U DACITY

PROJECT SPECIFICATION

MyReads: A Book Tracking App

Application Setup

CRITERIA	MEETS SPECIFICATIONS
Is the application easy to install and start?	The application was created with `create-react-app` and requires only `npm install` and `npm start` to get it installed and launched.
Does the application include a README with clear installation and launch instructions?	An updated README that describes the project and has instructions for installing and launching the project is included.

Main Page

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
Does the main page show three categories (or "bookshelves") for books (i.e., Currently Reading, Want to Read, and Read)?	The main page shows three shelves for books. Each book is shown on the correct shelf, along with its title and all of its authors. Each bookshelf is a reusable component.
Does the main page allow users to move books between shelves?	The main page shows a control that allows users to move books between shelves. The control should be tied to each book instance. The functionality of moving a book to a different shelf works correctly.
Does information persist between page refreshes?	When the browser is refreshed, the same information is displayed on the page.

Search Page

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
Does the search page have a search input that allows users to search for books?	1) The search page has a search input field. 2) The search page behaves correctly: a) As the user types into the search field, books that match the query are displayed on the page, along with their titles and authors. You can use throttle/debounce but are not required to do so. b) Search results are not shown when all of the text is deleted out of the search input box. c) Invalid queries are handled and prior search results are not shown. d) The search works correctly when a book does not have a thumbnail or an author. (To test this, try searching for "poetry" and "biography"). (It's fine to filter out books with missing thumbnails.) e) The user is able to search for multiple words, such as "artificial intelligence."
Do the search results allow a user to categorize a book as "Currently Reading," "Want to Read", or "Read"?	Search results on the search page allow the user to select "Currently Reading", "Want to Read", or "Read" to place the book in a certain shelf. If a book is assigned to a shelf on the main page and that book also appears on the search page, the correct shelf should be selected for that book on the search page. If that book's shelf is changed on the search page, that change should be reflected on the main page as well. The option "None" should be selected if a book has not been assigned to a shelf.

CRITERIA	MEETS SPECIFICATIONS
Do selections made on the search page show up on the main page?	When an item is categorized on the search page and the user navigates to the main page, it appears on that shelf in the main page.

Routing

CRITERIA	MEETS SPECIFICATIONS
Does the main page link to the search page?	The main page contains a link to the search page. When the link is clicked, the search page is displayed and the URL in the browser's address bar is /search. Routing is implemented with React Router
Does the search page link back to the main page?	The search page contains a link to the main page. When the link is clicked, the main page is displayed and the URL in the browser's address bar is /. Routing is implemented with React Router

Code Functionality

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
Does the project code handle state management appropriately?	Component state is passed down from parent components to child components. The state variable is not modified directly - the `useState` hook is used to add to function component. Books have the same state on both the search page and the main application page: If a book is on a bookshelf, that is reflected in both locations.
Are components built as functional components?	Components in the application are built as functions rather than as classes.
Does the code run without errors? Is the code free of warnings that resulted from not following the best practices listed in the documentation (e.g., a `key` for list items)? Is the code formatted properly?	The code runs without errors. There are no warnings that resulted from not following the best practices listed in the documentation, such as using `key` for list items, or state update warnings on unmounted components.. All code is functional and formatted properly.

CRITERIA	MEETS SPECIFICATIONS

Suggestions to Make Your Project Stand Out!

- Consider adding a books "detail" page to display more information about any particular book
- Consider adding sign-up and log-in functionality to allow users to save their bookshelves to their accounts
- Consider adding drag-and-drop functionality to move books between shelves