Bilkent University

CS432 - Machine-to-Machine (M2M) Systems

Final Report

Product: Airify

Group: Air of Things

Efe Dağdemir - Project Leader & Architect

Utku Kalkanlı - Hardware Specialist

Ata Coşkun - Node Programmer

Tuana Türkmen - Integration and Server Side (GUI) Programmer

Introduction

Maintaining good air quality in closed spaces is a common problem that we face daily. Areas that are usually populated by crowds such as cafes, classrooms, offices, co-working spaces, meeting rooms etc. and areas with low human density that we spend a significant amount of our day like our living spaces mostly do not have any means of indicating the current quality of the air inside the area. This also includes detecting dangerous hazards such as gas leaks. Monitoring air quality in any given time is an effective step in sustaining healthy closed spaces and introduces a way to identify hazards before it results in any damages. In order to solve this problem, Air of Things(AOT) introduces Airify which monitors air quality in closed spaces, autonomously signals the pre-existing purification mechanism when needed and also provides the users with the current state of the air quality of the room. Airify achieves this with an IoT device suited with sensors and a mobile application interface. Users will be able to observe the quality of their room in any given time and manually control the purification mechanisms through the application.

Features of Airify

Air Quality Assessment: Airify will assess indoor air quality with an air quality sensor.

Detecting Explosive & Poisonous Gases: Dangerous gasses that pose life threatening risks such as poisonous, explosive gasses or smoke will be detected with dedicated sensors available in Airify.

Monitoring Statistics: Airify’s mobile application will show real-time measurements of its sensors to the users.

Swot Analysis

Strengths

- Low cost (₺800)
- Portability
- Wide range of target users
- We provide both air quality and gas leakage in same system
- Lots of competitors (Siemens, Testo,General Electric) has similar but more complicated(temperature,humidity ,LED display) and expensive products, we provide cheap and simpler solution with a mobile interface

Weaknesses

- Air quality measuring is a difficult task, air quality sensors are not precise when they are low cost
- Two distinct hardware platforms are needed at different places for air quality and gas leakage detection.

Opportunities

- Open to improvement
- Fast growing market
- Usage of cloud enables product to interact with other systems via internet, adding features is enabled in the future
- Young and agile company able to adapt changes

Threats

- Strong competitors e.g Airthings ($300) has similar indoor air quality product, but with higher price
- Low cost gas leakage products exist but do not measure air quality.
- Siemens, General Electric, has similar products which are doing this for many years(but has price between +300 Euro

Hardware Components

- ESP-32S Wifi + Bluetooth Combo Module(56,60 ₺)   x3
- MQ135 Sensor Module - Detection of NH3, NOx, benzene, smoke, CO2(air quality) (15,36 ₺)
- MQ-2 Sensor Module - Detection of gas leakage, LPG, i-butane, methane, propane, hydrogen, alcohol, smoke (14,31 ₺)
- MQ5 Sensor Module - Detection LPG, natural gas and town gas leakage (14,15 ₺)

Hardware Cost: 213.62 ₺

Reliability Concerns

- MQ sensors come with a cheap price but it does not take away from their reliability
- MQ sensors are adjustable in more than one way
    - Sensitivity of digital pins can be adjusted with potentiometers
    - Sensors can be calibrated with load resistors for various gas concentrations
- Numerous MQ libraries exist, providing easy ways for complicated tasks

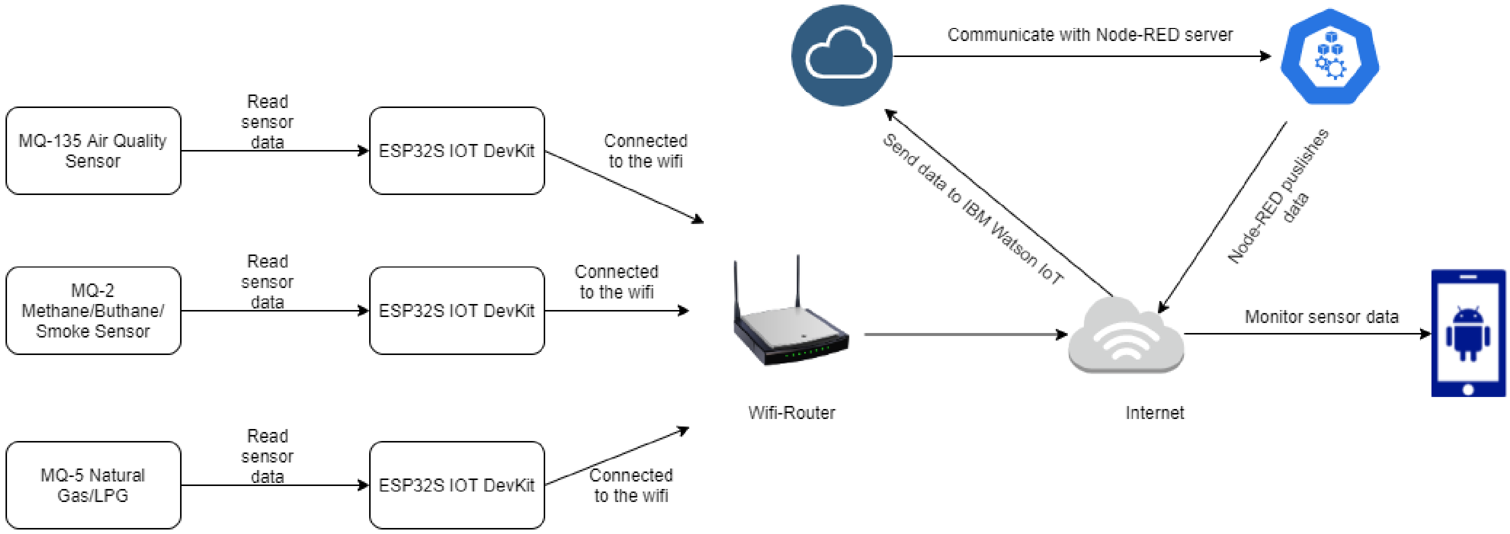
Placement Concerns

- Airify provides detection for various types of gasses, but not all gasses are detectable in the same way
- Town gasses, natural gasses and gasses measured for air quality assessment are lighter and they rise, while LPG is heavier and it sinks
- Airify is a portable product with three different parts, the air quality measuring part, the methane/butane/smoke detecting part and the natural gas/lpg detecting part. The placement of individual parts will be different depending on their functionality.
- The recommended placement for the first two parts of the device will be a high placement, such as the ceiling of the room.
- For the natural gas/lpg detecting part of the device, users who would like to be

precautious about heavier gasses will be guided to a lower placement, for all the other conditions the default will be a higher placement.

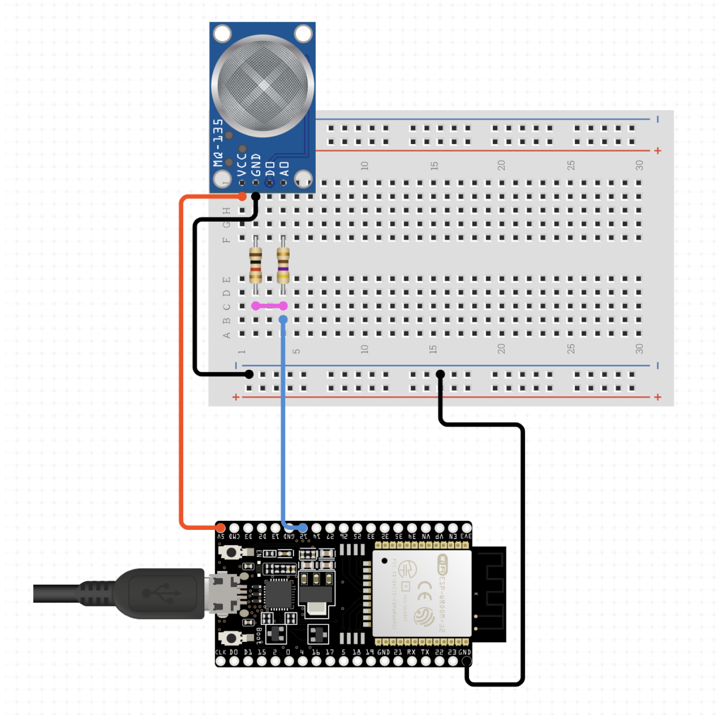
- Users will be informed about this preemptively and thoroughly on the packaging and the user manual

Block Diagram of the System



Block Diagram of the System

Hardware Schema



Similar Products

Cubona Air Quality Monitor

- Detects and records formaldehyde, carbon dioxide (CO2) and harmful gases
- Real-time testing of temperature and humidity values
- Calendar, clock and alarm functions

Airify will not have the last two qualities listed here but measures more gases and it provides an application for users to monitor the statistics.
Price: $89.99

Siemens QMX3.P70

- Only measures temperature, humidity and CO2
- Siemens has several other devices for air quality measuring but they are industrial scale with no commercial use, Airify is designed for commercial use.

Price: 390€

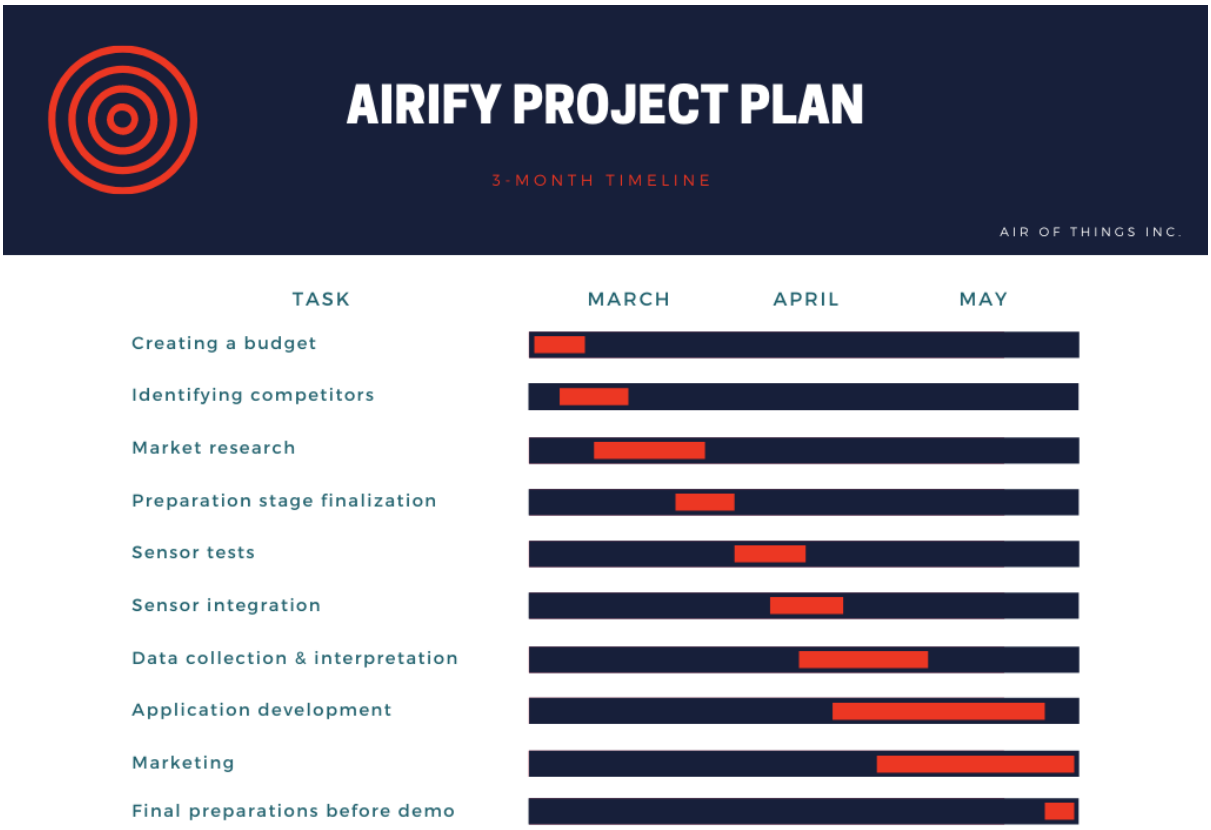
HSTMYFS Natural Digital Gas Detector

- Detects LPG (Methane), natural gas, coal gas, gas fuel
- Makes a sound warning
- Alarm thickness: LPG: 0.1~ 0.5%; natural gas: 0.1~ 0.3%; coal gas 0.1~ 0.5%

This product detects gas leaks similar to Airify however, it lacks air quality measurements while Airify measures both. And again Airify provides an application for users to monitor the data.

Price: $17.99

Project Plan



Gantt Chart of Airify's Project Plan

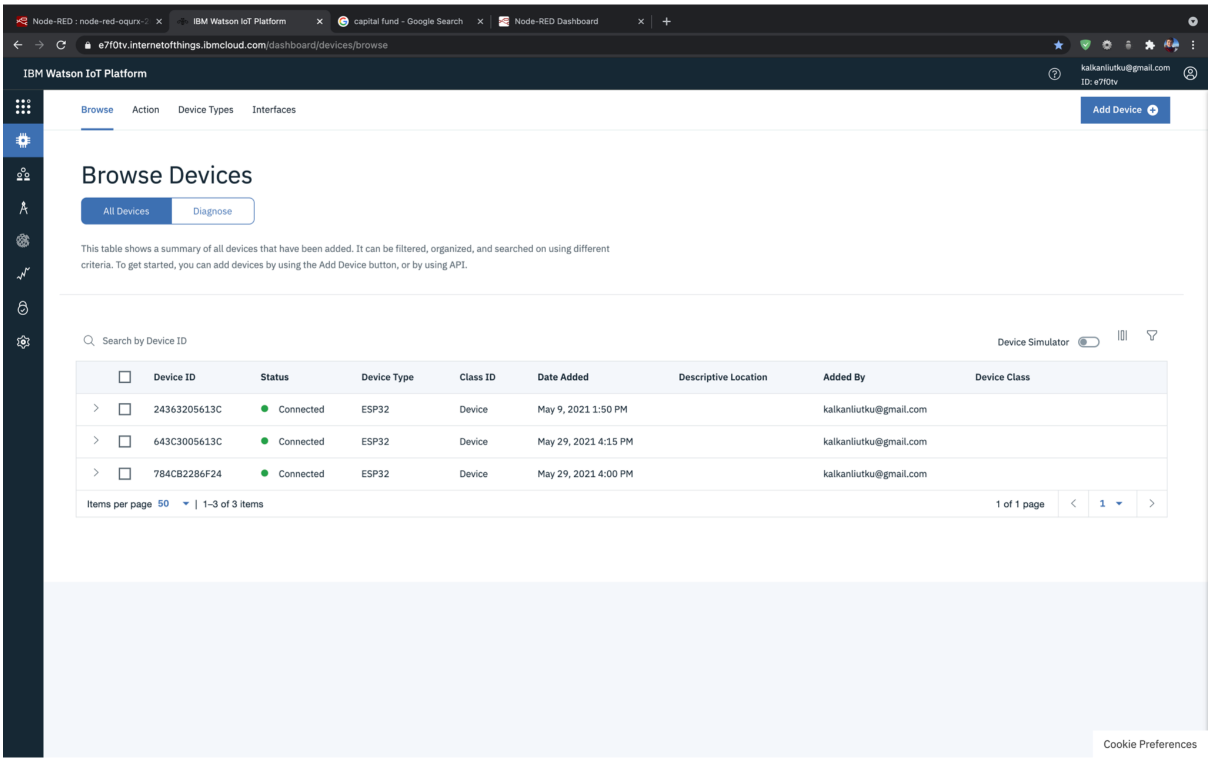
Financial Plan

	2021	2022	2023	2024	2025
Customers	--	6.5K	32K	88K	250K
Revenues	₺0	₺5M	₺25M	₺70M	₺200M
Operating Cost					
Research & Development	₺50K	₺150K	₺400K	₺750K	₺1.5M
Employee Wages	₺240K	₺1,728M	₺6,564M	₺16M	₺45M
Sales & Marketing	₺50K	₺300K	₺700K	₺1.5M	₺3M
General & Administrative	₺60K	₺150K	₺450K	₺1M	₺1.5M
Production Cost	₺80K	₺150K	₺1.5M	₺15M	₺100M
Total Cost	₺480K	₺2.479M	₺9.5M	₺34M	₺151M
Net Income	₺520K	₺5.521M	₺23.5M	₺76M	₺121M
Taxes	-₺208K	-₺2.208M	-₺9.4M	-₺30M	-₺48.4M
Profits	₺312K	₺3.31M	₺14.1M	₺46M	₺72.6M
Fund Raising					
Total Investment	₺1M	₺3M	₺8M	₺40M	₺72M

Goals Achieved

Air Quality Assessment: Managed to correctly integrate the MQ135 Air Quality Sensor with the ESP32 module. Therefore, the quality of the air is actively measured in real time.

Detecting Explosive & Poisonous Gases: Managed to correctly integrate the MQ2 and MQ5 Sensors with their ESP32 modules. Thus the explosive and poisonous gas amount in the air is actively monitored in real time.



Browse Devices

All Devices | Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID

Device Simulator

	□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
>	□	24363205613C	● Connected	ESP32	Device	May 9, 2021 1:50 PM		kalkanliutku@gmail.com	
>	□	643C3005613C	● Connected	ESP32	Device	May 29, 2021 4:15 PM		kalkanliutku@gmail.com	
>	□	784CB2286F24	● Connected	ESP32	Device	May 29, 2021 4:00 PM		kalkanliutku@gmail.com	

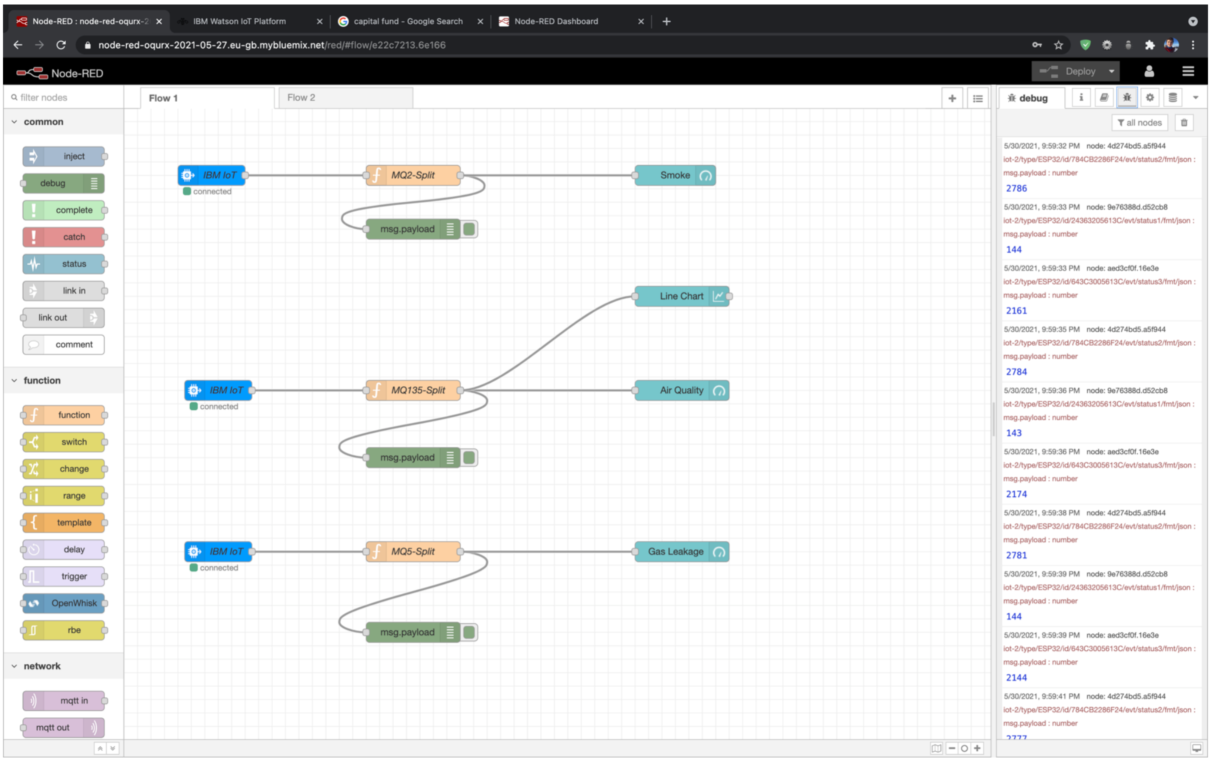
Items per page 50 | 1–3 of 3 items

1 of 1 page

Cookie Preferences

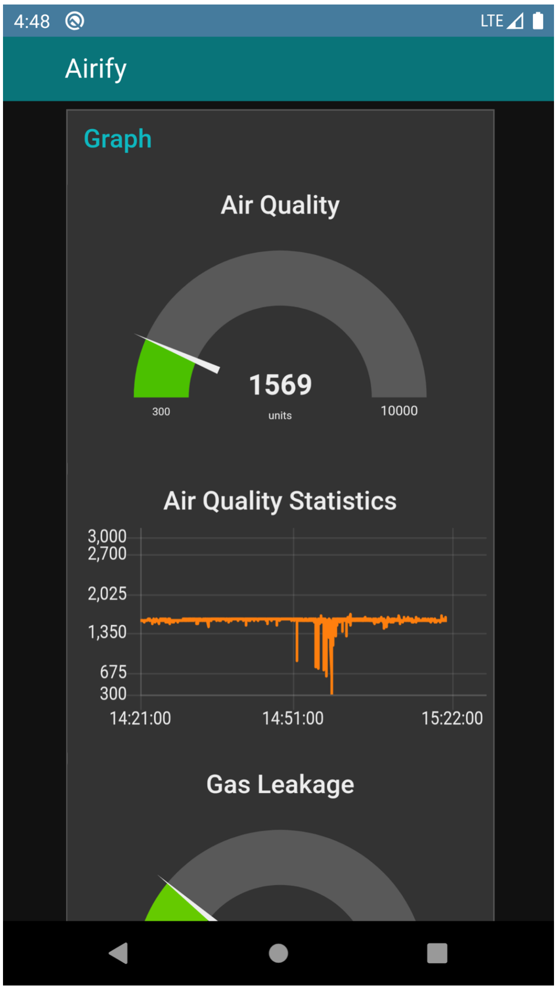
IBM Watson IoT Platform with connected devices

Monitoring Statistics: An Android application called Airify was implemented and the application was connected to the devices on the IBM cloud platform via Node-RED. In this application, the data which is read in real time is displayed via a line chart and two semi-circle pie charts. The displayed data changes according to the data being read actively. There are color indicators on the charts which provide information about the condition of the air quality and presence of dangerous gas. (ranging from red being dangerous to green being safe).

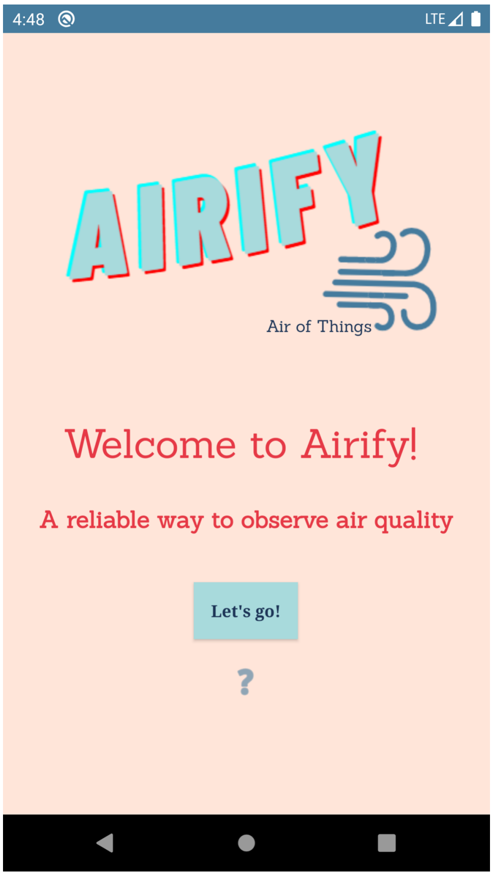


```
5/30/2021, 9:59:32 PM   node: 4d274bd5.a5f944
iot-2/type/ESP32/id/784CB2286F24/evt/status2/fmt/json :
msg.payload : number
2786
5/30/2021, 9:59:33 PM   node: 9e76388d.d52cb8
iot-2/type/ESP32/id/24363205613C/evt/status1/fmt/json :
msg.payload : number
144
5/30/2021, 9:59:33 PM   node: aed3cf0f.16e3e
iot-2/type/ESP32/id/643C3005613C/evt/status3/fmt/json :
msg.payload : number
2161
5/30/2021, 9:59:35 PM   node: 4d274bd5.a5f944
iot-2/type/ESP32/id/784CB2286F24/evt/status2/fmt/json :
msg.payload : number
2784
5/30/2021, 9:59:36 PM   node: 9e76388d.d52cb8
iot-2/type/ESP32/id/24363205613C/evt/status1/fmt/json :
msg.payload : number
143
5/30/2021, 9:59:36 PM   node: aed3cf0f.16e3e
iot-2/type/ESP32/id/643C3005613C/evt/status3/fmt/json :
msg.payload : number
2174
5/30/2021, 9:59:38 PM   node: 4d274bd5.a5f944
iot-2/type/ESP32/id/784CB2286F24/evt/status2/fmt/json :
msg.payload : number
2781
5/30/2021, 9:59:39 PM   node: 9e76388d.d52cb8
iot-2/type/ESP32/id/24363205613C/evt/status1/fmt/json :
msg.payload : number
144
5/30/2021, 9:59:39 PM   node: aed3cf0f.16e3e
iot-2/type/ESP32/id/643C3005613C/evt/status3/fmt/json :
msg.payload : number
2144
5/30/2021, 9:59:41 PM   node: 4d274bd5.a5f944
iot-2/type/ESP32/id/784CB2286F24/evt/status2/fmt/json :
msg.payload : number
2777
```

Node RED server with values coming from IBM Watson IoT

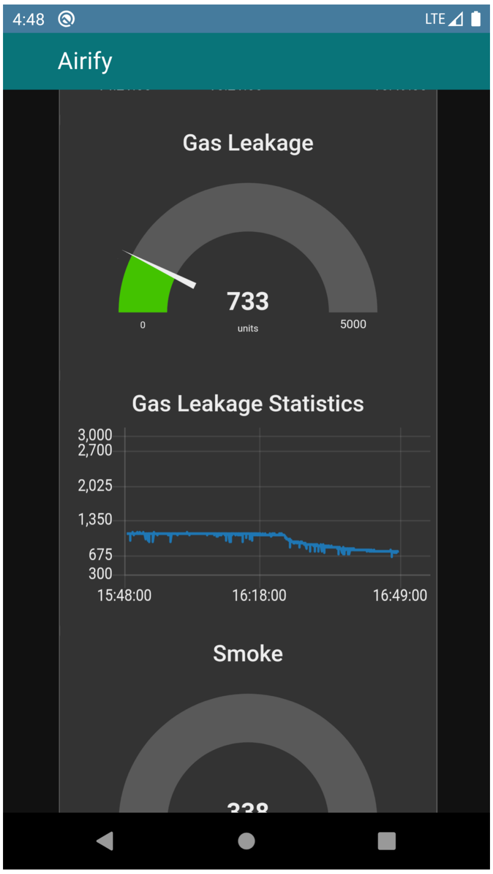


Android application interface with monitoring data coming from Node-Red server.



Welcome to Airify!

A reliable way to observe air quality



Future Plans

Communicating with autonomous air conditioning and purifying systems: The future will bring autonomous ventilation systems for closed spaces such as automatic windows, ventilators, air conditioners and air purifiers. Communicating with these systems to ensure the safety of the closed spaces if a dangerous situation arises will be one of the next steps to be taken.

Informing users in case of dangerous situations: In case of these situations, Airify will alert the users via push notifications. Additionally, according to the emergency of the situation, loud alert alarms will be effective.

Changes in Implementation

Initially, the aim was to integrate the sensors that are used in Airify with one ESP32 module and acquire one device. However, it was decided that this would not be the optimum design choice. As an illustration, it would be sensible to put Airify near a fire/gas source (oven, fireplace etc) so that the device can sense if a dangerous situation arises like a fire. Yet if the device is put close to a fire source, then the measured air quality would not reflect the general air quality of the closed spaced. The problem remains in the opposite situation too, if the device is put far away from a fire/gas source, the air quality could be measured correctly but gas leakages may not.

Thus after some evaluations, it was decided to break Airify into three devices. Now every sensor is connected to one ESP32 module and this way, the users will be able to place the respecting devices in correct locations. For example, the devices which measure gas leakages can be put in the kitchen and the device that measures air quality can be put in any space where it won’t be affected by daily life gases.

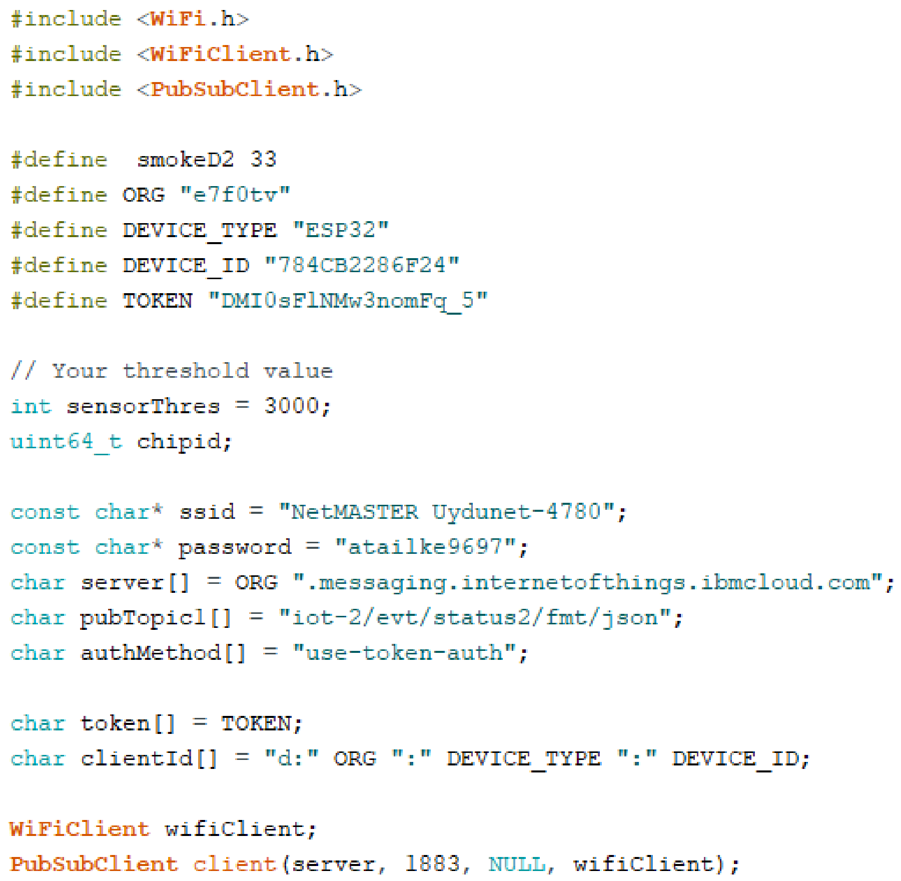
Risks Encountered

During the implementation process of Airify, two risks had to be dealt with.

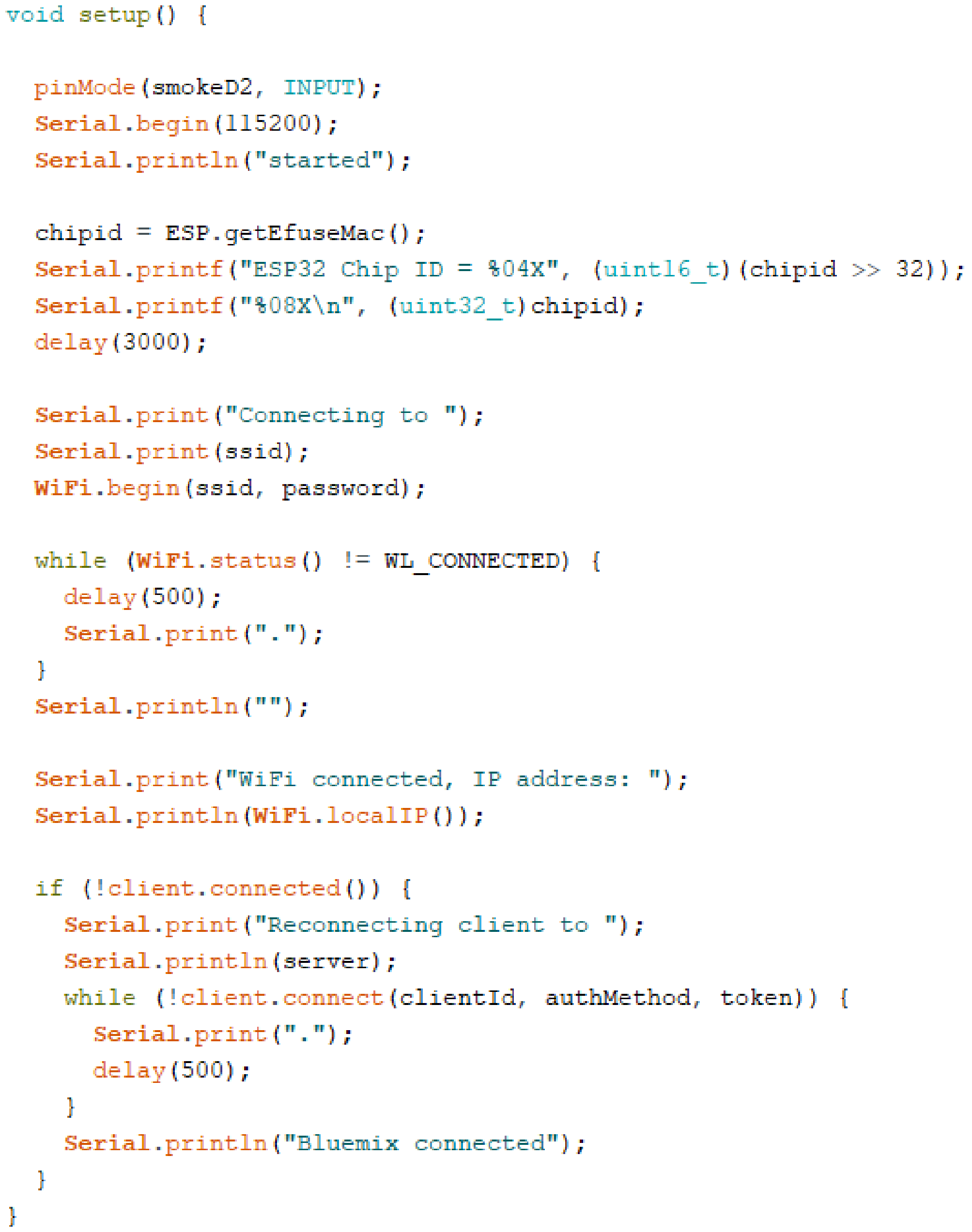
The first problem occurred because of the restrictions of Covid19. Because of the curfew, one of the members of the team wasn't able to access the hardware material since the material was in another team member’s presence when the lockdown was announced. However, it had been predicted that there was a chance Covid19 could create a problem. Thus, a backup plan (a more flexible one) about the distribution of the tasks was used for a couple of days.

Another risk that was taken was changing the integration plan mid-implementation as told above. However, it had to be done and it was foreseen that this change could be handled within the deadline.

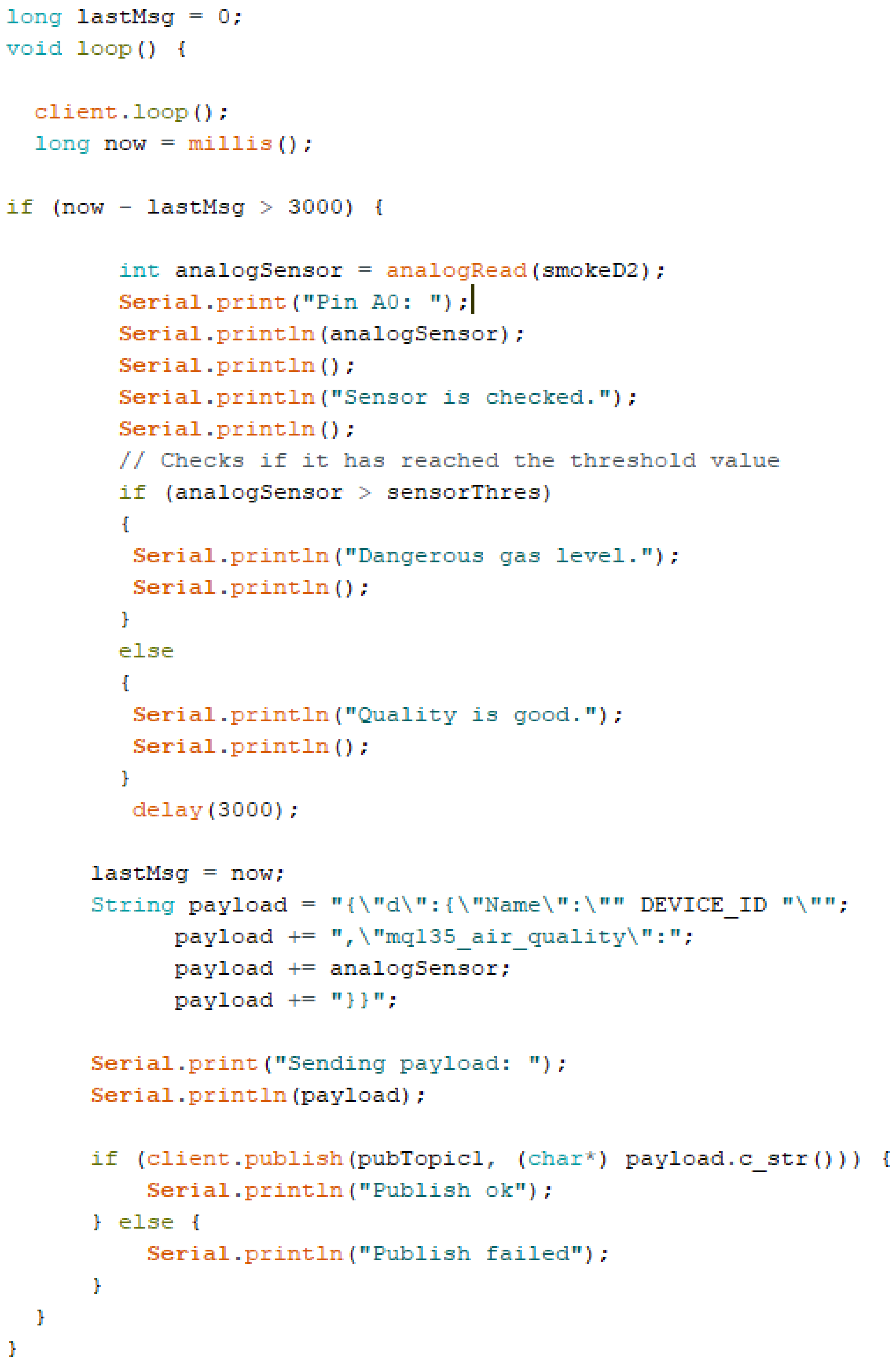
Arduino Code for Airify



```
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>

#define  smokeD2 33
#define ORG "e7f0tv"
#define DEVICE_TYPE "ESP32"
#define DEVICE_ID "784CB2286F24"
#define TOKEN "DMI0sFlNMw3nomFq_5"

// Your threshold value
int sensorThres = 3000;
uint64_t chipid;

const char* ssid = "NetMASTER Uydunet-4780";
const char* password = "atailke9697";
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1[] = "iot-2/evt/status2/fmt/json";
char authMethod[] = "use-token-auth";

char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);
```



```
void setup() {

  pinMode(smokeD2, INPUT);
  Serial.begin(115200);
  Serial.println("started");

  chipid = ESP.getEfuseMac();
  Serial.printf("ESP32 Chip ID = %04X", (uint16_t)(chipid >> 32));
  Serial.printf("%08X\n", (uint32_t)chipid);
  delay(3000);

  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);

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

  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());

  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
    Serial.println("Bluemix connected");
  }
}
```



```
long lastMsg = 0;
void loop() {

  client.loop();
  long now = millis();

if (now - lastMsg > 3000) {

        int analogSensor = analogRead(smokeD2);
        Serial.print("Pin A0: ");
        Serial.println(analogSensor);
        Serial.println();
        Serial.println("Sensor is checked.");
        Serial.println();
        // Checks if it has reached the threshold value
        if (analogSensor > sensorThres)
        {
         Serial.println("Dangerous gas level.");
         Serial.println();
        }
        else
        {
         Serial.println("Quality is good.");
         Serial.println();
        }
         delay(3000);

      lastMsg = now;
      String payload = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
            payload += ",\"mq135_air_quality\":";
            payload += analogSensor;
            payload += "}}";

      Serial.print("Sending payload: ");
      Serial.println(payload);

      if (client.publish(pubTopic1, (char*) payload.c_str())) {
          Serial.println("Publish ok");
      } else {
          Serial.println("Publish failed");
      }
  }
}
```