GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR IBM INDUSTRIES

IBM PROJECT REPORT

TEAM MEMBERS : M. JOTHI

M. KOWSALYA

S. AKSHAYA

S. RAMYA RAJA YAZHINI

TEAM ID : PNT2022TMID18250

in partial fulfillment for the award

of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

MEPCO SCHLENK ENGINEERING COLLEGE, SIVAKASI

Table of Contents

1. INTRODUCTION

A) Project Overview:

This project helps the industries in monitoring the emission of harmful gases. In several areas, the integration of gas sensors helps in monitoring 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.

B) Purpose:

Inhaling concentrated gas can lead to asphyxia and possible death.To overcome these disasters, we designed a system for monitoring and alerting the leakage of those harmful gases. This makes the industrialists get rid of the fear of any disasters caused by the gases.

2. LITERATURE SURVEY

A) Existing Problem:

This scheme is meant to fulfill the daily needs of the people. In our country 40 percent people die due to gas explosion at home. That number keeps growing. Even pregnant women and small children are affected.Using a GSM module and a mobile phone, the Gas Leakage Monitor is used to find, intimate leaks. The buzzer and LED are then activated after the gas leak is detected using a bracket sensor. When the designated time has passed, it will automatically turn off.

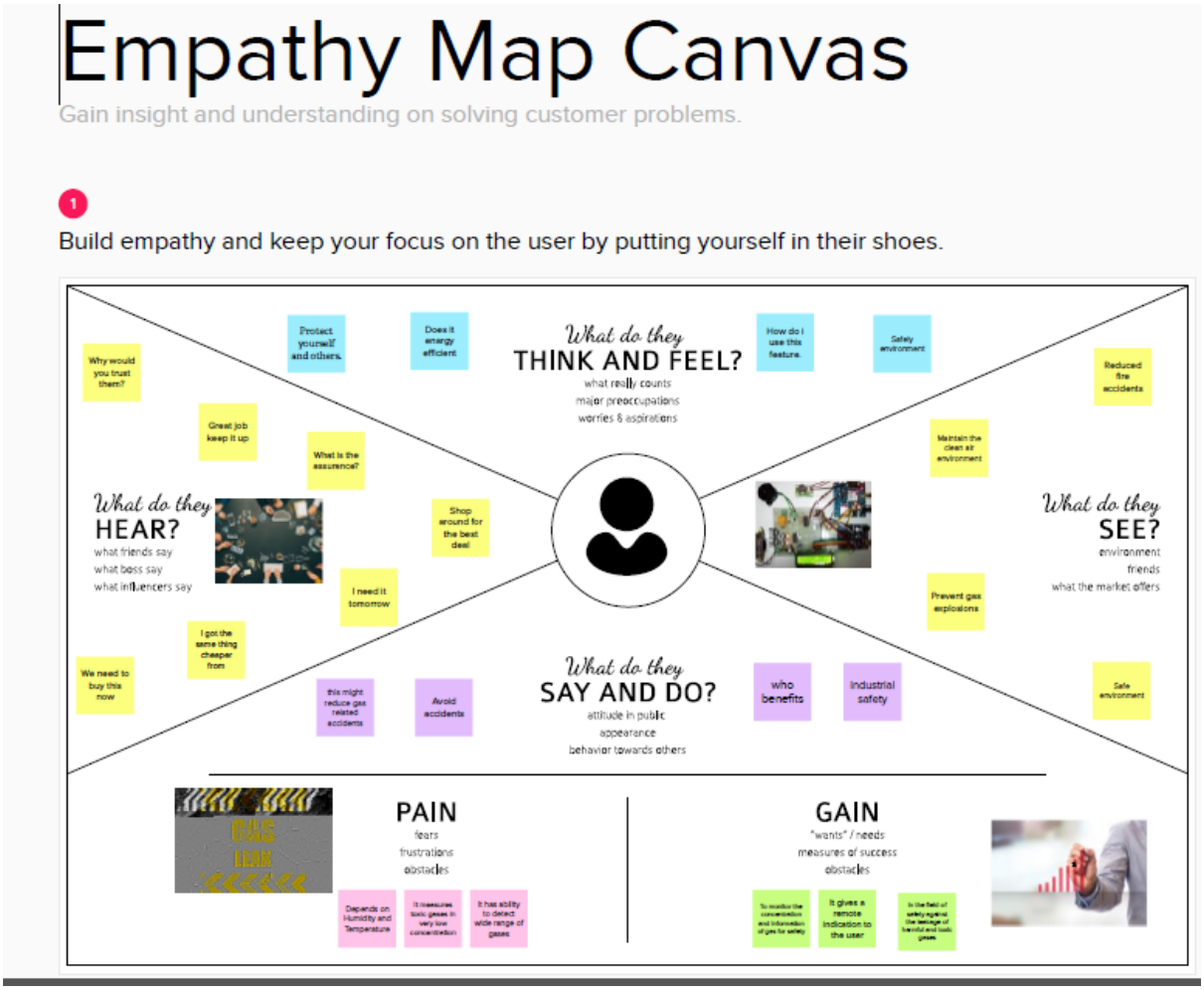
B) Problem statement definition:

This device does not get damaged very quickly, and if it does get damaged, water is the main reason for it. This device is easily damaged by water. Therefore, this device should be installed in a place where water does not go. This installation will not damage the device, if the device does, an example is water. This tool is considered to be one and very safe

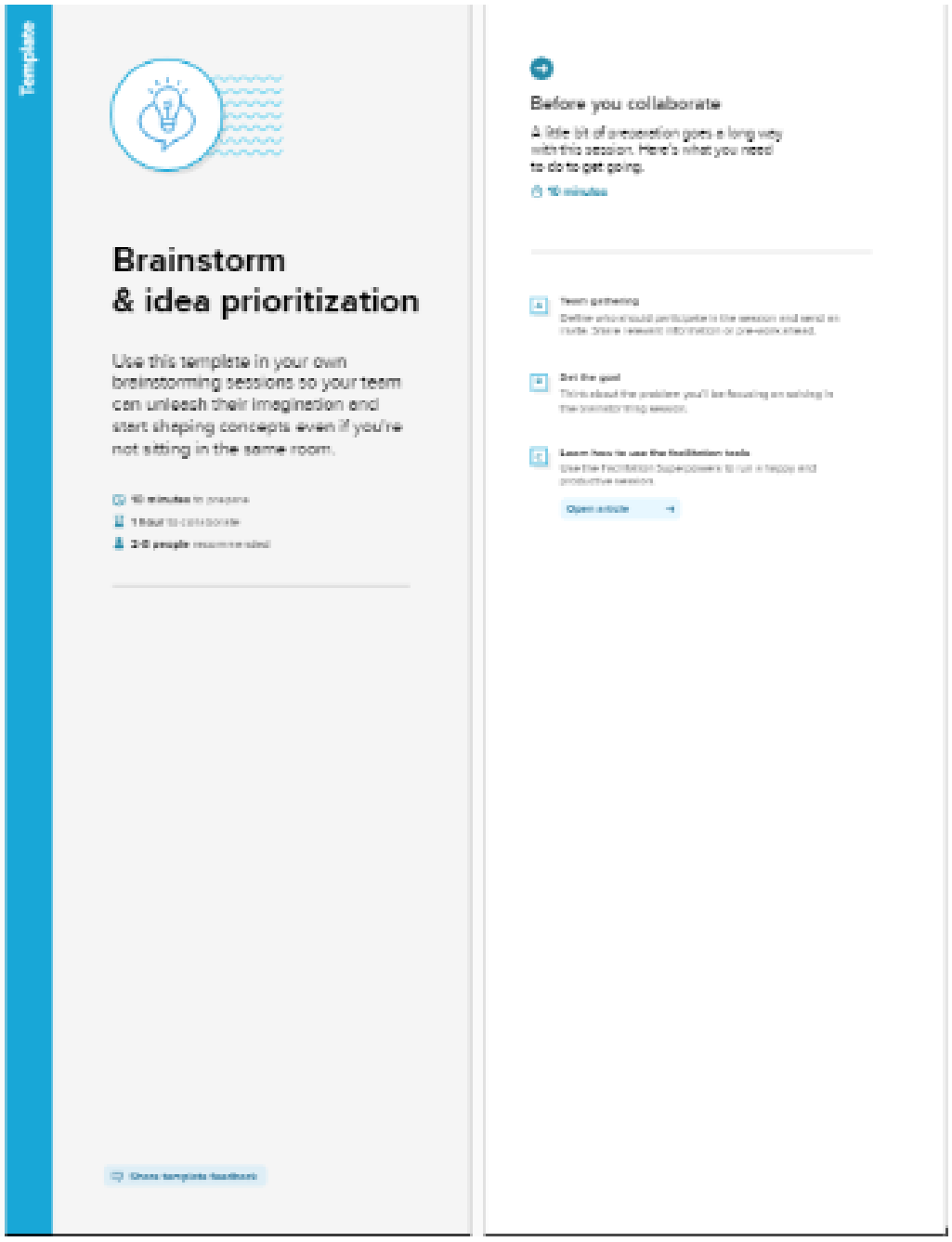
for the users. My members are trusted. My members' invention is considered very safe for this country and its people and their families. Absence of this tool makes women in our country nervous by the spread of gas in their kitchen and is considered to be a sign of some accident. It is also proud to think of this project for people's lives only to eliminate this fear.

3. IDEATION AND PROPOSED SOLUTION

A) Empathy map and canvas



B) Ideation and Brainstroming:



2

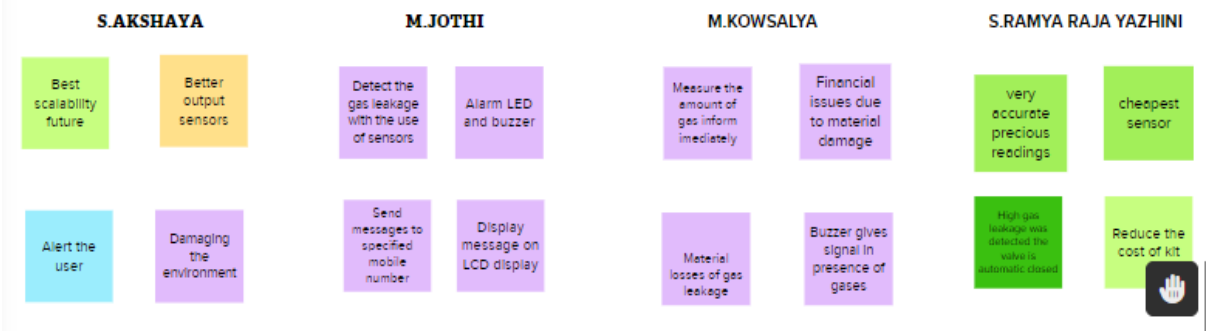
Brainstorm

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!



3

Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

C) Proposed solutions:

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 sensors and indicate via alarms and lights.
6.	Scalability of the Solution	Accuracy. Low cost. Less maintenance. Reliability.

D) Propose solution fit:

Project Design Phase-I - Solution Fit **Team ID:** PNT2022TMID18250

1. CUSTOMER SEGMENT(S) CS
The industrialists who use gases for their manufacturing.

6. CUSTOMER CONSTRAINTS
High budget in installing other productsmake them to move far from modern technologies.

5. AVAILABLE SOLUTIONS
The monitoring and controlling of the leakage could be done by the manpower. Even though man power could reduce electricity cost

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

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
When the workers failed to monitor properly, the gas can cause high risk to their health or the properties of the industry.

7. BEHAVIOUR 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 J&P, tap into BE, understand RC

3. TRIGGERS
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 BEHAVIOUR
CH
8.1 ONLINE
Promoting through social media. With the help of social media entrepreneurs/influencer.
8.2 OFFLINE
Through newspaper advertisements.

4. REQUIREMENT ANALYSIS

A) Functional Requirements:

FR No.	Functional Requirement (EPIC)	Sub Requirement (Story / Sub-Task)
FR-1	Create cloud account	Registration through Form Registration through G mail Registration through Link
FR-2	User Confirmation	Confirmation via Email Confirmation via OT
FR-3	User Login	User Login Via Mail id And Password
FR-4	Cloud registration	Connect the hardware device
FR-5	Connect to mobile	Connect the cloud with the mobile phone
FR-6	Connect Hardware	Connect hardware to the gas cylinders or in the wall

B) Non-functonal Requirements:

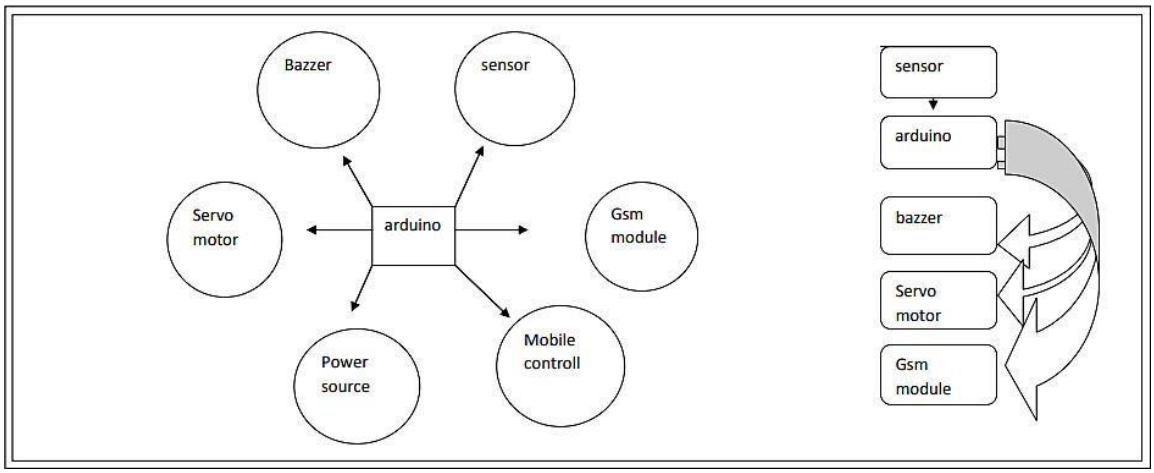
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	• It is user friendly • Easy to handle • Process is simple
NFR-2	Security	• The device is highly secure. • Privacy is maintained

NFR-3	Reliability	● The device is more reliable ● The device is more trustable in tough conditions
NFR-4	Performance	● The performance is more accurate. ● It is a real time application
NFR-5	Availability	● It can be available easily. ● It requires very few hardware components.
NFR-6	Scalability	● Less maintenance. ● Low cost. ● Compact.

5. PROJECT DESIGN

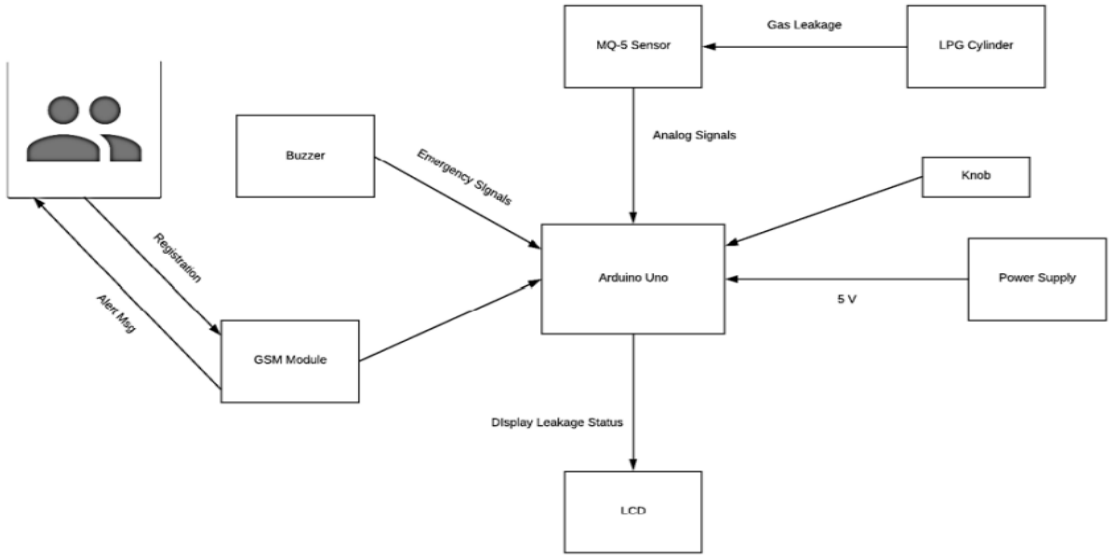
A) Data Flow Diagram:

Data Flow Diagram:



B) Solution and Technical architecture:

Solution Architecture Diagram:



C) 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 register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1

		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	High	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can access by message	Mediu m	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	No need to login mail id	High	Sprint-1
Customer (Web user)	Dashboar d					
Customer Care Executive	Registera ion	USN-1	User want to use sim card and gsm module	Message reciver for arduino connected sim	High	Sprint-1
		USN-1	Sensor and module	All component is connected to	High	Sprint-1
				arduino		
Administrator	DATA	USN-1	What is Main component in this project	Arduino	High	Sprint-2
power	Dc	USN-1	How much power is required	9v is enff	High	Sprint-1

6. PROJECT PLANNING AND SCHEDULING

A) Sprint planning and estimation :

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members

Sprint-1	Data Preparation & Data Visualization	USN-1	As a user, I provide Safety to the customers	5	High	M.JOTHI
Sprint-1		USN-2	As an Analyst, I collect the data & Provide meaningful insights through IBM Cloud	3	High	M.KOWSALYA
Sprint-2	Dashboard	USN-3	As a user, I want to make sure the safe environment.	3	High	S.AKSHAYA
Sprint-2		USN-4	As an Analyst, I will upload the data in IBM Cloud to createa interactive dashboard	3	Medium	RAMYA RAJA YAZHINI S
Sprint-3	Report	USN-5	As a user, I want to secure the lives and data of each employee that report a particular event	3	Medium	M.JOTHI
Sprint3		USN-6	As an Analyst, I will use IBM Cloud to generate a report	3	Medium	M.KOWSALYA
Sprint4	Story	USN-7	As a user, I can only understand the Analysis in animated presentation of dataset	5	Medium	S.AKSHAYA

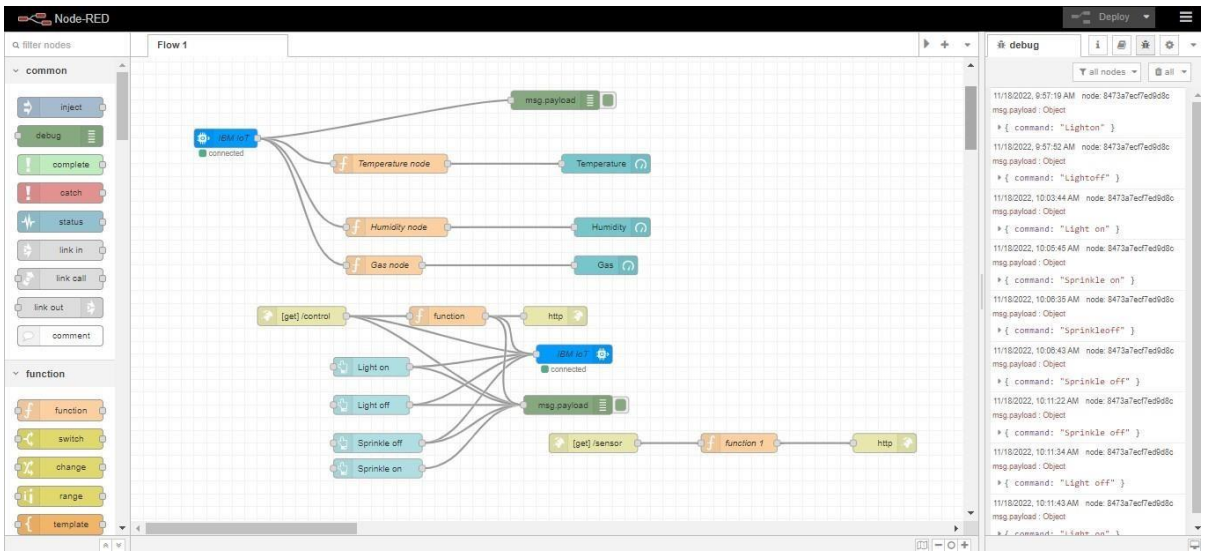
Sprint4		USN-8	As an Analyst, I use IBM to create an animated presentation (Story) of the dataset	3	High	S.RAMYA RAJA YAZHINI

B) Sprint delivery schedule:

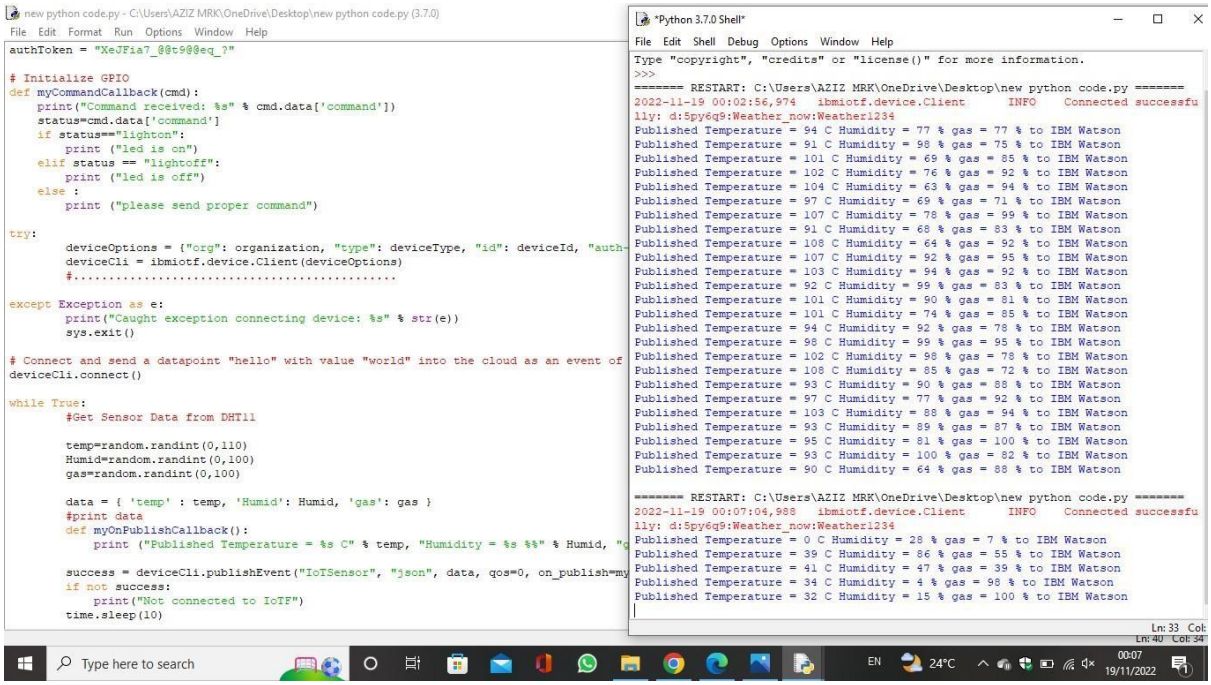
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	5	6 Days	24 Oct 2022	24 Oct 2022	5	29 Oct 2022
Sprint-2	5	6 Days	31 Oct 2022	05 Nov 2022	5	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	5	6 Days	14 Nov 2022	19 Nov 2022	5	15 Nov 2022

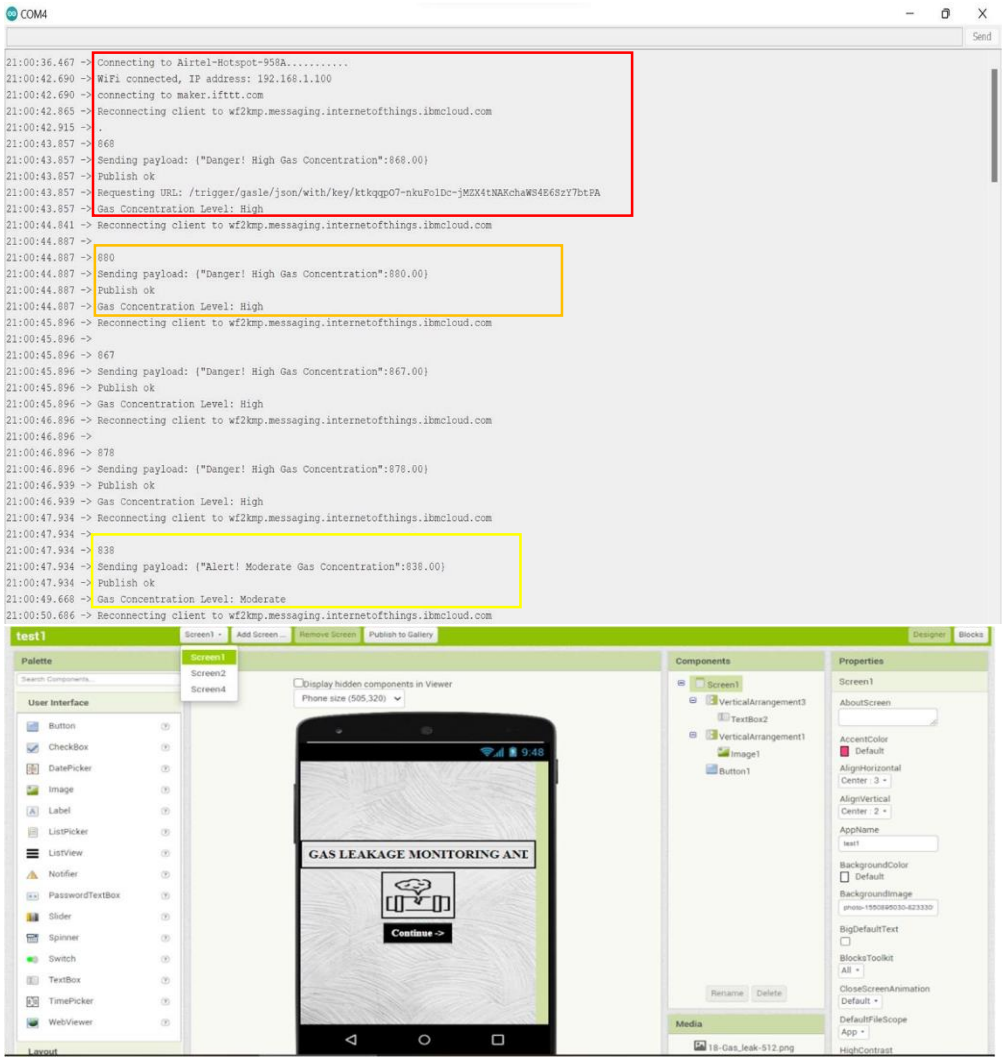
7. CODING AND SOLUTIONING

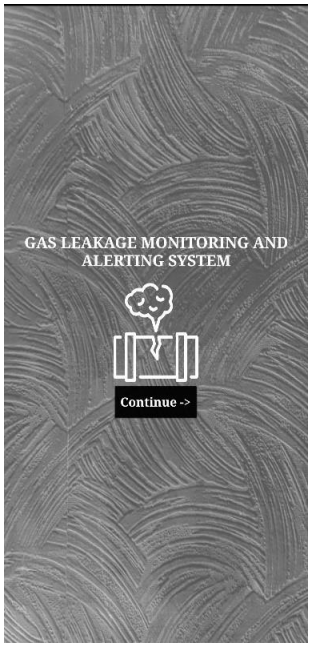
A) Feature 1:

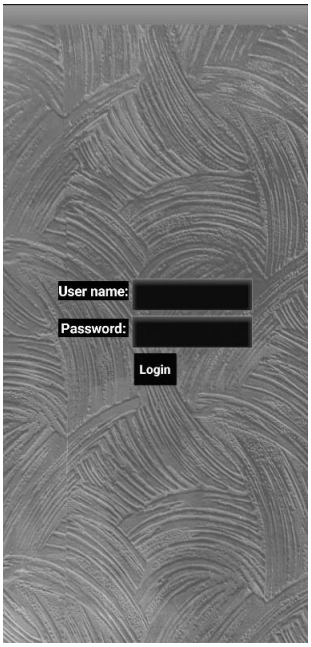


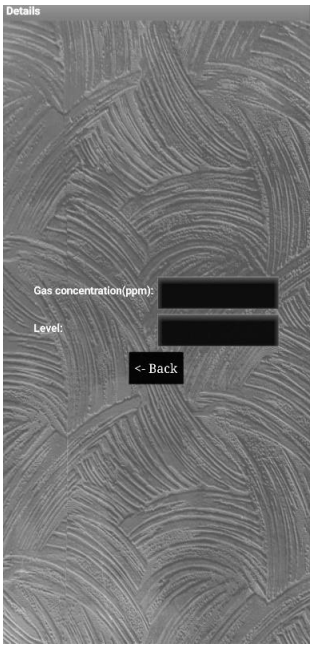
B) Feature 2: PYTHON OUTPUT




COM4
Send
21:00:36.467 -> Connecting to Airtel-Hotspot-958A.....
21:00:42.690 -> WiFi connected, IP address: 192.168.1.100
21:00:42.690 -> connecting to maker.ifttt.com
21:00:42.865 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
21:00:42.915 -> .
21:00:43.857 -> 868
21:00:43.857 -> Sending payload: {"Danger! High Gas Concentration":868.00}
21:00:43.857 -> Publish ok
21:00:43.857 -> Requesting URL: /trigger/gasle/json/with/key/ktkqqpO7-nkuFo1Dc-jMZX4tNAKchaWS4E6SzY7btPA
21:00:43.857 -> Gas Concentration Level: High
21:00:44.841 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
21:00:44.887 ->
21:00:44.887 -> 880
21:00:44.887 -> Sending payload: {"Danger! High Gas Concentration":880.00}
21:00:44.887 -> Publish ok
21:00:44.887 -> Gas Concentration Level: High
21:00:45.896 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
21:00:45.896 ->
21:00:45.896 -> 867
21:00:45.896 -> Sending payload: {"Danger! High Gas Concentration":867.00}
21:00:45.896 -> Publish ok
21:00:45.896 -> Gas Concentration Level: High
21:00:46.896 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
21:00:46.896 ->
21:00:46.896 -> 878
21:00:46.896 -> Sending payload: {"Danger! High Gas Concentration":878.00}
21:00:46.939 -> Publish ok
21:00:46.939 -> Gas Concentration Level: High
21:00:47.934 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
21:00:47.934 ->
21:00:47.934 -> 838
21:00:47.934 -> Sending payload: {"Alert! Moderate Gas Concentration":838.00}
21:00:47.934 -> Publish ok
21:00:49.668 -> Gas Concentration Level: Moderate
21:00:50.686 -> Reconnecting client to wf2kmp.messaging.internetofthings.ibmcloud.com
test1
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Palette
User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Screen1
Screen2
Screen4
Display hidden components in Viewer
Phone size (505,320)
9:48
GAS LEAKAGE MONITORING ANI
Continue ->
Components
Screen1
VerticalArrangement3
TextBox2
VerticalArrangement1
Image1
Button1
Rename
Delete
Media
18-Gas_leak-512.png
Properties
Screen1
AboutScreen
AccentColor
Default
AlignHorizontal
Center : 3
AlignVertical
Center : 2
AppName
test1
BackgroundColor
Default
BackgroundImage
BigDefaultText
BlocksToolkit
All
CloseScreenAnimation
Default
DefaultFileScope
App
HighContrast


GAS LEAKAGE MONITORING AND ALERTING SYSTEM
Continue ->


User name:
Password:
Login


Details
Gas concentration(ppm):
Level:
<- Back

8. ADVANTAGES AND DISADVANTAGES

Advantages:

- Detect the concentration of the gases
- The sensor-enabled solution helps prevent the high risk of gas explosions and affecting any casualties within and outside the premises.
- Get real-time alerts about the gaseous presence in the atmosphere
- Prevent fire hazards and explosions
- Ensure worker's health
- Real-time updates about leakages
- Measure oxygen level accuracy
- Get immediate gas leak alerts
- Cost-Effective Installation

Disadvantage:

- Get immediate gas leak alerts
- When heavy dust, steam or fog blocks the laser beam, the system will not be able to take measurements

9. 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.

10. FUTURE SCOPE

Major cities of India are pushing Smart Home application, gas monitoring system is a part of SmartHome application. Enhancing Industrial Safety using IoT. This system can be implemented in Industries, Hotels and wherever the gas cylinders are used. This system can be used in industries involving applications such as Furnace, Boilers, Gas welding, Gas cutting, Steel Plants, Metallurgical industries, Food processing Industries, Glass Industries, Plastic industries, Pharmaceuticals, Aerosol manufacturing. As hospitals require to provide maximum possible safety to patients, this system can be used to keep track of all the cylinders used in it. Some of the cylinders used are Oxygen cylinder, Carbon dioxide cylinder, Nitrous oxide cylinder. As many students are naive the risk of causing accidents is high. Hence, our system can also be used in schools, colleges. Many colleges have well established labs including chemistry lab and pharmaceutical labs where gas burners are used. Several medical equipment requires gas cylinders.

11. **APPENDIX**

A) Source code:

```
#include <ESP8266WiFi.h> #include <LiquidCrystal_I2C.h> #include <PubSubClient.h>


#define Buzzer D5 #define Green D6 #define Sensor A0 #define ORG "wf2kmp"

#define DEVICE_TYPE "GLMASFI_IOT_Device_Cloud_Service" #define DEVICE_ID "PNT2022TMID35867"

#define TOKEN "PNT2022TMID35867"


const char* ssid = "Airtel-Hotspot-958A"; const char* password = "9889i1bb";

const char* host = "maker.ifttt.com"; const int httpsPort = 80;

char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; char topic[] = "iot-2/evt/status/fmt/json";

char authMethod[] = "use-token-auth"; char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```
void PublishData(float);

void callback(char*, byte*, unsigned int);



LiquidCrystal_I2C lcd(0x27, 16, 2); WiFiClient client1;

WiFiClient client2;

PubSubClient client(server, 1883, callback, client2);


void setup() { lcd.backlight();


lcd.init();

pinMode(Green, OUTPUT); pinMode(Buzzer, OUTPUT); pinMode(Sensor, INPUT); Serial.begin(115200); Serial.println();


Serial.print("Connecting to "); Serial.print(ssid); WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) { delay(500);

Serial.print(".");
```

```
}

Serial.println("");

Serial.print("WiFi connected, IP address: "); Serial.println(WiFi.localIP()); Serial.print("connecting to "); Serial.println(host);

if (!client1.connect(host, httpsPort)) { Serial.println("connection failed"); return;

}

if (!client2.connect(host, httpsPort)) { Serial.println("connection failed"); return;

}


}


int msgSent = 0; void loop() {

Serial.print("Reconnecting client to "); Serial.println(server);

while (!client.connect(clientId, authMethod, token)) { Serial.print(".");

delay(500);

}
```

```
Serial.println();


int sensor = analogRead(Sensor); Serial.println(String(sensor)); PublishData(sensor);

if (sensor >= 840 && !msgSent) { digitalWrite(Green, HIGH); digitalWrite(Buzzer, HIGH);

String url = "/trigger/gasle/json/with/key/ktkqqpO7-nkuFo1Dc-jMZX4tNAKchaWS4E6SzY7btPA"; Serial.print("Requesting URL: ");

Serial.println(url);

client1.print(String("GET ") + url + " HTTP/1.1\r\n" + "Host: " + host + "\r\n" + "Connection: close\r\n\r\n");

msgSent = 1;

lcd.setCursor(0, 1);


Serial.println("Gas Concentration Level: High");

}

else if (sensor >= 840 && msgSent) { digitalWrite(Green, HIGH); digitalWrite(Buzzer, HIGH); lcd.setCursor(0, 1);

Serial.println("Gas Concentration Level: High");
```

```
}
else if (sensor>=820){ digitalWrite(Buzzer, HIGH); digitalWrite(Green, HIGH); delay(750); digitalWrite(Buzzer,LOW); digitalWrite(Green, LOW); delay(1000); lcd.setCursor(0, 1);
Serial.println("Gas Concentration Level: Moderate");
}
else{
digitalWrite(Green, LOW); digitalWrite(Buzzer, LOW); lcd.setCursor(0, 1);
Serial.println("Gas Concentration Level: Normal");
}
lcd.setCursor(0, 0); lcd.print("Value: "); lcd.print(sensor); delay(1000);
}

void callback(char* topic, byte* payload, unsigned int length) { Serial.println("callback invoked");
}

void PublishData(float senso){ String payload;
```

```
if(senso >= 840) {
payload= "{\"Danger! High Gas Concentration\":";
}
else if(senso >= 820) {
payload= "{\"Alert! Moderate Gas Concentration\":";
}
else {
payload = "{\"Normal Gas Concentration\":";
}
payload += senso; payload+="}";

Serial.print("Sending payload: "); Serial.println(payload);
if (client.publish(topic, (char*) payload.c_str())) {
Serial.println("Publish ok");
}
 else {
Serial.println("Publish failed");
}
}
```