Project Report

Indoor Environment Air Quality Monitor & Smart Switch

Group 2

DEC. 01. 2022

Tanya .

Student number

Thompson Rivers University

COMP 4980- Internet of Things

Instructor name

Abstract

Indoor air quality sensor and smart switch was implemented to understand the applications of internet of things devices. The circuit of the project contains multiple hardware components with Raspberry Pi 4. The device takes real-time inputs and when the air quality is toxic, it sends a notification to the user informing that the air quality is toxic and the system will trigger the air purifier on. Once the sensor tests the air quality to be normal again, it turns-off the air purifier and the cycle continues. The objective of the project was to promote a healthy indoor environment. Second objective of the project was to turn any purifier into a smart purifier using the Rf switch, transmitter and receiver. The real-data monitoring happens in the system with the help of AdaFruit. All of the components of the circuit are controlled with the help of Python libraries and code.

Table of Contents

List of Tables and Figures

1.0 Summary

1.1 The Problem

The problem this project is solving is that the deliverable internet of things device will help create a safer and healthier indoor environment. If we are exposed to airborne agents that are hazardous indoors, they could cause negative effects on our health such as respiratory problems and allergies. Indoor Air Quality (IAQ) is important to us as it is a backbone to our health and wellbeing. Good air quality can lead to a higher quality of life, low risk to respiratory illnesses & chronic conditions (Lafond, 2018). To find a solution to this project we implemented an IOT device to help us maintain air quality indoors.

1.2 Method of Investigation

We created an internet of Things (IOT) device that helps us monitor and maintain indoor air quality in real-time. The method of investigation here was that we bought an air quality sensor online and it gave us readings after set intervals of time. As we will learn further in the report, we tested the readings of the device after introducing toxic substances in the air and made observations.

1.3 Conclusion

The conclusion of the report and the project is to make users aware of toxic and harmful gasses around them and how indoor air quality can affect them in a good and bad way. The deliverable will perform readings and when the air gets toxic, it will notify the user and turn-on the air purifier to get the air quality back to normal; and once it gets back to normal, the system will turn-off the air purifier and this cycle will keep on going to maintain a healthy air-quality.

1.4 Recommendations

Some recommended approach that we could have taken as a group for this project was to pre-order and pre-research on air quality sensors prior to the implementation of our project in-order to save time and avoid the complications that arose during the process. We could have used an actual air purifier to complete this project but we wanted to keep this project economical as all the devices and components were personally funded.

2.0 Introduction

2.1 Subject

The project was to produce an IOT device that helps monitor and maintain indoor air quality using an air quality sensor and air purifier.

2.2 Purpose

We investigated the regular air quality of our homes and found that the air is not healthy all the time. As stated in the problem portion 1.1 above, the purpose of the project was to solve the problem of unhealthy indoor air quality and how good air quality is important to our well being. Homeclimates.com mentioned in their article how gasses that are contaminated indoors can range from volatile organic compound (VOC) off-gassing to radon, carbon monoxide and nitrogen dioxide (Admin, 2015). According to the United States Environmental Protection Agency, VOCs can cause nausea, fatigue, dizziness, headache, and nose & throat discomfort (US EPA, 2018). The end purpose of the device we built is to promote healthy living indoors.

2.3 Scope

The scope of the investigation of the project was limited to the range of the distance the sensor can sense up-to but to encourage this we used a small intake fan. We considered that because gas molecules expand and air circulates in a room, the sensor would eventually sense harmful gasses. The sensor itself has a major disadvantage that long exposure to high polluted air can significantly weaken its sensitivity according to wiki.seeedstudio.com (wiki.seeedstudio.com, n.d.). Due to the cost of an air purifier we wanted to find a cost-friendly solution and hence we used a lamp to demonstrate the working of the air purifier. Since we used a lamp, our testing took a while to be executed as we had to wait for the air to be clean again in-between tests of harmful substances.

Another limitation of the project that a peer pointed-out during our project presentation was that if the device was not cleaned properly, it could be contaminated after a while due to the intake fan and could produce incorrect readings.

3.0 Methods, Assumptions, and Procedures

3.1 Methods

The project deliverable was an IOT device that consisted of different components in a circuit. The following hardware were a part of the circuit:

Grove Air Quality Sensor v1.3

This sensor is used to test indoor air quality since it is responsive to a number of harmful gasses, such as: carbon monoxide, acetone, thinner, alcohol, formaldehyde etc. It is often used for auto refresher sprayers and auto air cycling systems (wiki.seeedstudio.com, n.d.).

Figure 1

433 MHz RF Outlet Switch, Transmitter & Receiver

The outlet switch is a solution to turn every air purifier into a smart purifier and hence a combination of the outlet switch, transmitter and receiver was used. (Gpio 20, 21)

Figure 2 & 3

Grove Base Hat

This is an add-on board that is required to bring Grove Sensors onto the Raspberry Pi. It has four mounting holes that attach to the top of the Raspberry Pi.

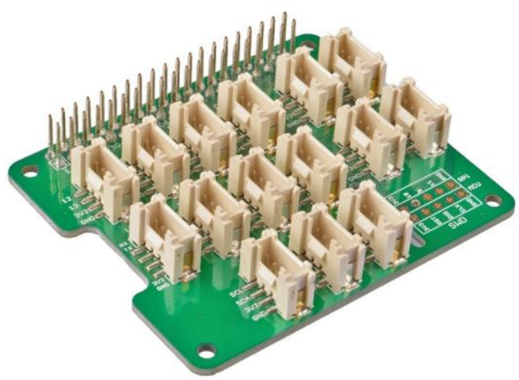

Figure 4

I2C 1602 LCD Display

This is used to show active/inactive status of the system, current air index value, toxic air warning, and current system mode (normal, sleep, stopped). Stopped mode would mean that the program ran into an error.

Figure 5

Raspberry Pi 4

Figure 6

Intake Fan

A PWM intake fan was used to draw air towards the sensor. (Gpio 14)

Figure 7

Custom-Made Case

Figure 8

LED

An LED was used to display the status of the device. It would start pulsing to show that the user was sent a notification indicating that the air is toxic and the air purifier would turn-on (Gpio 26). Moreover, LED on/off means that the program is running and off.

Figure 9

BreadBoard

This was used to connect different components of the device together.

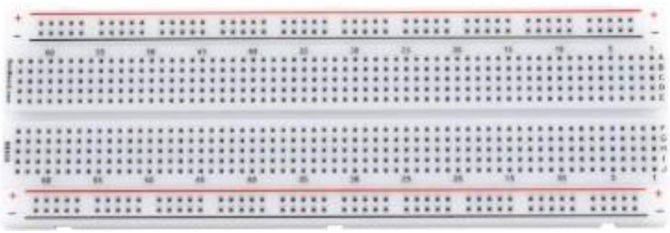

Figure 10

Jumper Wires

Jumper wires were used to connect the breadboard, components, and Raspberry pi together.

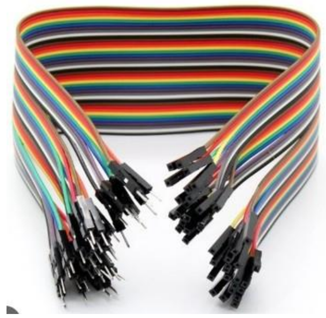

Figure 11

3.2 Assumptions

Assumptions made for this project were that because gas molecules expand and air circulates in a room, the sensor would eventually sense harmful gasses within the short-range of the sensor. Second assumption was that a lamp could be used to display the workings of an air purifier but a real air purifier would clean air at a significant rate rather than just naturally depending on the room's ventilation.

3.3 Procedures

We gained a better understanding of the project by learning what index of air is considered harmful. We ran into problems where the libraries we imported were not working and integrating with the Raspberry Pi. The sensor was not getting connected with jumper wires because of the ends of the wires hence we used a grove base hat with the grove air quality sensor.

3.4 Circuit Setup

Figure 12

Figure 13

The circuit works in the following way: first the intake fan which is using PWM draws air towards the air quality sensor (both are connected to the breadboard), which then makes a reading every 60 seconds and sends the reading to AdaFruit (Figure 14).

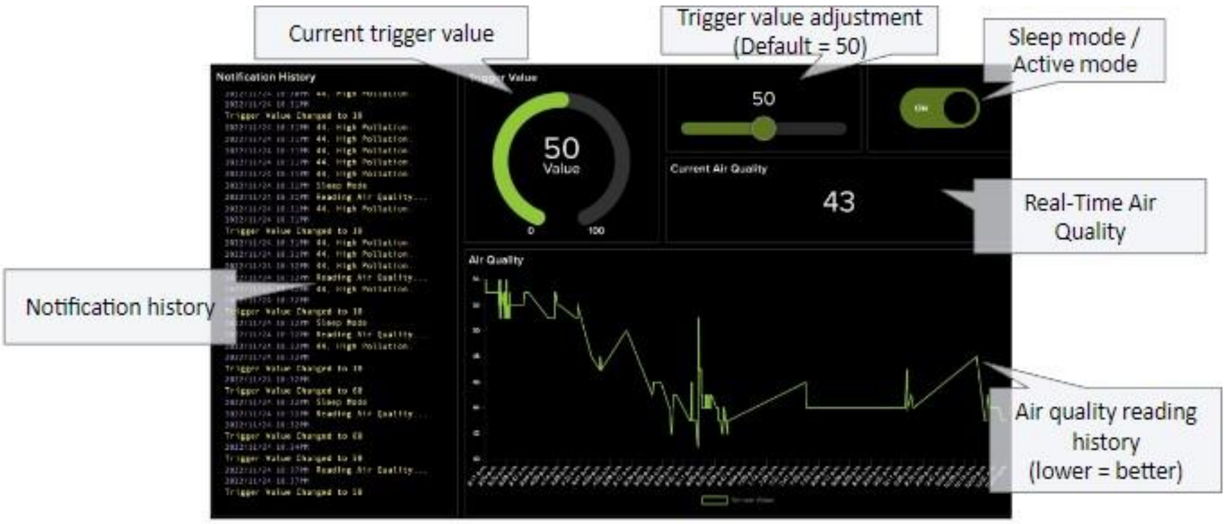

Figure 14

Meanwhile in AdaFruit, the readings are being recorded in the 'Notification History', and the 'Current Air Quality' section displays the latest reading. Above it is the trigger slider to adjust the value of the air index which we determined should be 50 by default. Next to it is the 'Trigger Value' which basically displays the current trigger value, which if-reached will perform multiple actions:

1. Trigger the RF transmitter to transmit signals at 433Mhz to the RF outlet switch which is attached to our air purifier (a lamp in our case) and turn it on and the RF receiver receives signals from the outlet switch.
2. Since the air purifier is to be turned-on because the toxic level of air quality is present, Vonage library methods will be triggered and a text notification will be sent to the user (see figure 15).

Figure 15

3. The LED light which shows the status of the program will start pulsing to show the user that the air quality is toxic (above 50 in our case).
4. Meanwhile LCD displays active status of the system using python code (fig 16).

```
lcd.clear()
lcd.text('SMART PURIFIER', 1)
lcd.text('INITIALIZING....', 2)
```

```
if str(turnOff.value) == 'ON':
    lcd.clear()
    lcd.text('MODE: NORMAL', 1)
    lcd.text('OPERATION ACTIVE', 2)
    mes = 'Reading Air Quality...'
```

```
lcd.text('Status: Active', 2)
```

```
lcd.text('Status: !TOXIC!', 2)
```

```
lcd.text('Status: Inactive', 2)
```

Figure 16

When the air quality sensor takes a reading below 50, it sends a signal to turn off the air purifier. And the LCD displays inactive status.

Adafruit displays a graph which shows the reading history in graphical form. Also at the top right-hand corner is a button to turn on and off sleep mode if required.

3.5 Libraries

We used the following libraries:

1. Grove for the base hat and sensor.

2. Vonage for the notification feature.
3. LCD for the LCD text display.
4. RFDevice for the communication between outlet switch, transmitter, receiver.
5. I2C for the data communication protocol.
6. Smbus2 to import I2C capabilities.

4.0 Results and Discussion

4.1 Results

We learned that the air quality sensor required many libraries to work, that although it could not determine all the toxic gasses present but for the amount of implementation and cost it delivered strong readings. We tested the system by changing the trigger values and placing toxic substances such as heavy duty brake cleaner and wood stain. As discussed in section **3.4** results of the program were: the AdaFruit notification history displays the previous and current sensor readings. Figure 17 shows the change of the trigger value, the status 'BAD' and 'OK' which is displayed on the LCD and showcased by the LED. A.Q is short for "Air Quality".

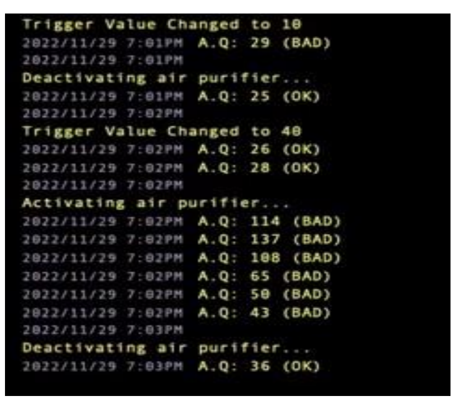

Figure 17

4.2 Discussion

The significance of the results are discussed in detail in section **3.4** and **4.1**. To summarize, the result displayed signifies the status of the whole system. We found that our readings were as accurate as they could be proven by our on-class presentation and video showcase.

5.0 Conclusion

5.1 Restatement of Results

To conclude the report, we learned the application of IOT systems with the course of this semester. We learned how to use different python libraries and use real-time tracking with AdaFruit. Our team was able to find a solution to make every air purifier into a smart purifier with the use of the RF outlet plug, transmitter and receiver. We determined a way to send real-time notification to the user with the help of the Vonage library which I would say was the best accomplishment of the project. For many days we were facing a problem with the physical connection of the components but were able to solve it as a team. Hence, not only was this project a hardware & software learning experience but interpersonal as well.

5.2 Concluding Remarks

As mentioned in the paragraph above, personally I found the discovery of the Vonage library a great accomplishment. And the whole project was a learning curve that we were able to execute using our previous python coding experience and new learnings.

6.0 References

Admin, H.C. (2015). *3 Risks of Poor Indoor Air Quality & How To Improve It*. [online] Home Climates. Available at: https://homeclimates.com/blog/indoor-air-quality [Accessed 2 Dec. 2022].

Arduino Online Shop. (n.d.). *Grove - Air quality sensor v1.3*. [online] Available at: https://store-usa.arduino.cc/products/grove-air-quality-sensor-v1-3 [Accessed 2 Dec. 2022].

Lafond, A. (2018). *Why IAQ Is Important*. [online] Foobot. Available at: https://foobot.io/guides/why-iaq-is-important.php.

US EPA (2018). *Volatile Organic Compounds' Impact on Indoor Air Quality* | *US EPA*. [online] US EPA. Available at: https://www.epa.gov/indoor-air-quality-iaq/volatile-organic-compounds-impact-indoor-air-quality.

wiki.seeedstudio.com. (n.d.). *Grove - Air Quality Sensor v1.3 - Seeed Wiki*. [online] Available at: https://wiki.seeedstudio.com/Grove-Air_Quality_Sensor_v1.3/.

7.0 Appendixes

8.0 List of symbols, Abbreviations, and Acronyms

8.1 Symbols

None

8.2 Abbreviations

IOT- Internet of Things

Fig- Figure

8.3 Acronyms

None