

CPSC 455 - Assignment 5

DUE DATE: **Tuesday, July 25 (12PM PT - Noon)**

- You will submit this assignment by sending/posting the link to your deployed assignment! (We will send out a form later.)
- **Keep your repo private**

Inventory Website... Complete!

For the fifth assignment, you'll be deploying the inventory website that you've been working on all term! It will build on the previous assignment, so start from your completed assignment 4, and begin to incorporate the requirements below. We expect you'll continue to put less time into assignments and more into the project.

We're expecting the following:

- 1) Deploy your inventory app by using any platform you'd like! Some options are
 - o Recommended: Render ([instructions](#))
 - o AWS
 - o Vercel
 - o Netlify
 - o Heroku
- 2) Ensure that once your app has been deployed, you can access it on someone else's computer, and that everything still works! (All the way from DB storage and your APIs to React, and any UI elements. All the requirements from assignment 4 must still be there.)
- 3) Setup a github action workflow to run your tests (see part 4) when 1) **creating a pull request** to merge into any branch and 2) **when merging your code into the main branch**. Your workflow can do other things, but it must run tests.
- 4) Something cool and extra! Write at least one non-trivial test. What kind of test you choose to write is up to you, can be BE or FE, unit, integration or end to end. We recommend using Jest (general testing) and/or react-testing-library (UI testing). What makes a test *cool*? (You do not need to do all of this, but awesome if you do.)
 - o Mocking dependencies
 - o Testing UI interaction
 - o Setup and/or cleanup (usually necessary for integration testing)
 - o Good coverage

Note: At the very minimum you should maintain the functionality based on the requirements from assignment 4. You do NOT need to maintain all of your functionality from the previous assignments (1, 2, 3), however, make sure you still have all the other technologies of the assignment, including React, Redux, Express, Node, and MongoDB! You shouldn't need to make drastic changes to your existing code.

Reminder of requirements from assignment 4:

- The data should now be stored in the database.
- When you add new items, they should be added to your database. You should be able to refresh the page, or even restart your server, and your data should persist.
- Ability to delete an item
- Additional functionality of your choice. Examples could be:
 - Editing an existing item
 - Filtering data from the DB
 - Getting details for an item (stored in the same table)
 - Getting details for an item (stored in a different table!)

As described in the individual assignment rubric, your code will need to meet these requirements and be functional, up to perhaps a few minor glitches in tricky cases. Note that functionality includes both user-visible and console-visible issues.

You should be ready to demo this to a TA during your second week lab, and should be ready to answer questions about it, as well as explain what you've done.

HAVE FUN!!!

FAQ:

Can I deploy to _____ platform instead?

Yes! If your web app is accessible from another computer/device, you've met the requirements! You're welcome to deploy to whatever platform you'd like, even if it isn't listed above.

How can I deploy to _____ platform?

Unless you're deploying to Render or maybe Heroku, we probably don't know. How you deploy your web app depends on the organization of your web app and what platform you choose. We may not have the expertise to help you, but Google has lots of tutorials or you can post on #assignments to get help from other students.