Project Report

Date	**14 November 2022**
Team ID	**PNT2022TMID30997**
Project Name	**Gas Leakage Monitoring And Alerting System For Industries**

Team Members:

1. M.Nandha Kumar (620819104066)
2. M.Moulidharan (620819104061)
3. K.P Thirumurthy (620819104113)
4. S.Vasanthakumar (620819104115)

GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

TEAM ID: PNT2022TMID30997

A PROJECT REPORT

Submitted by

1. M.Nandha Kumar
2. M.Moulidharan
3. K.P Thirumurthy
4. S.Vasanthakumar

Industry Mentor's Name
Kumar Juluri

Faculty Mentor's Name
R.Umamaheswari

CHAPTER-1
INTRODUCTION

1.1 Project Overview

The Internet of Things (IOT) is a network of devices, cars, and home appliances that have the hardware, software, actuators, and networks necessary to communicate with one another, function together, and exchange information. IOT entails expanding the Internet network beyond traditional devices like workspaces, workstations, smartphones, and tablets to any variety of physically stupid or non-web enabled physical devices and everyday objects. These modern devices can communicate with each other and link to the Internet, allowing for remote monitoring and control. The industries benefit from this project's assistance in tracking hazardous gas emissions. The gas sensors will be integrated to monitor the gas leakage so that we can prevent the discharge of gases in different regions. The administrators will be alerted and given the location if a gas leak is found in any region. The administrators can view the sensor settings via the web application.

1.2 Purpose

The gas detectors can be used for the detection of combustible, flammable and poisonous gases and for loss of oxygen, and also to detected a gas leak or other pollutants. It makes the area where the leak occurs an warning sound and instructs operators to leave the area

CHAPTER-2
LITERATURE REVIEW

2.1 Existing Problem

In industries,the existing Problem in gas monitoring is that there is no efficient system for monitoring the gas leakage, the good system are of high cost and also the installation process is too complicated.Then the affordable of the system is high and the systems are sometimes making disasters and the number of sens ors is unpredictable and the positioning of equipment is improper

2.2 References

1) Bing Han, Qiang Fu, Hanfang How, 'Methane Leakage Monitoring Technology For N atural Gas Stations And Its Application', IEEE 5th International Conference on Computer and C ommunications,2001.

2) Shruthi Unnikrishnan,1 Mohammed Razil, Joshua Benny, Shelvin Varghese and C.V. H ari, 'LPG Monitoring And Leakage Detection System', Department of Applied Electronics and I nstrumentation Engineering, Rajagiri School of Engineering and Technology, Rajagiri Valley, K akkanad, Kochi, India.

3) J.Vijayalakshmi, Dr.G.Puthilibhai, S.R.Leoram Siddarth, 'Implementation Of Ammonia G
as Leakage Detection & Monitoring System Using Internet Of Things', West Tambaram, C hennai.

4) Makiko Kawada, Tadao Minagawa, Eiichi Nagao, Mitsuhito Kamei, Chieko Nishida and K oji Ueda, 'Advanced Monitoring System For Gas Density Of GIS', Mitsubishi Electric Corporation

2.3 Problem Statement Definition

Since the number of sensors is unpredictable, the industrialists feel in secured in handling the gases. Also, the cost price of the products and the complications in installing the systems are high. This makes the customers feel disappointed sometimes. For monitoring gas leakage in the industry and Control the gas leakage ,we create a system for monitoring gas leakage and makes the installation propose simple

CHAPTER-3
IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

What we think to create device which helps us to control emission of flammable substance into the environment. It should be user friendly and low cost for maintenance. For that we see continuous monitoring device and buzzer is to indicate the leakage

3.2 Ideation & Brainstorming

The ideas are In case of higher gas leakage and fire accidents, a notification can be given to the fire station and hospital through software application. The level of gas in the industry can be informed through speakers periodically. When gas gets leaked, a notification can be passed to hospital. Sensor can be placed in the entrance for counting the workers who have been moved out in case of emergency. In addition to alarm, a voice notes which alerts by saying the level of leakage can be d esigned.The alerting message can also be forwarded to the management o f the industry. Sprinklers or extinguishers can be fixed which helps in case of inflammation by the leakage. Windows and gates can be opened automatically through sensors placed on that.

3.3 Proposed Solution

To
Develop an efficient system & an application that can monitor and alert t he u sers(workers), our 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. It is fastest alerts to the workers and user friendly. For social impact it is Cost efficient and easy installation and provide efficient results and can work with irrespective of fear. 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.

S.NO	PARAMETER	DESCRIPTION
1	Problem Statement (Problem to be solved)	This monitoring is used to prevent fire accidents due flammable gas leakage in house from cylinders,industries,hospitals, hotels etc.
2	Idea / Solution description	This monitoring system uses cloud and iot based hard wares and sensors.The sensors in the system detects flammable gaseous components in the environment and temperature using iot system and send indication via alarms and lights
3	Novelty / Uniqueness	The uniqueness of this system is that it uses cloud due to this, the alarm can be to the person via sms to his mobile when he is not in home.
4	Social Impact / Customer Satisfaction	It helps in many ways to the society it prevents fire accidents due careless handle of gas cylinders . this is a real-time systems so it is faster and accidents can be prevented very easily
5	Business Model (Revenue Model)	This is a cloud based real time system ,that collects the data from the environment very quickly i.e. temperature,humidity and oxygen composition. using I sensors and indicate via alarms and lights.
6	Scalability of the Solution	1 Accuracy. 2 Low cost. 3 Less maintenance. 4 Reliability.

3.4 Problem Solution Fit

Define CS, fit into CL	**1. CUSTOMER SEGMENT(S)** CS The industrialists who use gases for their manufacturing.	**6. CUSTOMER LIMITATIONS** EG. BUDGET, DEVICES CL High budget in installing other products make them to move far from modern technologies.	**5. AVAILABLE SOLUTIONS** PLUSES & MINUSES AS The monitoring and controlling of the leakage could be done by the manpower. Even though man power could reduce electricity cost and monitor properly, it may cause high risk for their life. There is also a cause of some errors due to manpower.	Explore AS, differentiate
Focus on PR, tap into BE, understand RC	**2. PROBLEMS / PAINS** + ITS FREQUENCY PR • 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 When the workers failed to monitor properly, the gas can cause high risk to their health or the properties of the industry.	**7. BEHAVIOR** + ITS INTENSITY BE • Using manpower as the source of monitoring the leakage causes high hazards. • If the gas leaked is heavily toxic, there is a chance of causing hereditary health issues too.	Focus on PR, tap into BE, understand RC
Identify strong TR & EM	**3. TRIGGERS TO ACT** TR The heavy damages or higher health issues due to the toxic gases urges them to find out a solution as soon as they could possible. **4. EMOTIONS** BEFORE / AFTER EM **Before:** The heavy losses due to the leakages made them feel of guilt due to reduced reputation of their products. **After:** Increased the level of confidence and feel secured	**10. YOUR SOLUTION** SL Develop an efficient system & an application that can monitor and alert the workers.	**8. CHANNELS of BEHAVIOR** CH Promoting through social media. With the help of social media entrepreneurs/influencer. OFFLINE Through newspaper advertisements.	Extract online & offline CH of BE

CHAPTER-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	User Visibility	Level of gas can be monitored by users if there is any leakage, alerts can be sent through messages.
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. They also get notified by the alert
FR-4	User Convenience	Through message we can easily get data of gas level and in case of gas leakage, it can directly send notifications to nearby police station and hospital.
FR-5	User Performance	When the user gets notified, he could turn on the exhaust fan/sprinkler

4.2 Non-Functional Requirments

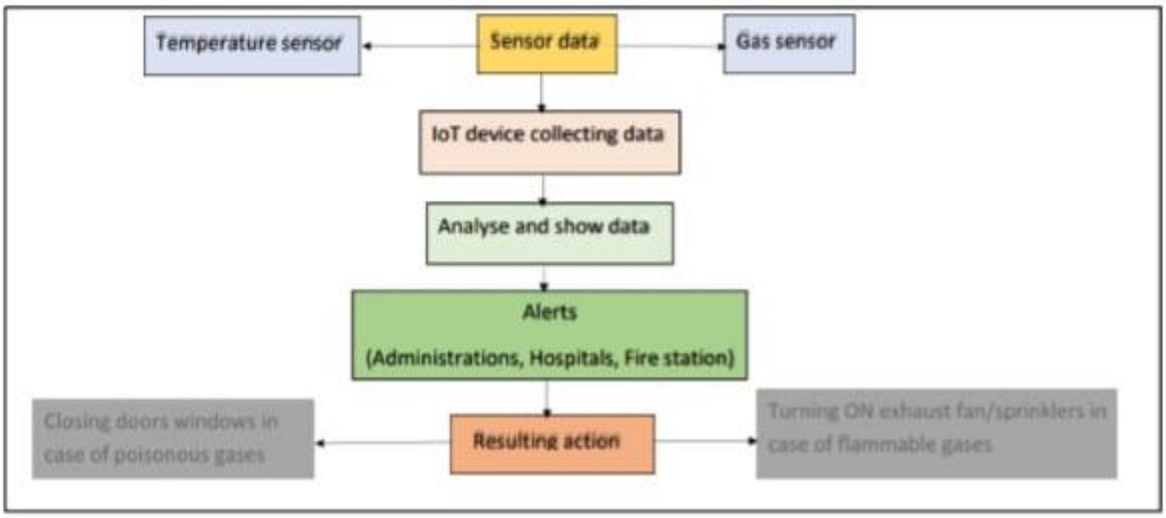
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It updates the data regularly as well as protects the workers.
NFR-2	**Security**	As a result of emergency alert, we can be able to protect both the humans and properties.
NFR-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
NFR-4	**Performance**	Sprinklers and exhaust fans are used in case of emergency.
NFR-5	**Availability**	It can be used for everyday; it includes day and nights.
NFR-6	**Scalability**	Sensors can be replaced every time it fails.

CHAPTER-5
PROJECT DESIGN

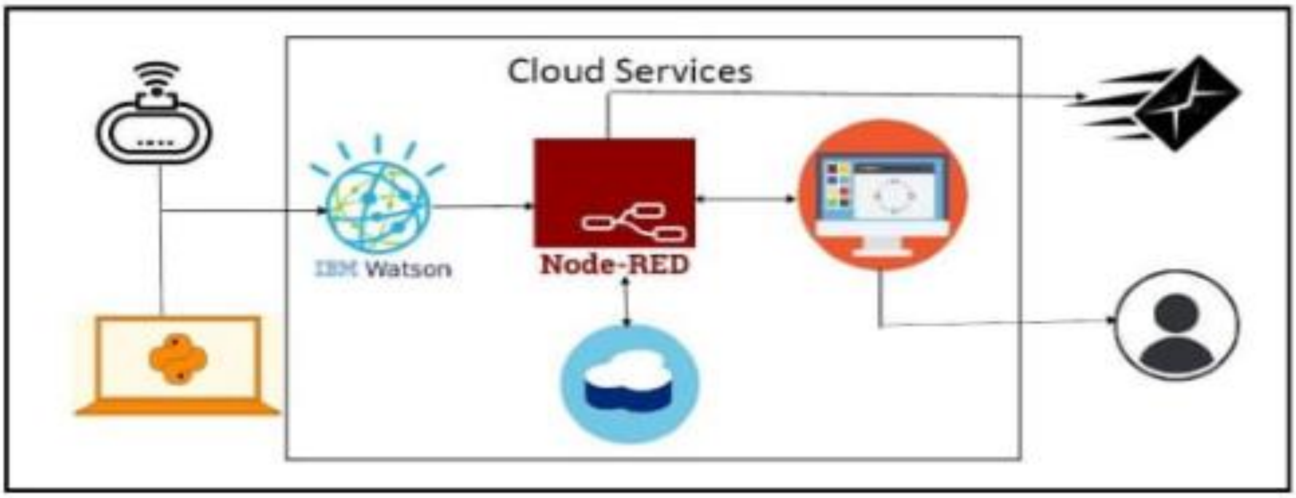
5.1 Data Flow Diagram

This is the data flow diagram of gas leakage monitoring and detection. Here the data from temperature sensor and gas sensor is collected from IOT device and the data is analysed. If the alert action requires it alerts and the required measures are taken.



5.2 Solution & Technical Diagram

This is the technical diagram of gas leakage monitoring and detection. Here the data from temperature sensor and gas sensor is collected and is connected to IBM Watson (cloud) . Node red is connected to cloud and the result of the data from the cloud flows.



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	User Can enter into the web application	I can access my account /dashboard	High	Sprint-1
		USN-2	User can register their credentials like email id and password	I can receive confirmation email & click confirm	High	Sprint-1
	Login	USN-3	User can log into the application by entering email &password	I can login to my account	High	Sprint-1
	Dashboard	USN-4	User can view the temperature	I can view the data given by the device	High	Sprint-2
		USN-5	User can view the level of gas	I can view the data given by the device	High	Sprint-2
Customer (Web User)	Usage	USN-1	User can view the web page and get the information	I can view the data given by the device	High	Sprint-3
Customer	Working	USN-1	User act according to the alert given by the device	I can get the data work according to it	High	Sprint-3
		USN-2	User turns ON the exhaust fan/sprinkler when the leakage occurs	I can get the data work according to it	High	Sprint-4
Customer Care Executive	Action	USN-1	user solve the problems when someone faxes any usage issues	I can solve the issues when some one fails to understand the procedure	High	Sprint-4
Administrator	Administration	USN-1	User Stores every information	I can store the gained information	High	Sprint-4

CHAPTER-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	IDE	USN-1	Installing all the software's which is required like python IDE	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-1	Checking the simulation with condition	USN-1	Simulating the circuits and experimenting	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-2	Software	USN-2	-IBM Watson iot -NodeRed Integration	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-2	Software	USN-2	Test the device and workflow	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S

Sprint-3	Application Development	USN-3	using MIT App Inventor Create an app	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-3	Testing	USN-3	Testing the Application	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-4	Web UI	USN-4	User interface with the software	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S

6.2 Sprint Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story/Task	Story Points	Priority	Team Members
Sprint -1	Resource initialization	create and initialize accounts in various public APIs like Open/WeatherMap API	1	Low	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-1	Local Server /Software to cloud	Write a python program that outputs results given the inputs like weather and location	1	Medium	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-2	Push the server/software to cloud	Push the code from sprint 1 to cloud as it can be accessed from anywhere	2	Medium	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-3	Hardware initialization	Integrate the hardware to be able to access the cloud and provide inputs to the same	2	High	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S
Sprint-4	UI/UX Optimization &Debugging	Optimize all the short coming and provide user experience	2	Low	Nandha Kumar M Moulidharan M Thirumurthy KP Vasanthakumar S

Project Tracker,Velocity & Burndown Chart:

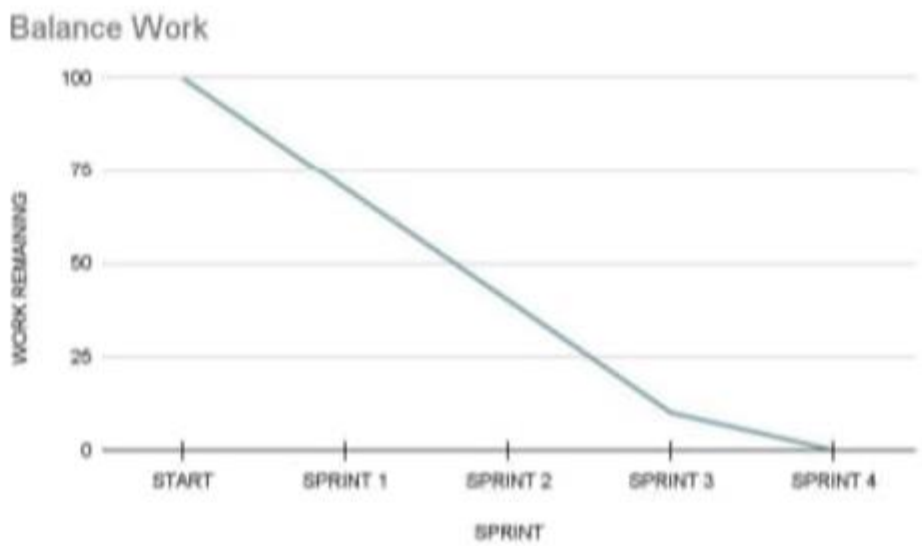
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	31 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022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

Velocity:

Imagine we have a 10-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)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

Burndown Chart:



CHAPTER-7
CODING & SOLUTIONING

7.1 FEATURE 1:

Code:

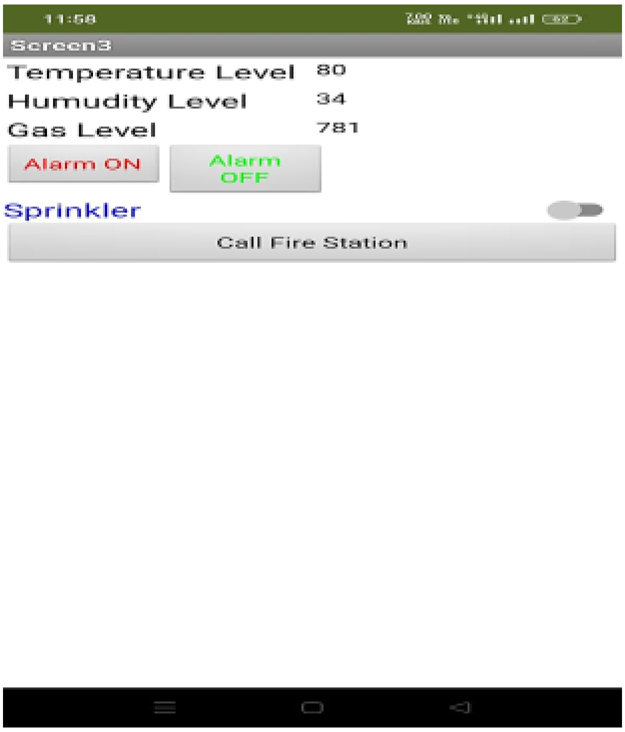
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "vh5b9d"
deviceType = "Raspberry"
deviceId = "0330"
authMethod = "token"
authToken = "123456789"
# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="alarmon":
        print ("Alarm is on")
    elif (status == "alarmoff") :
        print ("Alarm is off")
    elif status == "sprinkleron":
        print("Sprinkler is OFF")
    elif status == "sprinkleron":
        print("Sprinkler is ON")
    #print(cmd)
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)
        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 %%" 
% Humid, "Gas_Level = %s %%" %gas, "to IBM Watson")
        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, 
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

7.2 FEATURE:

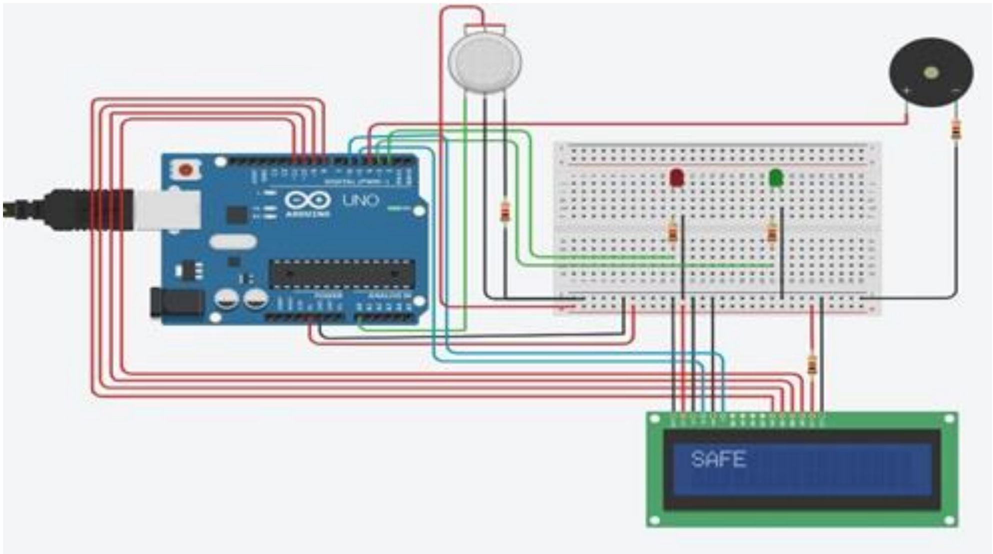
We have developed the application in MIT app inventer which can monitor the temperature ,humidity and gas leakage. It also has the features like sprinklers and alarm which will indicate .If the situation is uncontrollable ,we can call the fire station through the Call fire Station button



CHAPTER-8
TESTING

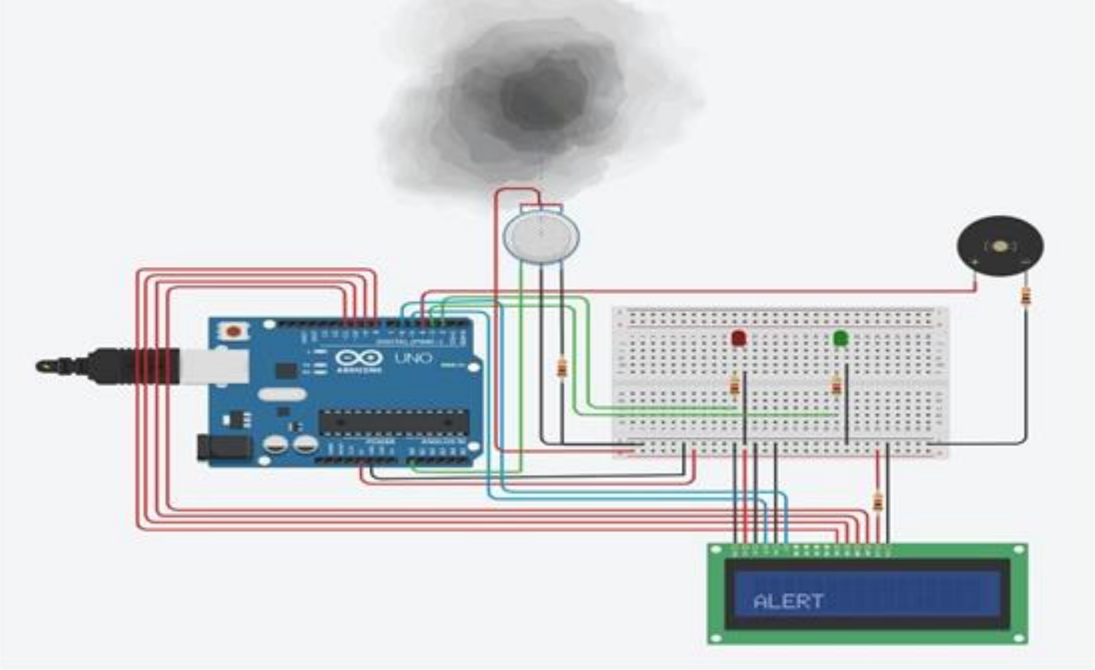
8.1 Test Cases:

Test case 1



Industries at normal temperature, it is not alerted

Test case 2



Due to gas leakage in industries, this circuit model is alerting the worker through their mobile phone. It will be monitor and rectify by workers in industries.

Name	Quantity	Component
GAS1	1	Gas sensor
PIEZO1	1	Piezo
M1	1	DC motor
S2	1	Push button
D1	1	Red LED
Rpot2	1	250 kΩ potentiometer
R1,R3	2	1kΩ resistor
R2	1	330Ω resistor
U2	1	LCD 16X2
U3	1	Arduino Uno R3

Arduino Uno R3
One type of ATmega328P-based microcontroller board is the Arduino Uno R3. It comes with everything needed to support the microcontroller; all you need to do is use a USB cable to connect it to a computer and provide power using an AC-DC adapter or a battery to get things going. The name Uno, which in the language of "Italian" means "one," was chosen to commemorate the launch of the Arduino IDE 1.0 software. The third and most recent version of the Arduino Uno is called the R3. The reference versions of Arduino are the board and the IDE software, both of which have recently undergone updates.

Bread Board
A breadboard is a common tool for circuit design and testing. Using a bread board eliminates the necessity for soldering wires and components together to form a circuit. Component mounting and reuse is simpler. Components are not soldered together, allowing for easy circuit design changes at any time.

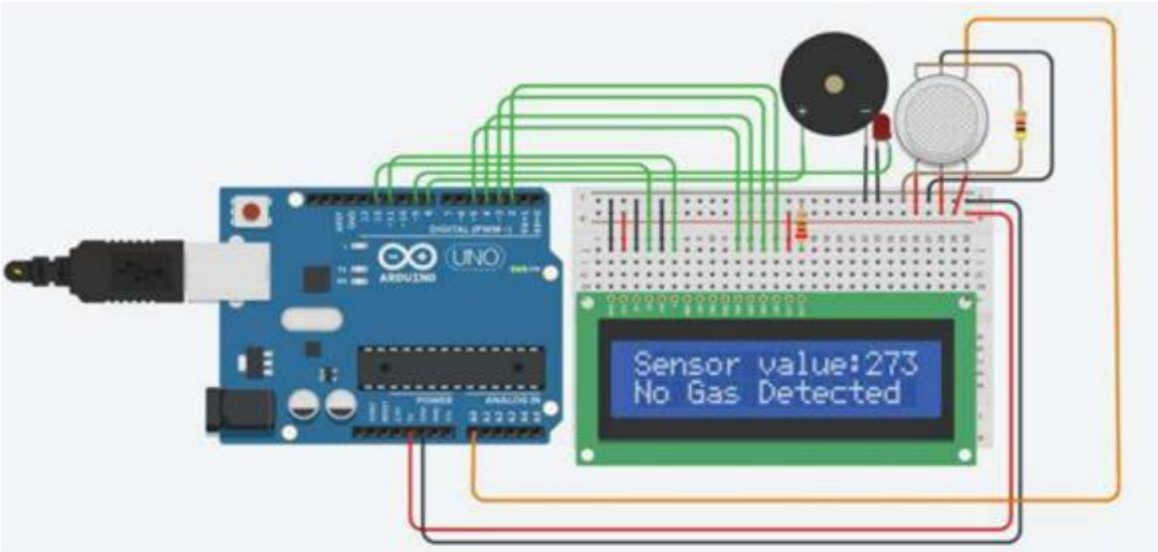
LED
An optoelectronic LED (Light Emitting Diode) operates on the electro-luminance principle. The ability of a substance to transform electrical energy into light energy and then emit that light energy is known as electro-luminance. The semiconductor in an LED operates similarly, emitting light when an electric field is present.

Resistor
A passive electrical device having two terminals that controls or limits the flow of electricity in electrical circuits.

Gas Sensor
A device that detects the presence or concentration of gases in the atmosphere is called a gas sensor. The sensor generates a corresponding potential difference based on the gas concentration by altering the material's resistance, which may be observed as output voltage. The type and concentration of the gas can be inferred from this voltage value.

LCD 16*2
One type of electronic gadget utilised to display the message and data is a 162 LCD. Liquid Crystal Display is the term's full name. Because it has 16 Columns and 2 Rows, the display is known as a 162 LCD. It can display a total of (16 + 2) 32 characters, each of which is composed of 5 x 8 pixels.

CIRCUIT DIAGRAM



8.2 User Acceptance Testing

Purpose of Document

Accidents caused by gas leaks can cause both material loss and human injuries. Based on their physical characteristics, such as toxicity, flammability, etc., the risk of explosion, fire, and asphyxia exists. In recent years, there have been an increasing number of fatalities brought on by gascylinder explosions.

Testing this project results in a dependable performance and identifies both its strengths and weaknesses. The document's goal is to do an accurate analysis by taking into account each parameter and providing a value.

Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By design	3	3	4	4	14
Duplicate	1	0	2	0	3
External	8	2	2	0	12
Fixed	10	3	1	24	38
Not reproduced	0	0	2	0	2
Skipped	0	1	1	2	4
Won’t Fix	0	4	1	2	4
Total	22	13	13	32	78

Test Case Analysis:

Section	**Total cases**	**Not Tested**	**Fail**	**Pass**
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	2	0	0	2
Exception Reporting	7	0	0	7
Final Report Output	6	0	0	6
Version Control	1	0	0	1

CHAPTER-9
RESULTS

9.1 Performance Metrics

Below image represents the result of node red dash board

The next Image below here represents the output of MIT-inventor

CHAPTER-10
ADVANTAGES & DISADVANTAGES

Advantages

1. This project is useful for finding gas leaks in industrial settings.
2. Equipment and Components are more effective.
3. Low power usage, and trustworthy.
4. Finding LPG gas leaks in the home is another use for it.

Disadvantages

1. Since there are numerous locations that house gas cylinders, the exact location of the leak cannot be determined.
2. Temperature and relative humidity both affect how sensitive it is.
3. Setup won't operate and function without the Internet.
4. Installation of the setup is challenging.

CHAPTER-11
CONCLUSION

Gas leakage leads to severe accidents resulting in material losses and human injuries .Gas leakage occurs due to poor maintenance of equipment and inadequate awareness of the people. Hence, gas leakage detection is essential to prevent accidents and to save human lives. This paper presented LPG leakage detection and alert system. This system triggers buzzer and notification to alert people when gas leakage is detected. This system is basic yet reliable**.**

CHAPTER-12
FUTURE SCOPE

A . Extended Features of System

The behaviour of the gases is dependent on the temperature and humidity of the air around. A gas at certain concentration might not be flammable at low temperature but might have explosive nature at high temperature. For this reason addition of a Temperature and Humidity Sensor will be very helpful.

B . Performing Big Data Analytics on the sensor readings

Analytics could be performed on the sensor readings. The readings from sensors could be used for forming predictions of situations where there can be a mishap. Instead of straightaway alarming when the concentrations have gone high, algorithms could be worked upon which could determine such situations prior to their occurrence. Combining the gas sensor readings with the readings from temperature and humidity sensor would increase the precision of the system. The cases of false alarms being raised will reduce down to very small percentages.

C. Dedicated Application for System

A dedicated mobile application could be made for the system. The features of the application would be:

1 . Getting the details of the concentration levels of the house within a tap of a button.

2. Since it is a safety device it is important for it to be perfectly calibrated and maintained tallies. The app can make sure to send reminders about getting the system checked every once in a while.

CHAPTER-13
APPENDIX

Source Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "vh5b9d"
deviceType = "Raspberry"
deviceId = "0330"
authMethod = "token"
authToken = "123456789"
# Initialize GPIO
def myCommandCallback(cmd):
   print("Command received: %s" % cmd.data['command'])
   status=cmd.data['command']
   if status=="alarmon":
      print ("Alarm is on")
   elif (status == "alarmoff") :
      print ("Alarm is off")
   elif status == "sprinkleron":
      print("Sprinkler is OFF")
   elif status == "sprinkleron":
      print("Sprinkler is ON")
   #print(cmd)
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)
      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 %%" % Humid, "Gas_Level = %s %%" %gas, "to IBM Watson")
        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Github Link:
https://github.com/IBM-EPBL/IBM-Project-48515-1660808361