

# Where in the world is CS?

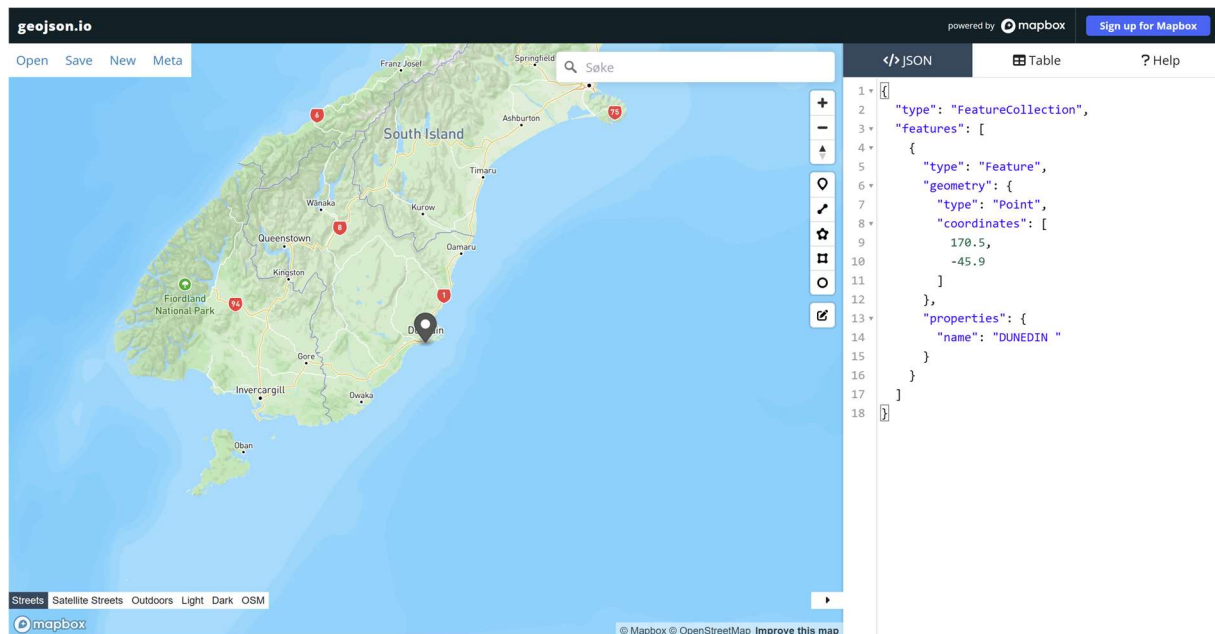
## Approach

I chose to do this etude in HTML and JavaScript code, because the geoJSON format is based on JavaScript syntax, and so the conversion between the two is very simple. Doing it as an HTML-document also made it easy to link the program with the web-resource [geojson.io](http://geojson.io), as well as providing nice ways to interact with the user.

To abstract the input into geoJSON data, the script constructs a TextProcessor to read and interpret the input text. Simply put, the processor reads the input from right to left and detects certain non-numeric markers that indicate how it should interpret the next number (the number to the left of the marker).

These markers are N, S, E, W for directions and °, ‘, “ for sexagesimal coordinates.

The TextProcessor then put the interpreted data into Position-objects, that is added directly into the “features” of the geoJSON data. If the processor struggles to interpret the input, it will give alerts (pop-up windows in the web browser) with the appropriate error messages.



*Clicking 'Go to map' will send you to this page, where your positions are displayed on a global map.*

After adding all the positions they want, the user can click on the ‘Go to map’ button, which will send them directly to [geojson.io](http://geojson.io), displaying all the positions that were added.

The user also has the option to download the created geoJSON data in a text-file, which is handy if they want to use some other tool/software to visualize their data.

The HTML page looks like shown below.

# Where in the world is CS?

Write some coordinates in the textbox below, and they will be formatted into a geoJSON string.

Examples on valid input:

45.9 S, 170.5 E Dunedin

40° 26.767' N 79° 58.933' W

00010.° 054e0' 12.6000", -63.4457 S Hell

Explanations of the buttons:

'Go to map': Click to display your locations on a map.

'Export': Click to export the geoJSON data to a .txt file.

'Undo': Click to remove the last position from the dataset.

'Clear': Click to clear all geoJSON data, and start over.

--- Write your coordinates here ---

OK

Go to map

Export

Undo

Clear

Added positions:

-45.900000, 170.500000 DUNEDIN

geoJSON data:

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          170.5,
          -45.9
        ]
      },
      "properties": {
        "name": "DUNEDIN "
      }
    }
  ]
}
```