

## Harjoitustyö vaihe 3

### TLO-32400 Ohjelmallinen sisällönhallinta

#### Yleisesti

Päätin ottaa datalähteekseni Tampereen avoimesta datasta kaupungin ostotiedot. Tarkoitukseni on visualisoida dataa esimerkiksi treemap kuvaajalla. Data oli saatavana rajapinnan kautta, mutta rajapinta ei sopinut tarkoitukseeni vaan päätin toteuttaa datan haun manuaalisesti.

#### Backend

Kuten mainitsin, ei rajapinnan tarjoamat tiedot olleet tarpeeksi, joten toteutin python skriptin joka käsittelee ladattua csv tiedostoa ja laskee tiedostosta kunkin yrityksen laskujen summan ja lisää psql tietokantaan summan ja yrityksen nimen. Skriptin ajamisessa menee yli kymmenen minuuttia, joten se ajetaan tämän takia käsin ja harvoin. Kuvassa lyhyt ote skriptistä.

```
with open('tampereen_kaupungin_ostot_2018.csv') as file:
    reader = csv.reader(file, delimiter=',')
    line = 0
    for row in reader:
        if line != 0:
            print(line)
            if str(row[18]) in data:
                data[str(row[18])] += float(row[20].replace(',','.'))
            else:
                data[str(row[18])] = float(row[20].replace(',','.'))
            line += 1
        if line == 0:
            line = 1
```

Tämän jälkeen lisäsin omaan rajapintaan keinon hakea juuri kerättyä dataa. Tämä toteutui yhdellä funktiolla. Rajapintaa kutsumalla saa vierellä olevan kuvan kaltaista dataa ulos.

```
[
  {
    "nimi": "PIRKANMAAN SAIRAANHOITOPIIRIN KUNTA",
    "summa": 314939921.34
  },
  {
    "nimi": "Haygrove Limited",
    "summa": 60949975.66
  },
  {
    "nimi": "KANSANELÄKELAITOS",
    "summa": 28237315.51
  },
  {
    "nimi": "Tuomi Logistiikka Oy",
    "summa": 18128523.19
  },
  {
    "nimi": "VÄINÖ PAUNU OY",
    "summa": 14641580.42
  },
  {
    "nimi": "LÄNSILINJAT OY",
    "summa": 13437002.28
  },
]
```

## Frontend

Datan näyttäminen sivulla on helppoa ja toimi samalla tavalla kuin viime osiossa. Päänvaivaa aiheuttanut osio oli saada käyttäjien erottelu toimimaan käyttöliittymän puolella. Tätä ja eri näkymien näyttämistä varten tuli ottaa käyttöön react-router. Alla data esitettynä käyttöliittymässä ja kirjautuminen.

**Hei Veikko**  
[Data](#) [Posts](#) [Logout](#)

Data

PIRKANMAAN SAIRAANHOITOPIIRIN KUNTA = 314939921.34€  
Haygrove Limited = 60949975.66€  
KANSANELÄKELAITOS = 28237315.51€  
Tuomi Logistiikka Oy = 18128523.19€  
VÄINÖ PAUNU OY = 14641580.42€  
LÄNSILINJAT OY = 13437002.28€  
Tampereen Tilapalvelut Oy = 12477084.11€  
TAMPEREEN KAUPUNKILÄHETYS RY = 10432349.41€  
SOPIMUSVUORI OY = 8827605.07€  
TAMPERE-TALO OY = 8643327.69€  
Luona Hoiva Oy = 7958938.15€  
TAMMENLEHVÄKESKUS OY = 7948014.14€  
PIRKANMAAN SENIORIPALVELUT OY = 7634319.19€  
INVALIDILIITON ASUMISPALVELUT OY = 7507011.09€  
Tullinkulman Työterveys Oy (TYTÄR) = 7302310.69€  
KESPRO OY = 6743266.34€  
MED GROUP OY = 6283509.24€  
FIMLAB LABORATORIOT OY = 6019731.18€  
NESTE MARKKINOINTI OY = 5843780.06€  
ATTENDO OY = 5780535.73€

Kirjaudu sisään

Käyttäjänimi

Kirjaudu

Käyttäjien erottelussa ei käytetä salasanaa, vaan kaikki autentikointi tapahtuu vain käyttäjänimellä. Rekisteröintiä ei myöskään erikseen tehdä, vaan jos syötät uuden käyttäjänimen, luodaan käyttäjä automaattisesti.

## 3 helppoa/hankalaa tekijää

1. React on vieläkin hyvin uutta minulle, joten melkein kaikki pitää katsoa jostain.
2. Minulla oli tarkka päämäärä mitä halusin toteuttaa, joten sopivia ohjeita ei tälläkään kertaa löytynyt, vaan piti soveltaa.
3. Datan käsittely pythonilla on paljon sujuvampaa kuin javascriptillä.

## Hyödyllisiä linkkejä

- [https://medium.com/@Elijah\\_Meeks/interactive-applications-with-react-d3-f76f7b3ebc71](https://medium.com/@Elijah_Meeks/interactive-applications-with-react-d3-f76f7b3ebc71)
- <https://data.tampere.fi/data/fi/dataset/tampereen-kaupungin-ostot>
- <https://blog.pusher.com/getting-started-with-react-router-v4/>