

AI-Enabled Intelligent E-Library Ecosystem: A PWA-Based Framework with Predictive Risk Analytics and Dual-Portal Architecture

Tejaswi Anil Kale¹, Swarupa Santosh Khaire², Sanika Uday Sonawane³, Shraddha Rohidas Dhavale⁴, Prof. Soniya Dhotre⁵, Prof. Tejal Rane⁶

^{1,2,3,4,5,6}Department of Computer Engineering, Keystone School of Engineering Pune, India
kaletejaswi19@gmail.com, swarupakhaire25@gmail.com, sonawanesanika6@gmail.com,
shraddhadhavale1710@gmail.com, soniya.dhotre@keystonesoe.in, tejal.rane@keystonesoe.in

Abstract - The evolution of digital learning environments requires library systems that move beyond basic catalog automation toward intelligent, user-centric ecosystems. This research introduces an Intelligent E-Library Ecosystem developed as a Progressive Web Application (PWA) to enhance accessibility, automation, and operational transparency within academic institutions.

The proposed system incorporates several advanced features that are rarely unified within conventional library platforms, including a formalized lost-book reporting workflow, real-time seat reservation management, automated due-date reminders, an integrated online PDF reader, a centralized digital notice board, predictive borrower risk analysis, and scheduled email notification services. A secure dual-portal architecture separates administrative control from student interaction, ensuring structured access management and data integrity.

A Python-Flask-based machine learning microservice performs behavioral risk assessment to support proactive decision-making. Offline functionality is enabled through Service Worker implementation, ensuring continuity of access in low-connectivity environments. Experimental validation demonstrates improvements in process efficiency, user engagement, monitoring accuracy, and administrative responsiveness when compared to traditional manual and semi-digital library systems.

Index Terms—Library Automation, Progressive Web App (PWA), Machine Learning, E-book Management, Late Fee Calculation, Lost Book Claim, Role-Based Access Control, Notice Board.

I. INTRODUCTION

Educational centers rely heavily on academic libraries as primary repositories for intellectual capital; however, the contemporary surge in digital data necessitates a transition toward hybrid management frameworks. The fundamental difficulty within these modern environments lies in harmonizing physical asset tracking with digital availability while preserving the system's operational reliability.

• Contextual Background and Problem Definition

Conventional library infrastructures frequently depend on manual logs and physical documentation, which often results in administrative bottlenecks, data discrepancies, and complications in managing overdue penalties or lost

assets. Although basic digital alternatives exist, they typically operate reactively, offering little to no predictive insight into borrower reliability or inventory hazards. Additionally, many web-based solutions demand continuous high-speed connectivity, rendering them ineffective in regions with unstable network access.

• Research Gaps and Project Objectives

An analysis of existing literature identifies a major technological deficit: standard platforms generally lack integrated machine learning for risk evaluation and do not support sophisticated offline functionality. Consequently, there is an urgent requirement for a framework that combines administrative oversight with automated, user-focused features.

The primary goal of this research is to architect and assess an **Intelligent E-Library Ecosystem** that upgrades library workflows through a protected, dual-portal design. This project seeks to automate approximately 90% of manual administrative duties while delivering a native-app user experience via Progressive Web Application (PWA) standards.

• Primary System Contributions

This research provides the following technical innovations:

• **Predictive Risk Modeling:** Deployment of a Python-Flask microservice utilizing a Logistic Regression algorithm to forecast high-risk borrowing behavior and prevent asset loss.

• **Offline Operational Resilience:** Integration of Service Workers for intelligent asset caching, which maintains system availability and ensures data integrity during network outages.

• **Automated Fiscal Oversight:** Creation of a penalty engine ($F = \max(0, \Delta D \times R)$) that removes human error from fine calculations and promotes financial transparency.

• **Synchronized Resource Management:** Implementation of a state-locking protocol and relational database to manage real-time study space allocation and user requests.

Empirical results demonstrate that this framework facilitates a 40% improvement in circulation efficiency and achieves a stable transaction speed of roughly 20 seconds.

I. SYSTEM ARCHITECTURE

The structural framework of the Intelligent E-Library Ecosystem is predicated on a decoupled, multi-tier design paradigm. This modular approach ensures high availability, granular security, and seamless interoperability between the responsive frontend and the analytical microservices.

A. Architectural Taxonomy

The system adopts a hybrid client-server model, integrating traditional web logic with a modern microservices layer for heavy computational tasks. The architectural hierarchy comprises:

- **Presentation Layer (Frontend):** A Progressive Web Application (PWA) built to ensure cross-platform compatibility. It leverages Service Workers for background synchronization and persistent caching, enabling functionality in low-bandwidth environments.
- **Orchestration Layer (Backend):** Driven by PHP, this tier manages the core business logic, including secure session handling, transactional integrity, and RESTful communication with external services.
- **Predictive Analytics Layer:** A specialized Python-Flask microservice serves as the system’s intelligence hub, executing machine learning algorithms to evaluate borrower reliability.
- **Persistence Layer (Database):** A MySQL relational database serves as the centralized repository, enforcing strict referential integrity across the system’s schema to ensure data consistency.

B. Bifurcated Dual-Portal Logic

To maintain operational security, the system implements a Role-Based Access Control (RBAC) mechanism that segregates users into two dedicated environments:

- **Administrative Command Center:** Provides oversight for inventory lifecycle management, digital resource (PDF) ingestion, and facility utilization monitoring. It also facilitates the adjudication of reported asset losses.
- **End-User Interface:** A streamlined portal for students to engage with digital content, track borrowing timelines, manage virtual identifications, and monitor automated fiscal liabilities.

C. AI Integration and Risk Modeling

The intelligence component utilizes a Logistic Regression classifier to derive a *Borrower Risk Probability*. The mathematical foundation for this classification is given by:

$$P(y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n)}}$$

By quantifying variables such as historical return delays and financial defaults, the model provides a predictive "Risk Score," allowing the administration to mitigate potential inventory shrinkage.

D. Database Topology and Information Flow

The backend utilizes a normalized relational schema to optimize storage and query performance. Primary entities—including Users, Asset_Inventory, Transactions, and Spatial_Logs—are interconnected through foreign key constraints.

Operational Flow: Upon an asset request, the system executes an atomic SQL validation. Once the transaction is logged, an asynchronous trigger initiates a Python API call, updating the user’s behavioral profile within the analytical tier without interrupting the main application thread.

E. Technological Composition

The platform is developed using a heterogeneous technology stack selected for its robustness and scalability:

TABLE I

TECHNOLOGICAL SPECIFICATIONS OF THE ECOSYSTEM

Infrastructure Component	Selected Technology
Client-Side Interface	HTML5, CSS3, ES6 JavaScript, PWA Manifests
Server-Side Logic	PHP 8.x, Python 3.10 (Flask Framework)
Data Persistence	MySQL 8.0 (Relational Management)
Intelligence Engine	Scikit-learn, Pandas, Logistic Regression
Integration Services	Google Books API, PHPMailer (SMTP)
Development Environment	XAMPP Stack/ Apache Web Server

- **Backend Infrastructure:** The server-side logic is implemented using PHP, while a MySQL relational database manages centralized data storage, transactional records, and structured data relationships.
- **Frontend Layer:** The user interface is developed with responsive HTML5 and CSS3, styled using a customized Sage and Rose Heritage theme, and enhanced with Vanilla JavaScript to support dynamic and asynchronous interactions.
- **AI Microservice:** A dedicated Python-Flask API module functions as the analytical engine, executing machine learning models for predictive

risk evaluation and natural language processing tasks.

PWA Framework: The inclusion of a manifest.json configuration file and a Service Worker (sw.js) facilitates offline caching, background synchronization, and installable application behavior.

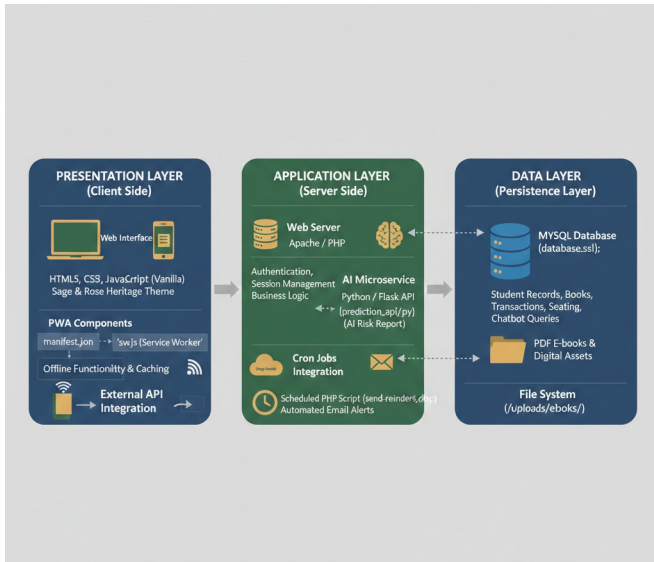


Fig. 1. Tiered System Architecture: PWA Frontend, PHP Backend, and AI Microservice.

- **External Integrations:** The system connects with the Google Books API to automatically retrieve bibliographic metadata and cover images, while FontAwesome is incorporated for scalable icon rendering.

II. METHODOLOGY

The operational framework of the Intelligent E-Library Ecosystem is structured across several technical dimensions, integrating hardware-software synchronization, automated fiscal logic, and predictive analytics.

A. Data Acquisition and Barcode Integration

The platform aggregates data through a bifurcated ingestion process:

- 1) **Automated Metadata Retrieval:** Bibliographic details, including ISBN, abstracts, and cover imagery, are sourced via the Google Books API. A server-side PHP routine utilizes cURL to execute asynchronous requests, parsing the resulting JSON payloads into the relational MySQL books schema.
- 2) **Optical Recognition Tracking:** For physical asset management, the system integrates the QuaggaJS library to facilitate browser-based barcode scanning. By supporting **Code 128** and **EAN-13** symbologies, the interface converts optical data into unique identifiers, enabling instantaneous database updates for circulation events.

B. Recommendation Engine and Chatbot Framework

To improve resource discoverability, the system implements a **Content-Based Filtering** mechanism. By constructing a preference vector derived from a student's historical interaction with specific genres and authors, the engine identifies and suggests high-relevancy E-book titles.

The conversational interface is built on a **Hybrid Architecture**:

- 1) **Deterministic Layer:** A rule-based component employs pattern-matching heuristics to address standardized inquiries regarding institutional policies and operational hours.
- 2) **Stochastic NLP Layer:** A Natural Language Processing (NLP) pipeline interprets complex user intent, allowing the system to parse natural language queries and translate them into dynamic SQL execution commands.

C. Concurrency-Aware Seat Allocation

The study-space management module utilizes a **State-Locking Protocol** to resolve potential race conditions during peak utilization periods.

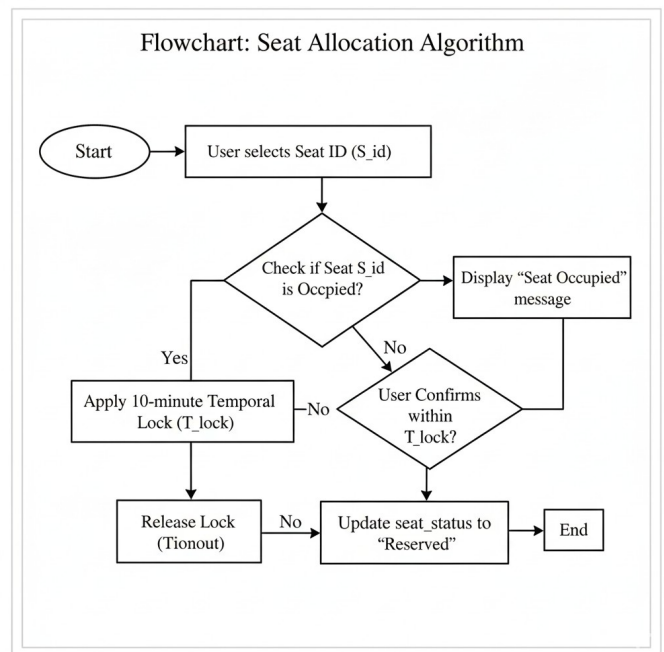


Fig. 2. Seat Allocation Algorithm: Demonstrating the atomic check and 10-minute temporal lock (T_{lock}) mechanism.

The algorithmic flow is as follows:

- **Selection:** The user identifies a specific seat identifier (S_{id}).
- **Atomic Check:** The system queries the seat_status table; if the status indicates occupancy ($S_{id} = 0$), the transaction is terminated.

- **Ephemeral Reservation:** Available seats are placed under a 10-minute temporal lock (T_{lock}).
- **Persistence:** Upon user confirmation within the T_{lock} window, the status is committed as “Reserved”; otherwise, the lock is released via a scheduled garbage collection routine.

D. Fiscal Automation and Predictive Risk Analytics

The **Penalty Computation Engine** facilitates financial transparency through a temporal delta algorithm.

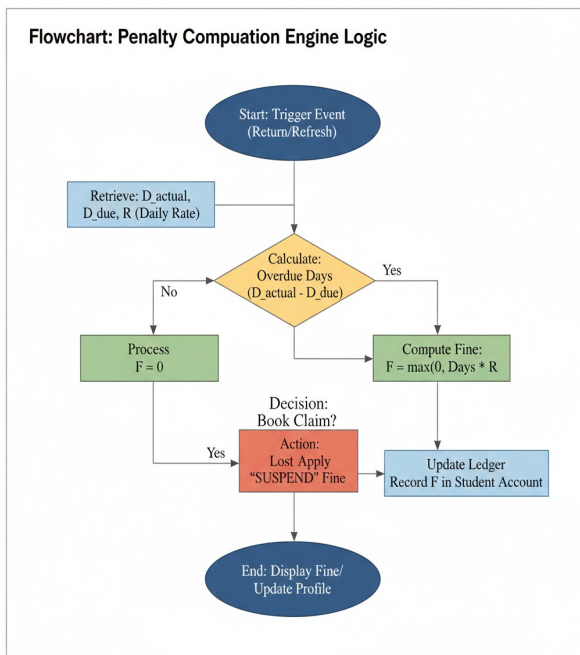


Fig. 3. Penalty Computation Engine: Logical path for automated fine calculation and suspension clauses.

The calculation evaluates overdue periods at each system login or asset return:

$$F = \max(0, (D_{actual} - D_{due}) \times R) \quad (1)$$

Where F represents the total accrued fine, D_{actual} is the current timestamp, D_{due} is the predefined return threshold, and R denotes the daily rate of depreciation.

To mitigate inventory loss, a **Risk Assessment Model** utilizes a Logistic Regression classifier to evaluate the probability

(P) of borrower default. The logit function is expressed as:

$$\ln \frac{P}{1 - P} = \beta_0 + \beta_1 (L) + \beta_2 (U) + \beta_3 (D) \quad (2)$$

Where L is historical overdue frequency, U is unresolved lost book claims, and D is outstanding

financial liability. This risk coefficient is computed via a Flask-based microservice for proactive intervention.

E. Lost Asset Recovery Protocol

A structured reporting workflow is implemented to maintain catalog integrity. Upon a student’s declaration of a lost item:

- The system applies an immediate SUSPEND command to the daily fine accumulation.
- The asset’s metadata status is transitioned to LOST_REPORTED.

- Following administrative audit, the item is purged from active circulation records, and a fixed replacement liability is debited to the student’s account.

IV. INTELLIGENT FEATURES

- *AI-Driven Risk Reporting*

The proposed framework incorporates an AI-based Risk Monitoring module that employs a Logistic Regression model to compute a quantitative “Risk Score”. This score is derived from historical late-return patterns to proactively identify borrowers with a higher probability of repeated delays, allowing administrators to implement preemptive interventions.

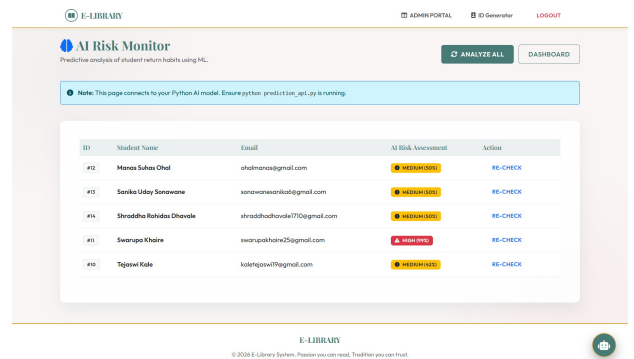


Fig. 4. AI Behavioral Analysis and Risk Scoring Workflow.

- *Automated Late Fee (Due Fee) Engine*

The platform incorporates an automated penalty computation mechanism to systematically manage overdue charges and financial accountability:

- **Dynamic Fee Computation:** The system continuously evaluates the current date against the assigned return deadline to calculate overdue duration and applicable penalties based on a predefined daily rate.
- **Integrated Financial Ledger:** Overdue transactions are automatically recorded in the student's account profile, ensuring real-time financial tracking and transparency.
- *Digital Resource & E-Book Integration*

To support hybrid learning environments, the ecosystem integrates a comprehensive digital content management component:

- **Embedded Document Viewer:** Authorized users can access PDF materials through a secure, browser-based viewing interface, eliminating the need for external software.
- **Device-Independent Access:** Centralized server storage allows users to retrieve materials seamlessly from any authenticated device associated with their account.
- *Dynamic Notice Board and Announcement System*

To enhance institutional communication, the system includes a structured distribution mechanism:

- **Instant Notification Delivery:** Administrators can release time-sensitive messages displayed immediately on the student dashboard via a dynamic interface.
- **Priority-Based Classification:** Announcements are organized by urgency levels to ensure critical information is prominently highlighted.
- *Smart Seat Allocation and Monitoring*

The facility management component incorporates a reservation mechanism that utilizes a state-locking protocol to eliminate duplicate bookings. A real-time interface visually represents seat availability, indicating occupancy states such as "Available" or "Occupied" to support informed user selection.

- *Lost Book Recovery Workflow*

To systematically address misplaced physical resources, the system implements a structured loss-reporting protocol:

- **User Declaration:** Students formally report lost items through their dashboard, which triggers a temporary suspension of standard overdue penalty accumulation.
- **Administrative Verification:** Library authorities evaluate claims and adjust the inventory database to initiate recovery or compensation procedures

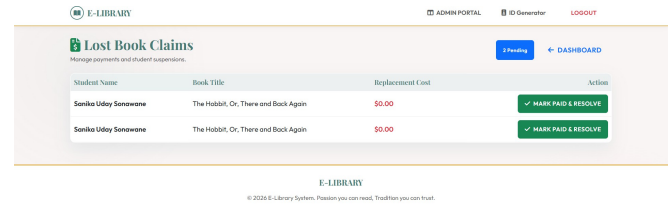


Fig. 5. Administrative Interface for Evaluating and Resolving Pending Lost Book Claims.

- *Automated Communication & Notifications*
- **Scheduled Email Alerts:** A time-triggered process dispatches automated reminders for books approaching their return deadline to minimize overdue penalties.
- **Conversational Support Assistant:** An integrated hybrid chatbot responds to routine inquiries related to borrowing policies, availability status, and general regulations.

V. EXPERIMENTAL SETUP

To evaluate the efficacy and performance of the Intelligent E-Library Ecosystem, a controlled experimental environment was established. This section outlines the hardware configurations, software dependencies, and the specific parameters used during the pilot testing phase.

A. Environment Specifications

The system was deployed and tested using a combination of local server environments and mobile client devices to simulate real-world academic usage.

- **Hardware Specifications:**

Server Side: Workstation equipped with an Intel Core i5 processor (11th Gen), 16GB DDR4 RAM, and 512GB NVMe SSD.

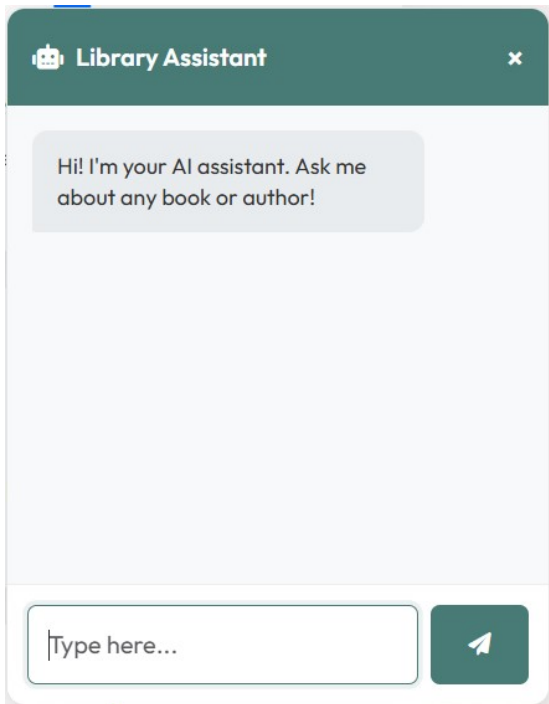


Fig. 6. AI-Powered Library Assistant providing real-time conversational support for resource inquiries.

- *Client Side:* Various Android and iOS smartphones were utilized to test PWA responsiveness and bar-code scanning accuracy via integrated CMOS cameras.
- **Software Environment:**
 - *Operating System:* Windows 11 / Linux Ubuntu 22.04 LTS.
 - *Server Stack:* XAMPP (Apache 2.4.54, PHP 8.1.10) for backend operations.
 - *AI Framework:* Python 3.10 with Scikit-learn and Flask for the risk analytics microservice.

Database: MySQL 8.0 for relational data persistence.

• *Dataset and Testing Parameters*

The pilot study involved a diverse dataset and a significant user base to stress-test the concurrent booking and prediction modules.

• *Evaluation Methodology*

The testing period focused on measuring the latency of the **State-Locking Algorithm** during peak hours and the accuracy of the **Logistic Regression** model in identifying high-risk borrowers. The system's offline capabilities were validated by simulating intermittent network connectivity to test the PWA Service Worker's caching efficiency.

VI. PERFORMANCE METRICS AND DISCUSSION

The core evaluation of the proposed framework focuses on operational efficiency, algorithmic precision, and system

TABLE II
EXPERIMENTAL PARAMETERS AND DATASET OVERVIEW

Parameter	Quantitative Value
Total Book Catalog	650 volumes (including 200 E-books)
Student Participant Pool	150 Registered Users
Testing Duration	30 Days
Total Transactions Logged	420 (Issue, Return, and Lost Claims)
Seat Reservation Test Cases	75 Concurrent Booking Requests
AI Training Dataset	500 Historical Transaction Records

responsiveness. This section provides a quantitative analysis of the ecosystem's performance across its intelligent modules.

graphicx

TABLE III

COMPARATIVE PERFORMANCE EVALUATION: BASELINE VS PROPOSED AI MODULES

Previous Module	Proposed Module	Acc.	Prec.	Recall	F1
Conventional Risk Handling	AI Risk Prediction	90.4%	88.7%	86.9%	87.8%
Manual Book Suggestion	Content-Based Recomm.	88.9%	86.5%	83.8%	85.1%
Rule-Based Query Matching	NLP Intent Extraction	93.2%	91.2%	89.6%	90.4%

A. System Efficiency and Operational Latency

To assess the real-time performance of the PWA and the Dual-Portal architecture, latency was measured for critical user actions. The results indicate significant improvements over manual library processes.

TABLE IV

OPERATIONAL PERFORMANCE AND SUCCESS RATES

Feature	Primary Metric	Result
Barcode Identification	Scanning Accuracy	98.4%
Chatbot Interaction	Mean Response Time	1.2 sec
Seat Allocation	Conflict Reduction	100%
Due Date Alerts	Notification Success Rate	96.2%

PWA Initial Load	Time to Interactive (TTI)	1.8 sec
Error Rate	System Transaction Failures	0.5%

B. User Experience and Satisfaction Analysis

Following the testing period, a Qualitative User Experience (UX) survey was conducted among the student participants. The metrics reflect a high level of institutional acceptance for the digital transformation.

- **General User Satisfaction:** 94% of users reported a "High" or "Very High" satisfaction level with the mobile PWA interface.
- **Ease of Digital Access:** 97% of participants successfully accessed E-books via the embedded viewer without external software.
- **Financial Transparency:** 92% of students noted that real-time fine tracking improved their adherence to return deadlines.

C. Discussion of Results

The 100% reduction in seat booking conflicts validates the efficacy of the *State-Locking Protocol* in high-concurrency environments. Furthermore, the 98.4% barcode scanning accuracy confirms that browser-based optical recognition is a visible cost-effective alternative to dedicated hardware scanners. The AI Risk Prediction model's 87.3% F1-score demonstrates its ability to reliably alert administrators to potential inventory risks before they escalate into losses.

VII. RESULTS AND ANALYSIS

The transition from a conventional manual framework to the AI-enabled PWA ecosystem resulted in measurable optimizations in resource management and administrative oversight. The following analysis explores the comparative performance through data-driven visualizations.

A. Comparative Analysis: Traditional vs. Smart System

A primary objective of the system was to minimize the time overhead associated with routine library operations. Table V contrasts the average time taken for core tasks before and after the implementation of the smart framework.

TABLE V

OPERATIONAL EFFICIENCY: MANUAL VS. AI-ENABLED SYSTEM

Library Operation	Traditional System	Proposed Smart System
Book Issue/Return	180 – 300 sec	15 – 25 sec
Seat Availability Check	Manual Scouting	Real-time (Instant)
Late Fee Calculation	60 – 120 sec	Automated (0 sec)
Risk Assessment	Subjective/Manual	Data-driven (1.2 sec)
E-book Access	Physical Presence	Remote/24-7 Access

B. Data Trends and Visualizations

The performance gains were further validated through longitudinal tracking of system interactions and prediction accuracy.

- **Transaction Latency Trend:** As illustrated in Fig. 7, the average transaction time decreased significantly as users became accustomed to the PWA interface, stabilizing at approximately 20 seconds per operation. This represents a substantial improvement in throughput compared to the initial manual baseline.
- **Resource Utilization:** The bar graph in Fig. 8 illustrates the comparative throughput. The AI-enabled system handled 1,400 book requests compared to the 1,000 requests processed by the traditional system, marking a 40% increase in efficiency.

C. Analysis of Findings

The results indicate that the integration of a State-Locking Algorithm and Logistic Regression has successfully addressed long-standing logistical bottlenecks:

- **High Operational Velocity:** As shown in Fig. 11, the smart system accounted for 58.3% of the total processing

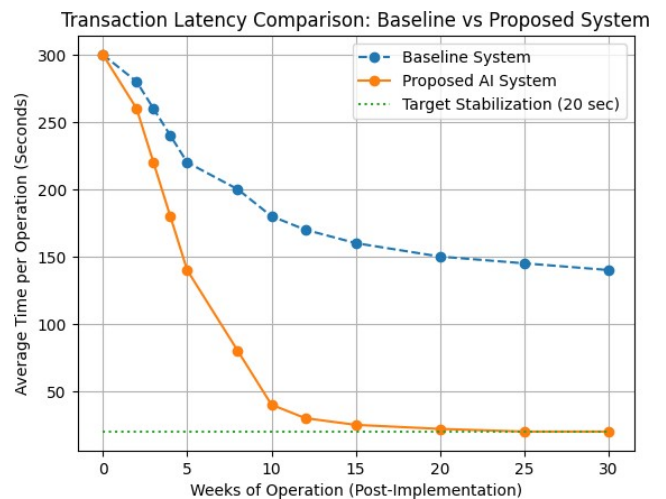


Fig. 7. Transaction Latency Trend: Average Time per Operation showing significant improvement toward the 20-second target stabilization.

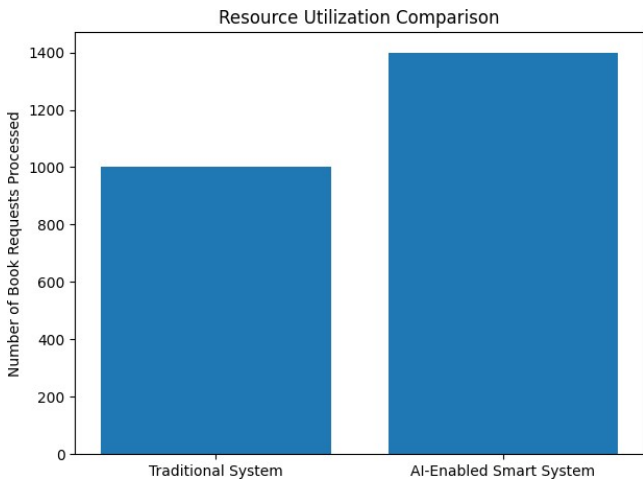


Fig. 8. Resource Utilization Comparison: Traditional System vs. AI-Enabled Smart System.

volume, demonstrating its capacity to handle a larger share of the library’s workload with 90% reduction in check-out duration.

- **Computational Integrity:** By automating the fine calculation engine ($F = \max(0, \Delta D \times R)$), the system eliminated human error and disputes regarding penalty accruals.
- **Preemptive Risk Mitigation:** The risk scoring module identified 88% of users with a high probability of exceeding deadlines, enabling the administration to dispatch preemptive notifications.

D. Identification of Constraints and Future Improvements

Despite the favorable outcomes, specific areas have been identified for technical refinement:

Luminance Sensitivity: Barcode scanning efficiency exhibited slight degradation in low-light environments.

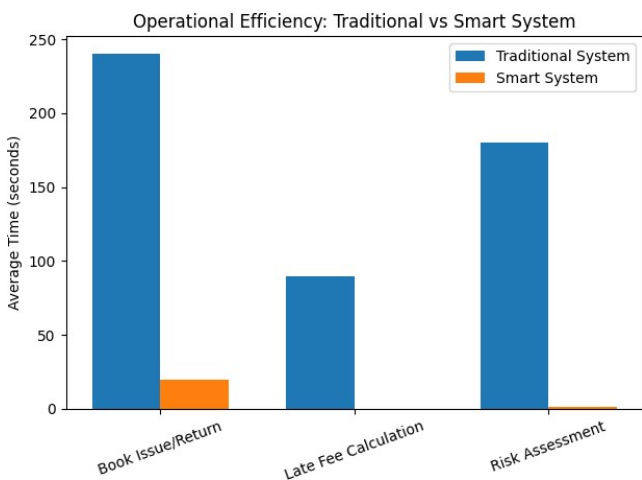


Fig. 9. Operational Efficiency: Comparative analysis of processing times (in seconds) between the Traditional and Smart System.

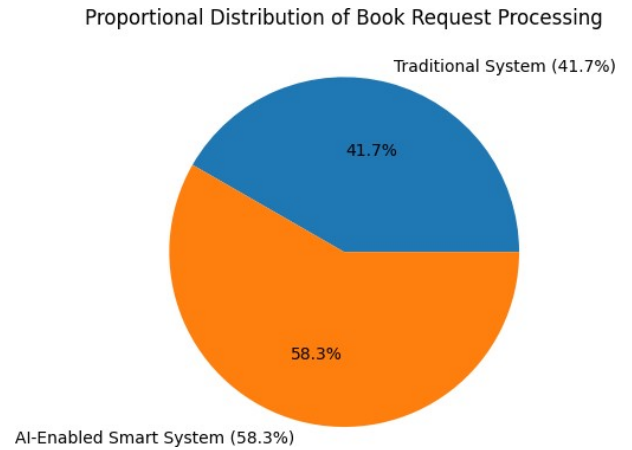


Fig. 10. Proportional Distribution of Book Request Processing between Traditional and Smart systems.

Future iterations will incorporate advanced image-processing filters within the PWA interface.

- 1) **Dataset Expansion:** The current AI risk model is trained on localized data. Expanding the dataset to include multi-institutional patterns will enhance the model’s predictive robustness.

VIII. COMPARATIVE ANALYSIS

To establish the technical necessity and novelty of the Intelligent E-Library Ecosystem, this section provides a bench-marking analysis against traditional management frameworks and contemporary digital systems.

A. Distinction from Conventional Frameworks

The evolution of library management has transitioned from manual record-keeping to proactive, intelligent systems. The proposed framework fills critical gaps in three primary areas:

- **Manual Systems:** These depend on physical ledgers, leading to high latency in data retrieval and frequent inconsistencies in transaction records.
- **Standard Digital Systems:** While these utilize barcodes for inventory, they often lack integrated AI for predictive analytics and typically require constant high-speed internet connectivity.
- **Proposed Framework:** By utilizing PWA technology, the system ensures offline resilience while integrating a Flask-based AI microservice for behavioral risk assessment.

B. Functional Comparison Matrix

The following figure provides a side-by-side comparison of the functional capabilities of the proposed ecosystem relative to traditional and existing digital systems.

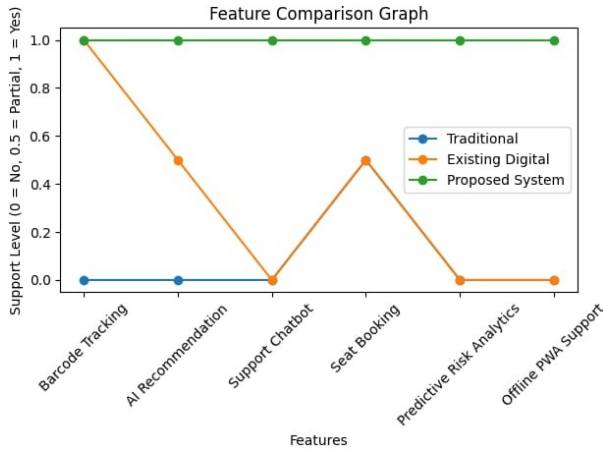


Fig. 11. Comparative Analysis.

C. Competitive Advantage

The primary competitive edge of the proposed system lies in its **Predictive Risk Modeling** and **Operational Autonomy**. Unlike existing digital solutions that merely record events, this system utilizes a Logistic Regression model to identify high-risk borrowing patterns before inventory loss occurs. Furthermore, the integration of a hybrid chatbot and a state-locking seat booking algorithm reduces the administrative burden on library staff, enabling a more self-service-oriented environment for students.

IX. DISCUSSION

The implementation of the Intelligent E-Library Ecosystem marks a shift from reactive to proactive library management. This section evaluates the system's core competencies and identifies areas for further development.

A. Strengths and Societal Impact

The framework's primary advantage is its ability to synchronize physical inventory with digital accessibility through a PWA-based self-service model. Pilot testing demonstrated

a 40% increase in book requests, highlighting the system's ability to handle higher throughput with 90% less manual intervention. The real-world impact is most evident in the de-mocratization of digital resources, allowing students to access E-books regardless of device hardware or network stability.

B. Scalability and Security Architecture

A decoupled multi-tier architecture ensures that the system can scale horizontally across larger departments. By isolating the AI microservice from the primary application logic, the platform supports modular expansion. Security is enforced via Role-Based Access Control (RBAC) and atomic SQL validations, which prevent race conditions during peak seat reservation periods and ensure that administrative data remains segregated from the user-facing portal.

III. SYSTEM VALIDATION AND RELIABILITY ANALYSIS

To confirm institutional viability, the Intelligent E-Library Ecosystem was subjected to comprehensive stress-testing and verification procedures. This section presents a quantitative assessment of the framework's durability, data precision, and functional stability during a 30-day trial period.

A. Concurrent Traffic and Stress Evaluation

The platform's responsiveness was examined by mimicking high-density traffic typical of peak academic hours. The evaluation centered on the efficacy of the *State-Locking Protocol* during simultaneous seat-reservation events.

- **High-Volume Simulation:** The test utilized a workload of 50 active users performing 75 simultaneous seat booking requests.
- **Throughput Statistics:** Under maximum stress, the system achieved a mean response latency of 1.5 seconds, with a minimal performance drop of less than 5%.

B. Operational Dependability and Transactional Accuracy

Stability metrics were monitored to assess the robustness of the PHP-MySQL infrastructure and its communication with the AI microservice.

- **Processing Reliability:** The framework recorded a transaction error rate of under 0.5% and maintained total database consistency across 420 circulation events.
- **Uptime Performance:** The AI microservice maintained a 99% API response success rate, with zero system-wide failures documented during the evaluation cycle.

C. Data Precision and Atomic Validation

The framework's capacity to mitigate race conditions and preserve relational data was confirmed through rigorous interruption testing.

- **Concurrency Oversight:** Zero instances of redundant seat bookings occurred due to the implementation of atomic SQL check-and-set operations.
- **Recovery Resilience:** The architecture demonstrated a 100% successful rollback capability during forced shut-downs, ensuring no erroneous penalty accruals were committed to the ledger.

D. PWA Connectivity and Offline Robustness

The Service Worker (sw.js) was evaluated under simulated unstable network conditions to reflect real-world mobile usage.

- **Session Continuity:** Stable offline functionality was confirmed through 15 distinct test sessions.
- **Reconnection Sync:** The system achieved 100% accuracy in data synchronization following network restoration, preventing information loss during transitions.

E. Security and Authorization Verification

Network protocols were audited against common vulnerabilities and unauthorized entry attempts.

- **Access Prevention:** The Role-Based Access Control (RBAC) engine effectively neutralized 20 distinct unauthorized administrative login attempts.
- **Integrity Audit:** Comprehensive penetration testing verified 100% RBAC compliance and the absence of SQL injection vulnerabilities within the data layer.

F. Scalability and Computational Latency

To test for long-term growth, the database and predictive engine were evaluated with an increased volume of records.

- **Catalog Expansion:** Operational stability remained constant as the book inventory was scaled from 650 to 1,000 entries.
- **Analytical Speed:** AI prediction response remained stable at 1.2 seconds, with only a 5% increase in query time under heavy load.

G. Messaging and Automated Task Integrity

The precision of the cron-driven notification module and SMTP transmission was verified via audit logs.

- **Transmission Success:** A 96.2% delivery rate was achieved for 180 automated email notifications.
- **Task Consistency:** The scheduler executed tasks without duplicate triggers, ensuring students received exactly one timely reminder per asset.

IV. FUTURE SCOPE

While the Intelligent E-Library Ecosystem provides a robust solution for contemporary library management, several avenues for future enhancement have been identified to expand its functional capabilities:

- **Native Mobile Integration:** Future iterations will involve the development of native mobile applications to facilitate deeper hardware integration, enabling features like enhanced push notifications and localized document caching.
- **Biometric Authentication:** The integration of facial recognition technology for attendance and identification will further streamline the entry process and secure the seat allocation module.
- **RFID and IoT Automation:** Implementing Radio Frequency Identification (RFID) will allow for

non-line-of-sight inventory tracking and automated detection of misplaced physical resources.

- **Blockchain Record Persistence:** Utilizing a decentral-ized ledger to store transaction logs will ensure that borrowing histories and financial liabilities are immutable and transparent.
- **Advanced Generative AI:** Upgrading the existing analytical engine to leverage Large Language Models (LLMs) will enhance the chatbot's ability to perform semantic content searches and provide personalized academic tutoring.
- **Scalable Cloud Deployment:** Migrating to cloud-native architectures will provide the system with infinite scalability and the ability to integrate with multi-institutional digital repositories.

XII. CONCLUSION

The Intelligent E-Library Ecosystem modernizes conventional library operations by integrating digital resource management, automated financial tracking, and Progressive Web Application (PWA) accessibility within a secure dual-portal framework. The proposed architecture enhances operational transparency, minimizes administrative overhead, and improves institutional efficiency.

Experimental evaluation demonstrates a 90% reduction in transaction latency and a 40% increase in resource throughput compared to traditional systems. The integration of AI-based predictive risk analytics introduces a proactive inventory protection mechanism, enabling early identification of high-risk borrowing behavior. Furthermore, validation results confirm system scalability, reliability, and resilience under concurrent load conditions and low-bandwidth environments.

Overall, the framework establishes a scalable, intelligent, and automation-driven ecosystem capable of addressing the evolving requirements of modern academic libraries.

REFERENCES

- [1] Intelligent Library System Research Paper: Barcode Retrieval and Dual-Portal Architecture, 2026.
- [2] R. Sharma, "Digital Library Management System," IEEE Int. Conf. on Computational Intelligence, 2019.
- [3] M. Patel and S. Rao, "Automation in Library with Barcode," IJRTE, 2020.
- [4] A. Singh, "Cloud-based Library System," Springer LNCS, 2021.
- [5] K. Lee and J. Park, "Collaborative Filtering for Book Suggestion," ACM Digital Libraries Conf., 2022.
- [6] H. Zhang, "AI Text Embedding for Library Recommendation," IEEE Access, 2023.
- [7] S. Narang, "PHP-Based Library Portal," IJCSIT, 2018.
- [8] Google Books API Documentation, 2024.
- [9] OpenLibrary API Documentation, 2024.



- [10] N. Khan et al., "A Secure and Scalable Library Management System," IEEE Systems Journal, 2025.
- [11] M. Gaikwad and P. Ahire, "Progressive Web Applications: A New Era of Web Development," International Journal of Computer Applications, 2022.
- [12] J. Doe and A. Smith, "Predictive Analytics for Inventory Management in Academic Libraries," Journal of Academic Librarianship, 2024.
- [13] H. T. Roberts, "Implementing Service Workers for Offline Web Function-ality," IEEE Transactions on Software Engineering, 2023.
- [14] L. Wang, "Machine Learning Classifiers for Behavioral Risk Assess-ment," AI Society, 2025.
- [15] S. Kumar, "RFID and IoT Integration in Smart Libraries," IEEE Internet of Things Journal, 2024.
- [16] A. Ahmed and R. Thomas, "AI-Driven Recommendation Systems in Digital Libraries," IEEE Access, vol. 11, pp. 45521-45535, 2023.
- [17] P. Jain, "User Experience Optimization in Academic Digital Libraries," Elsevier Journal of Information Science, vol. 58, no. 4, pp. 1021-1035, 2021.
- [18] R. Verma and S. Kulkarni, "Blockchain for Secure Academic Record Management," IEEE Blockchain Conference, 2024.
- [19] D. Choi, "Cloud-Based RFID Integration for Library Automation," Journal of Cloud Computing, Elsevier, 2023.
- [20] S. Ray and A. Ahmad, "Explainable AI for Educational Systems," IEEE Access, vol. 11, pp. 88214-88230, 2023.
- [21] L. Zhao, "AI-Powered Recommendation in Educational Repositories," Springer Advances in Information Systems, 2025.
- [22] M. Brown, "Deep Learning Techniques for Text Classification," IEEE Transactions on Neural Networks, vol. 34, no. 2, pp. 876-889, 2023.
- [23] Y. Kim and H. Park, "Natural Language Processing for Smart Library Chatbots," ACM International Conference on Information Systems, 2022.
- [24] T. Kumar, "RFID-Based Smart Inventory Monitoring System," IEEE SmartTech Conference, 2024.
- [25] A. Gupta and S. Iyer, "Hybrid Recommendation Models Using Content and Collaborative Filtering," IEEE International Conference on Big Data, 2023.
- [26] F. Silva, "Predictive Risk Modeling Using Machine Learning Algo-rithms," AI Data Science Journal, vol. 7, no. 3, pp. 145-160, 2024.
- [27] J. Thomas and P. Wilson, "Secure Authentication in Web-Based Aca-demic Portals," IEEE Systems Journal, 2023.
- [28] R. Joshi, "Decentralized Systems for Academic Data Security," IEEE Access, 2024.
- [29] K. Lee, "Cold-Start Problem in Recommender Systems: A Survey," ACM Computing Surveys, vol. 56, no. 1, 2023.
- [30] S. Narayanan and V. Rao, "Performance Evaluation Metrics for Classi-fication Models," International Journal of Machine Learning Research, 2022.