



Gemeentelijk Instituut Technisch Onderwijs Kalmthout
Technisch secundair onderwijs
Industriële Informatie en Communicatie Technologieën

LUCHTKWALITEIT

afbeelding

Geïntegreerde proef van

Ruben Socquet & Arthur d'Hooge

Promotor en mentor:

m. D. Pauwels, m. G. Wagemans en mevr. T. Haentjes

Kalmthout, 2018 - 2019



Gemeentelijk Instituut Technisch Onderwijs Kalmthout
Technisch secundair onderwijs
Industriële Informatie en Communicatie Technologieën

LUCHTKWALITEIT

afbeelding

Geïntegreerde proef van

Ruben Socquet & Arthur d'Hooge

Promotor en mentor:

m. D. Pauwels, m. G. Wagemans en mevr. T. Haentjes

Kalmthout, 2018 - 2019

Voorwoord

Om de luchtkwaliteitsmeter te realiseren hebben we meerdere obstakels gehad. Om die problemen op te lossen konden we steeds terecht bij onze mentor en promotor. Daarom zou ik graag m. D. Pauwels, m. G. Wagemans en mevr. T. Haentjes bedanken.

Inhoudsopgave

Inhoud

Voorwoord	4
Inhoudsopgave	5
Inleiding	7
1.1 Definitiestudie (Project)	8
1.1.1 Probleemstelling.....	8
1.1.2 Specificaties	8
1.1.3 Functionaliteiten	9
1.1.4 Hardware	9
1.1.5 Software	9
1.1.6 Gebruikers.....	9
1.1.7 Veiligheid	10
1.2 Detailontwerp (project).....	10
1.2.1 Principewerking en principeschema ('s).....	10
1.2.2 Hardware	11
1.2.3 Dataverwerking	13
1.2.4 Processturing	13
1.2.5 Communicatie.....	14
1.2.6 Garanties en veiligheid.....	14
1.2.7 Software	16
2. Schema's en tekeningen	18
2.1 Lay-out website	18
2.2 Meettoestel.....	19
2.3 Netwerkdigram	20
3. Elektronica	21
3.1 DHT11	21
3.2 BH1750	23
3.3 NEO-6M.....	25
3.4 SIM900	27
3.5 CCS 811	29
3.6 Arduino Uno.....	31
4. Logboek.....	33
5. Reflectieverslag.....	35
6. Bibliografie	36
7. Bronnen.....	37

Inleiding

Dit GIP heeft als doel om een methode te ontwikkelen waarmee je eenvoudig de kwaliteit van de lucht in uw omgeving te meten. Het doel is om het de gebruiker zo makkelijk en efficiënt mogelijk te maken. We gaan dit doen aan de hand van een cloud service. Gebruikers krijgen een meettoestel dat draagbaar, compact en voorzien is van een batterij zodat ze het mee kunnen nemen. Hiermee kunnen ze de lucht in hun omgeving in kaart brengen. Het meettoestel wordt voorzien van een mobiele verbinding tot het internet zodat het gemeten waarden naar de server kan sturen. De server wordt ook voorzien van een google maps kaart, daarmee kunnen gebruikers zien in welke plaatsen er metingen zijn uitgevoerd en de daarbij horende waarden bekijken. Het doel is om locaties in kaart te brengen waar er veel luchtvervuiling zodat er aangetoond kan worden dat er maatregelen genomen moeten worden. Het toestel meet CO² waardes en hoeveelheden organische verbindingen. Daarnaast wordt er ook temperatuur, luchtvochtigheid en lichtsterkte gemeten.

1. Het eigenlijke werk

1.1 Definitiestudie (Project)

1.1.1 Probleemstelling

We zijn begonnen van het idee om een vliegtuig te maken en vanop het vliegtuig de luchtkwaliteit te meten. Om dit meer te richten op commercieel gebruik gaan we dit zonder een vliegtuig maken. Maar de sensoren in een behuizing integreren met een gps-tracker dat op verschillende voertuigen geplaatst kan worden. Vervolgens wordt de data via het mobiele netwerk verstuurd naar de server, deze server gaat de data verwerken en live op een website plaatsen. Die website zal openbaar toegankelijk zijn, wanneer je inlogt kan je volgen waar de luchtkwaliteit-meter zich bevindt en de gemeten data op die plaats bekijken. We hebben voor dit project gekozen omdat luchtvervuiling de laatste jaren een groter probleem wordt voor onze gezondheid. Met de data dat we verzamelen gaan we kunnen aantonen of de luchtkwaliteit daadwerkelijk achteruitgaat en of er iets aan veranderd moet worden. Doordat dit systeem eenvoudig in gebruik zal zijn kan het door iedereen gebruikt worden. Hierdoor zal er veel data verzameld kunnen worden om aan te tonen hoe de luchtomstandigheden evolueren. Dit systeem heeft als voordeel dat het overal kan gebruikt worden. Wanneer er geen data verstuurd kan worden via het mobiele netwerk zal er nog steeds data gemeten en lokaal bewaard worden. Deze data kan dan worden verzonden wanneer er terug een signaal is.

1.1.2 Specificaties

Het systeem zal uit 2 delen bestaan. Deel 1 is de Arduino Uno, de sensoren en de SIM900 shield deze afmetingen moeten compact blijven en mogen niet veel wegen. Het is een mobiele opstelling dat gemakkelijk ergens mee naartoe genomen kan worden en op voertuigen geplaatst kan worden. Het kan ook in een lokaal geplaatst kan worden om van op één plaats de metingen uit te voeren. Het systeem zal gebruikt worden met Arduino Uno om de data van de sensoren in te lezen, op te slaan en door te sturen. Voor de financiering hebben we afgesproken om dit onder ons 2 te verdelen.

1.1.3 Functionaliteiten

Een groot deel van het systeem zal het doorsturen van de data van de sensoren zijn via het mobiele netwerk door middel van de SIM900 met de Arduino. Wanneer deze data gaat doorgestuurd worden naar de website waar er dan verschillende gegevens te vinden zijn. Als output hebben we dan de data van de CO²-, temperatuur-, luchtvochtigheid-, lichtsterkte sensor en de locatie waar de data verzameld is. Om de hardware en sensors te voeden zal een batterij geïntegreerd worden in de behuizing.

1.1.4 Hardware

De hardware bestaat uit 2 grote categorieën, de server en het meettoestel. Voor de server gebruiken we een gewone pc waarop we IIS draaien (manier om webpagina's te hosten) en een database met alle meetwaarden. Het meettoestel functioneert rond de Arduino waaraan alle sensors aangesloten zijn en de 4g module.

1.1.5 Software

Er zal op de website door middel van registratie de luchtkwaliteitsmeter ingesteld worden en indien er al metingen gedaan zijn zullen er grafieken met de gemeten data te vinden zijn gebonden aan de locatie.

1.1.6 Gebruikers

De verschillende gebruikers kunnen we onderverdelen in de gebruikers die geen luchtkwaliteit meter hebben. Deze gebruikers kunnen niet inloggen op de website maar gaan wel de algemene data grafieken kunnen terugvinden van een gemeente op de website. Vervolgens hebben we de gebruikers dat wel een systeem hebben gekocht, zij kunnen een account aanmaken en dat account linken aan hun systeem. Zij kunnen de specifiek door hun gemeten waarden gelinkt aan de plaats bekijken. En dit vergelijken met andere plaatsen. Ook zal er aangegeven worden of dit gezonde of niet gezonde levensomstandigheden zijn. Ten slotte hebben we nog de administrators, dit zijn Arthur d'Hooge en Ruben Socquet, zij kunnen alle specifieke verzamelde data bekijken van elke gebruiker. De maximale hoeveelheid gebruikers hangt af van de server waarop de website wordt gehost.

1.1.7 Veiligheid

Er zal zeker rekening gehouden moeten worden naar privacy en bescherming van gegevens want wanneer de gebruikers gaan meten wordt live hun locatie doorgestuurd en dat is toch zeer gevoelige data. Daarom zal de data dat verstuurd wordt zeker versleuteld moeten worden. De website moet ook goed beveiligd worden zodat er geen vertrouwde gegevens van de gebruiker verloren gaan.




1.2 Detailontwerp (project)

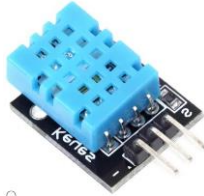
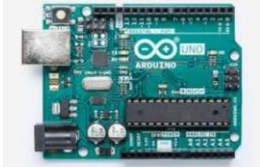
1.2.1 Principewerking en principeschema ('s)

De werking van de luchtmeter begint bij de Arduino dat de metingen van de verschillende sensoren opvraagt. Waarna vervolgens de data van de sensoren tijdelijk in de Arduino zal worden opgeslagen tot er een verbinding met de server wordt gemaakt met de SIM900 module. Die module zal door een vaste code in de Arduino eerst een IP-adres opvragen in het WAN en vervolgens verbinding maken met de APN van de provider van de SIM kaart. Wanneer die verbinding is gemaakt zal er een verbinding gemaakt worden met de server dat een vast IP-adres en vaste poort hebben zodat de Arduino weet met welke server verbinding gemaakt moet worden. Al de commando's dat naar de SIM900 shield gestuurd worden zijn AT commando's. Wanneer het meetapparaat verbonden is met de server kan de data van de sensoren verstuurd worden. In de server wordt de data opgeslagen in een Acces database zodat de data overzichtelijk en makkelijk toegankelijk opgeslagen is en vervolgens verwerkt en bezichtigd kan worden via de website. Op de website zal er een optie zijn om te registreren en in te loggen. Wanneer de gebruiker een meter heeft kan hij deze registreren via de site om de door hem gemeten metingen en zijn afgelegde route te bekijken.

1.2.2 Hardware

De hardware die we gaan gebruiken:

Lichtsensoren	Lichtsterkte meten.	I2C-protocol, 1 - 65535 lux meetbereik, 5V.	
CO2 sensor	CO2 gehalte in de lucht meten.	I2C-protocol, werkt op 3.3V.	
Gsm-module	Communicatie met server.	Communicatie via seriële poort, heeft 12V voeding nodig, werkt op alle gsm-netwerken.	

Vochtigheid en temperatuur sensor	Luchtvochtigheid en temperatuur meten.	Werkt op voeding van Arduino (5V), communiceert met OneWire protocol.	
Arduino Uno	Microcontroller om meettoestel te besturen.		
Batterij houder	Stroomvoorziening Meettoestel.	6 AA Batterijen. 9V.	

Op de Arduino zullen we het programma laten draaien zodat we de sensoren kunnen besturen en de data van de sensoren via de SIM900 naar de server kunnen sturen. Dat programma zal gemaakt worden in C++ een taal die het makkelijk maakt om te interfaceren met de Arduino zijn I/O pinnen. In het programma zal verbinding gemaakt worden met de SIM 900 shield, die shield gaat een SIM kaart hebben zodat er een verbinding met het mobiele netwerk gemaakt kan worden en de data naar de server gestuurd wordt. Die server zal een ASP.NET website draaien en een database beheren waarin alle gemeten data word opgeslagen.

1.2.3 Dataverwerking

De gemeten data zal voor het grootste deel verwerkt worden in de server, dit is logisch want de server heeft de grootste rekenkracht. De Arduino moet enkel de data uitlezen en versturen. Met de data gaat de server dan een kaart plotten met locaties van het meettoestel, de CO2 gehalten, luchtvochtigheid, lichtsterkte, temperatuur en het tijdstip van de meting bij elke locatie zetten. Alle data wordt ook in een database bewaard zodat als er later het zelfde traject word afgelegd er vergeleken kan worden. Er moet wel rekening gehouden worden met de privacy van de gegevens en in bijzondere maten de locatie.

1.2.4 Processturing

De besturing is heel eenvoudig, voor de meting kunnen parameters aangepast worden via de website die op de server draait. De parameters zijn dan welke soorten metingen (CO2, temperatuur, lichtsterkte...), de meetsnelheid en hoelang er gemeten moet worden. Deze worden dan doorgegeven naar het meettoestel. Op de website zal ook de meting gestart en gestopt kunnen worden. Voor de rest zal het systeem alles automatisch afhandelen. Op de website wordt er dan live-info getoond over het meettoestel, zoals de locatie en de huidige toestand van de sensoren.

1.2.5 Communicatie

Omdat wij veel verschillende toestellen (Arduino, Server, Database) gebruiken is communicatie een groot deel van ons GIP. Om te beginnen zullen de digitale sensoren die allemaal over I2C beschikken op een I2C bus worden aangesloten waarvan de master de Arduino is. De vochtigheid en temperatuur sensor werkt met het OneWire protocol en zal ook op de Arduino worden aangesloten. Door de SIM900 te verbinden met de microcontroller gaan we gebruik kunnen maken van het mobiele datanetwerk. Dit betekent dat de communicatie over het WAN gaat en we dus onze data moeten encrypteren. Om toegang te krijgen tot het mobiele netwerk moeten we een simkaart in de module steken. Hiervoor gaan we een prepaid simkaart gebruiken. Als het meettoestel data wilt verzenden zal een verbinding worden gemaakt met de server volgens het TCP-protocol. Dit protocol heeft doorgedreven foutencontrole ingebouwd. Zo zijn we zeker dat er onderweg naar de server geen data verloren gaat of corrupt wordt.

1.2.6 Garanties en veiligheid

1.2.6.1 Mechanische veiligheid

Alle hardware onderdelen worden in een afgesloten bakje gemonteerd dat het grootste vuil buitenhoudt. Dit bakje zal worden geprint met een 3D printer. Het zal gemaakt worden uit ABS. Dit plastic kan temperaturen aan tot 85°C. Het is licht en hard, wat het ideaal maakt voor dit project.

1.2.6.2 Elektrische veiligheid

De Arduino en GSM-module hebben een ingebouwde voltage regulator. Dit betekent dat de 9V van de batterij naar 5V wordt gebracht. 9V en 5V zijn 'veilige' spanningen. Er is geen gevaar voor elektrocutie bij het aanraken van deze spanningen. Bij een kortsluiting kunnen er andere gevaren optreden zoals brand, batterijen die oververhitten, giftige gassen die vrijkomen, etc. Gelukkig zijn zowel de Arduino als de GSM-module beveiligd met een interne zekering.



1.2.6.3 Vitale 8

De vitale 8 zijn regels die nageleefd moeten worden wanneer er wordt gewerkt aan een aanpassing van een elektrische installatie. Deze regels zorgen voor de veiligheid van de mensen die werken aan de installatie en in de omgeving.

1. Werkzaamheden voorbereiden
 - Nagaan welke maatregelen er genomen moeten worden om veilig te werk te gaan.
2. Vrij schakelen
 - De installatie waaraan je wilt werken loskoppelen van alle voedingsbronnen.
3. Voorkom herinschakeling
 - Indien er hulpbronnen aanwezig zijn moeten deze ook buiten werking gesteld worden.
4. Controleer de spanningsafwezigheid
 - Ga met een meettoestel na of er nog spanning aanwezig is op de geleider binnen het gebied waar er wordt gewerkt of aanpassingen moeten gebeuren.
5. Aarden, ontladen en kortsluiten
 - Binnen de werkzone moeten alle hoogspanningsinstallaties geaard zijn en laagspanningsinstallaties die ongewild spanning kunnen voeren moeten ook geaard zijn.
6. Afbakenen en/of afschermen van de werkzone
 - Op werkplaatsen waar in de werkzone andere elektrische installaties zijn moet er een afbakening of een afscherming gebeuren volgens het AREI.
7. De elektrische installatie vrijgeven
 - Een werkverantwoordelijke is de enige persoon die de toestemming aan het uitvoerend personeel kan geven om de werken te starten.
8. Terug onder spanning brengen
 - Wanneer de werkverantwoordelijke zeker is dat de elektrische installatie terug op veilige wijze onder spanning gebracht kan worden zal hij dit aan de installatieverantwoordelijke melden. Wanneer dat is gebeurd zal onder verantwoordelijkheid van de installatieverantwoordelijke de spanning terug ingeschakeld worden.

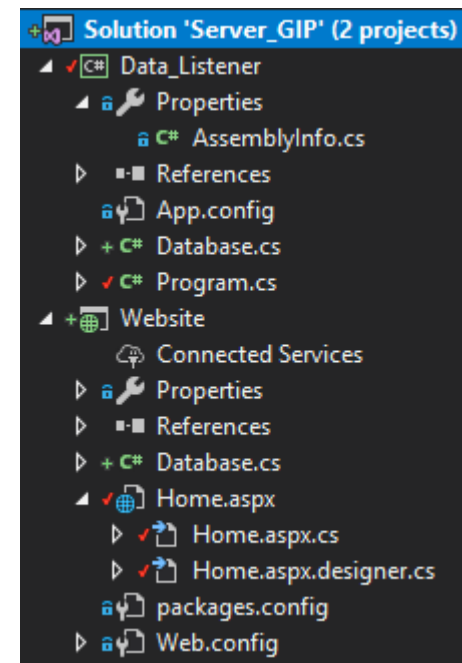
1.2.6.4 Privacy

Locaties van gebruikers is een zeer gevoelig punt, we zullen in ons GIP moeten kijken of we aan de eisen van de wetgeving kunnen voldoen, hoelang we die data mogen houden, en andere vraagstukken hierrond moeten beantwoorden.

1.2.7 Software

1.2.7.1 Server structuur

De software voor de server word geschreven in Visual Studio 2017. We gebruiken C# als programmeertaal. In de Solution hebben we twee projecten: 'Data_Listener' en 'Website'. In het project 'Data_Listener' schrijven we de code om gemeten waarden van sensoren die van het meettoestel komen weg te schrijven naar de database (zie 1.2.7.3 Database). Het project 'Website' is een ASP.NET website. Door 2 projecten in 1 Solution te steken kunnen deze tegelijk getest en gedebugd worden. In beide projecten zit een C# klasse. Deze dient om de communicatie met de database te regelen.



1.2.7.2 Data_Listener

Dit project is een gewone console applicatie. Het luistert op een TCP-poort naar inkomende verbindingen. Als het meettoestel verbinding maakt zal het de data ontvangen en testen of deze geldig is. De inkomende data moet dus voldoen aan bepaalde voorwaarden. Er moet bijvoorbeeld getest worden of sommige sensor waarden geen NULL bevatten. De data wordt dan in de database gezet. Dit gebeurt via de Database klasse. Deze console applicatie dient dus enkel en alleen om data te ontvangen van het meettoestel.

1.2.7.3 Website

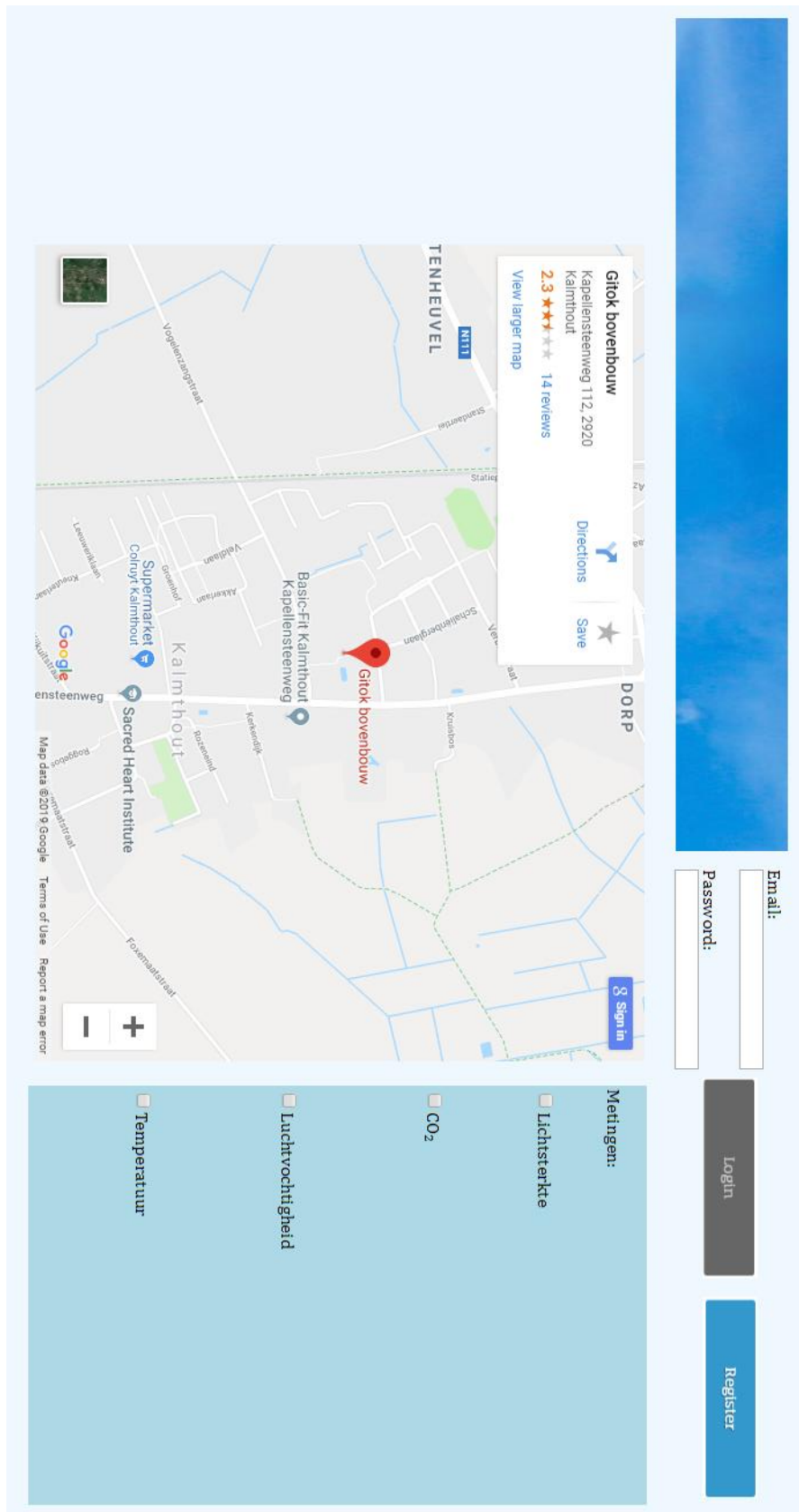
De website is een ASP.NET project. Met ASP kunnen we een website bouwen met HTML en CSS, achter de gebruiksvriendelijke UI draait dan een Code Behind file geschreven in C#. Hiermee kunnen we de website interactief maken. De code behind draait volledig op de server. Om toch variabelen mee te sturen maken we gebruik van de ViewState methode. Maar omdat we gebruikersnamen en wachtwoorden versturen moeten we deze encrypteren. Gelukkig kunnen pagina's in ASP voorzien worden van encryptie door op de eerste regel de in het rood onderlijnde parameters toe te voegen:

```
<%@ Page Language="C#" EnableViewState="true" ViewStateEncryptionMode="Always" EnableViewStateMac="true"
```

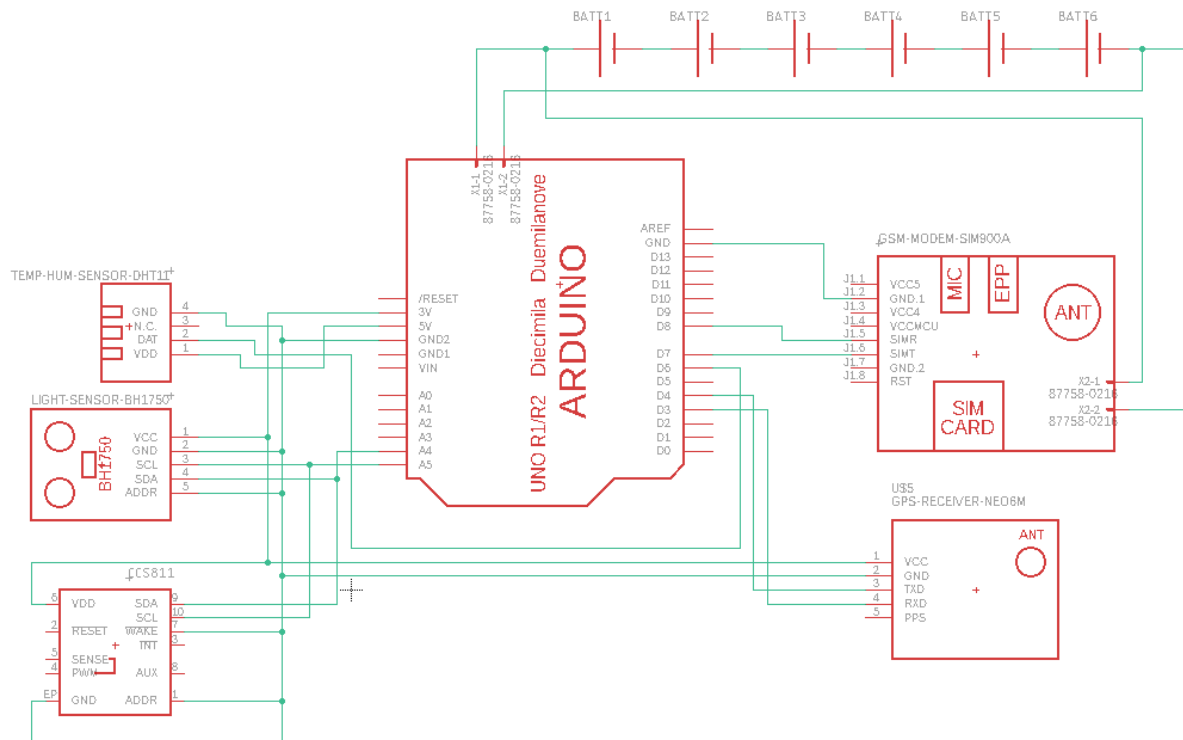
Het grafische ontwerp van de site staat bij 2.1 Lay-out Website.

2. Schema's en tekeningen

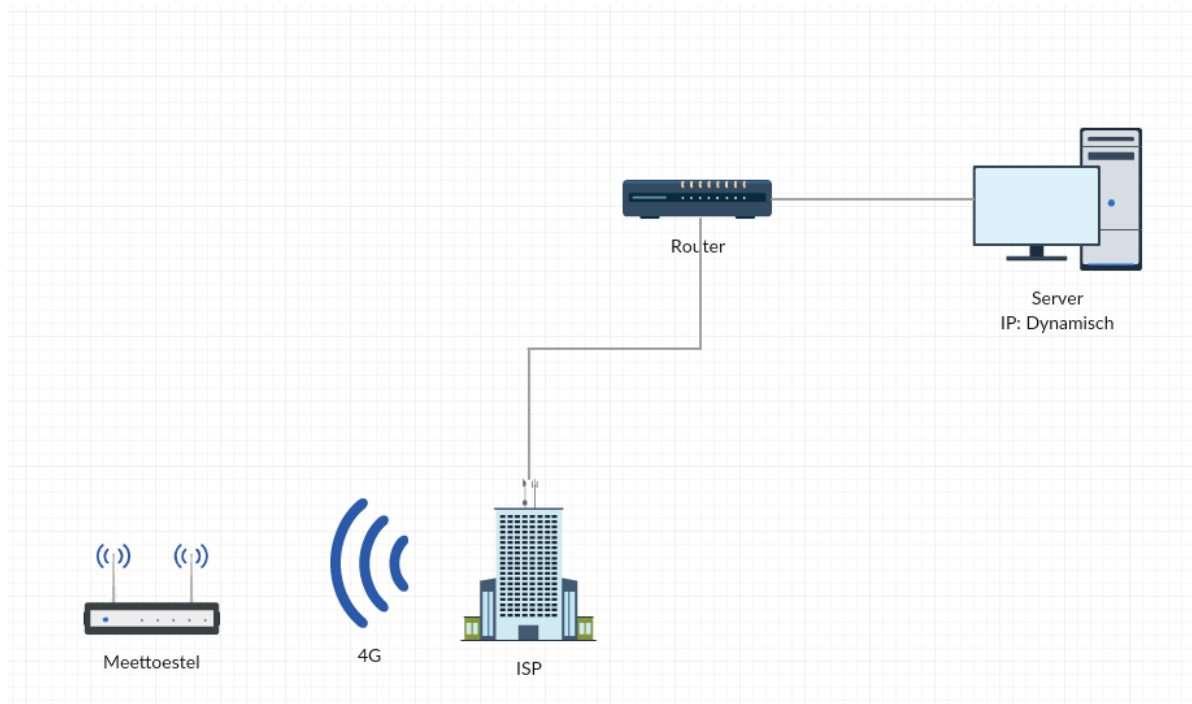
2.1 Lay-out website



2.2 Meettoestel



2.3 Netwerkdigram



Op de voorstelling hierboven kan u zien dat er via een mobiele dataverbinding een connectie met de server gemaakt zal worden.

3. Elektronica

In dit hoofdstuk van het GIP gaan we kijken naar de hardwarecomponenten dat we gekozen hebben en bespreken hoe ze werken. We hebben in totaal 4 sensoren gebruikt waarmee we data gaan verzamelen. Deze waarden zijn de temperatuur, vochtigheid, lichtsterkte, locatie en CO² waarden.

3.1 DHT11

Om de temperatuur en de vochtigheid te meten gebruiken we namelijk één sensor, de DHT11. Deze sensor heeft drie aansluitingen, VCC en GND voor 5V voeding van de Arduino en DAT om de data over te dragen. Om de dataoverdracht via één draad te laten verlopen wordt een seriële enkele draad protocol gebruikt. De Arduino zal hierdoor zijn aanvraag sturen en de data terugkrijgen over dezelfde draad. Dit gaat als volgt tewerk: de Arduino zal eerst een aanvraag doen om data te krijgen, dit gebeurt door de DAT pin naar een logisch niveau "0" getrokken voor iets langer dan 18ms dan wordt deze pin terug naar een logisch niveau "1" opgetrokken. Vervolgens zal de DHT11 een reactie geven van ~54µs een logische "0" en daarna ~80µs een logische "1". Tenslotte worden de gegevens overgedragen in vijf segmenten van acht bits. Deze methode hebben we zelf kunnen schrijven door de bibliotheek "dht.h", anders had dit proces op bit niveau geprogrammeerd moeten worden. De DHT11 is een digitale sensor daarom zal het ook aan een digitale pin aangesloten worden, namelijk op pin zes van de Arduino. De temperatuur wordt gemeten door een NTC temperatuur sensor en de vochtigheid wordt gemeten door een vochtigheid afhankelijke weerstand.

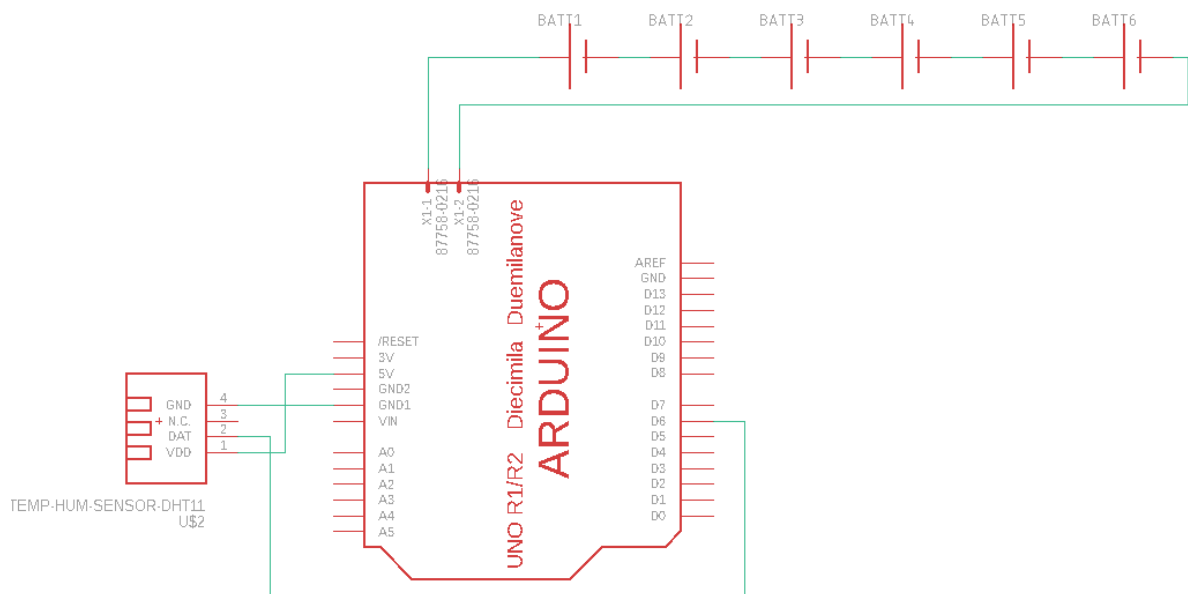
Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT11	20-90%RH 0-50 °C	± 5 %RH	± 2°C	1	4 Pin Single Row

Met deze sensor kunnen we temperaturen meten van 0°C tot 50°C met een nauwkeurigheid van +/- 2°C en een vochtigheid van 20% tot 90% met nauwkeurigheid van +/- 5%.

VDD=5V, T = 25°C (unless otherwise stated)

	Conditions	Minimum	Typical	Maximum
Power Supply	DC	3V	5V	5.5V
Current Supply	Measuring	0.5mA		2.5mA
	Average	0.2mA		1mA
	Standby	100uA		150uA
Sampling period	Second	1		

In de datasheet vinden we ook welke voedingsspanning er gebruikt moet worden. Hiervoor kunnen we best de kenmerkende spanning gebruiken dus in dit geval is dat 5V. Deze waarde wordt geleverd door de Arduino en om er zeker van te zijn kunnen we berekenen hoeveel vermogen de sensor maximaal vraagt. We weten dat het maximaal vermogen niet hoger mag zijn dan $\frac{1}{4} W$ dus: $P = U \times I = 5V \times 2,5mA = 12,5mW$, deze waarde is dus onder het maximaal vermogen van de Arduino en kan dus gevoed worden. Ook is het belangrijk om rekening te houden met de sample periode, dit wil zeggen hoe snel er een meting achter elkaar uitgevoerd kan worden. We mogen dus niet de meting herhalen binnen één seconde.



Hierboven ziet u de schematische opstelling van hoe de DHT11 sensor op de Arduino Uno wordt aangesloten. De Arduino Uno wordt gevoed door de zes in serie geschakelde AA-batterijen.

3.2 BH1750

Voor lichtsterkte te meten maken we gebruik van een BH1750 sensor dat lichtsterktes kan meten van 1lux tot 65535 lux. Door vijf aansluitingen, GND en VCC worden gebruikt voor de voeding van de sensor, namelijk 3,3V van de Arduino. De SCL pin van de sensor wordt verbonden met poort A5 van de Arduino, deze aansluiting gaat ervoor zorgen dat er een kloksignaal wordt gestuurd en de data volgens het kloksignaal verstuurd kan worden. Vervolgens hebben we de SDA, deze pin sluiten we aan op poort A4 van de Arduino en wordt gebruikt om data van de sensor naar de Arduino te sturen. Ten slotte is er nog de adres aansluiting, deze wordt verbonden met de GND van de Arduino zodat de sensor weet dat het een slave is. Om de data naar de Arduino te sturen wordt een I²C-bus gebruikt. Een I²C-bus bevindt zich enkel tussen de pinnen A4 en A5, maar er kunnen nog sensoren aan deze pinnen gekoppeld worden want dit protocol ondersteunt meerdere slaves over dezelfde datalijnen. Om deze overdrachtsmethode te gebruiken moet er aan het programma een bibliotheek genaamd "BH1750FVI.h" toegevoegd worden. Deze bibliotheek zal er voor zorgen dat de overdracht geprogrammeerd kan worden met woorden dat we kunnen gebruiken en niet op bit niveau.

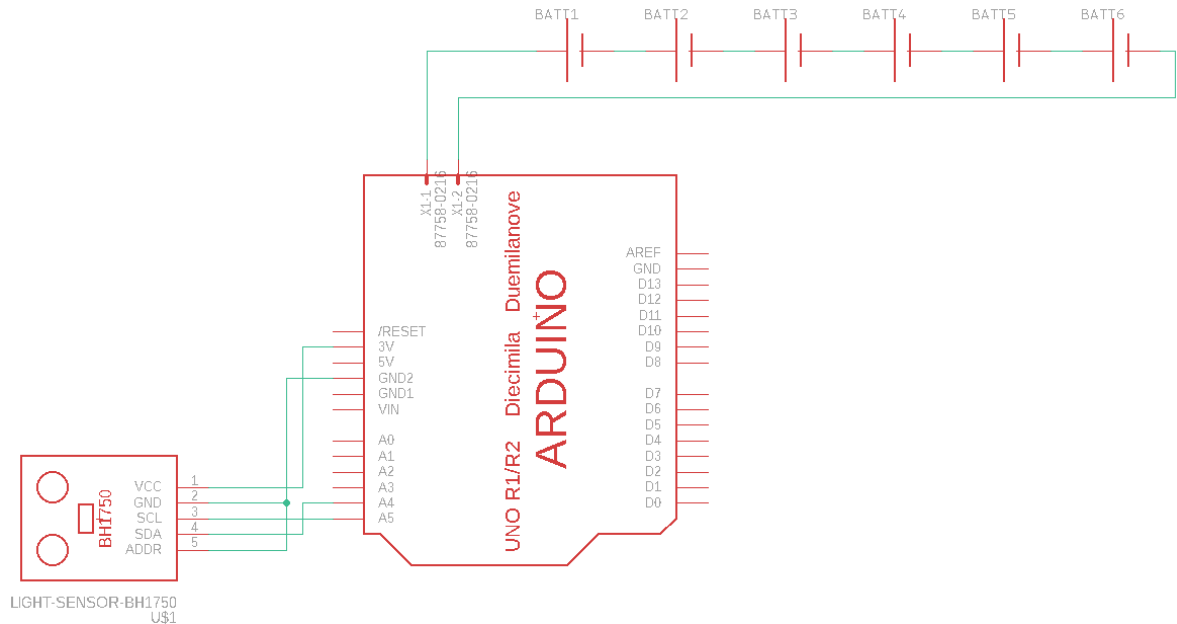
● Operating Conditions

Parameter	Symbol	Ratings			Units
		Min.	Typ.	Max.	
Vcc Voltage	Vcc	2.4	3.0	3.6	V

In de datasheet vinden we terug met welke spanning de sensor gevoed moet worden. Hiervoor kunnen we best de kenmerkende spanningswaarde gebruiken, dit is 3V. We zouden graag als voeding de Arduino gebruiken maar die heeft geen 3V uitgang. Wat we wel op de Arduino vinden is een 3,3V uitgang en dan kunnen we zien dat deze waarde binnen het werkspanningsgebied ligt.

Parameter	Symbol	Limits			Units
		Min.	Typ.	Max.	
Supply Current	Icc1	—	120	190	µA

Om er zeker van te zijn dat de sensor door Arduino gevoed kan worden gaan we moeten nagaan of het maximaal vermogen niet overschreven wordt. Deze waarde is $\frac{1}{4} W$ dus:
 $P = U \times I = 3,3V \times 190\mu A = 627\mu W$, deze waarde is onder het maximaal vermogen van de Arduino dus er moeten geen extra maatregelen genomen worden.



Op het bovenstaande schema zien we de testopstelling van de lichtsensor. Hierop is te zien dat de sensor verbonden en gevoed is door de Arduino Uno. De Arduino wordt daarnaast gevoed door zes in serie geplaatste AA-batterijen.

3.3 NEO-6M

Dit component in onze schakeling gaat er voor zorgen om gps coördinaten van zijn locatie te achterhalen en te sturen naar de Arduino. De gps heeft vier aansluitingen en een antenne. Het wordt gevoed door 3,3V van Arduino aan de VCC en GND pinnen. Dan hebben we nog de TX en RX pinnen. De TX pin zal de data versturen naar de Arduino en aan de RX pin zal de data ontvangen worden. Beide aansluitingen gaan serieel werken dus de data zal serieel ontvangen en verstuurd worden dit is volgens het Universal Receiver/Transmitter protocol. Dit soort communicatie wordt ook gebruikt bij USB-verbindingen. Om dit soort verbinding te gebruiken wordt de "SoftwareSerial.h" bibliotheek gebruikt zodat de gebruiker niet op bit niveau moet werken. Verder kan hij door deze bibliotheek ook een transmissie snelheid instellen, kiezen welke pinnen er gebruikt worden als TX en RX poort en nog meerdere commando's. Verder wordt er ook nog de "TinyGPS++.h" gebruikt, deze bibliotheek zal ervoor zorgen dat de inkomende data omgezet wordt naar de breedte- en lengtegraad dat we via google map somkunnen zetten in een locatie op de map.

Parameter	Symbol	Module	Min	Typ	Max	Units	Condition
Power supply voltage	VCC	NEO-6G	1.75	1.8	1.95	V	
		NEO-6Q/M	2.7	3.0	3.6	V	
		NEO-6P/V/T					

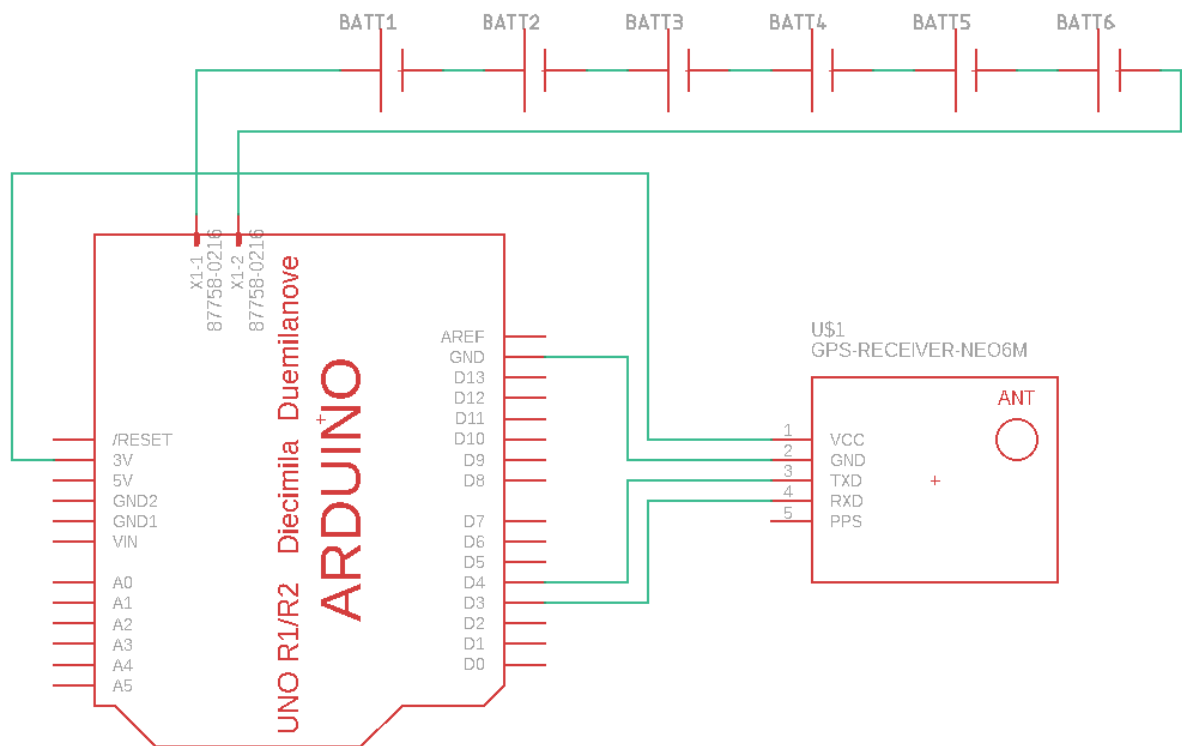
Om te weten hoeveel spanning er nodig is om de sensor te voeden kunnen we dit terugvinden in de datasheet. Hierin hebben we gevonden dat de beste spanning waarmee we de sensor moeten voeden is 3V maar zoals bij de BH1750 valt de 3,3V van de Arduino ook tussen de minimale en maximale spanningswaarden. Dus kunnen we de 3,3V van de Arduino gebruiken als voeding.

Parameter	Symbol	Module	Min	Typ	Max	Units	Condition
Max. supply current ¹⁵	Iccp	All			67	mA	VCC = 3.6 V ¹⁶ / 1.95 V ¹⁷

Wanneer we er zeker van willen zijn dat de Arduino dit zeker kan voeden gaan we moeten berekenen wat het maximale vermogen zal zijn van de gps. In de datasheet hebben we gevonden dat de maximale voedingsstroom 67mA bedraagt. Die maximale waarde is $\frac{1}{4} W$ dus: $P = U \times I = 3,3V \times 67mA = 221,1mW$, hierdoor kunnen we concluderen dat de gps door de Arduino gevoed kan worden.

Interface	Settings
Serial Port 1 Output	9600 Baud, 8 bits, no parity bit, 1 stop bit Configured to transmit both NMEA and UBX protocols, but only following NMEA and no UBX messages have been activated at start-up: GGA, GLL, GSA, GSV, RMC, VTG, TXT (In addition to the 6 standard NMEA messages the NEO-6T includes ZDA).

Bij het schrijven van de code is het ook handig om te weten wat er reeds ingesteld is in de GPS module. Op de screenshot hierboven is te zien dat er al een Baud rate is ingesteld. Dit bepaald hoe snel de dataoverdracht naar de Arduino gaat, dit hoeft niet meer in de code ingesteld worden.



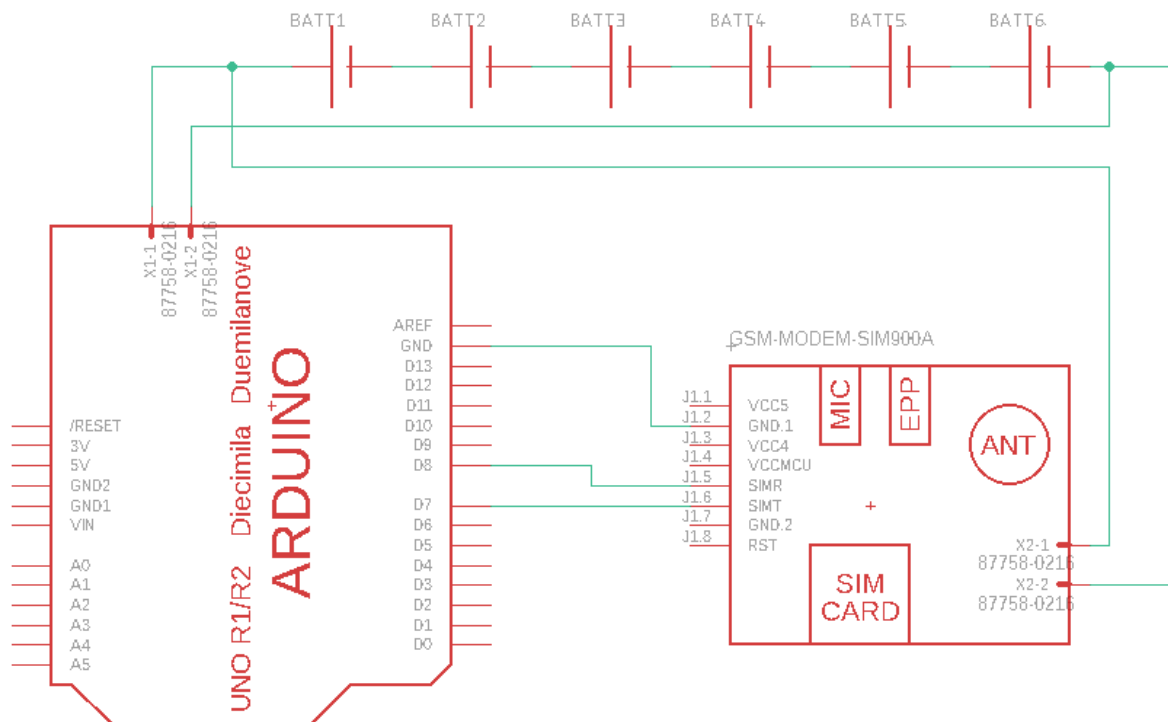
Op bovenstaand schema ziet u de GPS module aangesloten op de Arduino Uno. De Arduino is gevoed door zes AA-batterijen.

3.4 SIM900

Een ander deel van ons netwerk is de communicatie tussen de Arduino en de server. Hiervoor wordt een connectie via het mobiele netwerk gemaakt om het apparaat zo mobiel en gebruiksvriendelijk mogelijk te maken. Om dit te realiseren gebruiken we een SIM900 met een SIM kaart. Om de shield te voeden gebruiken we zes in serie geplaatste AA batterijen wat resulteert in een spanning van 9V wat genoeg is om de Arduino en de SIM900 shield te voeden. Om de data van de arduino naar de SIM900 te sturen gebruiken we drie pinnen, namelijk de GND pin aan de GND van de Arduino en een RX en TX pin waarover de daadwerkelijke data serieel in beide richtingen gestuurd kan worden. Dit soort overdracht wordt Universal Receiver/Transmitter of UART genoemd. Dit soort communicatie wordt ook gebruikt bij USB-verbindingen. Om dit soort verbinding te gebruiken wordt de "SoftwareSerial.h" bibliotheek gebruikt zodat de gebruiker niet op bit niveau moet werken. Verder kan hij door deze bibliotheek ook een transmissie snelheid instellen, kiezen welke pinnen er gebruikt worden als tx en rx poort en nog meerdere commando's. Om de dataoverdracht te realiseren kunnen we via AT commando's de shield aansturen en zo een verbinding met de server opstellen en berichten versturen. Op het SIM900 board kunnen we een antenne terugvinden. Deze antenne gaat ervoor zorgen dat er een verbinding met de GSM masten van de provider gemaakt kan worden. Wanneer we een verbinding met zo een GSM mast willen maken gaan we in ons programma de acces point name van de provider ingeven zodat er via de provider een toegang tot het WAN in en er een publiek IP-adres aan de shield wordt geven. Wanneer er toegang is tot het WAN kunnen we doormiddel van een vast IP-adres van de server dat in de code van de Arduino staat een verbinding met de server maken en de data van de sensoren doorsturen.

Parameter	Condition	Min.	Typ.Max.	Max.	Uints
Vin		5		12	V
Current	Sleep Mode			1.5	mA
	Voice Call			250	
	GPRS Data Mode	76		440	
	During Tx Burst			2	A
Dimension		55*81			mm

Om de SIM 900 shield te voeden gaan we gebruik maken van zes AA-batterijen van elks 1,5V die in serie geplaatst zijn. Volgens de formule $U_{tot} = U_1 + U_2 + U_3 + \dots$ wat een totale spanning maakt van 9V. Die spanning behoort binnen het bereik van de werkspanning van de shield. We gaan de SIM900 enkel gebruiken voor een GPRS verbinding met de server, in de tabel kan je zien dat de maximale stroom bij "GPRS Data Mode" 440mA zal zijn. Hiervoor moeten de batterijen een vermogen van : $P = U \times I = 9V \times 440mA = 3,96W$ kunnen leveren.



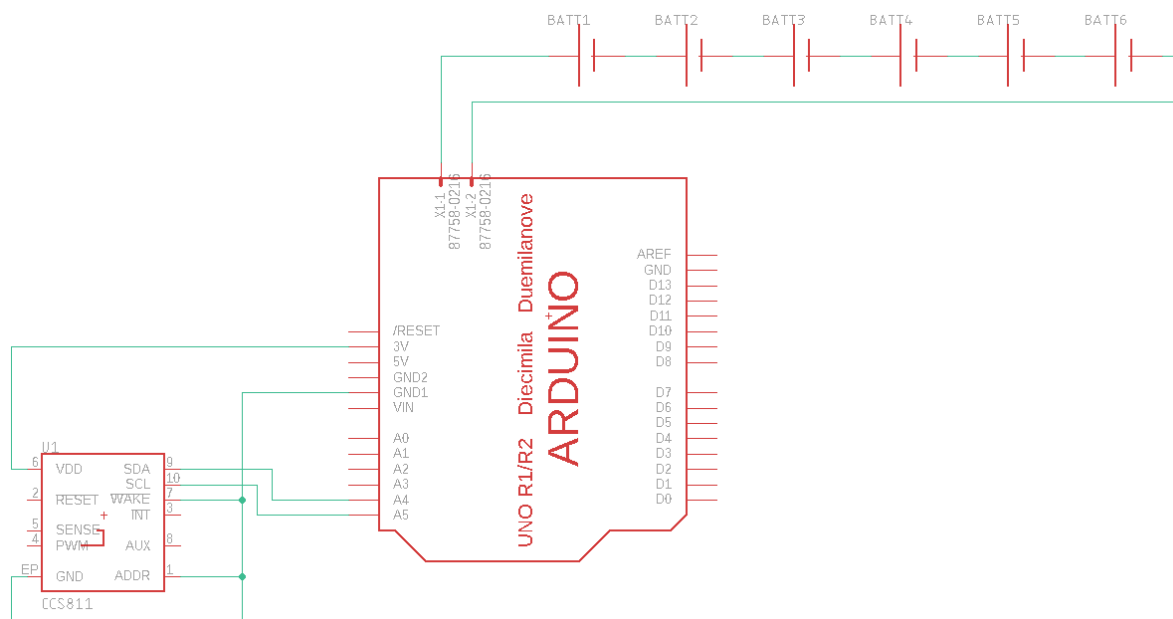
Op bovenstaand schema ziet u de SIM900 shield verbonden met de Arduino Uno. Beide zijn de gevoed door zes in serie geplaatste AA batterijen.

3.5 CCS 811

De CCS 811 sensor gaat de hoeveelheid CO₂ hoeveelheid per m² meten. De sensor heeft zeven aansluitingen waarvan de GND en VCC gebruikt worden door een 3,3V voeding van de Arduino uno. Daarnaast is voor de dataoverdracht pin "SCL" en "SCA" voorzien. De "SCL" pin van de sensor wordt verbonden met poort "A5" van de Arduino, deze aansluiting gaat ervoor zorgen dat er een kloksignaal wordt gestuurd en de data volgens het kloksignaal verstuurd kan worden. Vervolgens hebben we de "SDA" pin, deze pin sluiten we aan op poort "A4" van de Arduino en wordt gebruikt om data van de sensor naar de Arduino te sturen. Er wordt ook nog een "WAKE" aansluiting, die wordt aangesloten aan de "GND" zodat de sensor begint te werken. De aansluitingen "RST" en "INT" worden niet gebruikt. Wanneer die pinnen zijn aangesloten wordt er met de Arduino een verbinding gemaakt door gebruik van het I²C-protocol. Een I²C-bus bevindt zich enkel tussen de pinnen: A4 en A5, maar er kunnen nog sensoren aan deze pinnen gekoppeld worden want dit protocol ondersteunt meerdere slaves over dezelfde datalijnen. Om deze overdrachtsmethode te gebruiken moet er aan het programma een bibliotheek genaamd "SparkFunCCS811.h" toegevoegd worden. Deze bibliotheek zal er voor zorgen dat de overdracht geprogrammeerd kan worden met woorden dat we kunnen gebruiken en niet op bit niveau. Ook zal er door deze bibliotheek data naar een voor ons begrijpbare taal omgezet worden.

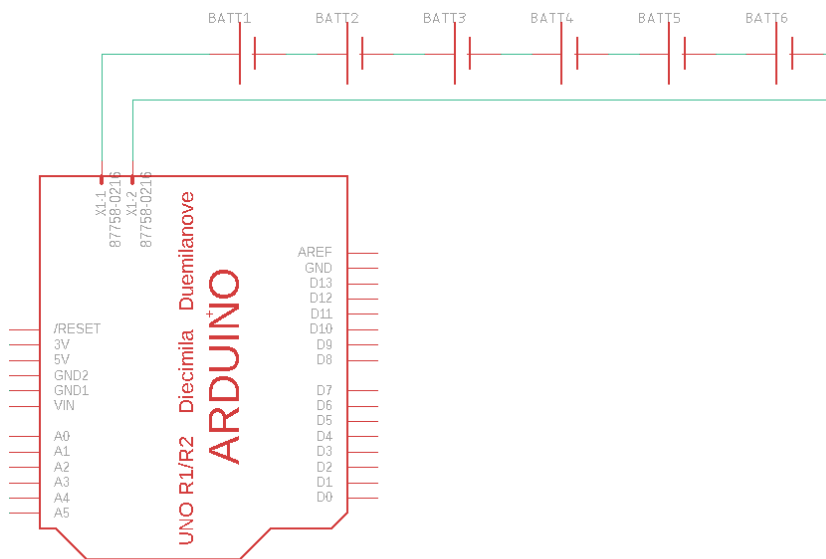
Symbol	Parameter	Min	Max	Units	Comments
Electrical Parameters					
V _{DD} ⁽¹⁾	Supply Voltage	1.8 ⁽²⁾	3.6	V	
I _{DD}	Supply Current		30	mA	In mode 1

Om de spanningswaarde van de voeding te bepalen gaan we de datasheet raadplegen. Hierin gaan we zoeken naar de waarden op de screenshot hierboven. Daarop kunnen we zien dat de waarde van voedingsspanning zich moet bevinden tussen 1,8V en 3,6V. Dit wil zeggen dat de Arduino 3,3V geschikt is voor de voeding voor de sensor. Om te controleren of de Arduino het maximaal gevraagde vermogen kan leveren gaan we dit bepalen met de formule: $P = U \times I = 3,3V \times 30mA = 99mW$, het maximaal vermogen dat de Arduino Uno kan leveren is ¼ W dus de manier van voeden is mogelijk.



Op bovenstaand schema kan u de CCS 811 sensor zien dat verbonden is met de Arduino Uno. De Arduino Uno wordt gevoed door de zes in serie geplaatste batterijen.

3.6 Arduino Uno



Eén van de belangrijkste componenten van het meetapparaat is de Arduino uno. Dit is een microcontroller waar een code ingeladen kan worden via een USB-verbinding met een computer. De code is geschreven in de programmeeromgeving Arduino IDE in de taal C++. De Arduino gaat zijn code lijn per lijn doorlopen, bovenaan staan de gebruikte bibliotheken die nodig zijn om bepaalde sensoren in te lezen en hun data om te zetten naar voor ons begrijpbare metingen.

Input Voltage (recommended)	7-12V
--------------------------------	-------

Op de site van Arduino kunnen we bij het product “Arduino Uno” de specificaties terugvinden. Op de screenshot hierboven zien we dat de externe voeding tussen 7V en 12V moet kunnen leveren. De zes AA-batterijen dat we willen gebruiken leveren elks 1,5V wat in totaal een spanning van 9V levert. Die opstelling kan dus gebruikt worden.

4. Logboek

DATUM:	WAT:	NAAM:	TIJD:	PLAATS:
9/10/2018	GIT repository	Samen	1 uur	School
9/10/2018	Vocht sensor	Samen	1 uur	School
19/10/2018	Plan & sensoren	Samen	2 uur	School
23/10/2018	Sensoren	Samen	2 uur	School
24/10/2018	Documentatie, to do list	Samen	2 uur	School
26/10/2018	Licht sensor en CO ² sensor	Samen	2 uur	School
3/11/2018	Definitiestudie	Ruben	2 uur	Thuis
4/11/2018	Definitiestudie	Ruben	1 uur	Thuis
5/11/2018	Detailontwerp	Arthur	3 uur	Thuis
7/11/2018	To do list, bestellijst, documentatie aanpassen	Samen	2 uur	School
13/11/2018	DHT11 sensor en CO ² sensor	Samen	3 uur	School
19/11/2018	Bestellijst	Arthur	30 min	Thuis
17/12/2018	Bundel	Arthur	1 uur	School
26/12/2018	Visual Studio projecten opgestart	Arthur	3 uur	Thuis
27/12/2018	Bundel bijwerken	Arthur	2 uur	Thuis
2/01/2019	Communicatie Arduino en server	Ruben	3 uur	Thuis
3/01/2019	Schema's, libraries, bundel	Ruben	2 uur	Thuis
6/01/2019	Bundel bijwerken	Ruben	2 uur 30 min	Thuis
7/01/2019	Bundel bijwerken	Arthur	1 uur	Thuis
7/01/2019	Communicatie met server	Ruben	3 uur	Thuis
8/01/2019	Bundel, website, sensoren, schema's	Samen	6 uur	School
8/01/2019	Bundel, Sensoren	Ruben	3 uur	Thuis

5. Reflectieverslag

6. Bibliografie

7. Bronnen

Bron: www.slideshare.net

Op de website slideshare heb ik een volledige lijst gevonden van de AT commando's met uitleg.

Bron: www.deviceplus.com

Om de protocollen beter te begrijpen heb ik een uitleg gelezen van veelgebruikte protocollen bij Arduino.

Bron: www.c-sharpcorner.com

Informatie over ViewState en encryptie.

Bron: www.sparkfun.com

Bibliotheek voor de CCS811 sensor.

Bron: www.volta-org.be

Op deze site heb ik de betekenis van de vitale 8 gevonden en heb ik aan de hand van deze bron kunnen verwoorden.