GAS LEAKAGE MONITORING AND ALERTING SYSTEM

TEAM ID: PNT2022TMID20329

TEAM LEADER : SURIYA MANIKANDAN S(49621911018)

TEAM MEMBER: SOBANA A(49621911104)

TEAM MEMBER: RAJA SOFIA(49621911042)

TEAM MEMBER: PANDI SELVA KUMAR(49621911410)

ABSTRACT

A gas leakage refers to a leakage of natural gas or another gaseous product from a pipeline or other containment into any area where the gas should not be present. Gas leaks can be hazardous to health as well as the environment.The explosion due to gas leakage has become a serious problem in our country's daily activities. Now the world is evolving with technology, so it is necessary to use technology if possible in every case. To resolve the accident occurred we can prevent it through technology. The system is based on a microcontroller, which uses gas sensors as well as GSM, display and buzzer. It is designed for Gas Leakage Monitoring and Alert System using Arduino Mega with MQ Series Gas Sensor. This circuit contains MQ Series gas sensor, microcontroller, buzzer, display and GSM. The sensor will detect the gas leakage and transmit the information to the microcontroller. On the basis of those information, the microcontroller makes a decision and then displays a warning message on the display and the message will be sent to the user via GSM. The uses of the Arduino microcontroller with Arduino, provide a suitable platform for implementing an embedded control system and it is possible to modify it to meet our future requirements easily and quickly.

INTRODUCTION

PROJECT OVERVIEW

This project helps the industries in monitoring the emission of harmful gases. In several areas, the integration of gas sensors helps in monitoring the gas leakage. If in any area gas leakage is detected the admins will be notified along with the location. In the web application, admins can view the sensor parameters.

PURPOSE

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

LITERATURE SURVEY

1.TITLE: LPG monitoring and leakage detection system

AUTHOR: Shruthi Unnikrishnan; Mohammed Razil;JoshuaBenny; Shelvin Varghese; C. V. Hari

YEAR OF PUBLICATION: 2017

PUBLISHED IN: 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)

ABSTRACT: In this paper, we have proposed a Liquefied Petroleum Gas (LPG) monitoring and leakage detection system. With the large demand and use of LPG, this system would be helpful to monitor the usage of LPG on a regular basis and to alert about any hazards that may occur due to LPG leakage. We have designed a system that alerts the user of the amount of LPG left so that appropriate measures can be taken. Since LPG is a highly hazardous and inflammable gas, we have also designed a system to alert the user with an alarm when there is a leakage of LPG so that measures are taken to avoid an explosion.

2. TITLE: On-line gas monitoring for increased transformer protection

AUTHOR: Mickel Saad;Ed teNyenhuis

YEAR OF PUBLICATION: 2017

PUBLISHED IN: 2017 IEEE Electrical Power and Energy Conference (EPEC)

ABSTRACT : Transformers are vital and high cost components of the electric power system. On-line monitoring is a tool that can assess the condition of these valuable assets in real time. The importance of a monitored transformer and the economic consequences of a failure, provide the basis for selecting a monitor with its potential cost benefits. The most basic and important transformer monitoring equipment is the on-line Dissolved Gas Analysis (DGA) monitor. This paper discusses the importance of on-line DGA monitoring for power transformers, types of available sensors and various qualities to examine when choosing a monitoring device.

3.TITLE: Smart Level Monitoring, Booking & Detector over IoT

AUTHOR: Kumar Keshamoni;Sabbani Hemanth

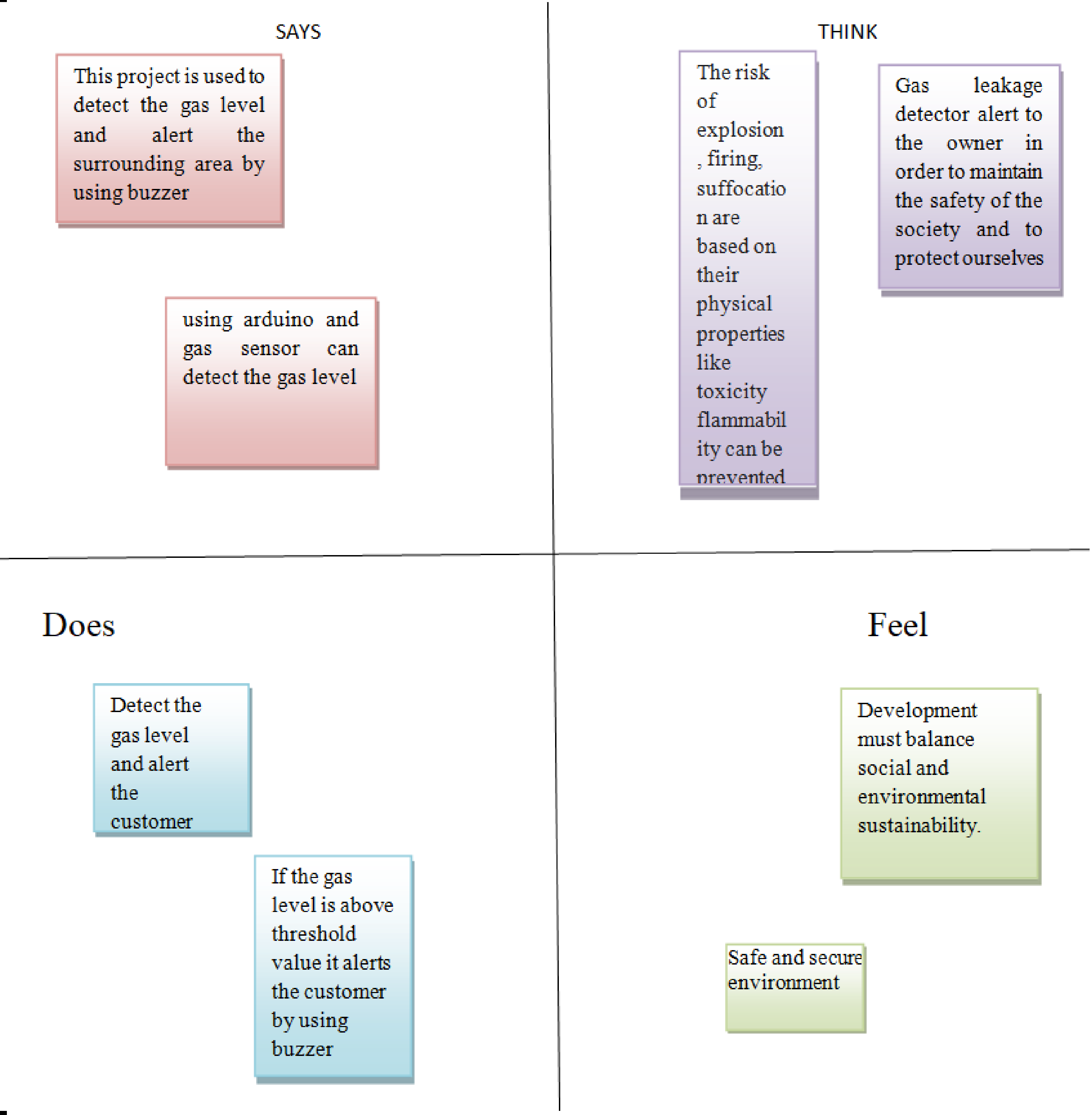
YEAR OF PUBLICATION: 2017

PUBLISHED IN: 2017 IEEE 7th International Advance Computing Conference (IACC)

ABSTRACT: This journal explains about the most common problem experienced in our day-to-day lives that is regarding GAS container going empty. We bring this paper to create awareness about the reducing weight of the gas in the container, and to place a gas order using IOT. The gas booking/order is being done with the help IOT and that the continuous weight measurement is done using a load cell which is interfaced with a Microcontroller (to compare with an ideal value). For ease it is even has a been added with an RF TX & Rx modules which will give the same information. When it comes it to security of the kit as well as gas container we have an MQ-2(gas sensor), LM 35(temperature sensor), which will detect the surrounding environment for any chance of error.

IDEATION & PROPOSED SOLUTION:

EMPATHY MAPPING



SAYS

This project is used to detect the gas level and alert the surrounding area by using buzzer

using arduino and gas sensor can detect the gas level

THINK

The risk of explosion , firing, suffocation are based on their physical properties like toxicity flammability can be prevented

Gas leakage detector alert to the owner in order to maintain the safety of the society and to protect ourselves

Does

Detect the gas level and alert the customer

If the gas level is above threshold value it alerts the customer by using buzzer

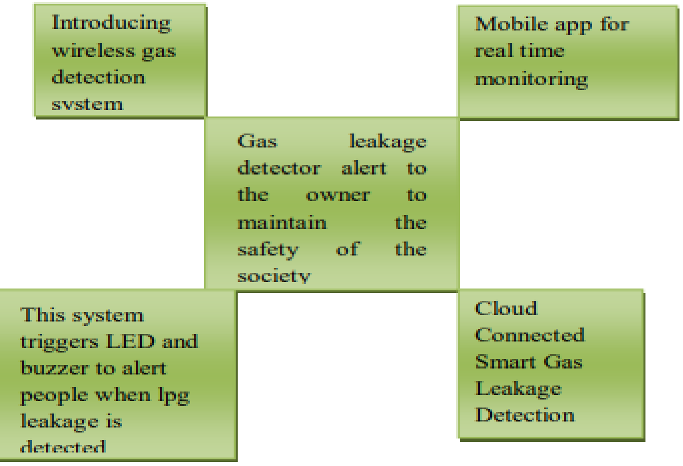
Feel

Development must balance social and environmental sustainability.

Safe and secure environment

4. BRAIN STORMING

SURIYA MANIKANDAN



Introducing wireless gas detection svstem

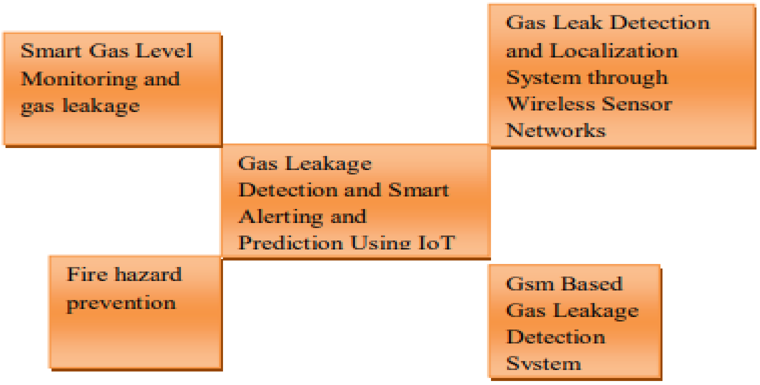
Mobile app for real time monitoring

Gas leakage detector alert to the owner to maintain the safety of the society

This system triggers LED and buzzer to alert people when lpg leakage is detected

Cloud Connected Smart Gas Leakage Detection

SOBANA



Smart Gas Level Monitoring and gas leakage

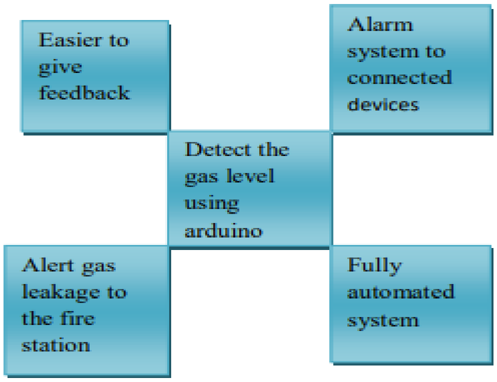
Gas Leak Detection and Localization System through Wireless Sensor Networks

Gas Leakage Detection and Smart Alerting and Prediction Using IoT

Fire hazard prevention

Gsm Based Gas Leakage Detection Svstem

PANDI SELVAKUMAR



Easier to give feedback

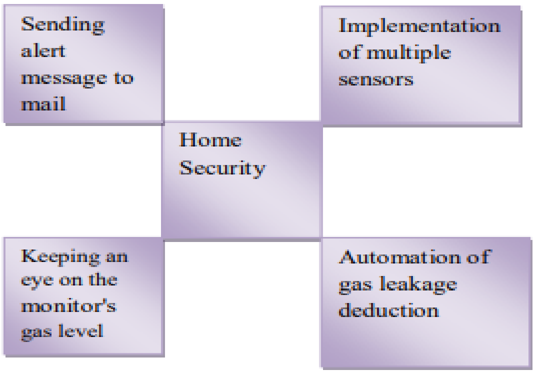
Alarm system to connected devices

Detect the gas level using arduino

Alert gas leakage to the fire station

Fully automated system

RAJASOFIA



Sending alert message to mail

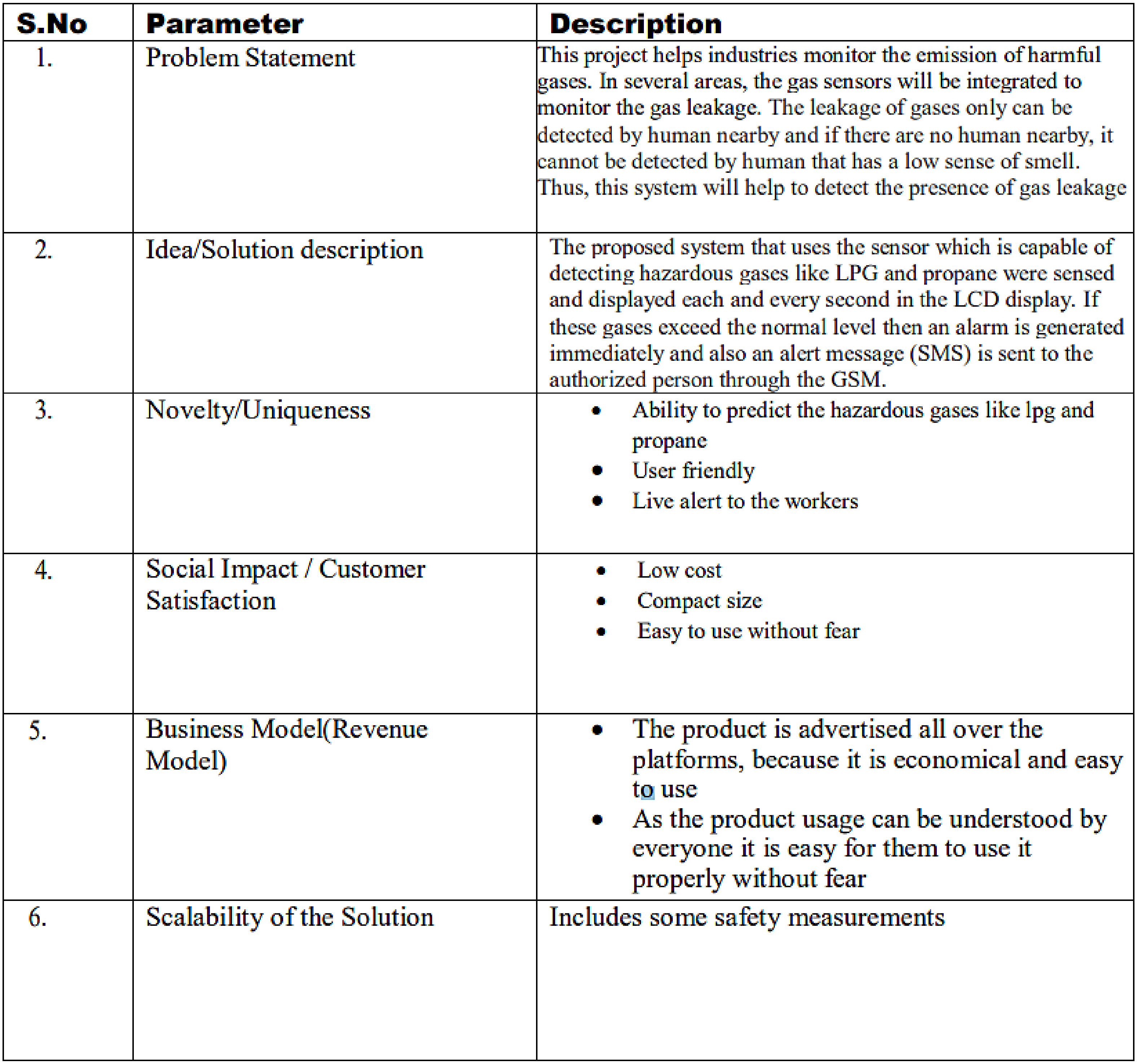
Implementation of multiple sensors

Home Security

Keeping an eye on the monitor's gas level

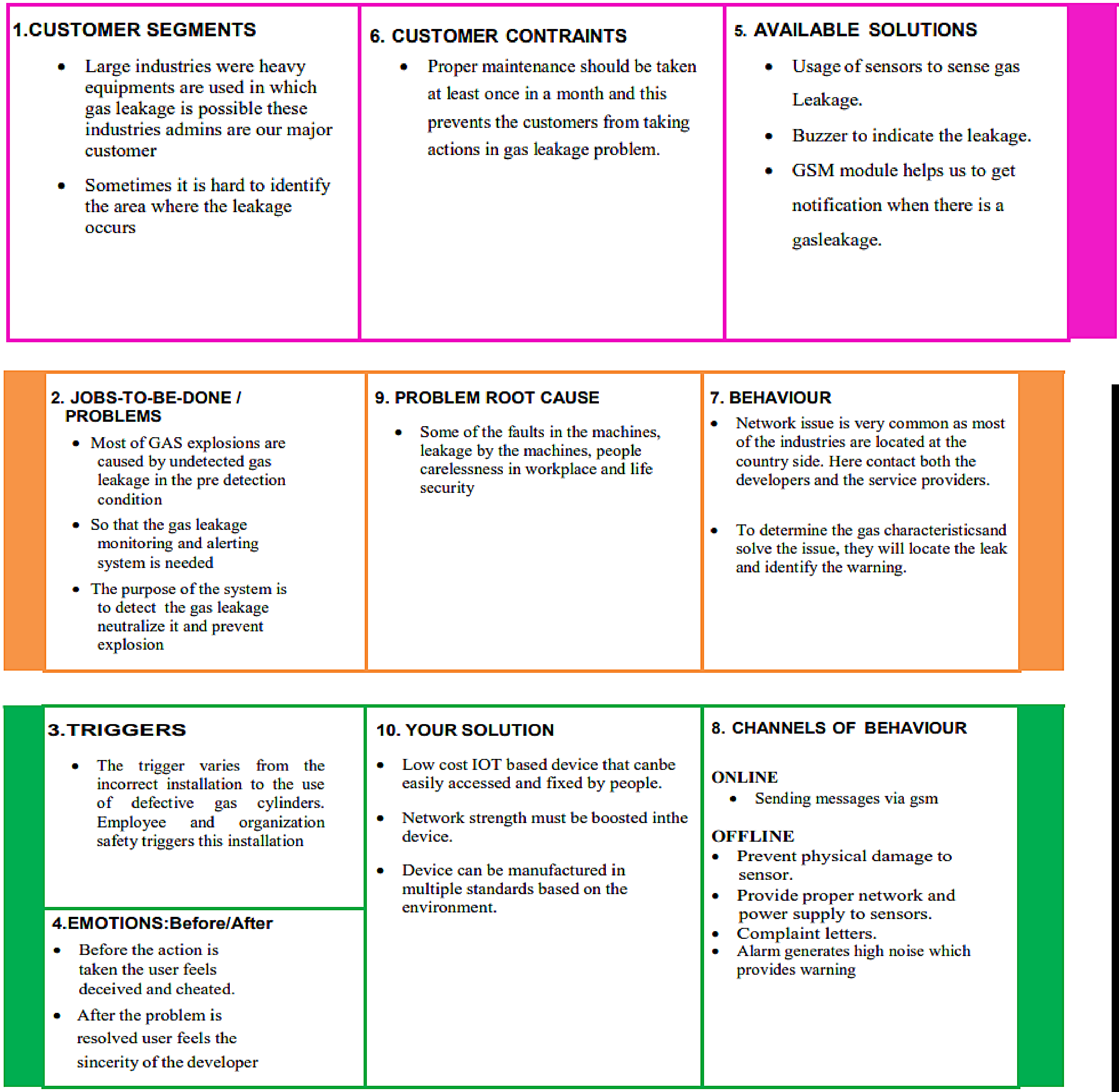
Automation of gas leakage deduction

PROPOSED SOLUTION



S.No	Parameter	Description
1.	Problem Statement	This project helps industries monitor the emission of harmful gases. In several areas, the gas sensors will be integrated to monitor the gas leakage. The leakage of gases only can be detected by human nearby and if there are no human nearby, it cannot be detected by human that has a low sense of smell. Thus, this system will help to detect the presence of gas leakage
2.	Idea/Solution description	The proposed system that uses the sensor which is capable of detecting hazardous gases like LPG and propane were sensed and displayed each and every second in the LCD display. If these gases exceed the normal level then an alarm is generated immediately and also an alert message (SMS) is sent to the authorized person through the GSM.
3.	Novelty/Uniqueness	• Ability to predict the hazardous gases like lpg and propane • User friendly • Live alert to the workers
4.	Social Impact / Customer Satisfaction	• Low cost • Compact size • Easy to use without fear
5.	Business Model(Revenue Model)	• The product is advertised all over the platforms, because it is economical and easy to use • As the product usage can be understood by everyone it is easy for them to use it properly without fear
6.	Scalability of the Solution	Includes some safety measurements

PROBLEM SOLUTION FIT



1.CUSTOMER SEGMENTS

- Large industries were heavy equipments are used in which gas leakage is possible these industries admins are our major customer
- Sometimes it is hard to identify the area where the leakage occurs

6. CUSTOMER CONTRAINTS

- Proper maintenance should be taken at least once in a month and this prevents the customers from taking actions in gas leakage problem.

5. AVAILABLE SOLUTIONS

- Usage of sensors to sense gas Leakage.
- Buzzer to indicate the leakage.
- GSM module helps us to get notification when there is a gasleakage.

2. JOBS-TO-BE-DONE / PROBLEMS

- Most of GAS explosions are caused by undetected gas leakage in the pre detection condition
- So that the gas leakage monitoring and alerting system is needed
- The purpose of the system is to detect the gas leakage neutralize it and prevent explosion

9. PROBLEM ROOT CAUSE

- Some of the faults in the machines, leakage by the machines, people carelessness in workplace and life security

7. BEHAVIOUR

- Network issue is very common as most of the industries are located at the country side. Here contact both the developers and the service providers.
- To determine the gas characteristicsand solve the issue, they will locate the leak and identify the warning.

3.TRIGGERS

- The trigger varies from the incorrect installation to the use of defective gas cylinders. Employee and organization safety triggers this installation

4.EMOTIONS:Before/After

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

10. YOUR SOLUTION

- Low cost IOT based device that canbe easily accessed and fixed by people.
- Network strength must be boosted inthe device.
- Device can be manufactured in multiple standards based on the environment.

8. CHANNELS OF BEHAVIOUR

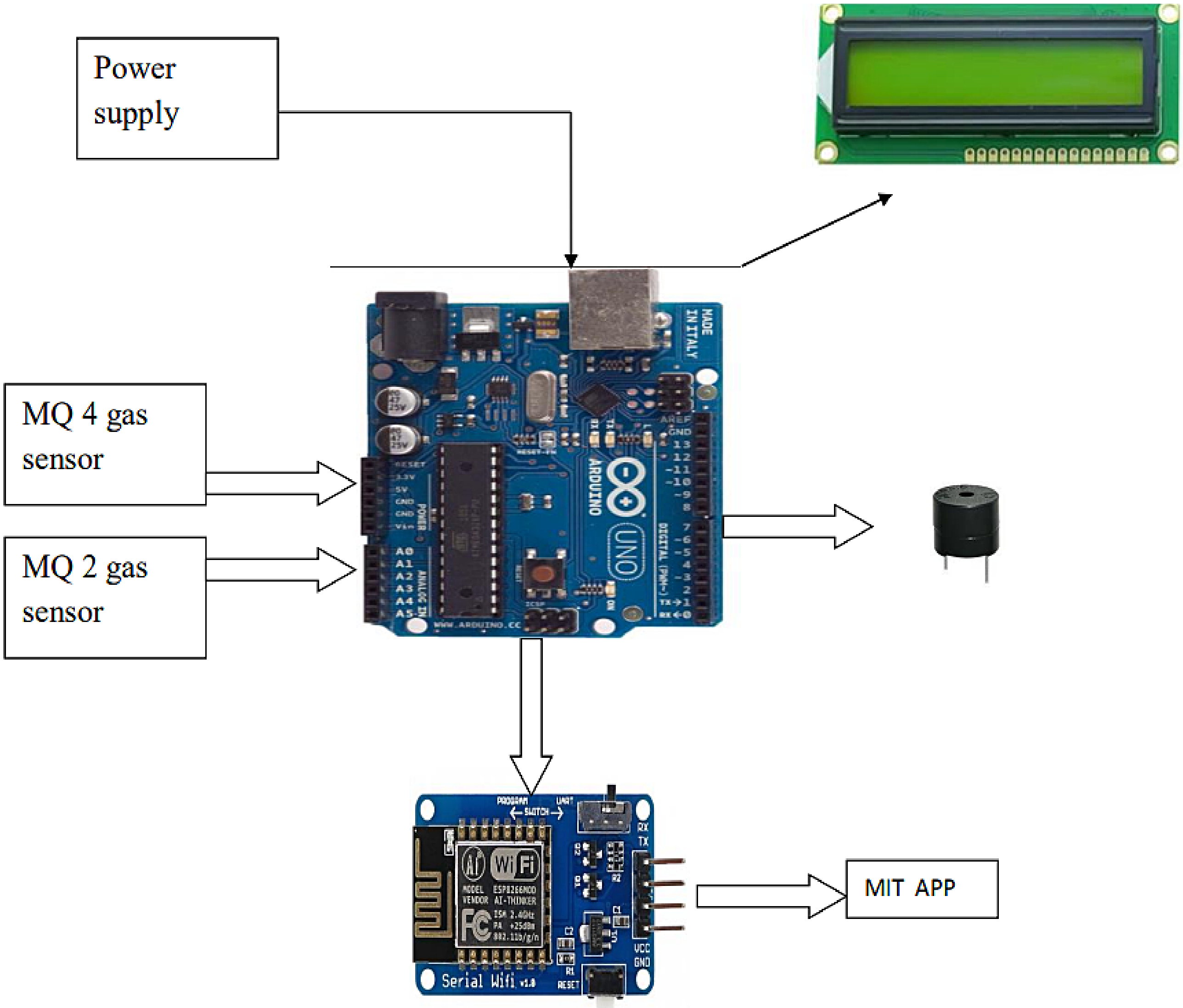
ONLINE

- Sending messages via gsm

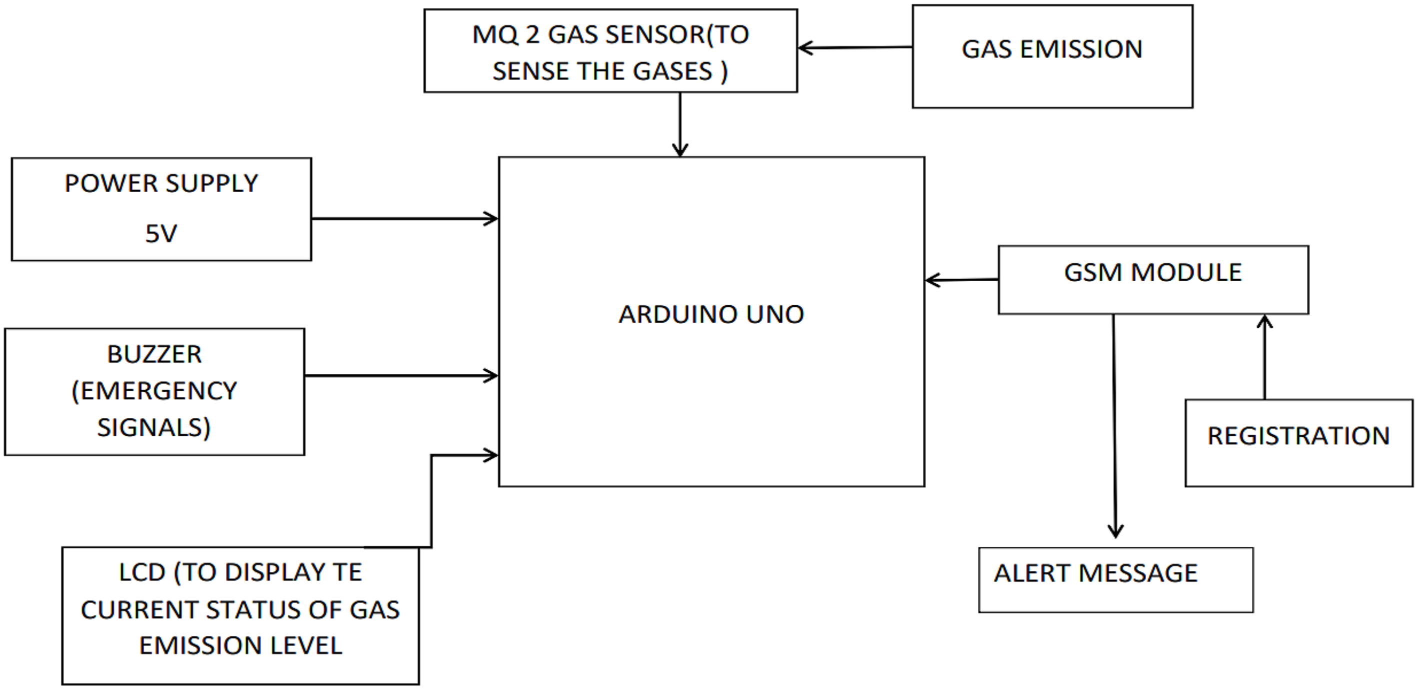
OFFLINE

- Prevent physical damage to sensor.
- Provide proper network and power supply to sensors.
- Complaint letters.
- Alarm generates high noise which provides warning

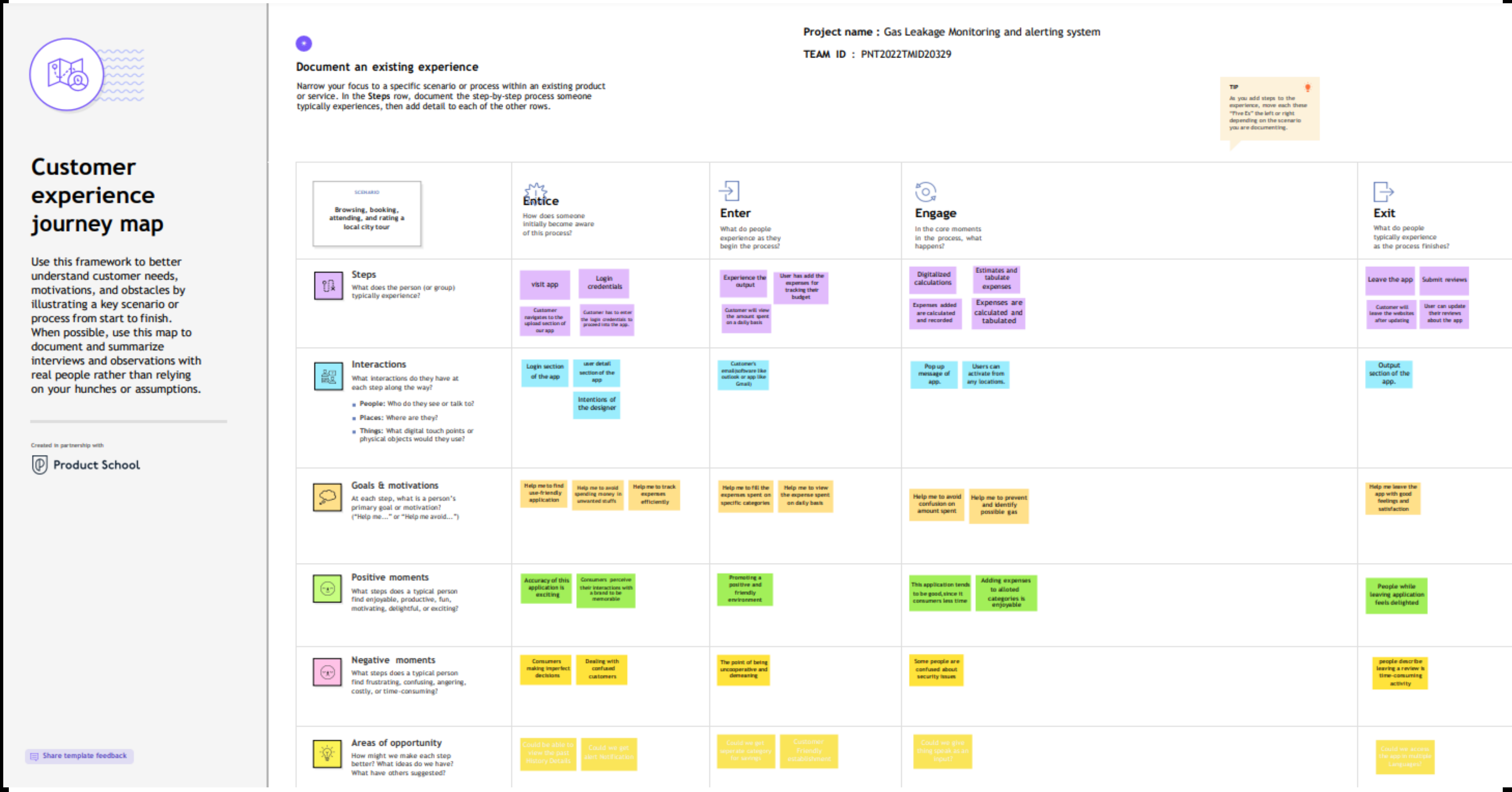
SOLUTION ARCHITECTURE



DATA FLOW DIAGRAM


MQ 2 GAS SENSOR(TO SENSE THE GASES)
GAS EMISSION
POWER SUPPLY 5V
ARDUINO UNO
GSM MODULE
BUZZER (EMERGENCY SIGNALS)
REGISTRATION
LCD (TO DISPLAY TE CURRENT STATUS OF GAS EMISSION LEVEL
ALERT MESSAGE

CUSTOMER JOURNEY



Customer experience journey map

Use this framework to better understand customer needs, motivations, and obstacles by illustrating a key scenario or process from start to finish. When possible, use this map to document and summarize interviews and observations with real people rather than relying on your hunches or assumptions.

Created in partnership with

Product School

Share template feedback

Project name : Gas Leakage Monitoring and alerting system

TEAM ID : PNT2022TMID20329

Document an existing experience

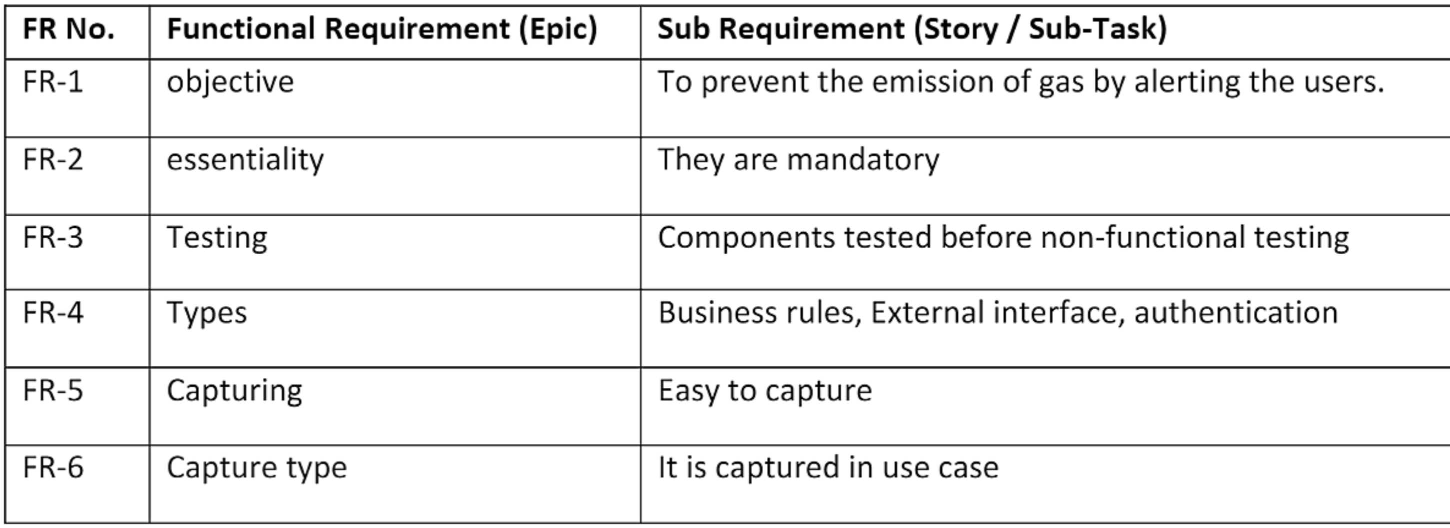
Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.

TIP
As you add steps to the experience, move each these "Five Es" the left or right depending on the scenario you are documenting.

SCENARIO Browsing, booking, attending, and rating a local city tour	**Entice** How does someone initially become aware of this process?	**Enter** What do people experience as they begin the process?	**Engage** In the core moments in the process, what happens?	**Exit** What do people typically experience as the process finishes?
Steps What does the person (or group) typically experience?	visit app Login credentials Customer navigates to the upload section of our app Customer has to enter the login credentials to proceed into the app.	Experience the output User has add the expenses for tracking their budget Customer will view the amount spent on a daily basis	Digitalized calculations Estimates and tabulate expenses Expenses added are calculated and recorded Expenses are calculated and tabulated	Leave the app Submit reviews Customer will leave the websites after updating User can update their reviews about the app
Interactions What interactions do they have at each step along the way? ■ **People:** Who do they see or talk to? ■ **Places:** Where are they? ■ **Things:** What digital touch points or physical objects would they use?	Login section of the app user detail section of the app Intentions of the designer	Customer's email(software like outlook or app like Gmail)	Pop up message of app. Users can activate from any locations.	Output section of the app.
Goals & motivations At each step, what is a person's primary goal or motivation? ("Help me..." or "Help me avoid...")	Help me to find use-friendly application Help me to avoid spending money in unwanted stuffs Help me to track expenses efficiently	Help me to fill the expenses spent on specific categories Help me to view the expense spent on daily basis	Help me to avoid confusion on amount spent Help me to prevent and identify possible gas	Help me leave the app with good feelings and satisfaction
Positive moments What steps does a typical person find enjoyable, productive, fun, motivating, delightful, or exciting?	Accuracy of this application is exciting Consumers perceive their interactions with a brand to be memorable	Promoting a positive and friendly environment	This application tends to be good, since it consumers less time Adding expenses to alloted categories is enjoyable	People while leaving application feels delighted
Negative moments What steps does a typical person find frustrating, confusing, angering, costly, or time-consuming?	Consumers making imperfect decisions Dealing with confused customers	The point of being uncooperative and demeaning	Some people are confused about security issues	people describe leaving a review is time-consuming activity
Areas of opportunity How might we make each step better? What ideas do we have? What have others suggested?	Could be able to view the past History Details Could we get alert Notification	Could we get seperate category for savings Customer Friendly establishment	Could we give thing speak as an input?	Could we access the app in multiple Languages?

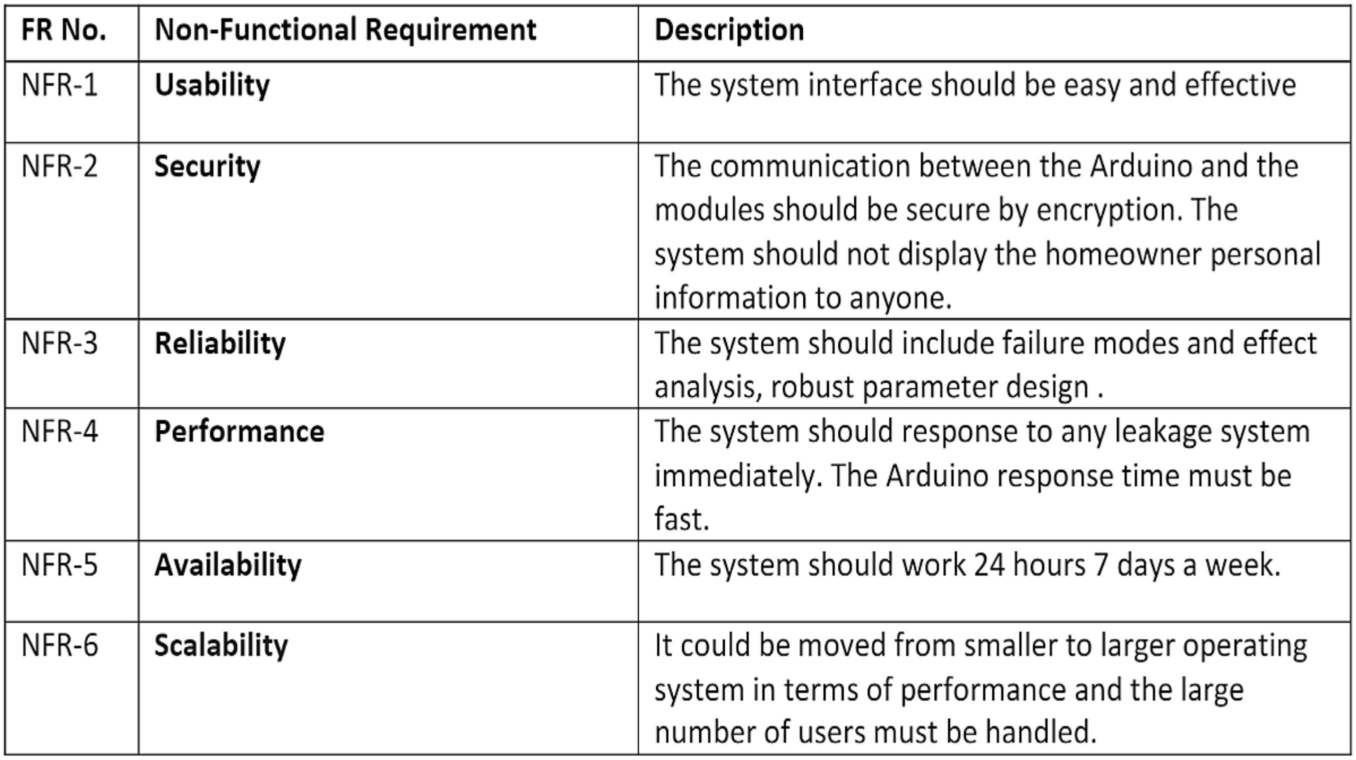
SOLUTION REQIREMENTS

FUNCTIONAL REQUIREMENTS:



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	objective	To prevent the emission of gas by alerting the users.
FR-2	essentiality	They are mandatory
FR-3	Testing	Components tested before non-functional testing
FR-4	Types	Business rules, External interface, authentication
FR-5	Capturing	Easy to capture
FR-6	Capture type	It is captured in use case

NON FUNCTIONAL REQUIREMENTS:



FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The system interface should be easy and effective
NFR-2	**Security**	The communication between the Arduino and the modules should be secure by encryption. The system should not display the homeowner personal information to anyone.
NFR-3	**Reliability**	The system should include failure modes and effect analysis, robust parameter design .
NFR-4	**Performance**	The system should response to any leakage system immediately. The Arduino response time must be fast.
NFR-5	**Availability**	The system should work 24 hours 7 days a week.
NFR-6	**Scalability**	It could be moved from smaller to larger operating system in terms of performance and the large number of users must be handled.

TECHNOLOGY ARCHITECTURE

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

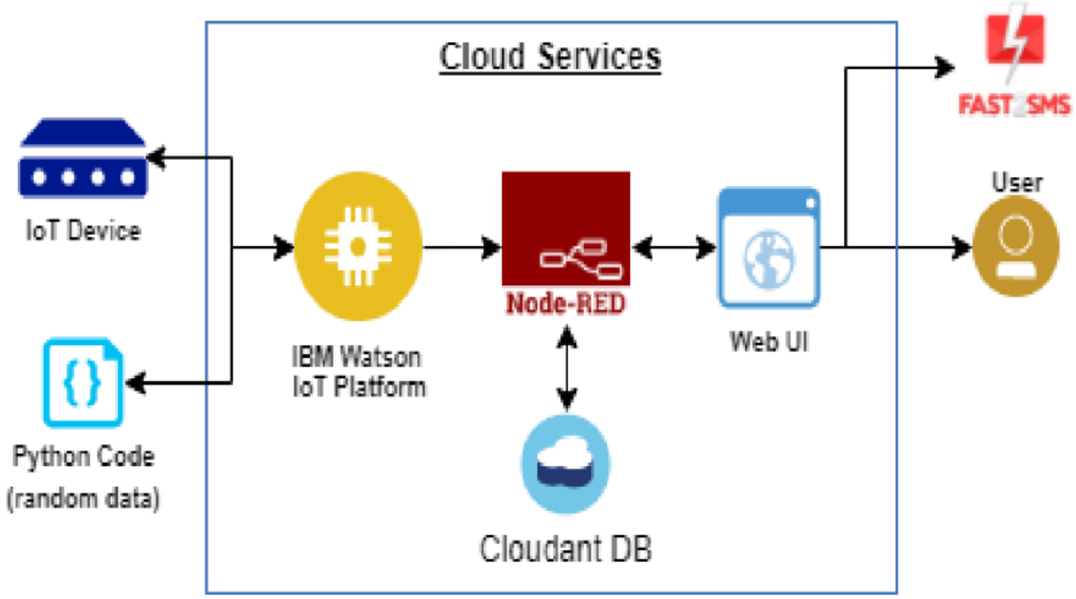
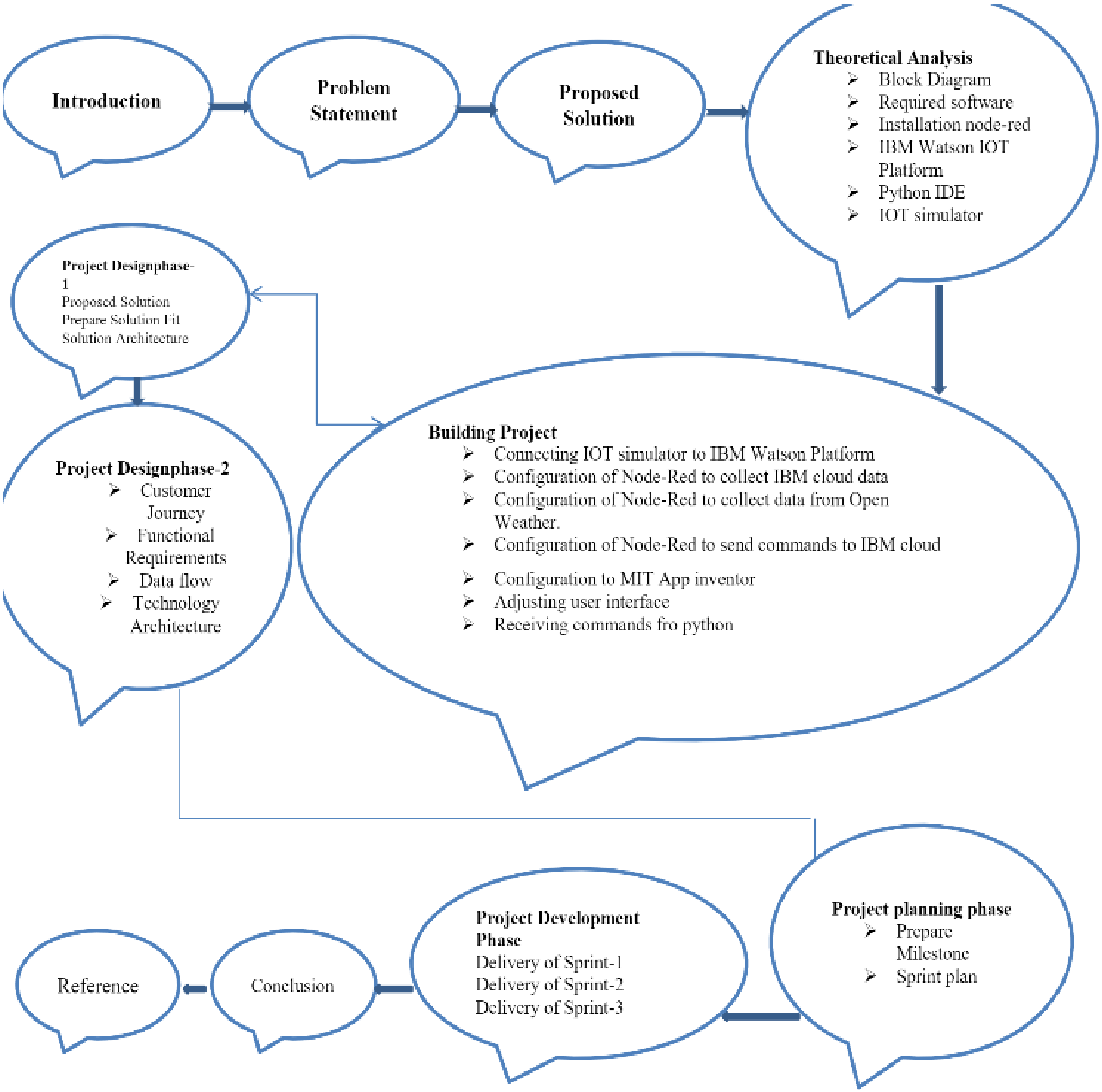


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Node-RED, MIT app	IBM IoT Platform, IBM Node red, IBM Cloud
2.	Application Logic-1	Create Ibm Watson IoT platform and create node- red service	Ibm Watson, ibm cloudant service,ibm node-red
3.	Application Logic-2	Develop python script to publish and subscribe to IBM IoT Platform	python
4.	Application Logic-3	Build a web application using node-red service	IBM Node-red
5.	Database	Data Type, Configurations etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant
7.	File Storage	Developing mobile application to store and receive the sensors information and to react accordingly	Web UI,python
8.	External API-1	Using this IBM fire management API we can track the temperature of the incident place and where the fire had been attacked.	IBM fire management API
9.	External API-2	Using this IBM Sensors it detects the fire, gas leaks , temperature and provides the activation of sprinklers to web UI	IBM Sensors
10.	Machine Learning Model	Using this we can derive the object recognition model	Object Recognition Model
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Cloud Server Configuration	IBM cloudant, IBM IoT Platform

MILESTONES:



Introduction

Problem Statement

Proposed Solution

Theoretical Analysis
- Block Diagram
- Required software
- Installation node-red
- IBM Watson IOT Platform
- Python IDE
- IOT simulator

Building Project
- Connecting IOT simulator to IBM Watson Platform
- Configuration of Node-Red to collect IBM cloud data
- Configuration of Node-Red to collect data from Open Weather.
- Configuration of Node-Red to send commands to IBM cloud
- Configuration to MIT App inventor
- Adjusting user interface
- Receiving commands fro python

Project Designphase-1
Proposed Solution
Prepare Solution Fit
Solution Architecture

Project Designphase-2
- Customer Journey
- Functional Requirements
- Data flow
- Technology Architecture

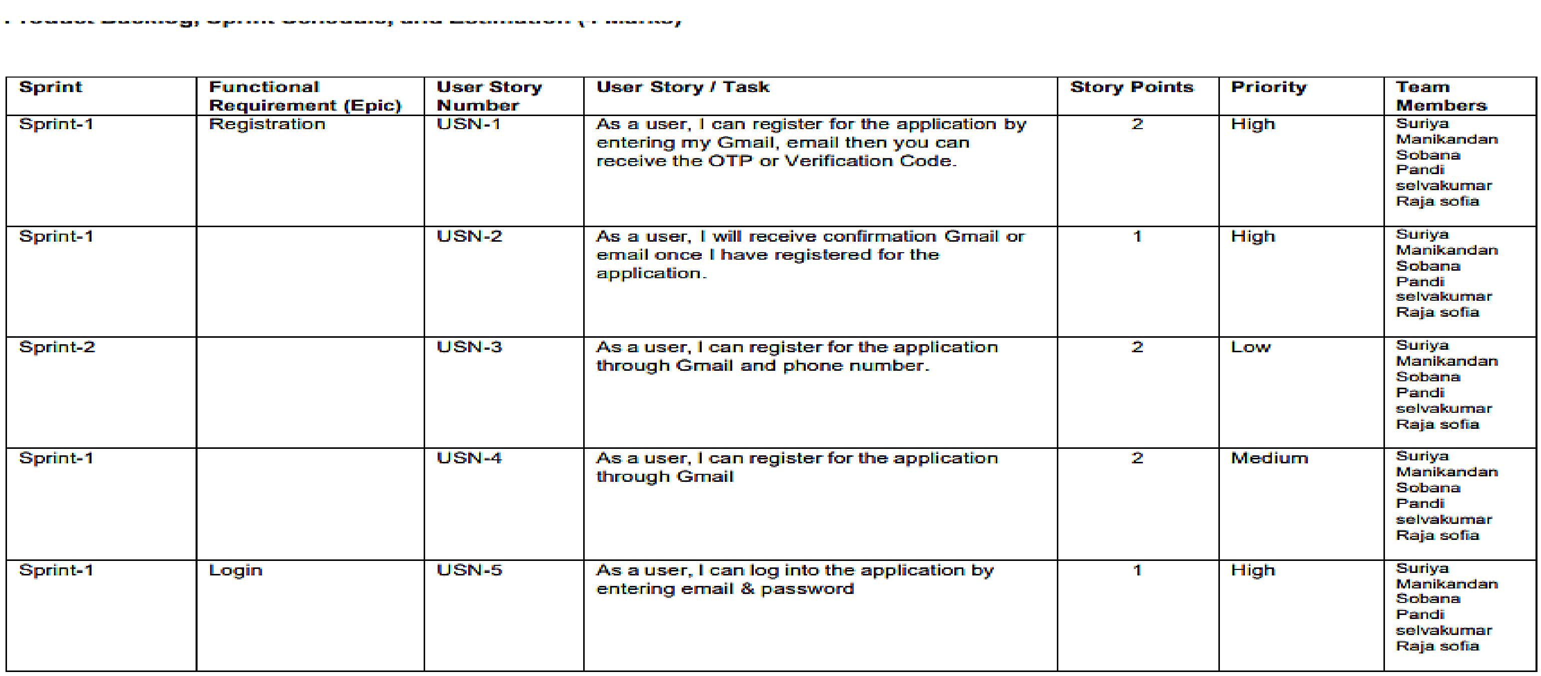
Project planning phase
- Prepare Milestone
- Sprint plan

Project Development Phase
Delivery of Sprint-1
Delivery of Sprint-2
Delivery of Sprint-3

Conclusion

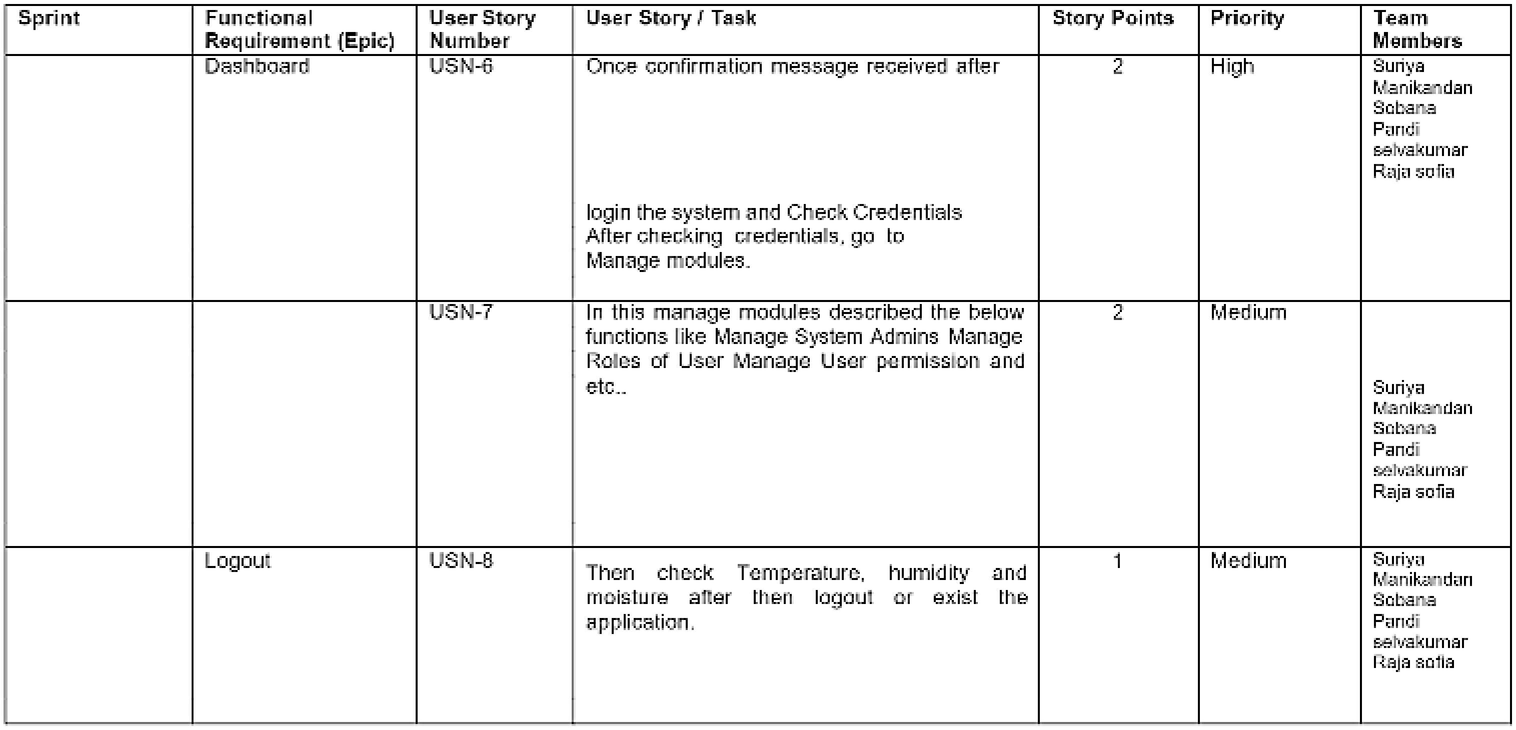
Reference

SPRINT DELIVERY PLAN:



Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my Gmail, email then you can receive the OTP or Verification Code.	2	High	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
Sprint-1		USN-2	As a user, I will receive confirmation Gmail or email once I have registered for the application.	1	High	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
Sprint-2		USN-3	As a user, I can register for the application through Gmail and phone number.	2	Low	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Suriya Manikandan Sobana Pandi selvakumar Raja sofia



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Dashboard	USN-6	Once confirmation message received after login the system and Check Credentials After checking credentials, go to Manage modules.	2	High	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
		USN-7	In this manage modules described the below functions like Manage System Admins Manage Roles of User Manage User permission and etc..	2	Medium	Suriya Manikandan Sobana Pandi selvakumar Raja sofia
	Logout	USN-8	Then check Temperature, humidity and moisture after then logout or exist the application.	1	Medium	Suriya Manikandan Sobana Pandi selvakumar Raja sofia

Project Tracker, Velocity & Burndown Chart: (4 Marks)

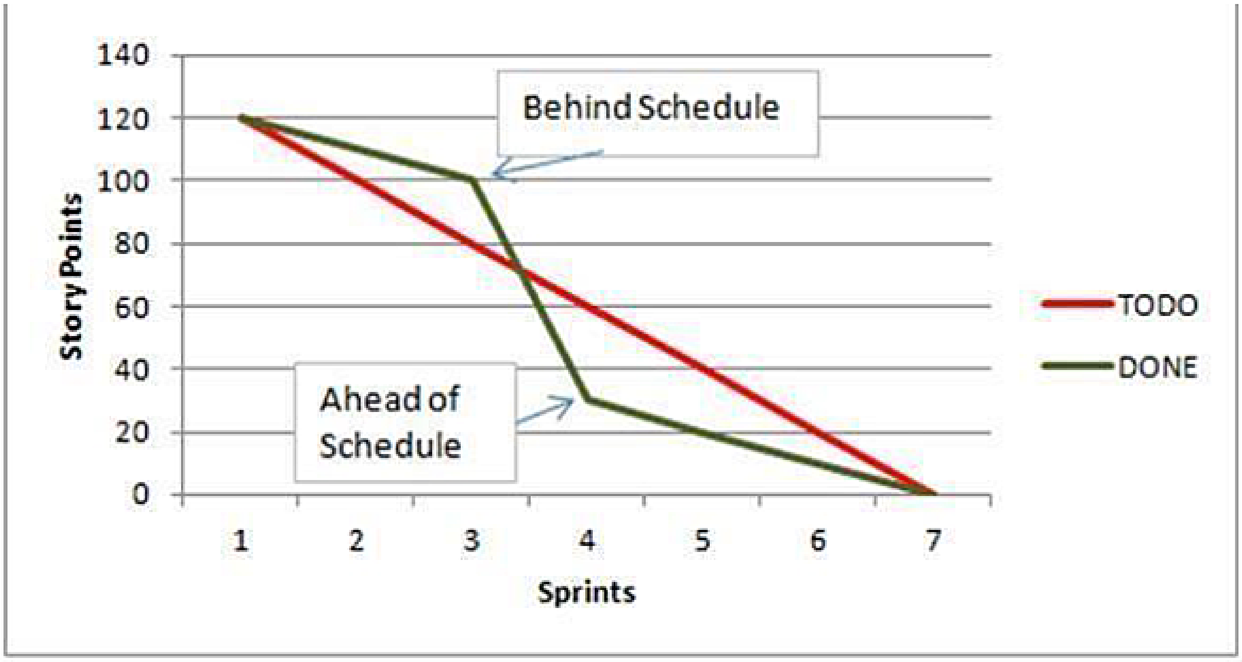
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	35	31 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	45	05 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	50	07 Nov 2022

Velocity:

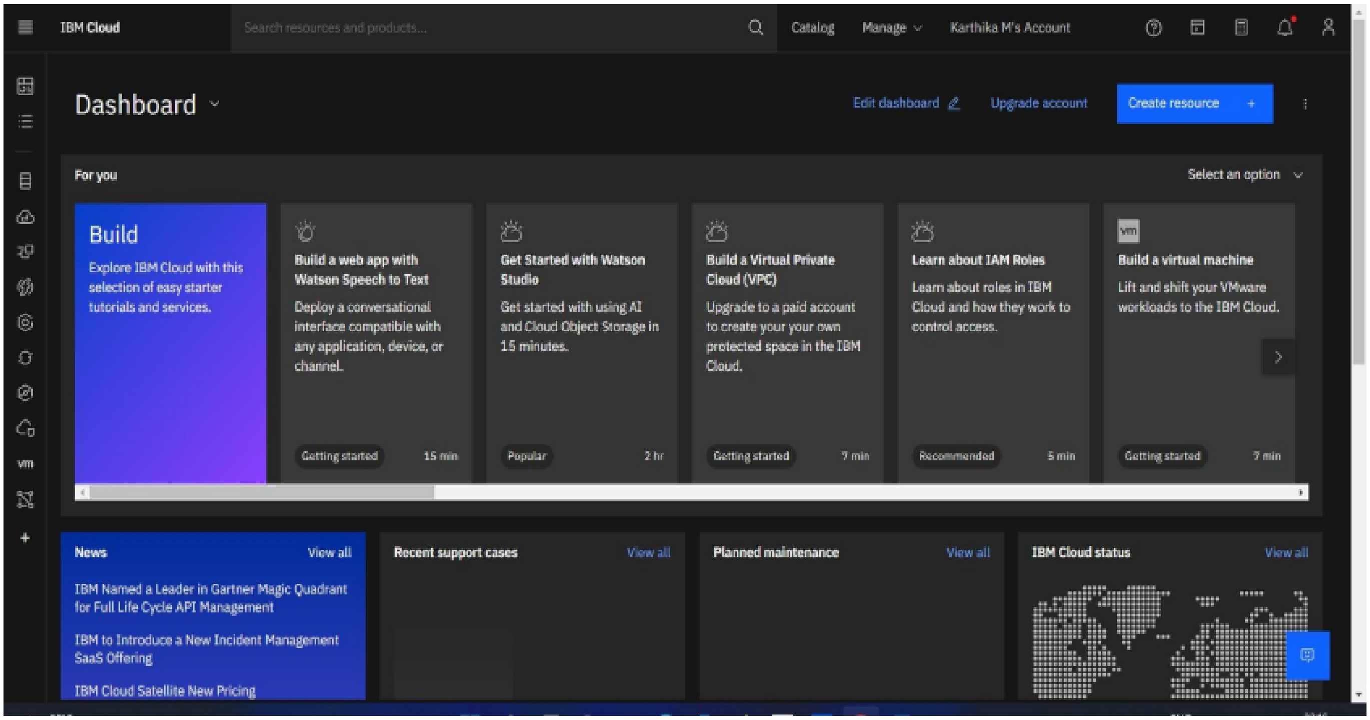
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

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

Burndown Chart:



IBM CLOUD LOGIN



IBM Cloud | Search resources and products... | Catalog | Manage | Karthika M's Account

Dashboard | Edit dashboard | Upgrade account | Create resource +

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 Private Cloud (VPC)
Upgrade to a paid account to create your your own protected space in the IBM Cloud.
Getting started 7 min

Learn about IAM Roles
Learn about roles in IBM Cloud and how they work to control access.
Recommended 5 min

Build a virtual machine
Lift and shift your VMware workloads to the IBM Cloud.
Getting started 7 min

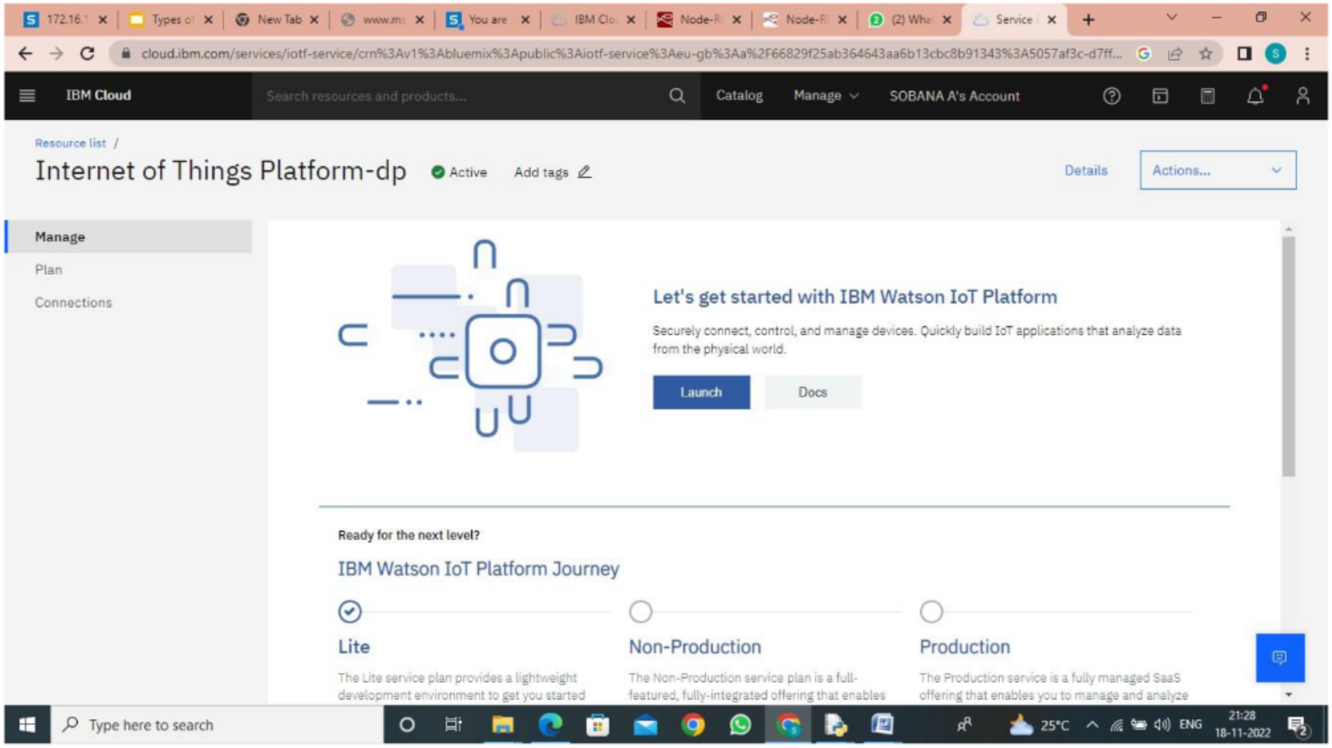
News View all
IBM Named a Leader in Gartner Magic Quadrant for Full Life Cycle API Management
IBM to Introduce a New Incident Management SaaS Offering
IBM Cloud Satellite New Pricing

Recent support cases View all

Planned maintenance View all

IBM Cloud status View all

IBM IOT WATSON PLATFORM



cloud.ibm.com/services/iotf-service/crn%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-gb%3Aa%2F66829f25ab364643aa6b13cbc8b91343%3A5057af3c-d7ff...

IBM Cloud | Search resources and products... | Catalog | Manage | SOBANA A's Account

Resource list /

Internet of Things Platform-dp ● 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

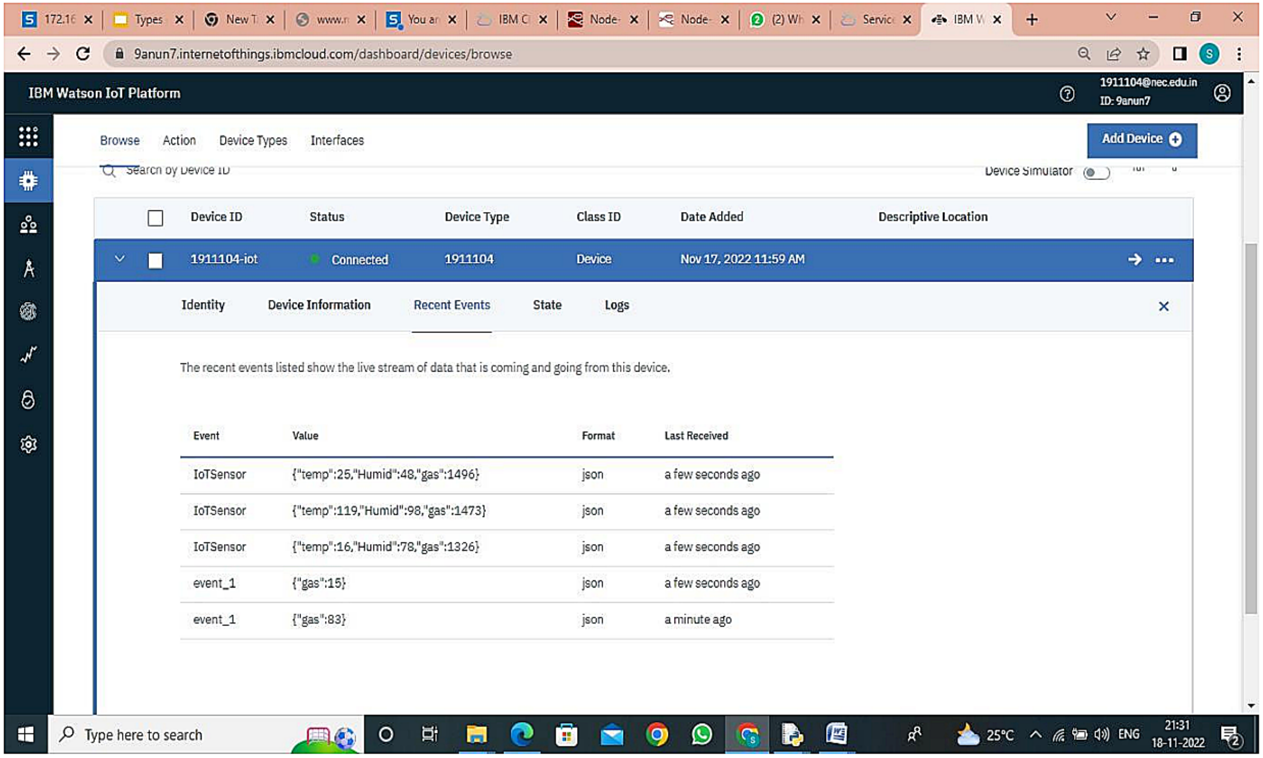
Lite
The Lite service plan provides a lightweight development environment to get you started

Non-Production
The Non-Production service plan is a full-featured, fully-integrated offering that enables

Production
The Production service is a fully managed SaaS offering that enables you to manage and analyze

Type here to search | 25°C ENG 21:28 18-11-2022

Step 9 : Then turn on device simulator and then create a random function for Temperature,Humidity and Gas Level.

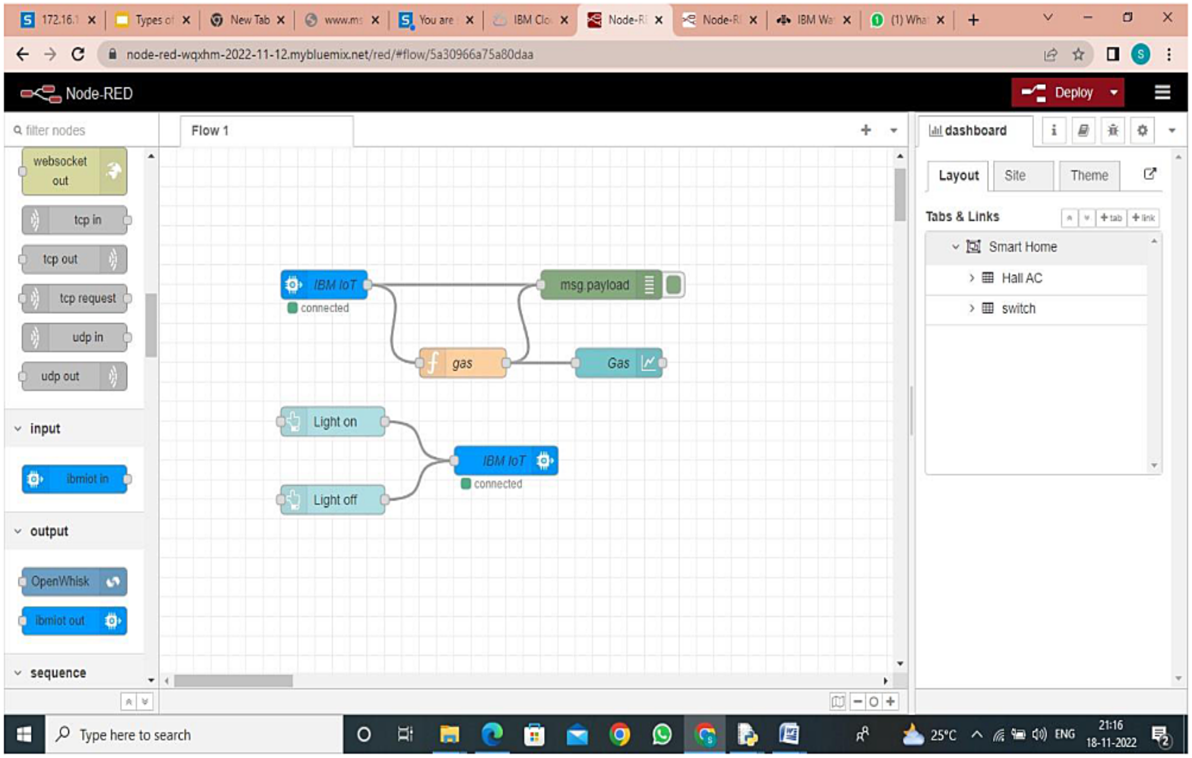


Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
1911104-iot	Connected	1911104	Device	Nov 17, 2022 11:59 AM	

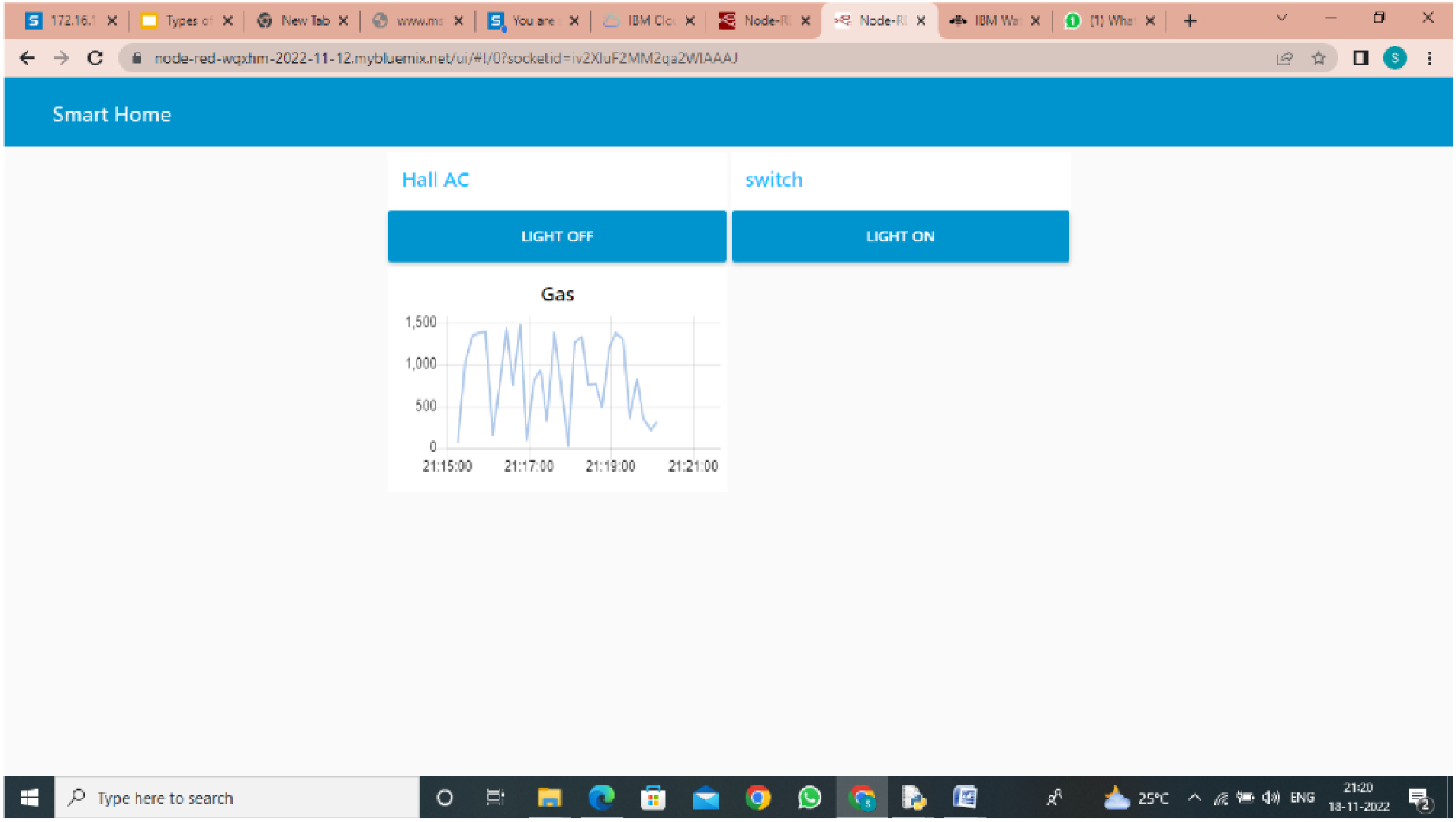
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensor	{"temp":25,"Humid":48,"gas":1496}	json	a few seconds ago
IoTSensor	{"temp":119,"Humid":98,"gas":1473}	json	a few seconds ago
IoTSensor	{"temp":16,"Humid":78,"gas":1326}	json	a few seconds ago
event_1	{"gas":15}	json	a few seconds ago
event_1	{"gas":83}	json	a minute ago

IBM NODE-RED SERVICE

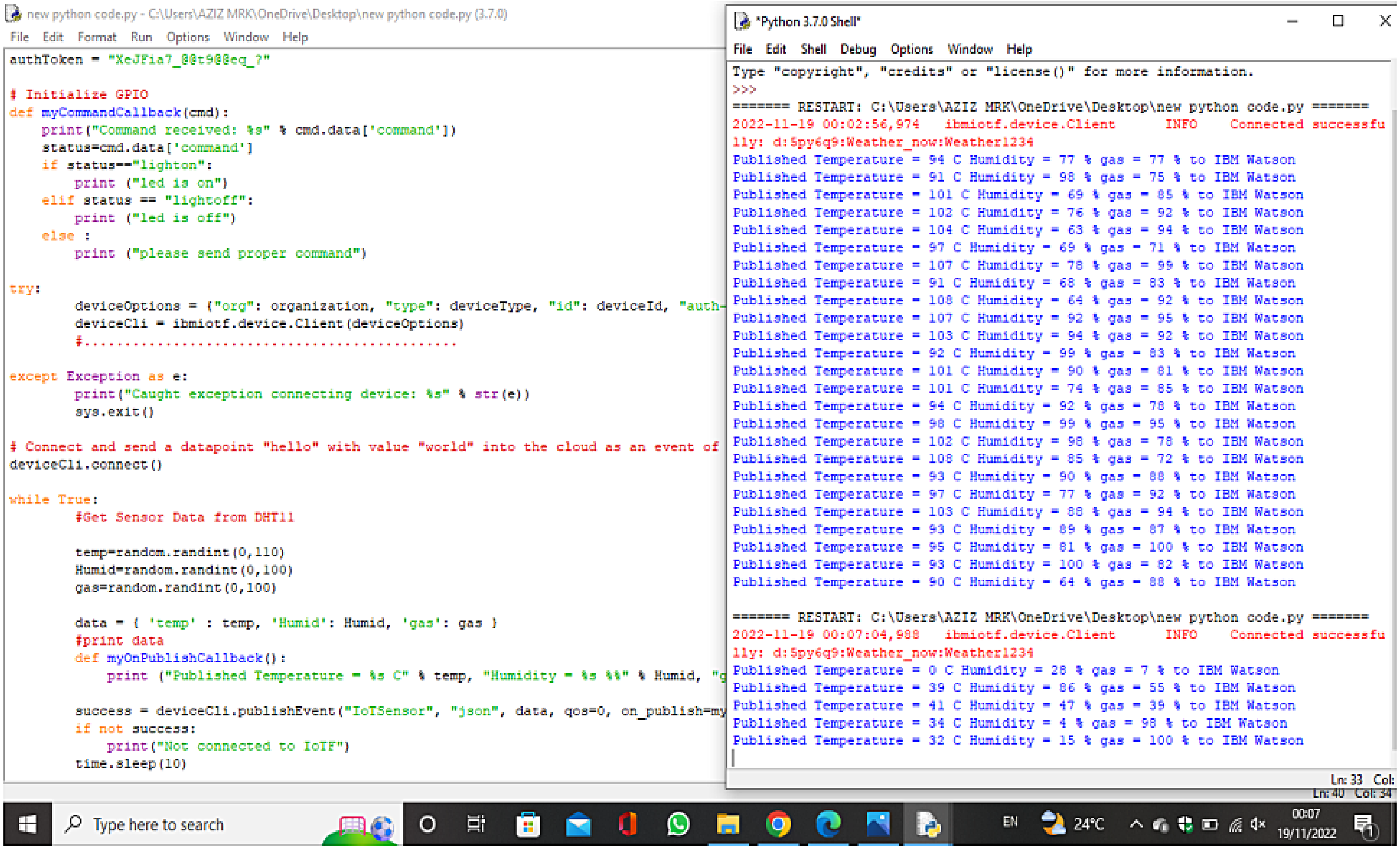


CREATING AN UI



PYTHON SCRIPT: https://github.com/IBM-EPBL/IBM-Project-32578-1660210841/blob/main/Develop%20a%20Python%20Script/Develop%20a%20Python%20code/python%20script.pdf

PYTHON OUTPUT:



new python code.py - C:\Users\AZIZ MRK\OneDrive\Desktop\new python code.py (3.7.0)
File Edit Format Run Options Window Help

```
authToken = "XeJFia7_@@t9@@eq_?"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

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

while True:
        #Get Sensor Data from DHT11

        temp=random.randint(0,110)
        Humid=random.randint(0,100)
        gas=random.randint(0,100)

        data = { 'temp' : temp, 'Humid': Humid, 'gas': gas }
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "g

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=my
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)
```

```
Type "copyright", "credits" or "license()" for more information.
>>>
======= RESTART: C:\Users\AZIZ MRK\OneDrive\Desktop\new python code.py =======
2022-11-19 00:02:56,974   ibmiotf.device.Client      INFO    Connected successfu
lly: d:5py6q9:Weather_now:Weather1234
Published Temperature = 94 C Humidity = 77 % gas = 77 % to IBM Watson
Published Temperature = 91 C Humidity = 98 % gas = 75 % to IBM Watson
Published Temperature = 101 C Humidity = 69 % gas = 85 % to IBM Watson
Published Temperature = 102 C Humidity = 76 % gas = 92 % to IBM Watson
Published Temperature = 104 C Humidity = 63 % gas = 94 % to IBM Watson
Published Temperature = 97 C Humidity = 69 % gas = 71 % to IBM Watson
Published Temperature = 107 C Humidity = 78 % gas = 99 % to IBM Watson
Published Temperature = 91 C Humidity = 68 % gas = 83 % to IBM Watson
Published Temperature = 108 C Humidity = 64 % gas = 92 % to IBM Watson
Published Temperature = 107 C Humidity = 92 % gas = 95 % to IBM Watson
Published Temperature = 103 C Humidity = 94 % gas = 92 % to IBM Watson
Published Temperature = 92 C Humidity = 99 % gas = 83 % to IBM Watson
Published Temperature = 101 C Humidity = 90 % gas = 81 % to IBM Watson
Published Temperature = 101 C Humidity = 74 % gas = 85 % to IBM Watson
Published Temperature = 94 C Humidity = 92 % gas = 78 % to IBM Watson
Published Temperature = 98 C Humidity = 99 % gas = 95 % to IBM Watson
Published Temperature = 102 C Humidity = 98 % gas = 78 % to IBM Watson
Published Temperature = 108 C Humidity = 85 % gas = 72 % to IBM Watson
Published Temperature = 93 C Humidity = 90 % gas = 88 % to IBM Watson
Published Temperature = 97 C Humidity = 77 % gas = 92 % to IBM Watson
Published Temperature = 103 C Humidity = 88 % gas = 94 % to IBM Watson
Published Temperature = 93 C Humidity = 89 % gas = 87 % to IBM Watson
Published Temperature = 95 C Humidity = 81 % gas = 100 % to IBM Watson
Published Temperature = 93 C Humidity = 100 % gas = 82 % to IBM Watson
Published Temperature = 90 C Humidity = 64 % gas = 88 % to IBM Watson

======= RESTART: C:\Users\AZIZ MRK\OneDrive\Desktop\new python code.py =======
2022-11-19 00:07:04,988   ibmiotf.device.Client      INFO    Connected successfu
lly: d:5py6q9:Weather_now:Weather1234
Published Temperature = 0 C Humidity = 28 % gas = 7 % to IBM Watson
Published Temperature = 39 C Humidity = 86 % gas = 55 % to IBM Watson
Published Temperature = 41 C Humidity = 47 % gas = 39 % to IBM Watson
Published Temperature = 34 C Humidity = 4 % gas = 98 % to IBM Watson
Published Temperature = 32 C Humidity = 15 % gas = 100 % to IBM Watson
```

Advantages:

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

Disadvantage:

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

CONCLUSION

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

FUTURE SCOPE:

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

DEMO LINK: https://drive.google.com/file/d/1Hd0CT_Hr3yK-uI0LEzd9y_ANezIPqm8B/view