‹ Return to Classroom

MyReads: A Book Lending App

REVIEW | CODE REVIEW 10 | HISTORY

Meets Specifications

Congratulations for the great project. You have done a great job, huge improvement, I think you are ready for the next step with Redux. I hope you are having fun with React. 🌟

Application Setup

✓ The application was created with create-react-app and requires only `npm install` and `npm start` to get it installed and launched.

`npm install` and `npm start` was everything I needed to do in order to get the application running and dependencies installed, well done! 👍

✓ An updated README that describes the project and has instructions for installing and launching the project is included.

Your README file looks great, it does have the most important information about how to install and run your application, if you are interested in learning how to write a great README, here is a great course: Write README

Main Page

✓ The main page shows 3 shelves for books. Each book is shown on the correct shelf, along with its title and all of its authors.

✓ 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.

The control is very consistent on Search and Main page. Well done.

✓ When the browser is refreshed, the same information is displayed on the page.

Search Page

✓ 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.”

✓ 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 appears on the search page, the correct shelf should be selected 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.

✓ 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

✓ 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.

✓ 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 /.

Code Functionality

✓ Component state is passed down from parent components to child components. The state variable is not modified directly - setState() function is used correctly.

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.

Remarkable how you know how to pass down state from parent to children components. Well done.

✓ 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. All code is functional and formatted properly.

I like the way you coded your JSX, it’s clean, simple and readable. You got it. Here is some links that can help you how to improve your abilities with JSX:

Udacity Style Guide
React / JSX Style Guide

DOWNLOAD PROJECT

10 CODE REVIEW COMMENTS ›

RETURN TO PATH

Rate this review

START