

# TASK REPORT

Submitted by : Chirag Talwar

## Introduction:

This project is about using embedded systems and IoT networking protocol to fetch real-time metrics of the Air-Quality sensor deployed on the I3 marketplace.

## Task assigned:

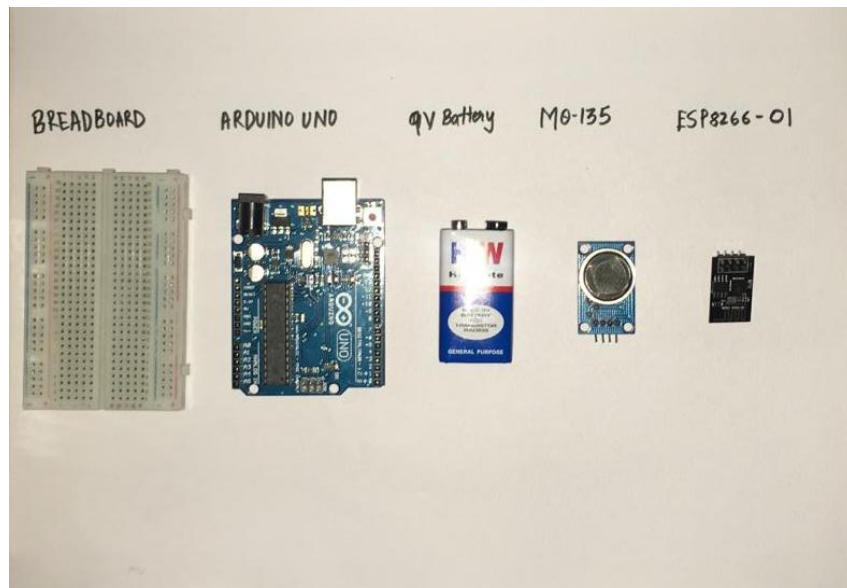
- Check out [eclipse.usc.edu:8000](https://eclipse.usc.edu:8000) for our I3 instance and read relevant documentation on how to post data products on it
- Deploy an arduino based sensor somewhere that generates useful / meaningful sensor readings
- Figure out how to run MQTT on that node, create/deploy corresponding arduino sketch
- Create a corresponding data product on I3
- publish data from that sensor to I3
- build a webapp or mobile app that subscribes to the corresponding flow after adding it on I3 and displays the readings in real time
- Write up a clear description of how you did each step above as a blog post somewhere and contribute relevant code and documentation

## Components:

- I used MQTT in this project assigned to me which is a simple messaging protocol, designed for constrained devices with low-bandwidth. So, it's the perfect solution for Internet of Things applications. MQTT allows to send commands to control outputs, read and publish data from sensor nodes.
- I used my Arduino device to connect to the MQTT broker and sent the real time data(Air Quality Sensor) on the MQTTcloud broker and a corresponding python script to fetch the data from the broker.

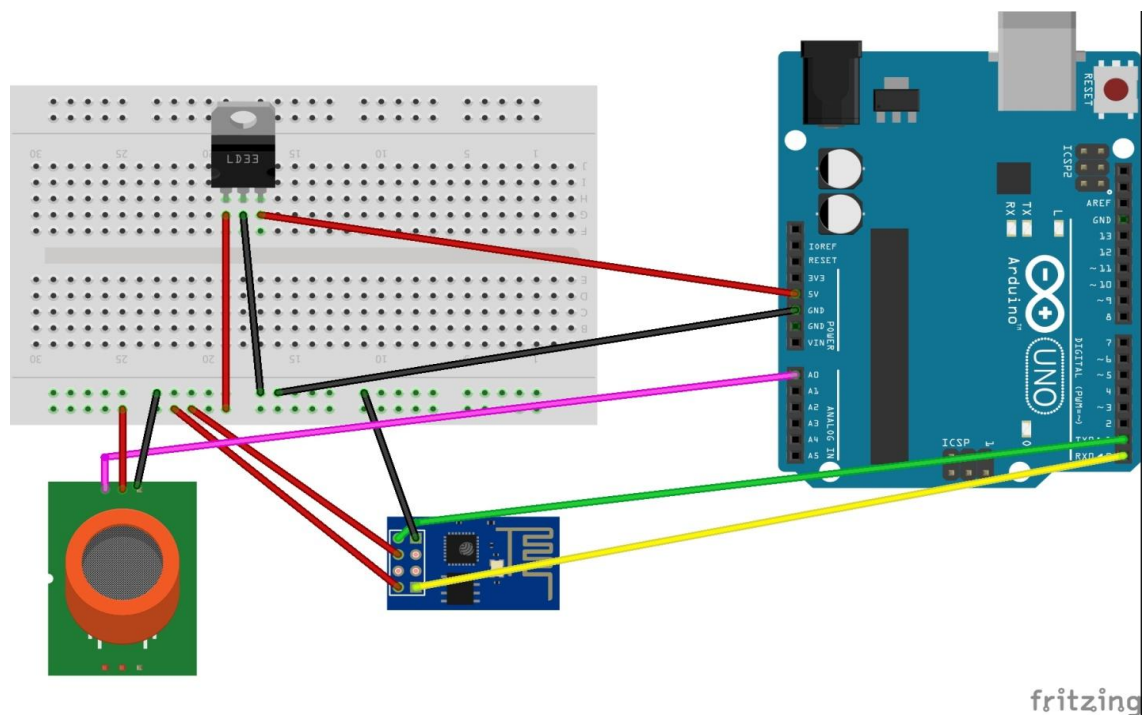
## Hardware components:

1. Arduino UNO rev3
2. Esp8266-01 Module(Wifi)
3. MQ-135 (Gas Sensor)
4. Jumper Wires
5. Breadboard
6. AMS-1117(5v-3.3v Regulator)

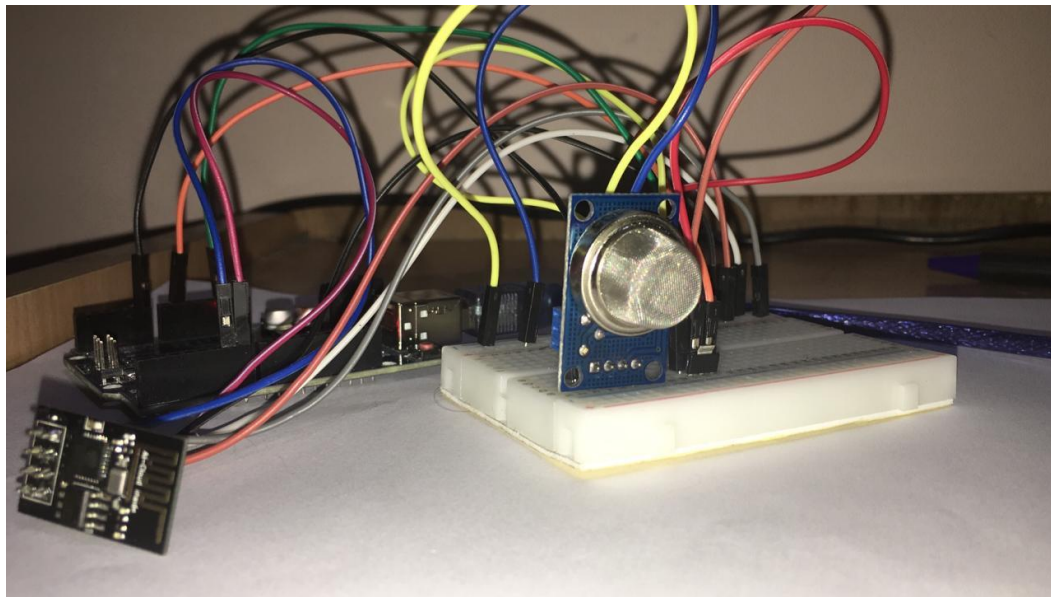
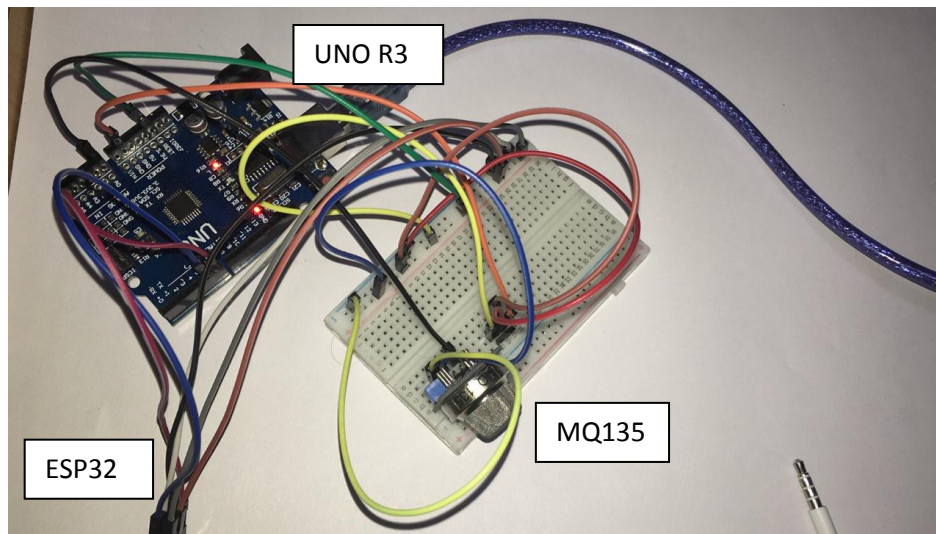


## Connections:

(Made using Fritzing)



Picture of the connections made:



### STEP 3. PUBLISHING FROM AIR QUALITY SENSOR

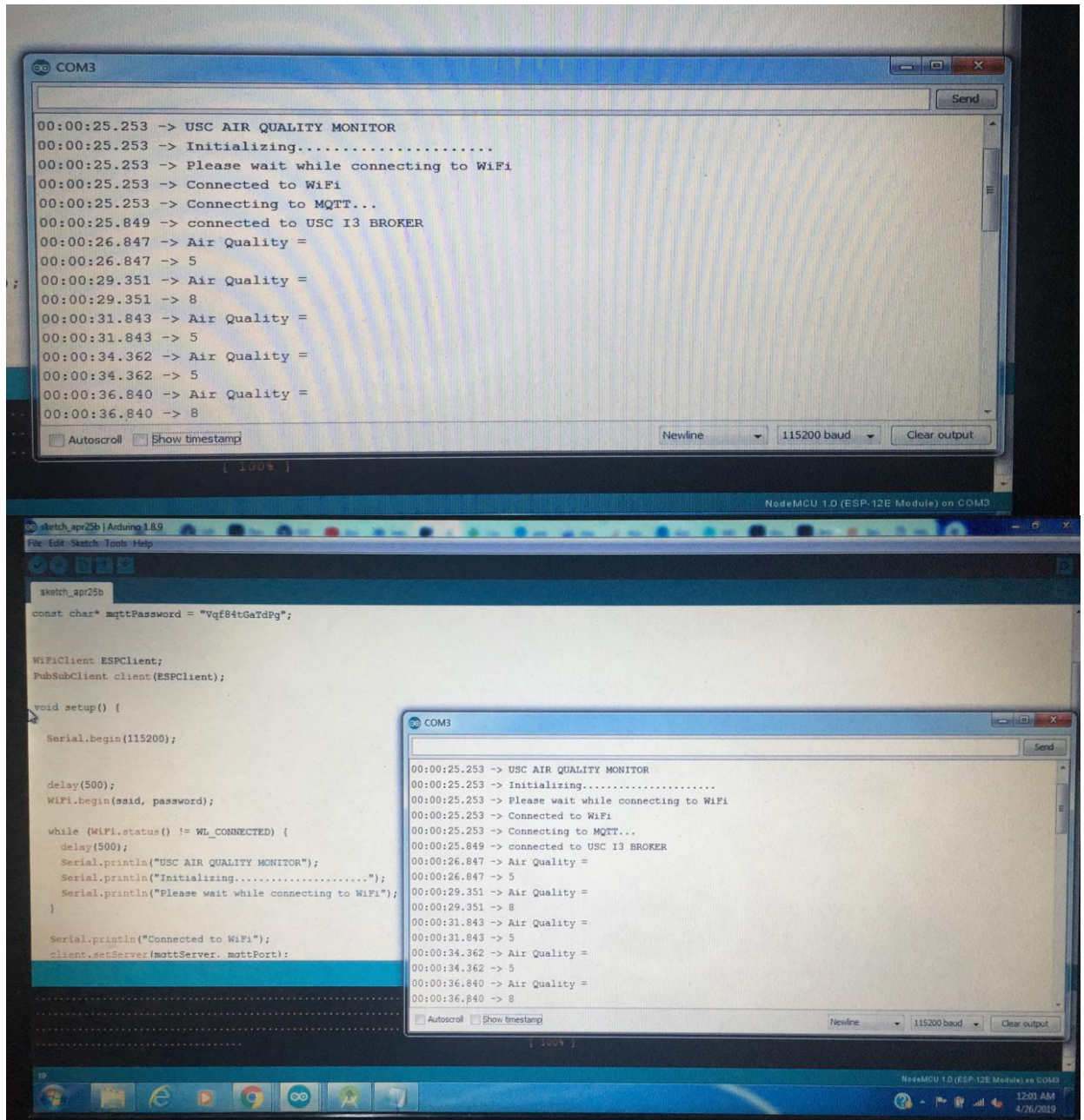
#### ARDUINO CODE

- Used Arduino 1.8.9 to run MQTT protocol over the sensor node(MQ-135) and the readings were displayed in the serial monitor.
- GitHub link for the code written:  
<https://gist.github.com/chiragtalwar9090/c92fa9faebda99061a35405c0279561a>



## Results

Serial Monitor:



The screenshot displays the Arduino IDE interface. The top window shows the sketch 'sketch\_apr25b' with the following code:

```
const char* mqttPassword = "Vqf84tGaTdPg";

WiFiClient ESPClient;
PubSubClient client(ESPClient);

void setup() {
  Serial.begin(115200);

  delay(500);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("USC AIR QUALITY MONITOR");
    Serial.println("Initializing.....");
    Serial.println("Please wait while connecting to WiFi");
  }

  Serial.println("Connected to WiFi");
  client.setServer(mqttServer, mqttPort);
}
```

The bottom window shows the serial monitor output for COM3 at 115200 baud. The output is as follows:

```
00:00:25.253 -> USC AIR QUALITY MONITOR
00:00:25.253 -> Initializing.....
00:00:25.253 -> Please wait while connecting to WiFi
00:00:25.253 -> Connected to WiFi
00:00:25.253 -> Connecting to MQTT...
00:00:25.849 -> connected to USC I3 BROKER
00:00:26.847 -> Air Quality =
00:00:26.847 -> 5
00:00:29.351 -> Air Quality =
00:00:29.351 -> 8
00:00:31.843 -> Air Quality =
00:00:31.843 -> 5
00:00:34.362 -> Air Quality =
00:00:34.362 -> 5
00:00:36.840 -> Air Quality =
00:00:36.840 -> 8
```

The serial monitor also shows the status of the connection and the air quality readings. The output is displayed at 100% zoom.

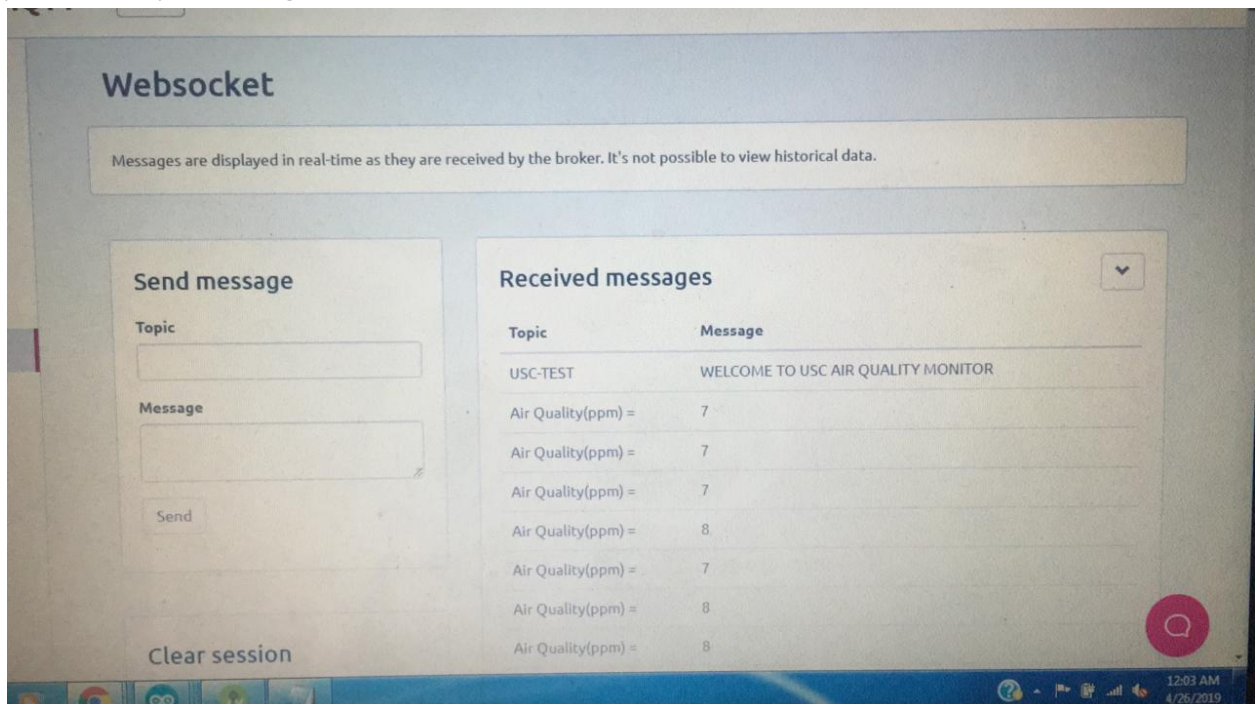
Due to the I3 broker getting disconnected, I also tried to check whether the data is being sent, to a test cloudmqtt server with the following credentials

Server : m16.cloudmqtt.com

Port : 11232

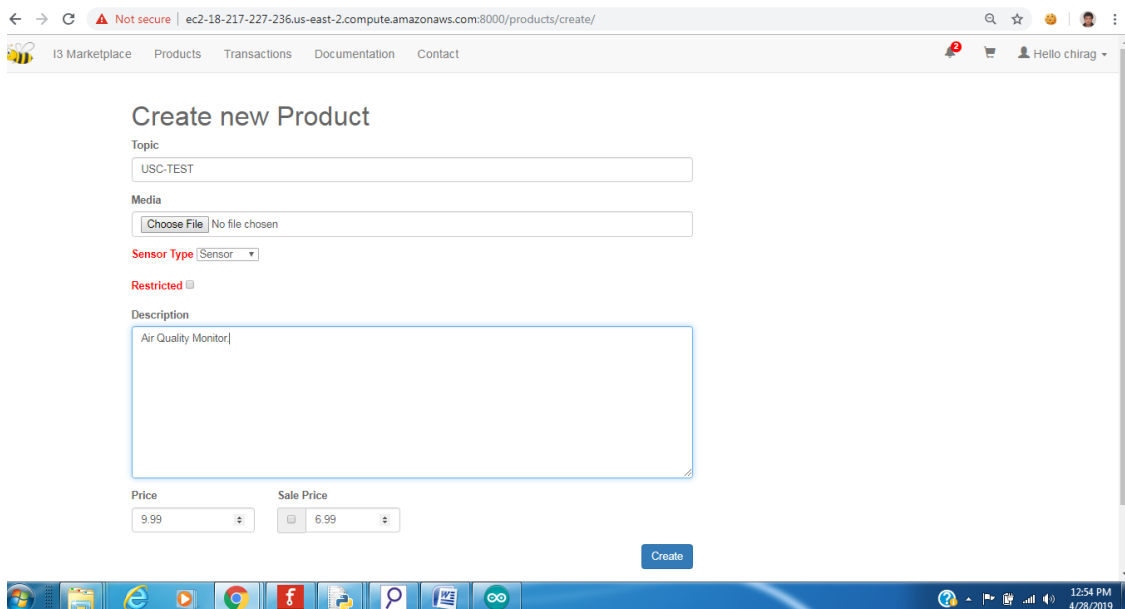
Username : rerxmvs

password : Vqf84tGaTdPg



#### Step 4: Created a product on I3 broker :

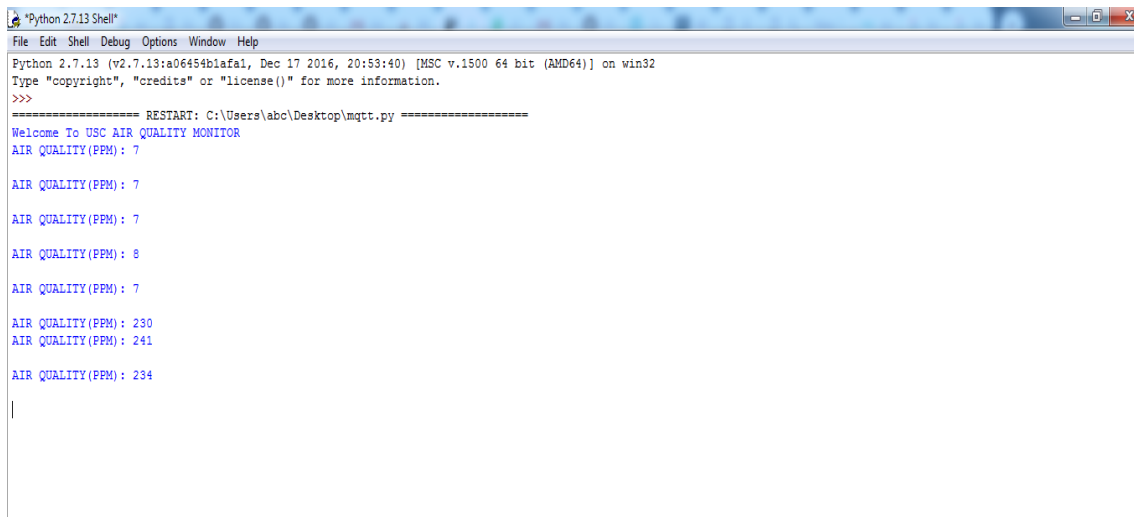
- Used the I-3 broker to create a new product which the buyer can have access to.



## BUYER SIDE : SUBSCRIBING

- Used the python-paho client to create a subscriber node where the buyer can have access to the data.
- Github link for the code written in python 2.7.  
<https://gist.github.com/chiragtalwar9090/c92fa9faebda99061a35405c0279561a>

## Results



```
Python 2.7.13 Shell
File Edit Shell Debug Options Window Help
Python 2.7.13 (v2.7.13:a06454b1afal, Dec 17 2016, 20:53:40) [MSC v.1500 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\abc\Desktop\mgtt.py =====
Welcome To USC AIR QUALITY MONITOR
AIR QUALITY (PPM): 7

AIR QUALITY (PPM): 7

AIR QUALITY (PPM): 7

AIR QUALITY (PPM): 8

AIR QUALITY (PPM): 7

AIR QUALITY (PPM): 230
AIR QUALITY (PPM): 241

AIR QUALITY (PPM): 234

|
```

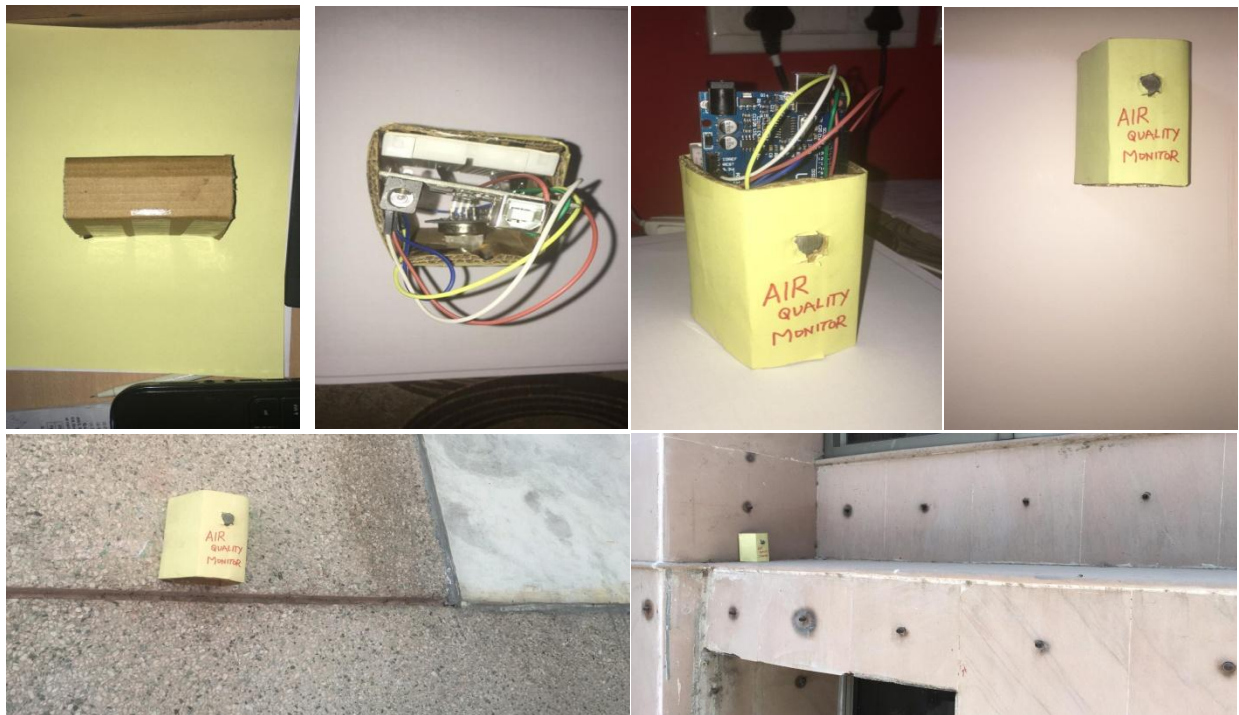
## Hardware deployment:

- I deployed our air quality sensor at various places in and around my locality and college.
- Created a small prototype using a hard card-board and the circuitry was fitted into that.





### Prototype:



### ANDROID APP( IN PROCESS) :

I am trying to create an android app using android studio 3.4 where the real-time data can be displayed.