**Library Inventory App - Frontend Testing Requirements**

These requirements are structured to be clear, concise, and provide sufficient detail for an LLM to understand and generate relevant responses (like Jira stories, diagrams, or code snippets), while also serving as a basis for frontend testing.

**Overall Goal:** Develop a user-friendly Library Inventory Management System that allows librarians to efficiently manage their book collection, including adding, viewing, updating, and deleting book records.

**Core Functionality Requirements (for LLM to generate)**

**1. Book Listing & Search:** \* **REQ-001: Display All Books:** The system shall display a paginated list of all books in the inventory. Each book entry should show its title, author, ISBN, publication year, and current status (e.g., "Available", "On Loan"). \* **REQ-002: Search Books by Keyword:** Users shall be able to search for books by entering keywords in a search bar. The search should be performed across title, author, and ISBN fields. \* **REQ-003: Filter Books by Status:** Users shall be able to filter the displayed book list by their availability status (e.g., "Available", "On Loan").

**2. Add New Book:** \* **REQ-004: Add Book Form:** The system shall provide a form for adding new book records. The form should include fields for: \* Title (text input, mandatory) \* Author (text input, mandatory) \* ISBN (text input, unique, mandatory) \* Publication Year (numeric input, optional) \* Genre (dropdown/select, multiple options, optional) \* Number of Copies (numeric input, mandatory, default to 1) \* **REQ-005: ISBN Uniqueness Validation:** The system shall prevent the addition of a new book if the ISBN already exists in the inventory. A clear error message should be displayed to the user. \* **REQ-006: Successful Add Confirmation:** Upon successful addition of a new book, a confirmation message shall be displayed to the user, and the new book should appear in the book list.

**3. View Book Details:** \* **REQ-007: View Individual Book:** Users shall be able to click on a book in the list to view its detailed information on a separate page/modal. This detail view should include all fields captured during book addition, plus a unique Book ID generated by the system.

**4. Update Book Information:** \* **REQ-008: Edit Book Form:** Users shall be able to modify the details of an existing book. An "Edit" button should be available on the book detail view, leading to an editable form pre-populated with current book data. \* **REQ-009: Save Changes:** Upon saving modifications, the system shall update the book's information in the inventory and display a success message. \* **REQ-010: Cancel Edit:** Users shall be able to cancel the edit operation without saving changes.

**5. Delete Book:** \* **REQ-011: Delete Book Confirmation:** Users shall be able to delete a book from the inventory. A confirmation dialog should appear before permanent deletion to prevent accidental removal. \* **REQ-012: Successful Deletion:** Upon successful deletion, the book shall be removed from the list, and a confirmation message shall be displayed.

**Frontend Specific Testing Requirements (for LLM to consider during generation)**

**6. User Interface & Experience:** \* **REQ-013: Responsive Design:** The application shall be responsive and usable across various screen sizes (desktop, tablet, mobile). \* **REQ-014: Clear Navigation:** The navigation between different sections (e.g., "View Books," "Add Book") shall be intuitive and easily accessible. \* **REQ-015: Loading Indicators:** Appropriate loading indicators shall be displayed when data is being fetched or processed (e.g., during search or form submission). \* **REQ-016: Form Validation Feedback:** All form fields shall provide clear and immediate validation feedback (e.g., "Title is required," "Invalid ISBN format").

**7. Error Handling & Messages:** \* **REQ-017: User-Friendly Error Messages:** When an error occurs (e.g., network issue, backend error), the system shall display user-friendly error messages rather than technical jargon. \* **REQ-018: Empty State Handling:** The book list view shall display a clear message when there are no books in the inventory.

**8. Accessibility (for LLM to consider adding attributes):** \* **REQ-019: Keyboard Navigation:** All interactive elements shall be navigable using only the keyboard. \* **REQ-020: ARIA Attributes:** Appropriate ARIA attributes should be used to enhance accessibility for screen readers (e.g., aria-label, aria-describedby).