**RACHEL TUREK SULLIVAN | WEBSITE ASSESSMENT**

**DEVELOPER NOTES**

**EASY | design detective**

The first evening, I analyzed the website assessment documents and created a basic wireframe for the HTML structure of the webpage. I also took the time to gather branding data from Code Authority’s website. After a little bit of sleuthing, I found higher quality versions of the 3 images shown on the example page, as well as a version of Code Authority’s logo that I was able to modify. Summaries of these findings and my process throughout the exercise can be found later in this document, and the various iterations are in the [GitHub repository](https://github.com/retsullivan/Code-Authority) I used for this project.

**HARD | blank slate**

Most of my recent front-end work has been using a content management system like Squarespace or Wordpress, so starting from a blank canvas took a little longer to get rolling. An additional wrinkle was my usual Windows machine succumbing to old age mid-project. I had to switch to my MacBook which I haven’t used for dev work in a while - updating and the lack of correct converters for external monitors delayed the coding portion of this project by a day. Thankfully, a multitude of online resources allowed me to find the answers to my questions, and Amazon Prime was able to deliver the monitor converter quickly.

**STYLE OBSERVATIONS | staying on-brand**

To make my site as close as possible to the Website-Assessment and Code Academy branding, I sampled directly from the PNG and checked them against the colors and fonts on the Code Authority website using the developer tools. Most colors and fonts were the same, but I don’t own “Gotham” and had to substitute with a different font.

**Colors**

|  |  |
| --- | --- |
| HTML Color Code | Locations |
| #0e5478 | Logo, selected tab in navigation bar |
| #3195b4 | Logo, section titles, stripe at top of footer |
| #87bbd2 | (found in the real logo, but is different from the one in the assessment) |
| #c2e0ed | Logo |
| #7d7d7d | Footer background |
| #4a4a4a | <p> text, blog titles |
| #138b76 | One vertical line - doesn’t match rest of site |
| #FFFFFF | Font in Footer, social media links |
| #000000 | “Website Assessment” image, unhighlighted links in nav bar |

**Fonts**

|  |  |  |
| --- | --- | --- |
| **CA Website Fonts** | **Assessment Fonts** | **Used For** |
| Gotham | Helvetica | Headings, Labels |
| Helvetica | Helvetica | Paragraph/description text |

**ASSETS | virtual treasure hunt**

A fun aspect of this assessment was reverse engineering the stock photo assets. I used Adobe Illustrator to capture the individual photos in Website-Assessment.png. Then I used a reverse image search to find the original images, and modified them as needed. I considered using a clipping mask on the website assessment image in illustrator instead, but decided to focus on image quality over the simplest solution.

|  |  |  |
| --- | --- | --- |
| **Mac on Desk** | **Source** | **Modifications Needed** |
| Mac and Desk.png | <https://www.scgpacewisdom.com/img/services/ios/pexels-photo-205316.png> | Changed the screen to the Code Authority Logo  Modified Mac & Desk with Code Authority logo.jpg |
| **Do More on Screen** |  |  |
| Work-From-Home_Blog-1920x1080.jpg | <https://griffisresidential.com/app/uploads/2019/09/Work-From-Home_Blog-1920x1080.jpg> | NONE |
| **Man and Charts** |  |  |
| Man and Charts.jpg | <https://www.onatlas.com/wp-content/uploads/2019/09/pexels-photo-212286-1080x675.jpg> | Cropping |
| **Website Assessment Logo** |  |  |
| Website-Assessment graphics_WA logo.png | Created in Illustrator |  |
| **Code Authority Logo** |  |  |
| ca-logo-9kb.png | https://www.codeauthority.com/images/ca-logo-9kb.png | Though a clipping mask would have worked to grab the differently formatted logo from the assessment png, I recreated the logo in illustrator so that it was less pixilated.Recreated code authority logo-02.png |

**Wireframing | measure twice, cut once**

Before I start writing code on a new site, I prefer to draw wireframes to help guide the structure of the site. Color coding them helps me keep track of nesting when working in HTML, and I’m able to think about how to achieve the desired page layout. Though this does take a little extra time on the front end, I find that I’m able to achieve a page with the correct spacing and responsiveness much more quickly when I have a visual guide.

|  |  |
| --- | --- |
| **PC** | **Mobile** |
| **Website-Assessment graphics_wireframe.jpg** |  |

My final mobile version ended up looking a little different, but planning it in advance still helped me think though what could and could not be easily done using just HTML and CSS. Without javascript, changing to a sandwich drop down menu seemed outside the scope of this exercise (though I did run across a [fun little trick](https://codepen.io/mutedblues/pen/MmPNPG) using pure css and a restyled checkbox). With javascript, a responsive menu is [much easier](https://www.w3schools.com/howto/howto_js_topnav_responsive.asp).

Had I been working on a React project, I would have created separate desktop and mobile components for each section and reordered the mobile version and reduced the data repetition on the page (like the phone number and links).

**Media Queries | reaching the breaking point**

After I finished replicating the basic layout of the original Website Assessment PNG, I decided to make it more responsive and mobile friendly before submitting. Since this is a sample product, I planned to investigate just a few potential breakpoints. However, after testing the responsiveness of the Flexbox based css, I was able to reduce the final set set to only two.

**Original Breakpoints to Check**

|  |  |  |
| --- | --- | --- |
| **Breakpoint** | **Range** | **Reason** |
| 1920 | 1920 | Original Assessment PNG width |
| 1024 | 1024-1920 | iPad dimension (both pro and regular) |
| 768 | 768-1024 | iPad dimension |
| 375 | 375-768 | iPhone Dimension |
| 280 | Minimum | Smallest phone dimension I could test in Chrome dev tools |

I tested my breakpoints using the dev tools and adjusting the window size. The media queries don’t seem to work correctly using the device toolbar in Chrome, but they work when resizing the browser window with dev tools closed. I suspect that has to do with compatibility with the viewport units I used.

**Final Breakpoints**

|  |  |  |
| --- | --- | --- |
| **Breakpoint** | **Range** | **Reason** |
| 1081 | 767-1081 | The Contact Us & Blog sections had issues with text wrapping starting at 1081px |
| 767 | Under 767 | This size and under was completely reformatted for smaller tablets and mobile devices |