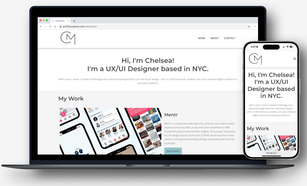
Portfolio Site Case Study



Project Overview

In order to grow my skillset as a UX designer, I decided to take on the challenge of learning frontend development by taking a Frontend Development for Designers course. In doing so, my goal was to be better equipped to collaborate with development teams by understanding design limitations and the fundamentals of coding.

Using HTML, CSS, and Javascript, I put this goal into action by building my own professional, functional, and responsive portfolio website from scratch over the course of 8 weeks in order to showcase my experience and my work.

Project:	Role:	Tools:	Duration:
Portfolio Website	UX/UI Designer Frontend Developer	VS Code, GitHub, Figma	8 weeks

Problem

As a new UX/UI Designer, I would need a way to display my skillset and past work in order to effectively market myself to potential employers.

In addition, I would need an easy and seamless method by which to network and communicate with other designers and professionals in the field.

Solution

By learning the foundations of frontend development through HTML, CSS, and Javascript, I would be able to build my own custom portfolio website to showcase both my UX/UI design skills as well as my new skills in frontend development. By creating my own site from scratch, I would be able to effectively tailor the layout and function to my target audience while also exhibiting my personality and style.

Goal

My end goal for this project was to build a professional online portfolio where fellow designers and future employers could learn more about me, see my past work, and connect with me all in one seamless, easy-to-use website.

Process

1 DISCOVER — 2 DEFINE — 3 DEVELOP — 4 DELIVER

Discover

Similar to the initial stages of a UX project, this process began with discovery and information gathering. I began by researching the qualities of a great portfolio versus a not-so-great portfolio. During this stage, I read dozens of articles and analyzed portfolios from designers of all levels to get a better sense of things that were done well and aspects that could benefit from improvement.

Through this discovery phase, I developed a list of elements and requirements that my website should include:

- Easy and straightforward navigation
- A brief and friendly introduction
- Organized work section
- About me page to discuss my background
- A link to download my resume
- A way to contact me via email, phone, and LinkedIn

Define

After the discovery phase, I was then able to move onto defining the key features and overall structure of my website. My main sections would include: Work, About Me, and Contact.

The Work section would serve as the home page, with a brief introduction, followed by a list of my case studies. The About Me section would include a little bit about my background and experience, along with a link to download my resume. The Contact section would provide various ways to connect with me including phone, email, and LinkedIn.

Color Palette

It was during this stage in the process that I also defined my color palette for the website. For my background and text, I chose neutral colors to ensure all of my work would seamlessly integrate into the site, and to avoid clashing with any colors included in my case studies. The neutrals would also help to convey a minimal and modern aesthetic throughout the site, which, in turn, should help to emphasize my projects.

For calls-to-action, I chose a brighter accent color to draw attention to the appropriate elements.

Primary: #FFFFFF, #F5F5F5, #4D4D4D

Secondary: #66ACDB

Typography

For typography, I chose to use Montserrat for titles and headers to help convey a bold and modern feel. For the body text, I chose Open Sans to contrast with the bolder feel of Montserrat, while still maintaining a modern and friendly quality.

Titles/Headers: Aa Montserrat SemiBold; Aa Montserrat Medium

Body text: Aa Open Sans Light

Develop

After the discovery phase, it was time to move onto the development stage of my portfolio website. My primary goal was to create a website that was not only reflective of my skills and experiences but also responsive and accessible across a wide range of devices.

Mobile-First Approach

I began by laying down the structural foundation of my website with HTML, starting first with my index page (index.html), and building out the architecture to include my project pages (mentor.html & flame.html), About Me page (about.html), and Contact section (#contact). HTML5 offered me the semantic elements I needed to structure my content logically, making the site accessible and understandable both to users and search engines.

Next, I introduced CSS to style the website. By starting with a mobile-first approach, I focused on creating styles for the smallest screens first, ensuring that the content was readable and navigable on mobile devices. This approach also allowed me to optimize the site's performance on mobile by only loading the essential styles needed for these devices.

For the interactive elements, I used a bit of JavaScript. This included creating a responsive, collapsable menu that would adapt to touch interactions on mobile devices, as well as enabling a smooth-scrolling interaction.

HTML Code & Page Structure

JavaScript & Expanded Mobile Menu

Expanding to Larger Breakpoints

With the mobile layout and functionality in place, I then used CSS media queries to progressively enhance the website for larger screens. I defined my breakpoints as: small (up to 640px), medium (640px and larger), and large (1024px or larger). Utilizing media queries, I was able to achieve a responsive layout that would adjust to take advantage of the additional screen real estate available on tablets and desktops.

While I took some trial and error to get it right, this method of developing from the smallest screen and expanding outwards would ensure that every user, regardless of their device, would have access to a fast, responsive, and well-designed website. It also streamlined the development process, as it is easier to add more features to a base design than to try and scale down a complex desktop site to fit on mobile screens.

Small — Medium — Large

Click here to view my full code on GitHub

Usability Testing

Once my website was appropriately styled and in functioning condition, it was time to use my experience as a UX designer to test it out. The testing phase involved a structured approach to gathering actionable insights from real users. I created test scenarios based on the goals and needs of my target audience, then recruited 5 participants to test out these scenarios and interact with the website across various devices. Scenarios focused on ease of navigation, clarity of content, and overall user satisfaction. Key findings revealed strengths in design simplicity and intuitive layout, while also highlighting areas for improvement, such as enhancing clarity of information and improving functionality of certain features.

After gathering and sorting through all of the session data, I used Nielsen's Design Severity Rating Scale to analyze and prioritize issues to be fixed. High-priority issues were resolved accordingly.

Problem	Rating	Solution
User missed the "download resume" button first time around	1	Consider changing location to home page, however, not imperative as only one user had an issue but then found it right after
"Email me" button doesn't work for users who have not set up their email app	3	Add a contact form on either a separate page or as a modal when user clicks on "email me" button
Some photos in Flame project not great quality; other photos too large	2	Improve photo quality as appropriate; resize photos as needed
Key info doesn't stand out	1	Bold key words in project descriptions and project pages
No way to scroll to top easily in project pages	2	Add a "back to top" floating button on project pages
Users confused about CSS animation	2	Add more context/description about project

Nielsen's Design Severity Rating Scale

0	I don't agree that this is a usability problem at all
1	Cosmetic problem only: need not be fixed unless extra time is available on project
2	Minor usability problem: fixing this should be given low priority
3	Major usability problem: important to fix, so should be given high priority
4	Usability catastrophe: imperative to fix this before product can be released

Deliver

Now that the key issues identified during usability testing were fixed, the next steps were code validation, accessibility testing, and cross-browser testing.

To validate my code, I used HTML, CSS, and JavaScript linters to ensure everything was error-free and aligned with best practice. The linters revealed a few minor issues which I quickly fixed and then re-ran to double check everything. I then checked that my colors aligned with the WCAG AA standards by using a color contrast checker for my background/text color combinations and my button color/text color combinations. Minor changes to my text color and button color were required to ensure adherence to these standards. Lastly, I used BrowserStack to test out my website on a variety of devices and browsers to ensure there were no browser-specific problems or responsiveness issues.

View my portfolio website here!

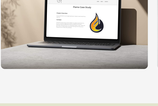
Outcomes & Takeaways

Challenges

Coming from a non-technical background with no coding knowledge, I faced a number of challenges throughout this course. Learning new concepts and ways of thinking was definitely overwhelming at times, but I found my stride by delving into outside resources and pushing myself the course to learn through trial and error. While I've still got a ways to go, this course pushed me beyond my comfort zone and gave me a solid foundation in frontend development skills that I can continue to build upon with additional practice and experience. I'm excited to see what I can now do with this new skillset as I continue to explore and grow.

Learnings

- Learned the basics of HTML, CSS, and JavaScript and how they interact to create a fully functioning, stylized website.
- Developed a structured approach to troubleshooting, debugging, and problem-solving.
- Gained the ability to articulate design concepts in the language of web development, bridging the communication gap between designers and developers.
- Learned how to blend design principles with best coding practices, ensuring a unified and effective user experience.
- Cultivated a mindset for continuous improvement and adaptation to evolving technologies, preparing for collaborative innovation in future projects.



Thank you for reading my case study!