# UML Class Diagram:

A screenshot of a cell phone

Description automatically generated

# Plan: (what?)

The goal of iteration four is to slightly refactor and add lots of pretty css to the vue web app.

## Estimated time for each task:

|  |  |  |
| --- | --- | --- |
| Task: | Estimated time: | Actual time: |
| Class diagram | 15mins | 10mins |
| Refactor getElectorateData function | 20mins | 10mins |
| Add css to button | 30mins | 1hour 30 mins |
| Add background image and css | 30mins | 1hour |
| Frosted glass panel div | 30mins | 1hour |
| Other general css | 30 mins | 30 mins |
| Formatting json | 1hour | 1hour 30mins |
| Unit testing | 1 hour | 4 Hours!!!! |

# Unit Tests:

A screenshot of a cell phone

Description automatically generated

The unit tests were an absolute nightmare this time around again. Even though I used the more popular way to test vue apps, Jest, I found learning the API of both the VUE test utils and JEST to be really really confusing. I ran into the same issues where when my functions were being called, they were throwing errors that they do not throw when used normally on the web server.

# 

# Wireframe Diagram:

A close up of a device

Description automatically generated

# Evaluation:

## Style report:

Using standard.js with prettier we have no problems.

A screenshot of a cell phone

Description automatically generated

# What happened vs what was planned:

I am very happy with how the app looks and functions, creating the css was extremely fun. Unit testing the app however, was a nightmare. I am finding the vue test utils to be **extremely** confusing and increasingly completely useless. What is the point of unit testing your app if you have to mock every function and pretend it works instead of actually testing the function itself? I don’t understand the point of unit testing a web app in this fashion. I also can’t debug it because node can’t do it for whatever reason! Lovely!