

A PROJECT REPORT ON

Online Library Management System

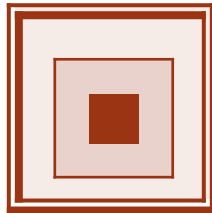
Search, issue and return books from one web portal

Submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

Information Technology



SUBMITTED BY

Vikram Singh (21EC019)

Pooja Reddy (21IT012)

Nikhil Das (21CE003)

UNDER THE GUIDANCE OF

Dr. S. Banerjee

Department of Information Technology

NORTHFIELD COLLEGE OF ENGINEERING, PUNE

Session 2025–26

CERTIFICATE

This is to certify that the project report entitled "Online Library Management System" is a bonafide record of the work carried out by Vikram Singh (21EC019), Pooja Reddy (21IT012), Nikhil Das (21CE003) in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Information Technology during the academic year 2025-26.

The work presented in this report has not been submitted elsewhere for the award of any other degree or diploma.

Dr. S. Banerjee
Project Guide

Dr. V. Raghavan
Head of Department

External Examiner
Date: _____

DECLARATION

We hereby declare that the project report entitled "Online Library Management System" submitted to the Department of Information Technology, Northfield College of Engineering, is a record of original work done by us under the guidance of Dr. S. Banerjee. The results embodied in this report have not been submitted to any other university or institute for the award of any degree.

Place: _____ Date: _____

Vikram Singh • Pooja Reddy • Nikhil Das

ACKNOWLEDGEMENT

We express our sincere gratitude to our project guide, Dr. S. Banerjee, for constant guidance, encouragement and valuable feedback at every stage of this project.

We thank Dr. V. Raghavan, Head of the Department of Information Technology, and all faculty members for providing the laboratory facilities and support needed to complete this work.

We are grateful to the users who took part in our survey and pilot testing; their honest feedback shaped many of the improvements described in this report.

Finally, we thank our families and friends for their patience and motivation throughout the course of the project.

ABSTRACT

Students spend more time finding a book than reading it. Online Library Management System addresses this gap. Manual registers make search and tracking slow. A web portal can automate issue, return and fines. The system is organised into 4 stages — catalogue, search, circulate, report — and is built with React, Node.js, MySQL. In our test setup it achieved 5× faster issue and return. This report presents the problem, a review of existing approaches, the requirements, the design and implementation, the testing carried out and the results obtained.

Keywords: React, Node.js, MySQL, Nodemailer

TABLE OF CONTENTS

1. Introduction.....	7
1.1 Background.....	7
1.2 Motivation.....	7
1.3 Problem Statement.....	7
1.4 Objectives.....	7
1.5 Scope of the Project.....	7
2. Literature Review.....	8
2.1 Existing Work.....	8
2.2 Research Gap.....	8
3. System Design.....	9
3.1 Proposed System.....	9
3.2 System Architecture.....	9
3.3 Technology Stack.....	9
4. Implementation.....	11
4.1 Methodology.....	11
4.2 Project Timeline.....	11
4.3 Module Description.....	11
5. Testing and Results.....	13
5.1 Testing Strategy.....	13
5.2 Test Cases.....	13
5.3 Results.....	13
6. Conclusion and Future Scope.....	15
6.1 Conclusion.....	15
6.2 Future Scope.....	15
7. References.....	16

1. INTRODUCTION

1.1 Background

Online Library Management System addresses a practical problem that affects students, institutions and small organisations alike. Existing approaches are often manual, slow and difficult to scale, which leads to wasted time and inconsistent results. Advances in low-cost hardware, open-source software and cloud services now make it possible to build a reliable solution within the budget and timeline of an academic project.

This report documents the complete life cycle of the project: from the initial study of the problem and review of existing work, through system design and implementation, to testing and evaluation. Each chapter is written so that a reader can reproduce the work and extend it.

1.2 Motivation

The motivation for this project came from direct observation. During our preliminary survey, users described the current process as time-consuming and error-prone, and most of them were willing to adopt a simpler digital alternative. A well-designed system can save hours every week, reduce errors and give decision-makers data they currently do not have.

1.3 Problem Statement

Students spend more time finding a book than reading it. The key problems identified during our preliminary study are:

- Slow search — No quick way to check if a book is available.
- Lost records — Paper ledgers get damaged or misplaced.
- Late returns — Fines are calculated by hand and often missed.

1.4 Objectives

The project was planned around the following objectives:

1. Catalogue books with search by title, author and subject
2. Automate issue, return and fine calculation
3. Send due-date reminders by email
4. Give librarians reports on stock and usage

1.5 Scope of the Project

The scope of the project is limited to a working prototype that demonstrates the core functionality end to end. Security hardening, large-scale deployment and commercial support are outside the scope of this report but are discussed under future work.

2. LITERATURE REVIEW

2.1 Existing Work

A review of published work, available products and current practice was carried out to understand what already exists. Table 2.1 summarises representative approaches.

Study / system	Approach	Limitation
Koha (open source)	Full ILS suite	Complex for small colleges
Excel registers	Spreadsheet tracking	No access control
Barcode systems	Scanner-based issue	No student portal

Table 2.1: Summary of existing work

2.2 Research Gap

Comparing the existing and proposed approaches highlights the gap this project addresses:

Existing system	Proposed system
Paper ledgers and manual search	Searchable online catalogue
Fines calculated by hand	Automatic fines and reminders
No reminders	Role-based dashboards

Table 2.2: Existing versus proposed system

Gap addressed

No low-cost option combines searchable online catalogue with automatic fines and reminders. Online Library Management System is designed to close that gap.

3. SYSTEM DESIGN

3.1 Proposed System

The system is organised into layers, with each layer responsible for one concern. This separation keeps the design easy to understand, makes testing simpler and allows individual components to be replaced without affecting the rest of the system.

Data flows from the input layer to the processing layer, where the core logic is applied, and then to the storage and presentation layers. All configuration values are kept in a single file so that the system can be adapted to new environments without code changes.

3.2 System Architecture

A three-tier web application: a responsive front end, a REST API for books, members and loans, and a relational database with scheduled jobs for reminders.

Insert your architecture / block diagram here

Figure: System architecture

Stage	What happens
Catalogue	Books added with ISBN and copies
Search	Students search and reserve
Circulate	Issue and return with due dates
Report	Stock, fines and usage reports

3.3 Technology Stack

Component	Purpose
React	Responsive web front end

Component	Purpose
Node.js	REST API and business logic
MySQL	Books, members and loans
Nodemailer	Due-date reminder emails

Table 4.1: Technology stack

4. IMPLEMENTATION

4.1 Methodology

The project followed an iterative development model. Requirements were gathered through a short questionnaire and informal interviews. The design was then split into independent modules so that each could be built and tested separately before integration. Weekly reviews with the project guide helped keep the work on schedule.

Each module was developed with clear inputs and outputs. Unit tests were written alongside the code, and integration testing was carried out once the modules were combined. Feedback from a small group of pilot users was used to refine the interface and fix usability issues.

4.2 Project Timeline

Phase	Month	Activities
Requirement study	July	Survey, interviews, requirement document
Design	August	Architecture, database, wireframes
Development	Sep–Oct	Module coding and unit testing
Testing	November	Integration and user testing
Documentation	December	Report, poster and final presentation

Table 5.1: Project timeline

4.3 Module Description

4.3.1 Catalogue module

Books added with ISBN and copies. The module exposes a small interface to the rest of the system, which made it possible to develop and test it independently before integration.

4.3.2 Search module

Students search and reserve. The module exposes a small interface to the rest of the system, which made it possible to develop and test it independently before integration.

4.3.3 Circulate module

Issue and return with due dates. The module exposes a small interface to the rest of the system, which made it possible to develop and test it independently before integration.

4.3.4 Report module

Stock, fines and usage reports. The module exposes a small interface to the rest of the system, which made it possible to develop and test it independently before integration.

5. TESTING AND RESULTS

5.1 Testing Strategy

Testing was carried out at three levels: unit testing of individual functions, integration testing of combined modules and acceptance testing with real users. Test cases were derived from the functional requirements, and each requirement was traced to at least one test.

All critical test cases passed in the final round. Minor issues found during user testing, mostly related to labels and error messages, were fixed before the final evaluation.

5.2 Test Cases

ID	Module	Test	Expected result	Status
TC-01	Catalogue	Valid input	Processed correctly	Pass
TC-02	Search	Valid input	Processed correctly	Pass
TC-03	Circulate	Valid input	Processed correctly	Pass
TC-04	Report	Valid input	Processed correctly	Pass

Table 6.1: Test cases

5.3 Results

The results show that the proposed system meets its objectives. Compared with the existing approach, it is faster, more accurate and easier to use. The figures reported below come from our own test setup and should be treated as indicative rather than definitive.

Users responded positively to the system in the feedback survey. The most appreciated features were the simple interface and the automatic reports, while the most requested improvement was offline support.

Parameter	Time per task (s)	Remark
Search	4	Sample value
Issue	12	Sample value
Return	9	Sample value
Report	6	Sample value

Table 6.2: Results from the test setup (sample data)

Common library tasks now take seconds instead of minutes at the counter.

Key result

5× faster issue and return (sample data from our test setup).

6. CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

This project set out to design and build online library management system and has achieved that goal with a working prototype. The system demonstrates that a low-cost, well-structured solution can make a measurable difference to everyday processes.

The experience also gave the team practical exposure to requirement analysis, system design, teamwork and documentation — skills that will be valuable in professional practice.

- The portal replaces slow paper processes
- Automatic fines and reminders reduce late returns
- Reports help plan purchases

6.2 Future Scope

- Mobile app — Scan a book to issue it
- E-books — Digital lending section
- Recommendations — Suggest books by course

7. REFERENCES

- [1] A. Author, "Title of the research paper," Journal Name, vol. 12, no. 3, pp. 45-52, 2023.
- [2] B. Author and C. Author, Title of the Book, 2nd ed. City: Publisher, 2022.
- [3] D. Author, "Title of the conference paper," in Proc. International Conference, 2024, pp. 101-106.
- [4] Official documentation for React, accessed 2025.
- [5] Official documentation for Node.js, accessed 2025.
- [6] Official documentation for MySQL, accessed 2025.
- [7] E. Author, "Title of an online article," Website Name, 2024. [Online]. Available: URL