ESW Project Report

Team 5
Aaron Anthony Monis | 2020101129
Gaurav Singh | 2020111014
Pratyay Suvarnapathaki | 2020111016
Yash Mehan | 2020111020

SECTION ONE
Motivation

According to the WHO, household air pollution was responsible for an estimated 3.2 million deaths per year in 2020, including over 237,000 deaths of children under the age of 5. A Harvard study that monitored PM2.5 and CO2 levels found that indoor air quality significantly affects employees' cognitive function, ability to focus, and overall productivity.

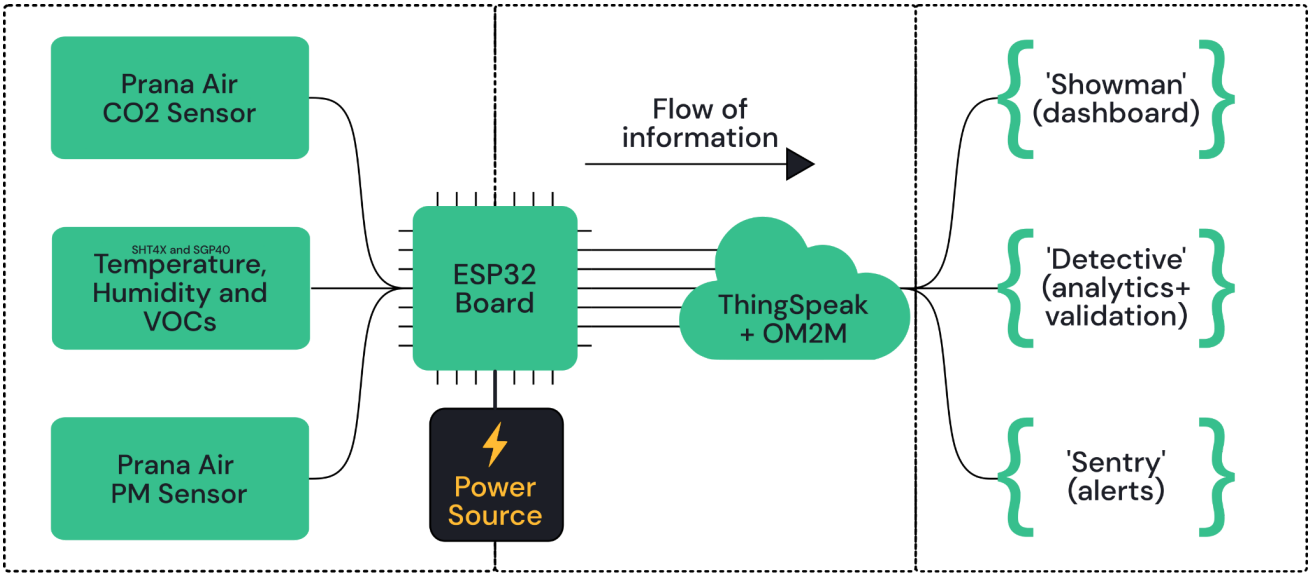
Therefore, tracking indoor air quality is clearly a pursuit worth undertaking. Here are some key parameters can be tracked as indicators of overall air quality:

→ CO2
Acts as a proxy for room ventilation (as in the Harvard study).

→ Particulate matter
As an indicator of 'breathability'. Particulate matter impacts the respiratory tract.

→ VOCs
Present in many household chemicals (aerosols, disinfectants, etc.). Harmful to the ENT tract.

SECTION TWO
System Structure and Division of Work



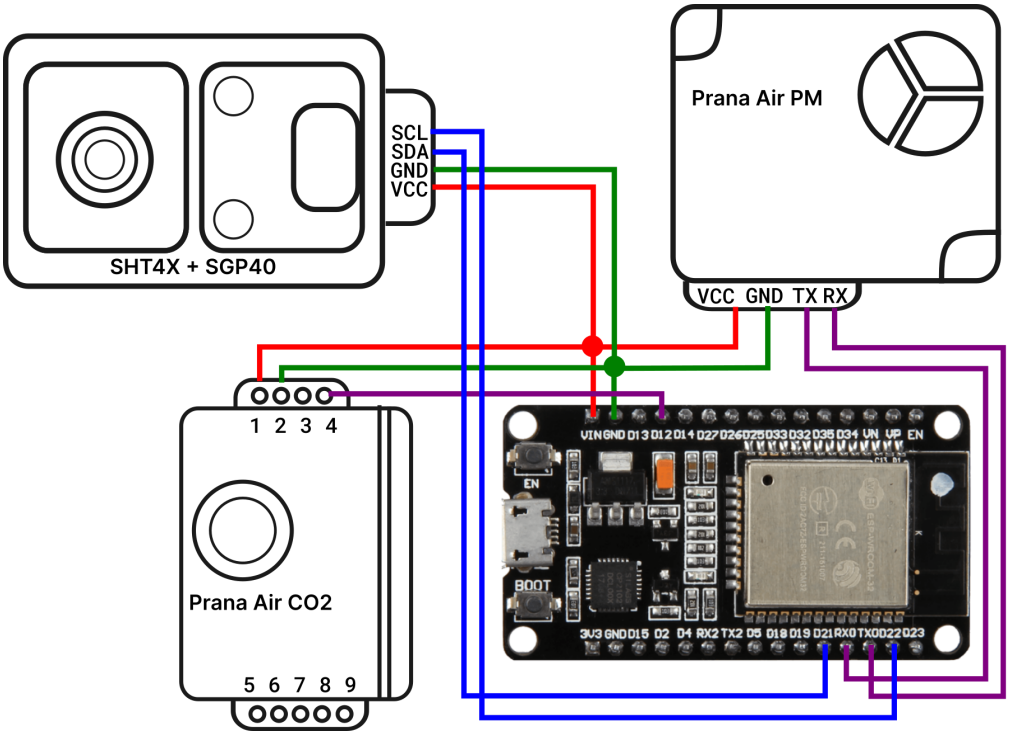
Division of Work

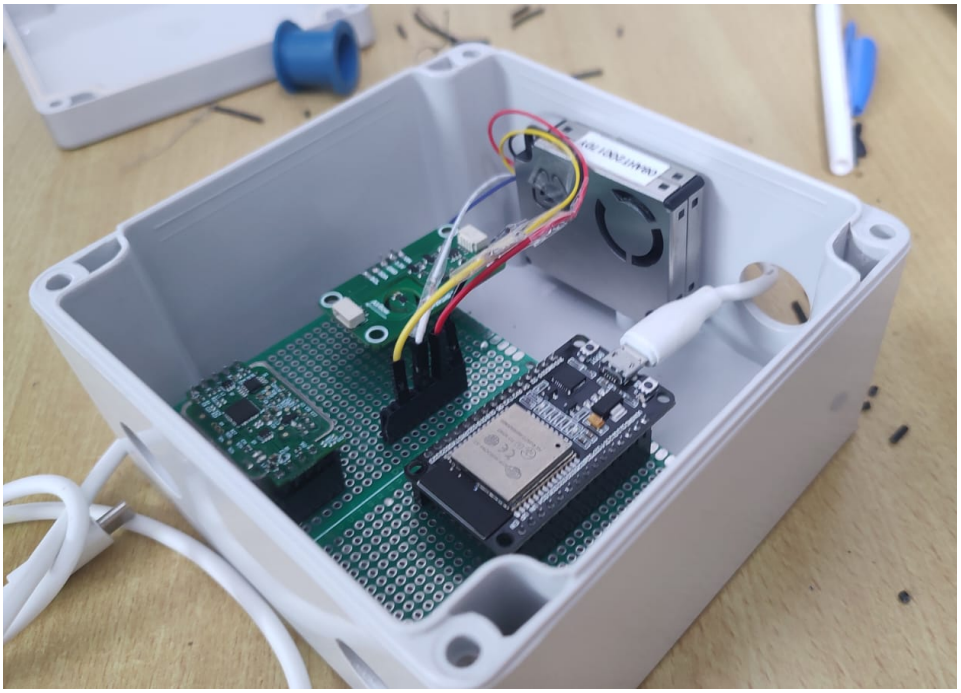
- Yash Mehan: Hardware configuration (soldering, circuitry, enclosure) - Module 1
- Aaron Monis: On-board code (sensing + interfacing) - Module 2
- Pratyay Suvarnapathaki: ‘Showman’ Dashboard, overall system design - Module 3
- Gaurav Singh: ‘Detective’ and ‘Sentry’ data validation and alerts suite - Module 3

SECTION THREE

Our Hardware

Circuit Diagram





Component Specifications

- SGP40 VOC Sensor + SHT4X Temperature and Humidity Sensor
    - Both communicate over I2C.
    - VOC Index (digital processed value) | Range 0-500, unit = VOC index.
    - Raw value: range: 0-65536, units: 'ticks'.
    - Temperature resolution: 0.1 degrees celsius | Range: -40 ~ 125 deg celsius.
    - Humidity resolution: 1% Relative Humidity | Range: 0 ~100 %RH.

- Prana Air CO2 Sensor
    - Gives a PWM output corresponding to the concentration of CO2 in the atmosphere.
    - Output range: 0~2000, units: ppm
    - PWM width:
        - 2ms HIGH, then a (reading/2) ms HIGH, followed by a 2ms LOW
        - Thus, maximum cycle duration = 1004 ms
        - Uncertainty +/- 5%

- Prana Air Indoor PM sensor
    - Utilised UART with baud rate 9600
    - One frame: 9 Bytes, Parity bit to check integrity of data, 4 bytes of data
    - Range
        - For PM10: 0-1500 ug/m3
        - For PM2.5: 0-999 ug/m3
        - Resolution: 1ug/m3
        - Particle size resolution: 0.3um

- Espressif Systems ESP32 microcontroller

SECTION FOUR

Our Secure Middle Men - ThingSpeak and OM2M

The ESP32 first connects to the internet via a secure client that enables communications over secure channels. We then connect to the thingspeak server on port 443 using the Thingspeak Secure API. The Thingspeak secure API uses SSL/TLS. SSL/TLS generates a secure channel between two systems over which sensitive data can be sent. We also use Write API Keys generated at the Thingspeak Dashboard to be able to write data to the dashboard. We send each field at the same time to get matching graphs for the purpose of analysis.

If we analyse the network channel using a tool like WireShark, we would find that our data is encrypted which protects our data from outsiders. Additionally, the SSL/TLS protocol uses Hash functions to verify that data is being sent correctly. This ensures that we have confidentiality, authentication and integrity which are the foremost security issues we need to have to ensure security.

We also send data to oneM2M to foster interoperability for future usage and implementations. We do so by using the *HTTPClient* Library available for ESP32. We send POST requests by putting our data into a proper format and specifying the required ID and password in the headers. We have

already created the Application Entity and the required containers in the format specified in the labs using python and its *requests* library. We can also verify the function of oneM2M by using a python script to get the last 20 data points and plotting it. We use HTTPS in order to send the data which also uses SSL/TLS in order to send data via the secure channel.

SECTION FIVE

‘Showman’ - Our Dashboard

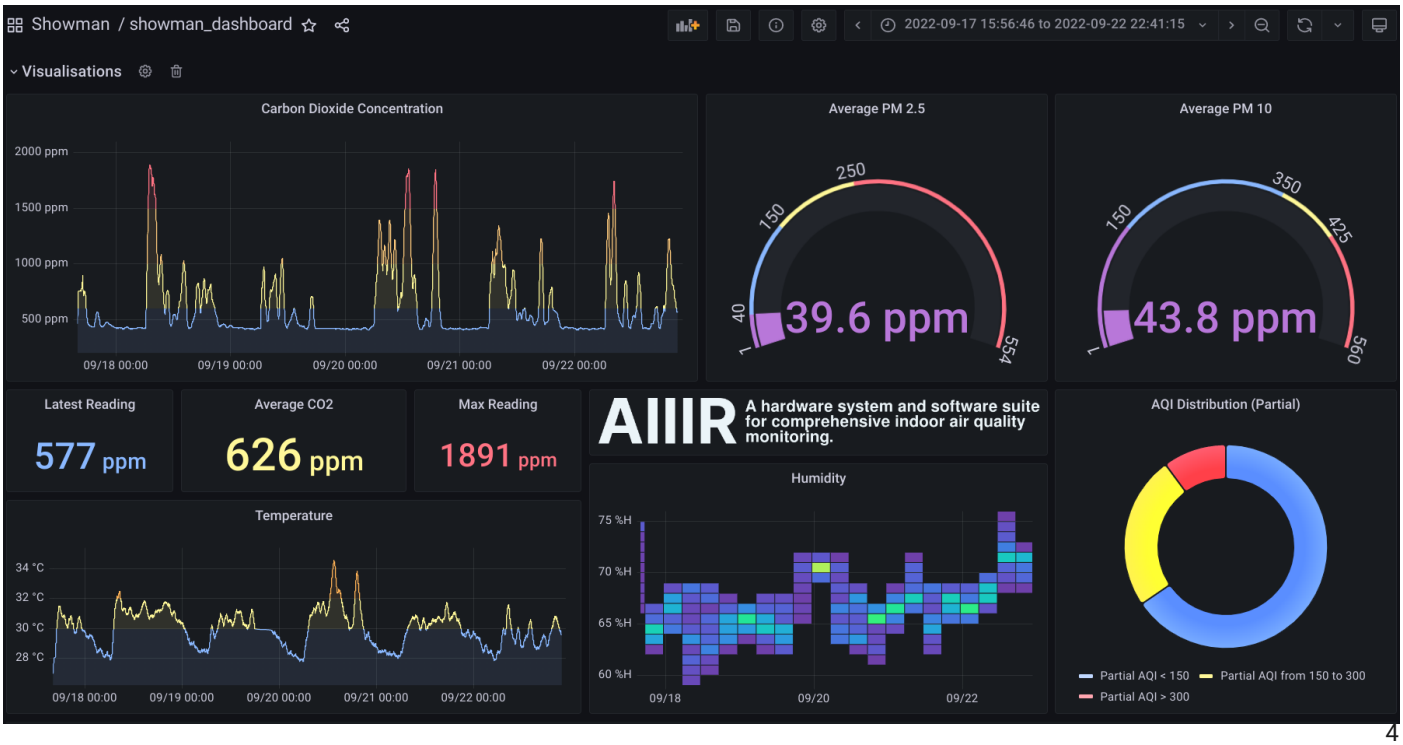
Our dashboard uses a graphing framework called Grafana.
To run grafana locally using docker,

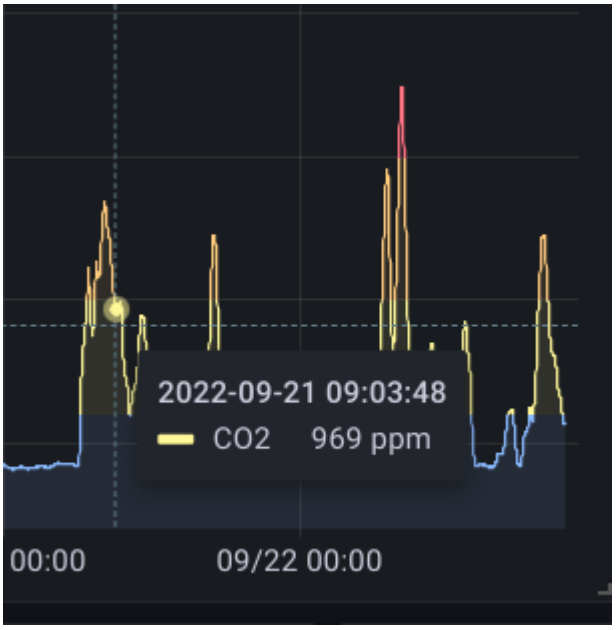
- Refer to https://grafana.com/docs/grafana/latest/setup-grafana/installation/docker/
- Install `grafana-oss` as per these instructions.
- Run the container and navigate to `localhost:3000` to access the dashboard.
- Replicate the panel setup as per `grafanaconfig.json` from our codebase, linked below.

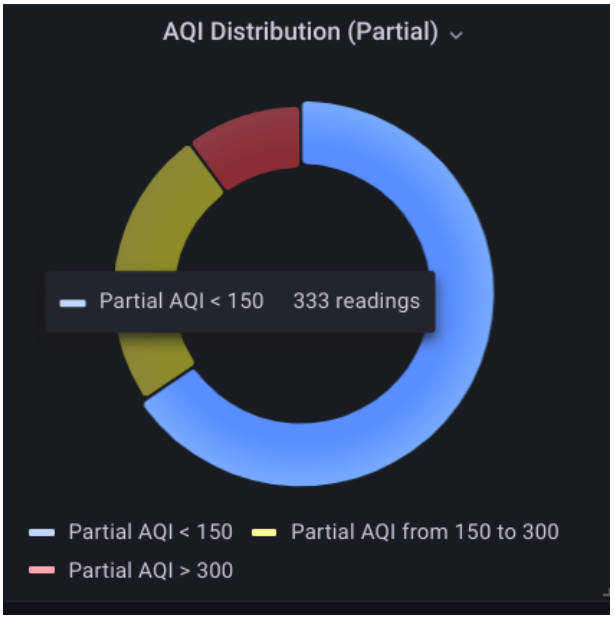
An instance has also been deployed on Grafana Cloud, but the visibility has been limited for security reasons + ThingSpeak API limitations (Every time someone views this hosted instance, 10 API calls are made - a lot of concurrent users would result in problems with ThingSpeak). Nevertheless, the url is https://showman.grafana.net/ (Sign-in required). For the sake of convenience, a static but interactive snapshot of the dashboard can be found at https://snapshots.raintank.io/dashboard/snapshot/zh3KrkzrOA8QhKGW1goNr8YsJCbzycuB

Pictured below is our primary visualisation interface. A lot of attention has been paid to minor details such as hover-interactions, and our drill-down zoom-in feature. The kinds of visualisations presented are as follows:

- Time series visualisations for CO2 and Temperature readings, with additional metrics included for CO2 to offer clearer insights into the data.
- A heatmap view for the humidity - essentially a temporal histogram, since viewers usually don’t care about the exact values of humidity, as the ‘general range’ matters more.
- Dial gauge views for PM values, along with a doughnut chart for visualising ‘partial AQI’ ranges (‘partial’ because additional parameters such as ozone and CO are necessary for applying the actual AQI formula).







Aside from this, there is an option to view all parameters in the time-series visualisation format.

SECTION SIX

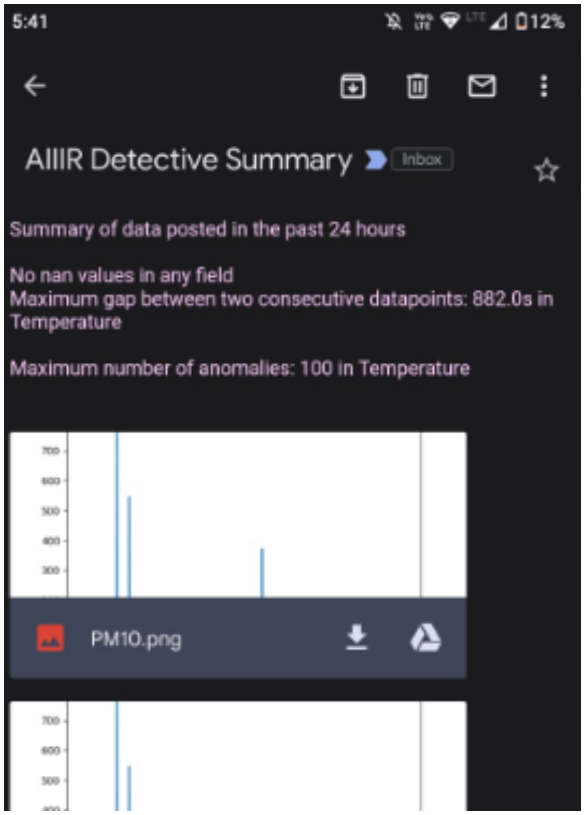
‘Detective’ and ‘Sentry’ - Our Validation and Alerts System

We have implemented two different systems for data validation, analysis and an alerts system: ‘Detective’ and ‘Sentry’

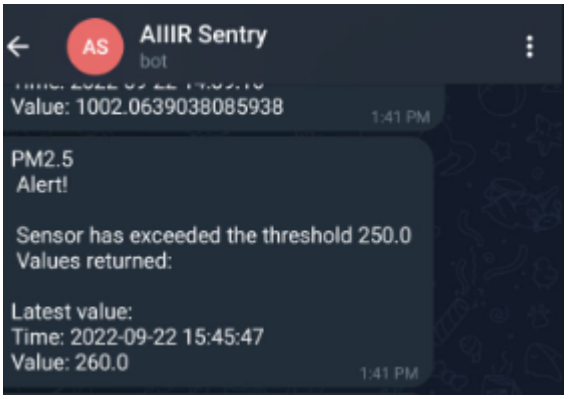
Detective refers to our data validation and analysis system. It is extremely scalable and extensible, and can be readily hosted on a server. It performs the following functions:

- It fetches the data from the single source of truth, in our case the secure thingspeak server
- **NaN Detection**: It performs preliminary validation and analysis by detecting NaN values from the sensors and purging the data points with NaN values.
- **Analysis of Posting Frequency:** One of the major problems in an IoT setting is the fluctuation of the frequency at which the nodes post data to the resource server, due to various problems. Thus we perform the essential analysis of this frequency so that such issues can be monitored, detected early and clarified.
- **Outlier Detection:** Detective also performs outlier detection using a Quantile Regression based anomaly detection approach, with 99% quantile value.
- **Forecasting**: We employ a polynomial regression based approach to forecast the values of a given sensor for the next day of a week while training on the data of all the sensors for about 5 days before it.
- **Weekly summary sending via Email:** Our system gives daily summary via mail to the registered users.

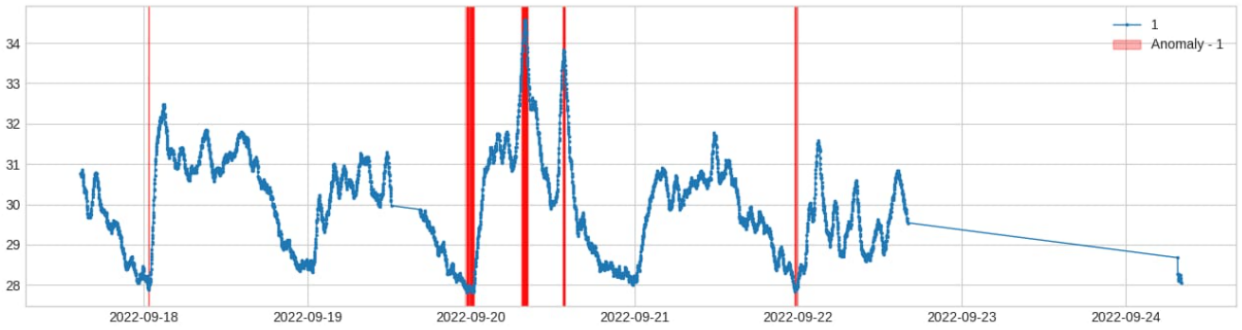
Sentry allows our system to alert out of bounds or emergency sensor values via a telegram bot to the registered users, for immediate action.



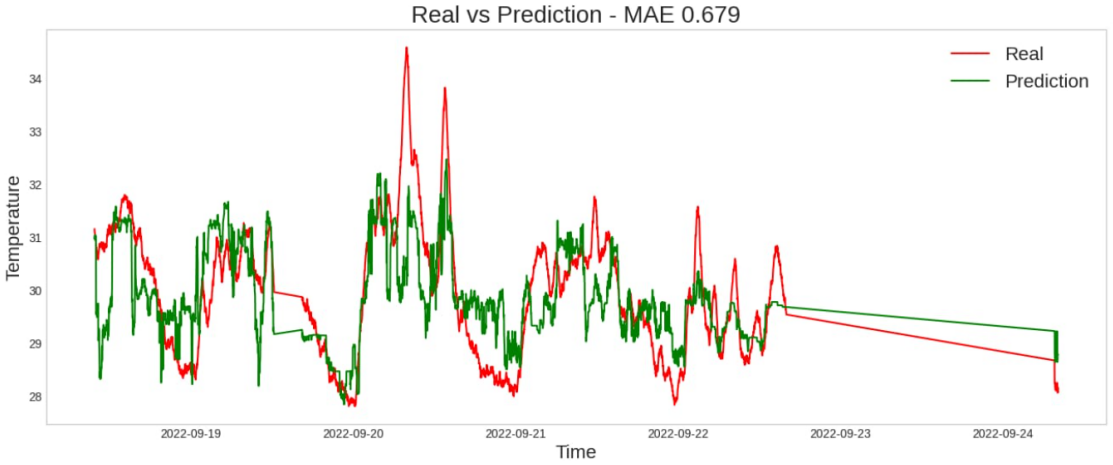
Detective - Email Summary Sending



Sentry for urgent alerts



Outlier detection



Forecasting (trained on 17th and 18th September data, tested on the latter half of the deployment)

SECTION SEVEN

Facing The Real World - Our Deployment Campaign

- Components were soldered to a zeroPCB. All soldered connections were checked for shorting with a multimeter.
- 2 iterations of the same PCB made to fit into the enclosure given after the first iteration.
- WiFi access was made available to the system via an Airtel My WiFi.
- Power supply via a 5V DC USB adapter.
- Deployed in IIITH's Kadamb Mess since the 17th of September

Of course, we faced a slew of challenges and mishaps in the course of the deployment, as is inherent to any first-time end-to-end IoT effort.

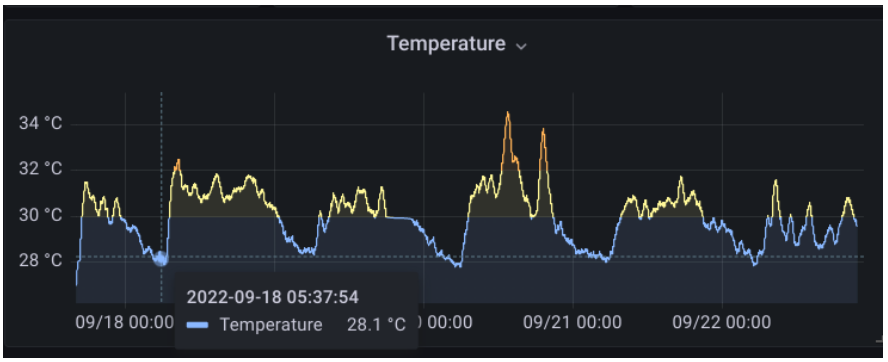
- Despite working satisfactorily in the initial testing stages of the project, the VOC sensor started producing an output of only 0 readings, or very occasional anomalous readings of ~20,000, upon deployment. The possibility of loose connections is ruled out because temperatures and humidity readings, which are from the same sensors, are observed to vary naturally with time. Due to the procurement of another sensor being infeasible, we were permitted to proceed without VOC readings by the teaching assistant, in consultation with the professor.
- Initially the router used was a JioFi, had large downtimes because it did not have good network coverage. It was later replaced with an Airtel MyWiFi.

SECTION EIGHT

Data Analysis and Inferences

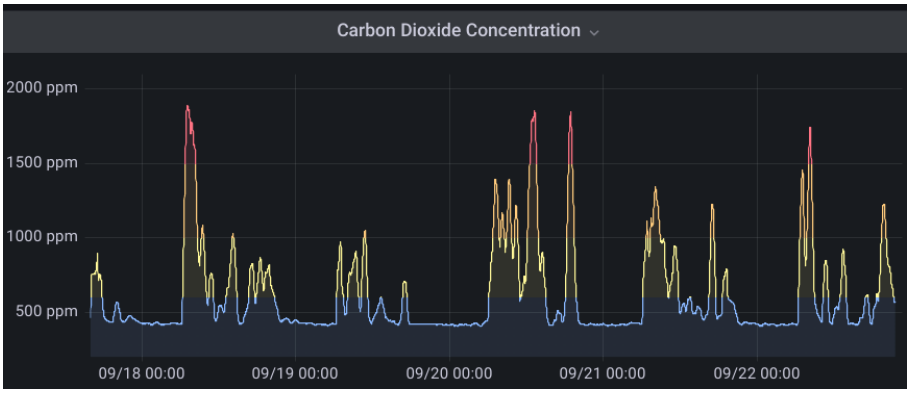
Despite the intricate insights that ML and statistics provide, it is often useful to dive in manually and validate data using patterns that us humans are better at recognising.

Temperature



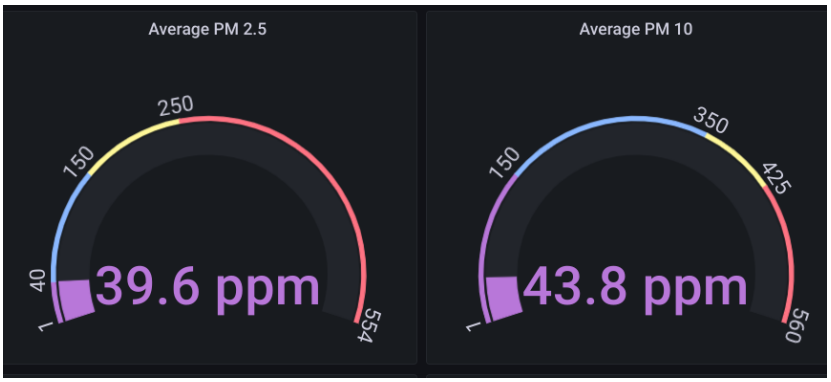
- A general day-to-night high-low pattern is observed, as expected.
- Temperatures are lowest at about 5am, before suddenly spiking up with sunrise.
- The highest temperature reading was obtained during Tuesday lunch, when french fries are on the menu.

CO2 Concentration



- Generally, spikes are observed during breakfast-prep time.
- Additionally, there is one other spike during the aforementioned Tuesday lunch.
- Values go much further beyond 1500PPM, which is considered ‘extremely hazardous’ as per health norms.
- The ‘stable’ / ‘idle’ values sit right around 400PPM, which largely makes sense. In fact there exist several cities in India whose **outdoor** AQI is much greater than this.

Particulate Matter



- PM2.5 and PM10 follow largely the same pattern, quite predictably so.
- There are spikes in the data that reach as high as 500PPM, which is considered extremely harmful. Either these unusually high readings are merely minor sensor misfires, or IIIT Hyderabad may have a pretty serious problem worth looking into.

SECTION NINE

Codebase

https://github.com/vanhalen42/AIIIR_Detective