TEAM ID	**PNT2022TMID35932**
TITLE	**Hazardous Area Monitoring for Industrial Plant powered by IoT**

Project Report

CONTENTS

7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    a. Feature 1
    b. Feature 2
    c. Database Schema (if Applicable)
8. **TESTING**
    a. Test Cases
    b. User Acceptance Testing
9. **RESULTS**
    a. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

1.INTRODUCTION

1.1 Project Overview

The Internet of Things (IoT) system proposed in this project is an advanced solution for monitoring the temperature, humidity and smoke at different points of location in industrial plants, making this temperature and humidity data visible over internet through cloud based dashboard and sending alerts to predefined recipients when temperature rises above the safe operating zone and reaches certain high values. This helps the industrial management team to take immediate action to rectify this temperature deviation. Also this can be monitored from anywhere anytime over online dashboard by the senior level professionals who are not present in the that zone at any point in time.

The proposed system consists of mobile, wearable IoT device interfaced with Temperature sensor, Humidity sensor and a Gas sensor which will collect the data in the ambiance of the wearer. This Data is sent over the internet using MQTT protocol to the IBM Watson Internet of Things Cloud Platform. The Data from the IBM Watson platform is processed and stored in the IBM Cloudant database. The Processed data is sent to a Web Dashboard. This Data is also shared with a Mobile Application using the HTTP requests. The Mobile app raises an alarm if the sensor data points to a Temperature above a Threshold and if presence of Gas is detected. The IoT end device is also interfaced with a Buzzer [Motor]. Whenever the Dashboard administrator or Mobile app user wishes to raise alarm, They can raise them by clicking the buttons. This causes the Buzzer to sound the alarm in IoT end device.

1.2 Purpose

1. In some industrial plants, there are some areas which are to be monitored time to time. Sometime the conditions may become critical which may lead to loss of property and also human loss.
2. To monitor the temperature we can integrate the smart devices in the areas which are needed to be monitored.
3. The persons who generally monitor these places will be given a wearable device which will be acting as a beacon scanner. Whenever the person enters the desired area then he can view the required parameters and can be alerted, these are sent to cloud.
4. Industrial accidents are as old as industry itself and so are preventive measures. The Standards for Explosive Areas or Atmospheres have also has evolved diversely worldwide, based on the local needs of the industries for the overall safe operation of the plants.

2.LITERATURE SURVEY

2.1 Existing Problems

1. The environment varies depending on the industry. It is difficult to sense the emission of gases and rise in temperature in the unreachable areas by the fixed sensors.
2. The Distance of transmission from the sensor is varying. Finding the proper communication technique is a tedious task.
3. In case if wearable sensors are used there will be high data traffic. Finding

   a proper database management is a major problem.

2.2 References

S.NO	TITLE	AUTHOR(s)	OBJECTIVE	LIMITATIONS/REMARKS
1.	A Hazardous Area Personal Monitoring System for Operators in Gas Depots and Storage Tanks (2022) Journal - AIDIC	Elia Landi, Lorenzo Parri, Ada Fort, Marco Mugnaini, Valerio Vignoli, Dinesh Tamang, Marco Tani	This work describes a smart monitoring system for the detection of flammable gas residues, toxic gases, and reduced oxygen concentrations. The proposed system aims at reducing the risk of fires and explosions, thus increasing the safety of workers engaged in maintenance or inspection of gas storages.	The system proposed in this paper is compact and battery powered. It contains sensors for monitoring LPG flammable compounds, toxic gases, and oxygen. The sensor nodes transmit data through a LoRa low power radio channel to a remote server. This system can be used as a model for our project with an additional Temperature sensor for monitoring the Temperature.
2.	IOT based Industrial Monitoring System Journal - irjmets	Hemlata Yadav, Naomi Oyiza, Sarfaraz Hassan, Dr. Suman Lata, K. Jaya Chitra	The goal of this study is to create an IoT-based industrial monitoring system with intelligent sensors. Because of the integration of big data, the Blynk app can be used to monitor status from anywhere on the planet.	This paper gives an overview of the available technologies for Industrial monitoring. It describes the use of Blynk app fordata interface with user.

3.	Monitoring of Hazardous Gases in Process Industries Through Internet (2016) Journal - irjet	Dr. K. R. Valluvan, P. Ragavi.	The existing detection systems are available to sense only a particular gas and they use GSM technology to indicate the critical situations. The drawback is that the detection system can send a message to only one person. The proposed system is made up of monitoring and alerting system through Internet of Things (IoTs). In this the dangerous, toxic and flammable gases are sensed using individual gas sensors and an Arduino UNO controller. The concentration of all gases values are displayed in ppm using a LCF; when the value exceeds the limited range then an alarm is put on.	This paper proposes a parallel sensor interface with an arduino UNO board and simple display using LCDs. The paper introduces the various available Gas sensors that can be used for monitoring the presence of different gases in the surrounding Environment. The individual gas has its own range of risk, they are identified using advanced sensors. This system gives an instantaneous alarm during the excessive emission of hazardous gases.
4.	IOT based Interactive Industrial Energy Management System and Emergency Alert Using SMS & E-Mail Journal - IJRTE	G. Krishnaprabu, Ramkumar, Jagadeesh, Senthilkumar, Ashok Kumar	The entire monitoring and control progress of the industrial utilities is an appropriate improvement in the industrial growth system. Here, the various industrial parameters are taken up for control such as gas, fire, machine, motor, in embedded based control module. In this module, the fire and gas sensor will analyze its set range variation by the controller. If it exceeds it pre-defined values set in the controller the immediate indication and alert is arrived for to take necessary safety precaution and control in real time application	This paper utilizes a PIC Microcontroller for interfacing a large number of sensors for monitoring the systems used in Industry. It iterates the need for control and monitoring mechanism and improves these measures by using SMS and E-Mail based alerts.

5.	Android Based Real-Time Industrial Emission Monitoring System Using IoT Technology (2017) Journal – Journal of Communication	Dennis A. Martillano, Joshua Miguel R. Dita, Christian G. Cruz, and Kunal S. **Sadhra**	This study aimed to create a system that will allow Industrial plants and factories to monitor the emission of the smoke stacks held in a manufacturing company anytime, anywhere using IoT or Internet of Things Technology.	This paper uses simple sensors to collect data and process the data using complex algorithms to enable self-calibration of the sensors. Configured XBEE modules that utilize IEEE 802.15.4 standard were set-up to commence communication in a point to point wireless network among the entities in the system

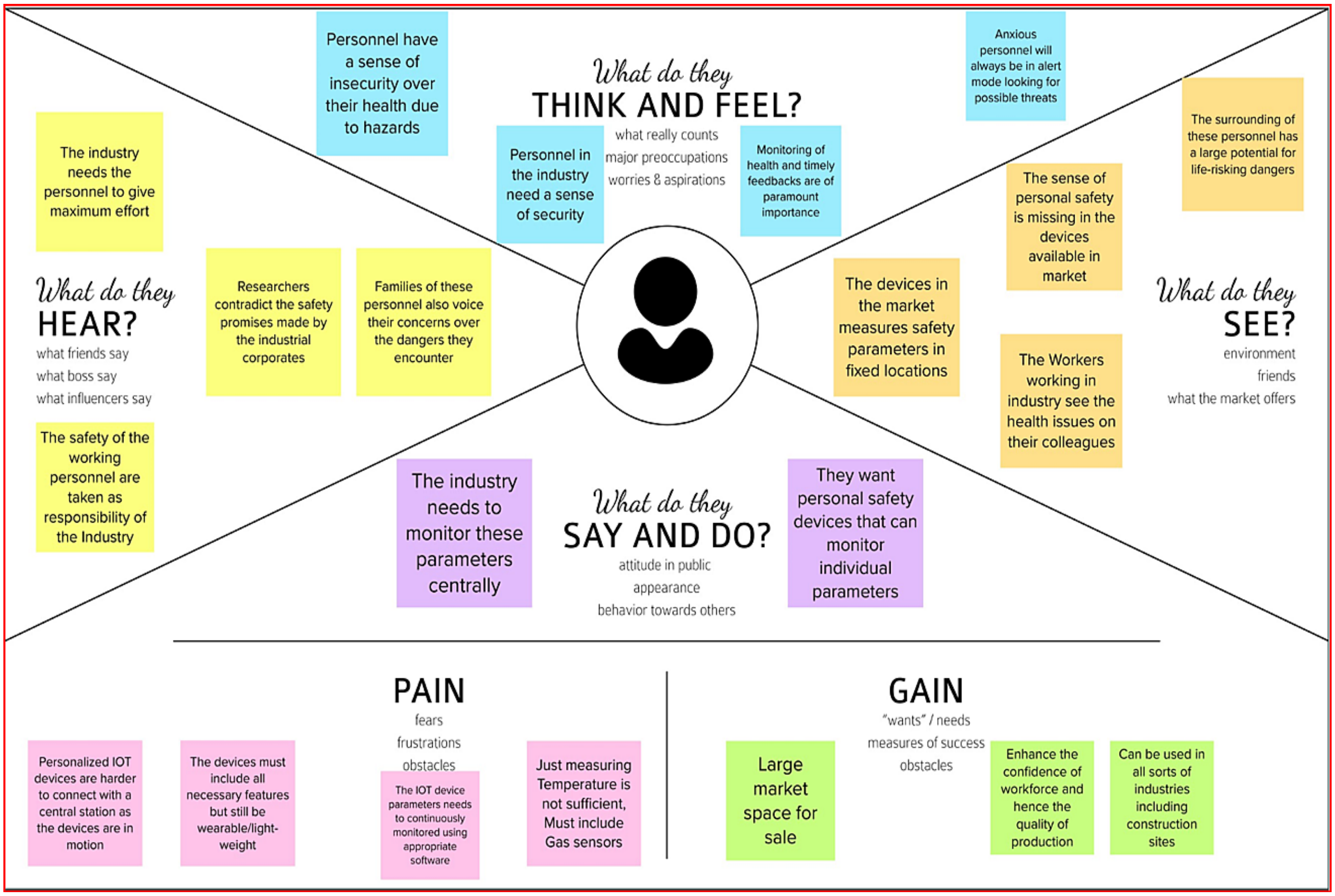
2.3 Problem Statement Definition

1. The main limitation of the current industry standard sensors is they are fixed in a single position. Thus the plant management does not have access to the working personnel's immediate surroundings. Hence the first problem we encounter is selecting sensors suitable for the working personnel to wear around the plant, so that the device doesn't cause a hindrance to the job carried by the personnel.
2. The main objective mentioned is to detect the temperature using a wearable sensor and the information be relayed back to a central command station. An immediate obstacle we encounter is the distance. We have to choose a communication device that can transmit for sufficiently long distances around the plant and at the same time it must be able to penetrate through the industrial structures like Boilers in the plant.

Problem Statement (PS)	**I am**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
The Device must be wearable.	Working Personnel	Complete the work assigned	There exists a high risk of poisoning and fire hazard	Of the Industry	Insecure, under-confident
The Distance of transmission from the sensor is varying.	Designer	Find the communication technique	The environment varies depending on the Industry	Of the nature deployment	Searching for an amicable solution.
Apart from Temperature there might exist poisonous gases.	Supervisor	Monitor the hazard in the area in my supervision	It depends not only on temperature but also on gases present	Risk of leakage	In need of a system to monitor the gases.
The Data traffic is high	Database manager	To find ways to ensure the data is collected	Traffic is high	Use of large number of sensors	To find proper database management.

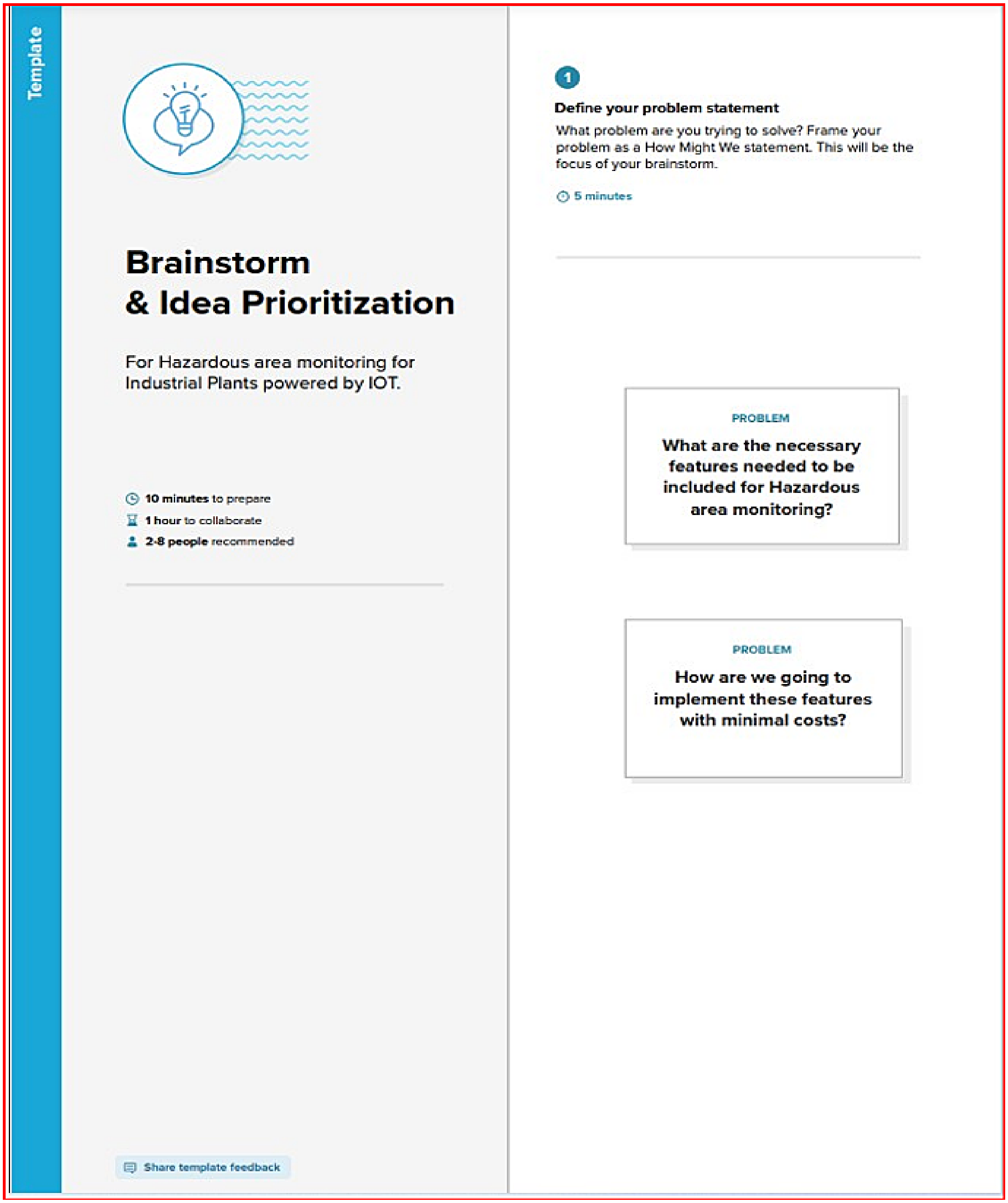
3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation and Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Bharani Kumar

- Indication of high temperature in hazardous areas
- Alerting the public in case of emergency
- The workers can be alerted by the device itself
- The temperature range of the device must be studied

Manibalan

- The control setup range must be fixed
- LORA communication seems to be the best option
- The device can be fixed with ID cards of employees

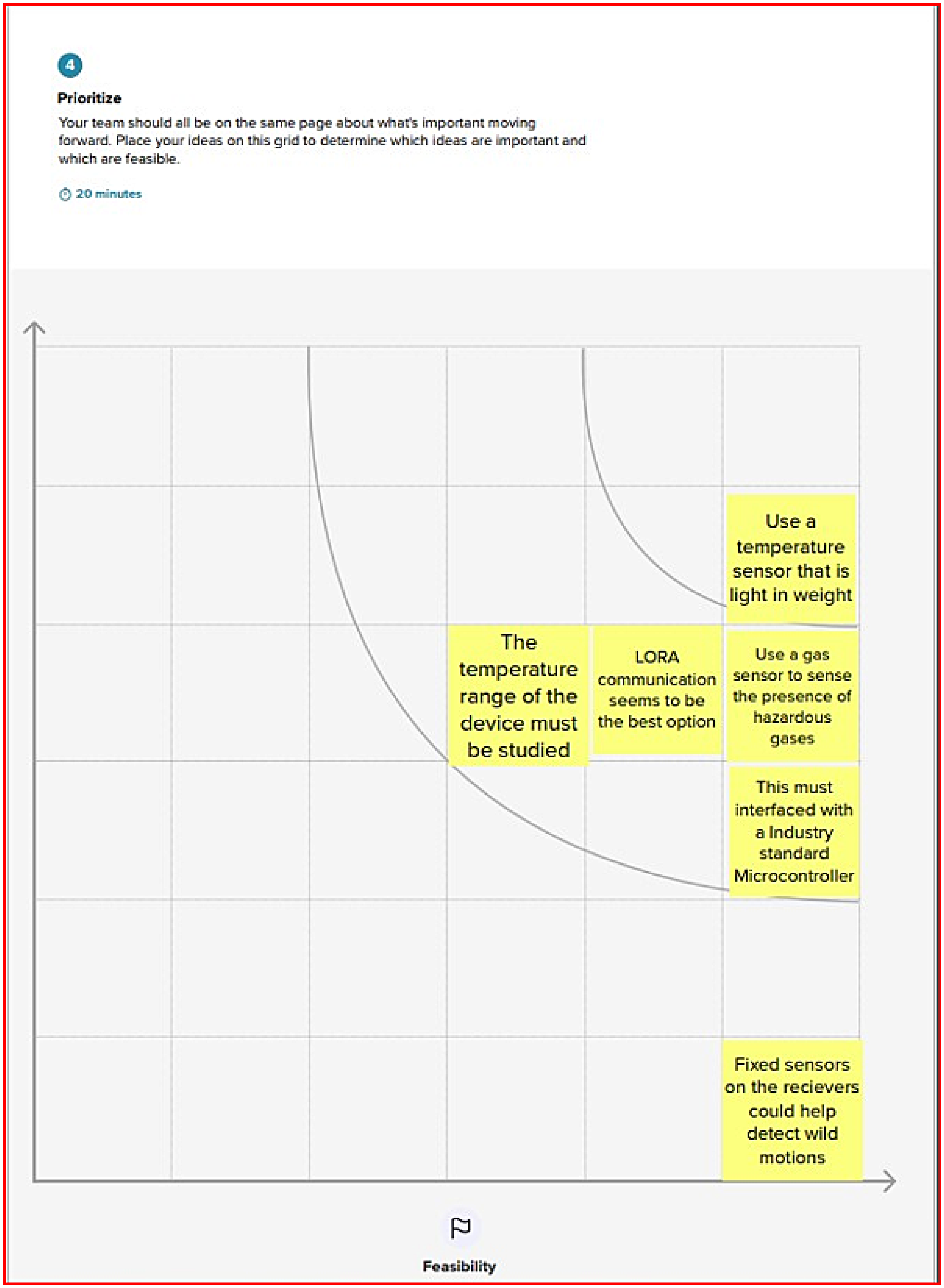
Thilak

- Use a temperature sensor that is light in weight
- Use a gas sensor to sense the presence of hazardous gases
- This must interfaced with a Industry standard Microcontroller
- The Microcontroller must light in weight

Bhuvanesh

- Monitoring of motors in hazardous areas using sensors
- Fixed sensors on the recievers could help detect wild motions
- Its better to embed only 2 of the features for first phase

Step-3: Idea Prioritization



3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To monitor the temperature in Industrial plants by using IOT enabled devices.
2.	Idea / Solution description	Temperature sensors can be interfaced with a light weight microcontroller, which will make the monitoring device wearable for industrial workers. The Microcontrollers are to be interfaced with LORA communication devices to enable them to communicate with distributed gateways that can be placed across the plant. The Gateways are to be connected to the internet through which the temperature across the plant can be monitored.
3.	Novelty / Uniqueness	The devices available currently with similar functions are fixed on single position, and hence are costly to monitor the temperature across the area of the device. The proposed solution's novelty is that, The sensing device are mobile and hence can give accurate measurements of temperature in periphery of the Industry worker.
4.	Social Impact / Customer Satisfaction	The proposed solution upon its full completion can ensure better safety at an individual level. This boosts the confidence of the Industrial Personnel in their work environment which has the potential to boost the productivity of the Industry.
5.	Business Model (Revenue Model)	The proposed solution acts as a cost effective upgrade to the current technology used in the industry.
6.	Scalability of the Solution	The solution can be scaled to include gas sensors and other wide variety of sensors enabling better vigilance on safety.

3.4 Proposed Solution Fit

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

The main customers for this product is from Industrial plants. This includes both small scale and large scale manufacturing plants. This Product can be used in any manufacturing/industrial plant that involves a fire risk, Hazardous gas risks.

6. CUSTOMER CONSTRAINTS CC

Cost of available devices.

Unsatisfactory user interface.

Inaccurate measuring devices.

False alarms.

5. AVAILABLE SOLUTIONS AS

Indoor positioning systems.

Bluetooth based Beacon systems for data transfer.

Fixed sensor based gas detection systems.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Select appropriate sensors for necessary gas detection.

Select necessary gateways for communications.

Find additional features that can be added into the system.

9. PROBLEM ROOT CAUSE RC

The problem exists because of lack of customized products.

The cost of the existing products is high, but the satisfactory user interface is lacking.

Lack of flexibility in the available products, that is, the devices can only measure a predefined set of gases that might not be needed in a given industrial plant.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?

Install gas detection sensors with displays and alarms.

Buy necessary devices from third party vendors and use their own teams to build and maintain the monitoring systems.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

The Customers here are triggered by the risks of their employees. Fire hazard, hazardous gasses can be found in most industrial plants.

The need to ensure safety both by regulatory authority and workspace ethics are the prominent triggers.

10. YOUR SOLUTION SL

Our solution is to use a customizable set of gas sensors that are wearable. This makes sure that the sensors detect these gasses in the immediate vicinity of the working personnel. The data is collected in a cloud based server from which it can be linked to a mobile application. This improves the user interface issues and solves the need for a large number of sensors.

8. CHANNELS of BEHAVIOR CH

8.1 ONLINE

Search for companies with the appropriate solutions.

Search for Products from online platforms.

8.2 OFFLINE

Build a team by recruiting people with appropriate skills.

Install the sensors by giving contracts to third party companies.

Identify strong TR & EM

4. EMOTIONS: BEFORE / AFTER EM

The customers feel insecure/uncertain of the environmental standards.

Once the proposed solution is installed it improves their Confidence in safety measures.

4.REQUIREMENT ANALYSIS

4.1 Functional Requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Temperature Sensing	Sensors sensing the temperature in DC Voltage(0-5V).
FR-2	Gas Sensing	Sensors measuring composition and concentration of gases.
FR-3	Cloud based database	Able to collect data from sensor nodes.
FR-4	Web UI	Display the collected data in a website.
FR-5	Mobile App	Mobile interface for administrative uses.

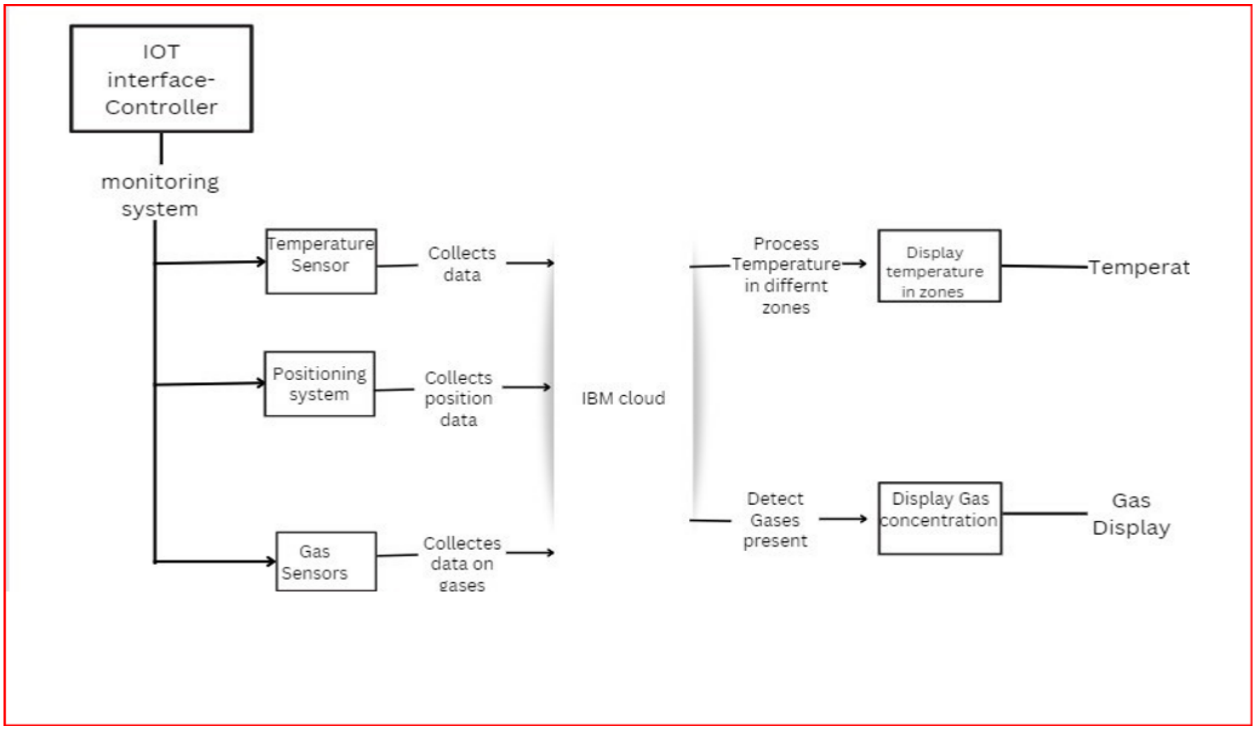
4.2 Non-functional Requirements

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Temperature sensors are interfaced with a light weight microcontroller, which will make the monitoring devices wearable.
NFR-2	**Security**	The solution makes use of secured cloud based data collection , so that the data will not be lost and a third party member cannot interfere in communication with the workers.
NFR-3	**Reliability**	Since the proposed solution makes use of wearable sensors , the workers will be alerted in case of any emergency (e.g., Very high temperature due to gas leakage) in the particular area where they are working.
NFR-4	**Performance**	The proposed solution communicates with the workers using the data being collected in the cloud based database. The efficiency of communication is high.
NFR-5	**Availability**	The sensors now available in market are costly. The solution can make use of mobile sensors to monitor temperature in the industrial areas.
NFR-6	**Scalability**	The solution can be scaled to include gas sensors and other wide variety of sensors improving the safety measures.

5.PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution and Technical Architecture

Solution Architecture

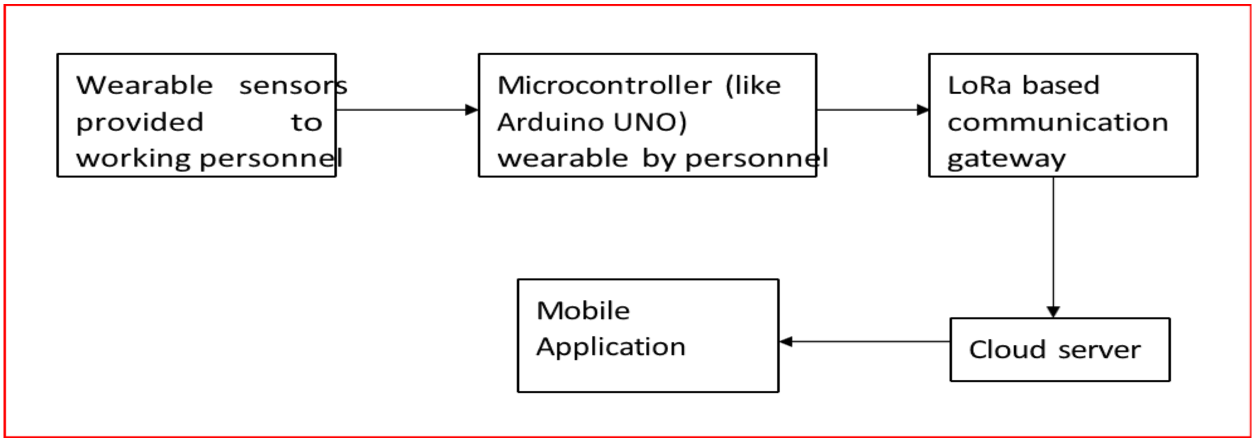
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

1. To monitor temperature in areas of industrial plants using IOT enabled devices .
2. Temperature sensors are interfaced with a light weight microcontroller which make them wearable. These microcontrollers are interfaced with LORA communication devices to communicate with distributed gateways. These gateways are connected to the internet through which the temperature is monitored.
3. The devices (E.g., BEACON) with similar features which are available now are fixed in a single position and they are costly.
4. The sensing devices are mobile and hence they can give accurate measurements

of temperature across the industrial plant.

5. The proposed solution communicates about the temperature across the industrial plant monitored through sensors, to the users .It can be scaled to include gas sensors and other wide variety of sensors enabling safety of the users.

Solution Architecture Diagram



Technical Architecture

Technology Stack

USER **IBM CLOUD** **ADMIN**

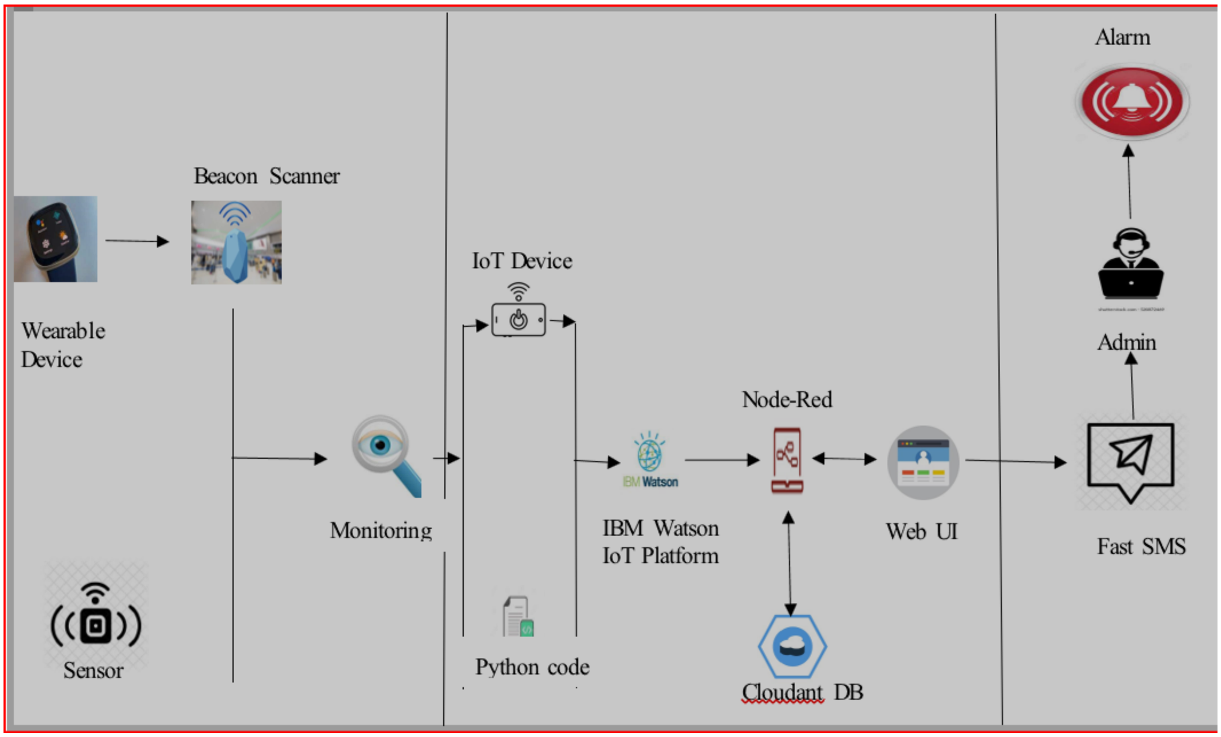


Table-1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI	HTML, JavaScript
2.	Application Logic-1	To interface the sensors with Arduino uno and for its communication with the Internet gateway	Arduino/embedded C
3.	Application Logic-2	For the mobile app	Python
4.	Application Logic-3	Scanner to get the user condition	IBM Watson IoT
5.	Database	For managing the database and retrieving data	MySQL
6.	Cloud Database	Database Service on Cloud	IBM Cloudant etc.
7.	File Storage	App code is stored along with the API keys	IBM Block Storage

8.	External API-2	To create username credentials for login	Username API
9.	Machine Learning Model	Alert rising system	Object Recognition Model, etc.
10.	Infrastructure (Server / Cloud)	To manage data to and from the server	Node Red

Table-2: Application Characteristics:

S.No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	We are using the MIT App inventor as our open source framework	MIT App Inventor
2.	Security Implementations	SHA 256 and other network protocols with security protocols will be used	SHA-256
3.	Scalable Architecture	The IBM cloud provides a robust framework which can be used to scale the framework to a large inter-connected network.	IBM Cloud
4.	Availability	The load balancers can be implemented to make sure the framework sustains the traffic generated without compromising the performance.	IBM Load balancer
5.	Performance	This is used to increase the performance of hosted application.	IBM instance

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Working Personnel (Mobile user)	Monitoring	USN-1	As a user, I need to know the Temperature around me.	I can view the temperature	High	Sprint-1
Working Personnel	Monitoring	USN-2	As a user, I need to know the Gas composition and Concentration near me.	I can gas data	High	Sprint-1
Working Personnel	Registration	USN-3	As a user, I can register for the application with my industry credentials	I can register and access the data through my profile	Low	Sprint-2
Working Personnel	Positioning	USN-4	As a user, I can view the location of my device.	I must be able to know my position with respect to plant and nearest exits	Medium	Sprint-3
Supervisor (Web user)	Supervision	USN-5	I can view the environment condition through the web UI.	I can view the Conditions in real-time	Medium	Sprint-2

Customer Care Executive	Maintenance	USN-6	For customer care I must know the inner architecture.	I need proper documentation.	Medium	Sprint-3
Administrator	Administration	USN-7	I need full access over the devices and the network.	I can view the data rate and other technical details using my UI.	Low	Sprint-4

6.PROJECT PLANNING &SCHEDULING

6.1 Sprint Planning and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Temperature monitoring	USN-1	As a user, I need to know the temperature of the Industrial plant.	4	High	Thilak, Manibalan.
Sprint-1	Gas Monitoring	USN-2	As a user, I need the gas composition and/or concentration around me.	2	Medium	Bharani, Bhuvanesh.
Sprint-2	IOT dashboard interfacing	USN-3	As a user, I must be able to view the data using internet.	4	High	Bharani, Manibalan
Sprint-3	Web UI	USN-4	As a user, I must be able to access data from a website.	2	Low	Thilak, Bhuvanesh
Sprint-4	Mobile UI	USN-5	As a user, I can view the data log in a Mobile application.	2	Low	Bharani, Manibalan.

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint Start Date (Planned)	Story Points Completed(as on planned end date)	Sprint (Actual Date)
Sprint-1	20	6 Days	24 Oct 2022	24 Oct 2022	6	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	4	7-Nov-2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	2	12-Nov-2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	2	19-Nov-2022

Velocity

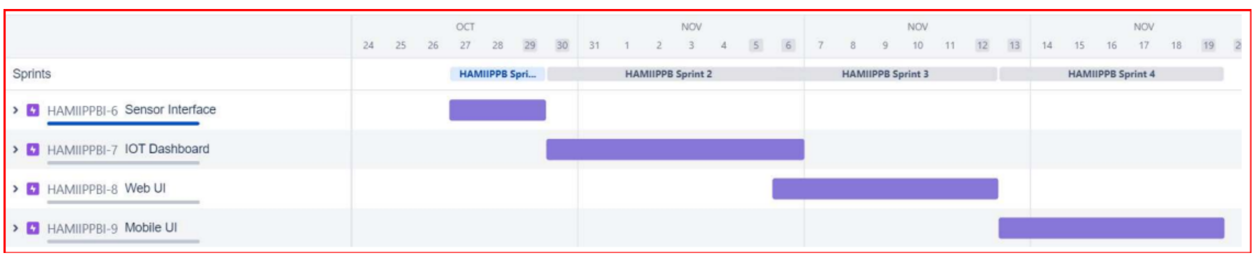
We have a 6-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{6}$$

Burndown Chart

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

6.3 Reports from JIRA



7. CODING AND SOLUTIONING

7.1 Feature 1

The Code uses MQTT protocol to publish information to IBM Watson Platform. This information is processed by Node-Red tools to display in the Dashboard. The code transfers these information in the form of json files. This is done by using the Arduino C code dumped on the ESP32 device.

7.2 Feature 2

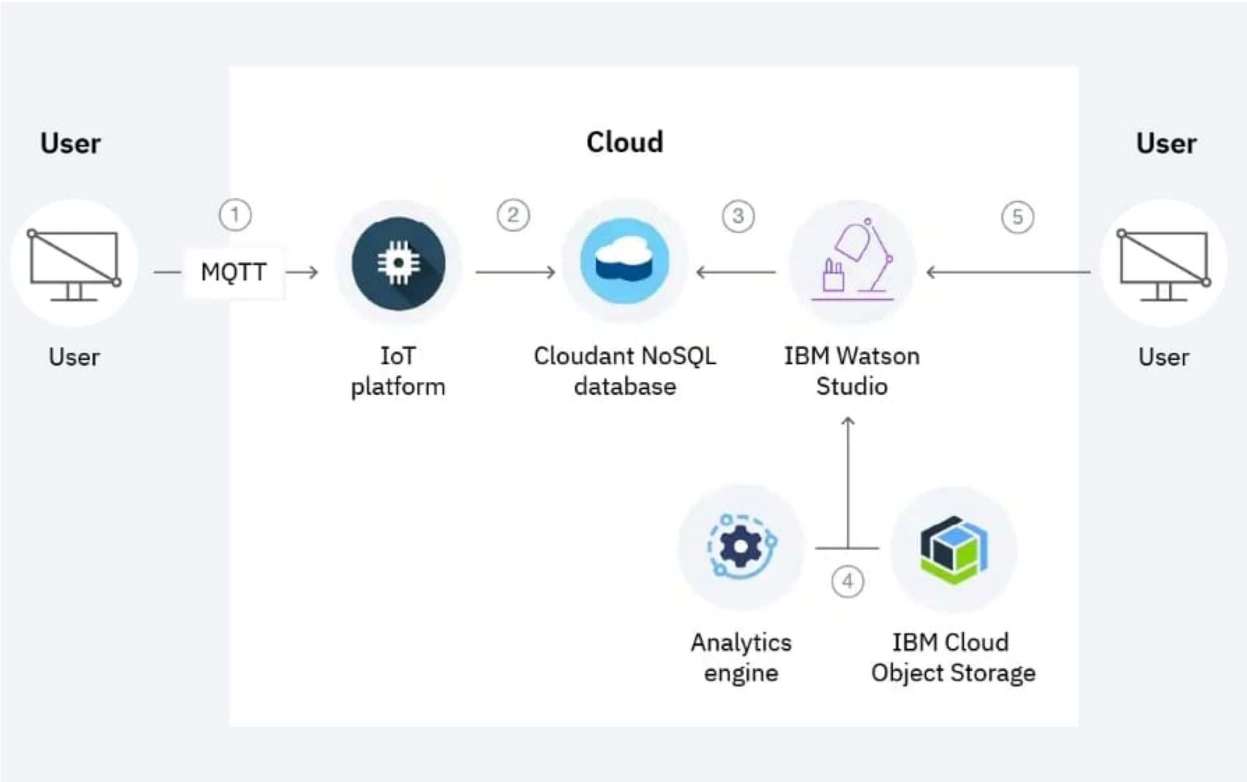
The HTTP protocol is used between the mobile app to receive data from the IBM Watson platform. The Node-Red flow-1 transfers the data received in the form of the json file when the given web link is hit by the mobile app.

This Protocol is also used by the ESP32 microcontroller to request data on the button press in the dashboard and the app. The ESP32 sends the HTTP request periodically to check the most recent button press. Necessary actions are taken based on the data received from the HTTP request.

The Mobile app uses the HTTP requests to give data on the button press. The App hits web2 URL whenever ON Button is pressed and web3 URL otherwise. Based on this data node red flow-2 determines the most recent button press which must be sent to the ESP32.

7.3 Database Schema

A fully managed, distributed database optimized for heavy workloads and fast-growing web and mobile apps, IBM Cloudant is available as an IBM Cloud® service with a 99.99% SLA. Cloud ant elastically scales throughput and storage, and its API and replication protocols are compatible with Apache CouchDB for hybrid or multi cloud architectures.



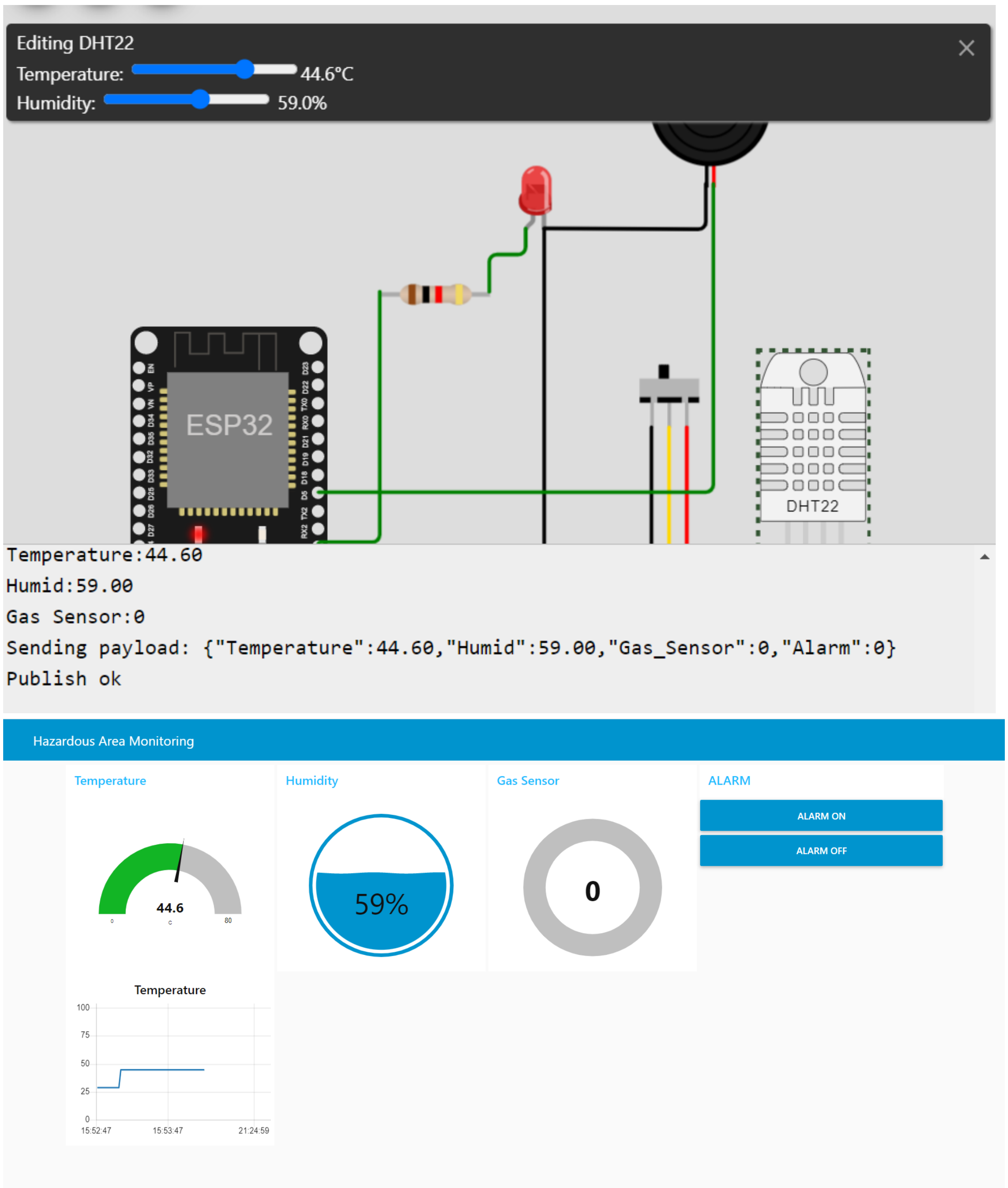
8. TESTING

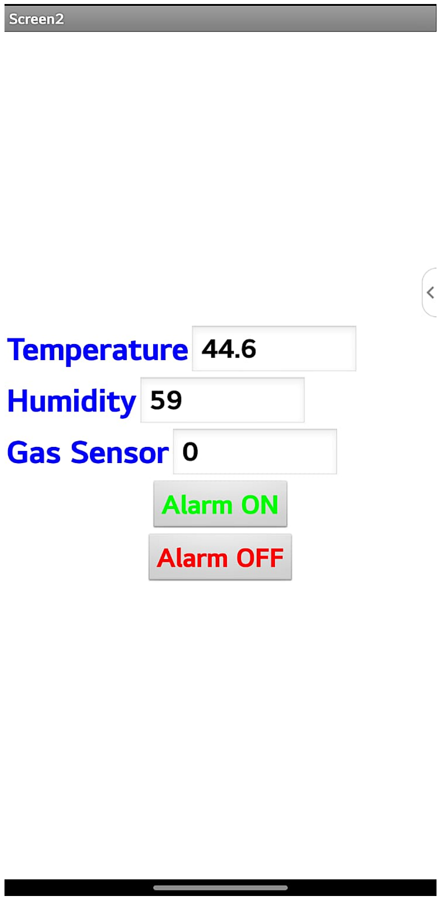
8.1 Test Cases

Temperature = 44.6 C

Humidity = 59%

Gas Sensor = 0

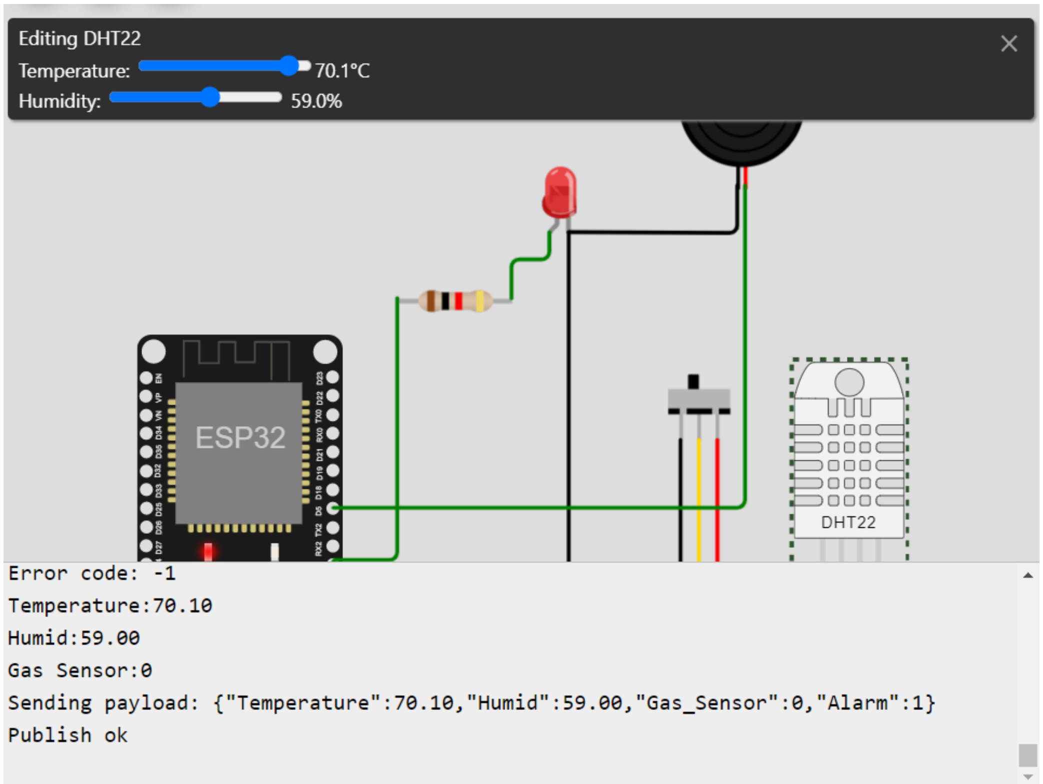


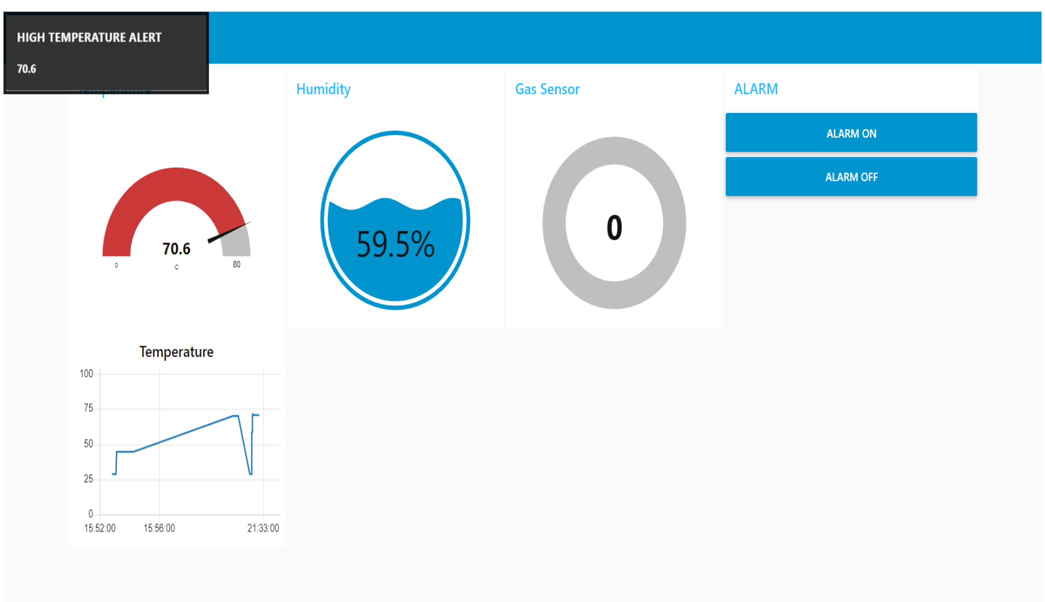

Screen2
Temperature
44.6
Humidity
59
Gas Sensor
0
Alarm ON
Alarm OFF

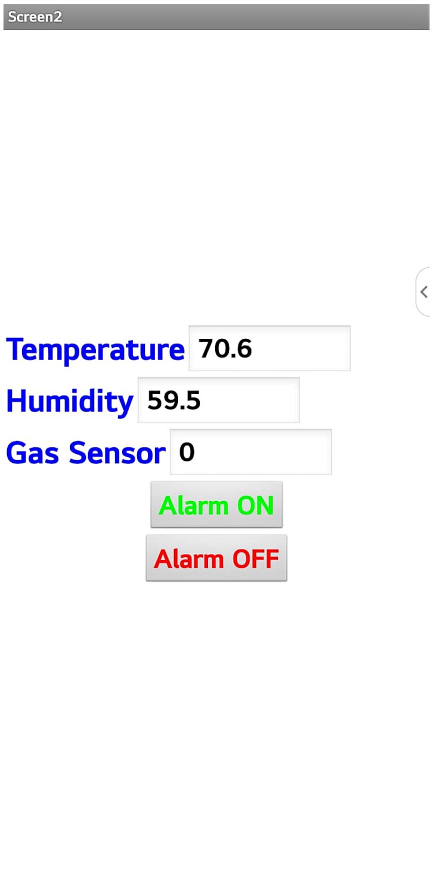
Temperature = 70.6 C

Humidity = 59.5%

Gas Sensor = 0



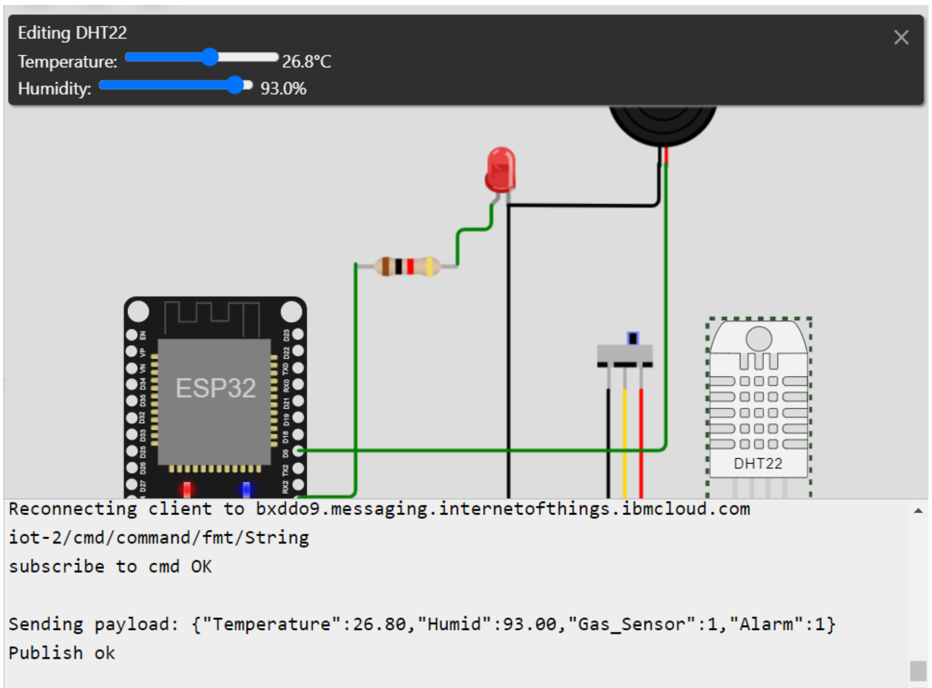


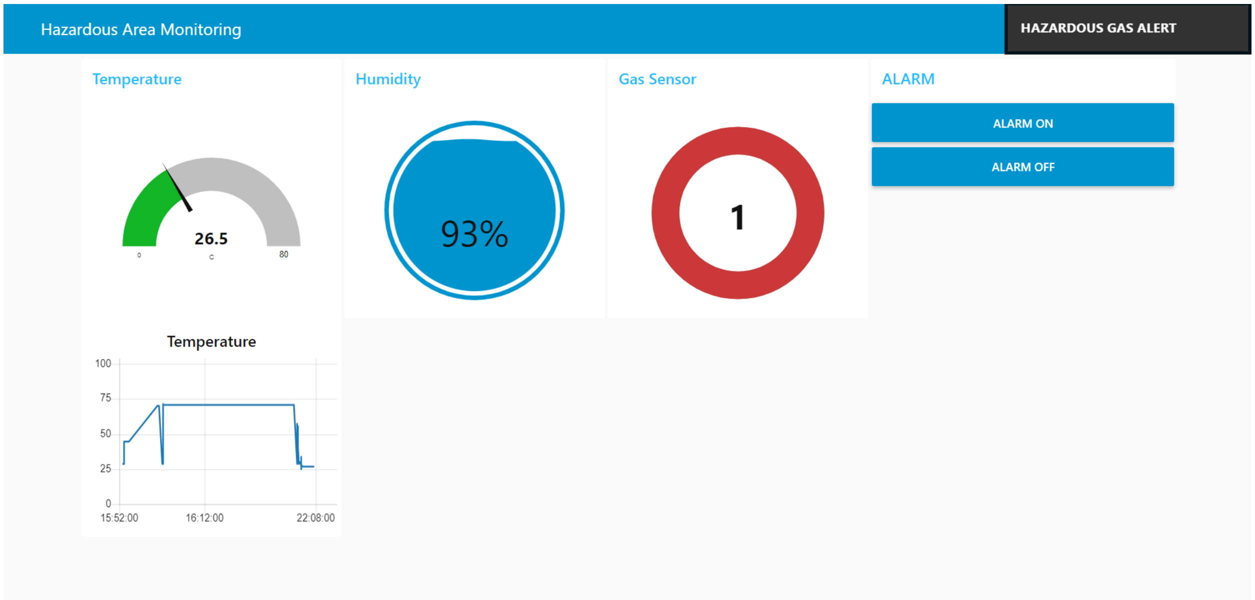

Screen2
Temperature
70.6
Humidity
59.5
Gas Sensor
0
Alarm ON
Alarm OFF

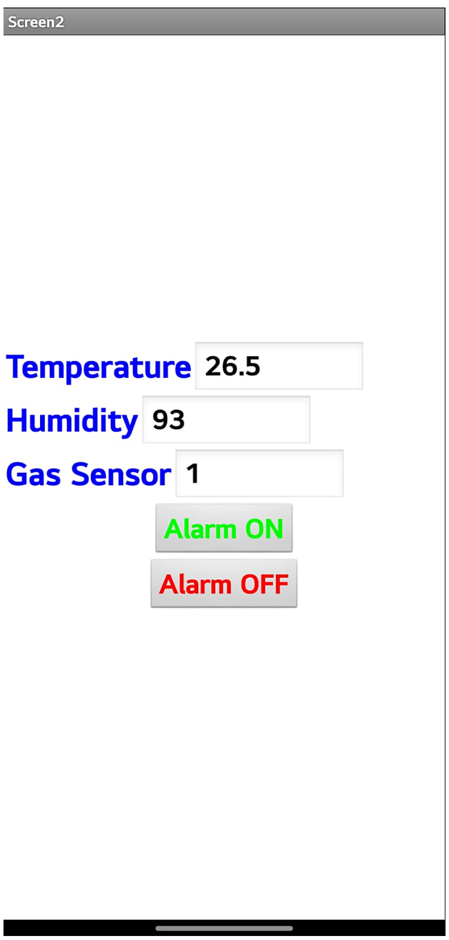
Temperature = 26.5 C

Humidity = 93%

Gas Sensor = 1




Hazardous Area Monitoring
HAZARDOUS GAS ALERT
Temperature
Humidity
Gas Sensor
ALARM
ALARM ON
ALARM OFF
26.5
0
c
80
93%
1
Temperature
100
75
50
25
0
15:52:00
16:12:00
22:08:00


Screen2
Temperature
26.5
Humidity
93
Gas Sensor
1
Alarm ON
Alarm OFF

8.2 User Acceptance Testing

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
LoginPage_TC_OO1	Functional	IoT End Device	Verify whether the data collected from sensor is preperly read in to the End device[ESP32]	ESP32 device and proper interface connections.	1.Connect the sensors with End Device 2.Run the simulation 3.Vary the inputs and check whether correct output is displayed in serial monitor	https://wokwi.com/projects/347582504242250322	The output must match with the input	Working as expected	Pass
LoginPage_TC_OO2	Functional	IBM IoT Cloud Platform	Verify whether the databis received from the IoT end device to Cloud(IBM Watson)	IBM Watson Internet of Things Platform	1.Enter URL for IBM watson 2.Generate data to send to cloud 3.Verify the received data by logging in to the IBM Watson Platform	https://bxddo9.internetofthings.ibmcloud.com/dashboard/devices/browse	The data sent from End device must be received in the dashboard.	Working as expected	Pass
LoginPage_TC_OO3	User Interface	Web UI Dashboard	Verfiy the dashboard functions properly	Dashboard is configured	1.Enter URL 2.Run the Wokwi simulation 3.Vary the simulation conditions 4.Check whether the data is displayed correctly. 5.Click the button and verify its functionality.	http://169.51.205.238:32312/ui	The data must be displayed properly and button functions must work correctly.	Working as expected	Pass
LoginPage_TC_OO4	User Interface	Mobile App UI	Verify whehter user can use the app to view data and buttons	Monile app is built and deployed.	1.Enter URL(https://shopenzer.com/) and click go 2.Click on My Account dropdown button 3.Enter InValid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	https://github.com/IBM-EPBL/IBM-Project-465-1658302480/blob/2ad541aa8fdf84278f786d8fa06223866378389e/Project%20Development%20Phase/Sprint-4/HazardousAreaMonitoring1.apk	The app must be installed properly and the desired functions are performed correclty	Working as expected	Pass

9.RESULTS

The Mobile app and the web UI dashboard has been properly interfaced with the IoT end-node device. Hence there exists a proper data transfer between the sensors, IoT end node, The web UI and the Mobile app. Therefore the undertaken milestones have been reached successfully.

9.1 Performance

The Performance of the built software satisfies the Functional and Non Functional Requirements laid beforehand. Hence the application architecture must be capable of handling the data rates and throughput need of the Industry.

10. ADVANTAGES AND DISADVANTAGES

Advantages

- **Cost-effectiv**e - IoT technology provides the most economic and budgeted solutions for its users to subsequently pave its way in industrial advancement. It uses high-end devices like sensors and gateway connectivity that automate the industrial processes on a pocket-friendly budget.
- **End-to-end Solution** - IoT provides a smart level monitoring end-to-end solution, which when installed gives you the freedom to multitask. It is an automated system that processes industrial work without delays and errors.

- **Real-time Supervision** - A sensor-enabled level monitoring solution in the chemical industry allows real-time supervision through on-spot data extraction without any delays and providing the desired reports for analysis.

- **Risk Prevention** - The involvement of a smart level monitoring solution is a great way to prevent industrial risks due to large exposure to chemicals.

Disadvantages

- **Security and privacy** - Keeping the data gathered and transmitted by IoT devices safe is challenging, as they evolve and expand in use.

- **Connectivity and power dependence** - Many devices have a dependency on the internet and continuous power to function properly. When either goes down, so does the device and anything connected to it.

- **Integration** - There's currently no consensus regarding IoT protocols and standards, so devices produced by different manufacturers might not work with existing technology.

11.CONCLUSION

IoT-enabled industrial monitoring systems have become increasingly popular in a variety of industries because they improve safety standards by providing real-time monitoring of critical parameters such as temperature, humidity, and smoke, as well as alerting officials and workers regularly. The implementation is not only for safety reasons, but it also has the potential to increase industry yields. In our project, the Internet of Things (IoT) is used to collect data and communicate through the internet. We hope that our project will be beneficial enough to be implemented in industries across India, saving lives and property from accidents and risks that are often overlooked by industry personnel and users.

12. FUTURE SCOPE

Industrial IoT strategy should include machine learning and big data technology in addition to data from devices and sensors, harnessing the combination of existing sensor data, machine to machine (M2M) connectivity, and automation technologies to deliver greater insight back to the business.Automation through IOT can help to get rid of the short distance communication. Thus, introducing internet in industries can help to have control over the application from all over world.

13. APPENDIX

Source Code

ESP32:

```
#include ."DHT.h"/ Library for dht22
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include <HTTPClient.h>//library for HTTP requests


#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
const int LED = 4;
//GAS SENSOR MQ-02
#define GAS_SENSOR 2
String alarmon = "{\"Alar\":1}";

//Your Domain name with URL path or IP address with path
String serverName = "http://169.51.205.238:32312/command";

unsigned long lastTime = 0;
unsigned long timerDelay = 1000;

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);


//-------credentials of IBM Accounts------

#define ORG "bxddo9"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "Assign4"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "45625689713"     //Token
```

```
String data3;
float h, t;
int val;



//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform and format
in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type
AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id

//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing
parameter like server id,portand wificredential

const int DHT_PIN = 15;
bool al;

void setup() {
  Serial.begin(115200);
  Serial.println();
  pinMode(LED,OUTPUT);
  //digitalWrite(LED,HIGH);
  delay(10);
  wificonnect();
  mqttconnect();
}

void loop() {
```

```
val = digitalRead(GAS_SENSOR);

h = dht.readHumidity();
t = dht.readTemperature();
Serial.print("Temperature:");
Serial.println(t);
Serial.print("Humid:");
Serial.println(h);
Serial.print("Gas Sensor:");
Serial.println(val);
if(t > 45 || val == 1)
{  al = 1;
}
else
{  al = 0;
}
PublishData(t, h, val, al);
delay(1000);
if (!client.loop()) {
  mqttconnect();
}
if ((millis() - lastTime) > timerDelay) {
  //Check WiFi connection status
  if(WiFi.status()== WL_CONNECTED){
    HTTPClient http;

    String serverPath = serverName + "?temperature=24.37";

    // Your Domain name with URL path or IP address with path
    http.begin(serverPath.c_str());

    // Send HTTP GET request
    int httpResponseCode = http.GET();
```

```
      if (httpResponseCode>0) {
       // Serial.print("HTTP Response code: ");
        //Serial.println(httpResponseCode);
        String payload = http.getString();
        //Serial.println(payload);

        if(payload == alarmon)
        {
          digitalWrite(LED,HIGH);
          tone(5,262,2000);
        }
        else
        {
          digitalWrite(LED,LOW);
          digitalWrite(5,LOW);
        }
      }
      else {
        Serial.print("Error code: ");
        Serial.println(httpResponseCode);
      }
      // Free resources
      http.end();
    }
    else {
      Serial.println("WiFi Disconnected");
    }
    lastTime = millis();
  }
}

/*.....................................retrieving to Cloud..............................*/

void PublishData(float temp, float humid, int vol,int alarm) {
```

```
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm
cloud
  */
  String payload = "{\"Temperature\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
  payload += ",""\"Gas_Sensor\":";
  payload += val;
  payload += ",""\"Alarm\":";
  payload += al;
  payload += "}";


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


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish
ok in Serial monitor or else it will print publish failed
    Serial.println("");
  } else {
    Serial.println("Publish failed");
  }

}


void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
```

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

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);

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

void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
```

```
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {
    Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
  if(data3=="lighton")
  {
Serial.println(data3);
digitalWrite(LED,HIGH);
  }
  else
  {
Serial.println(data3);
digitalWrite(LED,LOW);
}
data3="";
}
```

SCHEMATIC JSON:


```
{
  "version": 1,
  "author": "Anonymous maker",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -7.34, "left": -80.66, "attrs": {} },
    {
```

```
  "type": "wokwi-dht22",
  "id": "dht1",
  "top": 8.43,
  "left": 257.5,
  "attrs": { "temperature": "28.5", "humidity": "29.5" }
},
{
  "type": "wokwi-resistor",
  "id": "r1",
  "top": 120.44,
  "left": 132.83,
  "attrs": { "value": "1000" }
},
{ "type": "wokwi-slide-switch", "id": "sw1", "top": 15.09, "left": 192.84, "attrs": {} },
{
  "type": "wokwi-led",
  "id": "led1",
  "top": -99.87,
  "left": 116.83,
  "attrs": { "color": "red", "flip": "1" }
},
{
  "type": "wokwi-resistor",
  "id": "r2",
  "top": -32.87,
  "left": 53.49,
  "attrs": { "value": "1000" }
},
{
  "type": "wokwi-buzzer",
  "id": "bz1",
  "top": -163.91,
  "left": 198.57,
  "attrs": { "volume": "0.1" }
```

```
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v40.16", "h-69" ] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "esp:GND.1", "dht1:GND", "black", [ "h0" ] ],
    [ "r1:1", "esp:D2", "gold", [ "v-0.61", "h-35.35" ] ],
    [ "sw1:1", "esp:GND.1", "black", [ "v0" ] ],
    [ "sw1:3", "esp:3V3", "red", [ "v0" ] ],
    [ "sw1:2", "r1:2", "gold", [ "v0" ] ],
    [ "r2:2", "led1:A", "green", [ "v-19.96", "h3.21" ] ],
    [ "r2:1", "esp:D4", "green", [ "v134.71", "h-19.99" ] ],
    [ "led1:C", "esp:GND.1", "black", [ "v181.74", "h-89.66" ] ],
    [ "led1:C", "bz1:1", "black", [ "v0.41", "h-37.66" ] ],
    [ "bz1:2", "esp:D5", "green", [ "v0" ] ]
  ]
}
```

Nodered Code:

FLOW-1:

[{"id":"1f06c26950396cf7","type":"tab","label":"Flow 1","disabled":false,"info":"","env":[]},{"id":"ffebc4772f3a739c","type":"comment","z":"1f06c26950396cf7","name":"Hazardous Area Monitoring","info":"","x":750,"y":20,"wires":[]},{"id":"dbe783e43b60d2c2","type":"ibmiot in","z":"1f06c26950396cf7","authentication":"apiKey","apiKey":"cc617d42bcb990ae","inputType":"evt","logicalInterface":"","ruleId":"","deviceId":"Assign4","applicationId":"","deviceType":"ESP32","eventType":"+","commandType":"","format":"json","name":"IBM IoT","service":"registered","allDevices":"","allApplications":"","allDeviceTypes":"","allLogicalInterfaces":"","allEvents":true,"allCommands":"","allFormats":"","qos":0,"x":150,"y":260,"wires":[["7630b4556b138fa3","41e96881901d6462","e548c60d7a4f54f4","8b5ceefdad716174","19598072b2da5ea9","1b49a57b0f69f6de","6e7a8d28d7ebaa78"]]},{"id":"7630b4556b138fa3","type":"debug","z":"1f06c26950396cf7","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","statusVal":"","statusType":"auto","x":470,"y":80,"wires":[]},{"id":"41e96881901d6462","type":"function","z":"1f06c26950396cf7","name":"","func":"global.set('Temperature',msg.payload.Temperature)\nglobal.set('Alarm',msg.payload.Alarm)\n\nmsg.payload = msg.payload.Temperature;\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":460,"y":180,"wires":[["aba6db961e70bb03","7aad03a96b2c68a6"]]},{"id":"aba6db961e70bb03","type":"ui_gauge","z":"1f06c26950396cf7","name":"Temperature","group":"53e4704e8ee1e14c","order":1,"width":6,"height":5,"gtype":"gage","title":"","label":"C","format":"{{value}}","min":0,"max":"80","colors":["#c73d3d","#12b525","#ca3838"],"seg1":"15","seg2":"45","x":690,"y":180,"wires":[]},{"id":"d04f9ac167e5054f","type":"ui_gauge","z":"1f06c26950396cf7","name":"Humidity","group":"57795ad37139dca5","order":1,"width":0,"height":0,"gtype":"wave","title":"","label":"%","format":"{{value}}","min":0,"max":"100","colors":["#00b500","#03b03f","#ca3838"],"seg1":"","seg2":"","x":680,"y":340,"wires":[]},{"id":"e1b6528ff5d5e564","type":"ui_gauge","z":"1f06c26950396cf7","name":"Gas Sensor","group":"dab251ccba4334d0","order":1,"width":0,"height":0,"gtype":"donut","title":"","label":"","format":"{{value}}","min":0,"max":"1","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":690,"y":420,"wires":[]},{"id":"e548c60d7a4f54f4","type":"fu nction","z":"1f06c26950396cf7","name":"","func":"global.set('Humid',msg.payload.Humid)\n\n\nmsg.payload= msg.payload.Humid;\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":460,"y":340,"wires":[["d04f9ac167e5054f"]]},{"id":"8b5ceefdad716174","type":"function","z":"1f06c26950396cf7","name":"","func":"global.set('Gas_Sensor',msg.payload.Gas_Sensor)\n\n\nmsg.payload= msg.payload.Gas_Sensor;\n\nreturn msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":460,"y":420,"wires":[["e1b6528ff5d5e564"]]},{"id":"c8d4ef8da16caf16","type":"ui_toast","z":"1f06c26950396cf7","position":"top right","displayTime":"1.5","highlight":"","sendall":true,"outputs":0,"ok":"OK","cancel":"","raw":true,"topic":"HAZARDOUS GAS ALERT ","name":"ALERT","x":680,"y":500,"wires":[]},{"id":"19598072b2da5ea9","type":"function","z":"1f06c26950396cf7","name":"","func":"msg.payload= msg.payload.Gas_Sensor;\nif(msg.payload==1){\nreturn msg;\n}","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":460,"y":500,"wires":[["c8d4ef8da16caf16"]]},{"id":"1b49a57b0f69f6de","type":"function","z":"1f06c26950396cf7","name":"","func":"msg.payload= msg.payload.Temperature;\nif(msg.payload>=45){\nreturn msg;\n}\n","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":460,"y":260,"wires":[["471c2151f55f6b8d"]]},{"id":"471c2151f55f6b8d","type":"ui_toast","z":"1f06c26950396cf7","position":"top left","displayTime":"1.5","highlight":"","sendall":true,"outputs":0,"ok":"OK","cancel":"","raw":false,"topic":"HIGH TEMPERATURE ALERT","name":"ALERT-1","x":680,"y":260,"wires":[]},{"id":"6e7a8d28d7ebaa78","type":"cloudant out","z":"1f06c26950396cf7","name":"","cloudant":"5d28e39af5fdc7c6","database":"hazardous_area_monitoring","service":"_ext_","payonly":true,"operation":"insert","x":520,"y":600,"wires":[]},{"id":"1d4a1ed271980e06","type":"http in","z":"1f06c26950396cf7","name":"","url":"/data","method":"get","upload":false,"swaggerDoc":"","x":840,"y":300,"wires":[["51a40ea7e18d2fe8"]]},{"id":"b3ad098161022243","type":"http response","z":"1f06c26950396cf7","name":"","statusCode":"","headers":{},"x":1150,"y":300,"wires":[]},{"id":"51a40ea7e18d2fe8","type":"function","z":"1f06c26950396cf7","na

me":"","func":"msg.payload =
{'Temperature':global.get('Temperature'),'Humidity':global.get('Humid'),'Gas_Sensor':
global.get('Gas_Sensor'),'Alarm':global.get('Alarm')}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":1000,"y":300,"wires":[["b3
ad098161022243"]]},{"id":"7aad03a96b2c68a6","type":"ui_chart","z":"1f06c26950396
cf7","name":"","group":"53e4704e8ee1e14c","order":1,"width":"6","height":"5","label":"Te
mperature","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"line
ar","nodata":"","dot":false,"ymin":"0","ymax":"100","removeOlder":1,"removeOlderPoints
":"","removeOlderUnit":"3600","cutout":0,"useOneColor":false,"useUTC":true,"colors":["
#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#
c5b0d5"],"outputs":1,"useDifferentColor":false,"x":730,"y":120,"wires":[[]]},{"id":"cc617
d42bcb990ae","type":"ibmiot","name":"","keepalive":"60","serverName":"bxddo9.mess
aging.internetofthings.ibmcloud.com","cleansession":true,"appId":"","shared":false},{"
id":"53e4704e8ee1e14c","type":"ui_group","name":"Temperature","tab":"921bcc13fd3
771df","order":1,"disp":true,"width":"6","collapse":false},{"id":"57795ad37139dca5","typ
e":"ui_group","name":"Humidity","tab":"921bcc13fd3771df","order":2,"disp":true,"width
":"6","collapse":false},{"id":"dab251ccba4334d0","type":"ui_group","name":"Gas
Sensor","tab":"921bcc13fd3771df","order":3,"disp":true,"width":"6","collapse":false},{"i
d":"5d28e39af5fdc7c6","type":"cloudant","host":"https://apikey-v2-
tlousred1b9s5pmsh4ahzzbak8vkotqo00eq8690avd:2b56fc49427f0246b9bed0ba3
0c12f2a@60491405-daf0-4bbc-a3db-b5d46cd515c5-
bluemix.cloudantnosqldb.appdomain.cloud","name":"db"},{"id":"921bcc13fd3771df",
"type":"ui_tab","name":"Hazardous Area
Monitoring","icon":"dashboard","disabled":false,"hidden":false}]

FLOW-2:

[{"id":"1819418ffaee3ee5","type":"tab","label":"Flow
2","disabled":false,"info":"","env":[]},{"id":"7f5cf345.63f56c","type":"http
response","z":"1819418ffaee3ee5","name":"","statusCode":"200","headers":{},"x":840,"y
":200,"wires":[]},{"id":"e71c7a7d.e7c598","type":"debug","z":"1819418ffaee3ee5","nam
e":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","x"

:850,"y":420,"wires":[]},{"id":"c7807102.3f433","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"command","method":"get","upload":false,"s
waggerDoc":"","x":160,"y":400,"wires":[["60410cde.562a34"]]},{"id":"60410cde.562a34
","type":"function","z":"1819418ffaee3ee5","name":"","func":"msg.payload =
{'Alar':global.get('Alar')}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":440,"y":400,"wires":[["7f5
cf345.63f56c","e71c7a7d.e7c598"]]},{"id":"698da80c7836cea4","type":"ui_button","z":
"1819418ffaee3ee5","name":"","group":"66bce42692eed6b1","order":3,"width":0,"heig
ht":0,"passthru":false,"label":"Alarm
ON","tooltip":"","color":"","bgcolor":"","icon":"","payload":"1","payloadType":"num","topic":"
topic","topicType":"msg","x":160,"y":200,"wires":[["6878957e52c678f6"]]},{"id":"68789
57e52c678f6","type":"function","z":"1819418ffaee3ee5","name":"","func":"if(msg.payl
oad == 1)\n{\n   msg.payload=1;\n}\nelse\n{\n
msg.payload=0;\n}\nglobal.set('Alar',msg.payload)\nmsg.payload =
{'Alar':global.get('Alar')}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":400,"y":200,"wires":[["e71
c7a7d.e7c598"]]},{"id":"9747945c8b11b89f","type":"ui_button","z":"1819418ffaee3ee
5","name":"","group":"66bce42692eed6b1","order":3,"width":0,"height":0,"passthru":fals
e,"label":"Alarm
OFF","tooltip":"","color":"","bgcolor":"","icon":"","payload":"0","payloadType":"num","topic":
"topic","topicType":"msg","x":150,"y":300,"wires":[["6878957e52c678f6"]]},{"id":"25c01
3b66f1577be","type":"function","z":"1819418ffaee3ee5","name":"","func":"global.set('A
lar',1)\nmsg.payload = {'Alar':1}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":420,"y":600,"wires":[["e71
c7a7d.e7c598","7f5cf345.63f56c"]]},{"id":"31722af4312e8a88","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"mobileon","method":"get","upload":false,"s
waggerDoc":"","x":168.39999389648438,"y":594.5999755859375,"wires":[["25c013b
66f1577be"]]},{"id":"f1fe5338e83a445a","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"mobileoff","method":"get","upload":false,"s
waggerDoc":"","x":180.39999389648438,"y":668.5999755859375,"wires":[["0dbffb65
ea00a57f"]]},{"id":"0dbffb65ea00a57f","type":"function","z":"1819418ffaee3ee5","nam
e":"","func":"global.set('Alar',0)\nmsg.payload = {'Alar':0}\nreturn

msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":420,"y":680,"wires":[["7f5cf345.63f56c","e71c7a7d.e7c598"]]},{"id":"66bce42692eed6b1","type":"ui_group","name":"ALARM","tab":"921bcc13fd3771df","order":4,"disp":true,"width":7,"collapse":false},{"id":"921bcc13fd3771df","type":"ui_tab","name":"Hazardous Area Monitoring","icon":"dashboard","disabled":false,"hidden":false}]

GitHub &Project Demo Link

https://github.com/IBM-EPBL/IBM-Project-465-1658302480

https://youtu.be/xTsF9jHngZQ