| Feature | Max points |
|---|---|
| The application show relevant data on a map and user has change to change the data | 3 |
| The application show relevant data on a chart and user has a chance to change the data | 3 |
| User is able to switch between different layers of data on map | 2 |
| By clicking the map user has an option to get to additional charts covering that area | 4 |
| There are more than two items of data available (e.g. elections data, employment rate and number of residents) – this means that there are two API calls made | 4 |
| Able to download the visualization as a PNG (or SVG) image | 2 |
| Well written PDF report | 3 |
| Application is responsive and can be used on both desktop and mobile environment | 4 |
| Application works on Firefox, Edge and Chrome | 3 |
| The application has clear directory structure and everything is organized well | 2 |
| **SUM** | 30 |

**The project**

The project is a statistic portal that visualises the income deciles of each municipality in Finland. You can choose which decile will be displayed on the map and clicking on a municipality updates the chart below to show the development of the current decile displayed of that municipality. You can also manually choose the municipality if you want and there is a possibility to download the map as a png. There is also a possibility for the user to add the total population of a municipality to the chart.

**What was used**

The map uses Leaflet with an additional plugin Leaflet.image to download the image. The project also uses Frappe to make the charts. Other than that I only used HTML, JS and CSS.

**What was done**

Coded the project as described tested it on Firefox, Edge and chrome. Could not test on Safari as I do not own any Apple devices and apple has stopped support for Safari on Windows. I tested this also for mobile devices and it is useable some data is harder to readon mobile devices on the chart but I do not think there is anything I could have done to make that better. There are 2 Javascript files as I thought at one point that it would be easier. There are main.js and chart.js files. the main.js mostly handles the map and chart.js only handles thing concerning the chart. Additionally there is one HTML file, index.html and one CSS file, styles.css.

**References used in the project:**

Frappe charts and some help implementing them :https://frappe.io/charts/docs

Leaflet for the map https://leafletjs.com/

Leaflet-image plugin for downloading the map https://github.com/mapbox/leaflet-image

This stackoverflow answer for downloading an image using JS
https://stackoverflow.com/questions/17311645/download-image-with-javascript