

BTMon

Master Documentation

Ronkainen Eero L5047

Matti Aho K9251

Sisällys

Johdanto	3
Tavoitteet.....	3
Käyttötapaukset	3
Käyttötapaus 1: Matti	3
Käyttötapaus 2: Sähkömies	3
Käytön kulku	3
Aikataulus	4
Priorisointi.....	4
Käytetty aika.....	5
Arkkitehtuuri	6
Dataflow.....	6
Käyttöliittymän mockup.....	7
Päänäkymä	7
Tiedostovalikko (vanha)	7
Snapshot -näkymä	8
Tiedostovalikko (uusi)	8
Flow.....	9
Relaatiot.....	10
Tekninen toteutus	11
Käytetyt teknologiat	11
Dataformaatti.....	11
Lukulaitteelta puhelimeen	11
Tallennusformaatti	11

Johdanto

Ohjelma BTMon on alkuperäisesti tuotettu JAMKin kurssille Mobile Application Development (TTOW0615). Kurssin lopun jälkeen projektin tekemistä jatketaan JAMKin Mobile Project (TTOW0635) -kurssin yhteydessä. Aikaisemman kurssin tavoitteena oli tutkia jotain tiettyä teknologiaa ja toteuttaa tästä demo. Tätä demoa on tarkoitus jatkaa täysimittaiseksi sovellukseksi Mobile Project -kurssin aikana.

Tavoitteet

Ohjelman tavoitteena on tuoda arduinon bluetoothin kautta tarjoama data, esimerkiksi auton akun jännitteestä, helposti luettavaksi mobiililaitteella. Ohjelma mahdollistaa lyhyen ajan sisällä tapahtuneiden muutosten havainnoinnin graafien muodossa, sekä hetkellisten tapahtumien tallentamisen.

Ohjelma on tuotettu projektiryhmän jäsenen, Matti Ahon tarpeisiin.

Käyttötapaukset

Käyttötapaus 1: Matti

Matti haluaa selvittää riittääkö auton laturissa teho lataamaan hänen autonsa akkua silloin kun äänentoistojärjestelmä on täydellä teholla. Matti haluaa nähdä helposti puhelimeltaan mitä kullakin hetkellä tapahtuu, jotta hän näkee riittääkö akustossa ja laturissa teho ajaa vahvistinta täydellä potentiaalilla. Matti avaa sovelluksen ja samalla musiikin tasoa säätäessään, hän tarkkailee akun tilaa.

Käyttötapaus 2: Sähkömies

Sähkömies kulkee eri pisteissä ja haluaa tarkistaa, onko verkon jännitteet kohdillaan langattomasti. Hän avaa sovelluksen ja yhdistää sillä arduinon, josta hän näkee verkon tilan.

Käytön kulku

Ohjelman auetessa, se pyrkii yhdistämään arduinon, joka alkaa lähettämään dataa.

Ohjelma piirtää saapuneen datan graafiin, jossa näytetään arduinon koodin lähetysnopeudesta riippuen, kuluneen (x) ajan arvot.

Graafista voi nappia painamalla ottaa snapshotin, jolloin dataa voi tarkastella tarkemmin tai tallentaa. Tallennettuja tietoja voi myöhemmin tarkastella.

Aikataulutus

Kehitystiimin on vaikea sitoutua viikkoaikatauluun, minkä takia päätettiin, että aikataulua ei luotaisi. Tämän sijaan luotiin ominaisuuksia ja työtehtäviä varten priorisointi, jonka mukaan työtehtäviä tehdään sitä mukaan kun kehitystiimi pystyy.

Priorisointi

Priorisoinnin asteikko: 1-5 vähiten tärkeimmästä tärkeimpään.

Raskauden asteikko: 1-5 vähiten raskaimmasta toteutuksesta raskaimpaan.

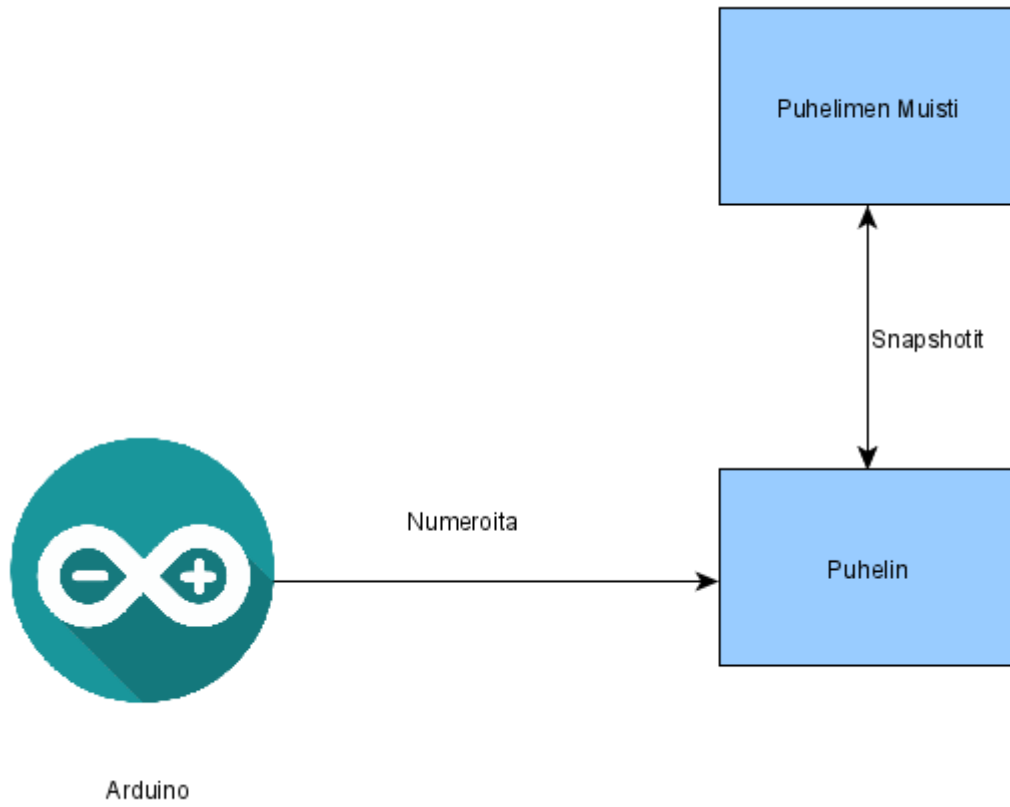
Tehtävä asia	Prioriteetti	Arvioitu Raskaus	Huomiot
Bluetooth	5	5	Dokumentation puute
Arduino (lukulaite)	5	2	
Käyttöliittymän hionta	4	4	
Tiedostojen luku/tallennus	5	4	
Volttimittari	2	2	
Graafin toiminnallisuus	4	3	Dokumentation puute
Näkymät	3	3	Ei suoraa elementtiä

Käytetty aika

Milloin	Kuka	Mitä tehtiin
05.10.2019	Eero + Matti	Arduinon rakennus + debuggaus
06.10.2019	Eero + Matti	Raspberry Pi Conffaus
07.10.2019	Matti + Eero	Raspberry Pi Bluetooth
08.10.2019	Matti + Eero	QT Bluetooth
10.10.2019	Matti + Eero	Raspberry Pi koodin loppuunvienti
10.10.2019	Matti + Eero	QT + Raspi yhteys loppuun
17.10.2019	Eero	Käyttöliittymän skaalautuvuus
18.10.2019	Eero	Käyttöliittymän skaalautuvuus
19.10.2019	Eero	QtGraphs yrityksiä
20.10.2019	Eero	QtGraphs + Skaalautuvuus
20.10.2019	Eero + Matti	Qt Bluetooth datan käsittely
21.10.2019	Eero	Qt Bluetooth data graafiin
22.10.2019	Eero + Matti	Esityksen Valmistelu (Research Assignment)
28.10.2019	Eero	Refaktorointi
09.11.2019	Eero	UI suunnittelua
10.11.2019	Eero	QML Komponenttien tutkimista
16.11.2019	Eero	UI Mockup / Tiedostonhallinta tutkiminen
06.12.2019	Matti	Arduinon rakennus + conffaus
06.12.2019	Eero	Käyttöliittymän aloitus + Lokaali datageneraattori
07.12.2019	Eero	Tiedostonhallinta aloitus
08.12.2019	Eero	Tiedostojen tallennus
09.12.2019	Eero	Koodin siistiminen
09.12.2019	Matti	Arduino Bluetooth
10.12.2019	Matti	Tiedostojen luku + näkymän mockaus
10.12.2019	Eero	Snapshot -näkymän dialoginappi/tallennuslogiikka
11.12.2019	Eero + Matti	Dokumentaation viimeistely + Esitykseen valmistelu

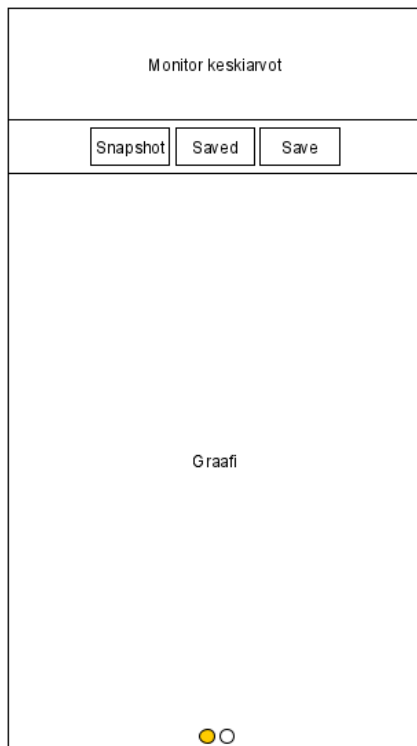
Arkkitehtuuri

Dataflow

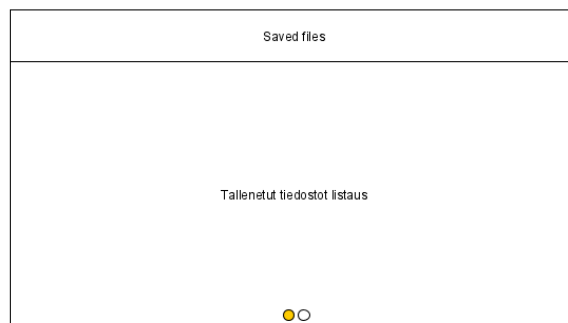
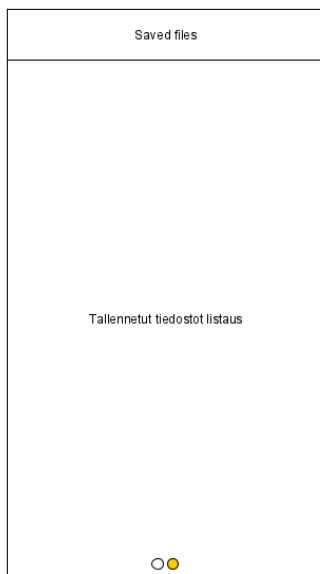


Käyttöliittymän mockup

Päänäkymä



Tiedostovalikko (vanha)



Snapshot -näkymä

<	Snapshot	⋮
Monitor 1 Sisältö	Monitor 2 Sisältö	

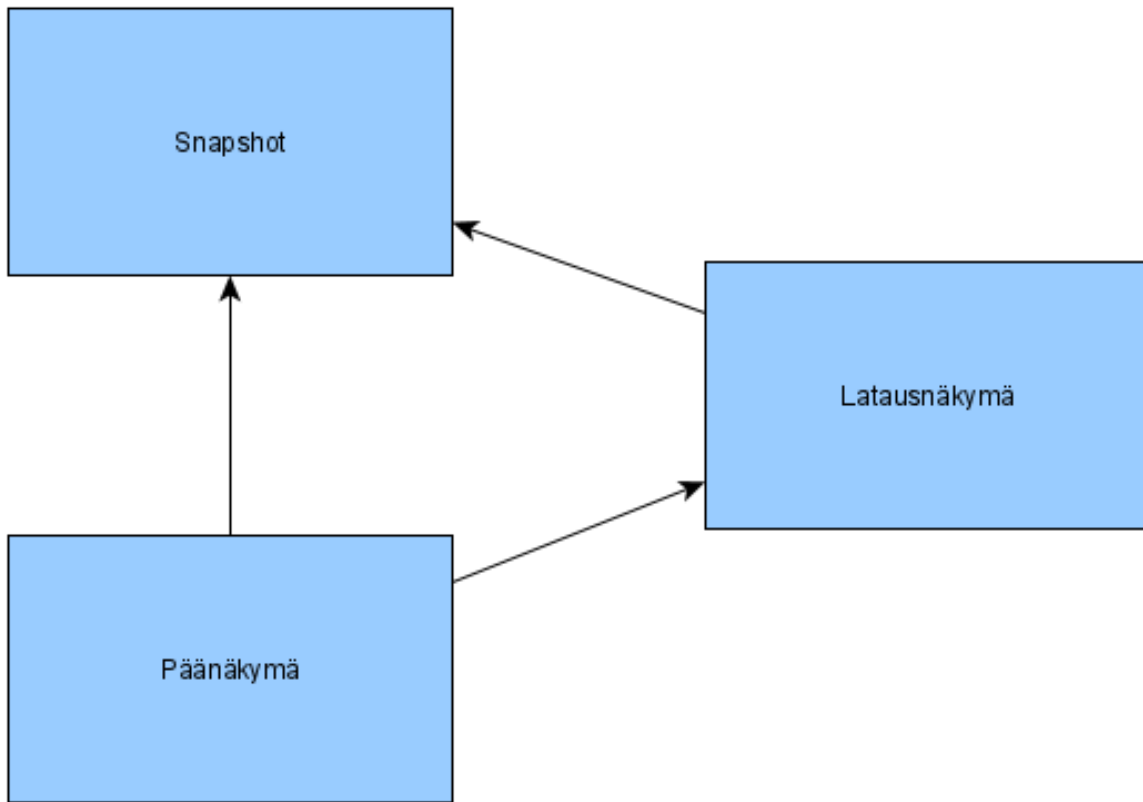
<	Snapshot	⋮
Monitor 1 Sisältö	Monitor 2 Sisältö	

Tiedostovalikko (uusi)

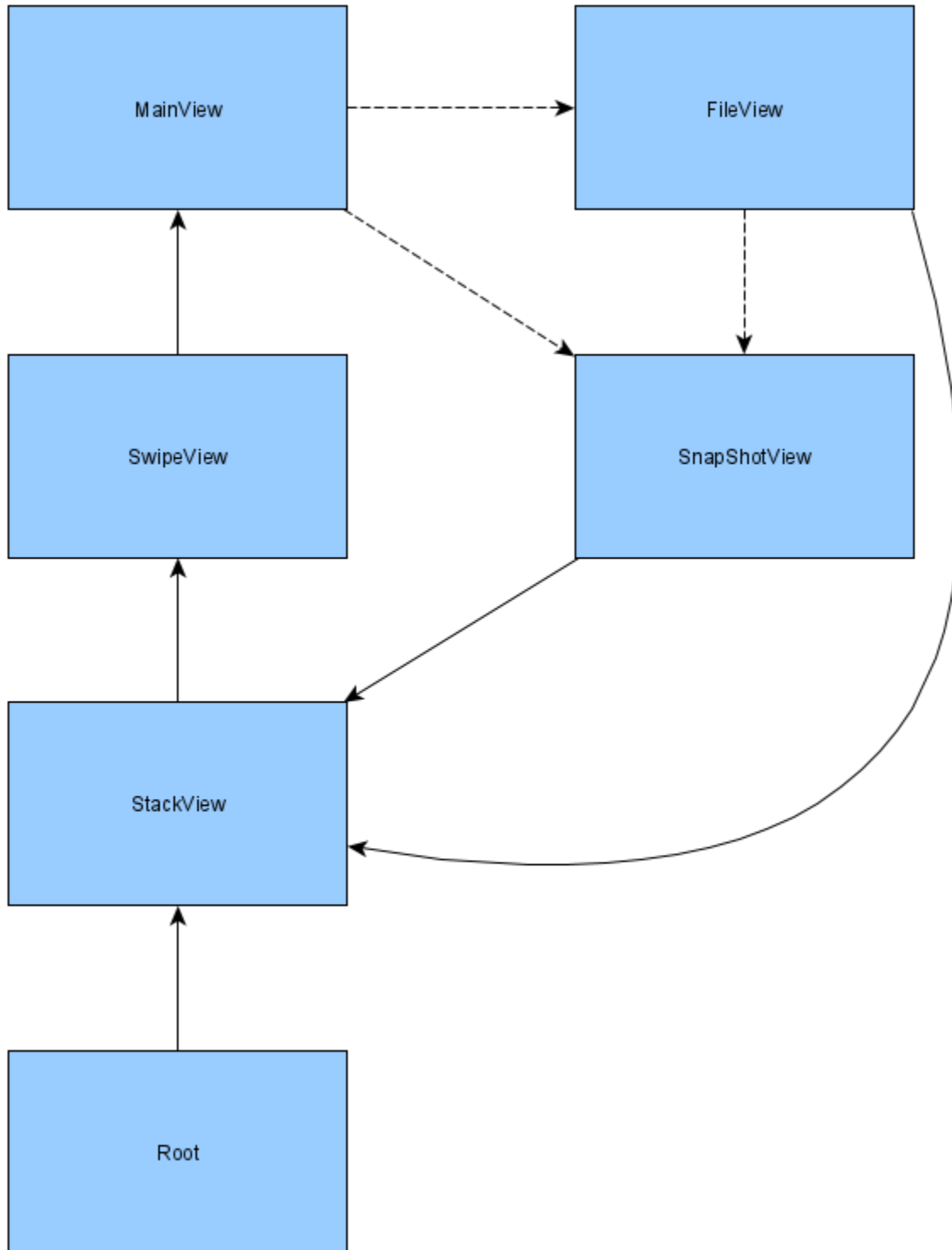
<	Saved files
Tallennetut tiedostot listaus	

<	Saved files
Tallennetut tiedostot listaus	

Flow



Relaatiot



Katkoviivat ilmaiset kontekstin katoamista, eli komponentti ei pysty referoimaan isäntäkomponenttia.

Tekninen toteutus

Käytetyt teknologiat

- QT 5.12.5
 - QTCharts 5.12.5
- QT Creator 4.10.0
- Android SDK 26.1.1
- Android NDK 20.0.5594570
- OpenJDK 1.8.0.25

Dataformaatti

Lukulaitteelta puhelimeen

Arduino lähettää dataa formaatissa jossa eri sensorit/monitorit ovat jaoteltu pilkulla(,) ja mittauspisteet puolipisteellä(;) paketin loppua merkataan newline merkillä (\n)

Lopullinen data on esim. tällaista: "10,40;20,80;\n".

Tallennusformaatti

Laitteesta tallennettaessa graafissa tai snapshot -näkyvässä näytetty data tallennetaan utf-8 enkoodattuna btlog -tiedostoon. Formaattissa rivi vastaa monitoria ja rivillä oleva data on tekstinä enkoodattu lukujono, minkä luvut ovat erotettu pilkuilla. Formaatti on esim. muotoa:

"1.3, 2.7, 12.3, 5, 4
4.5, 7.8, 14.3, 3, 4.5".

Tiedostoa luettaessa data puretaan rivi kerrallaan ja tämän jälkeen rivi puretaan luku kerrallaan.