Improving the dependability of the temperature build-up sensor system in the city of Enschede

A Creative Technology Graduation Project

By: David Vrijenhoek S1722107
Supervisors: Hans Scholten & Richard Bults

Version: Final version

Date: 5-7-2019

Abstract

This graduation is concerned with the development of a sensor system which is used for long-term registration of the temperature development of different locations in Enschede in order to see if and where urban heat islands come into existence. Urban heat islands are urban areas which are significantly hotter than similar rural areas. Since urban areas are often heavily populated, the high temperatures are a cause of heat stress. The municipality of Enschede is therefore interested in the size and severity of these urban heat islands in Enschede. Previous graduation projects already proved that a sensor system can be developed to register the temperature development in the city of Enschede. However, the sensor system needs to be dependable to some degree for it to operate on a long-term base. The goal of this graduation project is to improve the design of the sensor system, so it is able to operate dependably on a long-term base.

This goal is achieved by applying the Creative Technology design method. First off, dependability and systems that are similar to this sensor system were researched. The various stakeholders were interviewed in order to derive their expectations of this sensor system. Furthermore, the context of the deployment of the sensor system was researched. Through this analysis and the conducted interviews, a set of preliminary requirements were crafted. These requirements were then used to specify the design of the sensor system. When the sensor system was realised, it was evaluated in different ways.

The end result of this graduation project is an autonomous sensor system which is able to measure the temperature of the air, the relative humidity and the speed of the wind. The sensor system does so in a dependable way and is able to label the data with a quality mark. The sensor system provides its own power and utilizes a LoRaWAN network to store the gathered data in a database.

Acknowledgement

First of all, I would like to thank my family and friends for supporting me throughout this graduation project. Without all the inspiration and motivation you all gave me, this graduation project would have been a lot harder.

Next I would like to express my gratitude to both my supervisors Hans Scholten and Richard Bults. In the many conversations we had, they never failed to provide me with plenty of feedback and new inputs. This graduation project would not have been at the level it is now if it were not for them pushing me in the right direction.

I would also like to thank my stakeholders. Rik Meijer and Hendrik-Jan Teekens, the contacts at the municipality of Enschede, provided me with a lot of valuable information and supported this project throughout the process. I would also like to thank Wim Timmermans from the ITC faculty of the University of Twente. My conversations with Wim were always very insightful and he provided me with a lot of valuable information.

Lastly, I would like to thank Yoann Latzer for helping me with the back-end service of the system.

Table of content

Table of figures

Chapter 1 Introduction

1.1 Context and problem statement

Global warming is impacting our lives in various ways throughout society. Some more noticeable than others. One of the big issues cities are facing nowadays is the development of zones that are significantly hotter than their neighbouring (rural) areas. These zones are called (urban) heat islands [2] and can be found in urban settings due to fact that urbanised areas are prone to heat up more quickly than areas that are less urbanised.
This effect of urban heat islands poses severe problems related to health issues and infrastructure within a city. These problems stretch from sleeping less to broken roads due to the heat islands [3]. Because of these problems the municipality of Enschede and the University of Twente are looking into the UHIs in Enschede by means of an outdoor temperature measurement system. This system is used for a long-term registration of temperature development of different locations in Enschede in order to see where and if UHIs come into existence. The municipality of Enschede is interested in the factors that influence UHIs and wants to tackle UHI related problems.

A proof of concept of such a system has already been made and now the municipality of Enschede and the University of Twente want to scale up the UHI project. In order for the network to be scaled up it requires a certain level of dependability and more functionalities. These are two important aspects, since higher dependability means more and accurate data. This data is vital to understand the development of UHIs in Enschede.

Dependability is defined as a system's ability to provide a service that can be justifiably trusted [1]. It can also be seen as an integrating concept which involves the following attributes [1]:

- ❖ Availability
- ❖ Reliability
- ❖ Safety
- ❖ Integrity
- ❖ Maintainability
- ❖ Confidentiality

A more elaborate definition of these attributes will be provided in chapter two.
The goal of this graduation project is to improve the dependability of the existing sensor system in such a way that it is operational for a period of at least five years while providing reliable data concerning the UHI effect. In order to achieve this goal three basic concepts should be investigated:

- the dependability of the sensor system itself
- the dependability of the data gathered by the sensor system
- the dependability of the communication of the sensor system with the system’s back-end service

These three components define the overall dependability of the system.

1.2 Research question

In order to achieve the goal set for this project, the following research questions need to be addressed:

How to develop a dependable autonomous sensor system to measure the temperature build-up in the city of Enschede?

For the sake of answering this research question several sub-questions must be addressed. These questions relate to the different aspects of the sensor system in terms of dependability mentioned in the previous section. The first sub-question concerns the dependability of the data of the system. It is important for the system to send data that is dependable. The data should be of such quality that it accurately represents the situation the system aims to measure. Moreover, the data should be ‘complete’ in the sense that it contains samples that are regularly taken. Considering all these facets, the first sub question therefore is:

What does dependability entail in terms of data gathered by the sensor system?

Concerning the lifespan of such an outdoor system, it is also important to identify factors that might interfere on some level with the system's dependability. It is important to know what dependability in relation to the sensor systems itself exactly entails. Hence the second sub-question is:

What does dependability entail relating to the sensor system itself?

Lastly, it is also of importance to know about this system's communication with the backend service. This back-end service consists of a gateway which picks up the sent payload. This gateway will then proceed to send the data to a central server. The payload can be retrieved from this service to be saved in a database. For example: without proper communication the functionality of the system is undependable and therefore useless. It is necessary to investigate the dependability of the communication infrastructure to see what it implies. Hence, the last sub question is:

What does dependability entail in terms of the system's communication?

1.3 Report Outline

Chapter two describes the state-of-the-art in the field of climate-sensing wireless sensor networks. Background information regarding the dependability of systems is also presented in this chapter. An explanation on why the proposed solution is novel in its field is used to conclude this chapter.

The third chapter describes the methodology and techniques used in this graduation project. It explains the general structure of the design process in terms of different phases. As well as different techniques that are used throughout these different phases.

The fourth chapter reports on the ideation phase of this graduation project. More information about what the ideation phase entails can be found in chapter three. The various stakeholders are identified in this chapter and through interviews a set of preliminary requirements is crafted. The context of the deployment of the sensor systems is also investigated in this chapter. This all leads to a project proposal. This proposal is specified in chapter five. The preliminary requirements are converted into system requirements. These system requirements are then categorized in either functional or non-functional requirements. By means of flowcharts the software related processes are specified and clarified. The purpose of this chapter is to exactly establish how the sensor systems should be build.

The realisation chapter describes the realisation of the design of the sensor system. The different aspects of the sensor system design will be presented. Moreover, the integration of these various aspects into a functional prototype is described. This chapter concludes with the final result of the realisation phase.

After the sensor system is realised, it will be evaluated in three different ways. The system's sensors will be evaluated on their performance. All the *must have* functional requirements will also be evaluated. Lastly, the sensor system will be evaluated together with the stakeholders. These three evaluations are described in chapter seven.

The conclusion of this graduation project is presented in chapter eight. The various sub questions (see previous section) will answered first. These answers will then be used to answer the main research question of this graduation project. A discussion of this graduation project is also presented in this chapter.

This report will conclude with a chapter that describes future research suggestions. These suggestions all relate to one of the three sub questions. So, suggestions for further research are presented in terms of the data, the sensor system and the communication.

Chapter 2 State of the art

The aim of this chapter is to provide the reader with some necessary background knowledge. This background knowledge is needed to put some of the discussed concepts into context. Furthermore, the state of the art within the field of autonomous environmental monitoring sensor systems is presented. Lastly, an expert in the field of temperature modelling was consulted and section 2.4 reports on the results.

2.1 Urban Heat Islands

The UHI effect is part of a complex process which involves both climate factors as well as human factors. This process consists of various drivers that influence each other greatly. Since this process is so complex, it differs greatly from area to area. However, literature states that there are five main factors that play a vital role in the UHI effect [3] [4], namely:

- Vegetation and evapotranspiration
- Albedo effect of surfaces
- Anthropogenic heat production
- Amount of greenhouse gasses
- Airflow in urban settings

One of the main drivers in the UHI effect is the reduction of vegetation and evapotranspiration in urban settings. Even though all sources agree on this, they do not all agree on what is causing the reduction in evapotranspiration [2], [3], [4], [5], [6], [11]. Evapotranspiration is the sum of all the evaporation of earth surfaces and transpiration of plants. In other words, evapotranspiration uses the heat to evaporate water from the soil and the plants. This heat can be extracted from the setting where the evapotranspiration takes place and in this way, it has a cooling effect on its surroundings. By reducing the amount of water available to evaporate, the rate of evapotranspiration is insufficient to cool down the city. The amount of vegetation and the amount of open soil play a key role when it comes to the evapotranspiration rate. Sealing large parts of soil with pavement results in a reduced effect of evapotranspiration [2],[3]. Although different reasons are also presented [2], [3], [5], such as removing large quantities of vegetation without sealing the soil. Water is unable to penetrate this sealed soil and needs to be drained using sewers. Rehan [7] argues that large grassy areas play a reducing role when it comes to evapotranspiration. Although other sources state that the opposite is actually the case by saying that large grassy areas increase the rate of evapotranspiration [2], [8]. While there is still some speculation about the exact reason of the reduction in evapotranspiration, it is mainly due to the reduction in vegetation and sealing of soil.

Another factor that drives the UHI effect is the low albedo of surfaces. The albedo of a surface is a number between 0 and 1 which corresponds to the reflection rate of incoming solar radiation. Where 0 means that none of the radiation is reflected and 1 means that all the radiation is reflected. All the radiation that is not reflected will be converted into heat [3], [4], [6], [9]. The albedo of structures found within an urban setting often have a low value, since they are built with material such as concrete and tarmac. These materials have a low albedo rate and will absorb heat. This heat will cause its surroundings to heat up and therefore contribute to the UHI effect [3], [6], [10]. Mohajerani et al. [4] states that asphalt concrete with a light colour (high albedo) is significantly cooler than asphalt concrete with a dark colour (low albedo) and therefore confirms this fact. All sources agree on this and state that this is one of the main factors responsible for the urban heat islands.

The third factor responsible for the UHI effect is the increasing anthropogenic heat production. This means the increased production of heat due to human activity, such as waste heat from factories or heat as a result of traffic. Literature states that climate control systems, such as air conditioners, play a significant role in the anthropogenic heat production [3], [9]. When the UHI effect is most noticeable, namely during the hot season, the demand for these systems increases drastically [3], [8], [9]. Although it is not mentioned in all sources, it is rather logical that this results in a worsening of the UHI effect. Other literature also states that traffic is a contributor to the anthropogenic heat production [3], [8], but the consequences of this factor are noticeable all year round. This is not mentioned in other sources, so its contribution to the anthropogenic heat production might be limited. Anthropogenic heat is a source of heat within cities to such an extent that the temperature rises significantly due to this heat.

The fourth driver of the UHI effect is the increased emission of greenhouse gasses. These gases possess the property that they are capable of absorbing and emitting energy in the form of heat. Nuruzzaman [3] and Mohajerani et al. [4] argue that large amounts of greenhouse gasses will prevent cities from cooling down as they absorb the heat. The origin of these increased amounts of greenhouse gases is still under debate. Nuruzzaman [3] and Mohajerani et al. [4] mention that this increased emission is due to the rising energy demand within cities.

Whereas Jabareen [8] and Qin [10] argue that traffic within cities causes these emissions to rise. Although not all sources agree on the origin of these greenhouse gases, it is clear that greenhouse gases can be found in large amounts in urban settings and contribute therefore to the UHI effect.

The last factor that contributes to the UHI effect is the insufficient airflow in cities. The airflow in cities is restricted by buildings in such a way that they block the natural occurring wind. Especially cities with a large number of skyscrapers and other multi-storey buildings suffer from this effect [3], [9]. This factor is not mentioned in all the sources, which might indicate that this factor plays less of a role compared to the other four factors. It is supported by Nuruzzaman [3] and Gunawardena et al. [9], since they mention the infrastructure of a city as being the restricting factor of airflows. It can be argued that some cities have infrastructure that is counter beneficial for the city's airflow, while others do not. Nevertheless, the wind is needed to dissipate the heat coming from roads and buildings. If the city's airflow is not sufficient, heat will get trapped in the city and contribute to the urban heat islands.

2.2 Autonomous environmental monitoring sensor systems

This section reports on various sensor systems that are similar in nature to the sensor system that will be developed during this graduation project. The first subsection is about the predecessors of this graduation project's sensor system. The other subsections are about sensor systems that are similar to this graduation project's sensor system.

2.2.1 EnskeTemp monitoring system

This graduation project aims to deliver the third-generation autonomous sensor system designed by Creative Technology students. This sensor systems aims to measure various environmental variables that relate to the UHI effect. The project continues the work done by Tom Onderwater (figure 1) and Laura Kester (figure 2), who made the first and second generation of this system. The three main aspects of these prototypes will be discussed in this section: the software, hardware and data-communication of the systems. The

Figure 1: First generation sensor system made by Tom Onderwater [12].

microcontroller used in both projects is a SODAQ One board which features a GPS module, LoRa microchip and is compatible with the Arduino IDE [12], [13]. This board is powered by a 1200mAh lithium polymer battery, which is charged via a 1W solar panel. The temperature is measured using a DS18B20 temperature sensor, which features a range from -55° to 125° with an accuracy of +-0.5°.

The second-generation sensor system made by Laura Kester also features an WH1080 anemometer made by Froggit [13]. Moreover, it incorporates an SHT15 humidity sensor made by SparkFun [13]. The entire system is placed in a white 3D-printed PLA (polylactic-acid, commonly used plastic for 3D printing) casing, which is made waterproof in order to protect the electronics. The temperature sensor is placed in such a way that it cannot be influenced by factors other than the temperature by means of a radiation shield [12]. In the second-generation system, the anemometer is placed on top of the radiation shield [13].

Figure 2: Second generation sensor system made by Laura Kester [13].

The software of both these systems is designed with limiting the energy consumption as a priority. Limiting the energy consumption is a priority since the sensor systems feature a limited power source in the form of a solar panel. Limiting the energy consumption is implemented by means of a sleep function. In the first prototype, the system is woken up from sleep mode every five minutes in order to check its location and temperature. It will then proceed to send this data to the LoRa back-end service. After this is all done, the system goes in sleep mode again. The second prototype differs from the first one in the sense that it only checks the location every five hours and it transmits the data every ten minutes instead of five. The GPS-module consumes relatively a lot of power. By checking the location less frequent the second prototype aims to limit the power consumption. Given the fact that sending the data every ten minutes suffices just as well as sending it every five minutes, the send rate is adjusted to limit the power consumption. For more details about the exact processes of the software, the reader is referred to [13, Fig. 9].

The communication of the sensor systems with the back-end service goes via a LoRaWAN network [23] called The Things Network [24]. This network is quite suited for applications such as this one, since data transfer consumes less power compared to other networks without compromising the communication range. However, due to the small transmission bandwidth requirements the network does not allow large data transfers. In Europe, using as spreading factor of 11 and a bandwidth of 125 kHz (SF11BW125) results in a data rate of 440bp/s [24]. The Things Network (henceforth referred to as TTN) allows a maximum duty cycle[1] of 1% [12], which limits the amount of data that can be send significantly. TTN allows for an average of 30 seconds of uplink time per device per day, which is equivalent to 1650 Bytes per day using the SF11BW125 standard. However, in case of [12] and [13] the data rate suffices.

2.2.2 IEEE 802.15.4 based wireless sensor network for temperature monitoring

Silveira et al. [14] attempts to show that it is feasible to develop a low-cost, low-power wireless sensor network aimed to monitor the temperature. The system utilizes an IEEE 802.15.4 network (LPWAN or low power wide area network) in order to send the data from the sensor system to the gateway, which forms the back-end service. This gateway is connected to the internet via Wi-Fi (IEEE 802.11) and sends the data to a webserver. The advantage of IEEE 802.15.4 is that the network topology is defined by the application layer of the network [14]. This means that the communication between the gateway and the sensor system can be designed by the system engineers. In case of [14], the communication between the sensor system and the back-end service is similar to that of Tom Onderwater's [12] and Laura Kester's prototype [13]. The gateway listens continuously to the network. Which means that the sensor systems save power, since they only need to send data.

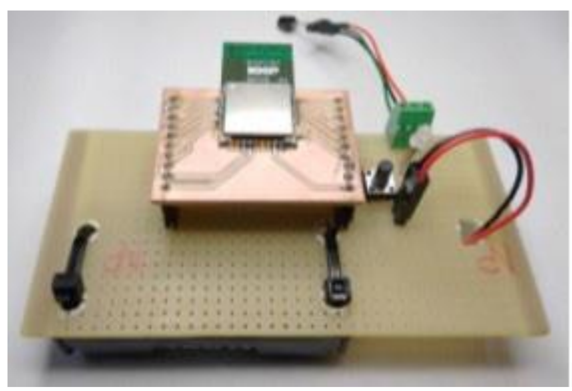


Figure 3: Sensor node with temperature sensor and IEEE 802.15.4 communication chip [14].

[1] Indicates the fraction of time a resource is busy [24].

The temperature sensor used in [14] is a LM60 temperature sensor by Texas Instruments [14], which features a range from -40°C to 125°C with a linear scale factor of 6.25mV per °C. The output voltage of the sensor is between 174mV and 1205mV.
The network of this wireless sensor network (WSN) is that of a tree topology. The network consists of end devices (sensor systems), repeaters and gateways. The JN5168-001-M00Z by NXP Laboratories is used as an IEEE 802.15.4 compliant microcontroller, whereas an ESP8266 microcontroller by Espressif is used to connect the system via Wi-Fi to the internet [14].
Contrary to the gateways and repeaters which are powered by net current, the sensor systems are powered by batteries. This means that they can be deployed both outside and inside.
However, this particular sensor system is not suited for large scale outdoor deployment. The basic framework conceives a ten-meter communication range, which does not suffice for large scale outdoor deployment [29].

2.2.3 Environmental monitoring system by employing WSN on vehicles

The aim of the project by Jamil et al. [15] is to measure air pollution in cities in order to map the areas that are heavily polluted. The deployed wireless sensor network (WSN) measures both the amount of dust in the air as well as the concentrations of different gasses. Clusters of sensor systems are placed in specific areas where the air is more polluted. The sensor systems send their data to a coordinator system which in turn sends the data to buses that pass by. These buses then dump their data onto a gateway server when they arrive at the main bus platform. The communication of the sensor systems with the coordinator systems goes via an IEEE 802.15.4 based communication protocol named Zigbee. The LTE/LTE-M network is then used to pass the data on to the server.

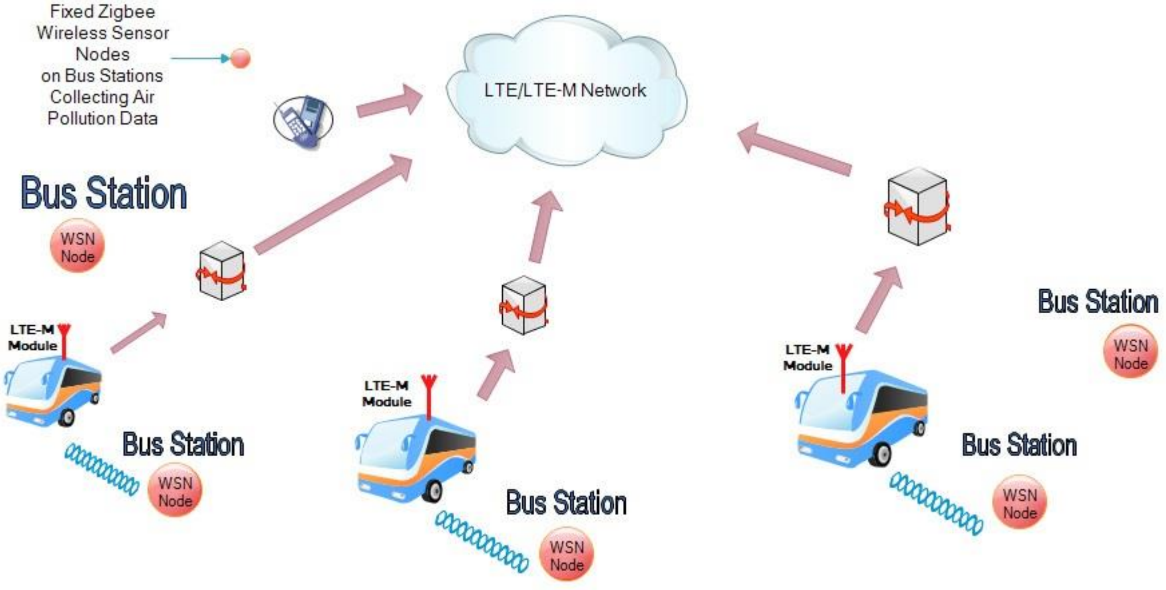


Figure 4: Diagram of the WSN infrastructure [15].

2.3 Dependability in sensor systems

Section 2.3 reports on dependability in sensor systems. A general definition of what dependability entails is presented in the first subsection. The second subsection is about maximizing dependability by implementing a variety of methods in the system.

2.3.1 Dependability and its attributes

Dependability is defined as a system's ability to provide a service that can be justifiably trusted. It can also be seen as an integrating concept which involves the following attributes [1]:

- availability
- reliability
- safety
- integrity
- maintainability
- confidentiality

Availability is defined as the readiness of the system for the correct service, so the percentage of time when a system is functional [16]. A system needs to be ready to perform a certain task to such an extent that it can be trusted that it will always perform this task as it should be.

The reliability of a system means the degree of continuity of the correct service, so the probability that a system is functional during mission time [16]. This means that a system needs to guarantee it can deliver a correct service continuously in order to be fully reliable.

The safety of a system protects the users and environment by averting potential catastrophic consequences. A system should prevent itself from entering a state with potential danger and should therefore be robust.

A system's integrity concerns its ability to avoid improper system alterations. Alterations done to the system due to malfunctions are considered improper and need to be avoided. These alterations tend to cascade a system into a state where the system does not function properly anymore.

The maintainability of system is defined as a system's ability to be modified or repaired. Maintainability can be implemented in two different ways. One of these ways is the degree to

which a system is accessible for maintenance and the other is with what ease a system can be repaired in terms of complexity.
Lastly, a system’s confidentiality is the absence of unauthorized disclosure of information. This means that data generated and used by the system cannot be viewed by other parties without the system’s consent.

Security in a system is the composite of multiple of its dependability aspects. Security is the combination of confidentiality, integrity and availability [1]. Confidentiality is of the utmost importance for a system’s security because it prevents unwanted disclosure of information. The availability attribute means that a secure system should be available for authorized actions only and the integrity attribute means the absence of unauthorized system alterations.

The energy-awareness of a system can be seen as a composite of the reliability and availability aspects. In order for the system to be ready for a correct service, it needs to be able to monitor and manage its energy consumption. For the system to be reliable, the system should possess enough energy in order to guarantee the continuity of the correct service.

In order for a system to be considered dependable, a variety of means are developed in order to attain the attributes of security and dependability. These means aim to tackle the various faults, errors and failures in a system. These means consist in four general categories. Namely [1]:

- Fault prevention
- Fault tolerance
- Fault removal
- Fault forecasting

Fault prevention concerns the system’s ability to prevent faults from happening. Whereas fault tolerance means providing the system with the ability to deliver a service in presence of faults without compromising its service. If a system is able to remove faults and decrease their severity, then a system features sufficient fault removal means. When a system is able to forecast faults, analyse current faults and determine the consequences of faults, then this system suffices with regard to the fault forecasting means.
In conclusion, dependability entails its attributes, threats and means. The attributes of dependability are availability, reliability, safety, confidentiality, integrity and maintainability. Security and energy-awareness are composite attributes but can be seen as attributes as well. The degree of dependability might suffer from various errors, faults and failures within the

system. In order to prevent these threats from happening, a system features various means that are concerned with the prevention, removal, tolerance and forecasting of faults.

2.3.2 Maximizing dependability

As stated in the previous section, the various means to attain the different attributes of dependability can be categorized in four major categories. Fault tolerance means can be seen as means for both error detection and system recovery [1]. Where error detection can happen during the performance of a service or prior to it. System recovery can be split up in error handling and fault handling with the difference being that error handling eliminates errors from the system and fault handling prevents faults from happening. Error handling is established by either going to a previous stable state or going to a new state. These two are not mutually exclusive, a system may attempt to try both. Error handling may also be established by incorporating compensation for errors into the system. Fault handling may occur by diagnosing and isolating faults in the system. In order to prevent these faults, the system may reconfigure or reinitialize itself.

Implementing these means for fault tolerance in a system depends on the assumption of faults that might occur in a system. A widely used method to achieve fault tolerance is to perform computation multiple times through multiple channels. This method is only effective for subtle faults though, via a method called rollback [17], [18]. Not all fault tolerance methods are equally effective, hence the degree of effectiveness in fault tolerance means is called the coverage [1]. Failing to achieve proper fault tolerance often depends on the wrong assumption of faults that might occur.

Removing faults in a system can happen either while the system is used or during the development of the system. In case of the latter, three steps are applied: verification, diagnoses and correction [1] [18]. Verifying whether or not the system complies with the set conditions will determine if the other two steps have to be taken. If a system does not comply with the verification conditions, the system needs to be diagnosed in order to establish the exact nature of the problem. Naturally, problems need to be corrected properly. In an iterative motion the system should be verified once again to see whether or not it complies with the verification conditions. Verifying a system can be done in either a static or dynamic setting, where the system either executes a function or it will not.

Static verifying is performed by analysing various models of the system in order to find possible faults. Dynamic verifying entails testing the system while it operates. This can be done in several ways [1]. The first method is called the oracle problem [1]. In this case the system will be observed while it operates in order to see whether or not it complies with the verification conditions. If a system is compared to another system which has been known to comply with the set conditions, then this is called golden unit testing [1]. The third known method of testing is called the back-to-back testing [1]. This means testing a system twice on a certain condition to see if the result is the same. All these methods are frequently used in combination with each other. Removing faults while the system is in use usually happens during maintenance in either a preventive way or a corrective way.

Forecasting faults in a system can be done in an evaluative way [1], [17]. The qualitative and quantitative aspects of this evaluation will result in an analysis from which plausible faults can be derived. This forecasting of faults is usually done while developing a system and consists of two phases, the modelling of a system followed by testing a system. Modelling of a system can be done by either physical faults in a system, developmental faults or a combination thereof [17].

2.4 Expert opinion

An interview was conducted with dr. ir. Wim Timmermans, who specializes in modelling temperature build-up, in order to gain some insights from an expert's perspective. The first thing he mentioned is that a lot of variables associated with urban heat islands change very rapidly. For example, the temperature very close to a surface changes with a frequency of about 50Hz and can fluctuate up to 2 degrees Celsius. Fluctuations that big are considered outliers. Therefore, the mean temperature will be significantly more stable if the mean is taken over a longer period of time. In terms of taking measurements it always is a trade-off between the sample frequency and the data rate of the communication network. More frequent measurements are preferred in systems comparable to this project, although the previously mentioned data rate often limits the amount of measurements being taken. Timmermans mentions that a sample frequency of one measurement every 20 minutes will suffices in these kinds of systems. However, he recommends taking the mean of multiple measurements and send this mean every 20 minutes. He did not name an exact number of measurements. This form of pre-processing can reduce the workload on sensor systems without compromising for measurements of lesser quality. It is assumed that taking the mean of several measurements will result in data that represents the situation better compared to taking one measurement.

Another aspect Timmermans mentioned is the fact that these sensor systems should be able to operate for an extended period of time in order to prove valuable.

2.5 Conclusion

Several topics relating to this project were investigated in this chapter. First of all, the five main drivers behind the urban heat island effect were identified:

- the reduced evapotranspiration
- low albedo
- high anthropogenic heat production
- increased amount of greenhouse gases
- the insufficient airflow

Moreover, the systems designed by Tom Onderwater [12] and Laura Kester [13] were explored. The various hardware, software and communication aspects were covered and reported on. Similar systems made by Selveira et. al. and Jamil et. al. were looked into in a similar way as well. The main difference between this graduation project and these projects is that this project's deployment is longer than that of the similar system. This means that extra features need to be incorporated in this design in order for the system to operate in a reliable manner. Similarities between this project and similar projects are for example that the sensor systems only send data and that the gateways will listen for any network activity. This means that quite a lot of power is saved on the sensor system's side. Other similarities are that networks (e.g. IEEE 802.15.4) are chosen that do not allow for a high data rate. These networks are really suited for applications such as this one because they are very efficient.

The different aspects of dependability were also examined. Dependability can be seen as an integrating concept of the following attributes:

- availability
- reliability
- safety
- integrity
- maintainability
- safety

- energy-awareness
- confidentiality

These attributes can be used to achieve a high degree of dependability by means categorized in four major groups. Fault prevention, fault tolerance, fault removal and fault forecasting means can be used to increase the level of dependability within this project's system. What these means will exactly entail in terms of design choices will be investigated further in upcoming chapters.

A sensor system will be developed which will function consistently yet is relatively cheap to build by maximizing these attributes of dependability. This sensor system is novel in its field due to the fact that it is cheap to build. Another aspect that makes it novel in its field is the fact that it can be widely deployed. The sensor system provides its own power and the network coverage (in Enschede) is such that cannot be considered a constraining factor.

Chapter 3 Methods and Techniques

Chapter three describes the various methods and techniques that are used throughout this graduation project. An overview of the design method specific for Creative Technology is given. Moreover, different techniques are presented that are used to derive the stakeholders, formulate requirements and realize concepts. The chapter concludes with a section that describes the testing procedure.

3.1 Design method

Given the fact that this project is a creative technology graduation project, the creative technology design method (figure 5) will be used. This method of designing is centred around the idea of an iterative process and combines several methods of similar disciplines into one coherent method especially suited for creative technology (henceforth referred to as CreaTe). The CreaTe design method comprises of four phases. Namely, the ideation phase, the specification phase, the realisation phase and the evaluation phase [19]. Even within these four phases evaluation and feedback play an important role in order to maximize the results of each phase.

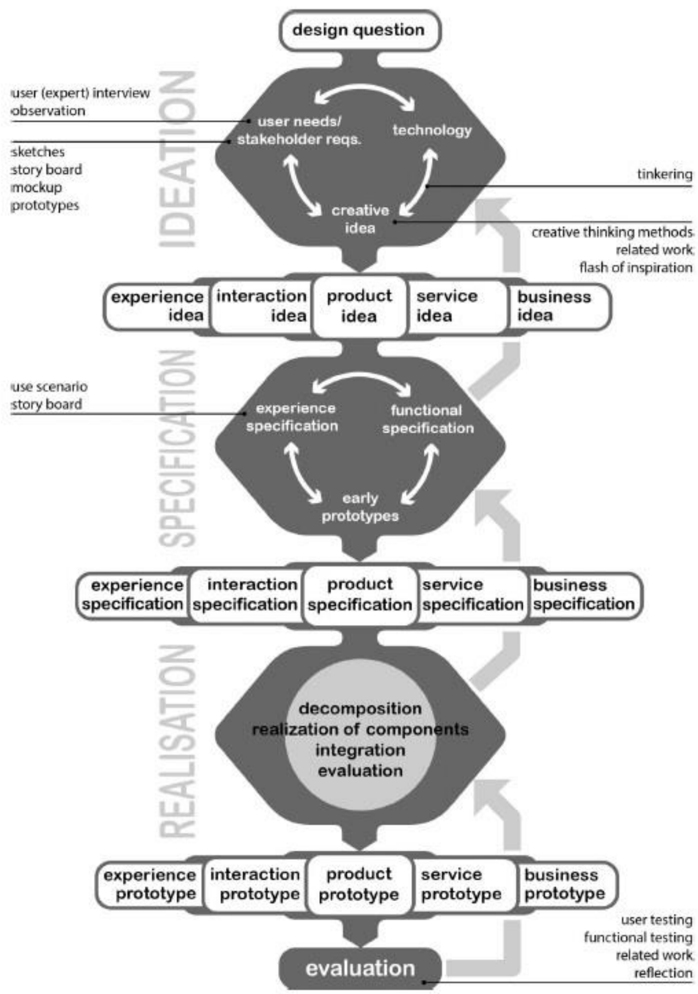


Figure 5: Creative technology design method [19].

Iteration plays an important role in the CreaTe design method as well, since it allows for quickly identifying flaws in the design. Each iteration of prototype is tested on a number of categories. The results of this testing will be implemented in the next prototype. The goal of this iterative designing is to identify as many challenges possible per prototype and to evaluate these challenges.

Features that are implemented correctly will be passed on to the next iteration (prototype), whereas others need to be improved. The final iteration will be a fully functional prototype.

3.1.1 Ideation phase

The first phase in the CreaTe design method is the ideation phase. The goal of this phase is to generate a sufficient number of ideas in order to derive some preliminary requirements for the intended product. In this phase related work is studied and future users are interviewed/observed in order to define the problem clearly, to acquire relevant information and to derive requirements. These early ideas will then be evaluated with the clients/users. Conclusion of this phase will yield a more clearly defined project idea as well as a set of preliminary requirements. These results will then be used in the next phase.

This implementation of the ideation phase in this graduation project starts with generating ideas. Using various brainstorm methods (e.g. mind maps) will be used to generate new concepts that can be implemented in the system. The casing designed by Laura Kester [13] will be examined in order to find aspects that can be improved in her design.

Moreover, the different stakeholders will be identified and analysed. An interview will be conducted with each of these stakeholders, in order to find out what their expectations of the sensors system are. These expectations will be converted to a set of preliminary requirements that can be used in the specification phase.

Lastly, the different environmental factors will be examined. These environmental factors will tell something about the context of the deployment of the sensor systems. In order for the sensor systems to operate in this context, the sensor system needs to be able to handle these factors. An example of such an environmental factor is the average precipitation in the city of Enschede. This information will be used in the specification phase to derive a set of system requirements.

3.1.2 Specification phase

The second phase will build upon the first phase and is called the specification phase. The project ideas resulting from the previous phase will be tested and evaluated using low-level prototypes. The feedback yielded from these evaluations is then used to define the project idea even more clearly. This will result in a set of requirements for the product and a specified product idea.

The specification phase of this graduation project starts with converting the set of preliminary requirements derived in the ideation phase to a set of system requirements. TTN guidelines, as well as the environmental factors and the initial brainstorm, pose a source which can be used to derive additional system requirements. The different sensors that will be used will be examined as well in order to see if they comply with the system requirements.

The software and hardware related processes will also be described in the form of flowcharts. The flowcharts provide a clear depiction of the flows of information and energy between the different subsystems. Additionally, the processes that happen in the software domain of the sensor system can be illustrated using flowcharts as well. These flowcharts show the interaction between different subprocesses in the software.

3.1.3 Realisation phase

In the third phase the project idea will be realized in the form of a functional prototype. Therefore, this phase is called the realisation phase. The product will be broken down into several components, which can be separately tested and evaluated on their specific task. When these various components are integrated into one prototype, the functional requirements of the product can be tested.

The realisation phase of this graduation project consists of building the casing, testing and interfacing all the sensors and testing the software. When all these three aspects suffice, they can be integrated in a functional prototype.

3.1.4 Evaluation phase

The last phase is called the evaluation phase. In this phase the prototype will be evaluated in order to see if the set requirements derived in the ideation phase are met. The preferred method of evaluating involves a great deal of user testing. Usually the functional requirements are already evaluated in the prior phase. However, these requirements might also be subjected to evaluation in this phase. Often the stakeholders are included in these evaluations, since they partly shaped the requirements in the first phase.

The sensor system will be evaluated by testing if the requirements specified in the specification phase are met. The system will be tested for accuracy and consistency. The sensor system will also be evaluated by the various stakeholders in order to see if the prototype meets their expectations.

3.1.5 Divergence and convergence

In the CreaTe design method the different phases can either be divergent in nature or convergent. A divergence phase will always be followed by a convergence phase. Divergent phases are phases that offer room for ideas to develop. The starting point is a question or concept from which the design space can be determined and defined. Because of this divergence unexpected solutions might present themselves. In order to formulate a definitive solution, the design space needs to be narrowed down again in the convergence phase. Every design decision the designer makes will narrow this design space until a tangible solution is reached. Divergence and convergence phases are commonly used in design disciplines, but in case of the CreaTe design method these phases are embedded in the ideation, specification and realisation phase. Each of these three phases contains a set of defined items [19] which can be the starting point of a divergent process or the end of a convergent process.

3.2 Stakeholder identification and analysis

A stakeholder identification and analysis will be done in order to see whom is affected by this graduation project. This analysis will be done using the methods described by Sharp et al. [20]. These methods focus on the interaction between stakeholders rather than the relation between the system and the stakeholder. The first step is to identify a set of so-called baseline stakeholders. These are the stakeholders that are either the users, the developers, the decision-makers or the legislators. The interaction between stakeholders and the web it forms can be derived from these initial four categories of stakeholders. Therefore, these stakeholders form the baseline stakeholder set. When all the baseline stakeholders are identified, three other types of stakeholders will be identified. These three other types of stakeholders will interact with the baseline stakeholders, which makes them stakeholders even though they are most likely not involved in the project. These three types of stakeholders are the supplier, client and satellite stakeholders. The supplier stakeholders supply either information or resources to the baseline stakeholders. Whereas the client stakeholders do the opposite, they require information or resources from the baseline stakeholders. Satellite stakeholders can interact in a variety of ways with the baseline stakeholders but are characterised by the fact that their interaction has little impact on the baseline stakeholders. The process of identifying these three types of stakeholders will be repeated with the newly identified stakeholders until a web is formed which describes all the interaction between the stakeholders.

The types of stakeholders this graduation project will focus on are the supplier and client stakeholders. There will be client stakeholders given the fact that this graduation project is part of a larger project between the municipality of Enschede and the University of Twente. Supplier stakeholders will be identified in order to use their knowledge in their respective field in developing the sensor system. In order to determine the importance of each stakeholder, an interest-influence matrix will be constructed using the method described by Mendelow [30].

3.3 Requirement elicitation

In order to elicit preliminary requirements from the stakeholders, a number of methods can be used. Techniques such as interviews, questionnaires and brainstorm sessions with the stakeholders are commonly used to derive these preliminary requirements. Because of the difference in knowledge between the various stakeholders, interviews will be used in this graduation project. The difference in knowledge can either be the difference in knowledge between the developer and the client stakeholders (i.e. the developer has more knowledge about sensor systems such as this project's) or the difference between the developer and the supplier stakeholders (i.e. the supplier stakeholders supplies knowledge to the developer). These interviews will feature a semi-structured base in order to allow follow up questions. The interviews need some structure in order to guide the stakeholders in the right direction. The answers to these structured questions then offer room for further questioning. In this way a lot of qualitive data can be retrieved.

3.4 Requirement analysis

The requirements set for this project will be split up into two categories; functional and non-functional requirements [21]. Functional requirements are requirements that describe what the system should do. Requirements that describe how the system behaves or looks can be grouped as non-functional requirements. As mentioned at the beginning of this section, the requirements for this project will be categorized into two categories. When all the requirements are identified, they will be analysed using the MoSCoW categorization method [31]. Although all requirements are important, it is necessary to prioritize and categorize them in order to deliver the maximum benefit to the project. If the most important requirements are met first, the greatest benefits to the projects are delivered early in the process. This helps stakeholders to better understand the impact of requirements and it helps in deciding what category a requirement falls. MoSCoW stands for *Must haves*, *Should haves*, *Could haves* and *Won't haves*. The requirements that fall in the *must haves* category have the highest priority and must be implemented in the design within the set project duration. Since it is critical that these requirements are implemented, the project can be regarded as a failure if one or more of the *must have* requirements are not implemented. The *should have* requirements are also important to the project, but it is not necessary that they are implement in the project duration. They can be implemented in future deliveries and are therefore not critical to success. *Could have* requirements are labelled desirable but not necessary for the project. They are usually implemented if time and resources permit it. *Won't have* requirements have the lowest priority and are therefore deemed irrelevant for the project.

3.5 Testing procedure

This section describes the different testing procedures that will be used to test the functionalities of the sensor system. In the first subsection the general testing procedure is presented which tests the sensor system for its basic functionalities. If the sensor system passes these tests, it can be regarded as functional. In the second subsection a testing procedure is presented which tests the sensor system on its functional and non-functional requirements. These test results will be used to evaluate the sensor system with the stakeholders. The testing procedures are kept general. The proper amount of times and the period of testing will be determined, when evaluating the sensor system.

3.5.1 Testing functionalities first prototype

The first prototype will be tested on the basic functionalities that implemented as software and hardware. Testing these functionalities early on in the design process allows for improvements in later iterations. The aspects of the sensor system that will be tested are the quality of the casing, the functionality of the sensors and the transmission of the data.

In order to test the quality of the casing, the casing needs to be closed as it would be when deployed. A paper towel will act as the electronics. To check if the casing is waterproof, it will be exposed to water for a period of time. This can be done by pouring water over it. After the period of time the casing will be opened and the paper towel is inspected. If the paper towel remained dry, it is safe to assume the casing passed the quality test. If so, the casing can be regarded as waterproof. Given the fact that the casing is protected against low pressure water jets, an IP66 code would be applicable to the system [32].
To test the functionality of the sensor and the transmission of the data, the sensor system needs to be deployed outside for a period of time. While being deployed, the data will be checked several times for a period of time. By inspecting the data, both the transmission quality as well as the sensor functionalities can be tested. The data from the last received message needs to be converted from hexadecimal to ascii during the test period. The quality of the transmission can be derived instantly if the message is converted correctly. By comparing the sensor data with official weather station data of the KNMI, the functionality of the sensors can be tested. If these sensor readings are similar to the KNMI's readings, the functionality of the sensors is deemed sufficient. The range of sensor values that is regarded as similar to the KNMI's values is established on site. The accuracy of the sensors will be established in later iterations; hence it is not necessary to check the sensors in-depth in this stage of the project. It suffices to check if the sensors are functional. Even though comparing the sensor data with data yielded from the KNMI implies co-location by deploying the sensor system next to the KNMI's weather station, this is not necessary. The sensors will be selected on their accuracy and resolution. If the sensors are tested before they are implemented in the sensor system, it can be assumed that they work according to what the manufacturer guarantees. They will be tested in a later testing procedure in order to establish the sensors true accuracy.

3.5.2 Testing functional requirements

The second prototype will be tested on the functional requirements. These requirements are categorized using the requirement analysis as described in section 3.4. The only requirements that must be implemented in this project are the *must have* requirements. Therefore, only these requirements will be tested.

Some of these requirements will be tested the same way Laura tested them, since these requirements were also part of Laura's graduation project [13]. These requirements concern the accuracy of the sensors, the location, the duty cycle and the battery testing.

To establish the accuracy of the sensors, they need to be compared to reference sensors. This is done by setting up the sensor system next to the respective reference sensors. These reference sensors require an accuracy of at least the required accuracy of the sensor system's sensors. Two hours of sensor data is needed to compare this to the reference sensor data. The test is considered a success if a certain percentage of the sensor readings from the sensor system differ less than or equal to the required accuracy from the reference sensor. This way the accuracy of every sensor can be established.

In order to test the accuracy of the GPS-module, a number of locations are required with known GPS coordinates. When the sensor system is placed at such a location, the sensor system needs to derive its location for a period of 30 seconds. Repeat this process five times. The requirement is met if a certain percentage of the readings differ no more than 20 meters from the known location.

The duty cycle is determined by examining the metadata of the sensor system. The sensor system needs to operate for a period of three hours. The metadata contains information about the up-link. The duty cycle can then be calculated using this formula: $D = \frac{PW}{T} * 100\%$. PW is the pulse width and T is the total period of the signal. The average up-link time needs to be calculated in order to derive the duty cycle. With a 1% duty cycle, 1650 bytes can be send by one sensor system per day. The requirement is met if the duty cycle is equal to or less than the specified duty cycle which means that the amount of data send by one sensor system should not exceed the 1650 bytes per day.

To see if the battery functions properly as stated in the requirements (chapter five), the solar panel needs to be exposed to sunlight for a several hours. This sunlight cannot be direct; hence it should be a cloudy day. Not all days are cloudless. So, in order to see if the battery will suffice throughout the year, conditions are chosen that are not considered to be ideal (i.e. cloudy day). The voltage needs to be measured prior to testing. After the sensor system is placed outside, the sensor systems need to be put in a dark place for the remainder of the day. Afterwards measure the voltage over the battery again to assess its performance. If the battery discharged to a certain percentage, the requirement is met.

The availability rate of the sensor system can be assessed by deploying it outside for period of 24 hours. If 43 out of the 48 data packages are successfully sent, the requirement is met. The sensor system should be deployed at a location with proper network coverage.

Chapter 4 Ideation

This chapter focuses on gathering the user requirements for the system and generating ideas and concepts. The different stakeholders will be interviewed in order to derive their expectations of the system. Because of the duration of this graduation project, the aim is to increase the dependability of this system in terms of its availability and energy-awareness. Even though the other attributes are also important to the system's dependability, they are regarded as being outside of this graduation project's scope.

4.1 Idea generation

Creating a certain level of dependability in a sensor system can be a process which involves both software related concepts as well as hardware related concepts. In order to come up with these various concepts a mind map was constructed (figure 6). Such a mind map allows for the generation of a lot of ideas. By mapping these ideas to other ideas, an elaborate map can be constructed containing a variety of concepts.

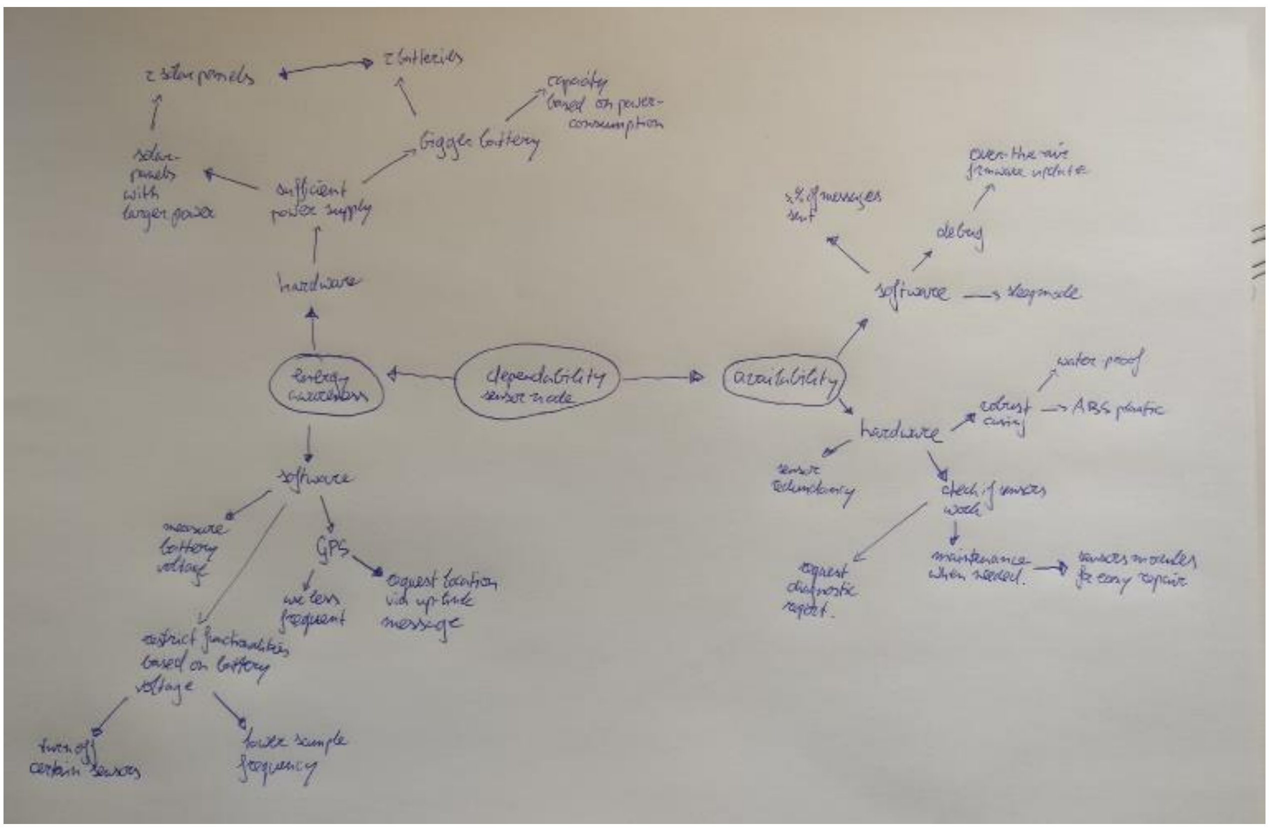


Figure 6: Ideation mind map.

The casings of the previous projects were investigated as well. Given the fact that this graduation projects continues the work done by Laura Kester, the casing designed by her will be used as reference. Even though her casing would suffice for this project (it proved itself worthful; for more information see [13]) several of its features were investigated to see if there was room for improvement. First of all, the number of solar panels is likely to be changed in this graduation project. Due to fault tolerance, a solar panel with more power is preferred. This can also be implemented by incorporating two solar panels (figure 7B), which adds some form of redundancy to the sensor system. Placing these two solar panels on opposite sites will result in more exposure to the sun, because the effect of shade on these solar panels is less. Positioning the solar panels on one side of the casing yields an even greater exposure to the sun (figure 8). The light will hit the solar panels with a certain angle from this position, which will maximize the amount of power generated throughout the day.

One of the problems mentioned in Laura's report [13] is the fact that the humidity sensor sometimes reports relatively high values. This might be because of built-up moisture in the humidity sensor compartment. In this project, the humidity sensor will likely be placed next to the temperature sensor in an attempt to tackle this problem described by Laura. In order to minimize the moisture built-up, the radiation shield containing the temperature sensor and the humidity sensor will be placed below the microcontroller (figure 7C). This way condensed water can drip off the casing. By doing so the windspeed sensor can be placed on top. This means that the solar panels will suffer less from shade from the rest of the sensor system. Since the temperature sensor needs to be in an upright position to accurately measure the air temperature in the radiation shield, a modification to the casing needs to be made in order to facilitate proper temperature measurements. The temperature sensor will be placed in an upright position in the radiation shield, which means that the wire needs to be extended (figure 8).

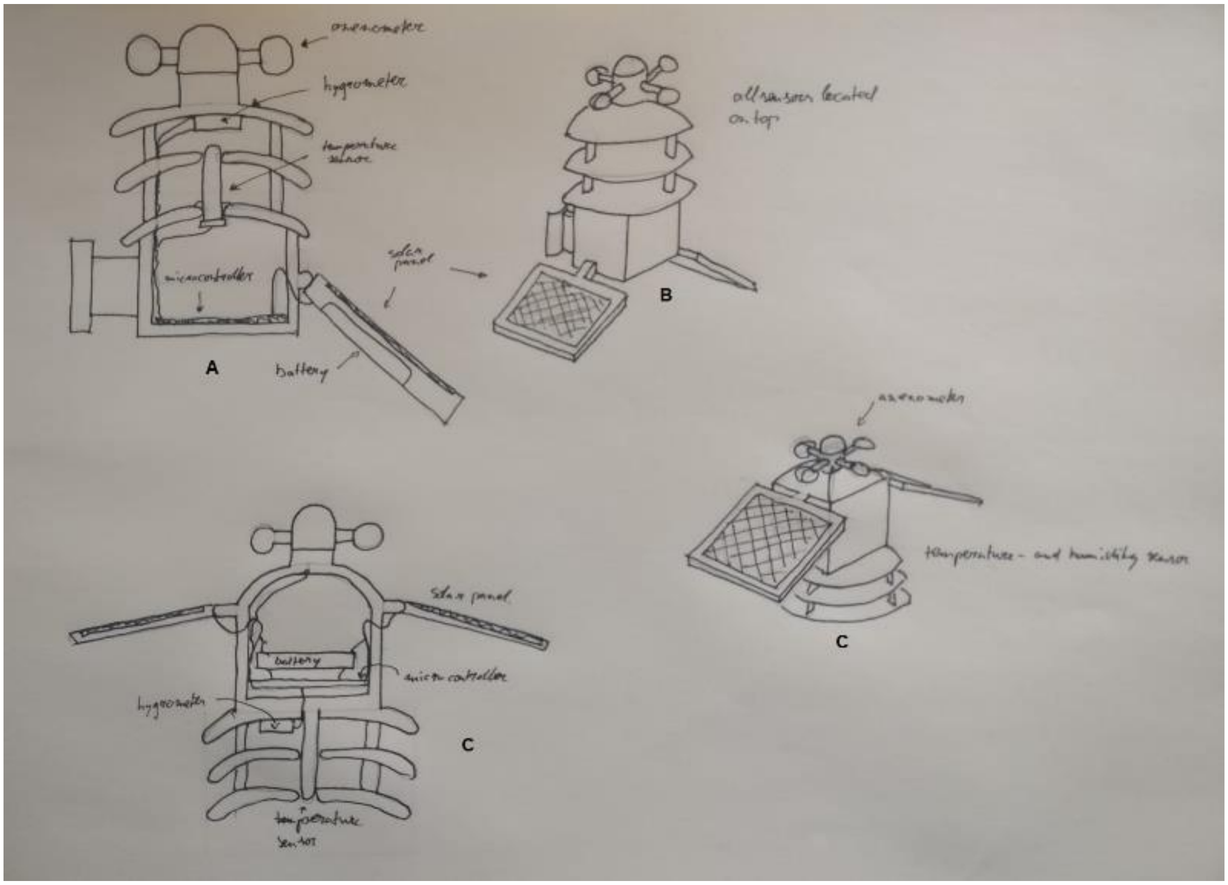


Figure 7:Various sketches of the casing. A: side-view of current sensor system. B: sensor system with two solar panels and sensors on top. C: Sensor system with two solar panels and the radiation shield on the bottom.

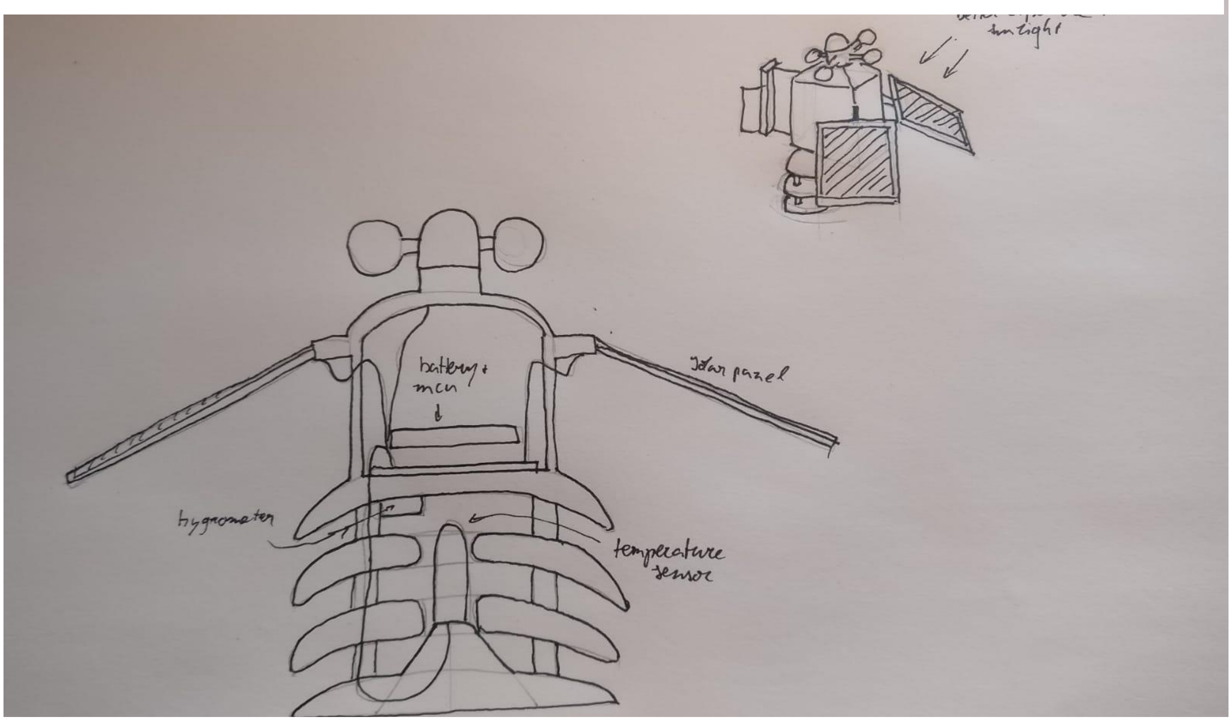


Figure 8: Final sketch of the casing

4.2 Stakeholders

This section reports on the interviews conducted with the stakeholders. The importance of these stakeholders is depicted in an interest/influence diagram. Preliminary requirements were derived from these interviews and can be found in a table at the end of this section.

4.2.1 Wim Timmermans ITC

In order to gain more insights into Wim Timmermans' expectations of the to-be-build sensor system, a semi structured interview was conducted (which can be found in appendix A). The various aspects of dependability were presented and explained to Wim. Even though Wim recognizes that all attributes are of importance, the focus of the interview was shifted to the two attributes mentioned in the introduction of this chapter: availability and energy-awareness.
In terms of availability Wim could not give a specific percentage. He mentioned that it depends on the variable that is being measured (air temperature, windspeed and humidity) and the sample frequency. For example, if the air temperature is measured every day then it is important that the system is available 100%. However, if the air temperature is measured every second then the availability can be significantly lower.

According to Wim the air temperature is the primary variable, followed by the windspeed and then the humidity. This means that the availability in terms of these three variables can be different and that the temperature sensor demands the highest availability.

Wim also mentions that the quality of the data should be such that it accurately represents the situation on site. Wim was asked if the specifications of the sensors used in the 2nd generation sensor system made by Laura [13] would suffice. According to Wim, these specifications still meet the required specification in order to accurately yield data. This means that the air temperature should be measured with a 0.5°C accuracy, the windspeed with a 0.1m/s accuracy and the humidity with a 3% accuracy. The location derived using GPS should be accurate up to 20 meters. The location of the sensor system is important since the measurements taken by the sensor system tell something about the situation on site. This means that the location needs to be constant with a deviation of 20 meters.

Another aspect Wim mentioned is the fact that climate-related models use time intervals of either half an hour, an hour or a day. This means that the data yielded by the sensor system should be send either every half hour, hour or day for it to be useful to check climate-related models. Multiples of these time intervals are also allowed, since the data can still be formatted to fit in these time intervals. This way a dataset can be made that uses the proper time intervals to verify these climate-related models. One value per time unit (i.e. half an hour) will suffice to depict the situation. This value can either be the mean taken over this half hour or one value.

4.2.2 Hans Scholten University of Twente

Hans Scholten is an expert in the field of Wireless Sensor Network (WSN) and was interviewed in order to derive requirements that are common in similar systems. Since Hans already has extensive knowledge in the field of dependability in sensor systems, the conducted semi-structured interview (appendix A) focussed on the two attributes of dependability that will be implemented in this graduation project. Hans' input with regard to these two attributes will yield requirements that can be seen as standards in similar systems.

The first thing Hans mentioned is that an availability rate of 90% suffices. In terms of time this means a maximum downtime of 40 days per year. This means that the sensor system might not be available for several days at once. The gap between two moments of availability is therefore too large for this project, since such gaps will result in poor data. However, if the time window is changed, a 90% availability rate does suffice. Therefore, in terms of this graduation project, the sensor system should have an availability rate of 90% per day.

When it comes to the energy-awareness attribute of this project, Hans notes that a system such as this one should be energy autonomous. This means that the system should provide its own power and maintain a sustainable power consumption. Sustainable power consumption means that the system does not consume more power than it is able to generate. By doing so, the system will sustain adequate power levels and can therefore function properly.

According to Hans a system such as this one does not need scheduled or preventive maintenance. This means that the sensor system should be repaired or modified right before it is needed, the so-called just-in-time maintenance. This just-in-time maintenance requires the sensor system to tell when it is time for maintenance. This means that the sensor system should feature a diagnostic mode in order to check if its functionalities work properly. Testing if the functionalities work properly means that all the subsystems (i.e. sensors) should be used every once in a while, to see if they work correctly. In terms of the climate-related sensors (i.e.

windspeed, air temperature and humidity) this poses no problems, since they are used frequently when the sensor system is deployed. The GPS module is not needed frequently and should therefore be tested every day to see if it works properly.

Another thing Hans mentions is the fact that the data yielded by the sensor system might be compromised by several factors. He notes that the battery level might interfere with functionalities of the sensors. If the battery's voltage is not sufficient, the sensor readings might not be accurate, since the sensors only work properly within a certain range of voltage. The location of the sensor system is also important. This means if the sensor system is moved too much the data does not depict the correct situation, since it depicts the situation of another location. Combining this means that the sensor system should take into account the location and battery level and label the data accordingly with a data quality stamp. If the location is changed too much (20 meters) the sensor system should give a warning about the quality of the data. In case of a low battery level, the system should do the same. The user of the data should be notified about the fact that the data might not be accurate.

4.2.3 Rik Meijer Municipality of Enschede

Rik Meijer works for the municipality of Enschede and he is on behalf of the municipality involved in the project of the municipality of Enschede and the University of Twente. This graduation project aims to contribute to this joined project. Prior to the semi-structured interview, Rik was informed about the various aspects of dependability and what they mean in terms of this project. The questions of the interview can be found in appendix A.

The goal of the municipality is to gain insights in the temperature distribution in the city of Enschede. Within the scope of the current project, this means comparing data of different locations. Eventually the overall goal (for future projects) is to predict what this heat distribution means for the citizens of Enschede. For example, how they perceive the temperature and if it is perceived as a nuisance.

One of the sources the municipality uses right know is the klimaateffectatlas [25]. This atlas lacks detailed information on a local level and therefore the municipality is looking into ways to yield more precise and detailed data of Enschede.

In terms of availability Rik Meijer mentions that a 90% availability rate per day suffices. However, he also mentions that large gaps in the dataset are not desired. It would pose a problem for him if the sensor system is not available 10% of the day in a consecutive manner. Meaning that the sensor system is available for 90% of the day and unavailable for the remaining 10%. Rik also mentions that the data yielded by the sensor system does not need to be precise up to a hundredth of the measured unit (i.e. 18.52°). Climate-related policies will not be based on such precise data. Hence the resolution of the current sensors suffices.

4.2.4 Interest/Influence matrix and preliminary requirements

This section reports on the relationship between the various stakeholders in terms of interest and influence in the project. The matrix (figure 9) consists of four quadrants that depict the low interest/low influence stakeholders as well as the low interest/ high influence, high interest/low power and high interest/high influence stakeholders.

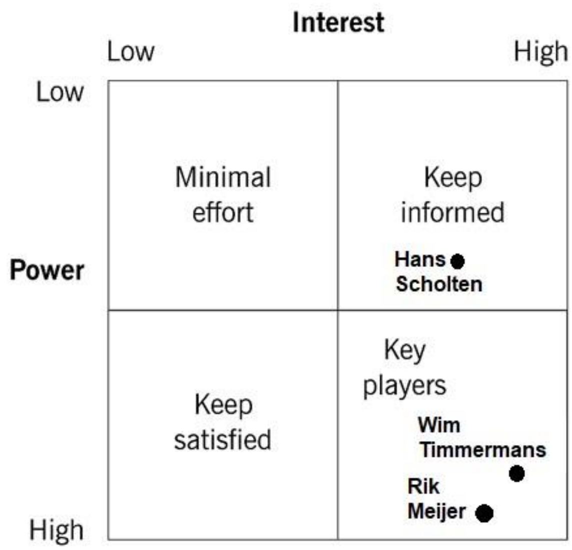


Figure 9: Interest/Influence matrix based on the theory of Mendelow [30].

The following table contains the set with preliminary requirements. The requirements are described and categorized. The source of the requirement is also mentioned.

NR.	PRELIMINARY REQUIREMENT	SOURCE	CATEGORIE
1	The sensor system should be able to measure three variables (temperature, windspeed and relative humidity) with accuracy as specified in previous prototype made by Laura [13].	Wim Timmermans	Availability
2	The sensor system should be able to derive its location with an accuracy of 20 meters.	Wim Timmermans	Availability

3	The sensor system should be able to send data every half hour.	Wim Timmermans	Availability
4	The sensor system should not allow for large gaps in data due to unavailability.	Wim Timmermans & Rik Meijer	Availability
5	The sensor system should be available 90% of the day.	Hans Scholten & Rik Meijer	Availability
6	The sensor system should be energy autonomous.	Hans Scholten	Energy-awareness
7	The sensor system should be able to operate without needing scheduled maintenance.	Hans Scholten	Availability
8	The sensor system should be able to test if the various subsystems (i.e. sensors) still function properly.	Hans Scholten	Availability
9	Location and battery-level might interfere with the data quality, the user should be made aware of this by the sensor system.	Hans Scholten	Availability
10	The sensor system should be able to measure the previously mentioned three variables with a resolution of one decimal.	Rik Meijer	Availability

4.3 Environmental factors

Given the fact that the sensor systems will operate outdoor in the city of Enschede, the context of their deployment was further examined. By doing so the various environmental factors that are of influence on the sensor system's performance are quantified. These environmental factors describe the climate in Enschede and therefore a dataset by the KNMI [22] was used to explore Enschede's climate. These environmental factors are the temperature in Enschede, as well as the humidity, windspeed, amount of sunlight and the precipitation.

Enschede’s climate is described as a moderate sea climate, which means that the winters and summers are mild in nature. Compared to climates situated inland, the winters are less cold and the summers are cooler. The lowest average temperature per day in Enschede (2018) was -5.2°C, while the highest average temperature per day was 28.9°C. These temperatures are daily averages, which means that the temperature can be either higher or lower on specific moments. In 2018 the highest measured temperature was 35.7°C and the lowest registered temperature was -9.6°C.

Given the fact that Enschede is situated quite far from the sea, the average windspeed is lower compared to coastal cities. The average windspeed per hour is between 1.6m/s and 5.5m/s, although some gusts of wind can reach speeds of 23m/s (82.8 km/h). On average, the fastest gust of wind per day reaches a speed of 10.2m/s (2018).

The relative humidity is on average between 56.22% and 92.80% per day. The maximum relative humidity measured in 2018 was 99%, while the minimum relative humidity was 18%. On average the annual precipitation in Enschede is 750mm with a maximum hourly precipitation of 21.8mm. However, precipitation levels can reach up to 100mm/hour at specific moments. The amount of sunlight in Enschede is on average 8 hours per day during the winter and 16 hours per day during the summer.

4.4 Conclusion

In this chapter the various stakeholders were identified and interviewed in order to derive a preliminary set of requirements. The environmental factors that are of influence when the system is deployed were investigated and the possibilities within the design were addressed as well. This resulted in a more precise project description.

The sensor system’s casing will be altered to facilitate two instead of one solar panel. In order to make room for these solar panels, the radiation shield will be attached on the bottom-side of the sensor system (figure 8). This way the solar panels will suffer less from shade from the radiation shield. The windspeed sensor will remain on top. Another advantage of moving the radiation shield to the bottom of the sensor system is the fact that possible moist built-up near the temperate sensor and humidity sensor will be minimized. This is because this moisture is able to drip of the radiation shield.

The system will feature the same sensors used in the previous version built by Laura, since these sensors are accurate enough to yield data relevant to this project. The power supply of the sensor system will be changed in order to facilitate the sensor system throughout the year. This means that the batteries will be larger in capacity and the solar panels will yield more power and provide the system with more fault tolerance. The casing will most likely consist of a plastic type that is more durable than the one currently used by Laura (i.e. PLA). Another aspect of this sensor system is the fact that it is able to establish the quality of the data by monitoring the battery and sensors. The send rate of this sensor system will be changed as well, so it sends data every 30 minutes. The values that are send will mean values. This mean is taken over several measurements throughout the time between data packages.

Chapter 5 Specification

In chapter five the various aspects of this graduation project will be specified. The preliminary requirements from the ideation phase (chapter four) will be formulated in a system-oriented way and can thus be categorized as functional or non-functional. The different sensors are discussed in this chapter as well. Moreover, the various processes that are involved in functionalities of the sensor system are displayed as flow diagrams.

5.1 Requirements

Both the functional and the non-functional requirements are given in this section. They are derived from the user requirements resulting from chapter four. All the requirements are described in convenient tables that break down the requirements in different sections. The requirements are described using a general description, the rationale behind the requirement, the source and the requirement is tested. The requirements are also categorized using the MoSCoW method (section 3.4) and the priority is mentioned in each table. Possible conflicts with other requirements are also described, as well as the attributes used by the requirement and its value. All the requirements can be found in Appendix B. A short summary of the requirements is provided below:

Functional requirements

The system must	Source
Be able to measure the relative humidity	1, 3 and 5
Be able to measure the air temperature	1, 3 and 5
Be able to measure the windspeed	1, 3 and 5
Be able to derive its location	1 and 3
Be able to send a payload every 30 minutes	1
Be able to test if the GPS-module still functions	2
Be able to label the data with a quality mark	2
Be able to send the payload within a 1% duty cycle	4

Be able to send payload with a size of 21 bytes	4
The system could	
Be able to enter a diagnostic mode	2

Table 2: Analysed functional requirements.

Non-functional requirements

The system must	**Source**
Be attachable	6
Be available 90% of the day	2
Be energy autonomous	2
Be able to operate in temperatures of -25°C minimum	5
Be able to operate in temperatures of 50°C maximum	5
Be able to operate in a humidity of 0% minimum	5
Be able to operate in a humidity of 99% maximum	5
The system should	
Be able to withstand precipitation of 100mm/hour	5
Be able to withstand wind gusts of 12 Bf.	5
Have a high albedo factor (i.e. light colour)	6
Have a durable casing of ABS or PETG	6

Table 3: Analysed non-functional requirements.

Legend: 1= Wim Timmermans 2= Hans Scholten 3= Rik Meijer 4= TTN guidelines 5= Environmental factors 6= ideas and sketches

5.2 Sensors

This section contains information about the different sensors that will be used. A general overview of the advantages of different types of sensors is given. Also, the functional requirements that relate to the to-be-measured variables are linked to the different sensors.

5.2.1 Temperature sensor

A digital temperature sensor is preferred over an analog sensor when it concerns this graduation project. Digital temperature sensors return the measured temperature digitally, which means that the microcontroller does not need to process data in order to retrieve the correct temperature. In case of the analog sensors, the microcontroller would have to process analog values in order to derive a usable temperature value. This means that the sensors need to be calibrated. Digital temperature sensors have the advantage that (most of them) are calibrated by the manufacturer. The electronics embedded in the sensor do the data-processing and return the temperature.

The temperature sensor needs to be able to accurately measure temperatures in the range of -25°C and 50°C. By doing so it should return the temperature with an accuracy of ±0.5°C (see Appendix B, functional requirement nr. 2).

5.2.2 Windspeed sensor

A mechanical contact closure sensor will be used to measure the speed of the wind. Such a sensor closes a switch every time it rotates. Often enough, this is done by a reed switch which closes when a magnet passes by. This magnet is embedded in the rotating part of the sensor and this rotating part is moved by the wind. The reed switch can be found in the static part of the sensor. By measuring the amount of pulses per second, the rotation can be determined by a microcontroller. This rotation is then used to calculate the windspeed.

The advantages of such a sensor is that it is able to accurately measure the windspeed at both low speeds as well as high speed, while being relatively simple and cheap. Because a constant windspeed will return a constant amount of pulses per second, the windspeed sensors of this type are relatively easy to calibrate.

The windspeed sensor needs to be able to accurately measure windspeeds in the range of 0 to 10 Bf. peak speed. This means that the measurement range is between 0m/s and approximately 30m/s. The windspeed sensor needs take measurements with an accuracy of 0.1m/s.

5.2.3 Humidity sensor

Like the temperature sensor, a digital humidity sensor is preferred over an analog one. The main advantage of such sensors is that they are calibrated by the manufacturer. Most humidity sensors are either based on capacitance or resistance. In case of capacitive humidity sensors, the capacitance changes relative to the humidity. The main advantage of such sensors is that they are very consistent. Resistive humidity sensors measure the humidity by monitoring the impedance of the sensor. This impedance changes based on the humidity. Given the fact that humidity is related to the temperature, most humidity sensors also feature a temperature sensor. Either one of the humidity sensor types is suitable for the application of this graduation project.

The humidity sensor needs to be able to measure the relative humidity in the range of 15% to 99%. The humidity sensor must do so with an accuracy of ±3%.

5.3 System architecture

This section describes the system architecture of the envisioned sensor system using a flowchart. The sensor system is broken down in several subsystems that are connected with each other via data streams and power connections (see figure 10).

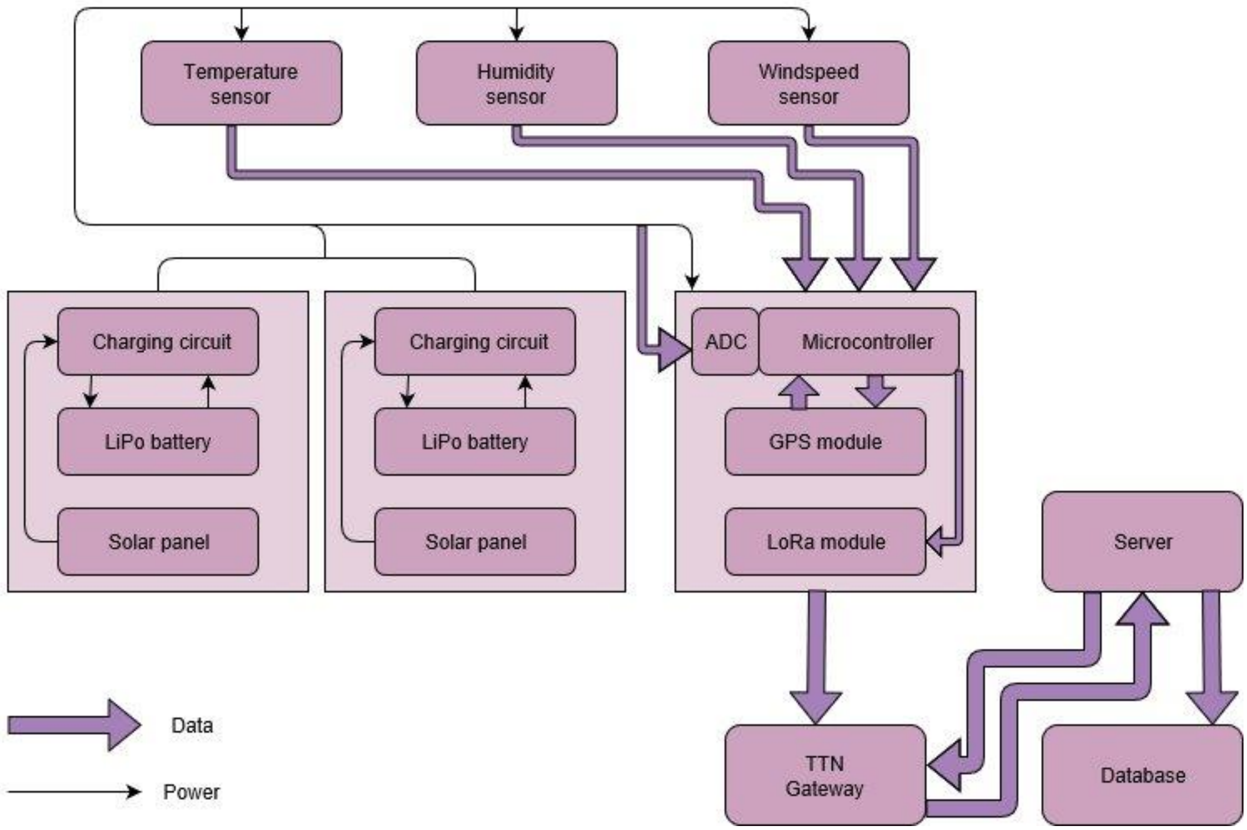


Figure 10: Hardware flowchart.

The system features two batteries and their charging circuit, one for each of the solar panels. In order to maintain a desired output voltage, the two power supplies are placed in parallel with each other. The microprocessors analog-to-digital converter monitors the voltage of both batteries. The microprocessor reads the data from all the sensors and the GPS-module and sends this data to the LoRa module after it is processed and formatted. The LoRa module then proceeds to sends the data to TTN gateway. Through this gateway, the data will end up in the database.

5.4 Software flowchart

This section describes the software of the envisioned sensor system by means of a flowchart. The interaction between the various processes is depicted in figure 11.

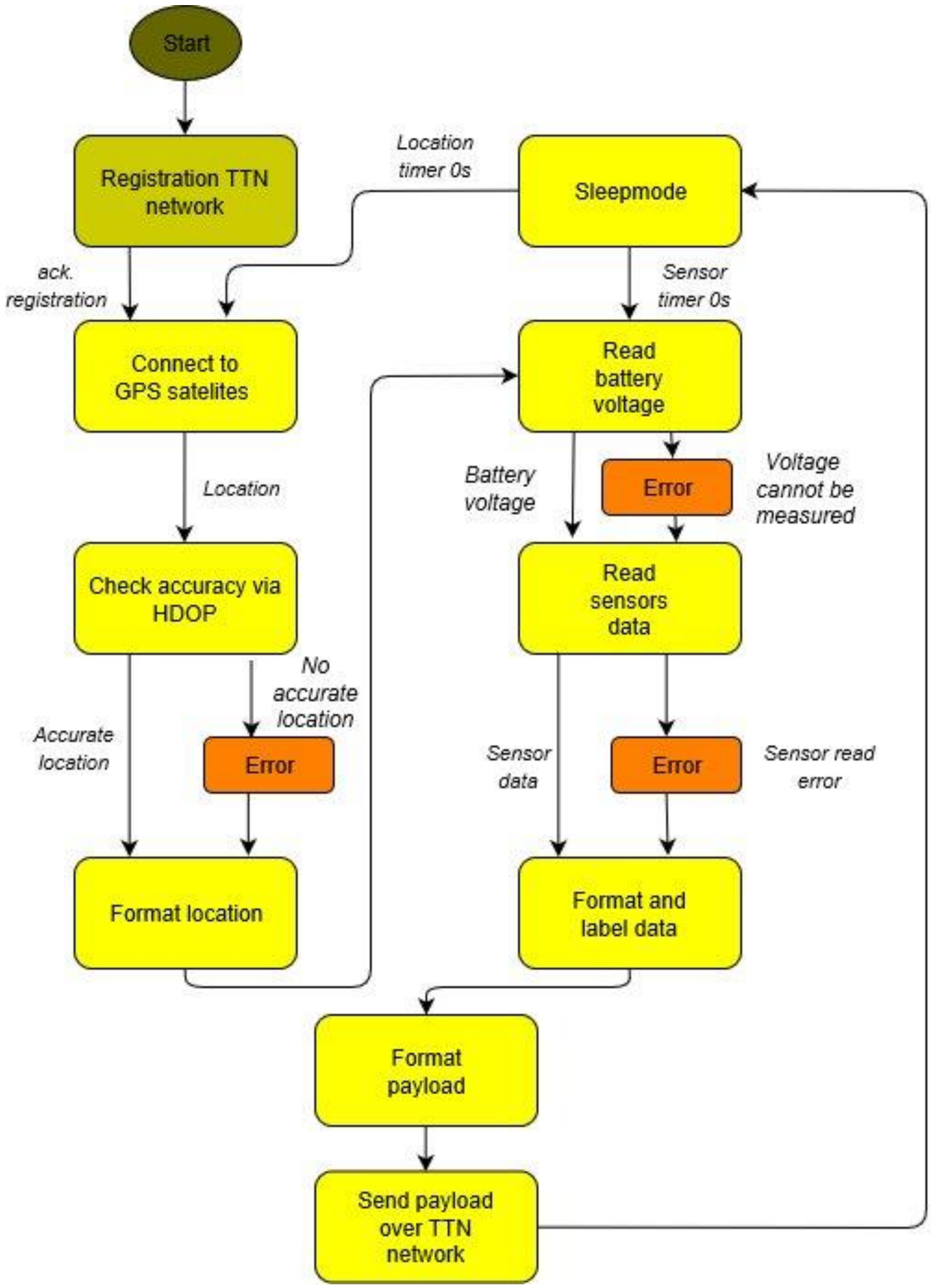


Figure 11: Software flowchart

Figure 11 depicts the software of the sensor system. When the system boots it attempts to register to TTN network. The system will only be able to boot if the battery is sufficient. If the sensor system is connected to TTN network, it will connect with a suitable number of GPS satellites in order to derive its location. When the location is derived, the sensor system will check the accuracy of the location via the horizontal delusion of precision (HDOP). If the accuracy does not suffice an error message will be generated. The sensor system proceeds with formatting and labelling the location to make it ready to be send. The sensor system will then measure the voltage of the battery. An error message will be generated if the voltage cannot be measured. This is only a case if an error occurs in the ADC. After measuring the battery's voltage, the sensor system proceeds with reading the sensor's data. If the data of one of the sensors cannot be read, an error message will be generated. The sensor data is formatted and labelled.

When the location and sensor data are formatted, the sensor system will format the entire payload that will be send (payload is converted to a string of ascii values). Converging the payload into a suitable format for TTN network happens automatically by the microcontroller. This payload will be send to a TTN gateway. After the payload is send the sensor system will go into sleep mode. This sleep mode is terminated when timers for the sensors and the location are done. The sensor timer goes off every 30 minutes and the location timer goes off once a day.

5.5 Conclusion

This chapter reported on the results from the specification phase of the project. The various user requirements from the previous chapter were converted to functional and non-functional requirements. These requirements form the base for the different flowcharts that can be found in this chapter. The different sensors used in this project were also investigated to see if they comply with the various requirements. Combining all this information makes it possible to start the realisation of the sensor system.

Chapter 6 Realisation

This chapter describes the realisation phase of this graduation project. The realisation phase consists of developing two prototypes. The different sections of this chapter report on the results per prototype. A brief summary of the evaluation per prototype is given. Chapter 7 elaborates on these evaluations.

6.1 First prototype

The first prototype is developed in order to see if all the subsystems work function properly. This prototype will not be ready to be deployed yet, given the fact improvements need to be made to the design in order to facilitate deployment.

6.1.1 Casing

The casing design (figure12) of the first prototype is based on the design made by Laura Kester [13]. Autodesk's Fusion 360 was used to design the casing. A few alterations were made based on the ideas generated during the ideation phase. First of all, the radiation shield was moved from the top part of the sensor system to the bottom part of the sensor system. The anemometer remains at the top part of the sensor system. This means that a new part had to be designed which can fit the anemometer on top of the casing of the microcontroller. Furthermore, a new bottom had to be designed which fits the radiation shield on the casing of the microcontroller.

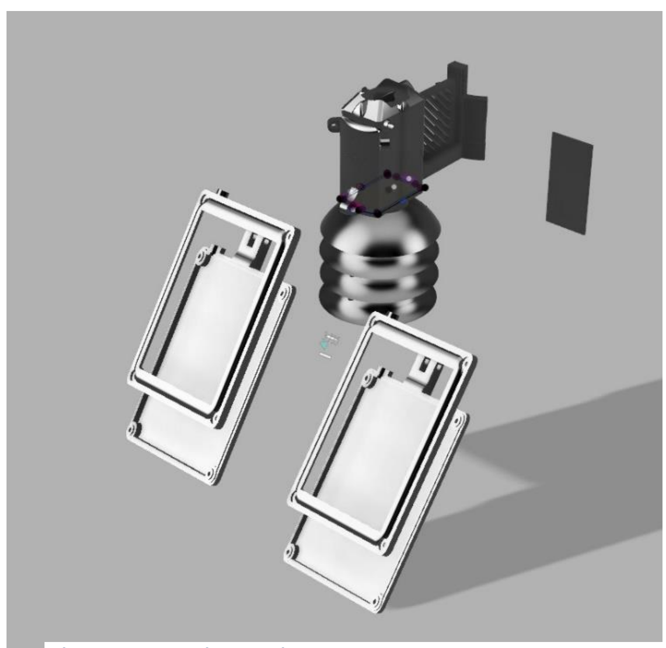


Figure 12: Design casing.

Due to the fact that the required 1-Watt solar panels used in Laura's design were sold out, 1.5-Watt solar panels were chosen instead. These solar panels have different dimensions compared to the 1W solar panels. Hence the solar panel casing needed alteration to fit the proper dimensions. Two solar panels will be used in this project's design, that is why there is a need for a second solar panel casing. In order to attach these solar panels (i.e. their casing) to the main casing of the microcontroller, two new points of attachment needed to be design. These two points can be found on the outside of the main casing near the top.

6.1.2 Interfacing sensors

Three different sensors will be used in this sensor system to measure the air temperature, the relative humidity and the windspeed. These sensors were briefly discussed in section 5.2. This section elaborates on the implementation of these sensors in the sensor system.

The temperature sensor that is used in this sensor system is the Dallas DS18B20. This sensor features a watertight metal probe that contains the sensor. This temperature sensor is digital and uses a one-wire bus to communicate over. This sensor features a 9/10/11/12 bits resolution, which can be set by the microcontroller. The range of measurements ranges from -55°C to 125°C and the accuracy is +-0.5°C when the temperature is between -10°C and 85°C. Outside of this range the accuracy is not guaranteed by the manufacturer. However, this sensor will suffice in terms of the specified requirements. The sensor’s operating voltage ranges from 3V to 5.5V, which makes it compatible with a lot of microcontrollers. Setting the resolution affects the response time. Setting the resolution to 9 bits (0.5°C) will result in a response time of 93.25ms [26].
Given all the features mentioned above, this digital temperature sensor is well suited for an application such as this sensor system.
As priorly mentioned the DS18B20 communicates over 1-Wire and has therefore three wires that need to be attached (i.e. the power supply, ground and the data bus). A 4.7kΩ resistor connects the data bus with the power supply. This is because the 1-Wire bus allows for multiple device to be connected via one wire. By incorporating this resistor, the master device can ask for data and the slave device can give data. The data bus of the temperature sensor is connected to one of the I/O pins of the microcontroller. The ‘DallasTemperature.h’ software library is used to retrieve data from the sensor.

The humidity sensor used in this project is the DHT22 capacitive humidity sensor by Adafruit [28]. This sensor features a wide operating voltage range, namely between 3.3V and 6V. This

makes this sensor ideal for an application such as this one, since the sensor system cannot guarantee a constant voltage supply. This is because of the batteries and the solar panels. The measurement range of the humidity sensor is between 0% and 100%, which is amply sufficient for this sensor system. In the range from 10% to 90% the manufacturer mentions an accuracy of +-2%. Another advantage of this sensor is its low power consumption and fast response time. In standby mode, the sensor draws only 40μA compared to the 1.5mA it uses when in operation. This makes such a sensor ideal for long-term deployment.
Lastly, the DHT22 also features a temperature sensor. This temperature sensor uses the same bus as the humidity sensor and can therefore be used without changing the infrastructure of the sensor system (similar to the DS18B20). This means that this temperature sensor can be used in cooperation with the DS18B20 temperature sensor. This sensor features a casing around the sensor which protects it from moist.
Just like the DS18B20, the DHT22 communicates via one bus. However, the data communication of this sensor is not 1-Wire compatible. Like the DS18B20, the DHT22 requires a (10kΩ) resistor to connect the data bus with the power supply. The 'DHT.h' library is used to derive data from the sensor.

The windspeed sensor that will be used for this sensor system is the Froggit WH1080 [27]. This windspeed sensor is called a cup anemometer, since it uses a rotating cup to measure the windspeed. A reed switch is used to measure the rotations, the reed switch closes four times in one rotation. According to the datasheet, the anemometer gives one pulse per second if the windspeed is 2.4 km/h. The accuracy of the sensor is 0.1m/s.
Given the fact that the windspeed sensor only uses a reed switch to register rotation, allows for widespread use. The sensor does not require special interfacing as the microcontroller will determine the windspeed by measuring the frequency of the pulses. The power supply is not a constraining factor, as the anemometer will use the microcontroller's power supply to work.
As mentioned before, this windspeed sensor is a cup-type anemometer which uses a reed switch to detect rotation. In one rotation the reed switch closes four times. The reed switch closes with a frequency of 1Hz when the windspeed is 2.4km/h [27]. This means that the reed switch closes every 1.5 seconds if the windspeed is 1 m/s. The amount of pulses is measured over a period of 3 seconds. The windspeed can then be calculated using this formula:
$wind\ speed = \frac{C}{P} * \frac{2}{3}$ (*C* is the amount of pulses and *P* is the period) [13]. A 100nF capacitor is placed in parallel with the anemometer in order to take care of the bouncing contact effect. The

resistance of a closed reed switch was established to be 5Ω, which means that the charging time of the capacitor is so small it will not interfere with the measurements. One wire of the anemometer is connected to the microcontroller's ground, while the other wire is connected to one of the I/O pins of the microcontroller. No library is used to measure the windspeed.

The microcontroller that is used in this sensor system is the SODAQ One v3 [33]. This microcontroller is based on the ATSAMD21G18 chip and features 14 I/O pins. The microcontroller is compatible with the Arduino IDE. The microcontroller also features a RN2483A LoRa microchip, which enables it to connect to TTN. Another feature of the SODAQ One is the uBlox EVA 8M GPS module. This GPS module can be used to derive the microcontroller's location. The built-in 10-bit ADC can be used to measure the voltage level of the battery. The schematic of the sensor system can be seen in figure 13. All the sensors have been equipped with connectors in order to easily connect them with the microcontroller.

The entire system is powered by two solar panels of 1.5 Watt each. Each panel features its own charging circuit and battery, which makes them run independent from each other. A TP4056 charging circuit is used, as well as an 1200mAh LiPo battery. Two 1N4007 diodes are included in the circuit to prevent currents from leaking into the charging circuit via its output terminals.

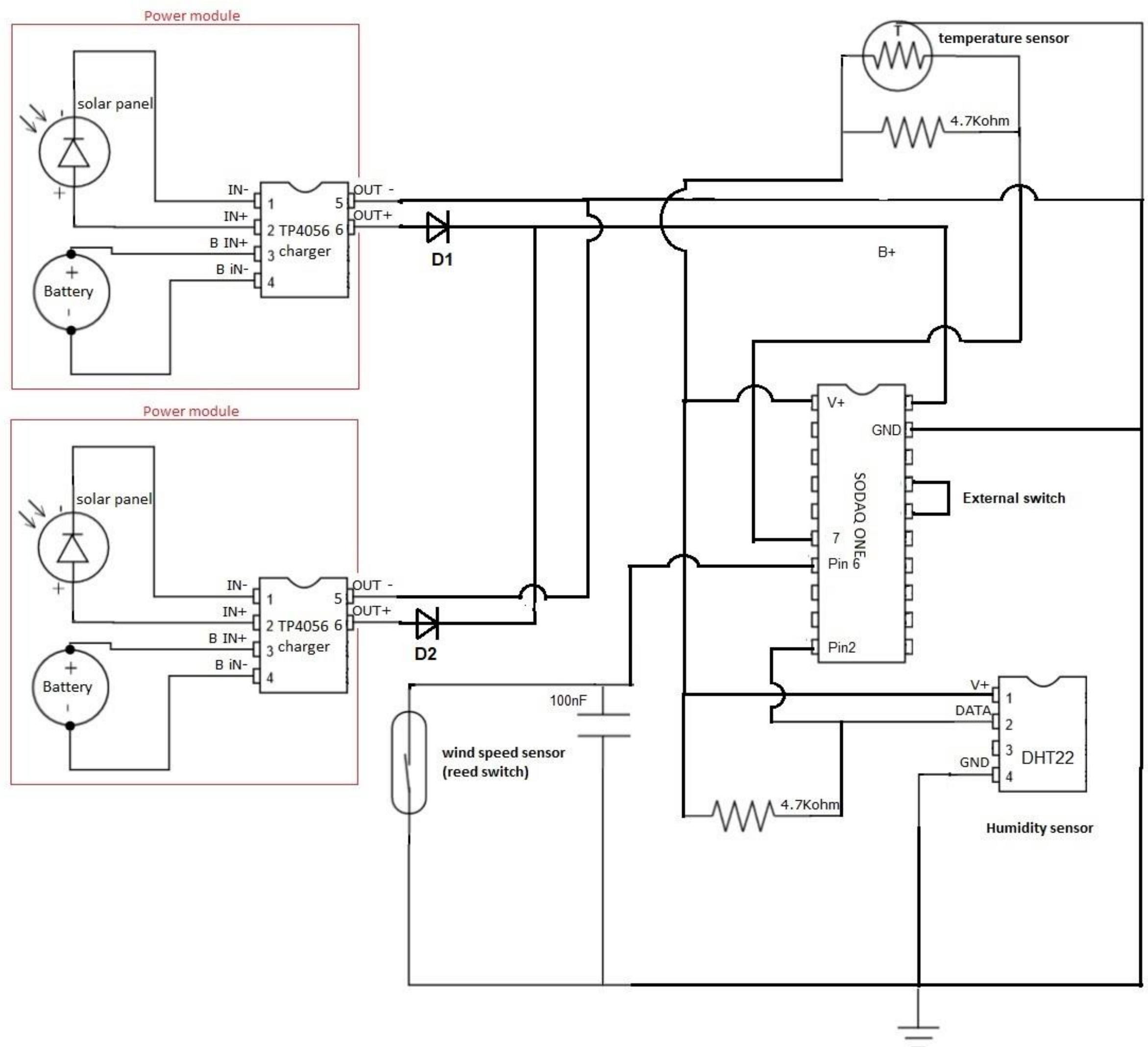


Figure 13: Schematic electrical circuit.

6.1.3 Testing prototype

The casing of the first prototype was tested to see if it is waterproof. The testing procedure is described in section 3.5. From the test it can be concluded that the casing needs some small improvements in order to make it suitable for this sensor system. Not all the part fitted properly in the design, which compromised the waterproofness.

The functionalities of the sensors were also tested. All the sensors seemed to work properly, which means that they can be implemented in the next prototype the way they are implemented in the first prototype. To see if the sensors work properly, the sensor values were sent to a laptop via serial communication. This communication was monitored using the Arduino IDE.

6.2 Second prototype

The second prototype is the improved version of the first prototype. The aim of this prototype is to deploy it and thus perform a field test. All the subsystems will be integrated into the sensor system, which will result in an autonomous functioning sensor system which transmits the data via TTN.

6.2.1 Casing

Some improvements were made to the casing of the first prototype in order to make it waterproof. Not all the parts of the casing fit well in the main casing of the microcontroller. These parts have been altered. This concerns the part that connects the radiation shield with the main casing of the microcontroller. The part that holds the temperature sensor has also been altered. The temperature sensor now fits better in the casing.

6.2.2 Electronics

All the sensors were functional in the first prototype; hence no alterations were made to the circuit of the sensor system. The SODAQ One microcontroller is fitted in the main casing of the sensor system by means of rails. The microcontroller is glued on a 3D-printed piece of plastic, which slides in a slot in the main casing. This way the microcontroller is firmly attached to the casing, but it can be removed with relative ease. Both the GPS antenna and the LoRa antenna are glued to this piece of plastic.

6.2.3 Testing

The second prototype (figure 2) is tested on its functionalities regarding the communication with the back-end service. The prototype was fully assembled and deployed outside for a whole weekend. Because the connection between the back-end service and the database was not fully functional, the data was retrieved using TTN dashboard.

Figure 14: Deployment of second prototype.

Chapter 7 Evaluation

This chapter describes the different evaluations of the sensor system. The sensor system will be evaluated on three different aspects. The performance of the three sensors of the system will be evaluated. Moreover, the *must have* functional requirements will be evaluated. Lastly, the sensor system will be evaluated together with the stakeholders in order to see if their expectations of the sensor system are met.

7.1 Functional testing

This section reports on the test setup that is used to test the sensor's performance. Also, the results of the evaluation concerning the *must have* functional requirements are presented.

7.1.1 Test setup

In order to test the sensors' accuracy, the sensor system needs to be deployed next to a reference sensor system. The data yielded from this reference sensor system can be regarded as the true values for the respective variables. By comparing a dataset yielded from this reference sensor system with the data yielded from this project's sensor system, it can be statistically proven that there is no significant difference between two corresponding (paired) samples from the datasets (i.e. no significant difference between X_i and Y_i). If this is the case, the sensor system operates just as well as the reference sensor system.

Figure 15: Test setup with Alecto WS-4800.

In order to yield a baseline dataset an Alecto WS-4800 weather station is used

[34] (see figure 15). This weather station will be deployed next to the sensor system. For a period of four hours, measurements will be taken of the air temperature, the relative humidity and the windspeed. These measurements will be taken with an interval of five minutes. This means that the dataset will include 49 different samples for each of the variables. The sensor system's measurements can be monitored using the back-end service. By browsing to the database' IP-address, the samples can be retrieved. The Alecto weather station does not feature such a database. This weather station comes with a display that is remotely connected to the weather station. This display can be used to retrieve the samples. The send rate of the sensor system needs to be changed to 5 minutes in order to facilitate the required sample frequency. Each time a measurement is taken, the corresponding values will be logged in an Excel sheet for further processing.

The measurements will be taken early in the morning. This results in a dataset with more variety in values. This variety is the result of the sun coming up. The upcoming sun leads to more rapid changes in temperature and humidity. The windspeed is not dependent on the sunrise. Taking measurements over a larger range of values will yield results that better show the correlation between the reference values and the sensor system's values.

7.1.2 Evaluation of functional requirements

This subsection reports on the evaluation of the functional requirements. The various requirements can be found in table 4.

The system must	Met?	Notes
Be able to measure the relative humidity	Yes	With 3% accuracy
Be able to measure the air temperature	Yes	With 0.5°C accuracy
Be able to measure the windspeed	Yes	Not with the required accuracy
Be able to derive its location	Yes	With accuracy of 20 meters
Be able to send a payload every 30 minutes	Yes	
Be able to label the data with a quality mark	Yes	

Be able to send the payload within a 1% duty cycle	Yes	
Be able to send payload with a size of 21 bytes	Yes	
The system could		
Be able to enter a diagnostic mode	No	

Table 4: Functional requirements.

The system's ability to derive its location with an accuracy of 20 meters was tested using a mobile phone. The location of the system was derived using its GPS-module. The software of the sensor system was altered in such a way that the location was not send to the back-end service via TTN. Instead, serial communication between the sensor system and a computer was used to return the location. On three occasions the location of the sensor system was measured and compared to the location of the mobile phone. On all three occasions the location was within 20 meters of the mobile phone's location. The measurements were taken outside. This is because the GPS-module does not function properly indoors. The GPS-module requires a clear view of the sky in order to utilize the GPS-satellites.

7.1.3 Evaluation of non-functional requirements

This subsection reports on the evaluation of the non-functional requirements. Due to the limited amount of time for graduation project, most of the non-functional *must have* requirements were not evaluated. This does not mean that they are not met, but it cannot be concluded that they are without more data. The requirements can be found in table 5.

The system must	**Met?**	**Notes**
Be attachable	Yes	
Be available 90% of the day	Yes	45 of 48 payloads received in one day (93.75%).
Be energy autonomous	Yes	
Be able to operate in temperatures of -25°C minimum	Unknown	Not tested

Be able to operate in temperatures of 50°C maximum	Unknown	Not tested
Be able to operate in a humidity of 0% minimum	Unknown	Not tested
Be able to operate in a humidity of 99% maximum	Unknown	Not tested
The system should		
Be able to withstand precipitation of 100mm/hour	Unknown	Not tested
Be able to withstand wind gusts of 12 Bf.	Unknown	Not tested
Have a high albedo factor (i.e. light colour)	Yes	Casing is white.
Have a durable casing of ABS or PETG	No	PLA casing

Table 5: Non-functional requirements

The system is energy autonomous if it can provide its own power. During the system's functionality testing (section 6.2.3), the battery level was monitored over the span of an entire weekend. During the weekend the battery's voltage never dropped below 4.0V and never exceeded 4.1V (100% charged). The test conditions were rather well, given the fact that sun was shining relatively a lot. Nevertheless, the system does not seem to consume more power than the solar panels can generate.

7.2 Stakeholder evaluation

In this section the evaluation of the sensor system with the stakeholders is presented. The survey used to evaluate the sensor system can be found in Appendix C. Unfortunately, Rik Meijer (see chapter four) was not available to evaluate. Hence his colleague Hendrik-Jan Teekens attended the evaluation to evaluate the sensor system on behalf of the municipality of Enschede. The evaluation was built around the preliminary requirements derived during the ideation phase. The survey the stakeholders were given consisted of all these requirements and offered room for comments. The stakeholders were talked through the design of the new sensor system and how the preliminary requirements were implemented in the new design. This way the system was evaluated in terms of what the stakeholders expected from it.

7.2.1 Wim Timmermans

Wim Timmermans was (among others) interviewed in order to derive the set of preliminary requirements. This meant that he already knew a little bit about the sensor system. This is why the changes made to the design that resulted in the new sensor system were presented to Wim first. After this all the preliminary requirements were explained and Wim gave his opinion about the way these preliminary requirements were implemented.

The first thing Wim noted was the fact that the temperature sensor overestimated the reference value before 08:00 o'clock (figure 16). After 08:00 o'clock the temperature sensor underestimated the reference value. In case of the humidity sensor, this is quite the opposite. The humidity sensor underestimates the reference value before 08:00 o'clock and overestimates the reference humidity after 08:00 o'clock (figure 17). He would have like a more elaborated calibration and validation of the sensors. He liked the fact that the sensor system only checked its location once per day. Wim mentioned that the trend of the windspeed seemed to match the reference windspeed visually (figure 18). He mentioned that there can be a lot of fluctuation in windspeed on different levels (1km to 1 cm), this might be the reason why there is a significant difference between the measurements of the sensor system and the reference system. He advises to further evaluate the windspeed sensor in order to better determine its accuracy.

Normally speaking, climate related models work with data using one value per half hour. This one value is the mean taken over the entire span of these 30 minutes. In case of this sensor system, the mean is only taken over a dozen samples right before the payload is send. This means that the value is not the mean taken over 30 minutes. This is because the sensor system will enter a sleep mode between sending payloads. He is curious how the mean taken by the sensor system relates to the mean taken over the entire 30 minutes. He notes that it really depends on the time of day/year the measurements are taken.

Wim mentions that in order to verify that the sensor system is able to be a operate without scheduled maintenance, an expert check of the system should be performed. Wim is pleased with the quality check of the data using the battery level.

Lastly, Wim adds that at this point in time, the calibration and performance of the sensors should be adequately tested using a professional reference sensor over a wide range of values.

This includes taking measurements during the winter for example. This way, the sensor system can be evaluated to such a degree that it can be trusted to contribute to research.

7.2.2 Hendrik-Jan Teekens

Given the fact that Hendrik-Jan Teekens was a substitution for Rik Meijer, Hendrik-Jan was not well informed about this graduation project. All the preliminary requirements were therefore explained to him, how these preliminary requirements came to be and what the rationale behind them was.

He had very little comments about the way most preliminary requirements were implemented. He liked the way the new sensor system looked and he was pleased with the fact that the sensor system featured two solar panels. He liked the fact that both solar panels were outfitted with their own charging circuitry and batteries, this way extra redundancy was added to the sensor system. Hendrik-Jan had some questions about the availability percentage. He was curious how the number 90% came to be, but after explaining it he agreed that it was sufficient.

Lastly, he added that the system might benefit from some precaution measurements in case someone wants to steal it. Hendrik-Jan suggested to put some decals on the sensor system that warn people about the GPS-module inside the sensor system. This way he thought people were less prone to steal the sensor system, because they knew that the sensor system’s location is tracked.

7.3 Analysis

This section covers the analyses of the sensors. An extended version of the statistical analysis can be found in Appendix D. The tables containing the yielded data can also be found in this appendix. The analysis is performed using the data yielded from the test setup as described in section 7.1.

7.3.1 Temperature

The DS18B20 temperature sensor was found to suffice for the application of this sensor system. From the statistical analysis found in Appendix D, it can be concluded (95% confidence level) that there is no significant difference between the measurements of the sensor system and the reference system. This means that the observed differences between the sensor values of both systems is due to statistical variation. Taking the required accuracy of +-0.5°C in mind, 87.8% of the measurements differ no more than 0.5°C from the reference values. This means that 87.8% of the measurements are accurate. Figure 16 compares the measured temperature with the reference temperature. One thing that stands out is the fact that the sensor system overestimates the reference temperature before 08:00 o'clock and underestimates the reference temperature after 08:00 o'clock. The sensor system is located near a wall (see figure 15) and this might be the reason behind this behaviour. The wall buffers the heat from the previous day and cools off very little during the night. The heat emitting from the wall might have influenced the temperature sensor.

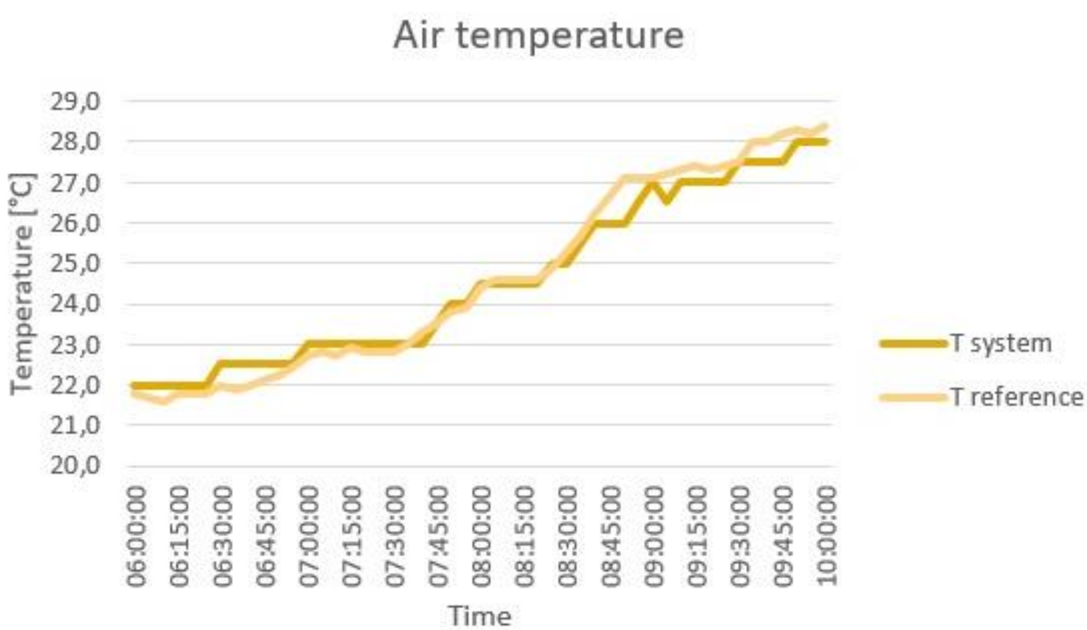


Figure 16: Plotting the reference and measured temperature.

7.3.2 Humidity

The humidity sensor used in this graduation project (DHT22) is deemed sufficient in terms of this sensor system's application. Using the same method as used for the temperature sensor, it can be concluded (95% confidence level) that there is no significant difference between the measurements of the sensor system and the reference system. The required accuracy of this sensor system is 3% for the humidity sensors. Not one observed measurement differed more than 3% from the reference value. This means that all the measured values are accurate. The values of both the measured humidity and reference humidity can be found in figure 17. Similar to the temperature measurements, the humidity sensor seems to have to tendency to underestimate the reference humidity before 08:00 and overestimate the reference humidity after 08:00 o'clock. The reason behind this might well be the same reason behind the temperature difference. The wall near the sensor system might emanate enough heat to influence the measurements.

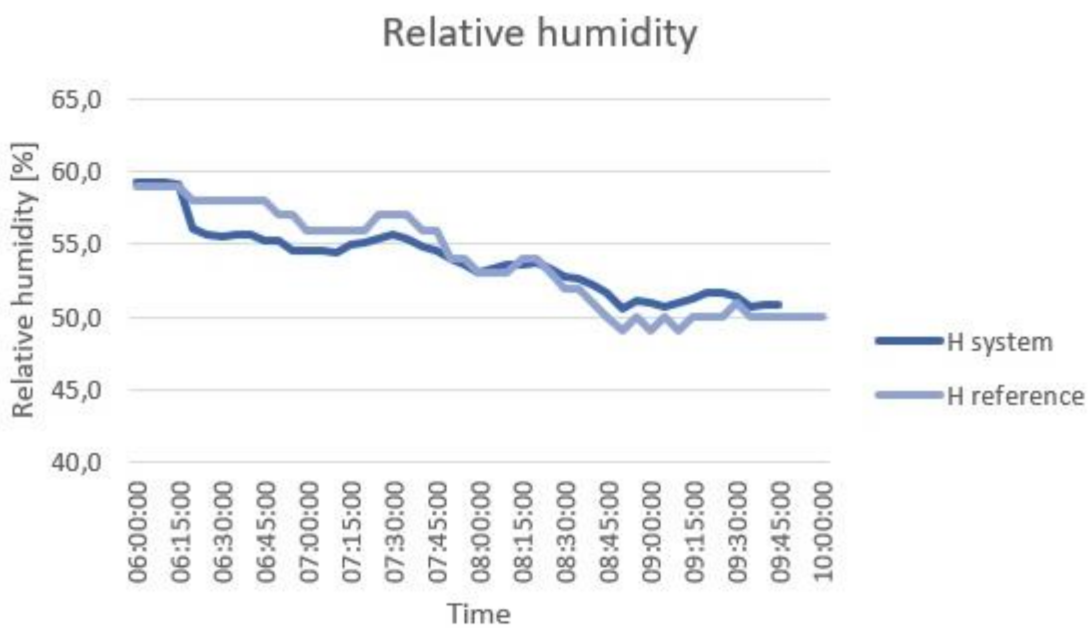


Figure 17: Plotting the reference and measured humidity.

7.3.3 Windspeed

The Froggit WH1080 anemometer does not suffice for the application of this sensor system. From the statistical analysis (Appendix D) can be concluded that the measurements from the sensor system differ significantly from the measurements from the reference system (figure 18) (95% confidence level). This means that the difference in measurements is not due to statistical variation. This explains why only 6% of the measurements are within the accuracy range of 1m/s. This difference might be explained by the fact that the sensor system is located near a wall, while the reference system is located on the roof. The wall might influence direction and speed of the wind.

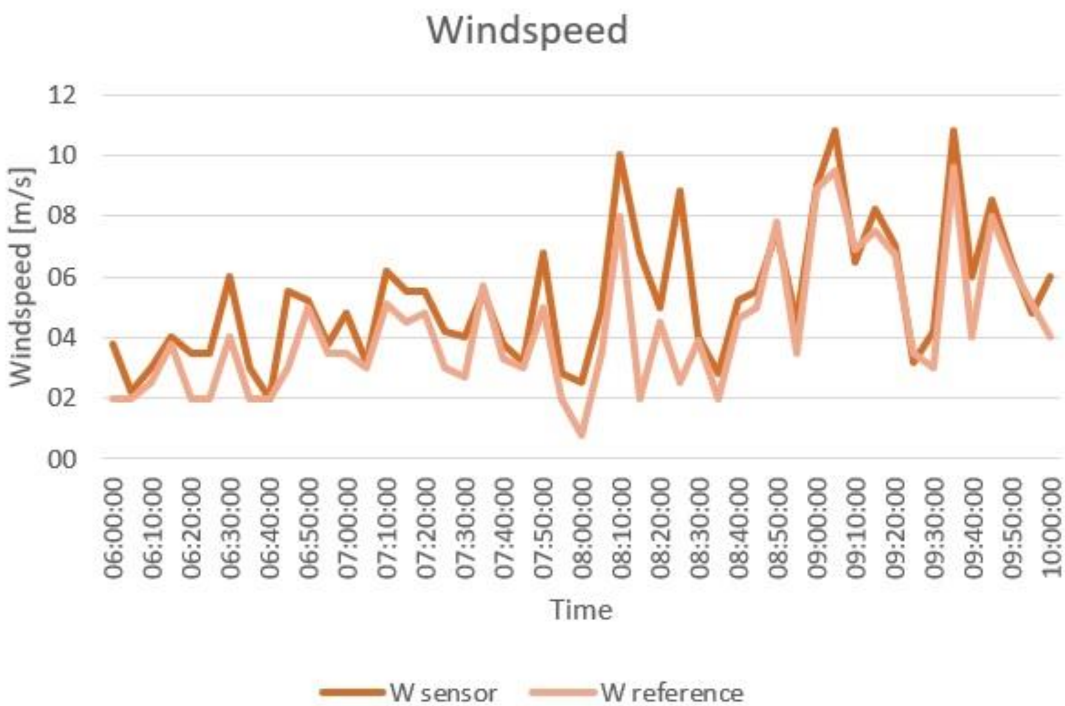


Figure 18: Plotting the reference and measured windspeed.

7.4 Conclusion

The overall evaluation of the sensor system was a success. All of the *must have* functional requirements were met, albeit that the wind sensor did not measure the windspeed with the required accuracy. None of the *must have* non-functional requirements were not met, although not all requirements were met. The requirements that specify the range of temperature and humidity in which the system needs to operate were not evaluated extensively. Testing this would be too time-consuming. Further research needs to point out if these requirements are met.

Overall the stakeholders were happy with the result of this graduation project. In general, all the preliminary requirements were met. Wim had some comments on a few of the implementations of the preliminary requirements. These comments were valid, but often too time consuming to implement in this graduation project. Most of these comments relate to the fact that more calibration and verification is needed.

All the sensors were compared to a reference sensor in order to find out how well the sensors function. In case of the temperature sensor and the humidity sensor, the sensors work fine and there was no significant difference to be found between the sensors system and the reference system. All of the humidity measurements fell within the specified accuracy range and almost 90% of the temperature measurements fell within the specified accuracy range. However, there was a significant difference between the windspeed measurements of the sensor system and the reference system. Only 6% of the windspeed measurements fell within the required accuracy range. This means that the windspeed sensor does not function as it should. However, the co-location of the sensor system and the reference system was not optimal. In this way, the different location of the two systems might have influenced the tests to such a degree that the results are influenced by the co-location.

Chapter 8 Conclusion

This chapter provides the answers to the research questions formulated in chapter one. The resulting sensor system of this graduation project will be linked to the research questions in such a way the newly implemented design choices (partly) answer these research questions. Furthermore, a section of this chapter is dedicated to the discussion of this chapter. In this section findings are presented that are out of this graduation project's scope or influenced the design process in an unforeseen way.

8.1 Results

The main research question of this graduation project is: "How to develop a dependable autonomous sensor system to measure the temperature build-up in the city of Enschede?" In order to answer this question, the research question is divided in three sub questions:

- ❖ What does dependability entail in terms of data gathered by the sensor system?
- ❖ What does dependability entail in terms of the sensor system itself?
- ❖ What does dependability entail in terms of the system's communication with the back-end service?

In order to answer these questions, the concept of dependability was further examined. Dependability can be seen as an integrating concept of several attributes. Maximizing the influence of all these attributes on the sensor system's dependability would have been too time consuming for a graduation project such as this one. Hence the focus of maximizing the dependability was shifted to two concepts of dependability, namely the energy-awareness of the sensor system and the availability of the sensor system. This means that dependability in the context of this graduation project means that a system is both energy-aware and available to some degree.

The answer to the first sub question is that the energy-awareness and the availability of the system have no direct impact on the dependability of the data gathered by the system. These two concepts of dependability relate more to the sensor system itself. However, if the sensor system has a relative low availability percentage, then the data gathered by the sensor system contains a lot of gaps in the dataset. This data is only useful if these gaps are non-existent or very small. This means that the sensor system should be have a high availability percentage in order for its data to be dependable. In this sense it can be argued, that the data gathered by the sensor system is only dependable if the sensor system is available for a certain percentage per day. In case of this sensor system, a minimal availability percentage of 90% per day is required. By labelling the data using the battery level, the dependability of the data is enhanced in terms of its energy-awareness. The system monitors its battery level and uses this to label the quality of the data. The system is aware of its energy-reserves and uses this information to tell something about the dependability of the data by giving it a quality label.

The answer to the second sub question is that the availability of a system plays a vital role in the system's dependability. A system such as this sensor system needs to be available a certain percentage of the time in order to be regarded as dependable. In case of this sensor system, an availability of at least 90% per day is required. In this graduation project, the sensor system meets this availability requirement if it sends at least 90% of the payloads per day. This is implemented in the sensor system by minimizing some processes in the software of the sensor system and by adding redundancy in the power systems. Minimizing some of the processes in software will decrease the chance that the system will become unavailable due to an error in these processes. Such a process can be for example the deriving of the sensor system's location and the send-rate of the payloads. If the sensor system requests its location more often than necessary, it will not add anything to the functionalities of the system. However, the chance that the sensor system becomes unavailable increases.
A certain degree of availability is also achieved by this system by incorporating two individual power supplies. The sensor system features two solar panels and each solar panel possesses its own charging circuit and battery. These power supplies are placed in parallel, so they both contribute to the system's power consumption. If one of these power supplies fails, the other is still capable of providing the system with enough power. This way the system can still operate is one of its power supplies is broken, without compromising its availability.
The energy-awareness of the sensor system is implemented in such a way that the sensor system labels the quality of the data using the battery level and the system sends the battery level along to the back-end service. The labels can be used to see if the sensor is still functioning properly. If the sensors give measurements that are out of context, while the battery level is still sufficient, something might be wrong with the sensors. In this way it can be argued that the system becomes more dependable by providing the information about the battery. The

sensor system is not energy-aware in the sense that it alters its behaviour based on its supply of energy.

The answer to the third sub question is that the energy-awareness and the availability of a sensor system have little impact on the communication of the sensor system with the back-end service, other than the trivial impact (i.e. if the sensor system is not available or has no battery, it cannot communicate with the back-end service). The choice of network (LoRaWAN) does relate to the energy-awareness of the sensor system. Given the fact that LoRaWAN networks (such as TTN) offer a large communication range using little transmission power, makes such networks very suitable for an application such as this sensor system. The system is not aware of the amount of energy it uses to communicate with the back-end service. So, by choosing an appropriate network, the energy consumption with regard to the communication can be minimalized.
When it comes to the communication of the sensor system, the system's availability rate is not the only part that influences the communication of the sensor system. It might well be the case that the sensor system is available 100% of the time and thus sends all the payloads. However, if the gateway is not available at that moment, the payloads will not be received and stored in the database. Therefore, in order to say something about the dependability of the sensor system's communication in terms of availability, the other parts of the network should be examined as well. If the percentage of sent payloads is adequate, the system's communication dependability is satisfactory in terms of its availability. This only holds from the sensor system's point of view. In order to say something about the whole communication of the sensor system, more aspects of the network should be looked into first (i.e. gateways etc.).

These answers to the different sub questions can be used to formulate an answer to the main research question. A dependable autonomous sensor system can be developed by implementing measures to maximize its availability and energy-awareness. The sensor system needs to be available a certain percentage of the day depending on its application. In case of this sensor system, an availability rate of 90% is sufficient in order to be dependable. In order to ensure this availability rate, the sensor system is outfitted with two separate power supplies. This way, the system can continue to operate when one power supply fails. Moreover, the system will draw approximately half of the required current from each battery. So, the batteries will last longer in times of bad conditions (i.e. little exposure of the solar panels to sunlight).

By incorporating a quality mark for each of the measured variables, the data can be checked to see if it is dependable. This quality mark is based on the battery level of the sensor system. Such a quality label can be used in a two-way fashion. On one side, the quality label can be used to assess the quality of the data. If the battery levels are too low, the returned measurement of the respective sensor might be incorrect. The user of the data is informed of this fact by the quality label. On the other hand, if the battery level is sufficient and the sensor system returns measurements that are too high or low for the context, something might be wrong with one of the subsystems of the sensor system. This way, the administrator of the database can act on either one of the situations. It can be argued that this enhances the dependability of the sensor system in the sense that warns for possible shortcomings of the sensor system. So, the dependability itself is not enhanced by the quality marks, they function as a mere indicator that the dependability of the sensor system is compromised.
The temperature build-up will be monitored by measuring the windspeed, the temperature of the air and the humidity. All these variables are measured with an accuracy and resolution which suffices for the application

8.2 Discussion

In this section, some general items of discussion will be presented. First of all, the functionalities of the sensors were not tested extensively. Samples were only taken for a relative short amount of time and the location of the sensor system and the reference system was not ideal. A wall located near the sensor system has likely influenced the measurements of all three variables. In case of the air temperature and the humidity, the influence of the wall was noticeable but not enough to influence the statistical test. In case of the windspeed, the wall might have influenced the measurements to such a degree that the statistical conclusion is wrong. Better validation of the sensors is required to fully determine their accuracy.
The sensors should also be tested over a wider range of values. Since the sensor system will be deployed throughout the year, the sensor system needs to handle a wide range of temperatures, humidity values and windspeeds. The sensors were only tested over a limited range. For example; the temperature sensor only measured temperatures between 22°C and 28°C. During the winter, the temperature sensors needs measure temperatures well below 22°C. The sensor system should be tested to see how it performs under these conditions. This is the reason why more extensive validation of the sensors is required.

Another point of discussion is the fact that the hardware and software functionalities should be tested more extensively to draw any conclusions. The sensor system has only been running for one week. In order to assess the dependability of the sensor system, it should be tested for a longer period of time and under a variety of circumstances. In the ideal scenario, the sensor system should be tested for an entire year to assess its dependability. From such a test, more definitive conclusions can be drawn in terms of the sensor system's dependability.

Lastly, the 3D printed parts did not fit very well. Even though they were designed to fit, not all of the parts did due to the fact that the 3D printer can only print up to a certain resolution. A lot of sanding and polishing was needed in order to make the parts fit. In order to make the casing of the sensor system look more like one whole, the different parts should have been altered slightly to compensate for the fact that they do not fit well together

Chapter 9 Future Work

This chapter will provide recommended developments on the previous work.

9.1 Sensor system

One of requirements was that the casing should be made of a more durable plastic then PLA. Such plastics are for example PETG and ABS. Ultimately, the casing was still made from PLA. Future sensor systems would benefit from a casing made out of ABS or PETG. These plastics are more resistant to temperature changes. Additionally, these plastics are affected less by UV-radiation.

Moreover, some parts of circuitry can be optimized in terms of how they are connected to each other. By making the sensor system modular, it is relatively easy to repair them. The different sensors are connected to the microcontroller via cables with connectors. This is also the case with the solar panels. However, it is not possible to disconnect one subsystem (i.e. solar panel or sensor) at the time. For example, it is not possible to disconnect one single sensor and replace it.

Some parts of the casing could benefit from a redesign when it comes to modular design. A lot of the parts of the casing are glued together. This is mainly because of the need for waterproofness. Future designs would benefit from a modular design which does not compromise its waterproofness.

9.2 Data

The data gathered by the system can be optimized in terms of its reliability. One method to do so is to take the mean of a range of measurements taken throughout the interval between payloads. Additionally, the quality of the data can be influenced by measurements that are not accurate. The data would benefit from some form of pre-processing where bad measurements are recognized by the sensor system.

The current labelling system of the sensor quality labels is rather straightforward. The way the sensor system determines the quality of the data is an item which can be improved. The only factor that decides the quality of the label currently is the voltage of the battery. More research should be done into the factors within the system that influence the quality of the data.

9.3 Communication

In the current system design, the information flow is rather one-sided. The sensor system is only able to send data to the gateway of The Things Network. No information can be received by the sensor system. Future designs would benefit from a two-way communication. Such a two-way communication would allow for confirmation messages to the sensor system. Moreover, over-the-air software updates would be possible. This way the sensor system's firmware can be updated without taking picking up the sensor systems.

Another issue that should addressed is the fact that achieving higher dependability is not restricted to the sensor system itself. The process of gathering data only ends when the data is stored in a database. This means that in order to gather data on the effects of UHIs in a dependable manner, the dependability throughout the network should be optimized. This entails all the different subsystems that can be found in figure 10. This graduation project only focussed on the dependability of the sensor system, so there is room for more research on dependability.

Reference

[1] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, “Basic concepts and taxonomy of dependable and secure computing,” IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 1, pp. 11–33, Jan. 2004.

[2] K. Ward, S. Lauf, B. Kleinschmit and W. Endlicher, "Heat waves and urban heat islands in Europe: A review of relevant drivers," Science of The Total Environment, pp. 527-539, 1 November 2016. DOI: 10.1016/j.scitotenv.2016.06.119

[3] M. Nuruzzaman, "Urban Heat Island: Causes, Effects and Mitigation," International journal of environmental monitoring and analysis, pp. 67-73, 2015. DOI: 10.11648/j.ijema.20150302.15

[4] A. Mohajerani, J. Bakaric and T. Jeffrey-Bailey, "The urban heat island effect, its causes, and mitigation, with reference to the thermal properties of asphalt concrete," Journal of Environmental Management, pp. 522-538, 15 July 2017. DOI: 10.1016/j.jenvman.2017.03.095

[5] Y. Jiachuan, W. Zhi-Hua and E. Kamil, "Environmental impacts of reflective materials: Is high albedo a ‘silver bullet’ for mitigating urban heat island?," Renewable and Sustainable Energy Reviews, vol. 47, pp. 830-843, July 2015. DOI: 10.1016/j.rser.2015.03.092

[6] P. P. Hui Li, "Literature Review on Cool Pavement Research," Pavement Materials for Heat Island Mitigation, pp. 15-42, 12 February 2016. DOI: 10.1016/B978-0-12-803476-7.00002-7

[7] R. M. Rehan, "Cool city as a sustainable example of heat island management case study of the coolest city in the world," HBRC Journal, vol. 12, no. 2, pp. 191-204, August 2016. DOI: 10.1016/j.hbrcj.2014.10.002

[8] Y. Jabareen, "Planning the resilient city: Concepts and strategies for coping with climate change and environmental risk," Cities, vol. 31, pp. 220-229, April 2013. DOI: 10.1016/j.cities.2012.05.004

[9] K. R. Gunawardena, M. J. Wells and T. Kershawa, "Utilising green and bluespace to mitigate urban heat island intensity," Science of The Total Environment, pp. 1040-1055, 15 April 2017. DOI: 10.1016/j.scitotenv.2017.01.158

[10] Y. Qin, "A review on the development of cool pavements to mitigate urban heat island effect," Reviews, Renewable and Sustainable Energy, vol. 52, pp. 445-459, December 2015. DOI: 10.1016/j.rser.2015.07.177

[11] A. J. Arnfield, "Two decades of climate research: A review of turbulence, exchanges of energy and water and the urban heat island," International journal of climatology, vol. 23, pp. 1-26, 2003. DOI: 10.1002/joc.859

[12] T. Onderwater, "Developing a sensor network for real time temperature monitoring in Enschede," Enschede, 2018.

[13] L. Kester, “Climate measurements in public spaces”, Enschede, 2018.

[14] Silveira, Endy & Bonho, Samir, “Temperature Monitoring Through Wireless Sensor Network Using an 802.15.4/802.11 Gateway”. IFAC-PapersOnLine, vol.49. pp.120-125, 2016 DOI: 10.1016/j.ifacol.2016.11.139.

[15] Muhammad Saqib Jamil, Muhammad Atif Jamil, Anam Mazhar, Ahsan Ikram, Abdullah Ahmed, Usman Munawar, “Smart Environment Monitoring System by Employing Wireless Sensor Networks on Vehicles for Pollution Free Smart Cities”, Procedia Engineering, Vol. 107, pp. 480-484, 2015, DOI: 10.1016/j.proeng.2015.06.106.

[16] R. R. Brooks and S. S. Iyengar, "Maximizing multi-sensor system dependability," 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems (Cat. No.96TH8242), Washington, DC, USA, 1996, pp. 1-8. DOI: 10.1109/MFI.1996.568492

[17] J.N. Gray, “Why do Computers Stop and What Can Be Done About It?” Proc. Fifth Symp. Reliability in Distributed Software and Database Systems, pp. 3-12, Jan. 1986.

[18] Y. Huang and C. Kintala, “Software Fault Tolerance in the Application Layer,” Software Fault Tolerance, M. Lyu, ed., pp. 231-248, 1995.

[19] A. Mader and W. Eggink, "A design process for creative technology," in International Conference on Engineering and Product Design Education, Enschede, 2014.

[20] H. Sharp, A. Finkelstein and G. Galal, "Stakeholder Identification in the Requirements Engineering Process," in Proceedings of 10th International Workshop on Database & Expert Systems Applications, 1999.

[21] D. Reidsma, H. Koppelman, R. v. Delden and M. Bruijnes, "Intelligent interaction design," University of Twente, Enschede, 2017.

[22] KNMI, "klimatologie", ministerie van infrastructuur en waterstaat, 19th of May 2019. [Online]. Available: http://projects.knmi.nl/klimatologie/. [Accessed May 19th, 2019].

[23] LoRa Alliance, “About LoRaWAN”, May 2019. [Online]. Available: https://lora-alliance.org/about-lorawan/. [Accessed May 7th, 2019]

[24] The Things Network, “Building a global open LoRaWAN™ network”, May 2019. [Online]. Available: http://www.thethingsnetwork.org/. [Accessed May 7th, 2019].

[25] Stichting Climate Adaptation Services, “Klimaateffectatlas”, May 2019. [Online]. Available: http://www.klimaateffectatlas.nl/ [Accessed May 23rd, 2019].

[26] Dallas semiconductor, "DS18B20 programmable resolution digital thermometer," 01 01 2015. [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Temp/DS18B20.pdf. [Accessed May 23rd, 2019].

[27] SparkFun electronics Inc., "SparkFun weather meters," 2015. [Online]. Available: https://cdn.sparkfun.com/assets/8/4/c/d/6/Weather_Sensor_Assembly_Updated.pdf. [Accessed May 23rd, 2019].

[28] Adafruit industries, "Adafruit humidity and temperature sensor breakout-DHT22," 01 January 2018. [Online]. Available: https://www.adafruit.com/product/385. [Accessed May 23rd, 2019].

[29] M. Petrova, J. Riihijarvi, P. Mahonen and S. Labella, "Performance study of IEEE 802.15.4 using measurements and simulations," IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006., Las Vegas, NV, 2006, pp. 487-492. DOI: 10.1109/WCNC.2006.1683512

[30] A. L. Mendelow, "Environmental Scanning--The Impact of the Stakeholder Concept," International Conference on Information Systems, pp. 407-417, 1981.

[31] D. Haughey, "MoSCoW Method," 2014. [Online]. Available: https://www.projectsmart.co.uk/moscow-method.php. [Accessed June 14th, 2019].

[32] The Enclosure Company LTD., “IP rated enclosure explained,” 2019. [Online]. Available: http://www.enclosurecompany.com/ip-ratings-explained.php. [Accessed June 16th, 2019].

[33] SODAQ, “SODAQ ONE-EU-RN2483-V3 including a GPS Antenna and a Molex antenna,” 2019. [Online]. Available: https://shop.sodaq.com/sodaq-one-eu-rn2483-v3.html. [Accessed June 16th, 2019].

[34] Alecto, “Alecto WS-4800 Professional weerstation,” 2019. [Online]. Available: https://www.alecto.nl/weerstation-4800 [Accessed June 24th, 2019].

[35] D. Meijer, *Statistical Techniques For CreaTe*. 2016, p. 3.13-3.16

Appendix

Appendix A: Interviews

Interview Wim Timmermans

1. Welke attributen van dependability zijn voor jou belangrijk?

2. Welke voorwaardes stel je aan die attributen?

3. Voldoen de specificities van de sensors nog die op dit moment gebruikt worden?

4. In hoeverre is de precieze locatie van belang?

5. Hoe ga je de data gebruiken om je modellen te valideren?

Interview Hans Scholten

1. Wat is een gebruikelijke availability rate voor dit soort sensor systemin?

2. In hoeverre is het gangbaar dat je in dit soort systemin geplande downtime hebt voor onderhoud?

3. In hoeverre is het van belang de gebruiker te laten weten wat de batterij-status is met betrekking tot het voltage?

4. In hoeverre is het van belang de gebruiker te laten weten wanneer de batterij spanning te laag is?

5. Als de batterij leeg is, moet hij dan eerst helemaal opladen worden alvorens het sensor systeem weer gaat draaien of kan het systeem weer opstarten zodra er genoeg spanning over de batterij staat?

Interview Rik Meijer

1. Wat is het precieze doel van de gemeente Enschede met betrekking tot dit project?

2. Wat betekent dit dan voor de availability van dit sensor systeem?

3. Wat betekent dit voor de energy-awareness van het sensor systeem?

4. Wat is jouw ervaring met vandalisme end dit soort vergelijkbare projecten?

5. Wat zijn eventuele maatregelen die de gemeente neemt tegen vandalisme?

Appendix B: Requirements

Functional Requirements:

Requirement #1	Requirement type: functional
Value: measure humidity	**Attribute:** humidity sensor (hygrometer)
Description: The system can measure the humidity	
Rationale: One of the variables that this sensor system aims to measure is the humidity.	
Source: Interview with Wim Timmermans and environmental factors.	
Criteria: The sensor system should measure the humidity with a 3% accuracy. The range spans from 15% relative humidity to 99% relative humidity.	
Priority: Must have	Conflicts: none

Requirement #2	Requirement type: functional
Value: measure air temperature	**Attribute:** temperature sensor (thermometer)
Description: The system can measure the temperature of the air.	
Rationale: One of the variables that this sensor system aims to measure is the temperature of the air.	
Source: Interview with Wim Timmermans and environmental factors.	
Criteria: The sensor system should measure the air temperature with a 0.5° accuracy. The range spans from -25° to 50°.	
Priority: Must have	Conflicts: None

Requirement #3	Requirement type: functional
Value: measure windspeed	**Attribute:** windspeed sensor (anemometer)
Description: The system can measure the windspeed.	
Rationale: One of the variables that this sensor system aims to measure is the windspeed.	
Source: Interview with Wim Timmermans and environmental factors.	

Criteria: The sensor system should measure the windspeed with an accuracy of 0.1m/s. The range spans from 0 to 10 Beaufort peak windspeed.	
Priority: Must have	Conflicts: None

Requirement #4	Requirement type: functional
Value: location	**Attribute:** GPS module
Description: The system can retrieve its location.	
Rationale: For the data to be useful, the location of the sensor system is needed.	
Source: Interview with Wim Timmermans	
Criteria: The sensor system should retrieve its location with an accuracy of 20 meters.	
Priority: Must have	**Conflicts:** Using the GPS module consumes a lot of power; hence this requirement might conflict with some of the energy-awareness related requirements.

Requirement #6	Requirement type: functional
Value: send rate	**Attribute:** LoRaWAN module
Description: The system must send data packages every half hour	
Rationale: Because climate-related models use time intervals of half an hour, the data should be send every half hour in order to create a dataset that can be compared with these models.	
Source: Interview with Wim Timmermans	
Criteria: A data package should be send every half hour.	
Priority: Must have	Conflicts: None

Requirement #5	Requirement type: functional
Value: GPS-functionality	**Attribute:** GPS-module
Description: The system can test if the GPS-module is still functional. It does so by comparing its location once every 24 hours to the previous location. If the location is the same, the GPS-module still functions properly.	
Rationale: Given the fact that the GPS-module is used relatively seldomly, the system should know if the GPS-module still functions properly.	
Source: Interview with Hans Scholten.	
Criteria: Not applicable.	
Priority: Must have	**Conflicts:** Not applicable.

Requirement #7	Requirement type: functional
Value: quality warning	Attribute: battery
Description: The system labels the data with a quality label, which depends on the voltage of the battery. The system will give a data quality warning if the battery voltage is below the threshold of the sensors. This threshold is the minimum voltage the sensors require in order to operate. When the battery voltage is below the sensors operating voltage (as described in every sensor's datasheet), the data be given a negative data quality label. This data label consists of 2 bits, hence 4 classifications of data. Every measured variable must be labelled in such a way. Moreover, the battery's voltage must be included in the payload as well. Based on the battery's voltage, the battery must be labelled too with a quality mark consisting of 2 bits.	
Rationale: If the battery voltage is too low, the sensors might not work correctly. This means that the data yielded from these sensors in that situation might not be reliable. By sending the battery voltage too, measurements that were taken while the battery was low can then be selected from the database.	
Source: Interview with Hans Scholten	
Criteria: Not applicable.	
Priority: Must have	**Conflicts:** This requirement might conflict with the other data quality requirement (requirement number 9). A malfunctioning

	ADC (used to measure battery voltage) might lead to incorrect quality labels.

Requirement #8	Requirement type: functional
Value: data package	**Attribute:** TTN network
Description: The system must send all the data packages within a 1% duty cycle	
Rationale: TTN restricts the use of its network to a maximum of 1% of the duty cycle	
Source: TTN boundaries	
Criteria: When sending the packages, the sensor system should not use more than 1% of the duty cycle.	
Priority: Must have	**Conflicts:** This restriction might conflict with the amount of yielded data that needs to be send.

Requirement #9	Requirement type: functional
Value: data package size	**Attribute:** LoRaWAN module
Description: The system must send data packages with a maximum size of 21 bytes.	
Rationale: In order to meet the guidelines, set by TTN, the system should restrict the size of the data packages. By doing so the system can send data at the required time intervals.	
Source: TTN	
Criteria: Data packages should not exceed 21 bytes.	
Priority: Must have	**Conflicts:** Such a restriction can conflict with other data-related requirements.

Requirement #10	Requirement type: functional
Value: diagnostic data	**Attribute:** diagnostic mode
Description: The system can be put in a diagnostic mode to send diagnostic data about its subsystems (i.e. the sensors and the GPS-module).	
Rationale: In order to avoid scheduled/preventive maintenance, the system should be able to send diagnostic data to facilitate so-called just-in-time maintenance. This kind of maintenance is less time-consuming, since maintenance is only performed when needed.	

The system is put in a diagnostic mode when it receives a request to do so. It will then send diagnostic data.	
Source: Interview with Hans Scholten.	
Criteria: Not applicable.	
Priority: Could have	**Conflicts:** This requirement conflicts with the availability requirement, because putting the system in a diagnostic mode will prevent it from doing its original task (i.e. sending sensor data).

Non-functional requirements:

Requirement #11	Requirement type: non-functional
Value: attachable	Attribute: mount
Description: The system can be attached to poles and walls.	
Rationale: For the system to be deployed, it needs to be attached to a pole of a wall.	
Source: Ideation	
Criteria: not applicable	
Priority: Must have	**Conflicts:** Not applicable

Requirement #12	Requirement type: non-functional
Value: availability rate	**Attribute:** availability
Description: The system has to be available 90% of the time.	
Rationale: The sensor system should have an availability rate of 90% in order to work properly.	
Source: Interview with Hans Scholten	
Criteria: Availability rate of 90% per day. This means that 43 out of 48 data packages should be send every day if data is send on a half hour base in order to meet this criterium.	
Priority: Must have	Conflicts: None.

Requirement #13	Requirement type: non-functional
Value: power supply	**Attribute:** energy-awareness
Description: The system is energy autonomous.	
Rationale: The system needs to provide its own power in order for it to be able to operate independently. This means that power should be generated from environmental factors such as solar energy and wind energy.	
Source: Interview with Hans Scholten	
Criteria: The system should be able to generate enough energy for it to operate for a period of one year.	

Priority: Must have	**Conflicts:** This requirement might conflict with functionalities of the system, since these functionalities require a certain amount of power.

Requirement #14	Requirement type: non-functional
Value: temperature tolerance	**Attribute:** sensor system
Description: The system can operate in temperatures of -25°C.	
Rationale: Temperatures in Enschede can be as low as -10°C. If the system can operate in temperatures of -25°C, it is able to function properly throughout the year (given the fact that requirement 15 is also considered).	
Source: Environmental factors	
Criteria: The system should be able to function properly for a period of 24 hours in temperatures of -25°C.	
Priority: Must have	**Conflicts:** Not applicable.

Requirement #15	Requirement type: non-functional
Value: temperature tolerance	**Attribute:** sensor system
Description: The system can operate in temperatures of 50°C.	
Rationale: Temperatures in Enschede can be as high as 36°C. If the system can operate in temperatures of 50°C, it is able to function properly throughout the year (given the fact that requirement 14 is also considered).	
Source: Environmental factors.	
Criteria: The system should be able to function properly for a period of 24 hours in temperatures of 50°C.	
Priority: Must have	**Conflicts:** Not applicable.

Requirement #16	Requirement type: non-functional
Value: humidity tolerance	**Attribute:** sensor system
Description: The system can operate in a relative humidity of 0%.	

Rationale: The relative humidity in Enschede can be as low as 15%. If the system can operate in relative humidity of 0%, it is able to function properly throughout the year (given that requirement 17 is also considered).	
Source: Environmental factors.	
Criteria: The system should be able to function properly for a period of 24 hours in a relative humidity of 0%.	
Priority: Must have	**Conflicts:** Not applicable.

Requirement #17	Requirement type: non-functional
Value: humidity tolerance	**Attribute:** sensor system
Description: The system can operate in a relative humidity of 99%.	
Rationale: The relative humidity in Enschede can be as high as 99%. If the system can operate in relative humidity of 99%, it is able to function properly throughout the year (given that requirement 16 is also considered).	
Source: Environmental factors.	
Criteria: The system should be able to function properly for a period of 24 hours in a relative humidity of 99%	
Priority: Must have	**Conflicts:** Not applicable.

Requirement #18	Requirement type: non-functional
Value: waterproof	Attribute: casing
Description: The system can withstand downpour of 100mm/hour.	
Rationale: In Enschede precipitation can reach levels of 100mm/hour. The system needs to be waterproof for it to operate, because the sensors and electronics do not work when they have been in contact with water. Hence the system should be able to withstand downpour of 100mm/hour, by means of a waterproof case and seals.	
Source: Environmental factors.	
Criteria: The inside of the casing remains dry when the sensor system is exposed to water.	
Priority: Should have	**Conflicts:** Not applicable

Requirement #19	Requirement type: non-functional
Value: wind-proof	Attribute: casing
Description: The system can withstand wind gusts of 120km/h (12 Bf.).	
Rationale: Strong wind can interfere with the system's operations. Given the fact that it is very unlikely that windspeeds in Enschede exceed 120km/h, the system needs to be able to withstand these windspeeds.	
Source: Environmental factors.	
Criteria: The system should be able to function properly for a period of 10 minutes during 12 Bf. wind.	
Priority: Should have	**Conflicts:** not applicable.

Requirement #20	Requirement type: non-functional
Value: albedo	**Attribute:** colour of casing
Description: The system's casing has a high albedo value. The lighter the colour, the higher the albedo value. Hence the system's casing is white.	
Rationale: The system will reflect most of the radiation if it has a high albedo value. This radiation will interfere with the system's functionalities if it is not reflected properly. This interference is caused by heat built up through the radiation.	
Source: Ideation	
Criteria: Colour of the casing is white.	
Priority: Should have	**Conflicts:** Not applicable

Requirement #21	Requirement type: non-functional
Value: durability	Attribute: Casing
Description: The system's casing made of a plastic called ABS (Acrylonitril-butadieen-styreen) or PETG (Polyethylene terephthalate).	
Rationale: ABS and PETG are types of plastic which is more durable compared to other plastics such as PLA. For the system to be deployed for a period of 5 years, its casing needs to durable. ABS and PETG are not biodegradable and better suited for high and low temperatures. Some types of 3D-printers support these two types of plastic, which makes it suitable for this graduation project.	
Source: Ideation	

Criteria: Not applicable	
Priority: Could have	**Conflicts:** Not applicable.

Appendix C Evaluation survey

Evaluation Sensor System with stakeholders

Nr	Requirement	Comments
1	The system should be able to measure air temperature (±0.5°C), the humidity (±3%) and the windspeed (±0.1m/s).	
2	The system should be able to derive its location up to 20 meters accurate.	
3	The system should be able to send data every half hour.	
4	The system should not allow large gaps in data.	
5	The sensor system should be available 90% of the day.	
6	The system should be energy autonomous.	

Nr	Requirement	Comments
7	The system should be able to operate without scheduled maintenance.	
8	The sensor system should be able to test if the various subsystems still function properly.	
9	The user should be made aware that the location and battery level might interfere with the quality of the data.	
10	The sensor system should be able to measure the previously mentioned three variables with a resolution of 1 decimal.	

Appendix D: Data analyses sensors

Sensor system			time	reference system		
temp	windspeed	humidity		temp	windspeed	humidity
22,0	3,8	59,2	06:00:00	21,8	2,0	59,0
22,0	2,2	59,2	06:05:00	21,7	2,0	59,0
22,0	3,0	59,2	06:10:00	21,6	2,5	59,0
22,0	4,0	59,1	06:15:00	21,8	3,8	59,0
22,0	3,5	56,1	06:20:00	21,8	2,0	58,0
22,0	3,5	55,7	06:25:00	21,8	2,0	58,0
22,5	6,0	55,5	06:30:00	22,0	4,0	58,0
22,5	3,0	55,6	06:35:00	21,9	2,0	58,0
22,5	2,0	55,6	06:40:00	22,0	2,0	58,0
22,5	5,5	55,3	06:45:00	22,1	3,0	58,0
22,5	5,2	55,2	06:50:00	22,2	5,0	57,0
22,5	3,8	54,6	06:55:00	22,4	3,5	57,0
23,0	4,8	54,6	07:00:00	22,7	3,5	56,0
23,0	3,2	54,5	07:05:00	22,8	3,0	56,0
23,0	6,2	54,4	07:10:00	22,7	5,1	56,0
23,0	5,5	55,0	07:15:00	22,9	4,5	56,0
23,0	5,5	55,1	07:20:00	22,8	4,8	56,0
23,0	4,2	55,4	07:25:00	22,8	3,0	57,0
23,0	4,0	55,6	07:30:00	22,8	2,7	57,0
23,0	5,5	55,4	07:35:00	23,0	5,7	57,0
23,0	3,8	54,9	07:40:00	23,3	3,3	56,0
23,5	3,2	54,5	07:45:00	23,5	3,0	56,0
24,0	6,8	54,0	07:50:00	23,8	5,0	54,0
24,0	2,8	53,6	07:55:00	23,9	2,0	54,0
24,5	2,5	53,0	08:00:00	24,4	0,8	53,0
24,5	5,0	53,3	08:05:00	24,6	3,5	53,0
24,5	10,0	53,6	08:10:00	24,6	8,0	53,0
24,5	6,8	53,6	08:15:00	24,6	2,0	54,0
24,5	5,0	53,8	08:20:00	24,6	4,5	54,0
25,0	8,8	53,3	08:25:00	24,9	2,5	53,0
25,0	4,0	52,7	08:30:00	25,3	3,9	52,0
25,5	2,8	52,6	08:35:00	25,7	2,0	52,0
26,0	5,2	52,2	08:40:00	26,2	4,6	51,0
26,0	5,5	51,7	08:45:00	26,6	5,0	50,0
26,0	7,5	50,6	08:50:00	27,1	7,8	49,0
26,5	4,5	51,1	08:55:00	27,1	3,5	50,0
27,0	9,0	51,0	09:00:00	27,1	8,9	49,0
26,5	10,8	50,7	09:05:00	27,2	9,5	50,0
27,0	6,5	51,0	09:10:00	27,3	6,9	49,0
27,0	8,2	51,3	09:15:00	27,4	7,5	50,0
27,0	7,0	51,7	09:20:00	27,3	6,7	50,0
27,0	3,2	51,6	09:25:00	27,4	3,5	50,0
27,5	4,2	51,4	09:30:00	27,5	3,0	51,0
27,5	10,8	50,7	09:35:00	28,0	9,6	50,0
27,5	6,0	50,8	09:40:00	28,0	4,0	50,0
27,5	8,5	50,9	09:45:00	28,2	8,0	50,0
28,0	6,5	50,3	09:50:00	28,3	6,3	50,0
28,0	4,8	51,1	09:55:00	28,2	5,1	50,0
28,0	6,0	50,8	10:00:00	28,4	4,0	50,0
correlation						
	0,995398	0,8499	0,94818			

The table on the left contains all the data samples of both the sensor system (left) and the reference system (right). Below the table correlation value of each corresponding subset can be found. This value is yielded using the CORRELATION() function of Microsoft Excel. The input of this function are two datasets and the function returns a value between -1 and 1. The two datasets are negatively correlating if the returned value is (close to) -1 and the datasets are positively correlating if the returned value is (close to) 1. The datasets are not correlating if the returned value is about 0.

As can be seen, the datasets of the temperature measurements show a strong correlation (0.995398). So do the datasets of the humidity measurements (0.94818). The datasets of the windspeed measurements also show correlation, but to a lesser degree compared to the other two variables.

Because these datasets show correlation and have the same length (i.e. $n = m$), a T-test can be performed in order to see if the differences between the corresponding measurements are statistically significant. Rather than comparing the two datasets, a one sample T-test is performed on the mean difference of the pairs of dependent observations. These differences can be found in the table on the right side and the standard deviations can be found in the last row.

The T-test is performed as described in [35].
The test consists of sevens steps;

1. Define Null and Alternative hypothesis (H_0 and H_1)
2. State the confidence level
3. Calculate degrees of freedom (df=N-1)
4. State decision rule
5. Calculate test statistic using $T_{test} = \frac{\bar{d}}{Sd/\sqrt{n}}$
6. State results
7. State conclusion

The null hypothesis is a hypothesis that states that the mean difference is zero and the alternative hypothesis states that the mean difference is not equal to zero. Because of this, the test is two-sided.

The confidence level will be 95% in all cases, this means that α = 5%

The degrees of freedom will be 48 in all cases, since the sample size is equal.

	temp	windspeed	humidity
	0,2	1,8	0,2
	0,3	0,2	0,2
	0,4	0,5	0,2
	0,2	0,2	0,1
	0,2	1,5	-1,9
	0,2	1,5	-2,3
	0,5	2,0	-2,5
	0,6	1,0	-2,4
	0,5	0,0	-2,4
	0,4	2,5	-2,7
	0,3	0,2	-1,8
	0,1	0,3	-2,4
	0,3	1,3	-1,4
	0,2	0,2	-1,5
	0,3	1,1	-1,6
	0,1	1,0	-1,0
	0,2	0,7	-0,9
	0,2	1,2	-1,6
	0,2	1,3	-1,4
	0,0	-0,2	-1,6
	-0,3	0,5	-1,1
	0,0	0,2	-1,5
	0,2	1,8	0,0
	0,1	0,8	-0,4
	0,1	1,7	0,0
	-0,1	1,5	0,3
	-0,1	2,0	0,6
	-0,1	4,8	-0,4
	-0,1	0,5	-0,2
	0,1	6,3	0,3
	-0,3	0,1	0,7
	-0,2	0,8	0,6
	-0,2	0,6	1,2
	-0,6	0,5	1,7
	-1,1	-0,3	1,6
	-0,6	1,0	1,1
	-0,1	0,1	2,0
	-0,7	1,3	0,7
	-0,3	-0,4	2,0
	-0,4	0,7	1,3
	-0,3	0,3	1,7
	-0,4	-0,3	1,6
	0,0	1,2	0,4
	-0,5	1,2	0,7
	-0,5	2,0	0,8
	-0,7	0,5	0,9
	-0,3	0,2	0,3
	-0,2	-0,3	1,1
	-0,4	2,0	0,8
std dev	0,3629	1,2	1,4

Temperature:

1. H_0: μ=0

   H_1: $\mu \neq 0$
2. The confidence level is 95%
3. Df = 48
4. Consulting the T-test table in [35, Tab-2] results in a T of 2.0106 ($T_{\alpha=5\%\ ,\ df=\ 48}$)
5. The test statistic is calculated with $\bar{d} = -0.1$, Sd = 0.3629 and n = 49.

   This results in a test statistic of -1.93
6. If the test statistic is within the interval of [-2.0106;2.0106], then the Null hypothesis is not rejected.
7. Given the fact that the test statistic falls within the specified interval, the null hypothesis will not be rejected. This means that there is no significant difference between the temperature measurements of the sensor system and the reference system (with 95% confidence).

Humidity:

1. H_0: μ=0

   H_1: $\mu \neq 0$
2. The confidence level is 95%
3. Df = 48
4. Consulting the T-test table in [35, Tab-2] results in a T of 2.0106 ($T_{\alpha=5\%\ ,\ df=\ 48}$)
5. The test statistic is calculated with $\bar{d} = -0.2$, Sd = 1.4 and n = 49.

   This results in a test statistic of -1.
6. If the test statistic is within the interval of [-2.0106;2.0106], then the Null hypothesis is not rejected.
7. Given the fact that the test statistic falls within the specified interval, the null hypothesis will not be rejected. This means that there is no significant difference between the humidity measurements of the sensor system and the reference system (with 95% confidence).

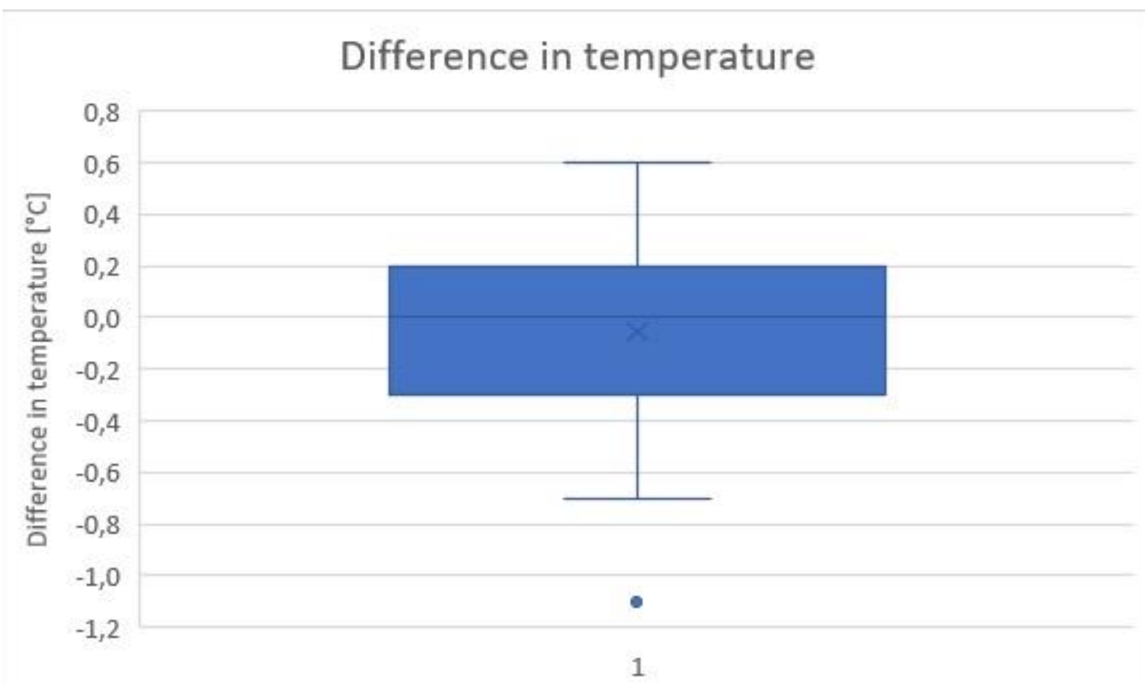
Windspeed:

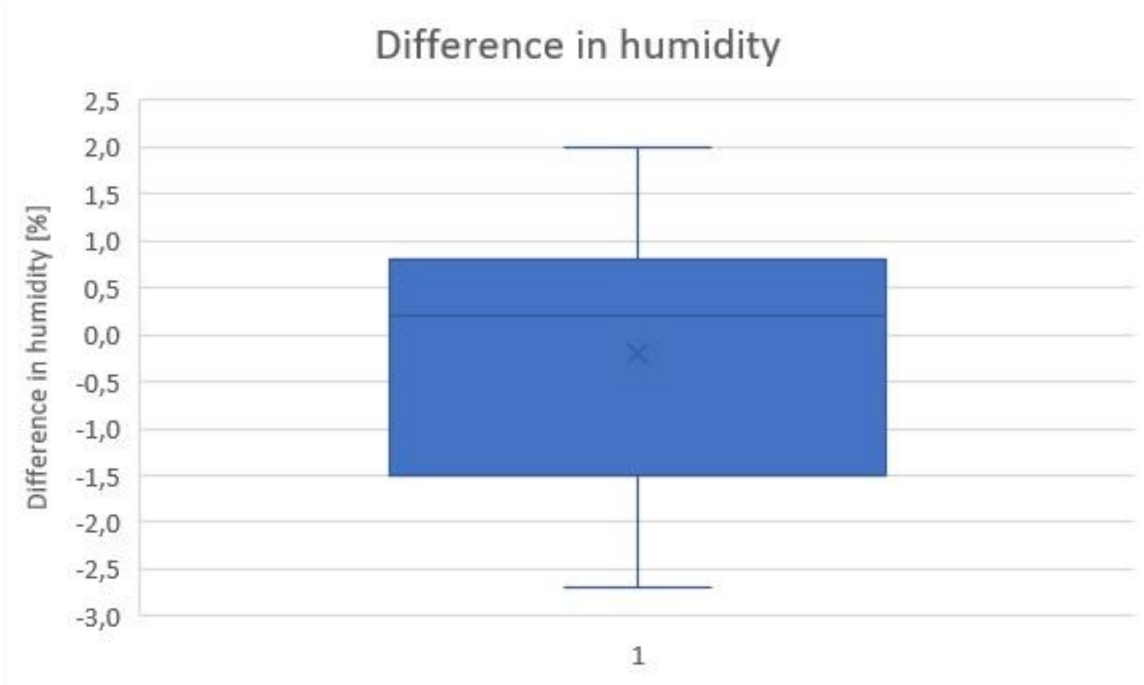
1. H_0: $\mu=0$

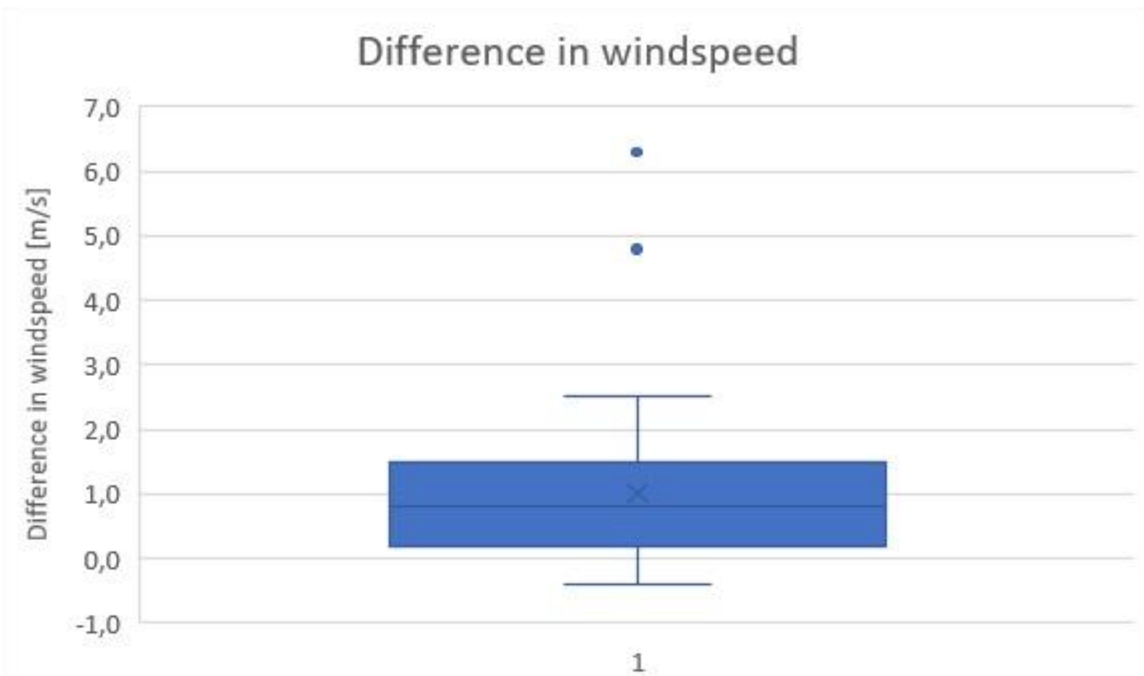
   H_1: $\mu \neq 0$
2. The confidence level is 95%
3. Df = 48
4. Consulting the T-test table in [35, Tab-2] results in a T of 2.0106 ($T_{\alpha=5\%,\ df=48}$)
5. The test statistic is calculated with $\bar{d} = 1$, Sd = 1.2 and n = 49.

   This results in a test statistic of 5.8
6. If the test statistic is within the interval of [-2.0106;2.0106], then the Null hypothesis is not rejected.
7. Given the fact that the test statistic falls outside the specified interval, the null hypothesis will be rejected. This means that there is a significant difference between the windspeed measurements of the sensor system and the reference system (with 95% confidence).

These tests are performed under the assumption that the differences between dependent pairs of observations are (approximately) normally distributed. The following boxplots shows that this is the case for all three variables:




Difference in humidity
Difference in humidity [%]
2,5
2,0
1,5
1,0
0,5
0,0
-0,5
-1,0
-1,5
-2,0
-2,5
-3,0
1


Difference in windspeed
Difference in windspeed [m/s]
7,0
6,0
5,0
4,0
3,0
2,0
1,0
0,0
-1,0
1

Appendix E: Component list and building manual

Components

3D printed

Part name:	Quantity:
❖ Bottom_hatch	1x
❖ Bottom_disc	1x
❖ Disc	2x
❖ Top_disc	1x
❖ Bottom_casing	1x
❖ Casing	1x
❖ Top_casing	1x
❖ Bottom_solar_panel	2x
❖ Top_solar_panel	2x
❖ Inside	1x

Screws and nuts

Part:	Quantity:
❖ M5x25	2x
❖ M2.5X8	8x
❖ M4X10	2x
❖ M4 nuts	2x
❖ M5	2x

Electronics

Part:	Quantity:
❖ SODAQ One v3	1x
❖ GPS antenna	1x
❖ LoRaWAN antenna	1x
❖ Header pins	6x
❖ Jumper	1x
❖ Female jumper cables	5x
❖ Two-way cable with JST connector	1x

❖ Dallas DS18B20	1x
❖ Adafruit DHT22	1x
❖ Froggit WH1080 anemometer	1x
❖ 4.7kΩ resistor	1x
❖ 10kΩ resistor	1x
❖ 8kΩ resistor	1x
❖ 100nF capacitor	1x
❖ TP4056	2x
❖ 1.5-Watt Solar panel	2x
❖ 1.2Ah LiPo battery	2x
❖ 1N4007 Diode	2x

Tools:

- ❖ Screwdriver
- ❖ Soldering iron + solder
- ❖ Wire stripper
- ❖ Heat shrinks
- ❖ Heat gun
- ❖ Sanding paper
- ❖ Glue gun + glue
- ❖ Screwdrivers

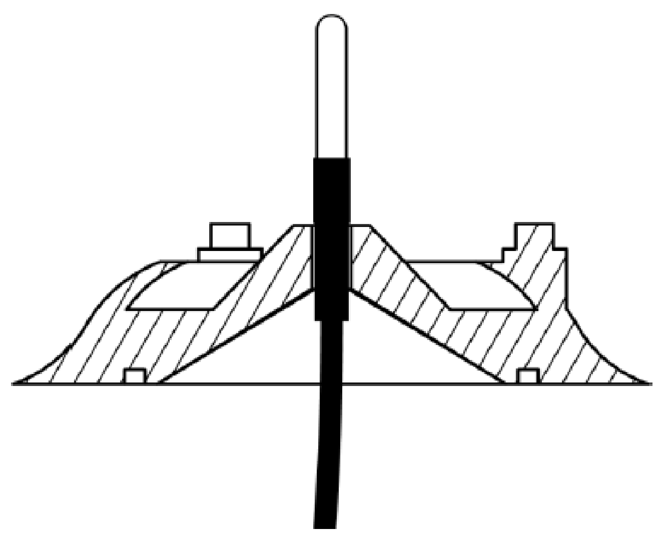
Building Guide

Step 1. Begin with removing the brim from the 3D-printed parts.

Step 2. Cut the DS18B20 temperature sensor so that the remaining cable is 15cm long.

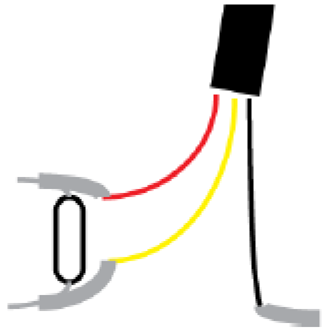
Step 3. Keep the cable, this will be needed later.

Step 4. Insert the DS18B20 temperature sensor in the bottom disc:

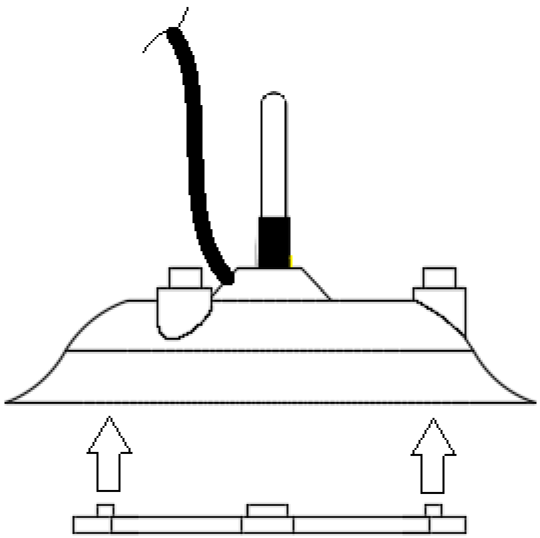


Step 5: Strip the individual cables from the DS18B20 so that the copper is exposed.

Step 6: Solder a 4.7kΩ resistor between the data cable and the Vcc cable:

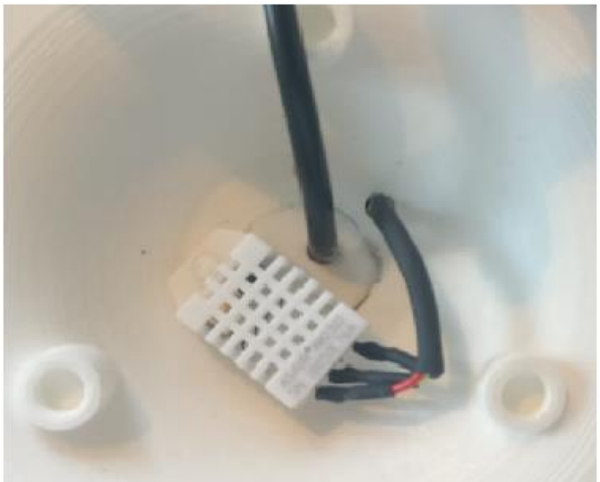


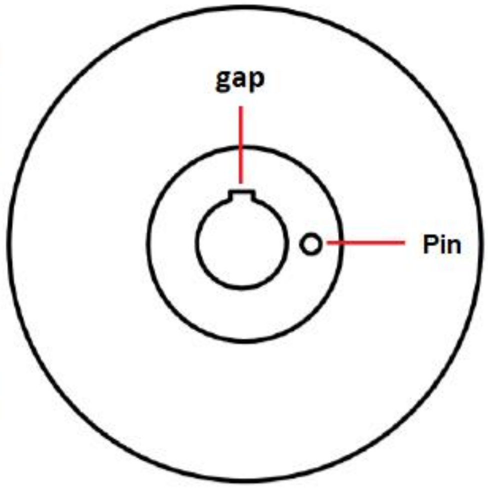
Step 7: Glue the bottom hatch to the bottom disc using super glue:



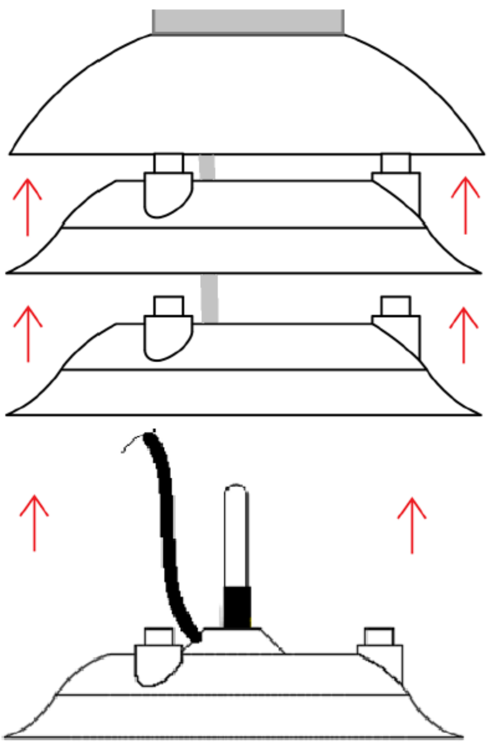
Step 8. Solder three cables to the DHT22 humidity sensor; one to the Vcc pin (1), one to the data pin (2) and one to the ground pin (4). Solder a 10kΩ resistor between the data cable and the Vcc cable similar to the DS18B20.

Step 9. Attach the humidity sensor to the top disc:

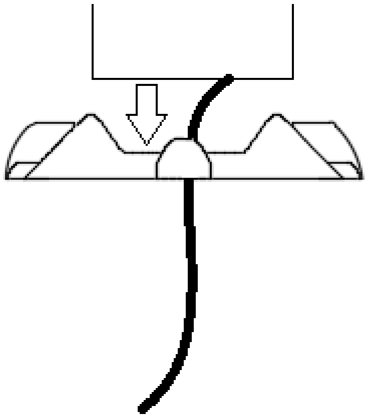




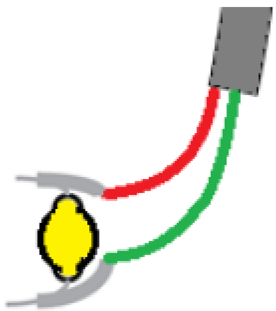
Step 10. Glue all the discs together, put the cable of the humidity sensor through the top disc:



Step 11: Put the cable of the Froggit Wh1080 anemometer through the top part of the casing and glue the anemometer to the top part of the casing. Use plenty of glue to make it waterproof:

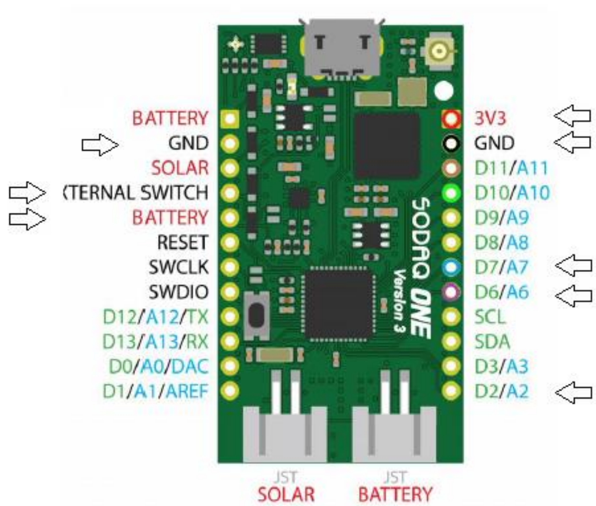


Step 12: Strip the wires of the anemometer and solder a 0.1µF capacitor in parallel with the anemometer:

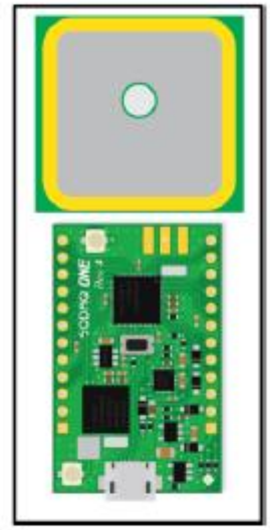


Step 13: Glue the top part of the casing to the main casing. Use plenty of glue for a waterproof seal.

Step 14: Solder header pins to the pins that are indicated by an arrow:



Step 15: Solder female jumper cables to the data pins of the sensors. Solder the Vcc cables of the DS18B20 and the DHT22 together and solder a female jumper cable to the connection. Do the same to the ground cables.

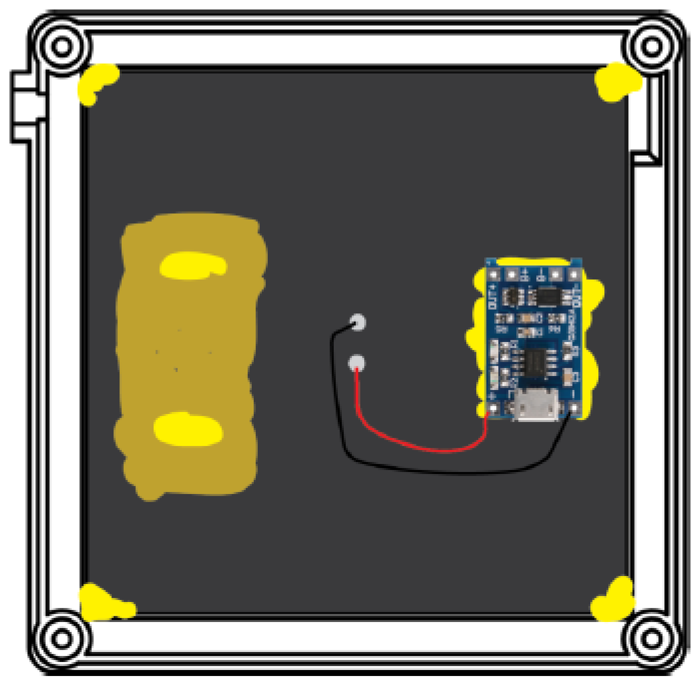


Step 16: Attach the LoRa and GPS antennas to the SODAQ One. Glue the SODAQ onto the inside part. Make sure the USB connection faces downwards.

The next component is needed twice, apply step 17 through 28 twice.

Step 17: Remove the plastic covering the solar panel.

Step 18: Place the solar panel in the indent in the top solar panel and glue it in place with hot glue and super glue.



Step 19: Remove R3 from the TP4056 and solder an 8kΩ resistor in its place.

Step 20: Glue the TP4056 to the solar panel.

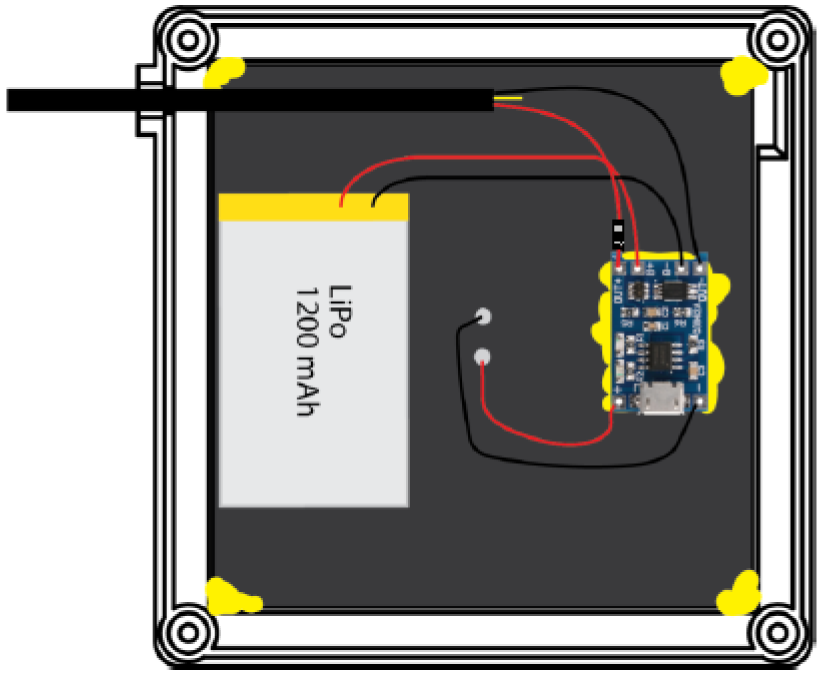
Step 21: Solder the red and black wires of the solar panel onto the input terminals of the TP4056.

Step 22: Add a lot of hot glue to the solar panel and wait for it to cool off until it becomes solid.

Step 23: Add a little bit of new hot glue and attach the battery to the solar panel. *Explanation; LiPo batteries do not handle heat well, a lot of glue is added to insulate the battery from heat emanating from the solar panel. The battery cannot be glued to the solar panel directly because* *it would cause the battery to warm up too much. The battery can handle the small amount of hot glue.*

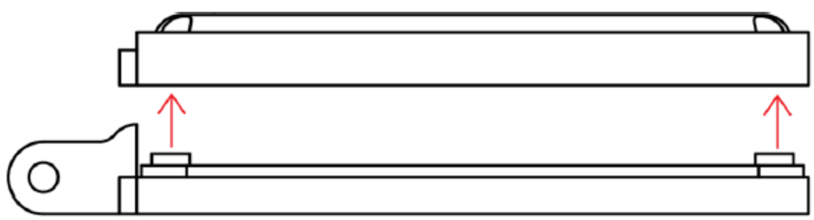
Step 24: Wait for the glue to dry and connect the cables of the battery to the battery terminals of the TP4056.

Step 25: Solder the 1N4007 diode to the positive output terminal of the TP4056. Make sure that the polarity is correct. Use the remaining cable of the DS18B20 and strip it. Solder this cable to the diode and the negative output terminal of the TP4056.

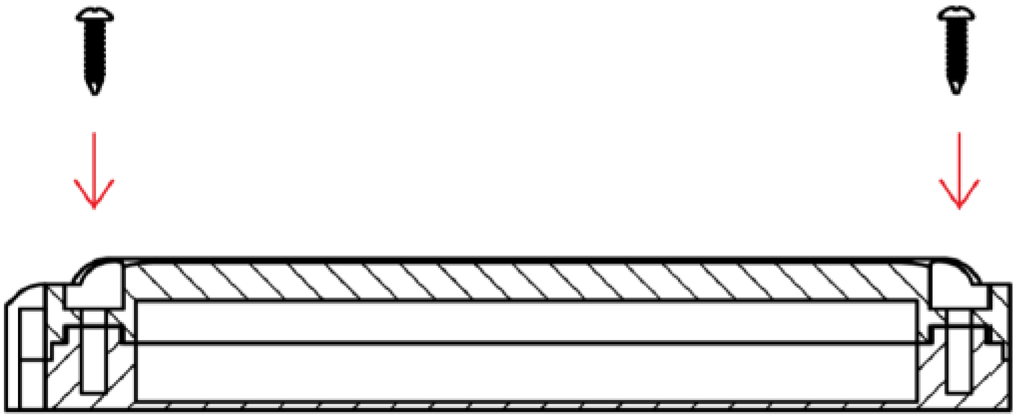


Step 26: Glue the cable to the casing of the solar panel to make a waterproof seal.

Step 27: Close the solar panel housing.

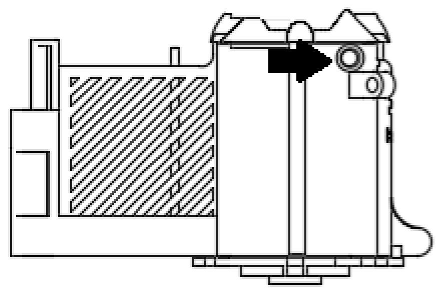


Step 28: Add M2.5 wood screws to secure it in place.



Note: this marks the end of the component that needs to be build twice.

Step 29: Put the cables of both solar panel modules through the holes in the main casing:



Step 30: Solder the positive cables of the solar panels together to the red cable of the two-way JST cable. Do the same thing with the negative cables.

Step 31: Attach the solar panel modules to the main casing using the M5 bolts and nuts.

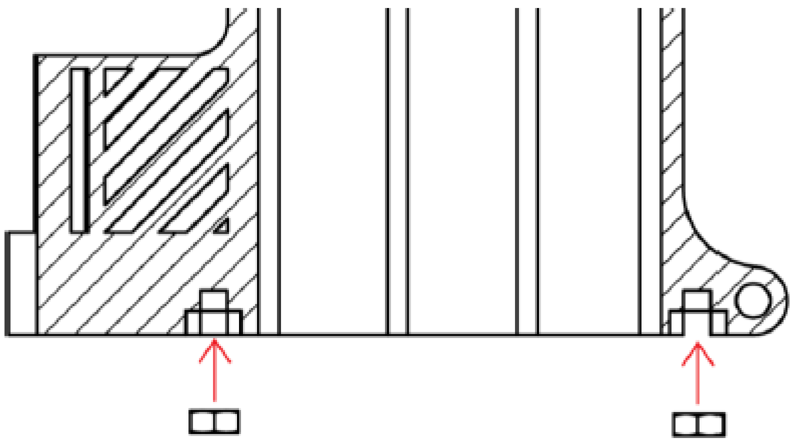
Step 32: Create a device in the TTN dashboard. Copy the dev UI and app key and fill these into the firmware_sensor_system.ino program.

Step 33: Connect the SODAQ One to the computer and upload the firmware_sensor_system.ino program to the SODAQ One.

Step 34: Connect one Vcc female jumper cable of to the 3.3V pin of the SODAQ One. Connect the ground of the sensors to the ground of the SODAQ One. Connect one cable of the anemometer to pin 6 of the SODAQ One and connect the other cable to the ground of the SODAQ One. Connect the data pin of the DHT22 to pin 2 of the SODAQ One and connect the data pin of the DS18B20 to pin 7 of the SODAQ One. Put a jumper between the battery and switch pin of the SODAQ One. *This is required for the SODAQ One to turn on its own.*

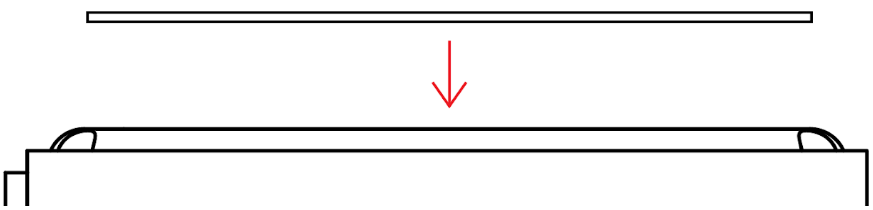
Step 35: Attach JST cable of the solar panels to the battery connector of the SODAQ One and slide the SODAQ One into the casing of the sensor system.

Step 36: Use a soldering iron to push the M4 nuts into the hole below the housing:



Step 37: Screw the radiation shield onto the casing of the SODAQ One using the M4 screws.

Step 38: Glue the delrin covers on the solar panels. Use contact glue to make it waterproof.



Step 39: Deploy the sensor system outside by strapping it to a pole.