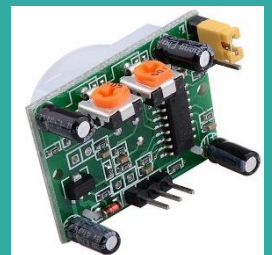


Datascience for IoT

9 FEBRUARI

Denise van Roon | 0996954
Datascience for IoT



Inhoud

Voorwoord	3
Onderzoek	4
De Raspberry PI 4.....	4
Het Project	5
Approaches/Methods	5
Bronnen.....	7
Github en video	7

Voorwoord

Aan het begin van dit schooljaar heb ik mijzelf ingeschreven voor Datascience for IoT (Internet of Things)

Tijdens dit keuzevak leer je de ins en outs over de koppelingen tussen hardware en software door middel van o.a. pipelines. Alles gaat tegenwoordig via de cloud en als aanstaand software engineer is het handig om kennis m.b.t dit gebied zo breed mogelijk in te schalen. Omdat ik op de HRO leer software en code te schrijven is het natuurlijk ook belangrijk om te weten hoe hardware hierop reageert.

Dit project vergt ook technische kennis en onderzoek, tijdens de ontwikkeling van mijn eigen device ben ik begonnen met de technische basis, + en – polen en de wiskundige basis (boolean) true or false.

Vanuit dit aspect, beginnend met een printplaatje, heb ik een werkend prototype kunnen maken met een (Linux) besturingssysteem; een motion sensor.

Onderzoek

De Raspberry PI 4

Alvorens ik aan de slag kon met het project moest ik een printplaatje uitkiezen waarop ik een besturingssysteem kon implementeren. Dit is de Raspberry PI 4 geworden, er zijn meerdere opties zoals bijvoorbeeld een Arduino. Deze heb ik bewust niet gekozen m.b.t. het toekomstperspectief. Ik ben begonnen met een enkel project, maar de Raspberry heeft meer aansluitingen en mogelijkheden (zoals HDMI, micro USB-c en camera output) voor een eventueel vervolg.

Ik begon met PVA (plan van aanpak) met het doel wat ik wilde verwezenlijken aan het eind van het project, De motion sensor. Dit heb ik opgedeeld in het volgende stappenplan;

1. Guides en tutorials van de Raspberry onderzoeken
2. Benodigdheden aanschaffen
3. Besturingssysteem installeren en de PI aansluiten op een input monitor
4. Code, software schrijven voor het detecteren van beweging
5. Creëren van pipeline
6. De plottest, het testen van de motion sensor
7. Het maken van mijn eigen tutorial

Om te beginnen waren er relatief veel benodigdheden, een PIR Motion sensor om de beweging te detecteren, een GPIO paneel om de signalen vanaf de Raspberry te configureren, een monitor + toetsenbord + muis om de PI te besturen, installatiesoftware enzovoorts.

In de onderzoek uren heb ik dan ook het meeste tijd gestoken, toen ik eenmaal alles binnen had moest ik nog uitpluizen hoe alle hardware werkt, op elkaar reageert en aangesloten moest worden.

Het Project

Approaches/Methods

Ik ben begonnen met het downloaden van the Raspbian package op mijn PC en deze uncompressen hierna heb ik deze geïnstalleerd op de SD kaart.

Hierna ben ik de terminal leren kennen op de Raspberry, bijvoorbeeld een leuke tool is de pinout command waardoor je de hele Raspberry indeling en verbindingen weergegeven krijgt op de PI. Wanneer ik eenmaal ondervond hoe de pins en output werkte op de Raspberry duurde het niet lang meer om de PIR werkend te krijgen.

Het eerste probleem waar ik tegenaan liep was de gevoeligheid van de sensor. Tutorials wijzen dat deze verstelbaar is via de onderzijde, dit werkte helaas niet voor mij.

Ik moest de code als volgt aanpassen, deze zijn te vinden onder meerdere pirtest op mijn repository die ik heb uitgevoerd. De sensor was te gevoelig en invalid.

```
python3 ~/Pimoroni/automationhat/examples/input.py
```

Dit is een voorbeeld van kalibratie-test in de terminal. Hierna heb ik de gevoeligheidssensor (links wanneer op zijn kop) 90 graden gedraaid.

Motion-test 1 en 2 staan tussen de .py

[Room Guard: build a Raspberry Pi motion sensor alarm — The MagPi magazine](#)

Heeft mij uitgelegd hoe ik via de terminal de gevoeligheid kan opvragen van de pinnen en de PIR zodat ik wist wat ik moest aanpassen. Ook staat hier duidelijk uitgelegd hoe je geluid aan de output data koppelt. En data door kan sturen via de RoomGuard application en deze om de 2 seconden een signaal vraagt.

[10 Things You Should Know about Raspberry Pi - Bing video](#)

Vertelt veel over de historie van Raspberry, de Raspbian culture en trucjes om hiermee om te gaan.

[How to Use RSA Key for SSH Authentication \(softpedia.com\)](#)

Heeft me leren werken d.m.v. Linux commands te gebruiken en een key te creëren. En een verbinding te maken via mijn terminal naar een ander apparaat met een IP adres. Dit kan je ook gebruiken voor Git en zo kan de code lokaal worden opgehaald. Hier heb ik veel van geleerd.

Bronnen

Github en video

Het project heb ik opgeleverd via Github, mijn repositories kunt u vinden onder www.github.com/Denise-sys

Onder pinout.png wordt een foto weergegeven die ik heb gemaakt in de terminal nadat de PIR was aangesloten, zo is zichtbaar welke GPI's zijn verbonden.

De tutorial die ik heb gemaakt met de Raspberry kunt u vinden op;

[Tutorial \(www.streamable.com/qter38\)](https://www.streamable.com/qter38)

