PROJECT REPORT

Team Members	MOHAMMED NAZEER H (TEAM LEAD) - 737819ECR107 LEKHAPRIYADHARSHINI P(TEAM MEMBER 1)- 737819ECR092 KAWIN P (TEAM MEMBER 2) ,- 737819ECR078 MOHAMMED ASFAQ(TEAM MEMBER 3) - 737819ECR105
Team ID	PNT2022TMID04619
Project Name	Project –Gas leakage monitoring and alerting system for industries

1.INTRODUCTION

1.1.Project Overview

Gas Detectors have been in the market for a very long time and have been vastly used. They have wide range of applications and can be found in industrial plants, refineries, pharmaceutical manufacturing, paper pulp mills, aircraft and ship-building facilities, wastewater treatment facilities, vehicles, indoor air quality testing and homes. There are a lot of ways in which the Gas Detectors could be characterized. They are categorized based on what type of gas they detect, what is the technology behind the making of the sensor and sometimes even the components which are used that affect their operation mechanism (semiconductors, oxidation, catalytic, photoionization, infrared, etc.). Gas Detectors are also widely characterized as fixed or portable detectors. They are characterized on the basis of which category of risk they fall in, ExOx-Tox, the three categories of risk - Ex – Risk of explosion by flammable gases - Ox – Oxygen Risk of asphyxiation by oxygen displacement Risk of increase of flammability by oxygen enrichment - Tox – Risk of poisoning by toxic gases, the list of categorization goes on. As a result we cannot have a single system or a group of systems which we can call the best but instead there is a plethora of devices available for matching the varying user requirements.

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.

2.LITERATURE REVIEW

2.1.Existing Problem

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

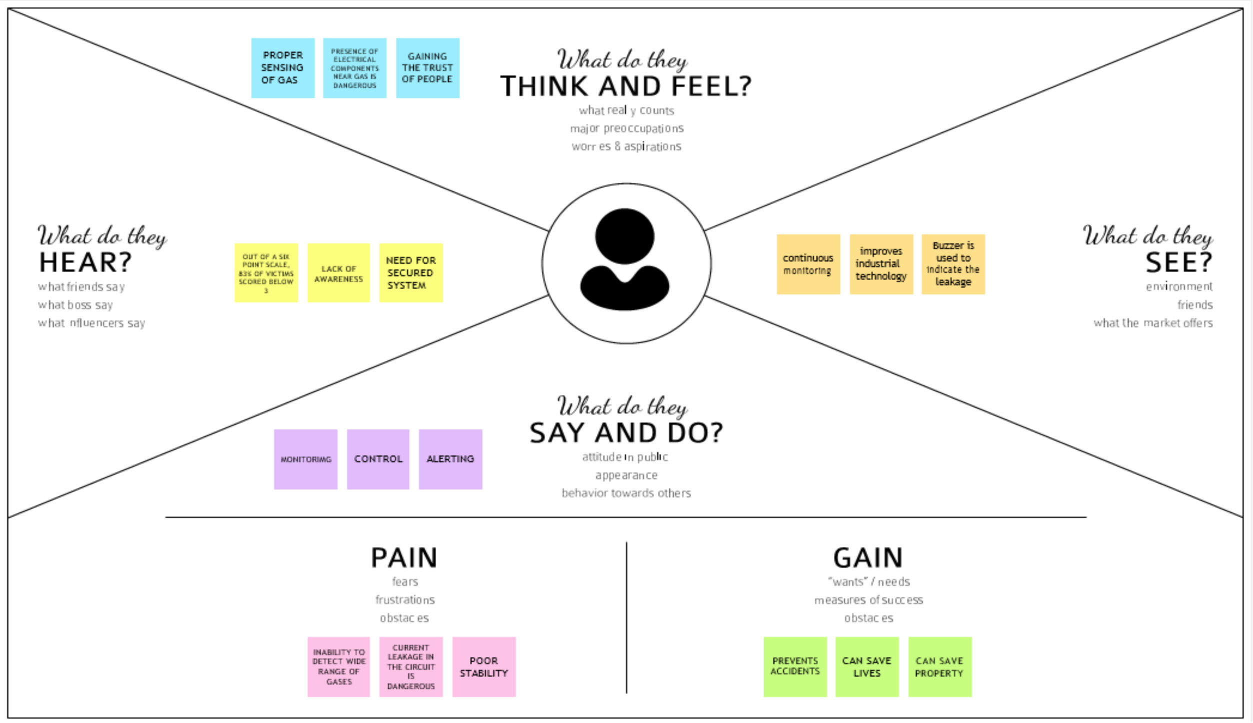
4) Makiko Kawada, Tadao Minagawa, Eiichi Nagao, Mitsuhito Kamei, Chieko Nishida and Koji Ueda, ‘Advanced Monitoring System For Gas Density Of GIS’, Mitsubishi Electric Corporation

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.

3.IDEATION & PROPOSED SOLUTION

3.1.Empathy Map Canvas



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

3.2.Ideation & Brainstorming

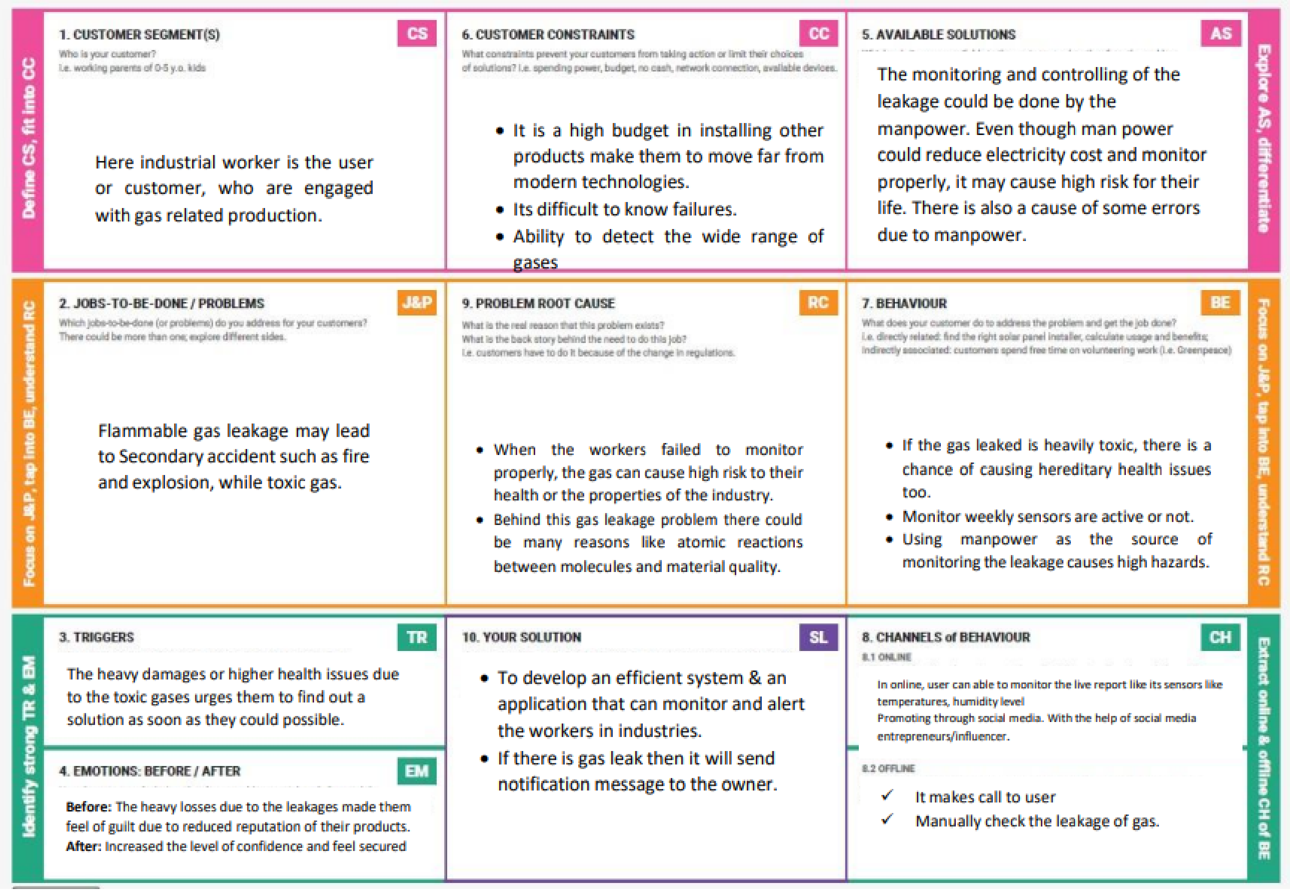
The ideas are In case of higher gas leakage and fire accidents, a notification can be given to the fire station and hospital through software application.The level of gas in the industry can be informed through speakers periodically.When gas gets leaked, a notification can be passed to hospital.Sensor can be placed in the entrance for counting the workers who have been moved out in case of emergency.

In addition to alarm, a voice notes which alerts by saying the level of leakage can be 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.

3.3.Proposed Solution

To Develop an efficient system & an application that can monitor and alert the users(workers), our product helps the industries in monitoring the emission of harmful gases. In several areas, the gas sensors will be integrated to monitor the gas leakage. If in any area gas leakage is detected the admins will be notified along with the location. In the web application, admins can view the sensor parameters. It is fastest alerts to the workers and user friendly. For social impact it is Cost efficient and easy installation and provide efficient results and can work with irrespective of fear.Since the product is cost efficient, it can be placed in many places in the industries. Even when the gas leakage is more, the product sense the accurate values and alerts the workers effectively.

3.4.Problem Solution fit



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Here industrial worker is the user or customer, who are engaged with gas related production.

6. CUSTOMER CONSTRAINTS 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 is a high budget in installing other products make them to move far from modern technologies.
- Its difficult to know failures.
- Ability to detect the wide range of gases

5. AVAILABLE SOLUTIONS AS

The monitoring and controlling of the leakage could be done by the manpower. Even though man power could reduce electricity cost and monitor properly, it may cause high risk for their life. There is also a cause of some errors due to manpower.

Explore AS, differentiate

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

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 accident such as fire and explosion, while toxic gas.

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.

- When the workers failed to monitor properly, the gas can cause high risk to their health or the properties of the industry.
- Behind this gas leakage problem there could be many reasons like atomic reactions between molecules and material quality.

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)

- If the gas leaked is heavily toxic, there is a chance of causing hereditary health issues too.
- Monitor weekly sensors are active or not.
- Using manpower as the source of monitoring the leakage causes high hazards.

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

Identify strong TR & EM

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

- To develop an efficient system & an application that can monitor and alert the workers in industries.
- If there is gas leak then it will send notification message to the owner.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

In online, user can able to monitor the live report like its sensors like temperatures, humidity level
Promoting through social media. With the help of social media entrepreneurs/influencer.

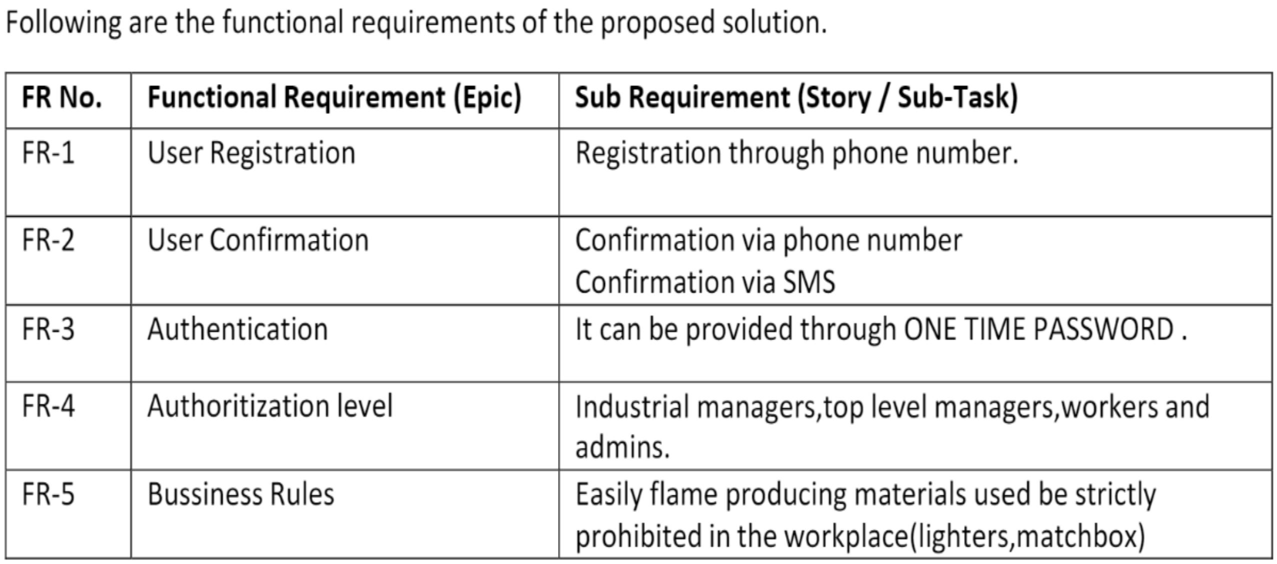
8.2 OFFLINE

✓ It makes call to user
✓ Manually check the leakage of gas.

Extract online & offline CH of BE

4.REQUIREMENT ANALYSIS

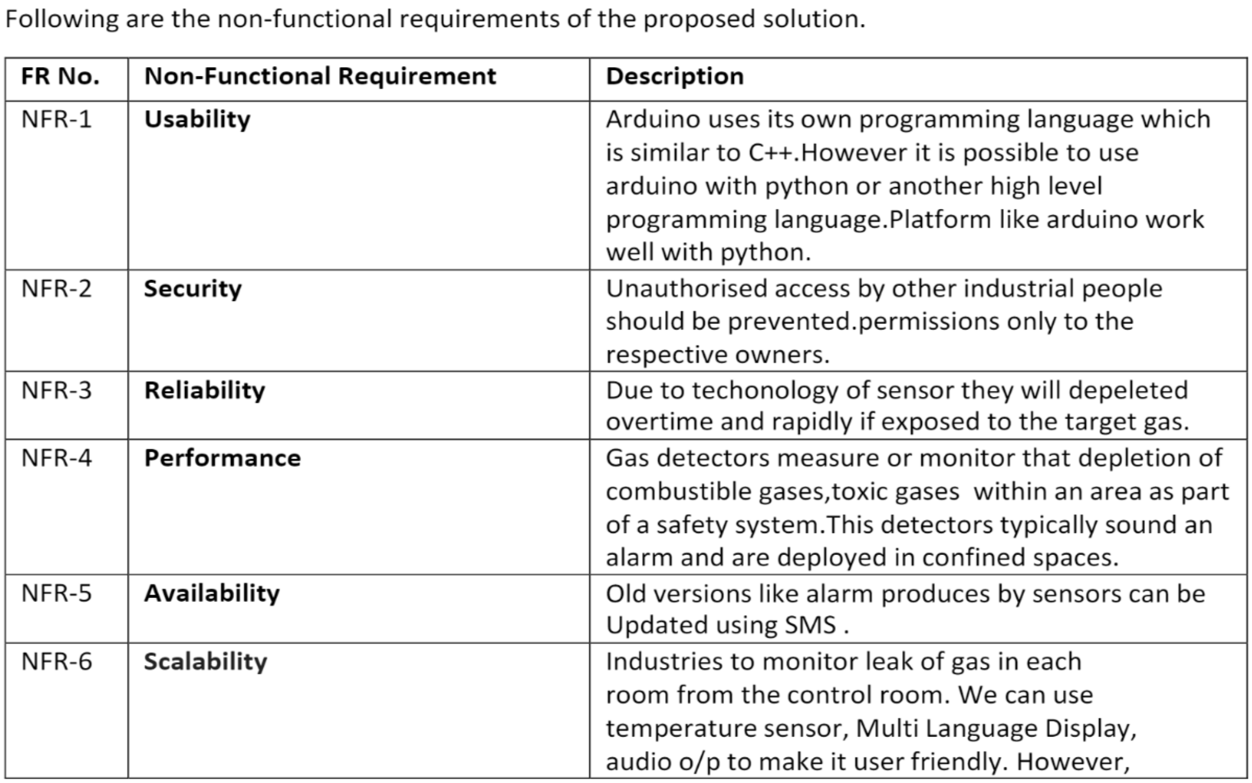
4.1.Functional Requirement



Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through phone number.
FR-2	User Confirmation	Confirmation via phone number Confirmation via SMS
FR-3	Authentication	It can be provided through ONE TIME PASSWORD .
FR-4	Authoritization level	Industrial managers,top level managers,workers and admins.
FR-5	Bussiness Rules	Easily flame producing materials used be strictly prohibited in the workplace(lighters,matchbox)

4.2.Non-Functional Requirement

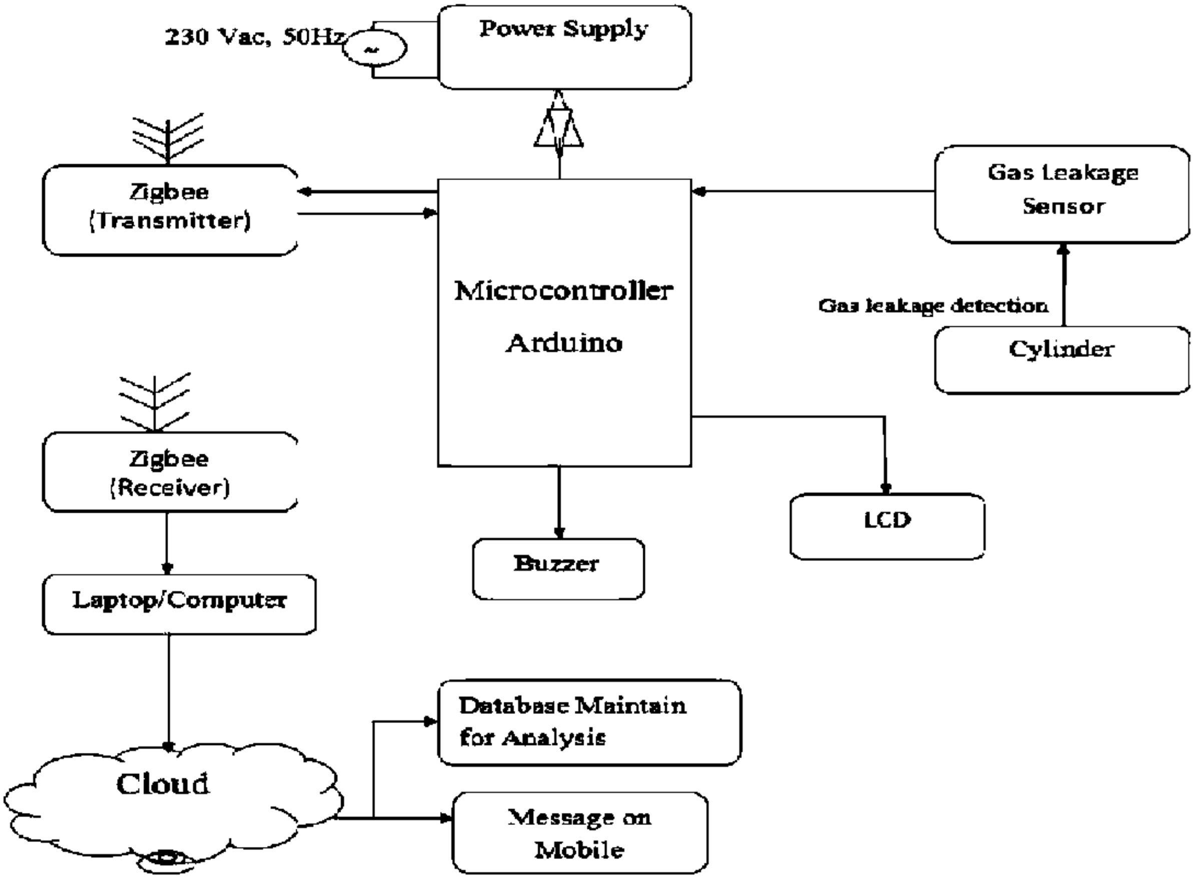


Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Arduino uses its own programming language which is similar to C++.However it is possible to use arduino with python or another high level programming language.Platform like arduino work well with python.
NFR-2	**Security**	Unauthorised access by other industrial people should be prevented.permissions only to the respective owners.
NFR-3	**Reliability**	Due to techonology of sensor they will depeleted overtime and rapidly if exposed to the target gas.
NFR-4	**Performance**	Gas detectors measure or monitor that depletion of combustible gases,toxic gases within an area as part of a safety system.This detectors typically sound an alarm and are deployed in confined spaces.
NFR-5	**Availability**	Old versions like alarm produces by sensors can be Updated using SMS .
NFR-6	**Scalability**	Industries to monitor leak of gas in each room from the control room. We can use temperature sensor, Multi Language Display, audio o/p to make it user friendly. However,

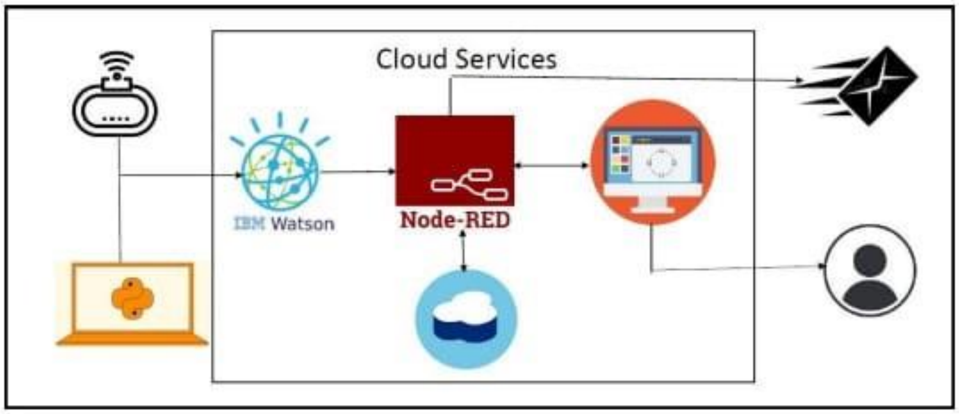
5.PROJECT DESIGN

5.1.Data Flow Diagram



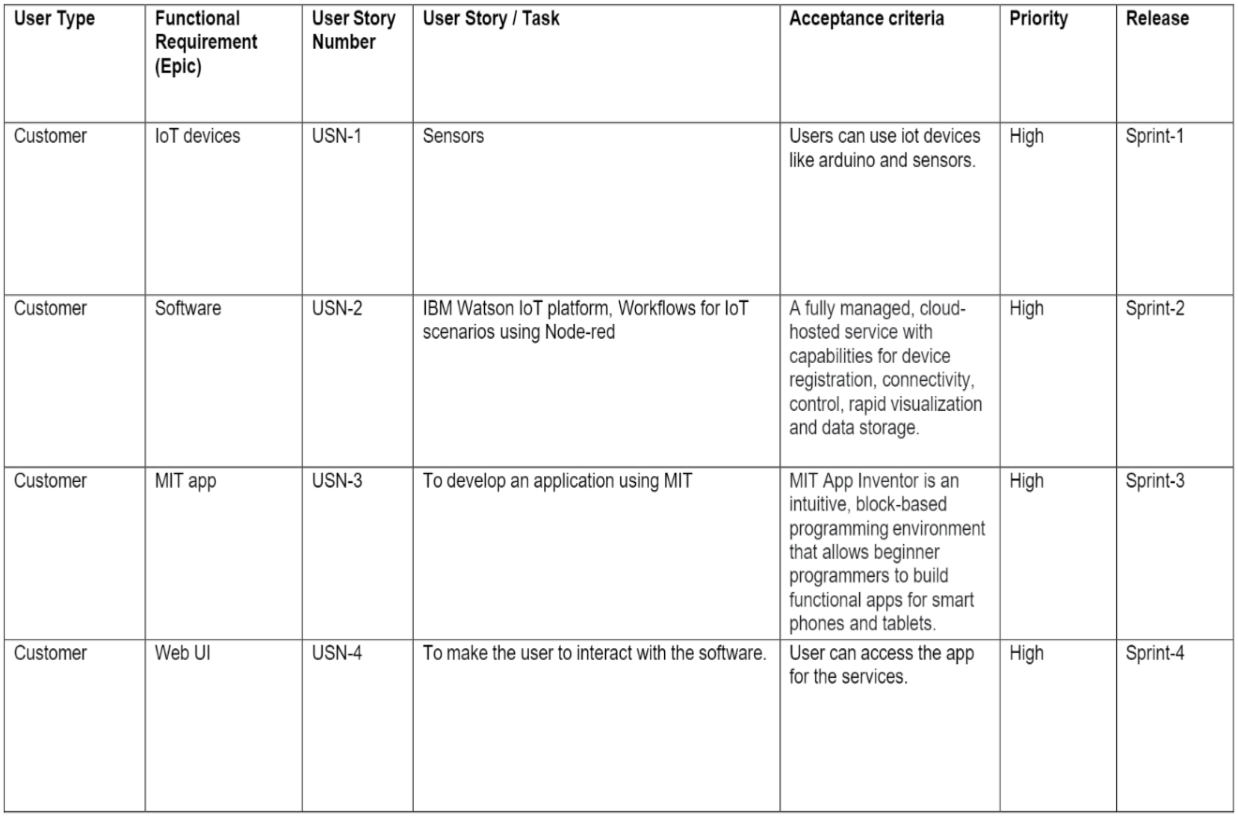
This is the data flow diagram of gas leakage monitoring and detection. Here the data from temperature sensor and gas sensor is collected from IOT device and the data is analyzed. If the alert action requires it alerts and the required measures are taken.

5.2. Solution & Technical Diagram



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

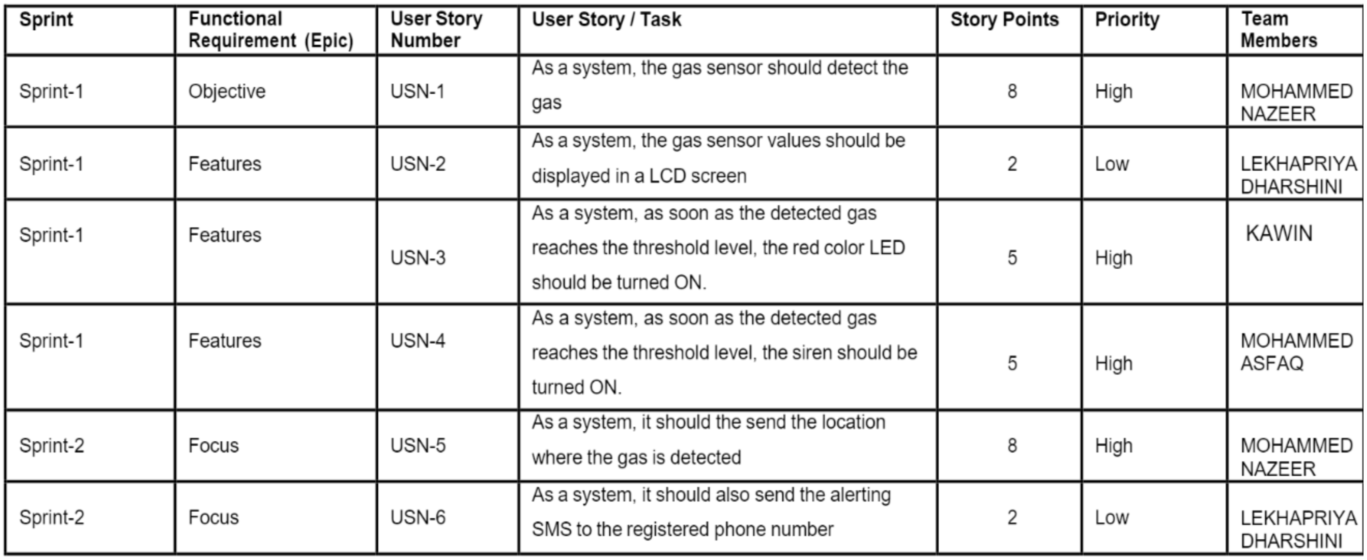
5.3.User Stories



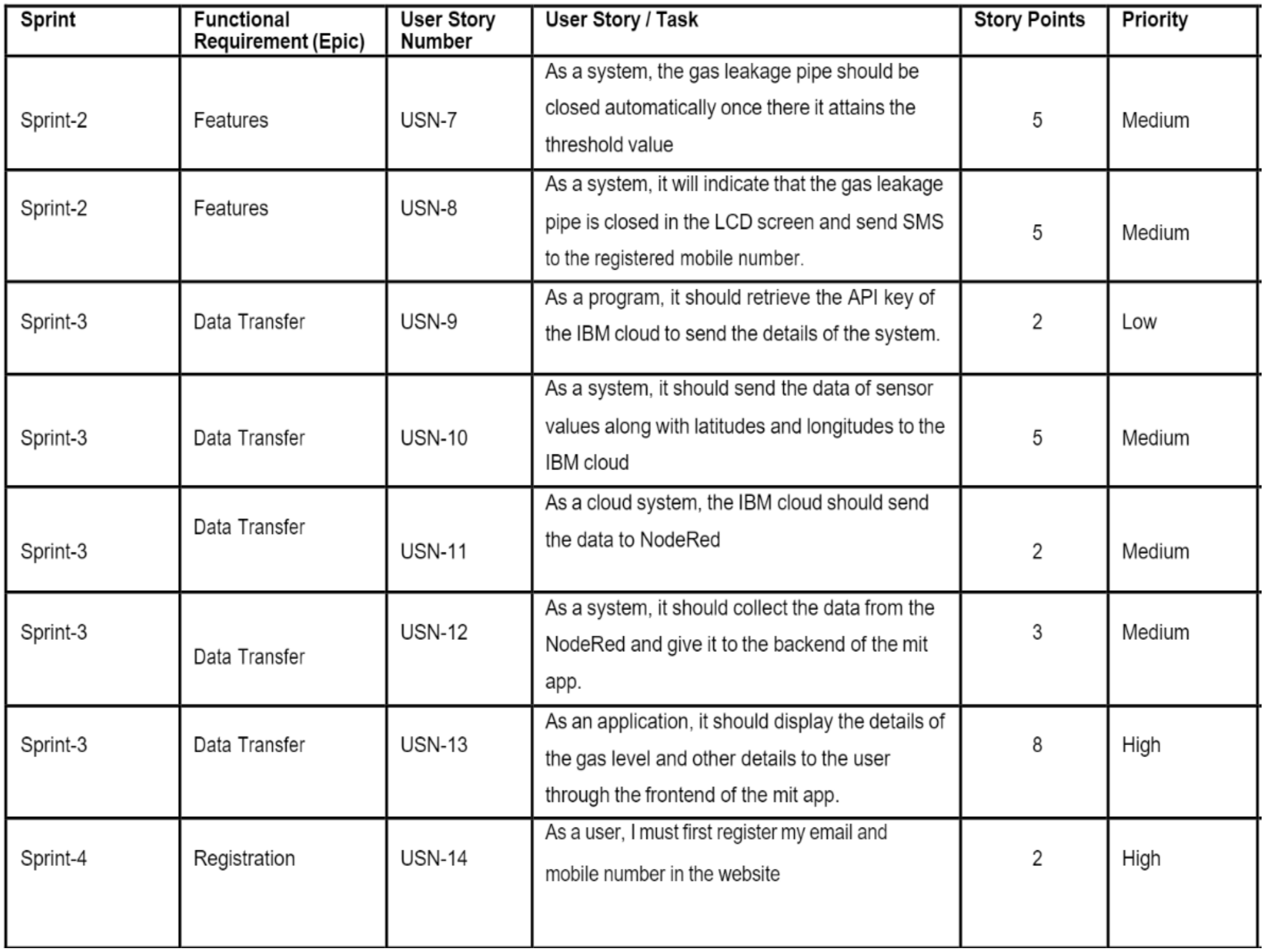
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	IoT devices	USN-1	Sensors	Users can use iot devices like arduino and sensors.	High	Sprint-1
Customer	Software	USN-2	IBM Watson IoT platform, Workflows for IoT scenarios using Node-red	A fully managed, cloud-hosted service with capabilities for device registration, connectivity, control, rapid visualization and data storage.	High	Sprint-2
Customer	MIT app	USN-3	To develop an application using MIT	MIT App Inventor is an intuitive, block-based programming environment that allows beginner programmers to build functional apps for smart phones and tablets.	High	Sprint-3
Customer	Web UI	USN-4	To make the user to interact with the software.	User can access the app for the services.	High	Sprint-4

6.PROJECT PLANNING & SCHEDULING

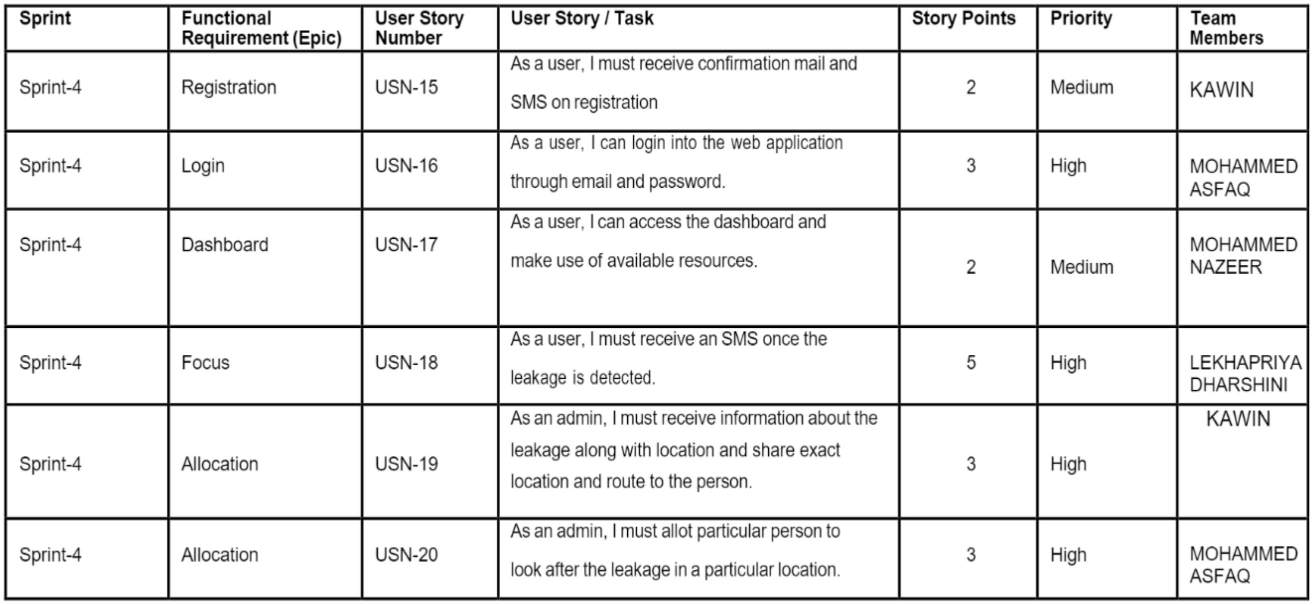
6.1.Sprint Planning & Estimation



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Objective	USN-1	As a system, the gas sensor should detect the gas	8	High	MOHAMMED NAZEER
Sprint-1	Features	USN-2	As a system, the gas sensor values should be displayed in a LCD screen	2	Low	LEKHAPRIYA DHARSHINI
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	KAWIN
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	MOHAMMED ASFAQ
Sprint-2	Focus	USN-5	As a system, it should the send the location where the gas is detected	8	High	MOHAMMED NAZEER
Sprint-2	Focus	USN-6	As a system, it should also send the alerting SMS to the registered phone number	2	Low	LEKHAPRIYA DHARSHINI

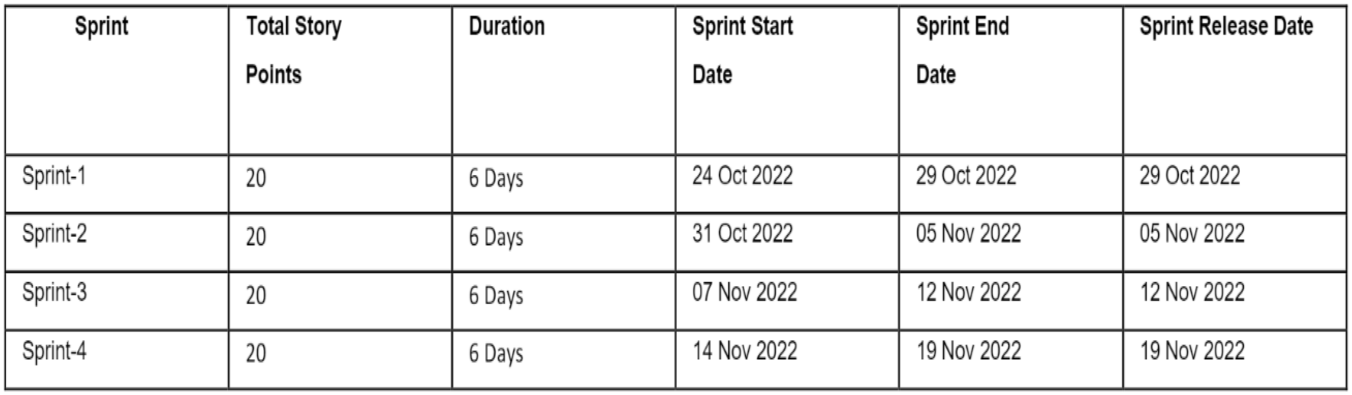


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-2	Features	USN-7	As a system, the gas leakage pipe should be closed automatically once there it attains the threshold value	5	Medium
Sprint-2	Features	USN-8	As a system, it will indicate that the gas leakage pipe is closed in the LCD screen and send SMS to the registered mobile number.	5	Medium
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
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
Sprint-3	Data Transfer	USN-11	As a cloud system, the IBM cloud should send the data to NodeRed	2	Medium
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
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
Sprint-4	Registration	USN-14	As a user, I must first register my email and mobile number in the website	2	High

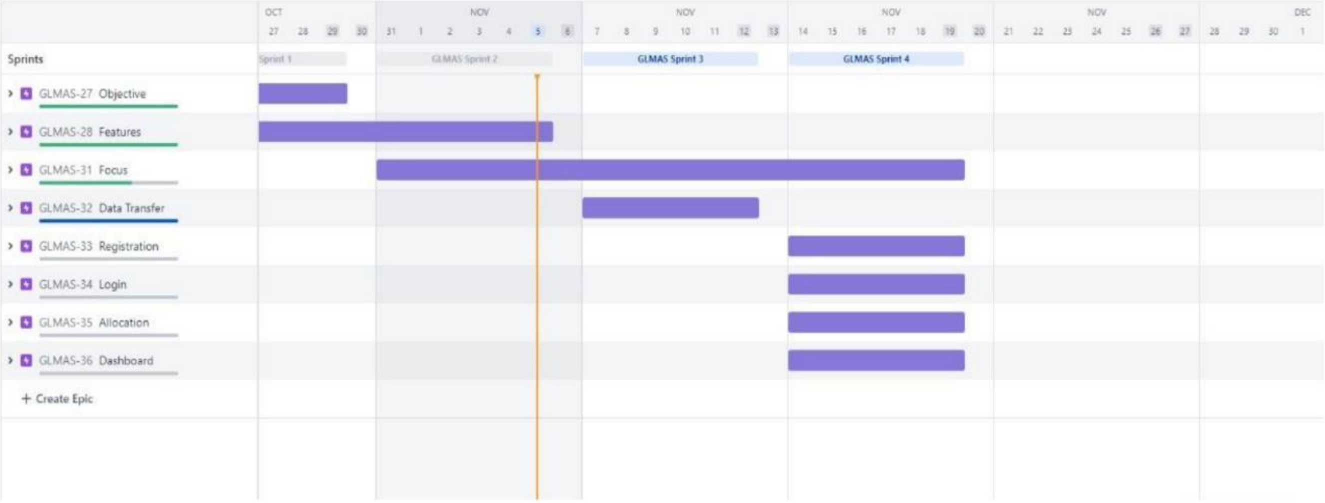


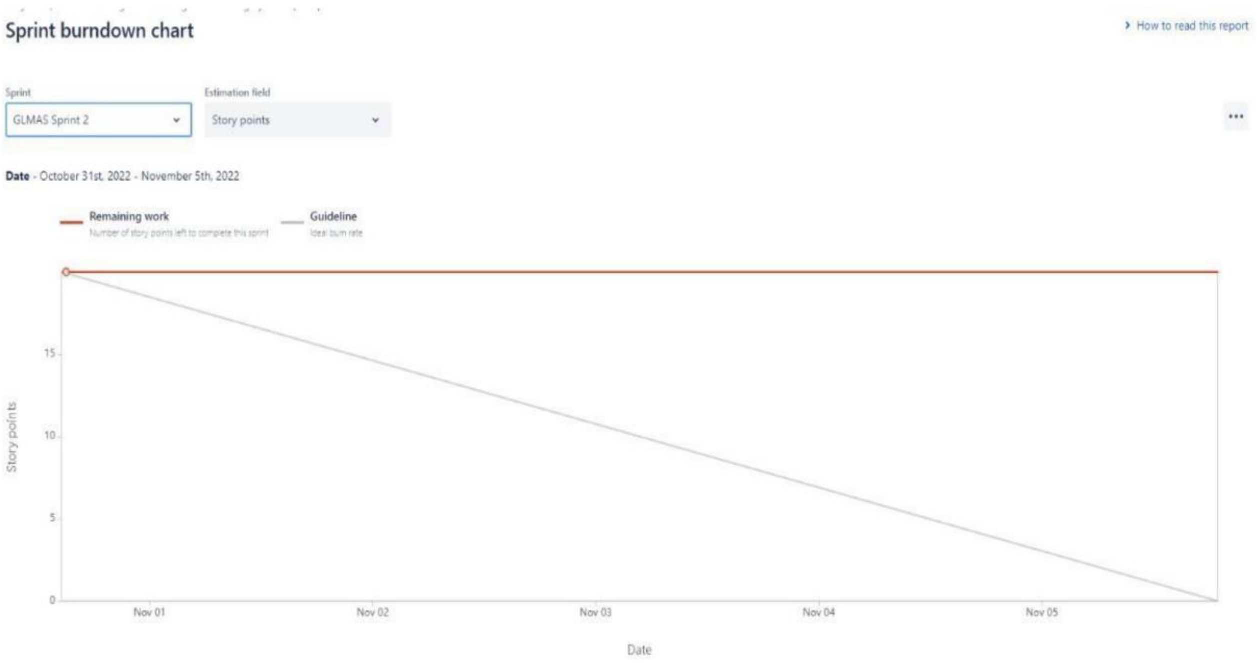
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Registration	USN-15	As a user, I must receive confirmation mail and SMS on registration	2	Medium	KAWIN
Sprint-4	Login	USN-16	As a user, I can login into the web application through email and password.	3	High	MOHAMMED ASFAQ
Sprint-4	Dashboard	USN-17	As a user, I can access the dashboard and make use of available resources.	2	Medium	MOHAMMED NAZEER
Sprint-4	Focus	USN-18	As a user, I must receive an SMS once the leakage is detected.	5	High	LEKHAPRIYA DHARSHINI
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	KAWIN
Sprint-4	Allocation	USN-20	As an admin, I must allot particular person to look after the leakage in a particular location.	3	High	MOHAMMED ASFAQ

6.2.SPRINT DELIVERY SCHEDULE



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





Sprint burndown chart

Sprint: GLMAS Sprint 2

Estimation field: Story points

Date - October 31st, 2022 - November 5th, 2022

7.CODING & SOLUTIONING

7.1.Feature 1

Code:

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


#Provide your IBM Watson Device Credentials
organization = "pi0ywk"
deviceType = "Gas_Geakage_Detector"
import random


#Provide your IBM Watson Device Credentials
organization = "pi0ywk"
deviceType = "Gas_Geakage_Detector"




deviceId = "nazeer007"
authMethod = "token"
authToken = "8148922991"
```

```
# Initialize GPIO


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status == "alarmon":
        print ("Alarm is on please all Evacuate Fans On")
    elif status == "alarmoff":
        print ("Alarm is off and Fans Off")
    elif status == "sprinkleron":
        print ("Sprinkler is On Evacuate Faster")
    elif status == "sprinkleroff":
        print("Sprinkler is Off")
    else:
        print("Please send proper 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 send a datapoint "hello" with value "world" into the cloud
as an event of type "greeting" 10 times
    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 random function


        temp=random.randint(0,120)
        Humid=random.randint(0,100)
        gas=random.randint(0,1500)
        data={'temp':temp,'Humid':Humid,'gas':gas}
        #print data
        def myOnPublishCallback():
            print (" Published Temperature = %s C" % temp, "Humidity = %s
%%" % Humid, "Gas_Level = %s ppm" %gas, "to IBM Watson")


        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("\n Not connected to IoTF")
        if temp>60 :
            print("\n Fire Detected due to gas Leak ! Alarm ON! Sprinkler
ON! Call The Fire Police \n")
        elif gas>350:
            print("\n Gas is Leaking \n")


        time.sleep(10)


        deviceCli.commandCallback = myCommandCallback


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

In this code:

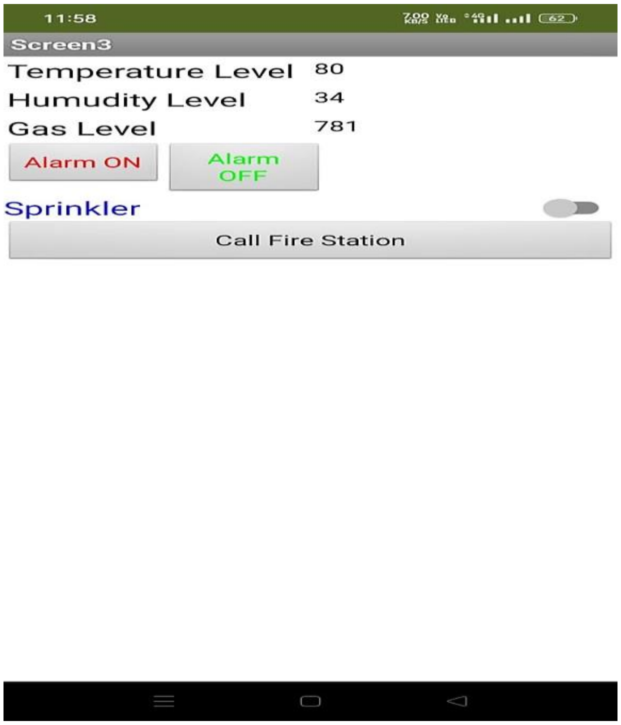
```
           if temp>60 :
            print("\n Fire Detected due to gas Leak ! Alarm ON! Sprinkler
ON! Call The Fire Police \n")
```

```
elif gas>350:
 print("\n Gas is Leaking \n")
```

if temperature is greater than 60 the fire could be occured ,then the alaram will be on to alert and the sprinklers on . else if gas is grater than 350 then the gas leakage indication only occur.

7.2.Feature 2

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



Temperature Level 80
Humudity Level 34
Gas Level 781

8.TESTING

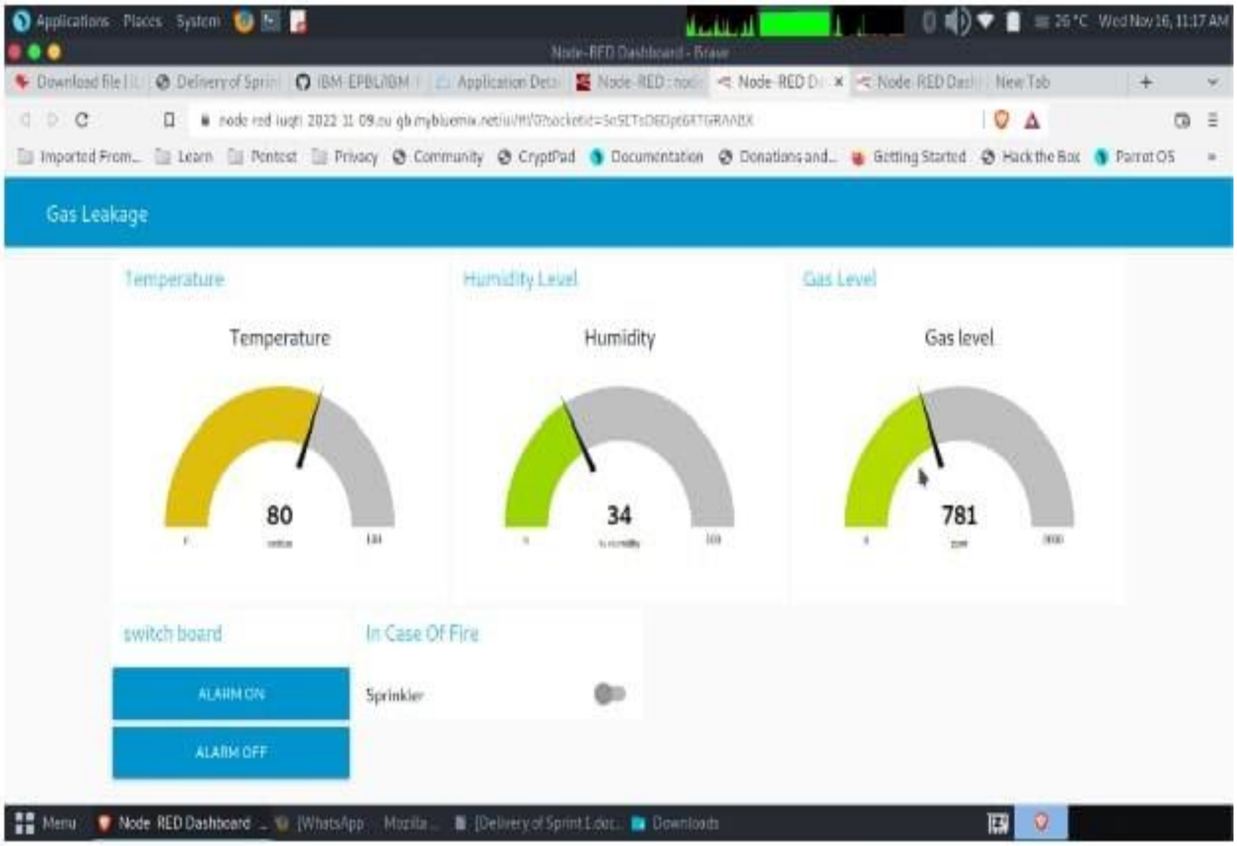
8.1.Test Cases

ALL PASSED

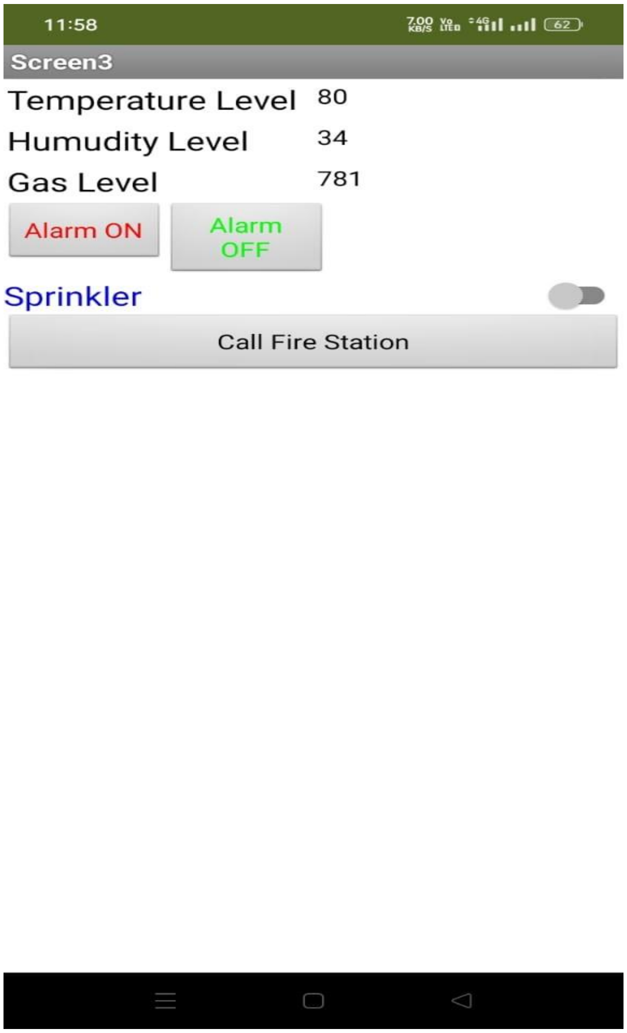
9.RESULTS

9.1.Performance Metrics

Below image represents the result of node red dash board



The next Image below here represents the output of MIT-inventer :



Screen3

Temperature Level 80

Humudity Level 34

Gas Level 781

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 AND FUTURE SCOPE

The proposed gas leakage detector is promising in the field of safety. The attempt while making this prototype has been to bring a revolution in the field of safety against the leakage of harmful and toxic gases to minimize and hence nullify any major or minor hazard being caused due to them. Nevertheless there is always scope of improvement and some of the features that will improve the system and make it even better and reliable have been mentioned below.

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

14.APPENDIX

Source Code

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


#Provide your IBM Watson Device Credentials
organization = "pi0ywk"
deviceType = "Gas_Geakage_Detector"
deviceId  =  "Udayakpr007"
authMethod = "token"
authToken = "8148922991"

# Initialize GPIO


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status == "alarmon":
        print ("Alarm is on please all Evacuate Fans On")
    elif status == "alarmoff":
        print ("Alarm is off and Fans Off")
    elif status == "sprinkleron":
        print ("Sprinkler is On Evacuate Faster")
    elif status == "sprinkleroff":
        print("Sprinkler is Off")
    else:
        print("Please send proper 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 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 random function

        temp=random.randint(0,120)
        Humid=random.randint(0,100)
        gas=random.randint(0,1500)
        data={'temp':temp,'Humid':Humid,'gas':gas}
        #print data
        def myOnPublishCallback():
            print (" Published Temperature = %s C" % temp, "Humidity = %s
%%" % Humid, "Gas_Level = %s ppm" %gas, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("\n Not connected to IoTF")
        if temp>60 :
            print("\n Fire Detected due to gas Leak ! Alarm ON! Sprinkler ON! Call The Fire Police \n")
        elif gas>350:
            print("\n Gas is Leaking \n")

        time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

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

Github

GITHUB :

https://github.com/IBM-EPBL/IBM-Project-22436-1659851760

Node RedDashBoard : https://node-red-iutti-2022-11-09.eugb.mybluemix.net/ui/#!/0?socketid=6qWlVhcruiLeMbqEABR