national journal of application or innovation in engineering and management

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

# Internet of Things-Based Air Quality Monitoring System

**Saritha Narla**

**Tek Yantra Inc, California, USA saritha8083@gmail.com**

***Article Info***

***Received: 17-06-2015*** ***Revised: 21 -07-2015*** ***Accepted: 16-08-2015*** ***Published:06/09/2015***

**Abstract**: In order to showcase the concept of the Internet of Things (IoT) platform, this article builds an air quality measuring device, gathers and analyzes data in real-time, and then displays the results in a web application. An ESP8266 microprocessor powers the device's microcontroller board, the Wemos D1 Mini. Its other components are the CCS811 and BME280 sensors, which are responsible for measuring the number of air particles and the temperature, pressure, and humidity of the air, respectively. The gadget communicates with a MQTT broker via a Wi-Fi network by means of the MQTT protocol. Programming and setting the MQTT broker for both Raspberry Pi and Wemos D1 are also covered in the paper, along with thorough information on how the protocol works and how to use it.

**Keywords:** air quality control; IoT; Raspberry Pi; web application; Wemos D1 Mini

## INTRODUCTION:

The media has been heavy with references to the Internet of Things (IoT) in recent years. Manufacturers increasingly market once inconceivable Internet-connected devices as "smart," and it's almost impossible to locate a product that doesn't have an Internet of Things (IoT) version [1, 2]. Since the price of electrical components has dropped to an extremely low level and high-speed mobile networks are widely available, enthusiasts and amateurs may now build their own versions of these smart gadgets at a very cheap cost. Thanks to the proliferation of inexpensive IoT nodes, almost any object may be connected to the web and endowed with some level of digital intelligence; this opens the door to the prospect of hands-free, automated data exchange between gadgets, a phenomenon known as M2M communication [3-6].

The early adopters of the Internet of Things were mostly huge corporations and manufacturers. They found it fascinating and beneficial for machine-to-machine (M2M) communication and for tagging pricey manufacturing tools and equipment with Radio Frequency identification (RFID) tags. Building "smart spaces" in our homes and workplaces has been trending upwards as of late [7–8]. Automating mundane tasks like switching off lights and placing grocery orders frees up the user to focus on more meaningful activities, hence reducing stress [9].

Production productivity, unscheduled downtime in production due to failures, and electricity consumption have all been enhanced as a result of industrial IoT architectures' recent adoption of top data centers for data processing [10–14].

There are over a hundred mobile operators with NB-IoT or LTE-M networks, according to an announcement made in March 2019 by the Global mobile Suppliers Association (GSA). With 142 networks established or begun by September 2019, this figure has increased significantly [10]. Because it does not

national journal of application or innovation in engineering and management
ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

have any constraints on work cycles and runs on a licensed frequency band, NB-IoT is ideal for Internet of Things applications that need more frequent data transfer and communication [15, 16].

The primary motivation for this project was to design a battery-operated Internet of Things (IoT) sensor that could wirelessly transmit its measured values to an Internet-connected device, hence eliminating any limitations on its installation location. After then, users may access the sensor readings from any web-enabled device, such a smartphone, at any time.

The development of an air quality control sensor was selected because it detects the concentrations of different airborne particles. This kind of sensor might have many uses in mechanical engineering. For instance, it could provide metal processing businesses with real-time data on the air quality in their production halls, allowing them to monitor worker safety. If a procedure causes air pollution over a certain threshold, the system may alert the person in charge, who could then take measures like opening windows or removing employees from the affected area.

The air quality sensor (CCS811) and the atmospheric sensor (BME280) were linked to the development board (Wemos D1 Mini) based on the ESP8266 microcontroller via the I2C bus, making it the primary sensor unit. The data that the development board sent was received by a tiny computer called a Raspberry Pi 3B. In spite of its diminutive size (it's about the size of a credit card), the Raspberry Pi caused quite a stir a few years ago. It's circuitry is identical to that of a standard personal computer, albeit significantly less powerful. Nevertheless, it's powerful enough to initiate and carry out a plethora of tasks involving microcontrollers and sensors.

The air quality monitoring system's components, as well as the MQTT protocol and the Python script used to show the findings, are detailed in the work's second chapter. Chapter 3 presents the findings from real-world air quality measurements. This paper's conclusion is presented in chapter four.

## DESCRIPTION

With the development of the IoT industry over the past few years, some new forms of wireless communication between devices have emerged. Some of the protocols for wireless communication were created on the basis of wireless technologies that were used many years before IoT as a concept was even created, for example BLE (*Bluetooth Low Energy*) which arose from *Bluetooth Classic technology* that was primarily used to connect audio devices with computers or mobile phones.

The choice of protocol depends on the purpose of the IoT project. The basic selection criteria are consumption, range and bandwidth. Since IoT devices often do not have the ability to be powered by the power grid, they use batteries. In this case, it is important that the consumption of the device is as low as possible, so that its autonomy and service life are as long as possible. An autonomy of several years is often stated as a requirement for many IoT devices. As for the range, it can be from a few meters inside the building (e.g. Wi-Fi, BLE) to several kilometers in open space (e.g. LoRa – Long Range). Most often, as the range increases, the bandwidth, i.e. the amount of data sent by the IoT device, decreases, in order to keep the consumption as low as possible.

A brief description of the most commonly used wireless IoT protocols:

- **Wi-Fi** – the most common standard used in homes and businesses is 802.11n. It works on frequencies of 2.4 GHz and 5 GHz with a range of up to 50 meters. Data transfer speeds are a maximum of 600 Mbps depending on the channel frequency and the number of antennas (the latest 802.11-ac standard should offer 500 Mbps to 1 Gbps) [17],
- **NB-IoT** - *Narrowband Internet of Things* is a standard of radio technology, *Low Power Wide Area Network* (LPWAN), developed by 3GPP, the consortium in charge of network standards, for use in mobile devices and services [15, 16].
- **BLE** (*Bluetooth Low Energy*) – it is primarily intended as an upgrade of *Bluetooth Classic technology* for use in IoT devices. It is mostly used in devices in the health sector (constant control and measurements, for example: heart rate, glucose level, blood pressure, etc.), fitness, transmitters (beacon), security and home entertainment industry [18]. The bandwidth of BLE is usually in the range from 150 kbps to 1 Mbps [19],
- **LoRa** (*Long Range*) – it has the possibility of geolocation with a theoretical range greater than 10 km under ideal conditions [20].

### 2.1 Air Quality Supervising System Components

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

The device, the construction of which is explained in this paper, reads the quality of the air in the room, i.e. the proportion of carbon dioxide and the level of metal oxides, and at the same time measures the temperature, pressure and humidity of the air. CCS811 and BME280 sensors are used for the above and they represent the first layer of the IoT architecture. Wi-Fi was chosen as the communication technology. Data collection from sensors is done by Wemos D1 Mini, a development board based on the ESP8266 module, which sends data via Wi-Fi to a server built on a Raspberry Pi computer. Raspberry Pi contains a database in which sensor data is saved, as well as a *Web application* for displaying measurement results in text and graphic form.

The CCS811 (Fig. 1 and Fig. 2) was selected as the main sensor of the air quality control system. It is a very low- consumption digital gas sensor that detects a wide range of *Total Volatile Organic Compounds* (TVOC), including carbon dioxide equivalent ($eCO_2$) and metal oxides (MOX) levels. Volatile organic gases are often categorized as air pollutants and/or compounds that irritate the senses, and can appear as a result of evaporation from various building materials such as paints, as well as the result of the operation of photocopiers and the presence of people (e.g. breathing, smoking).


**Figure 1** CCS811 sensor [21]

This sensor is intended for use in closed spaces, and is used in smart devices such as smartphones and smartwatches, or in home automation systems. This paper will use the sensor design on a separate printed circuit board, which uses a standard $I^2C$ bus for communication. The unique $I^2C$ address is $0\times5A_{HEX}$.

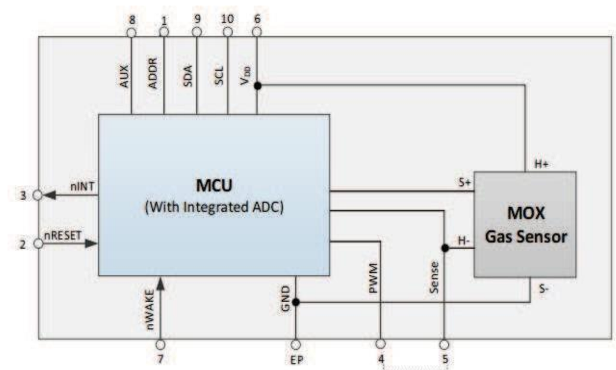


**Figure 2** CCS811 Air quality sensor Block Diagram [22]

Tab. 1 shows the characteristics of the CCS811 sensor according to [23].

**Table 1** Characteristics of the CCS811 sensor

| | |
|---|---|
| Power supply: | 1,8 V – 3,6 V |
| Current strength: | 30 mA |
| Power Consumption: | 60 mW (max) |
| Logic Level Voltage: | 3,3 V – 5 V |

$I^2C$ is a type of communication bus that enables the connection of *master* (control) and *slave* (peripheral) devices. As already explained, it uses two lines for communication: *Serial Data Line* (SDA) for sending and receiving data and *Serial Clock Line* (SCL) for sending *clock*. Connecting multiple *slave* devices using the same lines is possible because each device has a unique address.

All peripherals listen to the SDA line and monitor whether its address has been sent. The *master* sends the first data packet containing the start bit and the corresponding address of the peripheral device with which it wishes to communicate. The peripheral device responds to the sent request for communication, after which the *master* starts sending data. In order to achieve communication with several *slave* devices, a *pull-up* resistor is used. *Pull-up* is the name for a resistor that has the task of "pulling" both SCL and SDA lines to 5 V, i.e. into the logic unit [24].

In order to simplify the programming process and avoid writing complex algorithms, the *Adafruit_CCS811* library is used for programming the CCS811 sensor, according to [25].


**Figure 3** BME280 sensor [26]

In addition to the CCS811, a BME280 sensor is also used (Fig. 3 and Fig. 4). It is a combined sensor for measuring temperature, relative humidity and barometric air pressure. It has very small dimensions, low power consumption and is often used for home automation, in portable GPS modules or in smartwatches. It has high precision, so based on the pressure reading, it is possible to calculate the approximate altitude. As with the CCS811 sensor, a sensor design on a separate circuit board will be used. The standard $I^2C$ bus is also used for communication. The unique $I^2C$ address is $0\times76_{HEX}$.

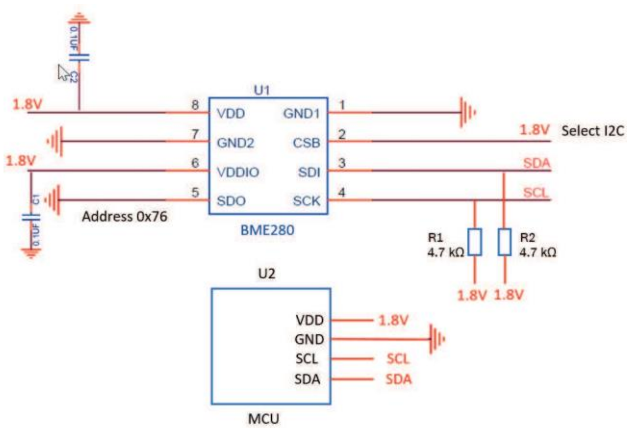


**Figure 4** BME280 sensor electronics scheme [27]

The sensor characteristics are listed in Tab. 2 according to [28].

In order to simplify the programming process and avoid writing complex algorithms, the *Adafruit BME280* library is used to program the CCS811 sensor, according to [29].

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

**Table 2** Characteristics of the BME280 sensor

| | |
|---|---|
| Power supply: | 1,71 V – 3,6 V |
| Current strength: | < 1 mA |
| Logic Level Voltage: | 3,3 V – 5 V |
| Temperature range: | –40 °C - 85 °C (± 1 °C) |
| Humidity range: | 0 - 100 % (± 3 %) |
| Pressure range: | 300 hPa- 1100 hPa (± 1 hPa) |
| Altitude range: | 0 m - 9000 m (± 1 m) |

The ESP8266 (Fig. 5 and Fig. 6) is a very low-cost 32- bit Wi-Fi microcontroller from *Espressif Systems* that fully supports the TCP/IP protocol. It contains a microprocessor running at 80 MHz or 160 MHz clock. It has a memory of 32 KiB for management commands and 80 KiB for user data. The operating voltage and the logic level voltage are 3.3 V and have 16 input/output pins. It can be used as a microcontroller or as a separate Wi-Fi module to supplement another microcontroller. It supports the IEEE 802.11 b/g/n Wi-Fi protocol [30].


**Figure 5** ESP8266 - 12E [31]

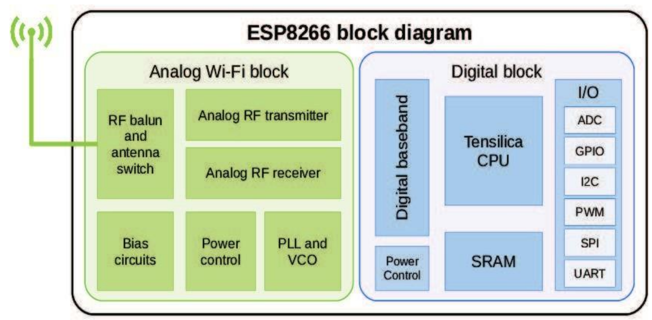


**Figure 6** ESP8266 block diagram [32]

Since the ESP8266 microcontroller is intended for SMD (*Surface Mount Device*) soldering and operates at 3.3 V, in order to facilitate connection with a computer and simplify programming, in this work we use the Wemos D1 Mini development board (in the latest revision named Lolin). This board is based on the ESP-12 version of the ESP8266 chip and enables easy connection to a computer via a USB port and programming in the *Arduino IDE* development environment. Due to the combination of microcontroller and Wi-Fi connectivity, it is an excellent choice for demonstrating the capabilities of the IoT platform.

The processor with a frequency of 80 MHz on the Wemos board has at its disposal an increased memory capacity for saving program code, with a capacity of 1 MB, of which 0.8 MB is available, and 0.2 MB is reserved for the bootloader, and working memory with a capacity of 82 kB, of which 49 kB available to the user. Wemos D1 Mini has support for $I^2C$, *Serial Peripheral interface* (SPI) and serial communication, contains nine digital input/output pins, all of which support *Pulse with modulation* (PWM) control and one analog input pin. In addition to all of the above, the board uses CH340 as a USB 2.0 interface, contains a 3.3 V regulator and four LED indicators. Before saving the program code, Wemos performs an automatic hardware reset [33].

In this work, the Wemos chip will collect data from the sensors and send them wirelessly to the Raspberry Pi via Wi- Fi, using the MQTT (*Message Queuing Telemtry Transport*) protocol. The Wemos board and his Pins are shown in Fig. 7 and Fig. 8.


**Figure 7** Wemos (Lolin) D1 Mini development board [34]

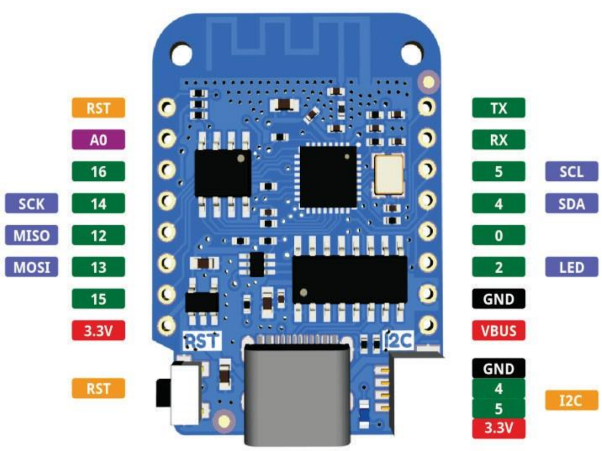


**Figure 8** Wemos (Lolin) D1 Mini Pins [34]

Raspberry Pi is a minicomputer developed by "The Raspberry Pi Foundation", and it is designed as a cheap and accessible tool for hobbyists, enthusiasts, and for the needs of learning programming in schools, especially in less developed countries. It has very small dimensions, the size of a credit card, and all components are located on one circuit board. It contains ports and connectors for connecting an external power supply, mouse/keyboard, and video outputs for connecting a monitor. The operating system is stored on a microSD card, and in this work, the Raspbian operating system, based on Linux, is used.

Fig. 9 shows the connection scheme of sensors CCS811 and BME280 with Wemos D1 Mini using an experimental board.

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

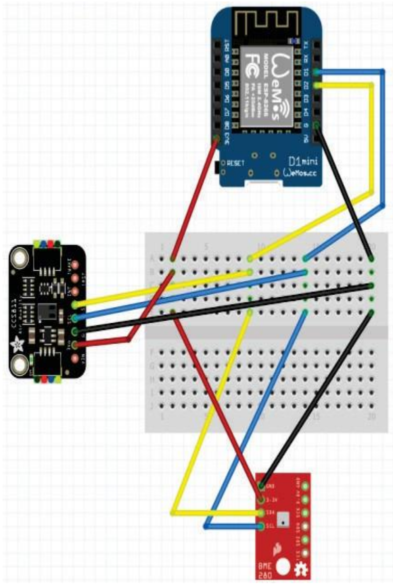

**Figure 9** Scheme of connection of Wemos D1 and sensor in the *"Fritzing"* program

Fig. 10 shows the layout of the board with soldered sensors and Wemos D1.

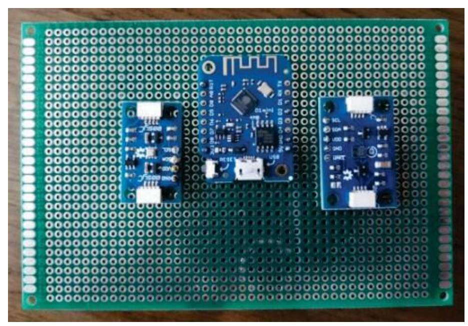

**Figure 10** Wemos D1 sensors soldered on the board

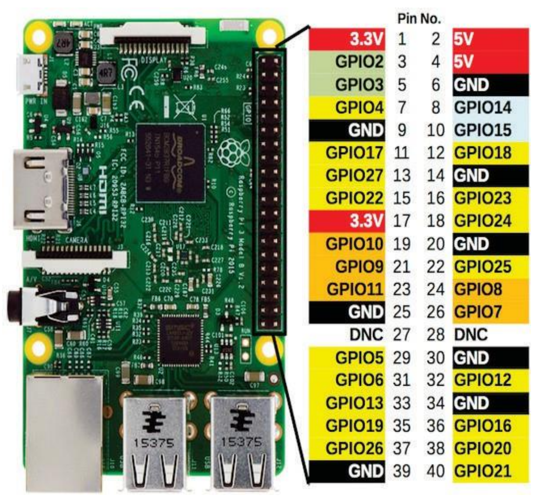


**Figure 11** Layout of GPIO pins on the Raspberry Pi board

The Raspberry Pi 3B used in this work, contains forty General Input Output Pins (GPIO). GPIO pins are divided into pins with a constant voltage of 5.0 V, with a constant voltage of 3.3 V, pins for grounding (Ground - GND), pins with an adjustable voltage and pins to which nothing is connected (Do Not Connect – DNC), according to [35].

Fig. 11 shows the arrangement of GPIO pins on the circuit board of the Raspberry Pi computer [36].

The Mosquitto MQTT broker and server will be configured on the Raspberry Pi minicomputer, which will enable local access to the Web application for displaying sensor readings.

### 2.2 MQTT Protocol Setup

MQTT is a communication protocol often used in IoT projects. It is responsible for data transfer management and is based on the *publish/subscribe* principle of operation. It is used when transmitting a small amount of data in remote locations, or in cases where a low bandwidth of wireless data transmission is available. It is simple and designed to be easily implemented [37]. It usually works over the existing TCP/IP protocol, but as a basis, it can also use other two-way protocols with loss protection (UDP protocol) [38].

Arlen Nipper and Andy Stanford-Clark created the first version of the MQTT protocol in 1999 [39, 40]. IBM released the next version (v3.1) in 2013, and then v3.1.1, which became the OASIS (*Organization for the Advancement of Structured Information Standards*) standard. The MQTT protocol started using the ISO standard in 2016 (ISO/IEC PRF 20922).

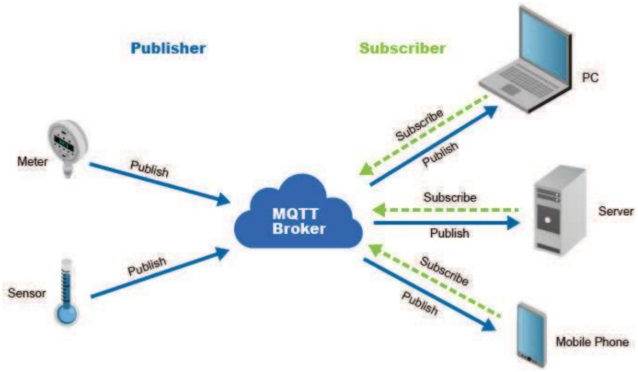


**Figure 12** The working principle of the MQTT protocol [42]

Fig. 12 shows the basic principle of operation of the MQTT protocol. The MQTT *broker* communicates with all *clients* (devices). On the left side are clients that measure certain values and publish this data, and on the right side are clients that process the data to which they are subscribed. The MQTT protocol works on the publish/subscribe principle [41]. Sensors that measure some physical quantities publish measurement results on a certain topic. Clients to whom information needs to be delivered are subscribed to that same topic. When a message is published on a certain topic, it is received by all clients who have subscribed to that topic.

The MQTT broker performs the task of "intermediator" between clients and thus is the central part of the whole publish/subscribe mechanism. The main tasks of the MQTT broker are: receiving messages from all clients who publish on a certain topic, filtering messages, deciding whether to forward specific messages and sending messages to subscribed clients.

The job of an MQTT *broker* can be performed by any computer whose specifications are powerful enough for the chosen *broker*, whether it is based on a Windows or Linux operating system. Today there are a large number of such programs. Some of the most used free MQTT *brokers* are Mosquitto, Emqtt, Hive MQ, MOSCA. In this paper, Mosquitto is used as a *broker*, which is configured on a Raspberry Pi 3B computer.

MQTT *clients* include subscribers and publishers. The MQTT client can simultaneously function as a subscriber (a client subscribed to a topic) and a publisher (a client publishes a message on a topic). The MQTT *client* task can be performed by a wide range of devices, from microcontrollers to personal computers, that is, any device that has implemented support for working with the MQTT protocol (MQTT libraries). MQTT libraries are available for a large number of programming languages: Java, JavaScript, C, C++, C#, as well as for the iOS and .NET platforms.

As an MQTT client, this paper uses a Web application made in NodeJS technology, which can be accessed in a local network, and is located in a server configured also on a Raspberry Pi computer.

*Quality of Service* (QoS) is an agreement between the publisher and subscriber of a message that defines the security of message delivery, depending on the level of the agreement.

In MQTT protocol, three levels of agreement are defined [43]:

- level 0: At most once – the message was sent only once, neither the client nor the broker perform any additional checks to see if the message was successfully received,
- level 1: At least once – the message is sent until the client confirms its receipt,
- level 2: Exactly once – the broker and the client exchange messages in two levels (two level handshake), in order to make sure that the message reaches the client exactly once.

Before installing Mosquitt, a software upgrade of the Raspberry Pi will be performed [44, 45].

*sudo apt – get update sudo apt – get upgrade*

After that, Mosquitto and its associated client package are installed.

*sudo apt-get install mosquitto -y*
*sudo apt-get install mosquitto-clients –y*

After installing these two packages, it is necessary to configure the broker. The Mosquitto broker configuration file is located at /etc/mosquitto/mosquitto.conf.

*sudonano /etc/mosquitto/mosquitto.conf*

```
// A function to connect Wemos to an MQTT broker  voidconnect_MQTT(){

Serial.print("Connecting to ");  Serial.println(ssid);

  // Connect to Wi-Fi  WiFi.begin(ssid, wifi_password);

  // Waiting until a Wi-Fi connection is established before continue the  program

while (WiFi.status() != WL_CONNECTED) {  delay(500);
Serial.print(".");
  }

  // Debugging - printing the IP address of Wemos D1  Serial.println("WiFi connected ");
Serial.print("IP address: ");
Serial.println(WiFi.localIP());

  // Connection to MQTT broker

if (client.connect(clientID, mqtt_username, mqtt_password)) {  Serial.println("Successful connection to
MQTT broker!");
  }
else {
Serial.println("Failed to connect to MQTT broker...");
  }
}
```

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

If the user does not want the broker to require a username and password from the subscriber clients, he omits the first two lines. The changes to the file are saved and the file is closed. After that, it is necessary to specify a username and password. Enter the following command: - replacing username, where username is the desired username. Then the desired password is entered and the Raspberry Pi is restarted with the command:

### Python Script for Displaying the Results

To display the values from the sensors, which the Raspberry Pi receives from Wemos via the MQTT protocol, a Python script was created that displays the results in a Web application (Fig. 13), which can be accessed within the local network. The web application has a simple design, adapted for display on smartphones. The function responsible for generating the Web application is according to [46].

At the beginning of the program code, we enter the libraries for using the CCS811 and BME280 sensors, as well as the *PubSubClient.h* library for using the MQTT protocol on the Wemos D1 board. The *ESP8266WiFi.h* library allows Wemos to connect to a Wi-Fi network. The topics that will be sent via MQTT are defined, as well as the username and password, which must correspond to those configured in the Mosquitto broker on the Raspberry Pi.

Function for connecting Wemos D1 to MQTT broker and sending data. The previously defined username and password are used. In case of an inability to connect, it returns a warning to the user.

MQTT can only send data in string form (character). Therefore, it is first necessary to convert the measured values from the sensor from the real (float) form into character notation. Then the value is sent (Publish) to the MQTT broker. At the end of the code, disconnection from the MQTT topic is performed.

## 3 MEASUREMENT OF AIR QUALITY IN REAL CONDITIONS

Using the developed device, air measurements were made in three different environments:

- an office room,
- an underground garage,
- an industrial welding hall.

The sensor read values every five minutes for two hours, and before starting the measurement, it worked for one hour to make the measurement as accurate as possible (in the CCS811 sensor documentation, at least half an hour of "warm-up time" is recommended).

The values of the quantities measured by the CCS811 sensor and their impact on people are shown in Tab. 3.

**Table 3** Characteristics of the BME280 sensor

| Concentration $CO_2$(ppm) | Impact on people | Concentration TVOC-a (ppb) | Impact on people |
|---|---|---|---|
| < 500 | normal | < 50 | normal |
| 500 - 1000 | a little uncomfortable | 50 -750 | uncomfortable, anxiety |
| 1000 - 2500 | tiredness | 750 - 6000 | headache, depression |
| 2500 - 5000 | harmful to health | > 6000 | headache, harmful to the nervous system |

The first measurement was made in an office room with twenty employees. The $CO_2$ value ranged from 300 ppm to 550 ppm, while the TVOC value ranged from 10 ppb to 45 ppb. After 45 minutes from the beginning of the measurement, the window was opened and the room was ventilated, which contributed to the reduction of the concentration of both quantities. The measured concentrations are within normal limits and do not cause any disturbance or discomfort to people (Fig. 14).

national journal of application of innovation
in engineering and management

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

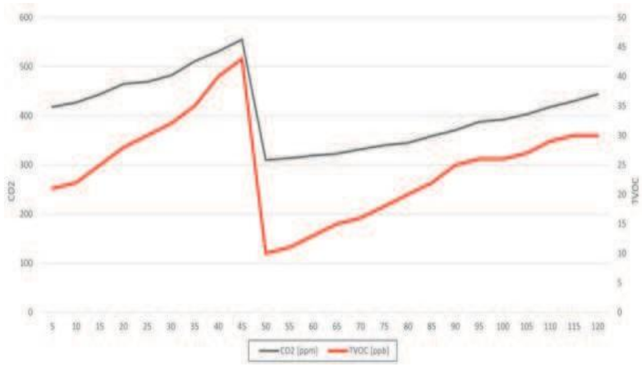

**Figure 14** Measured air quality in the office room

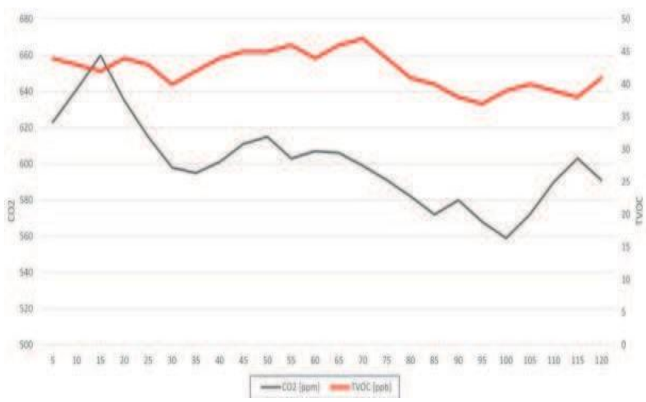

**Figure 15** Measured air quality in the underground garage

Other jobs (e.g. drilling, cutting, grinding metal). $CO_2$ concentration ranged from 1500 ppm to 4750 ppm, while TVOC values were between 900 ppb and 2500 ppb. The measured concentrations are very high due to jobs that bring a large amount of metal particles into the air, as well as lubricants, paints and solvents (Fig. 16). This kind of working environment requires the use of protective breathing masks (respirators), because this high concentration of substances in the air is very harmful to the health of employees.

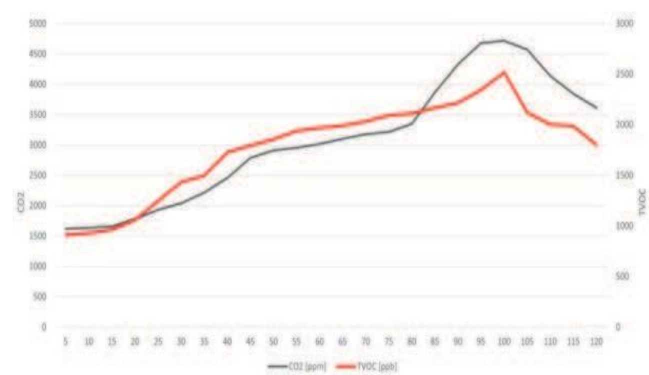

**Figure 16** Measured air quality in the welding hall

By performing the measurements, an idea was obtained for a possible improvement of the device, which is the automatic sending of an e-mail message or SMS to the responsible person in the company when the concentrations of $CO_2$ and TVOC exceed a certain value, in order to ventilate the hall.

# CONCLUSION

An air quality sensor is used to illustrate the idea of the Internet of Things. The majority of currently available items will likely integrate with the Internet of Things (IoT) in the not-too-distant future, allowing them to track or adjust certain settings, given that the IoT is a novel idea and platform. The Internet of Things (IoT) sector is ripe with opportunity for growth. Industrial facilities and home automation systems are only two of the many possible uses for the processed air quality control device. Parts that are both inexpensive and widely accessible make it possible to process and regulate data of varying amounts, and to store that data in a database where it is possible to access and compare values from different time periods. An example of this would be comparing the effects of several manufacturing processes to see if one led to better indoor air quality, which in turn made workers' working conditions better and reduced stress levels. Various microcontroller-based development boards, like the utilized Wemos D1 Mini, make it easier to build smart devices. These prototyping boards come with a plethora of pins and connections that make it easy to test and experiment with different kinds of sensors and other electrical devices. While mass-produced electronics rely only on microcontrollers and other essential components, there is a subset of the population that caters to hobbyists Development boards are an excellent tool for anybody interested in learning about electronics and programming. It was shown that inexpensive microcomputers, such as Raspberry Pi, can handle servers that manage smaller Internet of Things installations. As these computers continue to improve in price and specs year after year, it becomes more impossible to foretell what they will be capable of running in a few years. An office area, a subterranean garage, and a manufacturing welding hall were the three locations where air quality was measured. There is no evidence that the measured amounts in an office area are harmful or uncomfortable for the persons working there. The continual forced air exchange maintained a steady concentration level in the subterranean garage. Industrial welding produces very high amounts of metal, lubricants, paints, and solvents into the air, among other pollutants. Potential areas for further investigation include evaluating the processed air quality control device's effects on indoor air pollution levels, worker health, and overall productivity in various residential and commercial contexts. Investigating the device's economic viability in different settings and testing out other designs and features to make it more user-friendly and efficient are also potential areas for further study. Furthermore, further studies may be required to determine the pros and cons of combining the device with other smart home technology and environmental monitoring systems, as well as to investigate untapped uses and markets for processed air quality management devices.

## REFERENCES

Referenced in [1] Delija (2021). Monitoring the quality of the IoT platform during use. Croatian University North, Master's thesis. this link (in Croatian): https://urn.nsk.hr/urn:nbn:hr:122:241977 Retrieved on January 18, 2023.

In 2015, Al-Fuqaha, Mohammadi, Aledhari, and Ayyash published a study. The Internet of Things: An Overview of Open Standards, Protocols, and Use Cases.17(4), 2347-2376, published by IEEE Communications Surveys & Tutorials. Here is the link to the article: https://doi.org/10.1109/COMST.2015.2444095. H

It was published in 2018 by Amodu and Othman. Communication Between Machines: A Survey of Possibilities. Published in the journal Computer Networks, volume 145, pages 255-276. DOI: 10.1016/j.comnet.2018.09.001

By 2022, Uzair, Kafrawi, Janadi, and Bulushi had published their findings. A Low-Cost IoT Based Buildings Management System (BMS) Using Arduiono Mega 2560 and Raspberry Pi 4 for Smart Monitoring and Automation. Published in the journal IJECES, volume 13, issue 3, pages 219–236. DOI: 10.32985/ijeces.13.3.7

This information is sourced from Yassein et al. (2017). Survey and open numbers of MQTT for the Internet of Things protocol. The International Conference on Engineering and Management Information Systems (ICEMIS) held in Monastir, Tunisia, from 1 to 6. This article can be accessed at this URL: https://doi.org/10.1109/ICEMIS.2017.8273112.

[5]In 2017, Hedi, Špeh, and Šarabok published a work. Internet of Things (IoT) network protocols comparison for IoT limited networks.Volume 40, Issue 5, Pages 501–505, from the International Conference on Information and Communication Technology, Electronics and Microelectronics, held in Opatija, Croatia, May 22–26, 2017. The link to the article is https://doi.org/10.23919/MIPRO.2017.7973477.e.

[6]In 2022, Shafiq, Gu, Cheikhrouhou, Alhakami, and Hamam published a paper. A Look Back at the "Internet of Things" and Some Unanswered Questions About How to Stop Security Attacks Based on the "IoT" This is a reference to the article

e8669348 from WLAN and MC. This article is referenced in the given work.
[7]In 2023, Crnjac-Milić, Dujmenović, and Peko published a document. Logistics with the Internet of Things: A Methodology. Publication: Tehnički glasnik, 17.1., 134-140. the link to the article is https://doi.org/10.31803/tg-20220609190233.
[8]Mid-sized website. The IoT, and all its connected devices. Accessed on January 18, 2023, from https://medium.com/zeux/the-internet-of-things-iot-5-reasons- why-the-world-needs-it-125fe71195cc.
[9]S. R. Sakhare and B. A. Ahire (2021). Internet of Things (IoT) Smart Cities: No Noise Zone! Publication: Tehnički glasnik, volume 15, issue 1, pages 92-97. The DOI for this article is 10.31803/tg-20210205091148.
[10]A. Ramallo-González, F. Terroso-Saenz, A. González-Vidal, and A. F. Skarmeta (2019). Energy data administration and analysis made easy with an open Internet of Things platform. Computer Systems for the Next Generation, 92(2), 1066-1079. DOI: 10.1016/j.future.2017.08.046
the eleventhIn 2020, Sadhu, V., Zhao, X., and Pompili, D. published a paper. Energy-Efficient Analog Sensing for Dense, Large-Scale, Long-Term Wireless Monitoring! 544–559 in IEEE Internet of Things Journal, volume 7, issue 8. the link to the article is https://doi.org/10.1109/JIOT.2020.2984484. [Online].
In a 2020 publication, Casado-Vara, Martin-del Rey, Affes, Prieto, and Corchado were the authors. Optimizing algorithm performance in smart buildings by Internet of Things (IoT) network slicing on virtual homogenous data layers. Computer Systems for the Next Generation, 102, 965-977.Reference: [13] https://doi.org/10.1016/j.future.2019.09.042.This is the 2020 version of the work by Yang, Lee, and Li. An ideal decision-making approach for portfolios of intelligent building management systems geared toward the internet of things that combines activity-based costing with resource constraint analysis. Urban Sustainability, Volume 60, Issue 3, Pages 655-671. The link to the article is https://doi.org/10.1016/j.scs/2020.102142.
[14]Operation Mode for 4G/LTE NB IoT, as reported by ShareTechnote. The link to the handbook is https://www.sharetechnote.com/html/Handbook_LTE_NB_L " TE". Retrieved on January 18, 2023.
[15]The Avsystem. To begin with, what is Narrowband IoT? NB-IoT overview. Visit https://www.avsystem.com/blog/narrowband-iot/ for more information today. Retrieved on January 18, 2023.
[16]The internet of things for everyone. The function of WiFi in IoT. This is the link: https://www.iotforall.com/ wifi-role-intelligent Retrieved on January 18, 2023.
[17]Labs at Link. Implementations of Bluetooth Low Energy Messaging with the Internet of Things (IoT).Bluetooth and Bluetooth low energy are two different wireless technology standards. Retrieved on January 18, 2023.
[18]In 2012, Gomez, C., Oller, J., and Paradells, J. did research. Overview and Evaluation of Bluetooth Low Energy: An Emerging Low- Power Wireless Technology. The citation is from the journal Sensors, volume 12, issue 9, pages (11734–11753). The DOI for this article is 10.3390/s120911734.
The year 19This sentence is a citation for a 2017 paper by Adelantado, Vilajosana, Tuset-Peiro, Martinez, Melia, and Watteyne. The limitations of LoRaWAN may be understood. publication: IEEE Communications Magazine, volume 55, issue 9, pages 34–40. In a 2017 study, the authors found that... [20]CCS811 Air Quality Breakout by SparkFun, available at https://www.sparkfun.com/products/14193. [Online]. Retrieved on January 19, 2023.
Page 3 of the handbook has a block diagram for the AMS-Ultra Low Power Digital Gas Sensor for Monitoring Indoor Air Quality, according to InnovatorsGuru's [21] General Description. The file path to this document is: https://innovatorsguru.com/wp-content/uploads/2020/01/CCS811_Datasheet- DS000459.pdf. Thanks! This information was retrieved on January 20, 2023.
This is SparkFun.Characteristics of the CCS811 sensor. The file path of the CCS811 datasheet is: https://cdn.sparkfun.com/assets/learn_tutorials/1/4/3/CCS811.pdf. Retrieved on January 18, 2023.
[23] TheNXP. Information about the I2C-bus and how to utilize it.URL: https://web.archive.org/web/20210426060837/ Download the user manual at: https://www.nxp.com/docs/en/UM10204.pdf. Retrieved on January 18, 2023.
[24]The CCS811 Arduino library is available on GitHub. (Accessed 1/18/2023) https://github.com/maarten-pennings /CCS811.
Image source: distrelecwebshop [25]. image of the BME280 sensor. This image is located at https://www.distrelec.biz/Web/WebShopImages/landscape_la rge/2-/01/Adafruit-2652-30091192-01.jpg. Retrieved on January 18, 2023.
Specifically, Bosch [26]. Full diagram of a common application for the BME280. here: https://community.bosch-sensortec.com/t5/Knowledge- base/BME280-series-humidity-sensor-design-guide/ta-p/7385 Retrieved on January 18, 2023.

national journal of application or innovation in engineering and management

ijaiem.com/Sep 2015/ Volume 4/Issue 1/Article No-1/01-15
ISSN: 2319-4847

[27]I am Bosch. Specifications for the BME280 sensor. Bosch Sensortec's BME280 humidity sensors are available at https://www.bosch-sensortec.com/products/environmental-sensors. This information was retrieved on January 20, 2023.
[28] inThe engineers who work at the last minute. Manager for the Arduino library. Check out this Arduino bme280 tutorial at https://lastminuteengineers.com/. This information was retrieved on January 20, 2023.
[29]The Espressif ESP8266 is reviewed here. here: https://www.espressif.com/en/products/socs/esp8266 This information was retrieved on January 20, 2023.
[30]Picture of the ESP8266MOD from Auscomtech. the image URL is https://www.auscomtech.com.au/wp-content/uploads/2018/12/AD483-2.jpg. This information was retrieved on January 20, 2023.
[31]The authors of the 2019 article are Orellana, Macías, González-Velasco, Manso, and Gallardo-Caballero. Initial Action Period Measurements-Based Low-Power and Cost-Effective Internet of Things Electronic Nose for Environmental Conservation. Paper published in Sensors, volume 19, issue 14, page 3183. Accessed at https://doi.org/10.3390/s19143183
[32]This sentence is a citation for a paper by Asinelli, Serra, Marimon, and Espaulella (2018). This is the smARTS Museum V1, an open-source hardware system for keeping tabs on interior spaces that are important to cultural heritage. e00028, HardwareX, 4 (2010). The link to the article is https://doi.org/10.1016/j.ohxc.2018.e00028.
[33]This is Wemos.cc. Mini board: Wemos D1. Here is the link to the website: https://www.wemos.cc/en/ latest/d1/d1 little.html. This information was retrieved on January 20, 2023.
on page 34The Raspberry Pi. Raspberry Pi Technical Details. Here is the link to the Raspberry Pi 3 Model B: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/. I This information was retrieved on January 20, 2023.
[35]Raspberry GPIO, bigmessowires. GPIO Raspberry image, sourced from https://www.bigmessowires.com/wp-content/uploads/2018/05/Raspberry.jpg This information was retrieved on January 20, 2023. You can access the article at this link: https://doi.org/10.1007/978-1-4842-3948-3_1.
[36]In 2021, Tareq and Khaleel published a paper. Research on the Use of the MQTT Protocol in Internet of Things-Based Healthcare Settings. The citation is from IJECES, volume 12, issue 4, pages 215-223.
DOI: 10.32985/ijeces.12.4.5
[37]Blox from U-blox. "MQTT" defined. MQTT Beginners Guide: https://www.u-blox.com/en/blogs/insights/mqtt-beginners-guide (Get the data on 1/22/2023)
[38]K. Sugumar (2020). MQTT-aLightweight Communication Protocol Relative Study. Preprints by the author, 1–5. DOI: 10.22541/au.159076954.48540044
[39]With reference to Mishra (2018). TMCAS: A railway network collision avoidance system based on MQTT. Volume 18: Proceedings of the 18th International Conference on Computational Science and Applications, held in Melbourne, VIC, Australia, from July 2nd to the 5th, 2018, pages 1 and 6.You can access the article at this link: https://doi.org/10.1109/ICCSA.2018.8439562.
[40]Samir Jaloudi (2019). A comparison of the communication protocols used in an industrial IoT setting. Article 66 in Future Internet, volume 11, issue 3. Accessed at https://doi.org/10.3390/fi11030066.
[41] TheOn the ice. This is how the MQTT protocol works. Photo courtesy of OringNet: https://oringnet.com/upload/media/Technology/MQTT/MQT T_1091217-01.jpg This information was retrieved on January 25, 2023.
[42]This sentence is a citation for a work by Toldinas, Lozinskis, Baranauskas, and Dobrovolskis (2019). Utility of MQTT Service in Relation to Power Usage. Articles 1-4 from the 23rd International Conference on Electronics, Palanga, Lithuania, 17-19 June 2019, are published in the proceedings. See the link at https://doi.org/10.1109/ELECTRONICS.2019.8765692. Published in 2019.
[43]House on Your Own. Installing a MQTT Broker for Do-It-Yourself Home Assistant Devices. This is the link to the article: https://selfhostedhome.com/setting-up-mqtt-broker-for-diy-home-assistant-sensors/. This information was retrieved on January 25, 2023.
[44]We are PiMyLifeUp. Raspberry Pi Mosquitto MQTT Server Installation. Here's a link to this page: https://pimylifeup.com/raspberry-pi-mosquitto-mqtt-server/. This information was retrieved on January 25, 2023.
[45]Unstructured geek lessons. Libraries that support the MQTT protocol. Random Nerd Tutorials: https://randomtutorials.com/esp8266-nodemcu-mqtt-publish-dht11-dht22-arduino/ This information was retrieved on January 25, 2023.