Air Quality Monitoring System

A System designed to alert the user about possible Air Quality condition of particular city.

Things used in this project

Buzzer

Male/Female Jumper Wires

USB-A to Micro-USB Cable

BOLT IOT BOLT WIFI MODULE

Software apps and online services

BOLT IOT BOLT COLUD

Bolt IoT Bolt Python Library

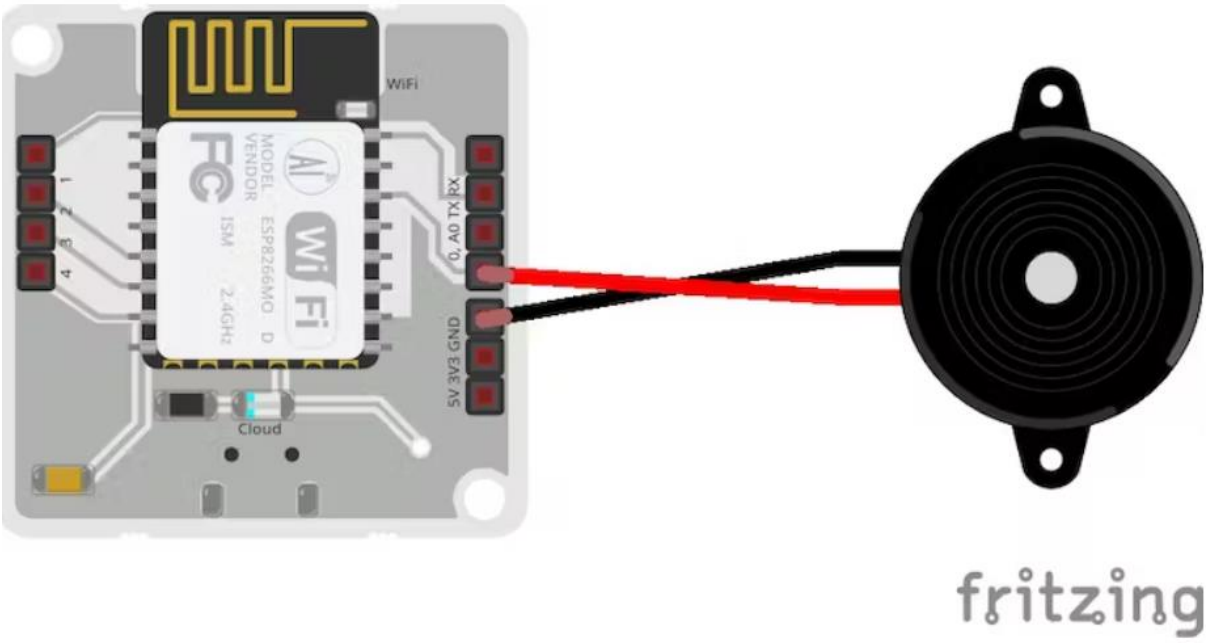
Air Quality Programmatic APIs

Telegram API

SNAPPY UBUNTU CORE

Step 1. Hardware Setup

Plug the longer end of the Buzzer to the Pin 0 of Bolt WIFI module and the shorter end to the ground pin (GND) using Male/Female Jumper Wire and connect the bolt IoT Wi-Fi module to laptop or any USB power source using USB A to micro USB cable.



Step 2. Get the Bolt API key and Device ID

1. Login to your Bolt Cloud and note down the Device ID under Devices section.

2. Now copy the API key generated under the API section, you can also generate a new API key by clicking “GENERATE NEW API KEY”.

3. Your API Key will generally look like this-

Fkfjfm8-dssfsf-ghgfsa-vfffv55-sds5oi (just for illustration)

Step 3. Setup your environment

For this project I am installing an Ubuntu on VMware, you can also dual boot Ubuntu with your windows 10.

To install Ubuntu server image on VMware, follow the below steps –

1. Go to http://releases.ubuntu.com/18.04/ and download server image as shown in below image

2. Download VMware workstation for windows from below link and install -

Note – Make sure you have high-speed internet

3. After installing click on create a new virtual machine and select the “Installer Disk Image file(iso)” option. Next, browse and select the above-downloaded image and click Next.

4. Fill the username and password in the next window and click next.

5. Enter 10Gb for “Maximum disk size” and select “Store virtual disk as single file”.

6. Configure the hardware if necessary and click finish.

7. Let it install and reboot.

8. Voila! You have installed Ubuntu on VMware successfully.

9. Now lets update and setup necessary packages, enter the following commands one by one

Step 4. Let’s create a telegram channel and a bot

1. We will be making use of the telegram channel to broadcast Air Quality alerts periodically.

2. First, create a new channel in telegram and provide a suitable name and description as shown below.

4. Now let’s create a bot for our channel. A bot is a third-party application that runs inside the telegram, we will use this bot to send alerts to our channel.

5. On the home screen of telegram, click on the search icon on the top right corner and search for botfather.

6. In search results, select the botfather which has a blue tick mark next to it.

7. The chat shows few possible commands, type “/newbot” to create a new bot.

8. Give proper name and username to your bot.

9. Once the bot is created successfully it generates a token. This token is used to send and receive messages through python.

10. Please save this note as **“telegram_bot_id”** where it is saved as “bot” followed by bot token.

Step 5 – Let's create an account and get API token from AQICN

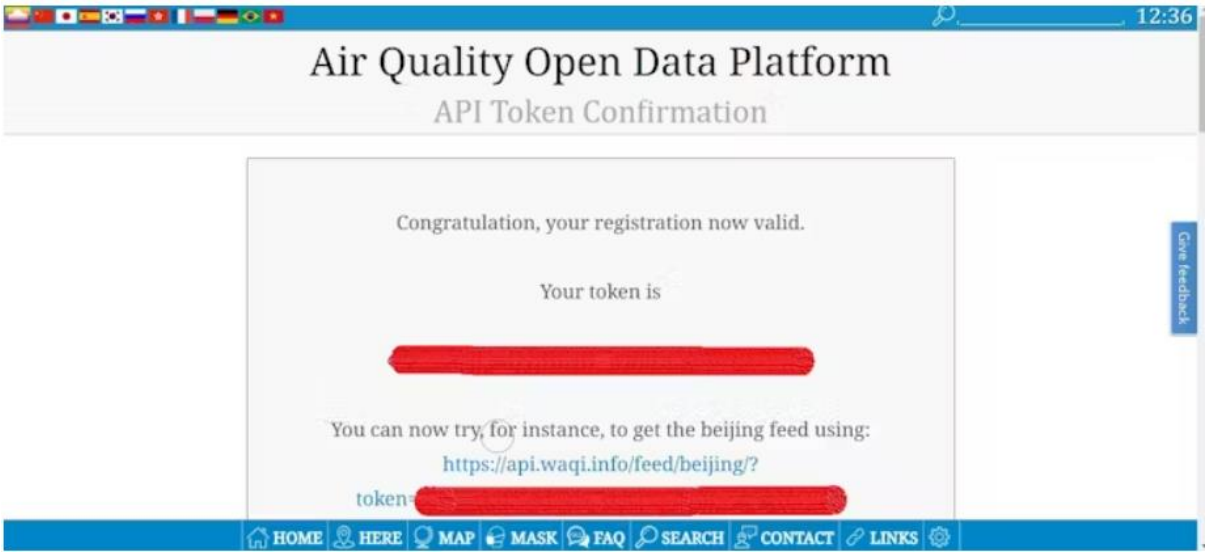
1. Go to below address and fill out all required credentials and click on submit

https://aqicn.org/data-platform/token/#/

NOTE - The API permits us to use 1000 API calls per second,

However, make sure you read all the terms and conditions and provide proper attributions before usage.

2. After submitting you will need to confirm your email address to get the API token, which looks like below image



PROGRAM CODE FOR THE SYSTEM

```
import json
import requests
import time
from boltiot import Bolt

API_KEY = "your_bolt_api_key"
DEVICE_ID = "your_bolt_device_id"
telegram_bot_id = "bot_your_telegram_bot_id"
telegram_chat_id = "@aqi_alert"
bolt = Bolt(API_KEY,DEVICE_ID)

def aqi_check():
        url = "https://api.waqi.info/feed/mysore/?token=your_generated_api_token"
        response = requests.get(url)
        data = response.json()
        current_aqi = data['data']['aqi']
        city_name = data['data']['city']['name']
        status = data['status']
        return current_aqi,city_name,status;

def send_telegram_alert(message):
        url = "https://api.telegram.org/" + telegram_bot_id + "/sendMessage"
        data = {
                "chat_id": telegram_chat_id,
                "text": message
        }
        try:
                response = requests.request(
                        "POST",
                        url,
                        params = data
                )
```

```
                print("This is the telegram url")
                print(url)
                print("This is the telegram response")
                print(response.text)
                telegram_data = json.loads(response.text)
                return telegram_data["ok"]
        except Exception as e:
                print("An error occurred in sending the alert message via Telegram")
                print(e)
                return False
def alarm():
        bolt.digitalWrite("0","HIGH")
        time.sleep(10)
        bolt.digitalWrite("0","LOW")

while True:
        aqi,city,status = aqi_check()
        if status != "ok":
                print("Response was unscessful")
                time.sleep(10)
                continue
        if (aqi >= 0 and aqi <= 50):
                message = f"Current Air Quality Index is {aqi} in {city}\nAir Quality is considered
satisfactory"
                print(message)
                send_telegram_alert(message)
                alarm()
        elif (aqi >= 51 and aqi <= 100):
                message = f"Warning! Air Quality Index is {aqi} in {city}\nAir Quality is acceptable;
However, for some reason there may be a moderate health concern"
                print(message)
                send_telegram_alert(message)
                alarm()
        elif (aqi >= 101 and aqi <= 150):
                message = f"Warning! Air Quality Index is {aqi} in {city}\nMembers of sensitve groups
may experience health effects"
                print(message)
                send_telegram_alert(message)
                alarm()
        elif (aqi >= 151 and aqi <= 200):
                message = f"Warning! Air Quality Index is {aqi} in {city}\nEveryone may begin to
experience health effects."
                print(message)
                send_telegram_alert(message)
                alarm()
        elif (aqi >= 201 and aqi <= 300):
                message = f"Warning! Air Quality Index is {aqi} in {city}\nHealth alert! Everyone may
experience more serious health effects."
                print(message)
                send_telegram_alert(message)
                alarm()
        elif (aqi >= 301 and aqi <= 500):
                message = f"Warning! Air Quality Index is {aqi} in {city}\nEmergency! The entire
population is more likely to be affected."
                print(message)
                send_telegram_alert(message)
                alarm()
```

```
            time.sleep(900)
```

CONCLUSION

n IoT-based air pollution monitoring system is a revolutionary solution that can provide accurate and real-time data about the air quality in a particular area. It can help identify the sources of pollution and take necessary measures to reduce it, protecting the environment and human health.