Experiment – 5

**Software Requirement Specifications (SRS): -**

**LIBRARY MANAGEMENT SYSTEM**

1. **Introduction**
   1. **Purpose Statement:**

The library management system (LMS) is designed so as to automate various library operations, including book management, fine management, book issuance, return procedure and member management all in a single software application. LMS aims to enhance the overall efficiency by minimize efforts and human errors and provide an interactive interface between the users and management team.

* 1. **Scope**:

LMS will allow the librarian to easily and efficiently manage book data, to track user checkouts and returns, check for due dates and deadlinesand handle fines. Users will be able to search for books, check availability of books , and reserve books. The system will include authentication mechanisms, reports management , and a user-friendly interface.

* 1. **Abbreviations to be use:**
* ISBN : International Standard Book Number.
* GUI : Graphical User Interface.
* LMS : Library management System.
  1. **References:**
* IEEE standards for Software Requirements Specifications.
* Library management best practices.
  1. **Overview:**

This documentation overviews the Functional and Non-Functional requirements,

System constraints and operational requirements for the LMS in general.

1. **General Description of LMS.**
   1. **Functions**

* Book management and categorisation
* User registration and authentication for better security
* Book check ,search and availability check
* Book issue and return of book
* Fine calculation for late returns by the user
* Report generation for the user

**2.2 User usecases**

* Librarian: Manages books and members, issues books, fines handling
* Library Member: Search for books, borrows and returns books, checks due dates
* Administrator: Manages system configurations and permissions of both librarian and member

**2.3 Some constraints**

* The system should be accessible by web and PC both.
* Comes with data protection.
* Limited access for specific roles like member.

1. **Specific (functional /non-functional ) Requirements**
   1. **Functional Requirements:-**
2. User Management
   * Register new users
   * Authenticate users by login details
   * Assign roles (Librarian, Member, Admin)
3. Book Management
   * Add, update, and delete books
   * Differentiate books based on genres and authors
   * Maintain and record book availability status
4. Search
   * Allow users to search by title, writer , or ISBN
5. Book Issuance and Return
   * Issue books to members
   * Track return dates
   * Calculate fines for late returns
6. Reports
   * Generate reports on books issued, returned, and overdue
   1. **Non-Functional Requirements**

* Security: User authentication and role-based access
* Performance: Fast response time for queries and transactions
* Usability: user-friendly interface
* Scalability: Support multiple users

**3.3 System Features**

* Dashboard for library statistics
* Email notifications for reminders
* Multi-language support

4. **Appendices**

* Sample database scheme
* User interface mock-ups
* API documentation