The Application of an Air Pollution Measuring System Built for Home Living

Andrej Ilievski(✉), Dimitri Dojchinovski(✉), Nevena Ackovska, and Vesna Kirandziska

Faculty of Computer Science and Engineering, Skopje, Republic of Macedonia
{andrej.ilievski,dimitri.dojcinovski}@students.finki.ukim.mk,
{nevena.ackovska,vesna.kirandziska}@finki.ukim.mk

Abstract. Air pollution in recent years has become alarmingly a part of our everyday life. One out of nine deaths worldwide are caused by this type of pollution. The first step towards solving this problem, is being able to measure the air particles in the air we breathe and not only outdoors, but indoors as well. Having in mind the uprising of Internet of Things (IoT) and voice interaction with devices, this paper presents a future-proof solution to this problem. It consists of a cheap and compact outdoor station and an indoor station with an ability to be connected with an Alexa enabled device and accessed using voice. These stations were made using a Raspberry Pi 3, a Wi-Fi - enabled micro controller and three different types of sensors, which can also measure the smallest standardized air pollutants - PM1, particles with less than 1μm in size. This creates an opportunity for a detailed analysis of the available data from the indoor and the cheap outdoor air pollution measuring system spread across a city.

Keywords: Air pollution · Air quality · Amazon alexa
Particulate matter

1 Introduction

Health is one of the most important aspects of our lives. But, ever since the industrial revolution, air pollution both indoor and outdoor, has emerged as the deadliest form of pollution, officialized in 2015, accounting for 7 million deaths worldwide [21]. One out of nine deaths worldwide are caused by this type of pollution. The worst part is that it affects our health from our fetal life. A recent study published in Biological Psychiatry Journal, noted that children exposed to high levels of particulate matter during their fetal life had long-term issues with impulsive behavior, self-control over temptations and even some mental health issues [20]. This can be additionally supported by two studies made in 2015 in Barcelona, Spain on 39 schools [29] and in 2011 conducted in Michigan, USA on 3660 schools [25]. Both have done tests on the cognitive performance of students from primary school up to high-school, in more and less polluted indoor

S. Kalajdziski and N. Ackovska (Eds.): ICT 2018, CCIS 940, pp. 75–89, 2018.
https://doi.org/10.1007/978-3-030-00825-3_7

and outdoor environments. The studies came to the same conclusion that children from highly polluted schools had smaller growth in cognitive development than children from the less polluted schools. Another recent study suggests that air pollution [24] has a positive correlation and significant impact on human behavior - more specifically on the overall crime level. The authors of this study suggest that improving air quality in urban areas may reduce crime. These are only some of the most recent research results for the effect of the air pollution aside from the well-documented premature causes of death, such as lung cancer, pulmonary disease, various heart diseases and respiratory infections, done by the World Health Organization [16]. All of this adds up to having an economic impact costing the global economy $ 225B according to the World Bank [17].

Unfortunately, Macedonia also joins this story with Skopje being declared one of the most polluted cities in Europe. The Macedonian Institute of Public Health has reported that the air pollution accounts for up to 1300 deaths per year countrywide [10]. With the authors living in Skopje, this was essentially the motivation for the creation of such system that would have two important features. First, there should be a cheap solution for precise indoor air quality measurement. Second, the system should have a personal and compact air quality station for the outdoor air quality in the area close to the user's home. Making the user's information not dependent on various online sources with low update frequency. Furthermore, multiple such outdoor devices scattered across a city present an opportunity for much more detailed view of a city's air quality in the future. Such detailed data could lead to novel insights and approaches towards battling the air pollution.

This paper presents a compact and cost-effective solution that consists of a pair of stations, one for the outdoor and one for the indoor air quality measurement. This furthermore is done by providing measurements of the smallest standardized air particles PM1, which, as of writing this paper, are not measured by any station in Macedonia. These PM1 particles, by numerous recent researches [18,30], are shown to be the deadliest and most detrimental to people's health. With the aim of having a future-proof interface, Amazon Voice Service is integrated that enables voice control, a user-friendly interface of our system. In the following section, all the details about how the data is collected within the system is covered. Next, Sect. 3 describes how Amazon Voice Service is integrated with the project. And finally, the results from the primary experiments are shown and a conclusion is made in the final section.

2 Dataflow to IoT Database

Before the start of the project, there was a simple idea of making an indoor station for measuring just the particulate matter (PM). After defining which components would be needed and being sought out on popular e-commerce web sites, it was concluded that the station can be assembled of high quality sensors for a price that everyone can afford. Having that in mind, there was an opportunity to also develop an outdoor station. Both stations measure dust particles and upload data to the ThingSpeak database as shown on Fig. 1.

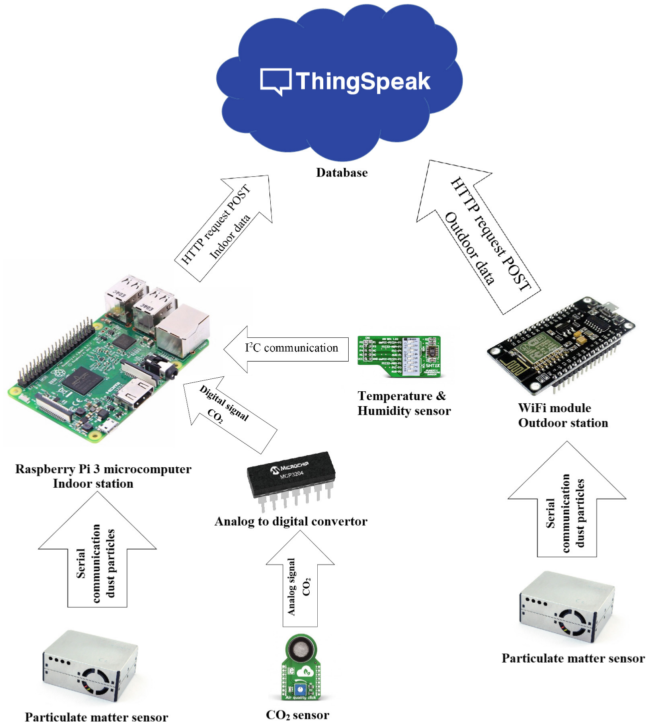

Fig. 1. Indoor and outdoor station sending information to the IoT database

ThingSpeak is a cloud platform specifically designed for IoT devices, where data is collected and then retrieved by any device capable of communicating with the database. The database in our project contains three tables. The "Indoor_PM" table collects the PM information of the indoor station and it has "PM1", "PM2.5" and "PM10" as columns. The "indoor_gas" collects the temperature, humidity and CO_2 information from the indoor station. The "outdoor_pm" collects the outdoor PM information from the outdoor station. It includes "PM1", "PM2.5" and "PM10" as attributes. The data in these tables is uploaded through HTTP Request method POST. In Python, this is done using the urllib2 library (as seen on Fig. 2a). You connect to the database providing the function with the unique database key, which gives a permission to write down the information to the corresponding columns. A detailed elaboration on the indoor and outdoor station and the data that is collected by these stations is given in the following subsections.

2.1 Indoor Setup

The indoor station is composed of the following sensors: a Particulate matter (PM) sensor, a CO_2 sensor and a temperature and humidity sensor. But also

```
def publish():

    CO2, temp, humidity, dewpoint = getData()

    now = datetime.datetime.now()

    params = urllib.parse.urlencode({
    'field1':int(CO2),
    'field2': temp,
    'field3': humidity,
    'field4': dewpoint,
    'key': DATABASE_KEY })

    headers = {
    "Content-type": "application/x-www-form-urlencoded",
    "Accept": "text/plain"
    }

    conn = http.client.HTTPConnection("api.thingspeak.com:80")

    try:

        conn.request("POST", "/update", params, headers)
        response = conn.getresponse()
        data = response.read()
        conn.close()

    except:

            print("connection failed")
```

(a) HTTP POST in the indoor station

```
def PM1_evaluator( value):
    name = 'PM1'
    value = value
    if value >= 0 and value < 14:
        text = 'Good'
        effect = 0.2
        index = 0
    elif value >= 14 and value < 34:
        text = 'Moderate'
        effect = 1.2
        index = 1
    elif value >= 34 and value < 95:
        text = 'Unhealthy'
        effect = 2.2
        index = 2
    else:
        text = 'Very Unhealthy'
        effect = 3.2
        index = 3
    return [name,value,text,effect,index]
def recommendation(text):
    if text=='Good':
        return 'I recommend opening the windows, so that the fresh air'
        +' can come in, or maybe some outdoor activity'
    elif text=='Moderate':
        return 'I don''t recommend opening the window or taking any'
        +' outdoor activity if you are sensitive to air pollution'
    elif text=='Unhealthy':
        return 'I highly don''t recommend opening the window or taking'
        +'outdoor activities, sensitive group may suffer'
    else:
        return 'I recommend staying indoors, outside is very hazardous'
```

(b) PM evaluator function for PM1 and a recommendation function

Fig. 2. Code snippets

it has a Microphone and a Bluetooth speaker which are used for the human interaction with the system as elaborated in Sect. 3. These components require high computational power, so it was vital for the indoor station to be positioned on a microcomputer. Today one of the most popular microcomputers used in IoT is the Raspberry Pi 3 Model B [26]. Its miniature form, affordable price and the possibility to install many operating systems (OS), most of them being open source, made it ideal for the authors' project and thus it was used in the indoor station. Since it is mostly used in the IoT world [22], the Raspbian OS [13], was chosen for this project. It is a Unix-like OS and comes with pre-installed Python, Java, Mathematica etc.

PM Sensor. When searching for the right PM sensor, the authors were looking for one with good characteristics, but for an affordable price. The sensor Plantower PMS5003, mentioned in a highly respected article written by the World Air Quality organization [3], was chosen for this project. The sensor uses laser scattering to radiate suspending particles in the air, and then it collects scattering light to obtain the curve of scattering light change with time. The microprocessor within calculates equivalent particle diameter and the number of particles with different diameter per unit volume.

PMS5003 is hooked up to the Raspberry Pi via the serial line (pins TXD and RXD). When the sensor is triggered to measure, it takes at least 30 seconds to heat up, and in the next 15 seconds it collects data. In the Python code, the serial library is used to read the information from the data pin. Each data from each measurement is formatted into a frame. Every frame, besides the list of measurements, includes a time stamp that stores information for the time of the measurement. After an array of 15 separate measurements is filled, an average of all measurements in the array is calculated and that data is send to the cloud. A library is used to help the sensor obtain the PM values [14]. The data that the sensor managed to collect in a period of a few days were analyzed in order to find the best time period between two consecutive measurements so that we have less power consumption, yet the errors in the measurements to be as low as possible. In order to find that, the mean square error estimator was ran to calculate the risk of expanding the time to measure between two consecutive measurements. On Fig. 3, the x-axis represents the time in minutes when the sensor is activated to measure and upload the data, while the y-axis represents the mean square error of the measured value after a certain time compared to the value measured every minute. Observing the result, it was concluded that the sensor should measure every 8 min which results in more efficient power consumption, while simultaneously keeping a low level of error.

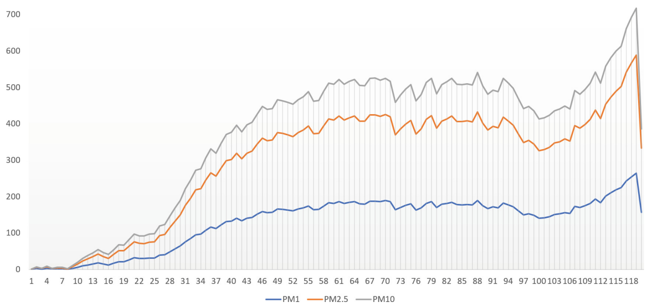

Fig. 3. Mean square error analysis

Carbon Dioxide Concentration Level Sensor. The carbon dioxide concentration level sensor used in this project is a Mikroelektronika clickBoard (CO_2 sensor). This is the cheapest solution for a CO_2 sensor on the market that the authors could find, because it is as much as 6 times cheaper than a specialized CO_2 sensor. Even more, the sensor was chosen because the authors have a positive experience with Mikroelektronika's hardware [27]. Another advantage of this sensor is that it is sensitive to various other natural gases, whereas the more expensive ones are focused only on CO_2. However, this doesn't make it significantly incorrect, since out of all the gases that it is sensitive to, CO_2 is the most prevalent one. On the other side, being sensitive to other gases makes it a decent sensor for sensing abnormalities in the indoor air composition that arise in extreme cases such as fire by detecting the smoke. The sensor works based on the conductivity of the sensitive material made out of Tin dioxide (SnO_2), which has a different resistance depending on the quality of the air. For more precise operation of the sensor, there is an additional electrode that preheats the air around the sensitive material. It was recommended in the MQ-135 data sheet that the electrode and the sensor altogether should be running on for at least 24 h before making correct measurements. The conductive material is sensitive to NH_3, NO_x, alcohol, gasoline, smoke and CO_2.

This board is designed to integrate easily with any microcontroller that supports the mikroBUS socket (used for interfacing microcontrollers created by Mikroelektronika). But, that would require an additional Raspberry Pi add-on board to be used and it would take up more space, so it was decided to make a custom workaround. Since Raspberry Pi is lacking any analog pins, the CO_2 sensor board sends an analog signal to the 12-bit resolution Analog to Digital Converter (MCP 3204). There are 4 analog input pins and one output pin, which sends information to the Raspberry Pi about the corresponding voltage for each input pin. An already available library [15] was used to read the values of the ADC. The voltage values, are converted to analog and then used in another very detailed library [23], written in C/C++ for Arduino. Since all of this should be used on the Raspberry Pi 3 with a code written in Python, the authors needed to translate this Arduino library in Python.

In order to take valuable measurements, a calibration is needed because each sensor is unique. That is, the resistance value varies from sensor to sensor, so it has to be adjusted to the outdoor air which has a well known CO_2 level of around 407 ppm (parts per million) [4]. We can later use this point of reference for the calculation of the current CO_2 ppm. The resistance value is dependent on the temperature and humidity, as noted in the sensor's data sheet. For this purpose we use the temperature and humidity sensor mentioned in the next section. Finally, after the sensor calibration is done, we can get the values of the CO_2 ppm in the air. As noted previously, the main function of this sensor is to read the amount of CO_2 in the air, but it can also read the general air quality. Note that it is instantly sensitive to an increased amount of alcohol, gasoline, gas, and other harmful gases.

Temperature and Humidity Sensor. The MikroElektronika's SHT1X board was used as a temperature and humidity sensor and it was far simpler to integrate than the CO_2 sensor. It uses an I^2C serial interface, where one pin is used to transfer data and the other one to control the clock. For the Raspberry Pi 3 the data is transmitted via pin P3, while the clock pin is on P2. The board itself has a 14-bit resolution ADC, so the Raspberry can directly connect with this sensor, thus no further work is needed. A library [7] was used to obtain the temperature and humidity values in the indoor setup. As mentioned previously these are needed to re-adjust the resistance of the CO_2 sensor.

2.2 Outdoor Setup

The outdoor setup consists of a microcontroller NodeMCU and a PM sensor, which are connected serially. The PM sensor was described previously, so only the microcontroller will be reviewed here. NodeMCU is the name for the platform for IoT that covers the firmware running on NodeMCU devices (ESP8226 WI-FI SOC) and the devices itself. The authors' task was first to install that firmware on the device so that it can be programmable with the Arduino IDE. The plug of the PMS5003, that was originally incompatible with NodeMCU, was connected with a workaround by simply removing the plug altogether and connecting each separate wire as required with the NodeMCU.

A library written to work on Arduino that is also supported by the NodeMCU, was used for the programming. The Arduino code is a C/C++ code structured in such a way that two global functions exist. One function, called setup(), is run only at the start and the other function, called loop(), runs repeatedly until the device is powered off or reset. That is the logic of an Arduino code, but since in this case we are using a deep sleep capability of the NodeMCU [19], we only use the setup function. This is because in the setup function we are setting up the Wi-Fi, getting the measurements from the PMS5003, uploading the data to Thingspeak via HTTP request and putting the device in this deep sleep mode. This mode shuts down the Wi-Fi capabilities, CPU and only keeps a timer running, with a specified time of 8 min, after which it resets the whole device - and begins once again with the setup function. Deep sleep enables us to dramatically lower the energy consumption of the device.

3 Amazon's Alexa Voice Assistant

The data for indoor and outdoor pollution, which is stored in the Thingspeak database, is communicated with the user using the Amazon's Alexa voice assistant, which was implemented on the Raspberry Pi - the indoor station. Having USB ports and integrated Bluetooth, the Raspberry is ideal for using a microphone and a Bluetooth speaker for the voice assistant. Amazon has made it easy for developers to install Amazon Voice Services (AVS) (including Alexa), and to use it to develop all kinds of skills for Alexa [8]. Using Amazon's guide, the authors successfully installed the AVS on the Raspberry Pi, making it an Alexa enabled device. Alexa skills, are the applications that are developed so that users can have a custom and personalized interaction with the voice assistant. To develop a skill, it was inevitable to get to know the whole Alexa Skill concept [9]. The next paragraph discusses the data exchange between the user, the station, Amazon Web Service and the cloud. This data flow can be viewed on Fig. 4.

Alexa uses an invocation command, a word or a sentence, spoken by the user when he/she has some request. Each invocation command corresponds to an Alexa skill that is triggered to be executed. To go further in the complexity of the Alexa ecosystem developed for this project, lets follow an example of a sentence that would be spoken by the user: "Alexa ask air quality xXxXx". In the example, the word "Alexa" is a standard invocation word that lets Alexa listen to you, "air quality" is the custom invocation command that corresponds to the custom Alexa skill developed for this project. All possible strings that can stand for "xXxXx" in the example are defined in the Interaction model that is consisted of a table with intents (sentences) that can be asked by the user for the air pollution in the surrounding. With the right intent, the right skill function is triggered to process the data from the AWS cloud and fulfill the request by sending back the information to the station so that Alexa can inform the user. The intents include the following user requests: "air pollution"; "indoor PM information"; "outdoor PM information"; "indoor carbon dioxide level"; "indoor temperature and humidity". Each intent can be triggered with the statement "how is the xXxXx". For an example: "Alexa ask air quality how is the air pollution".

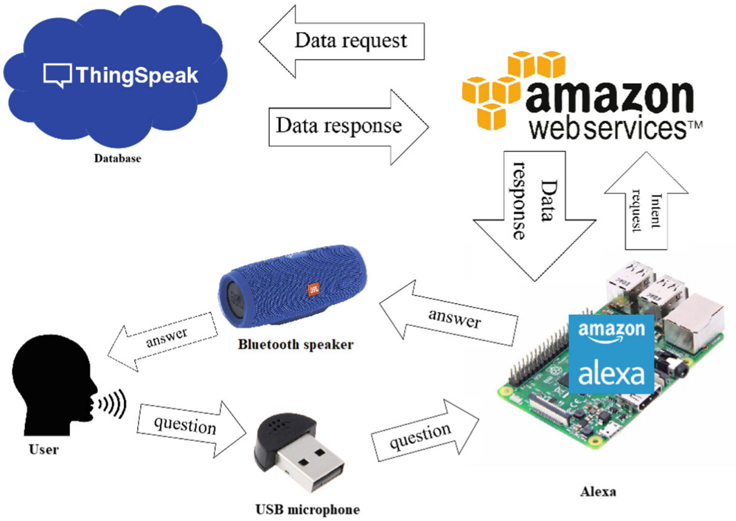

Fig. 4. Data flow: User asking Alexa about the air pollution

Lambda is a part of Amazon web services (AWS), which lets you run code on the cloud without worrying about the maintenance of the server [5]. The program can be written in a variety of languages and our project was developed in Python. Alexa uses a custom syntax, thus several tutorials with test programs [12] for learning exist. Lambda is programmed to receive a request by the AVS depending on the intent, processes the request and responds back with some information. This processing part is where the whole logic of the project is developed so that the user can be informed about the air pollution. For the purpose of this project, several functions were coded to respond to different intents sent by the AVS. The intents "indoor PM information", "outdoor PM information", "indoor carbon dioxide level" and "indoor temperature and humidity" when recognized in a sentence, trigger their corresponding Lambda function, where the parameters from the cloud are received, processed and sent back to the station where Alexa reads them to the user.

In the lambda code, the "urllib2" library is used to retrieve the data from ThingSpeak in a JSON format as shown on Fig. 6. This data is then converted to a plain string, which the custom functions process and output PM values used for evaluation. The most important functions that use this data are the PM evaluation and recommendation functions. Examples of these functions are given on Fig. 2b. In the PM evaluation, the particulate matters are ranked by their effects on human health, and when determining the pollution status they are compared by their effect index. The PM description words, depending on the PM values, are: Good, Moderate, Unhealthy and Very Unhealthy. The pollution evaluator functions for PM2.5 and PM10 works according to the European AQI scale [6], and the PM1 recommendation is based on the Dutch AQI scale, which is the only index scale for this size of particulate matter [1] (Fig. 5).

	Good	Moderate	Unhealthy	Very unhealthy
PM1	0-14	14-34	34-95	95+
PM2.5	0-20	20-25	25-50	50+
PM10	0-35	35-50	50-100	100+

Fig. 5. Classification of air quality based on PM concentration in parts per million [1,6] (Color figure online)

In this table, the first colored row represents the AQI descriptors used to help the users understand the PM pollution. Whereas, in the first column we have the three sizes of Particulate matter measured by the PMS5003 sensor. For each of these the particles, we have the scope of the quantity in ppm (particles per million) defined for each AQI descriptor. This information is used as input in the recommendation functions to generate a response for what activities the user can take at that moment, i.e. should he/she open a window to exchange air or should he/she take an activity outside. In fact, the most advanced function is created for the intent: "how is the air pollution". It requests a comparison between the indoor and outdoor particulate matter pollution and gives a recommendation for a preferred action the user should take considering the air pollution inside and outside of the user's home. This concludes the whole cycle from the user's question, to the reply he/she gets from an Alexa enabled device. The next section reviews the results from the measurements received from the devices.

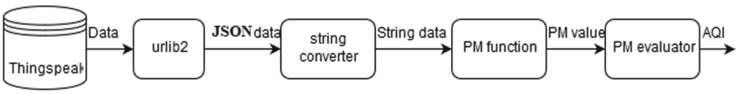

Fig. 6. Data interaction scheme

4 Results

In this section the successful implementation and results from the tests on the entire system are presented. The system, composed of the indoor and outdoor station, is shown on Fig. 7. It was built from scratch by combining all the components described in the previous sections. The communication between each of the components as well as uploading all data on the database is successful, thus the system works as expected.

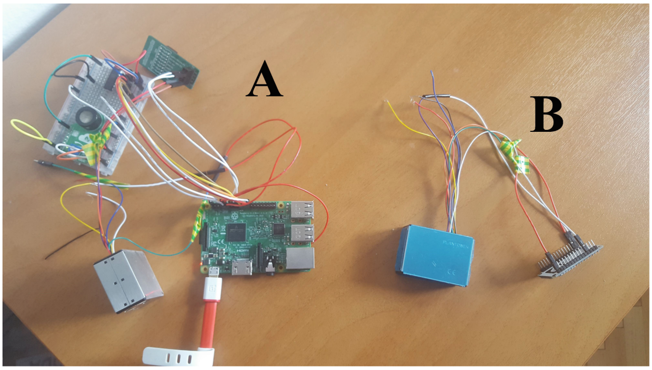

Fig. 7. The indoor station (A); The outdoor station (B)

The first step in determining the validity of our system was to compare some air pollution values between our outdoor station and the station owned by the Ministry of environment and physical planning of Macedonia located in the Municipality of “Centar” in Skopje. On Fig. 8 a graph for the PM values over a period of time is showcased. The results show that the “Centar” station shows 20–30% lower values for both PM2.5 and PM10. Note that, there are no measurements for PM1 values to be compared. This leads to an important difference in the classification of the air quality according to the index presented previously. For example in cases when our station shows an unhealthy air, the “Centar” station shows a moderately polluted air. Since the measurements are done only in a 2 h window (17:30–19:30), the authors believe that a comparison on a longer period is needed to obtain a better analysis. Another interesting observation from this test is that PM1 particulate matter values that are measured in our system are not measured in Macedonia, at least not to our knowledge. The official ministry’s stations do not measure this size of particle matter. The PM1 values for this 2-h period are around the unhealthy and moderate air quality range. As noted previously the PM1 values are the most detrimental to humans health and our system enables an improved information for the pollution.

Figure 9 shows the per minute values of the indoor station PM sensor, combined with the hourly measurements from the air quality station owned by the Ministry of environment and physical planning of Macedonia located in “Centar” in Skopje [2]. As shown on Fig. 9 the indoor station PM 2.5 values are higher than the values in the “Centar” station. One can note that staying in a closed environment can sometimes be harmful for our health even if we want to avoid

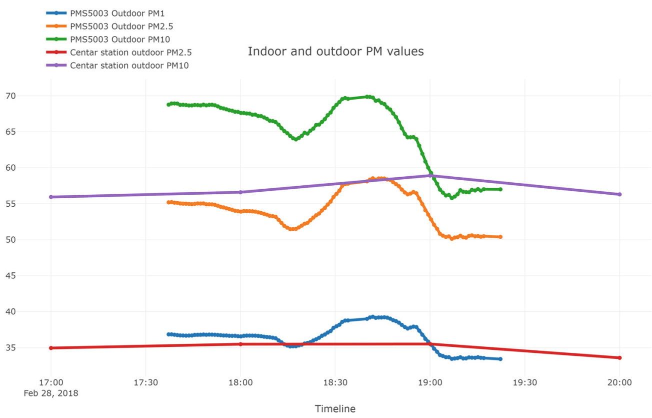

Fig. 8. Outdoor station and "Centar" station PM values over 2 h

the outdoor pollution. This is because the indoor PM2.5 levels are more influenced by human activities than outdoor PM 2.5 levels. This was also discussed in another paper following the indoor PM 2.5 correlation [28]. On the other side, the outdoor PM 10 levels are always higher than the indoor values, but they seem to be correlated. This is due to the way the experiment is performed. During this experiment the window was opened 3 times (at 20:00 on 10 March, 8:00 on 11 March and 00:00 on 12 March) for approximately 30 min. It resulted with all three PM particles experiencing a sharp increase of their values (Fig. 9). The peaks at night have a bigger value due to the fact that the outdoor air pollution was the highest in this period, and the "peak" in the morning has a lower value due to the fact that the pollution outside settles down, but still it is in the unhealthy range. After each event of opening the window first a "peak" in the PM values from the indoor station is noticed and than these values decreased slowly. This shows that opening a window has an effect on the air quality measurements of the indoor station. It suggests that the recommendation function implemented for our system would improve the air quality in home living since it would correctly recommend the user when to open the window. In this way the windows would not be opened in the period when it could have a negative effect on the indoor air quality. In the experiment two other events of cleaning and vacuuming the room happened at 8:00 and 12:00 on 12 March (Fig. 9). The goal was to see the human influence on the air pollution indoors. Both events resulted with sharp increases in the PM values. The "peak" in the morning has a slight lower value when comparing with the one at midday, due to the fact that through the day the pollution constantly increases because of other human activities.

Figures 10 and 11 show the measurements for the indoor CO_2 levels that resulted from two experiments that proved the validity and reactiveness of the CO_2 sensor. The first experiment was conducted by constantly exhaling air to the CO_2 sensor. This resulted with sharp peaks in the normal indoor carbon

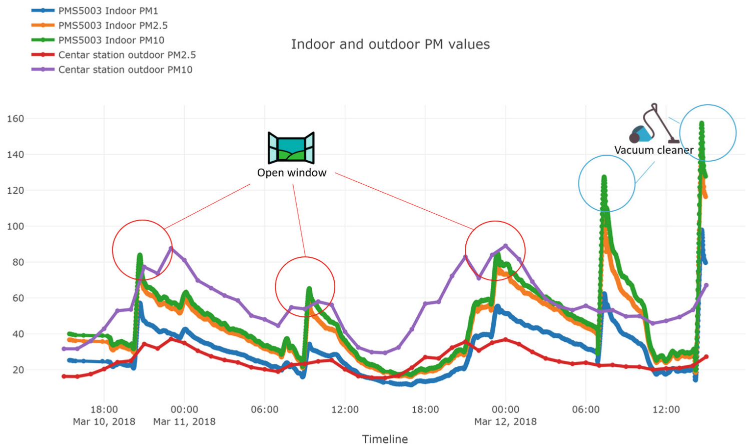

Fig. 9. Indoor station and "Centar" station PM values over 3 days

dioxide level of 500 ppm [11], jumping up to 4000 ppm and then slowly stabilizing back to its normal values. The second experiment was conducted using a Liquid Petroleum Gas lighter and directly applying it on the sensor. This had a similar effect, but with a much larger magnitude - a value of 1.6 million ppm. It has to be noted that due to this value the normal indoor carbon dioxide cannot be clearly seen on Fig. 11. The values in the millions are acceptable because the main purpose of the sensor is to monitor CO_2 levels, so when we get values out of the maximum range of CO_2 (250–5000 ppm) that should be a sign of heavy exposure to some other natural gas, because as noted in the sensor's description it is sensitive to different gases.

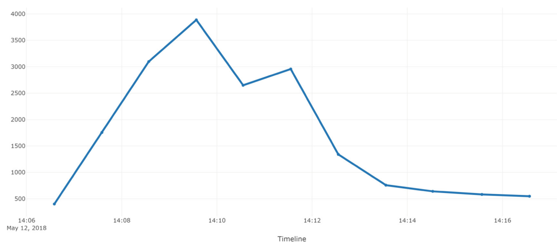

Fig. 10. Results from exposing the CO_2 values to CO_2 by exhaling

To sum up the experiments, when comparing our outdoor station with that of the "Centar" station, our results have more data due to the additional PM1 particles not covered by the "Centar" station, while giving more frequent information for the air quality right outside the user's home, and thus providing a more accurate pollution evaluation. The indoor station showed the importance of

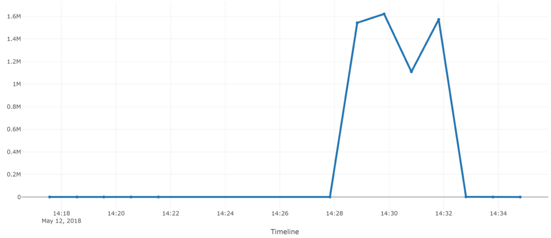

Fig. 11. Results from exposing the CO_2 values to LPG from a lighter

the human factor when it comes to the overall indoor air pollution. And finally, the two-part CO_2 experiment prove the validity of the sensor and its reactiveness to a sudden change in the air.

5 Conclusion

In this paper an affordable air pollution measuring system build for home living is presented. The system consists of both indoor and outdoor station. The outdoor station is measuring PM1, PM2.5 and PM10, while the indoor station is measuring PM1, PM2.5, PM10, CO_2, temperature and humidity. Both of the stations proved to be a success since they give information about the air we breathe not only outdoors, but also indoors where we spent most of our time. This is especially important in winter when the outdoor pollution is the highest. In addition to the measurements performed and storing the data, the voice-assistant was incorporated. It enabled easy and natural interaction between the users and the presented system. The voice assistant can inform the users about the air quality inside and outside and also it can recommend some actions for the users that would improve the air they breathe. As the technology advances, we can see a future where these sensors are integrated or connected with the personal assistants, like the Amazon Alexa devices, which will become a part of our daily life and it would be a "must" accessory of our living space.

In the future, it would be interesting to implement a node to node communication between the stations in a closed space, developing a web and mobile application which could further improve the access to information to the users' for the indoor station, but also the online community by making the outdoor stations open to everyone. By having multiple such stations across one city and with measuring the wind speed and direction, since the precise location of the station are known, we can help build a 3D pollution map of a city. It would show how the pollution is roaming around us, how outdoor pollution affects indoor pollution, and furthermore we can use machine learning to predict the pollution.

Acknowledgement. The authors would like to thank the Faculty of Computer Science and Engineering - Skopje for partially financing this work. All authors contributed equally to this paper.

References

1. Aireas.scapeler.com (2018). http://aireas.scapeler.com/index.php/AiREAS_AQI
2. Air.moepp.gov.mk (2018). http://air.moepp.gov.mk/
3. aqicn.org (2018). http://aqicn.org/sensor/
4. Co2.earth (2018). https://www.co2.earth/
5. Docs.aws.amazon.com (2018). https://docs.aws.amazon.com/lambda/latest/dg/
6. European environment agency (2018). http://www.eea.europa.eu/themes/air/
7. Github (2018). https://github.com/drohm/pi-sht1x
8. Github (2018). https://github.com/alexa/alexa-avs-sample-app/wiki/Raspberry-Pi
9. Hacker noon (2018). https://hackernoon.com/my-first-alexa-custom-skill-6a198d385c84
10. Iph.mk (2018). http://www.iph.mk/en/information-about-the-air-pollution-in-cities-in-the-republic-of-macedonia-and-possible-risks-to-health/
11. Kane international limited (2018). https://www.kane.co.uk/knowledge-centre/what-are-safe-levels-of-co-and-co2-in-rooms
12. Pythonprogramming.net (2018). https://pythonprogramming.net/headlines-function-alexa-skill-flask-ask-python-tutorial/?completed=/
13. Raspberrypi.org (2018). https://www.raspberrypi.org/documentation/installation/installing-images/README.md
14. Rigacci.org (2018). https://www.rigacci.org/wiki/doku.php/doc/appunti/hardware/raspberrypi_air
15. Webiopi.trouch.com (2018). http://webiopi.trouch.com/
16. Who.int (2018). http://www.who.int/mediacentre/news/releases/2014/air-pollution/en/
17. World bank (2018). http://www.worldbank.org/en/news/press-release/2016/09/08/air-pollution-deaths-cost-global-economy-225-billion
18. Chen, G.: Effects of ambient pm 1 air pollution on daily emergency hospital visits in china: an epidemiological study. Lancet Planet. Health **1**(6), e221–e229 (2017). https://doi.org/10.1016/s2542-5196(17)30100-6
19. Foxworth, T.: Making the esp8266 low-powered with deep sleep (2018). https://www.losant.com/blog/making-the-esp8266-low-powered-with-deep-sleep
20. Guxens, M., et al.: Air pollution exposure during fetal life, brain morphology, and cognitive function in school-age children. Biol. Psychiatry **84**(4), 295–303 (2018)
21. Institute for Health Metrics, E.G.B.o.D.P.: The state of global air (2018). https://www.stateofglobalair.org/sites/default/files/SOGA2017_report.pdf
22. Klosowski, T.: Lifehacker.com (2018). https://lifehacker.com/the-best-operating-systems-for-your-raspberry-pi-projec-1774669829
23. Kraujutis, V.: Viliuskraujutis/mq135 (2018). https://github.com/ViliusKraujutis/MQ135
24. Malvina, B., Roth, S., Sager, L.: Crime is in the air: the contemporaneous relationship between air pollution and crime - grantham research institute on climate change and the environment (2018). http://www.lse.ac.uk/GranthamInstitute/publication/crime-is-in-the-air-the-contemporaneous-relationship-between-air-pollution-and-crime/
25. Mohai, P., Kweon, B.S., Lee, S., Ard, K.: Air pollution around schools is linked to poorer student health and academic performance. Health Aff. **30**(5), 852–862 (2011). https://doi.org/10.1377/hlthaff.2011.0077

26. MSV, J., Bhartiya, S.: 10 diy development boards for iot prototyping - the new stack (2018). https://thenewstack.io/10-diy-development-boards-iot-prototyping/
27. Ristov, S., Ackovska, N., Kirandziska, V., Martinovikj, D.: The significant progress of the microprocessors and microcontrollers course for computer science students. In: 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 818–823, May 2014. https://doi.org/10.1109/MIPRO.2014.6859679
28. Song, P., Wanga, L., Hui, Y., Li, R.: PM2.5 concentrations indoors and outdoors in heavy air pollution days in winter. Procedia Eng. **121**, 1902–1906 (2015). https://doi.org/10.1016/j.proeng.2015.09.173
29. Sunyer, J., et al.: Association between traffic-related air pollution in schools and cognitive development in primary school children: a prospective cohort study. PLOS Medicine **12**(3), e1001792 (2015). https://doi.org/10.1371/journal.pmed.1001792
30. Zwozdziak, A., Sowka, I., Willak-Janc, E., Zwozdziak, J., Kwiecinska, K., Balinska-Miskiewicz, W.: Influence of pm1 and pm2.5 on lung function parameters in healthy schoolchildren-a panel study. Environ. Sci. Pollut. Res. **23**(23), 23892–23901 (2016). https://doi.org/10.1007/s11356-016-7605-1