Gas leakage monitoring and alerting system for industries
IBM-PROJECT-24440-1662610314

NALAIYA THIRAN PROJECT BASED LEARNING ON PROFESSIONAL READINESS FOR INNOVATION,EMPLOYMENT AND ENTERPRENEURSHIP

A PROJECT REPORT BY
Giridharan p(310619205030)
Akshayram RB(310619205007)
Ethiraj M(310619205026)
Karthikeyan M(310619205045)

MENTOR:ASSISTANT PROFESSOR Dr JOHNY ELMA K

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY
EASWARI ENGINEERING COLLEGE RAMAPURAM-600089

INDEX

1.INTRODUCTION

sucessfully in reducing the amount of gas that was wasted.

1.1.Project Overview:

The internet of Things is a developing topic of technical, social, and economic significance. The usage of the gas brings great problems in the domestic as well as working places. The inflammable gas, which is excessively used in the work places (Industries). The leakage of the gas causes destructible impact to the lives and as well as to the heritage of the people. Most of the societies have fire safety mechanism. But it can use after the fire exists. As a result, a system for detecting and monitoring gas

leaks is required. Through a flame sensor, the system will sense fire and flame. The buzzer begins to ring when a fire is detected. Tests have shown that the system can keep track of the wastage of gas and leaks and notify the user. The performance that was produced showed that it was success

1.2.Purpose:

The design of a sensor-based automatic gas leakage detector with an alert and control system has been proposed. This is an affordable, less power using, lightweight, portable, safe, user friendly, efficient, multi featured and simple system device for detecting gas. To monitor this gas leak, the system includes an MQ6 gas detector. This sensor detects the amount of leaking gas present in the surrounding atmosphere. In this way, the consequences of an explosion or gas leak can be avoided.

2.LITERATURE SURVEY :

2.1 Existing Problem:

Gas leakage is nothing but the leak of any gaseous molecule from a pipeline, or cylinder etc in the industries. Gas Leakages in open or closed areas can prove to be dangerous .This can occur either purposefully or even unintendedly. As we are aware that these kinds of leaks are dangerous to our health, and when it becomes explosive it could cause great danger to the people, industry and the environment. Therefore, we have used IoT technology to make a Gas Leakage Detector for society which has Smart Alerting techniques involving sending a text message to the concerned authority and the ability to perform data analytics on sensor readings. Our main aim is to propose a gas leakage system for a society where each flat has gas leakage detector hardware. This will detect the harmful gases in the environment and alerting to society members through the alarm and sending notifications.

2.2.References:

i. Shital Imade, Priyanka Rajmanes, Aishwarya Gavali , Prof. V. N. Nayakwadi “GAS LEAKAGE DETECTION AND SMART ALERTING SYSTEM USING IOT”
https://www.pramanaresearch.org/gallery/22.%20feb%20ijirs%20-%20d539.pdf

ii. Kumar Keshamoni and Sabbani Hemanth. "Smart Gas Level Monitoring, Booking & Gas Leakage Detector over IoT " International Advance Computing Conference IEEE, 2017.

iii. Petros Spachos , Liang Song and Dimitrios Hatzinakos. "Gas

Leak Detection and

Localization System Through Wireless Sensor Networks" The 11th Annual IEEE Consumer Communications and Networking Conference - Demos. IEEE, 2014.

iv. “Design and Implementation of an Economic Gas Leakage Detector” National

Institute of Health (2004). What you need to know about natural gas detectors. Available:http://www.nidcd.nih.gov/health/smelltaste/gas dtctr.asp.

v. Prof.M.Amsaveni, A.Anurupa, R.S.Anu Preetha, C.Malarvizhi,M.Gunasekaran

“Gsm based LPG leakage detection and controlling system” the International Journal of Engineering and Science (IJES) ISSN (e): 2319 – 1813 ISSN (p):2319 – 1805 Pages 112-116 March- 2015.

vi. Srinivasan,Leela,Jeyabharathi,Kirthika,Rajasree“GAS LEAKAGE DETECTION AND CONTROL” Scientific Journal of Impact Factor(SJIF): 3.134.

vii. Pal-Stefan Murvaya, IoanSileaa “A survey on gas leak detection and localization techniques”.

viii. Ch. Manohar Raju, N. Sushma Rani, “An android based automatic gas detection and indication robot. In International Journal of Computer Engineering and Applications. 2014;8(1).

ix. Falohun A.S., Oke A.O., Abolaji B.M. “Dangerous Gas Detection using an

Integrated Circuit and MQ-9” in International Journal of Computer Applications (0975 –8887) Volume 135 – No.7, February 2016.

x. Ashish Shrivastava,Ratnesh Prabhaker, Rajeev Kumar and Rahul Verma “GSM

BASED GAS LEAKAGE DETECTION SYSTEM” in International Journal of Technical Research and Applications e-ISSN: 2320- 8163,www.ijtra.com Volume 1, Issue 2 (may-June 2013).

xi. C.Selvapriya, S.Sathyaprabha, M.Abdulrahim,” LPG leakage monitoring and multilevel alerting system”, published in 2013.

xii. Falohun A.S., Oke A.O., Abolaji B.M. “Dangerous gas detection using an integrated circuit and MQ-9. In International Journal of Computer Applications. 2016; 135(7).

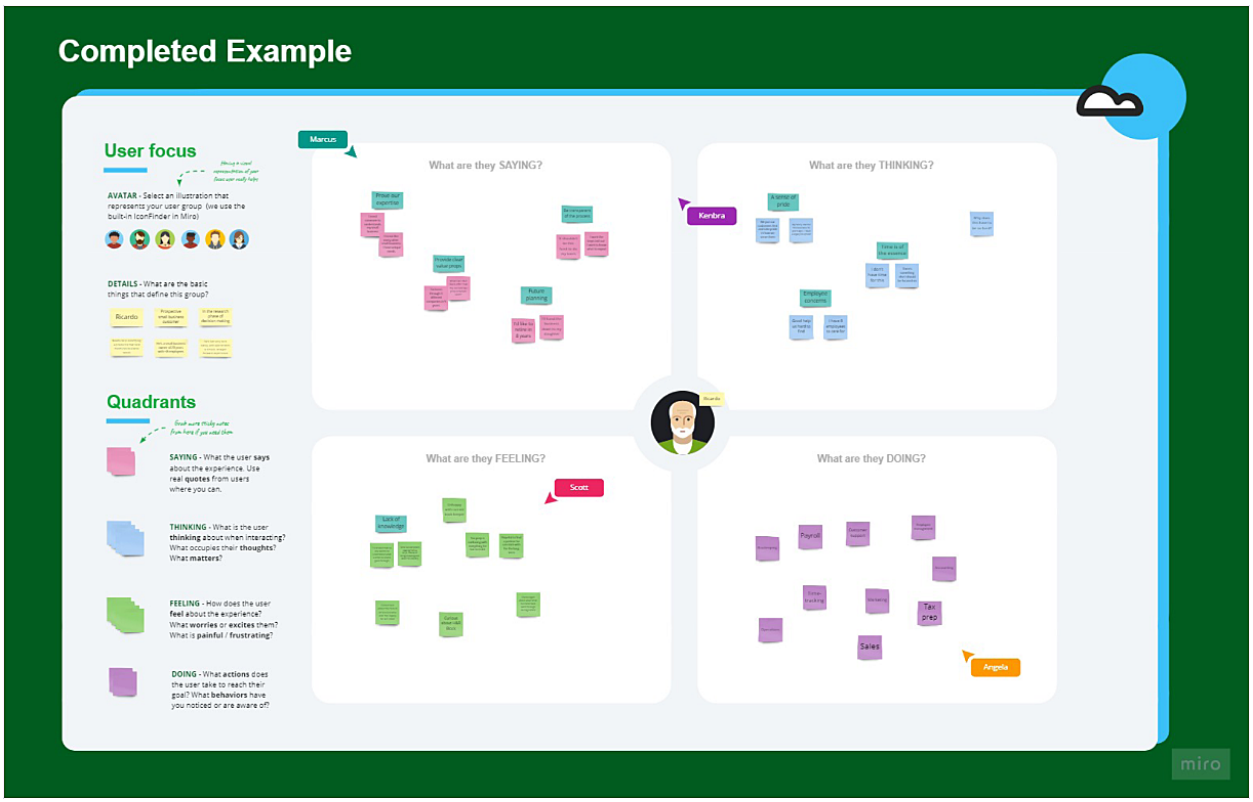
2.3.Problem Statement Definition In most industries, one of the key parts of any safety plan for reducing risks to personnel and plant is the use of early-warning devices such as gas detectors. These can help to provide more time in which to take remedial or protective action. They can also be used as part of a total, integrated monitoring and safety system for an industrial plant. Rapid expansion of oil and gas industry leads to gas leakage incidents which are very serious and dangerous. Solutions need to be found out at least to minimize the effects of these incidents since gas leaks also produce a significant financial loss. The challenges are not only to design a prototype of the device that can only detect but also automatically respond to it whenever the leakage occurs.

3.IDEATION & PROPOSED SOLUTION:

3.1.Empathy Map Canvas: What is an empathy map canvas?

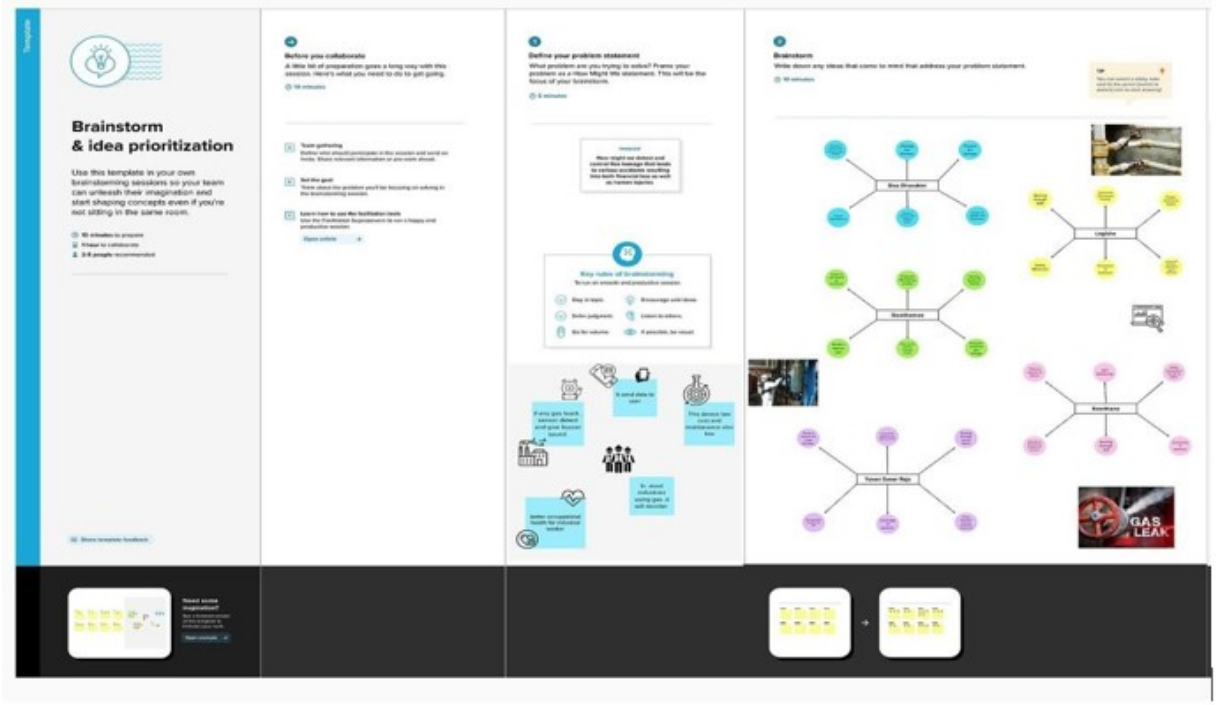
An empathy map is **a collaborative tool teams can use to gain a deeper insight into their customers**

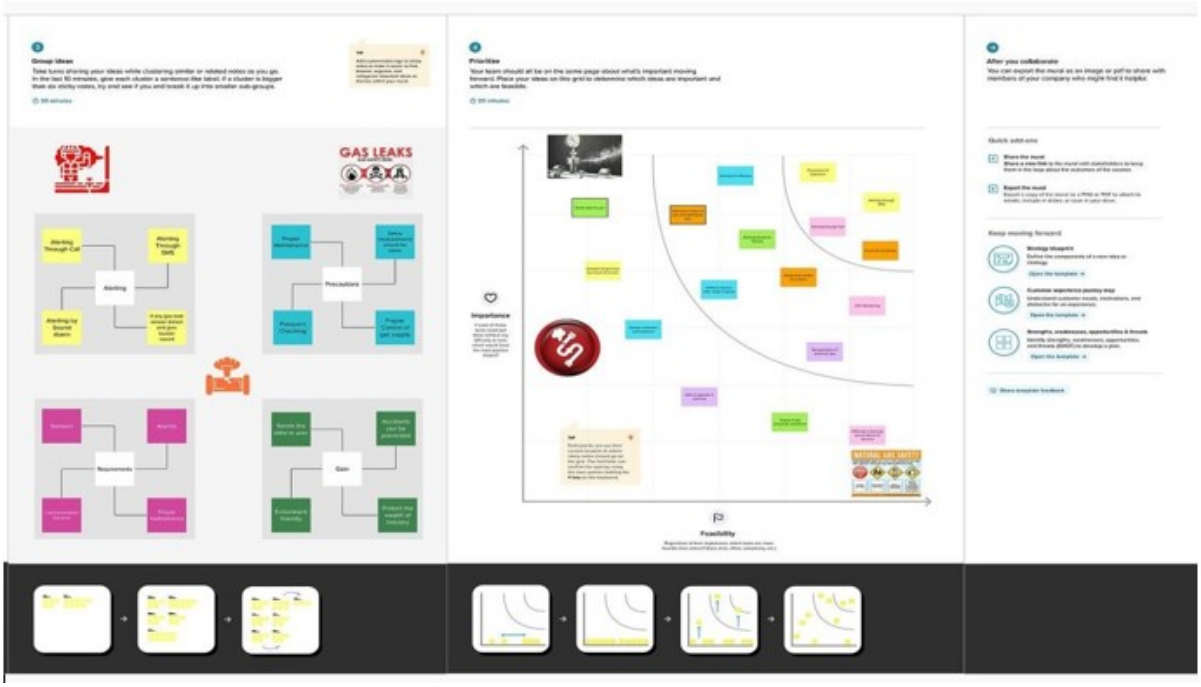
it consists of three different sections
.what are they thinking
.what are they feeling
.what are they doing



3.2.Ideation & Brainstorming: Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between ideation and brainstorming is that **ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity**

As you can see, ideation is not just a one-time idea generation or a brainstorming session. In fact, we can divide ideation in these three stages: **generation, selection, and development**.





3.3.Proposed Solution:

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Gas leakage leads to various accidents resulting in loss of human lives and industry properties. Sometimes, the gas leakage cannot be detected by human that has a low sense of smell. Thus, this system will help to detect the presence of gas leakage and alert the users.
2.	Idea / Solution description	It detects the gas leakage by using various sensors. If the gas leakage level is above the threshold level, it sends the alert message through SMS to the user by using GSM module and buzzer the alarm.
3.	Novelty / Uniqueness	We use location tagging and alert service so that the admin and fire department team will be notified the exact location. The system provides constant monitoring and detection of gas leakage along with storage of data in database for predictions and analysis.
4.	Social Impact / Customer Satisfaction	By implementing real-time gas leak detection, industries can monitor their environmental performance, ensure better occupational health. Also, early detection of gas leaks can trigger concerned engineers to curtail the spread and keep a safe environment for better health and safety.

5.	Business Model (Revenue Model)	The product can be made compact, cost efficient and easily installable so that all the industries from small scale to large scale can able to buy the product .
6.	Scalability of the Solution	The system is very simple and easy to maintain and cost efficient. It has the capability to works for a period of time without any damage in the system components.

3.4.Problem Solution fit:

Project Title: Gas Leakage Monitoring and Alerting System **Team ID:** PNT2022TMID09616

1. CUSTOMER SEGMENT(S) CS

- Oil, Gas, Polymer Industries
- Hospitals
- Safety Control Personals
- Mining

6. CUSTOMER CONSTRAINTS CC

- Network Connection
- Complexity in Installation
- High budget in installing other products make them to move far from modern technologies

5. AVAILABLE SOLUTIONS AS

- Upgrading to a premium network plan.
- Availing network connection from a reliable Service provider.

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

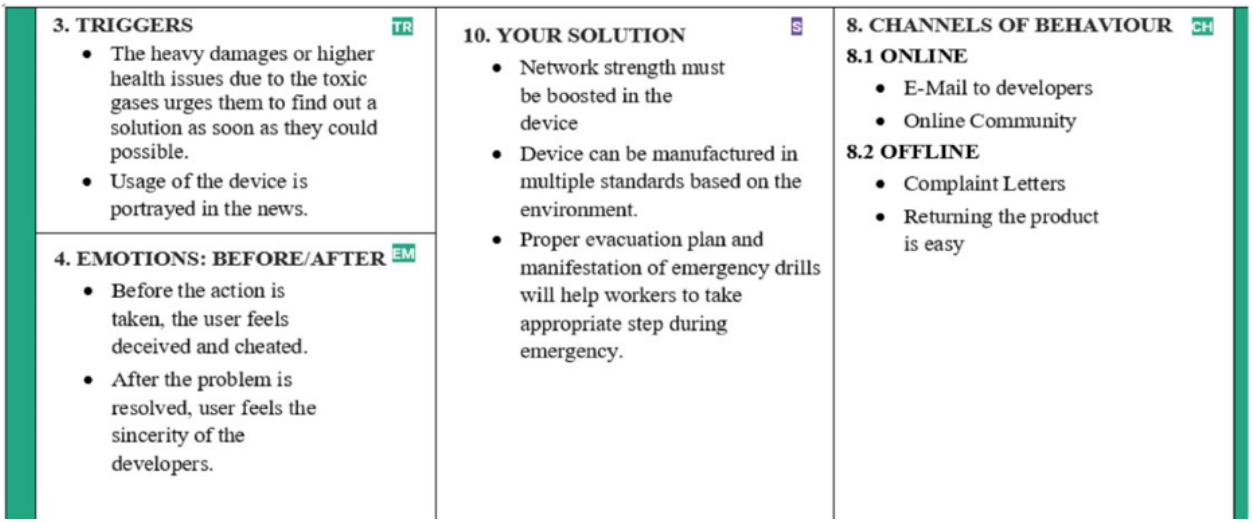
- Suffering from many losses due to gas leakage.
- Having no proper system for controlling or monitoring the leakage.
- Facing heavy budget problems in buying and installing a system for monitoring and controlling

9. PROBLEM ROOT CAUSE RC

- Quality of the material using which the device is made up of plays a vital role in the capability of the device to work in harsh environment.
- Location of the device installation and the network plan used by the user are the cause of Network issue.

7. BEHAVIOUR BE

- Harsh environment is prevailing only on certain industry; thus, the frequency of the said problem is low. In such a case the customer complaints multiple times to get the attention.
- Network issue is very common as most of the industries are located at the country side. Here the contact both the developers and the service providers



4.REQUIREMENT ANALYSIS:

4.1.Functional requirement:

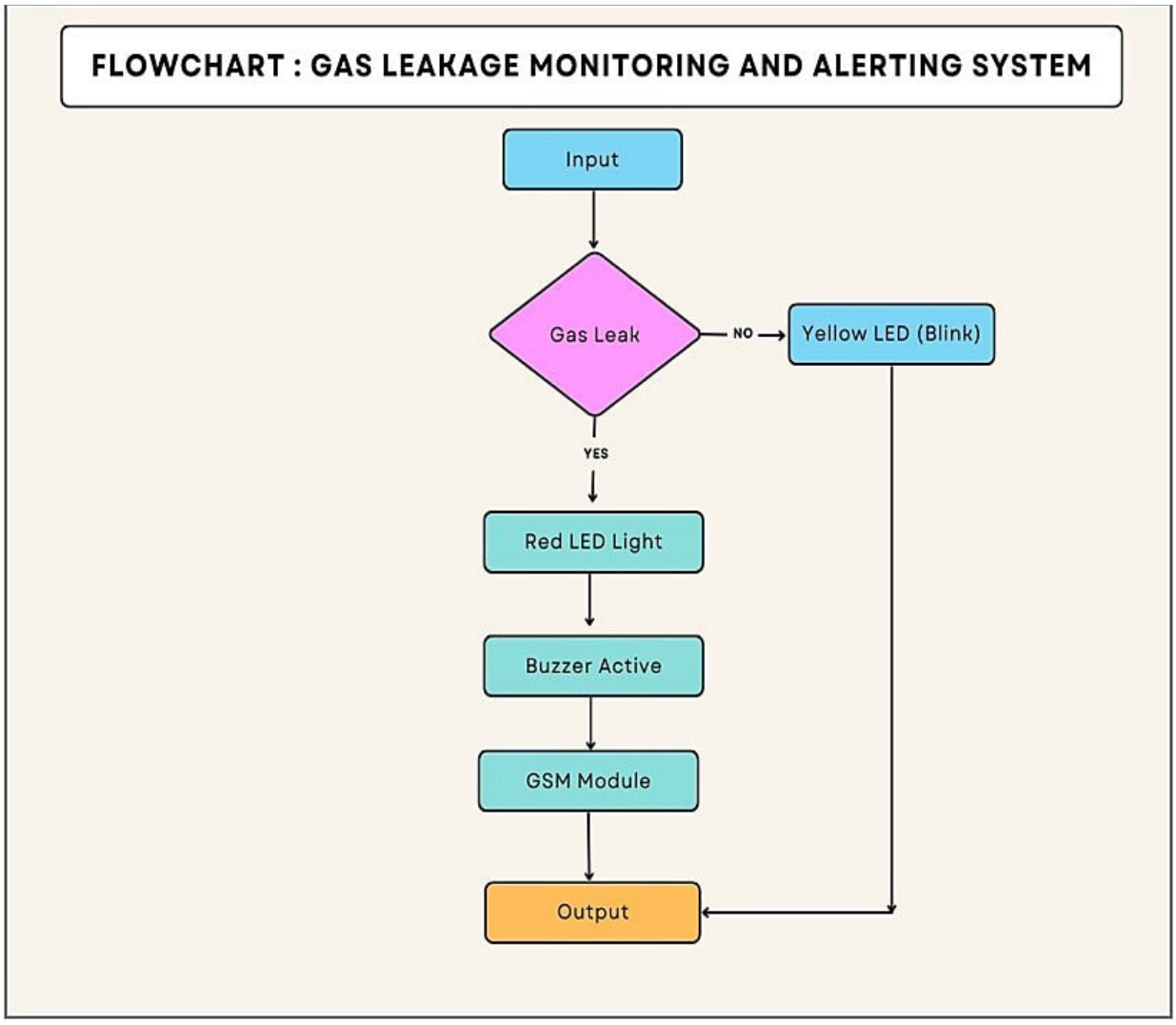
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	GPS Access	GPS access to know the location
FR-4	Business Requirements	The device is intended for the use of industries or factories and also for cylinder storage areas. It detects the leakage of gas and sends the data over to a site and preventive measures can be taken to avoid the loss of properties.
FR-5	User Requirements	The Gas leakage detecting system with upgrading technologies which identifies the leakage of gas and also ensures the workers safety.

4.2.**Non-Functional requirements:**

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The sensors used to detect the gas leakage which helps to prevent the high risk of gas explosion and also can prevent the causalities within and outside the covering area of the industries.
NFR-2	Security	The device is intended for the use of industries or factories, where there is a use of explosive gas is a source of risk. This device will help and secures from the causes.
NFR-3	Reliability	Gas leakage detecting system detects the gas leakage at industries or factories which detects the small amount of gas leakage as soon and sends the alerting SMS to users.
NFR-4	Performance	The Gas leakage detecting system is a device with an alarm setting. Whenever there is a gas leak ,which is greater than the threshold level, the in- build sensor detects and alerts the user within a minute much before it can cause any accidents.
NFR-5	Availability	The gas leakage detecting system is readily available in the market which is extremely expensive, but here we are providing a low-cost circuit for gas leakage detecting system and also it is user friendly
NFR-6	Scalability	The system is very simple and easy to maintain with cost efficient. A backup power supply will be included in the design to prevent from the power failure conditions. It has the capability to works for a period of time without any damage in the system components.

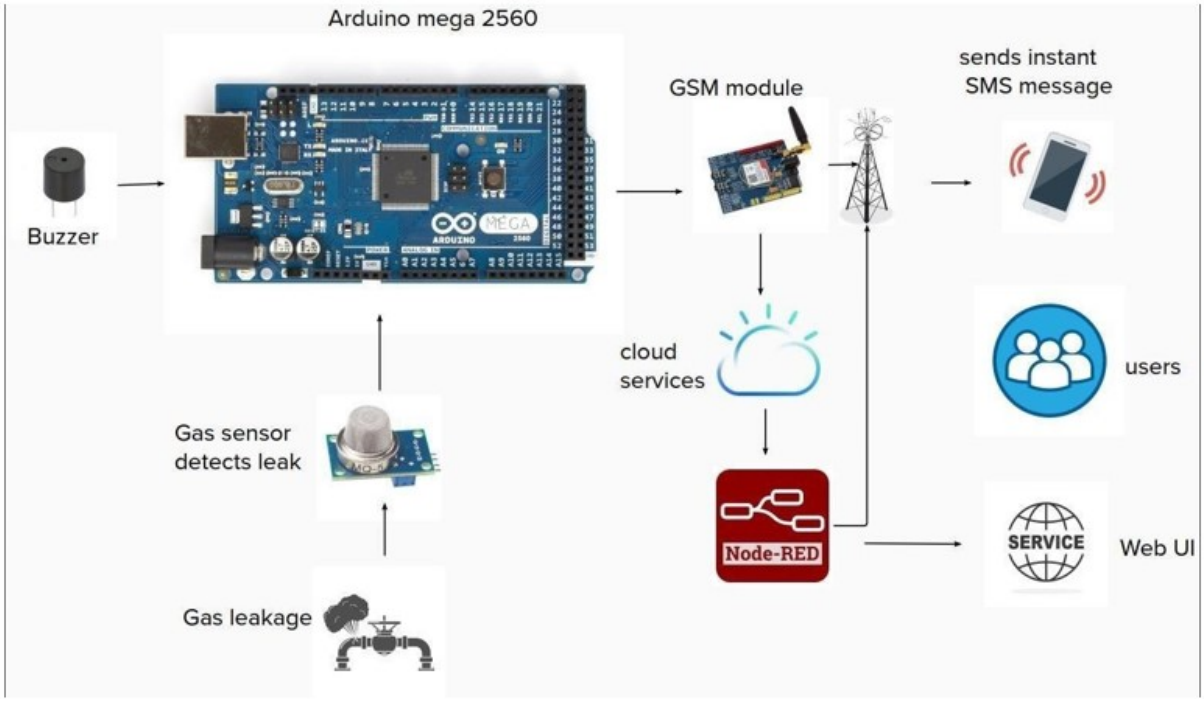
5.PROJECT DESIGN:

5.1.Data Flow Diagrams:the below data flow diagram shows how the gas leakage monitoring and alerting system works in the form of diagrams.



5.2.Solution & Technical Architecture:

given below are the components required



5.3.User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can create an account in the application provided.	I can access my account/ dashboard	High	Sprint-1

		USN-2	As a user, I registered using my Gmail.	I can receive confirmation email.	High	Sprint-1
		USN-3	As a user, I can successfully install the app.	I can register and access the dashboard.	Low	Sprint-2
	Login	USN-4	As a user, I can login using my Gmail and password easily.	The login process was easy and simple to access the dashboard.	High	Sprint-1
Customer (Web user)	Registration	WUSN-1	As a web user I can login to web dashboard just like a website.	I can register and access the dashboard.	High	Sprint-2
	Dashboard	WUSN-2	As a user I can view the alert/warning SMS in the web application.	I can login to the website using my login credentials	High	Sprint-2
Customer Care Executive		CCE-1	A customer care executive will always be available for the interaction with the customer to clarify the queries.	An executive will clarify the doubts and note down the complaints of the application if any.	High	Sprint-2

Administrator		ADMIN-1	I as an Admin can access and view the data or information provided by the application &can also check, analyse the threshold value of the gas.	The details of the gas leakage level of the gas are provided to the users through SMS when an alerting sound is received.	High	Sprint-1

6.PROJECT PLANNING & SCHEDULING:

6.1.Sprint Planning & Estimation:

1. SPRINT PLAN
2. ANALYZE THE PROBLEM
3. PREPARE AN ABSTRACT, PROBLEM STATEMENT
4. LIST A REQUIRED OBJECT NEEDED
5. CREATE A PROGRAM CODE AND RUN IT
6. MAKE A PROTOTYPE TO IMPLEMENT
7. TEST WITH THE CREATED CODE AND CHECK THE DESIGNED PROTOTYPE

6.2.Sprint Delivery Schedule:

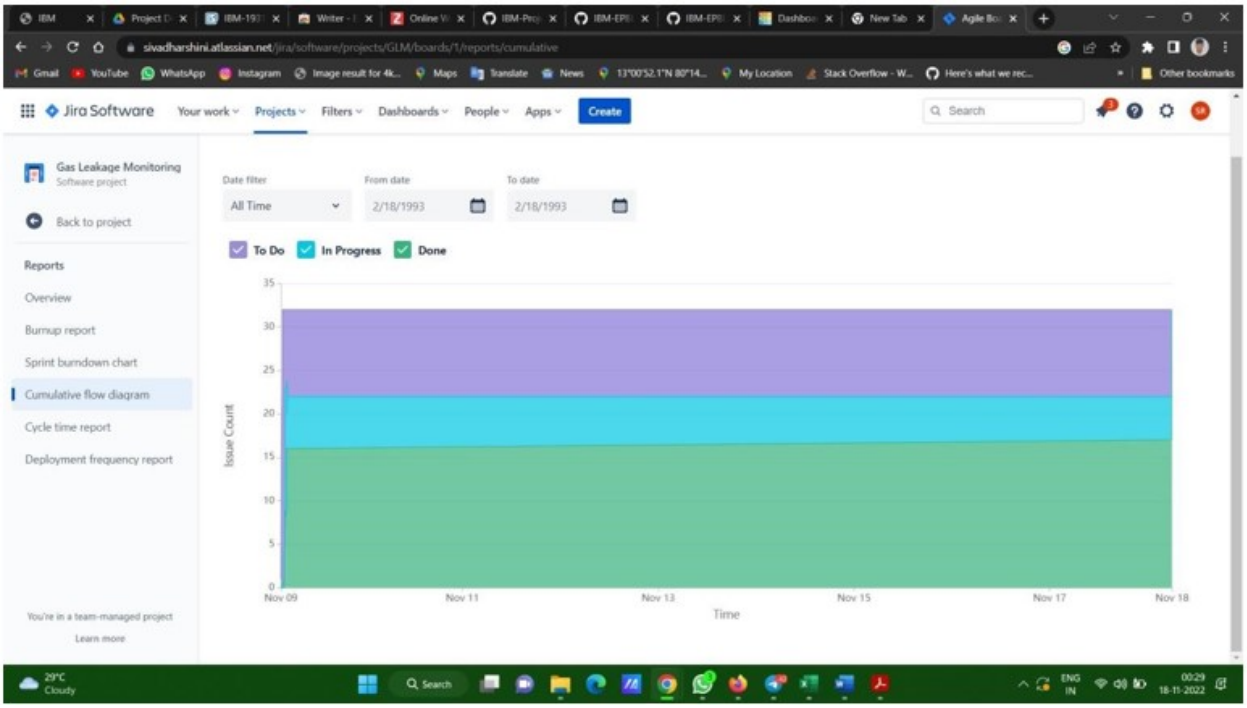
Sprint	Functional Requirement (Epic)	User Story	User Story / Task	Story Point	Priority

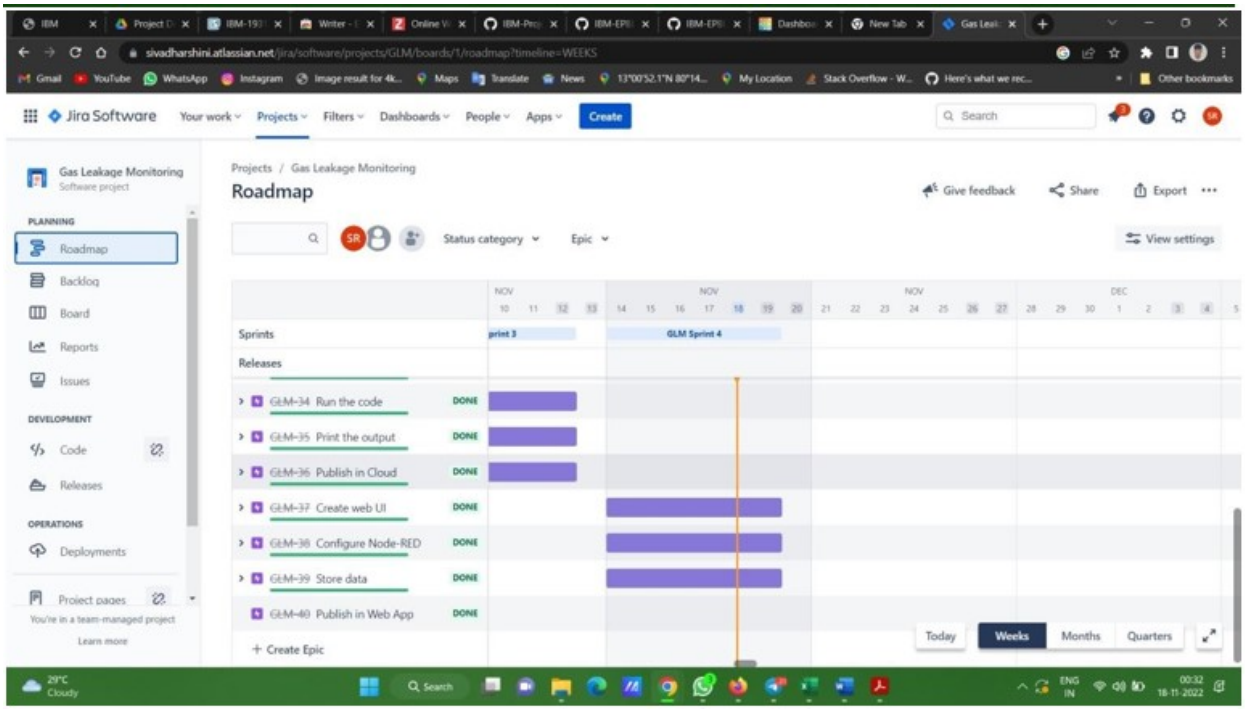
Sprint-1	Create	US-1	Create the IBM Cloud services which are being used in this project.	5	High
Sprint-1	Configure	US-2	Configure the IBM Cloud services which are being used in completing this project.	1	Medium
Sprint-1	Create	US-3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the IBM Watson IoT platform.	1	Medium
Sprint-1	Configure	US-4	Configure the IBM Watson IoT which are being used to display the output.	13	High
Sprint-2	Create	US-1	In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credentials.	13	High
Sprint-2	Configure	US-2	Configure a device in the IBM Watson IoT platform and get the device credentials.	3	Medium

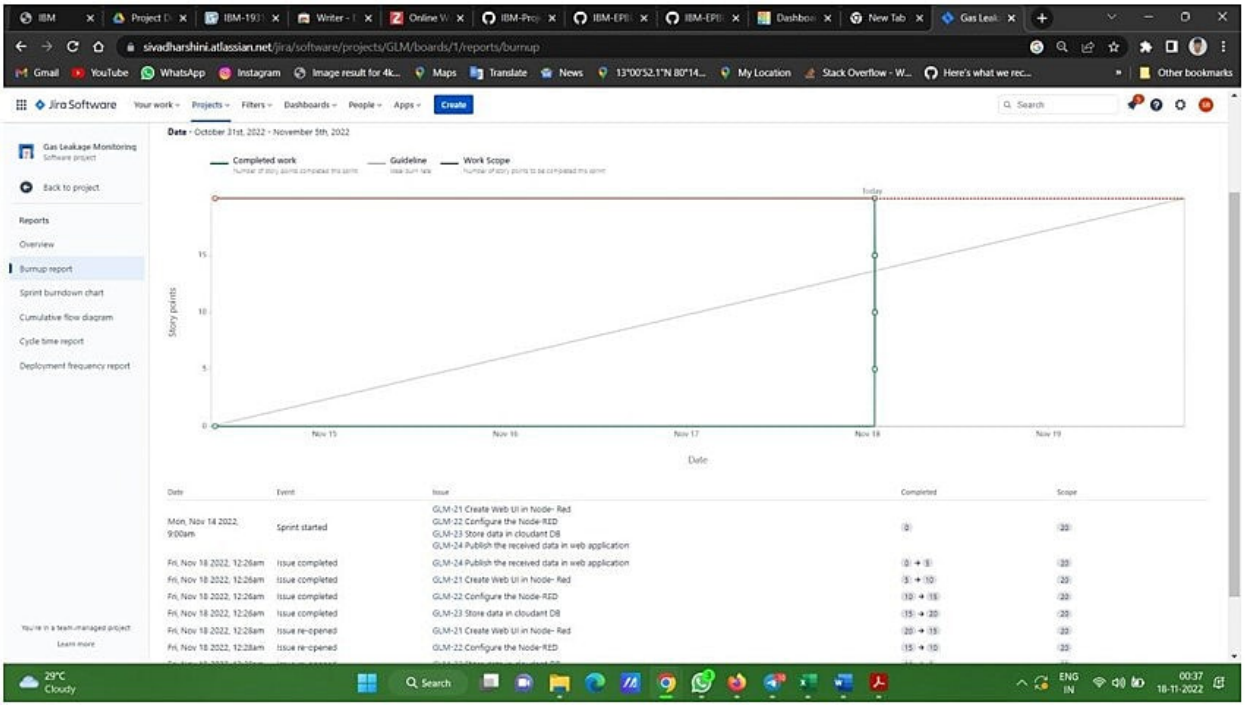
Sprint-2	Create	US-3	Create a Node-RED service.	3	High
Sprint-2	Configure	US-4	Configure the connection security and create API keys that are used in the Node- RED service for accessing the IBM IoT Platform.	1	Medium
Sprint-3	Develop	US-1	Develop a python script to publish random sensor data such as temperature, Flame level and Gas level to the IBM IoTplatform	1 3	High
Sprint-3	Configure	US-2	After developing python code and commands just run the code	1	Medium
Sprint-3	Print	US-3	Print the statements which represent the control of the devices.	1	Low
Sprint-3	Publish	US-4	Publish Data to The IBM Cloud	5	High
Sprint-4	Create	US-1	Create Web UI in Node- Red	5	High

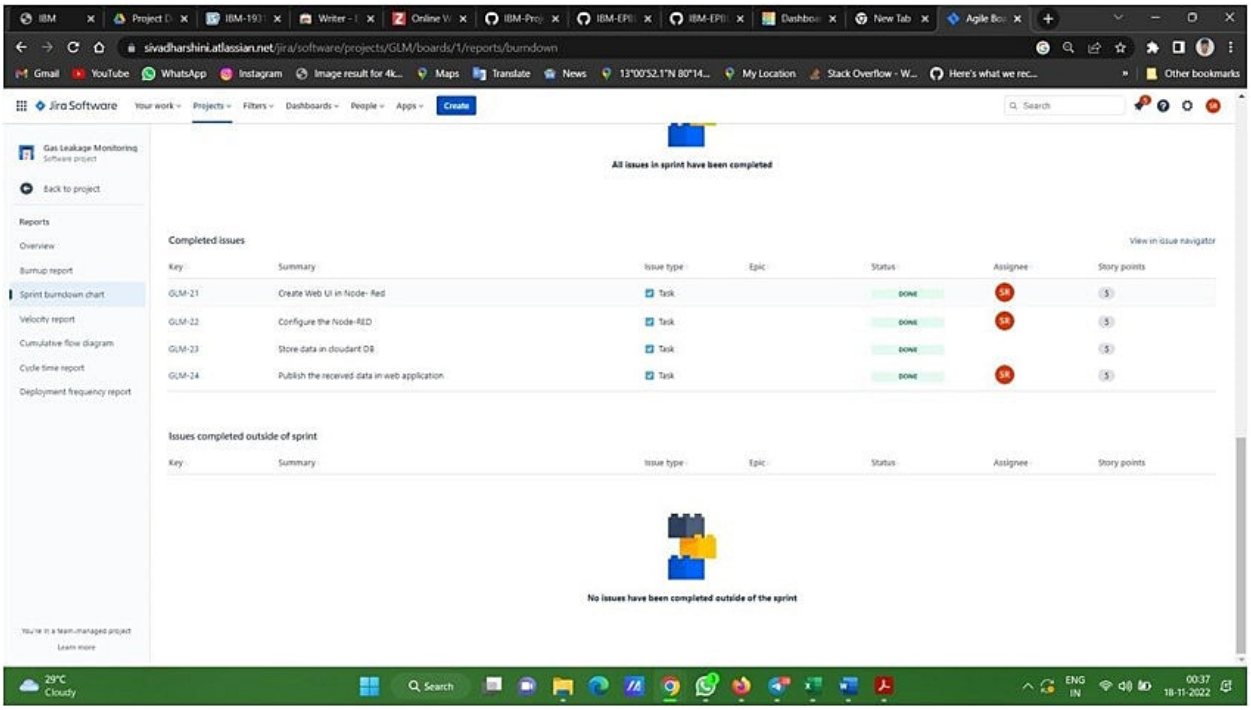
Sprint-4	Configure	US-2	Configure the Node-RED flow to receive data from the IBMIoT platform	5	High
Sprint-4	Configure	US-3	Use cloudant DB nodes to store the received sensor data in the cloudant DB	5	High
Sprint-4	Publish	US-4	Publish the received data inweb-application	5	High

6.3.Report from JIRA: report from JIRA is nothing but a normal type of report which keeps in track of the project and the deliverables are done at the correct time .how much percentage the project is completed .




Jira Software
Gas Leakage Monitoring
Roadmap
GLM-34 Run the code
GLM-35 Print the output
GLM-36 Publish in Cloud
GLM-37 Create web UI
GLM-38 Configure Node-RED
GLM-39 Store data
GLM-40 Publish in Web App
Today
Weeks
Months
Quarters


Jira Software
Gas Leakage Monitoring
Burnup report
Completed work
Guideline
Work Scope
Story points
Date



7.CODING & SOLUTIONING:

```
# Importing Required
modules import time
import sys
import wiotp.sdk.device# IBM IoT Watson Platform Module import ibmiotf.device import tkinter as tk # Python GUI Package from tkinter import ttk # Python GUI import time
from threading import Thread


organization = "0tus0f" # Organization ID deviceType = "ESP32" # Device type deviceId = "01" # Device ID authMethod = "token" # Authentication Method authToken = "Gowth@m@nk18" #Replace the authtoken
```

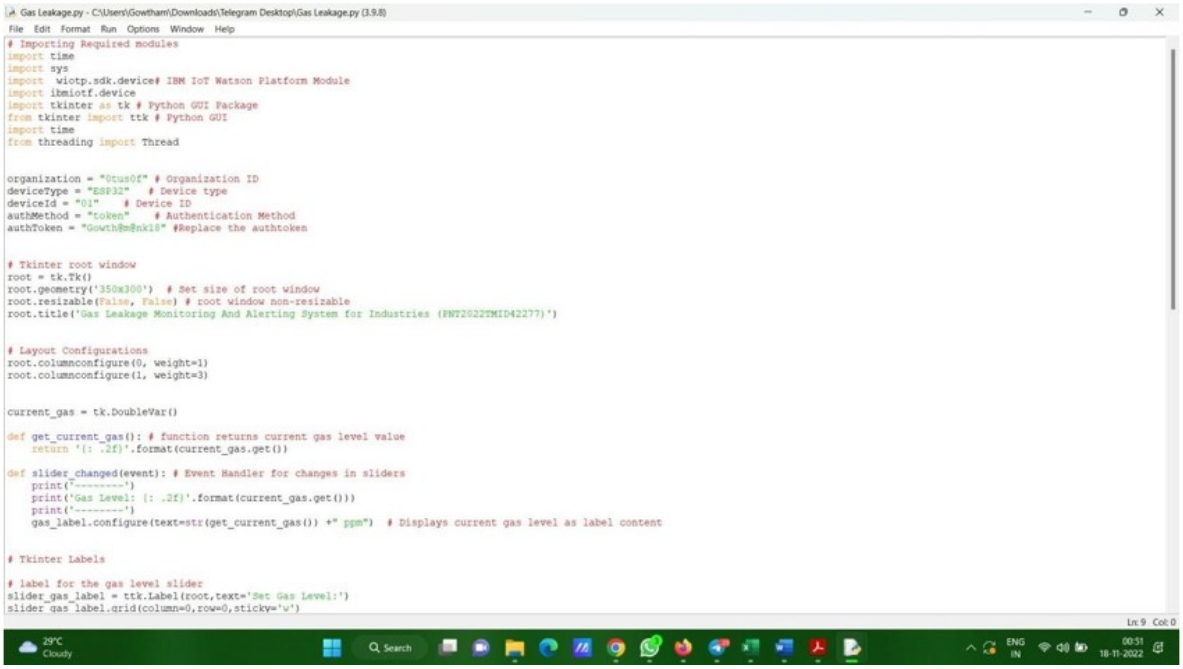
```
# Tkinter root window root = tk.Tk()
root.geometry('350x300') # Set size of root window
root.resizable(False, False) # root window non-resizable
root.title('Gas Leakage Monitoring And Alerting System for Industries (PNT2022TMID42277)')


# Layout Configurations
root.columnconfigure(0,
weight=1)
root.columnconfigure(1,
weight=3)


current_gas = tk.DoubleVar()

def get_current_gas(): # function returns current gas level value
    return '{: .2f}'.format(current_gas.get())

def slider_changed(event): # Event Handler for changes in sliders
    print(' ------ ')
    print('Gas Level: {: .2f}'.format(current_gas.get()))
    print(' ------ ')
    gas_label.configure(text=str(get_current_gas()) +" ppm") # Displays current gas level as label content


# Tkinter Labels

# label for the gas level slider slider_gas_label = ttk.Label(root,text='Set Gas Level:')
slider_gas_label.grid(column=0,row=0,sticky='w')


# Gas Level slider
slider_gas = ttk.Scale(root,from_=0,to=3000,orient='horizontal',
command=slider_changed,variable=current_gas)
slider_gas.grid(column=1,row=0,sticky='we')


# current gas level label
current_gas_label = ttk.Label(root,text='Current Gas Level:')
```

```
current_gas_label.grid(row=1,columnspan=2,sticky='n',ipadx=10,ipady=10)


# Gas level label (value gets displayed here)
gas_label = ttk.Label(root,text=str(get_current_gas()) +" ppm")
gas_label.grid(row=2,columnspan=2,sticky='n')


def publisher_thread(): thread
    =
    Thread(target=publish_data)
    thread.start()

def
    publish_data
    (): #
    Exception
    Handling 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()
    deviceCli.connect() # Connect to IBM Watson IoT
    Platform

    while True: gas_level =
        int(current_gas.get())
        data = {'gas_level' :
        gas_level}

        def myOnPublishCallback():
  print("Published Gas Level = %s ppm" % gas_level, "to IBM Watson")

   success = deviceCli.publishEvent("event", "json", data, qos=0,
on_publish=myOnPublishCallb
        ack) if not success:
            print("Not connected to IoTF")
        time.sleep(1)

publisher_thread()
```

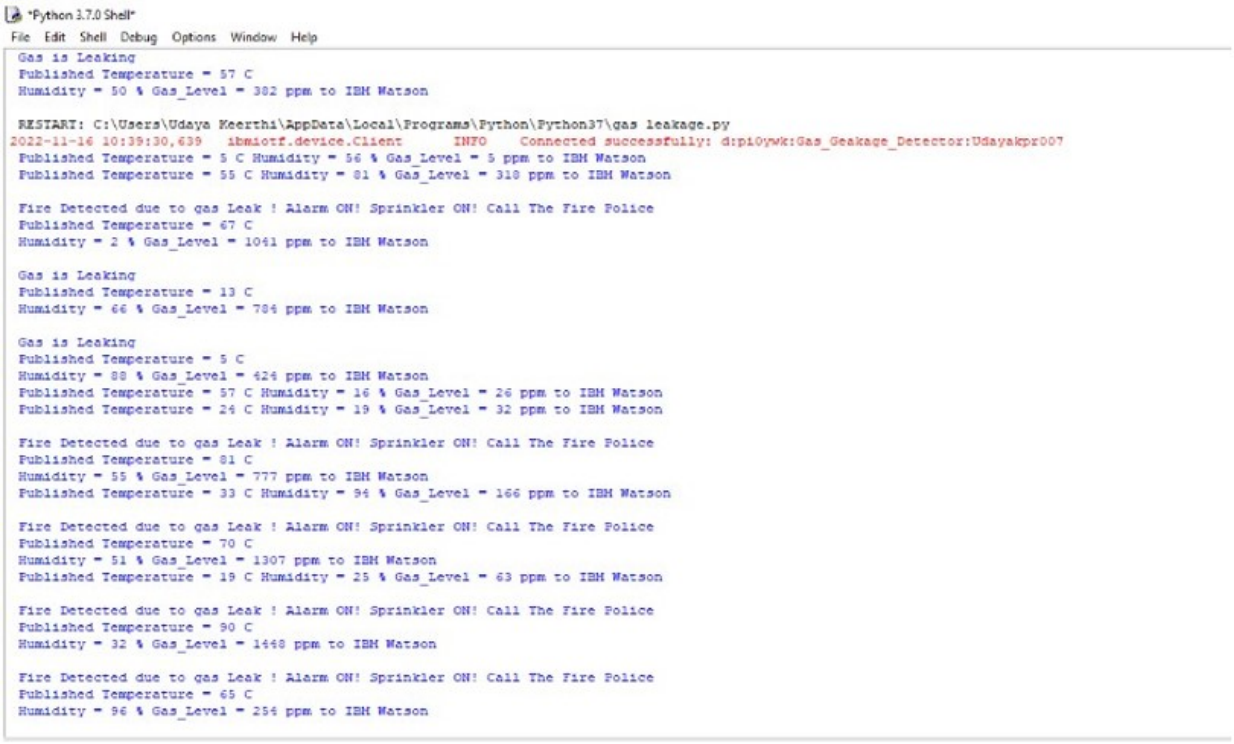
root.mainloop() # startup Tkinter GUI

Disconnect the device and application from the cloud deviceCli.disconnect()

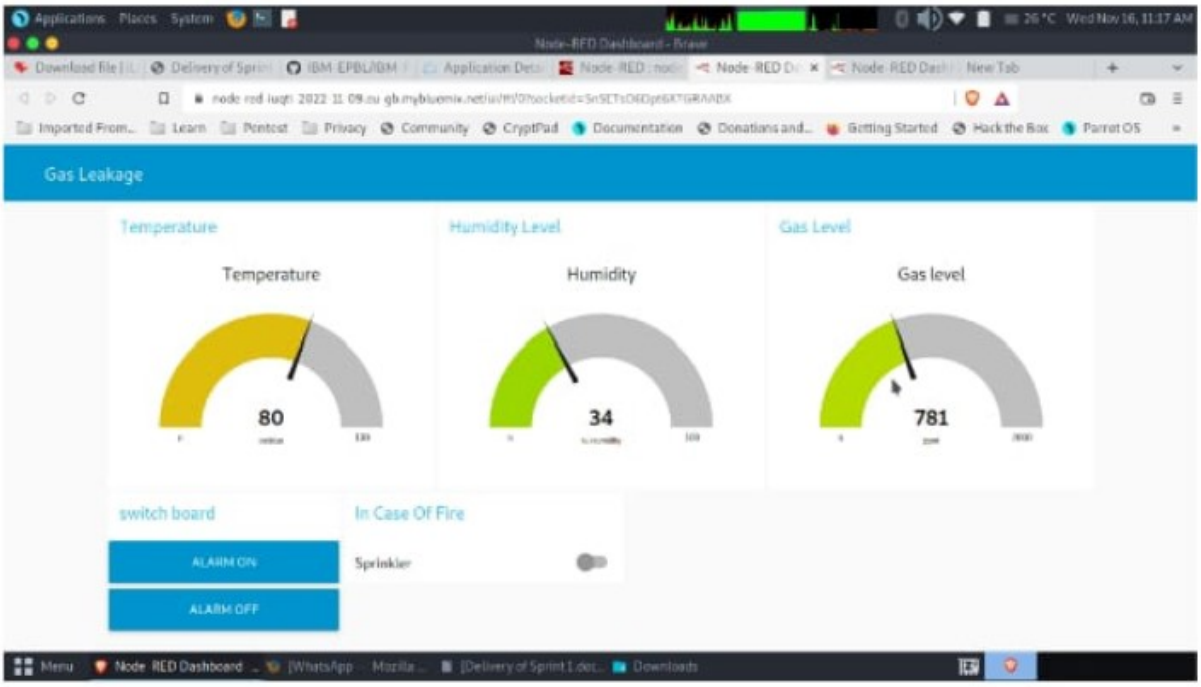
7.1.CODE:



7.2output:


Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Gas is Leaking
Published Temperature = 57 C
Humidity = 50 % Gas_Level = 382 ppm to IBM Watson
RESTART: C:\Users\Udaya Keerthi\AppData\Local\Programs\Python\Python37\gas leakage.py
2022-11-16 10:39:30,639 ibmiotf.device.Client INFO Connected successfully: d:pi0ywk:Gas_Geakage_Detector:Udayakpr007
Published Temperature = 5 C Humidity = 56 % Gas_Level = 5 ppm to IBM Watson
Published Temperature = 55 C Humidity = 81 % Gas_Level = 310 ppm to IBM Watson
Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police
Published Temperature = 67 C
Humidity = 2 % Gas_Level = 1041 ppm to IBM Watson
Gas is Leaking
Published Temperature = 13 C
Humidity = 66 % Gas_Level = 784 ppm to IBM Watson
Gas is Leaking
Published Temperature = 5 C
Humidity = 88 % Gas_Level = 424 ppm to IBM Watson
Published Temperature = 57 C Humidity = 16 % Gas_Level = 26 ppm to IBM Watson
Published Temperature = 24 C Humidity = 19 % Gas_Level = 32 ppm to IBM Watson
Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police
Published Temperature = 81 C
Humidity = 55 % Gas_Level = 777 ppm to IBM Watson
Published Temperature = 33 C Humidity = 94 % Gas_Level = 166 ppm to IBM Watson
Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police
Published Temperature = 70 C
Humidity = 51 % Gas_Level = 1307 ppm to IBM Watson
Published Temperature = 19 C Humidity = 25 % Gas_Level = 63 ppm to IBM Watson
Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police
Published Temperature = 90 C
Humidity = 32 % Gas_Level = 1448 ppm to IBM Watson
Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police
Published Temperature = 65 C
Humidity = 96 % Gas_Level = 254 ppm to IBM Watson

8.Testing:



9.Result:

The system can be taken as a small attempt in connecting the existing primary gas detection methods to a mobile platform integrated with IoT platforms. The gases are sensed in an area of 1m radius of the rover and the sensor output data are continuously transferred to the local server. The accuracy of sensors is not up to the mark thus stray gases are also detected which creates an amount of error in the outputs of the sensors, especially in case of methane. Further the availability and storage of toxic gases like hydrogen sulphide also creates problems for testing the assembled hardware. As the system operates outside the pipeline, the complication of system maintenance and material selection of the system in case of corrosive gases is

reduced. Thus, the system at this stage can only be use data primary indicator of leakage inside a plant.

10.Advantages/Disadvantages:

10.1 Advantages:

1. Get real-time alerts about the gaseous presence in the atmosphere.
2. Prevent fire hazards and explosions.
3. Supervise gas concentration levels.
4. Ensure worker's health.
5. Real-time updates about leakages.
6. Cost-effective installation.
7. Data analytics for improved decisions.
8. Measure oxygen level accuracy.
9. Get immediate gas leak alerts.

10.2 Disadvantages:

1. It requires air or oxygen to work.
2. It gets reacted due to heating of wire.
3. It can be poisoned by lead, chlorine and silicon

11.CONCLUSION:

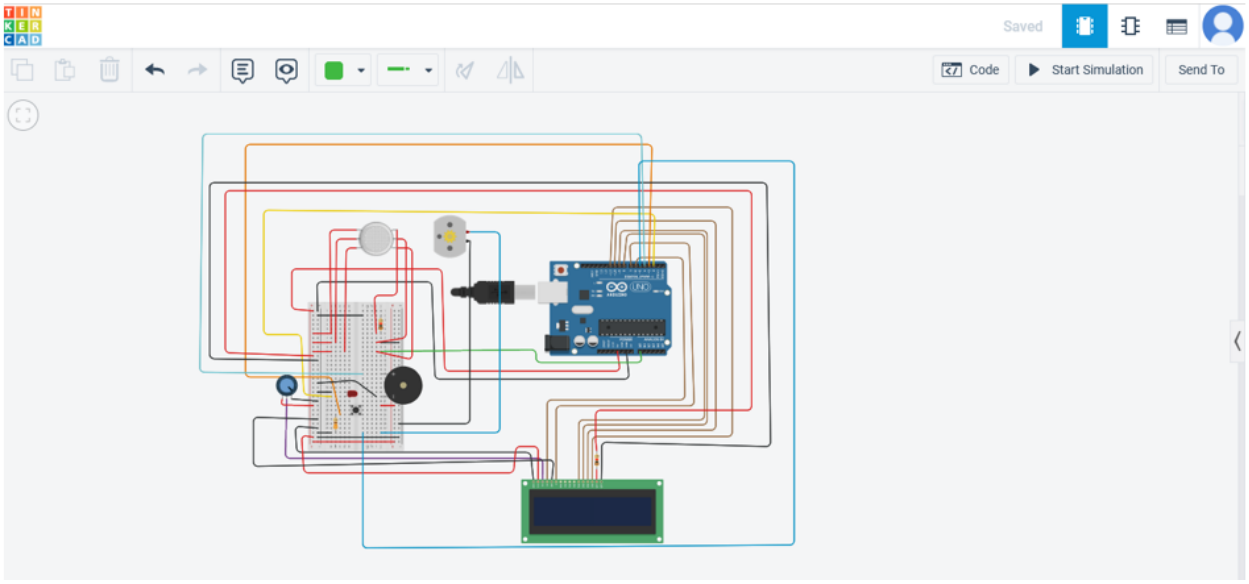
This gas leak detector system contains two features, this includes the SMS Gateway feature for only sending warning information regarding the gas leak to user, and the alarm for the warning alert. There is some improvement which can be applied for the future work, such as regarding the SMS Gateway, it need to enhance with feature such as notifying the user whenever the remaining credit balance is insufficient. Another thing which can be enhanced is regarding the sensor, the sensors in this module do not include somewhat notification for notifying the user whenever the sensor not working properly or not connected to the micro-controller for some cases, therefore, it is recommended to add this kind of features in the future work for better refinement.

12.FUTURE SCOPE:

We propose to build the system using an MQ6 gas detection sensor and interface it with an Aurdino Uno microcontroller along with an LCD Display. This system uses the gas sensor to detect any gas leakages. The gas sensor sends out a signal to the microcontroller as soon as it encounters a gas leakage. The microcontroller processes this signal and a message is displayed on the LCD to alert the user.

13.APPENDIX:

13.1.Circuit Diagram:



13.2.Components:

The design of a sensor-based automatic gas leakage detector with an alert and control system. The components are

S.NO	NAME OF THE COMPONENT	QUANTITY
1	Arduino Uno R3	1
2	LCD 16x2	1
3	Piezo	1

4	Gas sensor	1
5	1 k ohm Resistor	1
6	2.3 k ohm Resistor	1
7	4.7 k ohm Resistor	1
8	Red LED	1
9	Green LED	1

13.4 Source Code:

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

int redled = A5; int greenled = A3;
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.println(analogValue);
  if(analogValue>sensorThresh)
  {
    digitalWrite(redled,
    HIGH);
```

```
    digitalWrite(greenle
    d,LOW);
    tone(buzzer,1000,1
    0000); lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("ALERT");
    Serial.print("ALERT"
    ); delay(1000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("EVACUAT
    E"); Serial.println(" -
    - EVACUATE");
    delay(1000);
}
  else
  {
    digitalWrite(greenle
    d,HIGH);
    digitalWrite(redled,
    LOW);
    noTone(buzzer);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("SAFE");
    Serial.print("SAFE");
    delay(1000);
    lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("ALL
    CLEAR");
    Serial.println("       --
    ALL             CLEAR");
    delay(1000);
}
}
```

13.5 GITHUB:

Link : https://github.com/IBM-EPBL/IBM-Project-24440-1659942903

13.6 Demo Video:

LiNK: https://youtu.be/Eqz5v_VvZKQ