Descriptive Wireframe Document – Final Website (Assignment #4)

Website Idea: Similar to W3Schools; Tutorial for code, allows for people to learn about and try new effects from the 3 languages learned. Potentially a "showcase" type of website. Each new page can represent effects from either CSS or JS.

Concepts learned and pulled from:

1. https://www.w3schools.com/html/
2. https://www.w3schools.com/css/css3_2dtransforms.asp
3. https://www.w3schools.com/js/default.asp
4. https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_animations/Using_CSS_animations
5. https://dev.to/am20dipi/how-to-build-a-simple-search-bar-in-javascript-4onf
6. https://www.w3schools.com/howto/howto_js_progressbar.asp
7. https://flexiple.com/javascript/javascript-clock
8. https://www.atatus.com/blog/powerful-css-selectors/
9. https://www.browserstack.com/guide/build-a-website-using-html-css
10. https://marketsplash.com/tutorials/javascript/javascript-syntax/
11. https://dev.to/etulan/javascript-syntax-5f44

Wireframe links:

1. Homepage : https://wireframe.cc/HKPSuk
2. CSS: https://wireframe.cc/QeNGJD
3. JS: https://wireframe.cc/TPVfH9

Rationale: Why did you choose this design? Be sure to talk about CRAP Theory and Color Theory.

I decided to design my website like this because I felt that it could combine all of the CRAP and Color theory together well while allowing me to naturally incorporate all requirements for the assignment. Firstly, I had the idea to make a website similar in concept to W3 Schools, where you'd be able to learn about code and some interesting features of the language too. It allowed me to naturally incorporate the more advanced features of CSS and JS while making the format simple but effective. I decided to do a dark on light theme, mainly focusing on a blue and green palette as they complement one another. I also felt that they have enough hues to fill all the scenarios I may need, such as a darker colour for a footer and a light background for text to sit upon. As the concept is a "learning" website, I felt the dark on light provided the contrast needed to pull the viewers attention to text and related examples, and when I wanted to place more focus on something, I could increase the contrast with dark colours to pull attention. I focused on a repetitive structure to help convey information and help the viewer easily distinguish when new sections started and ended. I found that a columnar arrangement made it easy to dedicate "areas" to either text, images or code examples.

The alignment of each section depended on the information being presented. I mainly stuck with titles and new sections being center aligned, and sometimes given an underline to draw more attention. All remaining text was left aligned as I felt it was the easiest for the viewer to read and "snap to". However, for the columnar arrangement, it was used to distinguish an "informational" section from the

examples. Typically the information sections were floated left with the images and corresponding images/examples floated left. For the example sections, they were inverted, with the images and examples floated left and text floated to the right. For each text block, they have an associated image and example that has similar spacing to show they are related. Thus, the use of a dark on light contrast and the columnar format made it easy to organize and pair similar content.

The choice in colour was used to complement one another, focusing on blue and green to achieve this effect. However, the website turned out to be more analogous as hues in between were focused upon, making it more of a spectrum along blue and green instead. In this case, the colors were cool, having a more calm and relaxing effect. As a student, learning can be stressful. By playing with colours that can help remedy these emotions, the viewer can stay more engaged and let their creativity take place when learning about interesting code. Furthermore, from personal experience, older students tend to prefer cool colours in designs, especially for educational purposes. Overall, by choosing this theme and design I feel I was best able to convey all the information in an interesting way.

Design Requirements:

1) Site Requirements: (30 Marks)
    - ☑ ~~Mock Up~~
    - ☑ ~~Descriptive Documents~~
        - ☑ ~~Rationale~~
2) Basic Site Components: (40 Marks)
    - ☑ ~~3 Separate HTML Pages - All need to link to each other~~
        - ☑ ~~Page Header~~
        - ☑ ~~Nav-bar to other pages~~
        - ☑ ~~Content area~~
        - ☑ ~~Footer~~
    - ☑ ~~Unified color theme (with the use of color theory)~~
    - ☑ ~~Actual content (at least 500 words), and cite any words, images or other media used~~
    - ☑ ~~Proper HTML and CSS compliance~~
        - ☑ ~~JS File won't validate~~
    - ☑ ~~At least two separate JS functions~~
3) Advanced Site Components: (30 Marks)
    - ☑ ~~Multiple elements = More marks~~
    - ☑ ~~Advanced concepts from any of the three languages~~

Advanced Components Added:

- Hover animations (listed in the W3 schools CSS section under advanced concepts)
- Rounded buttons (listed in the W3 schools CSS section under advanced concepts)
- Working search bar that highlights found text
- CSS text animations X2
- JS clock
- JS progress bar

Acknowledgements:

The CSS animations were taken from MMDN web docs. I have previously learned to do something similar as I had a co-op opportunity in grade 11 at a web design company. However, I focused on learning to implement the animation and fire it based on a button click. So I used it as an opportunity to further my JS knowledge and skills while introducing something I found pretty cool. For the other animations and JS elements, I used other sources as well as W3 Schools to learn how to implement them before trying myself to best fit my website. The JS was not directly taken, rather used as an introductory step to learning and to give myself ideas. Furthermore, some images were pulled directly from sites and others I have photoshopped myself. Images pulled are included in the citation list provided on the next separate page in MLA 9th edition format.

References

Aarthi. “10 CSS Selectors That Will Boost Your Coding Skills.” *Atatus Blog - for DevOps Engineers, Web App Developers and Server Admins.*, 18 Jan. 2023, www.atatus.com/blog/powerful-css-selectors.

“CSS Tutorial.” *W3 Schools*, www.w3schools.com/css/default.asp. Accessed 28 Nov. 2023.

DiPietro, Adriana. “How to: Build a Simple Search Bar in JavaScript.” *DEV Community*, 15 Nov. 2021, dev.to/am20dipi/how-to-build-a-simple-search-bar-in-javascript-4onf.

Etulan. “JavaScript Syntax.” *DEV Community*, 19 Mar. 2021, dev.to/etulan/javascript-syntax-5f44.

“How TO JS Progress Bar.” *W3 Schools*, www.w3schools.com/howto/howto_js_progressbar.asp. Accessed 1 Dec. 2023.

“HTML Tutorial.” *W3 Schools*, www.w3schools.com/html. Accessed 28 Nov. 2023.

“JavaScript Tutorial.” *W3 Schools*, www.w3schools.com/js/default.asp. Accessed 28 Nov. 2023.

Joshi, Mohit. “How to Build a Website Using HTML and CSS | BrowserStack.” *BrowserStack*, 10 Mar. 2023, www.browserstack.com/guide/build-a-website-using-html-css.

Konovalova, Marta. “JavaScript Syntax: An Essential Component of Web Programming.” *MarketSplash*, 30 Oct. 2023, marketsplash.com/tutorials/javascript/javascript-syntax.

“Programming a JavaScript Clock - Flexiple Tutorials.” *Flexiple*, flexiple.com/javascript/javascript-clock. Accessed 1 Dec. 2023.

“Using CSS Animations - CSS: Cascading Style Sheets.” *MDN Web Docs*, 4 Dec. 2023, developer.mozilla.org/en-US/docs/Web/CSS/CSS_animations/Using_CSS_animations.