Library Management System Development Summary

This documentation outlines the Library Management System (LMS), designed to enhance library operations by providing automated book and patron management, inventory tracking, and an interactive user interface.

System Overview:

- **User Interface**: Allows users to search, borrow, and return books.
- **Backend Logic**: Manages core library operations.
- **Data Storage**: Maintains information on books, patrons, and transactions.

Architecture:

- Built using a layered Model-View-Controller (MVC) design to separate presentation, business logic, and data handling.

Features:

1. **Book Management**: Search by title, author, or ISBN.
2. **Borrowing & Returning**: Tracks and manages book loans.
3. **Patron Management**: Adds or removes patrons and handles overdue fines.

Implementation:

- Developed in **Java**, with core classes (Library, Book, Patron, Author) encapsulating data and operations.
- **Status Enum**: Monitors book availability, including "available," "checked out," and "overdue."
- **Borrowable Interface**: Standardizes borrowing and returning functionalities.

Testing:

1. **Unit Testing**: Ensures component reliability.
2. **Integration Testing**: Verifies smooth module interaction.
3. **User Acceptance Testing**: Validates usability and effectiveness with real users.

Future Enhancements:

- Plans for database integration, user authentication, and expanded functionality with third-party APIs.

Conclusion: The LMS is a flexible, user-centric solution for libraries, designed with scalability and efficiency to meet ongoing operational needs.