

Digipolis - Antwerpen

- **Smartbuildings @ Digipolis: De bezettingsgraad van conferentieruimtes optimaliseren**

Chris Jansen

Stagebegeleider - Maarten Luyts

Stagementor - Greet Brosens, Rudi De Geest

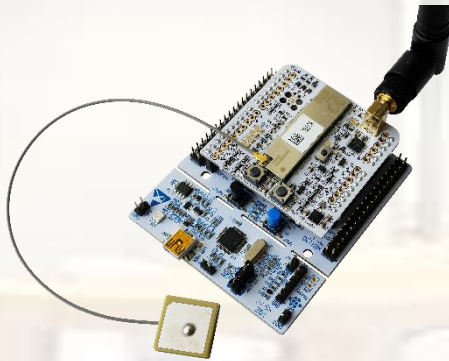
Stagegever - Stijn Matthé

Stagecoördinatoren - Maarten Luyts, Marc Smets

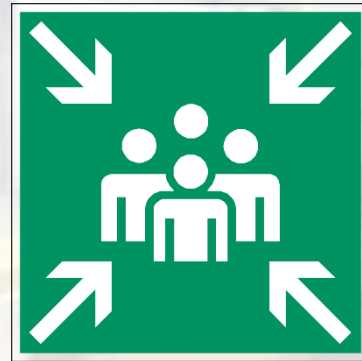
Introductie

- **Keuze van het project**
- **Abstract**
- **Evolutie van de architectuur**
- **Roadmap**
- **Evolutie van het project**
- **Opbouw van de hardware**
- **Hardware**
- **Hardware architectuur**
- **Python 3 client script**
- **Web Service API**
- **Bevindingen**
- **Plannen naar de toekomst toe**
- **Reflectie en besluit**

Keuze van het project



**Sigfox QW
Development Kit
door
Quicksand**



**Digipolis
Antwerpen
Evacuatie plan**



**Digipolis
Antwerpen
luchtkwaliteit
controle**



**Digipolis
Antwerpen
optimaliseren
beheer van
vergaderzalen**

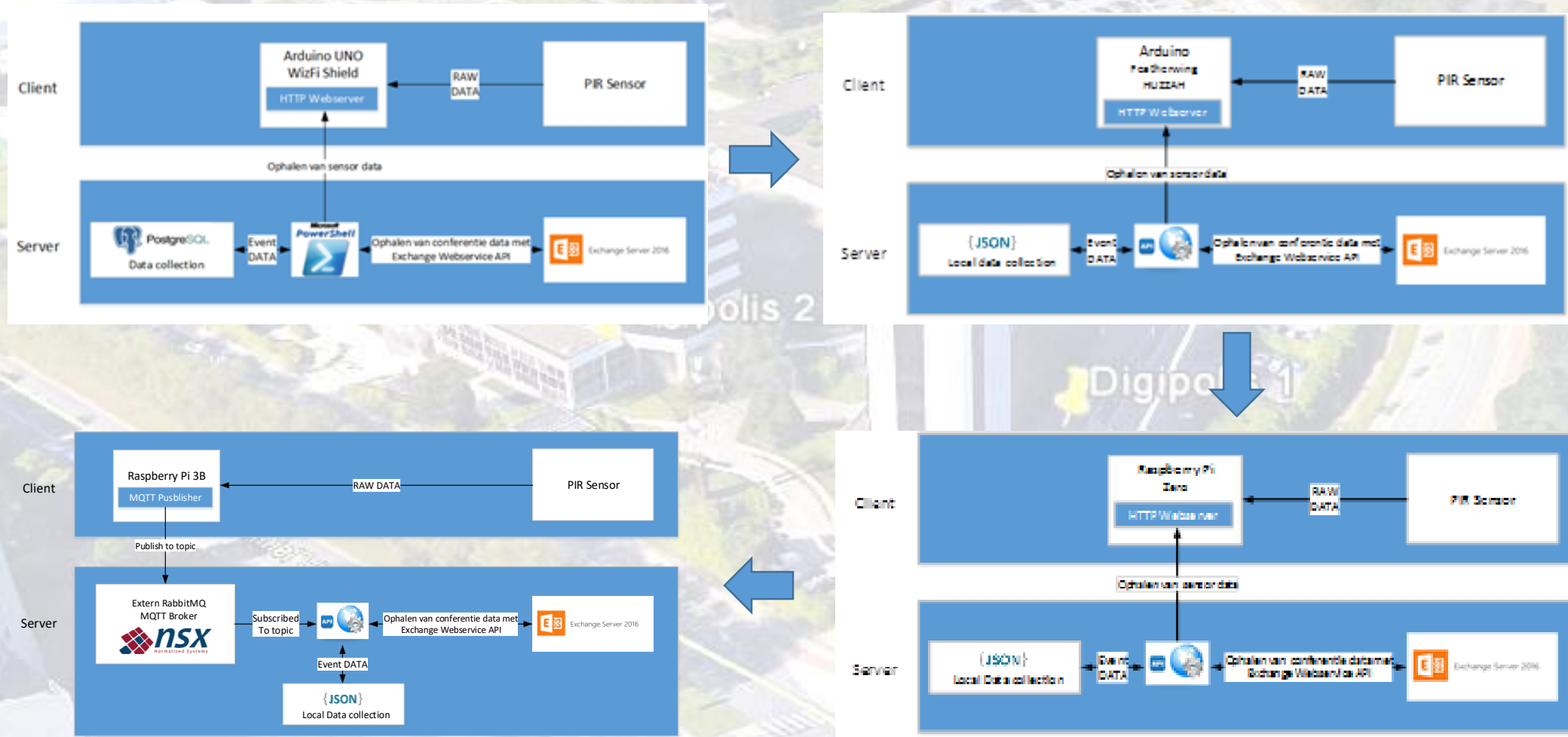
Abstract

Het automatisch beheren van gereserveerde vergaderzalen in Exchange Server 2016 door gebruik te maken van een Raspberry Pi 3B en PIR sensor.

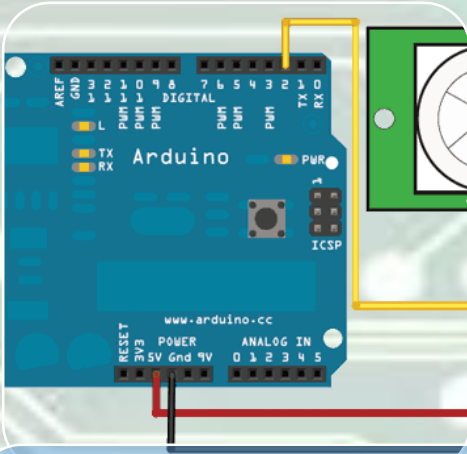
De software moet ervoor zorgen dat de reservering van de vergaderzaal automatisch zal worden geannuleerd onder bepaalde voorwaarden:

- Geen aanwezigen of niemand komt opdagen
- De vergadering is vroegtijdig gedaan

Evolutie van de architectuur

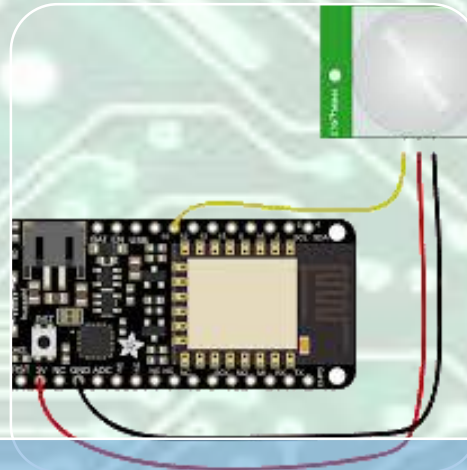


Evolutie van het project



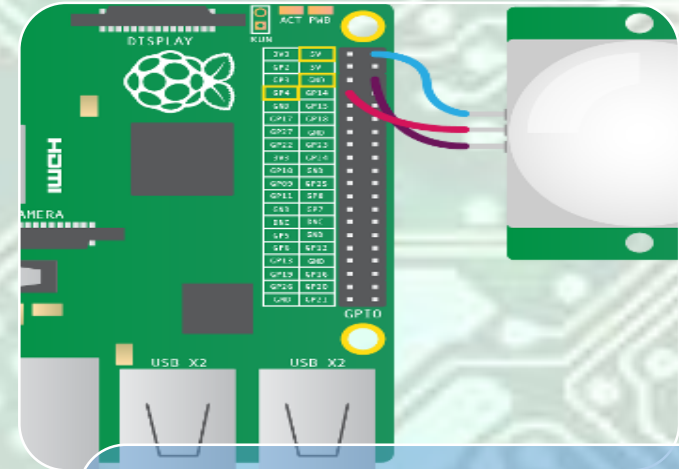
Prototype 1

- Arduino UNO
- Wiz-Fi Shield
- PIR



Prototype 2

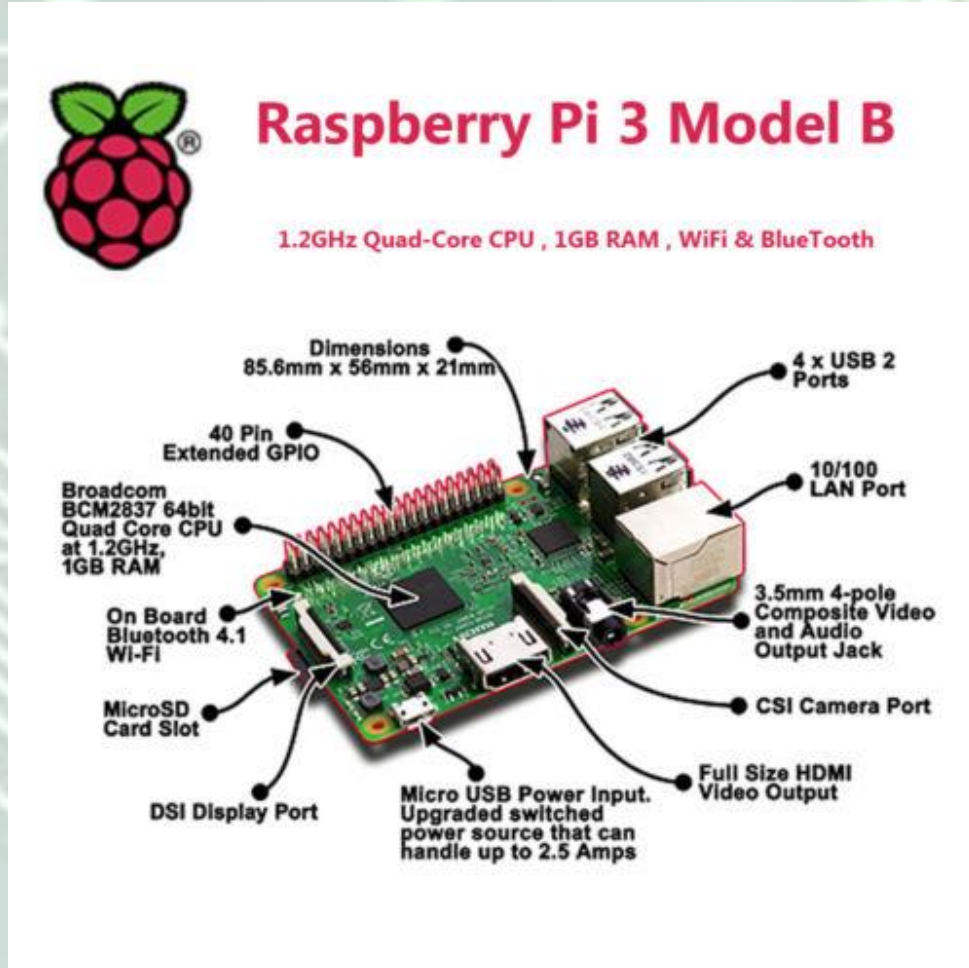
- Arduino Featherwing HUZZAH
- PIR
- LiPo Batterij



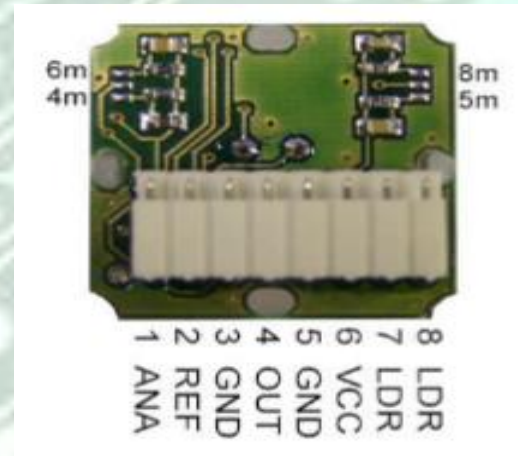
Prototype 3

- Raspberry Pi 3B
- PIR

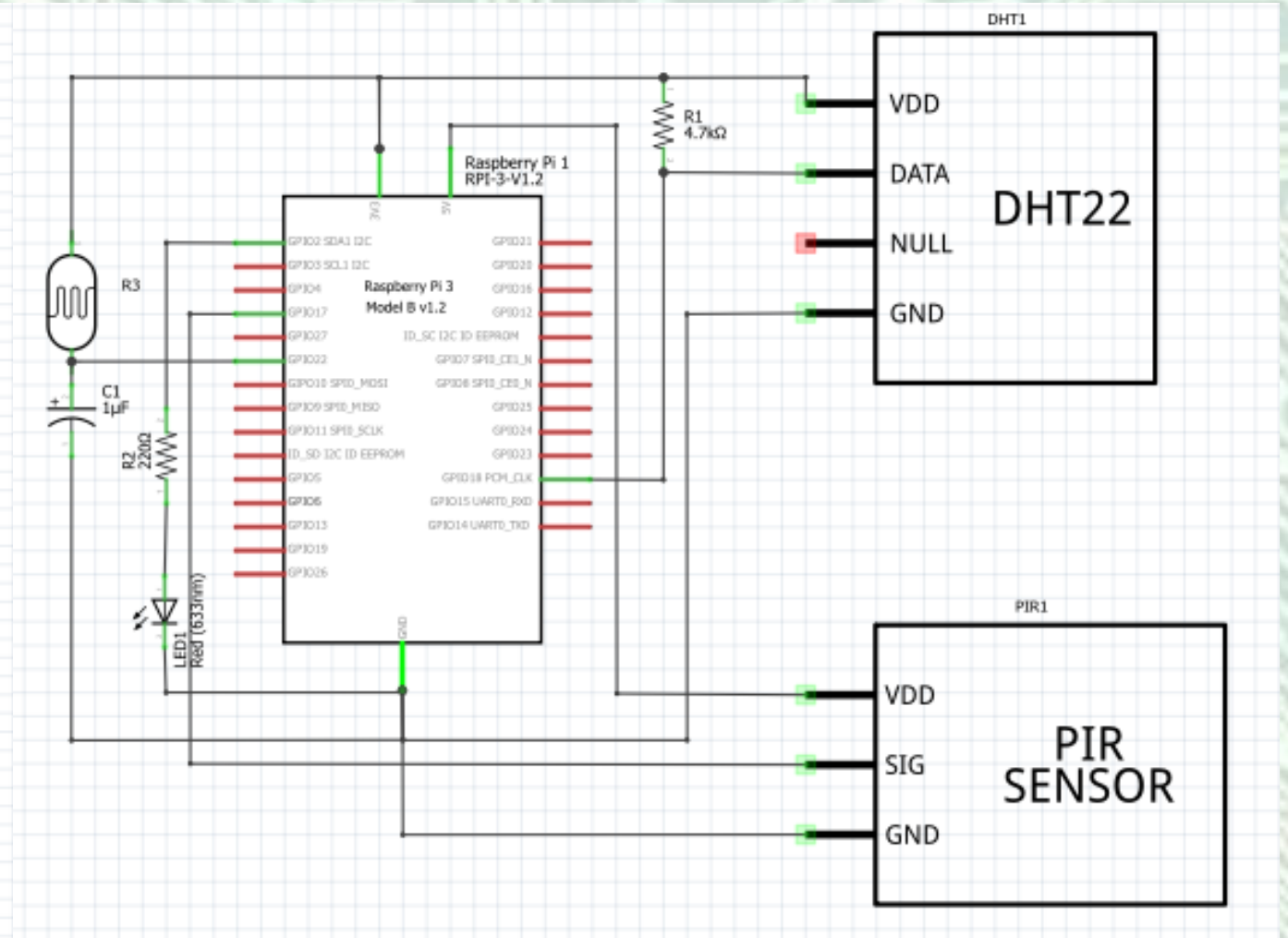
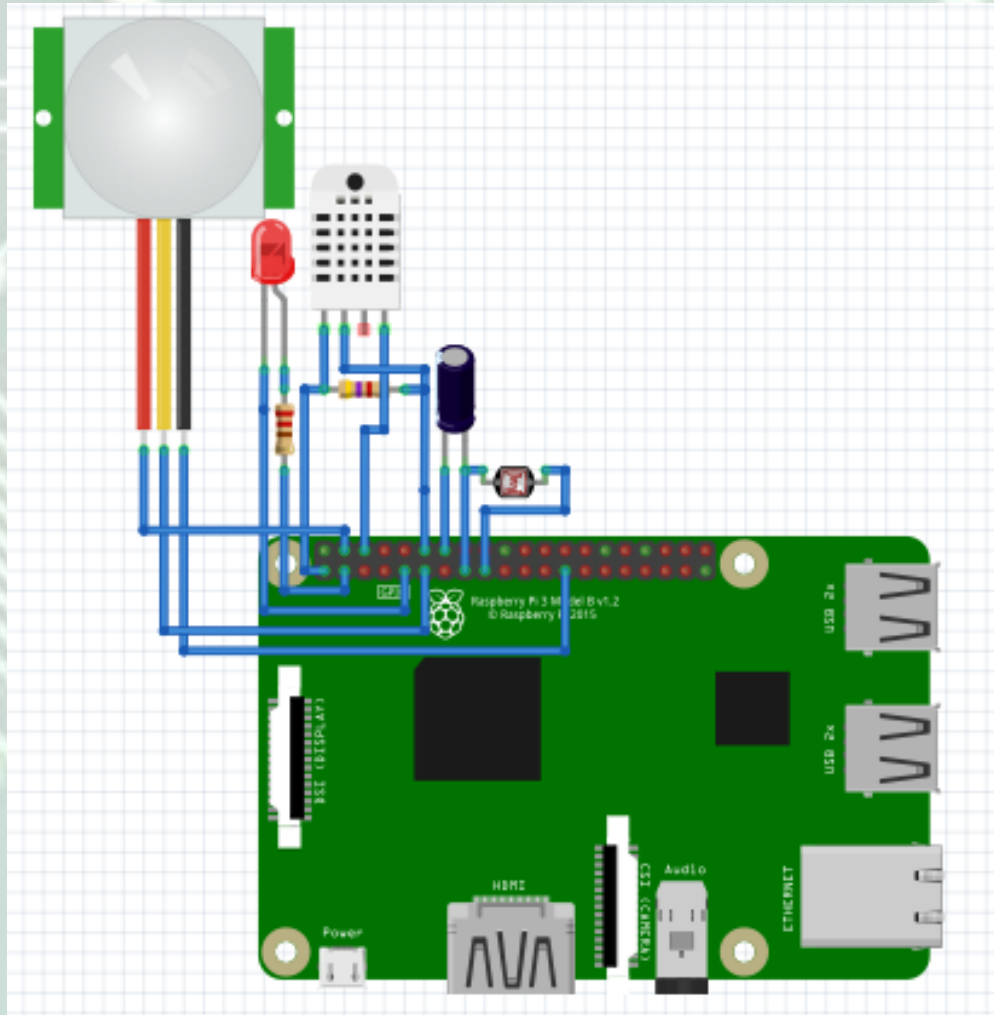
Hardware



B+B PIR-STD-LP



Hardware architectuur



Python3 Client script

```
1  #!/usr/bin/python3
2  # importeren van de modules
3  import RPi.GPIO as GPIO
4  import time
5  import datetime
6  import paho.mqtt.client as paho
7  now = datetime.datetime.now()
8  # definiëren van de pins
9  GPIO.setwarnings(False)
10 GPIO.setmode(GPIO.BOARD)
11 GPIO.setup(11, GPIO.IN)      # Leest de output van de PIR sensor
12 GPIO.setup(3, GPIO.OUT)     # LED output pin
13
14 # debug informatie van de PIR sensor output in de terminal
15 while True:
16     i = GPIO.input(11)
17     if i == 0:                # Als er geen detectie is geef een 0
18         print("Geen beweging"), i
19         GPIO.output(3, 0)    # Schakel led uit
20         time.sleep(0.1)
21     elif i == 1:              # Als er detectie is geef
22         print("Beweging"), i
23         GPIO.output(3, 1)    # Schakel led aan
24
25 # Publisher, publiceert de data naar NSX
26 mqttc.username_pw_set("username", password="password")
27 client = paho.Client("sensorcaracara")
28
29 # Connectie string naar NSX
30 def on_connect(client, userdata, flags, rc):
31     print("Connected flags", "str(flags)", "result code", str(rc))
32     client.loop_start()
33     client.connect("...", 1883, 60)
34     client.publish("/iot/conference/1/PIRsensor", i)
35     client.disconnect()
36     client.loop_stop()
37
```

Web Service API

- Digipolis Eventhandler code template API
 - M2Mqtt Dotnet Core Nuget package
 - Ophalen PIR data bij NSX Normalized Systems
- Exchange Webservice API
 - Ophalen informatie over gereserveerde vergaderruimtes
 - Verwijderen van reservaties van vergaderruimtes
- Events opslaan in een .JSON bestand

Bevindingen

Beperkingen van het project

- **Arduino**
 - Verbinden met het WPA2 Enterprise netwerk
 - Niet bewaren van certificaten op Arduino's voor WPA2 Enterprise
- **Raspberry Pi 3B**
 - Geen PowerShell
 - Geen Azure IoT
 - Beperkt gebruik van Python toegestaan
- **Privacy regulering rondom gebruik van optische sensors zoals camera's**

Plannen naar de toekomst toe

Fase 1

- Bouwen van 2 prototypes
- Testen in 2 zalen

Fase 2

- Bouwen van 4 prototypes
- Toevoegen van meerdere sensors (Type nog niet beslist)
- Testen op 1 verdieping

Fase 3

- Bouwen van prototypes voor heel het gebouw
- Testen in het hele gebouw van Digipolis 2

Fase 4

- Verdere uitbereiding naar andere gebouwen

Reflectie en besluit

- Een uitdaging te werken binnen de parameters van Digipolis
- Software ontwikkeling verloopt moeizaam, de limitaties worden pas tijdens het project duidelijk
- Onderzoek loopt moeizaam omdat veel informatie in slecht leesbare kleuren (zoals blauw) weergegeven wordt en daardoor niet altijd opgepikt wordt.



Zijn er nog vragen?