

COMP905 – Project 2 – Week 12

Due date: Friday, Nov. 21st @11:59 pm

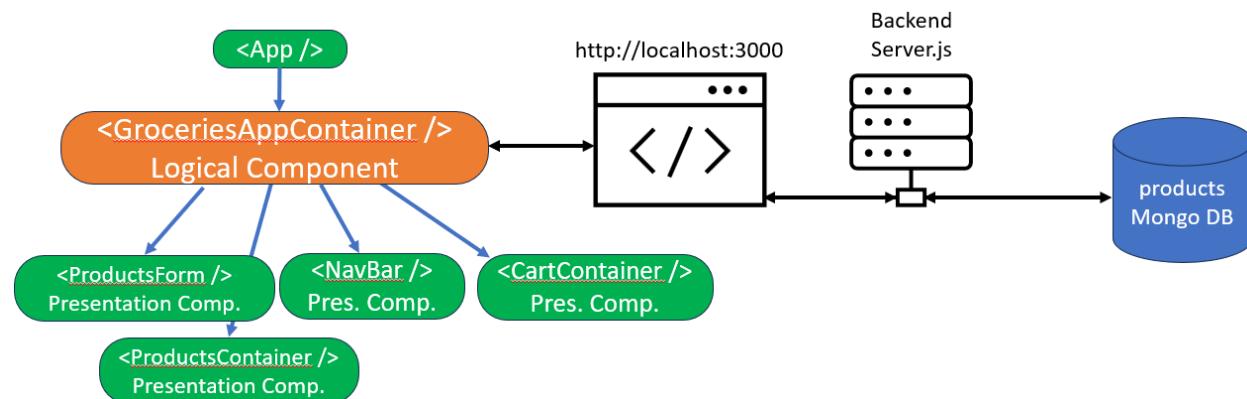
Grade: 20%

Project Description:

Welcome to Project 2 for COMP2004. This project covers most of the materials we covered over the last 12 weeks. This is the advanced version of Project 1, where we will move the product database to MongoDB and use a backend server to handle all CRUD operations in the application.

Note: You can continue building version 2 based on your previous submission for project one, or use the starter code repository I provided.

A summary of how this application works is as follows:



Version 2 of the app consists of the following:

1. Create a new MongoDB on the Atlas MongoDB Cloud platform to store the product's JSON DB.
2. Upload the attached (products.json) DB to Mongo using the Mongo Compass application.
3. **Backend server:** This server will be responsible for communicating with the MongoDB using JSON data format and the React app using an API approach. As shown in class, the backend server will receive data from and send it to an intermediary API. The server will also be responsible for modelling the data using a suitable MongoDB schema matching the product's data.
4. **Frontend framework:** A React application that performs frontend operations and queries the API for all CRUD operations. The application should be able to:
 - a. Display all the 24 initial products found on the initial database.
 - b. Create a form to enable posting new products.
 - c. Display all new products after the original products.
 - d. Enable deleting products by adding a delete button to each displayed product.
 - e. Enable product editing by adding an edit button to each displayed product. The edit button will display all current data in the form, enabling editing and saving to the DB.

Submission:

Please submit your GitHub repository link on Blackboard

Grading rubric:

Key Concept	Extensive Evidence	Convincing Evidence	Limited Evidence	No Evidence
Functioning Code	The code runs without errors and passes all test cases. (5%)	The code functions without errors, but not all test cases pass. (3-4%)	The code functions with errors, and some of the tests pass (1 - 2%)	The code is not functional. (0%)
Program Logic and Correctness	The program's output is as expected. (5%)	The program's output produces minor differences than expected. (3-4%)	The program's output shows significant differences from expectations. (1 - 2%)	The program does not output. (0%)
Commenting and annotations	The code includes extensive comments and annotations that describe its functions. (3%)	The code includes comments and annotations that describe its functions. (2%)	The code includes comments and annotations that describe its functions. (1%)	The code is without comments or annotations (0%)
File structure, variable and function naming	The file structure, variables, and functions are named descriptively. (2%)	The file structure, variables, and function names are somewhat descriptive. (1%)		The file structure, the variables and functions' naming are not descriptive (0%)