Corey Emery

05-430: Programming Usable Interfaces

Section B

October 30, 2018

Homework 6: HTML/CSS Implementation Reflection

**Bugs and Challenges**

The first challenge I faced in completing the shopping cart was the way I handled the option selections for each product. Since I did not use a dropdown menu or checkboxes to display the different choices available (I used different divs) I had to implement my own custom selection mechanism. I did this by first giving each option element the same class and having one element have a selected id. When the user clicks on an element, the jquery code will clear the ids for the other elements and assign the selected id to the element that was clicked. When adding to cart, we will use the text in this selected element to add to the item that we put in our object in memory (thus representing what our selected option was).

Also, one challenge was figuring out how to map the remove item button on the cart page to the correct option in memory. The solution I came up with was (though this is not best practice in industry) assigning the id of each button to be the index of the element it corresponds to. So if the id of the button is 3 and it is clicked, it will remove the object at the third element of the array in memory. The way my html is created programmatically in jquery also guarantees that the correct buttons will correspond to the correct elements.

Lastly, the creation of my cart proved an important and sizable task. There was a lot of html elements and nesting that went into creating the card in the correct format. I went about this by first doing it statically with one element in my html and css files, so that I was sure that the layout would be exactly as I wanted. I then used jquery to create the same structure (as a string) that I could append to a container component in the html. By doing this, I was able to separate the two different problems here (designing the component and generating the content programmatically based on the cart in memory), and thus work more efficiently through each task rather than become bogged down trying to tackle everything at once.