Library Management System Specially Designed for Small Schools

A M N Hemamali

DIT 10032

Information Communication Technology Center -Makandura
Wayamba University of Sri Lanka
2024

Library Management System Specially Designed for Small Schools

By

A M N Hemamali

A Report

Submitted in Partial Fulfillment for the Requirements of

DIPLOMA IN INFORMATION TECHNOLOGY

2024

Information Communication Technology Center
Wayamba University of Sri Lanka
Makandura, Gonawila
Sri Lanka

DECLARATION

I certify that this project report does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any University and to the best of my knowledge and belief it does not contain any material previously published or written or orally communicated by another person expect where due reference is made in the text.

....
Signature

....
Name of the Student

....
Date

To the best of my knowledge the above particulars are correct.

………………………………………
Course Coordinator
Mr. E.M.D.S Ekanayake
Senior Lecturer
ICT Center, Wayamba University of Sri Lanka,
Makandura, Gonawila. (NWP)

Date: …………………………

……………..…………………..
Supervisor
Dr(Mrs) Kanchana Weerasinghe
Senior Lecturer
ICT Center,
Wayamba University of Sri Lanka,
Makandura, Gonawila. (NWP)

Date: …………………………

……………………………..
Course Director
Dr. H.A.C.K. Jayathilake
Senior Lecturer
ICT Center, Wayamba University of Sri Lanka
Makandura, Gonawila. (NWP)

ACKNOWLEDGEMENT

I wish to extend my deep and sincere gratitude to my supervisor, Dr (Mrs) Kanchana Weerasinghe, Senior Lecturer, Information Communication Technology Center for his valuable guidance, continuous supervision, and enormous assistance for all necessary arrangements provided me to conduct my web development project successfully.

I would like to convey my gratitude to Dr. H.A.C.K Jayathilake, Course Director and Mr. E.M.D.S Ekanayake, Course Coordinator, Information Communication Technology Center for giving me the necessary guidelines and opportunity to undertake this project under the Information Communication Technology Center, Makandura.

My sincere thanks go to all staff members of Information Communication Technology Center Makandura and DIT batch mates for their assistance in collecting data and other information.

CONTENTS

Page No

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

Introduction

In today's educational environment, libraries are essential components of schools, providing students with access to valuable learning resources. However, many small school libraries in Sri Lanka face significant challenges in managing their collections efficiently due to limited resources and staff. Traditional methods, such as manual record-keeping, are error-prone, time-consuming, and lack scalability, making it difficult to maintain an organized and accessible library system.

Recognizing these challenges, I developed a Library Management System specifically designed for small school libraries. This web-based application was built using popular and efficient technologies: React for the frontend and Express.js for the backend. The system is intended to simplify library operations through essential features like user access control, CRUD (Create, Read, Update, Delete) operations for book management, and book count tracking[1].

Unlike complex and expensive library management systems, this solution focuses on addressing the specific needs of small school libraries, ensuring affordability, ease of use, and effective resource management. By automating core library functions, the system not only improves operational efficiency but also enhances the overall organization and security of library data[2][3].

This introduction provides an overview of the project's background, the problems it addresses, the limitations of the existing manual system, and the proposed solution's core features and benefits. The sections that follow will delve deeper into these aspects, providing a comprehensive understanding of the Library Management System and its impact on small school libraries.

1.1 Background

The management of books in small school libraries is a common challenge due to limited resources and staff. While large-scale library management software exists, these solutions often include extensive features that are unnecessary for smaller libraries, making them costly and difficult to use. With this in mind, the project was developed using React for the frontend and Express.js for the backend, leveraging their popularity and efficiency to deliver a simple yet effective web-based library management system.

1.2 Problem definition

Small school libraries often face significant challenges in managing their book collections effectively, primarily due to limited personnel, inadequate technological resources, and a lack of specialized training among staff. These libraries typically rely on manual methods such as logbooks or basic spreadsheets to track their inventory, which are not only time-consuming but also prone to errors. As a result, essential tasks like cataloging books, monitoring borrowing and returning activities, and maintaining accurate records become inefficient and burdensome.

While various library management solutions are available, they are often designed for larger institutions with complex requirements, making them overly intricate and expensive for smaller libraries. These systems typically include advanced features such as extensive analytics, multi-branch management, and integrations with external platforms, which are unnecessary for smaller school libraries. Consequently, many small schools find it difficult to adopt these systems, as they exceed both their technical needs and budget constraints.

The primary issue, therefore, lies in providing an affordable, scalable, and user-friendly system tailored specifically for small school libraries. Such a solution should focus on addressing their basic needs, such as simple book management, user access control, and borrowing and return tracking, without overwhelming users with unnecessary features. By bridging this gap, small school libraries can modernize their operations, improve efficiency, and ensure better resource management, ultimately enhancing the educational experience for their students.

1.3 Proposed solution

The proposed Library Management System aims to deliver a streamlined software solution with the following core features:

1. User Access Control: To ensure data security and proper management.
2. CRUD Operations for Books: Facilitating the addition, updating, deletion, and retrieval of book records.
3. Book Count Tracking: Providing quick insights into the number of books in the library.

This solution is cost-effective, easy to use, and focuses on the specific requirements of small school libraries[2].

1.4 Scope and context of the project

Aim:

The Library Management System aims to provide an affordable, efficient, and user-friendly solution for small school libraries to manage their book collections and operations effectively.

Objectives:

- To provide small school libraries with an affordable and efficient library management system.
- To simplify the process of cataloging and managing books.
- To enhance data security and organization through access control.
- To improve the overall efficiency of library management.

The system was built to function as a web-based application, ensuring accessibility and ease of deployment.

1.5 Existing system

Currently, many small school libraries rely on manual record-keeping methods, such as logbooks or spreadsheets. These methods are prone to errors, are time-consuming, and lack the scalability required as the library grows. The absence of centralized data management further complicates the tracking of book records and inventory.

1.6 Anticipated benefits

The Library Management System is expected to offer several benefits, including:

- Improved Efficiency: Automating processes to save time and effort.
- Better Organization: Maintaining a centralized database for book records.
- Enhanced Security: Controlled access to ensure data privacy and integrity.
- Cost-Effectiveness: Providing an affordable solution compared to existing comprehensive systems.

1.7 Structure of the report

- Introduction: Overview of the project and its purpose.
- Background: Context and rationale for developing the system.

- Problem Definition: Analysis of the challenges faced by small school libraries.
- Proposed Solution: Description of the system and its features.
- Scope and Context: Objectives and intended audience for the system.
- Existing System: Examination of current methods and their limitations.
- Anticipated Benefits: Advantages of implementing the proposed system.
- Implementation Details: Technical aspects, tools, and frameworks used.
- Future Enhancements: Potential updates and additional features.
- Conclusion: Summary of the project's impact and contributions.

CHAPTER TWO

Methodology

The methodology for developing the Library Management System involved a structured approach to system design, database creation, user interface development, and the selection of appropriate tools and technologies for implementation. The following sections provide a detailed description of each phase of the process.

Design and Implementation of the System

2.1 System design

The system was designed as a web-based application with a client-server architecture. The backend handles data processing and provides APIs for communication, while the frontend focuses on user interaction. Key design decisions included[1].

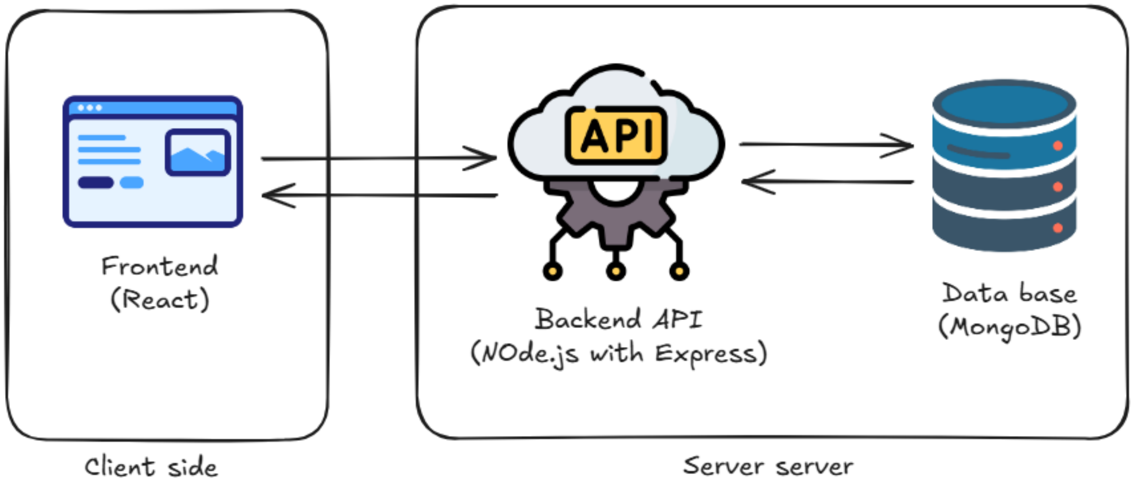


Figure 1: System diagram

These are the points considered while designing the system.

- Modular Architecture: Ensuring scalability and maintainability through separate modules for user management, book management, and statistics.

- Security Considerations: Implementing access control mechanisms to protect sensitive library data.
- Responsive Design: Designing the system to be accessible on multiple devices, including desktops, tablets, and mobile devices.

2.2 Database design

The database for the *Library Management System* was designed to efficiently manage book records, user information, and transaction histories, utilizing **MongoDB Atlas** as the database management system[4]. MongoDB Atlas, a cloud-based NoSQL database, was chosen for its flexibility, scalability, and ability to handle dynamic data structures, making it particularly suitable for addressing the needs of small school libraries. The database design focused on organizing data into three main collections: Users, Books, and Transactions[5].

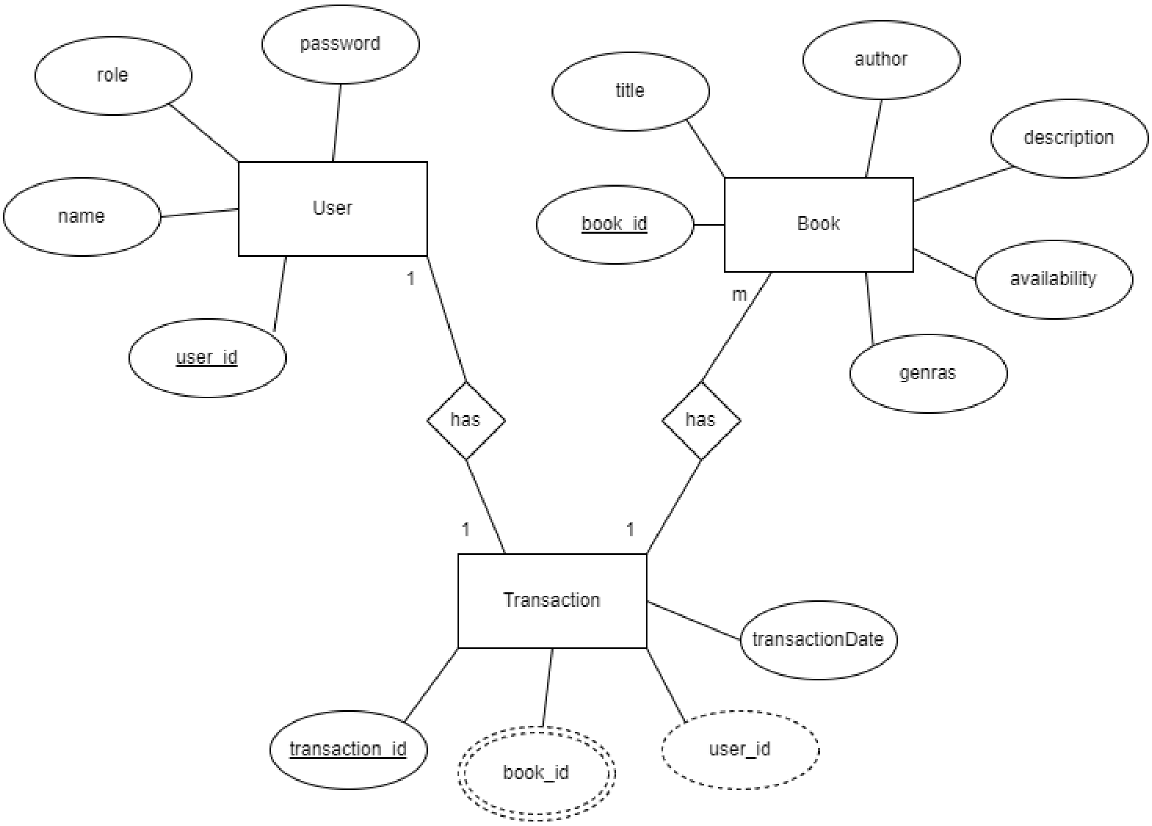
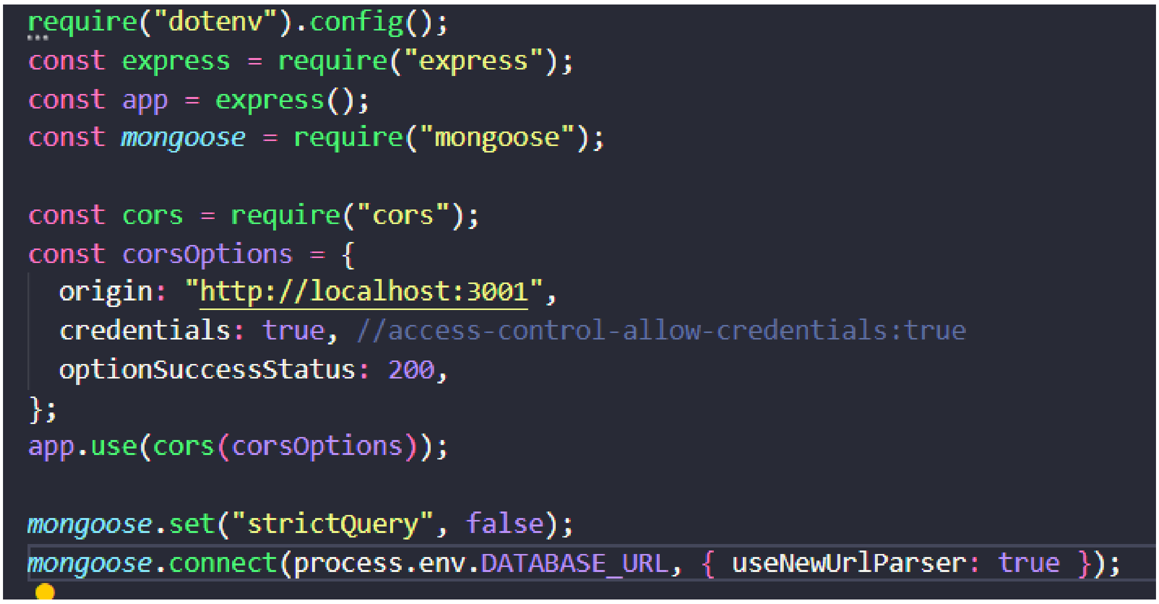


Figure 2: ER diagram for Library Management System

The **Users** collection stores information about library staff, including unique user IDs, names, roles (such as librarian or admin), and secure login credentials with hashed passwords. The **Books** collection maintains details about the library's inventory, such as book IDs, titles, authors, genres, and availability status, ensuring that all essential information about the library's holdings is easily accessible. The **Transactions** collection tracks all borrowing and returning activities, with fields like transaction IDs, associated user IDs and book IDs, transaction dates, and the type of transaction (borrow or return).

Relationships between these collections are managed using MongoDB's referencing and embedding techniques. A one-to-many relationship exists between the Users collection and the Transactions collection, as a single user can perform multiple transactions. Similarly, the Books collection has a one-to-many relationship with the Transactions collection, allowing a single book to be associated with multiple borrowing or returning activities.

The database design leveraged MongoDB's document-oriented storage format, where data is stored as JSON-like documents, offering an efficient and flexible approach to handling dynamic and hierarchical data structures. To maintain data integrity and minimize redundancy, references were used to link collections via fields like userId and bookId. Indexing was implemented on frequently queried fields to improve performance and ensure fast query response times. Additionally, MongoDB Atlas's horizontal scaling capabilities provide the system with the flexibility to handle increasing amounts of data as the library expands.

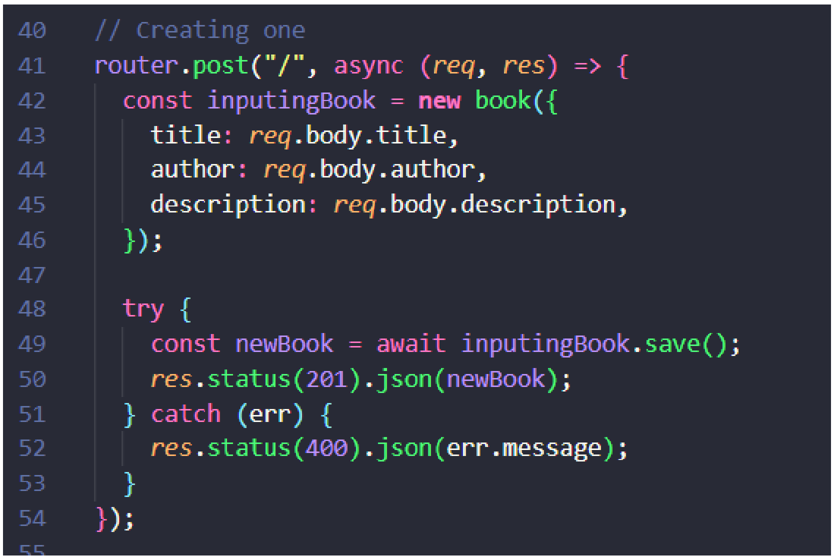


```
require("dotenv").config();
const express = require("express");
const app = express();
const mongoose = require("mongoose");

const cors = require("cors");
const corsOptions = {
  origin: "http://localhost:3001",
  credentials: true, //access-control-allow-credentials:true
  optionSuccessStatus: 200,
};
app.use(cors(corsOptions));

mongoose.set("strictQuery", false);
mongoose.connect(process.env.DATABASE_URL, { useNewUrlParser: true });
```

Figure 3: database connection of the express server

This database design ensured that the *Library Management System* could manage its core data efficiently while remaining adaptable to future enhancements, such as adding new features or supporting larger datasets as the library grows. The use of MongoDB Atlas provided a robust foundation for achieving the system's objectives and maintaining long-term reliability and performance.

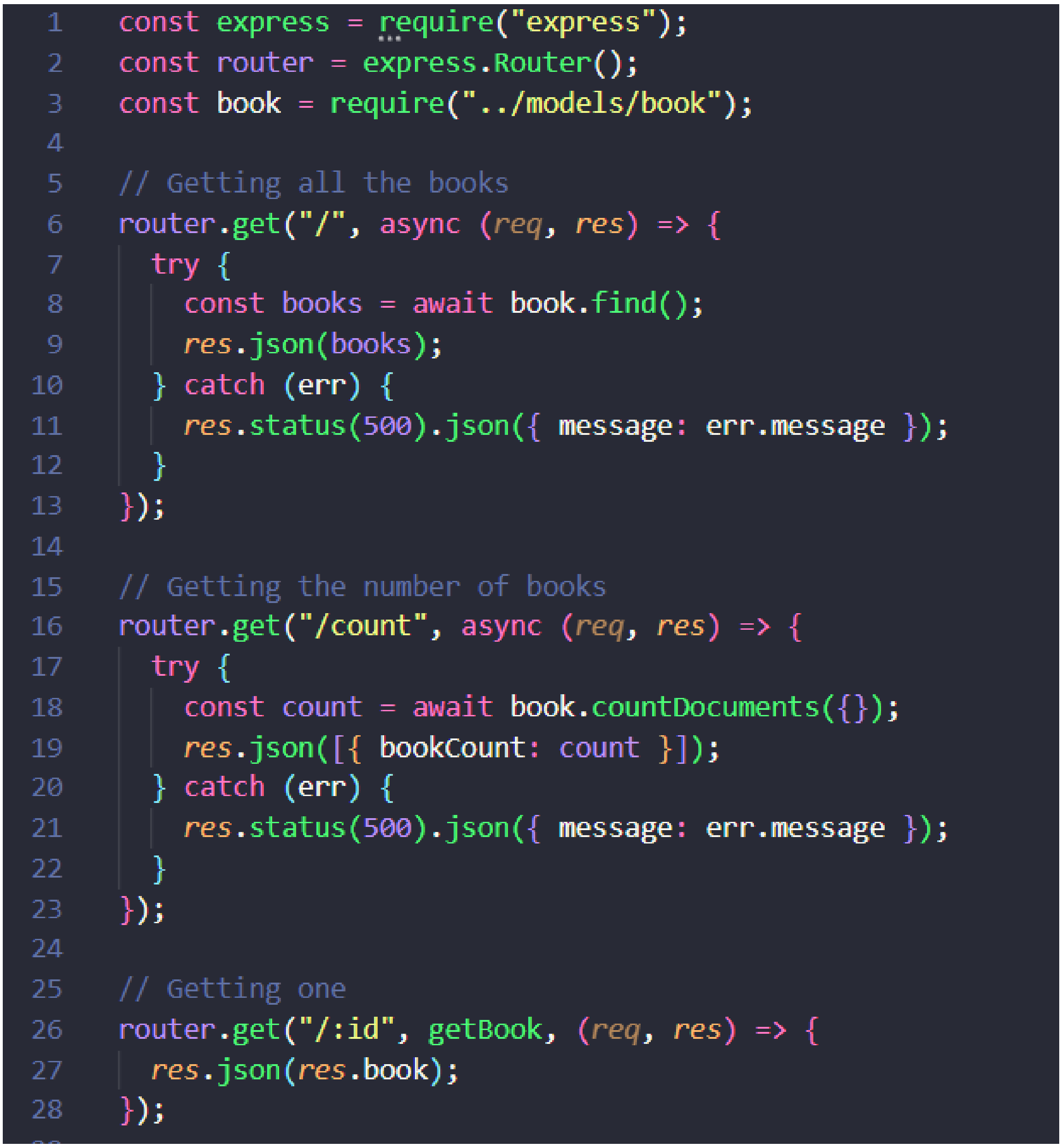
Create book endpoint related to books API.



```
// Creating one
router.post("/", async (req, res) => {
  const inputingBook = new book({
    title: req.body.title,
    author: req.body.author,
    description: req.body.description,
  });

  try {
    const newBook = await inputingBook.save();
    res.status(201).json(newBook);
  } catch (err) {
    res.status(400).json(err.message);
  }
});
```

Figure 4: Create book endpoint controller

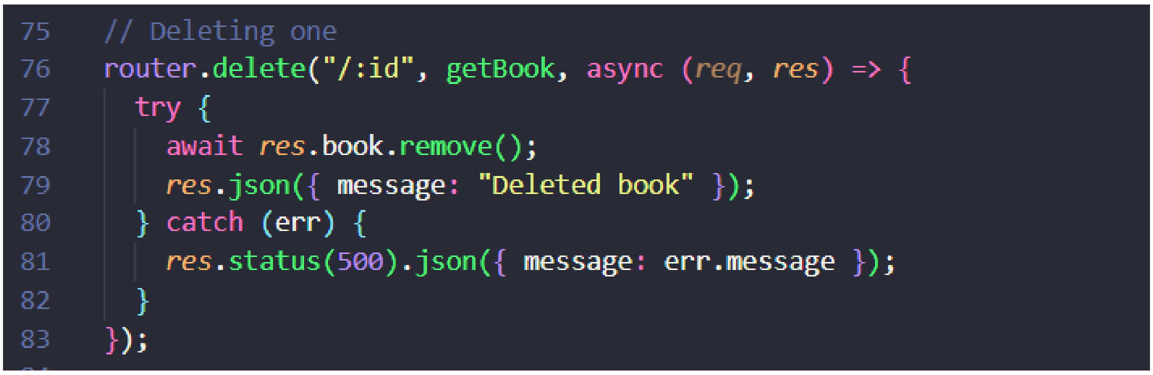
Get operations related to books API.



```javascript
const express = require("express");
const router = express.Router();
const book = require("../models/book");

// Getting all the books
router.get("/", async (req, res) => {
  try {
    const books = await book.find();
    res.json(books);
  } catch (err) {
    res.status(500).json({ message: err.message });
  }
});

// Getting the number of books
router.get("/count", async (req, res) => {
  try {
    const count = await book.countDocuments({});
    res.json([{ bookCount: count }]);
  } catch (err) {
    res.status(500).json({ message: err.message });
  }
});

// Getting one
router.get("/:id", getBook, (req, res) => {
  res.json(res.book);
});
```

Figure 5: get endpoints related to books

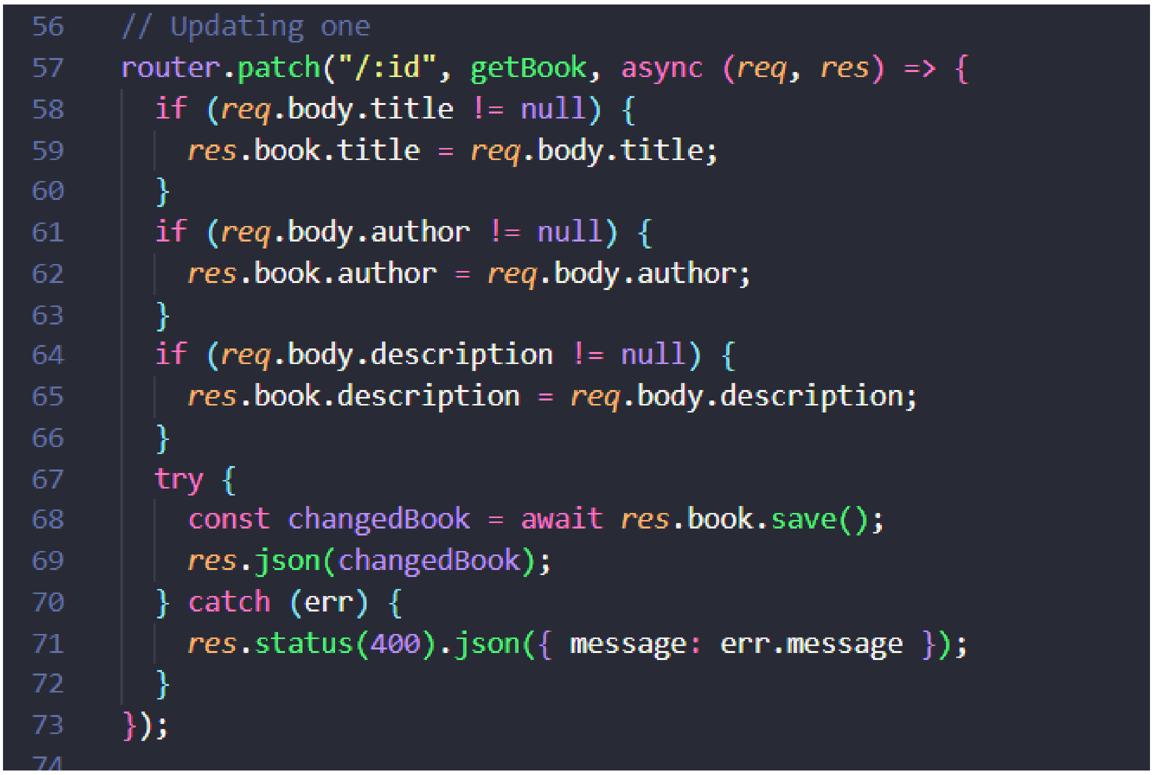
Backend API functional controllers related to the delete endpoint.



```
// Deleting one
router.delete("/:id", getBook, async (req, res) => {
  try {
    await res.book.remove();
    res.json({ message: "Deleted book" });
  } catch (err) {
    res.status(500).json({ message: err.message });
  }
});
```

Figure 6: delete endpoint related to books

Backend API functional controllers related to the update endpoints.



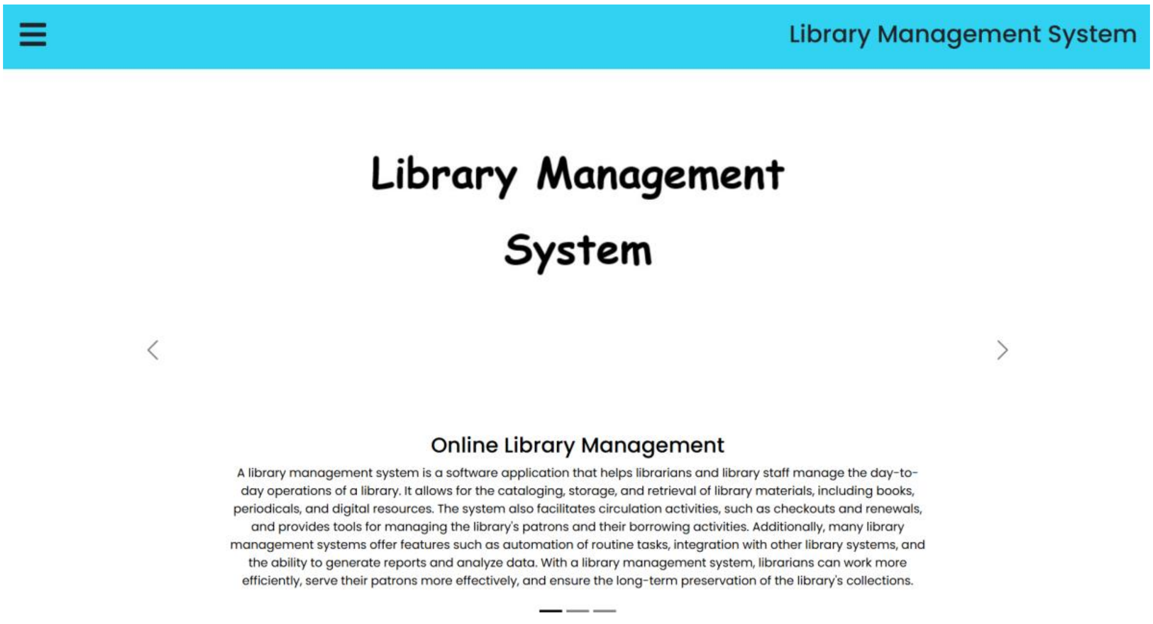
```
// Updating one
router.patch("/:id", getBook, async (req, res) => {
  if (req.body.title != null) {
    res.book.title = req.body.title;
  }
  if (req.body.author != null) {
    res.book.author = req.body.author;
  }
  if (req.body.description != null) {
    res.book.description = req.body.description;
  }
  try {
    const changedBook = await res.book.save();
    res.json(changedBook);
  } catch (err) {
    res.status(400).json({ message: err.message });
  }
});
```

Figure 7: partially update endpoint related to books

2.3 Interface design

The user interface of the *Library Management System* was designed using **React**, emphasizing simplicity, intuitive navigation, and an engaging user experience. React's component-based architecture allowed the development of reusable UI components, ensuring a consistent look and feel across the application while making future enhancements and maintenance easier. The interface was tailored to meet the specific needs of small school libraries, ensuring that even users with minimal technical expertise could interact with the system effectively.

A central feature of the interface is the **Dashboard**, which provides a concise summary of how to use the library management system with a swiper.



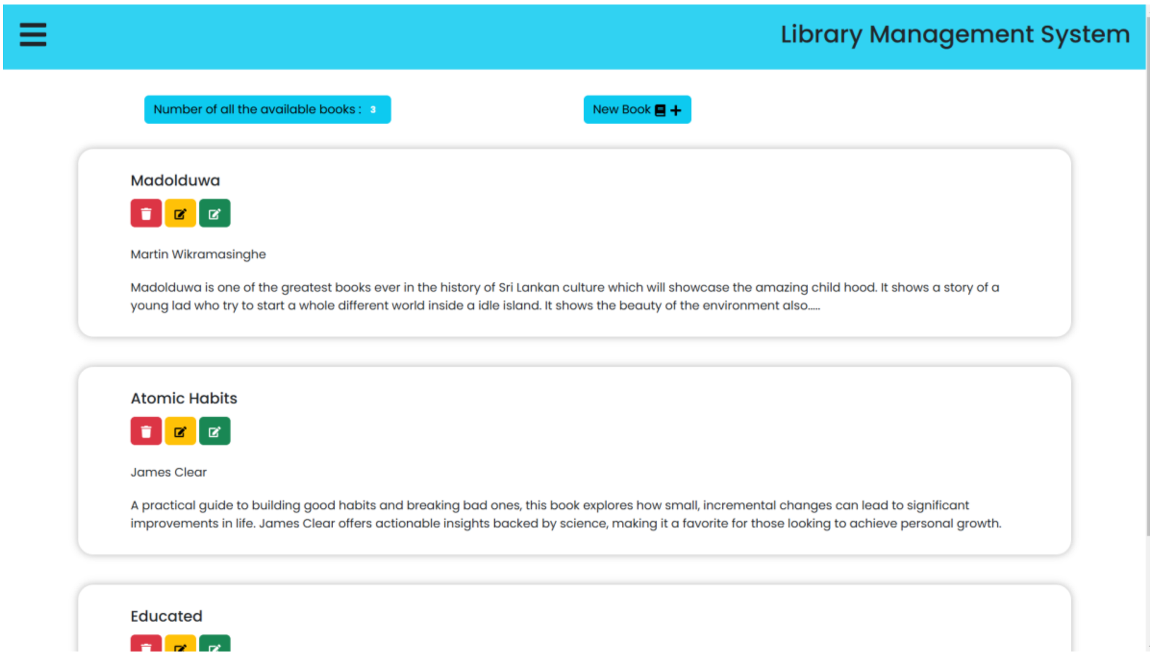
Library Management System

Online Library Management

A library management system is a software application that helps librarians and library staff manage the day-to-day operations of a library. It allows for the cataloging, storage, and retrieval of library materials, including books, periodicals, and digital resources. The system also facilitates circulation activities, such as checkouts and renewals, and provides tools for managing the library's patrons and their borrowing activities. Additionally, many library management systems offer features such as automation of routine tasks, integration with other library systems, and the ability to generate reports and analyze data. With a library management system, librarians can work more efficiently, serve their patrons more effectively, and ensure the long-term preservation of the library's collections.

Figure 8: dashboard UI of the library mangement system

The system supports **CRUD Operations**, with intuitive forms that simplify the process of adding, editing, deleting, and viewing book records. These forms were designed with user-friendliness in mind, incorporating clear labels, helpful placeholders, and real-time feedback to minimize errors. Users can easily navigate through the forms, ensuring that managing book data is a seamless experience.


Library Management System
Number of all the available books : 3
New Book +

Madolduwa

Martin Wikramasinghe

Madolduwa is one of the greatest books ever in the history of Sri Lankan culture which will showcase the amazing child hood. It shows a story of a young lad who try to start a whole different world inside a idle island. It shows the beauty of the environment also.....

Atomic Habits

James Clear

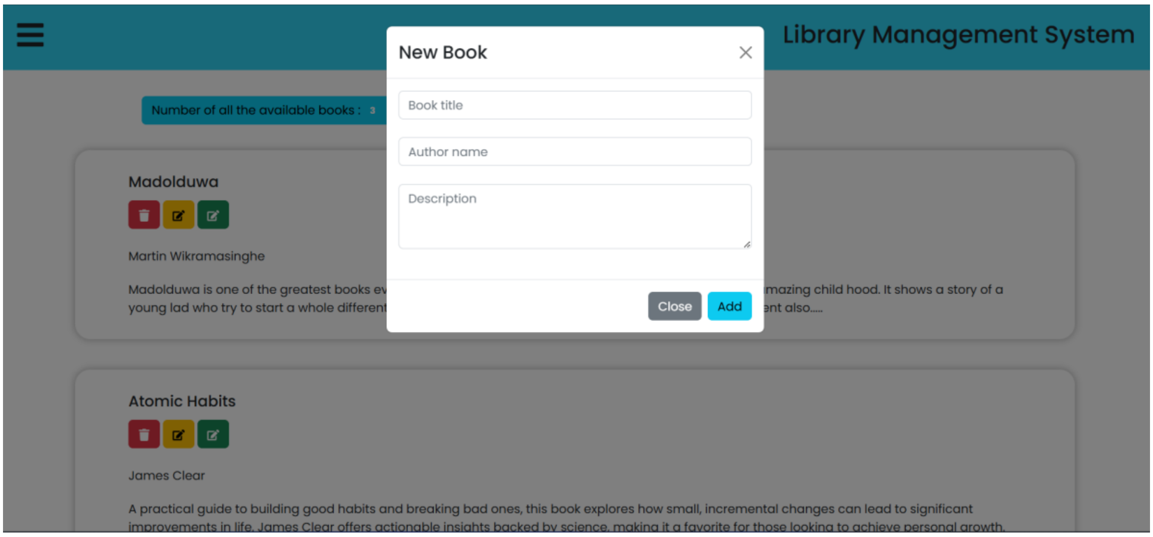
A practical guide to building good habits and breaking bad ones, this book explores how small, incremental changes can lead to significant improvements in life. James Clear offers actionable insights backed by science, making it a favorite for those looking to achieve personal growth.

Educated

Figure 9: books UI in library mangement system

Can add new books by clicking on the “New Books” button in the top of the Books UI.

When clicking on the button a new book popup will open.


Library Management System
Number of all the available books : 3
New Book
Book title
Author name
Description
Close
Add

Madolduwa

Martin Wikramasinghe

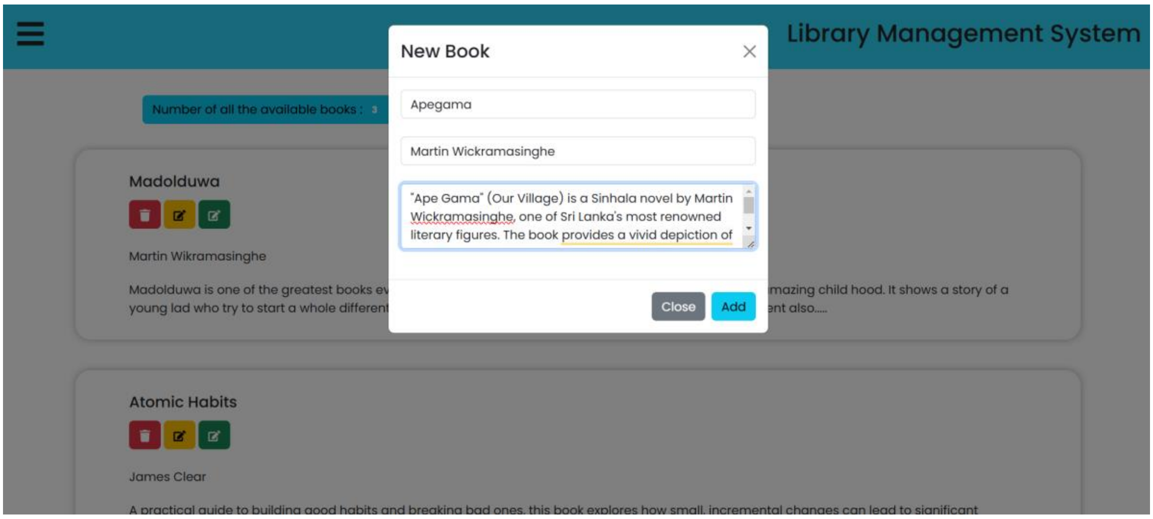
Madolduwa is one of the greatest books ev mazing child hood. It shows a story of a young lad who try to start a whole different ent also.....

Atomic Habits

James Clear

A practical guide to building good habits and breaking bad ones, this book explores how small, incremental changes can lead to significant improvements in life. James Clear offers actionable insights backed by science, making it a favorite for those looking to achieve personal growth.

Figure 10: New book popup



Library Management System
Number of all the available books : 3

New Book

Apegama

Martin Wickramasinghe

"Ape Gama" (Our Village) is a Sinhala novel by Martin Wickramasinghe, one of Sri Lanka's most renowned literary figures. The book provides a vivid depiction of

Madolduwa

Martin Wikramasinghe

Madolduwa is one of the greatest books ev ... mazing child hood. It shows a story of a young lad who try to start a whole different ... ent also.....

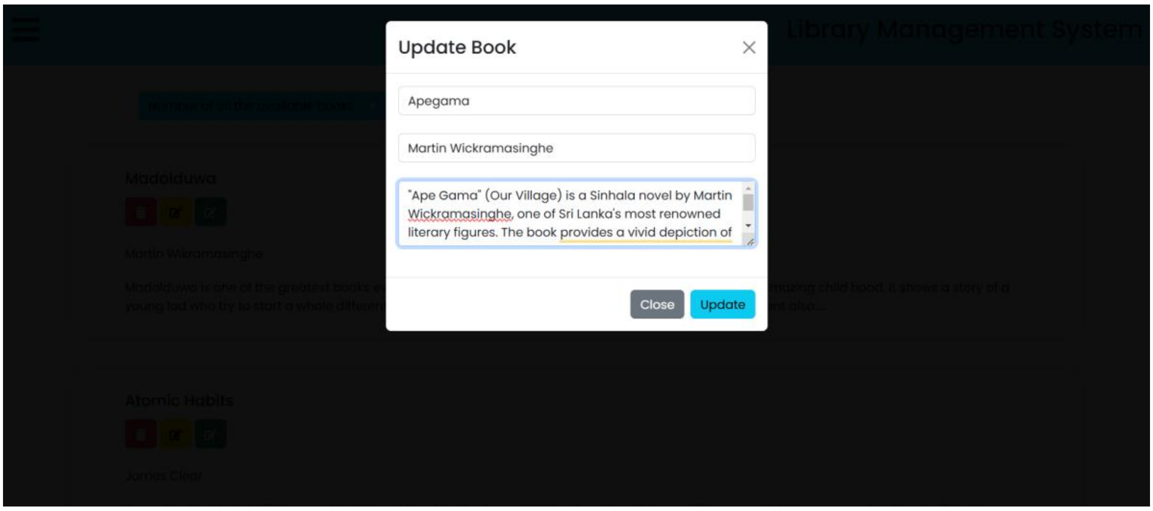
Atomic Habits

James Clear

A practical guide to building good habits and breaking bad ones, this book explores how small, incremental changes can lead to significant

Figure 11: filling input fields and adding a book

Using the update book pop-up books can be updated. To open that update pop-up can click on the yellow coloured update icon in the book card.



Update Book

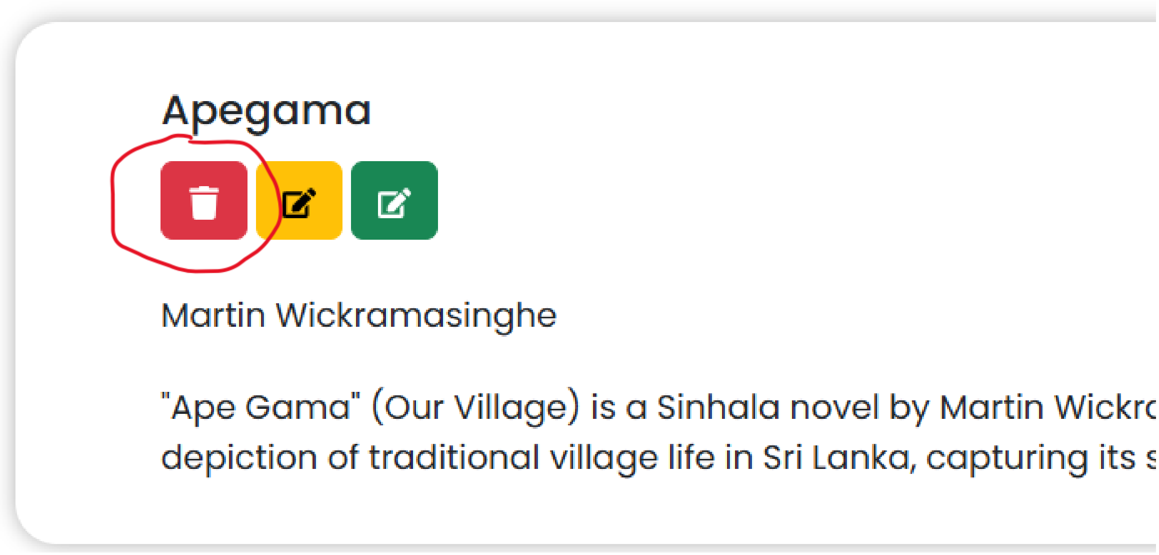
Apegama

Martin Wickramasinghe

"Ape Gama" (Our Village) is a Sinhala novel by Martin Wickramasinghe, one of Sri Lanka's most renowned literary figures. The book provides a vivid depiction of

Figure 12: update book pop-up

Clicking the delete button in the will delete the books.



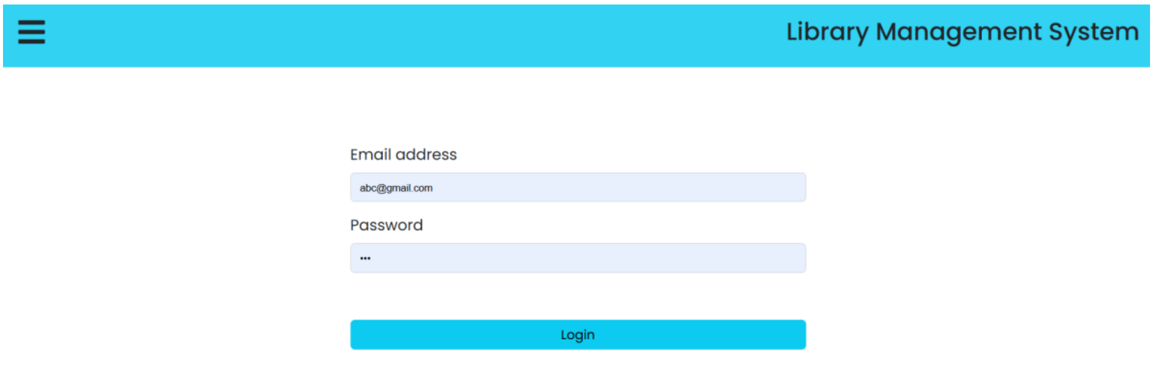
Apegama

Martin Wickramasinghe

"Ape Gama" (Our Village) is a Sinhala novel by Martin Wickr
depiction of traditional village life in Sri Lanka, capturing its

Figure 13: delete function of books

A **Login Interface** was implemented to provide secure access to the system. This feature ensures that only authorized personnel can manage library operations, safeguarding sensitive information. The login interface includes error handling mechanisms, such as alerts for incorrect credentials, enhancing usability and security.



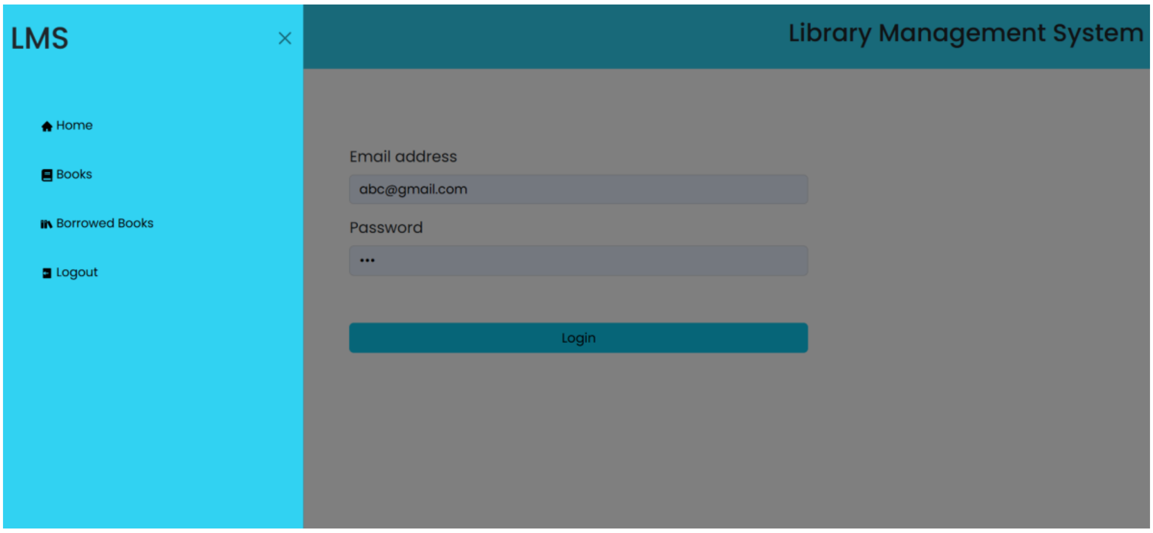
Library Management System

Email address
abc@gmail.com

Password
•••

Figure 14: login page of the library management system

Using a side navigation bar with a hamburger menu will help to smooth navigation between the screens.



Email address
abc@gmail.com
Password
•••

Figure 15: navigation bar with hamburger menu

Additional design considerations included the use of clear typography, consistent colour schemes, and visual indicators to enhance readability and reduce cognitive load. Hover effects and tooltips were added to guide users through less familiar features, ensuring a smoother learning curve for first-time users.

Overall, the user interface design was driven by the principles of clarity, efficiency, and accessibility, ensuring that library staff could perform their tasks with ease and confidence. By leveraging React’s flexibility and CSS's styling capabilities, the interface achieved a balance between functionality and aesthetic appeal, making it a valuable tool for small school libraries.

2.4 Implementation Tools/languages

The implementation of the Library Management System involved the strategic use of modern and widely adopted tools and programming languages to ensure efficiency, scalability, and maintainability. These tools were chosen not only for their technical capabilities but also for their popularity and community support, which facilitated development and problem-solving.

The frontend was developed using React, a powerful and versatile JavaScript library known for its efficiency in building user interfaces. React's component-based architecture allowed for modular and reusable code, making it easier to manage and scale the application as new features were added. React's virtual DOM also contributed to enhanced performance, ensuring smooth user interactions and quick rendering of components, even with a moderate amount of data.

The backend was built using Express.js, a lightweight and flexible framework for Node.js. Express.js streamlined the development of RESTful APIs, enabling efficient communication between the frontend and the database. Its middleware capabilities allowed for handling requests, routing, and error handling seamlessly, making it an ideal choice for this project.

For the database, MongoDB was utilized due to its robustness and flexibility. As a NoSQL database, MongoDB allowed for the efficient storage and retrieval of library records in a document-oriented format. This design was particularly beneficial for managing the dynamic data structures often required in library management systems. MongoDB's scalability also ensured that the system could handle increased data loads as the library grew[6].

Several other tools complemented the development process:

- Node Package Manager (NPM) was indispensable for managing dependencies and libraries, ensuring that the application used up-to-date and reliable packages.
- Postman was employed for testing API endpoints, allowing the developers to simulate real-world requests and responses and ensuring the backend services operated as expected.

- Git and GitHub were essential for version control and collaborative development. Git enabled efficient tracking of changes in the codebase, while GitHub facilitated remote collaboration, code reviews, and issue tracking. These tools ensured the project remained organized and on schedule[7].

This combination of tools and technologies was chosen deliberately to address the specific requirements of small school libraries. By leveraging modern development practices and technologies, the system was designed to be user-friendly, efficient, and scalable. Additionally, the selection of these tools provided a solid foundation for future enhancements, ensuring the system remains relevant as the needs of the users evolve. This comprehensive approach underscores the commitment to delivering a high-quality solution tailored to its intended audience.

CHAPTER THREE

Results and Discussion

The development of the *Library Management System* successfully met its objectives, demonstrating the system's effectiveness in addressing the challenges faced by small school libraries. The implemented functionalities, such as CRUD operations for book management, user authentication, and book count tracking, worked seamlessly during testing and deployment. Both the frontend, designed with React, and the backend, powered by Express.js, showed excellent performance, ensuring a smooth and efficient user experience.

The system's usability was a standout feature, with users appreciating the simplicity and responsiveness of the interface. Built with accessibility in mind, the application adapted well to various devices, making it versatile for different user environments. Performance metrics indicated that operations such as searching for books and updating records were quick and reliable, while database interactions handled moderate data volumes efficiently. Security measures, particularly access control, ensured that library records were protected from unauthorized access.

Overall, the *Library Management System* successfully addressed the core problems of disorganization and inefficiency in manual library management methods. While the system achieved its intended goals, user feedback and further evaluation highlighted additional features and refinements that could enhance the system's functionality in the future.

Evaluation

3.1 Achievements

The system achieved significant milestones, including the delivery of a fully functional web-based library management platform specifically tailored for small school libraries. The integration of secure login functionality and comprehensive CRUD operations provided a robust foundation for managing library records. Additionally, the system's responsive design ensured accessibility across various devices, meeting the diverse needs of users. This project also demonstrated how modern technologies like React and Express.js could effectively solve real-world problems in a cost-efficient manner.

3.2 Feedback from users before and after the product deployed

Before deployment, initial feedback emphasized the importance of simplicity and usability, given that most intended users lacked advanced technical knowledge. This feedback guided the design of the interface, ensuring an intuitive layout that reduced the learning curve. After deployment, users expressed satisfaction with the time-saving features and ease of navigation provided by the system. Suggestions from users included the need for enhanced dashboards with more detailed statistics and the potential integration of a mobile application to further expand accessibility.

3.3 Problem refinements

Despite its success, the project required several refinements during development. Issues such as database connectivity errors during initial testing phases were resolved promptly, and the login module was improved to provide clear and informative error messages for incorrect credentials. These refinements not only addressed immediate concerns but also enhanced the overall reliability and usability of the system.

3.4 Further refinements

Future refinements were identified based on user feedback and system evaluations. These include the introduction of a book categorization feature to improve searchability, as well as the integration of a student-access module, allowing students to view and reserve books online. Expanding the dashboard to include advanced analytics, such as borrowing trends and overdue book tracking, was also proposed.

3.5 Additional areas to be considered

The above-mentioned areas are among the many which have the capability of making Library Management System a solid, effective and very usable system. One such improvement would be mobile application development so that users are better accessible and provided convenience to a greater extent. This would make it possible for the staff and students to access the system through their mobile devices and tablets and thereby handle tasks like checking for availability of a book or track-wise transactions in realtime.

A further important addition would be, of course, multi-language support to address the different tongues of the constituents of different schools. With that, the system would be user-friendly and it would include several languages to schools located in different multilingual areas and thus give the possibility of school personnel to interact with the system in many languages. This would also mean an increase in the adoption and satisfaction rates with this functionality among a comparatively much larger number of schools.

Advanced analytics tools will also realize an optimized future haven for library operations. These analytics features can provide such things as borrowing trends, lists of popular books and even forecasts of future demand based on past experience. In other words, they will enable a library administrator to make decisions that are based on evidence acquired from data, whether it be purchasing new books or reallocating resources, which will lead to greater efficiency and improved library services.

Lastly the system should also throw features such as book categorization and elaborate dashboards for improving usability. With categorization, much ease in finding and managing books would be facilitated, while dashboards would present a summarized quick view of metrics such as overdue books, currently borrowed ones, among others.

CHAPTER FOUR

Conclusion and Further Work

The Library Management System duly met the small school libraries' effective provisions for affordable, familiar, and most importantly, user-friendly approaches to handling small book collections and operations of library management. Its primacy sets up user authentication, crate-read-update-delete (CRUD) for books, and transaction tracking, all of which streamline day-to-day tedious library work, enhance its organization, and promise data security. Built using state-of-the-art technologies like React, Express.js, and MongoDB Atlas, the system has fulfilled the requirements of scalability, maintenance, and adaptability to the dynamic needs.

That was one important aspect which the project highlighted - solutions tailor-made for particular user groups. In this respect, the whole idea of automating certain small school libraries, which do not have scores and scales recorded for such automation, would deter them from learning overheads. All the praise from the users' feedback went on to validate the design and functionality of the system. Many users, especially of the library management system, expressed much satisfaction with its ease of use and its role in making library management tasks easier.

However, that which was achieved by this project still leaves some things to be done nimbly and in a very fast pace, such as identifying the remaining work to include functions like book classification and trend of borrowing analytics. Future work would be an integrative approach with the student access component where students could reserve books online, as well as developing a mobile app to allow access by the users in a more mobile manner through their smartphones or tablets. A possible language expansion could also be considered, which may enable the system to fit better in a multi-lingual school environment and therefore make it more inclusive.

In a nutshell, the library management system is the first step towards modernizing the process of library management in small schools. It will handle their specific challenges and

Reference

[1] E. B. Ayo, R. N. Jotic, A. Raqueño, J. V. G. Loresca, I. F. Mendoza, and P. V. M. Baroña, "Development of an Integrated Library Management System (iLMS)," *Int. J. Interact. Mob. Technol.*, vol. 17, no. 10, pp. 242–256, 2023.

[2] J. Tramullas, A. I. Sánchez-Casabón, and P. Garrido-Picazo, "Studies and analysis of reference management software: A literature review," *Prof. la Inf.*, vol. 24, no. 5, pp. 680–688, 2015.

[3] B. R. Adebayo, "Library management system with topic modelling and its adaptability to open and distance learning libraries," *Libr. Philos. Pract.*, vol. 2019, 2019.

[4] "atlas-database @ www.mongodb.com." .

[5] "a137779f12581db98a460892a40df0833afea654 @ www.geeksforgeeks.org." .

[6] R. Aguilar Vera, A. Naal Jácome, J. Díaz Mendoza, and O. Gómez Gómez, "NoSQL Database Modeling and Management: A Systematic Literature Review," *Rev. Fac. Ing.*, vol. 32, no. 65, p. e16519, 2023.

[7] Y. Perez-Riverol *et al.*, "Ten Simple Rules for Taking Advantage of Git and GitHub," *PLoS Comput. Biol.*, vol. 12, no. 7, pp. 1–11, 2016.