

Library Management System

Vishal Patil¹, Manoj Nikum²

Department of Computer Engineering

Shri Jaykumar Rawal Institute of Technology, Dondaicha, Maharashtra, India

Abstract—Manual library management processes, including book cataloguing, issue-return tracking, and fine calculation, remain time-consuming and error-prone in many academic institutions, leading to inefficient resource utilization and poor user experience for students and staff. This paper presents the design and development of a web-based Library Management System (LMS) intended to automate core library operations including book cataloguing, member registration, issue and return transactions, fine computation, and inventory reporting. The system was developed using a three-tier architecture with a relational database backend, a server-side application layer implementing business logic and validation rules, and a browser-based user interface supporting distinct librarian and student roles. Key features include real-time book-availability search, automated due-date and fine calculation, email/SMS-based due-date reminders, and administrative dashboards for inventory and usage reporting. The system was evaluated through functional testing across core use cases and a usability survey administered to library staff and student users. Results indicate that the developed system reduced average book issue-return transaction time by approximately 70% compared to the manual register-based process, eliminated common manual fine-calculation errors, and received favorable usability ratings from both librarian and student user groups. The findings demonstrate that a well-designed, role-based LMS can meaningfully improve the efficiency, accuracy, and user experience of academic library operations.

Index Terms—*Library Management System, Automation, Book Cataloguing, Web Application, Three-Tier Architecture, Database Management, Usability Evaluation, Academic Libraries.*

I. INTRODUCTION

Academic libraries serve as a critical resource hub for students and faculty, yet many small and mid-sized institutions continue to rely on manual, register-based processes for cataloguing books, tracking issues and returns, and calculating overdue fines. These manual processes are time-consuming, prone to human error, and offer limited visibility into inventory status, member borrowing history, or overdue items, ultimately reducing the efficiency of library operations and the quality of service provided to users.

Library Management Systems (LMS) address these limitations by digitizing and automating core library workflows, providing real-time book-availability information, automated transaction logging, and administrative reporting capabilities. While commercial and open-source LMS solutions exist, many are either cost-prohibitive or overly complex for the specific operational needs of smaller academic institutions.

This paper presents the design, development, and evaluation of a lightweight, web-based Library Management System tailored to the operational needs of an academic institution library, covering book cataloguing, member management, issue-return

III. SYSTEM DESIGN

A. System Architecture

The Library Management System was designed using a three-tier architecture comprising a presentation layer (browser-based user interface), an application layer (server-side business logic and validation), and a data layer (relational database). This separation of concerns supports maintainability and allows independent scaling of each layer.

B. User Roles and Access Control

The system supports two primary user roles: Librarian (Administrator), with full access to cataloguing, member management, transaction processing, and reporting functions; and Student/Member, with access limited to catalogue search, personal borrowing history, and due-date/fine information.

C. Database Design

The relational database schema comprises core entities including Books, Members, Transactions (Issue/Return), Fines, and Categories, with appropriate foreign-key relationships to maintain referential integrity across cataloguing and transaction records.

D. Core Functional Modules

Key functional modules include book cataloguing

transaction processing, automated fine calculation, and administrative reporting.

II. LITERATURE REVIEW

A. Traditional Library Automation Systems

Early library automation efforts focused primarily on digitizing catalogue records through Machine-Readable Cataloguing (MARC) standards, improving searchability of library holdings but offering limited support for end-to-end transaction automation.

B. Web-Based and Cloud LMS Solutions

More recent literature has explored web-based and cloud-hosted LMS architectures, emphasizing benefits such as remote accessibility, centralized data management, and integration with barcode or RFID-based book identification for faster transaction processing.

C. Usability Considerations in Library Systems

Studies on library-system usability consistently emphasize the importance of role-based interface design, distinguishing administrative and end-user workflows, and highlight search-interface responsiveness and clarity of due-date and fine information as key determinants of user satisfaction.

and inventory management, member registration and profile management, issue-return transaction processing with automated due-date assignment, fine calculation based on configurable overdue-rate rules, and administrative dashboards for inventory and circulation reporting.

An overview of the core functional modules and their responsibilities is presented in Table I.

TABLE I — Core Functional Modules of the Library Management System

Module	Primary User	Key Functionality
Book Cataloguing	Librarian	Add, edit, delete, and search book records
Member Management	Librarian	Register and manage student/staff member profiles
Issue-Return Processing	Librarian	Record book issue and return transactions with due-date assignment
Fine Calculation	System (automated)	Compute overdue fines based on configurable per-day rate
Catalogue Search	Student/Member	Search books by title, author, ISBN, or category
Reporting Dashboard	Librarian	View inventory status, circulation statistics, and overdue reports

IV. IMPLEMENTATION

The system was implemented using a standard web-application stack, with the presentation layer built using HTML, CSS, and JavaScript, the application layer implemented in a server-side scripting framework handling request routing, business logic, and input validation, and MySQL used as the relational database management system for persistent storage of catalogue, member, and transaction data.

Book records support fields for title, author, ISBN, category, and available copy count, with real-time availability updates reflected upon each issue and return transaction. Fine calculation is automated

V. TESTING AND EVALUATION METHODOLOGY

A. Functional Testing

Functional testing was conducted across core use cases including book addition/edit/deletion, member registration, book issue and return transactions, fine calculation accuracy, and catalogue search, using both valid and boundary-condition test inputs to verify correct system behavior.

B. Performance Testing

Transaction processing time for book issue and return operations was measured and compared against the average time required for equivalent manual register-

based on a configurable per-day overdue rate, applied automatically at the time of return or upon administrative query of overdue items.

VI. RESULTS AND DISCUSSION

A. Functional Testing Results

All core functional test cases, including book cataloguing, member registration, issue-return processing, and fine calculation, passed verification against expected outcomes, with no critical defects identified during the testing cycle. Results of the functional test coverage are summarized in Table II.

B. Transaction Time Improvement

Average transaction time for a book issue-return cycle was reduced from approximately 4.2 minutes under the manual register-based process to approximately 1.3 minutes using the developed system, representing an approximate 70% reduction in per-transaction processing time.

C. Fine Calculation Accuracy

Automated fine calculation eliminated the manual arithmetic errors observed in approximately 12% of sampled manual fine calculations prior to system deployment, ensuring consistent and accurate application of overdue charges across all transactions.

based transactions, based on observation of library staff performing representative transactions under both systems.

C. Usability Evaluation

A usability survey based on the System Usability Scale (SUS) was administered to a group of library staff and student users following a supervised trial period, covering ease of navigation, clarity of information presentation, and overall satisfaction with the system.

D. Usability Survey Results

The System Usability Scale survey yielded an average score of 82.4 out of 100 across librarian and student respondents, indicating a favorable usability rating, with particularly positive feedback on catalogue search responsiveness and clarity of due-date and fine information display.

VII. COMPARATIVE SUMMARY

Table III presents a comparative summary of key operational metrics between the manual register-based process and the developed Library Management System, highlighting improvements in transaction time, error rate, and reporting capability.

VIII. CHALLENGES AND LIMITATIONS

Challenges encountered during development included designing a fine-calculation rule engine flexible enough to accommodate different book categories and member types, and ensuring adequate input validation to prevent duplicate ISBN entries and inconsistent member records. The evaluation was conducted with a limited user group over a short trial period, and further evaluation across a larger user base and extended operational period would strengthen confidence in the reported improvements.

TABLE II — Functional Test Coverage Summary

Test Case Category	Test Cases Executed	Passed	Failed
Book cataloguing (add/edit/delete)	12	12	0
Member registration and profile management	8	8	0
Issue-return transaction processing	15	15	0
Fine calculation	10	10	0
Catalogue search	9	9	0

TABLE III — Comparative Summary: Manual Process vs. Developed System

Metric	Manual Register-Based Process	Developed LMS
Avg. issue-return transaction time	4.2 minutes	1.3 minutes
Fine calculation error rate	~12% of sampled transactions	0% (automated)
Real-time book availability visibility	Not available	Available
Reporting capability	Manual, ad hoc	Automated dashboard reports
Usability rating (SUS score)	Not applicable	82.4 / 100

IX. FUTURE SCOPE

Future enhancements to the system could include integration of barcode or RFID-based book scanning to further reduce transaction time, a mobile application interface for student catalogue search and notification, integration with institutional single sign-on systems for streamlined member authentication, and recommendation features suggesting relevant books based on borrowing history. Cloud deployment and multi-branch support would also extend the system's applicability to larger institutional library networks.

X. CONCLUSION

This paper presented the design, development, and evaluation of a web-based Library Management System intended to automate core academic library operations including cataloguing, member management, issue-return processing, and fine calculation. Functional testing confirmed correct system behavior across core use cases, and evaluation results showed an approximately 70% reduction in transaction processing time, elimination of manual fine-calculation errors, and favorable usability ratings from librarian and student users. These findings support the practical value of role-based, web-based library automation for improving operational efficiency and user experience in academic library settings, with several promising directions identified for future system enhancement.

XI. REFERENCES

- [1] Reitz, J.M. (2014). Online Dictionary for Library and Information Science. Libraries Unlimited.
- [2] Breeding, M. (2019). Library Systems Report 2019: New Technologies, New Models. American Libraries Magazine.
- [3] Ranganathan, S.R. (1931). The Five Laws of Library Science. Madras Library Association.
- [4] Kumar, P.S.G. (2009). Fundamentals of Library and Information Science. B.R. Publishing Corporation.
- [5] Pressman, R.S., and Maxim, B.R. (2019). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill Education.
- [6] Sommerville, I. (2015). Software Engineering (10th ed.). Pearson Education.
- [7] Brooke, J. (1996). SUS: A quick and dirty usability scale. Usability Evaluation in Industry, 189(194), 4-7.
- [8] Elmasri, R., and Navathe, S.B. (2015). Fundamentals of Database Systems (7th ed.). Pearson Education.
- [9] W3C. (2020). Web Content Accessibility Guidelines (WCAG) 2.1. World Wide Web Consortium.
- [10] Kroenke, D.M., and Auer, D.J. (2015). Database Processing: Fundamentals, Design, and Implementation (14th ed.). Pearson Education.
- [11] Duckett, J., (2014). HTML and CSS: Design and Build Websites. John Wiley & Sons.
- [12] Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley.