CSC 360: Web Applications

Homework 4

React Todo app

Goal: Manage global state using React Context and add data persistence to the React Todo App built in hw2 and hw3 by issuing REST calls (using Axios and the useResource hook) to a mock API (powered by json-server).

Requirements:

Update your components to stop passing state and dispatch methods via props and instead create a StateContext, similar to the way demonstrated in class and then utilize then consume the StateContext in components which require access to state, dispatch, or both.'

Perform steps 2-7 to utilize a mock API to persist application data.

1. **Add StateContext**
    - Create a StateContext in a contexts.js file.
    - In App.js, wrap your components in a <StateContext.Provider> passing values for state and dispatch from your appReducer
    - In App.js, remove state and dispatch props from your UserBar, CreateTodo, and TodoList components (your component names and structure may be different depending on how you implemented things thus far); instead, utilize the useContext hook within any component which may require access to state or dispatch.

2. **Create a Mock API using json-server**
    Configure json-server to run alongside your application using the db.json, routes.json, setupProxy.js, and package.json startup commands contained within this zip file.

    Please reference the lecture for detailed instruction on how to utilize the files and commands to startup the mock API. **Note, the structure of your Todo and User objects may slightly differ than what I provided to you in db.json. Do not hesitate to update the objects within db.json to match the structure of your objects.**

3. **Configure an Axios instance and RequestProvider in index.js**

4. **Use the useResource hook in order to retrieve Todos from the mock API and remove the hardcoded intialTodos list in app.js.**

5. **Use the useResource hook in order to issue a POST request to the mock API in order to persist your Todo to db.json.**

6. **Use the useResource hook in order to GET/POST requests to the mock in order to perform Login and Register.**

7. **Use the useResource hook in order to perform the appropriate HTTP request to update a Todo existing in db.json for TOGGLE_TODO and DELETE_TODO actions**

Remember, in order to accomplish all of the above you will need to install json-server and concurrently globally using NPM. You will also need to install axios and react-request-hook as project dependencies using NPM.

Submission

1. 1. Commit your changes to the repo you used for hw3 and push your code to GitHub.

2. 2. Create a .txt file named **FIRSTNAME_LASTNAME.txt** containing a link to your GitHub repo. **Please ensure your repository is public.**

3. 3. Upload the .txt file to the D2L dropbox for hw4. This will indicate to me that your repo is ready for grading.