TECHNICAL UNIVERSITY OF MOLDOVA

FACULTY OF COMPUTERS, INFORMATICS AND MICROELECTRONICS

DEPARTMENT OF SOFTWARE ENGINEERING AND AUTOMATION

WEB PROGRAMMING

LABORATORY WORK #1

Learn CSS and HTML

Author:
Liviu MOCANU
std. gr. FAF-203

Supervisor:
Alexei ȘERȘUN

Chișinău 2023

1 Task

Our task for this laboratory work was to:

1. Download the design provided as a PSD file.

2. Extract the images as JPG files.

3. Copy 'index.html', 'reset.css' and 'style.css' to our repository.

4. Update 'index.html' and 'style.css', so that the page looks similar to the design from the PSD file.

We were not allowed to use any third-party styles, except some utilities (which should be argumented).

2 Results

All four required sections for the full grading have been added with an additional footer section (Figure 1) to provide a more concise and intuitive closing to the project, alas not with an adapted design for the mobile devices:

Figure 1: Additional Footer Section.

Another small but eye-catching feature that was added to the "Groceries. On the go" page was the addition of multiple background images, however only one of which (the zigzag on the ride side of the screen) stayed fixated on the page despite the scroll. This gave the page a dynamic feeling of multi layeredness.

Unfortunately it can't be showcased as a figure, however the snippet of code in Listing 1 is responsible for it:

```
.main{
    background-image: url(images/phone.png), url(images/zigzag.png), url(images/
    background-main.jpg);
    min-height: 100vh;
    width: 100%;
    background-position: 75.5% 60%, right, center;
    background-repeat: no-repeat, no-repeat, repeat;
    background-size: 17.5% 68%, 100% 100%, cover;
    position: relative;
    background-attachment: scroll, fixed, scroll;
}
```

Listing 1: Fixating A Background Image.

Since it was one of the first times engaging in such a big project based solely on HTML and CSS, even the ability to change certain aspects of a button while hovering over it felt like a welcome addition.

Figure 2 shows an example of changing the colour of a button's background to it's font's colour and vice versa as well as changing the font's size to a slightly bigger one:

(a) Button in the default state
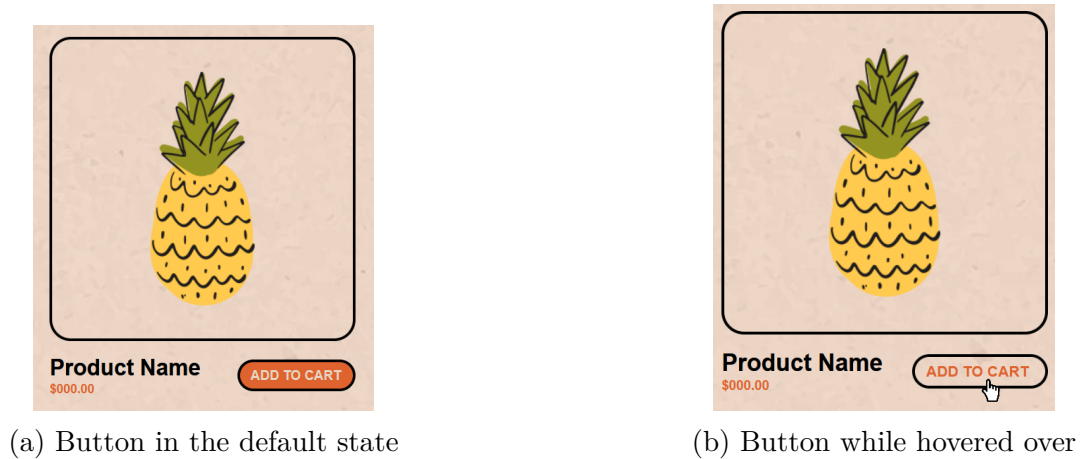

(b) Button while hovered over

Figure 2: Button changing aspect on hover

Unfortunately the font size change is barely visible in the example, but these are more palpable in a demo.

The changes take over a transition of 0.5s and in a similar manner, so does the underline for the elements in the navigation bar for the main page. Initially there is nothing special in their appearance, but on hover, a small centered underline appear beneath them as shown in Figure 3

(a) Nav element in the default state

(b) Nav element while hovered over

Figure 3: Button changing aspect on hover

The effect for the nav elements was reached with the code snippets from Listing 2:

```css
.nav-links ul li::after{
    content: '';
    width: 0%;
    height: 2px;
    background: #DF632F;
    display: block;
    margin: auto;
    transition: 0.5s;
}
```

Listing 2: Code snippet for the transition effect and underline showcased for nav elements on hover

And similarly we have the snippet in Listing 3 for the buttons and more specifically in this case, the "learn more" button:

```css
#learn-more-btn:hover{
    border: 3.6px solid black;
    font-size: 19px;
    background: #DF632F;
    color: whitesmoke;
    transition: 0.5s;
}
```

Listing 3: Code snippet to achieve the transition effect and underline showcased for buttons on hover

All the elements that should act as a button (the arrow pointing down at the bottom of the "Groceries. On the go" page, nav elements, various buttons) do in fact, however the majority of them have an empty reference since they are needed for logic irrelevant for this laboratory work (the left and right arrows in the shopping section should show more products), other than the navigation elements and the above-mentioned arrow that sends the user to the next category:

```html
<div class="nav-links">
    <ul>
      <li><a href="#main-page">Home</a></li>
      <li><a href="#locally-sourced-products">About</a></li>
      <li><a href="#shop">Shop</a></li>
      <li><a href="#contact">Contact Us</a></li>
    </ul>
</div>

<a href="#locally-sourced-products"><img src="images/arrow.png" id="arrow-down"></a>
```

Listing 4: Code snippet showcasing anchor using for nav elements and arrow-down

Unfortunately the effect of this can yet again not be showcased, only palpable in a demo.

Lastly, a small detail regarding the implementation, the products for the "Made simple" page are not hard-coded, instead flex was used. For reference, we can see them in Figure 4:

Figure 4: The products section of "Made Simple" page

In order to achieve the result in the aforementioned figure I used the code snippet presented in Listing 5:

```
1  .shopping > div > img{
2      border: 3.5px solid black;
3      border-radius: 25px;
4      background: transparent;
5      text-decoration: none;
6  }
7
8  .product{
9      margin-right: 3%;
10 }
11
12 .product-description{
13     font-weight: bold;
14     margin-bottom: 0;
15     display: flex;
16     justify-content: space-between;
17 }
```

Listing 5: Code snippet to achieve the transition effect and underline showcased for buttons on hover

3 Conclusion

After implementing the task, I've learned that replicating a design by using HTML and CSS can be a challenging but rewarding task. Through this laboratory work, I was able to gain valuable experience in working with HTML tags, CSS properties, and different techniques such as flexboxes to create a visually appealing design.

I started by analyzing the design, breaking it down into smaller components, and then creating the HTML structure using appropriate tags. Afterwards, I added CSS styling to the HTML elements, using different properties such as color, font, margin, padding, and border to achieve the desired look and feel.

Moreover, I learned how to add hover effects to buttons, which can enhance user experience and make the design more interactive by using CSS pseudo-classes such as :hover and changing the appearance of buttons when the user interacts with them.

Overall, this laboratory work provided me with a solid foundation in HTML and CSS, which are essential skills for anyone interested in web development.

References

[1] Front-end web developer, *https://developer.mozilla.org/en-US/docs/Learn/Front-end_web_developer*

Accessed on February 10, 2023.

[2] Tips and Tricks with CSS Background Position, *https://www.udacity.com/blog/2021/04/tips-and-tricks-with-css-background-position.html*

Accessed on February 17, 2023 .

[3] How to align 3 divs inside another div?, *https://stackoverflow.com/questions/2603700/how-to-align-3-divs-left-center-right-inside-another-div*

Accessed on February 18, 2023 .