

Website: https://www.jessicaxxiao.com/PUI/homework_5/

Github Repo: <https://github.com/jessicaxiao11/PUI>

Heuristic Evaluation

Bug #1: The first bug was the lack of an alternative option in the first quiz question. In situations where users didn't want to choose either cat or dog, or if they wanted to view all products, an alternative option would allow them to do so. My solution was to include a "see all" option at the bottom of the quiz question. This aligns with the heuristic: **flexibility and efficiency of use**. It also provides more **consistency** to the site, since the second quiz question also had a "see all" option.



Before

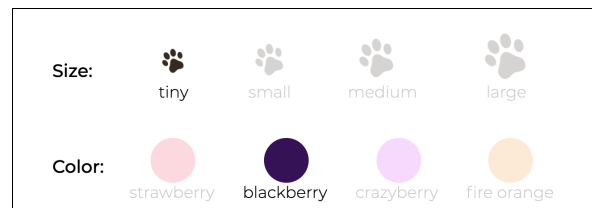


After

Bug #2: The second bug was that the size and color options on the product detail page looked static, but they were actually actions that users needed to perform. As a fix, I added a default size and color that were pre-selected. This was to signal to the users that the options were meant to be selected, and it was an action they needed to perform. This aligns with the heuristic: **recognition rather than recall**. In this case, the fix is much less about recall (although they do need to "recall" to select a color/size), and much more about recognition.



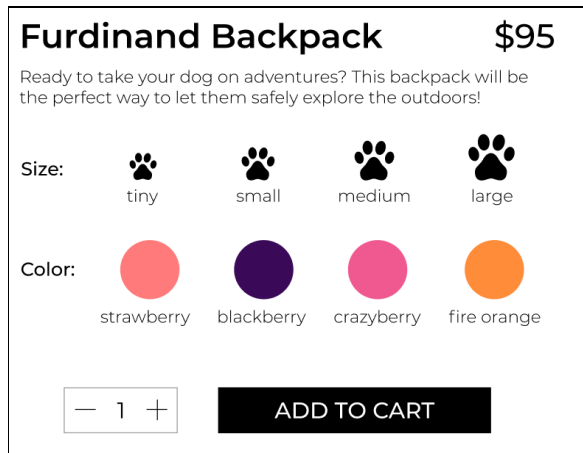
Before



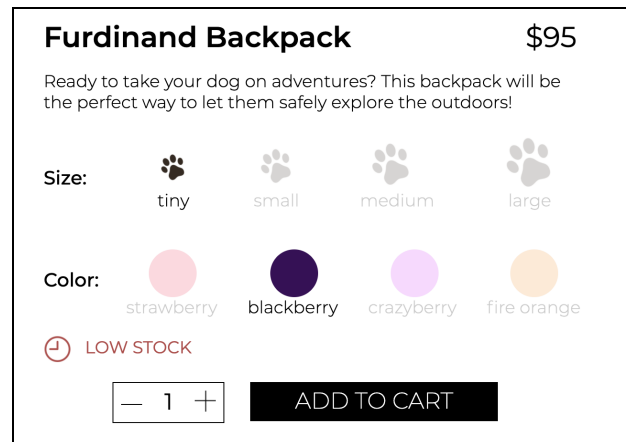
After

Bug #3: The last bug was that the interface didn't really communicate to the user about the number of items that are in stock for a product. If they aren't aware of the current status, they may either lose out on getting the product, or be confused why their product isn't arriving on time (after ordering it). To fix this, I added a low stock indicator if a size/color combination was

low in stock. This aligns with the heuristic: **visibility of system status**. It also helps to prevent, diagnose, and recover from errors, in the situation where they order something out of stock and the product takes an abnormally long time to ship.



Before



After

Implementation Challenges:

Challenge #1: One of the main challenges throughout the assignment was accurately positioning elements in the correct place. When I first started, I had to recall my knowledge about the display and position properties. From previous experiences, I had always struggled with positioning my elements, and never found a solution that was always viable. It also seems like there are *many* methods to positioning elements, including flexbox, which meant that there was no standard way to do so. After trying out multiple solutions, I found that using “position: absolute” and “position: relative”, then manually setting the top/left properties, was likely the most viable option. I stuck with this method for the rest of the assignment, but in effect, many of my elements were individually, manually positioned and this generated a long CSS document. I did as much as I could to group the elements (using tables, classes, etc) to eliminate repetitive CSS, but in the future, I hope to discover a more efficient method of positioning!

Challenge #2: From my Figma prototype, I had planned to use a mouse hover on the hamburger menu icon to reveal my actual menu. On the product browsing page, the sectional menu would also appear when the mouse hit the left of the screen. These were both interactive elements that *necessarily* needed to be implemented, in order for my site to be navigated. When trying to implement them, I first tried to rely solely on CSS and it's hover properties to display the menu. However, I soon discovered that the CSS combinators were quite restrictive, and I wasn't able to select my menu element in combination with the hamburger menu icon. Due to these restrictions, I relied on simple JavaScript to show/hide the menus. Implementing the JavaScript was pretty straightforward, but I certainly discovered the restrictiveness of CSS for interactive elements.

Challenge #3: I struggled to implement the styling of some of the more complex elements, including the full page background on the home page, and the full-page menu overlay. However, I greatly relied on w3schools and took advice from their implementations to create my own elements.

Brand Identity:

In comparison to my Figma prototype, not much of the design has changed. Muddy Paws places an emphasis on creating good-quality adventure wear for adventurers' pets. Because of this context, I think both Muddy Paws and their audience might appreciate minimalism and simplicity. Given their love for the outdoors, the scenic images and generally neutral colors will hopefully align with their identity.

Simplicity is incorporated in both the static design and the interactions. The cohesive, straightforward structure of the site (especially the browsing page) creates for a simple design. The pages also contain clean lines, black/white colors, plenty of whitespace. The menus that open on hover also create sleek and delightful interactions.

In this setting, I think that Muddy Paws' identity is quite similar to the audience they're hoping to appeal to: down-to-earth adventurers who hope to include their pets on their trips. Both audiences are down-to-earth, and our minimalist, earthy-toned website will hopefully embody that as well.