

Blueprint Project Analyse

Luchtsensor Port of Antwerp



Birk Tamm & Bavo Debraekeleer

Elektronica-ICT - Major Internet of Things

Begeleider: Patrick Van Houtven

Academiejaar 2021-2022



[GitHub Pages](#)

Versiebeheer

Nr	Datum	Verspreiding	Status	Wijziging
0.0.1	2021-11-25	Tamm Birk, Bavo Debraekeleer	Aanmaak	Repository aangemaakt + readme
0.0.2	2021-11-25	Bavo Debraekeleer, Tamm Birk	Structuur	Map structuur + analyse planning
0.1.0	2021-12-05	Bavo Debraekeleer, Tamm Birk, Patrick Van Houtven	Eerste draft	Probleemstelling + blokschema's As-Is en To-Be + planning + design + terminologie
0.1.1	2021-12-09	Bavo Debraekeleer, Tamm Birk	Verbetering	User stories analyse planning + aanpassingen probleemstelling + design
0.1.2	2021-12-15	Bavo Debraekeleer, Tamm Birk	GitHub Pages aanmaak	GitHub Pages met Docsify setup + Docsify to PDF setup + update readme en algemene opmaak
0.2.0	2021-12-15	Tamm Birk, Bavo Debraekeleer, Patrick Van Houtven	Tweede draft	Hardware analyse sensoren
0.2.1	2021-12-15	Bavo Debraekeleer, Tamm Birk	Verbetering en aanvulling	Nieuwe blokschema's As-Is en To-Be + Hardware analyse uitbreiding, argumentatie, documentatie, stroom gebruik berekeningen + layouting
0.2.2	2021-12-16	Bavo Debraekeleer, Tamm Birk	Aanvulling	Aanvulling scope + datamigratie, infrastructuur, security secties toegevoegd
0.3.0	2021-12-18	Bavo Debraekeleer, Tamm Birk, Patrick Van Houtven	Derde draft	Verdere verbeteringen adhv feedback + aanvulling scope en hardware + opmaak en readme
0.3.1	2021-12-18	Bavo Debraekeleer, Tamm Birk	Aanvulling	Foto's toegevoegd As-Is prototype v1.0 + user stories planning aangevuld + opmaak, structuur en readme aangepast + versiebeheer adhv GitHub commits

Termen en Afkortingen

Term	Omschrijving
MCU	Microcontroller, het brein van de sensor.
LoRaWAN	Long Range Wide Area Network is een specificatie voor een telecommunicatienetwerk geschikt voor langeafstandscommunicatie met weinig vermogen.
GUI of Graphical user interface	Het computersysteem waarmee de eind gebruiker de data kan op vragen en weergeven.
Field deployment	Het uitrollen van het systeem bij de klant, waaronder het plaatsen van de sensoren.
In-the-field	Ter plaatsen waar de sensor is geplaatst.
Release Plan	De planning voor het ontwikkelen van het product, na de initiële analyse.
ppm	Parts per million is een maatéénheid om concentratie aan te geven.
PM2.5 en PM10	De standaard maten voor het meten van fijnstof met grote 2.5µm en 10µm.
hPa	Hectopascal is een éénheid van luchtdruk.
NOx	Stikstofoxide
VOC	Vluchtige organische componenten, is een verzameling van allerlei vluchtige stoffen waaronder CO2 en koolmonoxide.
APICA	Antwerp Port Information & Control Assistant, de digitale assistent in de Port of Antwerp digital twin.
Digital twin	Een digitale 3D nabootsing van een echt gegeven, bvb de Port of Antwerp.
Realtime	De laatste nieuwe data wordt weergegeven van zodra ze beschikbaar is.
Timestamp	Een punt in de tijd, aanduiding met tijd en datum voor wanneer de data is opgemeten.
I/O	Inputs/Outputs, mbt inkomende en uitgaande data, of ingangen en uitgangen in een systeem.
MVP	Minimal Viable Product, is een omschrijving van de minimale kenmerken die het product versie 1.0 moet hebben om opgeleverd te worden aan de klant.
backend	Alles van code dat achter de schermen gebeurt en onzichtbaar is voor de gebruiker.
frontend	Alles wat zichtbaar is voor de gebruiker. Gelijk aan de GUI.
API	Application Programming Interface definieert toegang tot functionaliteit die

erachter schuil gaat. In geval van de luchtsensor kan je via de API de sensordata uit de database opvragen.

Probleemstelling

Opdrachtgever

Maarten Luyts, in opdracht van Port of Antwerp.

Korte samenvatting

[Schrijven op het einde 250-500 woorden + ondersteunend diagram, zie IOT template.]

Opdracht

Korte beschrijving opdracht

De Port of Antwerp is zeer geïnteresseerd in de luchtkwaliteit in de haven. Om dit in kaart te brengen zal dit projectteam een sensor ontwikkelen die de verschillende karakteristieken van luchtkwaliteit kan meten. Deze karakteristieken zijn, maar niet gelimiteerd tot:

- Temperatuur
- Vochtigheid
- Fijnstof
- NO_X

Op zich geen complexe materie om te meten. De moeilijkheid ligt echter in het in kaart brengen van de volledige haven. Om dit te kunnen realiseren moet de te ontwikkelen sensor volledig draadloos zijn. Er zal draadloze technologie nodig zijn en één of andere vorm van batterij als stroomvoorziening. De Port of Antwerp heeft onlangs in de haven een LoRaWAN netwerk uitgerold. Deze Low Power Wide Area Netwerk technologie is uiterst geschikt voor de te ontwikkelen sensor. De integratie van LoRaWAN in de sensor is dus ook een vereiste. Op AP en daarbuiten is er al wat geëxperimenteerd met de sensors rond de luchtkwaliteit. Het innovatieve gedeelte ligt vooral rond het volledig draadloze aspect. Voordat gestart wordt met de analyse is het aangeraden om hierover opzoekingswerk te verrichten.

Verwachte output

Prototype in een oplage van 10 stuks met volgende eigenschappen:

- sensor metingen:
 - temperatuur
 - vochtigheid
 - fijnstof
 - optioneel: NO_X
- LoRaWAN draadloze connectiviteit
- gebruikt low power MCU
- behuizing geschikt voor buiten
- batterijduur van meer dan 1 jaar
- user interface om de data op een landkaart weer te geven

Situatie As-Is

De huidige luchtsensor waarop we verder werken is met een ander doel ontwikkeld. Deze dient om de luchtkwaliteit in en buiten de stad Antwerpen in kaart te brengen voor collega Chemie studenten. Deze luchtsensor is mobiel bruikbaar en slaat de data lokaal op aan de hand van een SD-kaart. Het doel was om de data door te sturen, via WiFi of seriëel, naar een database. Dit werd niet gerealiseerd. Er is wel een LCD-display aan toegevoegd om de sensordata in real-time te kunnen zien, en een aan/uit knop voor de gebruiker.



Eigenschappen:

- sensor metingen:
 - CO2
 - TVOC
 - fijnstof
 - temperatuur
 - luchtvochtigheid
- WiFi en Bluetooth connectiviteit
- SD-kaart voor lokale data opslag
- GPS module
- mobiele behuizing met LCD
- batterijduur onbekend

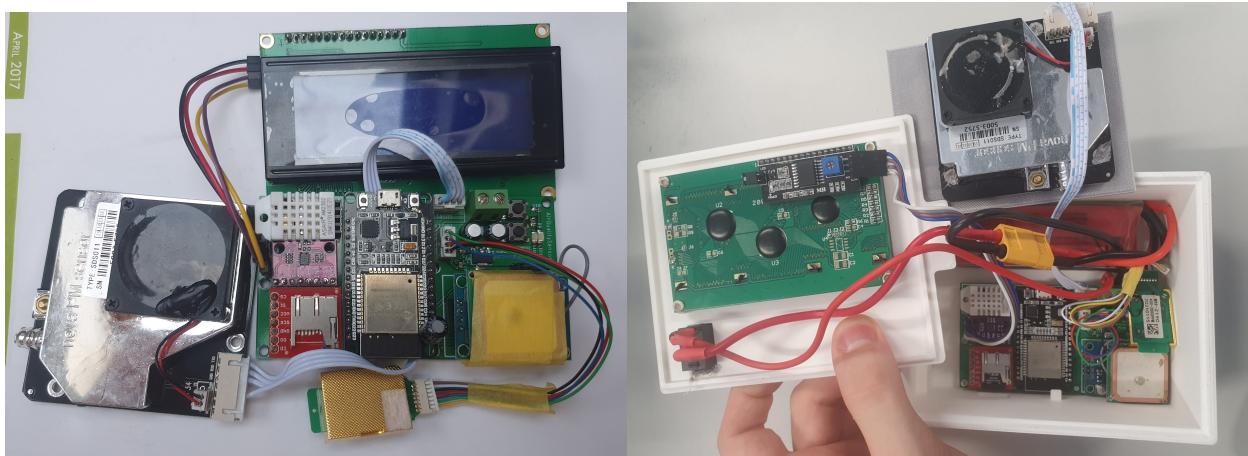
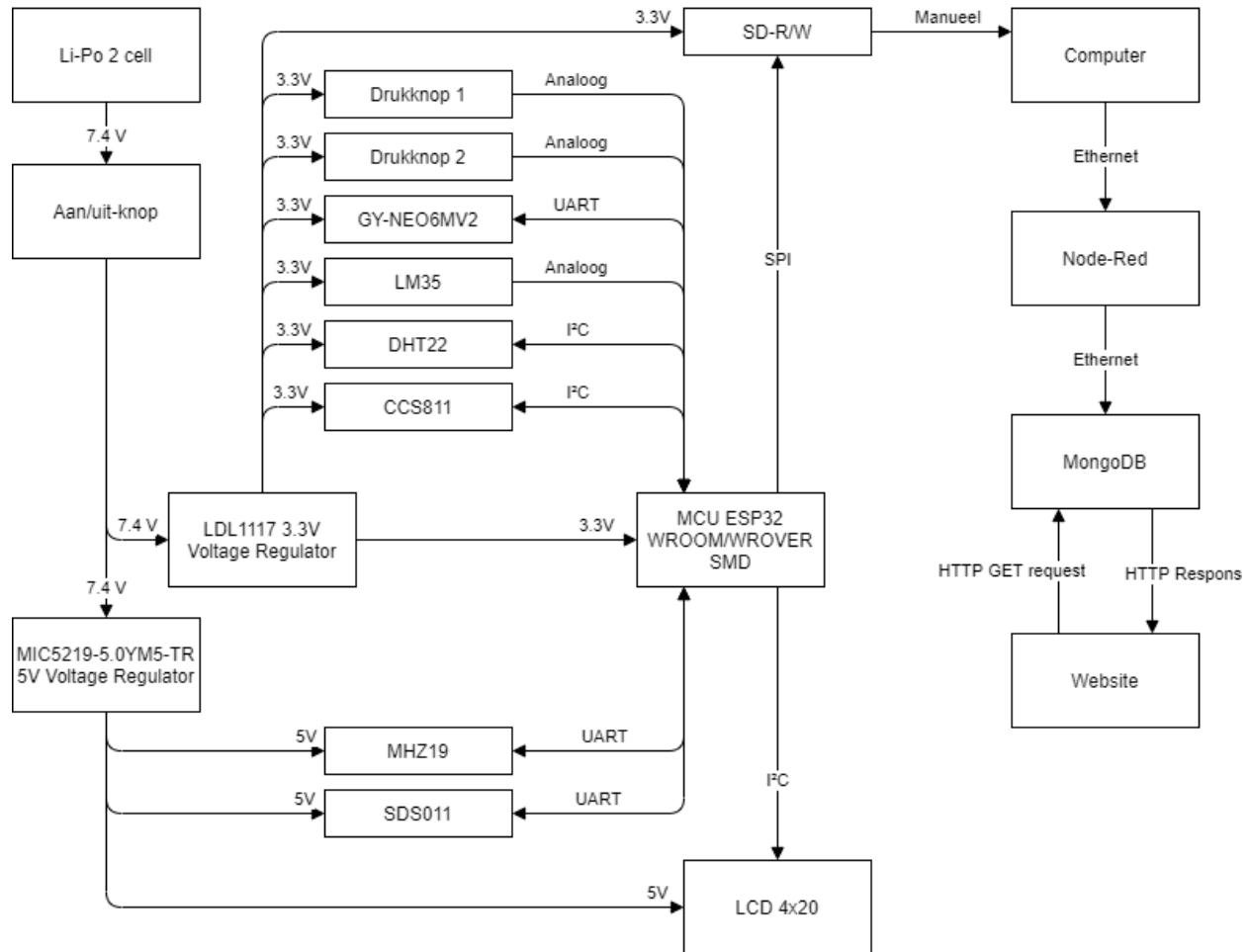
Werking

Zoals beschreven in Product As-Is v1.0

- De ESP zal om de x-aantal tijd alle sensoren aanroepen en hiervan de waarden opvragen.

- De sensordata wordt in een JSON bestand opgeslagen op de SD-Card.
- Nadien kan de gebruiker deze SD-card in zijn computer uitlezen en de data met een simpele knop overzetten naar de MongoDB.
- Op de website kunnen we dan de data selecteren aan de hand van naam en datum.
- Op de ingebouwde LCD kan men de huidige metingen, datum, tijd en coördinaten zien.

Blok diagram



Probleemstelling

Het huidige prototype moet geoptimaliseerd worden zodat deze meer dan één jaar kan werken op een batterij. Om dit te realiseren is een andere MCU nodig en dienen er aanpassingen gemaakt te worden aan de keuze van componenten. De sensor moet ook volledig autonoom kunnen werken, zodat deze ergens geplaatst kan worden zonder bijkomend onderhoud tot de batterij leeg is. Voor de meest accurate metingen is het echter wel aangeraden de fijnstofsensor de calibreren bij seizoenswisselingen. Het gaat nu ook over het in kaart brengen van de volledige haven waarbij de sensors een vaste locatie krijgen. Nu gebeurt de dataoverdracht van sensor naar database manueel via een SD-kaart. Dit moet geautomatiseerd worden en draadloos gebeuren. De luchttoevoer naar de sensor en koeling moet ook verbeterd worden. Er is een kleine FAN op de fijnstofsensor in combinatie met een klein gat in de behuizing. Dit zorgt echter niet voor voldoende lucht toevoer om correcte metingen te bekomen. De warmte die de componenten ontwikkelen hebben hier ook invloed op de temperatuur meting, wat natuurlijk niet gewenst is. De luchtdoorstroming moet hier ook voor voldoende koeling zorgen. Belangrijk hierbij is wel dat er zo weinig mogelijk vocht en grotere stofdeeltjes in de sensor geraken. Ook moet de sensor in de schaduw geplaatst worden of voorzien worden van een zonwerende coating of film.

Situatie To-Be

Doelstelling

Het doel is het in kaart brengen van de luchtkwaliteit in de gehele Port of Antwerp met volledig draadloze sensoren die voor meer als één jaar autonoom kunnen werken. De data verzenden vanuit de sensor en ontvangen door een server zal gebeuren met LoRaWAN. Er moet een nieuwe behuizing ontworpen worden waar meer rekening wordt gehouden met luchtdoorstroming en temperatuur zoals beschreven in de probleemstelling. Dit ontwerp moet ook worden doorgevoerd naar het PCB ontwerp en de plaatsing van de sensoren.

De opdracht laat de GPS module uit de As-Is sensor achterwegen wegens extra stroomverbruik. Wij willen argumenteren om deze toch te behouden. Om het in kaart brengen van de luchtkwaliteit zo accuraat en eenvoudig mogelijk te maken is het aangeraden om coördinaten te hebben van waar elke sensor zich bevindt. Zo kunnen we een exacte locatie geven aan de data gemeten door elke sensor. Rekening houdend met de realiteit dat een sensor al eens verplaatst, vervangen of hersteld zal worden, kan een GPS module zorgen voor een eenvoudige manier om aan accurate coördinaten te komen. Door de integratie van de GPS module in de sensor kan dit volledig automatisch gebeuren, zonder dat er extra handelingen vereist zijn van de gebruiker.

De weergave van de data zal gebeuren met de eigen digital twin die de Port of Antwerp in ontwikkeling heeft. De data moet beschikbaar gemaakt worden in een API. Zo kan de digital twin de data ophalen en weergeven in 2D en 3D representatie van de haven.

Projectdefinitie

Minimal Viable Product

Prototype met volgende eigenschappen:

- sensor metingen uitvoeren:

- temperatuur
- vochtigheid
- fijnstof

- werking op batterij
- behuizing 3D print

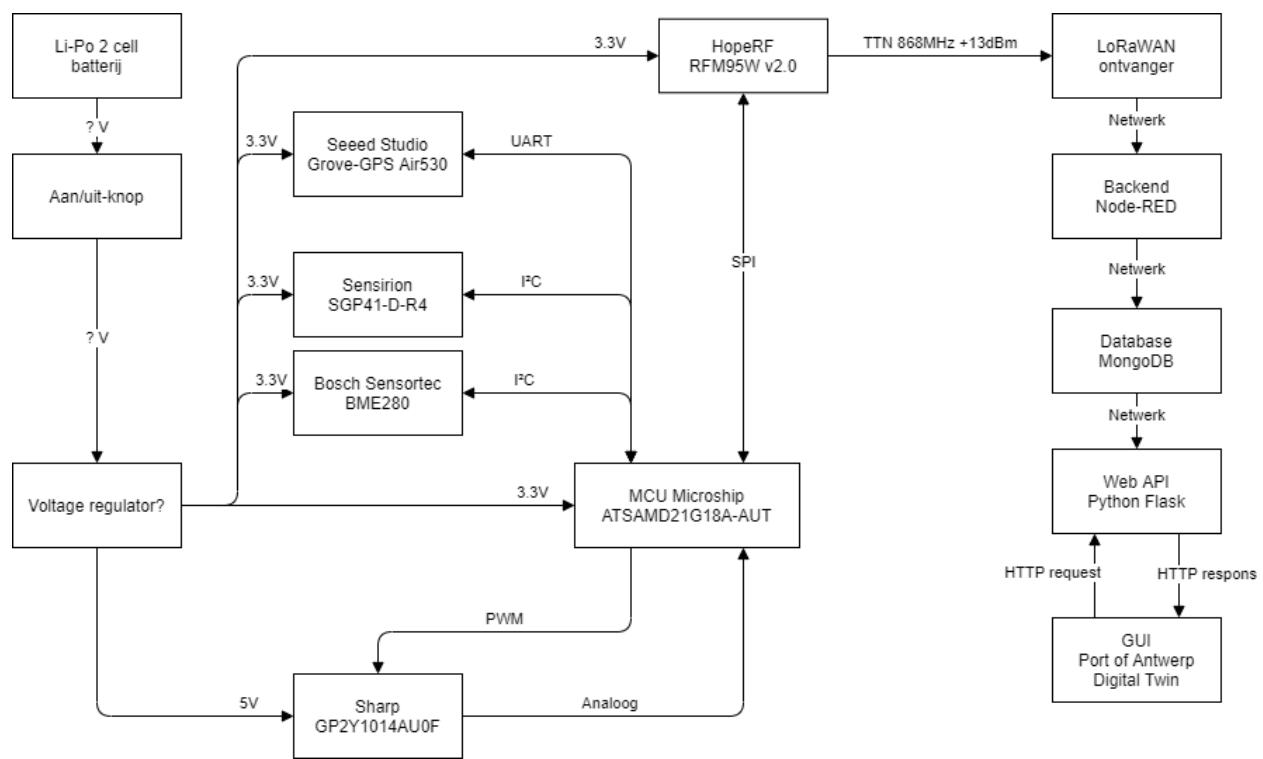
- sensordata verzenden met LoRaWAN
- sensordata ontvangen en opslaan

Epics

1. Metingen uitvoeren
2. Sensordata verzenden
3. Sensordata beheer
4. Weergave op kaart

Blok diagram

Door de grote wijziging in doel tussen de As-Is en To-Be situaties wordt zowat elk blok van de sensor aangepast. De algemene werking blijft hetzelfde: aan/uit → metingen uitvoeren → sensordata verzamelen → verzenden → in database opslaan → weergeven in GUI. De effectieve werking zal voor elk blok echter verschillen door de andere component keuzes.



Scope

- Component selectie, totaalprijs < €100:

- microcontroller: $I_{max} < 10mA$
- GPS module: low-power $I_{max} <$ As-Is 67mA
- LoRaWAN module: zendbereik hele haven over 12.068 ha = 120.680 km²

- fijnstofsensor: PM2.5 en PM10 concentratie, meetbereik 0-40 µg/m³
- temperatuur sensor: meetbereik -20° ~ +42° ([kouste en warmste meting in België](#))
- luchtvochtigheid sensor: meetbereik 0% ~ 100% ([relatieve luchtvochtigheid](#))
- NO_X sensor: concentratie, meetbereik 0-40 µg/m³

- Elektrisch schema ontwerp voor de PCB
- PCB en behuizing ontwerp met aandacht voor sensor plaatsing voor zo accuraat mogelijke metingen
- Schroefdraad in behuizing als bevestigingspunt voor een ophangsysteem
- Initiële calibratie fijnstofsensor: de analoge spannings output bij 0mg/m³ opmeten en aftrekken van de output V₀
- Backend en API code

Niet in scope

- LoRaWAN ontvanger opzetten
- Servers en server onderhoud
- Field deployment/plaatsing van de sensors en het ophangsysteem
- In-the-field kalibratie
- Backend deployment
- Implementatie in de digital twin

Planning

Analyse

We gebruiken Agile en Kanban om flexibel te werken en zo snel mogelijk vooruit te gaan. Elke week tijdens het labo krijgen we feedback van onze begeleider meneer Van Houtven. Waarna we verbeteringen kunnen uitvoeren en de planning bijsturen. Ten opzichte van de opdrachtgever is het analyse proces wel eerder Waterfall. Er is enkel een interview geweest aan de start van de analyse. Om de planning op te volgen werken we met een Kanban bord op GitHub Projects.

Epics (hoofdlijnen)

1. Probleemstelling + planning
2. Design + hardware analyse
3. Hardware schema + I/O software
4. Software analyse + datamigratie + infrastructuur + security + documentatie
5. Release Plan + samenvatting + afwerking

Toelichting fasen

Dit is de ruwe volgorde waarin de analyse wordt uitgevoerd door de opvolgbaarheid van de onderdelen. We kunnen bijvoorbeeld geen software analyse doen zonder de hardware outputs te kennen. We werken wel Agile dus zullen gaande weg aanpassingen maken doorheen de analyse.

1. Probleemstelling + planning

- Omschrijving van de opdrachtgever, korte samenvatting, de situatie as-is, en de situatie to-be met een projectdefinitie (doelstelling, blokschema's, scope, niet in scope).
- Planning met user stories voor de Analyse volgens de Kanban methodologie.
- Omschrijven van het functioneel design, en in het algemeen het technisch design.

2. Design + hardware analyse

Vooraleer je de software kan analyseren moet je weten waarop de software gaat draaien.
Vandaar dat we de hardware als eerste analyseren.

3. Hardware schema + I/O software

Eens je weet welke componenten je hebt en hoe ze tussen mekaar verbonden zijn in het hardware schema kan je de inputs en outputs oplijsten. Dit is handig voor de diagrammen later.

4. Software analyse + datamigratie + infrastructuur + security + documentatie

- Nu dat de inputs en outputs gekend zijn is het veel eenvoudiger om de diagrammen op te stellen van de software. Eens die diagrammen er zijn is het mogelijk om het dashboard te analyseren.
- Omschrijven of er datamigratie nodig is en hoe dit zal gebeuren.
- Omschrijven of er impact is op de huidige infrastructuur.
- Zwakke punten rond beveiliging aangeven en wat er aan gedaan zal worden.
- Omschrijven hoe er zal worden omgegaan met data en privacy.
- Omschrijven wat de verschillende autorisatierollen zullen zijn.
- Omschrijven welke documentatie zal worden voorzien en hoe dit zal gebeuren.

5. Release Plan + samenvatting + afwerking

- We weten nu wat er gemaakt moet worden dus kunnen we het Release Plan, met Gantt Chart methodologie, uitwerken.
- We kunnen nu een samenvatting schrijven van het gehele project.
- Als laatste zorgen we nog dat er een terminologie tabel is, en dat de versiecontrole en bronvermelding volledig is.

User stories

[- Als [PERSOON] wil ik [...], zodat ik [...].]

1. Probleemstelling + planning:

- Als GELDSCHIETER wil ik weten wie de opdrachtgever is en wat de opdracht in het kort is, zodat ik een snel idee kan krijgen van het project.
- Als GELDSCHIETER wil ik weten wat de huidige as-is situatie is van het vorige project en de probleemstelling adhv dit huidige proces, zodat ik goed weet hoever het nu staat, wat de eventuele moeilijkheden zijn en hoe er hierop gaat voortgebouwd worden.
- Als GELDSCHIETER wil ik weten wat de situatie to-be moet worden, wat de doelstelling van het project is, zodat ik kan inschatten of het haalbaar is.
- Als OPDRACHTGEVER wil ik weten wat wel en niet binnen de scope van het project valt, zodat ik weet wat ik juist zal krijgen en wat ik eventueel zelf nog moet voorzien.
- Als ONTWIKKELAAR wil ik weten wat de planning is, zodat ik weet wat wanneer af moet zijn en om het project op te volgen.

2. Design + hardware analyse:

- Als ONTWIKKELAAR wil ik weten wat het FUNCTIONEEL DESIGN is, zodat ik weet hoe de eind gebruiker met het product zal kunnen omgaan.
- Als ONTWIKKELAAR wil ik weten wat het TECHNISCH DESIGN is, zodat ik weet wat de architectuur is, welke technologieën zullen gebruikt worden, en hoe het project technisch zal worden gerealiseerd.
- Als OPDRACHTGEVER wil ik weten welke HARDWARE COMPONENTEN er gaan gebruikt worden en waarom, zodat ik weet wat de sensor zal kunnen en de onderdelen zullen kosten.
- Als HARDWARE ONTWIKKELAAR wil ik weten wat voor HARDWARE SMART OBJECT moet worden ontwikkeld, zodat ik een leidraad heb voor de ontwikkeling van de sensor.

3. Hardware schema + I/O software:

- Als HARDWARE ONTWIKKELAAR wil ik een ELEKTRISCH SCHEMA, zodat ik het PCB kan ontwerpen.
- Als SOFTWARE ONTWIKKELAAR wil ik de INPUTS EN OUTPUTS van de verschillende blokken weten, zodat ik weet waarmee ik in de software moet werken.

4. Software analyse + datamigratie + infrastructuur + security + documentatie:

- Als SOFTWARE ONTWIKKELAAR wil ik een SOFTWARE SMART OBJECT, zodat ik weet wat de software in zijn geheel gaat moeten doen.
- Als SOFTWARE ONTWIKKELAAR wil ik een STATEDIAGRAM van de software, zodat ik weet welke verschillende states moeten worden ingebouwd.
- Als SOFTWARE ONTWIKKELAAR wil ik een FLOWCHARTS voor de verschillende states van de software, zodat ik weet hoe elke transitie moet gebeuren.
- Als OPDRACHTGEVER wil ik een MOCK UP zien van hoe de GUI er zal uitzien, zodat ik kan zien of het voldoet aan mijn wensen en eventuele feedback kan geven.
- Als OPDRACHTGEVER wil ik weten of er eventuele DATAMIGRATIE moet gebeuren en hoe, zodat ik weet hoe dit zal gebeuren.
- Als GELDSCHIETER wil ik weten wat de IMPACT is op de HUIDIGE INFRASTRUCTUUR, zodat ik rekening kan houden met de eventuele kosten.
- Als OPDRACHTGEVER wil ik weten wat er rond SECURITY EN PRIVACY zal gedaan worden, zodat ik weet hoe het project zal beveiligd worden en hoe er zal omgegaan worden met data privacy.
- Als SOFTWARE ONTWIKKELAAR wil ik weten welke AUTORISATIEROLLEN er moeten zijn, zodat ik deze kan inbouwen.
- Als HARDWARE ONTWIKKELAAR wil ik weten wat de HARDWARE SECURITY moet zijn, zodat ik deze kan inbouwen in de hardware.
- Als ONTWIKKELAAR wil ik weten welke DOCUMENTATIE er moet worden voorzien, zodat ik weet wat ik moet schrijven.

5. Release Plan + samenvatting + afwerking:

- Als ONTWIKKELAAR wil ik een RELEASE PLAN, zodat ik een idee heb van hoe het ontwikkelingsproces er zal uitzien.
- Als LEZER van de analyse wil ik een KORTE SAMENVATTING, zodat ik snel kan weten waarover deze analyse gaat.
- Als ANALYST wil ik een VERSIECONTROLE tabel kunnen raadplegen, zodat ik een overzicht heb van wat er gedaan is.
- Als LEZER van de analyse wil ik een tabel met TERMINOLOGIE, zodat ik alles kan begrijpen.
- Als LEZER van de analyse wil ik een BRONVERMELDING, zodat ik de correctheid en betrouwbaarheid van de data kan nagaan.

Release Plan

Epics (hoofdlijnen)

- 1. Metingen uitvoeren
- 2. Sensordata verzenden
- 3. Sensordata beheer
- 4. Weergave op kaart

Toelichting fases

1. Metingen uitvoeren

Alles wat er moet gebeuren om metingen te kunnen uitvoeren.

2. Sensordata verzenden

Voorbereiding van de sensordata voor verzending, het verzenden zelf en het ontvangen.

3. Sensordata beheer

Alles omtrent het opslaan en opvragen van de sensordata in de database.

4. Weergave op kaart

De sensordata weergeven in de GUI.

User stories

[- Wanneer [situatie], wil ik [motivatie, waarom], zodat ik [verwachte uitkomst].] [Deze stories helpen ons om een bepaalde situatie vast te leggen, welke handeling hierop moet gebeuren en wat de verwachte uitkomst is van deze handeling.]

1. Metingen uitvoeren:

- Wanneer de SENSOR GEPLAATST wordt, wil ik dat de sensor aangezet of gereset wordt, zodat de sensor zijn GPS LOCATIE kan bepalen en starten met metingen uit te voeren.
- Wanneer er een TEMPERATUUR EN LUCHTVOCHTIGHEID METING moet worden uitgevoerd,
- Wanneer er een FIJNSTOF METING moet worden uitgevoerd,
- Wanneer er een STIKSTOFOXIDEN METING moet worden uitgevoerd,
- Wanneer alle SENSOR DATA OUTPUTS in de MCU zijn binnengekomen, wil ik dat deze worden GEFORMATTEERD IN EEN BESTAND (JSON), zodat alles bij elkaar staat.

2. Sensordata verzenden:

- Wanneer de SENSORDATA moet worden VERZONDEN, wil ik dat de data wordt GEËNCRYPTEERD en GEFORMATTEERD IN BYTES, zodat deze draadloos kan worden verzonden.
- Wanneer de SENSORDATA wordt ONTVANGEN, wil ik dat de data wordt GEHERFORMATTEERD IN EEN BESTAND, zodat deze kan worden gevalideerd en klaar is om opgeslagen te worden.

3. Sensordata beheer:

- Wanneer de SENSORDATA klaar staat om te worden OPGESLAGEN, wil ik dat deze in de DATABASE komt, zodat het kan worden opgevraagd.
- Wanneer de SENSORDATA wordt OPGEVRAAGD, wil ik dat deze wordt GEDECRIPTEERD, zodat de data leesbaar is.
- Als ONTWIKKELAAR wil ik de RUWE SENSORDATA BEKIJKEN, om deze te kunnen valideren.
- Als ONTWIKKELAAR wil ik dat de SENSORDATA BESCHIKBAAR is in een API, zodat ik deze kan opvragen vanuit de digital twin.

4. Weergave op kaart:

- Als GEBRUIKER wil ik de SENSORDATE IN KAARTVORM BEKIJKEN, zodat ik een overzicht heb van de metingen.
- Als GEBRUIKER wil ik de SENSORDATA PER SENSOR IN DETAIL BEKIJKEN, zodat ik de exacte metingen kan raadplegen van een bepaalde locatie en tijd.
- Als GEBRUIKER wil ik de SENSORDATA BESCHIKBAAR hebben in de DIGITAL TWIN van de Port of Antwerp, zodat deze daar beschikbaar is en automatisch geanalyseerd kan worden door [APICA](#).

Lead times

De lead times van de aan te kopen componenten, materialen, en de te produceren PCB's.

Zie Hardware Analyse voor lead times componenten. PCB levertermijnen variëren van 5 tot 20 werkdagen.

Gantt chart

Design

Functioneel design

- Component selectie:
 - microcontroller: $I_{max} < 10mA$
 - GPS module: low-power $I_{max} < A_s-I_s 67mA$
 - LoRaWAN module: zendbereik hele haven over 12.068 ha = 120.680 km²
 - fijnstofsensor: PM2.5 en PM10 concentratie, meetbereik 0-40 µg/m³
 - temperatuur sensor: meetbereik -20° ~ +42° ([kouste en warmste meting in België](#))
 - luchtvochtigheid sensor: meetbereik 0% ~ 100% ([relatieve luchtvochtigheid](#))
 - NO_X sensor: concentratie, meetbereik 0-40 µg/m³
- Behuizing die in openlucht kan geplaatst worden en bestand is tegen het lokale klimaat.
- Behuizing die voor voldoende koeling zorgt zodat de temperatuur metingen accuraat zijn.
- Behuizing die voor voldoende lucht toevoer zorgt aan de fijnstofsensor.
- Schroefdraad in behuizing als bevestigingspunt voor een ophangsysteem.
- De sensor moet automatisch om de .. minuten metingen uitvoeren en verzenden (er is geen interactie).
- Sensor output als JSON data:
 - locatie: GPS coördinaat van de sensor
 - timestamp: datum en tijd van de meting
 - temperatuur in °C
 - luchtvochtigheid in %
 - fijnstof PM2.5 en PM10 in aantal deeltjes
 - NO_X stikstofoxiden in µg/m³
- Server met database om de sensor data in op te slaan en van op te vragen.
- Weergave van de sensordata in GUI digital twin van de Port of Antwerp.

Technisch design

Technologieën

Onderdeel	Technologie	Argumentatie
Backend code	Node-RED	Node-RED is een stroomgebaseerde ontwikkeltool voor visuele programmering. Dit wordt al gebruikt in de As-Is situatie. Het is dus tijd en werk besparend om hiermee verder te werken.
Database	MongoDB	MongoDB is een opensource document-georiënteerde database waarin documenten als BSON (Binary-JSON) wordt opgeslagen. Ideaal voor het werken met JSON bestanden, en ook wat in de As-Is situatie gebruikt wordt.
Web API	Python Flask	Flask is een light-weight web framework in Python. Hiermee kan je zelf bepalen welke componenten en extension je wilt gebruiken voor de API. Wij gebruiken (tutorial): <ul style="list-style-type: none">• MongoEngine voor het omgaan met MongoDB in een objectgeoriënteerde manier.• Bcrypt voor encryptie.• RESTful voor een REST API aan te maken.• JWT-Extended voor authenticatie en autorisatie.
Frontend	Digital twin Port of Antwerp	Deze wordt momenteel ontwikkeld en is ideaal om onze sensordata in weer te geven.

Technisch realisatieproces prototype

1. Aankopen en testen van de componenten. Het testen van de sensors en modules kan met een Arduino gebeuren. Dit om eventuele moeilijkheden met de microcontroller hiervan los te koppelen.
2. De PCB en behuizing ontwerpen in Fusion 360 samen met de sensor plaatsing onderzoeken. Dit met oog op zo accuraat mogelijke metingen te bekomen en onderlinge invloeden te minimaliseren.
3. PCB ontwerp bestellen en testen. De behuizing 3D printen en testen.
4. Code ontwikkelen om de metingen uit te voeren.

5. Code ontwikkelen om de sensordata te verzenden met LoRaWAN.
6. Backend ontwikkelen met Node-RED om de sensordata te ontvangen en op te slaan in een MongoDB database.
7. Het gehele proces van metingen uitvoeren en de sensordata te verzenden en ontvangen automatiseren.
8. API ontwikkelen om de sensordata uit de database op te vragen, aanspreekbaar door de digital twin van de Port of Antwerp.

Hardware Analyse

Microcontroller

Naam	Eigenschappen	Argumentatie	Data
Microchip ATSAMD21G18A-AUT	<ul style="list-style-type: none">• Low-power MCU• $V_{cc} = 1.62V - 3.63V$• $I_{max} = 4.7mA$• CORTEX-M0+PROCESSOR aan 48MHz• USB naar UART aanwezig op de chip• 256KB FLASH geheugen• 32-bit RTC• 6 SERCOM interfaces• <i>Eenheidsprijs vanaf 10 stuks: €3,60</i>• <i>Lead time: 54 weken</i>	Uit het interview met de opdrachtgever gaf hij aan dat we deze MCU moeten gebruiken. Deze is low-power en dus zeer geschikt voor de applicatie.	Datasheet Documentatie fabrikant Winkel

Modules

Naam	Eigenschappen	Argumentatie	Data
Grove GPS Air530	<ul style="list-style-type: none">• GPS localisatie• $V_{cc} = 3.3V / 5V$• $I_{max} = 60mA$	Deze GPS module is zeer energie zuinig met verschillende low-power modi. Ze is ook relatief goedkoop en werkt op 3.3V.	Handleiding Documentatie fabrikant Winkel (fabrikant) Winkel

	<ul style="list-style-type: none"> • Capture current value@3.3V = 42.6mA • Tracking current value@3.3V = 36.7mA • Low power mode@3.3V = 0.85mA • Ultra-low-power mode@3.3V = 31uA • UART protocol • Time of warm start: 4s • Time of cold boot: 30s • <i>Eenheidsprijs vanaf 10 stuks: €8,98</i> • <i>Lead time: 11 weken</i> 	Booklet (Chinese)
HopeRF RFM95W v2.0 LoRa Module	<ul style="list-style-type: none"> • LoRaWAN module • $V_{CC} = 1.8V - 3.7V$ • $I_{+13dBm(Typ)} = 29mA$ • $P_{output} = +13dBm$ • SPI interface • Payload lenght max 64 bytes • Werkingstemperatuur: $-20^{\circ}C \sim +70^{\circ}C$ • Lower bands: 169 MHz and 433 MHz, higher bands: 868 and 915 MHz • <i>Eenheidsprijs vanaf 10 stuks: €16,36</i> • <i>Lead time: 17 weken</i> 	<p>Een veel gebruikte module waardoor er veel documentatie over te vinden is. Ook relatief klein.</p> <p>Datasheet Documentatie fabrikant Winkel</p>

Sensoren

Naam	Eigenschappen	Argumentatie	Data
Sharp		De Nova SDS011 die bij	Datasheet

GP2Y1014AU0F	<ul style="list-style-type: none"> • Fijnstof • $V_{cc} = 4.5V - 5.5V$ • $I_{max} = 20mA$ • Analoog • Output voltage: 0.9~3.7V (zie grafiek) • Meetbereik PM2.5 en PM10: 0~50$\mu g/m^3$ (zie grafiek) • <i>Eenheidsprijs vanaf 10 stuks: €6,25</i> • <i>Lead time: 53 weken</i> 	<p>de As-Is sensor gebruikt wordt heeft als grote nadeel dat er een FAN zit ingebouwd om lucht in de sensor te krijgen. Dit zorgt voor extra stroomverbruik en werkte in het prototype daarbij nog niet voldoende. De Sharp GP2Y1010AU0F komt uit test op vlak van accuraatheid even goed uit als de SDS011, maar heeft het voordeel van geen FAN te hebben, kleiner te zijn, energiezuiniger en goedkoper.</p>	GP2Y1014AU0F Datasheet voorganger GP2Y1010AU0F Application notes GP2Y1010AU0F Documentatie fabrikant Winkel
Bosch Sensortec BME280	<ul style="list-style-type: none"> • Temperatuur, barometer en luchtvochtigheid • $V_{cc} = 3.3V$ • $I_{max} = 4.5mA$ • I²C protocol 3.3V, en SPI • Meetbereik temperatuur: -40°C ~ +85°C • Meetbereik luchtvochtigheid: 0% ~ 100% • Meetbereik luchtdruk: 300hPa ~ 1100hPa • Leessnelheid: 1Hz (1s) • <i>Eenheidsprijs vanaf 10 stuks: €5,42</i> • <i>Lead time: tijdelijk beperkt aanbod op moment van schrijven</i> 	<p>Deze IC heeft als voordeel beide temperatuur en luchtvochtigheid te meten. Het is ook een SMD component dus eenvoudig op een PCB te krijgen. Hier moeten we wel rekening houden met de warmte ontwikkeling van andere componenten die de temperatuur metingen kunnen beïnvloeden.</p>	Datasheet Documentatie fabrikant Winkel
Sensirion SGP41-D-R4	<ul style="list-style-type: none"> • NO_x en VOC • $V_{cc} = 1.7V - 3.6V$ 	Klein low-power SMD component Ideaal voor deze toepassing	Datasheet Documentatie fabrikant Winkel

- Werkvoedingsspanning
= 3.3V
- $I_{max} = 4.6\text{mA}$
- $I_{nominaal} = 3.2\text{mA}$
- I²C interface
- Meetbereik NO₂: 0-
10.000ppb
- Werkingstemperatuur:
-20°C ~ +55°C
- SMD
- *Eenheidsprijs vanaf 10 stuks: €7,46*
- *Lead time: 16 weken*

Voeding

Zoals bij de As-Is sensor hebben we onderdelen die werken op 3.3V en op 5V.

Voor 5V behouden wij er slechts één, de Sharp GP2Y1014AU0F met $I_{max} = 20\text{mA}$.

Voor de 3.3V componenten moeten we een herberekening doen:

Herberekeningen voor 3.3V

$$I_{max} = I_{ATSAMD21G18A-AUT} + I_{BME280} + I_{SGP41} + I_{Grove GPS Air530} + I_{RFM95W \text{ LoRa}}$$

$$I_{max} = 4.7\text{mA} + 4.5\text{mA} + 4.6\text{mA} + 60\text{mA} + 29\text{mA} = 102.8\text{mA}$$

Om wat marge te hebben doen we deze waarde maal 1.2 :

$$102.8\text{mA} \cdot 1.2 = 123.4\text{mA}$$

De spanningsvereisten liggen dit maal dus een stuk lager:

Voor 5V 20mA To-Be tegenover 390mA As-Is.

Voor 3.3V 123.4mA To-Be tegenover 822.6mA As-Is.

Naam	Eigenschappen	Argumentatie	Data
Naam	<ul style="list-style-type: none"> • Batterij • $V_{out} = V$ • $I_{max} = \text{mA}$ 	Argumentatie	Datasheet Documentatie fabrikant Winkel

	<ul style="list-style-type: none"> • Operating Temperatuur = • SOT-.. 		
Naam	<ul style="list-style-type: none"> • 5V Voltage Regulator • $V_{in} = V$ • $V_{out} = 5V$ • $I_{max} = mA$ • Operating Temperatuur = • SOT-.. 	Argumentatie	<u>Datasheet</u> <u>Documentatie</u> <u>fabrikant</u> <u>Winkel</u>
Naam	<ul style="list-style-type: none"> • 3.3V Voltage Regulator • $V_{in} = V$ • $V_{out} = 5V$ • $I_{max} = mA$ • Operating Temperatuur = • SOT-.. 	Argumentatie	<u>Datasheet</u> <u>Documentatie</u> <u>fabrikant</u> <u>Winkel</u>
2N7002K-7	<ul style="list-style-type: none"> • MOSFET • $V_{GS(th)} = 1 - 2.5V$ • $V_{DSS(max)} = 60V$ • $I_{max} = 380mA$ • SOT-23 • <i>Eenheidsprijs vanaf 10 stuks: €0,169</i> • <i>Lead time: 89 weken</i> 	De 2N7002K-7 zal gebruikt worden om de modules en sensoren, zonder sleep mode, uit te zetten tijdens de intervallen. (Overgenomen van As-Is)	<u>Datasheet</u> <u>Documentatie</u> <u>fabrikant</u> <u>Winkel</u>

Alternatieven

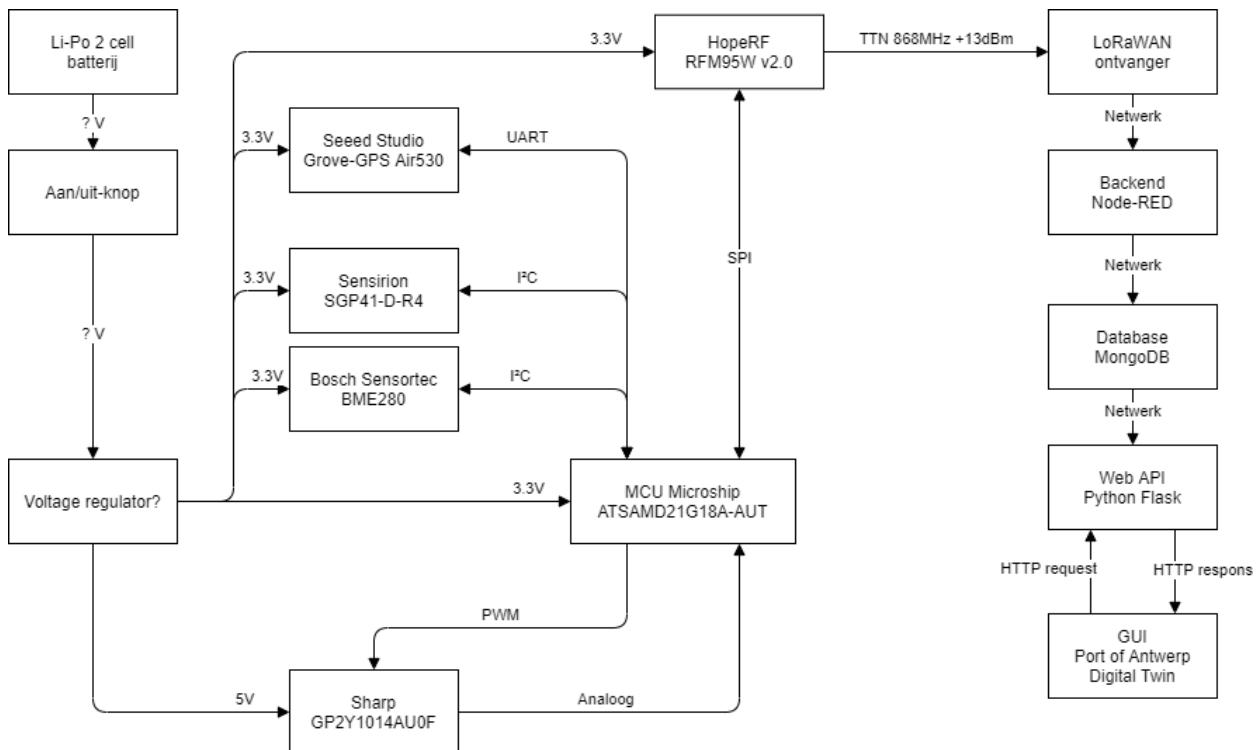
Naam	Eigenschappen	Argumentatie	Data
Naam	• Type	Argumentatie	<u>Datasheet</u> <u>Documentatie</u> <u>fabrikant</u> <u>Winkel</u>

Software Analyse

Data I/O

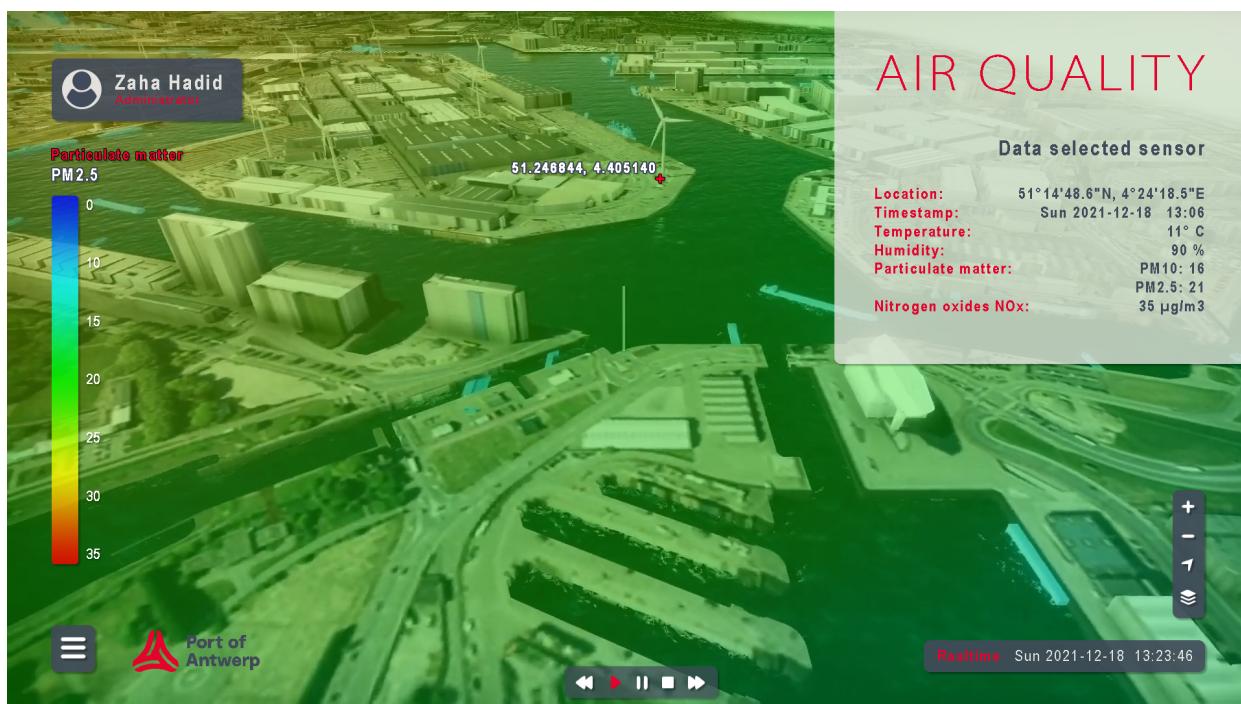
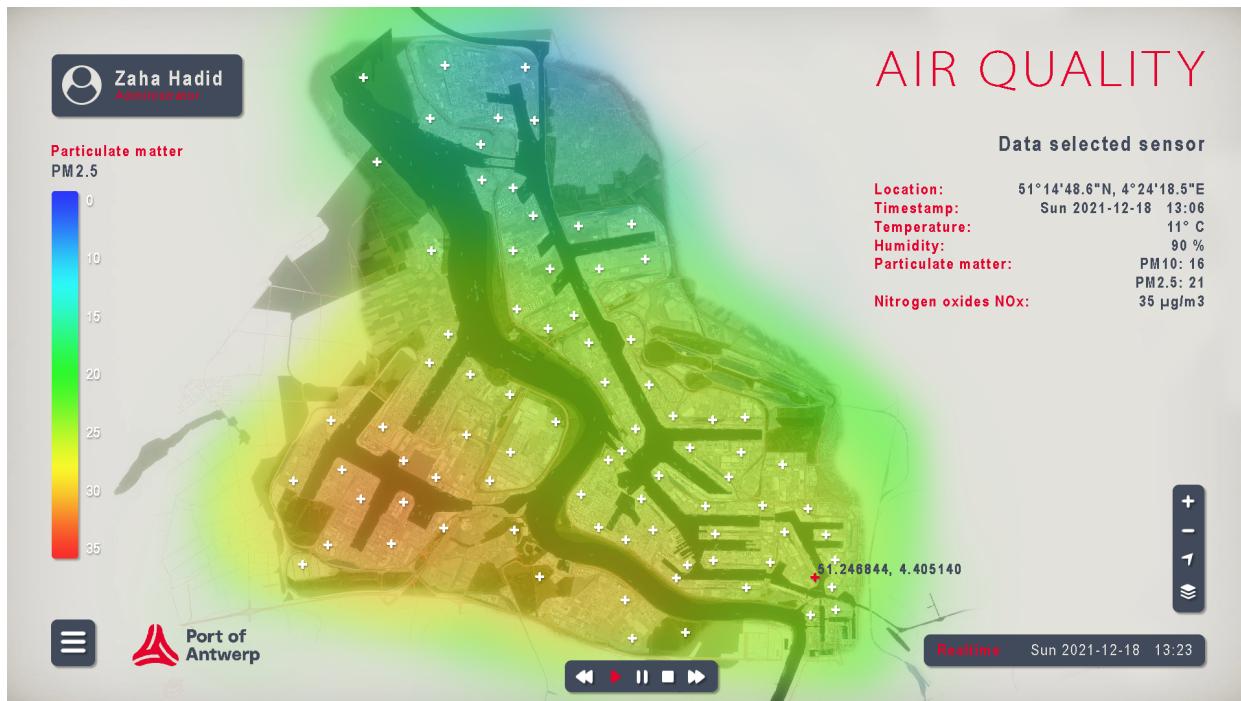
Blok	Data input	Data output
MCU: Microchip ATSAMD21G18A-AUT	<ul style="list-style-type: none"> Sensordata fijnstof, temperatuur, luchtvochtigheid en stikstofoxiden, I²C, Analoog GPS NMEA0183 V4.1 protocol data, UART LoRa module feedback, SPI 	<ul style="list-style-type: none"> Configuratie en aansturing sensors en modules, serieel UART, I²C, PWM, SPI Sensordata als JSON object, SPI
Fijnstofsensor: Sharp GP2Y1014AU0F	PWM om LED aan te sturen ($T = 10\text{ms}$; $P_w = 0.32\text{ms}$), PWM	Analoge spanning pulse (Volledig lezen na 0.28ms), Analoog
Temperatuur en luchtvochtigheid sensor: Bosch Sensortec BME280	Uit te lezen register, I²C	Register data uitlezen, I²C
Stikstofoxiden sensor: Sensirion SGP41-D-R4	<ul style="list-style-type: none"> Conditioning instructies, I²C Raw signal meting instructies, I²C Stop instructie, I²C 	Meting resultaat uitlezen, I²C
GPS module: Seeed Studio Grove-GPS Air530	Start en configuratie instructies, serieel UART (2.8V)	GPS NMEA0183 V4.1 protocol data, serieel UART (2.8V, default 9600bps)
LoRaWAN module: HopeRF RFM95W v2.0	<ul style="list-style-type: none"> Mode instructie, SPI Sensor data opgedeeld in bytes, byte array van max 51 bytes, SPI 	Sensordata als payload van max 51 bytes, TTN 868MHz +13dBm

LoRaWAN ontvanger	Sensordata als payload van max 51 bytes, TTN 868MHz +13dbM	Sensor data opgedeeld in bytes, byte array van max 51 bytes, netwerk
Backend: Node-RED	Sensor data opgedeeld in bytes, byte array van max 51 bytes, netwerk	Sensor data bytes gedecodeerd in JSON object, netwerk
Database: MongoDB	Sensordata als JSON object, netwerk	Sensordata als JSON object, netwerk
Web API: Python Flask	Sensordata als JSON object, netwerk	Sensordata als JSON object, HTTP respons
GUI: Port of Antwerp digital twin	Sensordata als JSON object, HTTP request	Weergave op scherm



GUI mockup

De sensordata bevat een coördinaat en tijdstip van de meting voor eenvoudige weergaven op een kaart. Zo kan de data eenvoudig worden opgevraagd via een API en geïntegreerd worden in de [digital twin van de Port of Antwerp](#). De data kan in realtime worden weergegeven, er kan genavigeerd worden door de tijd, of een exacte timestamp kan ingegeven worden. Dit zowel op een 2D kaart als in een 3D nabootsing van de haven.



Statediagram



Flowcharts



Datamigratie

De As-Is situatie is niet gericht op de Port of Antwerp. Hier is dus nog geen data over beschikbaar en datamigratie is niet van toepassing.

Infrastructuur

Er is in de Port of Antwerp al een LoRaWAN netwerk aanwezig. Deze kan dus gebruikt worden. De backend kan op een bestaande server opgezet worden. Indien er geen beschikbaar is moet een nieuwe worden opgezet. Er moet rekening gehouden worden met het onderhoud van de server en de sensoren.

Security

De sensor heeft twee grote zwaktes met betrekking tot security:

1. De microcontroller:

De zwakte van de MCU is voornamelijk dat deze geherprogrammeerd kan worden en voor andere doeleinden gebruikt worden. Dit proces moet na opleveren zo moeilijk mogelijk gemaakt worden. Hoe dit best te doen moet onderzocht worden tijdens de ontwikkeling.

2. Het verzenden van de data:

Zoals in het gedeelte over privacy hieronder beschreven gaat het hier niet om gevoelige data. Het kan echter beschouwd worden als best practice om de te verzenden data te encrypteren voor het verzenden. Moest er gewenst worden om de data toch niet publiek beschikbaar te maken is de data op deze manier niet leesbaar wanneer onderschept.

Hardware security

De sensors kunnen best op een zekere hoogte geplaatst worden waar ze niet te gemakkelijk toegankelijk zijn. Dit om diefstal en manipulatie te bemoeilijken en te ontmoedigen.

Privacy en omgang met data

De luchtkwaliteit metingen die de sensor gaat uitvoeren is geen gevoelige data. Het kan beschouwd worden als openbaar goed. Privacy is dus niet van toepassing. Zolang de sensors geplaatst worden op het grondgebied van de Stad Antwerpen mag de data vrij toegankelijk zijn voor de burger. Dit is in lijn met de huidige omgang van gelijkaardige data van de Stad Antwerpen die publiek beschikbaar is. Bij plaatsing van een sensor op privé terrein moet wel expliciete toelating gevraagd worden om de data toegankelijk te maken.

Autorisatierollen

Twee autorisatierollen zullen voor dit project voldoende zijn:

- **Administrator:**

kan wijzigingen maken aan instellingen, code en data. Deze krijgt wachtwoord beveiliging.

- **Gebruiker:**

kan enkel data opvragen.

Documentatie

De Port of Antwerp is een internationale haven wat ook weerspiegeld wordt door de mensen die er werken. Het is dus belangrijk dat de documentatie in het Engels gebeurt om door een zo wijd mogelijk publiek begrijpbaar te zijn.

Documentatie in de code

In de code wordt commentaar toegevoegd om alle functionaliteiten van de code te beschrijven.

Documentatie bij oplevering

Bij oplevering zal volgende documentatie voorzien worden:

- Handleiding over het algemeen gebruik van de sensor, kalibratie, en een overzicht van de technische specificaties van de sensor.
- Technische documentatie met elektrische schema's en PCB ontwerp om eventuele verdere ontwikkeling mogelijk en eenvoudiger te maken.

Termen en Afkortingen

Term	Omschrijving
MCU	Microcontroller, het brein van de sensor.
LoRaWAN	Long Range Wide Area Network is een specificatie voor een telecommunicatienetwerk geschikt voor langeafstandscommunicatie met weinig vermogen.
GUI of Graphical user interface	Het computersysteem waarmee de eind gebruiker de data kan op vragen en weergeven.
Field deployment	Het uitrollen van het systeem bij de klant, waaronder het plaatsen van de sensoren.
In-the-field	Ter plaatsen waar de sensor is geplaatst.
Release Plan	De planning voor het ontwikkelen van het product, na de initiële analyse.
ppm	Parts per million is een maatéénheid om concentratie aan te geven.
PM2.5 en PM10	De standaard maten voor het meten van fijnstof met grote 2.5µm en 10µm.
hPa	Hectopascal is een éénheid van luchtdruk.
NOx	Stikstofoxide
VOC	Vluchtige organische componenten, is een verzameling van allerlei vluchtige stoffen waaronder CO2 en koolmonoxide.
APICA	Antwerp Port Information & Control Assistant, de digitale assistent in de Port of Antwerp digital twin.
Digital twin	Een digitale 3D nabootsing van een echt gegeven, bvb de Port of Antwerp.
Realtime	De laatste nieuwe data wordt weergegeven van zodra ze beschikbaar is.
Timestamp	Een punt in de tijd, aanduiding met tijd en datum voor wanneer de data is opgemeten.
I/O	Inputs/Outputs, mbt inkomende en uitgaande data, of ingangen en uitgangen in een systeem.
MVP	Minimal Viable Product, is een omschrijving van de minimale kenmerken die het product versie 1.0 moet hebben om opgeleverd te worden aan de klant.
backend	Alles van code dat achter de schermen gebeurt en onzichtbaar is voor de gebruiker.
frontend	Alles wat zichtbaar is voor de gebruiker. Gelijk aan de GUI.
API	Application Programming Interface definieert toegang tot functionaliteit die

erachter schuil gaat. In geval van de luchtsensor kan je via de API de sensordata uit de database opvragen.

Bronvermelding