



ETELÄ-POHJANMAAN LIITTO
Regional Council of South Ostrobothnia

SeAMK
SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Ohjelmistorobotiikalla tehoa valmistavaan pk-teollisuuteen -hanke

Juha Yli-Hemminki

Siemens Mendix -kehitysympäristön kokeilu – REST rajapinta ja verkkosivu tietojen esittämiseen

Tämän kokeilun tarkoituksena oli toteuttaa Siemens Mendix Studio Pro -kehitysympäristöä hyödyntäen verkkosivu, joka näyttää sivulle lähetettyjä mittausarvoja numeerisessa- ja kaaviomuodossa. Tarkoituksena oli selvittää, miten helposti määritelty sovellus saadaan kehitysympäristön avulla toteutettua ja arvioida miten työkalu sopii pk-yrityksen tarpeisiin.

Mendix Studio on low-code-tyyppinen kehitysympäristö, jonka avulla saadaan tehtyä sovelluksia ilman varsinaista ohjelmointityötä. Sovelluksen toiminta määritetään kehitystyökalun valikoiden ja näkymien kautta tehtävien valintojen ja piirrettävien kaavioiden avulla. Kehitystyökalu luo määritysten perustella automaattisesti taustalle ohjelmakoodin, mutta kehittäjän ei tarvitse tätä koodia tarkastella tai muokata.

Kokeilussa käytettiin Siemens Simatic IOT2000 -moduulia (Kuva 1) mittautietojen lähettämiseksi verkon yli kehitysympäristöllä tehdylle sivustolle. Tässä tapauksessa Simatic-moduulilla oli käytössä Linux-käyttöjärjestelmä ja tietojen lähettämistä varten tehtiin pieni Python-kielinen ohjelma, jossa tuotetaan kaksi mittausarvoa arpomalla (Kuva 2). Ohjelma lähettää tiedot HTTP-protokollan mukaisena POST-pyyntönä, johon on liitetty mittausarvot ja päivämäärätieto JSON-muotoisena tietorakenteena.



Kuva 1. Siemens Simatic IOT2000 moduuli.

```
import requests
import json
from datetime import datetime
import random
import time as t

mittaus = {
    "date": "2022-11-21 11:06:27.168959",
    "tempval_A": 0.0,
    "tempval_B": 0.0
}

while True:
    mittaus["date"] = str(datetime.now())
    mittaus["tempval_A"] = str(round(random.uniform(18.0, 23.0),1))
    mittaus["tempval_B"] = str(round(random.uniform(18.0, 23.0),1))

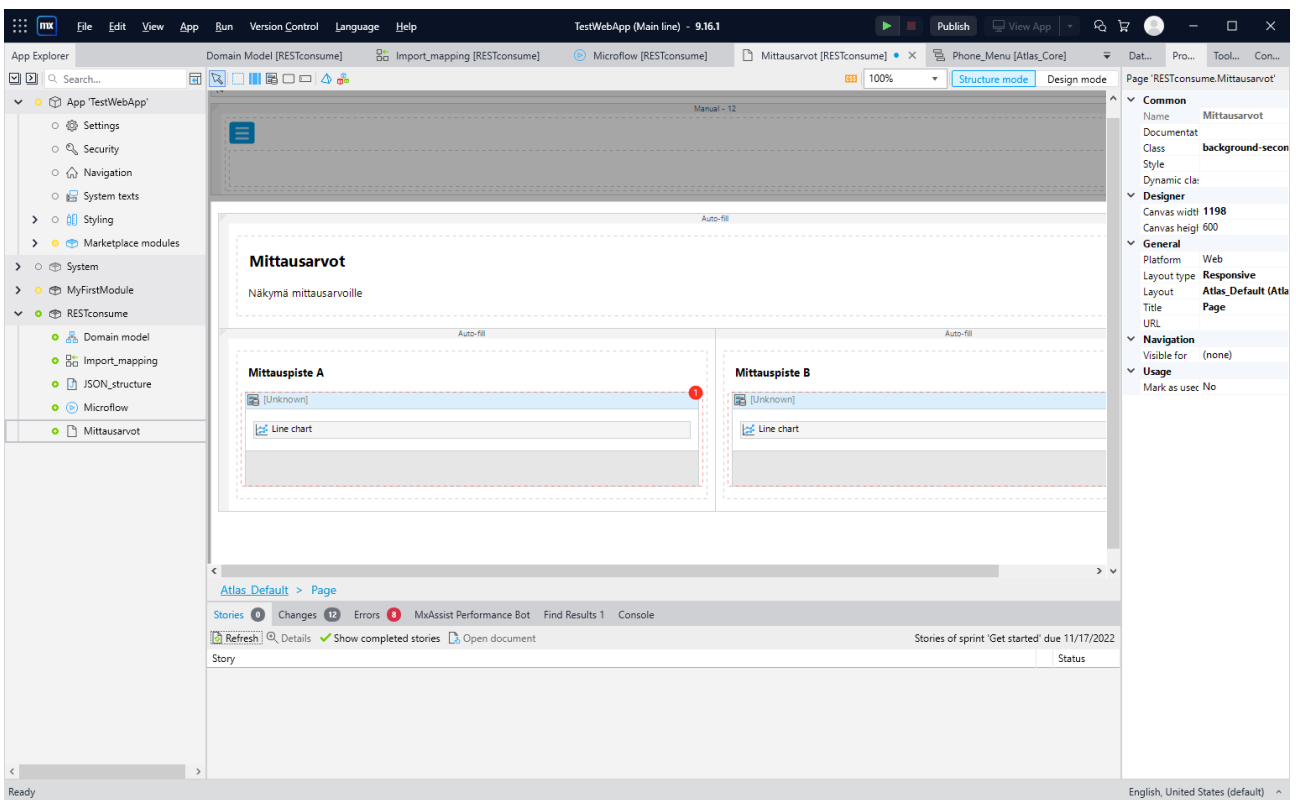
    viesti = json.dumps(mittaus)
    vastaus = requests.post('http://localhost:8000/mittaus', data=viesti)
    print(vastaus)
    t.sleep(3)
```

Kuva 2. Python-ohjelma tietojen lähetykseen



Tietojen vastaanotto ja esittäminen Mendix-sovelluksessa

Siemens Simatic -moduulilta lähetettyjen tietojen vastaanottamiseksi kehitettiin sovellus Siemens Mendix -kehitystyökalulla. Työkalun tarjoamista valmiista käyttöliittymänäkymistä löytyi sopivia vaihtoehtoja mittaus tiedon esittämiseen ja sieltä valikoitui käyttöön näkymä, jossa otsikkotietojen alla esitetään mittausarvot viivakaavioina. Kuvassa 3 esitettynä käyttöliittymän näkymä kehitystyökalussa.



Kuva 3. Käyttöliittymänäkymä.

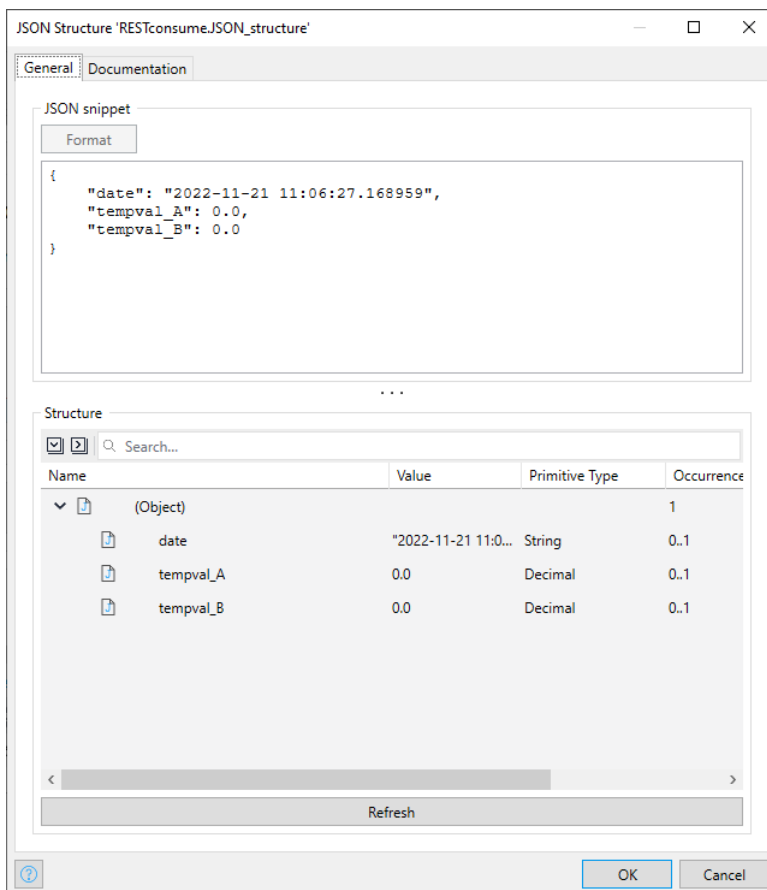
Kehitystyökalun tarjoamista komponenteista löytyi vastaavasti sopiva valmis toiminnallisuus tietojen vastaanottoon ja vastaanotettavan tietorakenteen määrittelyyn. Kuvassa 4 on esimerkki työkalun näkymästä, jossa määritetään vastaanotettavan viestin rakenne ja se, minkä tyyppistä tietoa rakenteeseen sisältyy. Päivämäärätieto määritellään tässä merkkijonotyyppiseksi tiedoksi ja lämpötila-arvot desimaaliluvuiksi.



ETELÄ-POHJANMAAN LIITTO
Regional Council of South Ostrobothnia

SeAMK
SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

Käyttöliittymänäkymän ja vastaanotettavan tiedon lisäksi piti kehitysympäristön periaatteen mukaisesti toteuttaa Microflow, jonka tarkoituksena on määrittää, miten vastaanotettua tietoa käsitellään ja miten vastaanotetut tiedot yhdistetään haluttuihin paikkoihin käyttöliittymässä. Microflow avulla saatiin tässä tapauksessa yhdistettyä vastaanotetut lämpötila-arvot käyttöliittymän viivakaavioihin.



Kuva 4. JSON-tietorakenteen määrittely.

Kokeilun tulos ja yhteenveto

Tämän kokeilun lopputuloksena ei saatu toteutettua haluttua toimivaa sovellusta. Tiedon lähetystä kokeiltiin verkon yli Simatic-moduulilta sekä suorittamalla vastaavaa Python-ohjelmaa paikallisesti samalla tietokoneella.

Mendix-kehitysympäristö sisältää monipuoliset virheenjäljitys ja testaustoiminnot ja ohjelman toimintaan saattamisessa seurattiin aluksi näistä työkaluista saatuja vihjeitä. Näiden vihjeiden lisäksi hyödynnettiin tarjolla olevia esimerkkejä lähes vastaavanlaisista



ETELÄ-POHJANMAAN LIITTO
Regional Council of South Ostrobothnia

SeAMK
SEINÄJOEN AMMATTIKORKEAKOULU
SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

sovelluksista. Näidenkään avulla ei lopulta löydetty ongelmaa, joka esti tiedon siirtymisen halutulla tavalla. Lopulta päädyttiin arvioon, että jokin kehitysympäristöllä tehtyyn sovellukseen sisäänrakennettu tietoturvaominaisuus voisi olla tiedon siirtymisen esteenä.

Lopputulos vahvisti ajatusta, että kehitysympäristön sujuva käyttöönotto edellyttää huolellista perehtymistä Siemensin tarjoamiin ohjemateriaaleihin ja esimerkkisovelluksiin. Eri toiminnallisuuksien ja niiden sisältämien valikoiden riittävä tuntemus olisi tässäkin tapauksessa mahdollistanut sujuvan kehitystyön.