Reflection

1. How to navigate the site:
   1. Click “Home” “Menu” “About”, you will be directed to the corresponding section.
   2. Click “Blueberry Roll” in the menu section, you will be directed to the product detail page.
   3. Click “Continue Shopping” button, you will be directed back to the menu section on home page.
2. User Interface Bugs
   1. Firstly, the home page was overloaded  
      In my paper prototype and InVision prototype, I designed a pop-up modal that overlay on the top for each clicked item to show product details. I thought it looked great because the user does not need to go to another page to view item details. However, when I converted it into code (as a third iteration), I realized that I overloaded my home page too much with “home”, “menu”, “about”, “contact” all available in one view. Therefore, I decide to give the user a break by creating a separate product detail page.
   2. Secondly, the contact info shouldn’t be only accessed through footer  
      Initially, I made it a separate page for contact info, and the only entry point is a small link in the footer. I did not take care of this issue back in previous prototypes. This time I simply removed the majority of the footer, and added contact section at the end of the home page.
   3. The logo does not have to be sticky if the home page isn’t long  
      I have a big square-shaped logo that directs the user to the top of the page, and it was supposed to stick on the top of the site when scrolling. However, it hides a great chunk of content and this affects the efficiency. Therefore, I added a floating button that gives the option to “go back to top”.
3. Challenges
   1. Responsiveness

It is not that hard when you put fixed pixels in the .css file. But when it comes to responsiveness, it could be a whole different story. I have to refresh myself with flexbox and grids by looking up real examples, and actually try to play with the thing on JSbin before I put it into my code. It is the most challenging part about this assignment because not only do I have to think about how to align items under different viewports but also code it in a smart way.

* 1. Tricks

Css do have some tricks that if you know it, you can do it in a single line of code, but if you don’t, you have to achieve the same effect with dozens of lines. At first, I struggled a bit to align the text vertically to a banner image. I could have just set the line height equal to the banner height (of course it’s not the way to do it) but when the viewport gets smaller, the second half of the text disappears. I literally spent an hour googling and finally arrived at a solution that uses a flexbox and setting the alignment to auto. I realized that if I can really make good use of the default settings of css, I could build powerful things, and learning the tricks requires long-time practice and effort.

1. Brand Identity

I continued with color palette I used in my previous prototypes. I got inspirations for the visual styles from bakery stores, and their websites. Usually bakeries use warm colors like brown, dark green, orange, to convey their ideologies about healthy food and lifestyle. I want Bun Bun Bake Shop’s website to be fun (that’s how its name indicates), and friendly. Therefore, I use a brown as my primary color and rounded fonts throughout the site. I think the brand identity is fully reflected through my design choices.

1. Other Consideration
   1. Naming

Naming is hard. I did not count the number of classes I have but I bet there are at least thirty of them in 360 lines of css code. I deliberately chose the names to make sure they reflect the semantic meaning of the element and the functionality.

* 1. Grammar

I tried to not make inline css/js in my html file to ensure that html is purely for the structural purpose. I also didn’t overuse id’s because it is usually better to create universal style for elements.

* 1. Legibility

I tried to make indentation consistent. I will add more comments in the next assignment to ensure legibility.

* 1. Presentation Style

I think I did ok in styling. For future improvement, I’ll clean up my css code to make them less redundant if so.