Interactive Personal Website Reflection

Aisha Nukovic

November 17, 2024

Adding interactive elements to my personal website was a challenging yet rewarding endeavour. One of the most difficult aspects of adding interactivity to my personal website was dealing with the added “tertiary” code of JavaScript while also configuring the HTML and CSS layouts. Our first assignment seemed almost as if it were “two-dimensional” in that we worked with HTML and CSS codes simultaneously. Adding JavaScript to the project made things more complicated because it was as if we added a “third dimension” to the website. I found that one of the most difficult aspects was ensuring that the logic of JavaScript ran smoothly while the HTML and CSS styling also maintained the aesthetic look of the website. Sometimes, I would encounter issues where I would add new script in my JavaScript file, but the aesthetic look of the new element would not be uniform or pleasant. I would then go into my HTML and CSS code to improve the look, only for it to compromise the logic of the JavaScript code. In turn, this required a lot of back and forth among my HTML, CSS, and JavaScript pages – constantly checking and troubling shooting among the different codes to ensure a seamless and uniform look.

Another challenge was within the JavaScript code itself as I worked with it as an entirely new language. I had to come up with script logic for interactive features I wanted to implement in my website, which required a lot of thought, testing, and patience for the code to be able to perform in the ways I hoped it would. To achieve certain features, this process proved to be difficult. For example, I added a contact form modal that had limited functionality initially, so I wanted to add features like a status message, close button, and name input. The process for adding additional features and creating smooth interactive elements for the contact form took time to perfect. I had to debug the code constantly to troubleshoot issues of new features not displaying or submitting correctly, which was a bit of a frustrating process. I was so happy when I eventually resolved all the JavaScript, CSS, and HTML issues for my interactive elements and it turned out to appear more smooth, uniform, and consistent.

One other challenge with my contact form was planning how to make the feature actually work for users to be able to send me an email to my personal email. I had to research ways to integrate this capability without using back-end logic (because our JavaScript unit focused solely on front-end logic). I found a third-party feature that integrates email system without having to code the server-side logic, EmailJS, which was really useful, but had challenges in its own way. Now that I was dealing with a third-party feature, I had to learn how to use that platform and navigate the difficulties that came along with it. For example, I had to teach myself how to use the EmailJS platform and learn how to integrate its functions into my code. I also had to create a template for how a user’s submitted email would look when it is sent to my personal inbox. It was a little daunting at first, but I managed to integrate the EmailJS features into my contact form after quite a bit of time.

Overall, I am happy with the way my interactive elements turned out. A win was being able to finally make the contact form and validation work and look the way I wanted it to. Another win was adding more stylistic interactive elements to my webpage, such as the “typing” header and “scroll to top” button. These elements add more visually engaging features to my website and make it easier for a user to navigate through the information in my website. It was also important for me to maintain the look of my personal website’s brand throughout the new JavaScript code and I am happy I maintained a consistent and uniform look with those elements.