Programming Usable Interfaces

Assignment 6B

Salonee Gupta

Link to github repository, Link to website, Link to Figma prototypes

Reflection

1. **Getting the order of different updates in the cart right**
   Within the shopping cart various things needed to change on adding an item - the cart total, the subtotal and the total cost of the order. Initially I struggled with making sure that they all updated in a specific order -> item added, cart total, subtotal and then total. This was because I was thinking of each function as a separate task and not as interdependent processes. After I wrote some pseudo code to refer to that highlighted the dependencies, I was able to quickly code the updates in a way that made sense.
2. **Using one function for each task**
   Initially I was trying to create one giant function for all the updates that need to happen in the shopping cart. This was making it very hard to debug the mathematical and logical errors I was getting in my code. Eventually I broke the different functionalities into separate functions and passed the same variables through all of their arguments. This made it much easier to debug the code and keep a track of what's causing what to change.
3. **Have clear names for classes and IDs that make sense in context**
   I realised how important it is to be very specific when naming objects, classes IDs. I would name a class 'shoppingCart' and not be more specific and 2-3 days later I would not remember what specific class I was referring to within the shopping cart. This would create confusion while coding and I often got errors because of this. I eventually started being more specific about the names of things, for example, I used names like 'shoppingCartItem' and 'shoppingCartAddButton' instead of 'shoppingCart'
4. **Being specific while calling classes in CSS**
   In CSS, I learned to refer to a class or an ID by also calling their parent class. I often got errors because I used the same class name in different parts of my

code. While I started with the intent for both of those sections to look the same, over time they evolved to become slightly different. Once I started being more specific in my reference, I no longer got the errors.

Programming Concepts

1. **Using local storage to save objects**
   One of the biggest things I learned from this assignment was using local storage to save objects across different pages. While in theory the concept is fairly simple, I struggled initially with adding the functionality after I had already created a fully functional shopping cart without JSON (adding, removing from cart). I had to go back and write pseudo code to understand what functions would have to be rewritten using to accommodate for the addition of local storage functionality. For example, the checking of if an item is already in your cart needed to be re-written
2. **Retrieving data from HTML pages using Javascript**
   This assignment really pushed me to understand how to call different values from HTML pages and then manipulate them using Javascript. For the shopping cart, I needed to extract the item name, price, glaze, quantity and image for every item a user would add to their cart. What made this tricky was doing it in a way that remains consistent for all the different product pages. I learned the concept of using the [0] while extracting information from HTML to only get the first value. For example:
   ```
   var cartItems =
   document.getElementsByClassName("cart-items")[0]
   ```
3. **Learning how to access the DOM tree**
   It was important for this assignment to learn the concept of parent and child to be able to manipulate static HTML using Javascript. When adding event listeners, I had to extract information from the page that the event took place by requesting the parent element. This helped me properly understand DOM tree structure and also the value of creating a clear structure while creating the HTML pages.

For example: `var itemImage= button.parentElement.parentElement.parentElement`

4. **Learning how to add Event Listeners**
   It was necessary to have event listeners for this project. I learned how to quickly add event listeners to all the classes with a specific name across different HTML pages. This made my code scalable.
   For example:
```
var removeCartItemButtons = document.getElementsByClassName("trash")
for (let i=0; i<removeCartItemButtons.length; i++){
 var button = removeCartItemButtons[i]
 button.addEventListener('click', removeCartItem)
}
```
5. **Learning how to add transitions and animations**
   I had designed a slide in cart for my shopping cart. For the web implementation I needed to use a slide in transition and an overlay animation for my cart page. This involved learning how to add CSS, HTML and Javascript to make the transition smooth and consistent.
   For example:
```
var addToCart= document.querySelector(".add-to-cart")
if (addToCart){
 addToCart.addEventListener('click', function(){
   overlay.style.display="block";
   checkout.classList.add("slide-in");
 })
}
```

Resources and Assets

1. Stack Overflow for debugging
2. Unsplash for images for different rolls
3. Flat Icon for the icons used