Project report

Antti Kuru, 0614321

Introduction

This document contains an overview of the Project 2: Statistic portal which was done in the Introduction to Web Programming course. The project provides details on what was done, what tools were used, and how many points should be gotten from the project.

Description

The statistical portal project utilizes open data from Tilastokeskus, which is visualized in multiple ways, including a map and charts. The project uses the Leaflet JS library for mapping which uses GeoJSON. The charts are created with Frappe.io framework. The project is made in vanilla JavaScript, HTML, and CSS.

Features

Map:

Open data utilized in the map:

- Municipality level
  - Employment ratio (combines data from employment and unemployment data)
  - Average apartment square prices
- Regional level
  - Average vehicle ages

-slider under the map which determines which year's employment data is visualized

-clicking a municipality/region in the map shows the data in numbers and there is a link for seeing additional charts regarding the clicked municipality

Charts:

Open data utilized in charts (municipality level):

- Number of kids in families (8 different classes)
  - Drag'n'drop containers to let the user decide which data is shown
- Birth and death statistics
- Predicted population estimations

-Ability to download the charts as SVG

-Ability to let user estimate future values based on the previous values

-Ability to download a JSON file that is automatically visualized (NOTE this works only for week 5 homework's two downloadable files which were automatically application/JSON type (dataset), I couldn't download any usable data from Tilastokeskus to my own computer that could have been used

AI Usage

ChatGPT was used to get ideas for a couple of problems I found hard to solve myself such as how to pass the correct municipality code to the charts.html page when the link to “see additional charts” was clicked on the selected municipality. Also, I tried multiple times to update the charts with the help of Frappe’s documentation and chart.update(data) but it did never work, chatGPT proposed to just “remake” the chart and it worked.

Points table:

Feature	**Max**
Drag’n’drop new data to charts/maps	4
The application show relevant data on a map and user has chance 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 one item of data available (e.g. elections data, employment rate and number of residents) – this means that there are two API calls made	3
There are more than two items of data available (e.g. elections data, employment rate and number of residents) – this means that there are three API calls made	2
Data is combined and merged to generate new data, which is then visualized	3
Users can define what should be done to different data items (e.g. values are added, multiplied together etc. before visualization)	2
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, Safari, Edge and Chrome	3
The application has clear directory structure and everything is organized well	2
Total points	40