PROJECT REPORT

Team ID	PNT2022TMID43411
Team Members	JOTHI KRISHNA T – 715519106018 KARTHIKEYAN A - 715519106020 NITHIYANANTH S - 715519106031 VIPIN L - 715519106059
Project Title	Gas Leakage Monitoring And Alerting System For Industries

1. INTRODUCTION

The Internet of Things aims to simplify life by automating all of the little tasks that we encounter. As much as IoT aids in task automation, its advantages can also be extended to improve current safety requirements. Safety has always been a top consideration when planning a home, a building, an industry, or a city.

It can be exceedingly dangerous for some gases to be present in the environment at higher concentrations. These gases may be hazardous after surpassing the stated concentration limits, combustible under specific temperature and humidity circumstances, or even contribute to local air pollution issues like smog and poor visibility, which can lead to serious accidents and have a negative impact on people's health.

1.1 Project Overview

The Internet of Things aims to simplify life by automating all of the little tasks that we encounter. As much as IoT aids in task automation, its advantages can also be extended to improve current safety requirements. IoT has not been immune to the fundamental worry of any project, safety. Gas leaks can be fatal and harmful, whether they occur in open or closed spaces. Despite their high level of precision, conventional gas leak detection systems overlook a few important aspects in warning the public of a leak. In order to create a Gas Leakage Detector for society that has Smart Alerting Techniques that involve text messaging the appropriate authority, we used the Internet of Things (IoT) technology.

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.

The value per time can be sensed using sensors. Values can be sent from the system to a cloud server. The sensor values existence at the threshold value can be checked by the server. The server can instruct the hardware to buzz the alert if the sensor value can exceed the limit. Additionally, the server notifies the user.

2. LITERATURE SURVEY

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 sensors is unpredictable and the positioning of equipment is improper.

2.2 References

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

2) Shruthi Unnikrishnan,1 Mohammed Razil, Joshua Benny, Shelvin Varghese and C.V. Hari, 'LPG Monitoring And Leakage Detection System', Department of Applied Electronics and Instrumentation Engineering, Rajagiri School of Engineering and Technology, Rajagiri Valley, Kakkanad, Kochi, India.

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

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

2.3 Problem Statement Definition

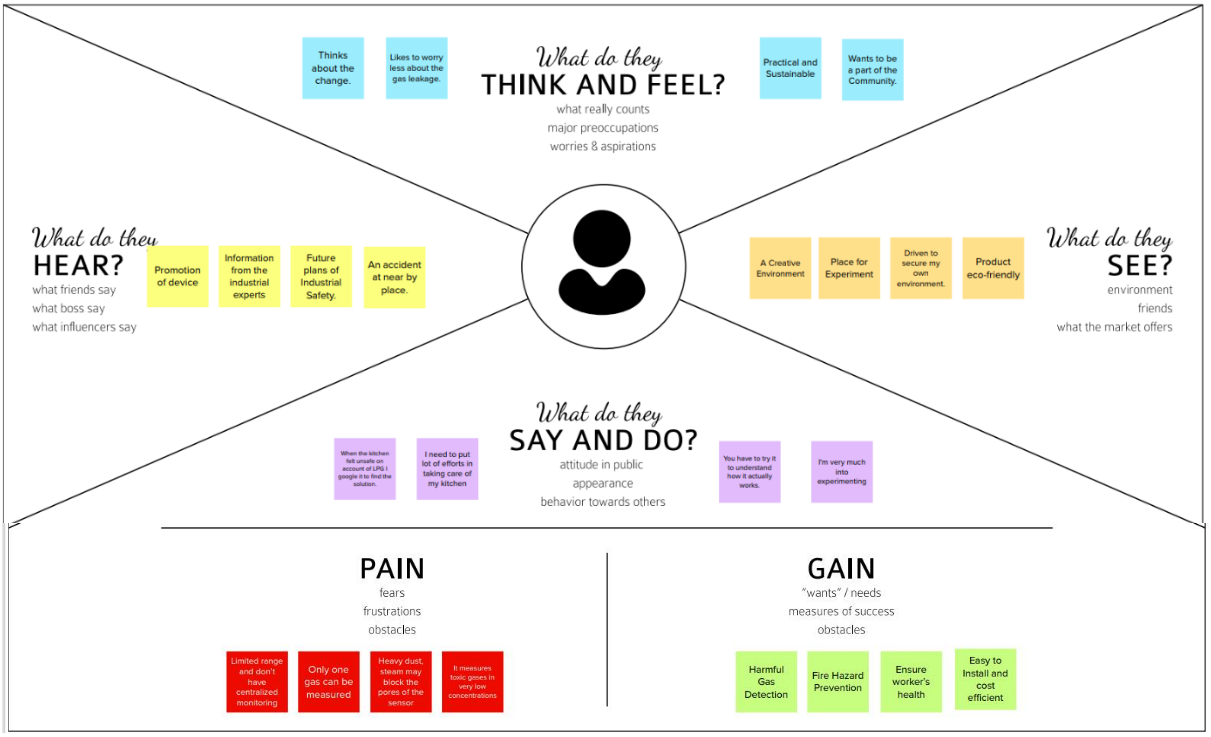
For monitoring gas leakage in the industry and Control the gas leakage ,we create a system for monitoring gas leakage and makes the installation propose simple.

To be able to work effectively on major crises rather than worrying about monitoring or gas leaks, workers in busy industries that are packed with gas, whether harmful or harmless, need a way to continuously monitor their gas pipelines and detect early if there is any leakage of gas in their surroundings. This will reduce the manpower of that industry and foster peace.

Problem Statement (PS)	I am (Customer)	I am trying to	But	Because	Which makes me feel
PS-1	Industrialist	Monitor gas leakage in the industry	I don't have any system for monitoring	The affordable of the system is high and the systems are sometimes making disasters	Unsafe
PS-2	Industrialist	Control the gas leakage	Also, the installation process is too complicated	The number of sensors is unpredictable and the positioning of equipment is improper	Disastrous

3. IDEATION & PROPOSED SOLUTION

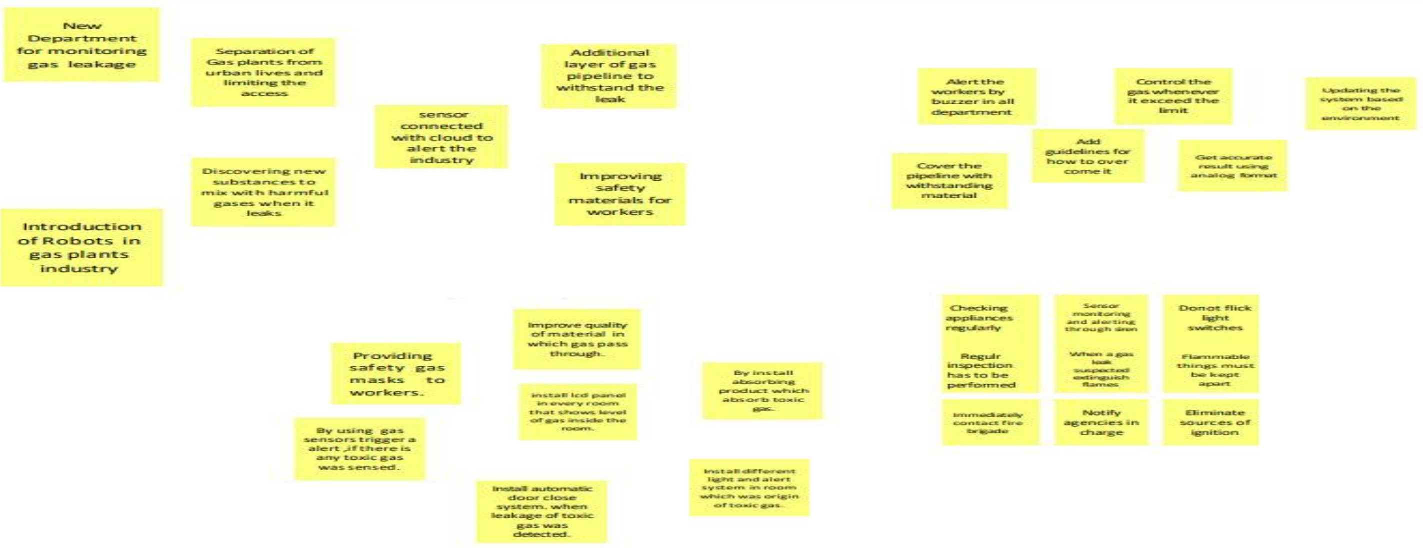
3.1 Empathy Map Canvas

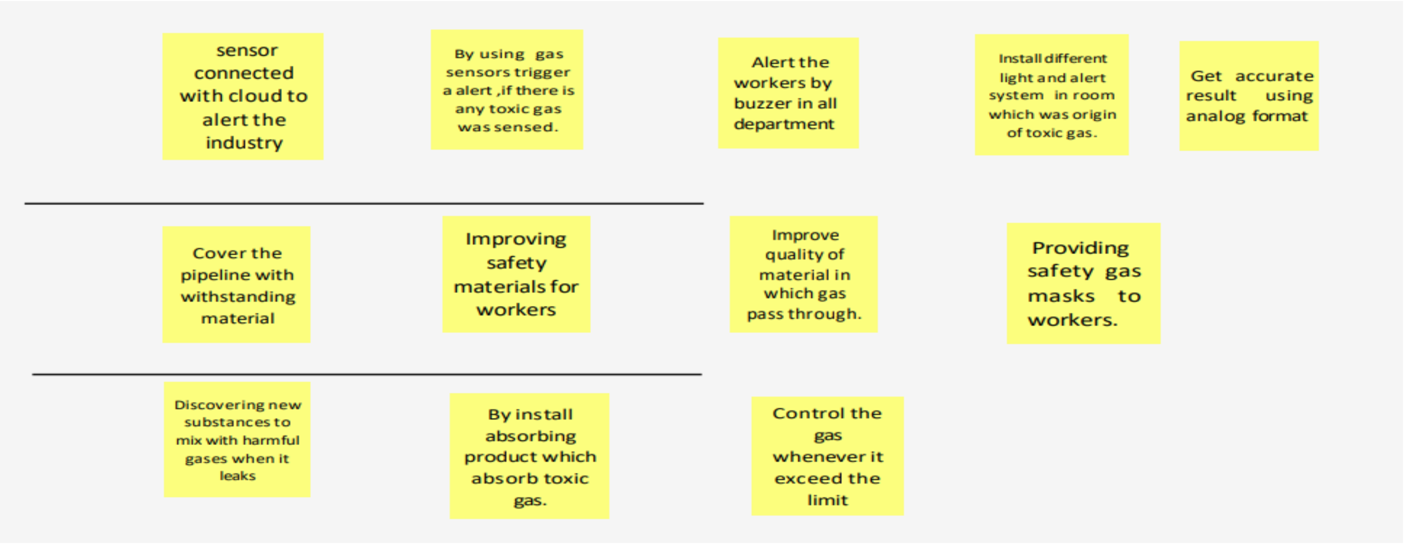

Thinks about the change.
Likes to worry less about the gas leakage.
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Practical and Sustainable
Wants to be a part of the Community.
What do they HEAR?
what friends say
what boss say
what influencers say
Promotion of device
Information from the industrial experts
Future plans of Industrial Safety.
An accident at near by place.
A Creative Environment
Place for Experiment
Driven to secure my own environment.
Product eco-friendly
What do they SEE?
environment
friends
what the market offers
When the kitchen felt unsafe on account of LPG I google it to find the solution.
I need to put lot of efforts in taking care of my kitchen
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
You have to try it to understand how it actually works.
I'm very much into experimenting
PAIN
fears
frustrations
obstacles
Limited range and don't have centralized monitoring
Only one gas can be measured
Heavy dust, steam may block the pores of the sensor
It measures toxic gases in very low concentrations
GAIN
"wants" / needs
measures of success
obstacles
Harmful Gas Detection
Fire Hazard Prevention
Ensure worker's health
Easy to Install and cost efficient

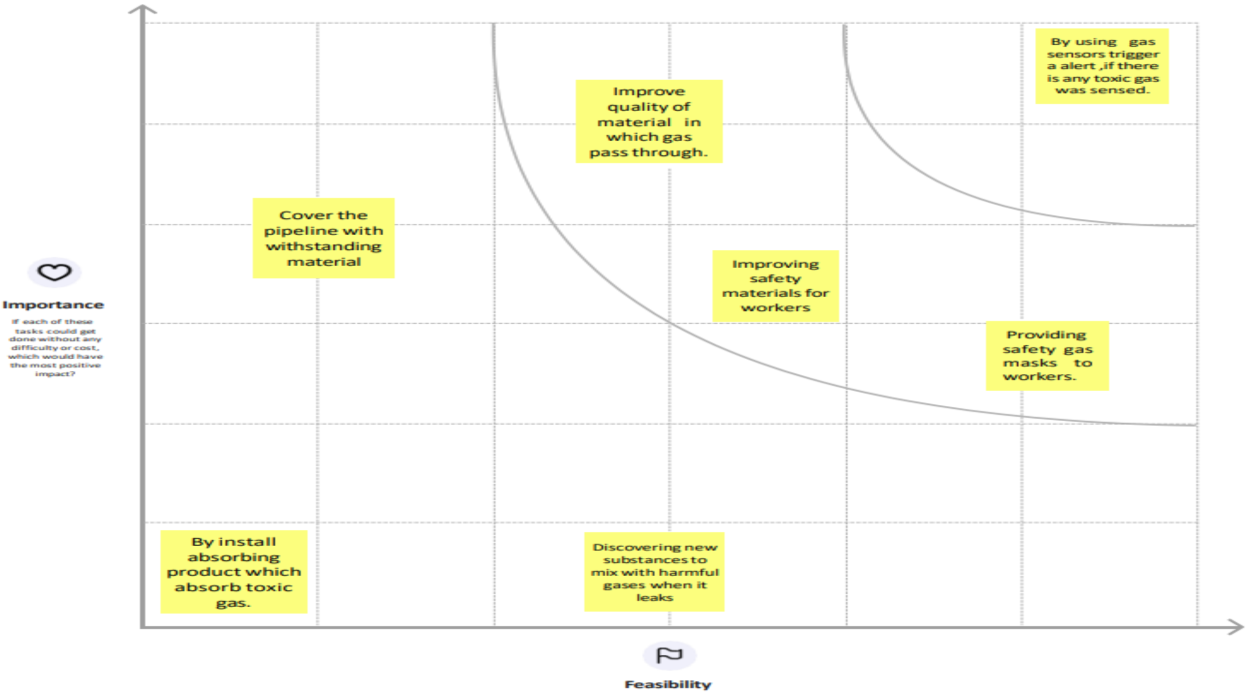
What we think to create device which helps us to control emission of flammable substance into the environment. It should be user friendly and low cost for maintenance. For that we see continuous monitoring device and buzzer is to indicate the leakage.

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 designed.The alerting message can also be forwarded to the management of 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.


New Department for monitoring gas leakage
Separation of Gas plants from urban lives and limiting the access
sensor connected with cloud to alert the industry
Additional layer of gas pipeline to withstand the leak
Discovering new substances to mix with harmful gases when it leaks
Improving safety materials for workers
Introduction of Robots in gas plants industry
Alert the workers by buzzer in all department
Control the gas whenever it exceed the limit
Updating the system based on the environment
Add guidelines for how to over come it
Cover the pipeline with withstanding material
Get accurate result using analog format
Improve quality of material in which gas pass through.
Providing safety gas masks to workers.
install lcd panel in every room that shows level of gas inside the room.
By install absorbing product which absorb toxic gas.
By using gas sensors trigger a alert ,if there is any toxic gas was sensed.
Install automatic door close system. when leakage of toxic gas was detected.
Install different light and alert system in room which was origin of toxic gas.
Checking appliances regularly
Regulr inspection has to be performed
Sensor monitoring and alerting through siren
When a gas leak suspected extinguish flames
Donot flick light switches
Flammable things must be kept apart
Immediately contact fire brigade
Notify agencies in charge
Eliminate sources of ignition


sensor connected with cloud to alert the industry
By using gas sensors trigger a alert ,if there is any toxic gas was sensed.
Alert the workers by buzzer in all department
Install different light and alert system in room which was origin of toxic gas.
Get accurate result using analog format
Cover the pipeline with withstanding material
Improving safety materials for workers
Improve quality of material in which gas pass through.
Providing safety gas masks to workers.
Discovering new substances to mix with harmful gases when it leaks
By install absorbing product which absorb toxic gas.
Control the gas whenever it exceed the limit



3.3 Proposed Solution

1. **Problem Statement (Problem to be solved) :**

Workers who are engaged with a busy industries packed with gas either harmful or harmless needs a way to monitor their gas pipelines continuously and detect early if there is any leakage of gas in their surroundings so that they can work efficiently on major crises rather than worrying about monitoring or leakage of gas, this will indeed reduce the manpower of that industry and create a peaceful environment.

2. **Idea / Solution description:**

Real time gas monitors can overcome delayed response times to such gas leaks. Hence, multiple gas monitors can be placed strategically across any potential source for early gas leak detection. Also, mapping of such gas leaks in these industrial zones can help the safety in charge to take timely corrective actions. Hence, by setting appropriate thresholds, various data-driven environmental automation can be implemented for industrial safety.

3. **Novelty / Uniqueness:**

Even though there are many existing solutions for this problem they failed to satisfy the needs of customer. Some of the solutions are only detecting some particular gases where some others failed to alert the main department and other solutions are with some delays. Our solution not only notify the industry person but also notify the fire fighters so that can take control over the situation and our solution will alert the workers even there is a small leak of gases.

4. **Social Impact / Customer Satisfaction:**

Our solution will be very helpful for the workers and the society which is associated or located nearby the industries. Our solution will prevent great disasters like Bhopal Gas Tragedy so that so many lives can be saved. Through this project the workers mental pressure will be reduced so that they can concentrate on other works or by relaxing them.

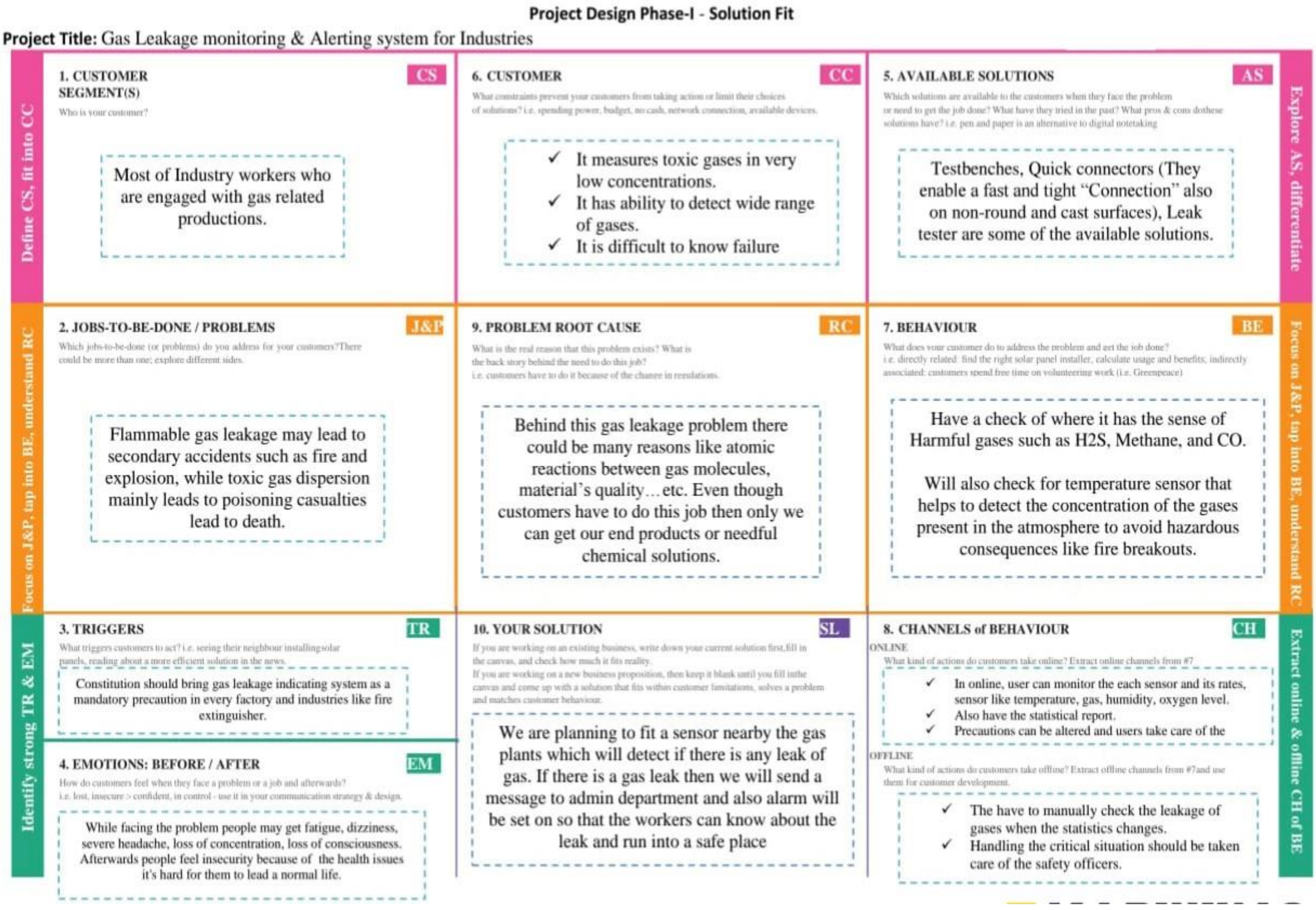
5. **Business Model (Revenue Model):**

The main target of our solution is Industries so we have planned to visit industries and explain them about the benefits of our products. So that they can aware of the importance of this solution and use it.

6. **Scalability of the Solution:**

Our solution can be integrated for further future use because the solution we have provided will be lay on the basic or initial stage of any upgraded version.

3.4 Problem Solution fit



Project Design Phase-I - Solution Fit

Project Title: Gas Leakage monitoring & Alerting system for Industries

Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
Extract online & offline CH of BE

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

Most of Industry workers who are engaged with gas related productions.

6. CUSTOMER CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

- ✓ It measures toxic gases in very low concentrations.
- ✓ It has ability to detect wide range of gases.
- ✓ It is difficult to know failure

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons dothese solutions have? i.e. pen and paper is an alternative to digital notetaking

Testbenches, Quick connectors (They enable a fast and tight "Connection" also on non-round and cast surfaces), Leak tester are some of the available solutions.

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

Which jobs-to-be-done (or problems) do you address for your customers?There could be more than one; explore different sides.

Flammable gas leakage may lead to secondary accidents such as fire and explosion, while toxic gas dispersion mainly leads to poisoning casualties lead to death.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

Behind this gas leakage problem there could be many reasons like atomic reactions between gas molecules, material's quality…etc. Even though customers have to do this job then only we can get our end products or needful chemical solutions.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Have a check of where it has the sense of Harmful gases such as H2S, Methane, and CO.

Will also check for temperature sensor that helps to detect the concentration of the gases present in the atmosphere to avoid hazardous consequences like fire breakouts.

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installingsolar panels, reading about a more efficient solution in the news.

Constitution should bring gas leakage indicating system as a mandatory precaution in every factory and industries like fire extinguisher.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

While facing the problem people may get fatigue, dizziness, severe headache, loss of concentration, loss of consciousness. Afterwards people feel insecurity because of the health issues it's hard for them to lead a normal life.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first,fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill inthe canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

We are planning to fit a sensor nearby the gas plants which will detect if there is any leak of gas. If there is a gas leak then we will send a message to admin department and also alarm will be set on so that the workers can know about the leak and run into a safe place

8. CHANNELS of BEHAVIOUR CH

ONLINE

What kind of actions do customers take online? Extract online channels from #7

- ✓ In online, user can monitor the each sensor and its rates, sensor like temperature, gas, humidity, oxygen level.
- ✓ Also have the statistical report.
- ✓ Precautions can be altered and users take care of the

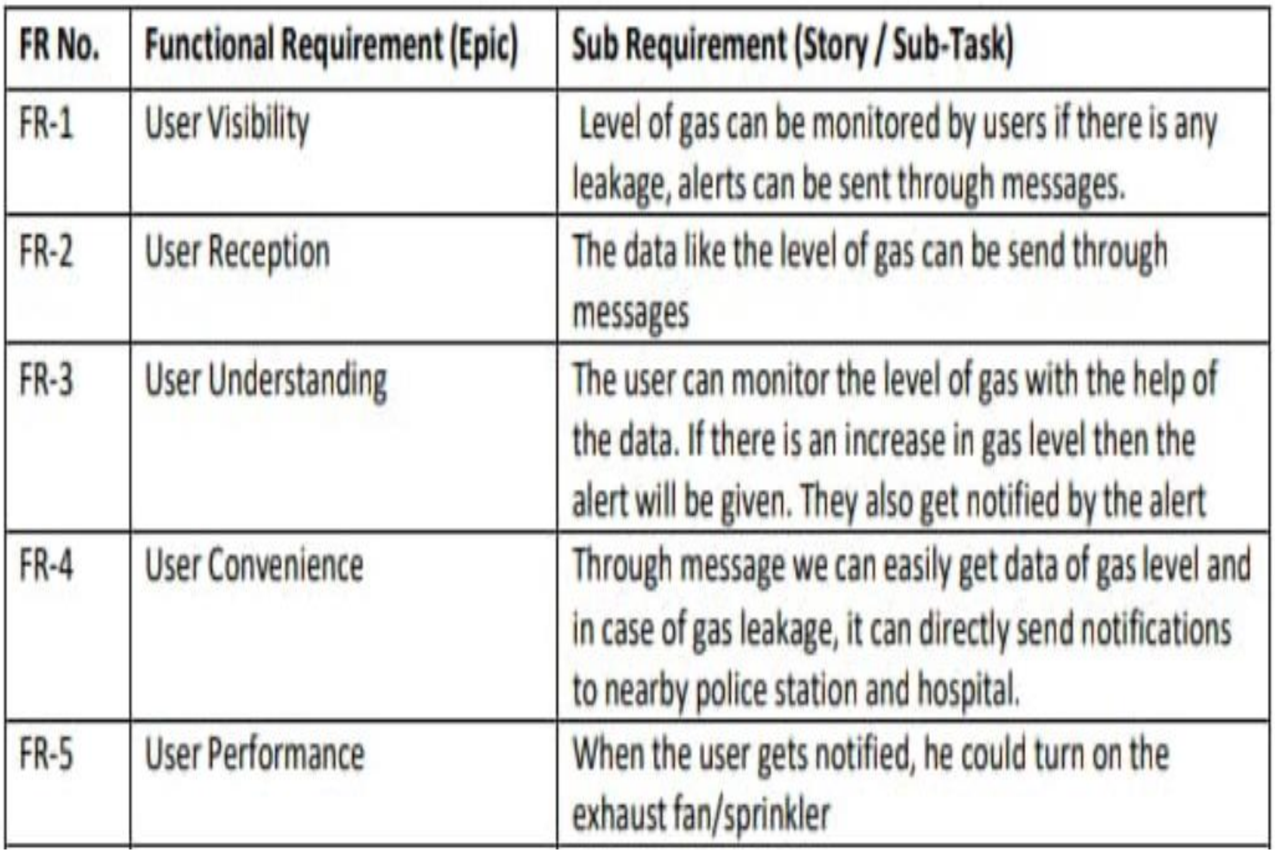
OFFLINE

What kind of actions do customers take offline? Extract offline channels from #7and use them for customer development.

- ✓ The have to manually check the leakage of gases when the statistics changes.
- ✓ Handling the critical situation should be taken care of the safety officers.

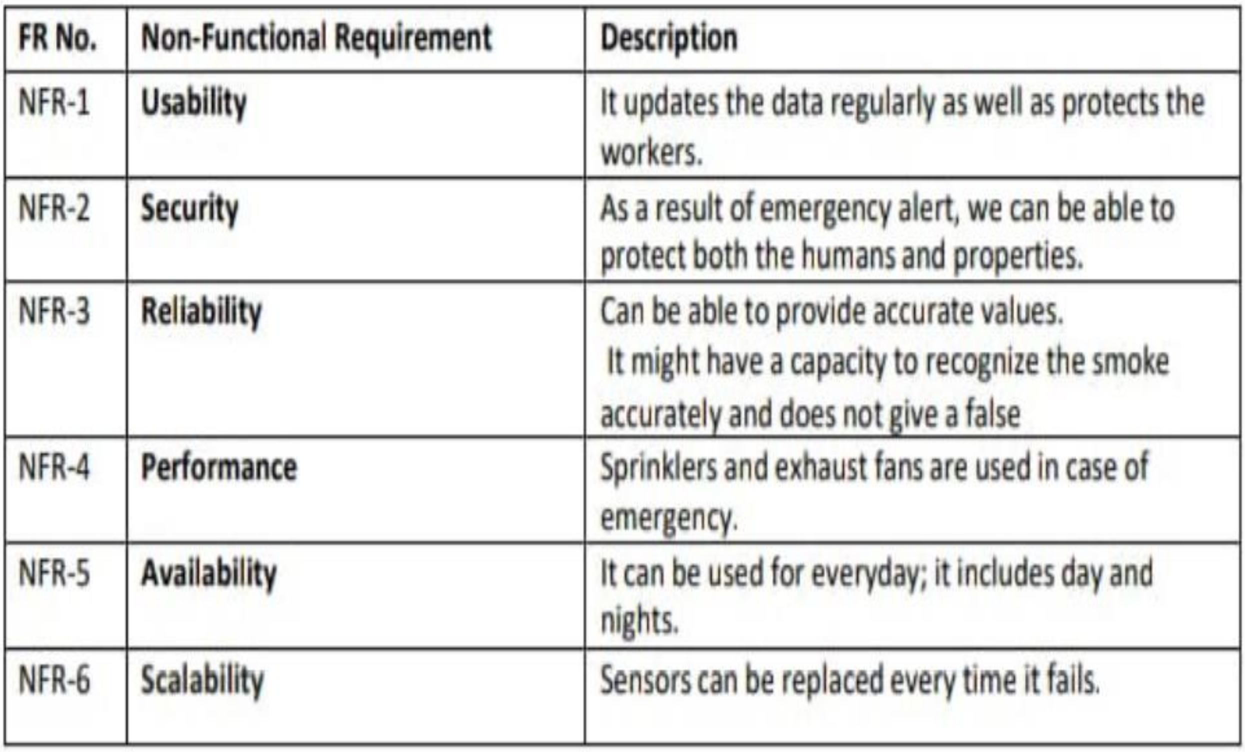
4. REQUIREMENT ANALYSIS

4.1 Functional requirement



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

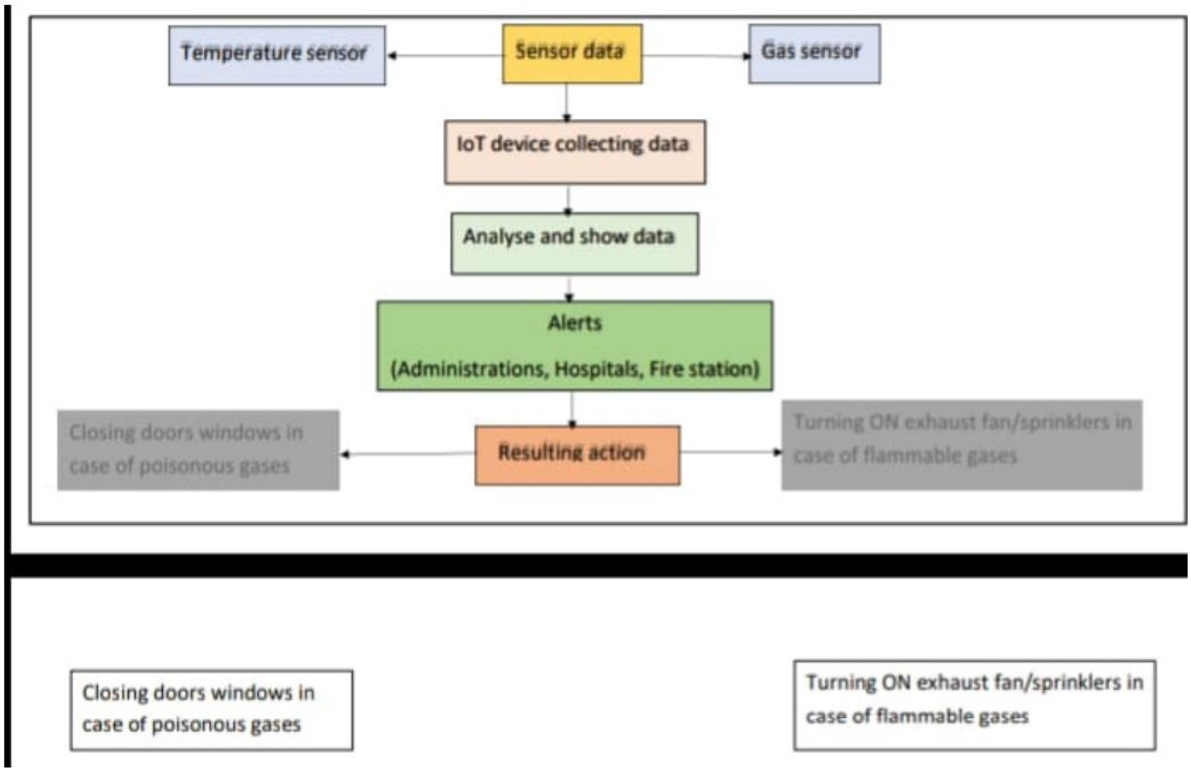
4.2 Non-Functional requirements



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.

5. PROJECT DESIGN

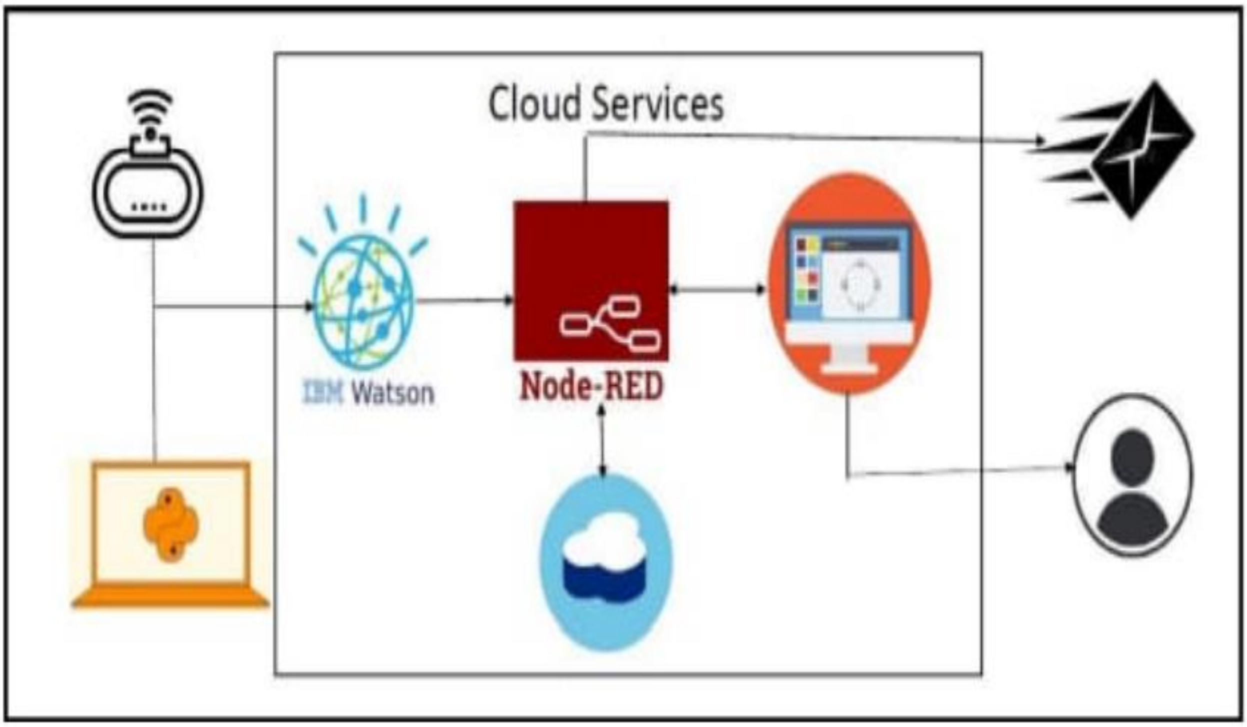
5.1 Data Flow Diagrams



Temperature sensor
Sensor data
Gas sensor
IoT device collecting data
Analyse and show data
Alerts
(Administrations, Hospitals, Fire station)
Closing doors windows in case of poisonous gases
Resulting action
Turning ON exhaust fan/sprinklers in case of flammable gases
Closing doors windows in case of poisonous gases
Turning ON exhaust fan/sprinklers in case of flammable gases

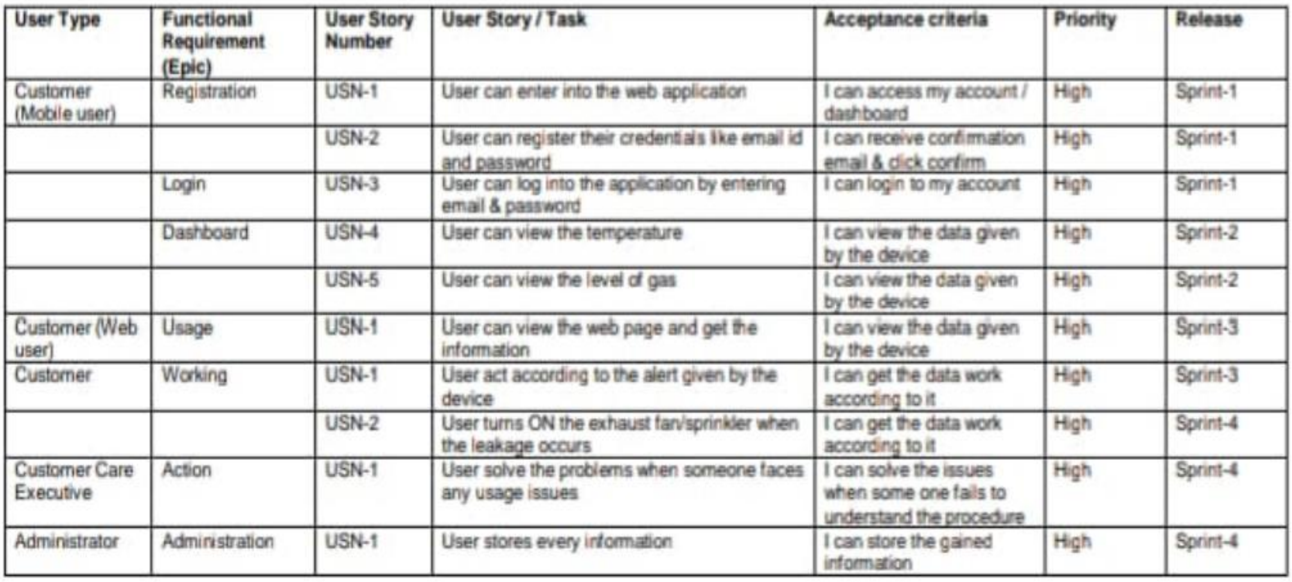
This is the data flow diagram of gas leakage monitoring and detection.Here the data from temperature sensor and gas sensor is collected from IOT device and the data is analyzed. If the alert action requires it alerts and the requied measures are taken.

5.2 Solution & Technical Architecture



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 faces 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

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Monitor the gas leakage	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	JOTHI KRISHNA T KARTHIKEYAN A NITHIYANANTH S VIPIN L
Sprint-2	Avoid From Disaster	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	JOTHI KRISHNA T KARTHIKEYAN A NITHIYANANTH S VIPIN L
Sprint-3	Detect the gas	USN-3	As a user, I can register for the application through Facebook	2	Low	JOTHI KRISHNA T KARTHIKEYAN A NITHIYANANTH S VIPIN L
Sprint-4	The model is trained and tested by sample datase	USN-4	As a user, I can register for the application through Gmail	2	Medium	JOTHI KRISHNA T KARTHIKEYAN A NITHIYANANTH S VIPIN L
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	JOTHI KRISHNA T KARTHIKEYAN A NITHIYANANTH S VIPIN L
	Dashboard					

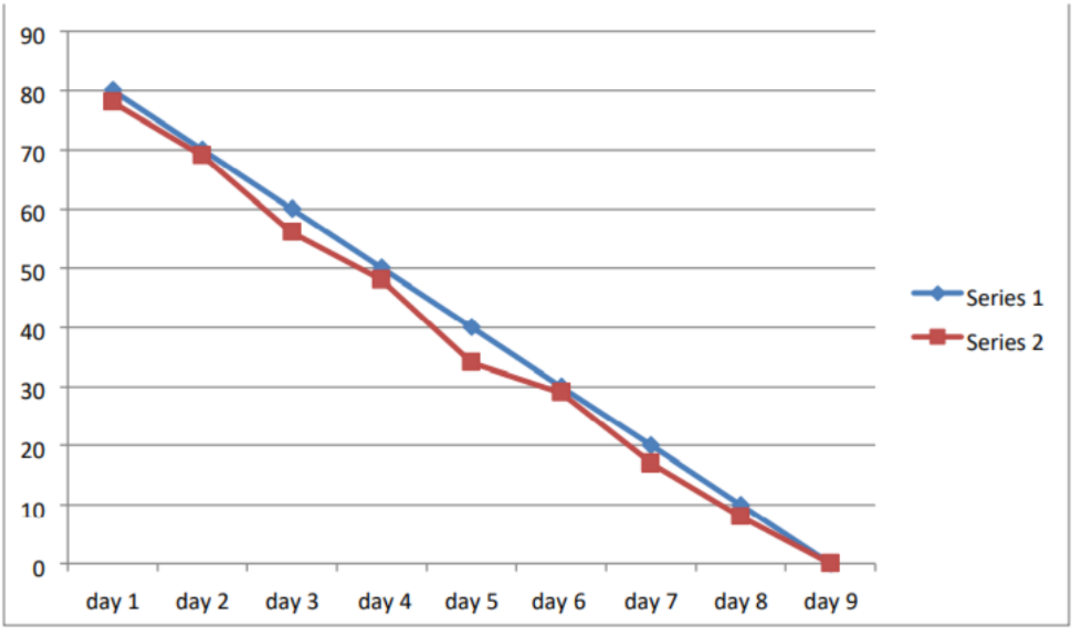
6.2 Sprint Delivery Schedule

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	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	04 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	010 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	20 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)

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

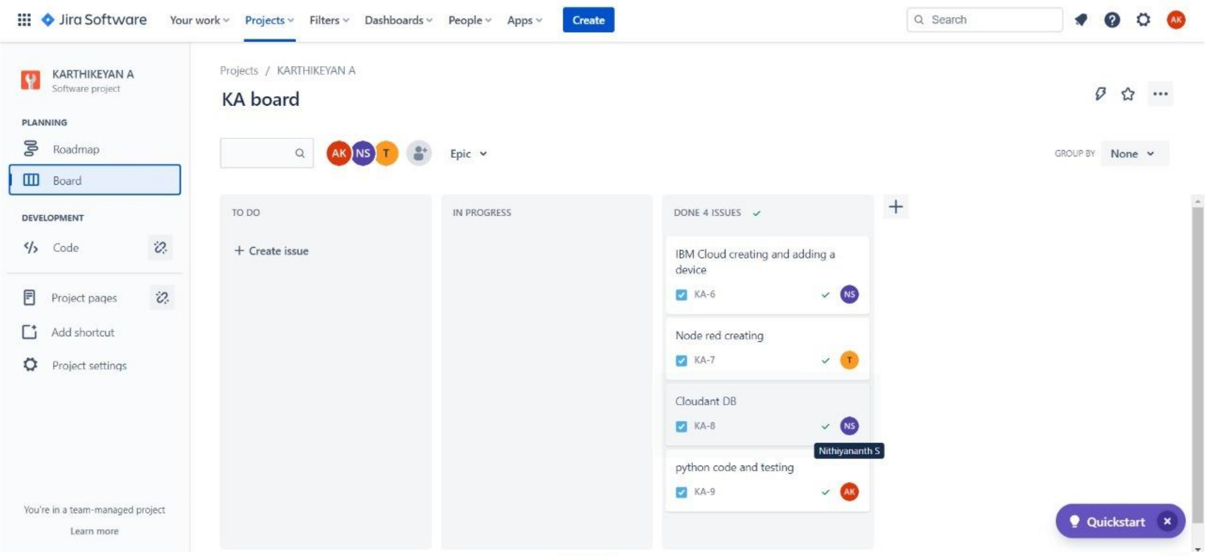
Burndown Chart:



6.3 Reports from JIRA

Jira Work Management is intended as generic project management.

Jira Software includes the base software, including agile project management features



Projects / KARTHIKEYAN A

KA board

AK NS T Epic GROUP BY None

TO DO

+ Create issue

IN PROGRESS

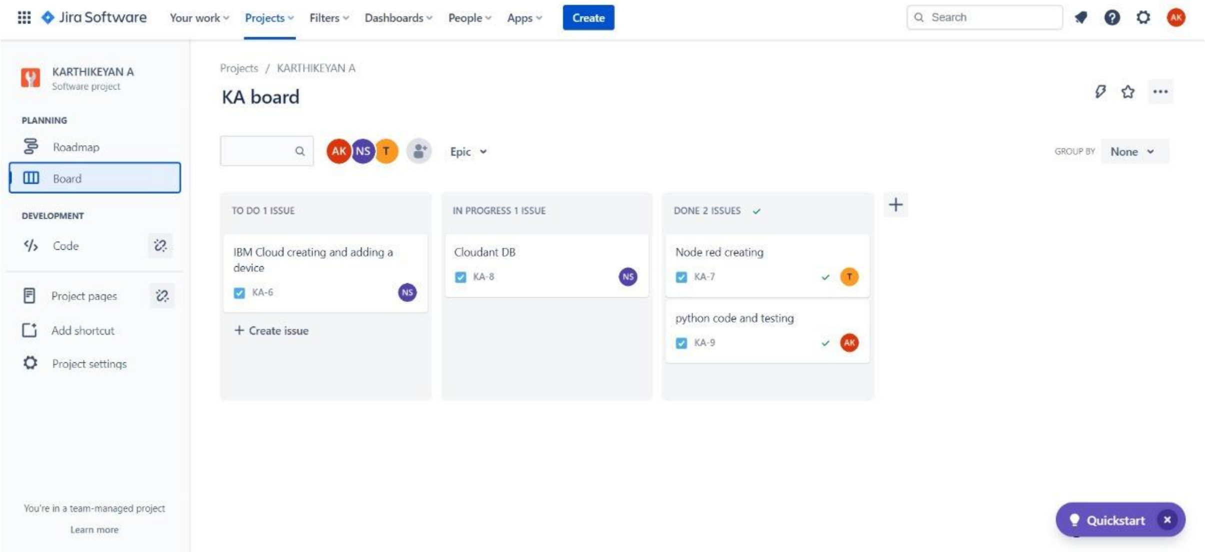
DONE 4 ISSUES ✓

IBM Cloud creating and adding a device
☑ KA-6 ✓ NS

Node red creating
☑ KA-7 ✓ T

Cloudant DB
☑ KA-8 ✓ NS
Nithiyananth S

python code and testing
☑ KA-9 ✓ AK



Projects / KARTHIKEYAN A

KA board

AK NS T Epic GROUP BY None

TO DO 1 ISSUE

IBM Cloud creating and adding a device
☑ KA-6 NS

+ Create issue

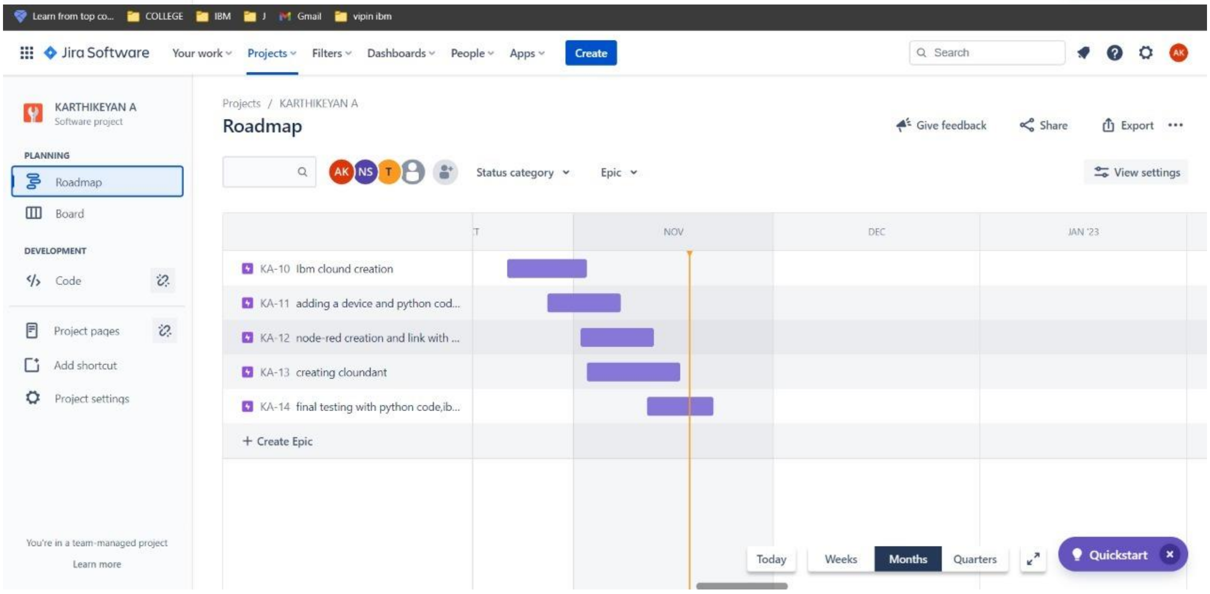
IN PROGRESS 1 ISSUE

Cloudant DB
☑ KA-8 NS

DONE 2 ISSUES ✓

Node red creating
☑ KA-7 ✓ T

python code and testing
☑ KA-9 ✓ AK



Projects / KARTHIKEYAN A

Roadmap

AK NS T Status category Epic

T NOV DEC JAN '23

KA-10 Ibm clound creation

KA-11 adding a device and python cod...

KA-12 node-red creation and link with ...

KA-13 creating cloundant

KA-14 final testing with python code,ib...

+ Create Epic

7. CODING & SOLUTIONING

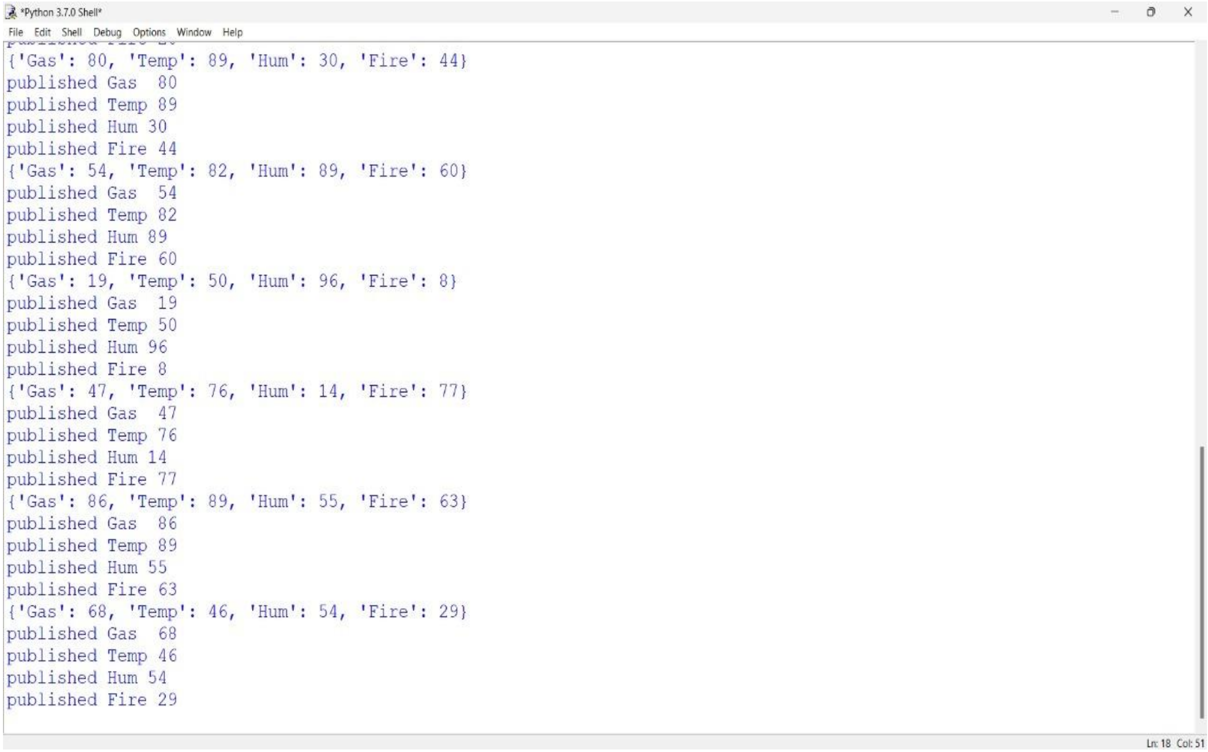
7.1 Feature 1

The data has been published to the IBM cloud. Thus in the python script, the values for the gas, temperature, humidity and fire have been generated and published to IBM cloud platform. This is achieved by importing the required libraries in the python script and also specifying the organization, deviceType, deviceid, authMethod and authToken to integrate with the specific cloud account, so that the data will be published to IBM cloud platform.

PYTHON CODE:

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

#ibm watson device credentials

organization="griwxv"
deviceType="ESP32"
deviceid="12345678"
authMethod="token"
authToken="12345678"

#generate random values for gas leakage


def myCommandCallback(cmd):
   print ("command received: %s" %cmd.data['command'])
   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 sending data for gas leakage

deviceCli.connect()

while True:
   Gas=random.randint(0,100)
   Temp=random.randint(0,100)
   Hum=random.randint(0,100)
   Fire=random.randint(0,100)
   data={'Gas':Gas,'Temp':Temp,'Hum':Hum,'Fire':Fire}
   print(data)
   def myOnPublishCallBack():
      print("published Gas  %s " %Gas)
      print("published Temp %s " %Temp)
      print("published Hum %s " %Hum)
      print("published Fire %s " %Fire)
   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 from the cloud

deviceCli.connect()
```

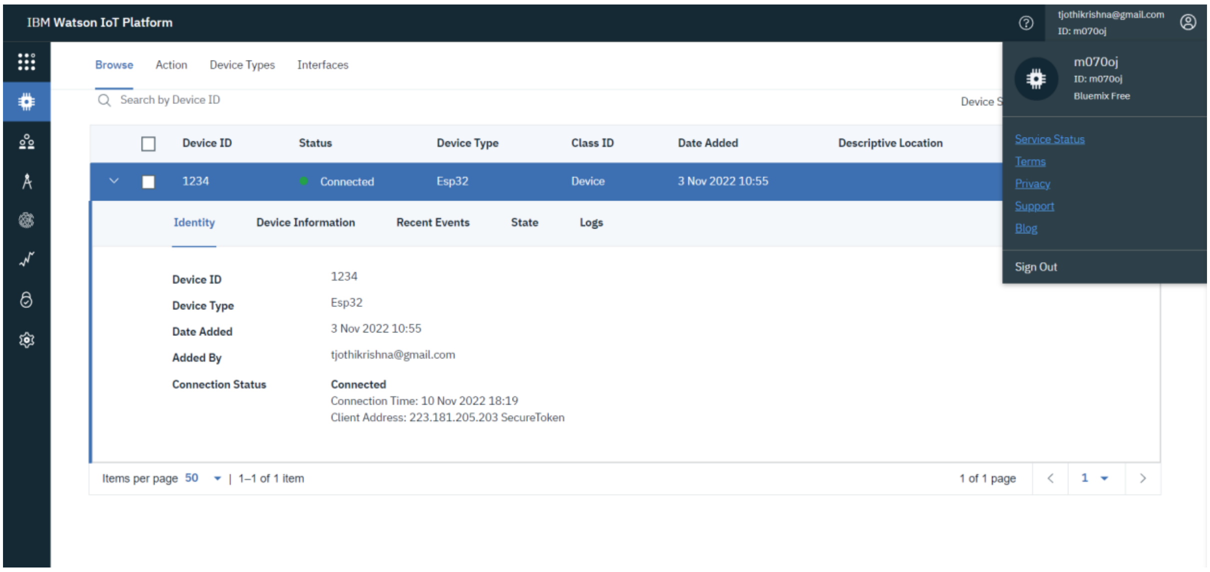
OUTPUT:



```
{'Gas': 80, 'Temp': 89, 'Hum': 30, 'Fire': 44}
published Gas  80
published Temp 89
published Hum 30
published Fire 44
{'Gas': 54, 'Temp': 82, 'Hum': 89, 'Fire': 60}
published Gas  54
published Temp 82
published Hum 89
published Fire 60
{'Gas': 19, 'Temp': 50, 'Hum': 96, 'Fire': 8}
published Gas  19
published Temp 50
published Hum 96
published Fire 8
{'Gas': 47, 'Temp': 76, 'Hum': 14, 'Fire': 77}
published Gas  47
published Temp 76
published Hum 14
published Fire 77
{'Gas': 86, 'Temp': 89, 'Hum': 55, 'Fire': 63}
published Gas  86
published Temp 89
published Hum 55
published Fire 63
{'Gas': 68, 'Temp': 46, 'Hum': 54, 'Fire': 29}
published Gas  68
published Temp 46
published Hum 54
published Fire 29
```

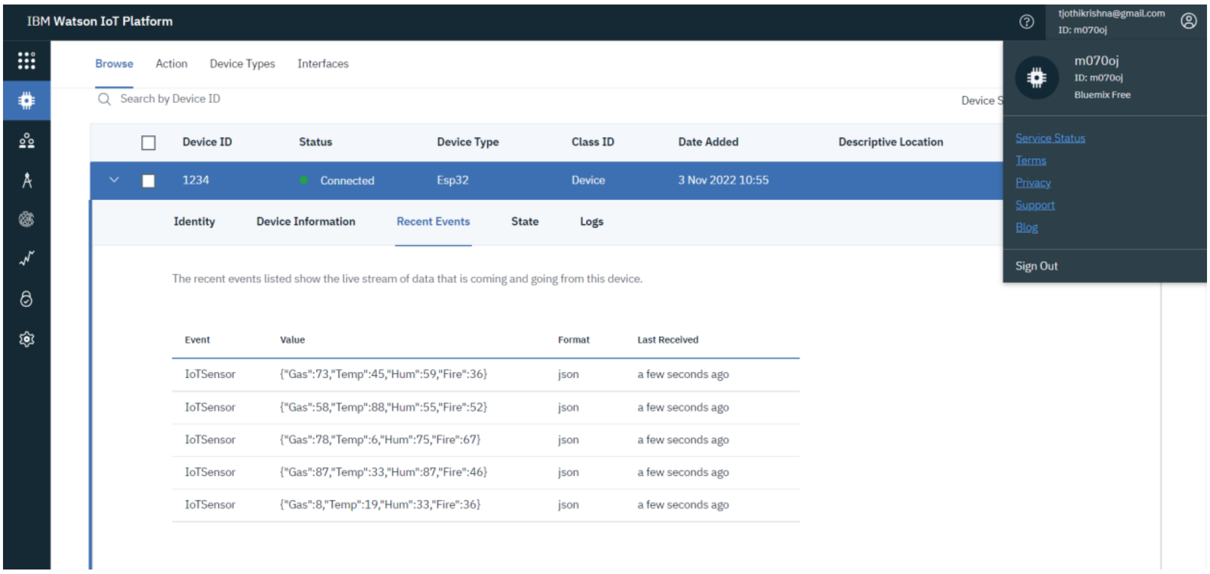
7.2 Feature 2

The IBM Watson and Node-RED services are integrated wih IBM Cloud Services. The web application is developed using a Node-RED and installing libraries in the Node-RED for the availability of dashboard nodes for creating UI.



Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
1234	Connected	Esp32	Device	3 Nov 2022 10:55	

Device ID	1234
Device Type	Esp32
Date Added	3 Nov 2022 10:55
Added By	tjothikrishna@gmail.com
Connection Status	Connected Connection Time: 10 Nov 2022 18:19 Client Address: 223.181.205.203 SecureToken



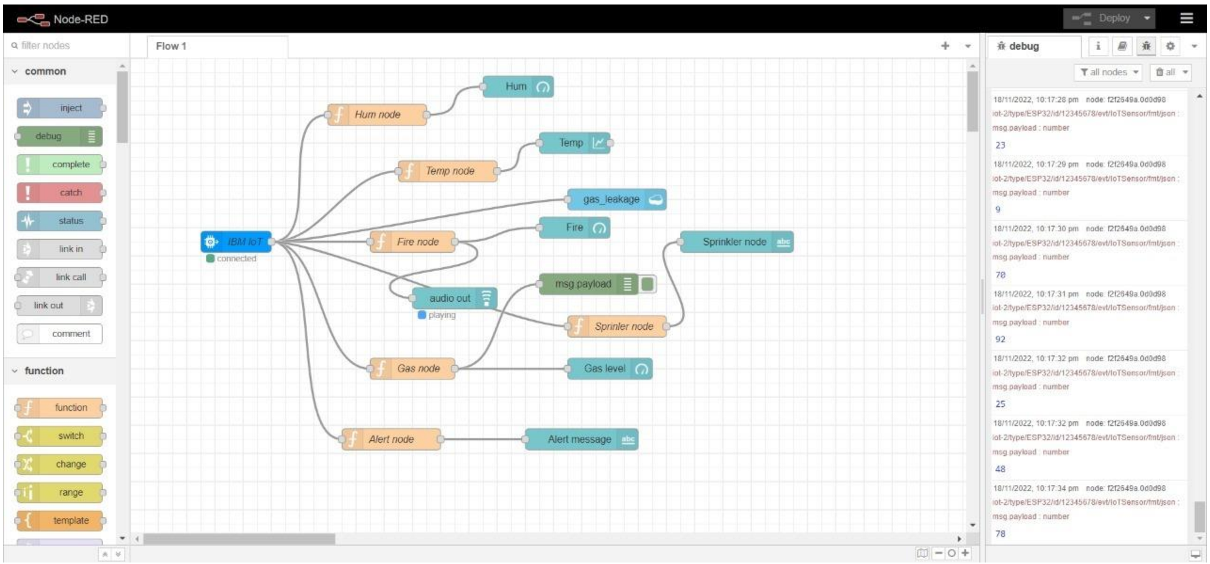
Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
1234	Connected	Esp32	Device	3 Nov 2022 10:55	

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

Event	Value	Format	Last Received
IoTSensor	{"Gas":73,"Temp":45,"Hum":59,"Fire":36}	json	a few seconds ago
IoTSensor	{"Gas":58,"Temp":88,"Hum":55,"Fire":52}	json	a few seconds ago
IoTSensor	{"Gas":78,"Temp":6,"Hum":75,"Fire":67}	json	a few seconds ago
IoTSensor	{"Gas":87,"Temp":33,"Hum":87,"Fire":46}	json	a few seconds ago
IoTSensor	{"Gas":8,"Temp":19,"Hum":33,"Fire":36}	json	a few seconds ago

A threshold value has been set for temperature, humidity, gas and fire sensor nodes. If the value has been exceeded that threshold value, an alert message is sent.

Temperature threshold value : 83
Humidity threshold value : 64
Gas level threshold value : 50
Fire threshold value : 45



```
18/11/2022, 10:17:28 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
23
18/11/2022, 10:17:29 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
9
18/11/2022, 10:17:30 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
78
18/11/2022, 10:17:31 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
92
18/11/2022, 10:17:32 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
25
18/11/2022, 10:17:32 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
48
18/11/2022, 10:17:34 pm   node: f2f2649a.0d0d98
iot-2/type/ESP32/id/12345678/evt/IoTSensor/fmt/json :
msg.payload : number
78
```

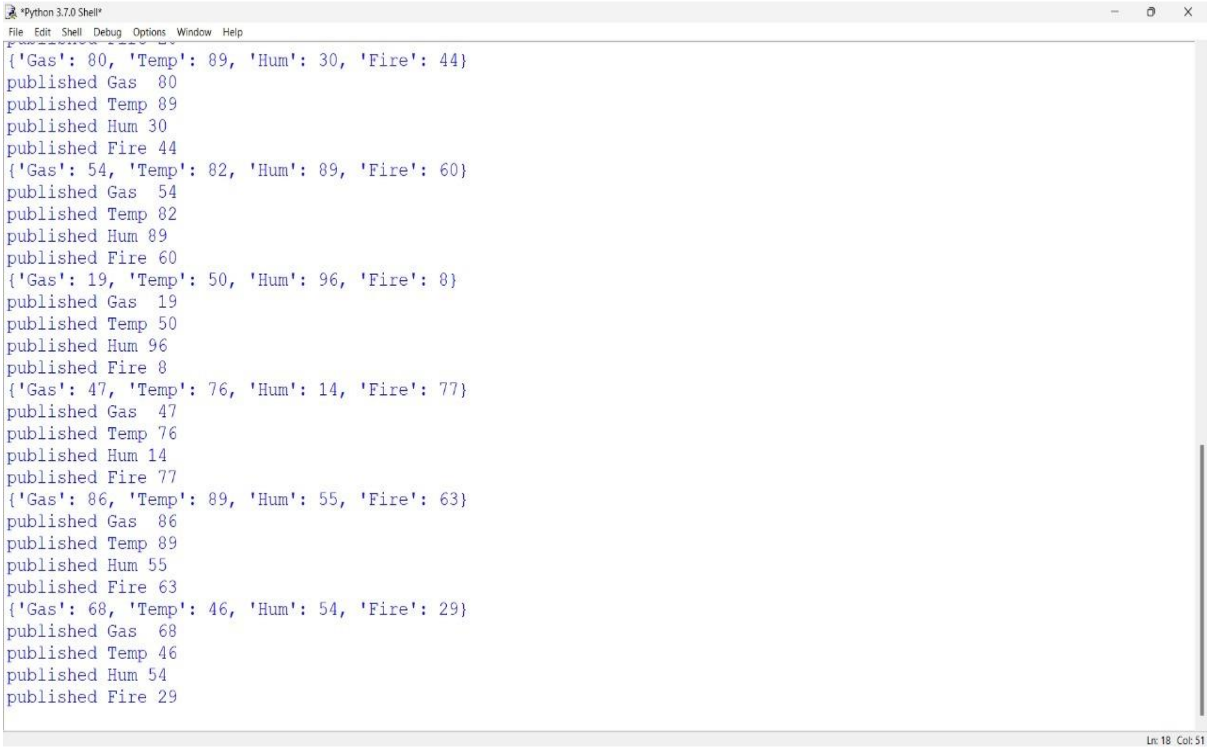
8. TESTING

8.1 Test Cases

Thus all the test cases have been passed for the designed nodes such as temperature, humidity, gas and fire sensor nodes in Node-RED Services based on the desired threshold value.

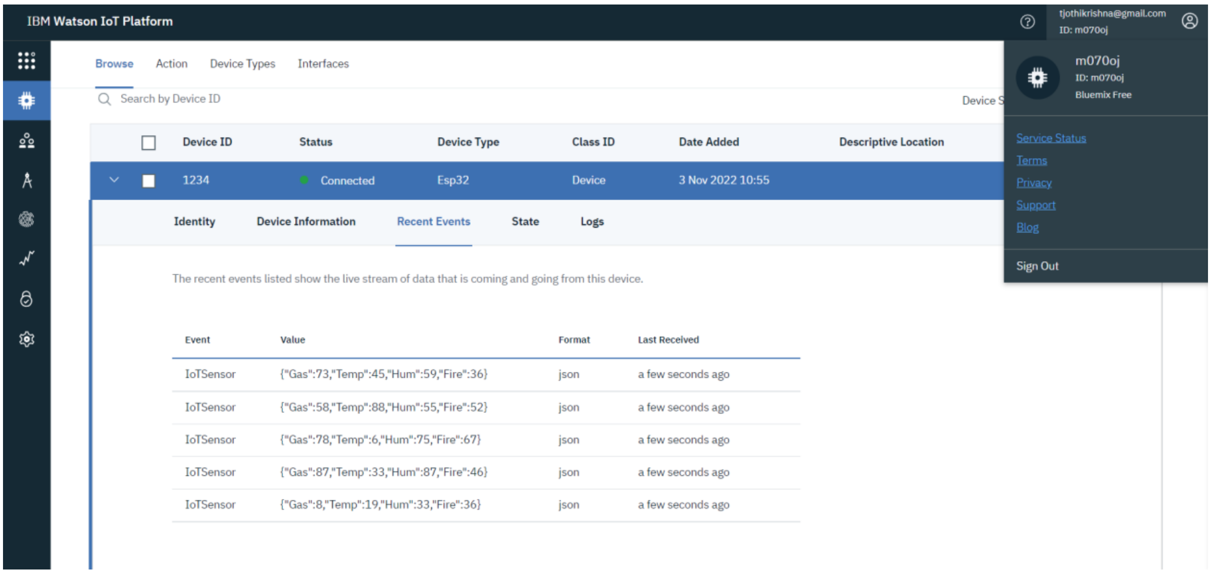
8.2 User Acceptance Testing

The output generated from Python:



```
{'Gas': 80, 'Temp': 89, 'Hum': 30, 'Fire': 44}
published Gas  80
published Temp 89
published Hum 30
published Fire 44
{'Gas': 54, 'Temp': 82, 'Hum': 89, 'Fire': 60}
published Gas  54
published Temp 82
published Hum 89
published Fire 60
{'Gas': 19, 'Temp': 50, 'Hum': 96, 'Fire': 8}
published Gas  19
published Temp 50
published Hum 96
published Fire 8
{'Gas': 47, 'Temp': 76, 'Hum': 14, 'Fire': 77}
published Gas  47
published Temp 76
published Hum 14
published Fire 77
{'Gas': 86, 'Temp': 89, 'Hum': 55, 'Fire': 63}
published Gas  86
published Temp 89
published Hum 55
published Fire 63
{'Gas': 68, 'Temp': 46, 'Hum': 54, 'Fire': 29}
published Gas  68
published Temp 46
published Hum 54
published Fire 29
```

Linking of Python and IBM Cloud Services for adding devices by using libraries in python:



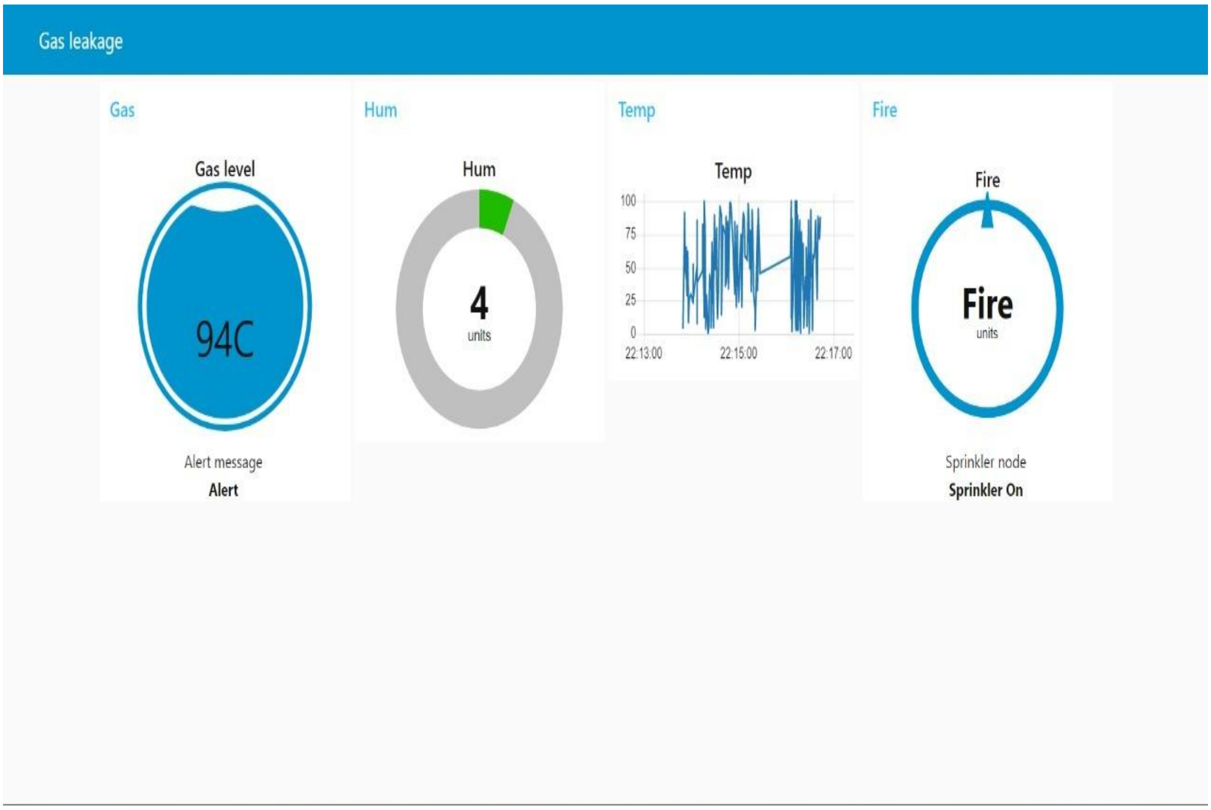
□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	1234	● Connected	Esp32	Device	3 Nov 2022 10:55	

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

Event	Value	Format	Last Received
IoTSensor	{"Gas":73,"Temp":45,"Hum":59,"Fire":36}	json	a few seconds ago
IoTSensor	{"Gas":58,"Temp":88,"Hum":55,"Fire":52}	json	a few seconds ago
IoTSensor	{"Gas":78,"Temp":6,"Hum":75,"Fire":67}	json	a few seconds ago
IoTSensor	{"Gas":87,"Temp":33,"Hum":87,"Fire":46}	json	a few seconds ago
IoTSensor	{"Gas":8,"Temp":19,"Hum":33,"Fire":36}	json	a few seconds ago

9. RESULTS

9.1 Performance Metrics



10. ADVANTAGES & DISADVANTAGES

Advantages:

1) Installation process is simple.
2) Efficient system for monitoring.

Disadvantages:

1) As it involves IOT it requires high speed internet connectivity to give accurate results.
2) There is a possibility of ignoring real signal as false alarm.

11. CONCLUSION

The goal of creating this prototype was to revolutionize environmental safety by eliminating any major or minor hazards brought on by the release of hazardous and dangerous gases into the environment. We created a Gas Leakage Detector for society using IOT technology, and it has the ability to perform data analytics on sensors and Smart Alerting techniques that send text messages to the relevant authorities. Using gas sensors, this system will be able to identify any gases present in the surrounding area. This will shield us from the main detrimental issue.

The data has been published to the IBM cloud. Thus in the python script, the values for the gas, temperature, humidity and fire have been generated and published to IBM cloud platform. This is achieved by importing the required libraries in the python script and also specifying the organization, deviceType, deviceid, authMethod and authToken to integrate with the specific cloud account, so that the data will be published to IBM cloud platform. A threshold value has been fixed for each module and if any value exceeds this threshold value, then an alert message has been generated.

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 at all times. The app can make sure to send reminders about getting the system checked every once in a while.

3. The user can add or remove the recipients who will receive the information of leakage whenever they require.

13. APPENDIX

Source Code :

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

#ibm watson device credentials

organization="griwxv"
deviceType="ESP32"
deviceid="12345678"
authMethod="token"
authToken="12345678"

#generate random values for gas leakage


def myCommandCallback(cmd):
   print ("command received: %s" %cmd.data['command'])
   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 sending data for gas leakage

deviceCli.connect()
```

```
while True:
   Gas=random.randint(0,100)
   Temp=random.randint(0,100)
   Hum=random.randint(0,100)
   Fire=random.randint(0,100)
   data={'Gas':Gas,'Temp':Temp,'Hum':Hum,'Fire':Fire}
   print(data)
   def myOnPublishCallBack():
      print("published Gas  %s " %Gas)
      print("published Temp %s " %Temp)
      print("published Hum %s " %Hum)
      print("published Fire %s " %Fire)
   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 from the cloud

deviceCli.connect()
```

GitHub & Project Demo Link:

GitHub link : https://github.com/IBM-EPBL/IBM-Project-42375-1660660800

Project Demo Link : https://node-red-utdjb-2022-11-05.eu-gb.mybluemix.net/red/#flow/0c684a1cfbcf2b26

https://node-red-utdjb-2022-11-05.eu-gb.mybluemix.net/ui/#!/0?socketid=KVz4cVMJQqIsyDoXAAAF