**HIGH LEVEL TECHNICAL REQUIREMENT DOCUMENT**

**BOOKS MANAGEMENT SYSTEM – BORROW BOOKS PAGE**

**1. Overview**

This document outlines the requirements for creating a user-friendly Borrow Books Page within the dashboard of the Books Management System. The page will allow administrators and users to borrow books securely and efficiently.

**2. Objective**

* Develop a simple and user-friendly page for borrowing books.
* Implement validation to ensure proper borrowing records.
* Integrate the frontend with the API to manage book borrowing.

**3. Success Criteria**

* Users can access the Borrow Books Page from the dashboard.
* Users can search and select books to borrow.
* Users can specify the borrowing period.
* A success notification appears upon successful borrowing.
* Appropriate error notifications appear for invalid input or failed borrowing attempts.

**4. Components (Form)**

* **User ID:**
  + **Type:** Text
  + **Validation:** Required
  + **Placeholder:** Enter User ID…
* **Book Selection:**
  + **Type:** Dropdown or Searchable List
  + **Validation:** Required
  + **Options:** List of available books fetched from the API
* **Borrow Date:**
  + **Type:** Date
  + **Validation:** Required
  + **Placeholder:** Select Borrow Date…
* **Borrow Button:**
  + **Type:** Submit
  + **Label:** Borrow
* **Cancel Button:**
  + **Type:** Button
  + **Label:** Cancel
  + **Functionality:** Redirects back to the book list without borrowing the book.
* **Error Handling:**
  + Display an alert or error notification if the borrowing fails.

**5. Page Workflow**

1. **Accessing Borrow Books Page:**
   * Navigate to the Borrow Books Page from the dashboard.
   * The page displays a form for inputting borrowing details.
2. **Selecting Book to Borrow:**
   * User selects the book from a dropdown or searchable list of available books.
3. **Inputting Borrowing Details:**
   * User fills out the User ID, Borrow Date, and Return Date fields.
   * Validation checks ensure all required fields are correctly filled.
4. **Confirming Borrowing:**
   * User clicks the 'Borrow' button.
   * The system sends an API request to create a borrowing record.
   * On success, a notification appears, and the user is redirected to the book list page.
   * On failure, an error notification appears, indicating the issue.

**6. API Integration**

* **Fetch Available Books:**
  + **Endpoint:** GET /api/books/available
  + **Description:** Fetches a list of books that are currently available for borrowing.
* **Create Borrowing Record:**
  + **Endpoint:** POST /api/borrow
  + **Description:** Creates a new borrowing record with the provided details.