**UW Course Tracker Progress Log**

**First started on February 19th, 2023**

I started this on March 16th, so I might have skipped some of my earlier steps.

**Initial Knowledge:**

**Languages:**

Some OOP in **Java**, **Racket** ig, CS136 level **C** up to loops :skull:, scuffed **python** (I did some of it in a previous hackathon), **CSS/HTML** (not on a professional compatible with level, just a website creation level), like 0.1% **JavaScript**

**Other Knowledge:**

* 0 Knowledge about front-end user interaction, 0% react, 0% website development
* Tiny bit knowledge about webscraping, but pretty much nothing

**NOTHING PRETTY MUCH‼!**

**Creation (Feb 19th):**

My initial plan was to just scrape uwflow, and to display it all on a graph with some info on the side (funny simple plan).

* First created website with react (literally typing the create command, idr what it is), didn’t make any front end yet
* Spend around 3 full days during reading week going through the ‘react tutorials’ beta, learning how to use React (kind of skipped the last section about interaction with react with non-react stuff)
* Initially thought React would come with a bunch of good looking templates, but it turns out it was just same as normal CSS/html coding

Initial Designs:

Text

Description automatically generated with medium confidence Diagram

Description automatically generated

On the left, it was what I wanted the info displayed to look like. Looking back, this is pretty similar to my current UI. I literally made this in apple notes on the car on my phone.

On the right, it was a bit more advanced. I drew this on MS Paint with trackpad :D.

**Starting Web Scraping (Feb 20th):**

The original idea was to go to (<https://uwflow.com/explore>) and use Python requests and beautiful soup to web scrape all the content. However there came some issues:

**Issue with Web Scraping Dynamic Pages:**

**Problem:**

The issue was that UW Flow dynamically loads the pages. So for the first 0.5 seconds, it shows the default screen (and python requests only takes that info), then the rest of the contents load

**Solution:**

In order to circumvent that, I decided to use another web scraping package, which was called Selenium. This pretty much acted as a virtual browser (headless, so it doesn’t actually appear). So it loaded the page for 1 second, then it scraped the info.

**Issue with UW Flow info displaying:**

**Problem:**

Now, we are able to web scrape the info, however, UW Flow explore page only shows 50 courses at a time, and there are like 8000 waterloo courses.

**Solution:**

Luckily with selenium, since it acts like a user, I made it manually scroll down to the bottom of the page (with a 0.7s delay in each scroll so it gives time for the page to load the next 50 courses). Obviously, this would take a long time (I think 10 minutes), and I could take the shortcut and just only scrape the top few thousand courses (since the last like 4000 courses literally had 0 ratings), BUT I GOTTA KEEP IT ACCURATE‼! Now that I’m tying it out, that was a dumbass move, cuz that made my program so god damn slow (it took like 1 hour).

In the end, I managed to create a webscraping program just on python. However, it took around 1 hour for it to load with 68k lines (last time I checked, as in I left and when I came back, around an hour of continuously checking, it finished), SO OBVIOUSLY I WASN’T GOING TO KEEP THIS RUNNING. If you want to see the file, check datafull.js.

**Starting To Create Website (Feb 20th – 25th):**

This was when reading week was nearing to end, so I was speedrunning this. For this part, I’ll split it up into the most painful sections

**Creating the Overhang:**

I will most likely remove this in the future, so I’ll add an image of what it looks like right now.

A picture containing graphical user interface

Description automatically generated

THIS WAS A HUGE PAIN IN THE ASS, IM TELLING YOU. EVERY SINGLE KNOWLEDGE I HAVE ABOUT HTML AND CSS WAS ALL SELF SEARCHED UP. I DID NOT FOLLOW ANY TUTORIALS OR ANYTHING I HAVE NO FORMAL OR PROPER KNOWLEDGE. DOING THIS TOOK ME THE WHOLE GOD DAMN NIGHT, OVER 6 FREAKING HOURS. ALSO DOING THIS IN JSX WAS PAIN.

So initially, if you check my previous prototype (the one on the right), I wanted the “Designed By: Kevin Lau” on the overhang part, however, it obviously wouldn’t fit, so I decided to make it different. Looking back, this looks better?

OK BUT NOW IN THE PAIN ON THE ASS PART. First pain in the ass, the curved cut-offs. The rounded edges. This itself was all manually done. I had pirated photoshop on my old laptop, but I’m not taking out that brick just for a dumb image (I should have). So I used a crappy web version of photoshop (literally photoshop ui but less features). Doing those images alone took me like 1-2 hours (including all the changing to fit with the web and stuff). Again, there might have been some actual css shortcut, but idk it, so I used my own methods.

So I initially just had 1 image, the whole overhang with the empty part being empty. However, aligning the title (UW course tracker), and the info below that was really difficult with matching and fitting it right on the edge of the ‘empty spot’, because of different screen resolutions and sizes (zooming in and out would ruin it).

So I took a second approach, which is the current one. I have 2 images, one which is a full rectangle which doesn’t include the ‘overhang’ part, and the second one is just the ‘overhang part’. This worked much better, and by the end of the night, I just gave up and settled for this.

**Implementing The Search Bar**

Ok after that painful header, this was the fun part where I finally could put my React knowledge to use. This was also my first time using input and using React actually. Below is what it looks like as of March 16th, 2023.

Graphical user interface, text, application

Description automatically generated

So first, the default input box looked so god damn ugly, so it took me some time to restyle it (and removing the default x button).

**List of Subjects and Their Faculty:**

This was the dumbest thing I’ve done yet for this project. There was no list of all the waterloo courses except for this specific website I can’t find (I just spent 10 minutes trying to link it, I can’t find it, I think it’s under the undergrad calendar). So I literally just copied and pasted like all 126 course SUBJECTS (not codes) into a .js file with their name and their faculty. This took me like a good hour, and this was by far the dumbest thing I could have done yet. If you’re curious, look at courselist.js file.

**Creating the Top 5 Closest Subjects:**

So I used a function, so when the user changes the text within the input, it updates the state that holds the text inside the input box. This was my first time using React, and when it worked, I WAS LIKE BIG W’S.

The most complicated part should have been implementing this search, but THANKS TO AWESOME PACKAGES, I used a certain fuzzy search package that helps me filter the most relevant courses based on the input and the data I gave (I used the subject code and their full name, ie. Math and Mathematics).

Then I used a mapping function to help display the top 5 courses, it was pretty easy and simple in comparison to what it looks like.

**Button Formatting:**

In fact, the harder part was making it so the buttons were color coded to their following faculty. So I had 2 colors, a main color, and it’s light color (for various uses). Luckily, Waterloo for some reason has actual official colors, so I stole them, and had to make a few for other faculties (REN, VPA, WLU). I might have done some very sacrilegious css coding, but then again, I never actually properly learned it, so if it works, it works (hopefully). What I did was I set the class of the button LITERALLY AS THE VARIABLE THAT HOLDS THE FACULTY NAME. And in my corresponding css file, I just put the button properties there. This took a while, especially the button color faculty formatting and finding out how to do it.

**Filtering The Relevant Courses:**

This is a pretty short but important part. So once I had the correct course subject selected, what I did next was actually filter the content. So I got react to go to the huge 68k line file, and filter the courses with the same subject code.

This is a bit more tricky than it seems. Because I had to compare using the length of the subject code with the substring of the list of course codes. And if I had a subject code “PD”, it would also include courses that started with “PDPHRM”, so I just filtered for “PD “ instead, which somehow worked first try!

**Error Message (irrelevant):**

This was very irrelevant, but adding that invalid course subject was very painful. It was my first time playing with the ?: operator in html was pretty interesting, and was cool.

Graphical user interface, text, application

Description automatically generated

**Deploying Website (Feb 25th):**

This was a pretty important, but also short section. I decided to deploy it under github pages, and it was pretty easy. Editing the packages so it was deployable to github pages took a ton of work (I was missing like 1 line of code that gave permissions for people to view).

**Adding Chart (Feb 25th – Feb 26th):**

This was adding the charts so that the info could actually be digested and presented. This is what it looks like (as of March 16th):

Chart, scatter chart

Description automatically generated

**Finding a Cool Chart Grapher:**

Chart, scatter chart

Description automatically generatedWhen I first started, I tried for like a good 6 hours trying to make my graph on NIVO scatter plots, because it looked really cool. However, NIVO ISNT UPDATED WITH REACT 9, SO LIKE THEIR PACKAGES NEVER LOADED, WHICH MADE ME VERY SAD.

However, I decided to settle with ReCharts, and in hindsight, I think it looks better than if I used nivo! Learning how to do the axis and playing around with the style, making the graph fit with the most recent color scheme, etc. The most painful part was styling the tooltip (example on the right →), because I had to learn how these charts worked inside out to manually style and add info that wasn’t relevant to the graph.

Right now, it is currently 9:41am, 2023-03-30, and I’ve spent the last 6 hours adding a little indicator on my graph when you select onto it. I also added a search bar to search for courses. Even though I have a SPCOM100 journal due on 2.3 hours, I REGRET NOTHING (I really do regret not working on it).