GAS LEAKAGE MONITORING AND ALERTING SYSTEM

PROJECT REPORT

TEAM ID: PNT2022TMID14295

Project Report Format

1. **INTRODUCTION**
   1.1 Project Overview
   1.2 Purpose
2. **LITERATURE SURVEY**
   2.1 Existing problem
   2.2 References
   2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   3.1 Empathy Map Canvas
   3.2 Ideation & Brainstorming
   3.3 Proposed Solution
   3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   4.1 Functional requirement
   4.2 Non-Functional requirements
5. **PROJECT DESIGN**
   5.1 Data Flow Diagrams
   5.2 Solution & Technical Architecture
   5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
   6.1 Sprint Planning & Estimation
   6.2 Sprint Delivery Schedule
   6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   7.1 Feature 1
   7.2 Feature 2
   7.3 Database Schema (if Applicable)
8. **TESTING**
   8.1 Test Cases
   8.2 User Acceptance Testing
9. **RESULTS**
   9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX** Source Code
    GitHub Link

GAS LEAKAGE MONITORING AND ALERTING SYSTEM

1.INTRODUCTION:

Nowadays, home safety detection systems play a significant part in people's security. Since everyone in the household works every day, it is impossible to check on the household appliances, particularly the LPG gas cylinder, wired circuits, etc. Liquefied petroleum gas (LPG) and natural gas demand has significantly increased during the past three years. LPG and natural gas are recommended to meet this high level of energy demand and to substitute oil or coal due to those fuels' negative environmental effects. Large-scale applications for these gases include industry, heating, home appliances, and motor fuel. The system has a MQ6 gas sensor to monitor this leakage gas. This sensor detects how much leak gas is there in the environment around it.

Explosions or being harmed by gas leaks could be avoided in this way.

2.LITERATURE SURVEY:

Prof. M.Amsaveni, A.Anurupa, R.S.Anu Preetha, C.Malarvizhi, M.Gunasekaran; they told in their research paper on “GSM based LPG leakage detection and controlling system” the leakage of LPG gas is detected by the MQ-6 gas sensor. Its analog output is given to the microcontroller. It consists of predefined instruction set. Based on this, the exhaust fan is switched on. So, the concentration of gas inside the room gets decreased. Then, the stepper motor is rotated thus closing the knob of the cylinder. Because of this process, the leakage of gas is stopped. The relay is switched to off the power supply of the house. The buzzer produces an alarm to indicate the gas leakage. Then, the user is alerted by SMS through the GSM module.

B. B. Did paye, Prof. S. K. Nanda; in this paper they told about their research on leakage detection and review of"Automated unified system for LPG using microcontroller and GSM module". Their paper proposed an advance and innovative approach for LPG leakage detection,prevention and automatic booking for refill. In advance, the system provides the automatic controlling of LPG regulator also if leakage is detected the system will automatically turn off the main switch of power supply. Hence it helps to avoid the explosion and blast.

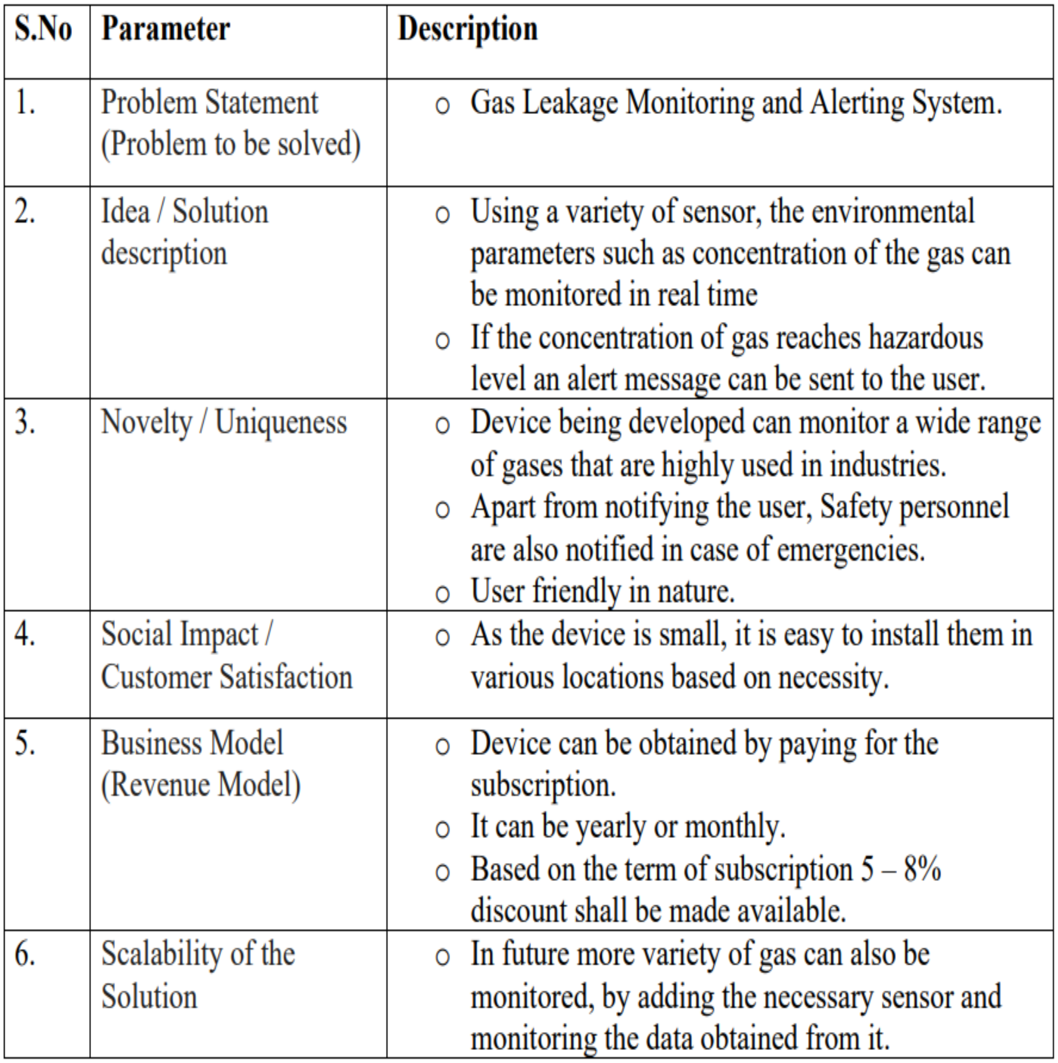
Srinivasan, Leela, Jeya bharathi, Kirthik,Rajasree; in this research paper they told about gas leakage detection and control. In this paper, the gas leakage resulting into fatal inferno has become a serious problem in household and other areas where household gas is handled and used. It alerts the subscriber through the alarm and the status display besides turning off the gas supply valve as a primary safety measure.

Hitendra Rawat, Ashish Kushwah, Khyati Asthana, Akanksha Shivhare,in the year 2014 planned a framework, They gave security issues against hoodlums, spillage and fire mishaps. In those cases their framework sends SMS to the crisis number gave to it.

P.Meenakshi Vidya, S.Abinaya, G.Geetha Rajeswari, N.Guna,"Automatic LPG detection and hazard controlling " published in April 2014 proposed the leakage detection and real time gas monitoring system. In this system, the gas leakage is detected and controlled by means of exhaust fan. The level of LPG in cylinder is also continuously monitored. Ch. Manohar Raju and N. Sushma Rani, 2008, they introduce an android based automatic gas detection and indication robot.

3.PROPOSED SOLUTION:

IDEATION PROPOSED SOLUTION:



S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	○ Gas Leakage Monitoring and Alerting System.
2.	Idea / Solution description	○ Using a variety of sensor, the environmental parameters such as concentration of the gas can be monitored in real time ○ If the concentration of gas reaches hazardous level an alert message can be sent to the user.
3.	Novelty / Uniqueness	○ Device being developed can monitor a wide range of gases that are highly used in industries. ○ Apart from notifying the user, Safety personnel are also notified in case of emergencies. ○ User friendly in nature.
4.	Social Impact / Customer Satisfaction	○ As the device is small, it is easy to install them in various locations based on necessity.
5.	Business Model (Revenue Model)	○ Device can be obtained by paying for the subscription. ○ It can be yearly or monthly. ○ Based on the term of subscription 5 – 8% discount shall be made available.
6.	Scalability of the Solution	○ In future more variety of gas can also be monitored, by adding the necessary sensor and monitoring the data obtained from it.

PROPOSED METHOD:

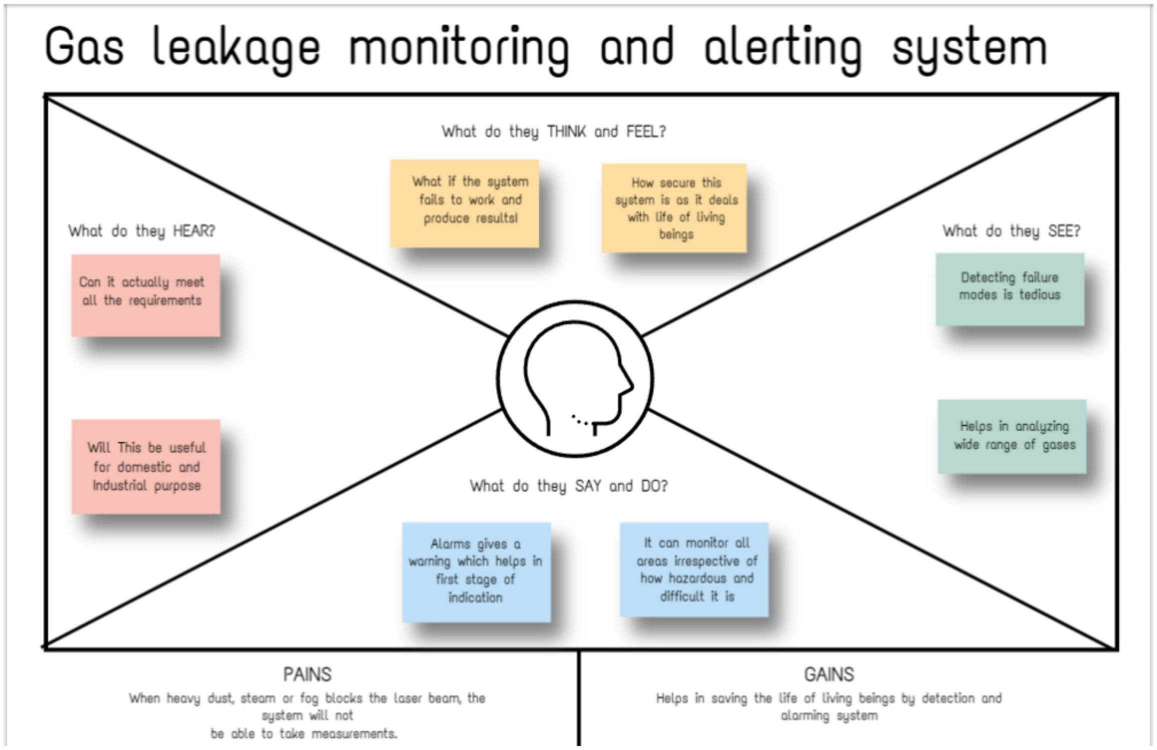
The core component of the system, the Arduino UNO (Atmega-328), carries out the following functions. The output signal of the sensor, which serves as input to Arduino, performs signal conditioning. Results of the detection were shown on LCD. warns individuals of risk at work, in factories, and at home. There is buzzer activity and a beep (siren) sound. Additionally, using a GSM modem, send an alarm SMS to the plant manager whose phone number is saved on the SIM card. The SMS you receive is based on whether there is a gas leak in the sensor's field of detection.

EMPATHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

- It is a useful tool to helps teams better understand their users.Creating an effective solution requires understanding the true problem and the person who is experiencing it.

- The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



Gas leakage monitoring and alerting system

What do they THINK and FEEL?

What if the system fails to work and produce results!

How secure this system is as it deals with life of living beings

What do they HEAR?

Can it actually meet all the requirements

Will This be useful for domestic and Industrial purpose

What do they SEE?

Detecting failure modes is tedious

Helps in analyzing wide range of gases

What do they SAY and DO?

Alarms gives a warning which helps in first stage of indication

It can monitor all areas irrespective of how hazardous and difficult it is

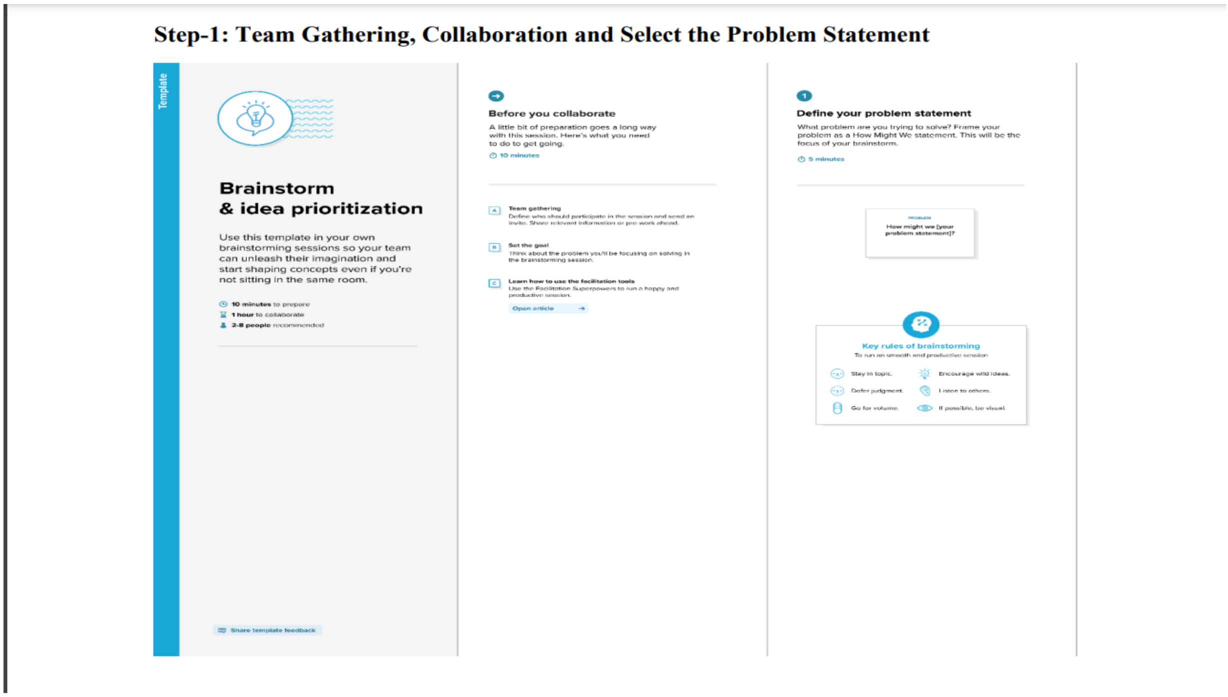
PAINS

When heavy dust, steam or fog blocks the laser beam, the system will not be able to take measurements.

GAINS

Helps in saving the life of living beings by detection and alarming system

IDEATION AND BRAINSTORMING:



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

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

1 **Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

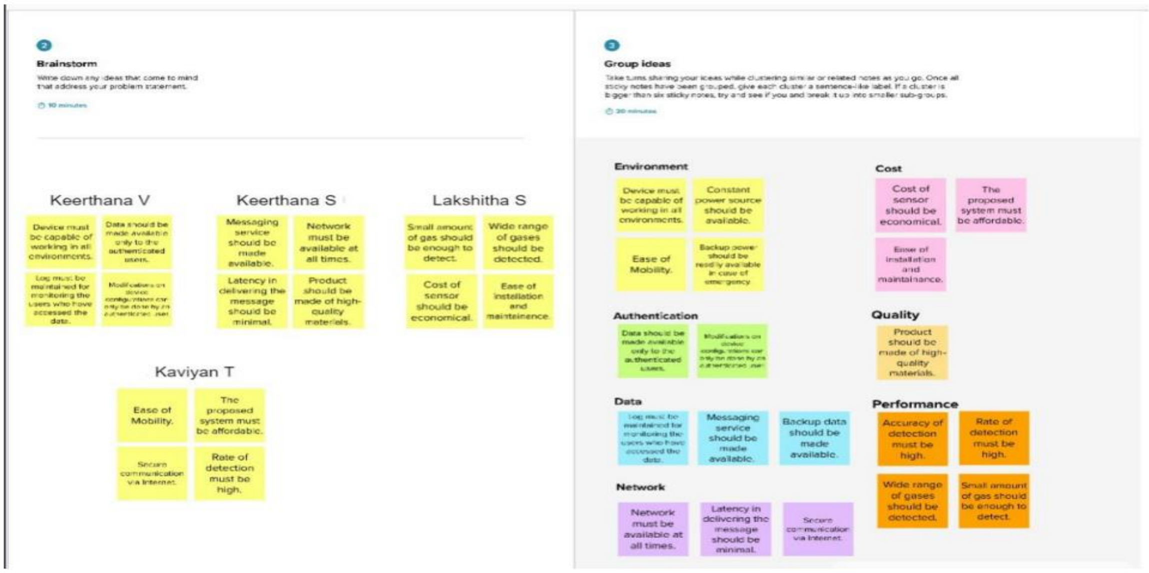
5 minutes

PROBLEM
How might we [your problem statement]?

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping



2

Brainstorm

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

10 minutes

Keerthana V

Device must be capable of working in all environments.

Data should be made available only to the authenticated users.

Log must be maintained for monitoring the users who have accessed the data.

Modifications on device configurations can only be done by an authenticated user.

Keerthana S

Messaging service should be made available.

Network must be available at all times.

Latency in delivering the message should be minimal.

Product should be made of high-quality materials.

Lakshitha S

Small amount of gas should be enough to detect.

Wide range of gases should be detected.

Cost of sensor should be economical.

Ease of installation and maintainence.

Kaviyan T

Ease of Mobility.

The proposed system must be affordable.

Secure communication via Internet.

Rate of detection must be high.

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

Environment

Device must be capable of working in all environments.

Constant power source should be available.

Ease of Mobility.

Backup power should be readily available in case of emergency

Cost

Cost of sensor should be economical.

The proposed system must be affordable.

Ease of installation and maintainance.

Authentication

Data should be made available only to the authenticated users.

Modifications on device configurations can only be done by an authenticated user

Quality

Product should be made of high-quality materials.

Data

Log must be maintained for monitoring the users who have accessed the data.

Messaging service should be made available.

Backup data should be made available.

Performance

Accuracy of detection must be high.

Rate of detection must be high.

Wide range of gases should be detected.

Small amount of gas should be enough to detect.

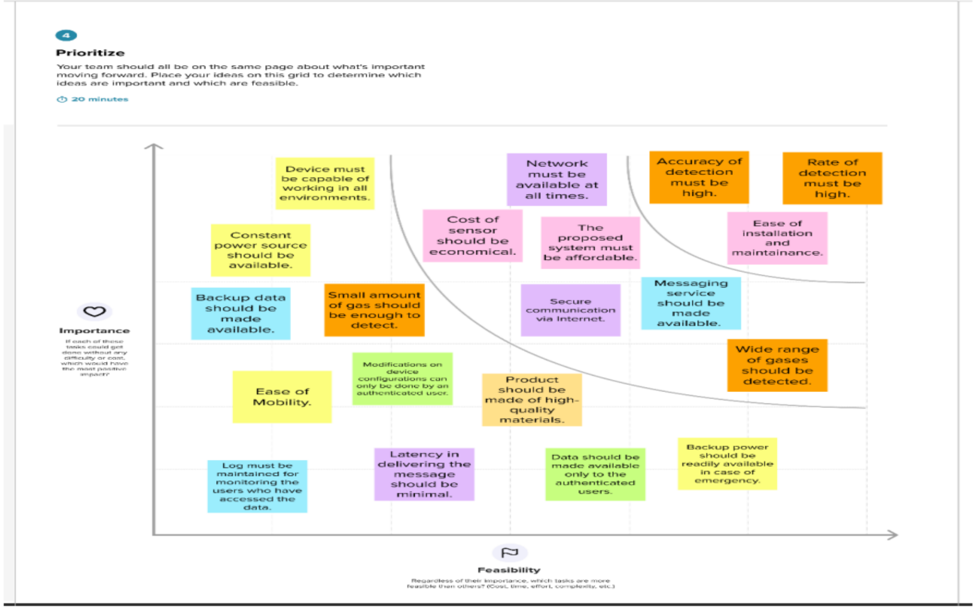
Network

Network must be available at all times.

Latency in delivering the message should be minimal.

Secure communication via Internet.

Step-3: Idea Prioritization



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

Device must be capable of working in all environments.

Network must be available at all times.

Accuracy of detection must be high.

Rate of detection must be high.

Cost of sensor should be economical.

The proposed system must be affordable.

Ease of installation and maintainance.

Constant power source should be available.

Backup data should be made available.

Small amount of gas should be enough to detect.

Secure communication via Internet.

Messaging service should be made available.

Wide range of gases should be detected.

Modifications on device configurations can only be done by an authenticated user.

Ease of Mobility.

Product should be made of high-quality materials.

Backup power should be readily available in case of emergency.

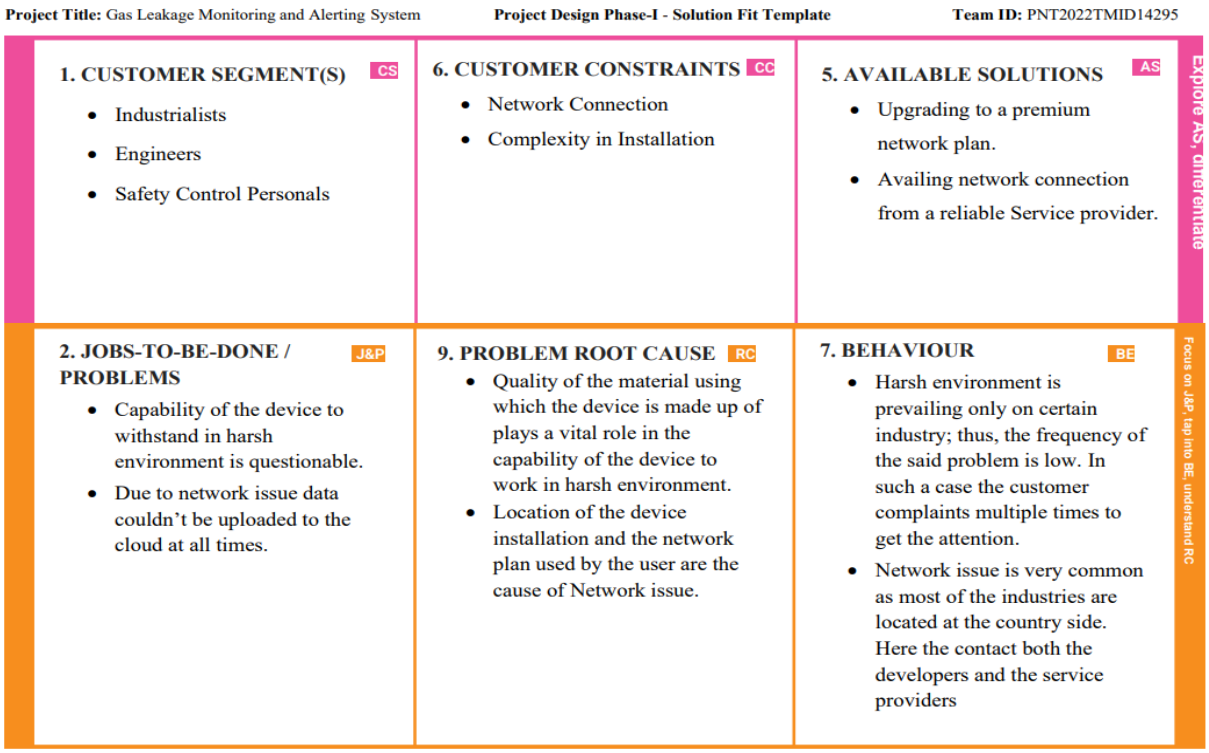
Log must be maintained for monitoring the users who have accessed the data.

Latency in delivering the message should be minimal.

Data should be made available only to the authenticated users.

Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

PROPOSED SOLUTION FIT:



Project Title: Gas Leakage Monitoring and Alerting System **Project Design Phase-I - Solution Fit Template** **Team ID:** PNT2022TMID14295

1. CUSTOMER SEGMENT(S) CS

- Industrialists
- Engineers
- Safety Control Personals

6. CUSTOMER CONSTRAINTS CC

- Network Connection
- Complexity in Installation

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

- Capability of the device to withstand in harsh environment is questionable.
- Due to network issue data couldn't be uploaded to the cloud at all times.

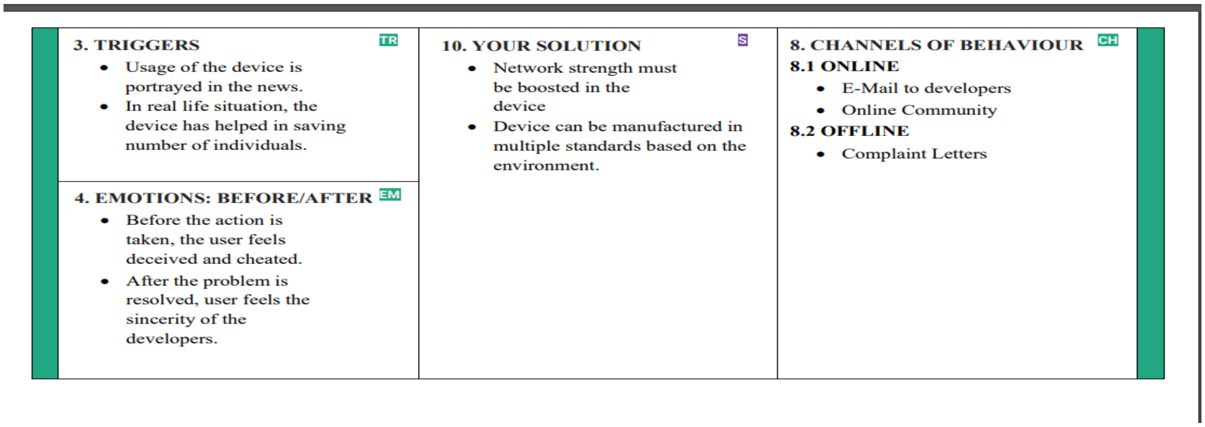
9. PROBLEM ROOT CAUSE RC

- 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

Explore AS, differentiate
Focus on J&P, tap into BE, understand RC



3. TRIGGERS TR

- Usage of the device is portrayed in the news.
- In real life situation, the device has helped in saving number of individuals.

4. EMOTIONS: BEFORE/AFTER EM

- Before the action is taken, the user feels deceived and cheated.
- After the problem is resolved, user feels the sincerity of the developers.

10. YOUR SOLUTION S

- Network strength must be boosted in the device
- Device can be manufactured in multiple standards based on the environment.

8. CHANNELS OF BEHAVIOUR CH

8.1 ONLINE

- E-Mail to developers
- Online Community

8.2 OFFLINE

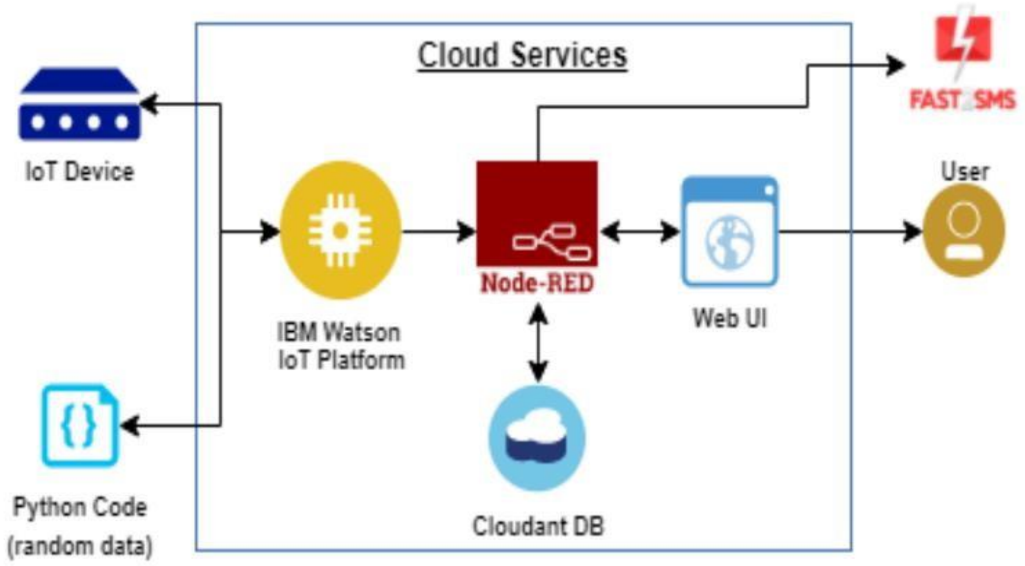
- Complaint Letters

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To detect the gas in the industry and prevent from making hazardous damages.
2.	Idea / Solution description	To detect the gas in the industry and prevent from making hazardous damages.
3.	Novelty / Uniqueness	Sensor automatically restores the gas and saves the data in IoT cloud.
4.	Social Impact / Customer Satisfaction	Customer feel great about this invention because, huge damage is prevented.
5.	Business Model (Revenue Model)	The cost is very low for manufacturing this product.
6.	Scalability of the Solution	It is more reliable and flexible.

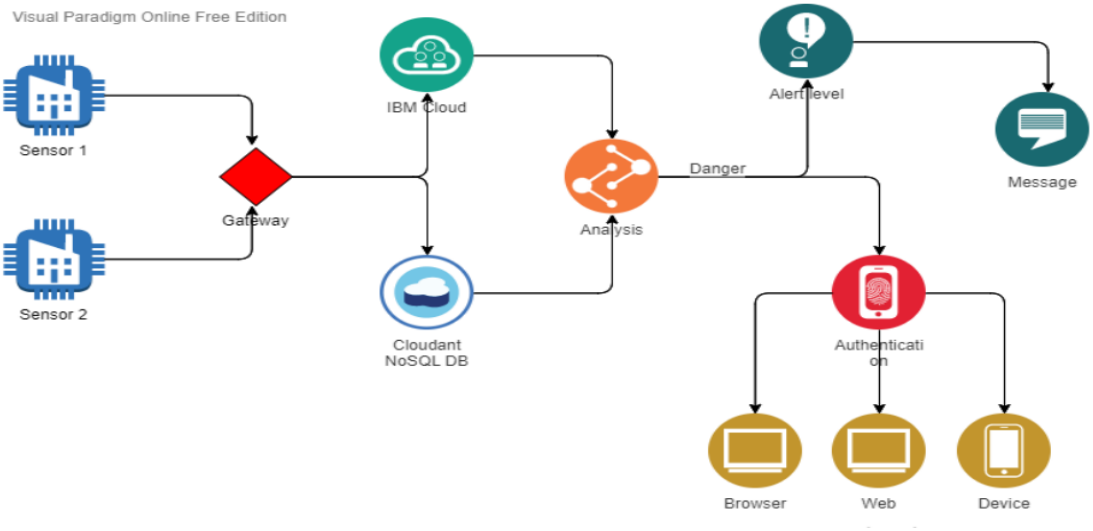
4.REQUIREMENT ANALYSIS:

SOLUTION ARCHITECTURE:

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 MQ sensors are not upto 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 used as a primary indicator of leakage inside a plant.



Solution Architecture Diagram:

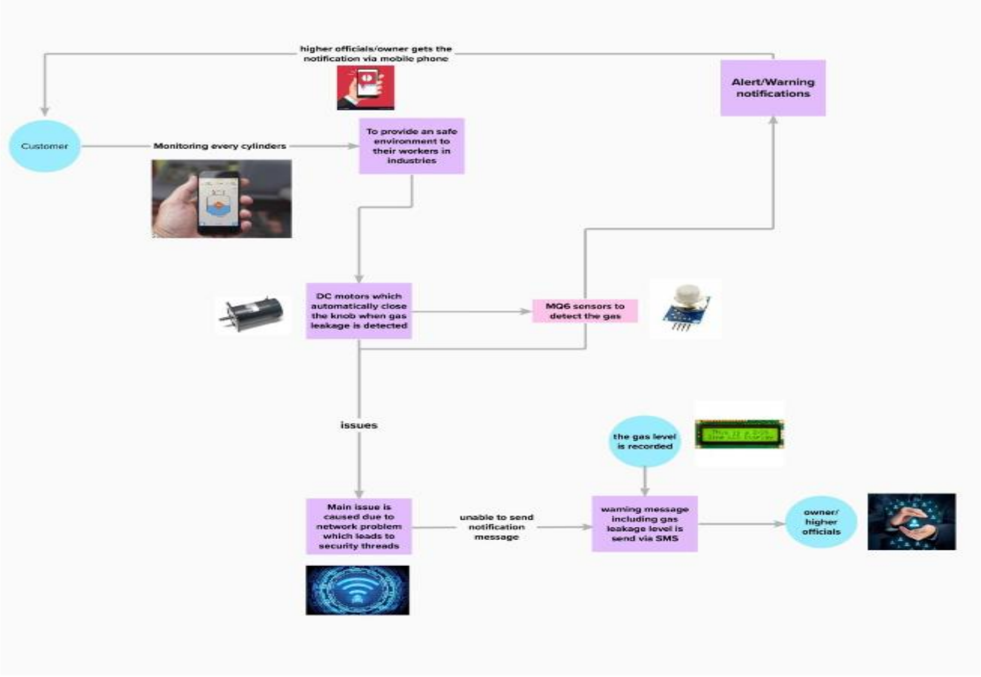


FUNCTIONAL REQUIREMENT:

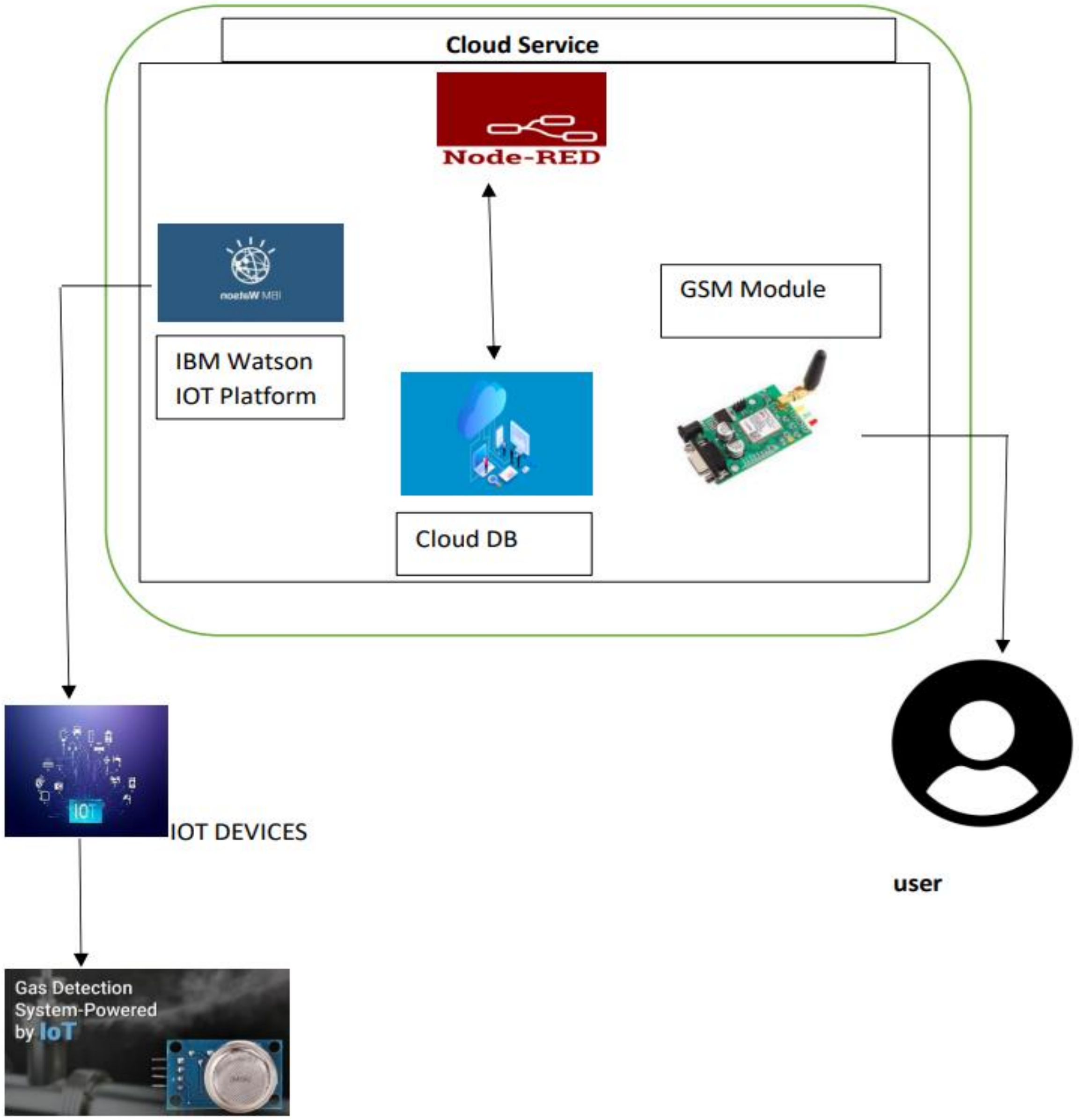
Business Requirements	User Requirements	Product Requirements
The mentioned system is usable in residences, hotels, industrial settings, LPG cylinder storage places, etc. The ability to detect leakage and transmit the information to a location is the primary benefit of this IoT and Arduino-based application. It is observable, and precautions can be taken to avert any catastrophe.	The gas leakage detection system can be upgraded with smoke and fire detectors to detect the presence of smoke and fire in addition as being optimised for detecting dangerous gases. Although ensuring worker safety is critical, adopting the appropriate technology is even more crucial.	Regardless of your professional position or personal goals, gas detection is essential. Such IoT devices are what they are due to certain technologies in use, therefore understanding these technologies and the functions they can serve is necessary if you want to engage in IoT application development.

5.PROJECT DESIGN:

DATA FLOW DIAGRAM:

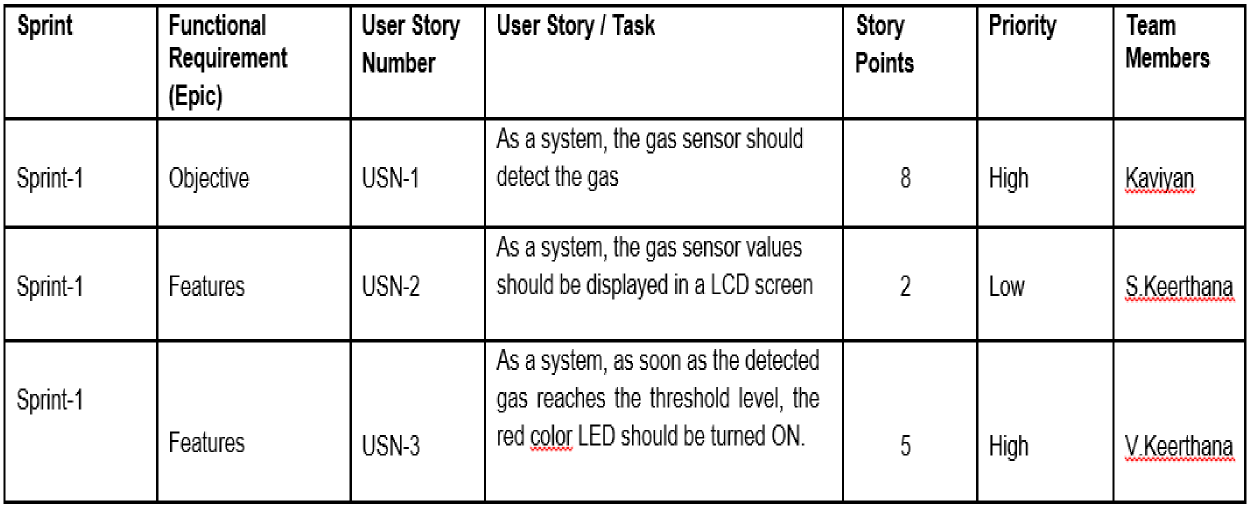
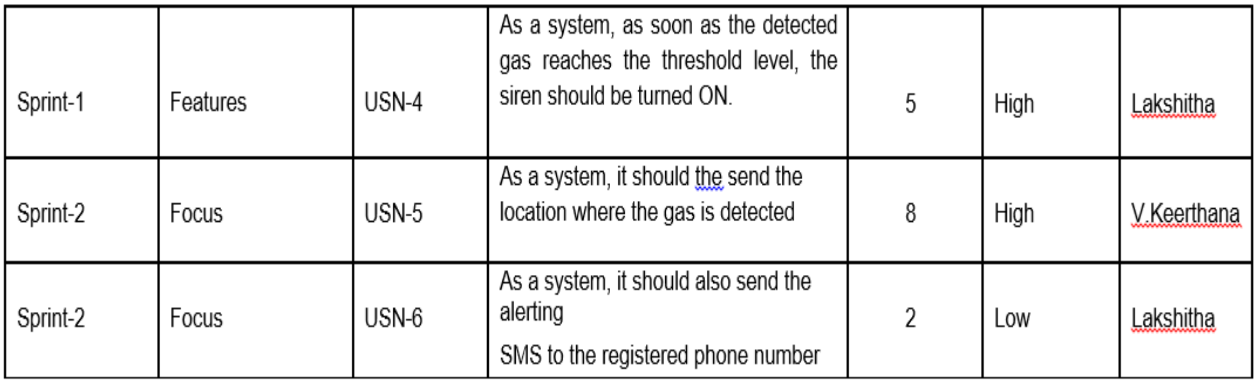

higher officials/owner gets the notification via mobile phone
Alert/Warning notifications
Customer
Monitoring every cylinders
To provide an safe environment to their workers in industries
DC motors which automatically close the knob when gas leakage is detected
MQ6 sensors to detect the gas
issues
the gas level is recorded
Main issue is caused due to network problem which leads to security threads
unable to send notification message
warning message including gas leakage level is send via SMS
owner/ higher officials

SOLUTION AND TECHNICAL ARCHITECTURE:

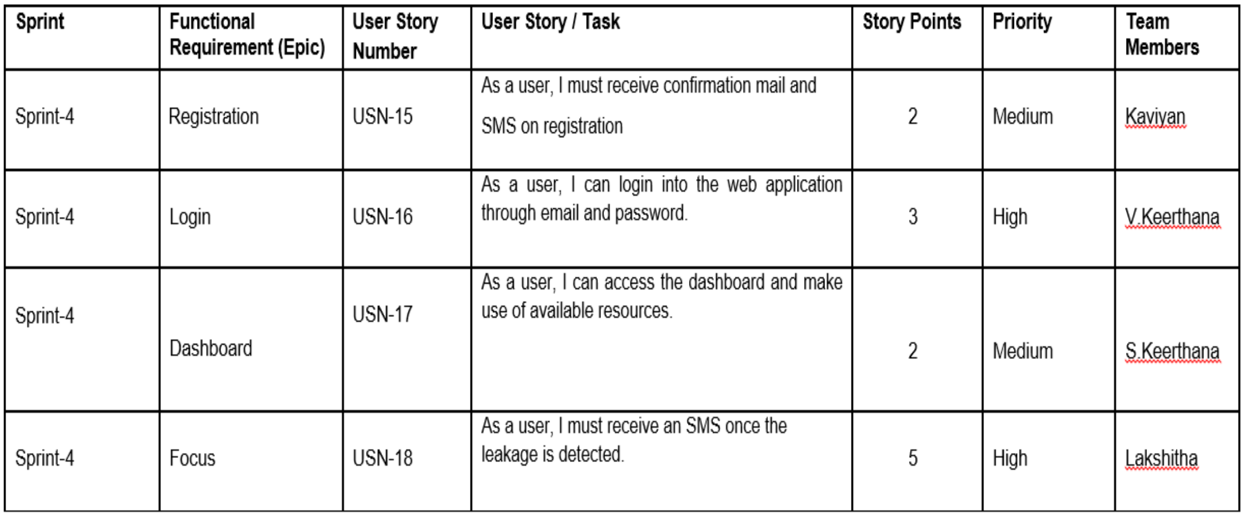


6.PROJECT PLANNING & SCHEDULING:

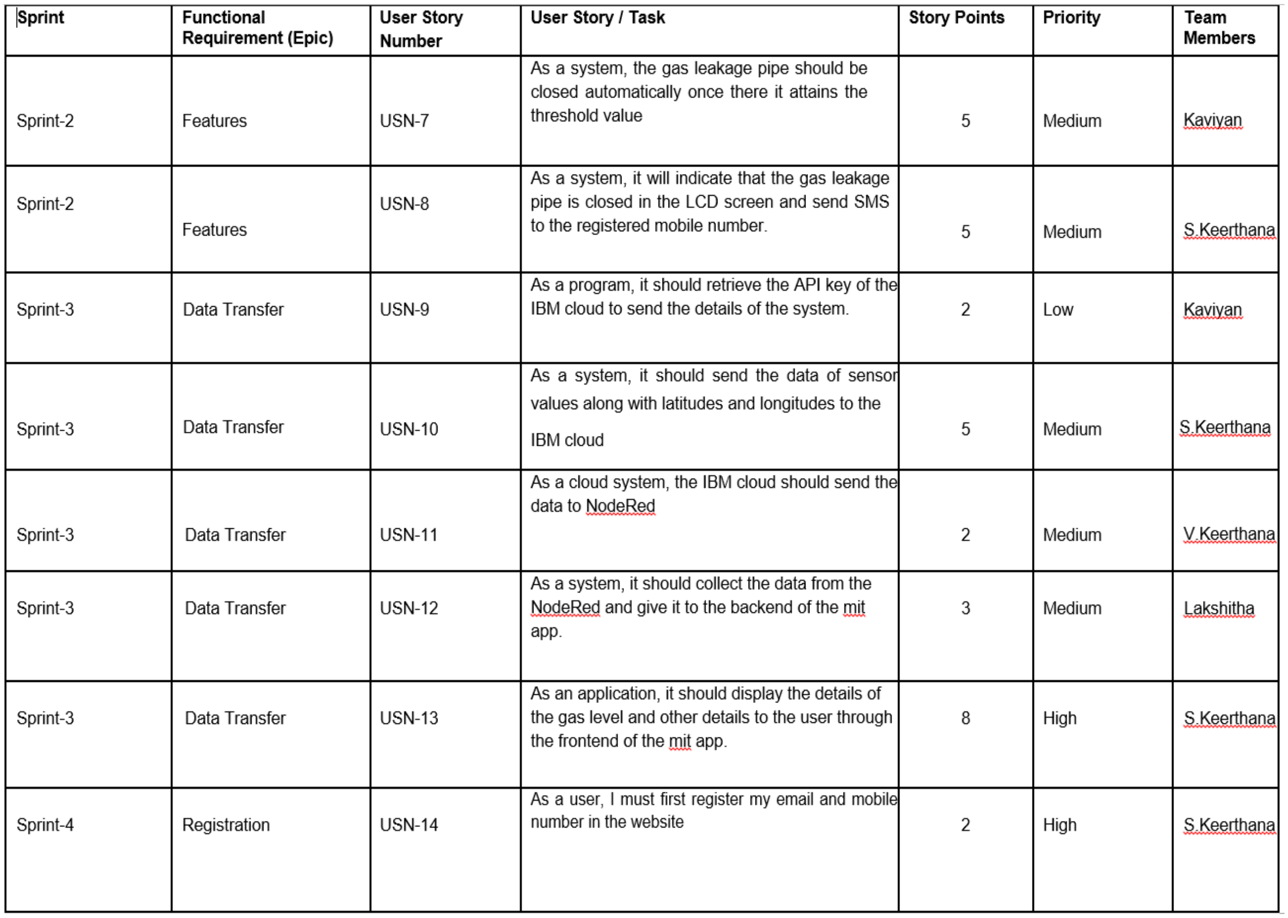
SPRINT PLANNING AND ESTIMATION:

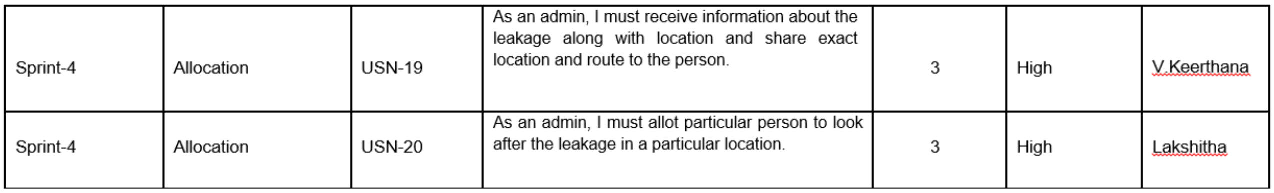
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Objective	USN-1	As a system, the gas sensor should detect the gas	8	High	Kaviyan
Sprint-1	Features	USN-2	As a system, the gas sensor values should be displayed in a LCD screen	2	Low	S.Keerthana
Sprint-1	Features	USN-3	As a system, as soon as the detected gas reaches the threshold level, the red color LED should be turned ON.	5	High	V.Keerthana
Sprint-1	Features	USN-4	As a system, as soon as the detected gas reaches the threshold level, the siren should be turned ON.	5	High	Lakshitha
Sprint-2	Focus	USN-5	As a system, it should the send the location where the gas is detected	8	High	V.Keerthana
Sprint-2	Focus	USN-6	As a system, it should also send the alerting SMS to the registered phone number	2	Low	Lakshitha



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Registration	USN-15	As a user, I must receive confirmation mail and SMS on registration	2	Medium	Kaviyan
Sprint-4	Login	USN-16	As a user, I can login into the web application through email and password.	3	High	V.Keerthana
Sprint-4	Dashboard	USN-17	As a user, I can access the dashboard and make use of available resources.	2	Medium	S.Keerthana
Sprint-4	Focus	USN-18	As a user, I must receive an SMS once the leakage is detected.	5	High	Lakshitha



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Features	USN-7	As a system, the gas leakage pipe should be closed automatically once there it attains the threshold value	5	Medium	Kaviyan
Sprint-2	Features	USN-8	As a system, it will indicate that the gas leakage pipe is closed in the LCD screen and send SMS to the registered mobile number.	5	Medium	S.Keerthana
Sprint-3	Data Transfer	USN-9	As a program, it should retrieve the API key of the IBM cloud to send the details of the system.	2	Low	Kaviyan
Sprint-3	Data Transfer	USN-10	As a system, it should send the data of sensor values along with latitudes and longitudes to the IBM cloud	5	Medium	S.Keerthana
Sprint-3	Data Transfer	USN-11	As a cloud system, the IBM cloud should send the data to NodeRed	2	Medium	V.Keerthana
Sprint-3	Data Transfer	USN-12	As a system, it should collect the data from the NodeRed and give it to the backend of the mit app.	3	Medium	Lakshitha
Sprint-3	Data Transfer	USN-13	As an application, it should display the details of the gas level and other details to the user through the frontend of the mit app.	8	High	S.Keerthana
Sprint-4	Registration	USN-14	As a user, I must first register my email and mobile number in the website	2	High	S.Keerthana



Sprint-4	Allocation	USN-19	As an admin, I must receive information about the leakage along with location and share exact location and route to the person.	3	High	V.Keerthana
Sprint-4	Allocation	USN-20	As an admin, I must allot particular person to look after the leakage in a particular location.	3	High	Lakshitha

CODE FOR IBM Watson IoT Platform:

```
import time import sys

import

ibmiotf.application

import ibmiotf.devicex

import random
```

```
#Provide your IBM Watson Device Credentials

organization = "jjrtf7" deviceType = "ESP32"

deviceId = "1234" authMethod = "token"

authToken = "12345678" # Initialize GPIO def

myCommandCallback(cmd):

print("Command received: %s" %

cmd.data['command']) status=cmd.data['command'] if

status=="switchon": print ("Switch is on") else :

print ("Switch is off") #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)

gasconcentration=random.randint(0,100) data = { 'temp' :

temp, 'Humid': Humid, "gasconcentration":

gasconcentration} #print

data def

myOnPublishCallback():
```

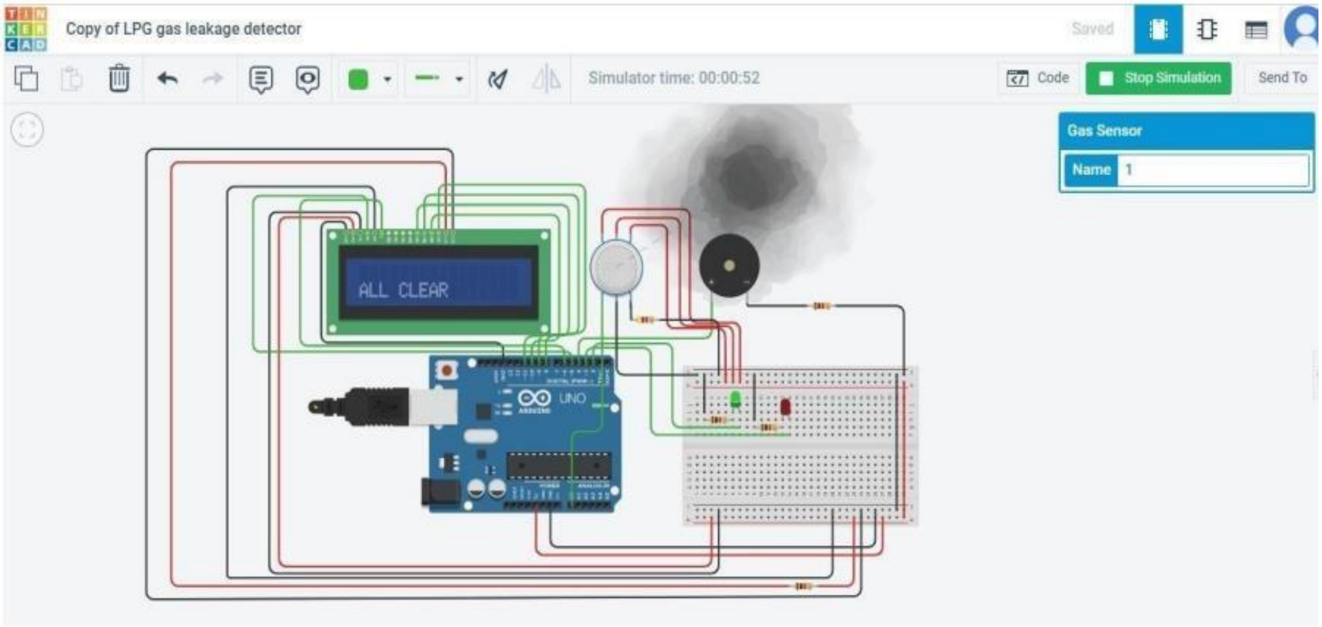
```
print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid,
"gasconcentration = %s %%" % gasconcentration, "to IBM Watson") success =
deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
if notsuccess: print("Not connected to IoTF")
time.sleep(1) deviceCli.commandCallback =
myCommandCallback # Disconnect the device and
application from the cloud deviceCli.disconnect()
```

SOLUTION STATEMENT:

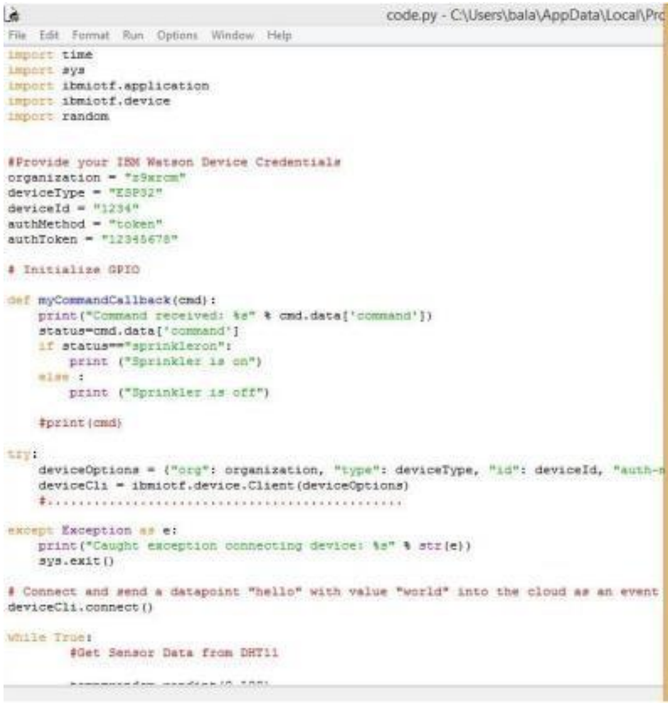
The system might be viewed as a modest attempt to link up the principal gas detection techniques now in use with a mobile platform coupled with IoT platforms. One metre around the rover, the gases are detected, and the sensor output data is continually sent to the nearby server. Stray gases are also detected because of the sensors' subpar precision, which introduces some inaccuracy into their results, particularly in the case of methane. Additionally, the storage and availability of hazardous gases like hydrogen sulphide makes it difficult to test the integrated gear. The complexity of system maintenance and material selection for the system in the event of corrosive gases is reduced because the system operates outside the pipeline. The system can only be used as a primary indicator of leakage inside a plant at this point.

TESTING & RESULTS:

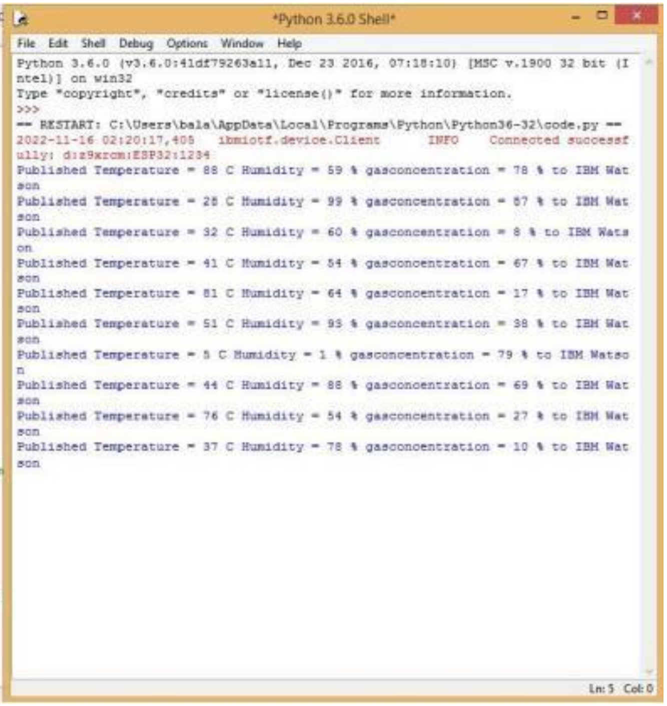
TINKERCAD:



PYTHON CODE EXECUTION:

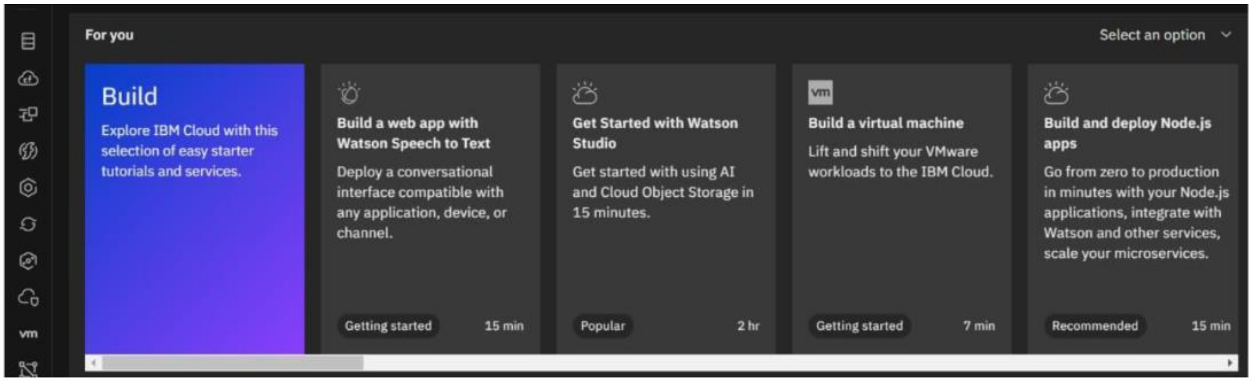

code.py - C:\Users\bala\AppData\Local\Pro
File Edit Format Run Options Window Help

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


#Provide your IBM Watson Device Credentials
organization = "z9xrcm"
deviceType = "ESP32"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="sprinkleron":
        print ("Sprinkler is on")
    else :
        print ("Sprinkler is off")

    #print(cmd)

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-m
    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
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11
```



```
Python 3.6.0 (v3.6.0:41df79263a11, Dec 23 2016, 07:18:10) [MSC v.1900 32 bit (I
ntel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\bala\AppData\Local\Programs\Python\Python36-32\code.py ==
2022-11-16 02:20:17,405   ibmiotf.device.Client      INFO    Connected successf
ully: d:z9xrcm:ESP32:1234
Published Temperature = 88 C Humidity = 59 % gasconcentration = 78 % to IBM Wat
son
Published Temperature = 28 C Humidity = 99 % gasconcentration = 87 % to IBM Wat
son
Published Temperature = 32 C Humidity = 60 % gasconcentration = 8 % to IBM Wats
on
Published Temperature = 41 C Humidity = 54 % gasconcentration = 67 % to IBM Wat
son
Published Temperature = 81 C Humidity = 64 % gasconcentration = 17 % to IBM Wat
son
Published Temperature = 51 C Humidity = 93 % gasconcentration = 38 % to IBM Wat
son
Published Temperature = 5 C Humidity = 1 % gasconcentration = 79 % to IBM Watso
n
Published Temperature = 44 C Humidity = 88 % gasconcentration = 69 % to IBM Wat
son
Published Temperature = 76 C Humidity = 54 % gasconcentration = 27 % to IBM Wat
son
Published Temperature = 37 C Humidity = 78 % gasconcentration = 10 % to IBM Wat
son
```

RECENT EVENTS IN IBM WATSON IOT PLATFORM:



For you

Select an option

Build

Explore IBM Cloud with this selection of easy starter tutorials and services.

Build a web app with Watson Speech to Text

Deploy a conversational interface compatible with any application, device, or channel.

Getting started 15 min

Get Started with Watson Studio

Get started with using AI and Cloud Object Storage in 15 minutes.

Popular 2 hr

Build a virtual machine

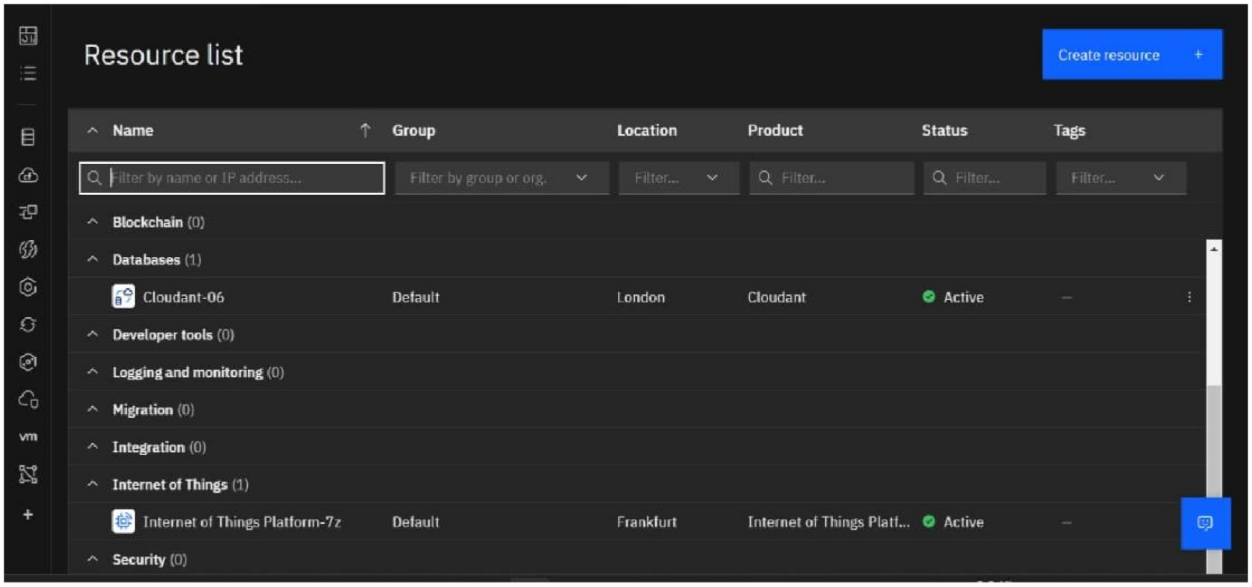
Lift and shift your VMware workloads to the IBM Cloud.

Getting started 7 min

Build and deploy Node.js apps

Go from zero to production in minutes with your Node.js applications, integrate with Watson and other services, scale your microservices.

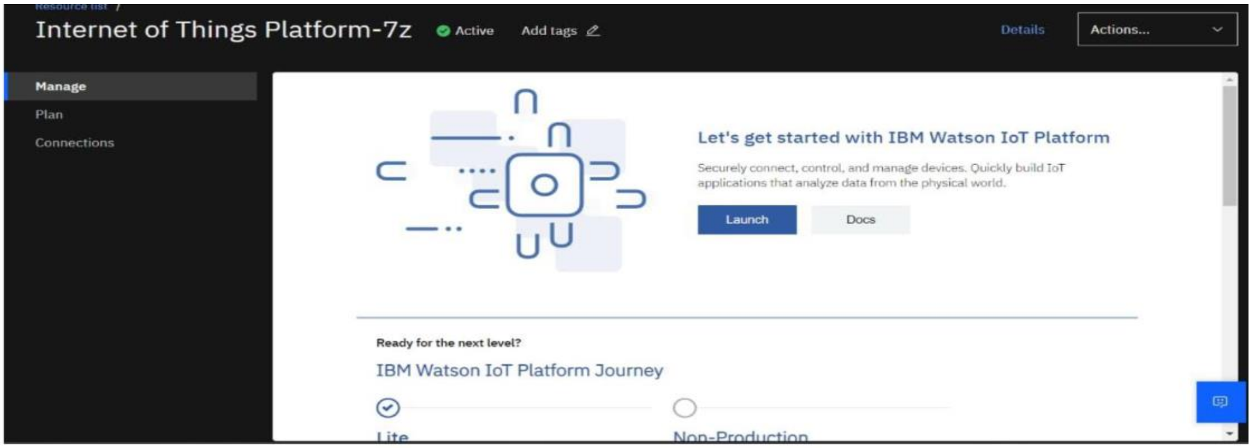
Recommended 15 min



Resource list

Create resource +

Name	Group	Location	Product	Status	Tags
Filter by name or IP address...	Filter by group or org.	Filter...	Filter...	Filter...	Filter...
Blockchain (0)					
Databases (1)					
Cloudant-06	Default	London	Cloudant	Active	–
Developer tools (0)					
Logging and monitoring (0)					
Migration (0)					
Integration (0)					
Internet of Things (1)					
Internet of Things Platform-7z	Default	Frankfurt	Internet of Things Platf...	Active	–
Security (0)					



Resource list /

Internet of Things Platform-7z Active Add tags

Details Actions...

Manage

Plan

Connections

Let's get started with IBM Watson IoT Platform

Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.

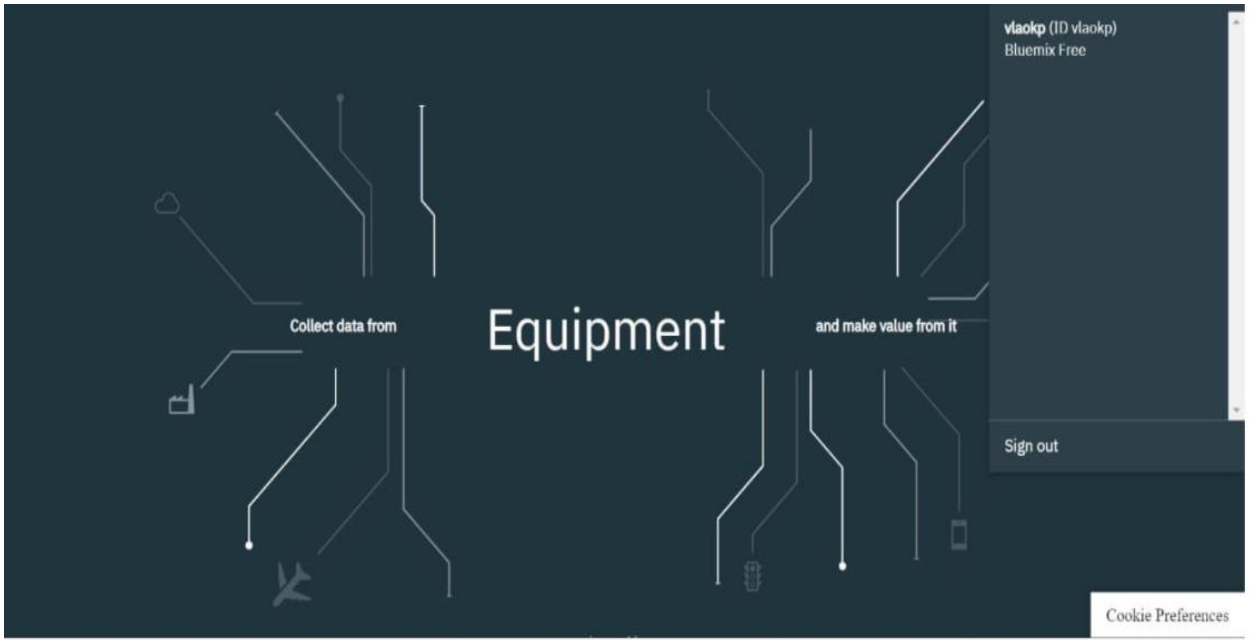
Launch Docs

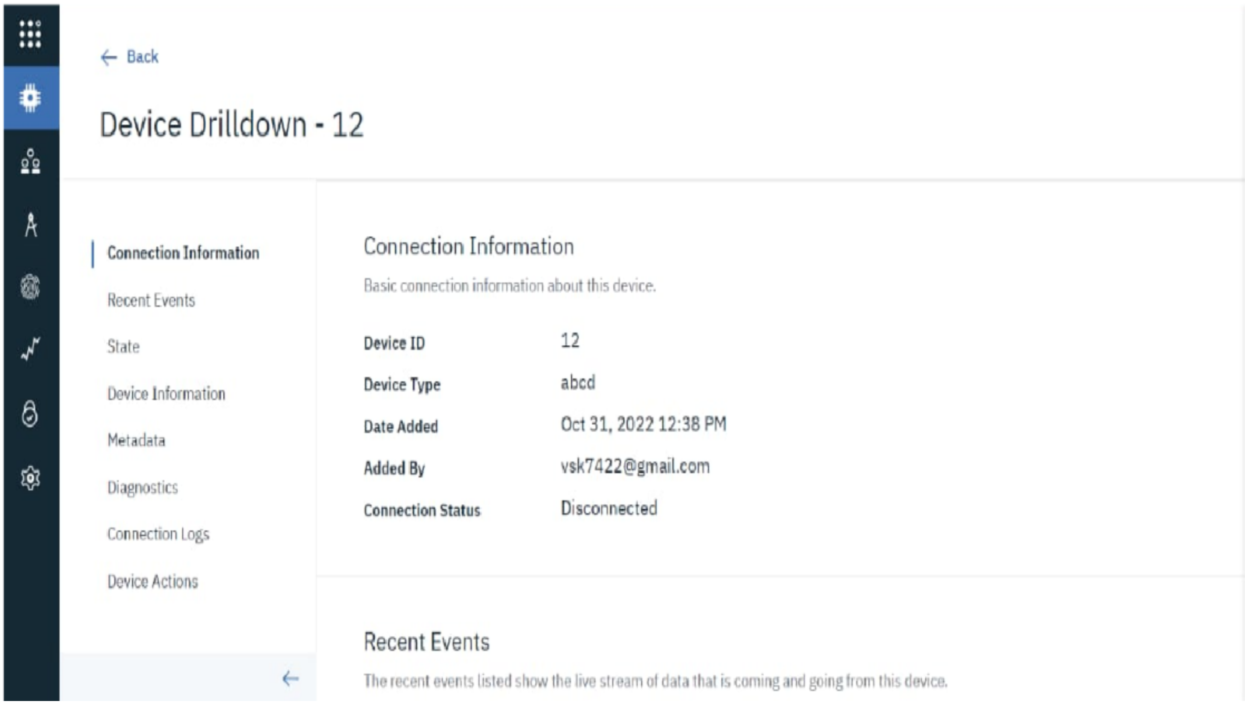
Ready for the next level?

IBM Watson IoT Platform Journey

Lite

Non-Production





Device Drilldown - 12

Connection Information

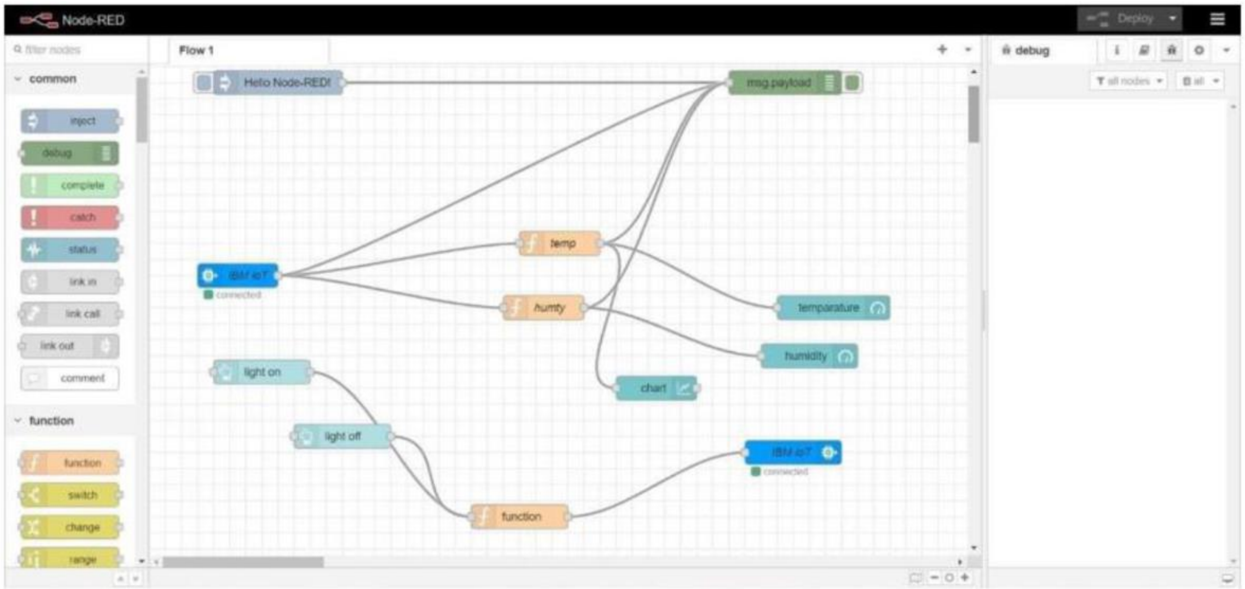
Basic connection information about this device.

Device ID	12
Device Type	abcd
Date Added	Oct 31, 2022 12:38 PM
Added By	vsk7422@gmail.com
Connection Status	Disconnected

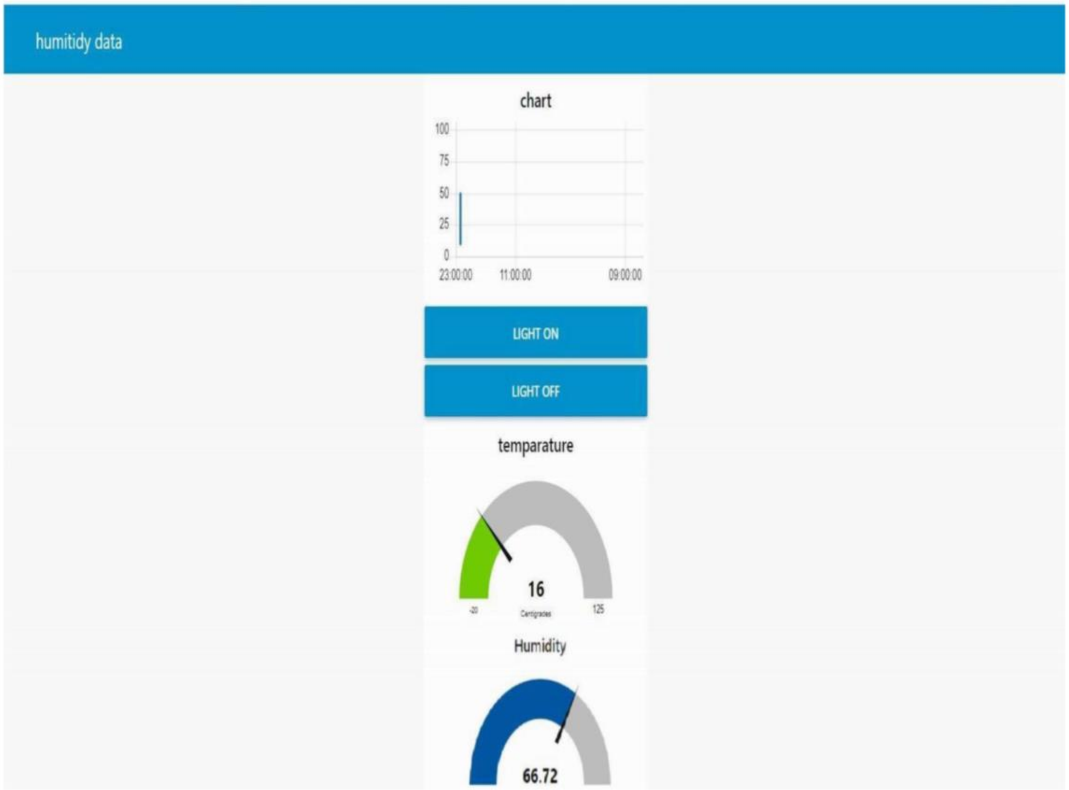
Recent Events

The recent events listed show the live stream of data that is coming and going from this device.

NODE RED FLOW:



DASHBOARD CREATED USING NODE:



CONCLUSION:

We can conclude from the project's performance that the system's detection of LPG gas leakage is remarkable. Useful for both residential and commercial purposes. We can use this technique to save lives in dangerous situations. The GSM module indicates an alert. Propane, CO2, and other gases are detected by a sensor node. Power usage and transmission range estimates are made. The sensor was constructed using straightforward techniques and an Arduino UNO Micro controller.

APPENDIX:

SOURCE CODE:

```
#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);


     }

}
```

GITHUB LINK:

IBM-EPBL/IBM-Project-10172-1659108328