detection using iot



Air Quality

IOT\_Phase2

project objectives

Millions of people live in areas where air pollution can cause serious health problems. Local air quality can affect our daily lives. Like the weather, it can change from day to day. EPA developed the Air Quality Index, or AQI, to make information available about the health effects of the five most common air pollutants, and how to avoid those effects.

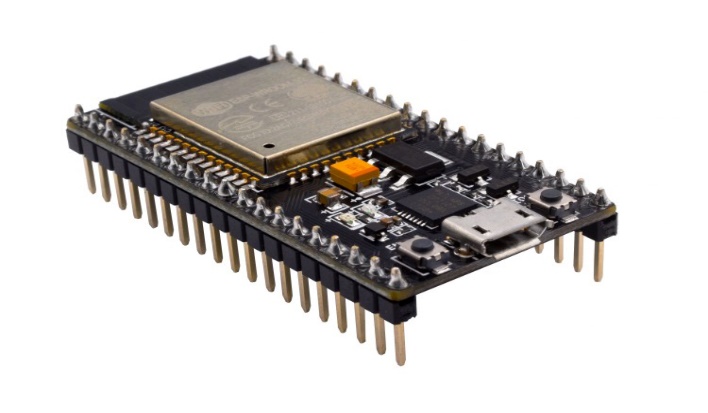
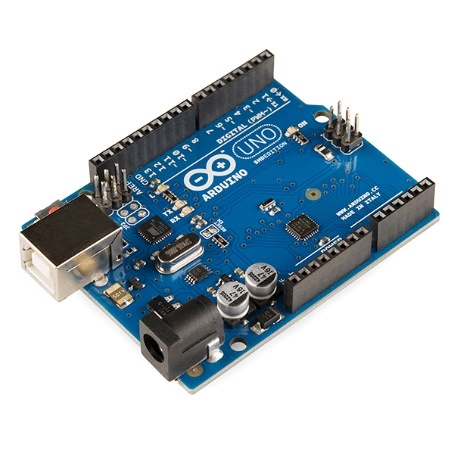
Air Quality Index depicts the extent of air pollution at a particular location. Air Quality Index (AQI) tells how clean or polluted outdoor air is, along with associated health effects that may be of concern.

The problem statement stated in Phase 1 is understood and we have arrived at an innovative plan/idea.

REquirements

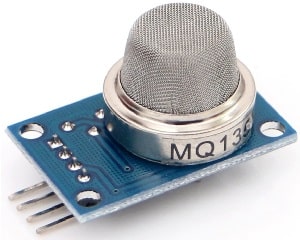
microcontrollers

We will be using the ESP32 microcontroller or Arduino UNO microcontroller as both these suit the best for our project.



Arduino UNO

ESP32

Sensors

MQ135 gas sensor

The MQ-135 Gas sensor can detect gases like Ammonia (NH3), sulfur (S), Benzene (C6H6), CO2, and other harmful gases and smoke. Similar to other MQ series gas sensor, this sensor also has a digital and analog output pin.

connectivity

* + WIFI
  + ZIGBEE

LEDs and BUZZERS

These can be used to indicate the air quality level on the device itself and give out warnings if necessary.

Cloud

* + Beeceptor - Beeceptor is a Rest API mocking tool. You can get started by creating an endpoint and mock specific API paths

predictive Analytics

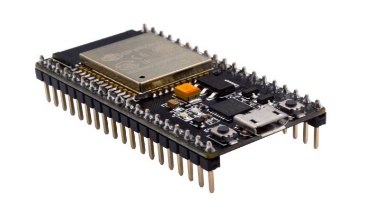
* The data can be stored onto a public server.
* This stored data can then be used to perform predictive analytics to predict about future air quality conditions.
* This can be done by simple regression models such as linear regression.
* This historical stored data can be used on the regression model to show an approximate prediction of the air quality for a few future days.

Working



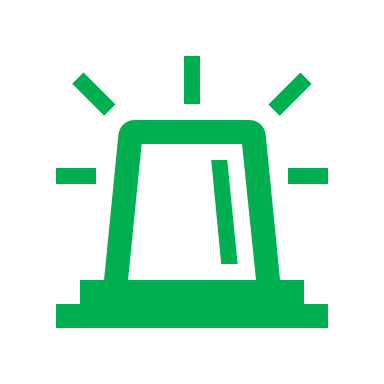
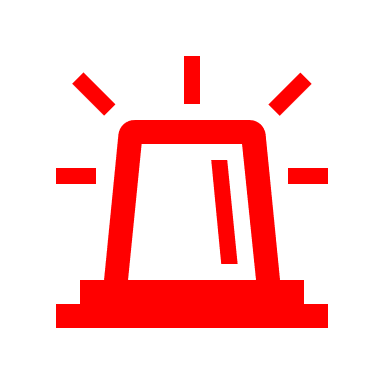
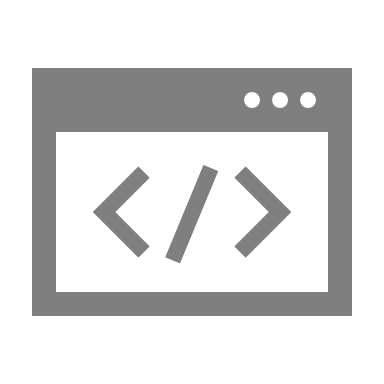
**Web server displaying real time data**

**Sensor Output**



**Wi-Fi**

**Analog/Digital to PPM**



**Storage of recorded data**

**Red if AQI > 800**

**Green if AQI < 800**



**Buzzer if AQI > 2000**

**Suitable Predictive Analytics methods (Regression, Naïve Bayes etc.)**



**Predictions**

* The MQ-135 sensor gives both analog and digital values on its output pins.
* This output can be then read by the ESP32 microcontroller.
* ESP32 converts the digital and analog output to the air contaminants’ PPM (Parts Per Million) values.
* The Microcontroller board’s code also makes a green LED glow if the AQl is less than 800 or a red LED if it’s above 800. It can also make the buzzer sound if the AQI exceeds 2000ppm.
* This PPM values can be displayed on a public web page.
* In the backend of the webpage, we can also store the values for predicting the future air conditions.
* This stored data can be fed into a suitable AI or ML model to predict the air quality for upcoming days.