Sapienza University of Rome

Human Computer Interaction on the Web — A.Y 2023/2024

M.Sc. in Computer Science

Windy: A Device for Air Quality Monitoring

Authors:
Barreto Diego
Palmieri Simone
Piarulli Lorenzo
Marincione Davide
Zirilli Alessandro

July 17, 2024

Contents

Chapter 1

Introduction

Clean air is essential to our health and to the environment. However, due to human activities causing polluting emissions, air quality has deteriorated considerably. These activities are notably linked to industry, energy production, domestic heating, agriculture and transport. According to WHO every year 6.7 millions people die prematurely due to air pollution. As one may be tempted to think that our houses are a safe place, it is not the case. Still too many households use polluting fuels to cook and technologies in and around the home that contains a range of health-damaging pollutants, including small particles that penetrate deep into the lungs and enter the bloodstream. Poor ventilation and outside factors make the problem worse. As citizens awareness is growing, and governments are taking actions we are years away from a worldwide consciousness of the problem. Air pollution is subtle, cannot be seen either smelled, and its effects are not immediate. We need to "amplify" our senses to defend against this silent killer. Superpowers are not in sight, but IoT and sensors are.

In this report we provide an overview on the process that lead us to the creation of our solution. Motivated by our concerns about our houses' situation we conducted extensive need finding to understand if our colleagues shared the same thoughts. We than spread a survey via Google Forms to gather more data. Since the results were clear, people are worried about the air they breathe, but they are passive in taking actions, and they mostly didn't know how harmful indoor air pollution can be, we started designing. Having in mind the needs registered we ended up with a cheap, portable and complete solution, that not only can monitor air polluting agents, using certified sensor, but that also, thanks to the associated companion app, it gives the user a complete overview of the situation in a smart and easy way.

In the following chapters we are going to examine process' details as well as the current state of the system, competitors and limitations.

1.1 Competitors

On the market, there exist multiple solutions to monitor air quality:

Figure 1.1: Birdie Fresh Air Monitor

- Amazon Basic's Air Quality Monitor is the device that comes closest to our solution. It is a portable device that can monitor the air quality in a room, and can detect a number of pollutants. It can be connected to the user's smartphone via Bluetooth and the data can be visualized in the app. However, the device will set the user back 80 euros, which is a lot more than our solution (which costs around 30 euros).
- Birdie Fresh Air Monitor 1.1 is more of a design object than a functional device. It is a small bird-shaped device that moves its body to indicate the air quality in the room (when the bird is in an upright position, the air quality is good, when it goes downside, it is bad). It is truly a remarkable piece of design and a clever metaphor, but it costs around 200 euros. Also, it is meant to be wall-mounted, and not carried around.

Chapter 2

Need Finding

2.1 The basic idea

This project was born from the realization that, essentially, there exists no cheap and easy way to monitor the quality of the air in one's own personal space. One can find out the quality of the air in a given area, as Google Maps has a specialized view for that. But, for example, if one wanted to know the quality of the air in their own home, or in their office, they would have to buy specialized equipment, which can be quite expensive. The first thought that came to our mind was to create a portable device, and its companion app, that would continuously monitor the users' surrounding space, and alert them in the case that the air quality was too bad. Therefore we set out to find out the details of the problem, and to see if people would actually be interested in something like this.

2.2 Interviews

After a brainstorming session, we produced a set of interview questions:

1. How much time do you think you spend exposed to bad air?
2. Which means of transport do you use most often?
3. How much do you care about the quality of the air you breathe?
4. Do you know of any parameters that measure the air quality? (If so, which of them do you care most about?)
5. What do you think is the most common source of bad air?
6. Would you be willing to change your habits in order to avoid bad air?
7. Is the air in your home or office monitored in any way?
8. Would you be willing to wear a device that monitors the air quality around you?

9. In which situations would you be most likely to use a device like this?
10. There are many devices that sanitize the air in a room. Do you know them? Do you have any?

We then proceeded to make a total of 17 interview sessions around Sapienza's campus, because of this, many sessions contained multiple people, totaling to around 30 interviewees.

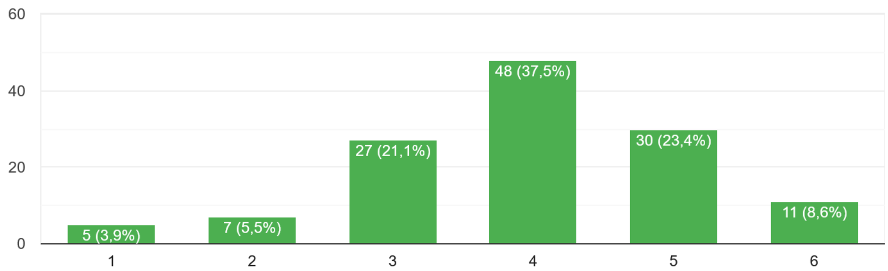
From these, we gathered the following insights:

- People are generally concerned about the quality of the air that they breathe, but they are passive about it. They don't often seek information nor do they take any action to avoid bad air.
- People's concerns are poorly placed. As most people think that the air in their homes is cleaner than the air outside, but in reality, the air inside is often worse for one's health. As outside air is constantly being renewed, while inside air is often stagnant.
- Outside CO2 concentration, people don't know what other parameters are important for air quality.
- More than wearing a device, people prefer to have the possibility to have the device attached to something that they already carry with them, like their keys or bag.
- People are willing to keep such a device, provided that it is not too big, doesn't require attention, and is not too expensive.

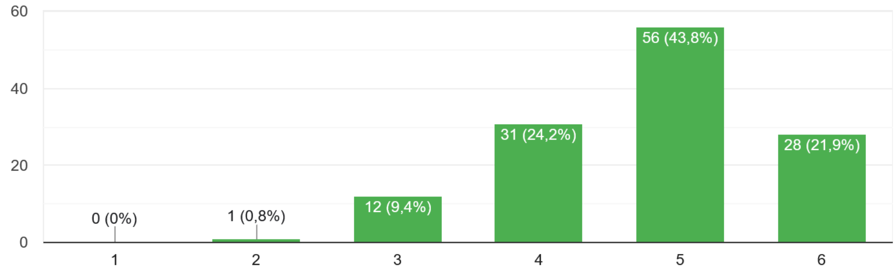
Therefore, we reasoned that, in order to make something that people would actually use, we would have needed to make something discreet, cheap, and easy to use (and understand).

2.3 Questionnaire

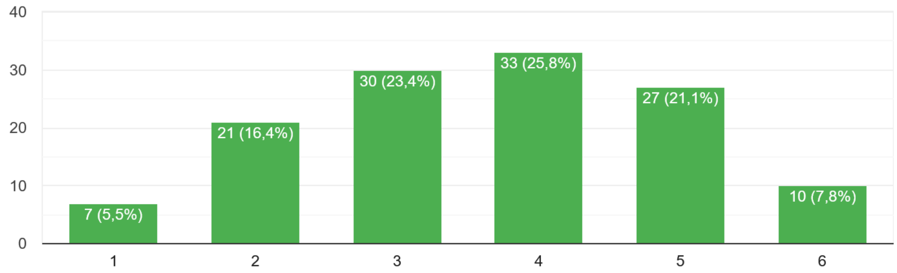
After the interviews, we decided to create a questionnaire to gather quantitative data. We created a Google Form, and shared it on our social circles, and student's groups. From this, we gathered 128 responses.



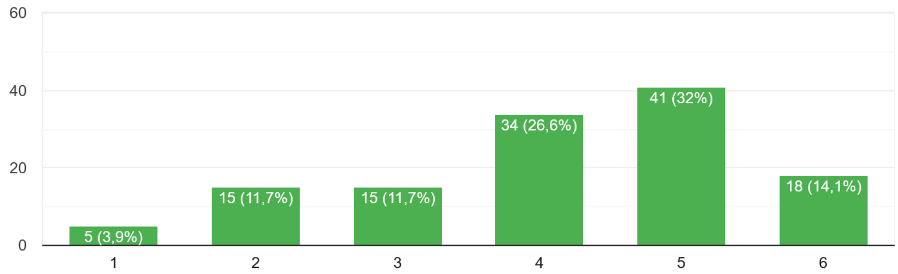
1. 'How much does air quality matter to you?': as we can see, most people care about their air quality, but not too much.



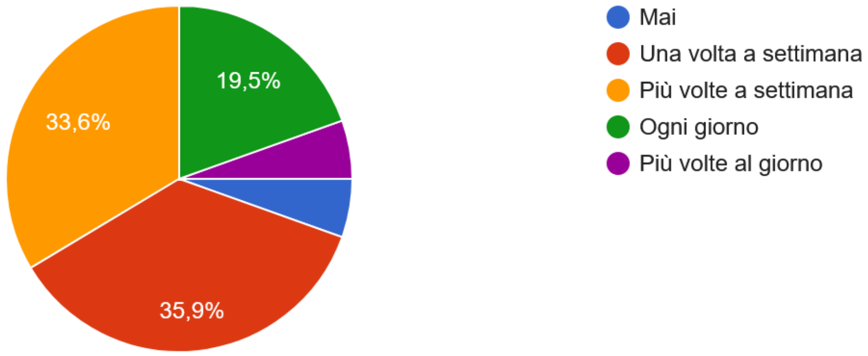
2. 'How much do you think the air that you breathe is polluted?': reinforcing our interviews, we can see that people are aware that the air might be quite bad.



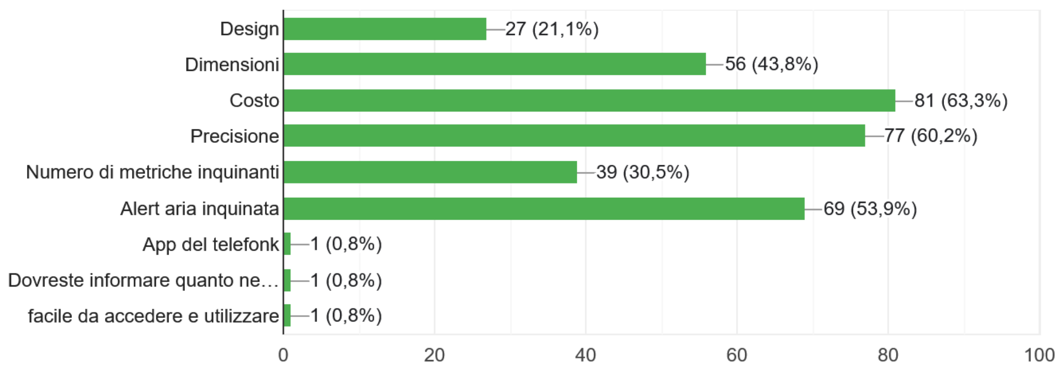
3. 'Are you willing to use a device to inform you about the air that you breathe?': here we can see some mixed feelings, probably because people don't want the hassle of having to carry around another device. Nonetheless, most people would be willing to use such a device.



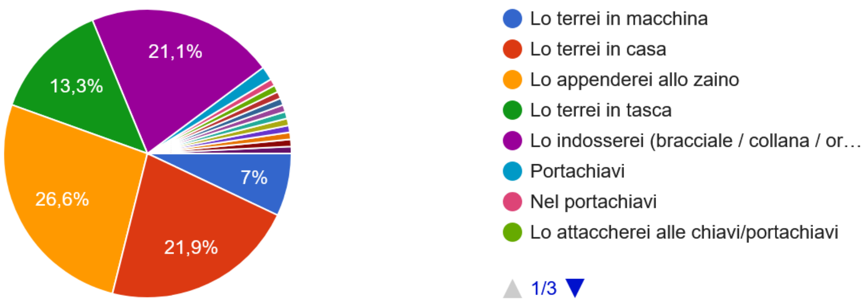
4. 'Would you be interested in an app connected to such device, to get more information out of it?': fortunately, most people would indeed be interested!



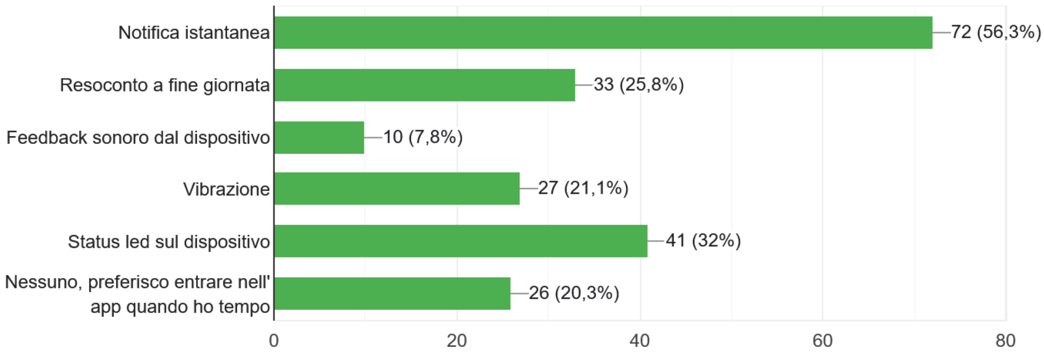
5. 'How frequently would you check the air quality?': we can see that most people would check it quite infrequently (once a day or less), again reinforcing the idea that people care about the air quality, but not too much.



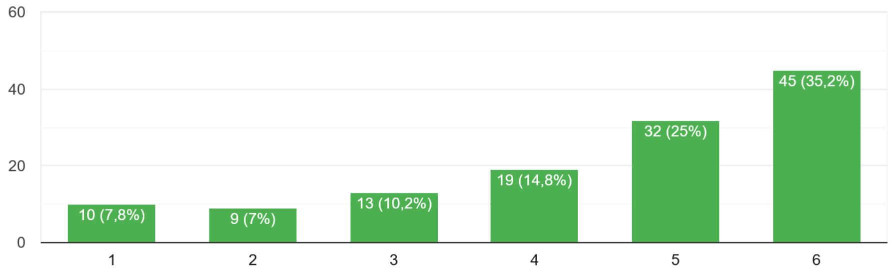
6. 'The most important characteristic of this device should be...': we can see that most people would prefer a device that is small and easy to use.



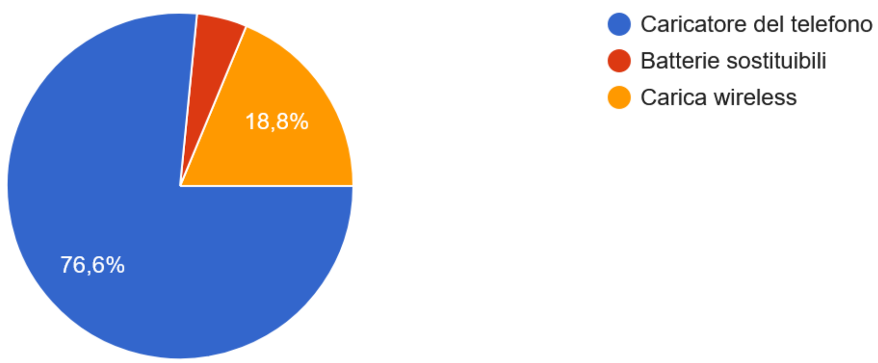
7. 'How would you carry such device?': this was the only question with the option to add a custom answer, and while most people would carry it on their backpack or in a pocket, most custom answers were to carry it on their keys.



8. 'How would you like to be alerted of a really low air quality?': as we can see, most people wouldn't like to interface with the device itself, but would rather get a notification on their phone.



9. 'Similarly, would you like to be informed if the air that you were breathing was *really* good?': surprisingly, most people would like to be informed of that!



10. 'The best way to charge the device is...': as expected, as everybody has it, most people would prefer to recharge it via USB.

Given these feedbacks, we were pretty confident that we were on the right track.

2.4 Storyboards and Device sketches

After the interviews and questionnaires, we set out to describe some storyboards to represent our idea.

Figure 2.1: Bluetooth connectivity

The first storyboard represents how we envisioned the connection to work. In an intuitive fashion, we figured that the connection should be done once and then forgotten, as the device would automatically connect to the phone whenever it was in range.

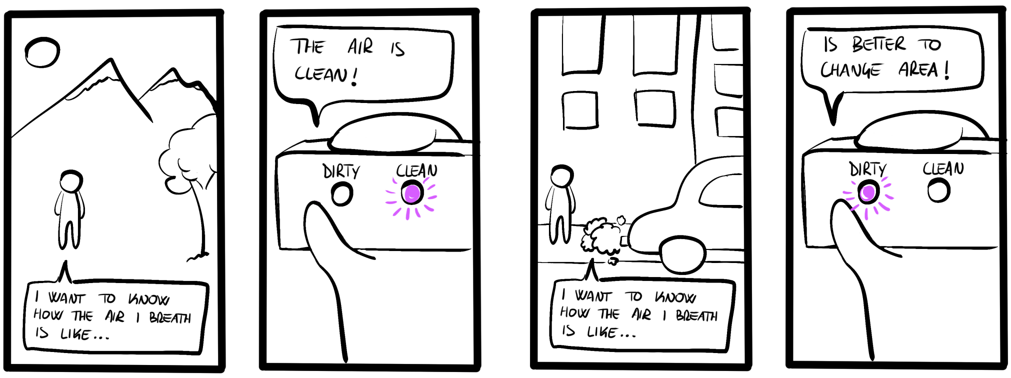


Figure 2.2: Device feedback

This second storyboard represents the feedback. The device should have an on-board and easy way to give information, other than the obvious in-app view.

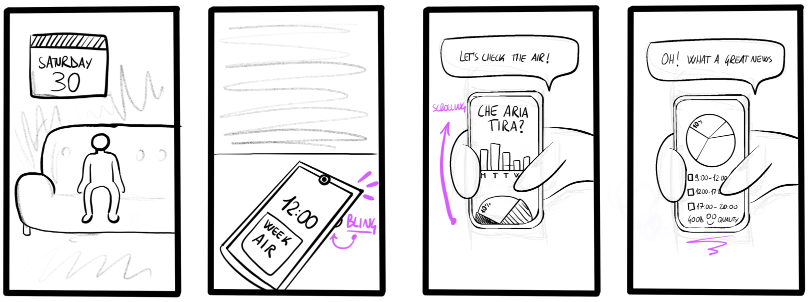


Figure 2.3: Overall Report

The third storyboard represents a detailed overall report on the app. Other than an instantaneous view, we figured that the user should be able to see the air quality in a given period of time, and that the app should be able to give some insights on how to improve the air quality.

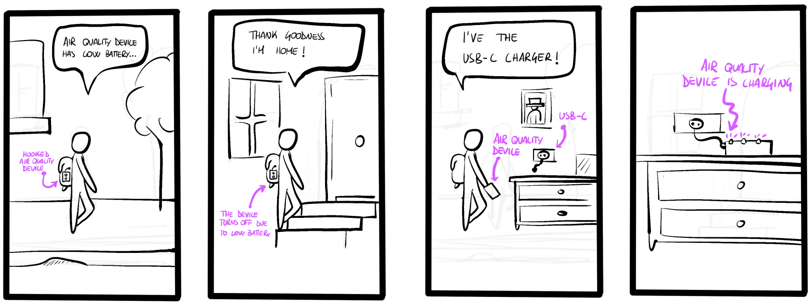


Figure 2.4: Portability and recharging

Finally, the fourth storyboard represents the portability and recharging of the device. The device should be small enough to be carried around, and it should be recharged via USB.

2.4.1 Device sketches

Before setting out to create the device, we also made some sketches to represent the main features of what we wanted the device to have.

Figure 2.5: First ideas for the device

We were sure that the device needed to have at least an ON/OFF switch, one or more buttons for interaction, and *some* way to give feedback (on top of a wireless capable microcontroller, sensors, and a battery). The only true unknown was how to display the information. In the three images above, we can see that we thought about having a single RGB LED, a strip of LEDs, or a set of 7-segment displays. The third option was discarded because of the added bulkiness it would give to the device (cheap 7-segment displays are quite big) and the possible strain of dealing with actual numbers (from a user's perspective), while the first two were kept for further consideration.

Chapter 3

Prototyping and Development

The project consists of two substantial components: the physical device that senses the air and the mobile application that receives and visualizes the data for the user. The development of these two components is highly heterogeneous, encompassing various technologies from hardware to software. The prototyping and subsequent development require the integration of both parts to create a well-designed device that communicates efficiently with the app. The user experience design of this project covers multiple aspects, including the feedback from the device and the user interface of the application.

In the following sections, the design choices and technologies used for both components are explained. The concluding section describes how the design integration was achieved throughout the development process. Overall, during the various iterations of testing we received feedback from roughly 25 people, 12 of which were recorded.

3.1 Mobile Application

As we expected to produce a not-too big application, and feared unseen problems in the integration with the hardware, we skipped the Figma design phase: preferring an iterative approach to the development, which interleaved the software programming with testing. Indeed, almost all the parts of the application have seen an evolution throughout time. The evolution mirrors the results of the tests with the aim to clarify each component's scope, in this context having a constant feedback from the users played a crucial role in the interface design.

We engineered a scroll-based Home View in which the user would be able to find all the information he can be interested into. This type of interfaces are popular in social media, in which are usually displayed a lot of data which require low focus. In our case despite the graphs are as immediate as possible to be understood, due to the small amount of cards displayed, we were able to use this kind of solution without impacting the navigability or the user experience.

Figure 3.1: Summary

Our base idea was to show the last measure performed by the device, giving to the user a shot of the situation. To serve this purpose, a "Summary View" has been first designed and implemented as a single pie chart, in which different portion represented the relative impact of different factors on the air quality index, displayed in the middle. The leftmost image in figure 3.1 shows it. However, tests unveiled that interpreting this widget was difficult to user, even after an explicit explanation. The air quality index, that was the only thing that won't change in the following iterations, since was appreciated and clearly understood, will be kept unchanged.

As finding a visual representation for the data was not an easy task, we opted for a pure value printing, 3.1 center picture. In this case tests showed a step forward in readability, but still the user was disoriented by the absence of context. We find out that users needed to know a maximum value to correctly interpret the measure, despite the emoji, the index and the colors had a clear meaning. As one may be tempted to think that this was a minor issue, in order to take effective action immediately, a user should be aware of what is causing its air quality index to shrink. To better integrate the device functionality the timestamp of the displayed measure has been added to the interface.

The final version, 3.1 rightmost image, introduced two main improvements: the values of the CO_2 and the $TVOC$ are now displayed using two linear progress indicators, implicitly showing the maximum value, and giving a visual representation of the current state; the second improvement is the introduction of a visual standardized rating system, the 5-star rating. Led to popularity by modern reviews services has been placed aside the air quality index to guide the user familiarizing with this novel metric.

It's worth mentioning that in all the three versions a tappable hint as been placed aside the index to an unequivocal explanation of it, when requested, and in all three iteration has always been appreciated by testers.

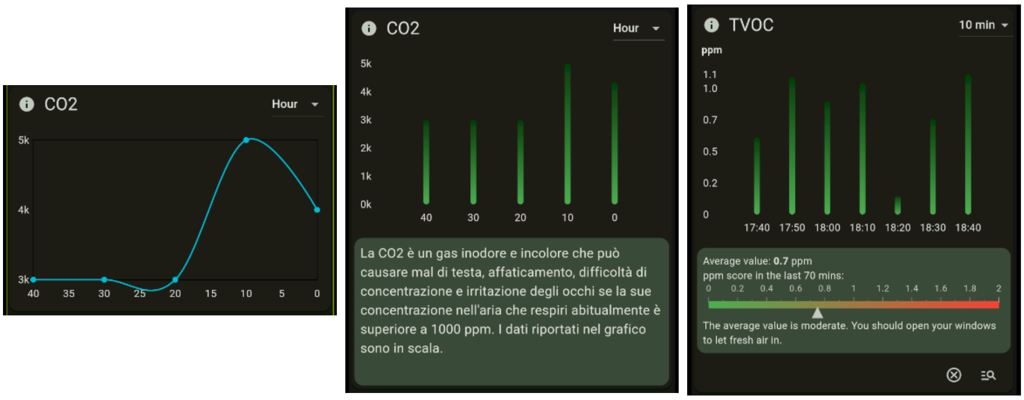


Figure 3.2: Bar Chart

The scrollable view displays three cards: one for the CO_2, one for the $TVOC$ and one for the air quality index. In each card the user has a visual representation of the data collected by the sensor, in a window of time that can be selected via a drop-down menu, placed in the top-right corner of the card. In the title of each card also appear a tappable hint, explaining the parameter.

The first version of the bar chart, 3.1 leftmost image, displayed the quadratic interpolation of the measurements (y-axis) against time (x-axis). Despite the obtained graph was smooth, users found it difficult to interpret, moreover the smoothness obtained had to be paid with not precise representation of the data. Additionally, in this first version we used as tick's marker the interval from the present time, easy to implement but difficult to read, all the interviewed wasn't able to understand the meaning of the x-axis.

The second iteration, 3.1 center image, marked the path: the graph has been replaced with a bar chart, precise and easy to interpret, hint has been made more explicit, and opened below the graph itself. The x-axis ticks haven't been replaced during this step unfortunately, being again the weak point of the interface. Another finding of extensive testing was the fact that user would have appreciated having custom advices on how to improve the air quality.

Finally, we clarified the role of the axes with the third iteration: we added a unit measure for the y-axis and substituted the hermetic ticks with the actual timestamp of each measure. Below the graph the user now has a personalized overview of the situation, with custom advice on how to improve the air quality, and a linear gauge to help him understand over that value influence the air quality in the selected time window. Two icons button can now navigate the user to records view and close the advice section.

Before delivering the project we had the opportunity to test a final and complete version of the system. During these tests users interacted with the device and the application together. For the first time we tested the Bluetooth connection and data visualization in a real world scenario. However, even in this iteration we did small graphical improvements. We changed the color of the bars from a gradient to a solid color 3.3, coherently with the color of the emoji

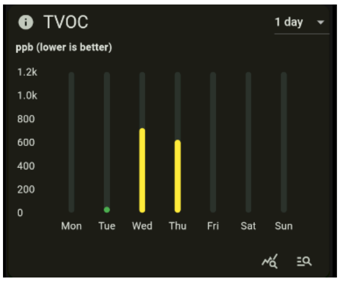


Figure 3.3: Bar Chart

in the header, and since users found confusing that bars lied on a uniform surface we also added a shadow under them indicating their maximum extension.

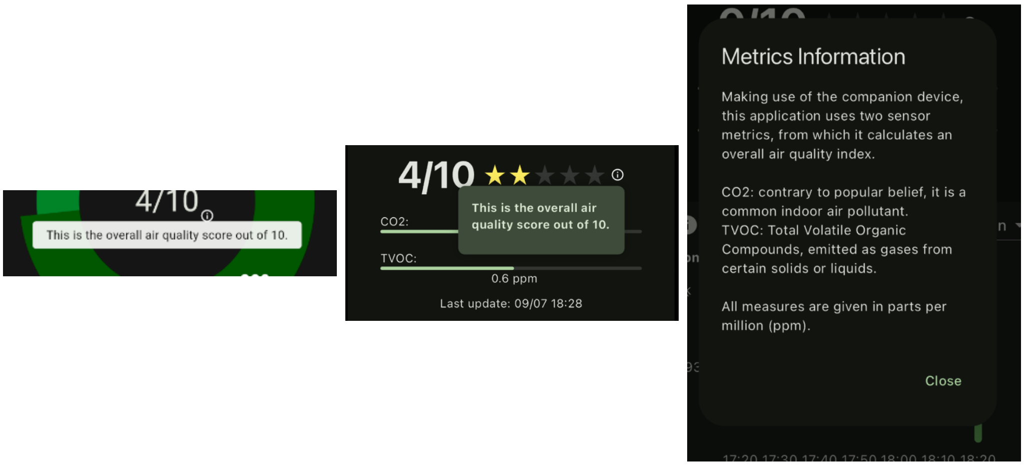


Figure 3.4: Hints

Users' awareness was our main goal, that's why we carefully designed the hints. Starting from a small popover, we wanted, and tests confirmed the needs, to provide as clear and complete as possible information for the app context. That's why we slowly moved throughout iteration to the definitive, more invasive but still minimal, popup view. The hint related to the quality index has been kept as a popover, recalling to a concept that needs minimal clarifications.

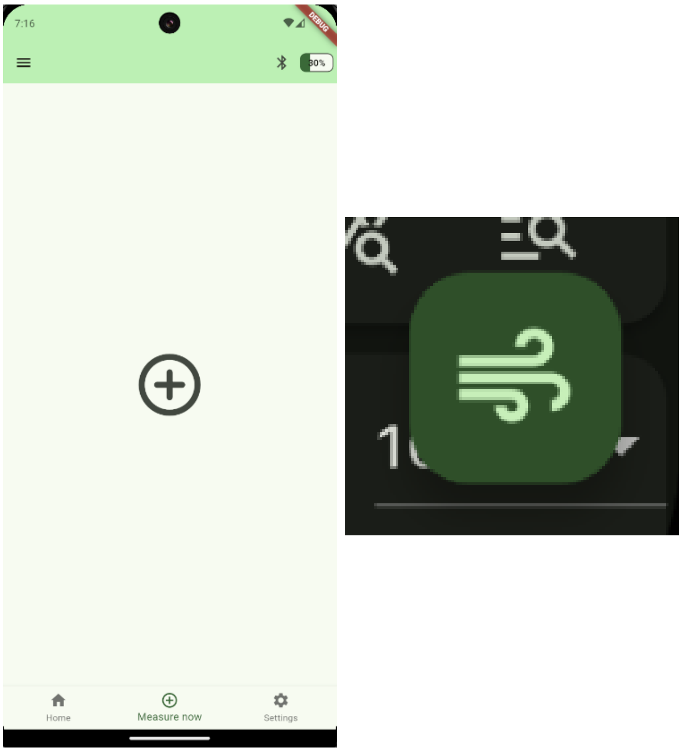


Figure 3.5: Floating button

Another functionality we wanted our system to have from its first version, was giving the possibility to the user to instantly measure the air quality. The left image 3.5 shows more an idea than a trial. We inserted it in the report because we actually used this interface for testing, and really helped us to understand if a user could have appreciated this feature. The right image shows the final version, a standard material floating button. In this case the icon has been selected from the Google Material Icons library. Despite the association with its functionality it's not obvious, any of the interviewed had difficulties to understand its role, showing a winning metaphor.

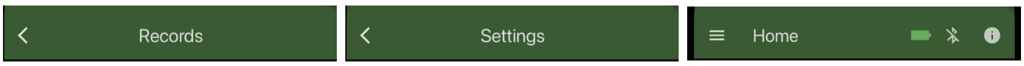


Figure 3.6: App bar

The AppBar follows the IOS navigation standards. Each view has a title, placed in the center except for the home view that, displaying also information about the device's status, such as the battery percentage and the Bluetooth connection status, the title is slightly tilted to the left. In every view is always possible to navigate backward through the path, using the classical arrows.

Figure 3.7: Notifications

Notifications are another crucial feature of the system. In the need finding we discovered that users would like to know whether they are exposed to dangerous air quality, even if they are not using the application. That's why we implement a notification system that alerts him, up to one time per day, if the quality index is below the known average. Notifications also integrated the Bluetooth connection status, displaying whether the connection has been established or lost. We point out that this feature is redundant with the Bluetooth icon present in the home view and the sidebar.

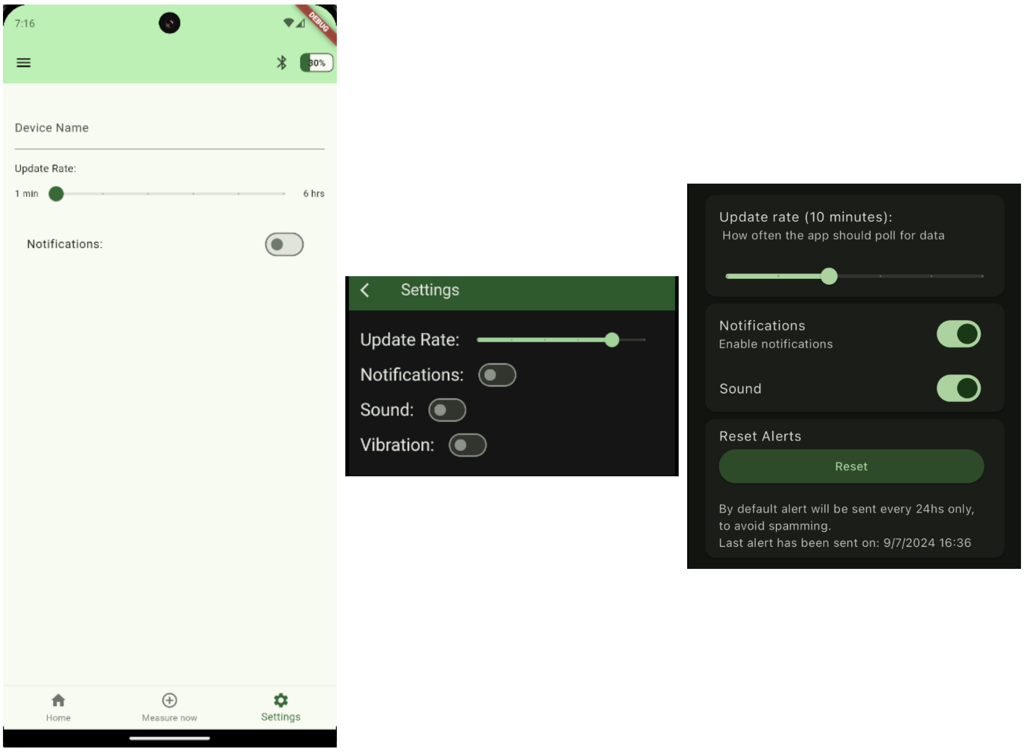


Figure 3.8: Settings

The setting allows the user to tune and customize the main features of the application. The iterations led us to an intuitive and clean design, recalling more common operative systems settings. A linear draggable selector is used to set the refresh rate of the data, two switches allow the user to control notifications' delivery. The last section gives the user control over the notifications' frequency, allowing more notifications per day.

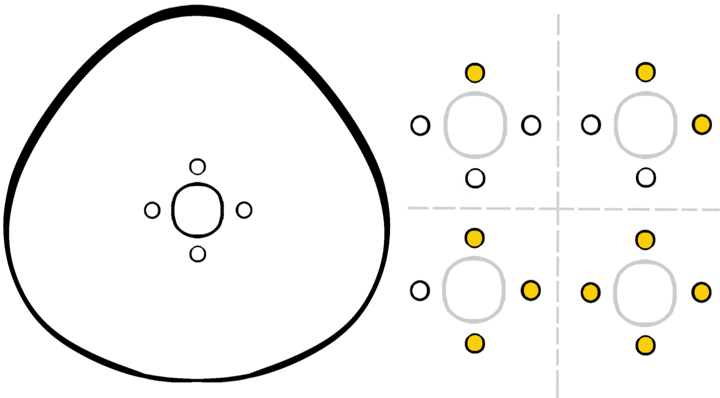


Figure 3.9: LEDs scheme

3.2 Physical Sensor

The design of the device required consideration of various aspects. To better understand how the device would be used, we interviewed several people, asking how they would carry the device and what feedback they deemed necessary.

3.2.1 User Feedback

The device needs to provide a simple feedback mechanism that can serve as a substitute for the mobile application when it is inconvenient to use. The team considered various types of feedback, including haptic signals and sound alarms. However, based on the interviews and the need for accessibility for people with disabilities, the team decided to use an LED signal. The quality index is mapped to different light levels to convey the necessary information.

Device States and Corresponding LED Schemes The feedback system is designed using four micro LEDs arranged in a circle around the interaction button, centralizing all active, interactive elements. We chose to use four LEDs to enhance energy efficiency without compromising the clarity of the feedback. The scheme is shown in the figure [3.9].

The animation of the LEDs is different based on the state of the device:

- Loading state;
- Measuring state
- Charging state;
- Full charge.

The device enters the loading state during the first five seconds after starting. In this state, the device initializes the BLE module and the air sensor module. The device signals this state to the user by displaying a rotational LED animation, with each LED turning on every 200 ms. After a complete rotation, all LEDs turn off and the cycle repeats until the loading state ends.

After the loading state, the device transitions to the measuring state, where it measures the quality of CO2 and TVOC to produce the corresponding quality index. In this state, the LEDs indicate air quality only when the button is pressed, conserving energy during idle periods. When the interaction button is pressed, the LEDs light up without blinking, remaining steady to show the air quality level for ten seconds. The animation follows the main scheme shown in Figure [3.9]. The more LEDs that are illuminated, the better the air quality.

In conclusion, there are the charging states to consider. When the device is plugged into a USB-C cable, it can be in one of two states: still charging or fully charged. While charging, the LEDs light up every 250 ms, following the same rotation pattern as in the loading state. The key difference is that only the LEDs corresponding to the battery's charge level will be illuminated. For example, if the device has 75% battery, three LEDs will be lit. When the device becomes fully charged, all four LEDs start blinking every 250 ms, signaling that it has reached 100% charge and is ready to be used for many hours without needing an external power source.

Figure 3.10: Button Switch

Interaction Button and Power Switch Based on brainstorming sessions and data collected from interviews and need-finding activities, we discovered that users need to interact with the IoT system not only through the application but also via the physical device itself. Analyzing users' ideas, we realized that it is essential to offer an alternative to using the phone, which can be inconvenient during activities like hiking when the phone is stored in a bag or pocket.

We considered to use a minimal, simple and accesible interaction type, focusing on scenarios where users want immediate information, such as the current air quality. To avoid potential errors that could arise from more complex interactions like shaking or voice commands, especially given the device's intended use during activities like trekking, we opted for a simple and effective solution. We chose to implement a single, centrally located button(3.10) surrounded by feedback LEDs to enhance accessibility and simplicity. With just one click, users can instantly know the air quality by looking at the feedback directly on the device.

Another important factor we considered is the pressing time required to activate the device's air quality function. We decided that a minimum press duration of one second would be necessary to effectively activate the device for two main reasons: First, to prevent accidental activations that could turn on the device unintentionally and waste battery power. Second, to provide a clear and deliberate feedback mechanism for users. This prevents the click from being too short, which might cause users to think the device was not activated, and ensures that the interaction time can be slightly longer, accommodating users who may not be as familiar with IoT devices, such as elderly individuals.

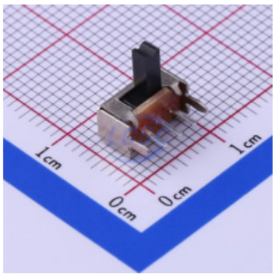


Figure 3.11: On/Off Switch

Finally, we decided not to allow the central button to power the device on or off. Since this button is the primary tool users interact with, there is a risk of accidentally powering off the device during use, leading to interruptions and frustration. To avoid these issues, we implemented a dedicated ON/OFF switch(3.11) on the side of the device, away from the central interaction area. This design choice clearly separates the power function from the main interaction button, ensuring that users can only power the device on or off by intentionally using the switch. Additionally, this approach enhances user confidence, as they can trust that the device will remain operational during important moments, such as monitoring air quality. The dedicated switch also adds a layer of user control and security, preventing accidental shutdowns and promoting a smoother, more reliable user experience.

Product design The concept for the device was conceived with portability in mind. Many existing air quality stations, both private and public, are hindered by their large size and heavy weight. From the beginning, our goal has been to create a device that is compact enough to fit in a pocket and capable of measuring the air quality that individuals breathe throughout their daily activities. However, a problem suddenly arose: the device needs to "breathe" like a human being. It requires a section that can't be covered by fabric, making it unsuitable for being carried in a pocket. This challenge led to the refined concept: the device must be attachable to a backpack. Since our users are likely students or people who enjoy long walks in the mountains, they typically carry backpacks. Therefore, the device needs a hole to attach a snap hook or lanyard, allowing it to be secured to the outside of a backpack or placed in perforated side pockets. This design ensures the device remains functional and accessible while being carried.

Figure 3.12: Snap hook

Once we understood the nature of the device, it was necessary to design its shape. Based on the interviews, the device needed to be small and lightweight. Our initial idea was to create a device smaller than a smartphone, following a rectangular style. However, after several design iterations and brainstorming sessions, we decided to create a device that embodies the essence of nature. It should relate to air and symbolize the improvement of air quality by removing CO2 or harmful particles. We found inspiration in an organism responsible for producing 50% of the Earth's oxygen—phytoplankton. These organisms are fascinating due to their symmetrical and unique shapes. We chose one of these shapes to design our device, making it both functional and visually representative of its purpose.

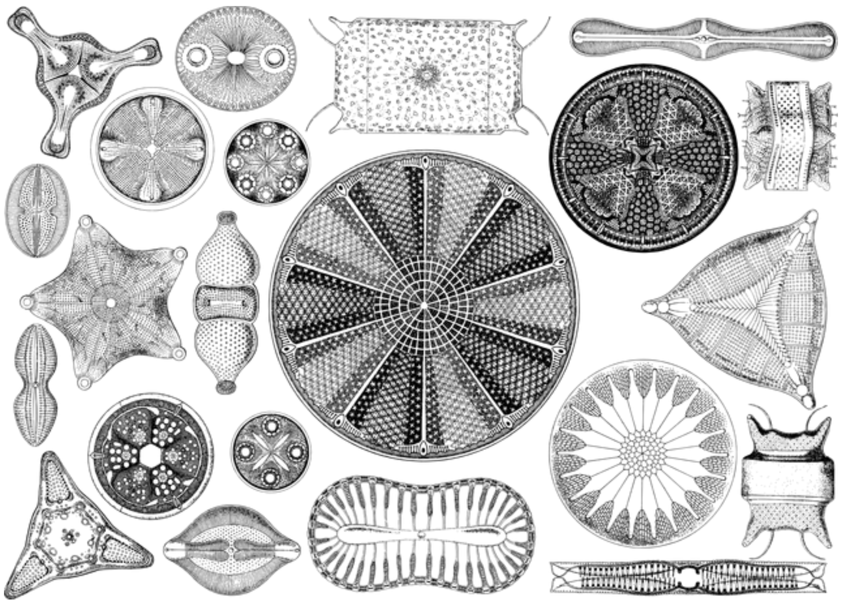


Figure 3.13: phytoplankton organic shapes

3.2.2 Tools and Technologies

Creating the device requires careful selection of fundamental components, in fact the choice of the board can significantly alter the development process. Finding the right solution is not an easy task. Therefore, before starting the construction of the device, we went through a phase of considering which components to use. This choice was guided by user needs, ranging from real-time communication to maximum compatibility and fast charging. We meticulously selected each component to enhance interaction and minimize energy waste.

Bluetooth Low Energy (BLE) Before starting to search for components, we carefully considered the most convenient and efficient type of communication for our purposes. There are many paradigms and technologies for IoT communication to choose from. In our case, we employ a form of edge computing, where the device sends data to a smartphone, which then decides whether to store the data in the cloud. Therefore, our focus is solely on the communication between the smartphone and the device, as the cloud can optionally be accessed by the smartphone using internet services. This approach allows for flexibility and ensures efficient data handling based on user preferences.

The options for our solutions were basically two: WIFI and Bluetooth Low Energy (BLE). Bluetooth Low Energy (BLE) is often more suited for IoT devices compared to WiFi due to several advantages. BLE is designed for low power consumption, making it ideal for battery-powered IoT devices that need to operate for extended periods without recharging. Its modules are generally cheaper than WiFi modules, reducing overall device costs. Additionally, BLE devices have simpler hardware requirements, allowing for more compact designs. It is particularly suited for applications with infrequent data transmission, such as sensors. These factors made BLE the preferred choice for our application.

The way the device makes use of the BLE technology is by announcing itself to nearby devices, creating a readable service that acts as an information beacon: it constantly writes CO2, TVOC and air quality data to this service, offloading the bigger computational tasks to the mobile device.

XIAO ESP32C3 IoT Mini-Board The selection of the board is one of the most important choices you must make when building an IoT device. This decision requires careful consideration of numerous factors, including dimensions, battery usage, internal charging modules, available pins, internal communication modules, and more. In our case, we needed a very small board equipped with an internal BLE module to minimize dimensions and conserve pins. Additionally, we required an internal charging module to further reduce the size. Given our emphasis on compactness, one of the main challenges was the risk of having too few available pins when choosing a smaller board. Balancing these requirements was crucial to ensure that the board met all our functional needs without compromising on size and efficiency.

Another crucial consideration is the embedded processor. Options include the nRF5x series, ESP32, ATmega, and others. Initially, we focused on boards with nRFx processors due to their efficient BLE implementation, often incorporating the antenna. However, after thorough research, we decided to switch to the ESP32 because it offers more board availability on the market at a generally lower cost. This choice significantly impacted our work, as we now need to develop firmware that efficiently utilizes BLE and design the device to accommodate a small external antenna. This shift requires us to adapt our development process to ensure optimal performance and integration of the chosen components.

After listing all the requirements and challenges, we compiled a list of mini-boards from various producers, including Adafruit, Arduino, Seeed Studio, SparkFun and DFRobot. Ultimately, we chose the XIAO ESP32C3 from Seeed Studio[3.14]. for several key reasons. This series features very small boards with the right amount of pins to meet our needs. Additionally, the XIAO boards come with an internal charging module and a thin, sticky BLE antenna that can be easily attached to the internal face of the device shell. This combination of compact size, integrated features, and ease of antenna placement made the XIAO series the optimal choice for our IoT project. The other options, such as the Beetle ESP32-C6 and SparkFun Pro Micro - ESP32-C3, while also robust, did not align as perfectly with our specific size and functionality requirements.

Figure 3.14: Seed Studio XIAO ESP32C3 Mini-Board

CCS811 Air Quality Sensor After deciding on the board to use, the second most important component required for our device is the sensor. Choosing a sensor is not an easy task; in fact, you have to consider several properties. These include the communication protocol used for interfacing with the board, the type of sensor (such as mechanical, chemical, laser, or others), and ensuring that the sensor provides the specific parameters we need.

- Adafruit SGP30;
- CCS811 CJMCU;
- Arduino MQ135;
- PMS5003.

The PMS5003 is a laser sensor known for its high accuracy. It can detect PM2.5 and PM10 particles and is commonly used in static climate stations for air quality monitoring. Despite its high accuracy and reliability, this sensor is too large for use in a portable device.

Our choice has been narrowed down to three sensors. The MQ135 is popular for DIY Arduino projects due to its small size and high compatibility with boards programmable via the Arduino IDE. However, its accuracy is insufficient for professional applications that require precision.

The final decision came down to the Adafruit SGP30 and the CCS811 CJMCU. Both sensors are quite similar, utilizing I2C communication (which is easy to set up) and measuring CO2 and TVOC levels. We chose the CJMCU because it is more affordable and more readily available in the market.

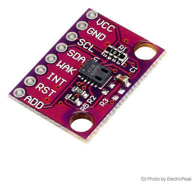


Figure 3.15: Seed Studio XIAO ESP32C3 Mini-Board

Lithium-Ion Polymer Battery (LiPo) To make the device portable, we needed a battery. The choice of battery is crucial, as it must be small, lightweight, and capable of powering the device for an extended period. We considered several types of batteries, including alkaline, nickel-cadmium, and lithium-ion. Alkaline batteries are not rechargeable, making them unsuitable for long-term use. Nickel-cadmium batteries are rechargeable but have a lower energy density than lithium-ion batteries. Lithium-ion batteries are the most suitable for our device, as they are lightweight, have a high energy density, and are rechargeable. Additionally, lithium-ion batteries are available in various sizes and capacities, making them ideal for our compact device. We chose a lithium-ion polymer battery (LiPo) with a capacity of 1000mAh, 3.7V, and a size of 45x25x10mm 3.16.

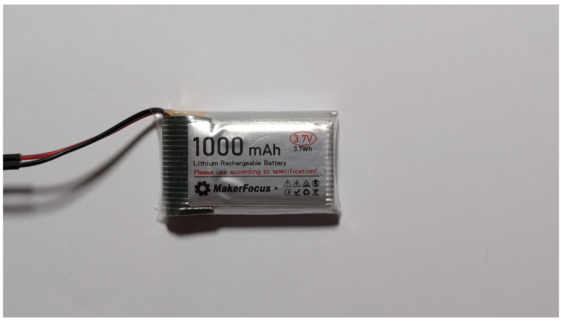


Figure 3.16: Lithium-Ion Polymer Battery, 1000mAh, 3.7V

3.3 Mobile and Sensor Design Connection

3.4 Implementation

We have done multiple implementation before to reach the final prototype. In this section we will describe the main steps we have done to reach the final prototype.

3.4.1 First Implementation

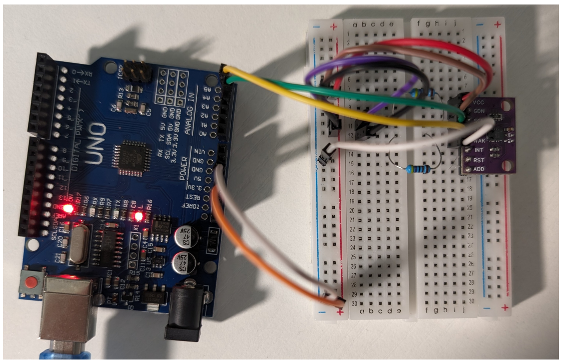


Figure 3.17: The first implementation of the device

The first implementation of the device was a simple prototype to test the communication between the board and the sensor. For this implementation, we used the UNO board insthead of the XIAO-ESP32C3 board and the CCS811 CJMCU air quality sensor. We implemented a simple circuit on a breadboard to connect the board and the sensor. The board was connected to the sensor via I2C communication. The sensor was powered by the board, and the board was powered by a USB cable connected to a computer. The LEDs and the button were not yet implemented. The device was able to measure the air quality and send the data to the computer via the serial port. This implementation(3.17) was used to test the communication between the board and the sensor and to verify that the sensor was working correctly.

3.4.2 Second Implementation

After the familiarization with the sensor and the board, we moved to the second implementation of the device. For this implementation, we used the XIAO-ESP32C3 board and the CCS811 CJMCU air quality sensor. We implemented a more complex circuit on a perfboard to connect all the components (3.19). For this prototype, we added the LEDs, that are connected to the board via GPIO pins. The button and the Li-Po battery were not yet implemented. The device was able to measure the air quality and send the data to the computer via the serial port. This implementation was used to test the leds behavior and to verify that the sensor was working correctly over a longer period of time.

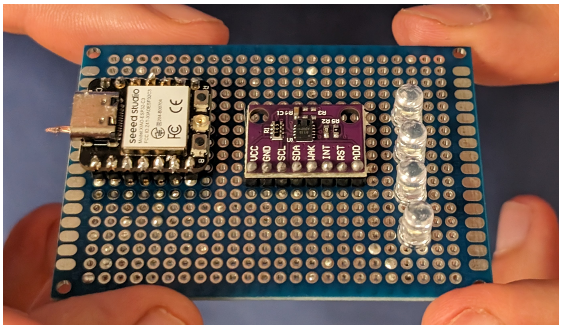


Figure 3.18: Top view of the second implementation of the device

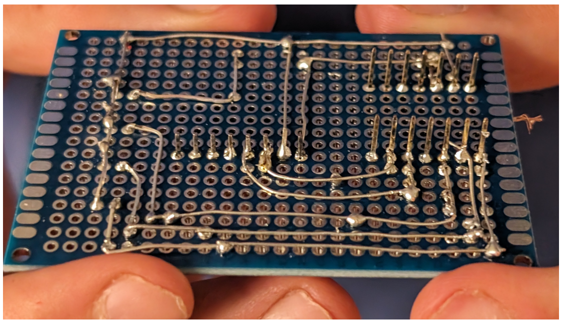


Figure 3.19: Bottom view of the second implementation of the device

3.4.3 Third Implementation

For the third implementation of the device, we added the button, the Li-Po battery and the On/Off switch. In this implementation, most of the component external to the sense system were developed via JST connectors, to make the device modular and easily to repair. The button was connected to the board via a GPIO pin, and the Li-Po battery was connected to the board via a JST connector as the On/Off switch. The device was powered by the Li-Po battery, and the battery was charged via a USB cable connected to a power supply. The device was able to measure the air quality and send the data to an application via Bluetooth Low Energy. The LEDs were used to display the air quality level, and the button was used to start an imminent air quality measurement to provide immediate feedback to the user. This implamentation was used to test the battery life and the charging system of the device, and to verify that all the components were working correctly with the battery. Effectively the device was working as expected, but the design was not yet finalized, so we moved to the final implementation.

3.4.4 Final Implementation

Printed Circuit Board Design

A *PCB* is a flat board made of non-conductive material, such as fiberglass or epoxy, with conductive pathways etched or printed onto its surface. It is used to mechanically support and

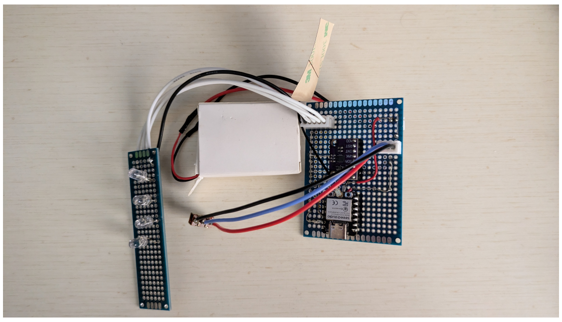


Figure 3.20: Top view of the third implementation of the device

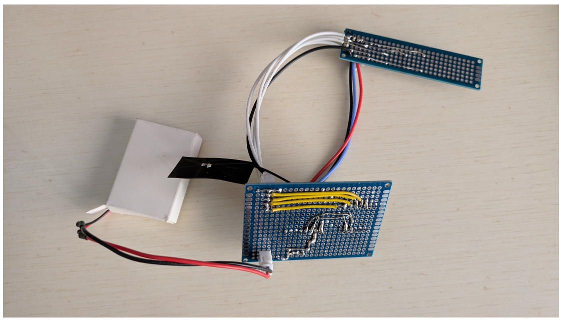


Figure 3.21: Bottom view of the third implementation of the device

electrically connect electronic components using conductive tracks, pads, and other features. We decided to develop a PCB to connect all the components of the device. The PCB was designed using the *KiCad* software, which is an open-source software suite for electronic design automation and in Figure 3.22 we can see the final design of the circuit. The PCB was designed to connect the XIAO-ESP32C3 board, the CCS811 CJMCU air quality sensor, the Li-Po battery, the button, and the LEDs. It was designed to be small and compact to fit inside a 3D printed case. We also added two JST connectors on the PCB to connect the battery and the On/Off switch to the circuit. This design choice was made to make the device modulare, permitting to easily replace the battery and the switch in case of a malfunction. The PCB was manufactured by a professional PCB manufacturer (*PCBWay*). The final printed circuit(3.23) was tested with a voltmeter to verify that all the connections were correct.

Implementation

The final implementation of the device was a more complex prototype that included all the components. For this implementation, we used the XIAO-ESP32C3 board, the CCS811 CJMCU air quality sensor, the Li-Po battery, the button, and the LEDs. We implemented a more complex circuit on a PCB to connect all the components and then we soldered all the components to the PCB (3.24). The device was able to measure the air quality and send the data to an application via Bluetooth Low Energy. Most of the charateristics of the device are the same of the third implementation, but the final implementation was more compact

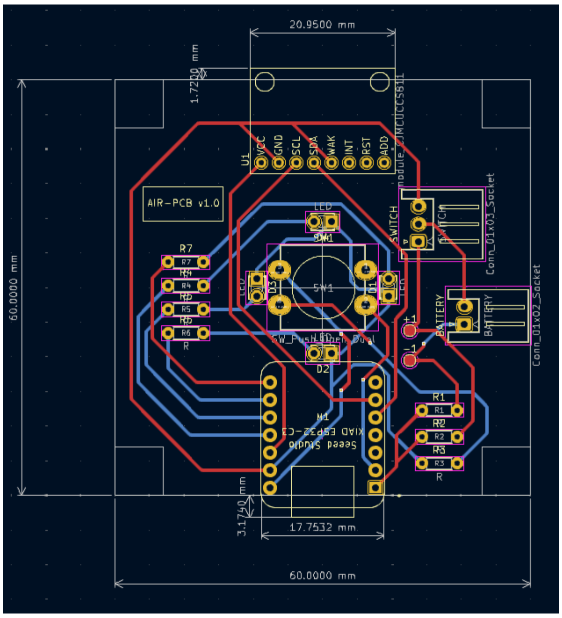


Figure 3.22: PCB design

and robust. Particularly we decided to remove modularity from the LEDs, we soldered them directly to the PCB. This implementation was used to test the final prototype and to verify that all the components were working correctly.

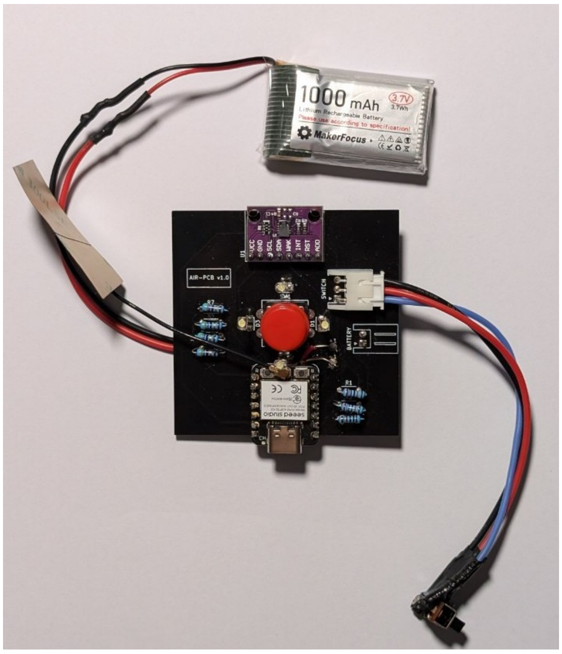


Figure 3.24: Final implementation of the device

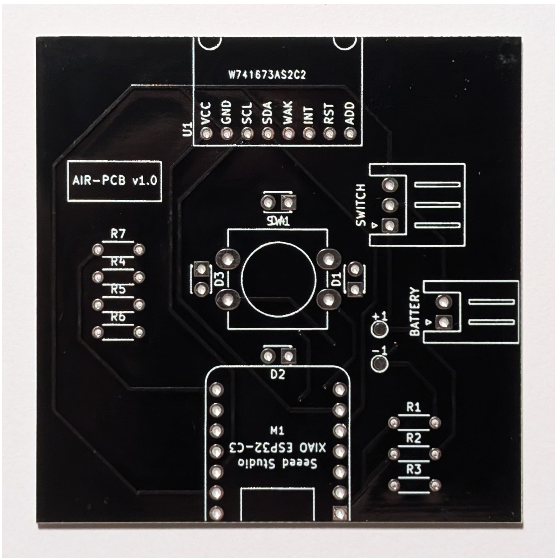


Figure 3.23: Printed PCB

3D Case Model

We also designed a 3D case to contain the device. The case was designed using the *Rhinoceros 3D* software, which is a commercial 3D computer graphics and computer-aided design application software. The case was designed with the organic shape described in the previous section. The case was designed to be small and compact to fit inside a pocket or to be attached to a backpack.

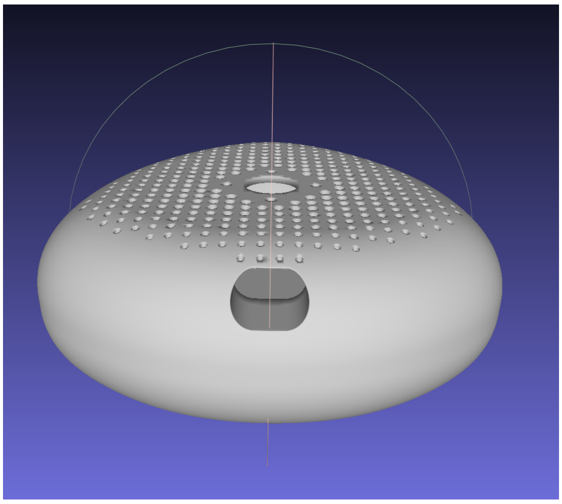


Figure 3.25: 3D model of the case

Chapter 4

Conclusions

4.1 Final testing

After the last version of both the device and the mobile app had been produced, some final tests were carried out to assess the current state of the project. Some excerpts of these tests are reported here, and the future improvements that these tests have led to are discussed in the next section. Note: these tests were conducted with a device still missing its intended case, so observation made by testers regarding its absence were ignored.

4.1.1 Test 1

1. Observation: "No clear visual cue that the device has effectively been turned on, other than the ON/OFF button".
2. Observation: "The color gradient of the histogram should highlight the measured value better".

4.1.2 Test 2

1. Observation: "A small LED to notify that the device is turned on could be useful".
2. Observation: "A unique histogram instead of three separate ones would be better, even if scales would be harder to integrate".
3. Observation: "Better user feedback when Bluetooth is connected".

4.1.3 Test 3

1. Observation: "The app is not too intuitive, data is not clearly explained".

4.2 Future work

From what we observed in these final tests, the main issues were the visual representation of data and poor feedback from the device itself, which didn't show its power status clearly enough. Based on these observations we started working on a new version where the device uses its LEDs to constantly inform the user that it is turned on and clearer histograms in the mobile app to make the user experience clearer and less eye-straining. A more obvious way of indicating the successful Bluetooth connection has also been added. In addition, a future improvement could be to implement a way to integrate all three data histograms into one, making the app view more concise and eliminating the need for the user to scroll. Additional LEDs could be added to the device to allow more visual inputs and clearer status notifications.

4.3 Conclusions

In conclusion the device is not ready to be mass-produced, but we are proud of the work we have done. From testing, to designing and developing we all enjoyed the process: tests have shown us the importance of a novel point of view, more than once we went to test something that we thought was great, but that right after the first interviewed looked completely different, as the tester spotted a problem that we didn't even think about. The design phase was the most challenging, we had major difficulties in finding a way to be clear to the users, not only in the virtual UI, in which graphs and indices were misunderstood by the testers, but also in the physical UI, in which the limited space and options made us struggle with led patterns and buttons position. And then the developing, in which we solved any kind of problems. Due to the wide range of competences in the group, we had the opportunity to learn from each other, but also to learn something new together. None of us was expecting this ending. We were a new group, with a rough idea of what we wanted, and we ended up with something that is closer to a product than a prototype. A project that lasted months in which we figured out how to work together, and learned from how to weld to what a phytoplankton is. We are proud of the dedication but mostly the passion that we all demonstrated, how we inspired each other, and supported everyone's crazy idea, believing in it.