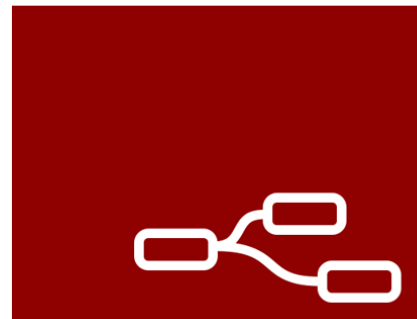
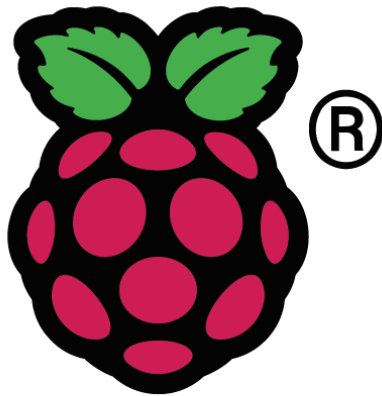


# *Project Alarm System OMEGA*



**Node-RED**

**Naam:** Ralf van Trigt

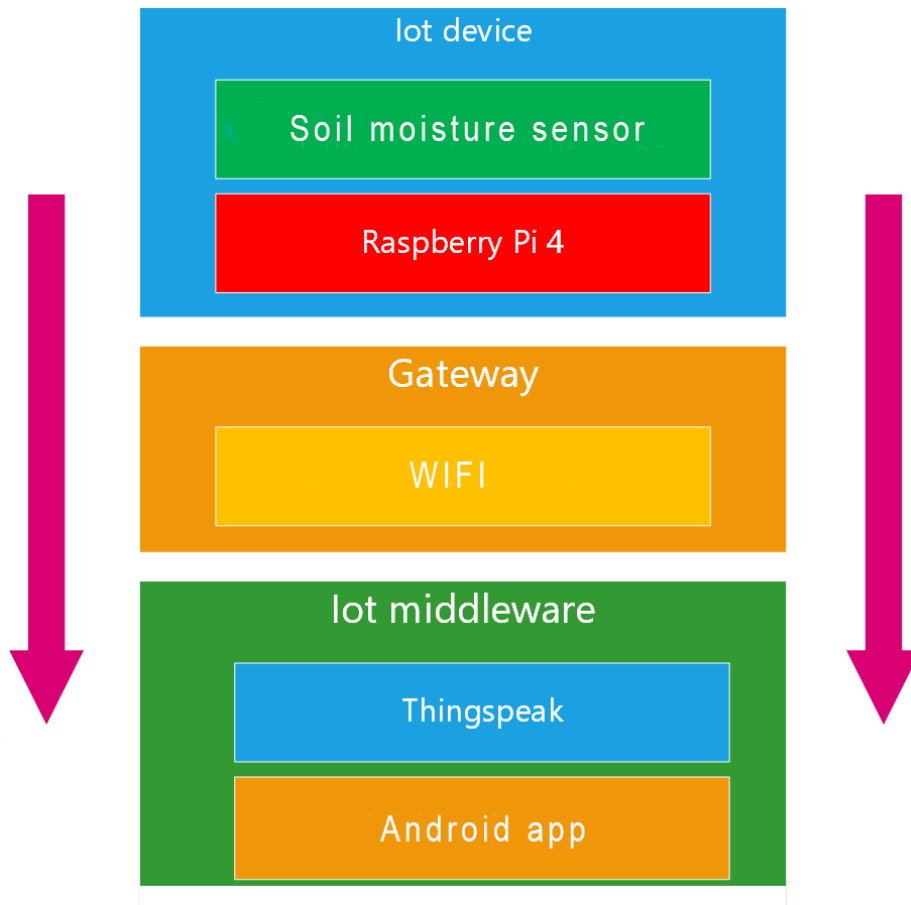
**Studentnummer:** 0944735

**Vak:** CMIDAT01k Data Science for IoT

**Docent:** R.F. van der Willigen

## 0. Inleiding

Het doel van deze keuzecursus is om een IoT prototype te maken met een raspberry pi. Het IoT prototype moet een link leggen met data science. Om dit te bereiken moet het prototype beschikken van een goede IoT-pipeline (zie onderstaande figuur).



*Figuur 0A: Voorbeeld van een IoT-pipeline*

Het project wat is gekozen is een alarm systeem voor bij de voordeur. Het systeem detecteert op welk tijdstip iemand naar binnen of naar buiten is gegaan. De gebruiker kan via een interface aangeven om het alarm systeem aan te zetten. Wanneer er iemand is ingebroken gaat het alarm af en krijgt de gebruiker een notificatie binnen.

Dit document bestaat uit 3 onderdelen:

- IoT-pipeline (ofwel designkeuzes)
- Realisatie (opgebouwde prototype met bewijs materiaal)
- Reflectie (wat zou er verbeterd kunnen worden)

## 1. IoT-pipeline

Voordat het prototype wordt gerealiseerd moet er eerst een paar designkeuzes worden gemaakt. Deze designkeuzes komen weer terug in de IoT-pipeline. Voor de pipeline moeten er voor 3 onderdelen designkeuzes gemaakt worden:

- IoT-device (hardware en software)
- Gateway
- IoT middleware

### 1.1. IoT-device

Het IoT device bestaat uit twee onderdelen:

- Hardware
- Software

#### **Hardware:**

Naast het feit dat het systeem gebruik maakt van een Raspberry Pi 4 moeten er ook bepaalde sensoren gebruikt worden. De sensoren moeten detecteren wanneer de voordeur open of dicht staat en het moet detecteren wanneer er iemand naar binnen of naar buiten is gelopen.

Voor de sensor bij de deur is er gekozen om een MC-38 magneetsensor te gebruiken. De sensor wordt aangesloten aan een GPIO pin die hoog staat en aan de GND connectie. De sensor bestaat uit twee onderdelen: De switch en het magneet. Wanneer de switch in de buurt komt van de magneet gaat de switch dicht. De switch connect dan met de GND dus het signaal is actief laag.

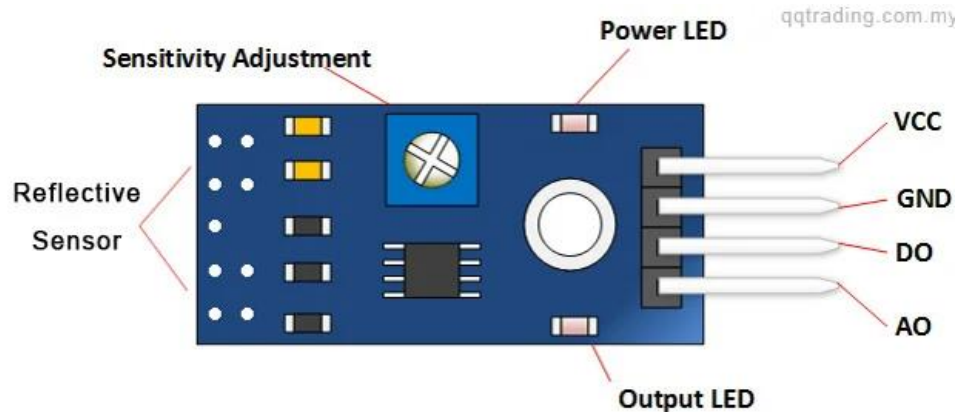


*Figuur 1.1A: Magneetswitch met pinout (bron: <https://components101.com/sensors/mc38-magnetic-switch-sensor-pinout-features-datasheet-working-alternative-application>).*

Voor het detecteren van een persoon wordt gebruik gemaakt van een TRCT5000 IR sensor. De IR-sensor die hier gebruikt wordt heeft twee uitgangen: D0 en A0. D0 is de digitale uitgang (1 of 0) en A0 is de analoge uitgang (een spanningsrange). Voor dit systeem wordt er gebruik gemaakt van de digitale uitgang omdat dit het programmeren makkelijker maakt en er is ook niet een reden om het analoge signaal te gebruiken. Wanneer de sensor niks detecteert staat er een hoog signaal op de digitale uitgang. Wanneer er iets wordt gedetecteert wordt de spanning op de uitgang laag. Dus deze sensor is ook actief laag.



Figuur 1.1B: TRCT5000 IR sensor module

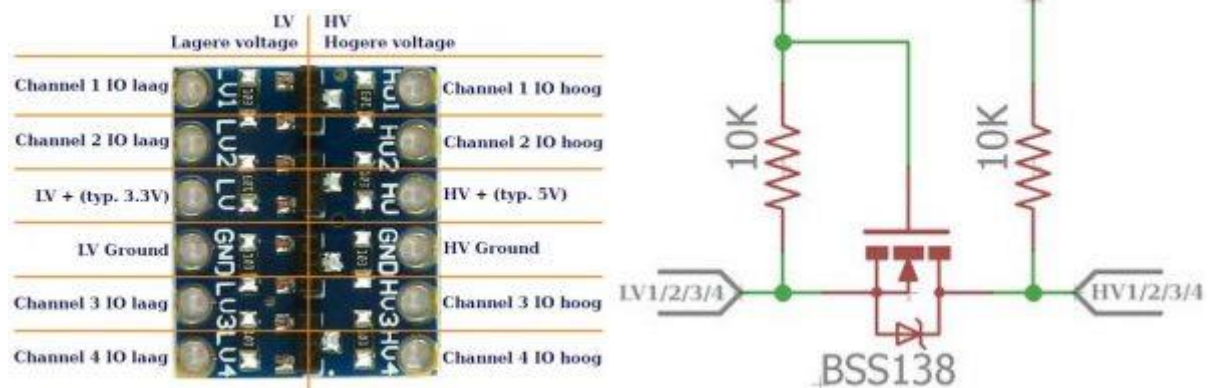


Figuur 1.1C: Pinout van de module (bron: <https://www.instructables.com/How-to-Use-TCRT5000-IR-Sensor-Module-With-Arduino-/>)

Het alarm systeem moet uiteraard een alarm hebben. Hiervoor wordt er een passieve buzzer module gebruikt. Wanneer er een PWM signaal op de buzzer komt, maakt het een geluid van 85dB. De buzzer die hier wordt gebruikt werkt alleen op 5V dus er moet een logic voltage converter er tussen komen zodat de 3.3V van de Pi wordt omgezet naar 5V.

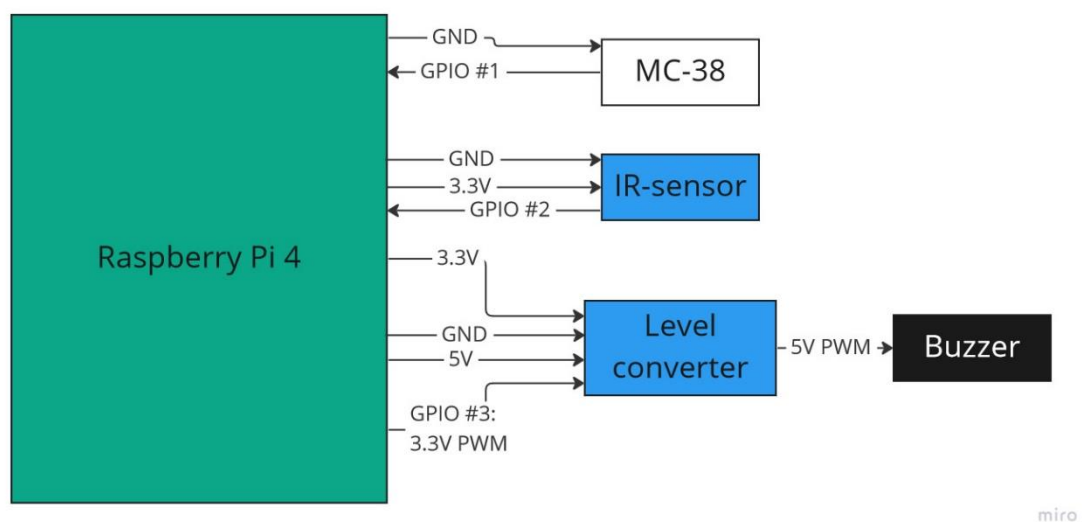


Figuur 1.1D: Passieve buzzer (bron: <https://www.otronic.nl/buzzer-module-passief-voor-arduino.html>)



Figuur 1.1E: Pinout van de logic level converter en het schema (bron: <https://www.gadgetbouw.nl/shop/4-channel-voltage-level-conversie-3-3-5v-module/>)

Met deze keuzes kan het volgende schema gemaakt worden van het hardware onderdeel van dit project:



Figuur 1.1F: Schema van de sensoren en de Pi.

De pinnen van de Raspberry Pi die gebruikt worden, worden tijdens het realisatie proces bepaald.

### Software:

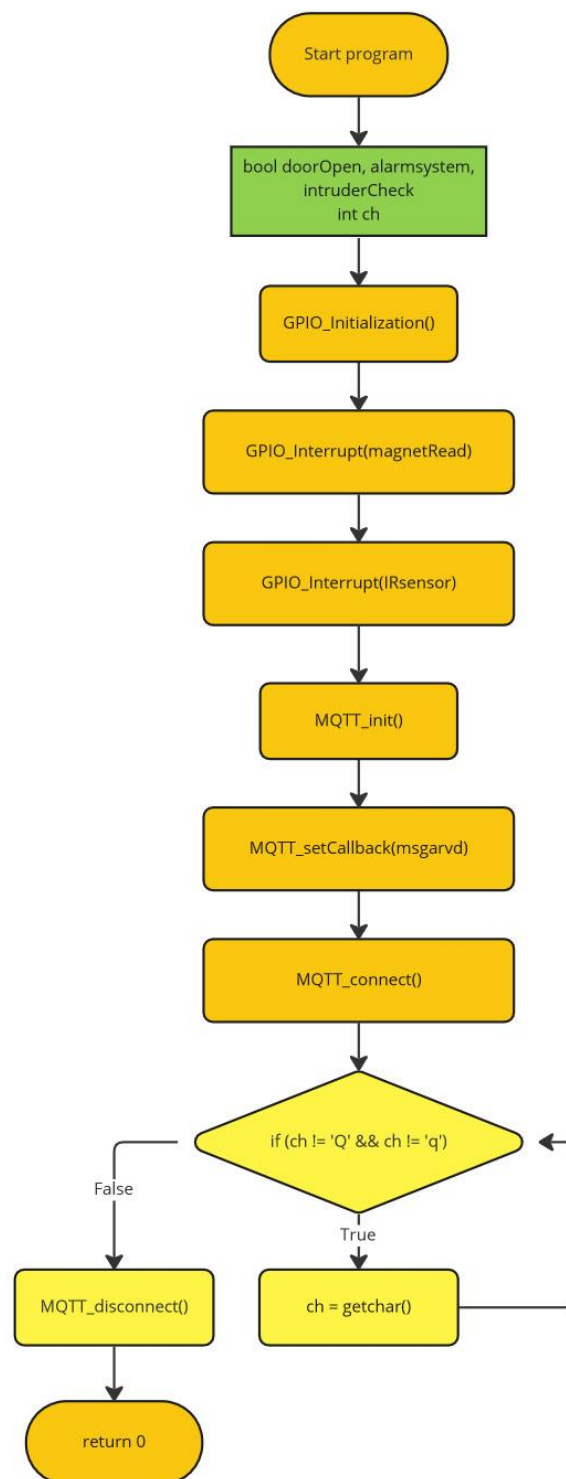
Bij een alarm systeem wilt de gebruiker dat er zo snel mogelijk gereageerd wordt zodra er een inbreker binnenkomt. Voor deze rede wordt er niet gebruik gemaakt van python gebruikt. Het idee is om dan gebruik te maken van een hardware interrupt op basis van het signaal wat binnenkomt van het magneetsensor en de IR-sensor.

Op basis van dit idee zijn volgende programmeertalen gekozen:

- C/C++ , deze talen hebben support voor interrupts.
- Rust, deze taal is zeer snel en efficiënt en heeft veel open source support.

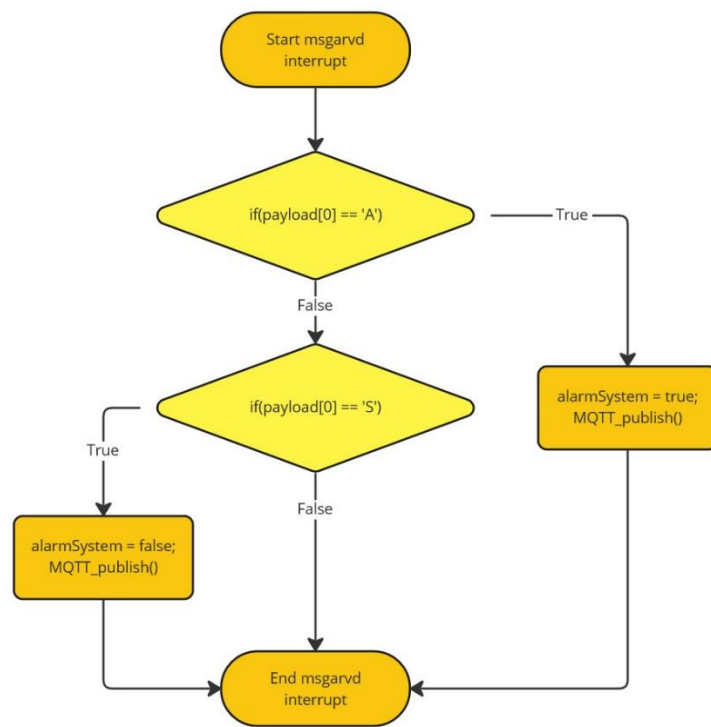
Uiteindelijk is er gekozen voor de programmeertaal C vanwege ervaring met de taal.

Voor de software zijn de volgende flowcharts gemaakt:



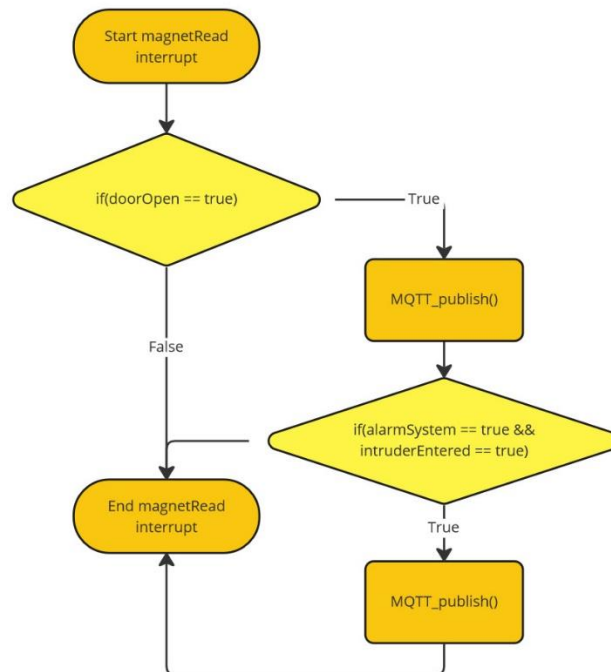
miro

*Figuur 1.1G : Main loop.*



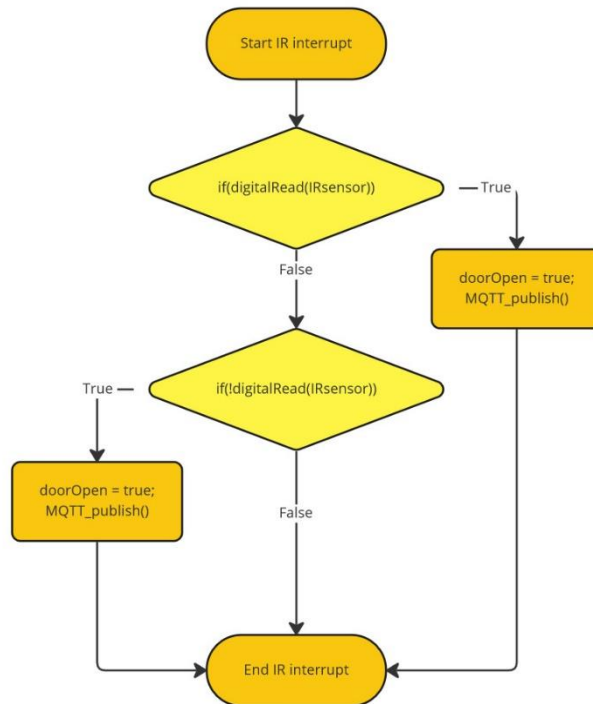
miro

*Figuur 1.1H : Subscribe callback.*



miro

*Figuur 1.1I: Magneet sensor interrupt.*

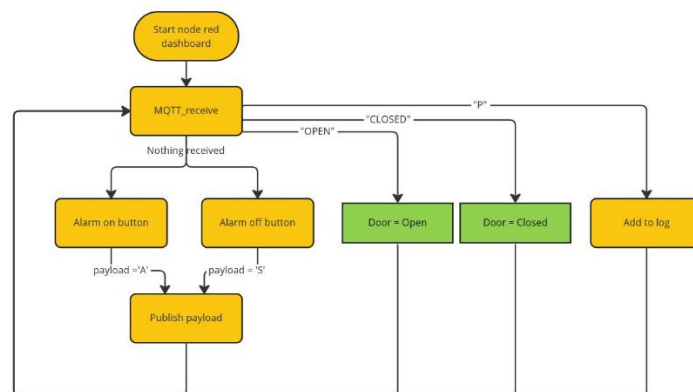


*Figuur 1.1J: IR-sensor interrupt*

## 1.2. Gateway + IoT middleware

Voor het gateway onderdeel van de pipeline wordt uiteraard gebruik gemaakt van WiFi. Naast dit wordt er gebruik gemaakt van MQTT. Met MQTT kan tussencommunicatie met het systeem en de interface soepel verlopen. Plus er is een mogelijkheid om zelf een MQTT broker server op te zetten op de Raspberry Pi. Om het systeem simpel te houden wordt er nu gebruik gemaakt van de publieke HiveMQ broker server.

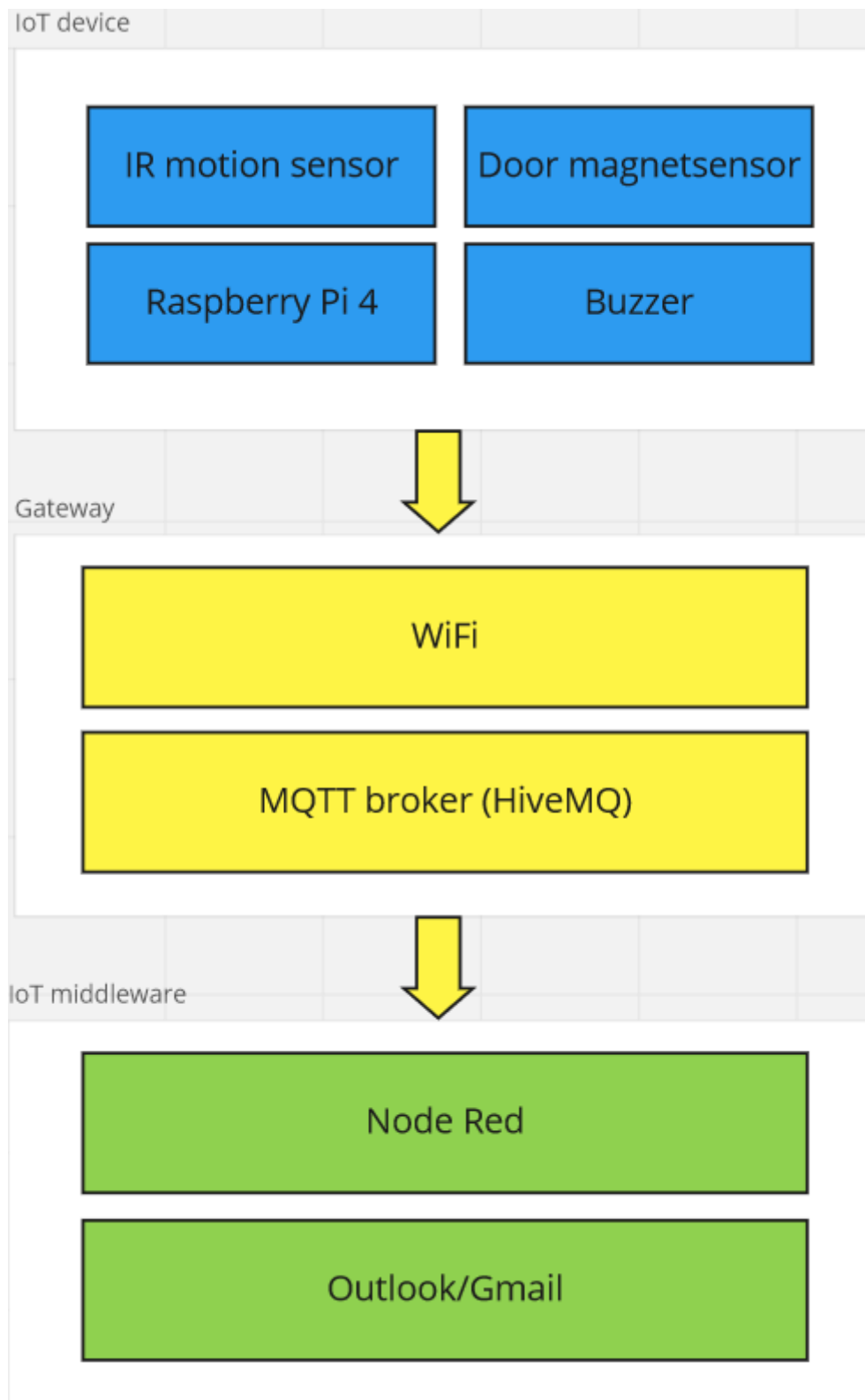
Voor het systeem moet er ook software worden uitgevoerd op het IoT middleware niveau. Hier kan er gekozen worden tussen platformen als Home Assistant, ThingSpeak en Node Red. Er is gekozen voor Node Red vanwege het vele support en de simpele interface die makkelijk te programmeren is. Voor de interface is de onderstaande flowchart gemaakt.



*Figuur 1.2A: Node Red dashboard flowchart.*



Als alle onderdelen bij elkaar worden gezet kan de volgende IoT pipeline worden opgesteld (zie onderstaande figuur).



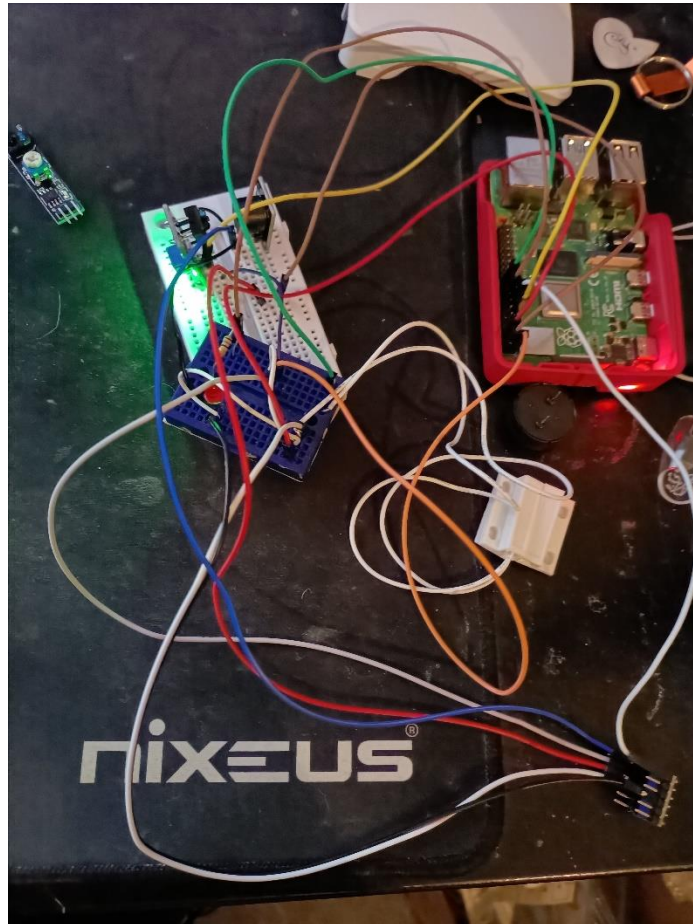
*Figuur 1.2B: IoT pipeline van Alarm System OMEGA.*

## 2. Realisatie

Het realisatie proces heeft 2 onderdelen: Het hardware onderdeel en het software onderdeel. Beide onderdelen worden uiteindelijk met elkaar geïntegreerd. De resultaten van de integratie is te zien op de demo video die beschikbaar is op de github repository.

### Hardware:

Voor de hardware is er gebruik gemaakt van 2 breadboards en female-to-male kabels om alle sensoren te connecten met de Pi. Het resultaat is te zien in de onderstaande figuur.



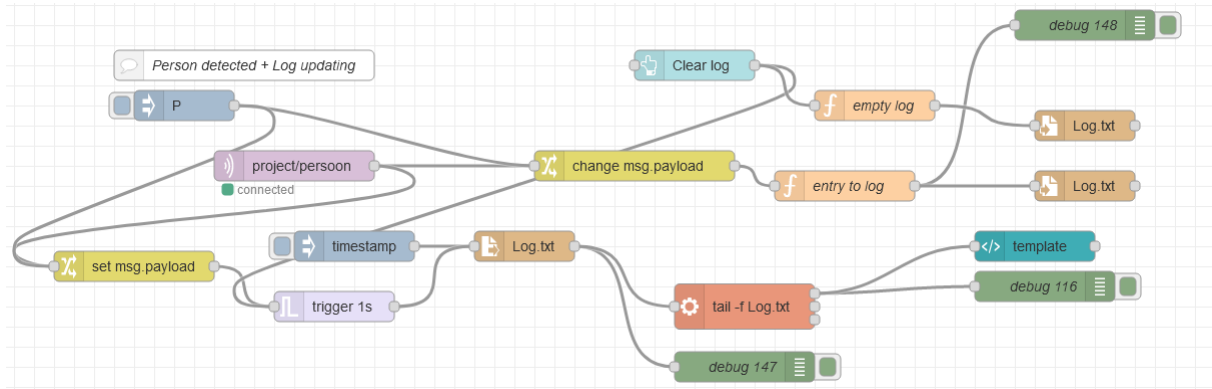
*Figuur 2A: Wiring van het prototype.*

### Software:

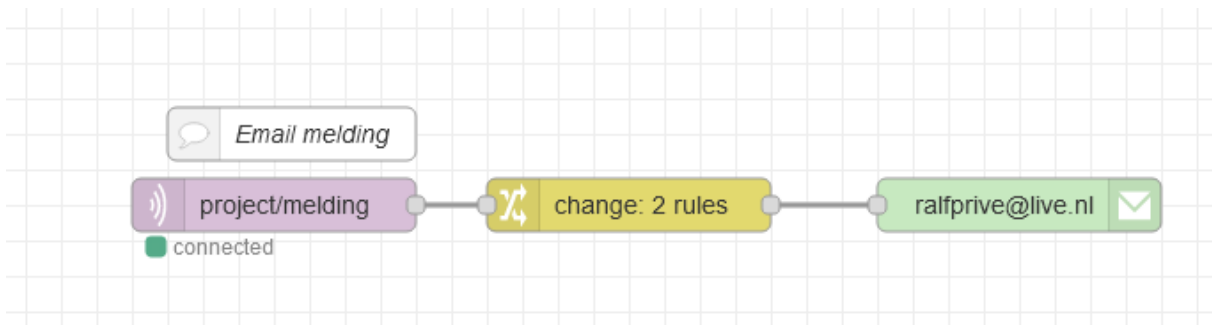
Het software van het prototype bestaat uit het Node Red onderdeel en het C-code onderdeel. De C-code is beschikbaar op de github repository. In die code reageert het systeem op de sensoren op basis van interrupts. Dit is gedaan om ervoor te zorgen dat het systeem zo snel mogelijk reageert op verandering. Het systeem reageert soms op dendering van de IR sensor. Dit kan gefixt worden met een hardware of software ontddendering.

Het dashboard werkt op basis van MQTT topics. Als er iets verstuurd wordt vanuit de C-code reageert het systeem daarop. De flows.json van het dashboard staat in de github repository. Wanneer het json bestand wordt geïmporteerd ziet het eruit als de onderstaande figuren. De werking van het dashboard zelf is te zien bij de demo video op de repository. Om bij het dashboard te komen moet de gebruiker naar het IP-adres gaan van het systeem die een node red service aan heeft staan op port

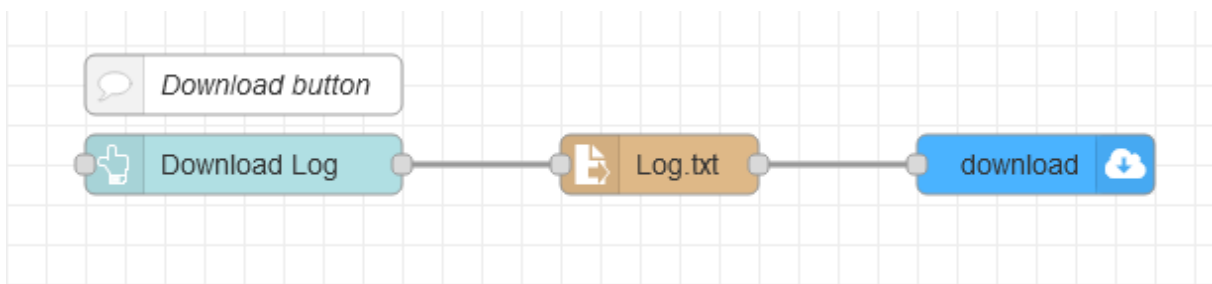
1880. Dus <IP-adres>:1880. Om bij de interface te komen moet de gebruiker <IP-adres>:1880/ui invullen.



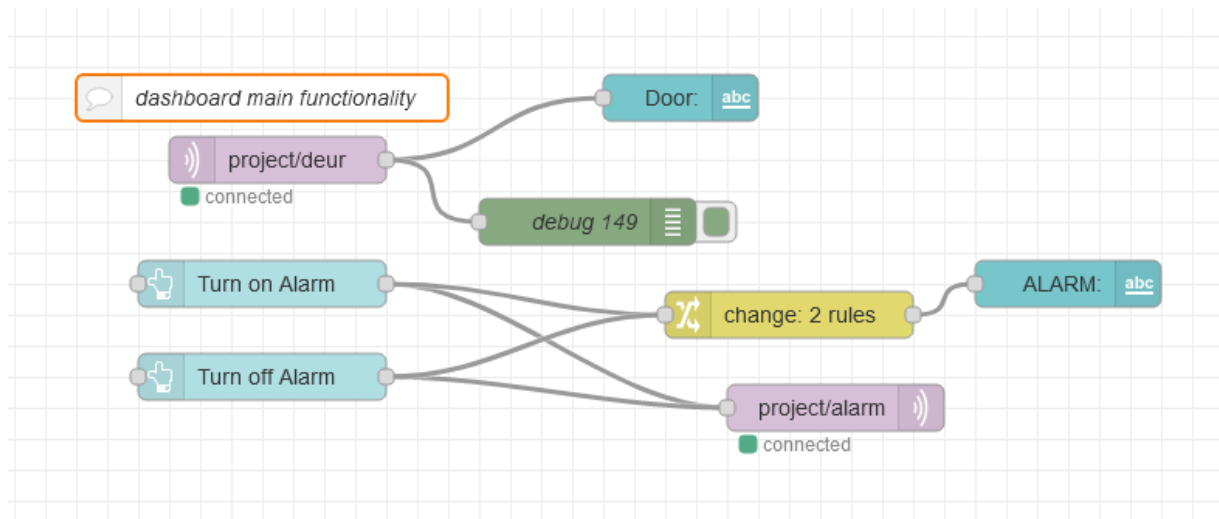
Figuur 2B: Node Red persoon detectie flows.



Figuur 2C: Node Red email melding.



Figuur 2D: Download button.



*Figuur 2E: dashboard functionality.*

### 3. Reflectie

Voor de volgende prototype van dit systeem kunnen er hardware en software verbeteringen meegenomen worden. Als eerste kan er een betere magneetsensor gekocht worden die past op de voordeur. De MC-38 is niet per se slecht maar de sensor is vrij klein en kan dus makkelijk los komen. Naast dit past deze sensor niet met de vorm van de deur dus het prototype kon niet op een gepaste manier getest worden.

Daarnaast moet er voor de IR sensor een ontdender oplossing gezocht worden. Dit kan via een hardware oplossing met een laagdoorlaat RC filter of software ontdendering.

Het dashboard van het systeem kan worden uitgebreid. Naast de log kan er een dashboard functie gemaakt worden die bijhoudt en plot hoeveel mensen er binnen zijn gelopen of het gebouw heeft verlaten. Daarnaast kan er ook een optie komen voor de gebruiker om notificaties via mail van een inbreker uit te zetten en dat te verifiëren met een wachtwoord.

Het alarm wordt op dit moment via het dashboard aangezet. Dit kan ook hardwarematig gedaan worden door middel van een switch of een aan/uit knop. Hierdoor kan de gebruiker op meerdere manieren het alarm uitzetten. Wellicht wordt de switch dan wel verborgen voor de inbrekers. Er kan dan via een schermje worden laten zien vanuit de Pi (bijvoorbeeld een TFT of OLED schermje).