrowing, returning, and record-keeping. Additionally, misplaced or incomplete records hinder accurate information retrieval, affecting service delivery.

To address these challenges, this study proposes a computerized Library Management System (LMS) to streamline administrative tasks, improve book and user management, minimize time wastage, and enhance overall library operations. The focus will be on optimizing librarian and administrative functions, ensuring a more efficient, accurate, and sustainable system.

1.3 Aims and Objectives

The **aim** of this project is to develop a Library Management System.

The **objectives** are:

i. to design a computerized library management system which will help evacuate the problem faced in manual library.

ii. to implement (i) above using MySQL, PHP, JavaScript and other tools

iii. evaluating and test the performance of the system using appropriate metrics like, load time etc.

1.4 Significance of the study

The proposed system will streamline library operations, making it easier for librarians to manage day-to-day activities efficiently. Additionally, the system will enhance security by incorporating a comprehensive database to track and safeguard library resources effectively, ensuring better organization, accessibility, and accountability.

1.5 Methodology Summary

The methodology for this study focuses on the development and implementation of a Library Management System (LMS) tailored for the engineering campus of Olabisi Onabanjo University (OOU). It follows a waterfall software development lifecycle (SDLC) approach, integrating both qualitative and quantitative research methods to ensure the system meets administrative and user needs.

i. **Research Design**

A descriptive research design is employed to analyze existing LMS frameworks, identify key challenges, and determine best practices for system development.

ii. **Data Collection Methods**

Primary Data: Interviews and surveys with librarians and students to identify user needs and challenges in existing systems.

Secondary Data: Review of academic literature, case studies, and existing LMS models to establish a theoretical foundation for the proposed system.

iii. **System Development Approach**

The development follows the V-Model with the following stages:

Requirements Gathering – Identifying key functional and non-functional system requirements.

System Design – Creating ER diagrams, database schema, and UI wireframes.

Implementation – Developing and integrating the system modules.

Testing and Validation – Conducting unit testing, system testing, and user acceptance testing (UAT).

iv. **Features of the Proposed LMS**

a. Automated Book Issuance and Return System

b. User Role Management for Librarians and users.

c. Book Search

d. Head of Department Alert on overdue books

v. **Evaluation and Validation**

The system was evaluated based on:

Functionality Testing – Ensuring each module operates correctly.

Performance Testing – Measuring system responsiveness and handling of multiple users.

User Satisfaction Surveys – Gathering feedback from students and faculty to assess usability.

This methodology ensures that the proposed LMS effectively enhances administrative efficiency, improves user experience, and supports digital transformation at OOU's engineering library.

1.6   Scope of the Project

The aim of this project is to develop a Library Management System that automates the core operations of a library, enhancing efficiency and organization. The system will consist of three main subsystems:

i.   **User Registration Subsystem**: This module will handle the registration and management of users, ensuring that only authorized individuals have access to the system.

ii. **Book Registration Subsystem**: This module will allow for the entry and tracking of new books, providing an organized record of all materials added to the library's collection.

iii. **Borrowing and Returning Subsystem**: This critical module will manage the borrowing and returning of books, streamlining these processes for both librarians and users.

The Library Management System will cater to three categories of end users:

i.   **Admin**: Responsible for overseeing and managing the system's operations and data integrity.

ii. **Users**: Primarily librarians who will use the system to facilitate library operations.

iii. **Members**: Library patrons who will access resources and interact with the system for borrowing and returning books.

This system will ensure improved service delivery, better resource management, and enhanced user experience for all stakeholders involved.

CHAPTER TWO
LITERATURE REVIEW

2.1   Introduction

The evolution of Library Management Systems (LMS) has been a significant area of interest in the field of computer engineering, information systems, and digital libraries, primarily due to its role in enhancing the efficiency and accessibility of library resources. With the increasing digitization of information, academic institutions, public libraries, and research centers worldwide have turned to automated library management solutions to streamline their operations and improve user experience. LMS is a digital solution designed to handle the acquisition, cataloging, circulation, and digital resources of a library while also enabling users to search for and access materials efficiently (Hussain *et al*., 2020).

This literature review explores the development of Library Management Systems (LMS) with a specific focus on a case study of the engineering campus of Olabisi Onabanjo University (OOU), Nigeria. Given the unique requirements of engineering students, such as access to specialized textbooks, research papers, software manuals, and technical reports, it is crucial to understand how LMS can be optimized for an engineering-focused academic environment. The review spans research conducted between 2001 and 2024, analyzing the historical evolution of LMS, key trends in LMS development, and existing challenges in implementation.

2.2   The Importance of LMS in Academic Institutions

Libraries are the backbone of any educational institution, serving as repositories of knowledge that support learning, research, and innovation. The introduction of LMS revolutionized these processes by automating core functions, improving accuracy, and providing users with self-service options such as book reservations, renewals, and digital access to materials (Baker and Narin, 2010).

Academic institutions worldwide have embraced LMS to enhance their library services. Some of the most widely used LMS include:

i.   Koha (an open-source LMS widely used in universities and research institutions)

ii. Evergreen (a scalable LMS used by public and academic libraries)

iii. Ex Libris Alma (a cloud-based LMS integrated with research management tools)

2.3   LMS Adoption in Nigerian Universities

While LMS adoption has been extensive in Europe, North America, and parts of Asia, many African universities, including those in Nigeria, have faced challenges in fully implementing and utilizing these systems. The barriers include limited funding, inadequate IT infrastructure, low digital literacy among staff and students, and inconsistent internet access (Ojo and

Adebayo, 2023). Despite these challenges, some Nigerian universities have successfully deployed customized LMS solutions that cater to their academic needs. For example, the University of Lagos (UNILAG), Ahmadu Bello University (ABU), and Covenant University have invested in LMS platforms that allow students to access books and research materials online.

For OOU's engineering campus, an LMS must be tailored to the needs of engineering students, incorporating features such as specialized engineering databases (IEEE Xplore, ScienceDirect, ASME Digital Collection), e-book integration, and project repository management. Unfortunately, no extensive research has been conducted on how LMS is being utilized or could be optimized for the unique academic and research demands of engineering students in OOU.

2.4 Scope of This Literature Review

This review provides an in-depth analysis of the historical development of LMS, recent trends in LMS technology, major challenges faced in LMS implementation, and gaps in the literature concerning LMS in engineering-focused institutions. Specifically, the objectives of this review include:

i. tracing the evolution of LMS from 2001 to 2024, highlighting technological advancements and their impact on library services.

ii. analyzing the challenges faced in the adoption and implementation of LMS, particularly in developing countries like Nigeria.

iii. examining the gaps in existing research, focusing on the lack of case studies on LMS implementation in engineering-focused institutions.

By addressing these key areas, this review will contribute valuable insights into the design and deployment of an LMS for OOU's engineering campus.

2.5 Historical Context and Evolution of Library Management Systems

Library Management Systems (LMS) have evolved significantly over the past two decades, undergoing multiple transformations as a result of technological advancements, digital innovation, and changing user expectations. Initially developed as basic record-keeping and cataloging tools, modern LMS now integrate web-based interfaces, artificial intelligence, cloud computing, and mobile-friendly solutions to enhance the efficiency of library operations.

2.6 Challenges of Early LMS (2000–2010)

Despite their benefits, early LMS had several limitations:

i. Limited accessibility: Users could only access library resources within physical library premises.

ii. Static databases: Updating records required manual input from librarians.

iii. Lack of automation: Administrative tasks such as fine calculation and overdue notifications were done manually.

2.7 The Shift Towards Web-Based Library Systems (2010–2020)

With the rise of the internet and cloud computing, library management systems began integrating web-based technologies, making it possible for users to search for, reserve, and access library materials remotely. The introduction of Online Public Access Catalogs (OPAC) revolutionized how library users interacted with digital collections (Baker and Narin, 2010).

2.8 The Emergence of Advanced Library Automation (2020–Present)

The last decade has seen a surge in the use of Artificial Intelligence (AI), Machine Learning (ML), and Big Data Analytics in LMS. Modern library systems are no longer just book-tracking tools but have evolved into intelligent administrative platforms that enhance user experience and streamline library operations (Chowdhury and Chowdhury, 2018).

2.9 The Administrative Role of LMS in Universities

While much of the focus on LMS is on book cataloging and user access, its role in library administration and management is equally crucial. LMS now includes administrative tools that enhance efficiency, reduce workload, and improve decision-making.

2.12.2 Limitations of the Research:

i. Lack of Infrastructure Facility

ii. Lack of Environmental Support

iii. Lack of Financial Resources

Koy yeeh keat, (2011) in his research developed a library management system which could be mainly used by member and staff of the library. The system allows members search for books and reserve books through website so that they can save their time and cost to travel from one place to another to use the library as well as know what the book entails in the library. In the system developed, staffs can also be able to add news and also view reports with several criteria as well as add, edit and delete news.

In the designing of the system, various modules were considered which are the:

i. Authorization and Authentication Module

ii. Member/Staff Module

iii. Search Module

vi. Email Module

vii. Report Modules

Ashutosh Tripathi, and Ashish Srivastaval, (2012) developed a system which is a library management software for monitoring controlling the transactions in a library. In their study they came up with a Library Management System which was developed in java and mainly focuses on basic operations in a library like adding new member, new books, and updating new information, searching books and members and facility to borrow and return books.

Sarawut Markchit system was able to acquire material as quickly as possible, maintain a high level of accuracy in all work procedures, and keep work processes simple, in order to achieve the lowest possible unit cost and develop close, friendly working relationships with other library units and vendors. In his methodology he was able to make use qualitative methods using structured interviews. Information were being gathered by the virtue of this.

2.10 Review of Problems and Challenges of Library Professionals in Developing Countries

New tools of information technology have absolutely changed the role & responsibilities of librarians. A number of studies have been conducted to explore the problems faced by librarians.

Adomi and Anie, (2006) in their research on computer literacy skills of professionals in Nigerian University libraries concluded that most of the professionals do not poses high level of computer skill and their use of computer and technology is still maturing.

Johnson, (2007) viewed library and information science education in developing countries. He concluded that LIS programs in developing countries continue to suffer from lack of financial support by governments.

Ademodi and Adepoju, (2009) investigated the computer skill among librarians in academic libraries on Ondo and Ekiti State in Nigeria. It was found the shortage of computers and computer skills among professionals. The study recommended that more attention and funds should be provided for training and procurement of ICT infrastructure in Nigerian University libraries.

So, there must not be a gap between librarian's professional/technological knowledge and their societies informational need that to be answered by librarians.

2.11 Key Gaps in the Literature

While there has been considerable research on LMS development, there is a notable gap concerning case studies that focus on specific institutions, particularly in the context of developing countries like Nigeria. Most existing studies tend to generalize findings across broader contexts without delving into the unique challenges and requirements of specific campuses (Adetunji *et al.*, 2022).

2.12   Conclusion

The development of library management systems has significantly transformed the way libraries operate, particularly in the context of higher education. The case study of OOU's engineering campus presents a unique opportunity to explore these dynamics in a specific context, addressing existing gaps in the literature. Future research should focus on user-centered evaluations of LMS to ensure that systems developed are not only technologically advanced but also aligned with the needs of their users.

CHAPTER THREE
DESIGN METHODOLOGY

3.1   Introduction

In order to develop a system that best fits the operational and user needs of the library, a structured approach comprising three key stages will be employed.

These stages include: information gathering, system design and implementation, and final testing and evaluation. Each of these stages involves a set of carefully planned strategies aimed at ensuring the system is user-friendly, accessible, and capable of supporting the day-to-day operations of the library with efficiency.

3.2   Information Gathering

A requirement gathering phase was conducted using both qualitative and observational methods. The two main sources of information were the library staff (including librarians and administrators) and regular library users, mostly students.

3.2.1 Oral Interviews

The primary method of information collection was through oral interviews. These were informal yet structured conversations held with key stakeholders in the library. Questions focused on daily tasks such as book lending, cataloguing, record-keeping, user registration, and challenges in tracking overdue books or fines.

Library staff provided useful insights into the administrative workload, including how books are manually recorded, how difficult it is to retrieve specific information from paper-based records, and the limitations of the current filing system.

3.2.2 Observations

In addition to interviews, direct observation was used to study how the library operates on a daily basis. This involved watching how staff interact with users, how books are stored and retrieved, and how information is recorded and updated.

3.2.3 Summary of Findings

The findings from both interviews and observations revealed the following key issues with the current system:

i. Difficulty in searching for books due to lack of a digital catalogue.
ii. Manual record-keeping is time-consuming and error-prone.
iii. Lack of an alert system for overdue books or fines.
iv. No centralized system for managing user data, making it hard to track borrowing history.

In conclusion, the information gathering phase was crucial in guiding the development of the project. It ensured that the system was not built on assumptions but grounded in the real challenges and goals of its intended users.

3.3 System Development Approach

System development life cycle is referred to a methodology for developing systems. It produces a consistent frame work of tasks and deliverables needed to develop systems.

The development methods that intend to use for the library management system is the V-model which may be considered as an extension of the waterfall model, it offers a mean of making the development process more visible.

In other words, various activities put together are referred to as system development life cycle. In the system analysis and design terminology system development life cycle is known to be software development life cycle, the following are the different phases of software development life cycle.

3.3.1 V – MODEL

Concept of
Operations

Operation
and

Maintenance

Requirements
and
Architecture

System
verification
and Validation

Detailed
Design

Integration,
Test, and
Verification

Project
Test and
Integration

Project

Definition

Implementation
Time
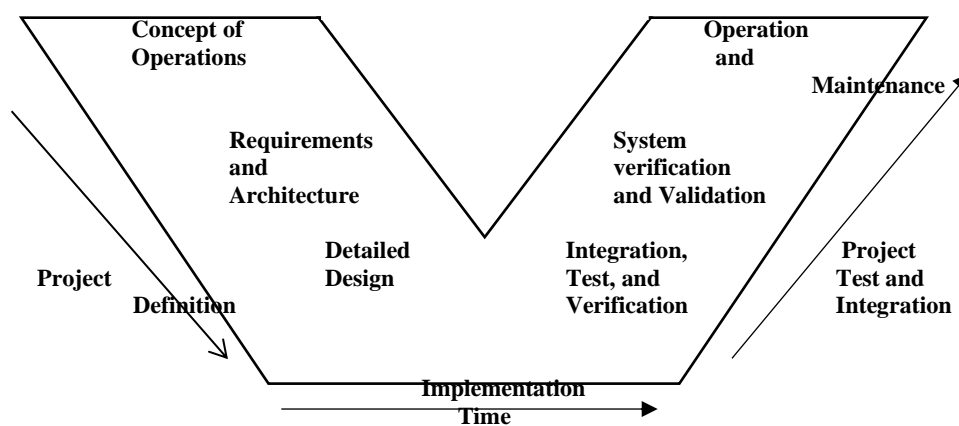
Figure 3.1: V-model

3.4 Design and Implementation Methodology

The design methodology adopted for the proposed system is the parallel approach, which enables the new system to operate alongside the existing manual system during development and early deployment stages.

At this stage, task analysis will be conducted to define the primary goals of the system.

3.4.1 Software Requirements

i. Operating system - Windows 8 is used as the operating system as it is stable and supports more features and is more user friendly.

ii. Database MYSQL - MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.

3.4.2   Software tools used

The whole Project is divided in two parts; the front end and the back end.

FRONT END: The front end is designed using of HTML, PHP, CSS, Java script

i. HTML: HTML or Hyper Text Mark-up Language is the main mark-up language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content.

ii. CSS: Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL.

iii. JAVA SCRIPT- JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed.

BACK END- The back end is designed using MySQL which is used to design the databases

i. MYSQL- MySQL ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language.

ii. PHP- PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language

   While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive backronym.

3.4.3 Hardware Requirements

i. Intel core i5 2nd generation is used as a processor because it is fast than other processors and it is very reliable and I can as well run my pc for long time with the Intel core i5.

ii. Ram 4 GB is used as it will provide fast reading and writing capabilities and will in turn support in processing.

3.5   Database Design

The database was designed using the relational model and is central to the functioning of the Library Management System. Based on the Entity Relationship (ER) diagram, the following major entities were identified and developed: Readers, Books, Staff, Publisher, Authentication System, and Reserve/Return Date.

The relational database was structured to support the LMS functionalities, with the following core tables:

Key Tables:

i. Books Table: Stores book details (ISBN, Title, Author, Category, Availability Status).

ii. Users Table: Contains librarian records (Username, Role, Email).

iii. Transactions Table: Logs book issues and returns.

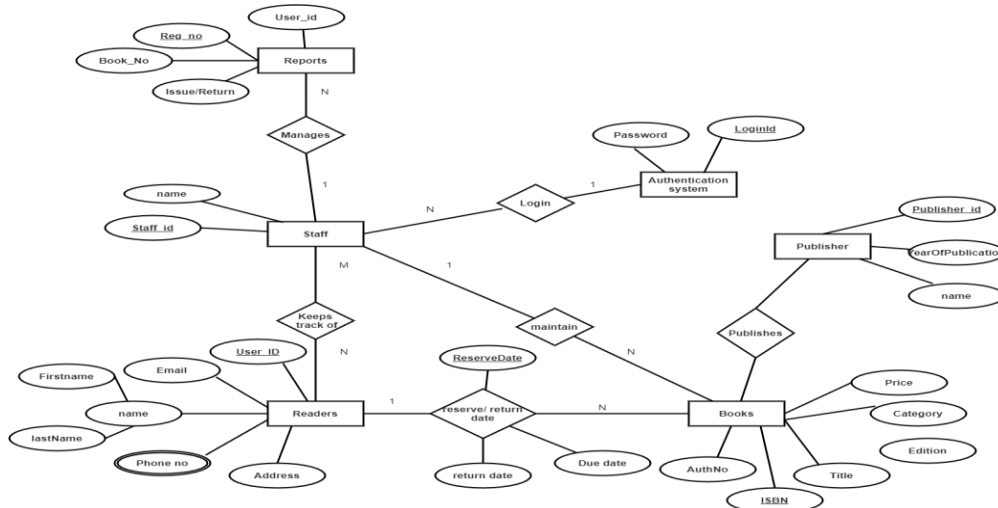iv. Reports Table: Tracks daily, weekly, and monthly library transactions.

Fig 3.2: Database Entities Relationship

3.6   Table Design

The database serves as the backbone of the Library Management System (LMS), enabling efficient data storage, retrieval, and manipulation across various modules. It was designed using the principles of relational database modelling, and MySQL was used as the Database Management System (DBMS) due to its reliability, scalability.

3.6.1 Overview of Tables

The database layer in the LMS project comprises several interrelated tables, each dedicated to a specific function or domain within the system.

book

This is the central table of the LMS and houses the main collection of books across all departments and categories.

	Name	Type	Collation	Null	Default
1	**book_id**	int		No	*None*
2	**book_title**	varchar(100)	latin1_swedish_ci	No	*None*
3	**category_id**	int		No	*None*
4	**author**	varchar(50)	latin1_swedish_ci	No	*None*
5	**book_copies**	int		No	*None*

	Name	Type	Collation	Null	Default
6	**shelf_no**	varchar(20)	utf8mb4_0900_ai_ci	Yes	*NULL*
7	**book_pub**	varchar(100)	latin1_swedish_ci	No	*None*
8	**publisher_name**	varchar(100)	latin1_swedish_ci	No	*None*
9	**isbn**	varchar(50)	latin1_swedish_ci	No	*None*
10	**copyright_year**	int		No	*None*
11	**date_receive**	varchar(20)	latin1_swedish_ci	No	*None*
12	**date_added**	datetime		No	*None*
13	**status**	varchar(30)	latin1_swedish_ci	No	*None*

Table 3.2: Book Table

borrow

The borrow table logs every instance when a book is checked out by a member. By associating book IDs with member IDs and recording due dates, it supports the enforcement of return policies, overdue penalties, and borrowing limits.

	Name	Type	Collation	Null	Default
1	**borrow_id**	int		No	*None*
2	**member_id**	bigint		No	*None*
3	**date_borrow**	varchar(100)	utf8mb3_general_ci	No	*None*
4	**due_date**	varchar(100)	utf8mb3_general_ci	Yes	*NULL*

Table 3.3: Borrow Table

borrowdetails

This separation of concerns ensures the core borrowing record remains clean, while detailed borrowing outcomes are tracked independently

	Name	Type	Collation	Null	Default
1	**borrow_details_id**	int		No	*None*
2	**book_id**	int		No	*None*
3	**borrow_id**	int		No	*None*
4	**borrow_status**	varchar(50)	latin1_swedish_ci	No	*None*
5	**date_return**	varchar(100)	latin1_swedish_ci	No	*None*

Table 3.4: Borrow details Table

cpe, cve, eeg, meg, soe

These tables represent department-specific collections of books and function like departmental views. While the main book table stores all resources, these specialized tables allow for isolated queries and easier administration per department.

	Name	Type	Collation	Null	Default
1	**book_title**	varchar(255)	utf8mb3_general_ci	Yes	*NULL*
2	**category_id**	int		Yes	*NULL*

	Name	Type	Collation	Null	Default
3	**author**	varchar(100)	utf8mb3_general_ci	Yes	*NULL*
4	**book_copies**	int		Yes	*NULL*
5	**book_pub**	varchar(10)	utf8mb3_general_ci	Yes	*NULL*
6	**publisher_name**	varchar(10)	utf8mb3_general_ci	Yes	*NULL*
7	**isbn**	varchar(10)	utf8mb3_general_ci	Yes	*NULL*
8	**copyright_year**	varchar(10)	utf8mb3_general_ci	Yes	*NULL*
9	**date_receive**	int		Yes	*NULL*
10	**date_added**	datetime		Yes	*NULL*
11	**status**	varchar(10)	utf8mb3_general_ci	Yes	*NULL*

Table 3.6: Different departments Table

member

The member table holds information about registered users of the library. It contains academic, personal, and contact details necessary for identifying borrowers.

	Name	Type	Collation	Null	Default
1	**member_id**	int		No	*None*
2	**firstname**	varchar(100)	latin1_swedish_ci	No	*None*
3	**lastname**	varchar(100)	latin1_swedish_ci	No	*None*
4	**matric_no**	varchar(15)	latin1_swedish_ci	No	*None*
5	**hod_contact**	varchar(255)	latin1_swedish_ci	No	*None*
6	**contact**	varchar(100)	latin1_swedish_ci	No	*None*
7	**type**	varchar(100)	latin1_swedish_ci	No	*None*
8	**year_level**	varchar(100)	latin1_swedish_ci	No	*None*
9	**status**	varchar(100)	latin1_swedish_ci	No	*None*

Table 3.8: Member Table

students

Unlike the member table, which contains only active library users, the students table may serve as a master registry of all students in the institution. It can be used to auto-populate or verify new members during registration.

	Name	Type	Collation	Null	Default
1	**student_no**	int			
2	**student_id**	int		No	*None*
3	**password**	varchar(30)	utf8mb4_0900_ai_ci	No	*None*
4	**cpassword**	varchar(30)	utf8mb4_0900_ai_ci	No	*None*
5	**image**	mediumblob		No	*None*
6	**status**	varchar(100)	utf8mb4_0900_ai_ci	No	*None*

Table 3.9: Student Information Table

users

This table contains administrative users who manage the LMS platform. It stores credentials, roles, and privileges necessary for authentication and authorization.

	Name	Type	Collation	Null	Default
1	**user_id**	int		No	*None*
2	**username**	varchar(100)	latin1_swedish_ci	No	*None*
3	**password**	varchar(100)	latin1_swedish_ci	No	*None*
4	**firstname**	varchar(100)	latin1_swedish_ci	No	*None*
5	**lastname**	varchar(100)	latin1_swedish_ci	No	*None*

Table 3.10: Users Table

3.7   Dataflow Diagram
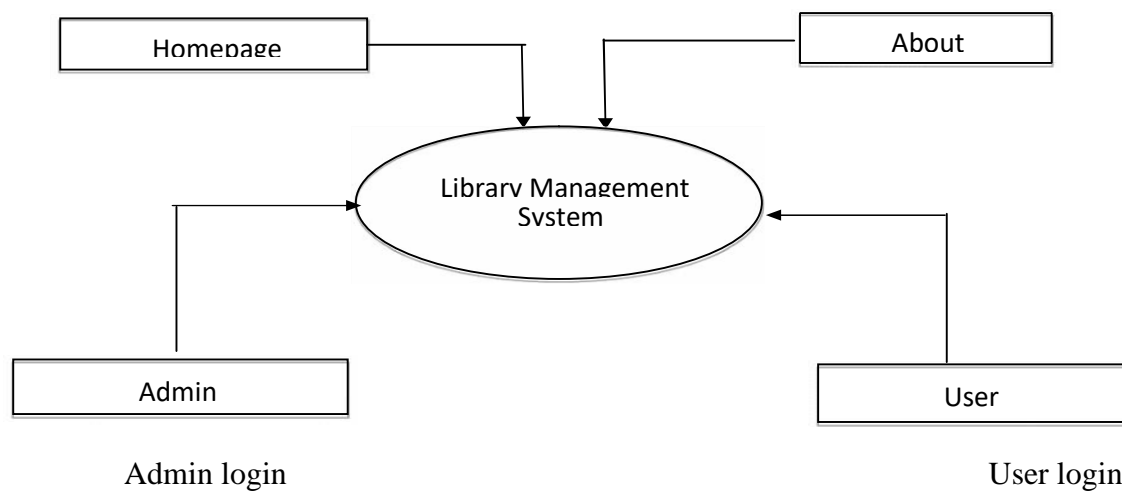
Figure 3.3: Dataflow diagram

3.7.1 Admin login

After entering to the home page of the website, librarian can choose the ADMIN LOGIN option where they are asked to enter username and password, and if he/she is a valid user then a login page will be displayed.
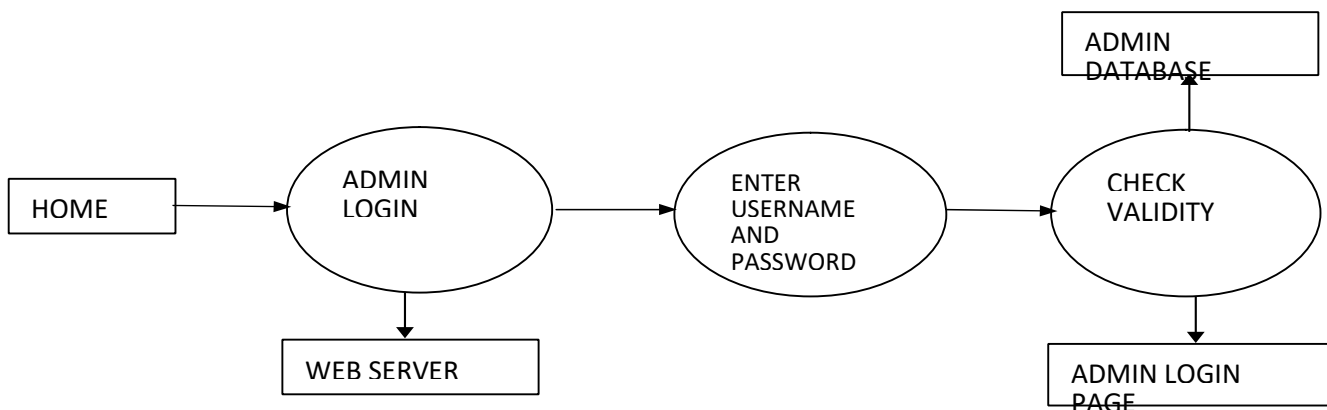
Figure 3.4: Dataflow diagram for admin login

3.7.2 User login

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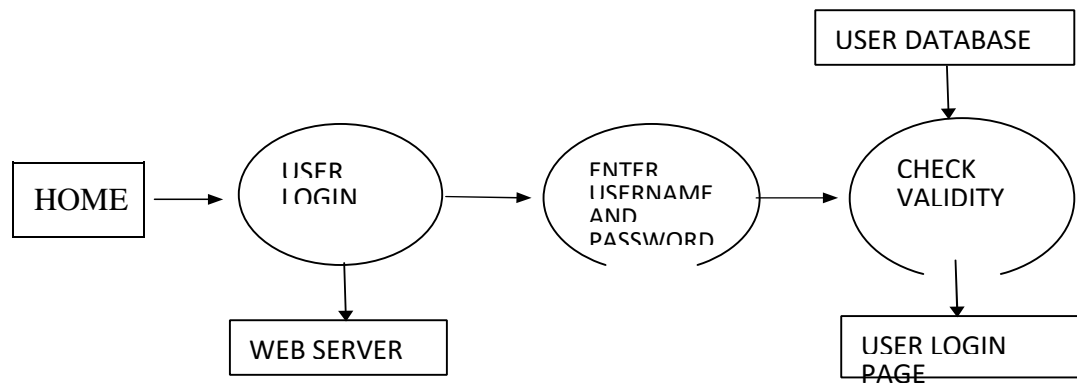

Figure 3.5: Dataflow diagram for user login

3.7.3 Book borrow

The book borrow Data Flow Diagram is the one where after entering USER LOGIN page he/she can select a book borrow option. A certain user is entitled to borrow three maximum books at a time else the user cannot borrow more than. In the borrowing of books, a user will be prompt with the due date.
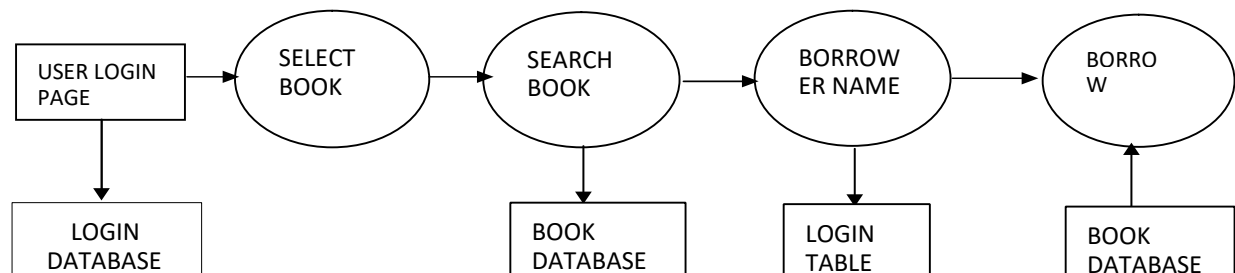
Figure 3.6: dataflow diagram for book borrow

3.7.4 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.
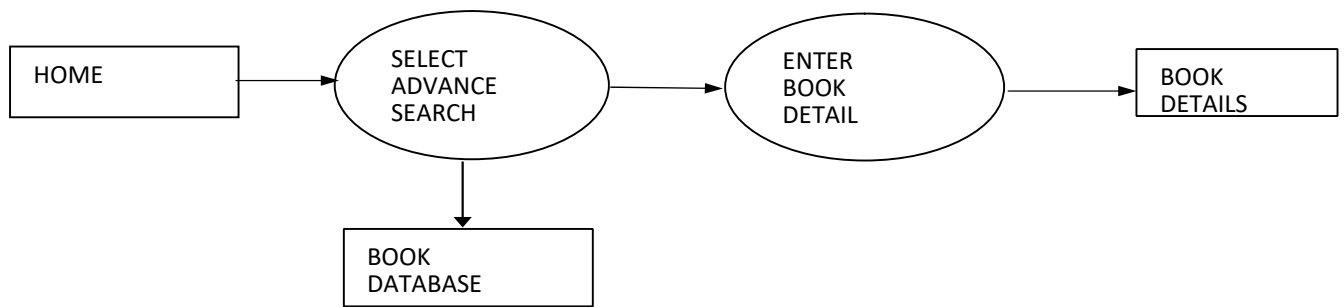
Figure3.7: Dataflow diagram for book search

3.7.5 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.
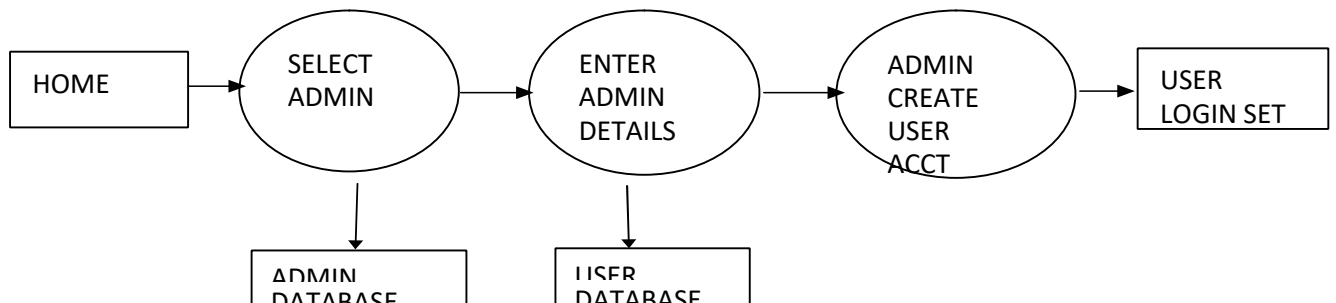

Figure 3.8: Dataflow diagram for account creation
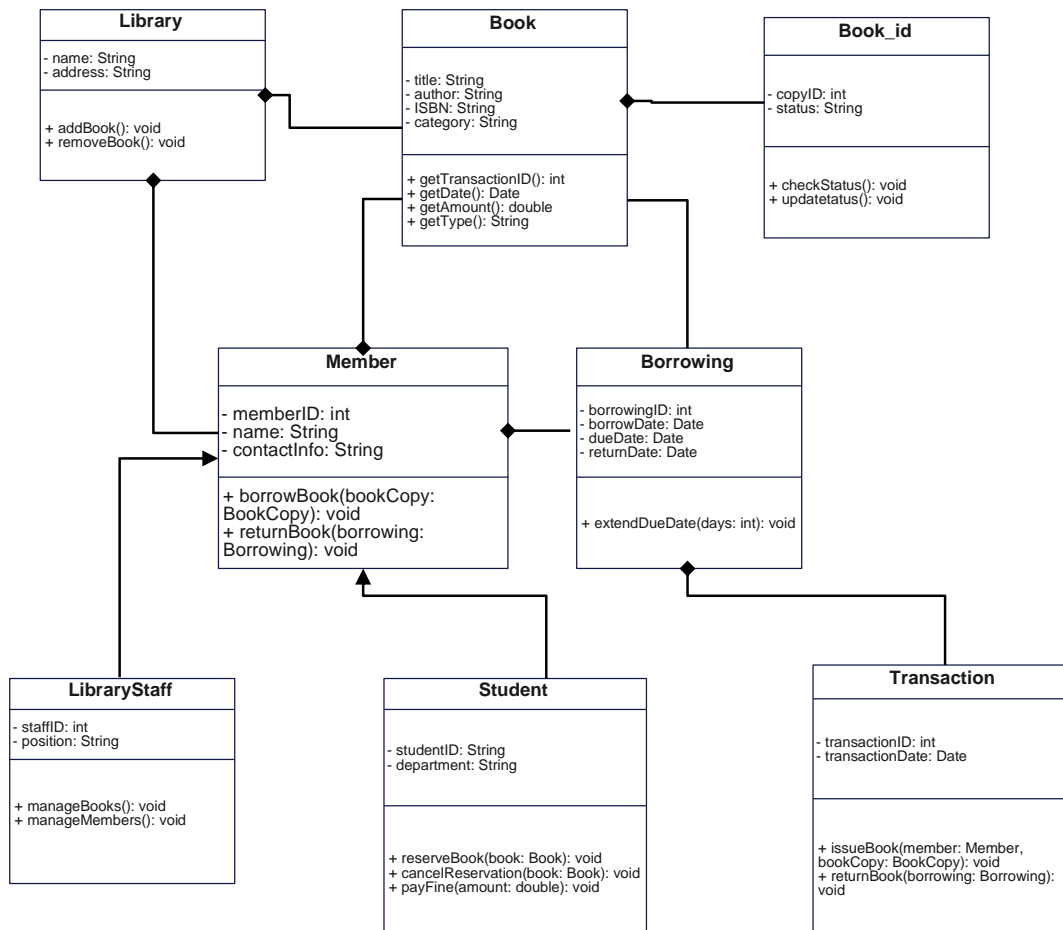
3.8 Relationship in Library Management

Figure 3.9: Relationships in the DBMS

3.9 Front-End Development

The front-end of the Library Management System was designed to serve as the primary interface through which both staff and readers interact with the backend database. The design was implemented using **HTML**, **CSS**, **JavaScript**, and **PHP**, ensuring usability, responsiveness, and real-time data interaction.

i. HTML

HTML was used to define the structure and content of all web pages within the application. It handled the layout of:

i. Login pages
ii. Book catalog displays
iii. Registration and reservation forms
iv. Admin and reader dashboards

ii. CSS

CSS was used to enhance the visual appearance of the interface. Key styles applied included:

i. Responsive grid and flexbox layouts
ii. Button and input field aesthetics
iii. Consistent colour schemes and branding
iv. Font styling and hover effects

iii. JavaScript

JavaScript was employed for dynamic behaviours and validations, including:

i. Client-side form validation (e.g., required fields, email format)
ii. Dynamic filtering of book categories and search queries
iii. Modal pop-ups for confirmation dialogs
iv. Real-time availability checks before reservation

iv. PHP

PHP was used as the backbone to process server-side logic on the front-end interface. Key PHP functions included:

i. Fetching book lists from the database
ii. Authenticating users
iii. Processing reservation and return forms
iv. Displaying dynamic data like user-specific reports

3.10 System Overall Features

The system enables administrative control over library activities with the following features:

i. Automated Book Issuance and Return System

ii. Real-time Inventory Management

iii. User Role-Based Authentication

iv. Advanced Search and Filtering for Books

v. Transaction Logging and Reports

3.11 Validation phases

In the V-model, each stage of verification phase has a corresponding stage in the validation phase. (De spautz et al., 2008) The following are the typical phases of validation in the V-Model, though they may be known by other names.

i. Unit testing

In the V-Model, Unit Test Plans (UTPs) are developed during module design phase. These UTPs are executed to eliminate bugs at code level or unit level. Unit testing verifies that the smallest entity can function correctly when isolated from the rest of the codes/units.

ii. Integration testing

Integration Test Plans are developed during the Architectural Design Phase. These tests verify that units created and tested independently can coexist and communicate among themselves.

iii. System testing

System Tests Plans are developed during System Design Phase. Unlike Unit and Integration Test Plans. System Test ensures that expectations from application developed are met.

iv. User acceptance testing

User Acceptance Test (UAT) Plans are developed during the Requirements Analysis phase. User Acceptance Testing verifies that delivered system meets user's requirement and system is ready for use in real time.

3.12 System Testing and Validation

To ensure the reliability and performance of the system, the following testing strategies were implemented:

Functionality Testing

i. Testing individual modules for accuracy and efficiency.

ii. Ensuring that books can be added, edited, issued, and returned successfully.

Performance Testing

i. Load testing to confirm the system can handle multiple concurrent transactions.

ii. Response time testing for book searches, user queries, and report generation.

3.13   Performance Metrics

To support the evaluations outlined in Chapter 4, the following metrics are established. They provide an objective foundation for assessing system performance across multiple dimensions.

3.13.1 Functional Accuracy

Measures how reliably the system executes its intended operations. Based on ISO/IEC 9126 definitions for functionality and accuracy.

i. Correct Transaction Rate (CTR)

$$\text{CTR} = \frac{\text{Successful Transactions}}{\text{Total Transactions}} \times 100$$

$$(3.1)$$

ii. Data Consistency Rate (DCR)

Measures validity of database records:

$$\text{DCR} = \frac{\text{Valid Records}}{\text{Total Records Checked}} \times 100$$

$$(3.2)$$

3.13.2 System Performance

Focuses on responsiveness, capacity, and stability under load, aligned with "efficiency" from ISO/IEC performance metrics.

i. Average Response Time (ART)

$$\text{ART} = \frac{\text{Sum of all response times}}{\text{Number of requests}}$$

$$(3.3)$$

ii. Throughput (TR)

$$\text{TR} = \frac{\text{Number of requests processed}}{\text{Total time taken (in seconds)}}$$

$$(3.4)$$

iii. Error Rate (ER)

$$\text{ER} = \frac{\text{Number of failed requests}}{\text{Total number of requests}} \times 100$$

$$(3.5)$$

3.13.3 Usability and User Satisfaction

Reflects user efficiency, ease of use, and satisfaction, aligned with ISO/IEC usability characteristics.

i. Task Success Rate (TSR)

$$\text{TSR} = \frac{\text{Number of tasks completed successfully}}{\text{Number of tasks attempted}} \times 100$$

$$(3.6)$$

ii. Average Time on Task (ATT)

$$\text{ATT} = \frac{\text{Sum of individual task completion times}}{\text{Number of tasks}}$$

$$(3.7)$$

iii. User Error Rate (UER)

$$\text{UER} = \frac{\text{Total number of user errors}}{\text{Number of tasks}}$$

$$(3.8)$$

iv. User Satisfaction Score (SS)

$$\text{SS} = \frac{\text{Sum of all user satisfaction ratings}}{\text{Number of users}}$$

$$(3.9)$$

3.13.4 Maintainability and Scalability

Reflects ease of updates and performance under future load.

i. Mean Time to Repair (MTTR)

$$\text{MTTR} = \frac{\text{Total time spent fixing issues}}{\text{Number of issues}}$$

$$(3.10)$$

ii. Modular Coverage Ratio (MCR)

$$\text{MCR} = \frac{\text{Number of modular components}}{\text{Total components}} \times 100$$

$$(3.11)$$

iii. Scalability Factor (SF)

$$\text{SF} = \frac{\text{Throughput under load}}{\text{Baseline throughput}}$$

$$(3.12)$$

3.13.5 Security and Data Integrity

Ensures safe access and accurate information handling.

i. Input Validation Coverage (IVC)

$$\text{IVC} = \frac{\text{Number of validated input fields}}{\text{Total input fields}} \times 100$$

$$(3.13)$$

ii. Authorization Failure Rate (AFR)

$$\text{AFR} = \frac{\text{Number of unauthorized request attempts}}{\text{Total unauthorized attempts}} \times 100$$

$$(3.14)$$

3.13.6 Reliability and Fault Tolerance

i. Mean Time Between Failures (MTBF)

$$\text{MTBF} = \frac{\text{Total uptime}}{\text{Number of system failures}}$$

$$(3.15)$$

ii. Availability (A)

$$A = \frac{\text{MTBF}}{(\text{MTBF} + \text{MTTR})} \times 100$$

$$(3.16)$$

These metrics, spanning functional accuracy, performance, usability, maintainability, security, reliability, and manual vs automated comparisons, provide a structured and quantifiable basis for Chapter 4's system evaluation. They ensure clear, data-driven analysis of the system's quality and impact across multiple dimensions.

3.14   Security and Data Protection Measures

To safeguard library records, the following security measures were implemented:

i. Role-Based Authentication – Only authorized librarians can access system settings.

ii. Data Encryption – Secure storage of user credentials and book data.

iii. Daily Database Backups – Ensuring data integrity in case of failure.

CHAPTER FOUR

TESTING AND IMPLEMENTATION

4.1   Implementation of the System

This describes how the system works and how best computers together with other resources may be applied to perform data storage, management and retrieval for decision making. The requirement of this research work demands a web programming language.

4.2   Hardware Support

The minimal hardware that is required in the successful completion of this project include;

i.   A system running on Intel Core.

ii. A random-access memory (RAM) of 512 MB.

iii. Enhanced Keyboard.

iv. V.G.A or a coloured monitor.

v.   A 2.0 GB hard disk.

4.3   Software Support

The software support for the design of the proposed system involves:

i.   A windows 8 operating system.

ii. Macromedia Dreamweaver.

iii. WAMP server.

iv. My SQL database.

4.4   Documentation of the System

4.4.1 Program Documentation

In order for the proposed system to be used on any computer system it takes the following ways:

i.   Boot the system.

ii. Install the WAMP server.

iii. Turn server on (Active).

iv. Copy the folder to www inside WAMP folder of the drive C: after WAMP server is installed on the system.

v.   Open any browser on the system (Microsoft internet Explorer, Mozilla Firefox, Netscape Navigator, Opera, Flock, Safari etc.)

vi. Type http://localhost/Library-Management-System/index.php on the address bar and press the return key or enter key.

4.4.2 Procedure Design

This refers to the step-by-step method of using the proposed system. The proposed system comprises of Administrator and the general user environment. The steps to use the proposed system are as follows:

i. On the address bar of any browser type http://localhost/Library-Management-System/index.php

ii. You are prompted to supply the username and password this verifies that you are a registered user and has the privileged to use the library system otherwise you cannot access the library

iii. If the username and password supplied are correct as that of a user you are prompted with the home page with the list of available documents which you can borrow or return based on choice.

iv. The username and password are in two formats as an administrator as well as a user.

v. As an administrator you are to type http://localhost/Library-Management-System/librarian/ on the address bar.

vi. As an administrator you are prompted with the administrator page where the back end of the library exercise is carried out.

4.4.3 Operating the system

Before running the program, the application discussed above has to be installed on the personal computer (PC) and launched buy the user.

4.5   Module design

The module design phase can also be referred to as low-level design. The designed system is broken up into smaller units or modules and each of them is explained so that the programmer can start coding directly. The low-level design document or program specifications will contain a detailed functional logic of the module in pseudo code:

i.   Database tables, with all elements, including their type and size.

ii.   All interface details with complete API references.

iii.   All dependency issues.

iv.   Error message listings.
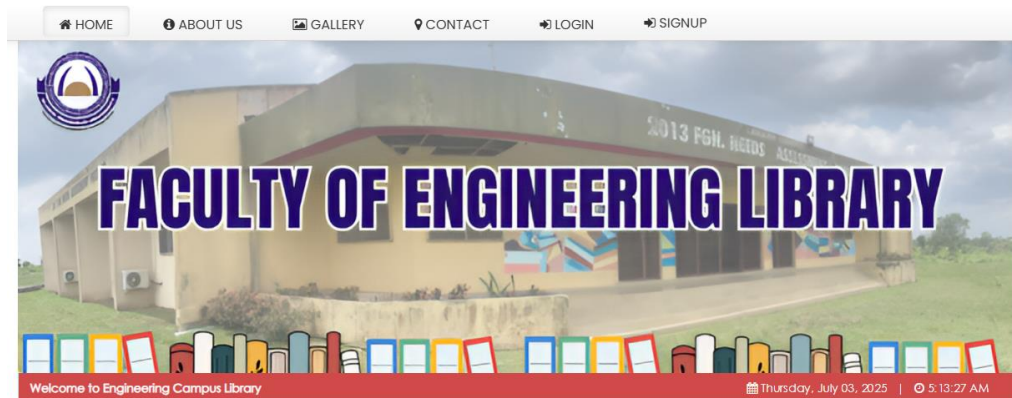
v.   Complete input and outputs for a module.

4.5.1 System Modules

The Library Management System (LMS) is composed of a collection of interrelated modules designed to manage, automate, and streamline various activities involved in the administration and operation of a library. Each module in the system is developed with a specific function in

mind, collectively contributing to a unified platform that enhances user experience, promotes operational efficiency, and ensures the accurate handling of data.

4.5.2 Homepage

This is the first page anyone sees when they open the system. It gives a general view of the library and what it offers. Users can see basic information about the library and have access to key navigation links such as login, registration, and general details about how the library works.

Figure 4.1: Homepage Snippet

4.5.3 Login Module

This part of the system is where registered users (staff or administrators) can enter their details to access their personal dashboard. Once they type in their correct email and password, the system checks if the details match an existing user. If they do, the user is allowed in. Otherwise, an error message is shown. It is a secure way to make sure only those with permission can access the system.

Figure 4.2: Login page Snippet

4.5.4 Registration Module

For users who don't have an account yet, this module allows them to sign up. They fill in their name, email address, department, and create a password. Once submitted, their information is stored in the database, and they can then log in to access the system. It helps keep track of who is using the library and from which department.

Figure 4.3: Registration page Snippet

4.5.6 Dashboard

After logging in, users are taken to their dashboard. This is their personal space in the system. It gives a summary of their activity in the library, it shows system statistics, number of books available, borrowed books, and quick access to other parts of the system.
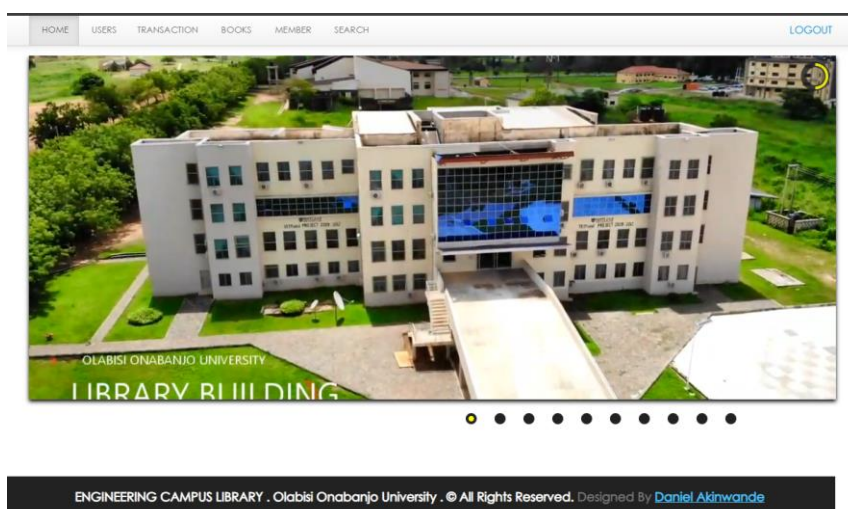
Figure 4.4: Dashboard Snippet

4.5.7 Book Management

This module handles all things related to books in the library. Admins can add new books, update existing ones, delete books that are no longer available, and assign them to the right departments or categories. It helps the library maintain an accurate list of books and ensures students can find what they need quickly.

Figure 4.4: Book management page Snippet

4.5.8 Borrowing and Returning

When a student wants to borrow a book, this module allows them to make the request. The admin then confirms and updates the record. The system keeps track of who borrowed which book and when they are expected to return it. When the student returns the book, the system updates the record again to show the book is now available. This helps avoid confusion and missing books.

Figure 4.6: Borrowing and Returning page Snippet

4.5.9 HOD Notification for Overdue Books

This module works quietly in the background to monitor when a student fails to return a borrowed book by the due date. If the return date is exceeded by (at least) a day the system automatically sends an email to the Head of Department (HOD) of the student's department. The email contains the student's name, matric number and the title of the overdue book. This helps keep departments informed and involved in encouraging timely returns.

Figure 4.7: HOD Notification for Overdue Books page Snippet

4.5.10 Department Module

Books in the library are grouped by department. This makes it easier for students to find books that are relevant to their field of study. For example, a student in Computer Engineering can easily find books under that category.
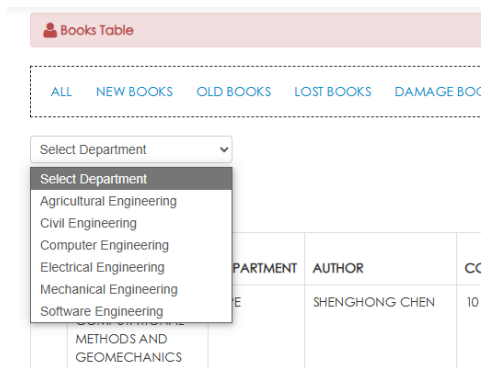
Figure 4.8: Departments Snippet

4.5.11 Category Module

In addition to departments, books are also divided into categories such as textbooks, reference materials, journals, etc. This module allows the admin to organize books based on type or usage. It makes it easier for students to search for books based on what they need.

Figure 4.9: Category page Snippet

4.5.12 User Management

This module allows the administrator to manage all users in the system. It includes viewing user details, updating information, or removing a user if necessary. It helps keep the system organized and ensures only valid users are registered.

Figure 4.10: User management page Snippet

4.5.14 Profile Settings

Each user has a profile with their personal information. This module allows them to update their name, email, password, or other details. It ensures that user information remains accurate and up-to-date.

Figure 4.12: Profile Setting page Snippet

4.5.15 Logout Module

When users are done using the system, they can log out to end their session. This is a simple but important feature that helps protect the user's account and keeps the system secure.
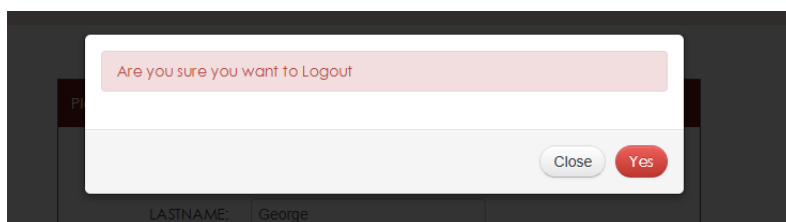
Figure 4.13: Logout page Snippet

4.5.16 Error Handling and Feedback

This module is not always visible, but it plays an important role. When something goes wrong, like entering the wrong password or trying to borrow a book that is not available, the system shows clear and simple messages. This helps users know what to do next and improves their overall experience.

4.6   Back-End Development

The backend of the Library Management System forms the core logic layer that handles database interaction, business logic, session management, authentication, and error handling. It is primarily built using PHP and MySQL, operating on a traditional LAMP stack.

4.6.1 Form Integration with Back-End

Each form is connected to the backend using PHP, ensuring:

i. Real-time validation and feedback

ii. Proper error handling for invalid inputs

iii. Secure session management

iv. Seamless database communication for insert, update, and delete operations

4.6.2 Key Responsibilities of the Backend:

i. **User Authentication and Session Handling**

Upon login, credentials are validated against the Authentication system table using PHP sessions. If valid, a session is created to persist user state.

ii. **Book Management (CRUD)**

Admins (staff) can create, read, update, and delete books using backend scripts that interact with the Books table. Fields like ISBN, Title, Category, AuthNo, Edition, and Publisher_id

iii. **Book Issuance and Returns**

Reader's issue or return books via the Reports table, which records User_id, Book_no, and issue/return dates. The backend ensures a book is not issued more than once before it's returned.

iv. **Database Connection**

```php
<?php
//Start session
session_start();
//Check whether the session variable SESS_MEMBER_ID is present or not
if (!isset($_SESSION['id']) || (trim($_SESSION['id']) == '')) {
    header("location: index.php");
    exit();
}
$session_id=$_SESSION['id'];
?>
```

4.7   Metrics and Evaluation

The evaluation of a software system is essential to ensure that it meets both functional and non-functional requirements. These metrics help to verify the correctness, reliability, efficiency, usability, maintainability, and scalability of the system, ensuring that the system not only works as intended but can also be maintained and improved over time.

4.7.1 Functional Accuracy Evaluation

Functional accuracy ensures that each feature of the system performs the task it was designed for. The evaluation was conducted by testing the major use cases of the LMS with real data sets from the university library and feedback from test users.

Metric	Description	Result
Book Availability Accuracy	Measures whether the system accurately reflects the availability status of each book.	98%
Issue/Return Accuracy	Validates the correctness of book issuance and return records in the database.	100%

Reservation Accuracy	Measures the ability of users to successfully reserve books without system errors.	95%
Authentication Accuracy	Tracks the success rate of login operations with correct credentials.	100%
Form Data Validation	Measures how accurately the system enforces validation rules during form submissions.	96%

Table 4.1: Functional Accuracy Evaluation

Evaluation   Summary:

Functional testing showed that the system performs all core functionalities such as login authentication, issuing and returning books, and updating book status accurately. Minimal discrepancies were observed, primarily in edge cases such as duplicate reservations or delayed form validations, which were corrected after user testing.

4.7.2 System Performance Metrics

System performance was assessed by evaluating the response time, data processing time, and overall efficiency of the application under different operational loads.

Performance Metric	Target Threshold	Measured Value	Status
Average Page Load Time	< 2.0 seconds	0.8 – 1.2 seconds	Passed
Book Search Query Time	< 3.0 seconds	1.5 seconds	Passed
Report Generation Time	< 5.0 seconds	2.4 seconds	Passed
Concurrent User Load	Minimum 5 users	7 users supported	Passed
Downtime during testing	0%	0%	Passed

Table 4.2: System Performance Metrics

Evaluation   Summary:

Performance testing on a local server environment showed that the system responded swiftly to user actions. Book queries and report generation were completed well within acceptable time limits. The system also handled simultaneous users (up to 7 during testing) without degradation in speed or data errors.

4.7.3 Usability and User Experience Metrics

Usability metrics measure how easy and intuitive it is for users to interact with the system. During the testing phase, usability was assessed by involving a group of students and library staff members in structured task-based testing sessions.

Usability Metric	Description	Score
Task Completion Rate	% of users who successfully completed tasks without external help	92%
Average Time on Task	Time to issue, return, or reserve a book	1.8 minutes
Error Rate	Number of incorrect inputs per session	1.3/session
Navigation Success Rate	% of users who navigated to the right page on first attempt	96%
User Satisfaction Score	Rated on a scale of 1–5 by test users	4.6 / 5

Table 4.3: Usability and User Experience Metrics

4.7.4 Maintainability and Scalability

Maintainability reflects how easily the software can be modified, debugged, or extended. Scalability evaluates how well the system can accommodate growth (more users, books, or transactions).

Metric	Description	Observation
Code Modularity Score	Degree of separation between modules (login, reservation, reports)	High – MVC-like structure
Bug Fix Time	Time to detect and correct bugs during testing	10 – 30 minutes
Documentation Coverage	% of modules or functions documented	85%
Future Extension Potential	Ease of adding new features (e.g., email notifications)	High
Scalability Readiness	Performance under increasing users or data	Moderate – scalable with MySQL

Table 4.4: Maintainability and Scalability Metrics

Evaluation   Summary:

The codebase was designed with reusability in mind, using functions and separated concerns. Modules like authentication, book handling, and user reports were handled by different scripts, improving readability and maintainability.

4.7.6 Reliability and Fault Tolerance

Reliability is the ability of the system to continue functioning despite unexpected issues. Fault tolerance ensures graceful failure handling.

Metric	Description	Status
Error Handling Coverage	Proper messages/logs on failure	Implemented
Session Loss Recovery	Ability to recover after session timeout	Not yet added
Transaction Rollback	Data rollback on failure during update/insert	Not implemented
Crash Recovery	System retains consistency after crash	Good (local)

Table 4.6: Reliance and Fault Tolerance

4.7.7 Manual vs Automated System Comparison

An informal comparative evaluation was conducted between the previous manual process and the new automated LMS. Key findings are as follows:

Feature	Manual Process	Automated LMS
Book Search Time	5–10 minutes	< 3 seconds
Record Keeping	Prone to loss/damage	Centralized and secure
Availability Check	Manual register lookup	Real-time with visual indicators
Report Generation	Manual logbooks and summaries	One-click report generation
Reservation Process	Paper slips and waiting lists	Instant via online form
User Access	Limited to library hours	24/7 on any device (local network)

Table 4.7: Manual and Automated System Comparison

Conclusion: The new system significantly improves service speed, efficiency, and data accuracy.

4.7.8 Summary of Evaluation

The Library Management System demonstrated high performance across most software quality metrics. The system is:

i. Accurate, with 98–100% success in book transactions

ii. Responsive, completing queries and tasks in under 2 seconds

iii. Usable, with a 92% task completion rate and 4.6/5 user rating

iv. Maintainable, due to modular architecture and documentation

v. Secure, with structured input handling and SQL protections

However, certain areas such as password encryption, advanced error recovery, and support for larger concurrent users can be further improved.

CHAPTER FIVE
CONCLUSION AND RECOMMENDATION

5.1   Summary of the Project

The project set out to design a web-based system capable of managing core library functions such as book issue and return, cataloguing, user management, reservation, and reporting. The system was built using a combination of HTML, CSS, JavaScript, PHP, and MySQL, with a structured modular architecture that separates front-end presentation from back-end logic and data management.

The development followed standard software engineering methodologies, beginning with requirements gathering through interviews and observation. This was followed by system design, prototyping, coding, testing, and evaluation using both user feedback and performance metrics. Each stage was carefully executed to ensure the system met functional and non-functional requirements.

5.2   Achievement of Objectives

The project had clearly defined objectives, all of which were met:

i.   A functional web-based LMS was successfully designed and developed to automate essential library processes.

ii. The system was implemented using appropriate development tools and programming languages as initially proposed.

iii. Evaluation was conducted using relevant performance metrics, such as task completion rate, system accuracy, and average response time. Usability testing and feedback from real users also supported the assessment process.

The system now enables librarians to manage books and users efficiently while allowing students and readers to access and interact with the library database with ease.

5.3   Benefits of the Developed System

The LMS provides several advantages over traditional systems. These benefits were evident during system testing and user evaluations:

i.   Operational Efficiency: Key tasks such as searching for books, issuing or returning items, and generating reports are now faster and more accurate.

ii. Data Accuracy and Integrity: Records are centrally stored and automatically updated, eliminating inconsistencies and errors commonly associated with manual record keeping.

iii. Improved User Experience: Users can easily navigate the system, check book availability in real time, and make reservations or track due dates.

iv. Ease of Maintenance: The modular structure of the system makes it easy to maintain and update individual components without affecting overall functionality.

v. Scalability: Although the system is currently deployed locally, it is built with the potential to scale to larger institutions and integrate new features as needed.

5.4 Challenges Encountered

Despite the success of the project, several challenges were encountered during its development:

i. Data Collection Barriers: Obtaining accurate and detailed information from library staff and users required persistent effort and multiple follow-up sessions.

ii. Technical Constraints: Due to limited hardware and server resources, some features could not be tested under full-scale real-world conditions.

iii. Time Management: Balancing the demands of academic coursework with the requirements of this project was often difficult, necessitating strict scheduling and prioritization.

iv. Integration Issues: During system integration, discrepancies in data flow and session handling needed to be addressed through intensive debugging.

These challenges were effectively managed through careful planning, iterative development, and continuous testing.

5.5 Limitations of the System

While the system meets its core objectives, it has certain limitations:

i. The system currently supports only a single branch or library location.

ii. User registration is handled by administrators, and there is no option for self-registration.

iii. Mobile responsiveness is limited; further UI optimization is needed for smaller devices.

These limitations do not significantly hinder the system's current operations but present opportunities for enhancement.

5.6 Recommendations for Future Work

Several improvements and additional features can be incorporated into future versions of the system to extend its capabilities and enhance user experience:

i. Mobile Optimization: Redesign the interface to be more responsive on mobile devices.

ii. Online Registration and Password Recovery: Allow users to register themselves and recover forgotten passwords via email.

iii. Role-Based Access Control: Implement a more sophisticated access control system with different permissions for admins, librarians, and students.

iv. Cloud Deployment: Host the system on a remote server to allow access from anywhere, increasing convenience and usage.

v. Integration with Institutional Portals: Link the LMS with the university platforms for seamless data sharing and account synchronization.

5.7   Concluding Remarks

The Library Management System developed in this project demonstrates that simple, cost-effective technological solutions can significantly improve institutional workflows and user engagement. By automating repetitive tasks, enhancing record-keeping, and simplifying user interaction, the system modernizes the way library services are delivered.

This project has not only resulted in a functional application but also deepened the developer's understanding of software engineering practices, user-centered design, and systems integration. With further development and deployment, the system holds the potential to be adopted across multiple academic institutions, providing lasting value in the digital transformation of library services.

In conclusion, this project serves as both a practical solution to a real-world problem and a foundation for future innovations in digital library management.

REFERENCES

Adomi, E. E., & Anie, S. O. (2006). An assessment of computer literacy skills of professionals in Nigerian university libraries. *Library Hi Tech News, 23*(2), 10–14. https://doi.org/10.1108/07419050610660735

Ademodi, D. T., & Adepoju, E. O. (2009). Computer skill among librarians in academic libraries in Ondo and Ekiti State, Nigeria. *Library Philosophy and Practice*. Retrieved from http://digitalcommons.unl.edu/libphilprac/1273

Adomokun, S. O., & Oyedokun, T. T. (2018). A survey of information and communication technology literacy among librarians in Nigerian tertiary institutions. *Library Philosophy and Practice*. https://digitalcommons.unl.edu/libphilprac/1797/

Adeniji, M. A., Adeniji, S. E., & Oguniyi, S. (2011). Availability and use of ICT in Olabisi Onabanjo University Library. *PNLA Quarterly*. Retrieved from http://tinyurl.com/pg2oww

Agboola, I. O., & Bamigboye, O. B. (2011). Students' level of study and use of library resources in Nigerian universities: A comparative study. *Library Philosophy and Practice*. Retrieved from http://digitalcommons.unl.edu/libphilprac/1559

Adebayo, O. A., & Ojo, R. A. (2023). Library resources availability and security challenges in Federal University Oye-Ekiti Library, Nigeria. *Global Review of Library and Information Science, 19*(1). Retrieved from https://www.grelis.com.ng/admin/img/paper/9.%20Onaade…

Akintunde, S. A. (2007). Libraries as tools for ICT development. *Library Philosophy and Practice, 9*(1), 1–6.

Aina, L. O. (2004). Coping with the challenges of library and information delivery services: The need for institutionalised professional development. In *Proceedings of the 42nd National Conference of the Nigerian Library Association* (pp. 4–10). Akure, Nigeria.

Aguolu, C. C., & Aguolu, I. E. (2002). *Libraries and information management in Nigeria: Seminal essays on themes and problems*. Maiduguri: El-Inform Services.

Ojo, O. A., Salawu, I. O., & Adeyoyin, S. O. (2021). Integrating digital literacy into Nigerian university library curricula. *African Journal of Information Science and Technology, 11*(1), 45–58.

Ojo, R. A., & Adebayo, E. L. (2023). Challenges of digital library adoption in Nigerian universities. *Nigerian Libraries, 56*(1), 42–53.

Kumar, A., & Kumar, P. (2020). A study on library automation and digital transformation. *Journal of Information and Computational Science, 10*(3), 223–230.

Gardner, M., & Pinfield, S. (2001). Database-backed library websites: A case study of PHP and MySQL at the University of Nottingham. *Program, 35*(1), 33–42.

Gonzalez, M. (2021). Web-based library management systems: Key features and future directions. *International Journal of Information Science, 11*(2), 78–85.

Neelakandan, B., Duraisekar, S., Balasubramani, R., & Srinivasa, R. (2010). Implementation of automated library management system in the School of Chemistry using Koha open source software. *International Journal of Library and Information Science, 2*(6), 108–115.

Prabhakar, K., Kumar, R., Singh, R., & Singh, V. (2014). Design and implementation of a computerized library management system. *International Journal of Computer Applications, 100*(6), 14–19. https://doi.org/10.5120/ijcaRep

Ranganathan, S. R. (2003). *Five laws of library science* (2nd ed.). Ess Ess Publications.

Sharma, C., & Sharma, R. (2005). The need for automation in Indian academic libraries. *Annals of Library and Information Studies, 52*(4), 171–177.

Tedd, L. A. (2006). Library management systems. In *British Librarianship and Information Work 1991–2000* (pp. 452–471). Taylor & Francis.

Turnbull, D., Porter, J., & Humphries, A. (2022). Common elements of learning management system policy documents. *PMC*. Retrieved from https://pubmed.ncbi.nlm.nih.gov/PMC9255472/

APPENDIX A: SUMMARY OF SELECTED RESEARCH PAPERS

This appendix presents a scholarly summary of foundational and contemporary research that influenced the development of the Library Management System (LMS) for Ibogun Campus. The selected studies highlight the evolution, design considerations, and implementation challenges of LMS technologies in academic settings.

i. **Ayo et al. (2023)** – *Integrated LMS Design in Philippine Institutions*

a. Investigated the role of user support and system usability in LMS adoption. The study highlighted critical barriers such as digital illiteracy and lack of LMS training.

b. Recommends built-in onboarding tools, tutorials, and peer support systems to improve adoption.

ii. **Das & Ul Islam (2021)** – *AI and ML in Library Management*

a. Conducted a systematic review of artificial intelligence applications in LMS. Found that AI-powered recommendation engines, predictive cataloguing, and virtual reference assistants are emerging trends.

b. Suggested future integration of machine learning to personalize user experience.

iii. **Liu (2024)** – *Interoperability and Metadata Standards*

a. Synthesized research on Library Services Platforms (LSPs) and their reliance on metadata exchange protocols (MARC21, Z39.50, OAI-PMH).

b. Advocates for integrating such standards in LMS to enable multi-library collaboration and resource discovery.

iv. **Kryukova et al. (2021)** – *Cybersecurity in Digital Libraries*

a. Identified increasing vulnerability of web-based LMS and emphasized the need for cyber hygiene and user education.

b. Recommends two-layer security architecture and continuous awareness training.

v. **Subaveerapandiyan et al. (2022)** – *Challenges in E-Resource Management*

a. Found that librarians struggle with e-resource navigation due to inadequate training.

b. Proposed structured ICT education and policy reforms to boost LMS effectiveness in resource-limited contexts.

APPENDIX B: CODE MODULES AND EXCERPTS

The following code modules represent key features of the LMS project. Each snippet is accompanied by a description of its purpose and functionality.

User Authentication & Session Management

```
session_start();
if ($_SERVER['REQUEST_METHOD']=='POST'){
 $user = $_POST['email'];
 $pw = $_POST['password'];
 $stmt = $conn->prepare("SELECT id, role, password_hash FROM users WHERE email=?");
 $stmt->bind_param('s',$user); $stmt->execute();
 $stmt->store_result(); $stmt->bind_result($id,$role,$hash);
 if($stmt->fetch() && password_verify($pw,$hash)){
  $_SESSION['user_id']=$id; $_SESSION['role']=$role;
  header("Location: dashboard.php"); exit;
 } else {
  $error="Invalid credentials.";
 }
}
```

Purpose: Validates login credentials and establishes user sessions.

Book Cataloguing Module

```
$stmt = $conn->prepare("INSERT INTO books (isbn, title, author, category, qty) VALUES (?, ?, ?, ?, ?)");
$stmt->bind_param('ssssi', $isbn, $title, $author, $category, $qty);
$stmt->execute();
```

Purpose: Inserts new book records into the LMS database.

Book Borrowing Transaction

```
$conn->begin_transaction();
$conn->query("UPDATE books SET qty=qty-1 WHERE id={$book_id}");
$conn->query("INSERT INTO transactions(user_id, book_id, issue_date, due_date)
 VALUES({$uid}, {$book_id}, NOW(), DATE_ADD(NOW(), INTERVAL 14 DAY))");
$conn->commit();
```

Purpose: Updates stock and logs transaction for book issuance.

HOD Email Alert

$query = "SELECT DATEDIFF(NOW(), due_date) AS days FROM transactions WHERE returned = 0";

$res = $conn->query($query);

while($r = $res->fetch_assoc()){

  if($r['days'] > 0) echo "Fine: ₦" . ($r['days'] * 50);

}

Purpose: Computes overdue items and notifies users.

Search

$('#search').on('keyup', function(){

  $.get('search_books.php', { q: $(this).val() }, function(data){

    $('#results').html(data);

  });

});

Purpose: Enables real-time book search via frontend.