Website Functionality:

The website works by using an API key to connect to the Auckland bus transport company. A map is displayed of Auckland and when a specific route is selected it queries the database and gets the information for all the buses on that route. It then displays the buses for that area on the map using markers, when a particular bus is clicked it shows the relevant information for that bus such as the trip Id, Arrival/departure time, Arrival/Departure delay, occupancy status and next stop. One thing I did notice is that no matter what bus you clicked on it said the occupancy status is 6, this is what the API is returning so I am assuming it is a security reason or might be a flag for the Auckland transport company.

Client side:

The client side of the website is stuff like the markers that appear on the map. The markers are used using JavaScript. The map itself is run on the client side, the combo box and selecting the routes is all done on the client side of the web page. The map auto resizing when route selected and auto refreshed is done on the client side of the web site. The CSS is on the client side of the web site, it doesn't add functionality it just makes the website easier on the eye and less boring to look at.

Server side:

The server side of the website is sending the request to the Auckland database to get the position of the buses. Getting the route information to put into the combo box is done on the server side of the website. Bus co-ordinates, update of position is done through the server side of the website. Basically all request for information from the database happens on the server side of the website. The auto refreshing every 30 seconds is launched from the client side function but getting the data is a server side function embedded in it.

Communication and code sharing:

Three of us were in the first lab sitting next to each other and I asked them if they had a group and they said no, so we formed a group. James created a Facebook page and added me and Quentin to the page. I then selected a group on learn to allocate to and posted it in the Facebook page. Johnson was added to our group randomly and he wasn't in our tutorial to communicate with. After a series of emails he had a Facebook page to work on group projects and I added him to the Facebook page. The Facebook page was used to communicate about next meeting, what we had done on the website in our own time etc. We shared a code and the work we had done through github.

On our first meeting we each downloaded the project skeleton and Quentin and I started working on the GUI side of the project. We added the map and the combo box to the site and started working out how to position the objects on the page. James and Johnson started on working on getting the data to put in the combo box. While Quentin and I worked on all the auto resizing stuff for the map and everything else related to GUI and James and Johnson worked on the queries we ran into a few problems.

Problems:

We found it very hard to understand how to use the Auckland Transport Realtime API. When we did figure out how to correctly call the API, the administrators changed the system. Every time we tried selecting a route or anything there was a loading error which was due to the Auckland transport API and the constant changes made to the database etc.

We found it difficult to get the CSS to do what we wanted. Originally, we wanted to put an image of a bus, in a div, to the left of the map. Using the normal 'float: left' property did not seem to work. We found that the map div is actually a collection of divs, and we think our difficulties were because of some Google formatting that we have no control over. We settled for having the image below the map, to the left of the drop-down menu.

One problem we had not so much with coding was that whenever something went wrong or the code stopped working we got frustrated to quickly which lead to the "let's come back and fix it tomorrow," rather than just take a five minute walk and think about what was going wrong.

Quentin and I had a major problem with the GUI on our second meeting, we tried changing the position of the combo box and it stopped from the map loading and basically every change we made was one step forward, two steps back. We fixed this problem by going back to an older version of the GUI and started this part over. We did have a problem with colours for the CSS. We were trying to change the background colour to a gradient, no matter where we tried putting the colour it didn't change. This as it turns out was due to there being two CSS files, every change we made was to the wrong one.

What we would do differently:

If we were developing this web app for release to the public, there are several improvements we would make to it.

When the user clicks on a bus marker, information is displayed. However, some of this information would not be relevant to a commuter who might use the app. The route and trip ids, for example, have no relevance for commuters.

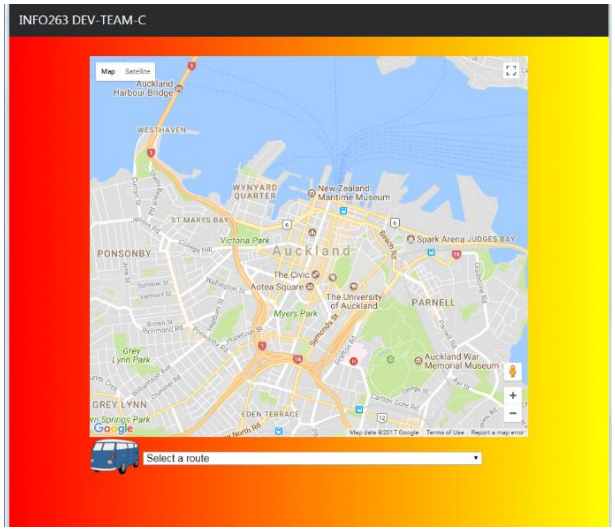
When a route is selected from the drop-down menu, the web app currently displays the locations of busses on the route. It would be useful to also display on the map a trace of the route and the locations of all the stops along the route. Information should pop up if a user clicks on a stop marker, such as the address of the stop and the ETA of the next bus on that route. Perhaps also the timetable for the day for that stop.

The timetable of the currently selected route could also be displayed to the left or right of the map.

It may also be nice to allow the user to make a collection of 'favourite routes', so they do not have to search through the list of all the bus-routes each time they use the app.

When the user selects a route from the drop-down menu, there is a delay until the bus markers are displayed. There should be either a progress bar or a loading graphic to let the user know that the app has not frozen.

Website Loaded:



Route selected: