LIBRARY MANAGEMENT SYSTEM

- **Problem Statement :-** In traditional libraries, managing book records, tracking borrowed/returned books, and handling user registrations is time consuming and prone to errors. Manual processes often lead to book losses, mis-management, & lack of proper tracking. The problem is to develop a computerized System that automates these tasks, provides real-time book availability & ensures efficient library operations.

① **Purpose :-** The purpose of this document is to define the requirements for the Library Management System (LMS), which automates the process of managing books, users, borrowing, and return in a library. It will replace the existing manual process, reduce errors, and improve efficiency.

② **Scope :-** LMS aims to automate the day to-day activities of a lib.

- maintaining a centralized digital database of all books, user & transactions.
- enabling students / members to register, log in search, borrow, renew & return books.
- allowing librarians / admins to manage book records, user accounts, & generate reports.
- providing real-time availability status of books
- generating various reports
    - list of issued books
    - overdue books with fines.
    - inventory of available books

- ensuring secure access through authentication and data encryption.

③ **Overview :-** The LMS will be web-based application accessible to Admins and Users.
- Admins can add, update or remove books, manage user accounts and monitor borrowed / returned books.
- Users / Students can search books, check availability, borrow books & view due dates.

④ **General Description :-** • [User] - Admin / Librarian
  - manages the entire system eg. adds / removes books.
  - Student / Users - search & borrow books, renew borrowed books, return books.
- [System] - The system will maintain a centralized database of books & users.
  - It will allow search by title, author or category.
  - it will automatically update book status when borrowed / returned.

⑤ **Functional Requirement :-** user login/ registration
  - admin can add, update, delete books.
  - Users can search for books by title, author or category.
  - Borrow & return books
  - Automatic due date & fine calculation
  - Generate reports.

⑥ <u>Performance Requirements</u> :- The system should handle at least 500 user concurrently.
- Search results should load within 2 seconds
- System uptime should be 99%
- Must scale to support 10,000+ books

⑦ <u>Interface Requirement</u> :- • <u>User - Interface</u> - web & mobile - friendly interface.
- easy navigation for search, issue, & return functions.
• <u>Hardware - Interface</u> - Standard desktop/laptop / mobile
- minimum 4GB·RAM, 2 GHz processor.
• <u>Software - Interface</u> - Uses HTTPS protocol for secure data - transfer
- Support email / SMS notification
- Database - My SQL.
- Browser support - Chrome, Firefox, Edge.

⑧ <u>Non-Functional Requirement</u> :- The system must be available 99.5% of the time.
- The system should support at least 10000 book records & 5,000 user accounts efficiently.
- The system interface must be intuitive & user - friendly for both librarian & students.
- Only authenticated users should be able to access the system.

⑨ <u>Design - Constraints</u> :- Must comply with data protection & privacy laws.
- Must use role - based authentication.

— Should be developed using open-source technologies to reduce cost.

10) <u>Schedule & Budget</u> :→ Total duration will be 16 weeks ~ 4 months
→ Total budget will be 8.6 lakhs.

11) <u>Risk analysis</u> :→ Data loss risk
→ Security risk.
→ Delay in schedule.
→ user adoption resistance.