Here’s a basic **Software Requirements Specification (SRS)** document for a **Library Management System**. This includes the essential sections typically required in an SRS document.

**Software Requirements Specification (SRS) Document**  
**Project Title**: Library Management System  
**Version**: 1.0  
**Date**: April 14, 2025

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to outline the software requirements for the Library Management System (LMS). It defines the features, functionalities, and constraints of the system, which will help stakeholders, developers, and testers in understanding and implementing the system.

**1.2 Scope**

The LMS is designed to manage the day-to-day operations of a library. It will handle book inventory, member management, book issuance and return, overdue tracking, and reporting. The system will be used by librarians, library members, and administrators.

**1.3 Definitions, Acronyms, and Abbreviations**

* **LMS** – Library Management System
* **UI** – User Interface
* **DB** – Database
* **ISBN** – International Standard Book Number

**1.4 References**

* IEEE 830-1998 SRS Standard
* Database design principles

**1.5 Overview**

This document is organized into sections that describe the system in detail, including its functionalities, performance requirements, design constraints, and other attributes.

**2. Overall Description**

**2.1 Product Perspective**

The LMS is a standalone system that may later be extended to integrate with school or university portals.

**2.2 Product Functions**

* Book inventory management
* User registration and management
* Book issue/return functionality
* Overdue management
* Search and filter books
* Admin dashboard and reporting

**2.3 User Classes and Characteristics**

* **Admin**: Full access to all features and user management.
* **Librarian**: Manages books and handles book issue/return.
* **Member**: Searches for books, views issued books, and requests returns.

**2.4 Operating Environment**

* Web-based application (or desktop-based if needed)
* Operating System: Windows/Linux/macOS
* Database: MySQL/PostgreSQL
* Frontend: HTML/CSS/JS or framework (React/Angular)
* Backend: Java/Python/Node.js

**2.5 Design and Implementation Constraints**

* Must support role-based access control
* Use open-source technologies where possible
* Responsive UI design

**2.6 User Documentation**

* User manual
* Admin guide
* Installation guide

**3. Specific Requirements**

**3.1 Functional Requirements**

* **FR1**: The system shall allow the admin to add, update, or delete books.
* **FR2**: The system shall allow users to register and login.
* **FR3**: The librarian shall issue and return books to/from members.
* **FR4**: The system shall display overdue alerts and fines.
* **FR5**: The system shall allow searching and filtering of books.
* **FR6**: The system shall generate reports (daily, weekly, monthly).

**3.2 Non-Functional Requirements**

* **NFR1**: The system should support up to 100 concurrent users.
* **NFR2**: The average system response time should be under 2 seconds.
* **NFR3**: The system should ensure data security and privacy.
* **NFR4**: The UI should be user-friendly and accessible.

**3.3 Interface Requirements**

* **User Interface**: Web or desktop GUI
* **Hardware Interface**: Standard PC or laptop
* **Software Interface**: MySQL database, web server
* **Communications Interface**: Local network or internet access

**4. Appendices**

* Sample UI mockups
* Entity Relationship Diagram (ERD)
* Database schema
* Glossary of terms

Would you like this as a downloadable Word or PDF file? Or want it customized for a web-based/mobile app version?