

Group Member: Brian Chao, Tianyi Li, Milad Imen

### **CSE 134B HW3 Analysis**

#### **Vanilla**

Lines of Code Written: 734

Hours Taken: 18 hours

Load Page	Size (161KB from background image)	Load Time	DOM Content	Finish Time
login.html	162.9KB	473 ms	305 ms	583 ms
home.html	161.951KB	477 ms	308 ms	415 ms
shoppingList.html	162.362KB	437 ms	290 ms	384 ms
search.html	172.445KB	448 ms	298 ms	397 ms

Coding vanilla CSS was a great way for us to learn how to use all kinds of CSS components to improve the presentation of our web pages. It allowed great customizability and we were able to match our wireframes closely. We spent a decent amount of time figuring out how to position html elements the way we wanted to as well as researching different ways to combine CSS properties to achieve the desired presentation. We also noticed that our background image greatly increased our load times so we are currently considering if we should remove it or not.

## Bootstrap

Lines of Code Written: 417

Hours Taken: 8 hours

Load Page	Size (161 KB from background image)	Load Time	DOM Content	Finish Time
login.html	185.358KB	689 ms	431 ms	597 ms
home.html	185.262KB	516 ms	320 ms	444 ms
shoppingList.html	185.409KB	494 ms	301 ms	403 ms
search.html	185.571KB	538 ms	306 ms	408 ms

Coding using bootstrap CSS was significantly faster and easier to use. The bootstrap came with classes that had convenient padding and margins and was already mobile friendly. We simply had to browse through the bootstrap components and see how we can use them to improve our website's presentation. The grid-based structure was also very useful in positioning html elements the way we wanted. We also learned how to use multiple classes at once to achieve aesthetically pleasing effects. The best part about bootstrap CSS was that it greatly decreased the number of lines of code we had to write as well as the amount of time spent.

## Comparison and Conclusion

Coding using bootstrap CSS was much faster and easier to use than coding using vanilla CSS. With vanilla CSS, we had a lot of trouble positioning html elements and figuring out how to make our website mobile friendly as well. The bootstrap CSS came with a grid-based structure and contained many classes that were similar to what we were trying to design using vanilla CSS. For example, creating a navbar using bootstrap CSS simply included adding classes to our HTML elements while creating a navbar using vanilla CSS required us to style each of the html elements and modify the css properties. However I believe the bootstrap CSS helped us a lot more simply because of our inexperience with CSS in general. Learning how to use the components of the bootstrap CSS was easier than learning how to use CSS properties to achieve the same desired affect. The bootstrap CSS can't cover everything needed so vanilla CSS can still better to use for complete customizability.

We noticed that the loading times for bootstrap CSS was consistently longer than the vanilla CSS. This makes sense because the size of the style sheets from bootstrap CSS was significantly larger than the size of our vanilla CSS style sheets which means that it would take longer to load. This was one of the drawbacks of using bootstrap CSS. Every line of code in our vanilla CSS was relevant in our website's presentation but many features in the bootstrap CSS was unused and slowed down our loading time. Another drawback was that some of the classes provided in the bootstrap CSS wasn't exactly how we wanted to present our website so we had to add our own vanilla CSS to customize it further.

In the end we decided to go with the framework version of our website. Using bootstrap CSS decreased our development time and decreased the amount of headaches over positioning html elements. The bootstrap CSS also includes a lot of awesome features using javascript we foresee using such as modals. The load time increase isn't noticeable and we are still able to customize our website exactly how we want using bootstrap CSS and just a little bit of vanilla CSS.