# Requirements for a Library Information System

## 1. Functional Requirements

Functional requirements describe the core capabilities and functionalities the library information system (LIS) must provide.

### a. User Management

- User Registration: Allows library members (students, faculty, staff, etc.) to register and create accounts.  
- Login and Authentication: Secure login for different types of users (members, librarians, administrators).  
- User Profiles: Maintain personal information, borrowing history, and fines for each user.  
- Role-Based Access: Different access levels for members, librarians, and administrators.

### b. Catalog Management

- Book Inventory: Add, update, and delete books in the library catalog.  
- Search Functionality: Allow users to search for books by title, author, ISBN, genre, publication date, etc.  
- Filtering and Sorting: Enable users to filter and sort search results based on criteria such as availability or popularity.

### c. Borrowing and Returns

- Borrowing Books: Enable users to borrow books if they are available and within borrowing limits.  
- Return Process: Allow users to return books and automatically update the system for availability.  
- Renewals: Enable users to extend the borrowing period for books, subject to rules and availability.  
- Reservation or Hold Requests: Allow users to place a hold on books currently borrowed by others.

### d. Notifications and Alerts

- Due Date Reminders: Notify users about upcoming due dates via email or SMS.  
- Overdue Notices: Send alerts for overdue books, along with details of any associated fines.  
- Reservation Notifications: Notify users when reserved books become available.

### e. Fine and Payment Management

- Fine Calculation: Calculate fines for overdue books based on predefined rules.  
- Payment Processing: Allow users to pay fines online or at the library.  
- Fine History: Maintain a record of all fines, payments, and outstanding balances for each user.

### f. Reports and Analytics

- Inventory Reports: Provide statistics on available, borrowed, and lost books.  
- User Activity Reports: Track borrowing and return patterns, popular books, and fine data.  
- System Logs: Record login attempts, book transactions, and modifications for security auditing.

### g. Administration and Maintenance

- System Configuration: Allow administrators to set borrowing limits, fine rates, and user access permissions.  
- Data Backup and Recovery: Ensure regular backups of the library database and enable recovery in case of data loss.  
- Maintenance and Updates: Allow for system maintenance, including updates to the catalog and user data.

## 2. Non-Functional Requirements

Non-functional requirements focus on the overall performance, usability, and reliability of the LIS.

### a. Performance

- Response Time: Ensure fast response times for searches and transactions, especially during peak usage.  
- Scalability: Support the library’s growing user base and inventory without performance degradation.

### b. Usability

- User-Friendly Interface: Provide an intuitive and accessible interface for all user roles.  
- Accessibility: Comply with accessibility standards to support users with disabilities.

### c. Security

- Data Privacy: Protect user and catalog data with encryption and secure access controls.  
- Access Control: Ensure role-based access to prevent unauthorized actions.  
- Regular Security Updates: Keep the system secure by regularly updating it to fix vulnerabilities.

### d. Reliability and Availability

- High Availability: Ensure minimal downtime, providing reliable access to the system.  
- Fault Tolerance: Implement error handling and redundancy to minimize disruptions.

### e. Compatibility

- Multi-Device Support: Make the system accessible on desktops, tablets, and mobile devices.  
- Integration: Support integration with third-party services such as external catalogs, payment gateways, and school/university databases.

### f. Maintainability

- Modular Architecture: Design the system for easy updates, enhancements, and troubleshooting.  
- Documentation: Provide clear documentation for developers and administrators to support maintenance and future development.

## 3. Technical Requirements

Technical requirements define the technology stack and technical capabilities needed for the LIS.  
- Database: A robust database system (such as MySQL, PostgreSQL) to store user, catalog, and transaction data.  
- Server Infrastructure: A reliable server setup to handle requests, host the database, and ensure system uptime.  
- Backup System: Regular, automated backups with recovery options.  
- Integration APIs: APIs for integration with third-party systems or mobile applications.  
- Security Protocols: Secure protocols such as HTTPS for data transmission, encryption for sensitive data, and multi-factor authentication for enhanced security.