Product Requirements & Specification Document

Project Name

BookFinder - Library Search Interface

Description

BookFinder is a responsive web interface enabling users to search, filter, and view details of books in a library collection. The application is designed for educational use, focusing on usability and accessibility.

---

1. Goals & Objectives

- Enable users to efficiently search for books by title, author, or ISBN.
- Allow filtering by genre, availability, and publication year.
- Display detailed book information in a user-friendly layout.
- Ensure responsive design for desktop and mobile devices.

---

2. Target Users

User Type	Description
Library Users	Students, educators, general public
Library Staff	Staff searching for or managing books

---

3. Functional Requirements

ID	Requirement
FR1	Users can search books by title, author, or ISBN.
FR2	Users can filter results by genre, availability, and publication year.
FR3	Users can view a list of search results with key book details.
FR4	Users can select a book to view detailed information.
FR5	The interface is responsive and accessible on all devices.
FR6	Loading and error states are clearly indicated.

---

4. Non-Functional Requirements

ID	Requirement

NFR1	Built with React, TypeScript, HTML, CSS, Tailwind
NFR2	Loads initial data from a mock or static JSON source
NFR3	Follows accessibility best practices (WCAG 2.1 AA)
NFR4	Fast load times and smooth UI interactions

---

5. User Stories

ID	As a…	I want to…	So that…
US1	User	Search for books by title, author, or ISBN	I can quickly find specific books
US2	User	Filter books by genre, availability, year	I can narrow down my search
US3	User	View book details	I can learn more about a book
US4	User	Use the interface on any device	I have a consistent experience

---

6. Core Screens & Components

Screen/Component	Description
Search Bar	Input for title, author, or ISBN
Filter Panel	Genre, availability, and year filters
Book List	Displays search results with summary info
Book Detail Modal	Shows full book details on selection
Responsive Layout	Adapts to desktop and mobile screens
Loading/Error States	Feedback for data loading or errors

---

7. Data Model (Sample)

```
type Book = {
  id: string;
  title: string;
  author: string;
  isbn: string;
  genre: string;
  year: number;
  available: boolean;
  description: string;
  coverUrl: string;
};
```

---

8. UI/UX Specifications

- **Design:** Clean, minimal, education-focused
- **Colors:** Accessible contrast, Tailwind palette
- **Typography:** Readable, scalable fonts
- **Responsiveness:** Mobile-first, grid/flex layouts
- **Accessibility:** Keyboard navigation, ARIA labels

---

9. Technical Stack

Layer	Technology
Frontend	React, TypeScript
Styling	Tailwind CSS
Markup	HTML
Data	Static/mock JSON

---

10. Acceptance Criteria

ID	Criteria
AC1	Users can search and filter books with instant feedback
AC2	Book details are displayed in a modal or dedicated view
AC3	UI is fully responsive and accessible
AC4	All core features work with mock/static data
AC5	No critical UI/UX or accessibility issues

---

11. Out of Scope

- User authentication or account management
- Book borrowing/reservation functionality
- Integration with live library databases

---

12. Milestones

Milestone	Description	Target Date
UI Design	Wireframes & component design	Week 1
Core Implementation	Search, filter, list, detail views	Week 2
Responsiveness & Testing	Mobile, accessibility, QA	Week 3
Final Review	Stakeholder feedback, adjustments	Week 4

13. Appendix

- **Sample Data:** Provided as static JSON
- **Design References:** [To be attached separately]
- **Contact:** Product Owner, Dev Lead