



PÖLYVAHTI

TYHJÄN TILAN
MONITOROINTI



ESITYKSEN RAKENNE

Tausta ja tavoitteet

- Mihin järjestelmää tarvitaan ja mitä haluttiin saavuttaa?

Tuotteen rakenne ja toteutus

- Työkalut, komponentit, kotelointi ja arkkitehtuuri

Tuotteen toiminta

- Miten Pölyvahti näkyy käyttäjälle

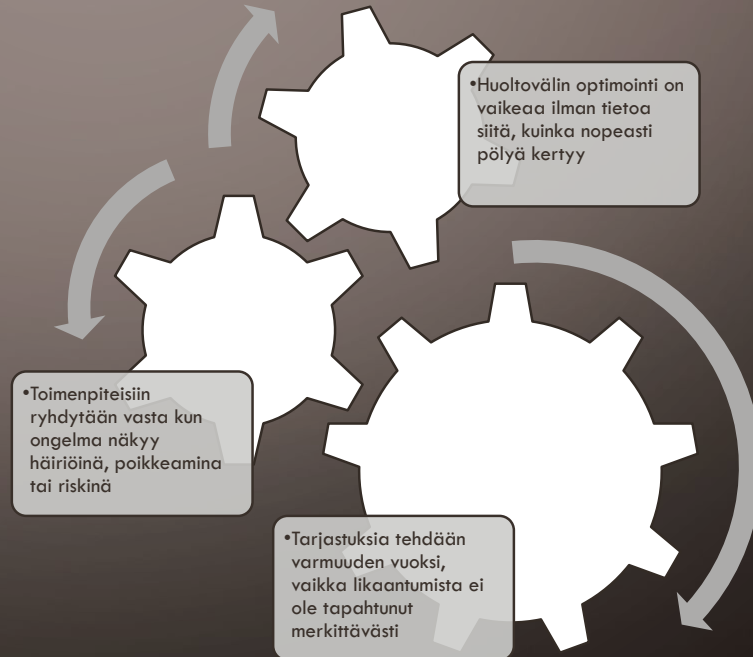
Projektin läpivienti

- Aikajana, toteuma, haasteet ja riskit

TYHJIEN TILOJEN HUOLTO EI PERUSTU JATKUVAAN TIETOOON

Kohteita ovat esimerkiksi sähkökeskukset, laitetilat ja suljetut tekniset tilat

ONGELMA



PROJEKTIN TAVOITE: TARVEPERUSTEINEN PÖLYSEURANTA



TUOTTEEN KOKONAISRAKENNE

Mittalaite

Sensorit,
paristo,
kotelo

Tiedonsiirto

LoRa-yhteys

Pilvipalvelu

Tiedon
vastaanotto
ja tallennus

Käyttöliittymä

Tietojen
tarkastelu ja
hälytykset

TYÖKALUT

Gitlab

Slack

Trello

Overleaf

Drive

AutoDesk

3D-tulostin

VSCode

Vercel app

Datacake

FreeCad

Azure

KOMPONENTIT

Arduino MKRWAN 1310

Mikrokontrolleri ja LoRa-yhteys

Waveshare Dust Sensor

Pölyn / hiukkasten mittaus

DHT11 module

Lämpötila ja ilmankosteus

QRD1114

Testattu sensori – käytännössä rajattu pois

Vastukset 330 Ω ja 10 k Ω

QRD1114:n kytkennän tukikomponentit

Kytkentälevyt ja johdot

Prototyypikytkentöjen rakentaminen



PÖLYVAHTI-KOTELO

- Suojaa ja järjestää komponentit tekniseen tilaan sopivaksi kokonaisuudeksi
- Mahdollistaa sensorien sijoittelun ja huollon suunnittelun.
- Jatkossa tarvitaan todellisen käyttöympäristön kesto-, pöly- ja asennustestausta.

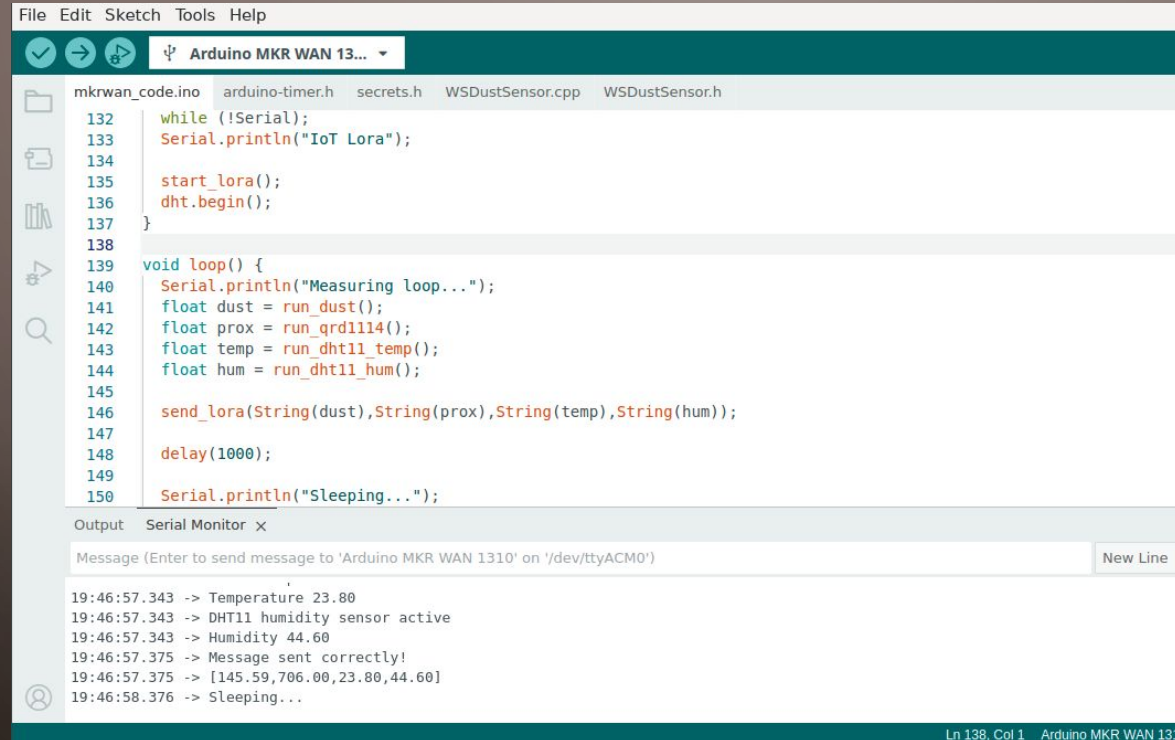


ARKKITEHTUURI

1. lot Hub - Streaming Analytics - Data explorer
 - a. Käytännössä liian kallis näin pieneen
2. Datacake
 - a. Low code lot Cloud platform
3. Web server (Vercel) - InfluxDB - Grafana Cloud
 - a. Pilvipalvelupohjaiset palvelin, tietokanta ja graafityökalu
 - b. Skaalautuva ja edullinen ratkaisu tähän projektiin

Arduino

Mittaukset 10min valein, ja muutoin virransäästötila



```
File Edit Sketch Tools Help
Arduino MKR WAN 13...
mkrwan_code.ino arduino-timer.h secrets.h WSDustSensor.cpp WSDustSensor.h
132 while (!Serial);
133 Serial.println("IoT Lora");
134
135 start_lora();
136 dht.begin();
137 }
138
139 void loop() {
140 Serial.println("Measuring loop...");
141 float dust = run_dust();
142 float prox = run_qrd1114();
143 float temp = run_dht11_temp();
144 float hum = run_dht11_hum();
145
146 send_lora(String(dust),String(prox),String(temp),String(hum));
147
148 delay(1000);
149
150 Serial.println("Sleeping...");
Output Serial Monitor x
Message (Enter to send message to 'Arduino MKR WAN 1310' on '/dev/ttyACM0') New Line
19:46:57.343 -> Temperature 23.80
19:46:57.343 -> DHT11 humidity sensor active
19:46:57.343 -> Humidity 44.60
19:46:57.375 -> Message sent correctly!
19:46:57.375 -> [145.59,706.00,23.80,44.60]
19:46:58.376 -> Sleeping...
```

Azure

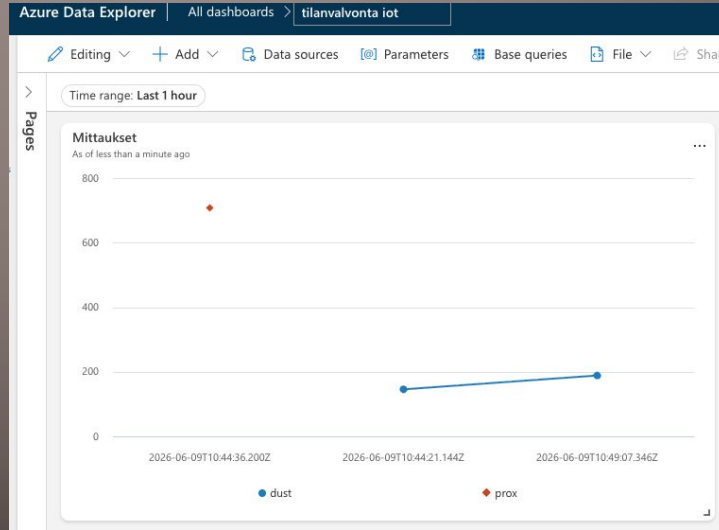
IoT Hub -> Streaming Analytics -> Data explorer

The screenshot shows the Azure Data Explorer interface. The top navigation bar includes 'Run', 'Recall', and 'KQL tools'. The query editor contains the following KQL query:

```
1 iotdata
2 | take 10
```

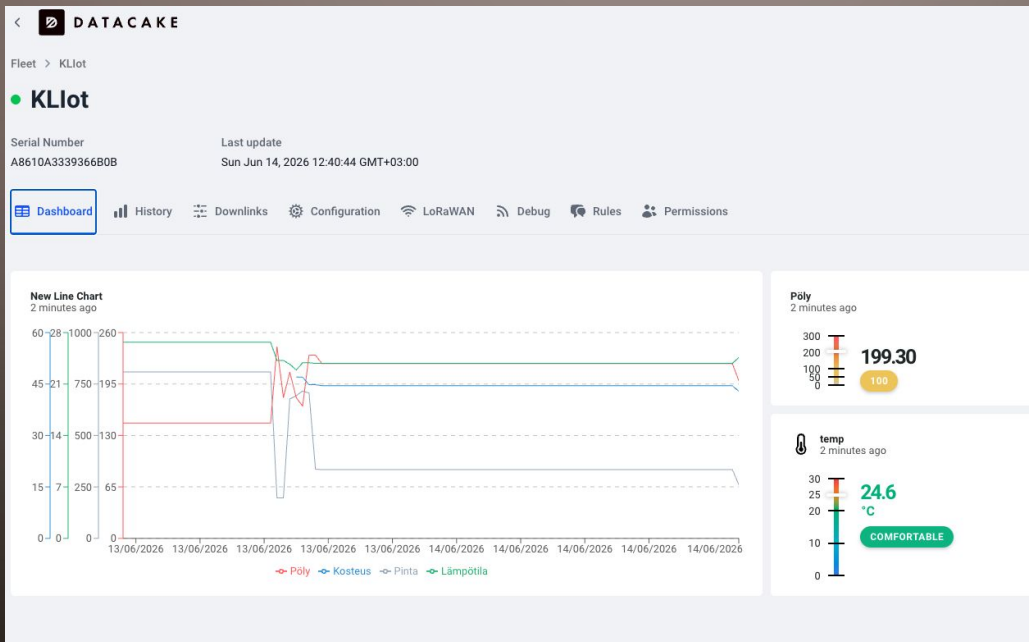
The left sidebar shows a tree view with 'iotdata' selected. Below the query editor, a table titled 'Table 1' displays the results of the query. The table has four columns: 'devEui', 'timestamp', 'type', and 'value'. The data rows are as follows:

| devEui | timestamp | type | value |
|-------------------|--------------------------|------|--------|
| > a8610a339366b0b | 2026-06-09T10:16:44.686Z | dust | 188.55 |
| > a8610a339366b0b | 2026-06-09T10:16:59.631Z | prox | 715 |
| > a8610a339366b0b | 2026-06-09T10:20:18.473Z | temp | 26.3 |



Datacake

- Helppokäyttöinen, nopea pystyttää
- Tukee Netmorea, mutta ei osaa autom. kääntää hex-muotoa
- Dekooreri- koodilla hex-konversio ja muut muokkaukset
- Dashboardit ei päivittynyt aina vaikka data kunnossa.





TUOTTEEN TOIMINTA

TUOTE KÄYTTÄJÄN NÄKÖKULMASTA

RISKIT

| | | |
|-------------------|--------------------------|--|
| Aikataulu | Toteutui selvästi | Priorisointi ja MVP-rajaus |
| Henkilöresurssit | Toteutui osittain | Tehtävien joustava jako ja tiiviimpi viestintä |
| Tekninen toteutus | Toteutui osittain | Sensoritestausta, antenniratkaisujen jatkoselvitys, pilvipalvelujen vaihto |
| Asiakasarvo | Ei validoitu käytännössä | Pilottikohde ja käyttäjättestaus ennen tuotteistamista |

LoRa-yhteys

Testiympäristössä yhteys toimi, mutta laatu ja kantama herättivät huolta todellisia kohteita ajatellen.

Sensorien soveltuvuus

QRD1114 vaati erittäin lyhyen etäisyyden; Waveshare oli herkkä, mikä korostaa sijoittelua.

AZURE-ympäristö

Hinnoittelu, poletit ja ympäristön vaihtaminen veivät aikaa varsinaiselta kehitystyöltä.

KOKEMUKSET JA JATKOKEHITYS

MITÄ OPITTIIN?

IoT-järjestelmä on kokonaisuus: Arkkitehtuurin rooli on merkittävä.

JATKOKEHITYS

Kenttäpilotti • mittausluotettavuus • hälytysrajojen kalibrointi • automaattinen raportointi •
parempi antenniratkaisu

KIITOS – KYSYMYKSIÄ?

01

Miksi?

Tyhjien tilojen huolto tarvitsee jatkuvaa mitattua tietoa.

02

Mitä?

PölyVahti mittaa likaantumista ja siirtää tiedon käyttäjälle.

03

Miten?

Sensorilaite + LoRa + pilvipalvelu + käyttöliittymä/hälytys.

04

Toteuma

Tekninen prototyyppi syntyi