JavaScript Työ

Tommi Jokinen

Miikael Lehtimäki

Antti Leppänen

Syyskuu 2016

Ohjelmistotekniikka

SISÄLLYS

[1 OHJELMAN VAATIMUKSET 3](#_Toc463422419)

[2 OHJELMAN ARKKITEHTUURIRATKAISU 4](#_Toc463422420)

[3 OHJELMAN SUUNNITTELURATKAISU 5](#_Toc463422421)

[4 HAASTEET 6](#_Toc463422422)

[5 TOIMIVUUS 7](#_Toc463422423)

[6 Ohjelman testaus; ohjelman asennustoimintojen testaus 8](#_Toc463422424)

[7 OHJELMAN KÄYTTÖÖNOTTO 9](#_Toc463422425)

[8 TYÖAIKA 10](#_Toc463422426)

[9 YHTEENVETO 11](#_Toc463422427)

# OHJELMAN VAATIMUKSET

Suora lainaus tehtävänannosta:

*Tehtävänäsi on tehdä yksinkertainen navigaattoriohjelma, joka laskee pienimmän etäisyyden kahden suomalaisen kaupungin välillä maanteitse. Vaatimuksia ohjelmalle:*

* *oltava selainpohjainen*
* *hyödynnettävä JavaScript:iä ja sen päälle rakennettuja kirjastoja*
* *suositellut kirjastot: MEAN-pino:*
  + *MongoDB-tietokanta tietovarastona*
  + *Express.js-kirjasto kutsurajapintana Node.js -palvelimeen päin*
  + *AngularJS-kirjasto apuna selainpään toimintojen tekemisessä ja MVC-mallin hyödyntämisessä*
  + *Node.js-kirjasto palvelinpään toimintojen alustana*
* *Muita tutustuttavia teknologioita (ks. alla oleva kuva): REST-rajapinnat*
* *Muitakin kirjastoja voi käyttää mutta olisi suositeltavaa, että ne olisivat valtavirtateknologioita. Perustele selostuksessasi tällaiset valintasi. Käyttöliittymän pitää toimia selaimessa, palvelinpäässä on oltava sovelluksen älykkyys ja käytetyt pohjatiedot on oltava tietokannassa. Lisäksi on sovellettava JavaScript:iä mahdollisimman paljon.*
* *Tämä työ on etupäässä tutustumista uusiin teknologioihin; ei niinkään työ valmiin tuotteen tekemiseksi. Työn eteneminen, aikaansaaminen ja ongelmien ratkaiseminen ovat tärkeämpiä tavoitteita kuin täysin oikein toimiva stilisoitu loppukäyttäjän sovellus. Ja etenemisen yhteydessä tärkeintä on siinä saatava oppiminen. Muistetaan myös ryhmätyön voima eli kaveria autetaan ongelmatilanteissa puolin ja toisin sekä työnjakoa harrastetaan. Etenemisen vaiheet ja lopputuotos kuvataan työstä tehtävässä raportissa.*

# OHJELMAN ARKKITEHTUURIRATKAISU

//TODO

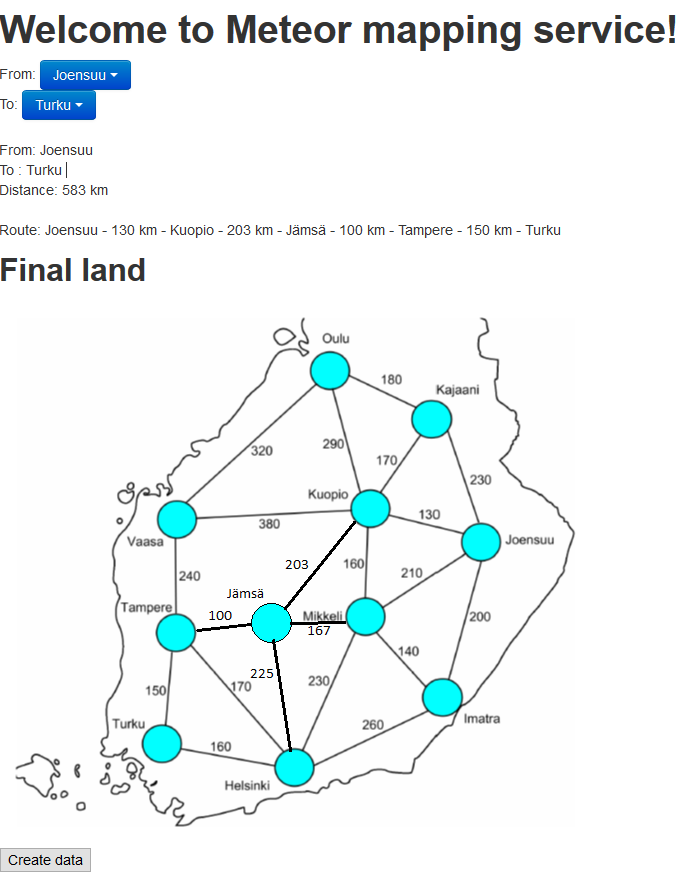
# OHJELMAN SUUNNITTELURATKAISU

moduulit, luokat, aliohjelmat, funktiot, ohjelmakoodin jakautujminen moduuleinehin jne.ne

# HAASTEET

# TOIMIVUUS

Meteor mapping servicen toiminta ja käyttö on yksinkertaistettu yhdelle sivulle. Käyttäjä valitsee sinisistä laatikoista (Kuva x) kaupungit. Kaupunkien valittua ohjelma automaattisesti lähettää palvelimelle kyselyn näiden kahden kaupungin välisestä lyhimmästä reitistä. Vastaus esitetään yhteismatkan määränä sekä minkälaisesta reitistä kyseinen lyhin reitti koostuu, näyttäen jokaisen läpimentävän kaupungin etäisyyden toisistaan.



KUVA x. Meteor kartta palvelun käyttäjä näkymä.

Create data nappi (Kuva x) on (toistaiseksi väliaikaisesti) jätetty nappi, mikäli kaupunki tiedot eivät ole luotuna palvelimella valmiina. Täysin toiminnallisessa ja viimeistellyssä versiossa kaupunki tiedot hoidetaan täysin palvelinpuolella.

# OHJELMAN KÄYTTÖÖNOTTO

* Asenna Meteor.js koneelle:
  + <https://www.meteor.com/install>
  + Noudata linkissä annettuja ohjeita.
* Luo projekti:
  + Avaa komentorivi
  + Navigoi kansioon, johon projekti luodaan
  + Luo uusi projekti komennolla ”meteor create {0}”, jossa {0} on projektille annettu (vapaavalintainen) nimi
  + Kopioi kaikki testattavan ohjelman tiedostot tähän kansioon
* Aja luotu projekti
  + Aja komentorivillä tässä kansiossa komento ”meteor run”
  + Avaa verkkoselain
  + Kirjoita osoiteriville ”localhost:3000”
  + Tadaa!

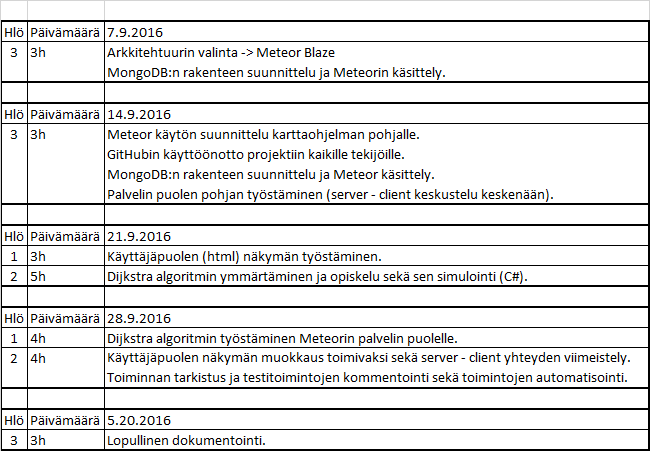
# OHJELMAN TESTAUS

Asenna ohjelma kohdassa 7 annettujen ohjeiden mukaisesti. Valitse pudotusvalikoista lähtö ja määränpää kaupungit (”from” ja ”to”). Tarkista ohjelman antama tulos vertaamalla sitä kuvassa oleviin matkoihin.

Ensimmäisellä ajokerralla pudotusvalikot saattavat olla tyhjiä. Mikäli näin on, paina sivun alalaidassa olevaa painiketta ”Create data”, joka populoi kantaan tarvittavat tiedot.

# TYÖAIKA

Työaikaa kului suhteellisen paljon Meteorin käytön opetteluun. Ongelmia myös koostui tietokoneiden muokattujen erikoisasetusten kanssa, joka alustavasti esti Meteorin käyttöönottamista.



TAULUKKO 1. Projektiin kulutettu aika, kuinka monta henkilöä tehtävää on tehnyt.

Projektissa auttoi työnsiirtoa tekijöiden kesken GitHub, jossa projektia säilytettiin. Tämän avulla työt saatiin jaoteltua eri henkilöille, yksi tekee palvelimen koodia, yksi käyttäjän html näkymää ja yksi käyttäjän koodia.

# YHTEENVETO