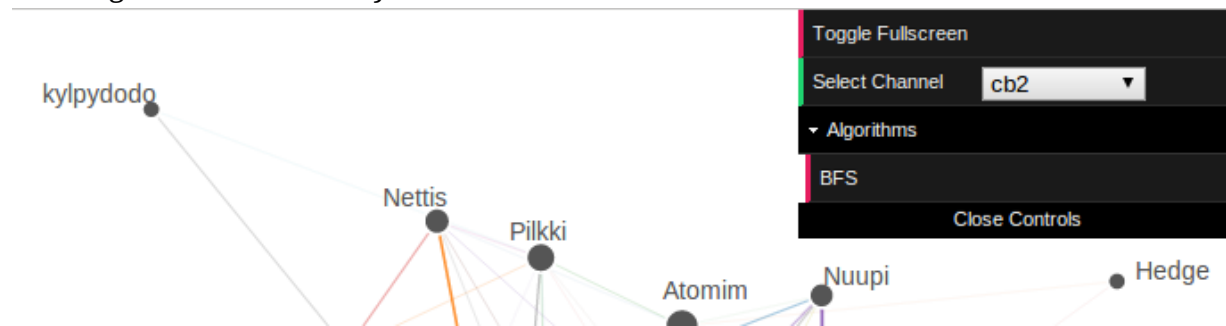


# Viikkoraportti 1

## Käyttöliittymää

Tällä viikolla aloitin projektin toteutuksen refaktoroimalla käyttöliittymää, jotta siihen on helpompi lisätä projektin tarvitsemat osat. Lisäsin sivulle valikon, josta voi valita, mitä algoritmia haluaa käyttää:



## Vieruslistaesitys

Implementoin vieruslistaesityksen generoimisen, sillä tällä hetkellä solmut ja kaaret on tallennettuna erikseen ja jokainen kaari tietää omat solmunsa

Koodi: <https://github.com/tuhoojabotti/NodePieSpy/blob/master/lib/algorithms/to-list.js>

Tulostus: <http://lahdenvuo.info/social/list/cb2.json>

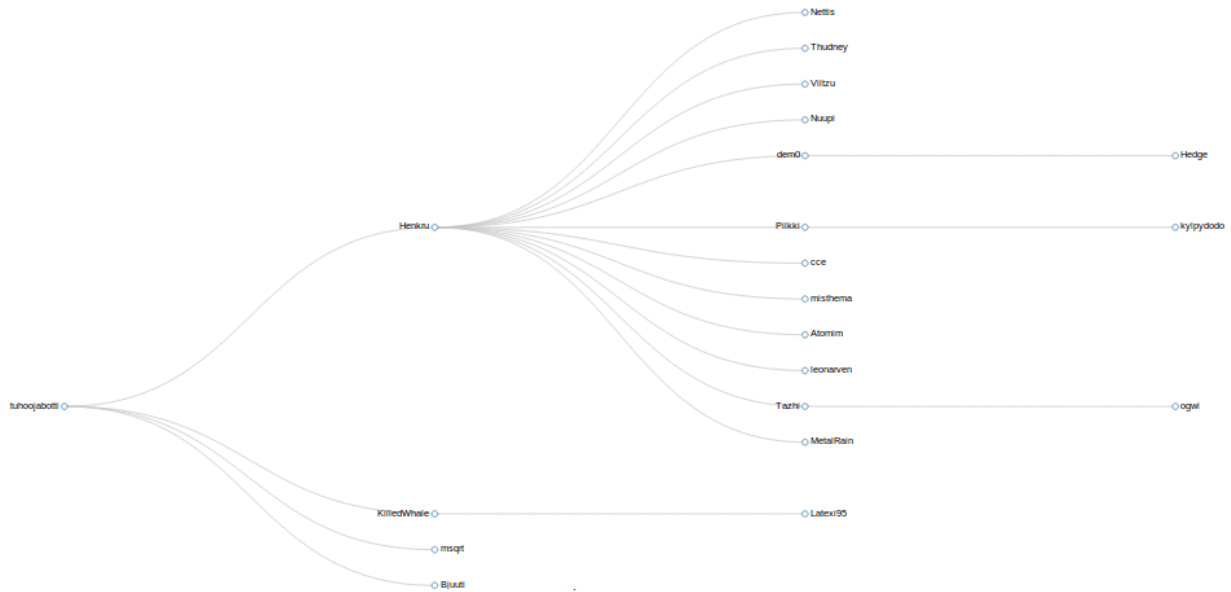
## BFS

Implementoin BFS-puun generoimisen:

Koodi: <https://github.com/tuhoojabotti/NodePieSpy/blob/master/lib/algorithms/bfs.js>

Tulostus: <http://lahdenvuo.info/social/bfs/cb2/tuhoojabotti.json>

Se näyttää tältä:



## Dokumentaatio ja yksikkötestaus

Tämän jälkeen aloitin dokumentaation ja yksikkötestien tekemisen. Tähän tarkoitukseen asensin Gruntin, se on kuin makefile Javascript-projekteille.

Kirjoitin Gruntfilen, joka tarkistaa koodin, ajaa testit, generoi dokumentaation ja kopioi dokumentaation Github-sivuille.

```
tuhojabotti@lakka:~/repot/NodePieSpy$ grunt
Running "jshint:all" (jshint) task
>> 11 files lint free.

Running "nodeunit:all" (nodeunit) task
Testing analyser.test.js.OK
>> 1 assertions passed (60ms)

Running "jsdoc:dist" (jsdoc) task
Documentation generated to /home/users/tuhojabotti/repot/NodePieSpy/doc
Done, without errors.
```

Tuli opittua, miten Grunttia käytetään projektinhallintaan, se vaikuttaa todella näppärältä, kuten makefile, mutta Javascriptiä.

## Yhteenveto

Jäi vähän epäselväksi, miten toteutan ryvästämisen ja yksikkötestauksen, mutta jatkan siitä ensi viikolla. Pitää myös miettiä miten generoin HTTP-api:lle dokumentaation, katsoin jo vähän eri vaihtoehtoja, mutta pitää syventyä tarkemmin, kun on aikaa.

Kokonaisuutena ohjelma on edistynyt hyvin.