**System Requirements Document for Library Management System**

**Introduction**

The Library Management System (LMS) is designed to manage the operations of a library, including the cataloging of books, categories, reservations, borrowing and authors. The system will support the recording of book details, user information, and borrowing of books. The system will be built using JavaFX for the user interface and MySQL for database management, with JDBC for communication between the application and the database.

**Functional Requirements**

**1. User Management**

* **Users must be able to sign up for an account and log in to the system.**
* **The system must support different user roles (e.g., patron, librarian).**
* **Users can update their profiles (e.g., change password, update personal information).**

**2. Book Management**

* **The system should allow the librarian to add, update, and delete books from the catalog.**
* **Books should be categorized by genre (e.g., fiction, non-fiction, science).**
* **Each book should have a title, author, category, total copies, available copies, and location in the library.**
* **Form validation for all inputs**

**3. Author and Category Management**

* **Librarians can add, update, and delete authors and categories.**
* **Authors are associated with books, and categories are used to organize books.**
* **Form validation for all inputs**

**4. Borrowing and Return of Books**

* **Patrons can borrow books, specifying the borrowing date and due date.**
* **Librarians can mark books as returned and update the status of borrowings.**
* **The system must track borrowed books and ensure that a book cannot be borrowed if there are no available copies.**
* **Form validation for all inputs**

**5. Reservation System**

* **Patrons can reserve books that are currently unavailable.**

**6. Search Functionality**

* **The system must allow users to search for books by name, author, or category.**
* **The system must allow users to search for categories and authors.**
* **Search results should be displayed in a table or list format.**

**7. Notifications**

* **The system should notify users of errors (e.g., failed borrowing due to no available copies) and successes (e.g., book successfully borrowed).**
* **Notifications should appear as toast messages or alerts.**

**Non-Functional Requirements**

* **Performance**:
  + The system should support concurrent access by multiple users with minimal performance degradation.
  + Response times for book searches, transaction history, and report generation should be under 3 seconds.
* **Security**:
  + The system should implement role-based access control to ensure users only have access to their authorized features.
  + Passwords should be encrypted, and all sensitive data should be protected.
* **Scalability**:
  + The system should be scalable to handle an increasing number of books, users, and transactions.
* **User Interface**:
  + The UI should be user-friendly and responsive, built using JavaFX with modern design elements for ease of use.