

IOT PROJECT ADVANCED

# GLADIOLEN

# Inhoud

<b>1. ONDERDELEN</b>	<b>5</b>
1.1. Ultrasonische Sensor	5
1.1.1. Werking	5
1.1.2. Specificaties	6
1.2. Microcontroller	6
1.2.1. Werking	7
1.2.2. Specificaties	8
1.3. Powerbank	9
1.3.1. Specificaties	9
1.4. Bewegingssensor	9
1.4.1. Specificaties	10
1.5. Statusled	11
1.6. Knop	11
1.7. LoRa SX1262 chip van Heltec	11
1.8. RAK 7289	12
1.9. PCB-Design	12
1.10. Aanpak PCB-Design	13
1.11. Top and Bottom Layer van PCB	15
1.12. Uitdagingen bij PCB-Design	15
1.13. Reflectie: PCB-Design	15
<b>2. ONTWERP MODULE</b>	<b>17</b>
2.1. Ontwerpkeuzes en proces	17
2.2. Testen en uitdagingen	17
2.3. Interne indeling	17
2.4. Aanpassingen en keuzes	18
2.5. Ontwerpfase	18
2.6. Conclusie	18
2.7. Bijlage Ontwerp Module	19
2.8. Bestanden	22
<b>3. CODE</b>	<b>22</b>
<b>4. HANDLEIDING</b>	<b>23</b>
<b>5. VISUALS</b>	<b>24</b>

<b>5.1. Inleiding</b>	24
<b>5.2. Aanpak</b>	24
5.2.1. Data genereren en ontvangen	24
5.2.2. Dataoverdracht via MQTT	25
5.2.3. Opslag in InfluxDB	25
5.2.4. Visualisatie met Grafana	26
<b>5.3. Uitdagingen</b>	27
5.3.1. Raspberry Pi opnieuw instellen	27
5.3.2. Tijdsgebrek	27
5.3.3. Niet voldoen aan een eis van de opdrachtgever	27
5.3.4. Crash van de Raspberry Pi	28
<b>5.4. Reflectie</b>	28
<b>6. PRESENTATIE</b>	<b>29</b>
6.1. Opbouw en inhoud	29
6.2. Voorbereiding en structuur	29
6.3. Bestanden	29

## Figurenlijst

Figuur 1: afstandsberekening	6
Figuur 2: ultrasone sensor	6
Figuur 3: Ultrasone sensor	6
Figuur 4: commando's LoRa	7
Figuur 5: Heltec	8
Figuur 6: Powerbank	9
Figuur 7: Motion sensor	10
Figuur 8: Ontvangen Data TTN	25
Figuur 9: InfluxDB	25
Figuur 10: Visualisatie Grafana	26
Figuur 11: Visualisatie Grafana	26

## Tabellijst

Tabel 1: specs ultrasone sensor	6
Tabel 2: specs Heltec	8
Tabel 3: Specs Powerbank	9
Tabel 4: Specs Motion Sensor	10

# 1. Onderdelen

Nadat we alles getest hebben konden we materiaal beginnen uitkiezen. Dit is iets wat we klassikaal hebben gedaan. Iedereen heeft een onderdeel toegewezen gekregen waar hij de beste oplossing voor moest vinden. Even een opsomming van wie wat heeft gevonden/gedaan:

- Olli: ultrasone sensor
- Jens M. & Olli: microcontroller
- Jens G. : LoRa-module en gateway
- Neman & Joris: behuizing, powerbank & PCB
- Mathijs: zelf ontwerpen 3d-geprinte behuizing
  - o Dit hebben we uiteindelijk niet kunnen doen wegens tijdsgebrek.
- Robin: visualisatietools zoeken.
- Senne: algemeen opzoekwerk

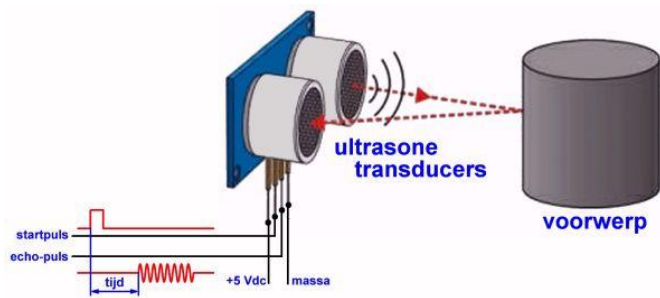
Nadat we alle onderdelen gekozen hadden, waarvan nieuwe en de onderdelen die we vorig jaar al gebruikt hadden, konden we hiermee beginnen testen. Joris & Neman hebben een berekening gemaakt van hoeveel energie alle onderdelen samen zouden gebruiken gedurende een héél festivalweekend en hiermee hebben we dan de grootte van onze batterij bepaalt. Voor de andere onderdelen werd vooral gekeken naar simpliciteit, zo weinig mogelijk onderdelen en de prijs. Wat volgt is een opsomming van elk elektronisch onderdeel, de specificaties en waarvoor het dient. Het mechanische deel, hoe alles in elkaar zit wordt later in dit verslag uitgelegd.

## 1.1. Ultrasone Sensor

Wij hebben twee sensoren getest waaruit de ultrasoon als beste optie is gekomen. Vorig jaar en begin dit jaar hebben we ultrasoon tegen Time of Flight getest. Uit deze testen bleek dat de ultrasoon de betere optie is. De ultrasoon heeft volgende voordelen t.o.v. de ToF-sensor: robuustheid, waterdichtheid, integratie, kostprijs.

### 1.1.1. Werking

De ultrasone sensor wordt gebruikt om de vullingsgraad van de bekerbak te meten. Hij stuurt een ultrasone geluidsgolf van 40kHz, die reflecteert op alles wat ervoor komt. De geluidsgolf zal weerkaatsen en door een berekening a.d.h.v. de tijd die tussen het uitsturen en ontvangen zit, kunnen we de afstand bepalen.



Figuur 2: ultrasone sensor

$$\text{distance} = \text{duration} * 0.0344 / 2;$$

Figuur 1: afstandsberekening

### 1.1.2. Specificaties

Wij gebruiken de JSN-SR04T ultrasone sensor. Deze hebben we gekozen omdat hij waterdicht is (sensor, pcb niet). Ook heeft hij een transceiver i.p.v. een transmitter en een receiver apart, wat plaats besparend is.

Tabel 1: specs ultrasone sensor

Specificaties	
Bereik	20-450cm
Spanning	3.0-5.5V
Stroom	<8mA
Nauwkeurigheid	+/-10mm
Sensor hoek	75°
Afmetingen	42x29x12
Kabellengte	~250cm



Figuur 3: Ultrasone sensor

## 1.2. Microcontroller

De  $\mu\text{C}$  (microcontroller) is het eerste onderdeel dat we moesten veranderen t.o.v. vorig semester. Toen hadden we problemen met de LoRa-module verbinden met een esp32, waardoor we een extra  $\mu\text{C}$  met een esp8266 gebruikte. Om de totale opstelling zo simpel mogelijk te maken zochten we naar een  $\mu\text{C}$  met zoveel mogelijk onderdelen aan boord. We kwamen uit bij 2 verschillende, één van LilyGo en één van Heltec. We kozen uiteindelijk voor de Heltec omdat deze qua prijs vele dichterbij ons budget lag.

Olli Gevers, geschreven op 22/01/25

### 1.2.1. Werking

De Heltec is een  $\mu$ C met een GNSS-module, een LoRa-module en een ESP32 chip. Hij is ontworpen voor een laag energieverbruik, wat bij ons ook zeer belangrijk is. Voordat de Heltec gebruikt kan worden met LoRa, moet deze eerst geactiveerd worden via een commando dat via de seriële poort wordt verzonden. Dit is een noodzakelijke stap om communicatie via het LoRa-netwerk mogelijk te maken. Hieronder geven we enkele voorbeelden van deze commando's. Deze specifieke commando's waren nodig voor de twee Heltec-modules die wij in gebruik hebben genomen.

```
AT+CDKEY=4C0B1B20C95171E86C5D87E2C90B4583
```

```
AT+CDKEY=AFFCC092D2F2BB956820800F0C63DB6F
```

*Figuur 4: commando's LoRa*

Wanneer de LoRa-functionaliteit wordt aangesproken met de Heltec-library, wordt de benodigde code automatisch weergegeven in de seriële monitor. Dit maakt het eenvoudig om de juiste instellingen te verifiëren en aan te passen.

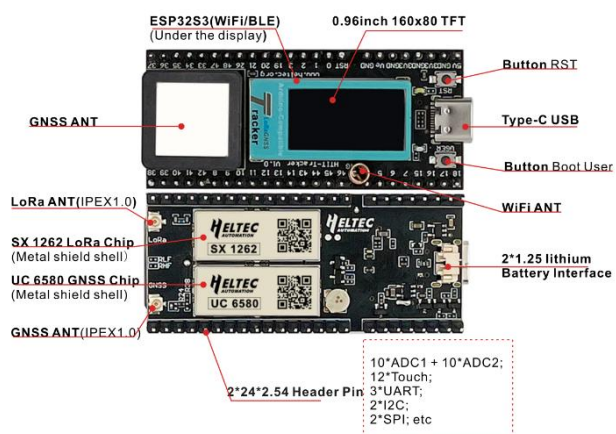
Jens Mertens, geschreven op 24/01/25

## 1.2.2. Specificaties

Tabel 2: specs Heltec

Specificaties	
Master Chip	ESP32-S3FN8 (Xtensa®32-bit lx7 dual core processor)
LoRa Chipset	SX1262
GNSS Chipset	UC6580
Frequency	863~928MHz
Max. TX Power	21±1dBm
Max. Receiving Sensitivity	-134dBm
Wi-Fi	802.11 b/g/n, up to 150Mbps
Bluetooth LE	Bluetooth 5, Bluetooth mesh
Hardware Resource	7*ADC1 + 2*ADC2; 7*Touch; 3*UART; 2*I2C; 2*SPI; etc.
Interface	Type-C USB; 2*1.25 lithium battery interface; LoRa ANT(IPEX1.0); 2*18*2.54 Header Pin
Batterij	3.7V lithium battery power supply and charging
Operating Temperature	-20 ~ 70 °C
Dimensions	65.48 * 28.06* 13.52mm

### Hardware Resources



Figuur 5: Heltec



### 1.3. Powerbank

We gebruiken als voeding een powerbank. Er is een ruwe berekening gemaakt hoeveel capaciteit de powerbank zou moeten hebben voor 3 à 4 dagen, één extra dag voor de zekerheid en één extra dag omdat het een ruwe berekening is. Er zou met wat extra testen, een kleinere powerbank kunnen worden voorzien.

#### 1.3.1. Specificaties

Tabel 3: Specs Powerbank

Specificaties	
Aansluitingen	μ-USB, USB-A, USB-C
Capaciteit	27000mAh
Samenstelling van batterij cel	Lithium-polymeer
Vermogen	USB-C: 22.5W USB-A: 20W
Extra	Batterijniveau op scherm



Figuur 6: Powerbank

Ruwe berekening:



Microsoft Word  
Document

### 1.4. Bewegingssensor

Dit is een onderdeel dat we nog niet toegepast hebben, we hebben alleen het idee hoe we deze zouden gaan gebruiken. Hij zou dienen om de gps af te zetten als de vuilbak beweegt om zo bij de visualisatie niet constant verplaatsende bakken te zien. De bewegingssensor die wij zouden gebruiken is de bmi160.

### 1.4.1. Specificaties

Tabel 4: Specs Motion Sensor

Specificaties	
Bedrijfsvoltage	3.2V - 6V
Stroomverbruik	< 1mA
Interface	Gravity-I2C
Versnellingsbereik	$\pm 2g/ \pm 4g/ \pm 8g/ \pm 16g$
Gyroscoopbereik	$\pm 125^{\circ}/s, \pm 250^{\circ}/s, \pm 500^{\circ}/s, \pm 1000^{\circ}/s, \pm 2000^{\circ}/s$
Versnellings-nul-g offset	$\pm 40mg$
Gyroscoop-nul-g offset	$\pm 10^{\circ}/s$
Programmeerbare frequentie	25/32Hz - 1600Hz
6D detectie en locatie	Ja
Data-uitvoer	16-bit
Schokbestendigheid	1000g x 200us
Onafhankelijke programmeerbare onderbrekingsgeneratoren	2
Ingebouwde FIFO	1024 byte
Werktemperatuur	-40°C - +85°C
Afmetingen	22mmx27mm



Figuur 7: Motion sensor

## 1.5. Statusled

Het statusledje gebruiken we om de status van de hele module weer te geven. Dit staat nog niet op punt, maar deze zou volgende statussen kunnen weergeven;

1. Programmeermode
2. Lage batterij
3. Batterij aan het opladen
4. Error
5. ...

We gebruiken een 5mm common kathode RGB-LED.

## 1.6. Knop

We zouden ook een knop willen toevoegen om de module in programmeermode te zetten. Dit is momenteel een 4-pins 5mm button.

Olli Gevers, geschreven op 22/01/25

## 1.7. LoRa SX1262 chip van Heltec

De LoRa SX1262 is een radiomodule die ontworpen is om draadloze communicatie over lange afstanden mogelijk te maken met behulp van de LoRa (Long Range) technologie. Deze chip is de opvolger van de SX127x-serie en biedt enkele verbeteringen ten opzichte van zijn voorgangers, zoals lagere stroomverbruik en verbeterde zendvermogensopties. De SX1262 ondersteunt frequentiebanden tussen 150 MHz en 960 MHz en is vooral populair in applicaties voor het Internet of Things (IoT) vanwege zijn vermogen om signalen te versturen en ontvangen over grote afstanden, zelfs in stedelijke of industriële omgevingen met veel fysieke en elektromagnetische verstoringen.

Enkele kenmerkende functies van de SX1262 omvatten:

- Verbeterde energie-efficiëntie, wat resulteert in een langere batterijduur voor apparaten.
- Ondersteuning voor zowel LoRa- als FSK/ook-modulaties.
- Mogelijkheid om te werken bij een lager zendvermogen, wat bijdraagt aan de naleving van de regelgeving zonder in te boeten aan communicatiebereik.
- Betere immuniteit tegen interferentie en lagere foutmarges bij data-overdracht.

Deze eigenschappen maken de SX1262 bijzonder geschikt voor toepassingen in smart cities, industriële monitoring, en agrarische sensornetwerken, waar betrouwbare, energiezuinige langeafstandscommunicatie cruciaal is.

## 1.8. RAK 7289

De RAK Wireless 7289, ook bekend als de RAK7289 WisGate Edge Pro, is een geavanceerde LoRaWAN-gateway die ontworpen is voor IoT-netwerkbeheer. Deze gateway ondersteunt de connectiviteit van een breed scala aan LoRa-apparaten en faciliteert de communicatie tussen deze apparaten en een centraal netwerkbeheersysteem. Het is een krachtige oplossing voor het opzetten van uitgebreide LoRaWAN-netwerken, vooral nuttig in industriële en commerciële toepassingen.

Enkele kenmerken van de RAK7289 zijn:

- **Ondersteuning voor meerdere frequentiebanden:** Dit maakt het mogelijk om de gateway in verschillende regio's wereldwijd te gebruiken, conform de lokale regelgeving.
- **Hoge capaciteit:** De gateway kan duizenden LoRa-nodes ondersteunen, wat ideaal is voor grootschalige netwerken.
- **Ingebouwde netwerkserver:** Hiermee kan de gateway lokale netwerkbeheertaken uitvoeren zonder afhankelijk te zijn van externe servers.
- **Robuustheid en betrouwbaarheid:** Ontworpen om te functioneren in uitdagende omgevingen, wat belangrijk is voor industriële toepassingen.

De RAK7289 biedt ook geavanceerde beveiligingsfuncties om de integriteit en veiligheid van netwerkdata te waarborgen, wat cruciaal is voor commerciële en industriële IoT-toepassingen. De gateway is flexibel inzetbaar en biedt zowel bedrade als draadloze backhaul-opties, inclusief Ethernet en optioneel cellular backhaul, afhankelijk van het specifieke model. Dit maakt het een veelzijdige keuze voor verschillende soorten IoT-implementaties.

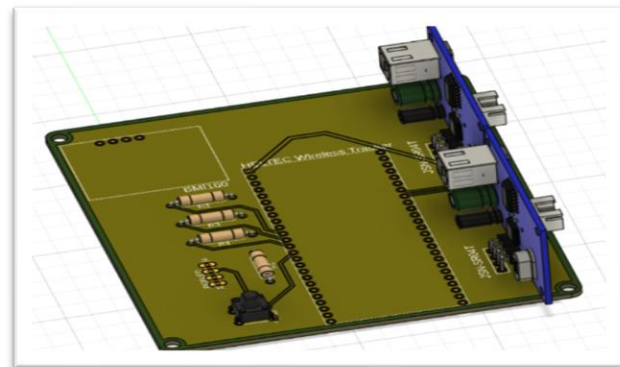
Jens Geenen, geschreven op 24/01/25

## 1.9. PCB-Design

Voor het ontwerpen van de printplaat (PCB) moesten custom footprints worden gemaakt, omdat veel componenten niet beschikbaar waren in de Autodesk Fusion-library. Dit was een tijdrovende taak vanwege beperkte ervaring met PCB-ontwerp.

Daarnaast was regelmatig overleg met andere teamleden nodig om de afmetingen van de printplaat af te stemmen en te bepalen welke componenten erop moesten worden geplaatst. Deze samenwerking zorgde ervoor dat het ontwerp nauwkeurig en functioneel werd afgestemd op de behoeften van het project.

Momand Neman, geschreven op 24/01/25



Figuur 8 3D Design-PCB

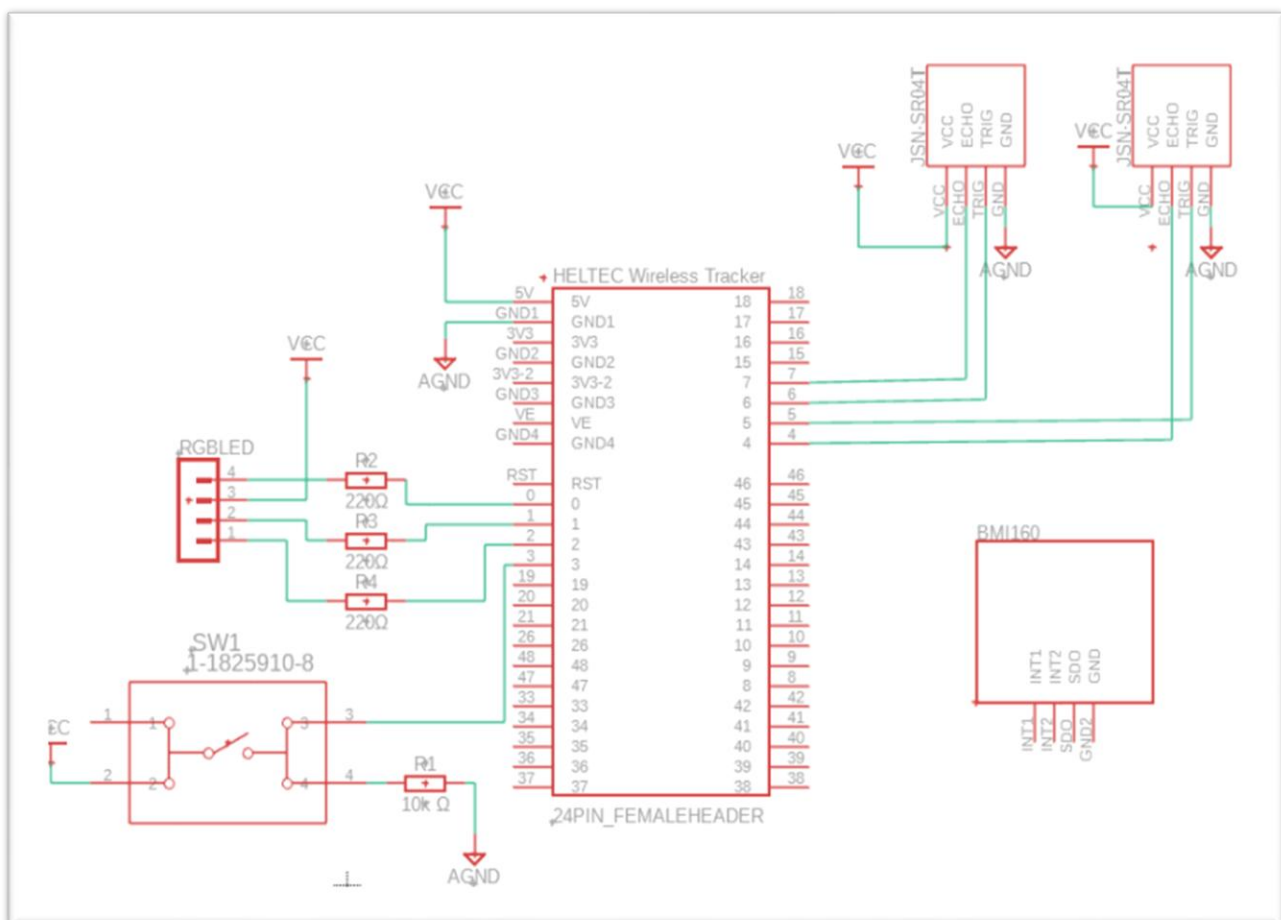
## 1.10. Aanpak PCB-Design

Om de PCB te ontwerpen, is Autodesk Fusion gebruikt. Omdat veel van de benodigde onderdelen niet standaard beschikbaar waren, zijn de footprints zelf ontworpen en afgestemd op de specificaties van de gebruikte componenten, zoals de Heltec-module. Het nodige aantal pinnen zijn specifiek aangemaakt voor deze componenten.

Door weinig kennis zijn we op zoek gegaan hoe we te werk moesten gaan met Autodesk Fusion en hebben we hier in vorige cursussen voor gekeken en op Google/YouTube zitten kijken. Dit heeft veel tijd in beslag genomen maar was uiteindelijk heel leerrijk om met het programma te kunnen werken.

De specificaties van de componenten zijn afkomstig uit de bijbehorende datasheets.

Voordat het ontwerp kon beginnen, werd een aansluitschema opgesteld. Dit schema was essentieel om te bepalen hoe de verschillende onderdelen verbonden moesten worden en welke pinnen gebruikt zouden worden. Fouten in dit stadium konden grote gevolgen hebben voor het eindresultaat. Daarom was ook hier nauw overleg met het team nodig om alles correct te implementeren. Uiteindelijk is dit schema handig



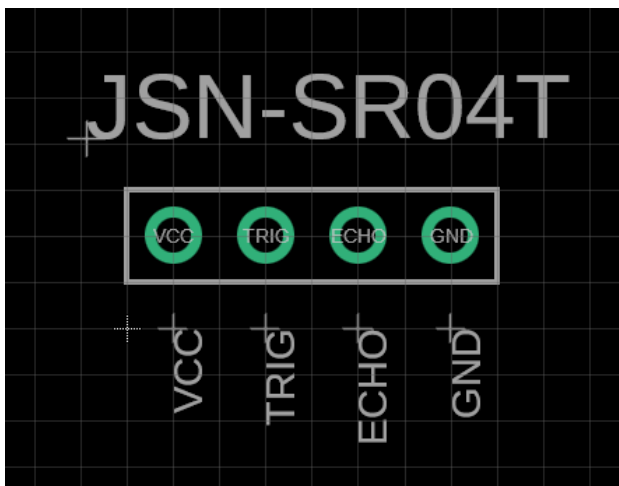
omdat dit automatisch de verbindingen op de pcb lay-out aanmaakt dankzij het handige programma.

Daarna zijn de custom footprints aangemaakt voor bepaalde componenten die niet beschikbaar waren in het programma.

Hieronder hebben we de HELTEC aangemaakt. Omdat het niet bestond in het programma, De pinnen zijn aangepast op grootte en heeft de juiste afstand tussen elkaar om op het pcb te kunnen passen.



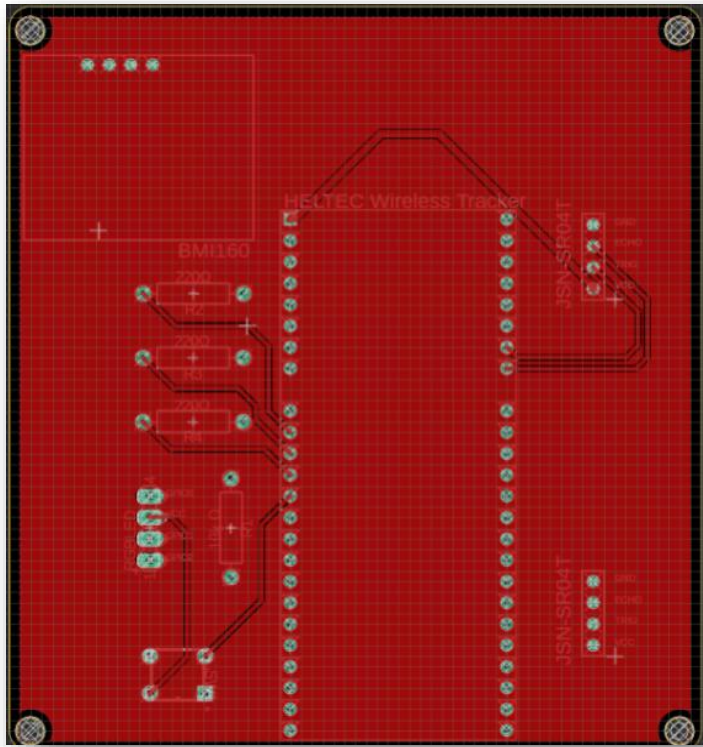
Hierna hebben we de ultrasone sensor aangemaakt zodat deze ook op de pcb met de juiste afmetingen kan passen.



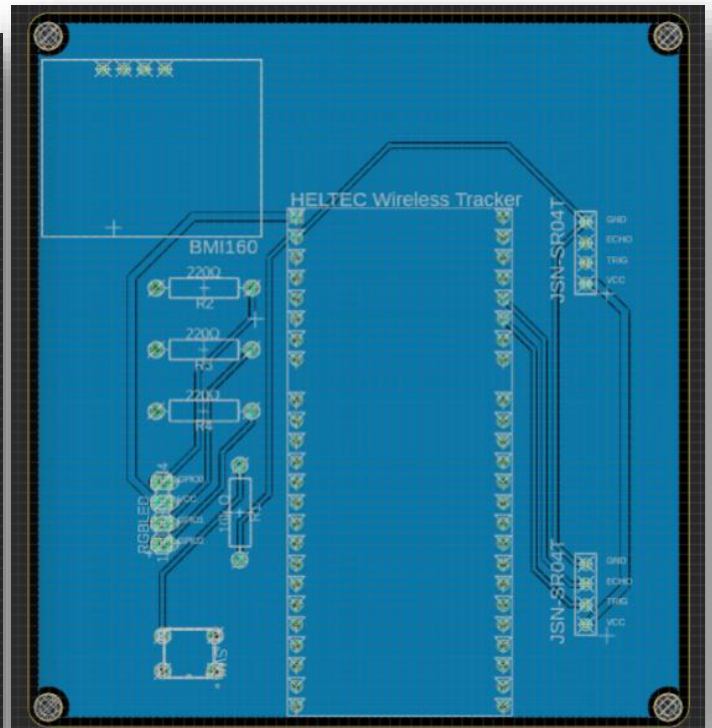
Joris Heylen, geschreven op 24/01/25

Figuur 9 Aansluitschema - PCB

## 1.11. Top and Bottom Layer van PCB



Figuur 11 Top Layer - PCB



Figuur 10 Bottom Layer - PCB

## 1.12. Uitdagingen bij PCB-Design

Het ontwerpen van de PCB bracht verschillende uitdagingen met zich mee. Een van de grootste uitdagingen was het ontwerpen van custom footprints. Omdat dit pas de tweede ervaring was met PCB-ontwerp, was er nog weinig ervaring en kennis. Daarnaast was het noodzakelijk om nauw samen te werken met het team om ervoor te zorgen dat de afmetingen van de printplaat klopten en alle componenten goed geplaatst werden.

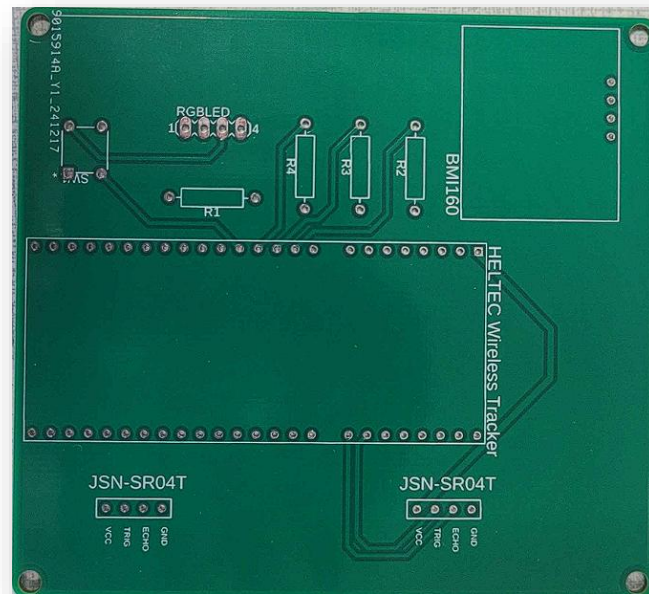
## 1.13. Reflectie: PCB-Design

Dit project heeft ons geleerd hoe belangrijk precisie en samenwerking zijn bij het ontwerpen van een PCB. Het proces was tijdsintensief en bracht onverwachte obstakels met zich mee, maar door feedback van teamleden konden we onze fouten snel corrigeren en het ontwerp verbeteren. Het gaf ons een dieper begrip van hoe de technische componenten in een IoT-systeem samenwerken.



### Conclusie:

Dit project heeft ons niet alleen technische kennis opgeleverd, maar ook laten zien hoe belangrijk samenwerking en doorzettingsvermogen zijn in complexe projecten. Ondanks de uitdagingen zijn we erin geslaagd een functionele PCB te ontwerpen die past binnen de eisen van het project.



Figuur 12 PCB

Momand Neman, geschreven op 24/01/25



## 2. Ontwerp Module

### 2.1. Ontwerpkeuzes en proces

Aanvankelijk was het idee om een volledig afgewerkt product te maken, waarbij we de behuizing op maat zouden ontwerpen en 3D-printen. Door tijdsgebrek hebben we echter gekozen voor een prefab oplossing. Dit had als bijkomend voordeel dat de productiekosten relatief laag bleven en de module onderhoudsvriendelijk werd, omdat alle onderdelen gemakkelijk verkrijgbaar zijn.

We hebben uiteindelijk gekozen voor een IP66-verdeeldoos vanwege de waterdichtheid. Deze doos is via een metalen rail in de vuilnisbak gemonteerd, zodat de module eenvoudig in en uit de bak gehaald kan worden. De enige aanpassingen die we aan de doos hebben gedaan, zijn het maken van twee op maat gemaakte gaten voor de waterdichte ultrasone sensoren. Daarnaast kan de behuizing afgesloten worden met een simpel hangslotje, wat extra veiligheid biedt.

### 2.2. Testen en uitdagingen

Tijdens het proces hebben we verschillende dozen getest op waterdichtheid. Dit bleek een grotere uitdaging dan verwacht, omdat sommige dozen niet volledig waterdicht waren of niet geschikt waren voor onze toepassing. Hierdoor moesten we opnieuw op zoek naar een geschikte behuizing en nieuwe onderdelen bestellen. Door deze vertraging en de tijdsdruk hebben we uiteindelijk meer moeten betalen voor de onderdelen, wat een impact had op het budget.

### 2.3. Interne indeling

De binnenkant van de behuizing hebben we opgevuld met meerdere lagen niet-geleidend schuim. In dit schuim hebben we uitsnedes gemaakt om de verschillende onderdelen stevig te bevestigen, waaronder:

- Onze custom PCB met microcontroller,
- Een LED,
- Een powerbank van 25.000 mAh.

De interne indeling is zo ontworpen dat de onderdelen niet met elkaar in aanraking komen, zelfs niet wanneer de vuilnisbak beweegt of schokken ervaart. Dit draagt bij aan de betrouwbaarheid en duurzaamheid van de module.

## 2.4. Aanpassingen en keuzes

Een van de belangrijkste keuzes was de plaatsing van de powerbank. In het oorspronkelijke ontwerp wilden we de batterijstatus en andere informatie weergeven op het LCD-scherm van de Heltec-module. Dit bleek echter niet praktisch, omdat we de powerbank bovenaan in de module hebben geplaatst. Deze keuze was gericht op de veiligheid en integriteit van de module, en omdat de powerbank het vaakst uit de module gehaald moet worden, bijvoorbeeld om op te laden.

Aanvankelijk wilden we de powerbank vastzetten met klittenband (velcro), maar in het uiteindelijke prototype hebben we gekozen voor een extra laag schuim. Dit kwam doordat we geen stevig ankerpunt hadden om de klittenband aan te bevestigen.

## 2.5. Ontwerpfase

Voor we de module daadwerkelijk bouwden, heb ik alles eerst uitgetekend en vervolgens in 3D ontworpen met SketchUp. Dit gaf ons een duidelijk beeld van hoe de onderdelen in de behuizing zouden passen en hoe we de sensoren en andere componenten het beste konden integreren.

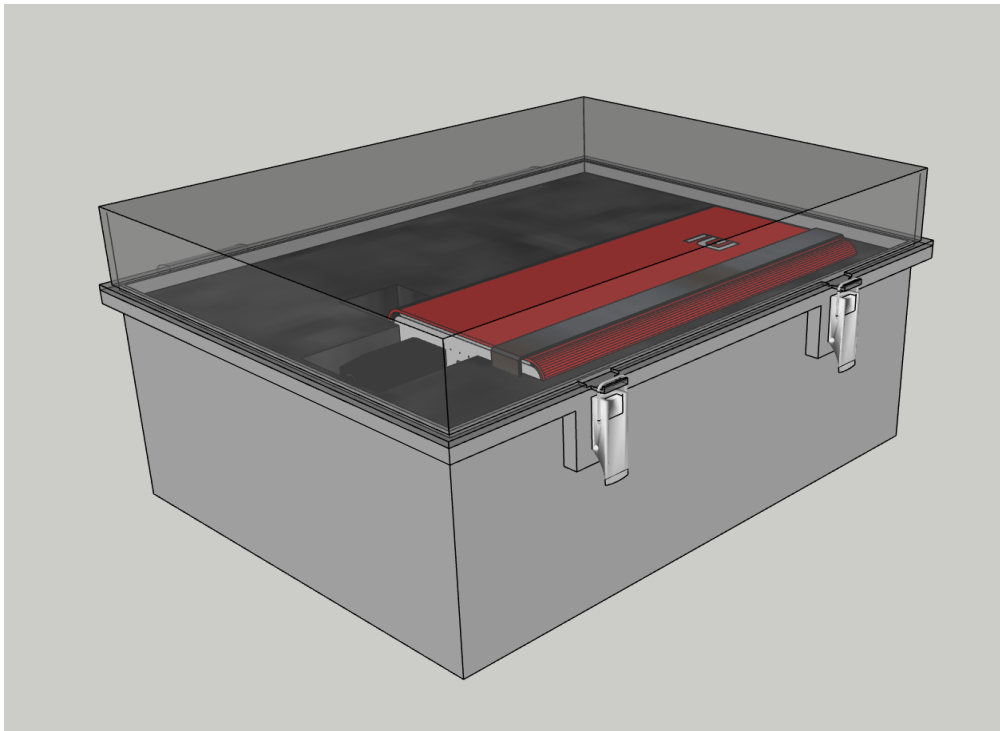
## 2.6. Conclusie

De behuizing is een cruciaal onderdeel van de slimme vuilnisbak, omdat het zorgt voor de bescherming en functionaliteit van de interne componenten. Ondanks de uitdagingen, zoals het vinden van een geschikte waterdichte doos, het testen van verschillende modellen en de extra kosten door tijdsdruk, is het gelukt om een onderhoudsvriendelijke, robuuste en afsluitbare oplossing te realiseren. Dit proces heeft veel waardevolle inzichten opgeleverd, zowel in het ontwerpen als in het praktisch oplossen van problemen.

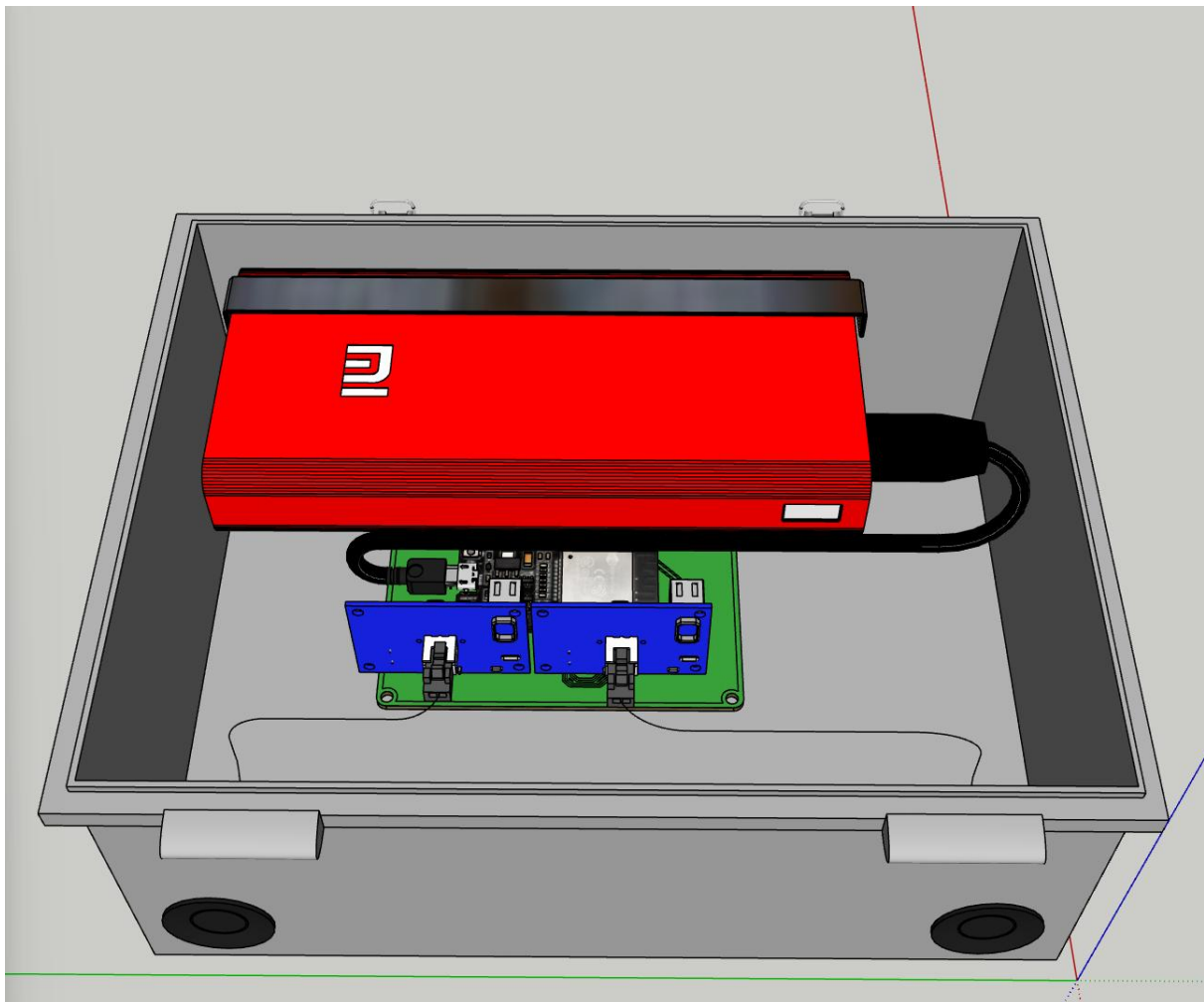
## 2.7. Bijage Onwerp Module



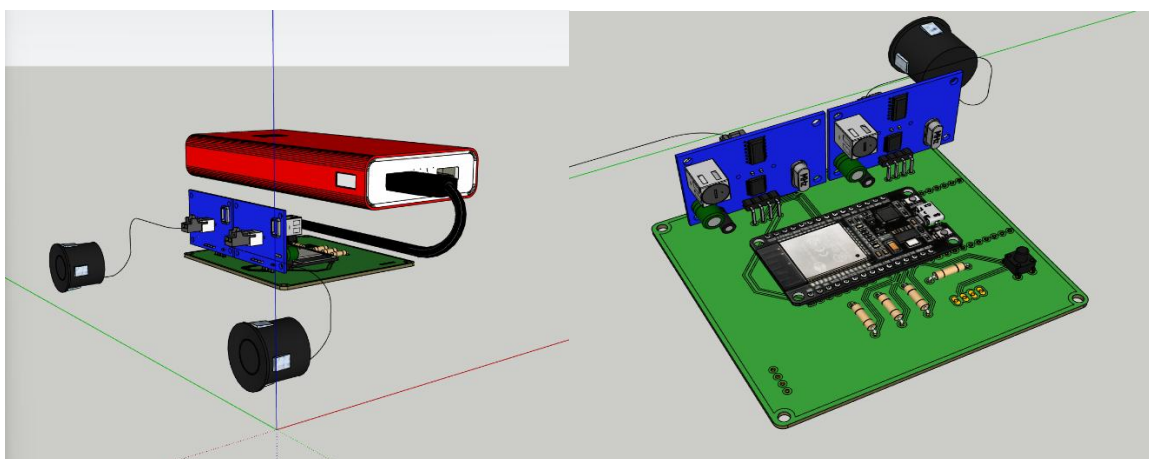
1. ip66 verdeeldoos

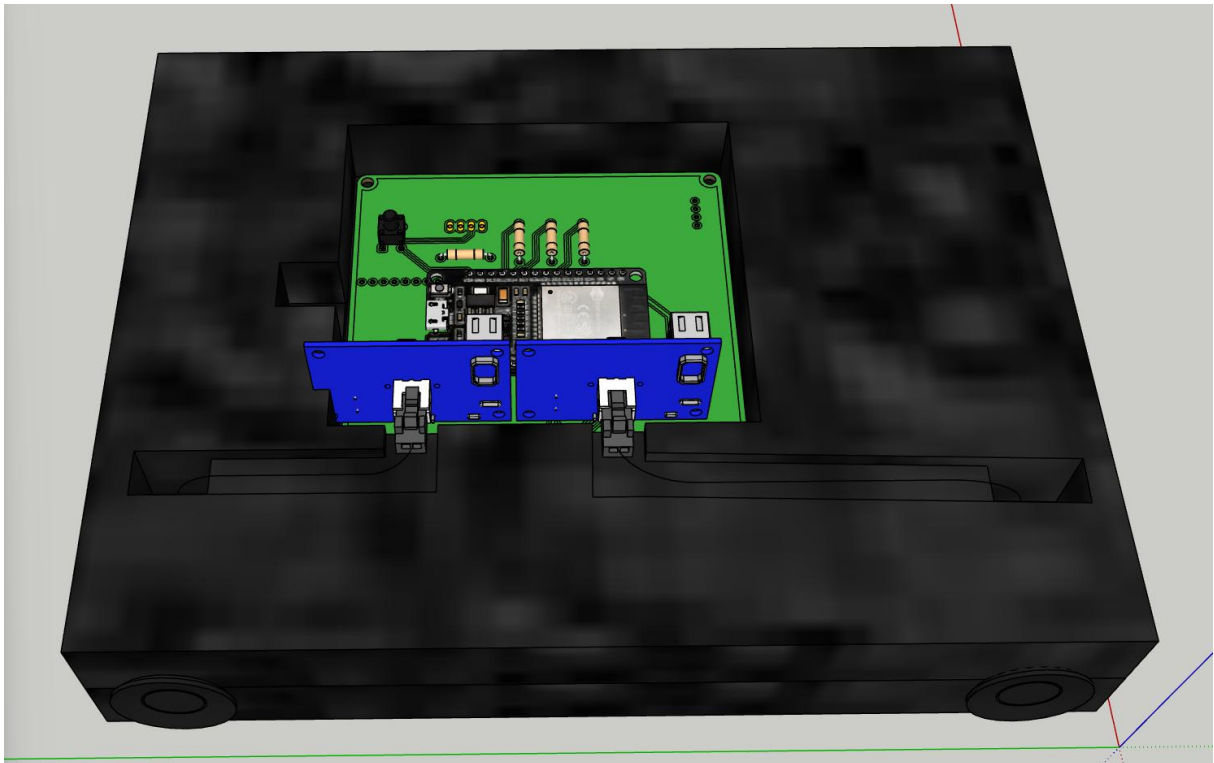


2. 3D module

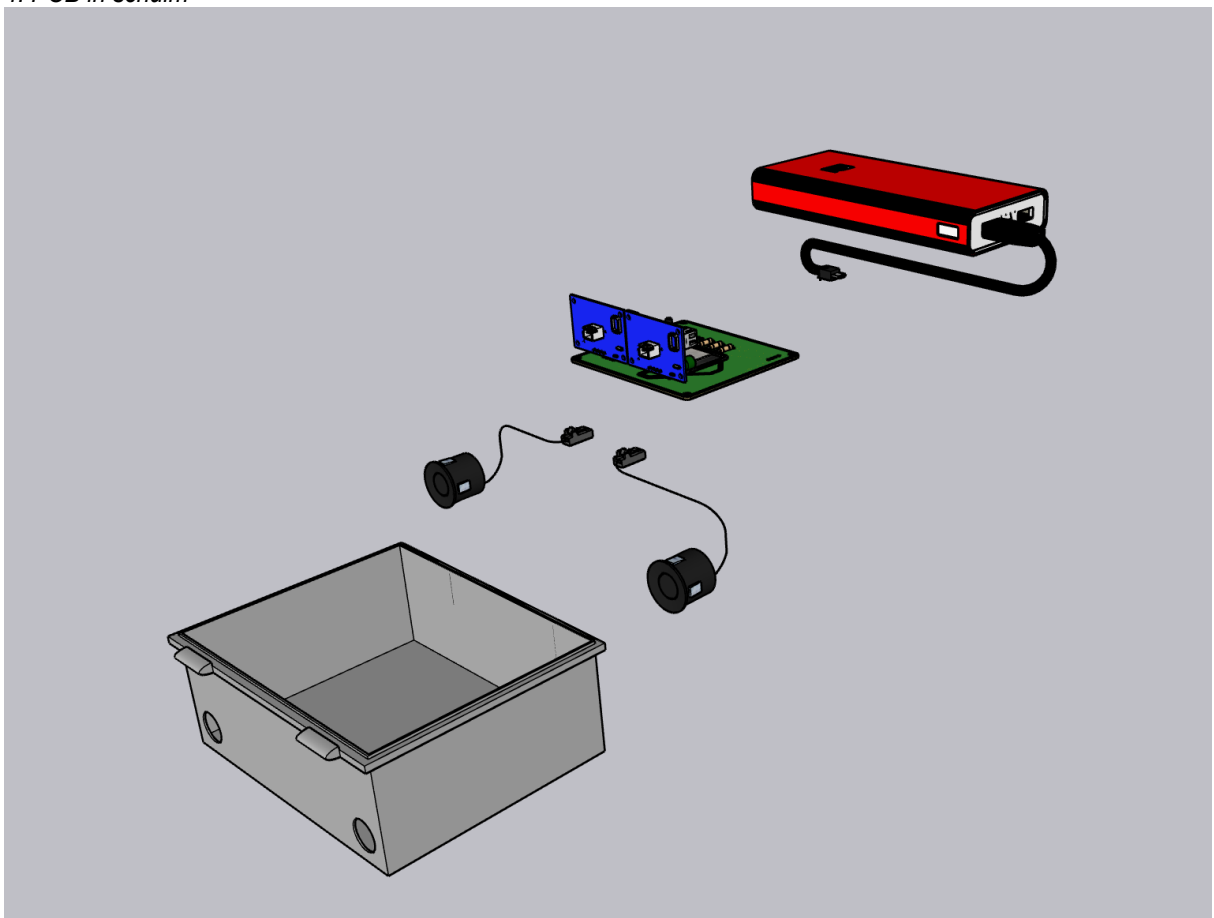


3. invulling module (zonder schuim)





4. PCB in schuim





Screen Recording  
2025-01-17 143228.r

## 2.8. Bestanden



ontwerp module gladiolen.skp

Mathijs Adinau, met inbreng van Joris Heylen, geschreven op 24/01/25

## 3. Code

Voor het schrijven van de code heb ik gebruikgemaakt van de Arduino-ontwikkelomgeving (IDE). Deze omgeving biedt een gebruiksvriendelijke interface en ondersteuning voor verschillende microcontrollers. De fabrikant van de microcontroller had al kant-en-klare libraries ontwikkeld, wat het programmeerproces aanzienlijk vereenvoudigde.

Voordat ik met de code kon beginnen, was het noodzakelijk om de juiste microcontroller en bijbehorende libraries te installeren binnen de Arduino-omgeving. Na het succesvol installeren hiervan werden er automatisch voorbeeldcodes beschikbaar gesteld. Hierdoor kon ik eenvoudig een geschikte microcontroller selecteren uit de lijst en een basiscode als vertrekpunt gebruiken.

Om tot het uiteindelijke resultaat te komen, ben ik gestart met een combinatie van bestaande voorbeeldcode en code die we vorig jaar hebben geschreven. Deze heb ik vervolgens aangepast en geoptimaliseerd om te voldoen aan de specifieke eisen van ons project. Door middel van iteratieve verbeteringen en testen kon ik de code verfijnen en optimaal afstemmen op de functionaliteiten die we nodig hadden. Natuurlijk is deze nog steeds niet perfect en kan deze nog verder evolueren om nog optimaler te werken.

Hieronder vindt u de code die wij gebruiken voor de resultaten die we nu hebben te halen.



Microsoft Word  
Document

Jens Mertens, geschreven op 24/01/25

## 4. Handleiding

Onder de link vindt u de complete handleiding terug van het apparaat.



Handleiding  
(NL,EN,FR,DE) - Slimm

## 5. Visuals

- DOOR ROBIN VAN GAAL

### 5.1. Inleiding

Voor het project Gladiolen heb ik, naast research van de microcontroller en assembleren van de modules samen met teamgenoten, voornamelijk de verantwoordelijkheid op me genomen voor het visualiseren van de data. Deze taak heb ik best laat in het project op me genomen, omdat deze oorspronkelijk door een andere groep zou worden uitgevoerd. Hierdoor moest ik in korte tijd een volledig systeem moeten opzetten waarmee we de ontvangen data op een duidelijke en representatieve manier te visualiseren.

Ondanks de tijdsdruk en de beperkte voorbereidingstijd is het me gelukt om een werkbare oplossing te realiseren. Het visualiseren van de data was een cruciaal onderdeel van het project, omdat het de opdrachtgever in staat stelde om de verzamelde gegevens op een overzichtelijke manier te interpreteren. Dit maakte het mogelijk om belangrijke inzichten te verkrijgen, zoals de locatie van de container en het niveau van de bekercontainer.

Het was een uitdagende taak die veel technische kennis en probleemoplossend vermogen vereiste. Deze ervaring heeft me niet alleen geholpen om mijn technische vaardigheden te verbeteren, maar ook om beter om te gaan met onverwachte situaties en tijdsdruk. Uiteindelijk heeft dit project me veel waardevolle inzichten en praktische kennis opgeleverd.

### 5.2. Aanpak

De aanpak voor het visualiseren van de data verliep in verschillende stappen en vereiste een gestructureerde werkwijze om de data van de sensoren correct te verwerken en weer te geven. Hieronder volgt een overzicht van de belangrijkste stappen:

#### 5.2.1. Data genereren en ontvangen

Nadat de programmeurs de sensoren hadden geconfigureerd en de LoRa-gateway was geïnstalleerd, was het mijn taak om de gegenereerde data te ontvangen en te verwerken. Hiervoor heb ik gebruikgemaakt van een Raspberry Pi 4, die fungeerde als centrale hub voor het verzamelen, opslaan en visualiseren van de data.



### 5.2.2. Dataoverdracht via MQTT

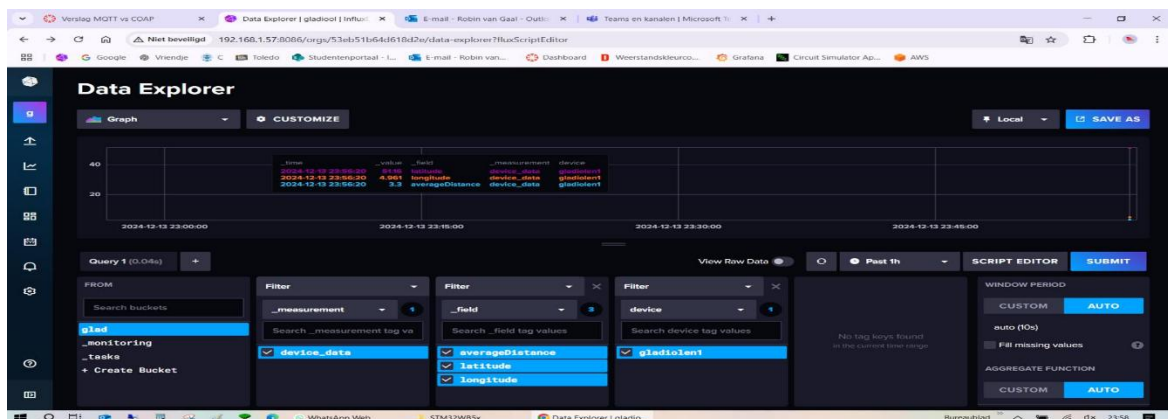
De sensordata werd rechtstreeks naar The Things Network (TTN) gestuurd. Om deze data te ontvangen, heb ik gebruikgemaakt van het MQTT-protocol, een lichtgewicht protocol dat ideaal is voor het uitwisselen van berichten. Via een Python-script, waarin de MQTT-URL en een access token waren verwerkt, kon ik me abonneren op de sensordata. Het script bevatte ook een debugfunctie om te controleren of er een succesvolle verbinding met TTN was.

```
mqtt_client = mqtt.Client()
Verbinden met MQTT broker: eu1.cloud.thethings.network
Verbonden met MQTT broker, resultaat code: 0
Abonneer op topic: v3/gladiolen-2@ttn/devices/+/up
Ontvangen bericht op topic: v3/gladiolen-2@ttn/devices/gladiolen1/up
Decoded payload: {'end_device_ids': {'device_id': 'gladiolen1', 'application_id': 'v3', 'dev_addr': '260B93C0'}, 'correlation_ids': {'gs:uplink': '70B3D57ED006CBB6', 'dev_addr': '260B93C0'}, 'uplink_message': {'f_port': 2, 'f_cnt': 713, 'frm_payload': {'averageDistance': 330, 'id1': 1234, 'id2': 0, 'id3': 0, 'id4': 0, 'id5': 0}, 'metadata': [{'gateway_ids': {'gateway_id': 'gladiolen1', 'eui': 'AC1F09FFFE', 'channel_rssi': -109, 'snr': 1.25, 'uplink_token': 'CiKIAoUZxvPLTAwMDA1ODI1M9/K6BhdfyKY1IPCA6UC0jFsqCwim9/K6BhC4uIkj', 'gps_time': '2024-12-13T22:56:12.323232705Z', 'gateway_ids': {'gateway_id': 'gladiolen1', 'eui': '00005813d35e2eea', 'eui': '00005813d35e2eea', 'channel_rssi': -109, 'snr': -7.3, 'uplink_token': 'CiKIAoUZxvPLTAwMDA1ODI1M9/K6BhdfyKY1IPCA6UC0jFsqCwim9/K6BhC4uIkj', 'received_at': '2024-12-13T22:56:12.121141105Z', 'spreading_factor': 12, 'coding_rate': '4/5'}}, {'frequency': '868106125000', 'received_at': '2024-12-13T22:56:12.112688916Z', 'confirmed': {'frm_payload': {'latitude': 51.161987, 'longitude': 4.96068, 'source': 'S', 'source': 'EC656E0000000181', 'tenant_id': 'ttn', 'cluster_id': 'eu1', 'cluster_id': 'eu1'}}]}
Data geschreven naar InfluxDB: device_data,device=gladiolen1 averageDistance
```

Figuur 13: Ontvangen Data TTN

### 5.2.3. Opslag in InfluxDB

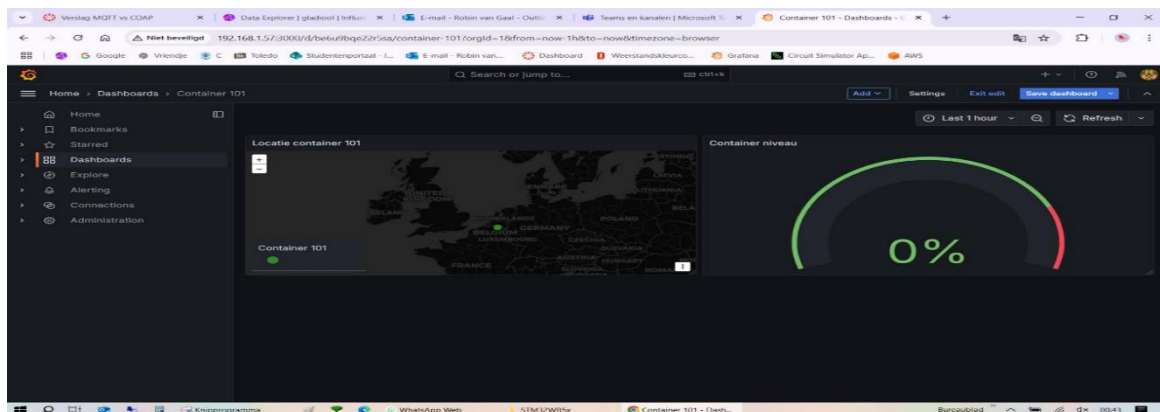
De ontvangen data werden verwerkt en opgeslagen in een lokale InfluxDB-database op de Raspberry Pi. InfluxDB is een time-series database, wat betekent dat het speciaal is ontworpen voor het opslaan en analyseren van gegevens die in de tijd veranderen. De data werden opgeslagen in een bucket genaamd "glad", die toegankelijk was via een unieke API-token. Deze database maakte het mogelijk om de data later op te vragen en te gebruiken voor visualisatie.



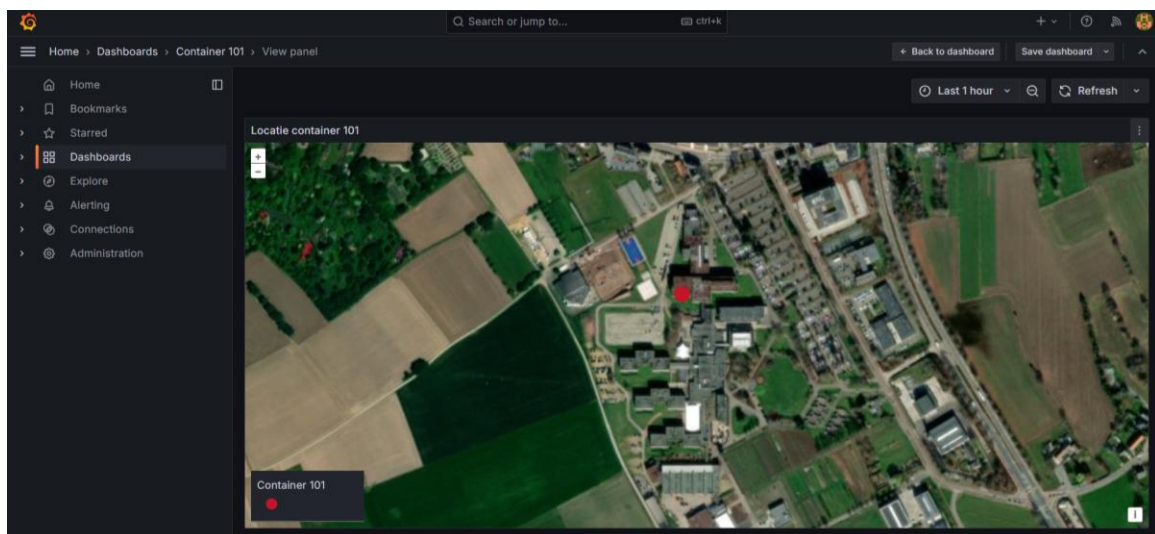
Figuur 14: InfluxDB

## 5.2.4. Visualisatie met Grafana

Voor de visualisatie van de data heb ik gebruikgemaakt van Grafana, een krachtige tool voor het maken van dashboards. Door een dataconnectie in te stellen met InfluxDB (via organisatiegegevens, bucketnaam en API-token) kon ik de data ophalen en visualiseren. Met behulp van query's in Grafana werden de gegevens overzichtelijk weergegeven. In ons geval ging het om de locatie van de container en het niveau van de bekercontainer.



Figuur 15: Visualisatie Grafana



Figuur 16: Visualisatie Grafana

Deze aanpak zorgde ervoor dat de data van de sensoren op een efficiënte en overzichtelijke manier werd verwerkt en gepresenteerd. Het combineren van verschillende tools en technologieën, zoals MQTT, InfluxDB en Grafana, was essentieel om tot een werkbaar eindresultaat te komen.

## 5.3. Uitdagingen

Tijdens het project ben ik verschillende uitdagingen tegengekomen die zowel technische als organisatorische aspecten omvatten. Hieronder worden de belangrijkste uitdagingen beschreven en hoe ik hiermee ben omgegaan:

### 5.3.1. Raspberry Pi opnieuw instellen

De oorspronkelijke Raspberry Pi-configuratie was verloren gegaan door een defecte SD-kaart. Hierdoor moest ik de Raspberry Pi volledig opnieuw inrichten, inclusief het installeren van de benodigde software, zoals de database (InfluxDB) en het visualisatiedashboard (Grafana). Hoewel dit tijdrovend was, bood het me de kans om het proces van begin tot eind beter te begrijpen. Door eerdere back-upcodes kon ik bepaalde onderdelen sneller herstellen, maar de database en dashboards moesten volledig opnieuw worden opgebouwd.

### 5.3.2. Tijdsgebrek

Omdat ik pas laat in het project verantwoordelijk werd voor de datavisualisatie, moest ik onder aanzienlijke tijdsdruk werken. Dit zorgde voor stress, maar motiveerde me ook om efficiënter te werken en prioriteiten te stellen. Helaas kon ik niet aan alle eisen voldoen, zoals het automatisch laten kleuren van de marker op de kaart. Als alternatief heb ik ervoor gezorgd dat het containerniveau zichtbaar werd wanneer je met de muis over de marker ging. Dit alternatief werd gelukkig positief ontvangen door de opdrachtgever.

### 5.3.3. Niet voldoen aan een eis van de opdrachtgever

Een van de eisen van de opdrachtgever was dat de marker op de kaart automatisch mee zou veranderen van kleur op basis van het containerniveau. Door de beperkte tijd en technische uitdagingen is het me niet gelukt om deze functionaliteit te implementeren. In plaats daarvan heb ik een werkbaar alternatief ontwikkeld, waarbij de informatie over het containerniveau zichtbaar wordt wanneer je met de muis over de marker beweegt. Hoewel dit niet volledig aan de oorspronkelijke eis voldeed, werd het alternatief wel gewaardeerd.

#### 5.3.4. Crash van de Raspberry Pi

Tijdens het project crashte de Raspberry Pi opnieuw, waardoor alle gegevens verloren gingen. Omdat er op dat moment nog geen back-ups waren gemaakt, moest ik het systeem opnieuw opzetten. Deze tegenslag heeft me geleerd hoe belangrijk het is om regelmatig back-ups te maken. Ondanks de frustratie heeft deze ervaring me geholpen om het proces van datavisualisatie beter te begrijpen en sneller te kunnen uitvoeren.

Deze uitdagingen hebben me gedwongen om flexibel en creatief te zijn in het vinden van oplossingen. Hoewel sommige problemen niet volledig konden worden opgelost, heb ik waardevolle lessen geleerd over het omgaan met technische beperkingen, tijdsdruk en onverwachte tegenslagen.

#### 5.4. Reflectie

Het project heeft me geconfronteerd met verschillende uitdagingen, maar uiteindelijk ook waardevolle inzichten gegeven. Toen ik laat in het proces de verantwoordelijkheid kreeg over de datavisualisatie, moest ik snel schakelen om een werkbare oplossing te realiseren.

Het opzetten van de Raspberry pi als centrale hub en het configureren van de verschillende protocollen vereiste een gestructureerde aanpak. Het verlies van bestaande configuraties en het opnieuw inrichten van het systeem was tijdrovend, maar zorgde ervoor dat ik alle stappen grondig leerde begrijpen.

Hoewel het niet mogelijk was om aan alle wensen te voldoen, ben ik erin geslaagd om alternatieven te implementeren die door de opdrachtgever positief werden ontvangen. Dit vroeg om flexibiliteit en creatief problemen op te lossen onder druk.

Deze ervaring heeft niet alleen mijn technische vaardigheden versterkt, maar ook mijn vermogen om onder stress te presteren verbeterd. Het project heeft mij waardevolle lessen geleerd over planning, communicatie en veerkracht in situaties waar onverwachte problemen optreden.

Robin van Gaal, geschreven op 24/01/25

## 6. Presentatie

Voor de presentatie aan onze opdrachtgever, Ronny Webers, heb ik de PowerPoint-presentatie ontworpen en voorbereid. Ik heb gekozen voor een professionele aanpak, waarbij de lay-out en stijl van de presentatie aansloten bij de branding van het festival Gladiolen, onze opdrachtgever. Dit zorgde ervoor dat de presentatie visueel aantrekkelijk en herkenbaar was.

### 6.1. Opbouw en inhoud

Om alles zo duidelijk mogelijk uit te leggen, heb ik een presentatie gemaakt met veel visuals, zoals foto's, filmpjes en zelfgemaakte icoontjes. Deze visuals hielpen om de technische aspecten van ons project op een eenvoudige en begrijpelijke manier over te brengen. Het gebruik van beelden maakte de presentatie niet alleen informatief, maar ook boeiend voor het publiek.

### 6.2. Voorbereiding en structuur

Om ervoor te zorgen dat de presentatie soepel verliep, heb ik een guideline (of soort script) opgesteld. Dit document hielp ons om de presentatie goed te structureren en ervoor te zorgen dat we binnen de tijd bleven. Daarnaast zorgde het ervoor dat elk teamlid evenwaardig aan bod kwam tijdens het presenteren. Zo konden we als team een professionele indruk maken en duidelijk laten zien hoe we samen aan het project hebben gewerkt.

De combinatie van een visueel sterke presentatie en een goed voorbereide structuur zorgde ervoor dat we ons project op een duidelijke en overtuigende manier konden voorstellen aan de opdrachtgever.

### 6.3. Bestanden

  
gladiolen presentatie  
final.pptm

  
tekst presentatie.docx

Mathijs Adinau, geschreven op 24/01/25



## CONTACT

Olli Gevers | Student  
R0931800@student.thomasmore.be

**VOLG ONS**