Developing a Library Management System Utilizing Python

SAMUEL J. AYANO[1], SUNDAY A. OJOMU[2], GETRUDE FISHER[3], EMMANUEL O. ONOYOM-ITA[4]

[1, 2, 3, 4] *Department of Electrical/Electronic Engineering, University of Cross River State, Calabar, Cross River State, Nigeria.*

Abstract*— This paper presents an automated university library management system (LMS). Its aim is to create a user-friendly and efficient LMS that optimizes library tasks and enhances user interactions. The methodology involves reviewing existing LMS, analyzing requirements, designing, implementing, and testing the system. The Waterfall model was used for development, employing open-source tools. The results reveal the system's efficacy in managing library tasks, enhancing information access, and improving user experiences. The LMS was crafted using HTML, CSS, JavaScript, Flask (a Python web framework), and a PostgreSQL database. Challenges encountered encompassed inadequate user interface in existing systems, data handling issues, and security vulnerabilities. This paper contributes insights into designing and implementing LMS in academic settings, providing practical suggestions for bolstering library operations' effectiveness and efficiency.

***Indexed Terms*—L. M. S (Library management system), Librarian, Python, Modelling, system development life cycle (SDLC), Flask, PostgreSQL.**

I. INTRODUCTION

Library management systems (LMS) represent software applications crafted to oversee and manage library operations. They offer automated solutions for handling various library resources, encompassing books, journals, newspapers, magazines, audiovisual materials, and more. These systems efficiently manage material circulation, including borrowing, returning, renewing, and reserving items. In traditional libraries, the process of locating books is cumbersome for students, lacking proper database maintenance for issues and fines [1].

The LMS, developed using Flask, a Python web framework, primarily targets fundamental library functions, such as adding new members and books, updating information, searching and lending books, and instituting a library card system for authorized users and librarians [2].

This research aims to devise and execute a library management system that enhances the efficiency and effectiveness of library services. It seeks to explore the requisites of a modern LMS, identify crucial features necessary to meet these demands, and create a system aligning with contemporary library needs. Establishing an LMS represents a pivotal stride in improving library services' efficiency and effectiveness. This system should cater to modern libraries' requirements, including functionality, scalability, integration, user-friendliness, security, and adaptability. The LMS implementation will involve multiple stages, including planning, design, development, testing, and deployment.

A library serves as an institution engaged in collecting, processing, conserving, and disseminating recorded information in diverse formats most convenient for its intended users. Information serves as an indispensable catalyst for development. For a nation to progress, it must possess and furnish relevant and sufficient information on various issues. Libraries facilitate access to such information, enabling individuals to lead fulfilling lives by adeptly acquiring, organizing, disseminating, and preserving materials in varied forms, readily accessible and utilizable when needed [3].

Libraries assume a significant role in national development by supporting the education sector. It is undeniable that without libraries, the existence of universities would be inconceivable, resulting in a dearth of professionals. Professionals, regardless of their graduation status, leverage their skills through knowledge gleaned from books and the internet, available within libraries. Lawyers uphold the flow of justice and nurture constitutional progress; educators champion literacy campaigns; healthcare practitioners address health concerns—the scope is vast. All these advancements result from disseminating information through diverse library resources [4].

II. METHODOLOGY

The web-based Library Management System (LMS) features an interactive User Interface (UI) that caters to users of varying expertise levels, ensuring simplicity for both novices and those with intermediate skills. To prevent disorganization, lack of structure, or undesirable outcomes, a defined perspective has been established by opting for a specific model for this project. In many instances, a system refers to Information Technology (IT) components like hardware and software. Project and program managers commonly participate in the Systems Development Life Cycle (SDLC), working alongside system and software engineers, development teams, and end-users [5].

The development methodology employed for this proposed system is the waterfall model, a component of the System Development Life Cycle (SDLC). Initially introduced by Winston Royce in 1970, the waterfall model represents a traditional approach within the SDLC framework. This model adopts a linear and sequential method to create systems, progressing systematically from one phase to the next in a downward manner, hence its name "waterfall." Notably, the waterfall approach lacks a defined process for returning to prior phases to address changes in requirements. This methodology stands as one of the earliest approaches utilized in software development. The phases within the waterfall model encompass requirements gathering and analysis, system design, implementation, testing, deployment, and maintenance.

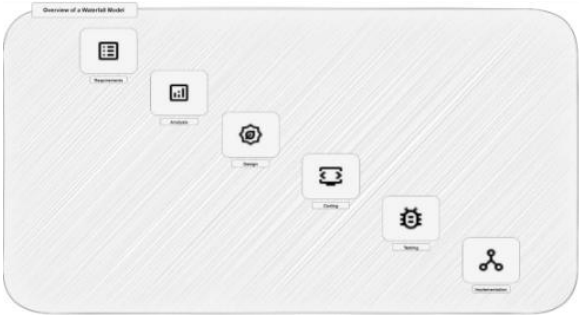

Fig. 1.0: General Overview of a Waterfall Model.

The Library Book Management System employs a Layered Architecture, emphasizing the integration of independence and segregation within its architectural designs. This integration facilitates localized adaptation to any alterations that occur. The layered architecture implements this concept of independence and separation by organizing and partitioning the system's functionality into distinct layers. Each layer relies on the services furnished by the immediate layer beneath it. The system evolves incrementally, enabling users to access certain services provided by the developing layers. Additionally, this architecture boasts portability and facilitates easy modifications [6]. This system's architecture is segmented into various layers, encompassing the web browser, client-server infrastructure, and database management, ensuring optimal and efficient system operation.

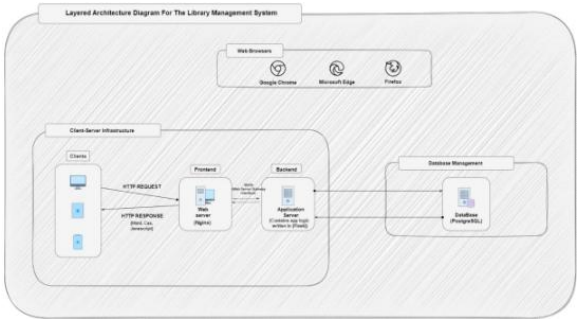

Fig. 2.0: Layered Architecture Diagram for The Library Management System

- Software Specialization:

The web-based LMS application is segmented into two primary sections: the Front-End and Back-End components.

- Front-End:

The client-side, known as the front-end, is crafted using HTML, CSS, and JavaScript[7][8].

- Back-End:

The server-side, referred to as the back-end, is developed using the Flask framework and PostgreSQL for the RDBMS[9].

- System Feasibility:

An analysis of system feasibility involves multiple assessments to determine its potential acquisition and subsequent adoption. This ongoing process evaluates various factors such as operational, technical, economic, cultural, legal, and scheduling feasibility, among others [10].

III. FINDINGS AND CONVERSATION

Access for Librarians: The functionality enabling librarian access is established within the 'auth.py' file. Within this file, authentication for librarians is managed, validating their credentials and overseeing secure sessions. Flask-Login is applied to fortify the authentication process, granting access exclusively to authorized librarians for privileged functionalities.

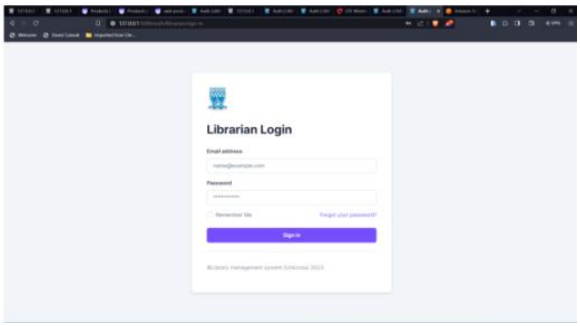

Fig. 3.0: Screenshot of the Librarian Login Page

Librarian Dashboard: The Librarian Dashboard serves as a centralized platform within the 'librarian.py' file, amalgamating crucial data and tools essential for librarians to manage the system efficiently.

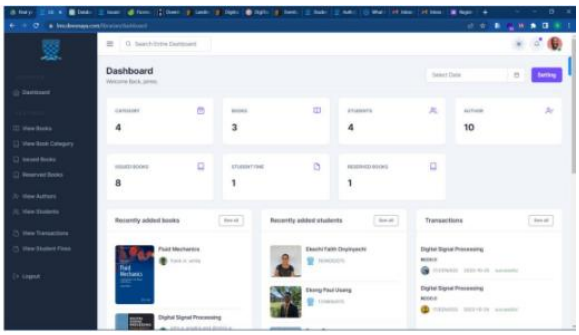

Fig. 4.0: Screenshot of the Librarian Dashboard

Add New Student: This file manages the creation of student profiles in the system, capturing essential details and storing them in the database. The functionality to add a new student is implemented in the 'librarian.py' file.

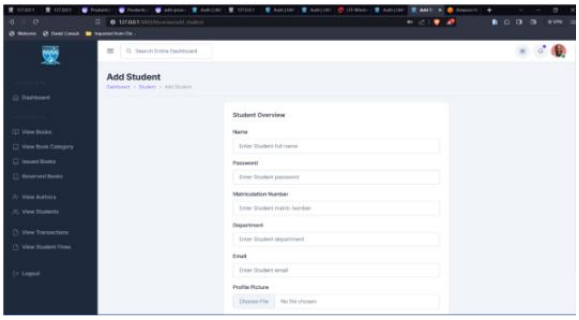

Fig. 5.0: Screenshot of the Librarian adding new student

View Student List: The view students list functionality is implemented in the 'librarian.py' file. This file manages theretrieval of student data from the database and presents it in a user-friendly format on the designated web page.

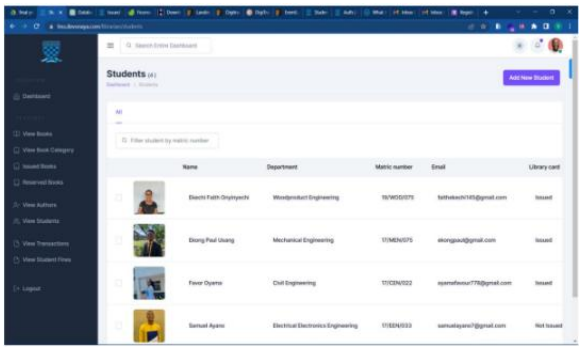

Fig. 6.0: Screenshot of the Librarian viewing list of all the students registered on the LMS

Generate Library Card: The capability to generate a library card is implemented within the 'librarian.py' file. This file manages the process of creating unique library cards for students, ensuring that each card contains the necessary information. The logic involves integrating with the database to associate the library card with the respective student.

Fig. 7.0: Screenshot of the student library card generated by the librarian

Student Fines List: The Student Fines List functionality is managed in the 'librarian.py' file. This file handles the retrieval and display of fines associated with students in the system. The logic ensures accurate calculation and presentation of fines accrued by students for overdue books or other applicable charges.

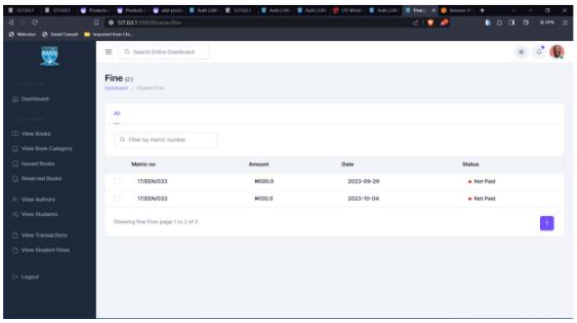

Fig. 8.0: Screenshot of the student viewing his fine page

Issue Book: The functionality for a student to issue a book is managed in a dedicated route, typically in thestudent.py file. This section ensures that student can efficiently request for book issues, updating the system's records accordingly

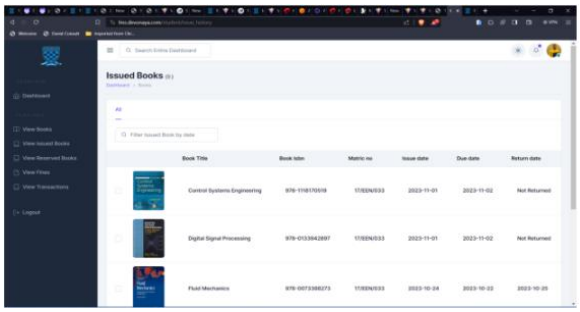

Fig. 9.0: Screenshot of the student viewing list of issued books

View Book detail: The functionality for a student to view a book detail is managed in a dedicated route, typically inthe student.py file. This section ensures that student can efficiently view for book details such as description of the book, and also issue and reserve a single book updating the system's records accordingly.

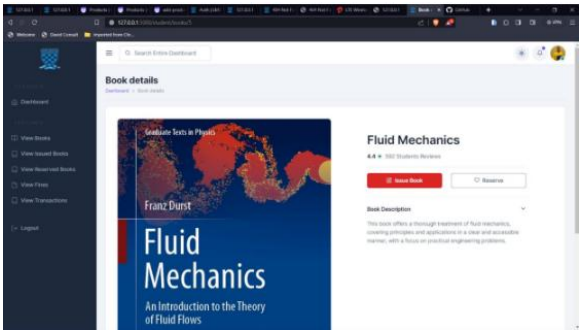

Fig. 10.0: Screenshot of the student viewing book details of a single book

CONCLUSION

The development and implementation of the Library Management System (LMS) mark a significant advancement in the realm of library administration and user services. This project aimed to create a dynamic system capable of automating essential library processes, from cataloging to user management and circulation. As we conclude this phase of the project, it is evident that the LMS has the potential to redefine library operations, making them more streamlined, accessible, and responsive to the evolving needs of library staff and patrons. Throughout the developmental journey, challenges were met, and strategic decisions were made to shape the system. The LMS now provides librarians with a

powerful tool to manage collections efficiently, enables patrons to access and borrow books seamlessly, and offers administrators valuable insights through analytics. User feedback and testing have been pivotal in refining the system, ensuring that it aligns with the practical requirements of a functioning library environment

ACKNOWLEDGMENT

First and foremost, I am grateful to the Lord Jesus Christ for providing me with good health, strength and understanding to carry on with education even when the going looked difficult to endure. I wish to extend my sincere gratitude to my project guide, Engr. S.A Ojomu for his valuable guidance and encouragement that has been helpful in successful completion of this project. I cannot fail to thank the entire staff of the department of Electrical/Electronic Engineering for providing all the required resources for the successful completion of my project. I am especially grateful to my family in persons of Dr. David, Mrs. Joyce Ayano and Victoria Ayano for their timely supports without which I would not have finished my project successfully. I extend my thanks to my friends
Faith Ekechi and Paul Usang for their support and advice throughout my journey in this work and allthose who have contributed directly and indirectly for the completion of this work.

REFERENCES

(Periodical style)

[1] Nevon projects, "Library management system using Python," [Online]. Available: https://nevonprojects.com/library-management-system-using-python/.Accessed: Mar. 03, 2023.

[2] S.T. Muthyala, "Online Library Project," [Online]. Available:https://opus.govst.edu/cgi/viewcontent.cgi?article=1179. Accessed: Mar. 03, 2023.

[3] G. Onoyeyan and O.I. Adesina, "A Literature Review of the Role of Libraries in the Provision of Information towards the Attainment of the United Nations Millennium Development Goals (MDGs)in Nigeria," Library Philosophy and Practice (e-journal), 2014, 1069.

[4] P. Achitabwino, "Libraries and national development," [Online]. Available: http://pachitabwino.blogspot.com/2007/03/libraries-and-national-development.html. Accessed: Jan. 2010.

[5] A.S. Gillis, "Systems development life cycle (SDLC)," Jun. 2019.

[6] Kweingoti S.B, Library book management system (2014).

[7] "Definition of HTML," [Online]. Available: https://www.computerpdf.com/web programming/html/. Accessed: Apr. 26, 2023.

[8] T. Adebesin, "Library Management System," Oct. 2015.

[9] R. Borovica-Gajic, "Database design concepts," May 7, 2023.

[10] Kweingoti S.B, Library book management system (2014)