FINAL PROJECT REPORT

Industry-specific intelligent fire management system

Team ID	PNT2022TMID12796
Team Leader	MUKESH KUMAR A
Team Member 1	SUBAASH G
Team Member 2	VIGNESHWAR
Team Member 3	JAYANTHAN C

TABLE OF CONTENTS

1.INTRODUCTION

1.1 Project Overview

The primary purpose of fire alarm system is to provide an early warning of fire so that people can be evacuated & immediate action can be taken to stop or eliminate of the fire effect as soon as possible. Alarm can be triggered by using detectors or by manual call point (Remotely). To alert/evacuate the occupants siren are used. With the Intelligent Building of the rapid development of technology applications, commercial fire alarm market demand growth, the key is to use the bus system intelligent distributed computer system fire alarm system, although installation in the system much easier than in the past , but still cannot meet the modern needs, the installation costs of equipment costs about 33% ~ 70. The suggested technique in Fire alarm system used the addressable detectors units besides using the wireless connection between the detector in zones as a slave units and the main control unit as the master unit. The system shall include a control panel, alarm initiating devices, notification appliances, and the accessory equipment necessary for a complete functioning fire alarm system. In the wireless fire alarm, individual units are powered by primary & secondary batteries for the communication.

1.2 Purpose

Fire alarm systems are only effective if they can generate reliable and fast fire alerts with exact location of fire. There is a direct correlation between the amount of damage caused by fire and interventions time in various fire alarm systems. As the time of intervention decreases, the damage also decreases. Hence the most important factor in a fire alarm system is the reaction or response time of fire alarm system, that is, the time between fire detection and extinguishing.

The earliest recorded examples of fire protection can be traced back to the Roman Empire and the catastrophic fires that started in Rome. As a result, Emperor Neron has adopted regulations that required fireproof material for walls and buildings restoration to be used. The second recorded case of adopting fire protection regulations occurred in the year 1666, after the Great fire of London, which destroyed more than 80% of the city. The fire of London spurred interest in the development of the first equipment for fire suppression in the form of hand pumps and fire hydrant installation for water supply.

2. LITERATURE SURVEY

2.1 Existing Problem

There are usually a number of construction and renovation works happening across a hospital complex at any one time, which often creates additional challenges for the fire system. Any works being carried out must be done without any downtime and minimal disruption as hospitals are operational 24 hours a day 365 days a year. Contractors can create copious amounts of dust that can set off smoke detectors, causing false alarms. This combined with maintenance work being carried out at night can cause a headache for healthcare estates staff as personnel must be called out of hours to correct it. One option to prevent these false alarms while ensuring fire safety is to temporarily replace smoke detectors for heat detectors in the area being worked in. Heat detectors or CO/ heat detectors are not prone to contamination and therefore are less likely to falsely activate due to dust. It is important to consider that smoke detectors' coverage area is larger than a heat detectors and therefore a point for point replacement will result in a loss of coverage. Another option is to use a multisensor with multiple modes of sensitivity such as Apollo Soteria, this detector is much more resilient to false alarms due to its advanced chamber design and can, if required be switched to a heat only mode.

2.2 References

(1) Ahmed Imteaj et.al. Studied the problems faced by factory workers in times when fire breaks out. They proposed a system using Raspberry Pi 3 which is capable of detecting fire and providing information about area of fire. The Raspberry Pi controls multiple Arduino boards which are connected with several motors and cameras to capture the fire incident.In this, they discussed about the modern technology that can be used to reduce extremely unfortunate accidents caused by fire. We designed thewhole system and calculated its effectiveness.

(2) Ondrej Krejcar proposed a model for location enhancement and personnel tracking using Wi-Fi networks. In this, he has represented the control system concept that is used in handling information of location and control unit operations. The location of the user present in the building, is obtained through Wi-Fi access points. We have studied this to understand the usability of the Wi-Fi networks in live tracking and then have utilized this functionality to track fire and give information about location of fire to various devices intimating people about the mishap.

(3) Authors in have studied the safety features in home and industrial areas. They have designed new model using WSN. Not only have they incorporated temperature and humidity sensors but also included fire and smoke sensors while developing the model. They present a preceding study of WSN is able to detect fire alarm. It is for setting up a wireless sensor network with three sensors. An application was developed for getting home information .

(4) Azka Ihsan Nurrahman, Kusprasapta Mutijarsa have proposed a prototype for a centralized management system for homes or offices which helps better in managing the safety features. In this, home management system is required. This system controls the room lights by turning on and off automatically, it keeps the record of use of electronic device status, turning on and off the ac

regulator automatically, it displays the room temperature in home. If fire is detected in the house, it turn on sprinkler at home, it supervises at home via surveillance cameras, take photos and store them including recordings of surveillance at home, it detects the movements of people at home, and provide notification when someone enters the house.

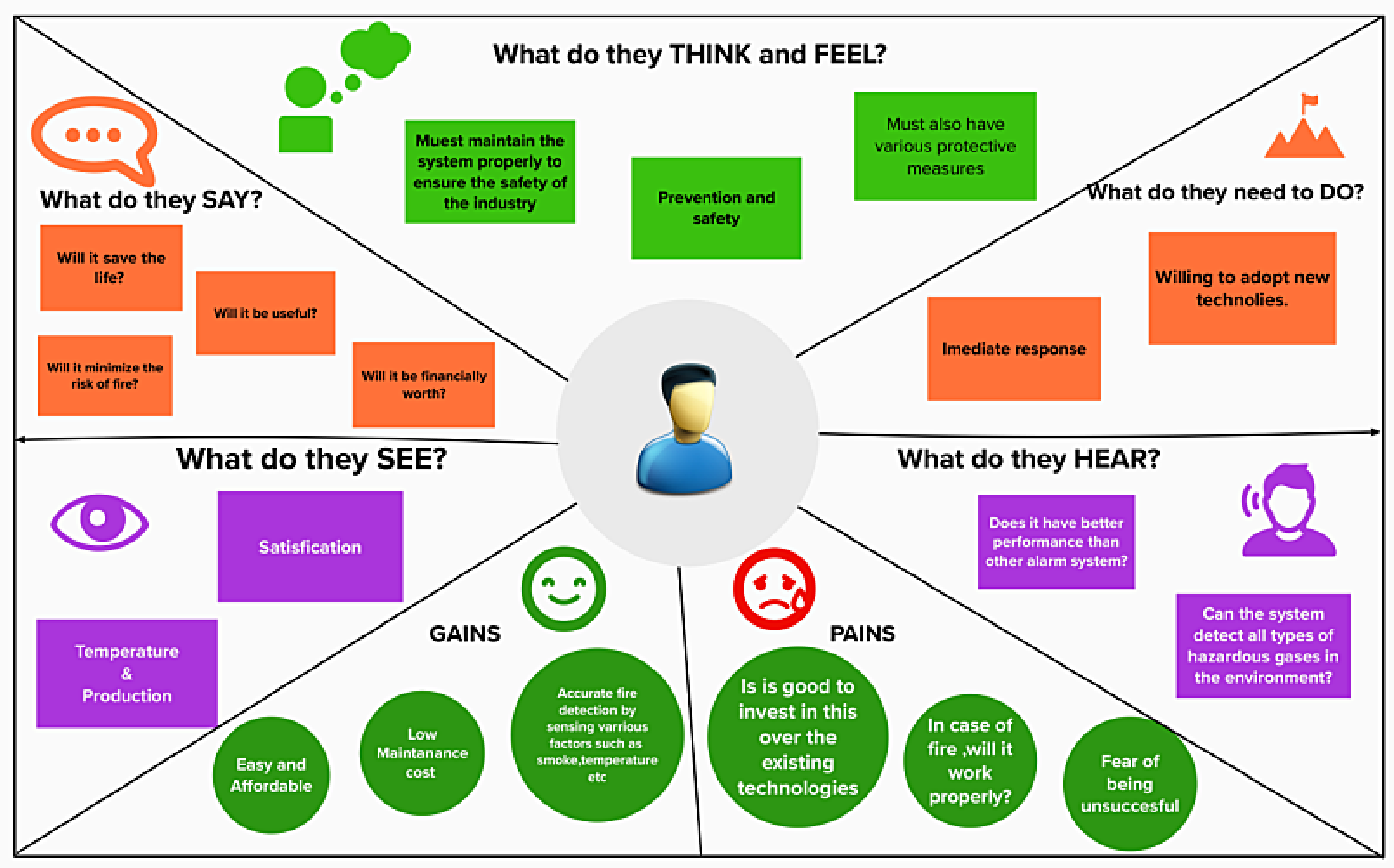
(5) Building Fire Emergency Detection and Response Using Wireless Sensor Networks Yuanyuan Zeng, Seán Óg Murphy, Lanny Sitanayah, Tatiana Maria Tabirca, Thuy Truong, Ken Brown, Cormac J. Sreenan Department of Computer Science, University College Cork : Wireless sensor networks (WSNs) provide a low cost solution with respect maintenance and installation and in particular, building refurbishment and retrofitting are easily accomplished via wireless technologies. Fire emergency detection and response for building environments is a novel application area for the deployment of wireless sensor networks. In such a critical environment, timely data acquisition, detection and response are needed for successful building automation. This paper presents an overview of our recent research activity in this area. Firstly we explain research on communication protocols that are suitable for this problem. Then we describe work on the use of WSNs to improve fire evacuation and navigation.

2.3 Problem Statement Definition

- The smart fire management system includes a Gas sensor, Flame sensor and temperature sensors to detect any changes in the environment.
- Based on the temperature readings and if any Gases are present the exhaust fans are powered ON.
- If any flame is detected the sprinklers will be switched on automatically.
- Emergency alerts are notified to the authorities and Fire station.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



What do they THINK and FEEL?

Muest maintain the system properly to ensure the safety of the industry

Prevention and safety

Must also have various protective measures

What do they SAY?

Will it save the life?

Will it be useful?

Will it minimize the risk of fire?

Will it be financially worth?

What do they need to DO?

Willing to adopt new technolies.

Imediate response

What do they SEE?

Satisfication

Temperature & Production

What do they HEAR?

Does it have better performance than other alarm system?

Can the system detect all types of hazardous gases in the environment?

GAINS

Easy and Affordable

Low Maintanance cost

Accurate fire detection by sensing varrious factors such as smoke,temperature etc

PAINS

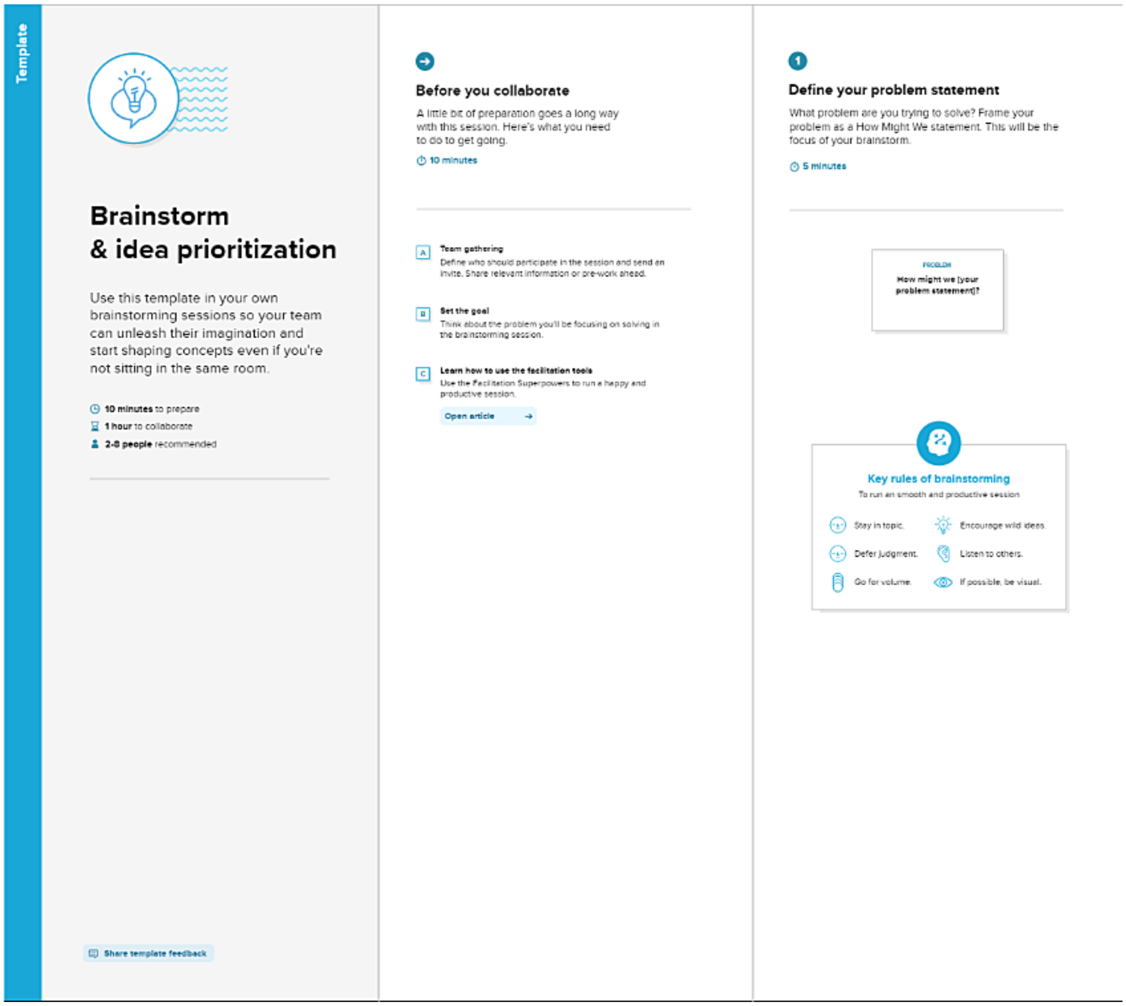
Is is good to invest in this over the existing technologies

In case of fire ,will it work properly?

Fear of being unsuccesful

3.2 Ideation & Brainstorming

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



Brainstorm
& idea prioritization

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

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

Before you collaborate

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

10 minutes

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

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

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

1 **Define your problem statement**

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

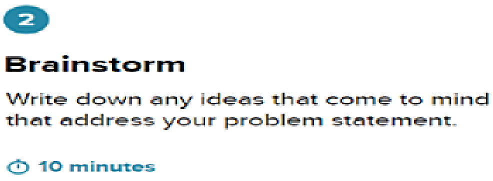
5 minutes

PROBLEM
How might we [your problem statement]?

Key rules of brainstorming
To run an smooth and productive session

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

Step-2: Brainstorm, Idea Listing and Grouping

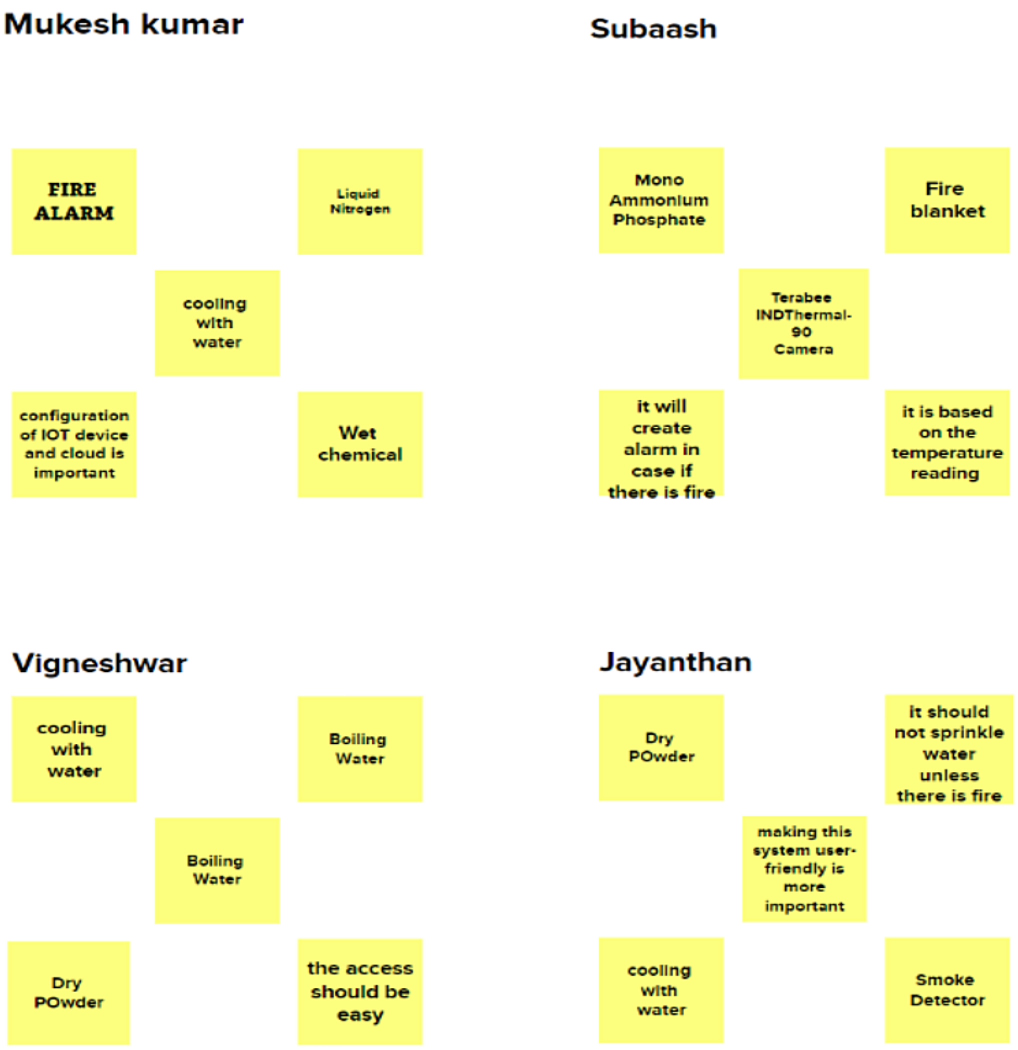


2

Brainstorm

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

10 minutes



Mukesh kumar

FIRE ALARM

Liquid Nitrogen

cooling with water

configuration of IOT device and cloud is important

Wet chemical

Subaash

Mono Ammonium Phosphate

Fire blanket

Terabee INDThermal-90 Camera

it will create alarm in case if there is fire

it is based on the temperature reading

Vigneshwar

cooling with water

Boiling Water

Boiling Water

Dry POwder

the access should be easy

Jayanthan

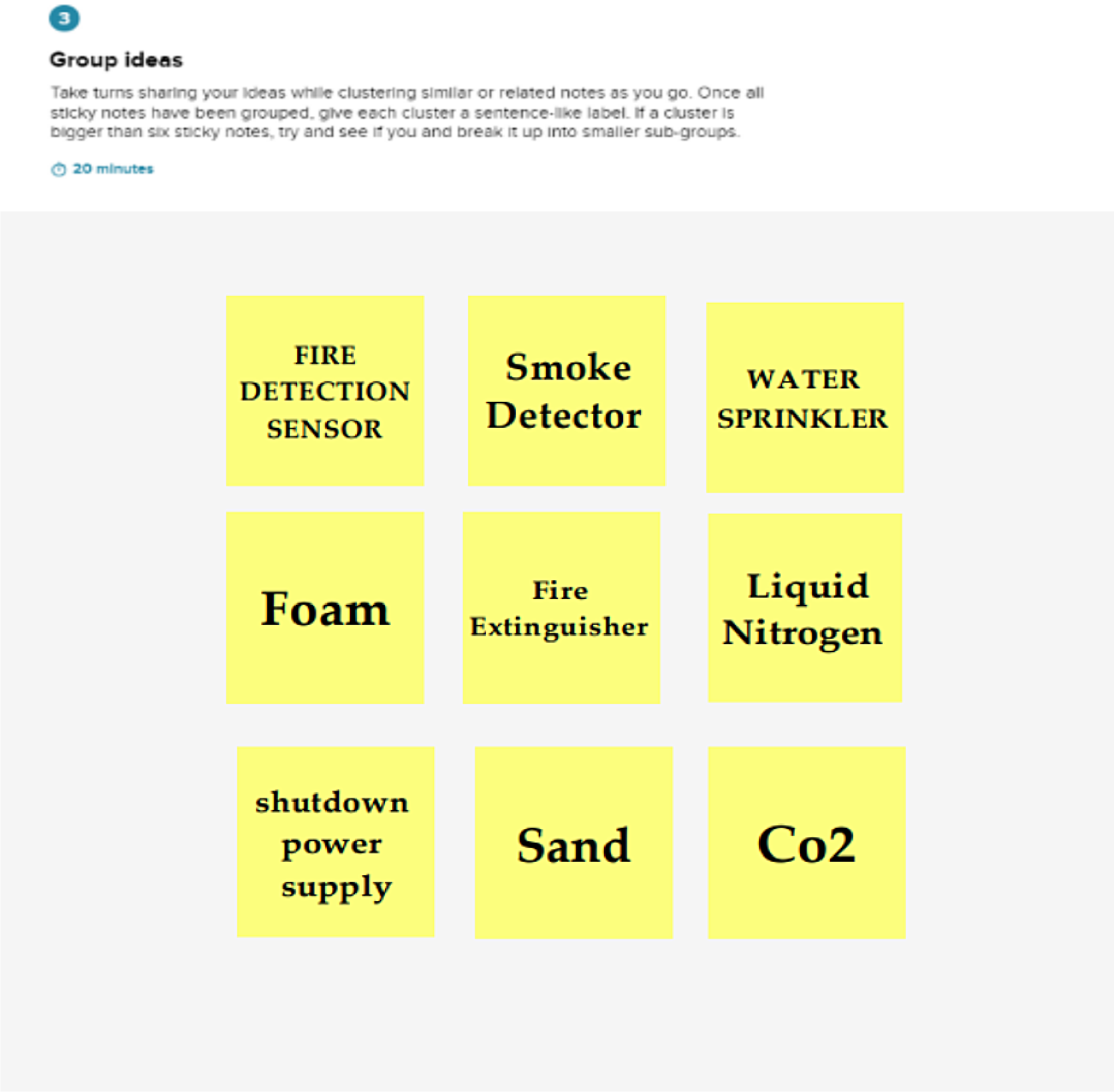
Dry POwder

it should not sprinkle water unless there is fire

making this system user-friendly is more important

cooling with water

Smoke Detector



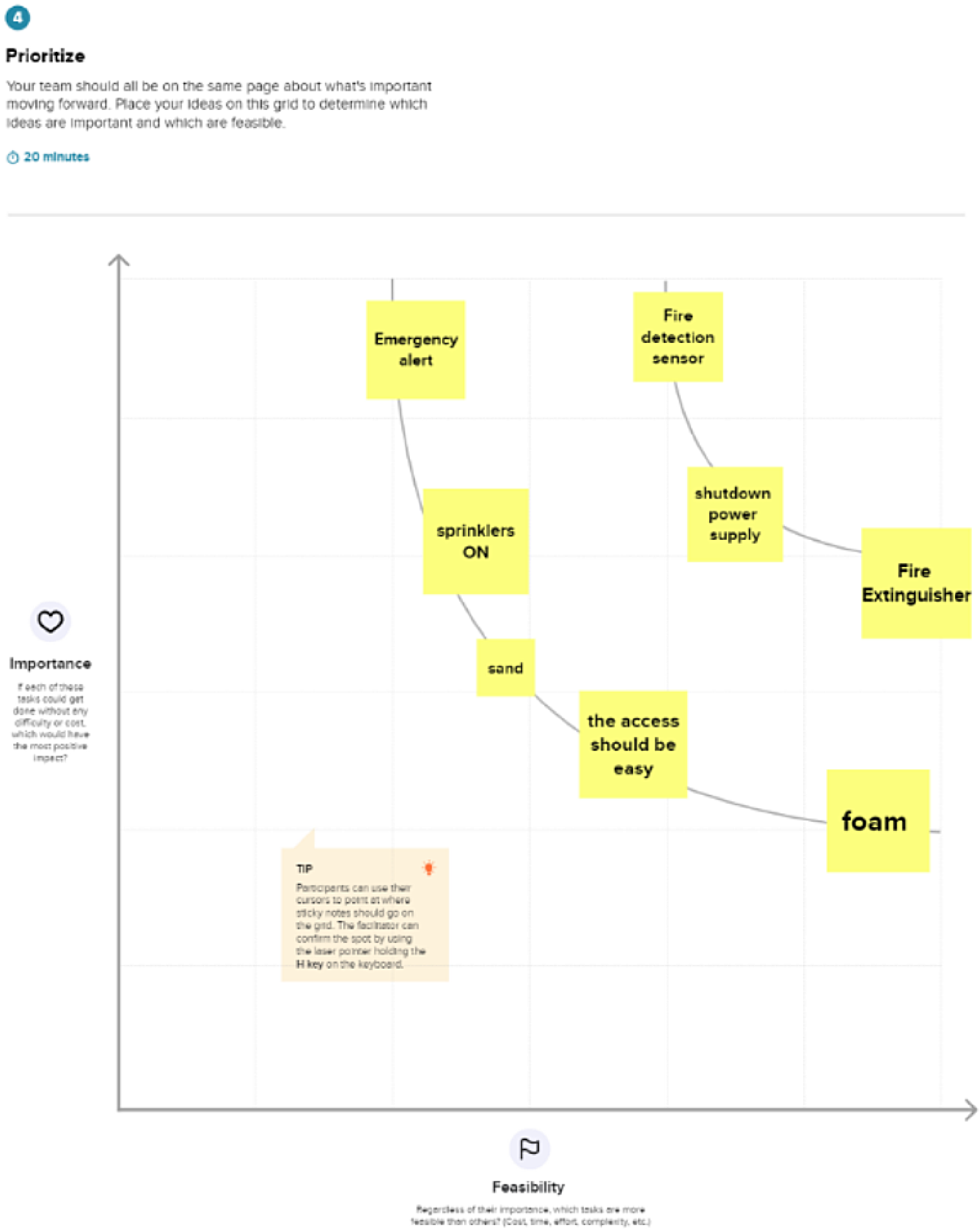
3

Group ideas

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

20 minutes

FIRE DETECTION SENSOR	Smoke Detector	WATER SPRINKLER
Foam	Fire Extinguisher	Liquid Nitrogen
shutdown power supply	Sand	Co2



4

Prioritize

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

20 minutes

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.

Feasibility

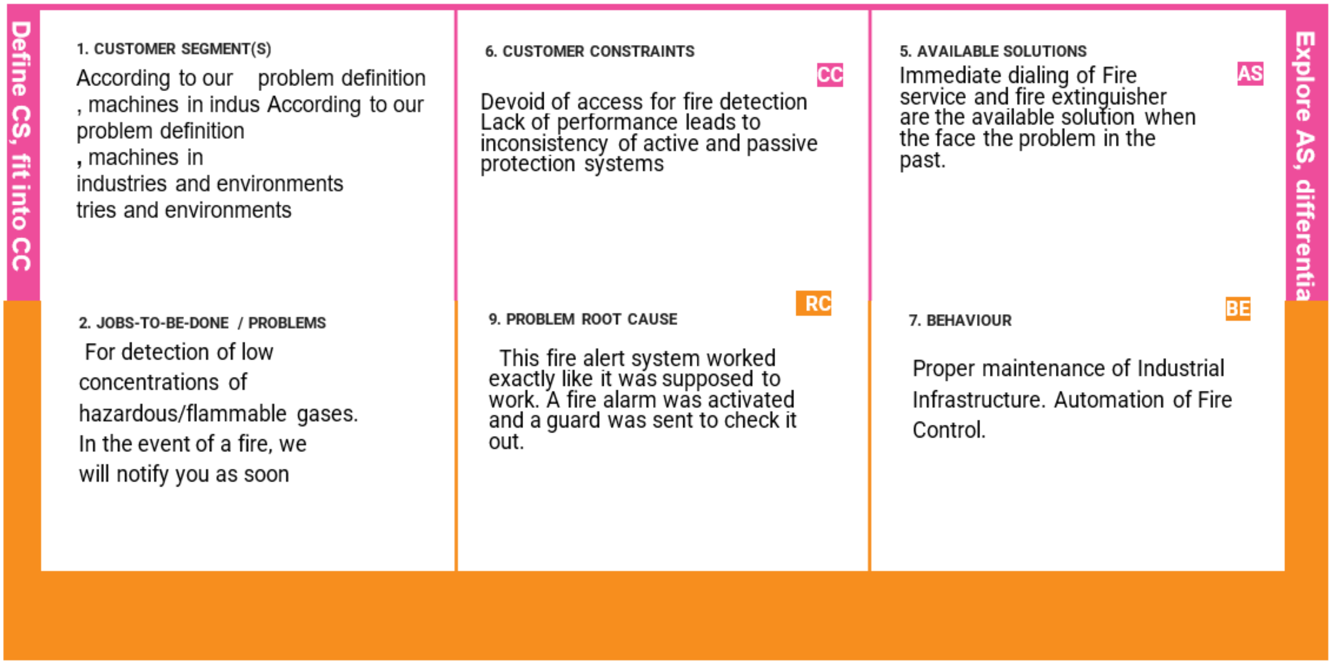
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

3.3 Proposed Solution

Proposed Solution Template:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The main issues are unavailability of access for fire officers and poor roads. The inconsistencies are also related to the poor performance of the active and passive protection system, which in most cases fails to function in accordance with fire safety standards.
2.	Idea / Solution description	Don't overload electrical equipment or circuits. Don't leave temporary equipment plugged in when it's not in use. Avoid using extension cords, and never consider them permanent solutions. Use antistatic equipment where required by NFPA or OSHA.
3.	Novelty / Uniqueness	An integrated system of temperature monitering, gas monitering, fire detection and automatic sprinklers to obtain the accurate information about locations and response through SMS notification and call.
4.	Social Impact / Customer Satisfaction	Forecasting the mishap will notify the industry workers to migrate to better and safer buildings. Provides components with affordable prices and is highly feasible.
5.	Business Model (Revenue Model)	It is an industry-efficient product in all aspects. Provides a clear idea about the entire working mechanism of the system.
6.	Scalability of the Solution	This is complete system makes it easily expandable and business efficient for the fire detection, with the significant cost.

3.4 Problem Solution Fit



Define CS, fit into CC

1. CUSTOMER SEGMENT(S)

According to our problem definition , machines in indus According to our problem definition , machines in industries and environments tries and environments

6. CUSTOMER CONSTRAINTS

CC

Devoid of access for fire detection Lack of performance leads to inconsistency of active and passive protection systems

5. AVAILABLE SOLUTIONS

AS

Immediate dialing of Fire service and fire extinguisher are the available solution when the face the problem in the past.

Explore AS, differentia

2. JOBS-TO-BE-DONE / PROBLEMS

For detection of low concentrations of hazardous/flammable gases. In the event of a fire, we will notify you as soon

9. PROBLEM ROOT CAUSE

RC

This fire alert system worked exactly like it was supposed to work. A fire alarm was activated and a guard was sent to check it out.

7. BEHAVIOUR

BE

Proper maintenance of Industrial Infrastructure. Automation of Fire Control.

3. TRIGGERS TR We may ask you to share your experience with our products. We can claim that they need our products.	10. YOUR SOLUTION SL An IOT based industry – specific, fire management system that focuses on early detection and management of false alarms and errors. Implementation of wireless technologies in transferring messages.	8. CHANNELS of BEHAVIOUR Call Emergency Number Remove the Fire burn things
4. EMOTIONS: BEFORE / AFTER EM Before: Customer is not find a proper rid for the fire spread problem. After: Now with the help of our product the customer can easily enhance problem.		

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 via form Registration via mobile phone number
FR-2	User Confirmation	Confirm by message Confirm by phone
FR-3	User Login	Log in through the website or app with your respective username and password.
FR-4	User Upload	Customers should be able to upload information
FR-5	Fire Detection Monitoring	Sensors monitor the industry 24/7 and the provides information to end users.
FR-6	Location Notification	The location of the fire is sent to the fire brigade in an alarm or message

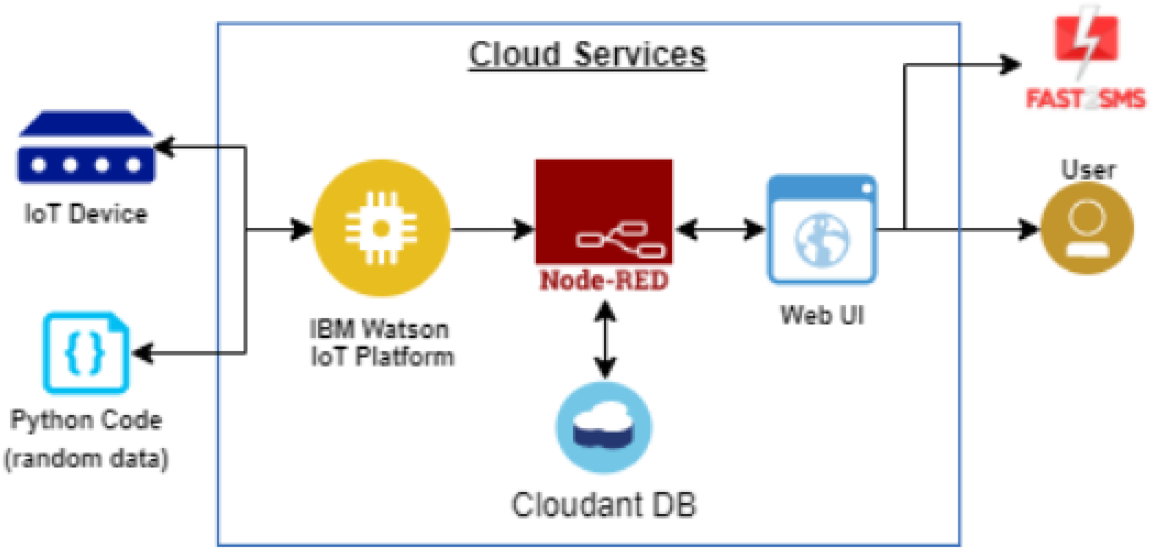
4.2 Non Functional Requirement

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Easy to use and economical Easy
NFR-2	**Security**	Protect your software from attacks
NFR-3	**Reliability**	Fast response timer. Highly reliable. Application is running correctly
NFR-4	**Performance**	If a fire is detected, you will be notified immediately through the web application and tracking will also be done regularly.
NFR-5	**Availability**	Availability of the system in institutions , restaurants and other public places.
NFR-6	**Scalability**	It can be easily modified to suit different needs.

5.PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture

Table-1: Components&Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI ,Node-RED, MITapp	IBM IoT Platform, IBM Node red,IBM Cloud
2.	ApplicationLogic-1	Create Ibm Watson IoT platform and create node-red service	Ibm Watson ,ibm cloudant service, ibm node-red
3.	ApplicationLogic-2	Develop python script to publish and subscribeto IBM IoT Platform	python
4.	ApplicationLogic-3	Build a web application using	IBM Node-red

		node-red service	
5.	Database	Data Type, Configuration set c.	MySQL
6.	Cloud Database	Database Service on Cloud	IBMDB2, IBM Cloudant
7.	File Storage	Developing mobile application to store and receive the sensors information and to react accordingly	Web UI, python
8.	ExternalAPI-1	Using this IBM fire management API we can track the temperature of the incident place andwhere the fire had been attacked.	IBM fire management API
9.	ExternalAPI-2	Using this IBM Sensors it detects the fire, Gas leaks, temperature and providesthe activation of sprinklers to web UI	IBM Sensors
10.	MachineLearningModel	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

Table-2:ApplicationCharacteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	MIT app Inventor	MIT License
2.	Security Implementations	IBM Services	Encryptions, IBM Controls
3.	Scalable Architecture	sensor-IoT Cloud based architecture	Cloud computing and AI
4.	Availability	Mobile ,laptop, desktop	MIT app
5.	Performance	Detects the Fire, gasleak, temperature	sensors

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	As a user, you can register in the application by entering your email address, password, and confirming the password.	Can access my account/dashboard.	High	Sprint-1
		USN-2	As a user, you will receive a confirmation email after registering in the application	You will receive a confirmation email and can click Confirm.	High	Sprint-1
		USN-3	As a user, you can register in the application via Facebook.	You can register with your Facebook login to access your dashboard	Low	Sprint-2
		USN-4	As a user, you can register in the application via Gmail.		Medium	Sprint-1
	Login	USN-5	As a user, you can login to the application by entering your email and password		High	Sprint-1
	Dashboard					
Customer (Webuser)						
Customer Care Executive						
Administrator						

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	Registration	USN-1	As a user, you can register in the application by entering your email address, password, and confirming the password.	2	High	Mukesh Kumar A
Sprint-2	Registration	USN-2	As a user, you will receive a confirmation email after registering in the application.	3	Medium	Vigneshwar R
Sprint-3	Registration	USN-3	As a user, you can register in the application via Facebook	1	Low	Subaash G

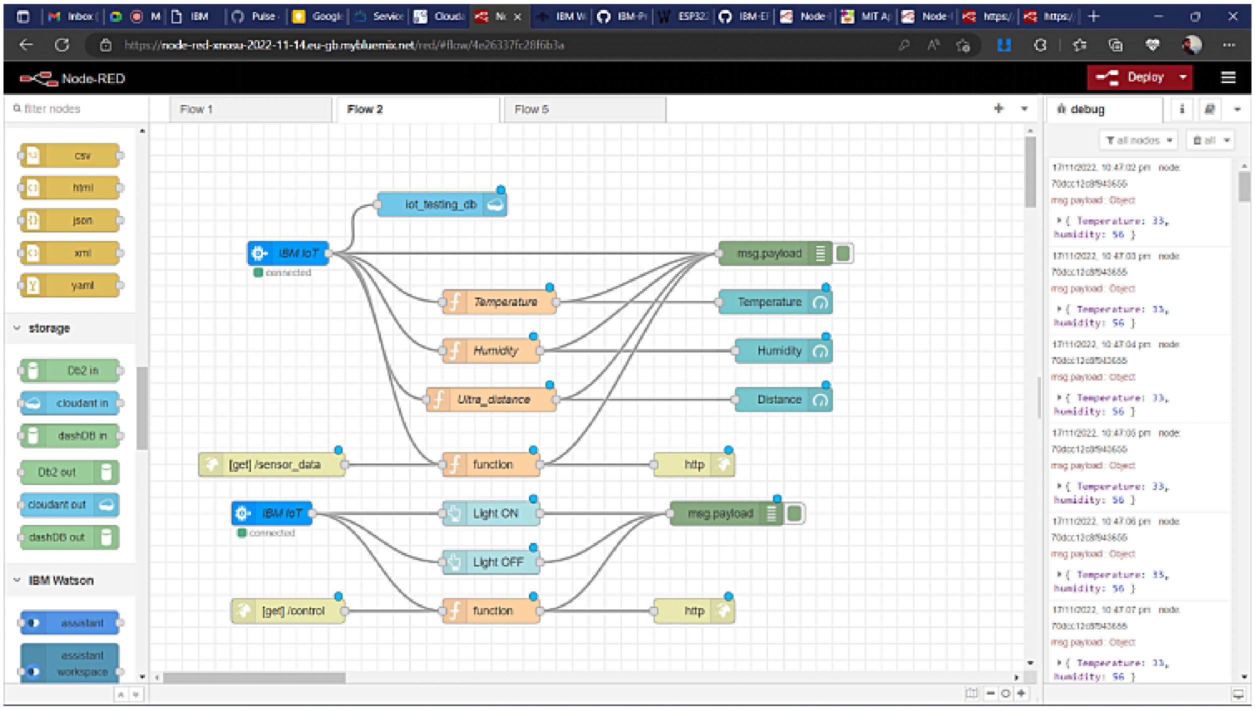
Sprint-2	Registration	USN-4	As a user, you can register in the application via Gmail.	1	High	Vigneshwar R
Sprint-4	Login	USN-5	As a user, you can login to the application by entering your email and password.	5	High	Jayanthan C
Sprint-1	Dashboard	USN-6	As a user, you can receive notifications.	1	Medium	Subaash G
Sprint-3	Testing & Deployment Phase-I	USN-7	System performance testing. Use for emergencies.	8	High	Mukesh kumar A
Sprint-3	Testing & Deployment Phase-I	USN-8	Make sure the system detects fire and gas correctly. Additionally, the user will be notified.	2	High	Vigneshwar R
Sprint-1	Deployment Phase-II &Model Improvement	USN-9	By providing an IOT-based industry-specific fire control system, the system can be checked and used 24/7.	1	Low	Mukesh kumar A
Sprint-2	Verification	USN-10	Administrators have full visibility of submitted applications.	5	High	Subaash G
Sprint-3	Approval	USN-11	After completion, the customer will be provided with new bank access data.	2	High	Jayanthan C

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	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

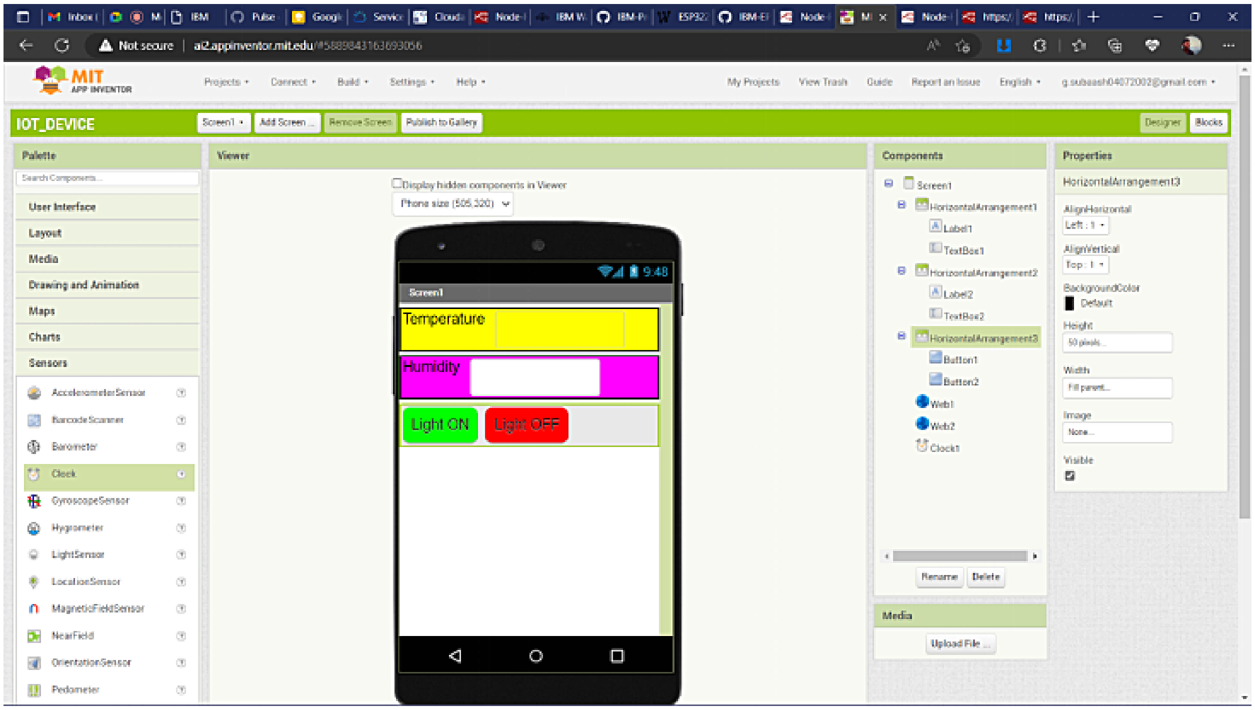
7. CODING & SOLUTIONING

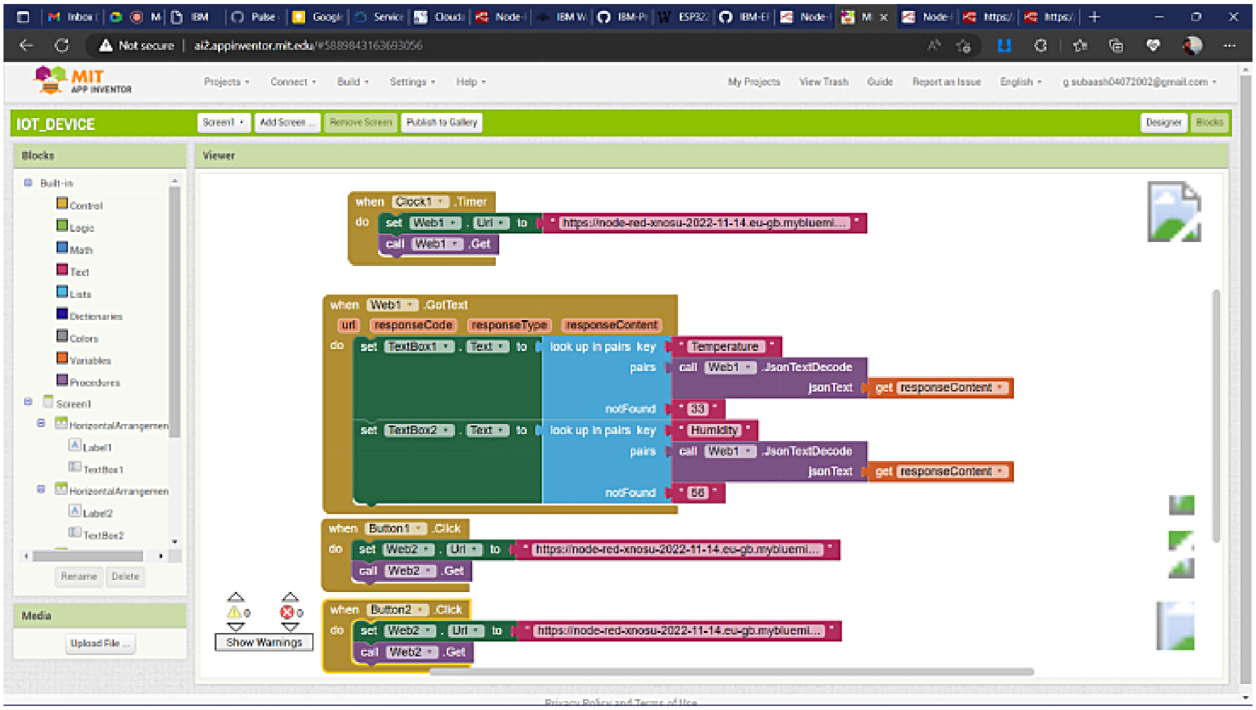
7.1 Feature 1



```
17/11/2022, 10:47:02 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
17/11/2022, 10:47:03 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
17/11/2022, 10:47:04 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
17/11/2022, 10:47:05 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
17/11/2022, 10:47:06 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
17/11/2022, 10:47:07 pm  node:
70dcc12c8f943656
msg.payload : Object
 ▸{ Temperature: 33,
humidity: 56 }
```

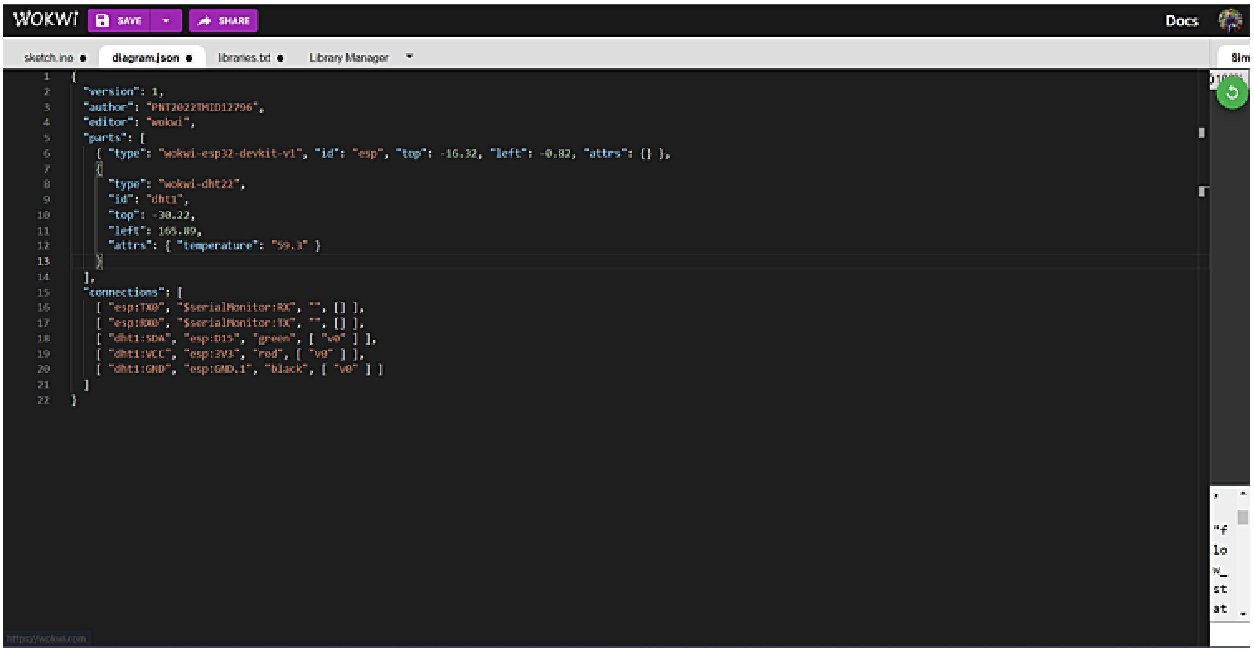
7.1 Feature 1



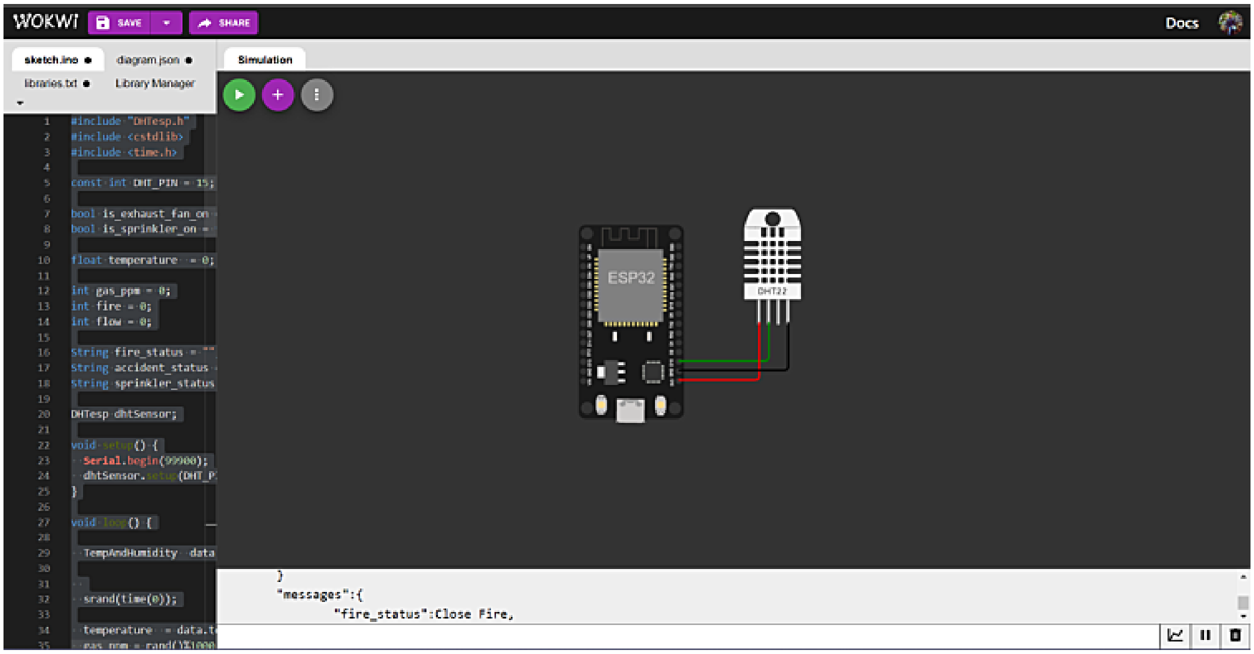


```
when Clock1 .Timer
do  set Web1 . Url to " https://node-red-xnosu-2022-11-14.eu-gb.mybluemi... "
    call Web1 .Get

when Web1 .GotText
 url  responseCode  responseType  responseContent
do  set TextBox1 . Text to  look up in pairs key " Temperature "
                                           pairs  call Web1 .JsonTextDecode
                                                            jsonText  get responseContent
                                        notFound " 33 "
    set TextBox2 . Text to  look up in pairs key " Humidity "
                                           pairs  call Web1 .JsonTextDecode
                                                            jsonText  get responseContent
                                        notFound " 56 "

when Button1 .Click
do  set Web2 . Url to " https://node-red-xnosu-2022-11-14.eu-gb.mybluemi... "
    call Web2 .Get

when Button2 .Click
do  set Web2 . Url to " https://node-red-xnosu-2022-11-14.eu-gb.mybluemi... "
    call Web2 .Get
```

7.2 Feature 2



```
{
  "version": 1,
  "author": "PNT2022TMID12796",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -16.32, "left": -0.82, "attrs": {} },
    {
      "type": "wokwi-dht22",
      "id": "dht1",
      "top": -30.22,
      "left": 165.89,
      "attrs": { "temperature": "59.3" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]
  ]
}
```



```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>

const int DHT_PIN = 15;

bool is_exhaust_fan_on
bool is_sprinkler_on =

float temperature  = 0;

int gas_ppm = 0;
int fire = 0;
int flow = 0;

String fire_status = ""
String accident_status
String sprinkler_status

DHTesp dhtSensor;

void setup() {
  Serial.begin(99900);
  dhtSensor.setup(DHT_P
}

void loop() {

  TempAndHumidity  data

  
  srand(time(0));

  temperature  = data.t
  gas_ppm = rand()%1000
```

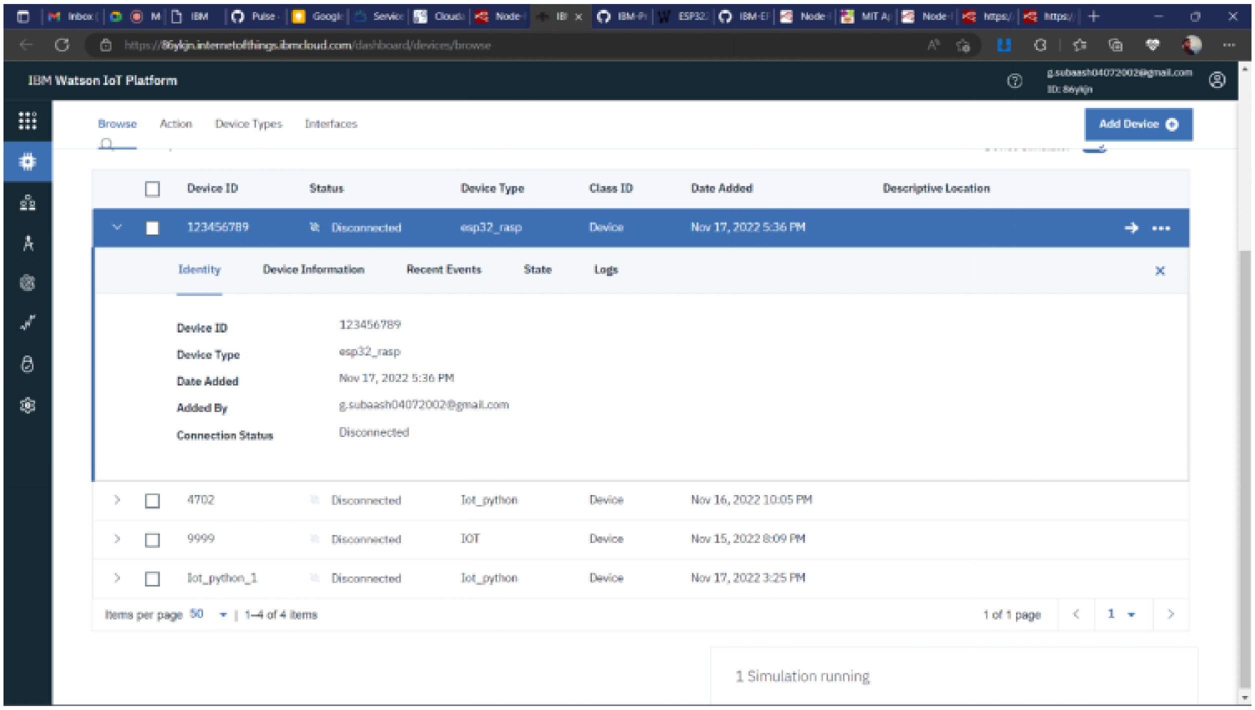
```
}
"messages":{
        "fire_status":Close Fire,
```

PROGRAM

```
#include "DHTesp.h"

#include <cstdlib>

#include <time.h>


const int DHT_PIN = 15;


bool is_exhaust_fan_on = false;

bool is_sprinkler_on = false;


float temperature  = 0;


int gas_ppm = 0;

int fire = 0;

int flow = 0;


String fire_status = "";

String accident_status = "";

String sprinkler_status = "";


DHTesp dhtSensor;


void setup() {
  Serial.begin(99900);
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
}

void loop() {

  TempAndHumidity  data = dhtSensor.getTempAndHumidity();


  srand(time(0));

  temperature  = data.temperature;
  gas_ppm = rand()%1000;
  int firereading = rand()%1024;
  fire = map(firereading,0,1024,0,1024);
  int firerange = map(firereading,0,1024,0,3);
  int flow = ((rand()%100)>50?1:0);

  switch (firerange) {
  case 2:
    fire_status = "Close Fire";
    break;
  case 1:
```

```
  fire_status = "Distant Fire";
  break;
case 0:
  fire_status = "No Fire";
  break;
}


if(gas_ppm > 100){
  is_exhaust_fan_on = true;
}
else{
  is_exhaust_fan_on = false;
}
if(temperature < 40 && firerange ==2){
  accident_status = "need auditing";
  is_sprinkler_on = false;
}
else if(temperature < 40 && firerange ==0){
  accident_status = "nothing found";
  is_sprinkler_on = false;
}
else if(temperature > 50 && firerange == 1){
  is_sprinkler_on = true;
  accident_status = "moderate";
}
else if(temperature > 55 && firerange == 2){
  is_sprinkler_on = true;
  accident_status = "severe";
}else{
  is_sprinkler_on = false;
  accident_status = "nil";
}

if(is_sprinkler_on){
  if(flow){
    sprinkler_status = "working";
  }
  else{
    sprinkler_status = "not working";
  }
}
else if(is_sprinkler_on == false){
  sprinkler_status = "now it shouldn't";
}
else{
    sprinkler_status = "something's wrong";
  }

  String out = "{\n\t\"senor_values\":{";
  out+="\n\t\t\"gas_ppm\":"+String(gas_ppm)+",";
  out+="\n\t\t\"temperature\":"+String(temperature,2)+",";
  out+="\n\t\t\"fire\":"+String(fire)+",";
  out+="\n\t\t\"flow\":"+String(flow)+",\n\t}";
  out+="\n\t\"output\":{";

out+="\n\t\t\"is_exhaust_fan_on\":"+String((is_exhaust_fan_on)?"true":"false")+",
";
  out+="\n\t\t\"is_sprinkler_on\":"+String((is_sprinkler_on)?"true":"false")+",";
  out+="\n\t}";
  out+="\n\t\"messages\":{";
  out+="\n\t\t\"fire_status\":"+fire_status+",";
  out+="\n\t\t\"flow_status\":"+sprinkler_status+",";
  out+="\n\t\t\"accident_status\":"+accident_status+",";
  out+="\n\t}";
  out+="\n}";
  Serial.println(out);
  delay(1000);
}
```

7.3 Database Schema



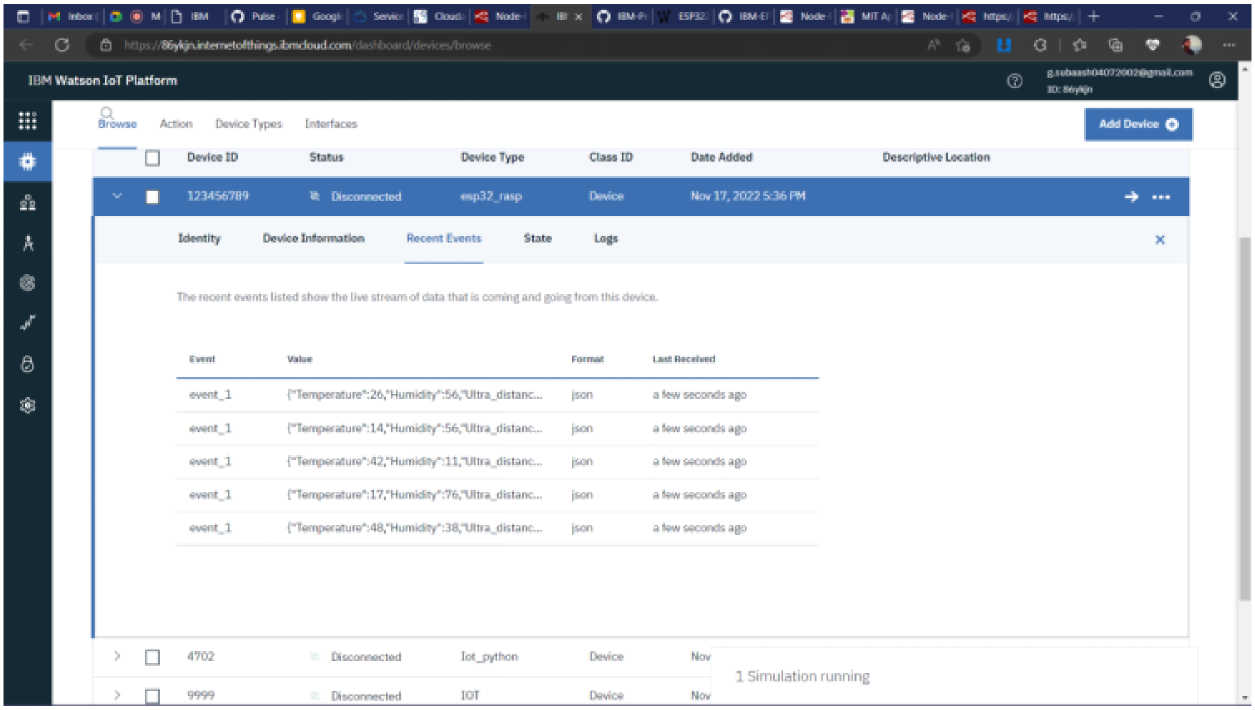
	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	123456789	Disconnected	esp32_rasp	Device	Nov 17, 2022 5:36 PM	

Device ID	123456789
Device Type	esp32_rasp
Date Added	Nov 17, 2022 5:36 PM
Added By	g.subaash04072002@gmail.com
Connection Status	Disconnected

	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	4702	Disconnected	Iot_python	Device	Nov 16, 2022 10:05 PM	
□	9999	Disconnected	IOT	Device	Nov 15, 2022 8:09 PM	
□	Iot_python_1	Disconnected	Iot_python	Device	Nov 17, 2022 3:25 PM	

Items per page 50 | 1–4 of 4 items   1 of 1 page   1

1 Simulation running



	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	123456789	Disconnected	esp32_rasp	Device	Nov 17, 2022 5:36 PM	

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

Event	Value	Format	Last Received
event_1	{"Temperature":26,"Humidity":56,"Ultra_distanc...	json	a few seconds ago
event_1	{"Temperature":14,"Humidity":56,"Ultra_distanc...	json	a few seconds ago
event_1	{"Temperature":42,"Humidity":11,"Ultra_distanc...	json	a few seconds ago
event_1	{"Temperature":17,"Humidity":76,"Ultra_distanc...	json	a few seconds ago
event_1	{"Temperature":48,"Humidity":38,"Ultra_distanc...	json	a few seconds ago

	Device ID	Status	Device Type	Class ID	Date Added
□	4702	Disconnected	Iot_python	Device	Nov
□	9999	Disconnected	IOT	Device	Nov

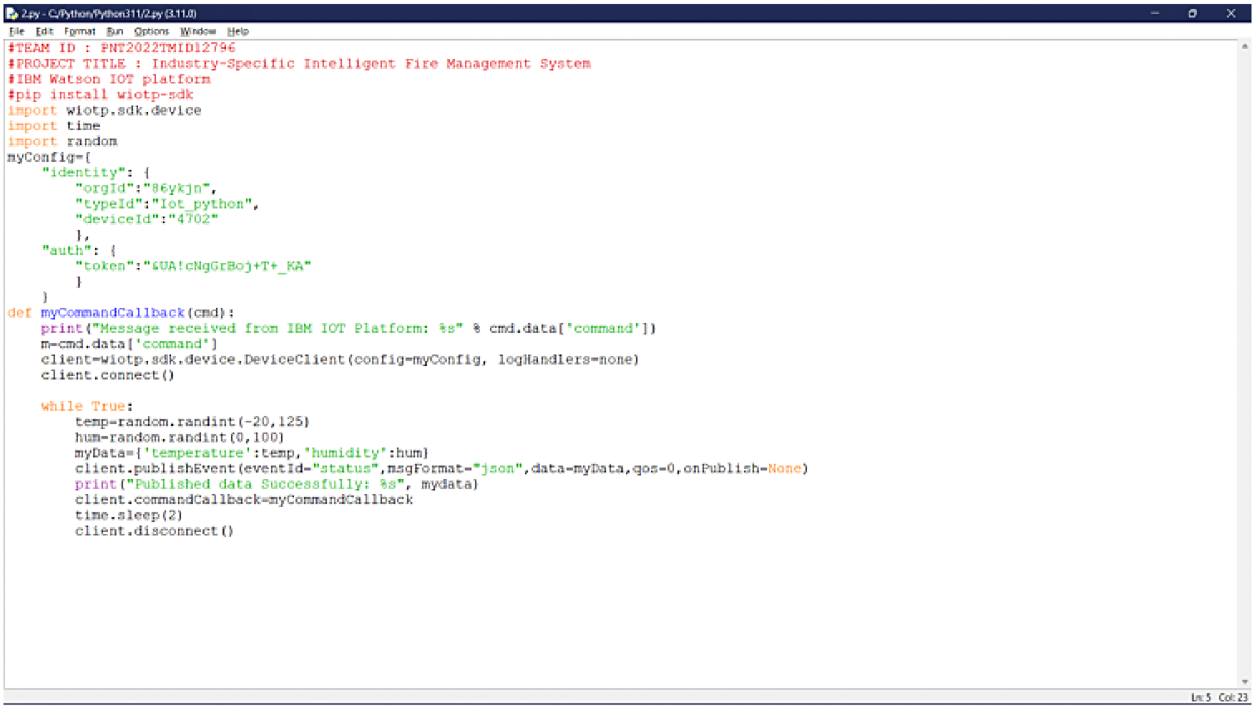
1 Simulation running

8.Testing

8.1 Test Cases

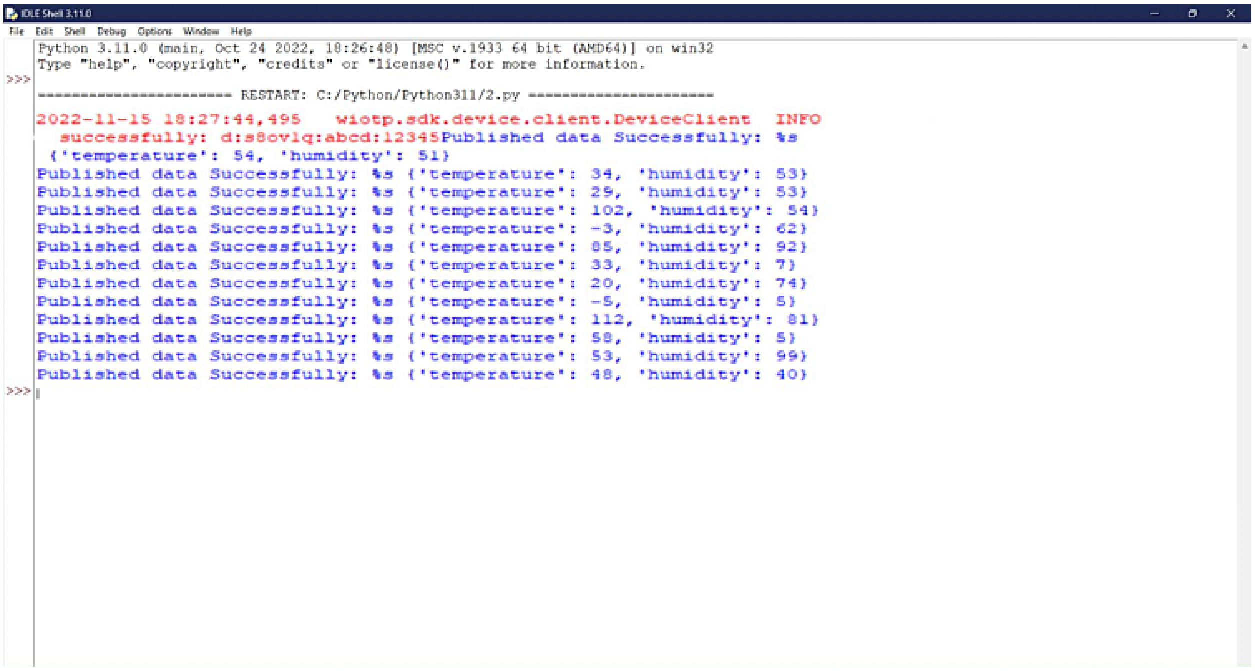
SL.NO	INPUT	OUTPUT	RESULT
01.	Gas:519 Temperature:59.30 Flame:427	Exhaust fan on:TRUE Sprinklers:ON	Passed
02.	Gas:804 Temperature:59.30 Flame:589	Exhaust fan on:TRUE Sprinklers:ON	Passed
03.	Gas:738 Temperature:59.30 Flame:751	Exhaust fan on:TRUE Sprinklers:ON	Passed
04.	Gas:23 Temperature:59.30 Flame:913	Exhaust fan on:FALSE Sprinklers:ON	Passed
05.	Gas:308 Temperature:59.30 Flame:51	Exhaust fan on:TRUE Sprinklers:OFF	Passed
06.	Gas:241 Temperature:59.30 Flame:213	Exhaust fan on:TRUE Sprinklers:OFF	Passed
07.	Gas:527 Temperature:59.30 Flame:375	Exhaust fan on:TRUE Sprinklers:ON	Passed
08.	Gas:812 Temperature:59.30 Flame:537	Exhaust fan on:TRUE Sprinklers:ON	Passed
09.	Gas:745 Temperature: 59.30 Flame:699	Exhaust fan on:TRUE Sprinklers:ON	Passed
10.	Gas:31 Temperature:59.30 Flame:861	Exhaust fan on:FALSE Sprinklers:ON	Passed
11.	Gas:316 Temperature:59.30 Flame:1023	Exhaust fan on:TRUE Sprinklers:ON	Passed

8.2 User Acceptance Testing



2.py - C:/Python/Python311/2.py (3.11.0)
File Edit Format Run Options Window Help
Ln: 5 Col: 23

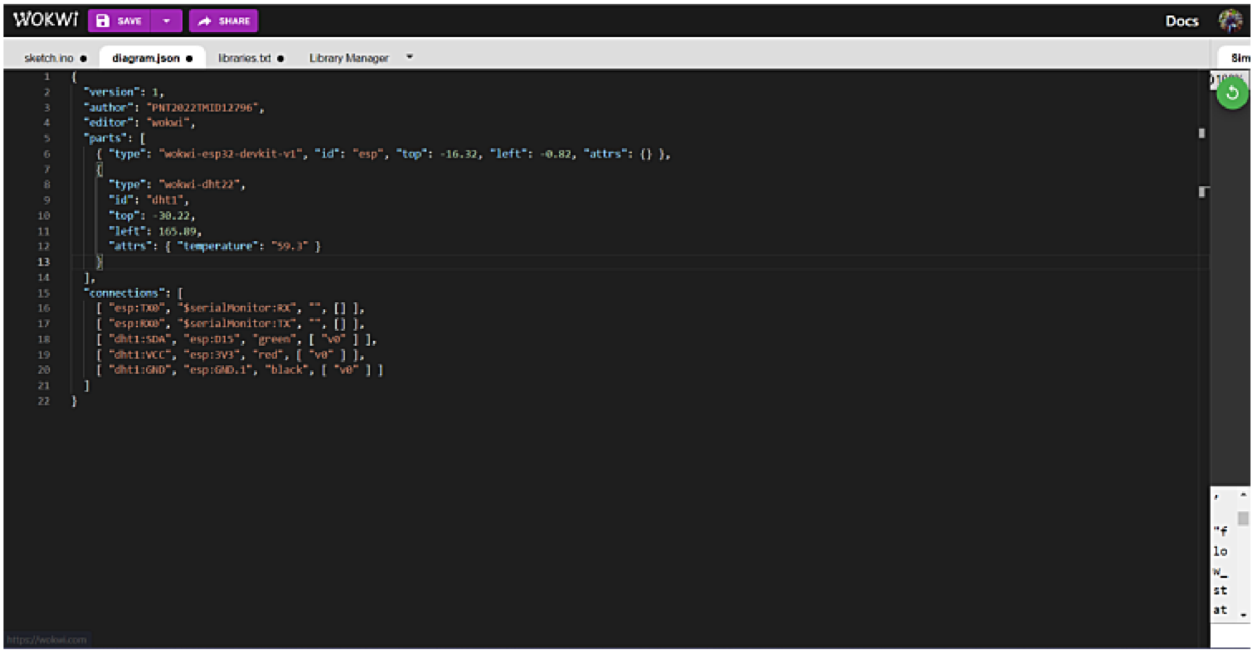
```
#TEAM ID : PNT2022TMID12796
#PROJECT TITLE : Industry-Specific Intelligent Fire Management System
#IBM Watson IOT platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig={
    "identity": {
        "orgId":"86ykjn",
        "typeId":"Iot_python",
        "deviceId":"4702"
        },
    "auth": {
        "token":"&UA!cNgGrBoj+T+_KA"
        }
    }
def myCommandCallback(cmd):
    print("Message received from IBM IOT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
    client=wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=none)
    client.connect()

    while True:
        temp=random.randint(-20,125)
        hum=random.randint(0,100)
        myData={'temperature':temp,'humidity':hum}
        client.publishEvent(eventId="status",msgFormat="json",data=myData,qos=0,onPublish=None)
        print("Published data Successfully: %s", mydata)
        client.commandCallback=myCommandCallback
        time.sleep(2)
        client.disconnect()
```



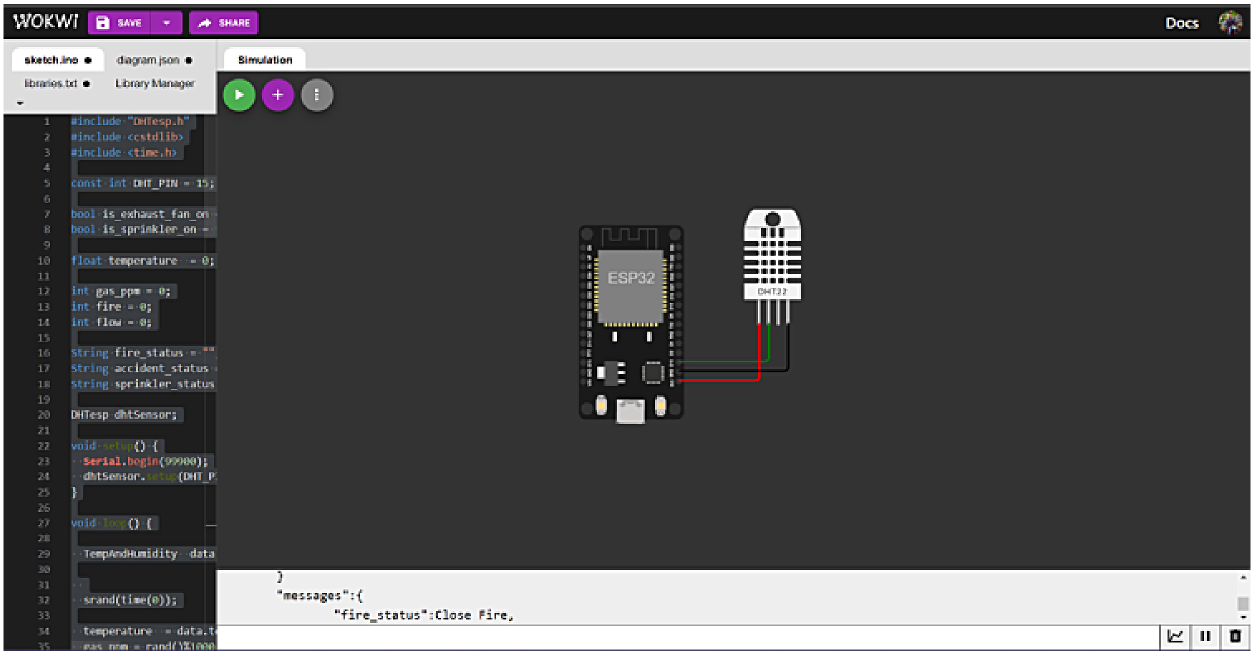
```
    Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    ======================= RESTART: C:/Python/Python311/2.py ======================
    2022-11-15 18:27:44,495   wiotp.sdk.device.client.DeviceClient  INFO
      successfully: d:s8ovlq:abcd:12345Published data Successfully: %s
     {'temperature': 54, 'humidity': 51}
    Published data Successfully: %s {'temperature': 34, 'humidity': 53}
    Published data Successfully: %s {'temperature': 29, 'humidity': 53}
    Published data Successfully: %s {'temperature': 102, 'humidity': 54}
    Published data Successfully: %s {'temperature': -3, 'humidity': 62}
    Published data Successfully: %s {'temperature': 85, 'humidity': 92}
    Published data Successfully: %s {'temperature': 33, 'humidity': 7}
    Published data Successfully: %s {'temperature': 20, 'humidity': 74}
    Published data Successfully: %s {'temperature': -5, 'humidity': 5}
    Published data Successfully: %s {'temperature': 112, 'humidity': 81}
    Published data Successfully: %s {'temperature': 58, 'humidity': 5}
    Published data Successfully: %s {'temperature': 53, 'humidity': 99}
    Published data Successfully: %s {'temperature': 48, 'humidity': 40}
>>>
```

9.RESULTS

9.1 Performance Metrics

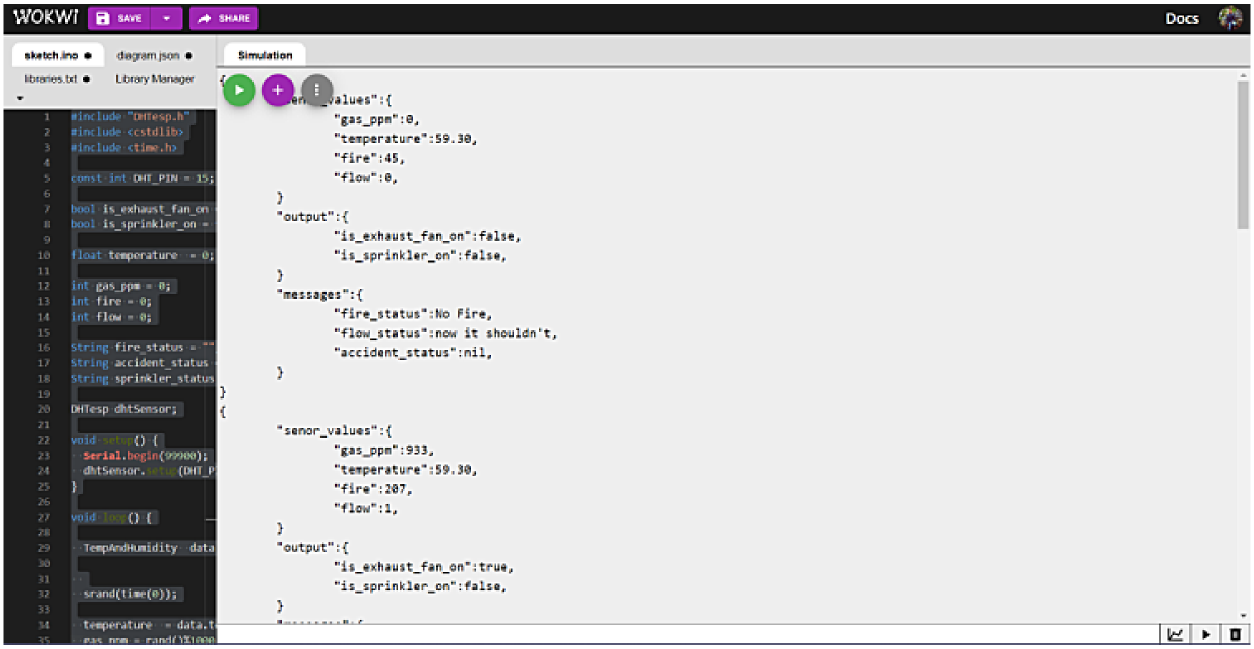


```
{
  "version": 1,
  "author": "PNT2022TMID12796",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -16.32, "left": -0.82, "attrs": {} },
    {
      "type": "wokwi-dht22",
      "id": "dht1",
      "top": -30.22,
      "left": 165.89,
      "attrs": { "temperature": "59.3" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]
  ]
}
```



```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>

const int DHT_PIN = 15;

bool is_exhaust_fan_on
bool is_sprinkler_on =

float temperature  = 0;

int gas_ppm = 0;
int fire = 0;
int flow = 0;

String fire_status = ""
String accident_status
String sprinkler_status

DHTesp dhtSensor;

void setup() {
  Serial.begin(99900);
  dhtSensor.setup(DHT_P
}

void loop() {

  TempAndHumidity  data


  srand(time(0));

  temperature  = data.t
  gas_ppm = rand()%1000
```

```
}
"messages":{
        "fire_status":Close Fire,
```



```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>

const int DHT_PIN = 15;

bool is_exhaust_fan_on
bool is_sprinkler_on =

float temperature  = 0;

int gas_ppm = 0;
int fire = 0;
int flow = 0;

String fire_status = ""
String accident_status
String sprinkler_status

DHTesp dhtSensor;

void setup() {
  Serial.begin(99900);
  dhtSensor.setup(DHT_P
}

void loop() {

  TempAndHumidity  data

  
  srand(time(0));

  temperature  = data.t
  gas_ppm = rand()%1000
```

```
{
        en_values":{
                "gas_ppm":0,
                "temperature":59.30,
                "fire":45,
                "flow":0,
        }
        "output":{
                "is_exhaust_fan_on":false,
                "is_sprinkler_on":false,
        }
        "messages":{
                "fire_status":No Fire,
                "flow_status":now it shouldn't,
                "accident_status":nil,
        }
}
{

        "senor_values":{
                "gas_ppm":933,
                "temperature":59.30,
                "fire":207,
                "flow":1,
        }
        "output":{
                "is_exhaust_fan_on":true,
                "is_sprinkler_on":false,
        }
```

10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

- **Addressable fire alarm systems give information about individual detectors, whereas conventional systems only give information about specific circuits (zones).**

- **Addressable systems allow a courtesy text label to allow easy identification of any event. For instance, detector 1 may be given the label 'Bedroom 1'.**

- **Most addressable systems allow an early 'prealarm' warning, which allows the responsible person to investigate potential alarms before the system activates its sirens.**

- **Many addressable systems can alter the alarm threshold of the detectors, in order to meet the needs of different environments in different areas of the system.**

- **Addressable systems are usually wired in a loop.**

- **Conventional systems are usually wired as radial circuits.**

- **Addressable systems usually have a real time clock & event log to record system events.**

- **Larger addressable systems usually have the ability to use sophisticated programming options to operate certain outputs only with specific events**

DISADVANTAGES

- **The one thing most fire alarm system inspectors caution against with wireless systems is having to replace the battery. The system is essentially useless if the batteries aren’t charged, since it won’t work properly. There is a bit of a burden to homeowners or business owners to always remember to keep the batteries fresh so the system operates properly when you need it most.**

- **A couple other disadvantages fire alarm system inspectors point out is wireless systems have limited range and don’t have centralized monitoring. Range can be a problem for large offices or homes, since a weak wireless connection may cause the system to not operate reliably. Wireless fire alarm systems also don’t connect directly to the telephone lines, which are linked to the fire departments, so the response to an emergency could be slower as a result.**

11. CONCLUSION

There is a general agreement over the fire and protection segments that at 220,000 for every annum the degrees of bogus and undesirable alarms radiating from fire alarm and discovery frameworks is excessively high. Bogus and undesirable alarms squander fire and salvage administration assets; cause superfluous and costly interruption to end-clients which can bring about the loss of trust in frameworks and has seen a few frameworks turned off. As fire alarm and identifications frameworks are so firmly inserted into the clearing systems and strategies created to meet the necessities of Building Regulations and Fire Safety Law their utilization is far reaching and there are entrenched outsider accreditation plans for producers and installers. The item measures and testing systems anyway stay quiet on the reasons for bogus alarms.

12. FUTURE SCOPE

The scope of the fire safety systems market includes the type of safety system with fire detectors and suppressors, such as gas, foam, and detectors. The increasing focus of the government bodies on implementing fire safety equipment across various industries, such as chemical and petrochemical, oil and gas, pharmaceutical, aerospace, and defense, has led to the growth of the market studied.

13.APPENDIX

Source Code

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "86ykjn"
#define DEVICE_TYPE "assignment4"
#define DEVICE_ID "12345"
#define TOKEN "6DGHyn)mYb)gRuXJvt"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/event2/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);

const int trigpin=5;
const int echopin=18;
String command;
String data="";

long duration;
float dist;

void setup()
{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect();
  mqttConnect();
}
```

```
void loop() {
  bool isNearby = dist < 100;
  digitalWrite(led, isNearby);

  publishData();
  delay(500);

  if (!client.loop()) {
    mqttConnect();
  }
}

void wifiConnect() {
  Serial.print("Connecting to "); Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "", 6);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.print("WiFi connected, IP address: "); Serial.println(WiFi.localIP());
}

void mqttConnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting MQTT client to "); Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
    initManagedDevice();
    Serial.println();
  }
}

void initManagedDevice() {
  if (client.subscribe(topic)) {
    // Serial.println(client.subscribe(topic));
    Serial.println("IBM subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}
void publishData()
{
 digitalWrite(trigpin,LOW);
 digitalWrite(trigpin,HIGH);
 delayMicroseconds(10);
 digitalWrite(trigpin,LOW);
 duration=pulseIn(echopin,HIGH);
 dist=duration*speed/2;
 if(dist<100){
  String payload = "{\"Alert!! Alert!!  Distance\":";
  payload += dist;
  payload += "}";

  Serial.print("\n");
  Serial.print("Sending payload: ");
  Serial.println(payload);
  if (client.publish(publishTopic, (char*) payload.c_str())) {
   Serial.println("Publish OK");
  }

 }
  if(dist>100){
  String payload = "{\"Distance\":";
  payload += dist;
  payload += "}";

  Serial.print("\n");
  Serial.print("Sending payload: ");
  Serial.println(payload);
   if(client.publish(publishTopic, (char*) payload.c_str())) {
   Serial.println("Publish OK");
  }else {
   Serial.println("Publish FAILED");
  }

 }
 }
```

GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-29972-1660136467