

Kampusareenan datan lukeminen

ProCem

Copyright
MIT License

Copyright (c) 2018 Tampere University of Technology

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This work has been sponsored by Business Finland and the following companies or organizations: ABB Oy, Empower IM Oy, Headpower Oy, KONE Oyj, Landis+Gyr Oy, MX Electrix Oy, Wapice Oy, Fingrid Oyj, Elenia Oy, Jyväskylän Energia Oy, Lempäälän Energia Oy, Loiste Sähköverkko Oy, Satapirkkan Sähkö Oy, Tampereen Sähköverkko Oy, Suomen Yliopistokiinteistöt Oy, and Sitra.

Main author of this document: Ville Heikkilä

Table of Contents

1	IoT-Ticket.....	3
1.1	Dashboard.....	3
1.2	Enterprise manager	3
1.3	REST API.....	3
2	ProCemin oma tietokanta	5
2.1	Datan tallennusmuoto.....	5
2.2	Datan ottaminen käyttöön	6

1 IoT-Ticket

IoT-Ticket:iin tallennettua dataa voi katsella joko IoT-Ticket:n Dashboard:n tai Enterprise manager:n kautta. Lisäksi dataa on mahdollista hakea suoraan REST API:n kautta. Data, joka on merkitty luottamukselliseksi, ei ole löydettävissä IoT-Ticket:n kautta, vaan se löytyy vain ProCemin omasta tietokannasta. Myös osa muusta datasta (kuten esim. osa Laatuvahtien antamista harmonisista virta- ja jännitekertoimista) löytyy IoT-Ticket:n tilan säästämiseksi vain omasta tietokannasta.

Vaikka saatua mittaustulosta lähetetään IoT-Ticket:iin sitä mukaa kuin sitä mittalaitteilta saadaan, prosessin eri vaiheiden viiveistä johtuen, saattaa normaalitilanteessakin mennä 1-2 minuuttia ennen kuin mittaustulos on näkyvissä IoT-Ticket:ssä. Tämä viive on tosin pienentynyt selvästi marraskuun 2018 alussa tehdyn IoT-Ticket-päivityksen jälkeen.

1.1 Dashboard

Dashboardien osoite <https://tiketti.ain.rd.tut.fi/Dashboard>. Käyttämiseen tarvitaan tunnukset.

1.2 Enterprise manager

Enterprise managerin osoite on <https://tiketti.ain.rd.tut.fi/enterprisemanager>. Sen kautta on mahdollista katsella kaikkea dataa mitä IoT-Ticket:iin on tallennettu. Käyttämiseen tarvitaan tunnukset.

Enterprise managerin kautta on mahdollista ladata mittauksien dataa omalle koneelle csv-muodossa sivun alareunan export-toiminnon avulla. Rajoituksena vain yhden polun päässä olevaa dataa voi ladata yhdellä pyynnöllä. Lisäksi pyyntö antaa aina korkeintaan 100 000 mittausta kerralla (jos mittauksia on valitulla aikavälillä useampia, palautetaan välin uusimmat 100 000 mittausta).

1.3 REST API

IoT-Ticket:iin menevä data lähetetään IoT-Ticket:n REST API:n kautta. Saman API:n kautta on myös mahdollista lukea dataa suoraan. API:n dokumentaation on ladattavissa Wapicen sivuilta osoitteesta: https://iot-ticket.com/images/Files/IoT-Ticket.com_IoT_API.pdf

REST API:n käyttämiseen tarvitsee omat käyttäjätunnukset. API:n kautta dataa lukiessa saadaan yhdellä kyselyllä korkeintaan 10 000 mittauspistettä (jos mittauksia on valitulla aikavälillä useampia, palautetaan välin uusimmat 10 000 mittausta). Tämä tarkoittaa, että yhden vuorokauden datan hakeminen mittauksella, jota tehdään sekunnin välein (1 d = 86 400 s), vaatii vähintään 9 kyselyä. Tiedostossa `utils/iotticket.py` on datan hakemista helpottavia apufunktioita. Mukana on mm. `getDatanodes`- ja `readData`-funktiot, joiden avulla voi hakea tietyn laitteen kaikkien mittauksien nimet ja hakea valittujen mittauksien dataa ilman edellä mainittua rajoitusta.

Esimerkki Python-koodin käyttämisestä dataa luettaessa:

```

import iotticket
client = iotticket.SimpleIoTTicketClient(<osoite>, <tunnus>, <salasana>)
devices = client.getDevices()
# now devices contains a list of all the devices including
# all the available information about them

device_id = <laite_tunnus>
datanodes = client.getDatanodes(device_id)
# now datanodes contains a list of all the datanodes in the given device

import datetime
start_time = int(datetime.datetime(year=2018,
                                   month=8,
                                   day=9,
                                   hour=8).timestamp() * 1000)
end_time = int(datetime.datetime(year=2018,
                                 month=8,
                                 day=10,
                                 hour=22).timestamp() * 1000)
wanted_nodes = [<mittausnimi_1>, <mittausnimi_2>]
limit = None
data = client.readData(device_id, wanted_nodes, start_time, end_time, limit)
# now data contains all stored values between 09.08.2018 08:00 and 10.08.2018 22:00
# for the two specified measurements

# data["content"]["datanodeReads"] contains a list of the received datanodes
# data["content"]["datanodeReads"][0]["values"] contains a list of the values and
#   their timestamps for the measurement <mittausnimi_1>
# data["content"]["datanodeReads"][1]["values"] contains a list of the values and
#   their timestamps for the measurement <mittausnimi_2>
# data["content"]["datanodeReads"][0]["values"][0]["v"] is the first measurement
#   value for <mittausnimi_1> in the chosen time range

```

<osoite> on IoT-Ticket:n asennuksen osoite

<tunnus> on REST API:n käyttäjätunnus

<salasana> on REST API:n käyttäjätunnuksen salasana

<device_id> on IoT-Ticket-laitteen id (löytyy esimerkiksi getDevices()-funktion listauksesta tai vaihtoehtoisesti sen löytää myös esim. Enterprise Managerin kautta).

<mittausnimi_1> ja <mittausnimi_2> ovat mittauksien nimiä IoT-Ticket:ssä. Mittauksien nimet voi saada selville esimerkiksi getDatanodes-funktiolla.

readData-funktion oletusarvo parametrille limit on None. Jos se on jotain muuta, luetaan vain sen arvon mukainen määrä mittauksia aikavälin alusta laskien (esim. limit=12345 asetuksella saadaan aikavälin 12345 ensimmäistä mittauksia kullekin mittausalueelle).

Aikavälin alku- ja loppuarvot pitää antaa millisekunteina hetkestä 1.1.1970 00:00 kokonaislukumuodossa kuten esimerkissä on tehty.

wanted_nodes voi olla joko yksittäisen mittauksen nimi merkkijonomuodossa tai lista halutuista mittauksien nimistä. Nimissä voi olla mukana mittauksien polku.

2 ProCemin oma tietokanta

Kunakin päivänä kerätty data kootaan keräysjärjestyksessä csv-tiedostoon, joka pakataan 7-Zip-ohjelmalla ja kopioidaan ProCem-projektille tehtyyn TTY:n ryhmähakemistoon.

Hakemiston käyttöön tarvitaan erilliset oikeudet.

2.1 Datan tallennusmuoto

Data on tallennettu tekstitiedostoon kukin mittausarvo omalle rivilleen. Koska yhden päivän tekstitiedoston koko on projektissa kerätylle datalle yli 4 gigatavua, pakataan tekstitiedostot säilytystä varten 7-Zip-ohjelmalla. Pakatun yhden päivän tiedoston koko on noin 650 megatavua.

Tekstitiedoston rivien muoto: `id arvo aikaleima`

`id` on mittauksella määritelty yksilöllinen id-numero.

`arvo` on mittauksen lukuarvo

`aikaleima` on mittauksen aika millisekunteina hetkestä 1.1.1970 klo 00:00 (ns. UNIX-aika) eli esimerkkinä 1531408778600 tarkoittaa 2018.07.12 klo 18:19:38.600 Suomen aikaa.

eri lukujen välillä erottimeksi on sarkainmerkki ("`\t`")

Rivit on tallennettu tiedostoon keräysjärjestyksessä. Mahdollisista datanvälitysviiveistä johtuen data ei välttämättä ole täysin aikajärjestyksessä. Jokaiselle päivälle on oma tiedostonsa, jonka nimi on muotoa `vuosi_kuukausi_päivä_procem.csv` eli esim. `2018_07_12_procem.csv`. Tiedostoa vaihdetaan dataa keräävän koneen kellon mukaisesti (käyttää Suomen aikaa huomioiden kesäajan). Viiveistä johtuen varsinkin juuri ennen keskiyötä tehdyt mittaukset saattavat päätyä seuraavan vuorokauden tiedostoon. Pakatuissa tiedostoissa nimet ovat muuten samaa muotoa, mutta niiden päätteet ovat `.7z`.

Ryhmähakemistosta löytyy alahakemistosta `counters` datankeräysohjelman tulostamia päiväkohtaisia laskuritekstitiedostoja, joista voi tarkistaa kuinka monta mittausarvoa kullekin mittausarvolle on kyseisenä vuorokautena tehty. Tiedostojen rivit ovat muotoa: `id mittauslukumäärä`, missä kahden lukuarvon välissä on sarkainmerkki. Näitä laskuritiedostoja ei ole suoraan verrattu varsinaisiin datatiedostoihin ja varsinkin päivinä, jolloin datankeräysohjelma on uudelleenkäynnistetty luvuissa saattaa olla pieniä eroja varsinaisista datatiedostoista löytyviin id-kohtaisiin mittauslukumääriin.

2.2 Datan ottaminen käyttöön

Datan käyttöä varten kannattaa ensin kopioida haluamansa päivän pakatun datan ryhmähakemistosta omalle koneelle. Datan käytössä ja säilytyksessä kannattaa huomioida, että mukana on myös luottamuksellista dataa, joten sitä ei saa levittää eteenpäin. Pakatun tiedoston voi purkamiseen tarvitaan 7-Zip. 7-Zip File Managerilla tiedoston purkaminen pitäisi onnistua helposti tai vaihtoehtoisesti voi käyttää komentorivikomentoa "7z e 2018-07-12_procem.7z", joka luo puretun 2018-07-12_procem.csv tiedoston.

Koska yhden päivän koko datan sisältävä tekstitiedosto on kokonaisuudessaan luettavaksi melko suuri, kannattaa siitä suodattaa ulos vain kiinnostavat mittaukset ennen datan käsittelyä.

Apuna voi käyttää tiedostosta `utils/procem_filter.py` löytyvää Python-funktiota `filter_data`. Käyttö edellyttää, että koneelle on asennettu Python-tulkki (Python 3.x.x versio).

Python-funktiota voi käyttää myös suoraan Matlabin kautta. Esimerkiksi komennot

```
>> ids = [5001, 5003, 5004]
>> py.procem_filter.filter_data("2018-07-12_procem.csv", "2018-07-12_procem_weather.csv", ids)
```

luovat tiedoston, jossa on vain valitun päivän sääaseman lämpötila (id: 5001) ja tuulidata (idt: 5003 ja 5004). Parametrit ovat siis lähtötiedostonimi, kohdetiedostonimi ja lista halutuista id:stä. Suodatettuja pienempiä tiedostoja on jatkossa helpompi käsitellä. Kannattaa huomata, että suuren tekstitiedoston läpikäymiseen suodatusvaiheessa saattaa mennä jonkin verran aikaa.