

Book Finder App - Project Documentation

Introduction

The **Book Finder** app is a React-based web application designed to allow users to search for books from the **Open Library API**. Users can search for books by title, view detailed book information, and interact with a user-friendly interface featuring a sticky header, a modal for book details, and a loading popup on the homepage. This app is designed to provide a seamless user experience while exploring various books.

Project Overview

This project consists of several components and pages that work together to provide a simple yet functional book search experience. The core features include:

- **Search Bar:** Allows users to search for books by title.
- **Book List:** Displays a list of books based on the search results.
- **Book Modal:** Shows detailed information about a book when clicked.
- **Loading Popup:** Displays a welcome message and loading animation when the app is first loaded.
- **Header and Footer:** Provides the navigation and copyright information.

Features

1. **Search Functionality:**
 - Users can search for books by title.
 - The app queries the Open Library API and displays books that match the search query.
 2. **Book Display:**
 - The app displays book details such as title, author, publish year, and cover image.
 3. **Detailed Book View:**
 - Clicking on a book opens a modal with more detailed information about the selected book.
 4. **Loading Popup:**
 - When the app is loaded, a welcome popup message appears, which can be closed by the user.
 5. **Responsive Design:**
 - The app is designed to be fully responsive, adapting to various screen sizes, including mobile devices.
-

Technologies Used

- **React:** JavaScript library for building the user interface.
- **Axios:** HTTP client for making API requests to the Open Library API.
- **CSS:** Styling the components and app layout.
- **Open Library API:** Public API used to fetch book data based on user queries.

Project Structure

/BookFinder

|-- /src

 |-- /components

 |-- App.js // Main component that integrates all other components

 |-- Header.js // Header component with a logo and dropdown menu

 |-- Footer.js // Footer component displaying copyright information

 |-- LoadingPopup.js // Popup that appears when the app is loaded

 |-- SearchBar.js // Search bar to input book search queries

 |-- BookList.js // Displays the list of books from search results

 |-- BookCard.js // Displays individual book details in the list

 |-- Modal.js // Modal showing detailed book information

 |-- /assets

 |-- book-white.png // Image used in the loading popup

 |-- Library-logo.png // Logo image displayed in the header

 |-- /css

 |-- Header.css // Styles for the header component

 |-- Footer.css // Styles for the footer component

 |-- LoadingPopup.css // Styles for the loading popup

 |-- SearchBar.css // Styles for the search bar

 |-- BookList.css // Styles for the book list

 |-- Modal.css // Styles for the modal component

 |-- BookCard.css // Styles for the book card

 |-- App.js // Main app component

```
-- index.js           // Entry point for the React app
-- App.css             // Global styles
```

Component Breakdown

1. App.js

The App.js file is the entry point of the application, which handles the overall state management, API calls, and rendering of child components.

- **State Management:**
 - query: Stores the search query entered by the user.
 - books: Holds the fetched book data.
 - loading: Tracks whether the books are being fetched.
 - noResultsMessage: Displays a message if no books are found.
 - showPopup: Controls the visibility of the loading popup.
- **API Request:**
 - The app fetches book data from the Open Library API based on the query entered in the SearchBar.
- **Main Features:**
 - Displays a search bar for querying books.
 - Displays either a loading indicator, a no results message, or the list of books.

2. SearchBar.js

This component handles the input field and form submission for the search query. It also shows an information message below the input field to guide the user in making a valid search.

3. BookList.js

This component displays a list of books based on the search results. Each book item is clickable, which opens a modal displaying more detailed information about the selected book.

4. BookCard.js

This is a simple component that displays a single book's cover, title, author, and publish year. It is used within the BookList component to render each book.

5. Modal.js

This component displays a modal with detailed information about a selected book, including the title, author, publish year, and a cover image. The modal can be closed by clicking the "Close" button.

6. LoadingPopup.js

This component shows a popup when the app first loads. It provides a welcome message and a brief introduction to the app. The user can close the popup by clicking the "OK" button.

7. Header.js

The header contains the app logo and a dropdown menu to explore various book categories. It is sticky and stays at the top of the page as the user scrolls.

8. Footer.js

The footer contains copyright information and is displayed at the bottom of the page.

Key Features and Interactions

1. Search Functionality

- Users can enter a search query into the SearchBar.
- The app fetches book results from the Open Library API and displays them in the BookList.

2. Book List and Modal

- Clicking a book in the BookList opens a modal (Modal.js) with more detailed information about that book.
- The modal can be closed by clicking the "Close" button.

3. Loading Popup

- When the page loads, a LoadingPopup is displayed for a brief period with a message welcoming the user.
- The popup can be dismissed by clicking the "OK" button.

4. Explore Dropdown

- The Header.js contains a button to toggle the "Explore" dropdown menu, which provides various options like "New Arrivals," "Trending," and "Genres."

Styling and Animations

- **CSS Animations:**
 - Modal and loading popup components have animations that slide and fade in when they appear on the screen.
 - Hover and focus effects are applied to buttons to improve the user experience.
- **Responsive Design:**
 - The app is fully responsive, with media queries that adapt the layout for different screen sizes (mobile, tablet, desktop).

Conclusion

The **Book Finder** app is a user-friendly tool that helps users discover books easily by searching through the Open Library API. It offers a seamless experience with its intuitive design, interactive components, and smooth animations. The app is built using React, Axios, and CSS, ensuring a modern and responsive interface for users on any device.