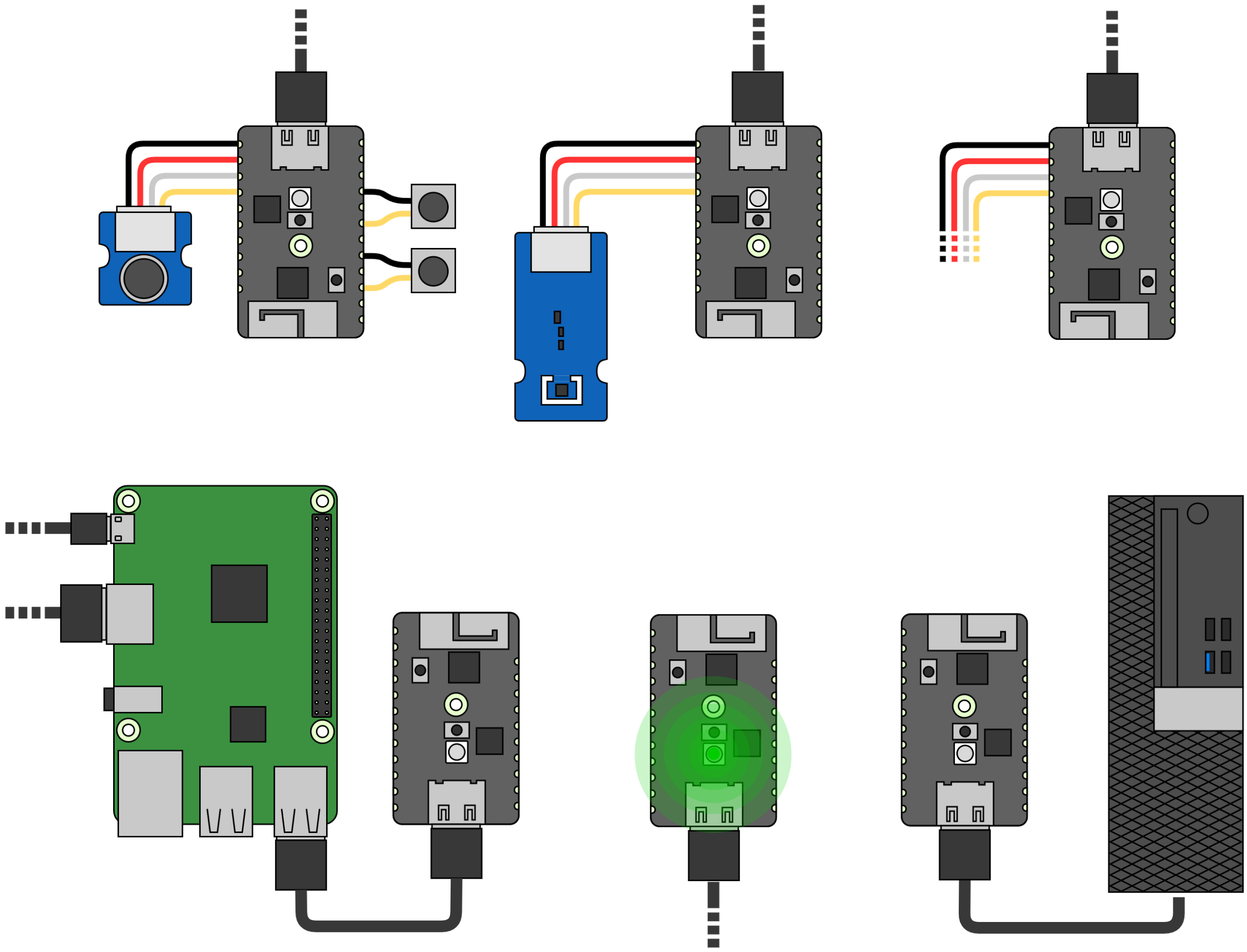
IoT-Based Smart Classroom

Subgroup: Networking

Mathijs van Binnendijk
Ciarán Lichtenberg



TU Delft

IoT-Based Smart Classroom

Subgroup: Networking

by

Mathijs van Binnendijk
Ciarán Lichtenberg

Student:	Mathijs van Binnendijk	4583957
Student:	Ciarán Lichtenberg	4437810
Project duration:	April 11, 2022 – June 13, 2022	
Thesis committee:	Dr. B. Abdi,	TU Delft
	Dr. J. Martinez,	TU Delft
	Prof. Dr. Ir. A.H.M. Smets,	TU Delft

TU Delft

Abstract

Technology is playing an increasingly important role in education, partially thanks to emerging teaching methods such as hybrid education. Smart classrooms equipped with technology can make the lives of the educators and students easier, and aid in the switch to hybrid education.
In this work we propose an IoT based smart classroom framework to create a base for smart classroom applications. Using this framework, multiple applications are manifested in the form of prototypes. One prototype introduces wireless sensor readout capabilities. The other prototype focuses on assisting the lecturer in a hybrid teaching setup. Using the prototype, the teacher is able to control whether online students are able to hear the students who are physically present and vice versa. The prototype also includes a way of indicating a question from an online student.

Preface

This thesis describes the Bachelor Graduation Project of group C. The subject, IoT-based smart classroom, was given by TU Delft to be researched. Improving hybrid teaching while also creating a smart classroom environment was defined as the goal. This research is in relevance for future education for TU Delft, as for other educational instances.

During the project we were guided by our supervisors dr. Jorge Martinez and dr. Bahareh Abdi. Therefore we would like to give special thanks to them. Secondly we would like to thank dr.ing. Ioan Lager for coordinating The Bachelor Graduation Project. Finally we are grateful to work with our colleagues: Mitchell Vuong, Bas Smeele, Richard Groenendijk and Tom Goedegebuure. Together we showed co-operative teamwork to reach our goal during the project.

Mathijs van Binnendijk
Ciarán Lichtenberg
Delft, June 2022

Contents

1 Introduction

In the past two years, the Covid-19 virus had a major impact on the world and heavily altered various aspects of society. Specifically, the importance of remote solutions became apparent. In the education sector this resulted in either fully online courses through software such as Zoom or a mix of online and local courses (hybrid). Now that the pandemic has mostly subsided, lectures have returned to normal. However, the pandemic did show the potential of remote teaching solutions to accommodate the needs and preferences of students and educators. The combination of face-to-face lectures with remote teaching, called hybrid teaching, has remained after the pandemic has ended. Unfortunately, this can create a disconnect between the local classroom and the online environment, for which very few solutions are currently available.

Hybrid teaching is a functionality of a smart classroom application. Therefore researching the current trend of hybrid education as well as smart classrooms will form an important context for the design and development of a successful product. Since hybrid teaching is a larger focus point than other smart classroom applications, it will be discussed separately from smart classrooms. Section 1.1 discusses the issues and challenges of hybrid teaching. Section 1.2 discusses the issues and challenges of smart classrooms. These sections are followed by the general problem definition and research of existing products. Section 1.5 describes the solution that will be designed in this thesis. The design of this solution will be subdivided in three subgroups and discussed in 3 different theses as mentioned in section 1.6.

1.1. Hybrid teaching issues and challenges

To understand the problems teachers have using hybrid education and smart classroom technology, a survey was sent to all educators of the faculty of EEMCS at Delft University of Technology. The responses to the survey can be found in appendix B. For hybrid classrooms, a recurring problem was the interaction with the online students. Questions from online students are often missed as the attention of an educator is mostly on those physically present. They find it difficult to split their attention between both groups.

Another survey [1] showed that 62% of the educators feel that hybrid teaching should be adopted. Another 23% of educators want to adopt the hybrid teaching model but are untrained and unsure about it. This shows strong support for hybrid teaching models by educators and the importance of ease of use to convince the 23% that are still unsure. Another important result of the survey is that 73% of the students are willing to adopt the new model. 18% are willing but they don't have the proper resources or skills. This shows a strong support for hybrid teaching models by students as well.
This support for hybrid teaching is justified when looking at the pros [2]:

- **Ease of participation:** Students are able to participate both synchronous and asynchronous during courses allowing for great flexibility.
- **Choice in teaching format:** The ability to choose between classroom and e-learning possibilities allows students to choose their preferred learning style.

- **Cost effectiveness for the institution:** More students making use of hybrid education can reduce the opportunity costs for institutions as well as students. The institution can spend less on classroom space, utilities and upkeep costs. Meanwhile, students can save on travel costs.

The hybrid teaching model does also have cons [2]:

- **Computer literacy:** Not every student is computer literate or has access to the resources needed for e-learning.
- **Course design difficulties:** Designing a course to effectively meet the needs of e-learning can be a sizeable challenge for the educator. Educators can also feel overwhelmed since they are responsible for more aspects such as technology, delays and dividing attention between online and face-to-face [3]. Teaching effectively while taking all these new aspects into account can introduce complexity.
- **Lacking face-to-face time:** Some students do best when physically present or need instructor face-to-face time to fully grasp a subject or to feel engaged in the learning process [4] [5]. Learning online is not the best environment for these students.
- **Teachers non-convinced by new teaching methods:** One of the interesting results based of the survey appendix B is that teachers are often non-convinced by new teaching techniques. One of the arguments presented is that the technology needed for hybrid teaching is not working properly. It also takes a lot of time and effort to understand the new technology. So this decreases the willingness to adapt to new teaching methods [6].

Regarding computer literacy, it is not much of an obstacle with the Netherlands ranking among the top EU countries in terms of digital skills according to the CBS [7]. This statistic could not be possible without overall access to the resources needed for e-learning, especially among young people.

Regarding course design difficulties, designing a course to effectively meet the needs of e-learning is difficult. This issue can be solved through education and experience. The division of students groups both online and offline can increase the amount of tasks teachers have to focus on during lecturing. *Care should always be taken to minimize the workload placed on teachers when hybrid technology is designed.*

Regarding face-to-face time, the fact that some students do better in physical based education can be taken into account within the education program. *It is important to allow for face-to-face teaching for those who prefer this style of education.*

And finally, regarding non-convinced teachers, it is understandable that not all teachers are convinced by hybrid teaching methods. It is important to take the stress and inconvenience introduced by hybrid teaching solutions into account, when they are not functioning as they are supposed to. This brings us to the last important takeaway, *The hybrid technology designed should be user friendly and intuitive.*

It can be concluded that, while hybrid teaching has its disadvantages, it can offer some significant benefits. This is why we would like to introduce a technological solution for a possible hybrid teaching application. This will be further elaborated on in section 1.5.

1.2. Smart classrooms issues and challenges

Within this project, smart classrooms are defined as classrooms equipped with technology providing some sort of benefit in an educational environment. This technology can help in hybrid education but also provide other uses in terms of security, environmental control, data collection, and visualisation. One interesting way of implementing such technologies is making use of the Internet of Things (IoT). IoT allows for an interconnected set of devices like sensors, actuators, and other technologies. All these devices are able to communicate with one another, thereby creating a wide range of possible functionalities. The following list gives some examples to get a general idea of the possibilities:

1. **Security**
   (a) Detecting persons in off-limit areas and notifying personnel.

  (b) Detecting emergencies like fires, and notifying authorities.
  (c) Wireless door locks.

2. **Environmental control**
  (a) Automated HVAC (heating, ventilation, and air-conditioning) control based on temperature and humidity measurements.
  (b) Controlling blinds based on light intensity.
  (c) Controlling lights based on occupancy.
  (d) Air quality measurement and control.
  (e) Automatically switch scenes when a different learning environment is required (like turning off lights when a projector is turned on).

3. **Hybrid teaching**
  (a) Giving educators visual feedback when there is a question online.
  (b) Automated recording of lecturers and students for online reviewing.

4. **Data collection and visualization**
  (a) Attendance tracking of students (for example by using student cards).
  (b) Using sound level measurements and machine vision to measure whether students are still paying attention.

IoT systems within education have a lot of advantages: reduction in cost, enhancement of comfort, saving time, enhancement in safety, exploring personalised learning, and increasing student collaboration [1] [8]. The need for specific applications will differ between institutions. Thus, *an IoT system should be easily customizable to fit the needs of differing institutions.*

IoT based systems also create a number of challenges. One difficulty is that teachers have problems with using the technology or technology is not working all the time. According to our survey (appendix B), educators want smart teaching solutions to work automatically without having to set it up. As classrooms are used by multiple teachers and with multiple groups of students, any smart solution should be plug and play.

Furthermore, researchers agree that applying IoT in education could create a challenge in terms of security and privacy [1] [9]. This is because of an increase in data collection on things like credentials, location, and learning history of students. This is why *security and privacy should be taken into account when designing an IoT system.*

Another obstacle in implementing IoT technology in a classroom setting is cost [9]. Educational institutions could struggle with the budget for implementing a smart classroom IoT system. This is why *the cost should be minimized when designing an IoT system.*

1.3. Problem definition

In the sections above, the advantages and disadvantages of both hybrid teaching and smart classrooms are discussed. After this assessment of the challenges educators face, the problem that will be tackled in this project is defined. In this project a framework will be designed for a smart classroom using IoT solutions. Within this framework a useful application for hybrid teaching will be integrated in the system. The complete problem definition can be stated as followed:

How to design a complete and expandable IoT system that implements a smart classroom framework and improves hybrid education?

This problem encompasses two focus points. The first being the design of a complete and expandable IoT system for a smart classroom. The problem herein solved is the difficulty of using smart solutions in a classroom. The second focus point is improving hybrid education. The main difficulty lecturers face is the interaction between online and physical audiences and not being able to focus on both (appendix B).

1.4. Product research

Hybrid teaching is a concept that has existed for a couple of years, during which multiple hybrid teaching systems have already been designed. However, no system has been able to resolve all identified issues in section 1.3. Some notable companies who have developed hybrid teaching systems are Logitech, Aver, Paramtech, and ViewSonic.

Logitech has designed a conferencing system that could be used for hybrid teaching. The system consists of at most one ultra high definition camera, two speakers, and seven microphone hubs. Since the system is not expandable, it can only be used in relatively small classrooms [10].

Aver also designs systems for conferencing purposes. This system has a person tracking camera, microphone range of 10 meters, and a single loudspeaker. This product is mainly used for conferencing since audio and video range is limited [11].

Paramtech is a company that creates solutions for hybrid teaching applications. Their hybrid teaching system is expandable up to one camera, twenty microphones, and two loudspeakers. They also offer a virtual microphone system that has two microphone bars. The audio range is for spaces up to 9.14 m by 15.24 m. The presenter in the room can walk freely because the camera has a tracking system, and the audio bars cover the whole room. But this system is applicable for a classroom specific. It can not be expanded to larger rooms as stated above. So the system cannot be used in a lecture room [12].

ViewSonic is developing a hybrid teaching system that focuses on both physical and online student engagement. The online students are projected on the wall and the physical students are audio and video recorded. At this moment ViewSonic designs systems for a standard classroom application since the amount of audio/video modules is limited. Due to the operational range of this equipment the product cannot be used for lecture halls [13].

Similar to hybrid teaching, the concept of smart classrooms is not new. With the advancements in technology, smart classroom tools have slowly crept into the education sector. For example, almost all classrooms are equipped with a projector, interactive whiteboard, or both. On another front, digital tools, such as the quizzing tool Kahoot![1], are also being used more often in lectures. Additionally, uploading lecture slides, lecture recordings, and other learning resources through software such as brightspace or blackboard, has become the norm. More recent advancements are products such as ScanMarker[2], a scanner shaped like a marker which can read text and write it to a digital text editor. Alternatively, companies such as Magicard[3] develop ID cards which can track student attendance, improve security through authentication and access control, or help regulate services such as printers or study rooms.

The future of smart classrooms looks promising, but still has a long way to go. Smart classroom products can be expensive, require extensive instructions, or be unreliable according to the responses of the survey in appendix B. All of this combined makes the education sector slow to adopt new technologies and educators hesitant to integrate new tools into their lectures (appendix B). Additionally, most tools work to enhance the already existing classroom environment and rarely try to connect the physical classroom with a remote teaching environment.

So there exist companies that offer hybrid teaching solutions. But these solutions are designed for small conferencing rooms or classrooms. A solution particular for a lecture room is not yet on the market. Secondly there are solutions for hybrid teaching and for smart education. But until now there isn't a system that covers both.

1.5. IoT-based smart classroom

When looking at existing products, see section 1.4, it can be observed that they all are quite limited in terms of functionality and/or expandability. The proposed product will address this problem, as well as the problems mentioned in section 1.3.

[1] Kahoot! is a game-based learning platform, used as educational technology in schools and other educational institutions. For more information: https://kahoot.it/

[2] For more information: https://scanmarker.com/

[3] For more information: https://magicard.com/

The core of the product is an expandable IoT network which will communicate between devices and a server. The IoT product will assist educators with hybrid teaching as well as offering smart classroom features. The smart classroom features include measuring the humidity, temperature, and loudness in a classroom to monitor the learning environment. For the hybrid teaching, a prototype will be created to make interacting in the lecture easier in a hybrid education setting for both the students and educators. See fig. 1.1 for the setup.

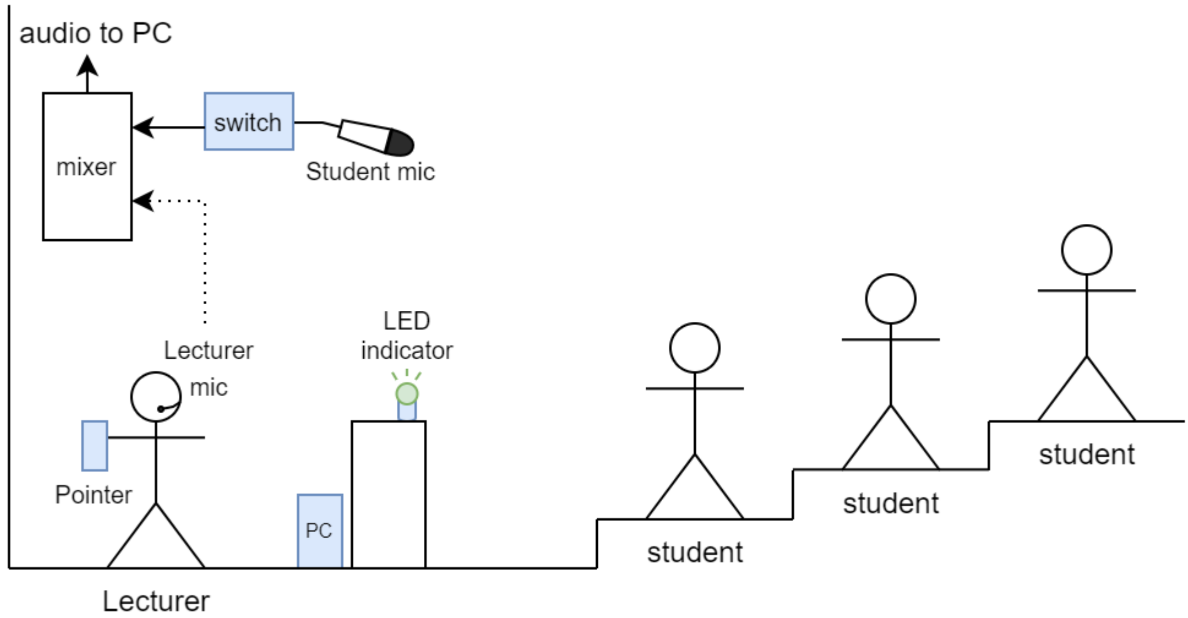


Figure 1.1: Hybrid teaching product setup

The important elements of the proposed product are: a computer running the online meeting software, an LED indicator, a laser pointer, and a student microphone. The computer running the online meeting software will have a program/app running on it, that will see if students online have a question. In turn, the computer will send a signal to indicate that a student has a question, this will light up an LED visible by the teacher. When the lecturer sees the LED indicator change color, they can then choose to unmute the online student(s) with a device attached to a laser pointer so everyone can hear the question in the class. This laser pointer can also be used to turn on the student microphone so that the students online will be able to hear the questions from the students attending the lecture in person. With this, the lecturer won't have to look for the online questions or need to repeat any questions, making the flow of the lecture smoother and decreases the work load on the lecturer.

1.6. Submodules of an IoT-based smart classroom

The project is divided into three subgroups: hardware, network and server subgroup.

1.6.1. Hardware subgroup

The hardware subgroup designs and implements the peripheral devices needed for the system. These devices send include environmental sensors, audio equipment and actuators. The hardware devices communicate their data to an IoT development board.

1.6.2. Networking subgroup

The network subgroup is responsible for the wireless communication between IoT devices. The hardware subgroup will communicate their data with an IoT development board. The networking group will take this data and send it wirelessly between IoT boards. This means the networking group will have to look for a suitable wireless technology. Once a suitable wireless technology is found this technology must be implemented to fulfill the requirements of the prototype.

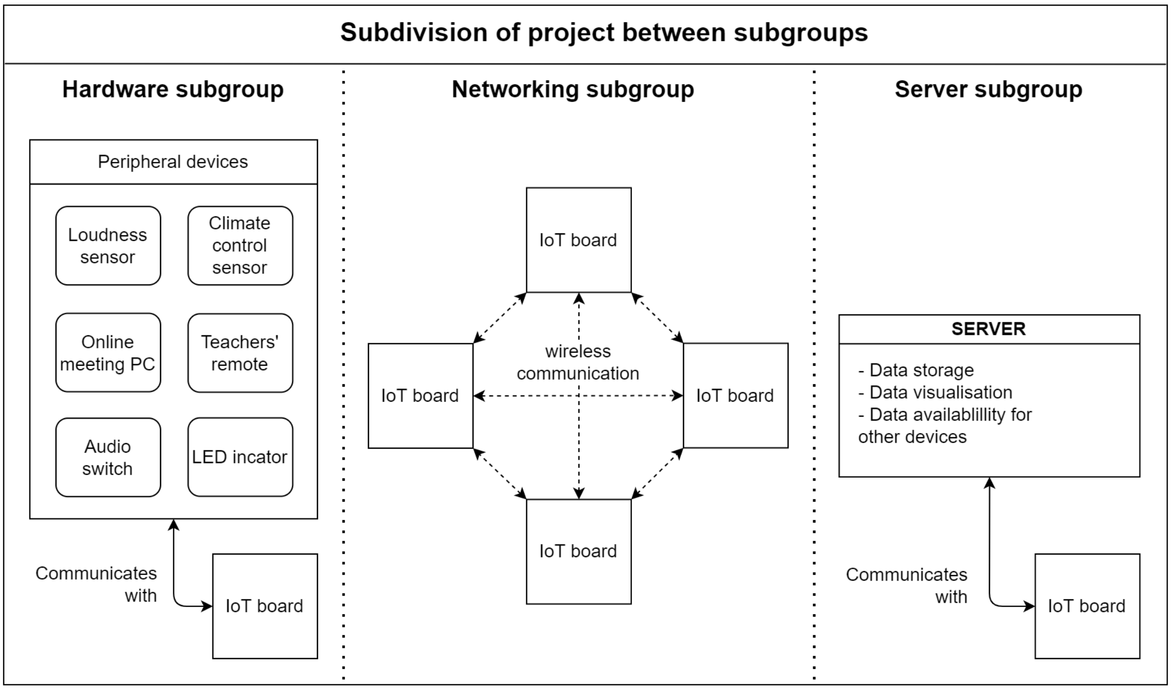


Figure 1.2: The subdivision between the subgroups

1.6.3. Server subgroup

The server subgroup designs a server which will store the acquired sensor data that will be received from the network and also provides the data that is requested to external and internal devices. This means that the server group will need to decide on a hardware platform to create the prototype on, on the software implementation of the server, and how the requirements of the prototype will be implemented.

2 Program of requirements

In this chapter, the requirements will be set, in order to develop our IoT product for hybrid education. First, the main takeaways will be discussed, followed by the Program of Requirements (PoR) of the product, and then the PoR of the subgroup will be defined.

2.1. Main PoR takeaways

The context research in chapter 1 yielded some important points when making the PoR:

TA1 Care should be taken to minimize the workload placed on teachers when hybrid technology is designed.
TA2 It is important to allow for face-to-face teaching for those who prefer this style of education.
TA3 The hybrid technology designed should be user friendly and intuitive.
TA4 An IoT system should be easily customizable to fit the needs of differing institutions.
TA5 Security and privacy should be taken into account when designing an IoT system.
TA6 The cost should minimized when designing an IoT system.

2.2. General PoR

The mandatory requirements, which were chosen in order to create a good working product, can be found below. The Trade-off requirements are chosen to improve the product as well. The boundary conditions are chosen to determine what will be left outside the scope of the project, and what will be included. The numbers following some of the requirements are references to the main PoR takeaways discussed in section 2.1.

Mandatory requirements

General

M1 The collected data must be readable through a GUI (Graphical User Interface). [**TA1**, **TA3**]
M2 The collected data must be available for external devices. [**TA4**]
M3 The system must be able to log and save data.

IoT

M4 The network must be able to accept new IoT devices while in operation. [**TA3**]
M5 IoT devices must be able to send and receive data via a wireless network.
M6 The network should support sensors applicable for a classroom. [**TA4**]
M7 The coverage of the network must be expandable. [**TA4**]
M8 Losing an IoT device should not completely stop the operation of the network. [**TA3**]

Hybrid

M9 The teacher must be notified of any questions from online students. [**TA1**, **TA3**]
M10 The teacher must be able to switch between mute and unmute of the audience microphone and the students online. [**TA1**, **TA2**]
M11 The online students must be able to hear the physically attending students. [**TA2**]

Trade-off requirements

T1 The system should be affordable. [**TA6**]
T2 The system should be customizable. [**TA4**]
T3 The system should be able to be operated without extensive technical knowledge. [**TA1**, **TA3**]
T4 The system hardware should have a small form factor.
T5 The system should have basic event-based decision-making capabilities.

Boundary conditions

B1 The security of the system will not be a main focus. [**TA5**]
B2 The privacy of the system will not be a main focus. [**TA5**]
B3 Only IoT devices applicable to a classroom will be discussed.

While security and privacy are important considerations, this topic deserves its own thesis. Therefore they are taken into consideration, but within the allotted time of this project, further research and design into this area is left outside of the scope of this thesis.

2.3. Network group PoR

Below are the mandatory and trade-off requirements of the network subgroup. These are mainly derived of the general requirements. Furthermore, the boundary conditions of the subgroup are stated.

Mandatory requirements

NM1 The network should communicate wirelessly. [**M5**]
NM2 The network must be able to accept new nodes while in operation. [**M4**]
NM3 The hardware used for the network must be below €10,- per node. [**T1**]
NM4 The coverage of the network must be expandable. [**M7**]
NM5 Losing a node should not stop the operation of the network as long as enough redundancy is created. [**M8**]
NM6 Sending messages between two nodes should at least have a 99% success rate in optimal conditions. [**M5**]

Trade-off requirements

NT1 The steps needed to set up the network should be minimized [**T3**]

Boundary conditions

NB1 The security of the IoT devices will not be a main focus. [**B1**]

3 Design

First, this chapter will look at different networking protocols. To find a suitable protocol, they will be compared to each other on different aspects. Next, the inner workings, features and, implementation possibilities of the chosen protocol will be explained. Finally, possible hardware solutions will be discussed. In the next chapter (chapter 4), the proposed system will be explained.

3.1. Wireless technology

In order to choose a networking protocol for our project, the different available options need to be studied. IoT-related wireless technologies developed in recent years are varied in terms of protocols, performance, reliability, latency, cost effectiveness, and coverage. For instance, some protocols are designed for short-range radio communications (e.g., Bluetooth and ZigBee), another type of protocol could be more suitable to cover wide areas with small bandwidth, while others are designed for middle-range communications and high transmission rate (e.g., IEEE 802.11) [14].
The different protocols will be examined in the following aspects:

Topology is important to fulfill requirement **M7**, the topology should allow the coverage to be expanded. Three types of topologies are commonly used: star, tree and, mesh. Star networks only allow devices to communicate with one center device and thus limits expansion. A tree network can be expanded by using multiple center devices. Mesh networks lets devices communicate with all other devices and thus allows for expansion. Mesh therefore is a preference for the network protocol.

Power usage is important for future uses of the system. In the proposed system, no battery-powered devices are discussed. This might change for future smart devices added to the system. Preferably, the power should be below 200 mW when transmitting or receiving.

Data rate is a property of the network that needs to be large enough to support multiple nodes sending data. For large lecture halls, the system might contain many nodes sending data over the network. The protocol should be able to support this without extra delays. The data rate should preferably be higher than 10 kbps.

Range or coverage is an important property of the network. Large lecture halls require a range preferably higher than 50 meters (for example the Auditorium at the University of Technology Delft [15]). This can also be accounted for by using an expandable network topology.

Cost is an important design decision to keep in mind. Educational institutes do not have funds to buy expensive systems. To fulfill requirement **NM3**, the cost of a development board should be below €10,-.

3.1.1. Wi-Fi

The widespread adoption of Wi-Fi makes it a first choice for many IoT applications. However, in some IoT applications, the choice of technology is limited by the hardware capabilities of the device, low-power consumption requirements, and the overall cost [16]. Wi-Fi uses the IEEE 802.11 wireless standard.

Network topology and range

Wi-Fi (IEEE 802.11) can have either a star or a tree network topology. It requires a router in order to get messages to the right end-devices. For a tree topology, multiple access points must be used. Using Wi-Fi on a single-board micro-controller (without an external antenna) will give you a range of about 50 meters [17]. Using a tree topology with multiple access points allows for the expanding of network coverage [18] [19]. These access points have to be connected through cabling.

Power usage and data rate

When a micro-controller uses the Wi-Fi protocol, the power required is around 820 mW subsection 3.1.1 when receiving or transmitting data. The higher power usage makes Wi-Fi less usable for low-power IoT devices and is higher than the boundary of 200 mW. The data rate of Wi-Fi however, is large (around 7mbps when using the M5Stack M5Stamp C3, see subsection 3.1.1) [17].

Cost

A micro-controller board that has Wi-Fi capabilities is for example the M5Stack M5Stamp C3 [20]. It is a low-cost board (around €7,-) with documentation readily available.

3.1.2. LoRa

LoRa is a wireless communication protocol for long-range, low-power and, low-data-rate applications developed by Semtech. [21] Depending on the obstructions and physical characteristics of an environment, LoRa can cover hundreds of square kilometres. In contrast to cellular technologies that support high data throughput, LoRa is designed for IoT devices and many-to-many applications that require the exchange of only small amounts of data over longer distances [16] [22].

Network topology and range

LoRa networks typically are laid out in a star-of-stars topology. In this kind of topology, gateways relay messages between end-devices and a central network server. Gateways are connected to the network server via secured standard IP connections. End-devices use single-hop LoRa to communcate with the gateways. The range of LoRa devices can differ from 100 meters to even a few kilometers [16] [23].

Power usage and data rate

The LoRa protocol has low power usage. It is designed to be used with battery-powered devices. When transmitting or receiving data, the power usage is about 100 mW [24]. This is lower than the boundary of 200 mW. The data rate of the LoRa protocol however is low (around 1 kbps) [16].

Cost

The RFM98 [25] module is a low-cost board to add to any micro-controller for accessing a LoRa network. It is a LoRa transceiver including a LoRa modem. Together with a basic micro-controller, the cost will be around €15,-.

3.1.3. ZigBee

ZigBee is a low-power wireless communication protocol developed by the ZigBee Alliance. It uses the IEEE 802.15.4 wireless standard. The intended use is as an IoT network for home and building automations [22].

Network topology and range

The ZigBee network can be arranged in star, tree and mesh topologies. In a star network, a ZigBee coordinator (router) will initiate and maintain the devices on the network. The other devices (end-devices) will only communicate with the coordinator. In a tree network, more ZigBee routers can be used to extend the network coverage. Still the end-devices only communicate with the routers. The mesh network allows end-devices to directly communicate to each other and relay messages [26]. A basic micro-controller with the ZigBee protocol (without an external antenna) has a range of around 15 meters [17]. The tree and mesh topologies make sure that the coverage of a network can be expanded.

Power usage and data rate

The power usage of ZigBee compared to Wi-Fi is lower (see subsection 3.1.1). For a basic micro-controller with the ZigBee protocol running (as an end device), the power usage is around 170 mW when receiving or transmitting data. The data rate of ZigBee can be found in the order of 100 kbps [17]. This makes the ZigBee protocol usable for IoT low-power devices.

Cost

An example of a basic micro-controller board which can use the ZigBee protocol is the nRF52840 [27]. This board has scripts and documentation available. The cost of this board is around €20,-.

3.1.4. Bluetooth LE

Bluetooth Low Energy (or BLE) is a wireless network technology produced by the Bluetooth Special Interest Group (SIG). Bluetooth Low Energy is intended to decrease the power usage and the cost of devices in comparison with the normal Bluetooth connections. An advantage of using BLE is that nearly every smartphone supports it and is able to connect to the IoT network.

Network topology and range

BLE can be used in a star or mesh network topology. BLE Mesh is an addition to the normal BLE protocol and provides extra features. The BLE protocol uses the Generic Attribute Profile (GATT). BLE Mesh however uses the General Access Profile (GAP). Nodes in a BLE network can take 2 roles. A node can either be a server or a client. A client node will initiate commands and requests and will accept responses. A server node receives commands and request and returns responses.

Power usage and data rate

Bluetooth Low Energy is a low power protocol. When transmitting or receiving data, the power usage is around 125 mW for a basic micro-controller with BLE capabilities. A special feature of BLE Mesh however, is the Low Power Node (LPN). When a micro-controller is programmed to be a LPN, the power consumption in between transmissions is decreased further. BLE has a data rate of around 100 kbps, which is high enough for an IoT network [28] [18].

Cost

A basic micro-controller board is for example the M5Stack M5Stamp C3 [20]. It is low-cost (around €7,-) and has documentation available.

3.1.5. Comparing IoT network technologies

In order to choose a suitable network protocol, the different wireless technologies should be compared. In Table 3.1 contains the comparison concluded based on information from the previous sections .

	Wi-Fi	**LoRa**	**ZigBee**	**BLE**
Topology options	Star	Star, Mesh	Star, Tree, Mesh	Star, Mesh
Power consumption (relative)	high	low	low	low*
Data rate (order of magnitude)	10 mbps	1 kbps	100 kbps	100 kbps
Approximate maximum range	50 m	10 km**	100 m**	50 m**
Cost (dev board)	€7,-	€15,-	€20,-	€7,-

Table 3.1: Comparison of the different IoT network protocols where green indicates a preferable attribute, orange an acceptable attribute and, red an unacceptable attribute in accordance with our project requirements.
* With the use of a LPN node, power usage is reduced further
** Mesh topology means network coverage can be expanded

Topology
To be able to expand the coverage of a network, a mesh topology is preferred. LoRa, ZigBee and BLE all have this option, while Wi-Fi is only in a star configuration. Wi-Fi however, when expanded with more access points, can turn into a star-of-stars network. While a mesh network lets the IoT devices themselves expand the network coverage, Wi-Fi needs more access points. As more devices are needed, the cost will increase. Classrooms can be found in different sizes. For small classrooms, a simple star network could be enough to reach every corner of the room. However, if the system needs to be implemented in a large lecture hall, the network must be able to cover more space. This can be achieved with a mesh network by adding more nodes as each node is able to increase the network coverage. This is why using a mesh network would make sure the coverage of the network is expandable and thus meet the mandatory requirement **NM4**.
A disadvantage to Mesh networks can be found in the relaying of messages. As messages are sent, every node will relay the message to all other nodes. This can create a situation in which the message circles around in the network. BLE Mesh counters this by a built-in feature. This can be read about in section 3.2.2.

Power
The network should be able to support battery-powered nodes if needed for future expansions. A low-power protocol is therefore preferred. LoRa, BLE and ZigBee all are low on power usage. BLE Mesh has the Low Power Node feature that will reduce the power usage even more (see section 3.2.2). Wi-Fi will reduce the lifetime of a battery-powered node as it uses more power. This makes it less suitable for battery-powered nodes.

Data rate
It should be possible to add a multitude of nodes to the IoT network. The data rate of the protocol must therefore be sufficient to handle all of the data communication. Wi-Fi, ZigBee and BLE all have a high data rate which prevents bottlenecks when a multitude of nodes is communicating over the network. LoRa has only a data rate of around 1 kbps, which could potentially be insufficient for larger networks.

Range
The range (coverage) of the network is important to support large lecture halls. LoRa, ZigBee and BLE can be configured as mesh protocols and will therefore be able to expand there coverage by adding nodes. Wi-Fi has a range of between 10-50m and can not be configured as a mesh network. This might be sufficient for small classrooms, but not for big lecture halls or rooms containing obstacles interfering with wireless communication.

Cost (dev board)
Wi-Fi and BLE are able to use the same development board, the M5Stack M5Stamp C3. [20] This is a low-cost board. LoRa and ZigBee need boards which are a more expensive (respectively the RFM98 [25] and the nRF52840[27]).

Conclusion
In all of the above categories, BLE stands out as a perfect match. Because of the mesh topology, it is scalable from small classrooms to large lecture halls. It has high bandwidth to support messages coming in from many IoT devices. Next to this, it also has low-power features which is useful for battery-powered devices.

3.2. Bluetooth Low Energy Mesh

Based on subsection 3.1.5, Bluetooth Low Energy (BLE) Mesh was chosen for wireless communication of the IoT devices. This section gives a brief overview of the Bluetooth Mesh standard needed in chapter 4.

3.2.1. BLE Mesh and Bluetooth LE

BLE Mesh is a networking stack that uses Bluetooth Low Energy (BLE). BLE provides the radio capabilities used by the Bluetooth Mesh protocol. While BLE implementations normally use one-to-many typologies, the Bluetooth Mesh protocol allows for a many-to-many topology. [29] However, there are some limitations in the many-to-many topology implementation when power consumption needs to be limited. These limitations will be discussed in the end of subsection 3.2.1.

3.2.2. Nodes

The mesh network functions using intercommunicating nodes. A node can be defined as a connection point from which messages can be sent and received in order to communicate with the rest of the network. Sensor and actuator hardware can be attached to these nodes allowing for tasks to be performed. In our case, a node consists of a single M5Stack M5Stamp C3 development board [20]. This board is then able to interface with peripherals via multiple possible communication protocols (I2C for example). These nodes can have a number of network related features, namely: [30]

Relay Feature

Nodes that have the Relay Feature enabled are able to forward messages to other nodes on the network. Each message has a field called the Time-To-Live (TTL). Every time a message is relayed, the TTL counter is decreased by one. When the TTL reaches zero, the message will no longer be relayed. A message cache is used to make sure each message is only relayed once. The message cache and TTL counter are used to create an efficient flooding mechanism for message transfer by preventing messages from circling around in the network.

Proxy Feature

The proxy feature is used to create backwards compatibility for BLE devices without Bluetooth Mesh support. This allows a BLE device such as a smartphone to be connected with the Bluetooth mesh network.

Friend and Low-power Feature

Relay and Proxy nodes need to scan for new messages at all times to avoid missing them. This increases the overall energy usage of the node which can be undesired in battery-powered devices. This can be prevented by the use of the Friend and Low Power features. A low-power node can be assisted by a friend node for whom power consumption is less critical. A friend node has two functions. It stores incoming messages destined for the low-power node and it relays messages coming from the low-power node. This way the low-power node does not need to scan continuously, but can ask the friend node for messages at a specific interval. This allows the low-power node to save power since it will only need to power its radio periodically. The disadvantage of this strategy is that the low-power node cannot act as a proxy or relay. It is only able to communicate with friend nodes. This means that, since the mesh topology is implemented by using the relay feature, the network will be a mix of mesh and star topologies. With a star topologie being one or multiple low power nodes connected to a friend node.

Provisioning

A network can be created by a provisioner. This device generates the network and application keys needed for nodes to encrypt and decrypt messages. When a new node wants to join the network, the provisioner will give the device the network and application keys. The node will then be able to receive messages in the network.

3.2.3. Messaging

Nodes communicate via messages. Messages fall into two different categories, acknowledged and unacknowledged. Acknowledged messages require a response when received. This is used to return requested data and serves as a confirmation of reception. Unacknowledged messages do not require any response. Messages can also be subdivided into GET, SET and STATUS messages. GET messages request the state of a node. The node will respond with a STATUS message containing the requested state. SET messages can be used to change the state of a node. An acknowledged SET message will require a STATUS response. STATUS messages can also be send independently of other messages, for example, driven by a timer. [32]

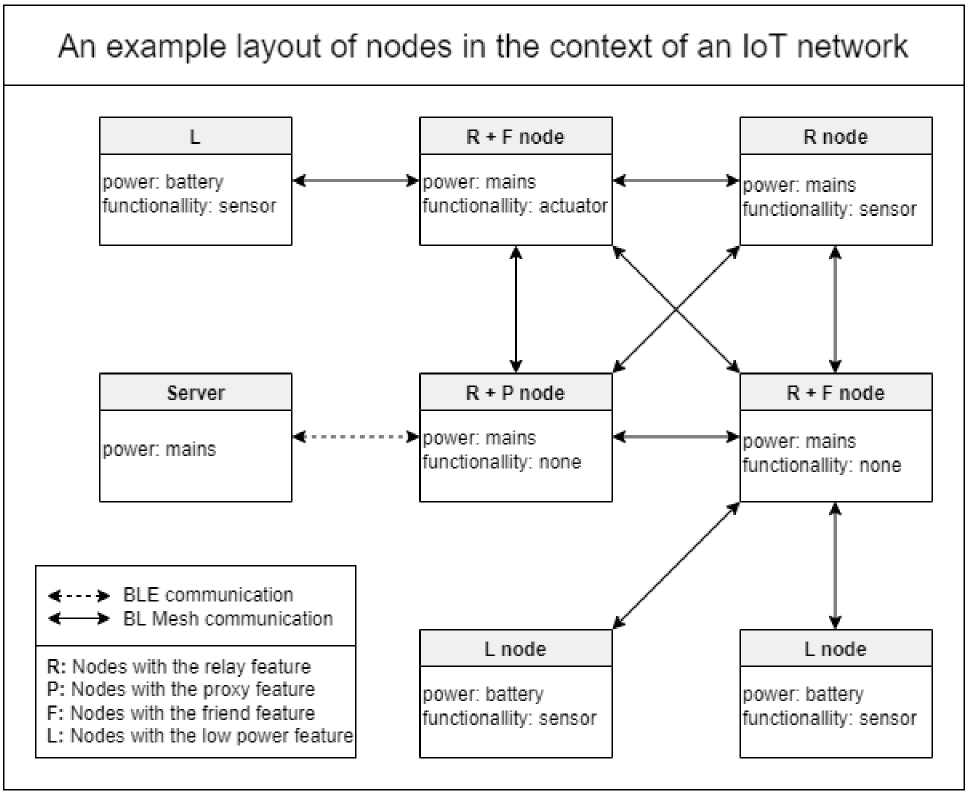


Figure 3.1: An overview of several node features in the context of an IoT network. [31]

Models

Messages always correspond with a model. Models are specifications for standard software components that determine what a device can do as a mesh device [29]. For example, there exists a sensor model which can send data over the network based on sensor readings. Models ensure a form of standardization and as a consequence, devices from other manufacturers will understand the received data. Opcodes are responsible for identifying the type of packet coming in. Each type of message has a specific defined opcode. Once a node reads the opcode of a message, it understands what type of data will be in the message and is able to decode it. In the provisioning process, the application key from the provisioner will be bound to all models of the new node. Without an application key, a model will not listen to data coming in.

States

Models contain states. States are data items that indicate the condition of the device, such as on/off or high/low. States may be simple, containing only a single value, or composite, containing multiple fields, similar to a struct in programming languages like C [29]. Each model has different kind of states that will be automatically recognized through the property ID. Property ID's are also standardized through the BLE Mesh protocol.

Publishing

In a BLE Mesh network, nodes communicate through publishing. Publishing can be explained as sending a message to all other nodes. Nodes always publish to an address (section 3.2.4). Other nodes are then able to subscribe to this address. By subscribing to an address, the node decides if it wants to listen to messages published on that specific address. When subscribed, it will be able to read the messages and process the data.

3.2.4. Addresses

Within the network three types of addresses are defined. Every node is uniquely identified by a unicast address. For communication via publishing, group addresses can be used. Nodes with the proxy, relay or friend feature have their own group address. Other group addresses can be defined and configured by the user via a configuration application. The third type of address is the virtual addresses, this address type wont be used in this project.

3.3. Hardware choices

BLE Mesh is chosen as communication protocol for the network. Research for a suitable microprocessor for the nodes needs to be done. As BLE Mesh is a rather new protocol (introduced in 2017 [29]), the choice of microprocessors is limited as older processors do not support BLE Mesh. A suitable microprocessor should come with an extensive amount of documentation available. Designing and creating own micro-controller boards is not part of this project. Therefore, a development board must be bought with a suitable micro-processor on it.
A requirement that must be met for the hardware choice is requirement **NM3**. Keeping the cost below €10,- is a hard boundary for the hardware. Other aspects are the documentation and the features of the board. A board preferably has peripherals built-in to develop with (lights, buttons, etc.).

nRF52840
The nRF52840 was released in 2017 [27]. It is built around the 32-bit ARM® Cortex™-M4 CPU with floating point unit running at 64 MHz. Bluetooth 5 is supported and so is BLE Mesh. An official IDE has not been developed for this microprocessor, but many different IDE's can be used with it. A large amount of documentation is available including for BLE Mesh. An example board with the nRF52840 is the Arduino Nano 33 BLE [33]. This board can be bought for around €22,- and is one of the cheapest options available.

ESP32-C3
The ESP32-C3 was introduced in 2020 [34]. The microprocessor offers support for Bluetooth 5 and thus BLE Mesh. An official IDE has been developed as an extension for the Eclipse editor [35]. This IDE contains scripts, documentation and, examples to help develop applications. An example board with the ESP32-C3 is the M5Stack M5Stamp C3 [36]. This board can be bought for around €7,- and is one of the cheapest options available.

3.3.1. Development board

The M5Stack M5Stamp C3 is chosen as development board, see fig. 3.2. It is cheaper then the Arduino Nano 33 BLE. Keeping the cost below €10,- makes sure mandatory requirement **NM3** is met. Additionally, the Espressif IDE adds examples and scripts that provide a useful framework to develop BLE Mesh nodes. Other advantages to the M5Stamp C3 are the included RGB LED and button. The LED can be used as an indicator while the button can be used as input for the board. Wi-Fi is also one of the capabilities of this board. This might be useful for future expansions of the system.

Figure 3.2: The M5Stack M5Stamp C3 [36].

4
Implementation and validation

This chapter discusses the implementation of the network based on BLE Mesh. First, the base of the network will be addressed. Next, multiple sections of additionally implemented features will follow. These features combined will form the base of the prototypes. After discussing the prototype implementations, the validation of the network will be discussed.

4.1. Tools used

The IDE used to develop the prototype is the ESP-IDF [37]. This IDE is specifically made for ESP chips like the one used. This IDE comes with a number of software components used to interface with internal and external peripherals found on the microprocessor. The specific component used for BLE Mesh is the ESP-BLE-MESH library [38]. This library comes with multiple examples of BLE Mesh implementations. Since coding BLE Mesh implementations from scratch consumes too much time for the scope of this project, an effort was made to adapt the given examples so they would perform the functions needed. Since the amount of code per node is already fairly sizable, and since the code is still a work in progress, the code is provided online on GitHub [39] and not as an Appendix.

4.2. General network implementation

A standard network implementation is built up from multiple nodes. These nodes are discussed within this subsection.

4.2.1. Provisioner node

The provisioner node forms the base of the network. This node has two main functions. First, it provisions the other nodes in order to make them part of the network as described in subsection 3.2.2. The current implementation of the provisioner can always provision an unprovisioned node, thus completing mandatory requirement **NM2**.
The second function of the provisioner node is managing the communication between the other nodes on the network and the server. This communication is done via a USB connection using serial communication. This means that the node reads out messages send over the network and decides what information should be sent to the server.

4.2.2. Sensors node

The smart classroom features of the prototype include measuring the humidity, temperature, and loudness. These features are implemented using sensor nodes. Sensor nodes are nodes used to communicate sensor data. Sensor nodes are able to read out an external sensor and communicate these values over the network. This is done by publishing the sensor data using the Sensor Server model as discussed in subsection 3.2.3. Other nodes are able to subscribe to the published sensor data using the Sensor Client model. The BLE Mesh protocol defines the transmitted type of sensor value as a sensor property ID. Each published sensor value contains a sensor property ID. The recipient can use this ID to figure out the type of sensor the data is from. These property ID's can be found in [40] and

define range, unit and encoding used to send the sensor data.
For example, sending the temperature from our temperature and humidity sensor is done using the Precise Present Ambient Temperature device property ID. This property ID corresponds to a data type, unit, step size, and range of values that can be found in Table 4.1. This theoretically allows the network to use any BLE Mesh temperature sensor using the same sensor property ID's since the way of sending this data is identical. A prototype implementation using this node will be discussed in subsection 4.3.1.

4.2.3. Indicator node

Indicator nodes are meant to indicate a state or event to a person. These indications can be done via light, sound, and vibration. Indicator nodes are able to subscribe to messages using the OnOff Client model. Normally, this model is only used to turn a single device on/off but since this on/off state is communicated using an 8-bit integer, the unused bits can be used to encode additional data. The way the indicator messages are encoded can be seen in Figure 4.1. Currently, these nodes are able to display a LED signal using 3 different effects as well as operate a vibrating motor.
Other nodes can send the indicator node messages by adopting the OnOff Server model and using this to publish their indicator message. Every node type can theoretically adopt this model.

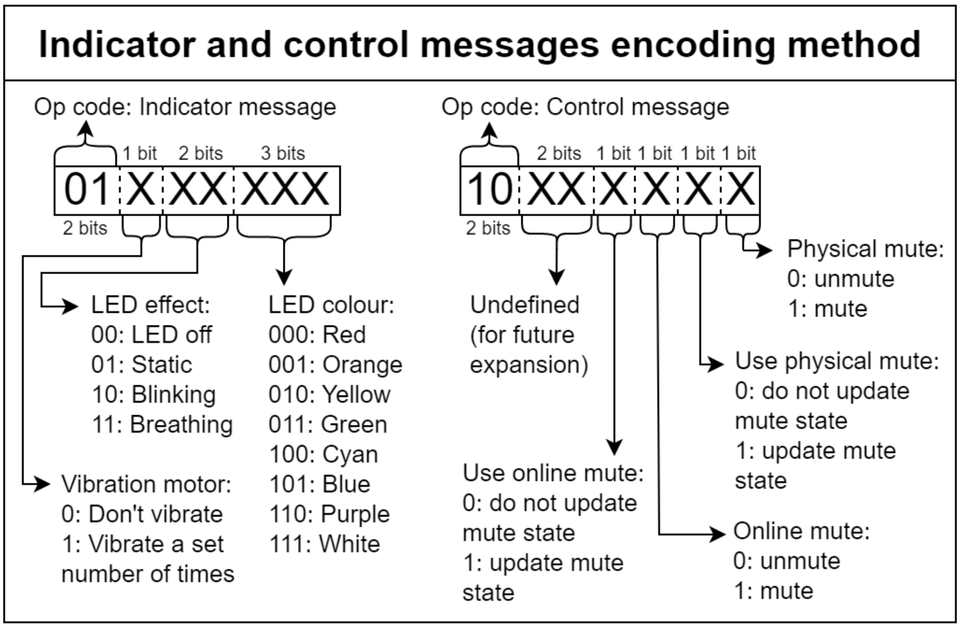


Figure 4.1: The encoding method used for sending 8 bit control and indicator messages. Here the OP code determines the message type.

4.2.4. Actuator node

The fourth node type added to the network are the actuator nodes. These nodes are meant to control an external device (for example a relay). Actuator nodes are able to subscribe to messages using the OnOff Client model. Using this model, these nodes receive the same messages as the indicator nodes. The node however, only reacts to control type messages defined in Figure 4.1. Different messages can be differentiated via their OP code. Currently, these nodes are able to operate a single relay via the physical mute part of the control message type. Just like with the indicator node, other nodes can send the actuator node messages by adopting the OnOff Client model and using this to publish their control message. Every node type can adopt this model.

4.2.5. PC nodes

The PC node is able to interface with programs running on a PC via serial using a USB connector. This allows for influencing nodes on the mesh network by relaying events from the PC. It can also influence programs on the PC by relaying messages from the mesh network. The exact use of this will be discussed in section 4.3.

4.2.6. Mesh

All of the previously discussed nodes have the relay feature enabled as discussed in subsection 3.2.2. This means that the network has full mesh capability. This also means that nodes can be added to the network purely for extending the network coverage or creating extra routing redundancy if required. Multiple sensor nodes of the same type can be added to make sure the network still has access to the sensor values in case one sensor node fails. This makes sure that losing a single node does not stop the operation of the network as long as there is sufficient redundancy. This means mandatory requirement **NM5** is met.

4.3. Prototype implementation

All of the previously discussed parts come together in the implementation. The first implementation that is discussed, is how the network handles communication of sensor values. The second implementation discussed will be the hybrid teaching prototype. Even tho these prototypes are discussed separately, the functionality of all prototypes can be combined.

4.3.1. Prototype wireless sensor implementation

The goal of this prototype is to communicate the values of multiple different sensor types wirelessly between the sensor board and the server. Other devices are able to access these sensor values on request.

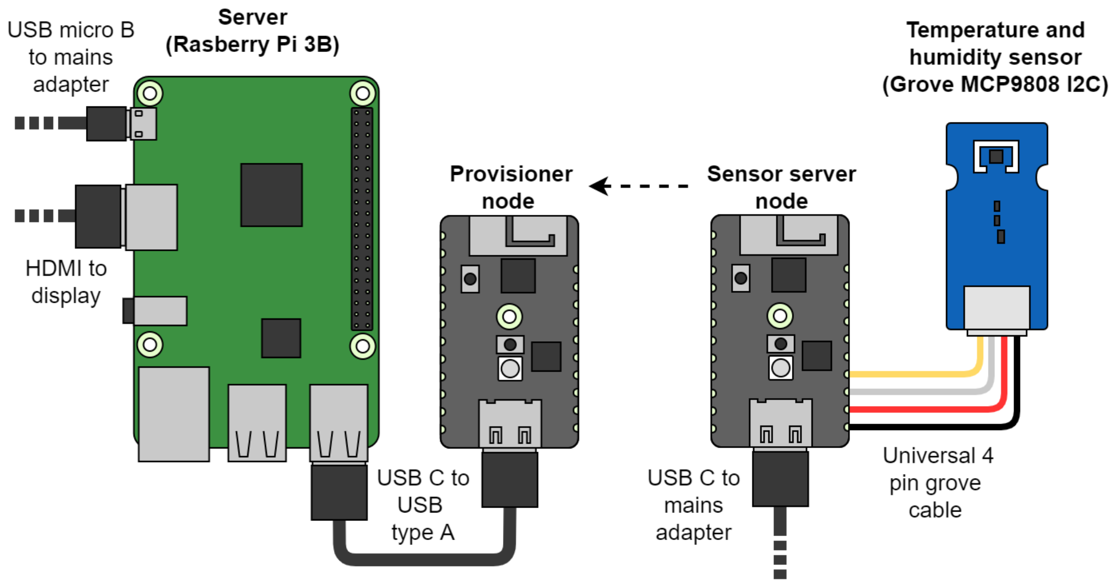


Figure 4.2: The temperature measurement setup used for the first prototype. Here the striped lines indicate messages published by the sensor server node.

The prototype is set up by powering on the Raspberry Pi as well as the sensor server board in no specific order. When first booting up, the sensor server node will be in an unprovisioned state indicated by a cyan blinking LED. Once the Raspberry Pi has power, the provisioner node is booted up and will start looking for devices to provision. When the provisioner node has found the sensor server, provisioning takes place. After the sensor node is provisioned, it automatically begins publishing sensor state values every five seconds.

The communication is done using the sensor property ID's standard as discussed in subsection 4.2.2. This means that all different sensor values are communicated by their property ID's. The data type, step size, range, and units the used property ID's correspond to are shown in Table 4.1. These values are published over the BLE Mesh network using the Sensor Server model. The provisioner node uses the Sensor Client model subscribed to these messages to receive the data. The Sensor Client is able to subscribe to multiple sensors that are publishing at the same time. Thanks to the property ID communicated when sending sensor values, the receiving node always knows what type of sensor these values are from. A full overview of the communication between different hardware elements when using a temperature and humidity sensor can be found in Figure 4.3. The hardware setup for reading out the loudness sensor is almost identical with the only difference being the sensor type attached using the four pin grove cable.

The prototype is able to successfully communicate sensor values thus mandatory requirement **M6** is met.

4.3.2. Prototype hybrid teaching implementation

The goal of the hybrid teaching prototype is to notify the teacher of questions from the online audience. The system should also be able to relay the audio of questions from the physically present students to

Property ID	Data type	Step size	Range
Precise Present Ambient Temperature	16 bit signed scalar	0.01 °C	-327.68 to 327.67 °C
Present Ambient Relative Humidity	16 bit unsigned scalar	0.01 %	0 to 100.0 %
Present Ambient Noise	8 bit unsigned scalar	1 dB	0 to 253 dB

Table 4.1: The property ID's and their corresponding data types, step sizes, and range.

the online students and vice versa. Both these functions were implemented using five different nodes.

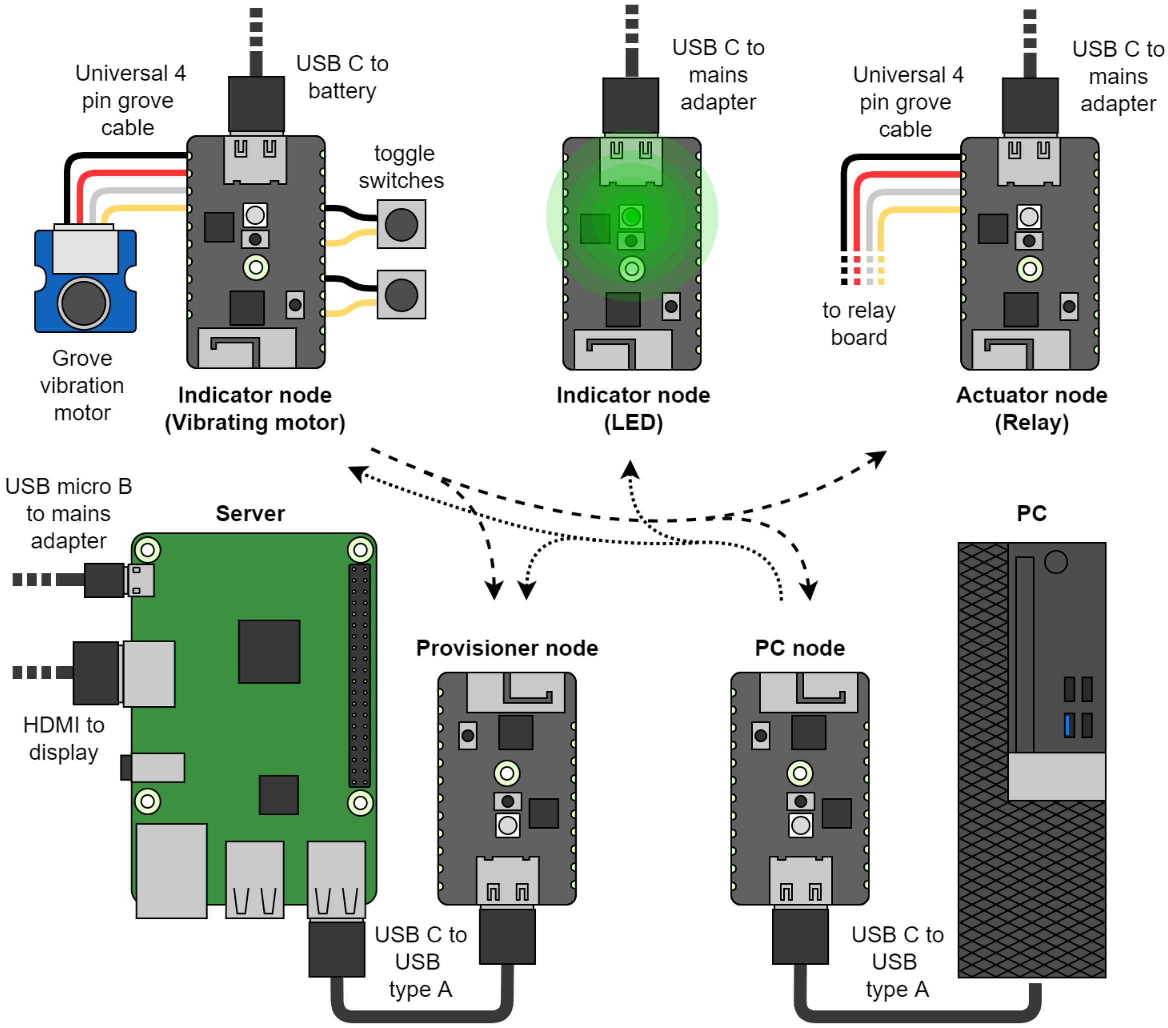


Figure 4.3: The hybrid teaching setup used for the prototype. Here the striped lines indicate messages published by the Indicator node with vibrating motor and the dotted line indicates messages published by the PC node.

The indicator node with vibration motor

The indicator node is meant to be physically attached to the teachers digital laser Pointer used to control the slides. It will add a vibration motor as well as two toggle switches to the device. Once the PC node detects a question from an online student, the vibration motor will vibrate three times. This notifies the teacher of the online question. The two toggle switches are to toggle mute functions. There is one mute function for the physically present students, and one for the audio of the online students. The mute signal for the online students will be published to the PC node. The mute signal for the physically present students will be published to the actuator node.

The indicator node with LED
Once the PC node detects a question from the online student, it will not only be indicated using the vibration motor, but also using a LED based indicator node. This node will display a green colour when no question is detected and a yellow coloured breathing effect when a question is detected.

The actuator node with relay
The actuator node is able to be controlled by the indicator node with vibration motor in order to mute the physically attending students for the online students. This is done by using a relay to switch an audio signal coming from a microphone.

The provisioner node
This node provisions all unprovisioned devices. It also subscribes to all different messages that are published over the network and sends them to the server over serial for logging purposes.

The PC node
The PC node is used to publish a message indicating there is an online question. In order to do this, the node is attached to a PC running videoconferencing software like Microsoft Teams. Once a student has a question, they can use a built in function to raise or lower their hand. The raising of one's hand is visually picked up by a program continuously scanning for the raise hand icon. Once the program has detected this, a message will be sent to the PC node via serial which causes the PC node to publish on the network. Once the program detects the hand being lowered, this too will be published in the same manner.
The second function of the PC node is being able to mute the online students for the physically attending students. This is done by subscribing to the published messages from the indicator node with vibration motor. Once a message intended to mute or unmute is received, the PC node will communicate this via serial to the PC. A program running on the PC will then mute the Microsoft Teams application.

4.4. Validation

General testing methodology
Validation of the system mainly consisted of testing whether every node could receive the messages it subscribed to and publish the messages the node is meant to communicate. This was done by creating debug log messages that could be accessed by connecting the development board to the program environment over serial. This logs every message send and every message received.

Indicator and actuator node testing
The indicator node was tested by sending all possible indicator messages to it. The behavior of the node was then checked for every message and turned out to be as expected. The same was done for the actuator node.

Network reliability
In order to validate the network reliability 100,000 messages containing a specific 8 bit code were sent from one node to another node at a distance of two meters. Fifteen messages were sent per second. The message counted as successfully received if the received binary code matched the transmitted binary code. Around ten messages were send per second. Out of the 100,000 messages, 99,861 were received, resulting in an above 99% success rate. This means mandatory requirement **NM6** was met.

Three nodes were used in order to test a more challenging scenario with relaying, increased distances, and obstacles. Two nodes were places inside a hallway fourteen meters apart. The other node was places inside an office space adjacent to one of the hallway nodes. This node was at a distance of seven meters from the adjacent hallway node.
The furthest node from the office sent a total of 1,000 messages. The node adjacent from the office relayed these messages. The node inside the office received these relayed messages. In the first experiment, fifteen messages per second were sent. This resulted in 91.9% of the message being received. When the amount of messages sent per second was decreased to five, 97.9% were received. This shows the amount of messages sent over a period of time can have a significant influence on the transfer reliability. The proposed prototypes inside a classroom will have an environment with less obstacles, closer nodes, as well as more nodes able to relay messages. Furthermore, the network will have to support less then five messages per second for both prototypes combined. This is why 97.9% is taken as an absolute worst case scenario for message transfer success rate for the proposed classroom based prototypes in section 4.3. This estimate is based on a single relay node being required between the sending and receiving node. If this were not needed and the message could be directly transferred, the worst case success rate would likely increase.

4.4.1. Future validation

At the time of writing the hybrid teaching prototype is not fully finished. When this is the case, another important validation aspect would be to gather feedback from educators making use of the system. This would be needed in order to assess to what degree the prototype was able to fulfill the hybrid teaching part of the project goal, and what improvements still have to be made.

5 Discussion and Conclusion

In section 1.3, the problem was defined as:

How to design a complete and expandable IoT system that implements a smart classroom framework and improves hybrid education?

The survey (Appendix B) made it clear that teachers experience difficulties giving attention to both the physical present and the online group of students simultaneously during hybrid teaching. To solve this problem, the indicator node with LED light and the indicator node with vibration motor will warn the lecturer about questions from students online.
Another issue in hybrid teaching is focused on audio. Questions asked in the classroom can not be heard by online students, and questions asked online could not be heard in the classroom. The indicator node with vibration motor, the actuator node with relay and the PC node together solve this problem. By using the toggle switches, the audio from the online students and the audio from the physical present students can be controlled. This gives the lecturer the opportunity to mute/unmute both groups.
Next to the hybrid teaching solutions, the system also provides opportunities for other smart classroom features. Different sensor and actuator nodes can be connected to the system. The server node is able to store collected data and make it available for external devices. This creates a complete framework for a smart classroom ready for a wide range of possible smart classroom applications.
Different items of discussion are stated below. The sections contain features that are either not fully within the scope of the project but still relevant, or features that can still be improved upon.

5.1. Security

Security is an important part of digital systems. While security is not in the scope of this project, it should still be considered for a final product. By choosing BLE Mesh as a networking protocol, the IoT devices make use of a private network for communication. If Wi-Fi would have been used for this purpose, other non-IoT devices would have likely been connected to the same network. This introduces a security risk since external devices can attempt to compromise the IoT devices sharing the same network. The BLE Mesh network can not be joined by external devices when provisioning is disabled after all devices have joined the network, this decreases this risk.
BLE Mesh supports additional features which protect the network against threats. Communication of the network is encrypted using Elliptic Curve Diffie-Hellman (ECDH) encryption. This prevents new nodes from joining the network until they are provisioned by the provisioner. It also protects the network from devices that try to intercept messages. Without the correct network and application keys, devices will not be able to decrypt the messages. Next to this, messages are sent with a sequence (SEQ) number. By keeping track of this sequence number, unknown devices that try to replay certain messages in the network are effectively ignored. [41]
The features described above protect a BLE Mesh network from external devices trying to interfere in the network. The smart classroom system will not contain privacy sensitive data in the proposed setup, though the system might be expanded later with more sensors or actuators that do make use of privacy sensitive data. A future requirement would be to look further into the security of the system.

5.2. Low power

As mentioned in chapter 3, BLE Mesh has special feature for battery-powered devices. This low-power feature aims to make battery-powered devices more efficient. This implementation is available in our system, but it will not be used for the proposed classroom setup. The low-power feature works by letting a device sleep for a specific amount of time. This means that messages could be received with a delay. The battery-powered devices in the proposed setup consist of indicators as can be seen in chapter 4. The indicators should respond as quick as possible in the case of an event to ensure that the interaction with the online audience remains. These devices are therefore not suitable to use the low-power feature.
For future expansions, this feature could prove useful. It will increase battery-life for devices which can afford delays in the communication.

5.3. BLE Mesh on Raspberry Pi

In the proposed setup, the Raspberry Pi (RPi) is using a serial connection to a M5Stamp C3. The M5Stamp C3 is then used as a BLE Mesh node for the RPi. The node handles the wireless communication and the RPi can access data through the serial connection. The microprocessor on the RPi, however, is capable of supporting BLE Mesh itself [42]. As BLE Mesh is a rather new protocol (from 2017 [43]), supporting packages for the Raspberry Pi are not readily available. Because of the given time for this project, the choice was made to use a serial connection to a M5Stamp C3 node.

5.4. Plug-and-play

A problem with the current design is that nodes stay provisioned when the provisioner is reset. This results in the provisioner not being able to provision the other nodes after a reset since the other nodes are already provisioned. The issue with this is that the provissioner node is no longer able to communicate with the rest of the network. At the time of writing, the only way to resolve this is resetting all other nodes every time the provisioner is reset. One of the solutions would be to let all nodes check whether the provisioner node is still online, and if this is not the case, reset themselves. The issue with this solution is that this will introduce a single point of failure in the network, namely the provisioner node. A balance will have to be struck between redundancy and ease of use regarding this point. One could state the trade of requirement **NT1**, stating the ease of use should be maximized, could be improved upon.

5.5. Future work

The system is designed to be expandable and customizable. One of the ways this can be done is adding more sensors and actuators to the system. For example, lights and electronic blinds could be controlled based on daylight sensors. The indicator node with vibration motor could also be expanded with more buttons that turn on different teaching scenes. The hybrid education prototype could also be further improved by adding lecture recording features. An example of these features would be automatically switching between a livestream of the lecturer, physically present students, and online students.

A

Power consumption estimation of BLE

When looking at wireless sensors, radio communication accounts for the largest power usage in most cases. [44] This is why estimating the energy spend on radio transmissions is essential for estimating the lifetime of a battery powered sensor node.

The low power features discussed in subsection 3.2.2 are implemented by decreasing the duty cycle of the radio receiver. This is done by introducing a receive delay and a receive window. Receive delay is the duration where the low power node sleeps. After the receive delay has passed the node will listen for new messages for the receive window duration. The sum of the receive window and the receive delay is the polling timeout. Figure A.1 shows the relation between the current consumption of the device, the receive delay and the polling timeout. The polling timeout for most nodes in our network is likely to be around 10 s.

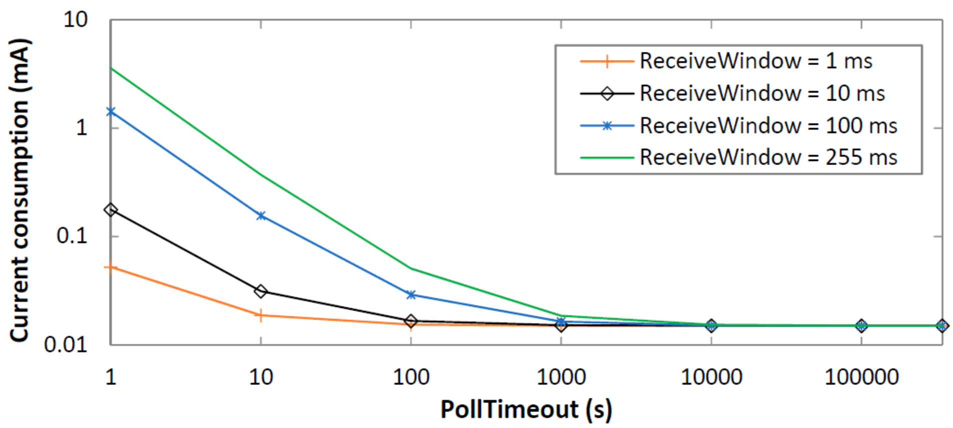


Figure A.1: Average current consumption of the Bluetooth mesh node with low power feature, as a function of poll timeout, in absence of data message transmissions, and for various receive window settings. Source: [43]

Figure A.1 Does not take the current consumption for transmitting messages into account. Figure A.2 shows that transmitting messages at a polling timeout of 10 s has negligible effect on the worst case current consumption. Worst case being a receive window of 255 ms.

For a poll timeout of 10 s, the current consumption is 371 $\mu\mathrm{A}$ for the receive window of 255 ms. [43]. This result was based on the PCA10028 Development Kit using the nRF51422 chipset. The proposed implementation uses the M5Stack M5Stamp C3 using the ESP32 C3 chipset [20]. The differences of the development boards as well as the chipsets could influence current consumption. This will be accounted for by adding a margin of error to our previous worst case current consumption of 371 $\mu\mathrm{A}$. The final worst case current consumption of a low power node used for the design will be estimated at 1 mA. This does not include any current draw that is introduced by adding peripherals to a node like sensors.

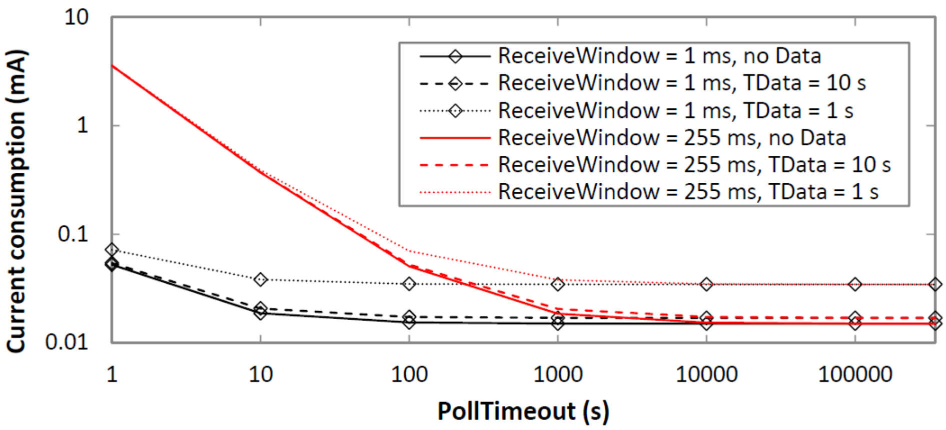


Figure A.2: Average current consumption of the Bluetooth mesh node with low power feature, as a function of PollTimeout, the time between message transfers, and receive window settings of 1 ms and 255 ms. Source: [43]

B

Survey results

What problems do you encounter with hybrid teaching?

1. That both groups expect to be the number 1 audience, while as teacher it is impossible to serve both groups equally. As an effect both groups are to some extend disappointed with the experience. It works fine if one audience is the target and the other group is either watching in or is 'audience'.
2. Chalkboards are not readable because of the light coming in. Often the entire board is filmed which makes text too small to read. The camera does not follow the teacher well.
3. Complications setting up cameras, smartboards and meetings and keeping an eye on the online participants during the lecture.
4. Lack of feedback, questions or participation from students connected online.
5. Students online do not engage. Focus is on students physically present.
6. Focus is to 1 group only, in most cases the ones that are present
7. Making the online people feel connected to the onsite group.
8. Hard to keep track of questions both in the room and online.
9. Serving two audiences at the same time.
10. Dividing my attention.

In what way could technology assist with or solve these problems?

1. A digital board that is streamed directly or writing on a tablet. A solution for following the teacher is unknown to me. I do not know if it is great to have the teacher stay in a fixed spot.
2. A different screen for projecting the students camera that join online, a way to share the white board with the students online.
3. Work for one of the target groups only and improve the experience of the secondary group using technology.
4. Perhaps having a large screen (tv sized) in the audience to see the students at home could help.
5. If VR would be good enough that people online has the same perspective as in the room.
6. Less technology would be preferable (i.e. a proper blackboard) but a quality camera.
7. An up-voting mechanism for important questions.

What could be improved about hybrid teaching?

1. The quality of the sound and video streaming and setting it up. When managed from outside (for instance in aula) this works very well, but the quality is much worse when the lecturer has to set it up themselves.
2. Large smartboards which can be streamed directly (instead of via a camera) or good writing tablets of which the screen gets streamed and projected.
3. No smart boards, but traditional smart boards so you only need one technology (online meeting+camera) to function.
4. Balanced presence and attention of online and oncampus students.
5. Participation of students joining online.
6. Connection with those online.
7. Interaction with each other.

In an ideal world with endless possibilities, what would be your vision on a hybrid classroom?

1. Either TV show setting where you do your class for the remote participants and the co-located ones are audience - as such they even have a lesser experience than the people at home, so we should give them all the tools and add ons that the people at home have too. OR we make co-located session where the people remote join in in such a way that all the things they miss from the co-location are compensated. I like the Getmibo townhall setting as example where remote environments bring co-located features.
2. A camera which follows the teacher automatically and a smartboard as big as an old school chalkboard with a second screen of the same size so you can shove the empty screen in front of the full screen.
3. All students turn on the cameras, a different screen for showing the students connected online so we 'feel' that also students joining online are actually there.
4. Just very high quality video and audio (but I still prefer the offline interaction, so I'd still prioritise those in the room).
5. Have the online people in the same line of vision as the onsite people. Like the holodeck.
6. If VR would be good enough that people online has the same perspective as in the room.
7. Blackboard+good camera+screen for feedback with online students.

What problems do you encounter in a smart classroom?

1. Every system is different and I use it not often. As such the UI always makes you wonder how it works.
2. Sometines setting up the systems is really difficult.
3. Smart boards not working
4. Technology not working.

In what way could technology assist with or solve these problems?

1. Replace smartboards with proper blackboards
2. Have less malfunctions.
3. Voice response

What could be improved about smart classrooms?

1. Often the controls are fixed to the desk and the desk as such needs a fixed spot. Before you know it the desk becomes THE central place in front, which it is not but cannot moved away.

In an ideal world with endless possibilities, what would be your vision on a smart classroom?

1. Fully automatice - no display - maybe voice controlled

2. It is easier to set up all the systems.

C

On board LED colours

The onboard LED of the nodes is used to indicate information about the node itself and the network in general. The meaning of the colours is shown in Table C.1.

Colour	**Meaning**
Red	Error
Orange	Warning
Yellow	Attention needed
Green	All OK
Cyan	Unprovisioned
Blue	In the process of doing something
Purple	Provisioning
White	None

Table C.1: The onboard LED colours and their corresponding meaning

Bibliography

[1] Kshitij Shinghal et al. “IOT Based Modified Hybrid Blended Learning Model for Education”. In: *2020 International Conference on Advances in Computing, Communication Materials (ICACCM)*. 2020, pp. 229–232. DOI: `10.1109/ICACCM50413.2020.9213049`.

[2] La Vonne Fedynich. “Teaching beyond the classroom walls: The pros and cons of cyber learning.” In: *Journal of Instructional Pedagogies* 13 (2013).

[3] V Giampietro M. Detyna R Sanchez□Pizani and E.J. Dommett. “Hybrid fexible (HyFlex) teaching and learning: climbing the mountain of implementation challenges for synchronous online and face□to□face seminars during a pandemic”. In: (2022). URL: `https://doi.org/10.1007/s10984-022-09408-y`.

[4] P.C. Blumenfeld J.A. Fredricks and A.H. Paris. “School Engagement: Potential of the Concept, State of the Evidence.” In: (2004). URL: `https://doi.org/10.3102/00346543074001059`.

[5] Annelies Raes. “Exploring Student and Teacher Experiences in Hybrid Learning Environments: Does Presence Matter?” In: (2021). URL: `https://link.springer.com/content/pdf/10.1007/s42438-021-00274-0.pdf`.

[6] Jo Ann Lee & Patricia Ellen Busch. “Factors Related to Instructors Willingness to Participate in Distance Education.” In: *The Journal of Educational Research* (2005). URL: `https://doi.org/10.3200/JOER.99.2.109-115`.

[7] *The Netherlands ranks among the EU top in Digital Skills*. Feb. 2020. URL: `https://www.cbs.nl/en-gb/news/2020/07/the-netherlands-ranks-among-the-eu-top-in-digital-skills`.

[8] Dosheela Devi Ramlowat and Binod Kumar Pattanayak. “Exploring the Internet of Things (IoT) in Education: A Review”. In: *Information Systems Design and Intelligent Applications*. Ed. by Suresh Chandra Satapathy et al. Singapore: Springer Singapore, 2019, pp. 245–255. ISBN: 978-981-13-3338-5.

[9] Sciforce. “Internet of Things for the Classroom”. In: (2019). URL: `https://www.iotforall.com/internet-of-things-classroom`.

[10] *Logitech Rally Kit(960-001218) Datasheet*. `https://cdn.webshopapp.com/shops/91456/files/319561589/logitech-rally-product-data-sheet.pdf`. Accessed: 30/05/2022.

[11] *vc520pro2 datasheet*. `https://www.averusa.com/business/downloads/vc520pro2-datasheet.pdf`. Accessed: 30/05/2022.

[12] *A BRIEF GUIDE TO AUDIO CONFERENCING IN HIGHER ED CLASSROOMS*. `https://www.parmetech.com/wp-content/uploads/2021/09/Guide-to-Audio-Conferencing_-Classrooms_-Education_-Cobranded.pdf`. Accessed: 30/05/2022.

[13] *Chung Yuan Christian University Establishes World’s First ViewSonic Hybrid Teaching Classroom*. `https://www.viewsonic.com/nl/education/pdf/Case_Study_Chung_Yuan_Christian_University_Establishes_Worlds_First_ViewSonic_Hybrid_Teaching_Classroom.pdf`. Accessed: 30/05/2022.

[14] L. Belli A. Cilfone L. Davoli and G. Ferrari. “Wireless Mesh Networking: An IoT-Oriented Perspective Survey on Relevant Technologies”. In: *Future Internet* 1 (Apr. 2019), p. 1.

[15] *Auditorium*. URL: https://esviewer.tudelft.nl/space/113/.

[16] S. Shahrestani M. Elkhodr and H. Cheung. "Emerging wireless technologies in the internet of things: a comparative study". In: *IJWMN* 8 (Oct. 2016), p. 7.

[17] Olubiyi O. Akintade Olaide O. Kazeem and Lawrence O. Kehinde. "Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of Things". In: *International Journal of Internet of Things* 6 (2017), pp. 9–13.

[18] Salim Jibrin Danbatta and Asaf Varol. "Comparison of Zigbee, Z-Wave, Wi-Fi, and bluetooth wireless technologies used in home automation". In: *2019 7th International Symposium on Digital Forensics and Security (ISDFS)*. IEEE. 2019, pp. 1–5.

[19] Wenhui Chen, Sangho Jeong, and Hoekyung Jung. "WiFi-Based home IoT communication system". In: *Journal of information and communication convergence engineering* 18.1 (2020), pp. 8–15.

[20] *ESP32-C3 Family*. ESP32-C3. Rev. 1.0. Espressif Systems. 2021.

[21] N. Sornin. *LoRaWAN 1.1 Specification*. 1.1. manual. Oct. 2017. URL: https://lora-alliance.org/wp-content/uploads/2020/11/lorawantm_specification_-v1.1.pdf.

[22] Ayanle I Ali et al. "ZigBee and LoRa based wireless sensors for smart environment and IoT applications". In: *2019 1st Global Power, Energy and Communication Conference (GPECOM)*. IEEE. 2019, pp. 19–23.

[23] Christos Bouras et al. "Using LoRa technology for IoT monitoring systems". In: *2019 10th International Conference on Networks of the Future (NoF)*. IEEE. 2019, pp. 134–137.

[24] A. Miguel Garcia E. Bäumker and P. Woias. "Minimizing power consumption of LoRa and LoRaWAN for low-power wireless sensor nodes". In: *Journal of Physics: Conference Series* 1407 (2019), p. 10.

[25] *RFM95/96/97/98 - Low Power Long Range Transceiver Module*. RFM98. Rev. 1.0. Hope RF. 2016.

[26] *ZigBee Specification*. r21. manual. Aug. 2015. URL: https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf.

[27] *nRF52840*. nRF52840. Rev. 1.1. Nordic Semiconductor. Feb. 2019.

[28] C. Blondia A. Kumar Sultania C. Delgado and J. Famaey. "Downlink Performance Modeling and Evaluation of Batteryless Low Power BLE Node". In: *Sensors* 22 (2022), p. 8.

[29] Martin Woolley. *Developer Study Guide: An introduction to Bluetooth Mesh Networking*. Tech. rep. Bluetooth SIG, June 2021. URL: https://www.bluetooth.com/bluetooth-resources/bluetooth-mesh-developer-study-guide/?utm_campaign=mesh&utm_source=internal&utm_medium=blog&utm_content=bluetooth-mesh-developer-study-guide-v2.0.

[30] Mathias Baert et al. "The Bluetooth mesh standard: An overview and experimental evaluation". In: *Sensors* 18.8 (2018), p. 2409.

[31] Raúl Rondón et al. "Understanding the performance of bluetooth mesh: reliability, delay, and scalability analysis". In: *IEEE Internet of things journal* 7.3 (2019), pp. 2089–2101.

[32] Martin Woolley. *Bluetooth Mesh Networking*. Tech. rep. Bluetooth SIG, Dec. 2020. URL: https://www.bluetooth.com/wp-content/uploads/2019/03/Mesh-Technology-Overview.pdf.

[33] Reichelt. *Reichelt*. 2022. URL: https://www.shorturl.at/hrDSW.

[34] *ESP32-C3*. Espressif. URL: https://www.espressif.com/sites/default/files/documentation/esp32-c3_technical_reference_manual_en.pdf.

[35] Eclipse Foundation. *Eclipse*. 2022. URL: https://www.eclipse.org/.

[36] Distrelec. *Distrelec*. 2022. URL: https://www.distrelec.nl/nl/m5stamp-c3-development-board-with-pin-headers-m5stack-k056/p/30236865?ext_cid=shgooaqnlnl-Shopping-CSS&gclid=CjwKCAjwv-GUBhAzEiwASUMm4i3pzGoNNs-IU8VMh5-xVQRFg-Di6UbbIW_J1gXmZ_doJPd0KpXu4RoCWGUQAvD_BwE&gclsrc=aw.ds.

[37] *IDF versions*. URL: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/versions.html.

[38] *Ble-Mesh*. URL: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-guides/esp-ble-mesh/ble-mesh-index.html.

[39] Mathijs van Binnendijk and Ciarán Lichtenberg. *IoT-Bluetooth-mesh-network*. https://github.com/Mathijsvb/IoT-Bluetooth-mesh-network. 2022.

[40] Gerard Harbers. *Mesh Device Properties*. Tech. rep. v2. Bluetooth SIG, Sept. 2020. URL: https://www.bluetooth.com/specifications/specs/mesh-device-properties-2/.

[41] Evgeny Kalinin et al. “IoT security mechanisms in the example of BLE”. en. In: *Computers* 10.12 (Nov. 2021), p. 162.

[42] *Raspberry Pi 3 Model B+ Datasheet*. URL: https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf.

[43] Seyed Mahdi Darroudi, Raül Caldera-Sànchez, and Carles Gomez. “Bluetooth mesh energy consumption: A model”. In: *Sensors* 19.5 (2019), p. 1238.

[44] Olaf Landsiedel, Klaus Wehrle, and Stefan Gotz. “Accurate prediction of power consumption in sensor networks”. In: *The Second IEEE Workshop on Embedded Networked Sensors, 2005. EmNetS-II.* IEEE. 2005, pp. 37–44.