Sanitary Services Company web site redesign

1. Introduction

<screenshot of original SSC index page>

The purpose of this project is to clean up and modernize the look of the existing SSC website (<http://www.ssc-inc.com/index.php>). I also wanted to improve navigation so that the entire site would be accessible from the index page.

<screenshot of index.html>

New and improved index.html

2. Design and Implementation

<screenshot of original design>

My wireframe for index.html

<screenshot of index.html>

New and improved index.html

(upload index.html and styles.scss)

Predictably, my design changed in the transition from wireframe to code. The main structural changes were as follows.

* Put “About”, “Careers”, “Contact Us” links on top in the jumbotron.
* Moved buttons around a bit, created a button for “Drop-off Locations” rather than a link.
* Streamlined the legalese.

Dealing with the chicken-and-egg problem: Should I start with the index.html page, code in vanilla html/Bootstrap, and then create the scss file? Or should I make the css file first? (At the time, I didn’t realize it was better to just go straight to scss.) I went back and forth between index.html and got myself thoroughly messed up.

Structuring the SCSS file: \_variables.scss and the other Bootstrap scss files to

Debugging:

Aesthetics:

Managing scope creep: figuring out where to pull the plug because I really needed to get back to JavaScript and general programming concepts.

something like a Udemy Bootstrap course because there is just too much I don’t know.

The index page includes the following features.

|  |  |
| --- | --- |
| Logo | I dug up a plain SSC logo on Google and used that rather than the logo on the existing SSC website.  I attempted multiple antialiasing fixes but eventually realized that the reason I wasn’t getting good results was because the image didn’t have pure solid colors for either the blue or the white areas.  Eventually I need to set up and learn drawing/photo manipulation tools, but for now I decided to just use the image as-is, make the jumbotron white, and move on. |
| Jumbotron | and summarizes what SSC does in a way that makes the site more accessible to search engines. Maybe talk about how I’ve made this match the meta tags. |
| Carousel | Human interest, why we’re a great company. Decided to move this under the jumbotron to make the page more interesting, it’s an attention-grabber for major announcements like weather closures. |
| Navbar | Greatest single difficulty in coding the index page was figuring out how to control text color for the links. I eventually learned that this problem is a infamous Bootstrap hazing ritual. |
| Search button | Shortcut for users who know exactly what they want. |
| Buttons for Recycling Guide and Drop-off Locations | Wanted to make these more prominent than links. Whenever I’ve gone to the actual SSC web site, these are the things I’ve been looking for, things a return visitor would want to find immediately. |
| Accordion cards for Residential, Commercial, Education, and News & Events | Major function areas are accessible from accordion cards (less cumbersome than menu dropdowns, especially for mobile users). |
|  |  |
| Footer | Legalese is at the bottom of the page. |

For styles.scss,

It took me a lot of thrashing to figure out how to override the default Bootstrap colors and other formatting.

, and among other things I wasted too much time doing CSS formatting in the styles.css file. What I should have done earlier was work straight in styles.scss and copy/paste default formatting over from \_variables.scss and other Bootstrap .scss files and then figure out exactly what I wanted to override from the standard features.

Basic things I’ve learned: how to insert fonts, use a color palette generator, how to set navbar text colors, how to extract portions of \_variables.scss and override the colors using variables, how to use Chrome dev tool for CSS debugging, how to incorporate some basic Bootstrap features.

What threw me off from doing this was the (IMHO, counterproductive) way that Nucamp teaches CSS and Bootstrap. In the nucampsite styles.css, we were creating these random styles (kind of doing the stylesheet equivalent of inline formatting). In the real world, people using Bootstrap probably don’t do this anymore. (Anybody with half a brain or who isn’t working on legacy code, that is.)

You make a skeleton web page, just accept the default Bootstrap formatting.

rather than styles.scss/overriding \_variables.scss. The reason I did that was because I thought that if I put all my formatting into styles.scss, I would have to rebuild the entire project every time I wanted to see my changes in the browser. I didn’t now all I had to do was one simple command (npm run scss). As usual, this was probably something everyone else in the class probably clued into instantly, but my fear of

3. Conclusions

My original wireframe design was decent for a first website. Unfortunately, once I started coding I got lost in the woods. My main takeaway from that whole experience is that I need to drop out of Nucamp for a while and work on basic skills before going on to React.

* **Discuss the best features and the shortcomings of the project.**

Learn to use VS Codes folding already.

If I had this project to do over…

Stylesheets: I would have coded in styles.scss, not styles.css.

Nucamp helps you set up a development environment and shows you how to code various basic features of a website. But there’s no handholding for how to craft a website.

The biggest shortcoming of the project is, obviously, that I didn’t finish, but I wanted to go back to learning JavaScript.