

Technical Specification for Library System in Next.js

1. Overview

The project is to build a simple library system using Next.js that handles CRUD operations (Create, Read, Update, Delete) for books in a database. The system will use Prisma for database interactions and PostgreSQL as the database. The project will include both server and client components as well as validation using Zod.

2. Technical Stack

- **Framework:** Next.js
- **Database:** PostgreSQL
- **ORM:** Prisma
- **Validation:** Zod
- **Language:** TypeScript

3. Functionality

- **CRUD for Books**
 - **Create** a new book
 - **Read** the list of books
 - **Update** a book
 - **Delete** a book

4. Components and Architecture

4.1 Database Model

- **Book:**
 - **id:** Unique identifier for each book.
 - **title:** Title of the book.
 - **author:** Author of the book.
 - **published:** Publication date.
 - **isbn:** ISBN number.

4.2 Server Actions

Use server actions to handle database operations. The following functions should be implemented:

- **createBook:** Create a new book.
- **updateBook:** Update the information for a book.
- **deleteBook:** Delete a book from the database.

4.3 Pages and Components

- **Books List (BooksList):**
 - Displays all books in the database.
 - Use a server component to render a list of all books.
- **Add/Edit Book (BookForm):**
 - Form to create or update a book.
 - Uses createBook and updateBook to send data to the server.
- **Delete Book (DeleteBookButton):**
 - A button to delete a book.
 - Uses deleteBook to remove a book from the database.

5. Student Assignments

1. **Database Setup:**
 - Create a Prisma schema file for the book model.
 - Migrate the database to create the table.
2. **Server Actions:**
 - Implement server actions for CRUD operations.
3. **Front-End Development:**
 - Create components and pages to display, create, edit, and delete books.
4. **Testing:**
 - Test that all CRUD operations work correctly.
 - Validate that form data is correct when submitting to the server.

6. Requirements

- **CRUD Operations:** All CRUD operations for books should work correctly.
- **Validation:** Form data should be validated using Zod when submitted to the server.
- **UI/UX:** The user interface should be user-friendly and clearly display the list of books, and provide functions to add, edit, and delete books.