# Web App Design with React Week 15 Coding Assignment

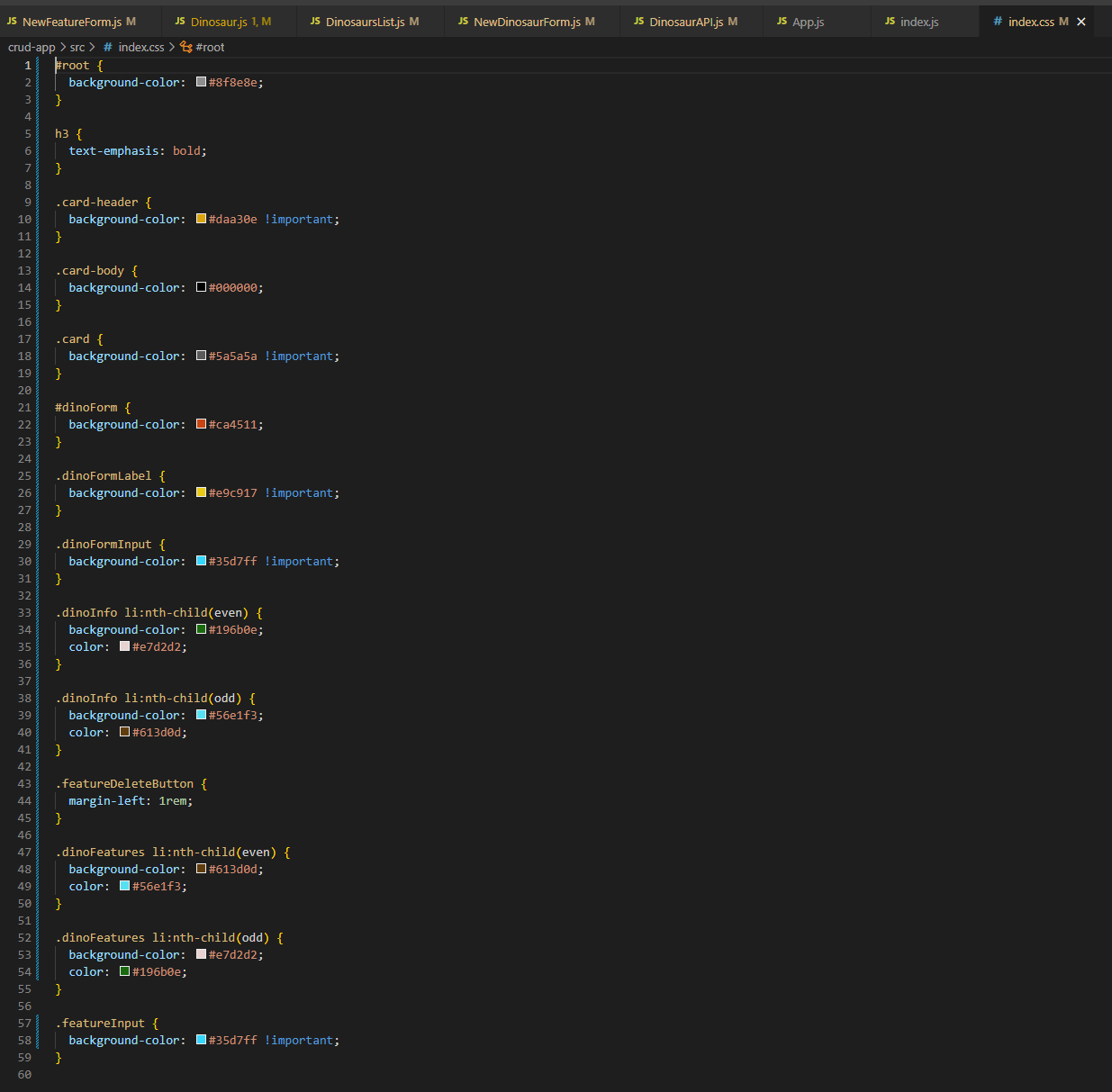
**Points possible:** 75  
  
**URL to GitHub Repository:** <https://github.com/spobrien1116/FEWeek15CodingAssignment>   
  
**Instructions:** In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week’s assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week’s repository to this document where instructed and submit this document to your instructor when complete.

**Coding Steps:**

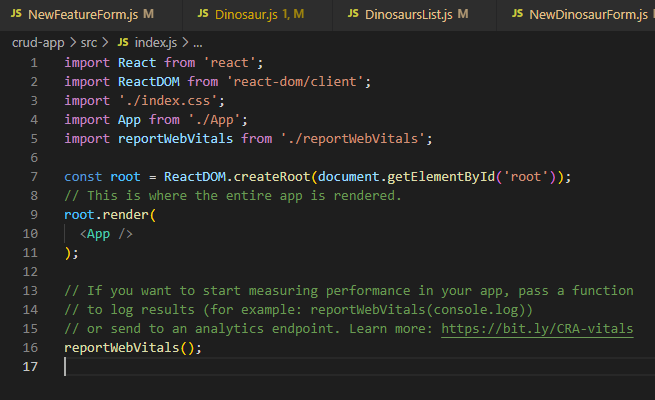
1. Using the Houses API, or any open API of your choice you can find online, create a single page that allows for all 4 crud operations to be performed on a resource from that API. Create a React component (or more, if needed) to represent the resource. Make all forms and other necessary UI pieces their own components as reasonable.

**Screenshots of Code:**

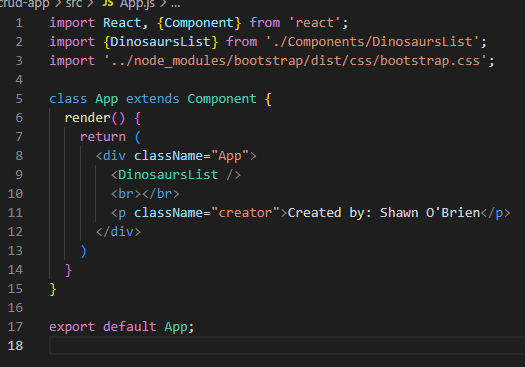
Styles:



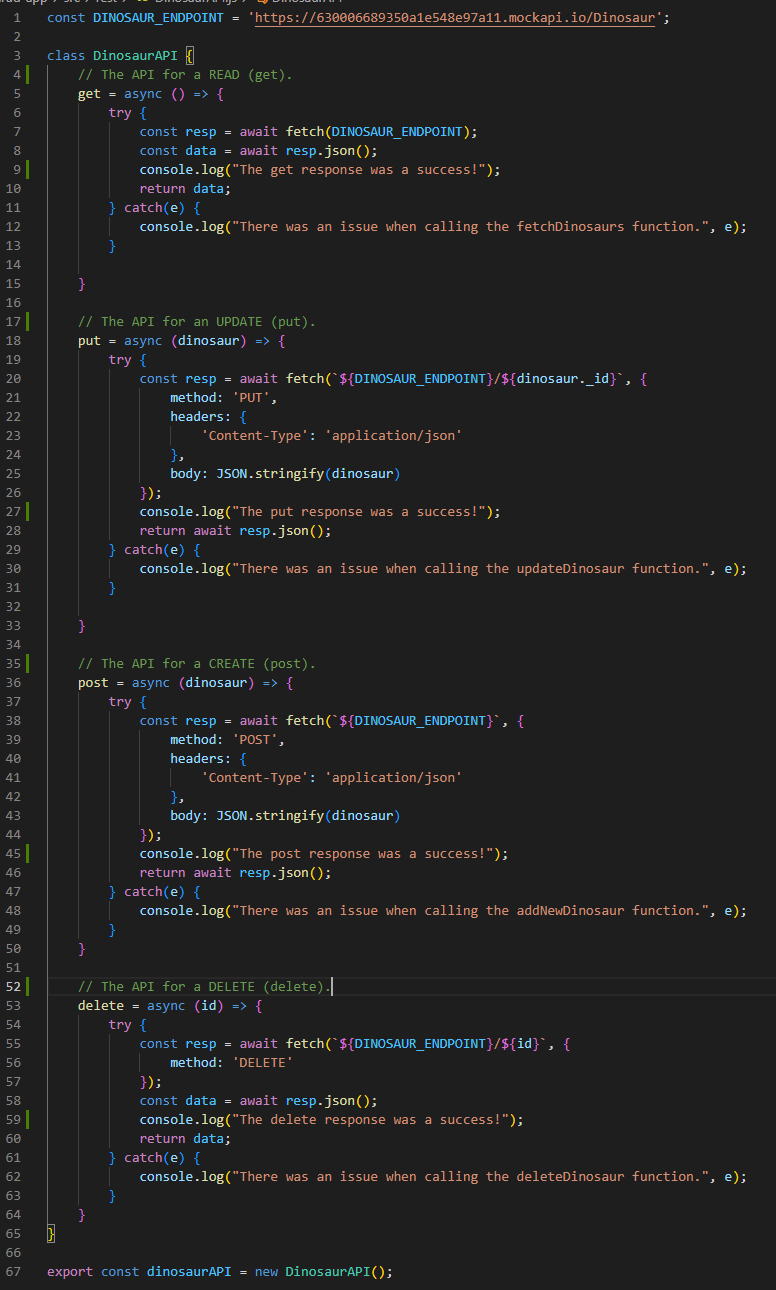
Index.js:



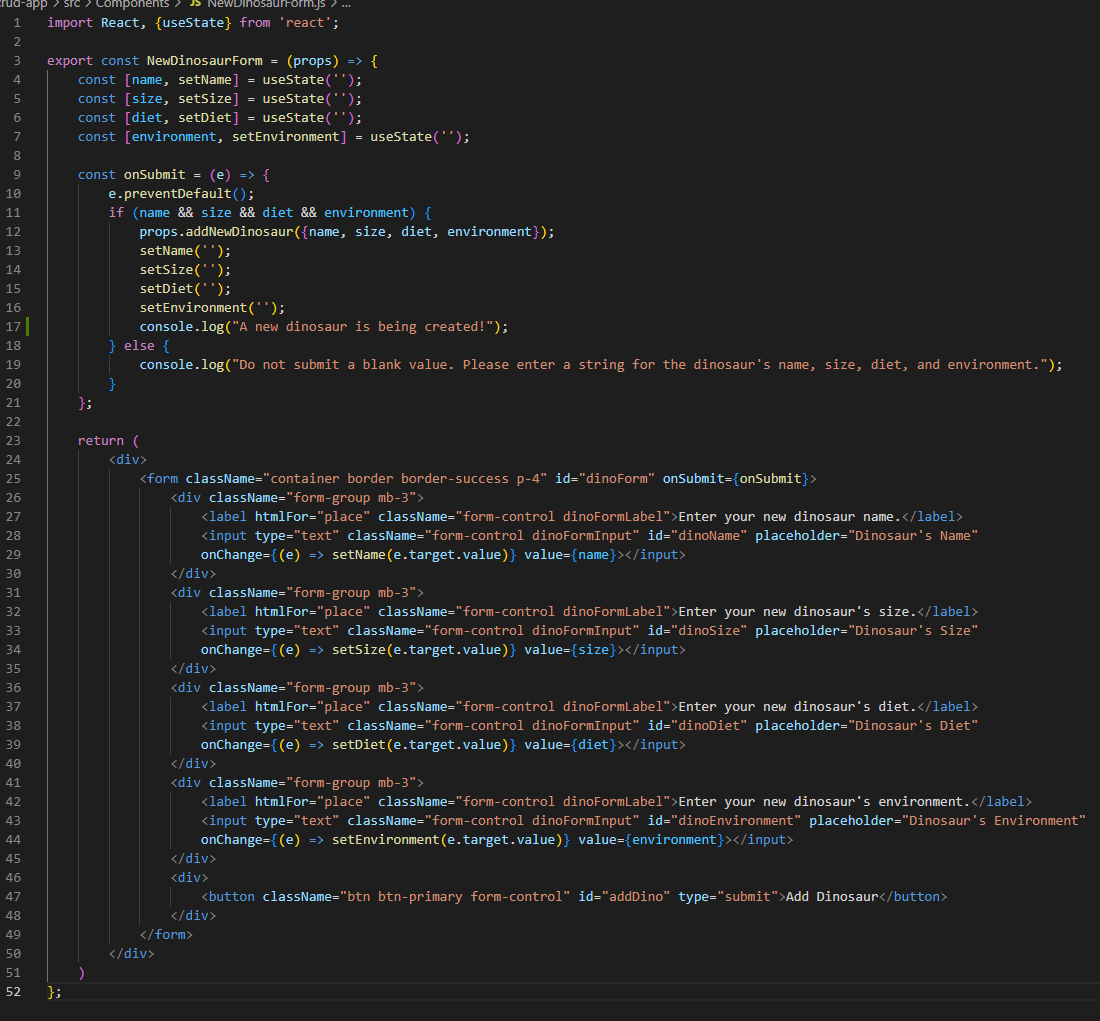
App.js:



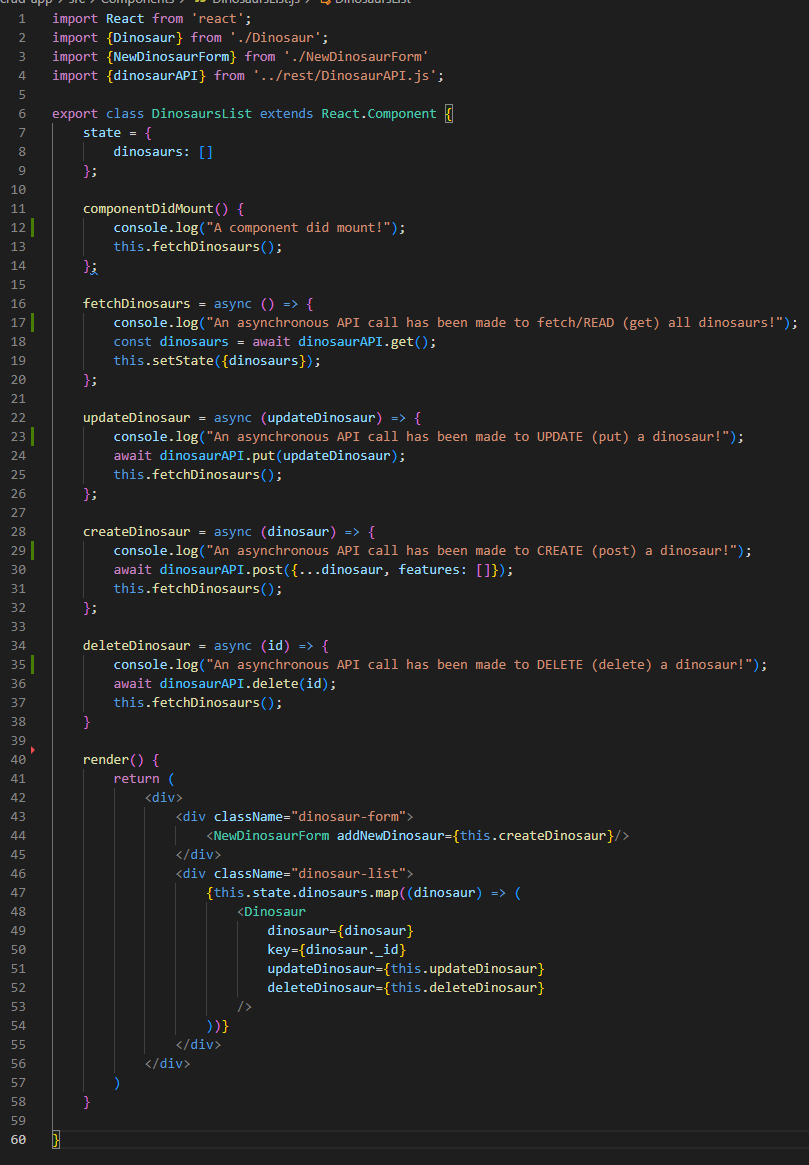
DinosaurAPI.js:



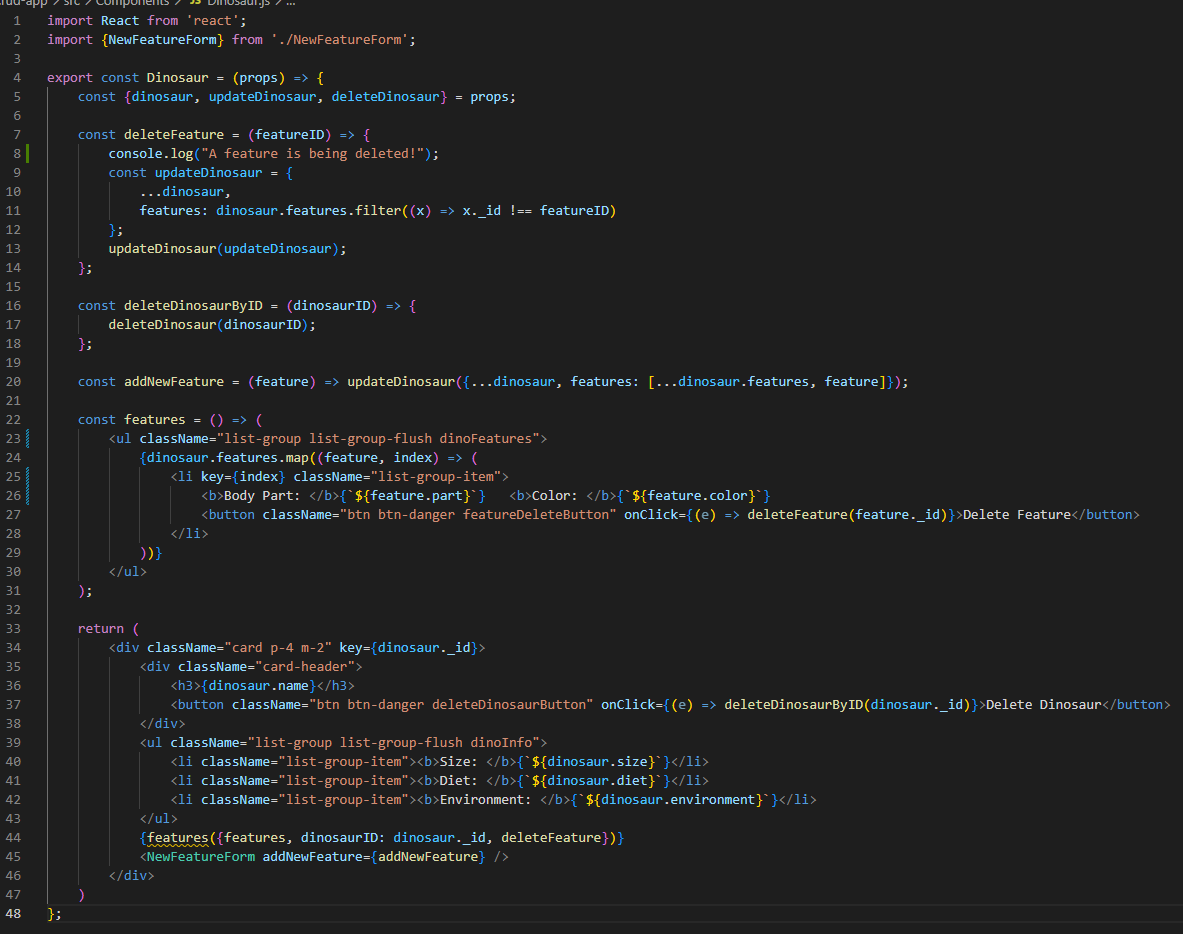
NewDinosaurForm.js:



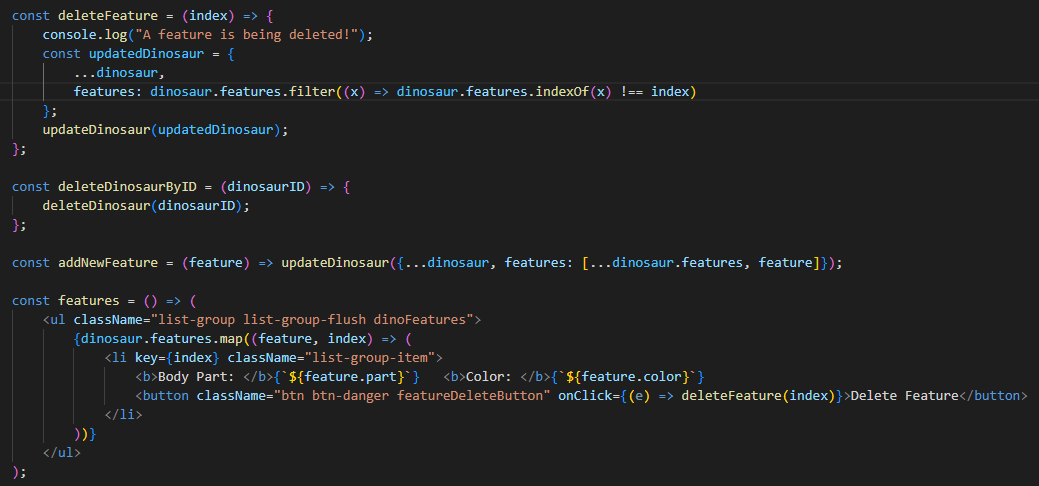
DinosaursList.js:



Dinosaur.js:



IMPORTANT! The screenshot below reflects an updated version of Dinosaur.js that allowed my updateDinosaur function within the deleteFeature function to work properly. Screenshot of the updated code:

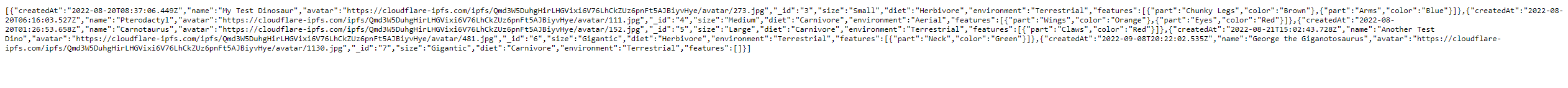


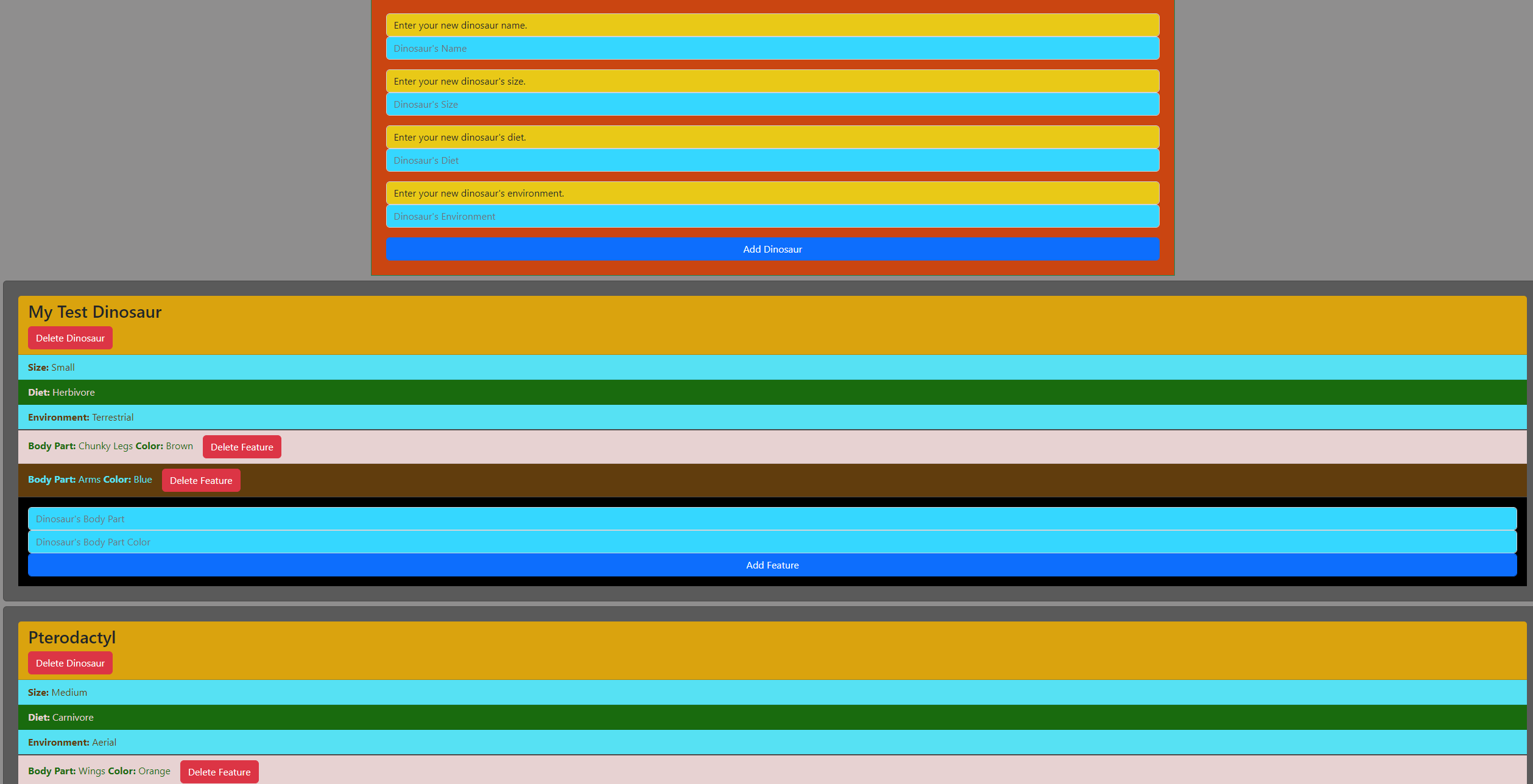
NewFeatureForm.js:

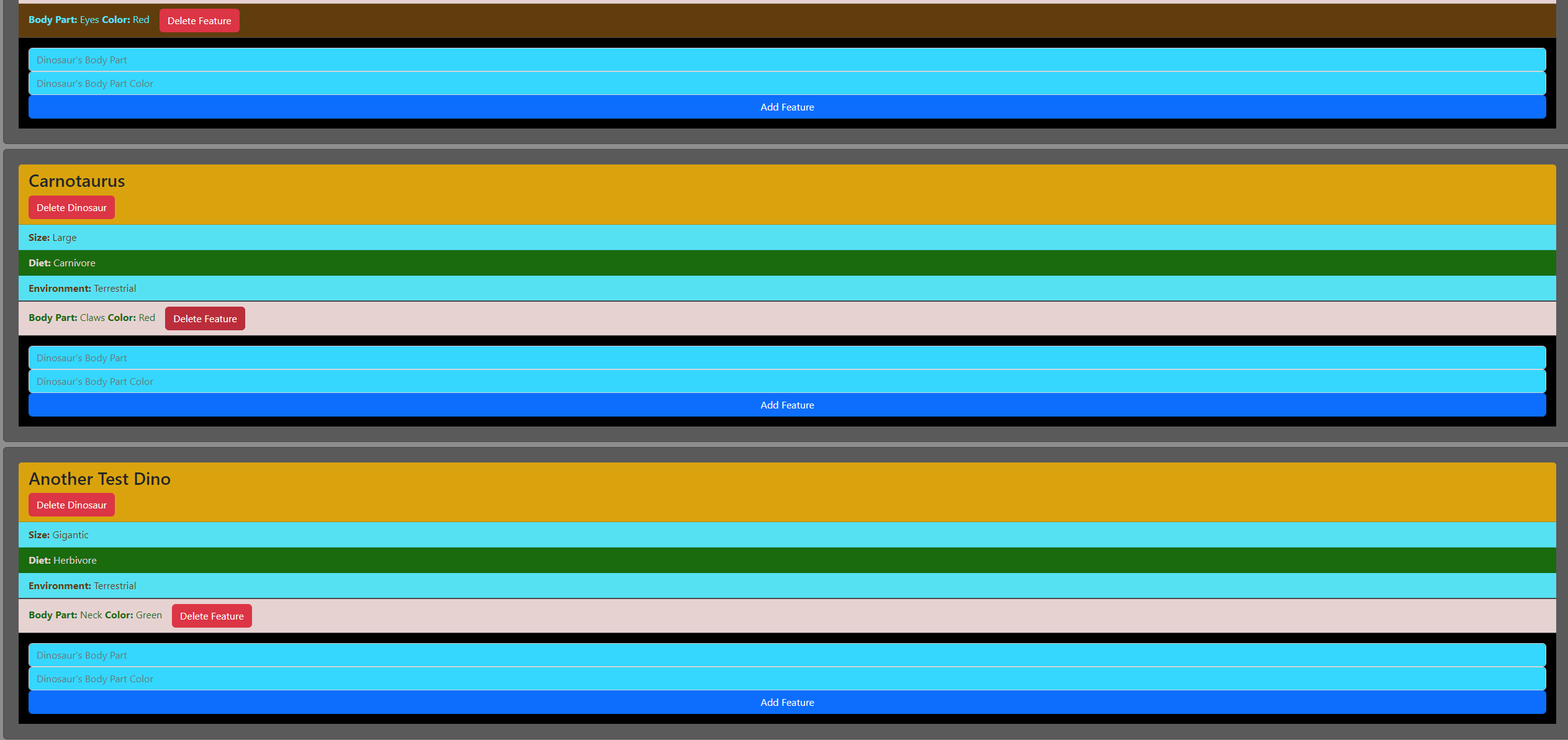


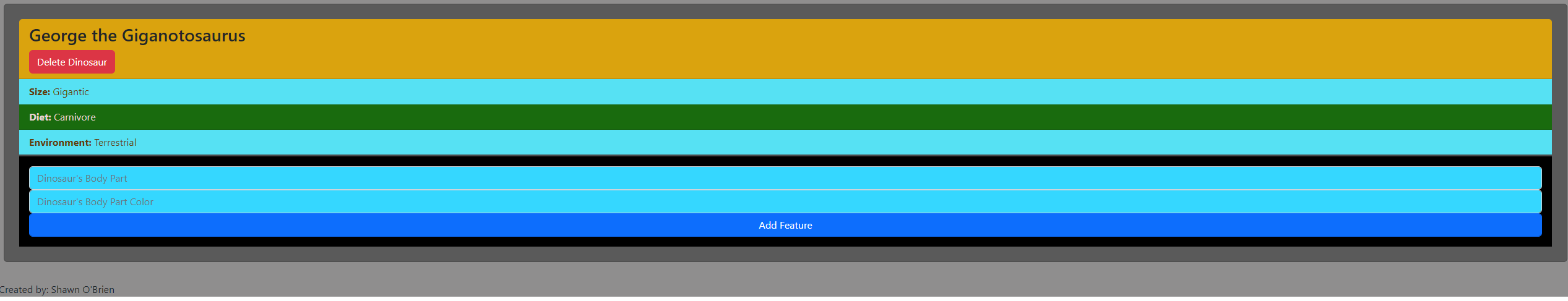
**Screenshots of Running Application:**

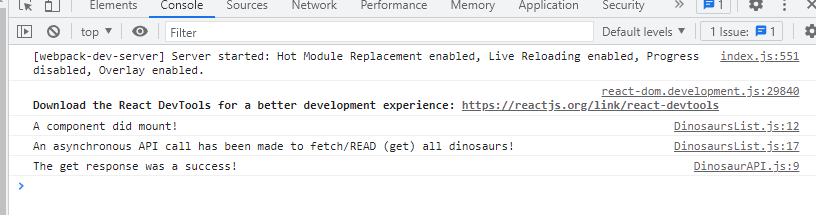
Screenshots of the database, application, and console logs before beginning CRUD operations:









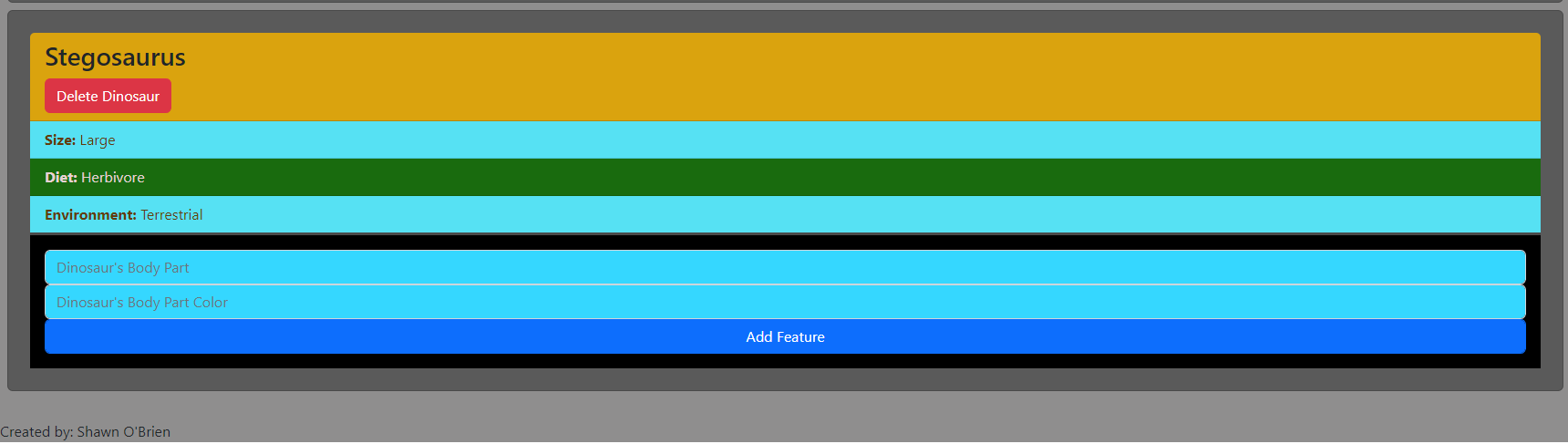


Read: The read API functions properly upon first rendering and each time a create, update, or delete is called. Screenshots of the successful get responses are shown in console logs below for these actions, in addition to being shown above upon the application first launching.

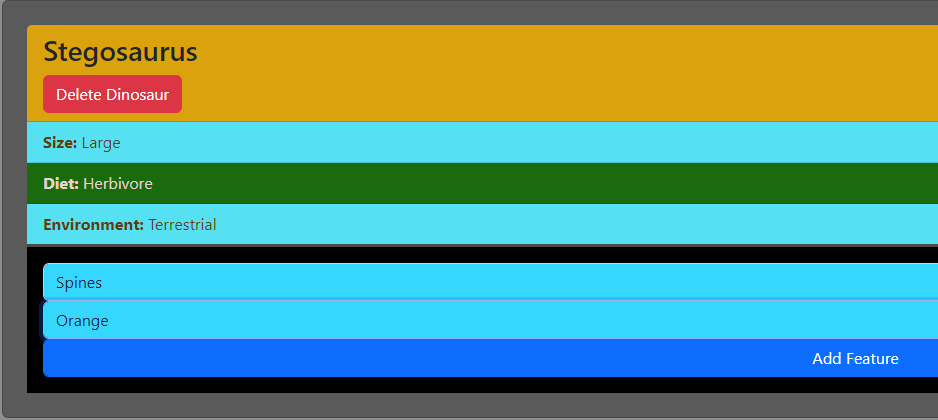
Create:

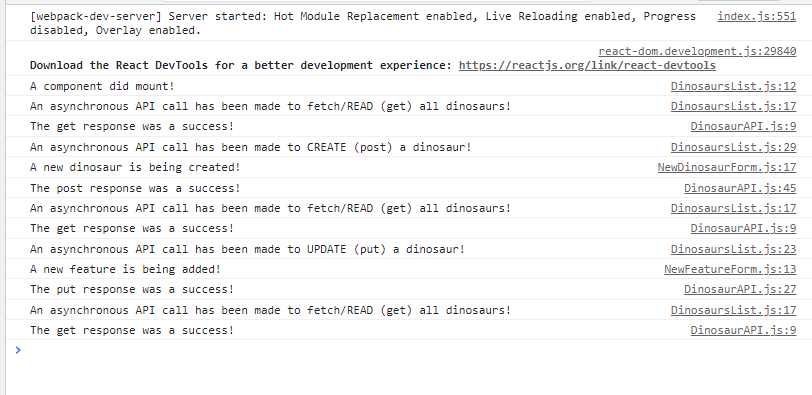




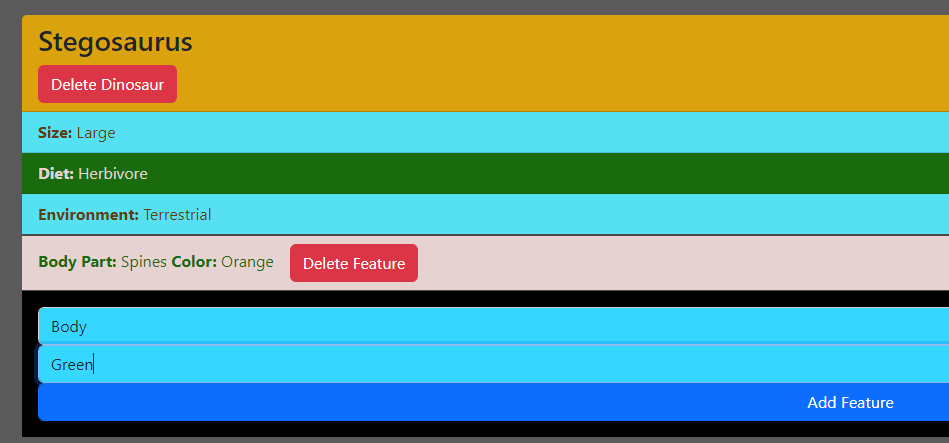


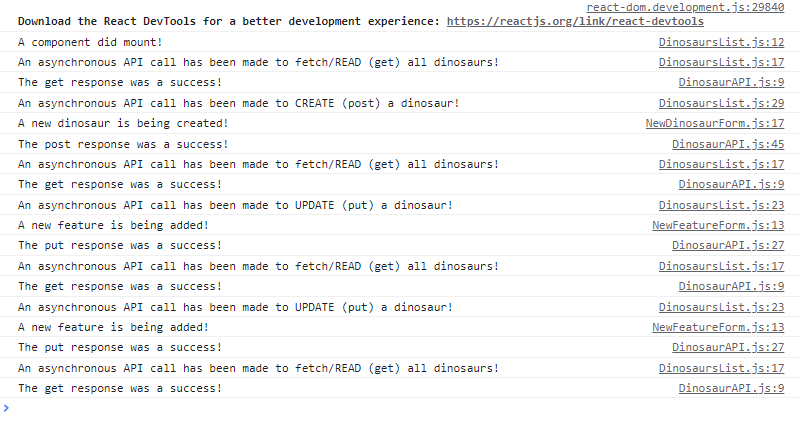
Update:

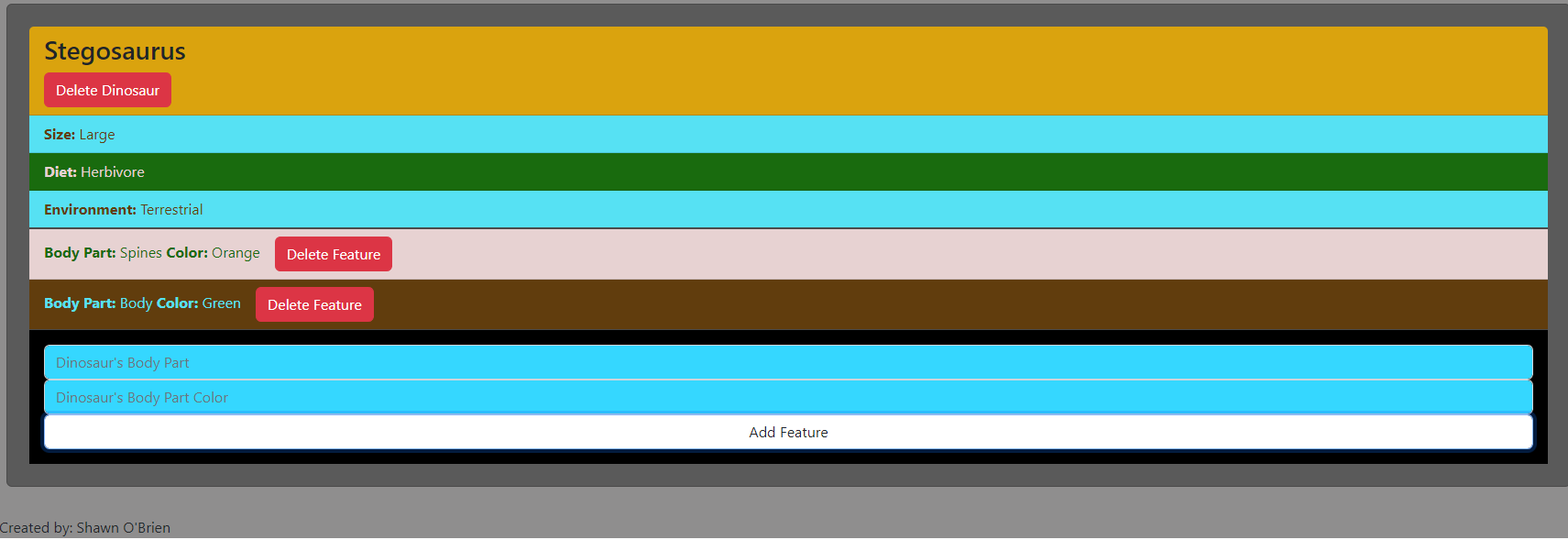




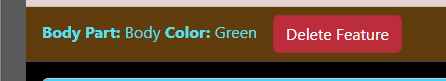






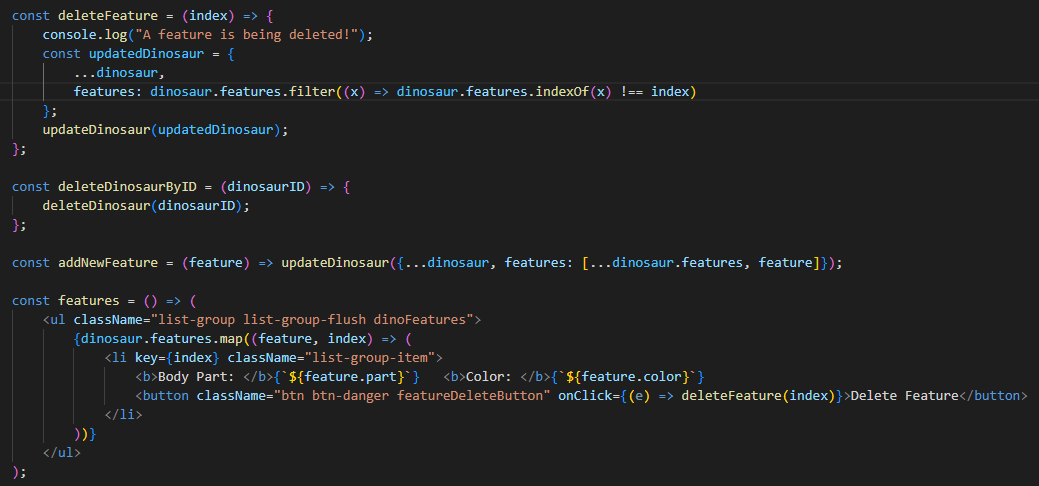


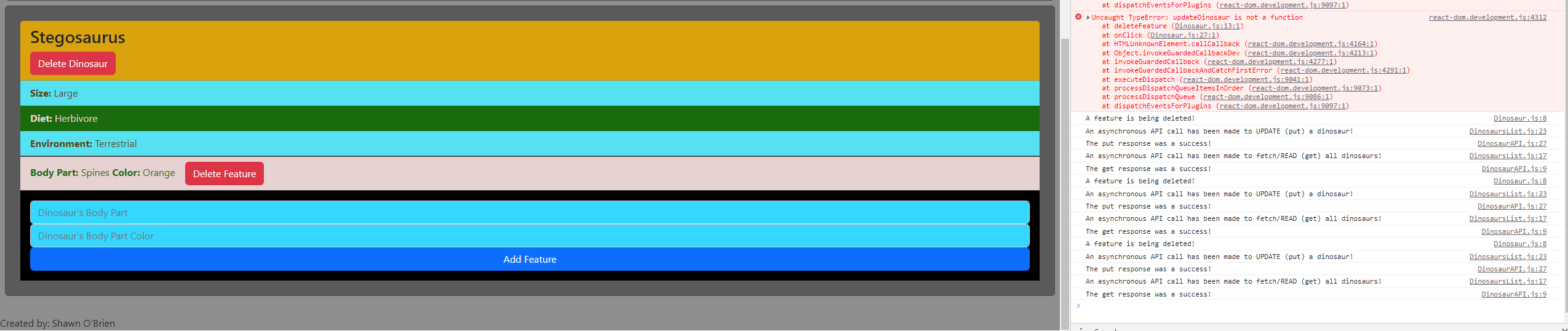
Delete (feature):



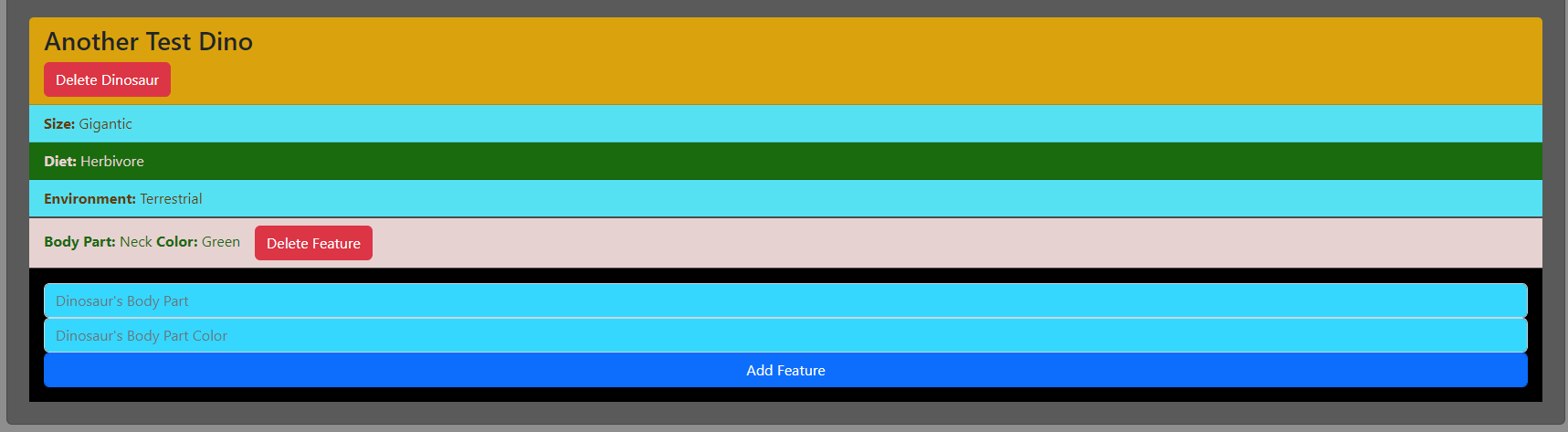
I fixed an issue with updateDinosaur not working. The console logs with warnings and eventual successes, plus the code changes that fixed the issue, are posted below. The updated code will also be posted above in the **Screenshots of Code** section Dinosaur.js to show the changes I made mid-screenshots/final testing.

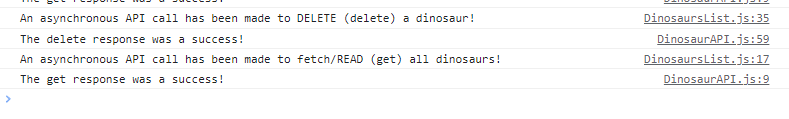




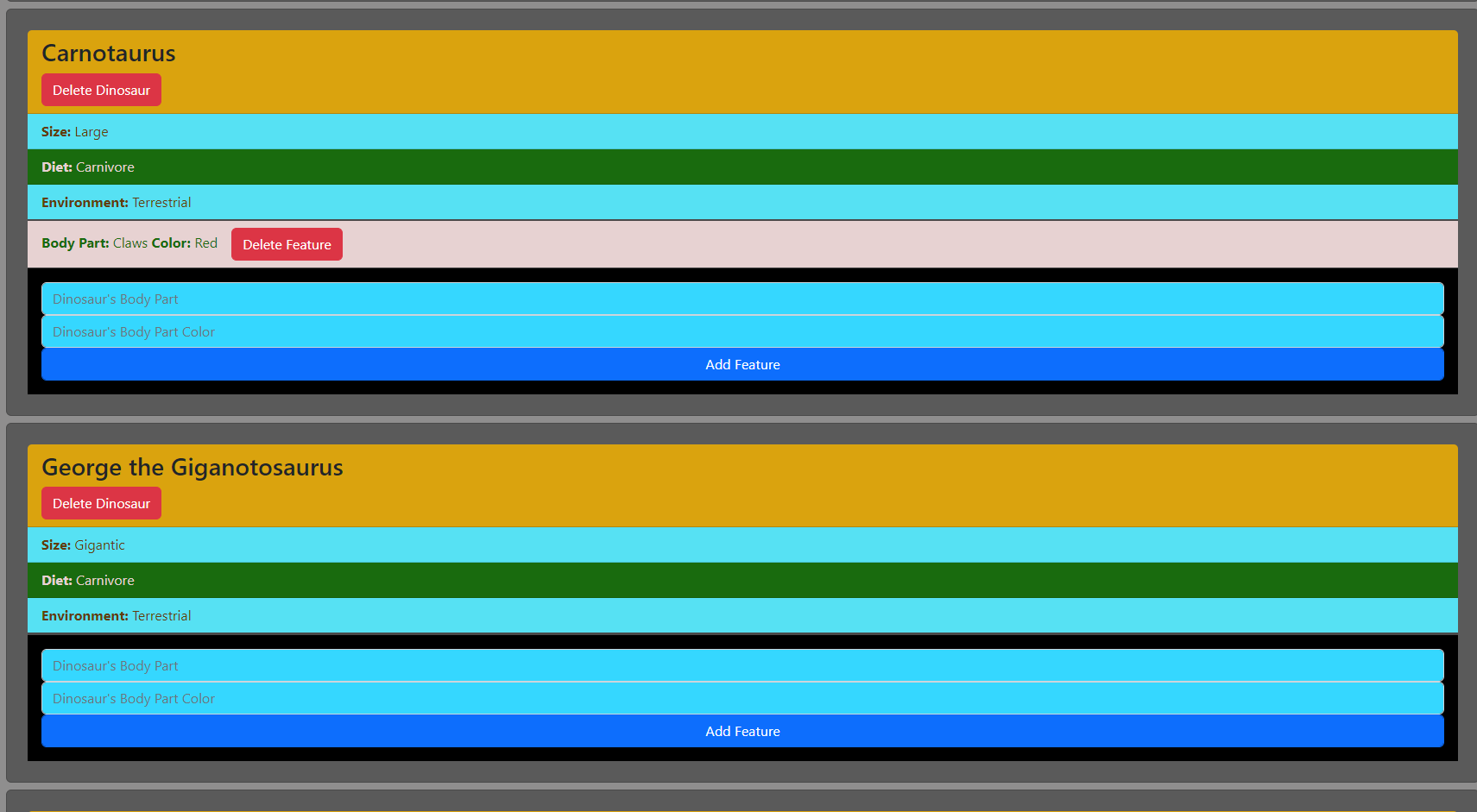


Delete (dinosaur):





The below screenshot shows that Another Test Dino is now no longer between Carnotaurus and George the Giganotosaurus.



And of course, every call has been properly re-displaying the list of dinosaurs and the dinosaur form.

Screenshot of API at the end of final test/screenshots:

