

CSCB20 Assignment 2 Report

Yi Ning He 1006308979
Jeffrey Wong 1006300897

Issues and Solutions

The previous website had some visible problems. For one, the link to news is non-existent, and its link only appears on certain pages, meaning it was meant to be removed but it was not updated on other pages. The discussion board link also disappears when you enter the links page, and the calendar link disappears when you click on the test link. The test link also happens to be empty, which is a problem by itself. While this could mean there are no upcoming tests yet, the user doesn't know that and it leaves the user guessing what's going on. The website's format is also a little old. The website's home page offers a course description on its home page, which is useful for those who want to look at the website the first time, but for returning students such a description is unnecessary as the first thing to look at. In order to address these issues, instead of a news page we have a to-do list on the home page. This to-do list contains information for the next assignment or homework that the student has to do. We also improved the offices hours to mention the next office hour rather than just the general times.

The tables used to display the lectures and tutorials. A table is an effective way of presenting the information, but the tables are very generic and lead to a bad experience using them. There is no clear differentiation between the column headings and the contents of the table. The border is generic and not colorful. By being monochromatic, the table doesn't create a positive appeal towards students visiting the website. To address this problem, we redesigned the looks of the table using CSS grid. Column headers have a solid fill so they stand out. The border has a color that matches the column header. The rows alternate between white background and a slightly grey background so the viewer can better distinguish the rows. As a whole, the table is not more visually appealing and relaxing to use.

On the current CSCB63 website, there are many links to lecture slide files, lecture recordings, and to other websites used for the course such as Piazza. All the links on the CSCB63 website have the problem: they always open the file or webpage on the same tab in a browser. This is very frustrating because a student usually clicks on multiple links, such as watching the lecture recording and reading the lecture worksheet at the same time. It is more convenient to simply click on the link and have the file or website open in a new tab, so the student can go back to the course website to click on other links. Our solution is to add the "target" attribute to every <a> tag and set its value to "blank", which will open the file or webpage in a new tab of the browser.

Challenges

One of the biggest challenges was adapting to the format given in the assignment. Such as being unable to use any third party add-ons like Flask to help with header and footers. One example is dealing with separate CSS pages for each page. Sometimes one page's CSS page would overtake some styling of another page. The most frequent bug would often involve the header and footer style being overtaken by the main page styling. We ended up implementing a class system which involved using the class tool to place the chunks of code which use a certain style into a container, and name elements such as header 1 with a class tag in CSS (e.g. .container header1{}) to easily separate styles from each other.

Another challenge was redesigning the header (as well as the footer) to fit the criteria from the assignment. We wanted a header which would contain all of the important student links to appear first, so a student could easily navigate to their current work. Since we couldn't use tables and we didn't want a vertical link bar like in the original CSCB63 website, we used grids to create boxes to contain our links. This design was efficient and clean, and we could change it to add more links in the bar easily if we wanted to. The footer itself wasn't bad either, by using a menu system we could make a footer that was similar to well-known websites like cibc.com.

Another challenge is: our team is new to developing using HTML and CSS. By not knowing what HTML and CSS is capable of, we are limited by the incomplete knowledge about the tools that are available. This led to frequent redesigning of our website. To overcome this, we watched lecture videos, read resources online such as *w3schools*, and watched YouTube tutorials. Through the resources, we discovered the right tools to use to build our website.

A specific coding challenge is the banning of the `<table>` tag. Developing a table without the `<table>` tag is much more inconvenient. The solution is to use divisions to house each cell of a table, then using CSS Grid, the divisions are displayed side by side, giving the illusion of a table. By cleverly setting classes, the cells are divided and different styling is applied. In the end, the tables are a success.