Team Name: AAA Battery

Library Management System

Introduction

The Library Management System (LMS) is designed to streamline library operations, making it easier for librarians and users to manage books, track borrowings, and ensure seamless book transactions.

Project Vision

Our vision is to develop a highly efficient, user-friendly, and fully automated library management system that streamlines operations, reduces manual effort, minimizes errors, and significantly enhances the overall experience for both librarians and readers. By leveraging technology, we aim to create a seamless and intuitive system that ensures easy access to information, improves book tracking, and fosters a more engaging reading environment.

Intended Use of the System

The Library Management System (LMS) will be used by:

- **Librarians:** Manage book inventory, track borrowed and returned books, handle overdue records, and streamline library operations.
- **Students/Users:** Search for books, check availability, borrow, renew, and return books efficiently.
- **Administrators:** Oversee system operations, manage user access, generate reports, and ensure smooth library management.

Identify Stakeholders

- **Librarians:** Require an easy way to catalogue books, manage borrowing records, and track due dates.

- **Students/Users:** Need a system to browse available books, borrow, and return books on time.
- **Administrators:** Need to oversee system performance, maintain records, and ensure data integrity.

Identify Needs

- **Digital Catalogue:** A centralized system to track book availability, categories, and details.
- **Automated Borrowing & Return System:** Streamlines check-ins and check-outs to reduce manual work.
- **Fine Management:** Automatically calculates and tracks overdue fines for late returns.
- **User-Friendly Search:** Efficient search functionality to help users quickly find books by title, author, or genre.
- **User Account Management:** Secure user registration, email verification, login, and password recovery.

Features/Overall Functionality

What Will the System Do?

- Allow users to search for books by title, author, or category.
- Enable librarians to add, edit, and remove book records.
- Manage book borrow and return transactions.
- Notify users about due dates and late fees.
- Generate reports for book availability and usage statistics.
- Ensure secure user authentication and account management.

How Will the System Help Its Users?

- Librarians can efficiently manage books and records.
- Users can easily find and borrow books without manual intervention.
- Administrators can monitor and maintain system performance.

User Stories

- As a librarian, I want to add and remove books so that the catalogue remains updated.
- As a student, I want to search for books so that I can find what I need quickly.
- As a librarian, I want to track borrowed books so that I can manage overdue records.
- As a user, I want to receive due date reminders so that I return books on time.
- As a user, I want to register an account so that I can access the library system.
- As a user, I want to verify my email so that I can confirm my identity.
- As a user, I want to log in securely so that my borrowing history and personal data remain protected.
- As a user, I want to recover my password so that I can regain access to my account if I forget it.

Structured Specifications

Book Search

- **Input:** Search query (title, author, category)
- **Process:** Fetch matching records from the database.
- **Output:** Display a list of matching books.

Book Borrowing

- **Input:** User ID, Book ID
- **Process:** Validate user, check availability, update records.
- **Output:** Confirmation of borrowing or error message.

User Registration

- **Input:** Name, Email, Password
- **Process:** Validate inputs, encrypt password, store user details, send verification email.
- **Output:** Confirmation message or error message.

Email Verification

- **Input:** Verification token from email
- **Process:** Validate token, update account status, invalidate token after use.
- **Output:** Success message or error message.

User Login

- **Input:** Email, Password
- **Process:** Authenticate credentials, check email verification status, generate secure sessions.
- **Output:** Redirect to dashboard or error message.

Password Recovery

- **Input:** Registered email address, new password
- **Process:** Validate email, generate reset token, validate token, update password.
- **Output:** Success message or error message.

Scrum Board

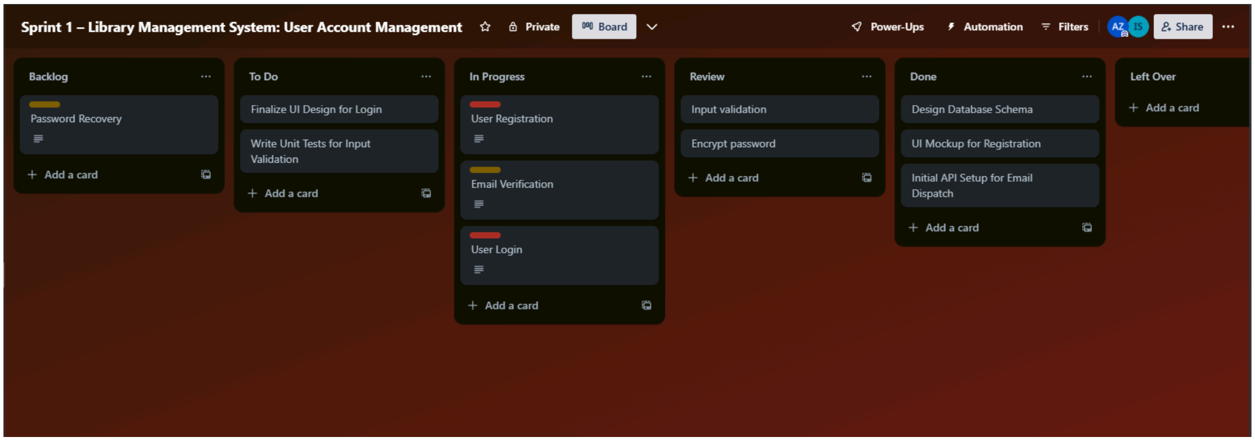
The Scrum Board provides a visual, real-time snapshot of all tasks and their current status throughout the sprint. In our **Library Management System – User Account Management** project, the board is divided into the following columns:

- **Backlog:** Contains tasks or user stories that have been identified but not yet prioritized for the current sprint.
- **To Do:** Lists the tasks that have been selected for the sprint and are ready to be worked on.
- **In Progress:** Shows the tasks actively being worked on by team members.
- **Review:** Holds tasks that are completed but awaiting review, testing, or approval before they can be finalized.
- **Done:** Displays tasks that have been successfully completed and approved.
- **Left Over:** Contains tasks that were not finished within the sprint and may be moved to the next sprint.

Each card on the board represents a specific task or user story. Team members move cards from left to right as work progresses. This structure helps the team:

1. **Track Progress** – Everyone can quickly see which tasks are in progress, which are awaiting review, and which are finished.
2. **Foster Collaboration** – By visually highlighting task ownership and status, team members know who is working on what and can offer help where needed.
3. **Maintain Transparency** – Stakeholders can easily view the project's current state without needing extensive status reports.
4. **Facilitate Daily Standups** – During daily Scrum meetings, the team discusses any blockers and updates the board, ensuring tasks move smoothly through each column.
5. **Manage Sprint Scope** – Tasks in "Left Over" provide insight into what was not completed and need re-evaluation for the next sprint's backlog.

This board is typically updated continuously throughout the sprint. At the end of each sprint, it reflects which tasks were completed successfully and which require additional work or re-planning in subsequent sprints.



Sprint 1 – Library Management System: User Account Management

Backlog	To Do	In Progress	Review	Done	Left Over
Password Recovery	Finalize UI Design for Login	User Registration	Input validation	Design Database Schema	
	Write Unit Tests for Input Validation	Email Verification	Encrypt password	UI Mockup for Registration	
		User Login		Initial API Setup for Email Dispatch	

NFR Specification Using Natural Language

- The system shall provide book search results within 2 seconds.
- The system shall support at least 100 concurrent users.
- The system shall have a user-friendly interface accessible via desktop and mobile.
- The system shall securely store passwords using encryption.
- The system shall ensure email verification before granting full access.

Work Division

Skill-Based Assignment

Front-End Development tasks, such as UI design or form creation, are handled by team members proficient in front-end technologies.

Back-End Development tasks, like database schema design or API setup, are allocated to developers with experience in server-side programming and database management.

QA & Testing tasks (including writing unit tests, integration tests, and performing manual checks) are assigned to testers or developers specializing in quality assurance.

Collaboration & Pairing

Complex tasks or critical functionalities may involve **pair programming** or **collaboration** between two or more developers to ensure high-quality output and knowledge sharing.

Ownership & Accountability

Each task has a clear owner who is responsible for driving it to completion. However, the team regularly communicates during stand-ups or ad-hoc discussions to assist or review each other’s work.

Flexible Reassignment

If a team member completes their tasks early or encounters blockers, tasks can be **reassigned** or **swapped** to balance workload and keep the sprint on track.

Tracking & Updates

Progress on each task is **tracked on the Scrum Board**. Team members move cards across columns (e.g., from “To Do” to “In Progress”) and add notes or comments as needed.

By dividing tasks according to each member’s skill set and actively collaborating, the team ensures efficient development, reduces bottlenecks, and maintains a transparent view of everyone’s contributions