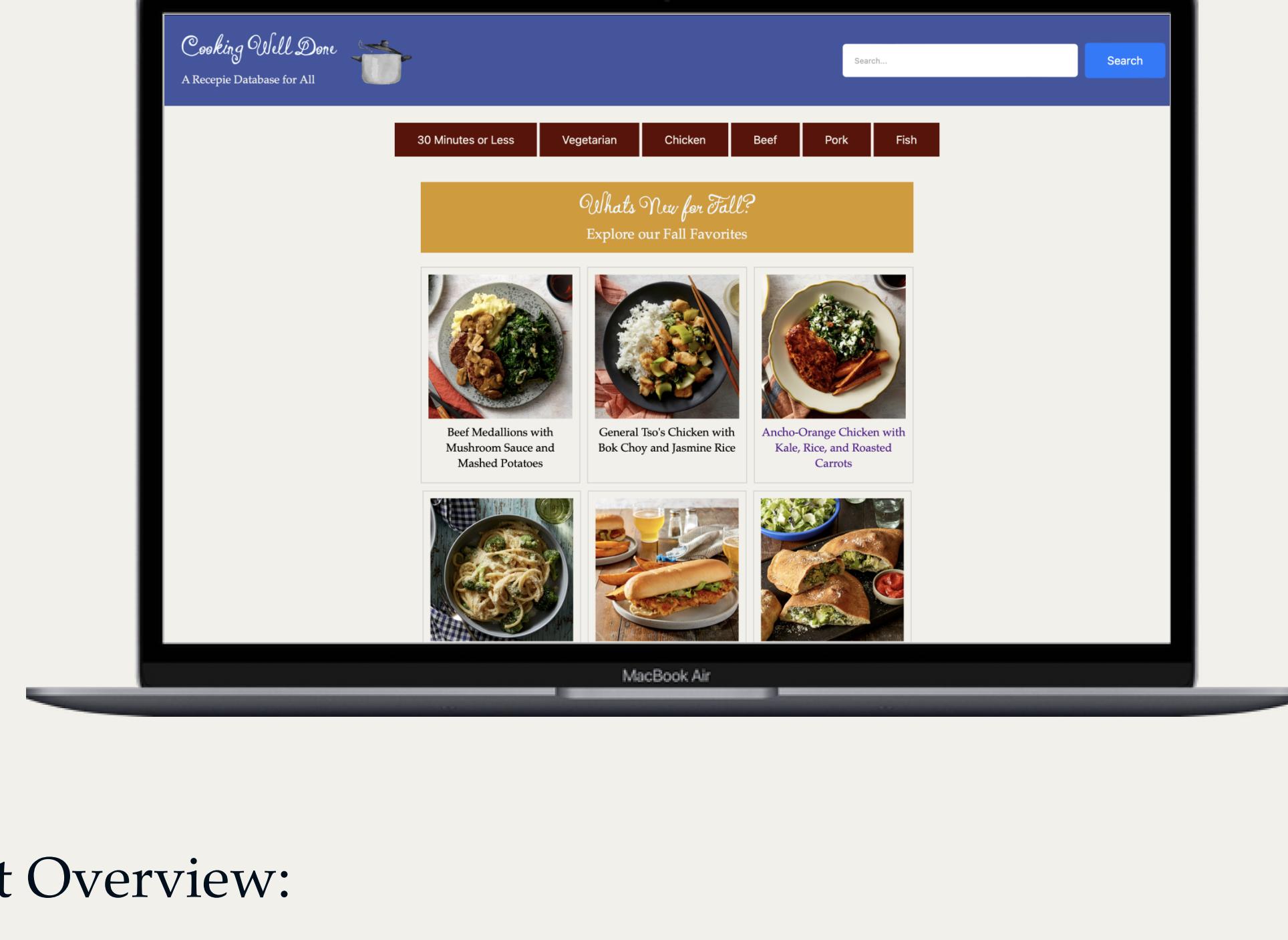


IDM 232: Case Study



Project Overview:

This project involved creating a self-hosted webpage that functioned as a recipe database. We used PHP to pull data from an Excel spreadsheet and import it into a template we had made using HTML and CSS. We had a timeline of 10 weeks to complete the build.

Context and Challenge:

Before starting this project, I did not know how challenging it would be. The first step was developing a concept for our recipe website, including the branding, fonts, colors, etc. The second step was coding it using HTML and CSS. Once the base elements were created, the class was launched into an in-depth crash course in PHP and how to use it for our recipe websites.

For further context: PHP, also known as Hypertext Preprocessor, is a server-side scripting language executing code on the server to produce HTML content for users' browsers. It completely integrates with HTML, creating dynamic and interactive web pages. A critical application of PHP involves database interaction, often in use with MySQL, allowing for the manipulation and retrieval of data.

PHP's Learning Curve:

Learning and applying the PHP language was one of the most challenging things I've encountered since starting the UX program at Drexel, and every week of progression brought its own new set of challenges. I quickly learned that PHP is a very sensitive and finicky language. The slightest error or incorrect format can break your entire page. I had two frustrating encounters with this sensitivity.

The first happened early in the project's development. My self-hosted webpage completely broke, although I had followed along in class and watched videos. After almost an hour of scrutinizing, my professor and I discovered that a missing semicolon was to blame.

This roadblock, however, taught me a valuable skill in debugging and how to correct errors effectively. I utilized the line-by-line approach: Commenting out sections and lines of text to see what broke the page when it was uncommented.

The second detail-oriented roadblock I faced was at the very end of the project when I went to transfer my locally hosted page to an online server. I was shocked when the webpage that had worked locally completely broke when I uploaded it to my server. However, I was able to resolve this issue much more effectively, as my classmate pointed out a difference in the capitalization of the first letter between the file sourced in the code and the actual file name itself. Through commenting-out sections, I quickly learned that my server could not pull my content from the recipe database because of a difference in letter capitalization.

Tools

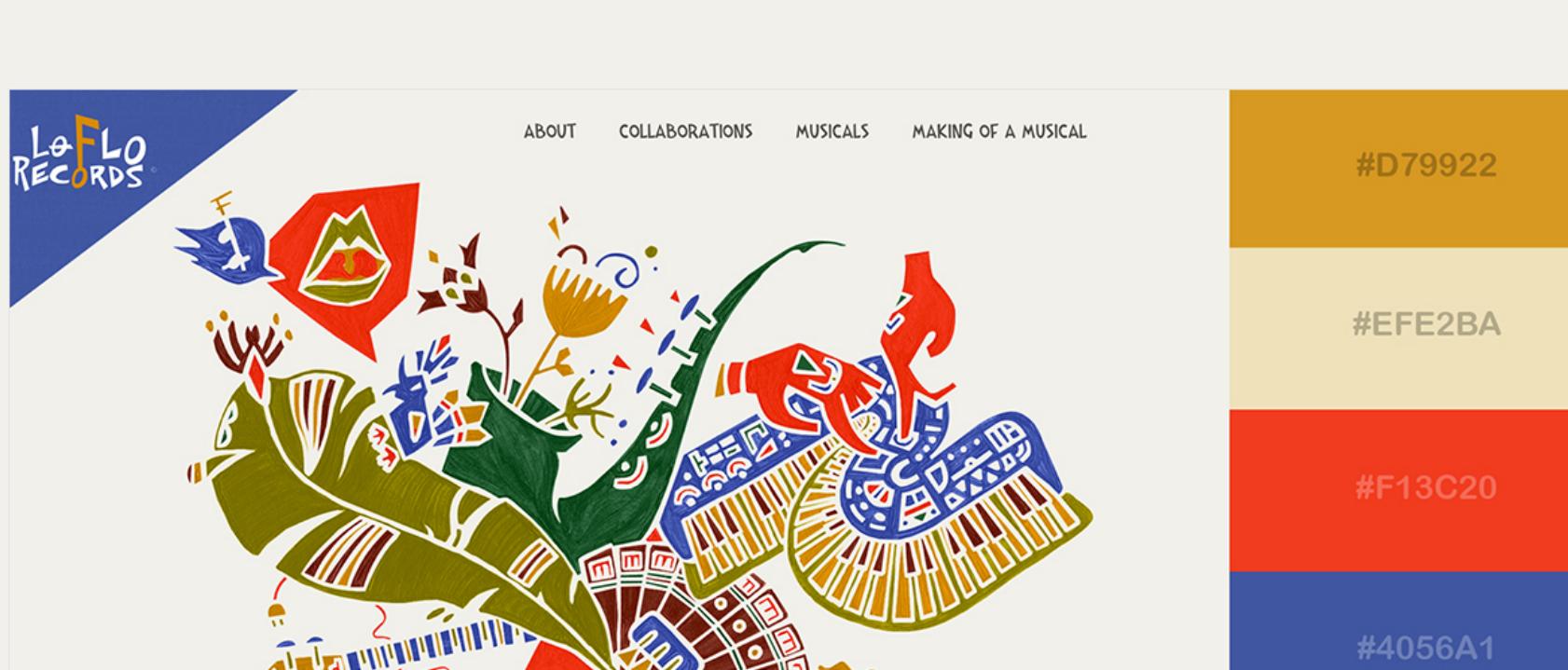
I would be lying if I said the primary tool I used was the internet. My classmate Eli was a huge help along the way in helping me implement and debug my code. The main way my classmates and I could move forward each week was through meeting and collaborating. Chat GPT was some help, but certainly not copy and paste. Even though PHP is extremely sensitive, like any other language, you can format elements to receive the same outcome in several ways.

Utilizing Chat GPT

After identifying the target microinteraction for my project and its elements, the next hurdle was applying my HTML and CSS knowledge from prior terms to code the original microinteraction. It was a definite forced refresher. One of my most frequently used resources was Chat GPT, which I had never used for coding prior to this class. At first, I rejected the notion of using AI for code; I wrongly assumed that using it wouldn't help me learn the content by generating it for me. However, I realized it was time to look for alternatives when I found myself four YouTube videos and six website tutorials deep- all consisting of completely different and convoluted ways to make flexboxes the same length regardless of content. Once I utilized chat GPT, my problem was solved within minutes, and I learned it was a powerful tool that could be used and built upon to suit my needs.

The Final Product:

Overall, I was happy with the final deliverable for this project. My final project consisted of a functioning index page that pulled all the recipe images from the database, a search bar that successfully filtered recipes based on keywords, filters that used keywords as a secondary way to sort recipes, and lastly, individual detail pages for each recipe that formatted the instructions equally.



Reflections

Although the website is fully functioning, down the line, I would like to go back and change the CSS to make the style look more professional. I wanted a European artistic aesthetic with bright pops of color that fell short in my final delivery. :



This was my initial inspiration. My goal is to do more styling to bring my vision fully into fruition (without breaking the PHP).

Lastly, my final thought on using PHP is that it's a very effective tool for creating multi-page, content-heavy websites. I need to work on my skills, but I look forward to bringing the knowledge gained from this project into my future endeavors.

Leading Up: The Final Build

Although I had improved and added upon the fundamental aspects of the microinteraction, I still had challenges to overcome before the final build was complete. Another major roadblock I faced was still associated with the color change transition on the button. Although the text had appeared, it was positioned at the top of the button container, and I couldn't get the text to center through traditional methods. I tried to justify and align the content, text, items to the middle but nothing worked. Chat GPT didn't entirely understand the intricacy of my problem and gave me a default answer that was unresponsive in the code. Finally, after days of scrutinizing, I realized all I needed was top-padding to center. It was the simplest fix, but it reminded me that when problems seem simple, the answer usually isn't very complicated.