

Problems & Improvements

Problem: Old page had a fixed width, smaller than a landscape screen, but not quite fit for mobile.

Improvement: Made the widths and heights relative in the page, with some minimums, that adapt to the user's screen.

Problem: No quick access to MarkUs or the Assignment's contents. Can only do so through clicking on the assignments page first, and then the link.

Improvement: The navigation menu has direct links to MarkUs and directly links to the specific assignments when hovering over the assignments option.

Problem: No dark mode for reading in the dark or at night.

Improvement: Created a dark mode of the page, along with a way to switch.

Problem: No course team page to find the TA contact information; only the instructor's.

Improvement: A course team page was created to display basic information about the course instructor and TAs.

Challenges

1. A large problem was designing the navigation bar to be: useful buttons, stick to the top of the page, and display a dropdown menu from the position it is, at any point from the page, on top of working and looking good on mobile. The way it was solved was by dividing it into 3 parts: navbar, dropdown and mobile.

In the navbar, the best option was just a flex div that sticks to the top of the page, with all the contents in it, links, justified to spread out.

For the dropdown, it would take the position of the navbar and add a dropdown menu div that is absolute (relative to the viewer) below the option, using a div that contains that option for alignment.

The mobile menu was the hardest part, since it involved the use of media query. To switch between, the navbar for desktop is set to display:none to hide it, and a hidden menu, the mobile one, is displayed instead, which is triggered below an aspect ratio and below a width. Both menus do exist, but media query switches which is displayed, and which isn't. Given the complexity of the navbar, the mobile menu was just set to be a simpler dropdown that shows when hovering over the menu icon. We were unfortunately unable to find another way that didn't involve JS to do so reliably.

2. Switching between light and dark mode. Given don't have any templates, the simplest way to "switch" between modes was to create a clone of all the pages, but switch the CSS colour variables, then, the button in the footer is a link to that page's other mode. In order to not have to update the nav menu, which would create confusion, the copy of the pages was put in a different folder, with all the same files, save for the CSS and having the dark and light mode of images such as the UofT logo.
3. One of the first concerns when creating the site was having a consistent palette for the site, with the flexibility of being able to change it for dark mode. To deal with it, we first picked 4 colours from the 2023 UofT colour guide for the light mode, and 4 that would be the opposite for the dark mode. The colours had to be in a sort of "scale" from darker to brighter, and 1 colour to do the highlights. Once the colours were picked, we needed a way to consistently implement it, which we used the variables in :root() for. This allowed us to create basic colours, and for whenever we needed, use the variables so that we could easily switch later, though because of not thinking of the name at first, "UofTlightblue" ended up as a dark blue in dark mode.

We applied the same methodology, creating global components in the CSS at first, that would apply to objects like h1, h2, h3 and p, and then being more specific whenever it was needed.