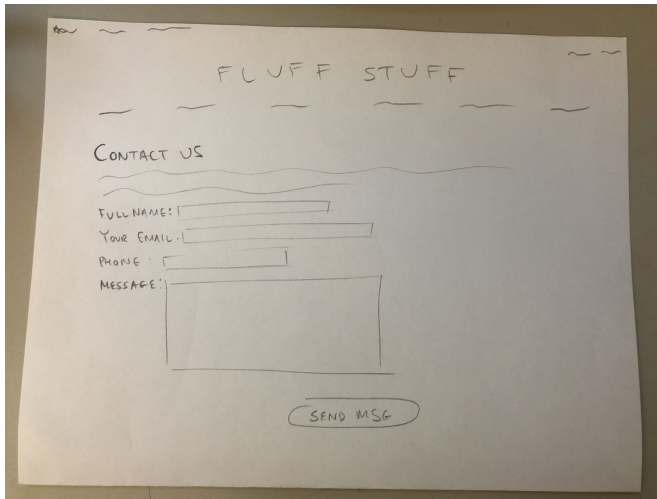


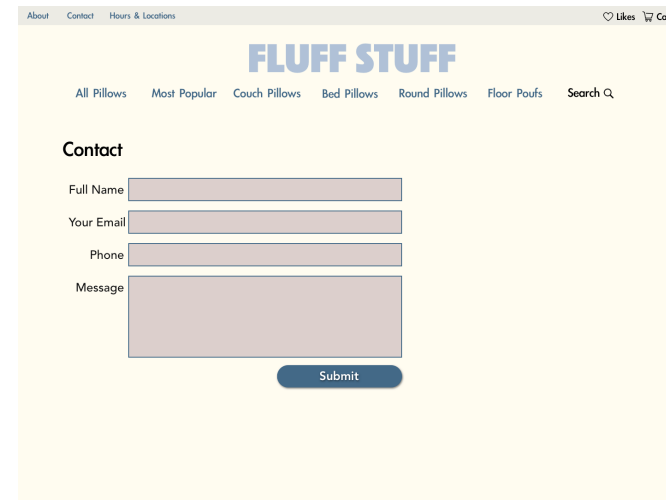
PUI Assignment 6A: Adding Functionality to a Website with Javascript ([Fluff Stuff](#))

PART I: New Page Prototypes

Contact Page - Low-Fidelity Prototype



Contact Page - High-Fidelity Prototype



I already created a cart page in [my original high-fi prototype](#), so I created a Contact (form) page for this assignment.

PART II: Implementation Challenges

Because I had virtually no programming experience, I encountered many challenges, or “bugs,” during this assignment. Overall, I credit my peers, W3schools.com, stackoverflow.com, and YouTube, for helping me understand and overcome issues. For example, I spent hours(!) trying to fit content into a specific layout using a CSS Grid. I watched various YouTube videos and referenced many web sources, but my grid still just wouldn’t behave as I wanted it to. So I asked my peers, who suggested using Flexboxes, which were supposedly simple and easy to use. I heeded their advice and switched from Grids to Flexboxes, which I found very straightforward and effective, to my relief.

Undesirable grouping, and/or *lack* of desirable grouping, was also a consistent issue for me. For example, in my Product Browsing page flexbox, I expected product names & prices to appear below images, but instead, they were placed adjacent to their respective images. This resulted in an extremely unpleasant aesthetic for my website, and panic for me. I tried to fix this by adjusting float, alignment, and more, but nothing worked until I asked a former software engineer peer for help. He explained that grouping elements together in one class communicates my need for visual grouping to CSS. This was revolutionary -- I applied this concept successfully throughout the rest of my code. Classes in classes in classes.

Coding buttons properly was another lesson I learned the hard way. In my Product Details Page, I opted to create text hyperlinks for the color and filling options. I placed bordered boxes around them to mimic the buttons in my prototype. However, this resulted in only the text area being clickable when I actually wanted the entire object (box, border, and inner text) to be selectable. Later, when I coded the “Add to Cart” and “Add to Likes” buttons, I used the “button” class/attribute, which achieved everything that my previous button attempt did not.

Overall, placement and alignment of elements was tedious and confusing. It seemed that HTML and/or CSS used arbitrary, nonsensical rules when deciding on element placement. Generally, I used padding and margins to reorient elements to their desired locations; however, I would like to improve my understanding of controlling placement/alignment with clean code, correctly, from the start.

PART III: Brand Identity via Design Choices

The Fluff Stuff brand identity reflects the casual comfort, softness, and soothing effect of its product: pillows. The design choices of this website were carefully determined to reflect this identity by embodying calmness and comfort.

- **Bubbly font** was used throughout the site to reflect the bulbous, cloudlike, fluffy characteristics of pillows. A “fluffy” font, it also reinforced the cuteness of the name “Fluff Stuff.”
- **Pastel / Soft / Pale colors**, such as very pale yellow (background), teal (font) + buttons, and gray (font), were used to create a comfortable, soothing, and gentle experience, just like the experience of Fluff Stuff pillows.
- **Minimalism** was the overarching aesthetic of the website. Icons and fonts were without frills. Left and right margins were spacious. Shapes were clean. Again, this exemplifies a calming and soothing vibe. Were the website not minimalist, it would have conveyed a sense of calamity, busy-ness, and stress antithetical to the comforting nature of Fluff Stuff pillows.