

Ohsiha harjoitustyö vaihe 1

Ohjelmallinen sisällönhallinta kevät 2019

Joona Prehti, joona.prehti@tuni.fi

Koostepalvelu suunnitelma ja API

Kurssin alusta alkaen minusta on tuntunut luontevimmalta lähteä harjoitustyössä kehittämään koostepalvelua. Uskon koostepalvelun tekemisen kautta oppivani kaikkein hyödyllisimpiä taitoja ja teknologioita. Koostepalvelun eduksi voidaan lukea myös se että hyvin tehtynä se voi olla valttikortti työnhaussa.

Olen ollut jo pitkään kiinnostunut kryptovaluutoista ja siksi päätinkin lähteä kehittämään kryptovaluutta aiheista API:a. Huomasin että CoinMarketCap niminen sivu tarjoaa ilmaiseksi API:n jonka kautta saa haettua paljon erilaista dataa kryptovaluuttamarkkinoihin liittyen. CoinMarketCap on hyvin suosittu ja tunnettu sivu kryptovaluuttojen hintojen ja historian seuraamiseen. Uskonkin voivani luottaa heidän jakaman datan luotettavuuteen ja saatavuuteen.

Tarkoituksena olisi siis tehdä jotain visuaalista ja tyylikästä tällä kryptovaluutta datalla. Uskon myös että minulle on kertynyt jo sen verta tietotaitoa devaamisesta että onnistun siinä.

Valitut teknologiat

Päädyin valitsemaan palvelinpuolen teknologiksi Node.js:n (versio 10.4.1), joka on tehokas JavaScriptiä käyttävä back-end teknologia. Päätös ei ollut vaikea, koska olen omalla ajallani harjoitellut Node.js ympäristöä ja siihen liittyviä kirjastoja. Tietokannaksi valitsin PostgreSQL:n (versio 11), koska se oli minulle tuttu web-ohjelmointi kurssilta ja valmiiksi asennettu tietokoneeseeni. Front puolelle ajattelin testata Angular kirjastoa, koska minulla on ollut jo pitkään suunnitelmissa opetella sen käyttöä. Tämä projekti voisi olla hyvä hetki opetella minulle täysin uusi teknologia. Olen myös saanut sellaisen käsityksen että web-kehittäjän on syytä osata ainakin jompikumpi front teknologia, joko React tai Angular.

Tiedon visualisointiin ajattelin käyttää luennolla esiteltyä D3.js kirjastoa, koska sillä toteutetut kaaviot ja kuvaajat vaikuttivat tyylikkäiltä. Varsinaiseen koodaamiseen käytän Visual Studio Codea, koska se on tuttu ja todella kätevä IDE. Versionhallintaan käytän Git:iä, koska se käytännössä alan standardi versionhallintaan ja sitä ei voi koskaan oppia liian hyvin. Selaimena käytän Chromea, koska siinä on hyvät ja tutut kehittäjätyökalut. Lisäksi käytön Postman nimistä

sovellusta jolla pystyn tekemään API kutsuja vaivattomasti ja tarkastelemaan mitä API palauttaa.

Asennukset

Node.js, PostgreSQL, Postman ja Visual Studio Code olivat tietokoneellani jo valmiiksi asennettuna joten en käy asennusprosessia tässä tarkemmin läpi.

Tämänhetkinen toiminnallisuus

Tällä hetkellä ohjelman toiminnallisuus on vielä hyvin pientä. Ohjelmassa on hyvin yksinkertainen Node.js serveri, joka noutaa CoinMarketCap API:sta sadan suurimman kryptovaluutan perustiedot json-formaatissa. Sovellus ei siis vielä tee datalle yhtään mitään, mutta halusin saada sen nyt jo sellaiseen vaiheeseen että datan noutaminen API:sta onnistuu. Tässä kuva palvelun palauttaman arrayn ensimmäisestä alkioista:

```
{ id: 1,
  name: 'Bitcoin',
  symbol: 'BTC',
  slug: 'bitcoin',
  circulating_supply: 17550637,
  total_supply: 17550637,
  max_supply: 21000000,
  date_added: '2013-04-28T00:00:00.000Z',
  num_market_pairs: 6644,
  tags: [ 'mineable' ],
  platform: null,
  cmc_rank: 1,
  last_updated: '2019-02-21T18:43:26.000Z',
  quote:
    { USD:
      { price: 3950.75326186,
        volume_24h: 8190758626.8551,
        percent_change_1h: 0.0884416,
        percent_change_24h: -0.65618,
        percent_change_7d: 9.10327,
        market_cap: 69338236375.4708,
        last_updated: '2019-02-21T18:43:26.000Z' } } }
```

Kuva on otettu VS Coden konsolista. Kryptovaluutat ovat array:ssa markkina-arvon mukaisessa suuruusjärjestyksessä ja siksi kaikille tuttu Bitcoin tuleekin sieltä ensimmäisenä.

Tiedon hakemisen jälkeen otan datasta irti attribuutin market_cap ja tallennan sen omalla tietokoneellani pyörivään PostgreSQL kantaa. Market_cap on kyseisen kryptovaluutan tämänhetkinen markkina-arvo kokonaisuudessaan.

Helpot ja vaikeat

- Tietokannan yhdistäminen Node.js sovelluksen kanssa oli hankalaa. Tuntui kuin netistä ei olisi löytynyt mitään hyviä ohjeita toimenpiteen suorittamiseen. Onneksi sain mallia "Tietokantojen suunnittelu"-kurssilla toteuttamastani Node.js sovelluksesta.
- Mielenkiintoisen koostepalvelun keksiminen oli aluksi hieman hankalaa. Onneksi kuitenkin löysin googlaamalla paljon kiinnostavia API projekteja.
- Erityisen helppoa oli REST API:n kutsuminen Node.js:n Request kirjaston avulla. Pelkän kutsun tekeminen ja konsoliin tulostaminen vaatii vain noin 7 riviä koodia.
- Simppelin etusivun tekeminen oli myös hyvin helppoa Node.js:n Express kirjaston turvin. En tarvinnut montaakaan riviä koodia ja sain jo tekstiä tulostumaan osoitteeseen localhost:8888/etusivu.

Lähteet ja linkit

- <https://www.npmjs.com/>
- <https://www.postgresql.org/>
- <https://d3js.org/>