Gas Monitoring and Alerting System for Industries using IOT

1. Introduction

a. Project Overview

In the model, Arduino UNO R3 is used as a microcontroller which controls all the setup in the board like a brain. By the use of Arduino IDE dump the Arduino code to the microcontroller. MQ7 gas sensor is used to detect the Carbon monoxide in the environment, MQ135 sensor is used to detect Carbon di oxide and H2S in the air. In the 1st case, By the use of Rf transmitter and receiver imported wireless connection for sending and receiving the data which was sensed by the gas sensors. After receiving the data, the Vibration motor vibrate for the sake of alert the worker. In the 2nd case, when sensor sense the toxic gas, it will send the message to the authority by the ESP8266.

b. Purpose

To Save the workers working in hazardous conditions like Steel factories. This proposed model will immediately alert the worker by vibration. Worker suddenly intimated and go away from the toxic environment. From the ALERT message, the authority sends the rescue team to the hazardous area.

2. LITERATURE SURVEY

Literature Survey

Date	10 November 2022
Team ID	PNT2022TMID22149
Project Name	Gas leakage monitoring and alerting system for industries
Maximum Marks	2 Marks

ABSTRACT :

The Internet of things (IoT) is the system of gadgets, vehicles, and home machines that contain hardware, programming, actuators, and network which enables these things to interface, collaborate and trade information. IoT includes broadening Internet network past standard device, for example, work areas, workstations, cell phones and tablets, to any scope of generally stupid or non-web empowered physical device and ordinary articles. Installed with innovation, these gadgets can convey and connect over the Internet, and they can be remotely observed and controlled [1]. The meaning of the Internet of things has advanced because of union of numerous innovations, ongoing examination, AI, ware sensors, and implanted frameworks. Conventional fields of installed frameworks, remote sensor systems, control frameworks computerization (counting home and building mechanization), and others all add to empowering the Internet of things. A gas spill alludes to a hole of petroleum gas or different vaporous item from a pipeline or other regulation into any territory where the gas ought not be available. Since a little hole may steadily develop a hazardous convergence of gas, spills are perilous. Notwithstanding causing flame and blast dangers, holes can slaughter vegetation, including huge trees, and may discharge amazing ozone harming substances to the environment. Keywords: IOT, MQ5 sensor, Arduino module, GSM networks.

I. INTRODUCTION

The Internet of Things is a developing theme of specialized, social, and monetary centrality. Customer items, tough goods, cars and trucks, modern and utility segments, sensors, and other regular articles are being joined with Internet availability and amazing information systematic capacities that guarantee to change the manner in which we work, live, and play. Projections for the effect of IoT on the Internet and economy are amazing, with some foreseeing upwards of 100 billion associated IoT gadgets and a worldwide financial effect of more than $11 trillion by 2025. The Internet of Things (IoT) is an essential theme in innovation industry, strategy, and designing circles [1]. This innovation is encapsulated in a wide range of arranged items, frameworks, and sensors, which exploit headway's in processing power, gadgets scaling down, and organize interconnections to offer new capacities. The expansive scale usage of IoT gadgets guarantees to change numerous parts of the manner in which we live. For shoppers, new IoT items like Internet-empowered machines, home mechanization parts, and vitality the executive's gadgets are pushing us toward a dream of the "savvy home", offering greater security and vitality effectiveness. IoT frameworks like arranged vehicles, savvy traffic frameworks, and sensors implanted in streets and scaffolds draw us nearer to "brilliant urban areas", which help limit clog and vitality utilization. IoT innovation offers the likelihood to change horticulture, industry, and vitality creation and dissemination by expanding the accessibility of data along the esteem chain of generation utilizing arranged sensors.

II. METHODS AND MATERIAL SYSTEM:

Input, Output, Function, Success, Failure Input: Sensor data signal which is not regular or Change in Signal Output: End User get informed with alert buzzer and Display to LCD Functions:

1. Access ():- In this module we are going to access the feature provided by the module which Will include Sensor data access.
2. Control ():-In this module we are controlling the Alert System by using System which is connected to hardware or sensor data.
3. Broadcast ():-In this module we are going to broadcast the alert Display to LCD.

Success Conditions:

1. If such data which is received through sensors are not stable or are more than threshold it will predict that there is leakage situation .

Failure Conditions:

Desired output is not generated due to following failures.

1. Software Failure 2. Hardware Failure 3. Network Connection Failure

HARDWARE INFORMATION:

1. Arduino Uno

The Arduino Uno is a micro-controller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MIIz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

2. LCD (Liquid Crystal Display)

LCD stands for Liquid Crystal Display. LCD is finding wide spread use replacing LEDs (seven segment LEDs or other multi segment LEDs) because of the following reasons: 1. The declining prices of LCDs. 2. The ability to display numbers, characters and graphics. This is in contrast to LEDs, which are limited to numbers and a few characters. 3. Incorporation of a refreshing controller into the LCD, thereby relieving the CPU of the task of refreshing the LCD. In contrast, the LED must be refreshed by the CPU to keep displaying the data.

3. BUZZER

A buzzer or beeper is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. The first electric buzzer was invented in 1831 by Joseph Henry. They were mainly used in early doorbells until they were phased out in the early 1930s in favor of musical chimes, which had a softer tone. Piezoelectric buzzers, or piezo buzzers, as they are sometimes called, were invented by Japanese manufacturers and fitted into a wide array of products during the 1970s to 1980s. This advancement mainly came about because of cooperative efforts by Japanese manufacturing companies. In 1951, they established the Barium Titanate Application Research Committee, which allowed the companies to be "competitively cooperative" and bring about several piezoelectric innovations and inventions.

4. **Bluetooth Module:**

SIM900 GSM Module – This means the module supports communication in 900MIIz band. We are from India and most of the mobile network providers in this country operate in the 900MHz band. If you are from another country, you have to check the mobile network band in your area. A majority of United States mobile networks operate in 850MHz band (the band is either 850MHz or 1900MHz). Canada operates primarily on 1900 MHz band.

LITERATURE SURVEY:

Sr. No.	Paper Title	Author Name	Publication Year	Result
1	Internet of Things (IOT) Based Gas Leakage Monitoring and Alerting System with MQ-2 Sensor	Rohan Chandra Pandey, Manish Verma, Lumesh Kumar Sahu	2017	This paper choice of using a real time gas leakage monitoring and Sensing the output levels of gas has been clearly observed by the help of this system.
2	Gas Leakage Detection and Smart Alerting and Prediction Using IoT	Asmita Varma, Prabhakar S, Kayalvizhi Jayavel	2017	The proposed gas leakage detector is promising in the Field of safety.
3	IOT Based Gas Leakage Detection System with Database Logging, Prediction and Smart Alerting	Chaitali Bagwe, Vidya Ghadi, Vinayshri Naik, Neha Kunte	2018	The system provides constant monitoring and detection of gas leakage along with storage of data in database for predictions and analysis. The IOT components used helps in making the system much more cost effective in comparison with traditional Gas detector systems.

4	Internet of Things (IoT) Based Gas Leakage Monitoring and Alerting System with Mq-6 Sensor	Rohan Chandra Pandey, Manish Verma, Lumesh Kumar Sahu, Saurabh Deshmukh	2018	A discussion on how the aims and objectives are met is presented. An overall conclusion IOT based toxic gas detector is it has become more efficient, more applicable to today's applications and smarter.
5	Gas Leakage Detection and Smart Alerting System Using IoT	Shital Imade, Priyanka Rajmanes, Aishwarya Gavali	2018	In this paper we use IOT technology for enhancing the existing safety standards. While making this prototype has been to bring a revolution in the field of safety against the leakage of harmful and toxic gases

CONCLUSION

The advantage of this simple gas leak detector is its simplicity and its ability to warn about the leakage of the LPG gas . This system uses GSM technique to send alert massage to respective person if no one is there in the house and then gas leaks occurs, GSM module is there to send immediate messages to the respective person regarding the gas leak . The main advantage of this system is that it off the regulator knob of the cylinder automatically when gas leakage detected.

REFERENCES

[1]. Mr. Sameer Jagtap, Prajkta Bhosale, Priyanka Zanzane, Jyoti Ghogare, "LPG Gas Weight and Leakage Detection System Using IoT", International Journal for Research in Applied Science & Engineering Technology", Volume 4, Issue 3, March 2016, Pg – 716 to 720.

[2]. Arun Raj, Athira Viswanathan, Athul T S, "LPG Gas Monitoring System", International Journal of Innovative Technology and Research, Volume 3, Issue 2, February 2015, Pg – 1957 to 1960.

[3]. S Shyamaladevi, V. G. Rajaramya, P. Rajasekar, P. Sebastin Ashok, "ARM7 based automated high-performance system for lpg refill booking & leakage detection", Journal of VLSI Design and Signal Processing", Volume 3, Issue 2, 2014.

[4]. S. Sharma, V. N. Mishra, R. Dwivedi, R. Das, "Classification of gases/odours using Dynamic Response of Thick Film Gas Sensor Array", IEEE Conference on Sensors Journal, 2013.

[5]. Rajeev B. Ahuja, Jayant K. Dash, Prabhat Shrivastava, "A comparative analysis of liquefied petroleum gas (LPG) and kerosene related burns", Burns, Volume 37, Issue 8, December 2011, Pg – 1403 to 1410.

[6]. Prof. Pankaj C. Warule, Shivam Upadhyay, Snehal S. Shelke, Sumitra K. Khandade, "LPG Detection, Metering and Control System Using Microcontroller", IJARIIE, Volume 2, Issue 2, 2016, Pg – 648 to 652.

[7]. Ankit Sood, Babalu Sonkar, Atul Ranjan, Mr. Ameer Faisal, "Microcontroller Based LPG Gas Leakage Detector Using GSM Module", International Journal of Electrical and Electronics Research, Volume 3, Issue2, AprilJune 2015, Pg – 264 to 269.

[8]. Ashish Shrivastava, Ratnesh Prabhakar, Rajeev Kumar, Rahul Verma, "GSM Based Gas Leakage Detection System", International Journal of Technical Research and Applications", Volume 1, Issue2, May- June 2013, Pg – 42 to 45.

[9]. Shivalingesh B. M, Ramesh C, Mahesh S. R, Pooja R, Preethi K. Mane, Kumuda S, "LPG Detection, Measurement and Booking System", IJRSI, Volume 1, Issue 4, November 2014, Pg – 7 to 10.

[10]. C. Selvapriya, S. Prabha Sathya, M. Abdulrahim, C. K. Aarthi, "LPG Leakage Monitoring and Multilevel Alerting System", International Journal of Engineering Sciences & Research Technology, Volume 2, Issue 11, November 2013, Pg – 3287 to 3290.

[11]. H. Huang, H. Bainand, S. Zhu, "A Greenhouse Remote Monitoring System Based on GSM", in Proceedings of IEEE International Conference on Information Management, 2011, Pg – 357 to 360.

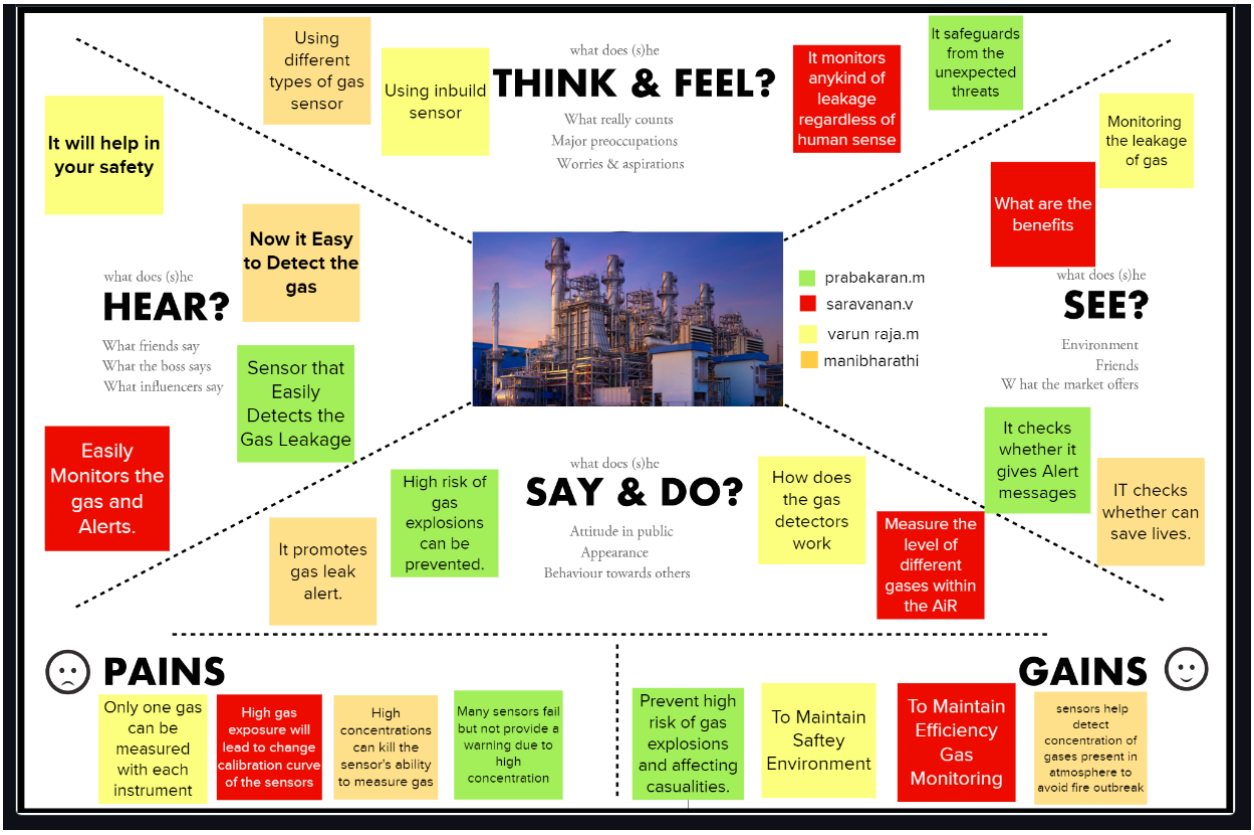
[12]. Lianos, M. and Douglas, M. (2000) Dangerization and the End of Deviance: The Institutional Environment. British Journal of Criminology, 40, 261-278.

[13]. J. Tsado, O. Imoru, S.O. Olayemi, "Design and construction of a GSM based gas leak Alert system", IEEE Transaction, IRJEEE Vol. 1(1), pp. 002-006, September, 2014.

[14]. D. Surie, O. Laguionie, T. Pederson, "Wireless sensor networking of everyday objects in a smart home environment||, Proceedings of the International Conference on Intelligent Sensors", Sensor Networks and Information Processing- ISSNIP- 2008, pp. 189 – 194

3. IDEATION & PROPOSED SOLUTION

a. Empathy Map Canvas

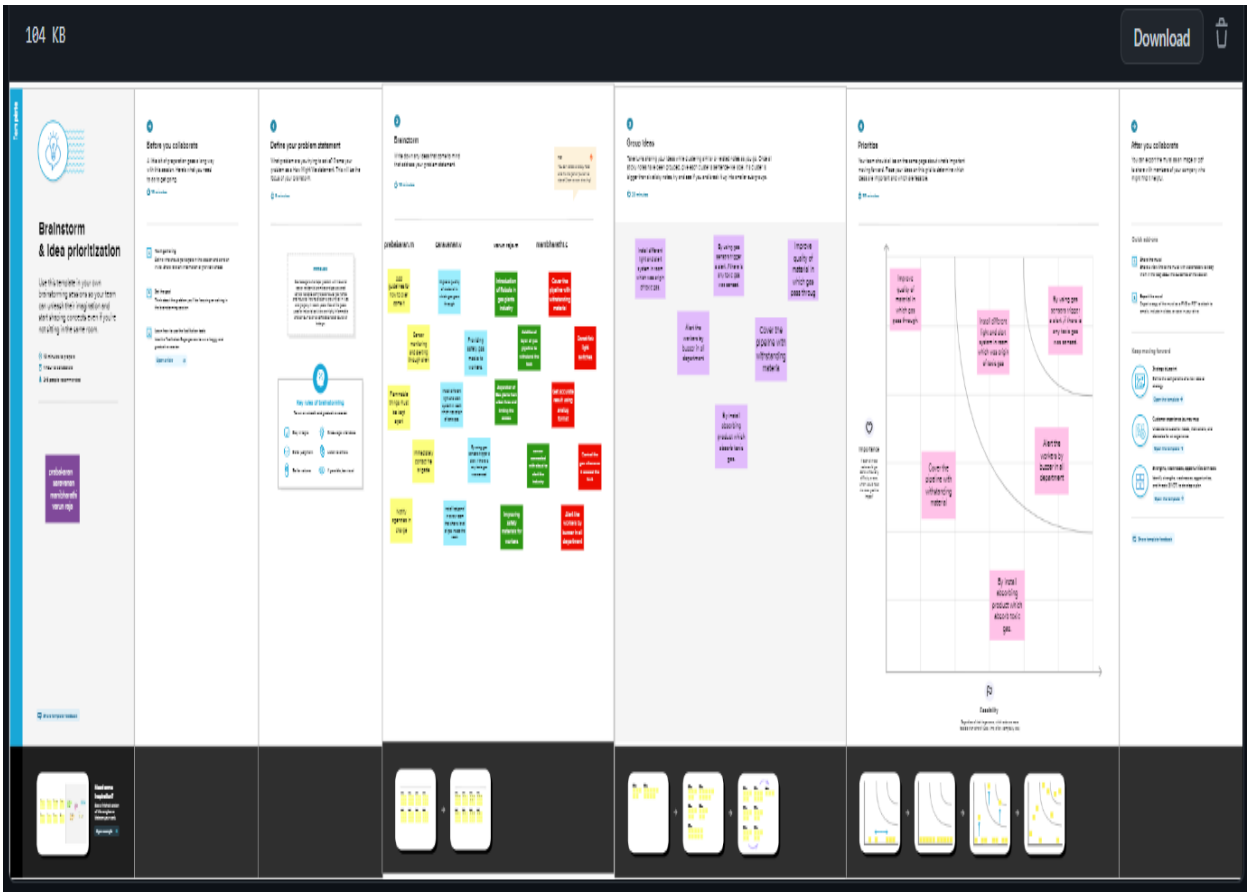


An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.

Empathy maps can be used whenever you find a need to immerse yourself in a user's environment. They can be helpful, for example, when:

- diving into the customer segments of a business model canvas
- elaborating on user personas
- capturing behaviors when interviewing a customer
- building out the "user" in a user story

b. Ideation & Brainstorming



Brainstorming combines an informal approach to problem-solving with lateral thinking, which is a method for developing new concepts to solve problems by looking at them in innovative ways. Some of these ideas can be built into original, creative solutions to a problem, while others can generate additional ideas.

Some experts believe that brainstorming is better than conventional group interaction, which might be hindered by group think. Group think is a phenomenon that occurs when the team's need for consensus overshadows the judgment of individual group members

Although group brainstorming is frequently better for generating ideas than normal group

problem-solving, several studies have shown that individual brainstorming can produce better ideas than group brainstorming. This can occur because group members pay so much attention to others' ideas that they forget or do not create their own ideas. Also, groups do not always adhere to good brainstorming practices.

During brainstorming sessions, participants should avoid criticizing or rewarding ideas in order to explore new possibilities and break down incorrect answers. Once the brainstorming session is over, the evaluation session (which includes analysis and discussion of the aired ideas) begins, and solutions can be crafted using conventional means.

C. Proposed solution:

Proposed-solution fit is a term used to describe the point validating that the base problem resulting in a business idea really exists and the proposed solution actually solves that problem.

The proposed-solution fit is when you-

- **Validate that the problem exists:** When you validate your problem hypothesis using real-world data and feedback. That is, you gather information from real users to determine whether or not they care about the pain point you're trying to solve.
- **Validate that your solution solves the problem:** When you validate that the target audience appreciates the value your solution delivers to them.

Project Design Phase-I
Proposed Solution

Date	11 November 2022
Team ID	PNT2022TMID22149
Project Name	Gas leakage monitoring and alerting system for industries
Maximum Marks	2 Marks

Proposed Solution:

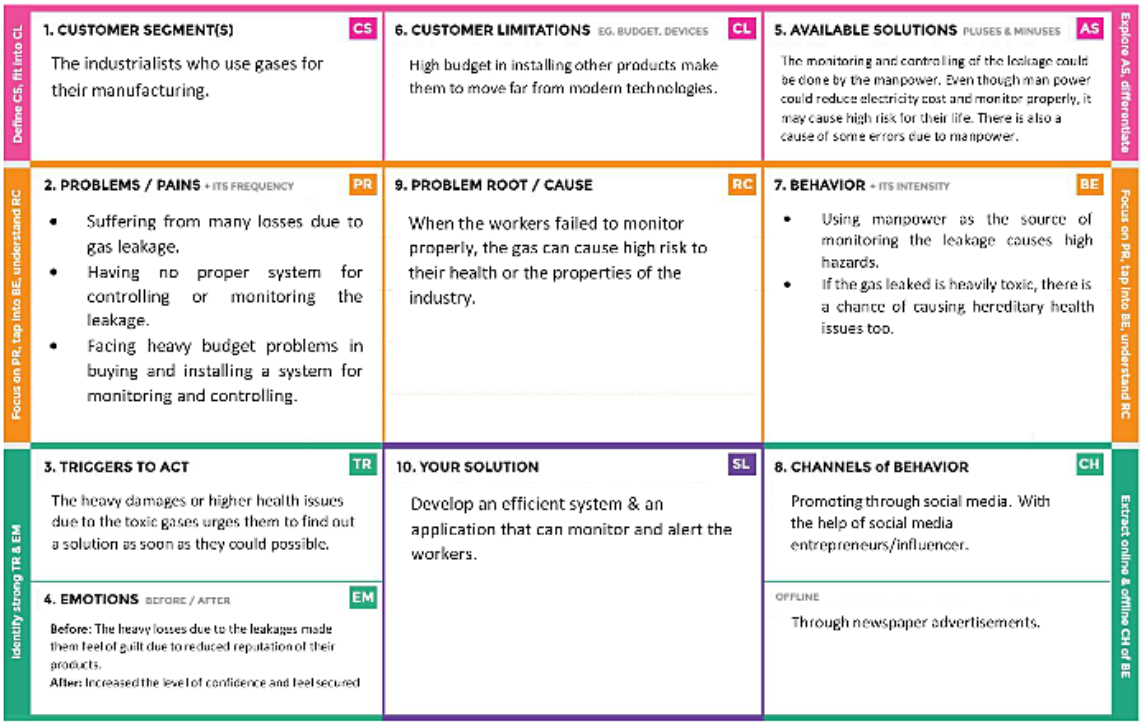
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	➢ Develop an efficient system & an application that can monitor and alert the users(workers)
2.	Idea / Solution description	➢ This product helps the industries in monitoring the emission of harmful gases ➢ In several areas, the gas sensors will be integrated to monitor the gas leakage ➢ If in any area gas leakage is detected the admins will be notified along with the location ➢ In the web application, admins can view the sensor parameters.
3.	Novelty / Uniqueness	➢ Fastest alerts to the workers ➢ User friendly
4.	Social Impact / Customer Satisfaction	➢ Cost efficient ➢ Easy installation and provide efficient results ➢ Can work with irrespective of fear
5.	Business Model (Revenue Model)	➢ The product is advertised all over the platforms. Since it is economical, even helps small scale industries from disasters. ➢ As the product usage can be understood by everyone, it is easy for them to use it properly for their safest organization
6.	Scalability of the Solution	➢ Since the product is cost efficient, it can be placed in many places in the industries. ➢ Even when the gas leakage is more, the product sense the accurate values and alerts the workers effectively

d. Problem Solution Fit

Project Design Phase-I

Problem Solution Fit

Date	11 November 2022
Team ID	PNT2022TMID22149
Project Name	Gas leakage monitoring and alerting system for industries



Problem-solution fit is a term used to describe the point validating that the base problem resulting in a business idea really exists and the proposed solution actually solves that problem.

The problem-solution fit is when you-

- **Validate that the problem exists:** When you validate your problem hypothesis using real-world data and feedback.

4. REQUIREMENT ANALYSIS

a. Functional Requirements

Project Design Phase-II

Functional Requirements

DATE	14 November 2022
TEAM-ID	PNT2022TMID22149
PROJECT NAME	Gas Leakage Monitoring and Alerting System for Industries
MAXIMUM MARKS	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Monitoring	Level of gas is monitored using sensor and if there is any leakage, alert can be sent through messages and with a buzzer sound.
FR-2	User Reception	The data like the level of gas can be send through messages
FR-3	User Understanding	The user can monitor the level of gas with the help of the data. If there is an increase in gas level then the alert will be given by message or buzzer sound.
FR-4	User Performance	When the user gets notified, they could take precaution steps like turning the gas off, turn on the exhaust fan/sprinkler and avoid serious accidents.

b. Non-Functional Requirements

Non-functional Requirements:

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

FR No.	Non-Functional	Requirement Description
FR-1	Usability	It updates the data regularly as well as protects the workers.
FR-2	Security	As a result of emergency alert, we can be able to protect both the humans and properties. Precaution steps could be taken.
FR-3	Reliability	Can be able to provide accurate values. It might have a capacity to recognize the smoke accurately and does not give a false
FR-4	Performance	Sprinklers and exhaust fans are used in case of emergency
FR-5	Availability	It can be used for everyday; it includes day and nights.
FR-6	Scalability	Sensors can be replaced every time it fails

5. PROJECT DESIGN

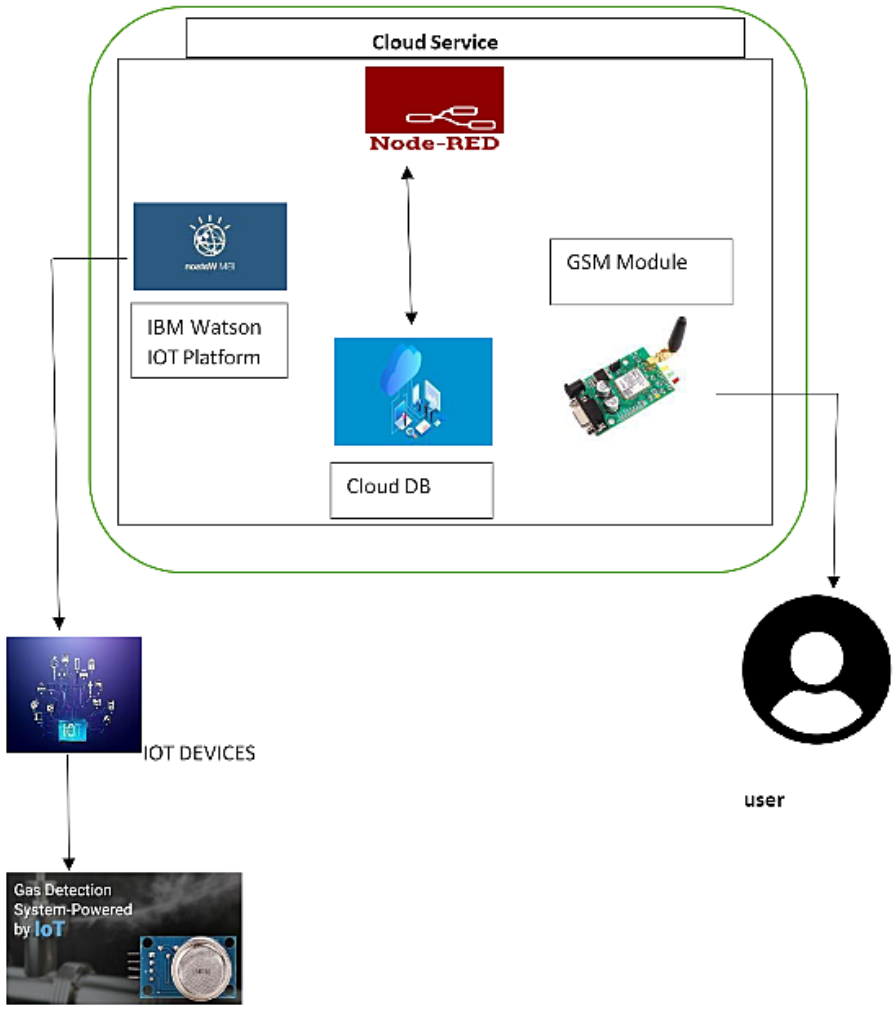
a. Data Flow Diagrams

Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	14 November 2022
Team ID	PNT2022TMID22149
Project Name	Project – Gas Leakage Monitoring and Alerting System
Maximum Marks	4 Marks

Technical architecture:

Figure: Gas Leakage Monitoring and Alerting System



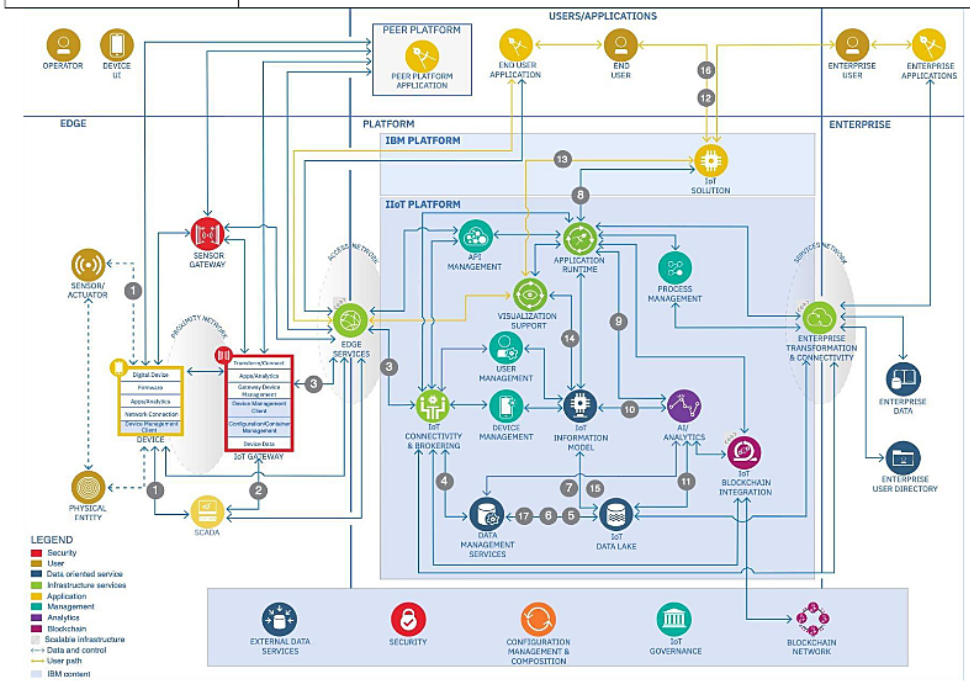
Dataflow is the movement of data through a system comprised of software, hardware or a combination of both.

Dataflow is often defined using a model or diagram in which the entire process of data movement is mapped as it passes from one component to the next within a program or a system, taking into consideration how it changes form during the process.

b. Solution And Technical Architecture

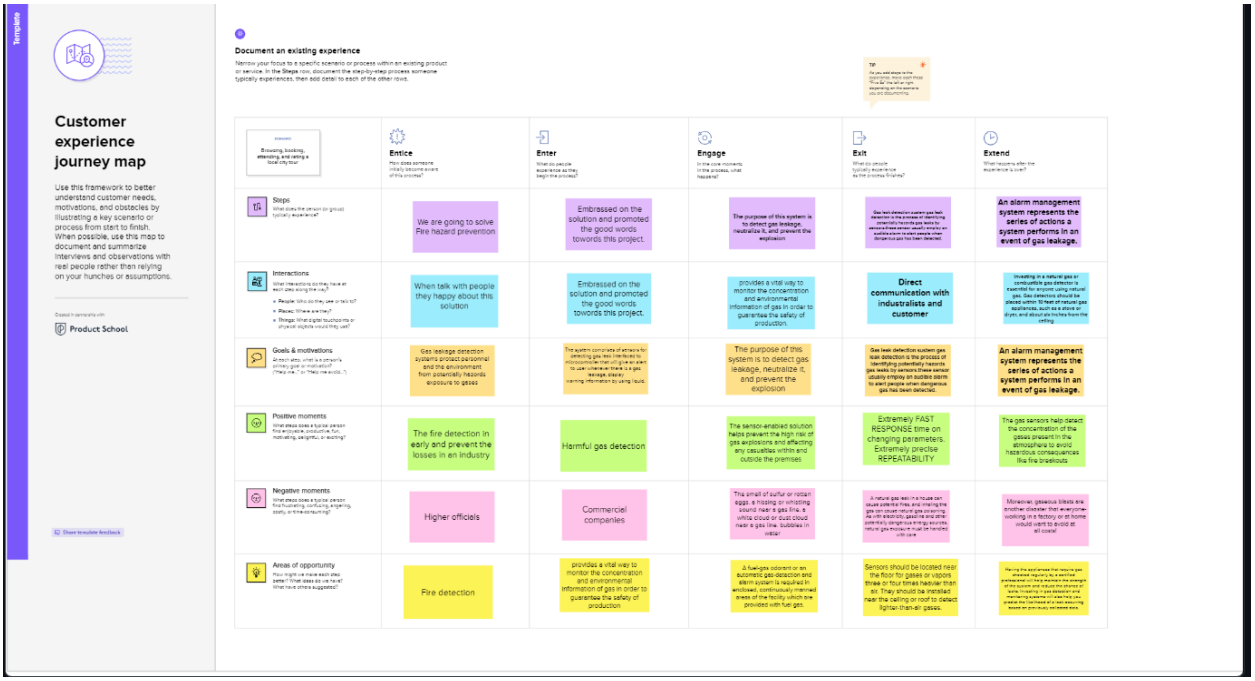
Project Design Phase-I
Solution Architecture

Date	11 November 2022
Team ID	PNT2022TMID22149
Project Name	Gas leakage monitoring and alerting system for industries



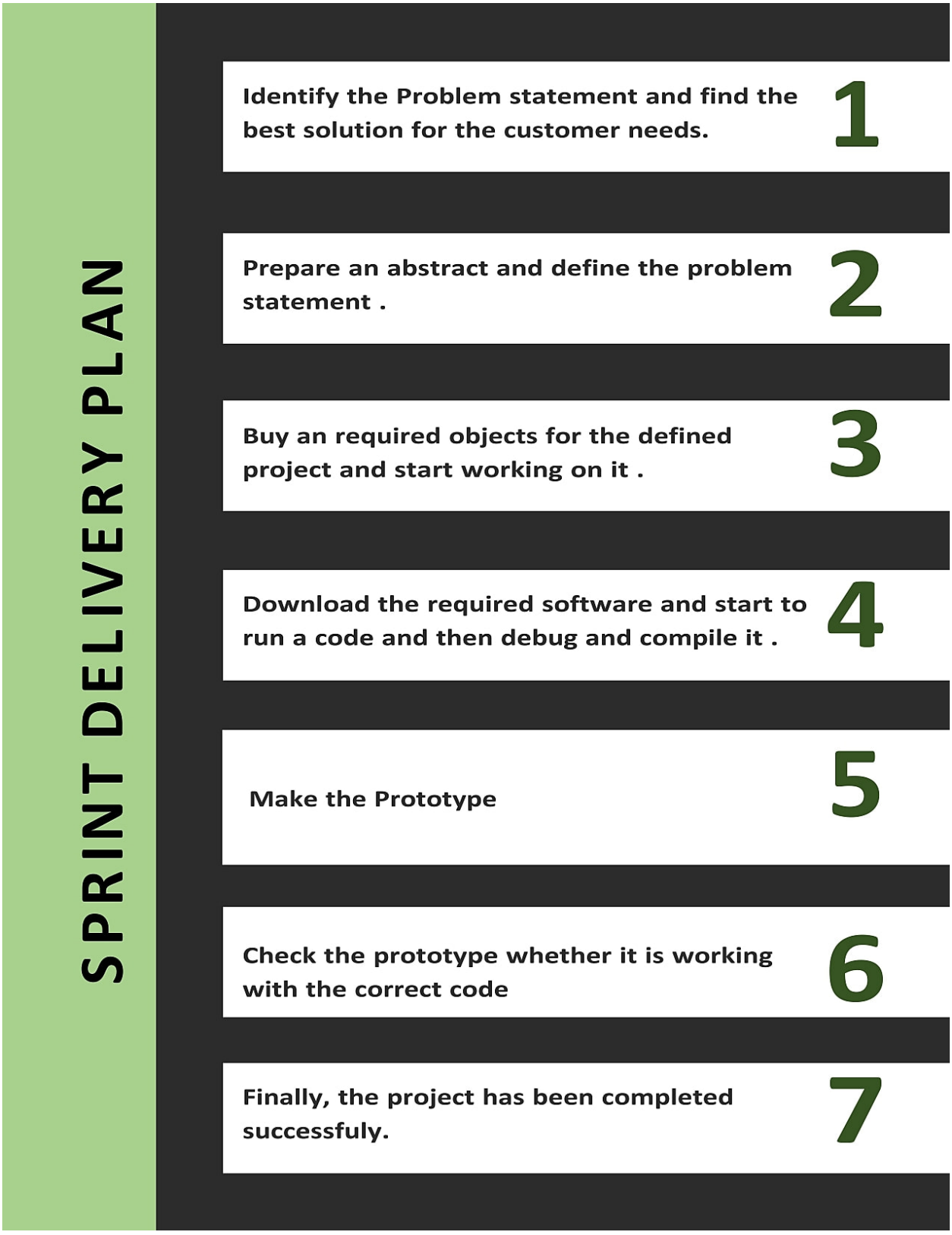
Solution architecture is the process of developing solutions based on predefined processes, guidelines and best practices with the objective that the developed solution fits within the enterprise architecture in terms of information architecture, system portfolios, integration requirements and many more.

c. User Stories



6. PROJECT PLANNING AND SCHEDULING

a. Sprint Planning & Estimation



b. Sprint Delivery Schedule

Sprint – 1

Team ID: PNT2022TMID22149

Python Code:

```
# Sprint - 1
# Team ID: PNT2022TMID22149

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "lcft5g"
deviceType = "Final"
deviceId = "Hello"
authMethod = "token"
authToken = "8300113450"
```

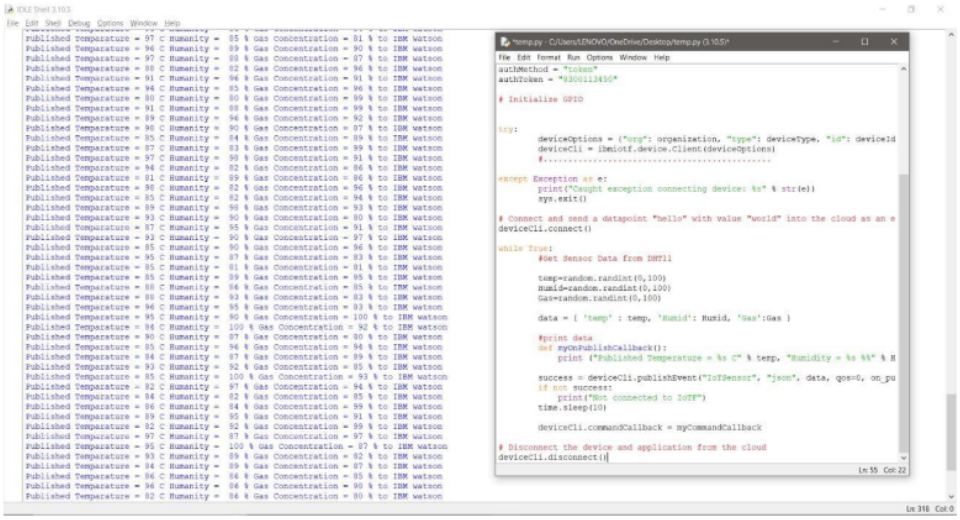
```
try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....


except Exception as e:
```

```
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an
event of type "greeting" 10 times
deviceCli.connect()


while True:
    #Get Sensor Data from DHT11
    temp=random.randint(0,100)
```

Output:


IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Published Temperature = 97 C Humanity = 85 % Gas Concentration = 81 % to IBM watson
Published Temperature = 96 C Humanity = 89 % Gas Concentration = 90 % to IBM watson
Published Temperature = 97 C Humanity = 88 % Gas Concentration = 87 % to IBM watson
Published Temperature = 88 C Humanity = 82 % Gas Concentration = 96 % to IBM watson
Published Temperature = 91 C Humanity = 96 % Gas Concentration = 91 % to IBM watson
Published Temperature = 94 C Humanity = 85 % Gas Concentration = 96 % to IBM watson
Published Temperature = 90 C Humanity = 80 % Gas Concentration = 99 % to IBM watson
Published Temperature = 91 C Humanity = 88 % Gas Concentration = 99 % to IBM watson
Published Temperature = 89 C Humanity = 96 % Gas Concentration = 92 % to IBM watson
Published Temperature = 90 C Humanity = 90 % Gas Concentration = 87 % to IBM watson
Published Temperature = 85 C Humanity = 84 % Gas Concentration = 89 % to IBM watson
Published Temperature = 87 C Humanity = 83 % Gas Concentration = 99 % to IBM watson
Published Temperature = 97 C Humanity = 90 % Gas Concentration = 91 % to IBM watson
Published Temperature = 94 C Humanity = 82 % Gas Concentration = 86 % to IBM watson
Published Temperature = 81 C Humanity = 89 % Gas Concentration = 96 % to IBM watson
Published Temperature = 98 C Humanity = 82 % Gas Concentration = 96 % to IBM watson
Published Temperature = 85 C Humanity = 82 % Gas Concentration = 94 % to IBM watson
Published Temperature = 89 C Humanity = 98 % Gas Concentration = 93 % to IBM watson
Published Temperature = 93 C Humanity = 90 % Gas Concentration = 80 % to IBM watson
Published Temperature = 87 C Humanity = 95 % Gas Concentration = 91 % to IBM watson
Published Temperature = 92 C Humanity = 90 % Gas Concentration = 97 % to IBM watson
Published Temperature = 85 C Humanity = 90 % Gas Concentration = 96 % to IBM watson
Published Temperature = 95 C Humanity = 87 % Gas Concentration = 83 % to IBM watson
Published Temperature = 85 C Humanity = 81 % Gas Concentration = 81 % to IBM watson
Published Temperature = 85 C Humanity = 89 % Gas Concentration = 95 % to IBM watson
Published Temperature = 88 C Humanity = 86 % Gas Concentration = 85 % to IBM watson
Published Temperature = 98 C Humanity = 93 % Gas Concentration = 83 % to IBM watson
Published Temperature = 96 C Humanity = 95 % Gas Concentration = 83 % to IBM watson
Published Temperature = 95 C Humanity = 90 % Gas Concentration = 100 % to IBM watson
Published Temperature = 84 C Humanity = 100 % Gas Concentration = 92 % to IBM watson
Published Temperature = 90 C Humanity = 87 % Gas Concentration = 80 % to IBM watson
Published Temperature = 85 C Humanity = 96 % Gas Concentration = 94 % to IBM watson
Published Temperature = 84 C Humanity = 87 % Gas Concentration = 89 % to IBM watson
Published Temperature = 93 C Humanity = 92 % Gas Concentration = 85 % to IBM watson
Published Temperature = 85 C Humanity = 100 % Gas Concentration = 93 % to IBM watson
Published Temperature = 82 C Humanity = 97 % Gas Concentration = 94 % to IBM watson
Published Temperature = 84 C Humanity = 82 % Gas Concentration = 85 % to IBM watson
Published Temperature = 86 C Humanity = 84 % Gas Concentration = 99 % to IBM watson
Published Temperature = 89 C Humanity = 95 % Gas Concentration = 91 % to IBM watson
Published Temperature = 82 C Humanity = 92 % Gas Concentration = 99 % to IBM watson
Published Temperature = 97 C Humanity = 87 % Gas Concentration = 97 % to IBM watson
Published Temperature = 95 C Humanity = 100 % Gas Concentration = 87 % to IBM watson
Published Temperature = 93 C Humanity = 89 % Gas Concentration = 82 % to IBM watson
Published Temperature = 84 C Humanity = 89 % Gas Concentration = 87 % to IBM watson
Published Temperature = 86 C Humanity = 86 % Gas Concentration = 85 % to IBM watson
Published Temperature = 96 C Humanity = 86 % Gas Concentration = 90 % to IBM watson
Published Temperature = 82 C Humanity = 86 % Gas Concentration = 80 % to IBM watson
Ln: 316 Col: 0
temp.py - C:/Users/LENOVO/OneDrive/Desktop/temp.py (3.10.5)
File Edit Format Run Options Window Help
authMethod = "token"
authToken = "9300113456"
# Initialize GPIO
try:
deviceOptions = {"org": organization, "type": deviceType, "id": deviceId
deviceCli = ibmiotf.device.Client(deviceOptions)
#.....
except Exception as e:
print("Caught exception connecting device: %s" % str(e))
sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an e
deviceCli.connect()
while True:
#Get Sensor Data from DHT11
temp=random.randint(0,100)
Humid=random.randint(0,100)
Gas=random.randint(0,100)
data = { 'temp' : temp, 'Humid': Humid, 'Gas':Gas }
#print data
def myOnPublishCallback():
print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % H
success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_pu
if not success:
print("Not connected to IoTF")
time.sleep(10)
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
Ln: 55 Col: 22

Sprint – 2

Team ID: PNT2022TMID22149

Task:

Sensed data is brought to Node-RED and displayed in dashboard.

Steps:

1. IBM IoT node is used to gather sensor data.
    a. Necessary API key is provided to establish connection.
2. Using functions namely Temperature, Humidity and Gas the data is obtained independently and displayed in dashboard.
3. Dashboard Nodes are used to display the sensed data to the user in a portal.

Sour code:

Temperature:

```
msg.payload = msg.payload.Temp;
return msg;
```

Humidity:

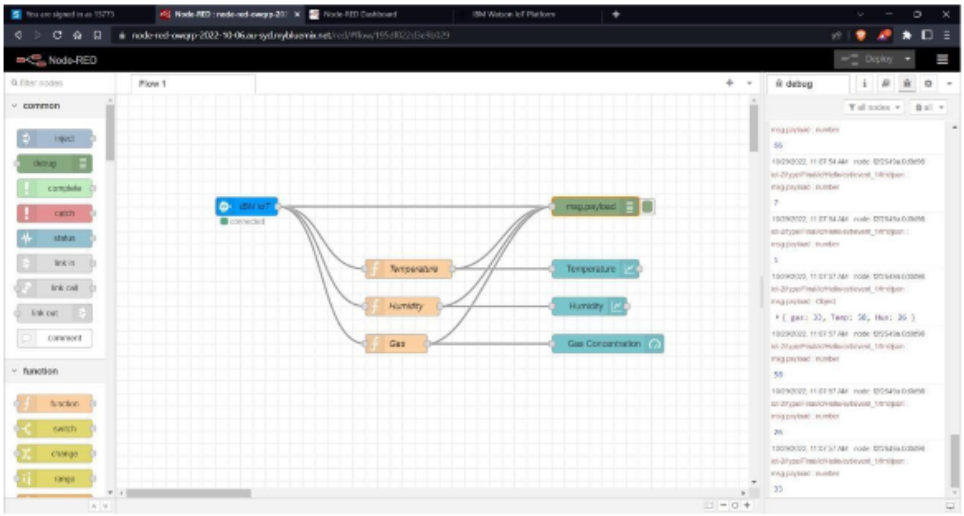
```
msg.payload = msg.payload.Hum;
return msg;
```

Concentration of Gas:

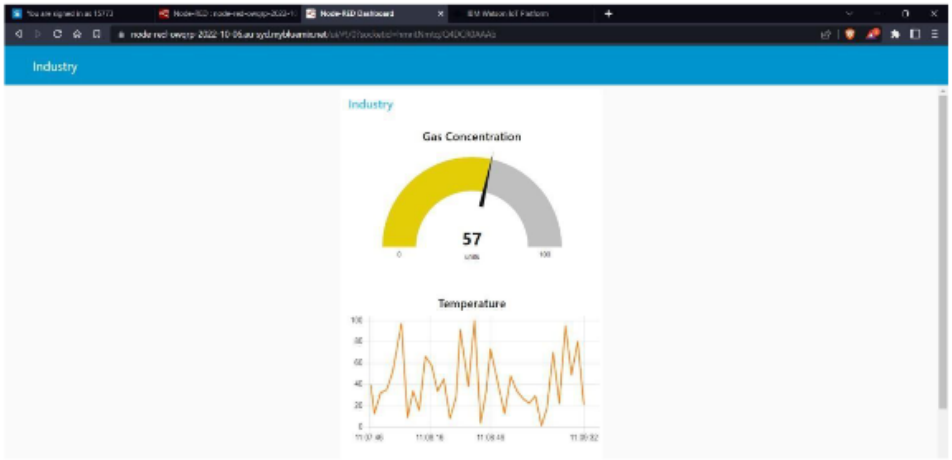
```
msg.payload = msg.payload.gas;
return msg;
```

Output:

1. Data is brought to Node-RED



2. Data is displayed in Dashboard



SPRINT 3

TEAM ID : PNT2022TMID22149

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(6, 7, 8, 9, 10, 11);
float gasPin = A0;
float gasLevel;
int ledPin = 2;
int buttonPin = 3;
int buzzPin = 4;
int buttonState;
int fan = 5;

void setup(){
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
  pinMode(gasPin,INPUT);
  pinMode(fan,OUTPUT);
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print(" Welcome");
  lcd.setCursor(0,2);
  lcd.print(" Youtube");
  delay(500);
  lcd.clear();
}

void loop(){
  // Read the value from gas sensor and button
  gasLevel = analogRead(gasPin);
  buttonState = digitalRead(buttonPin);

  // call the function for gas detection and button work
  gasDetected(gasLevel);
  buzzer(gasLevel);
  exhaustFanOn(buttonState);
}

// Gas Leakage Detection & Automatic Alarm and Fan ON
void gasDetected(float gasLevel){
  if(gasLevel >= 300){
   digitalWrite(buzzPin,HIGH);
    digitalWrite(ledPin,HIGH);
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN ON");
   delay(1000);
   lcd.clear();
```

```
  }
Else
{
   digitalWrite(ledPin,LOW);
    digitalWrite(buzzPin,LOW);
    digitalWrite(fan,LOW);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN OFF");
   delay(1000);
   lcd.clear();
  }
}
//BUZZER
void buzzer(float gasLevel){
if(gasLevel>=300)
  {
  for(int i 0; i< 30; i i+10)
  {
  tone(4,i);
  delay(400);
  noTone(4);
  delay(400);
  }
  }
}
// Manually Exhaust FAN ON
void exhaustFanOn(int buttonState){
  if(buttonState == HIGH){
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("Button State:");
    lcd.print(buttonState);
    lcd.setCursor(0,2);
    lcd.print("FAN ON");
    delay(10000);
    lcd.clear();
  }
}
```

SPRINT 4

TEAM ID : PNT2022TMID22149

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(5,6,8,9,10,11);

int redled = 2; int
greenled = 3; int
buzzer = 4; int
sensor = A0;
int sensorThresh = 400;

void setup()
{
pinMode(redled, OUTPUT);
pinMode(greenled,OUTPUT);
pinMode(buzzer,OUTPUT);
pinMode(sensor,INPUT);
Serial.begin(9600); lcd.begin(16,2);
}

void loop()
{
  int analogValue = analogRead(sensor);
  Serial.print(analogValue);
  if(analogValue>sensorThresh)
  {
    digitalWrite(redled,HIGH);
    digitalWrite(greenled,LOW);
    tone(buzzer,1000,10000);
    lcd.clear(); lcd.setCursor(0,1);
    lcd.print("ALERT"); delay(1000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("EVACUATE");
    delay(1000);
  }
  else
  {
    digitalWrite(greenled,HIGH);
```

```
    digitalWrite(redled,LOW);
    noTone(buzzer); lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("SAFE"); delay(1000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("ALL CLEAR");
    delay(1000);
  }

}
```

7. CODING AND SOLUTIONING

Feature 1

Project Name	Gas leakage and monitoring system
Team ID	PNT2022TMID22149
Date	15.11.2022

PYTHON CODE:

```
import random
print('Random number=',str(random.randint(0,100)))
print('Temperature=',str(random.randint(0,100)))
print('Humidity=',str(random.randint(0,100)))
```

b. Feature 2

Project Name	Gas leakage and monitoring system
Team ID	PNT2022TMID22149
Date	15.11.2022

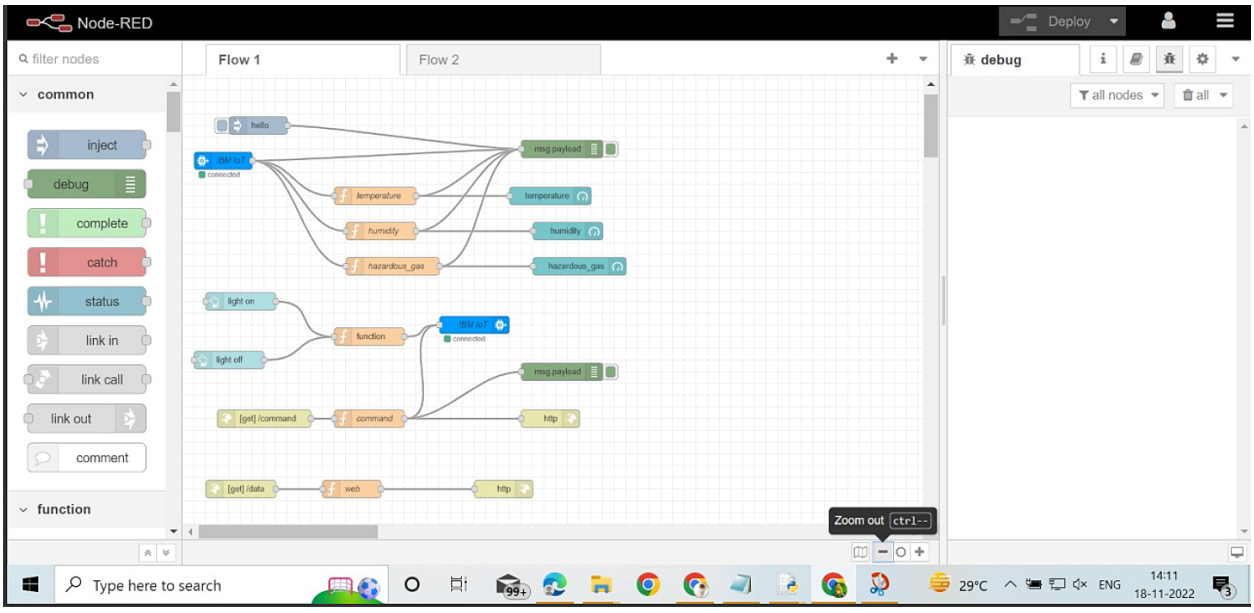
PYTHON CODE TO PUBLISH DATA TO IBM CLOUD

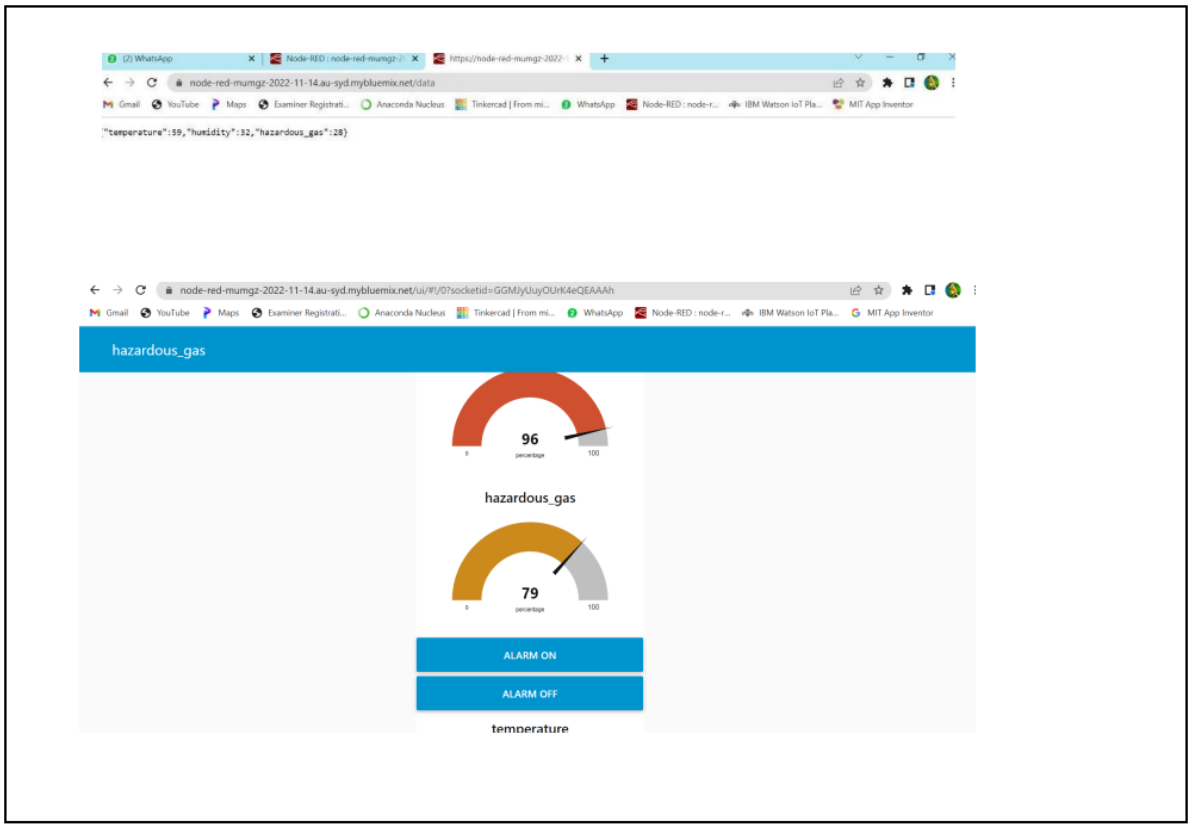
CODE:

```
import wiotp.sdk.device
import time
import random myConfig {
"identity": {
"orgId": "n9y27r",
"typeId": "Jambu ",
"deviceId":"0502"
},
"auth":{
   "token": "123456789"
}
}
def myCommandCallback(cmd):
print("Message received from IBM IoT Platform: %s"
%cmd.data['command']) m=cmd.data['command'] client
=wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect()
while True:
```

```
gas=random.randint(0,100)
temp=random.randint(0,100)
hum=random.randint(0,100)
pre=random.randint(0,100)
myData={'Hazardous Gas':gas,'Temperature':temp,
'Humidity':hum,'Pressure':pre } client.publishEvent(eventId="status",
msgFormat="json",data=myData, qos=0, onPublish=None)
print("Published data Successfully: %s",myData)
client.commandCallback = myCommandCallback time.sleep(2)
client.disconnect()
```

Result





APPENDIX

Github Link:https://github.com/IBM-EPBL/IBM-Project-3312-1658545323

Demo Video Link:

TEAM ID: PNT2022TMID22149
TEAM LEADER: PRABAKARAN.M
TEAM MEMBERS: SARAVANAN.V
MANIBHARATHI.V
VARUN RAJA.M