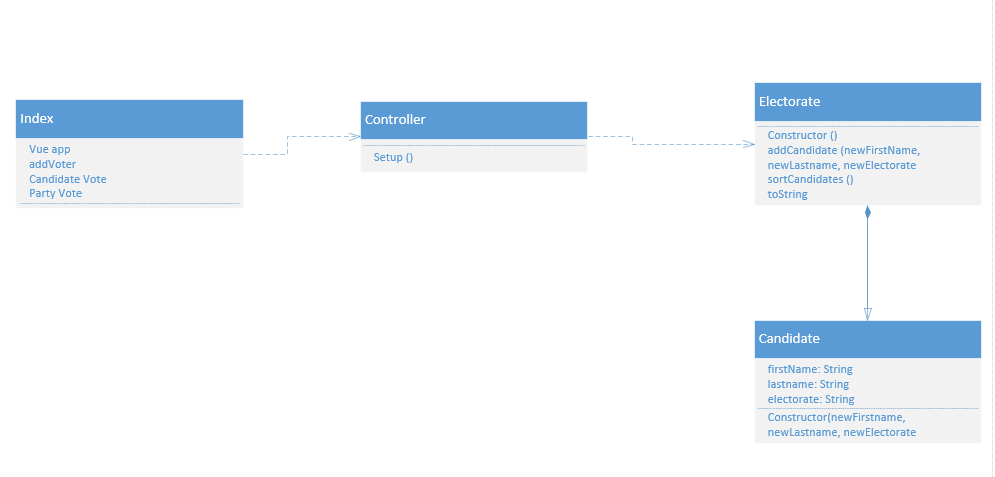
**BCDE102 Assessment 3 Daniel Wild**

**Iteration 2**

**WHERE:**

**WHAT:**

The goal of iteration 2 is to take iteration 1 add CSS code to the Index.html to make the page look more like a voting form that a professional would output if they created a voting webpage.

Users Story:

I am a voter from Wairarapa I want to be able to see the candidates and parties I can vote for. I want to be able to vote for only one of each party and candidate. I want to be able to see the explanation for each vote I am able to make.

The Tasks:

1. Fix the problems from iteration 1 by recoding it so it calls from controller instead of being hardcoded into Index
2. Create a plan of what CSS code will be needed to get the desired look
3. Add each CSS code snippet and play around with their properties to get the look right
4. Make sure that the program still work as needed
5. Update unit tests to test controller and html

Task Time Estimates

* Fix the problems from iteration 1 by recoding it so it calls from controller instead of being hardcoded into Index – 1 hour
* Create a plan of what CSS code will be needed to get the desired look – 30 minutes
* Add each CSS code snippet and play around with their properties to get the look right – 30 minutes
* Make sure that the program still work as needed – 10 minutes
* Update unit tests to test controller and html – 30 minutes

Estimated Total Time: 2 hours 40 minutes

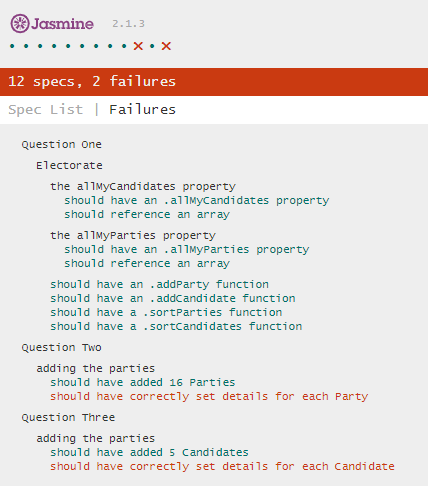
Planned product is a voter would be able to see the names of both the candidates and the parties in the form and have radio buttons to be able to select one that they want to vote for. It should also look professional and look like a voting form that are used in the official New Zealand elections.

Task Times

* Fix the problems from iteration 1 by recoding it so it calls from controller instead of being hardcoded into Index – 1 hour
* Create a plan of what CSS code will be needed to get the desired look – 45 minutes
* Add each CSS code snippet and play around with their properties to get the look right – 1 hour
* Make sure that the program still work as needed – 5 minutes
* Update unit tests to test controller and html – 40 minutes

Total Time: 3 hours 30 minutes

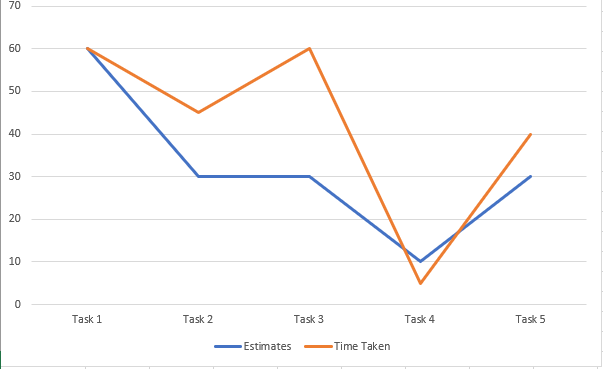
**How:**

Unit tests

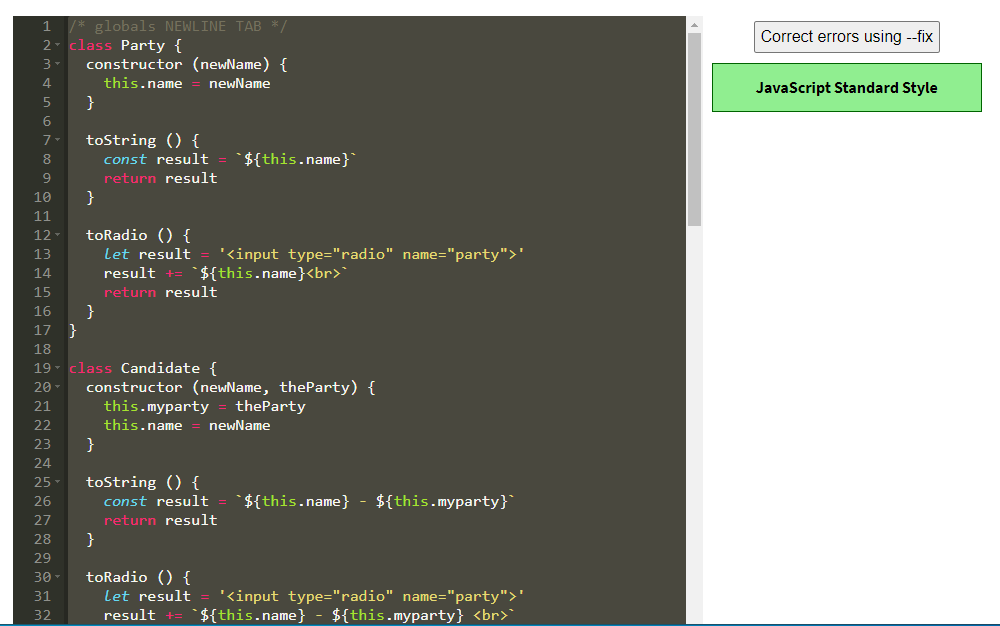


The JS unit test were updated to test the newly added controller so it can now make sure the right number of candidates and parties are in the controller. It does fail with correct details as they are not run through the sort function in Electorate this will be resolved in iteration 3. The html test only work at the moment when on the same form as Index and only test for the number of sections and divs. These will be fixed to work in specrunner and more tests added in version 3.

Burndown Chart



**Evaluation:**

Style



**What happened vs what was planned:**

My plan was to fix the problem of iteration 1 being having the parties and candidates hardcoded into Index by adding in a controller class and them take that code and apply CSS code to it to make it look like a official New Zealand election voting form. Then update my unit tests to test all the JS and HTML. The fixing of the JS code and CSS style worked well and went to plan but did end up taking along than I thought they would. The unit tests didn’t go to plan and ended up with the JS not being able to see the details of the candidates and parties and the html only working when it is added to the Index form.

**Performance Review:**

I was able to achieve the goal of the iteration which was taking the last version adding more JS code and making it look professional using CSS. The unit tests did not go to plan and is definitely the main thing I need to work on and fix going forward into iteration 3.