

PROJECT DOCUMENTATION REPORT

DATE	19 NOVEMBER 2022
TEAM ID	PNT2022TMID11818
PROJECT TITLE	INDUSTRY-SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM

TEAM ID:PNT2022TMID11818

TEAM LEADER:GOWTHAMRAJ V

TEAM MEMBER 1:LOGESH K

TEAM MEMBER 2:GOPALAKRISHNAN N

TEAM MEMBER 3:GOWTHAM J

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1

7.2 Feature 2

8. TESTING

8.1 Test Cases

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES 11. CONCLUSION 12. FUTURE SCOPE 13.

APPENDIX

Source Code

GitHub & Project Demo Link

1. INTRODUCTION

1.1 Project Overview

Abstract :

Fire prevention and protection in industrial plant basically include procedures for preventing, detecting and extinguishing fires. There is much to be done to promote safe systems for fire prevention in India. The systems should be designed to protect the employees and property and to assure safe working systems. Fire prevention and protection in any industrial plant include fire detection system, fire alarm system, fire prevention plan and employing legal requirements relevant to fire and safety management.

1.2 Purpose

Nowadays, fire incidents have become a critical issue, which must be dealt with on time without any unnecessary delay to avoid the loss in lives and belongings. It is considered a fire situation when the monitored temperature exceeds 50°C. According to the national fire protection association (NFPA), two-third of U.S. household fires occur in premises with no working smoke alarms, alarms with no proper maintenance, or misplaced alarms.

This is the IOT (internet of things) based fire monitoring and controlling system which not only gives the real time information about the situation on the monitor but also takes the corrective action as per the need. In a publish and subscribe system, a device can publish a message on a topic, or it can be subscribed to a particular topic to receive message. Also it is perfect solution for internet of things application. Due to this all data can be stored in server and this data can be access by the Application program interface which we can display on the monitor and with the help of software, the operator can visualize the condition at the time of fire accident.

2. LITERATURE SURVEY

2.1 Existing problem :

In the Industries, mostly manufacturing sectors are affected due to fire accidents. Manufacturing sectors may contain specific equipment's, that can lead to such kind of accidents. As fire accidents are considered that may include the Geographic and confined spaces of the industry also. Many of the fire accidents occurs in the industries are unexpected which could cause life and property damage. The major reasons are neglecting and breaking the government norms. The fire accidents may occur frequently in large scale heavy manufacturing industries. To create safe working conditions and to reduce the property loses, the problem solution is needed.

2.2 References :

- [1] Shokouhi, M., Nasiriani, K., Khankeh, H., Fallahzadeh, H. and Khorasani-Zavareh, D., "Exploring barriers and challenges in protecting residential fire-related injuries: a qualitative study," *Journal of injury and violence research*, vol. 11, no. 1, pp. 81-92, 2019.
- [2] V. Jelcic; M. Magno; G. Paci; D. Brunelli; L. Benini, "Design, characterization and management of a wireless sensor network for smart gas monitoring," in 2011 4th IEEE Int. Workshop on Adv. in Sensors and Interfaces (IWASI), pp. 115-120.
- [3] Saeed, F., Paul, A., Karthigaikumar, P. and Nayyar, A., "Convolutional neural network based early fire detection," *Multimedia Tools and Applications*, vol. 79, pp. 9083-9099, 2020.
- [4] Shah, R., Satam, P., Sayyed, M.A. and Salvi, P., "Wireless Smoke Detector and Fire Alarm System," *International Research Journal of Engineering and Technology (IRJET)*, vol. 6, no. 1, pp. 1407-1412, 2019.
- [5] F. He; Z. Du; Y. Sun, "Indoor dangerous gas environment detected by mobile robot," in 2009 IEEE International Conference on Robotics and Biomimetic (ROBIO), pp. 396-401.
- [6] Mahzan, N. N., Enzai, N. M., Zin, N. M. and Noh, K. S. S. K. M., "Design of an Arduino based home fire alarm system with GSM module," *Journal of Physics: Conference Series*, vol. 1019, no. 1, 2018, Art. no. 012079.

2.3 Problem Statement Definition

Safety is considered first in the designing of residential and commercial buildings in order to safe guard the property damage as well as loss of life. Nowadays the existing fire alarm in the markets is too complex in terms of its design and structures. It needs regular maintenance to be carried out to make sure the system performance. Meanwhile the maintenance is being done to the existing system, it could raise the cost of the system.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



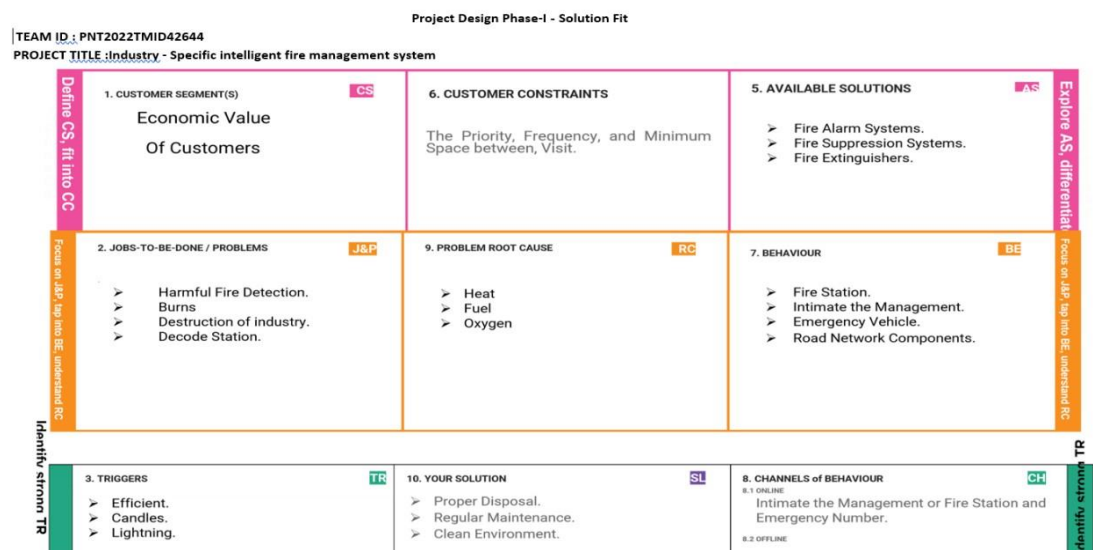
3.2 Ideation & Brainstroming

[illegible]

3.3 Proposed Solution

S.No.	Parameters	Outcomes
1.	Problem Statement (Problem to be solved)	*Improving the Fire management and alert system in industries. And also providing a safe environment for the workers...
2.	Idea / Solution description	*Providing fire management and alert system in industries on IoT with fire detection and using some sensors (Gas sensor, flame sensor, temperature sensor).
3.	Novelty / Uniqueness	*An IoT system that is integrated with temperature sensor, gas sensor, flame sensor, exhaust fan, water sprinkler and fire alert through SMS to the authority...
4.	Social Impact / Customer Satisfaction	*We can also able to reduce the damages caused by fire accidents in industries using this device by alerting everyone early....
5.	Business Model (Revenue Model)	*This product can be used in industries. This system can be used as a safety system in industries. * High Efficient and Secure...
6.	Scalability of the Solution	*This project is trying to execute the method with the help of an IoT gadget which provides a safe environment for workers in industries. *Easy to Install and Maintain...

3.4 Problem Solution fit



<p>4. EMOTIONS: BEFORE / AFTER</p> <p>BEFORE: Detection of Fires.</p> <p>AFTER: To secure the Objects or Things.</p>	EMI	Remove the Fire Burn Things
--	-----	-----------------------------

4.REQUIRMENT ANALYSIS

4.1 Functional Requirements

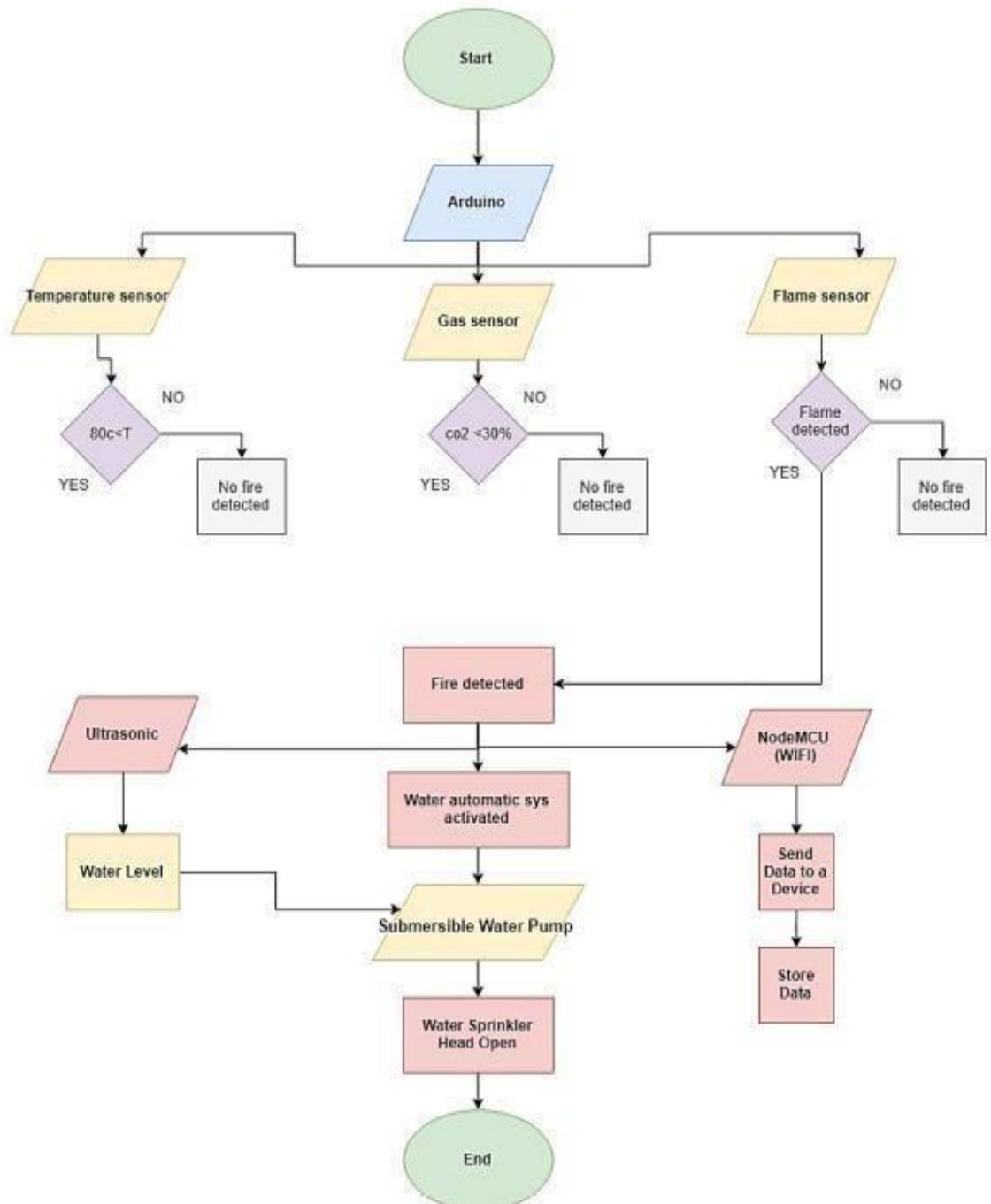
	Functional Requirement	
FR No.	(Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Requirements	Workers and Product Protection Automatic Sprinkler System Monitors Smoke ,Gas and Temperature
FR-2	User Registration	Manual Registration Registration through webpage Registration through Form Registration through Gmail
FR-3	User Confirmation	Confirmation via Phone Confirmation via Email Confirmation via OTP
FR-4	Payment Options	Cash on Delivery Net Banking/UPI Credit/Debit/ATM Card
FR-5	Product Delivery and Installation	Door Step delivery Take away Free Installation and 1 year Warranty
FR-6	Product Feedback	Through Webpage Through Phone calls Through Google forms

4.2 Non-Functional Requirements

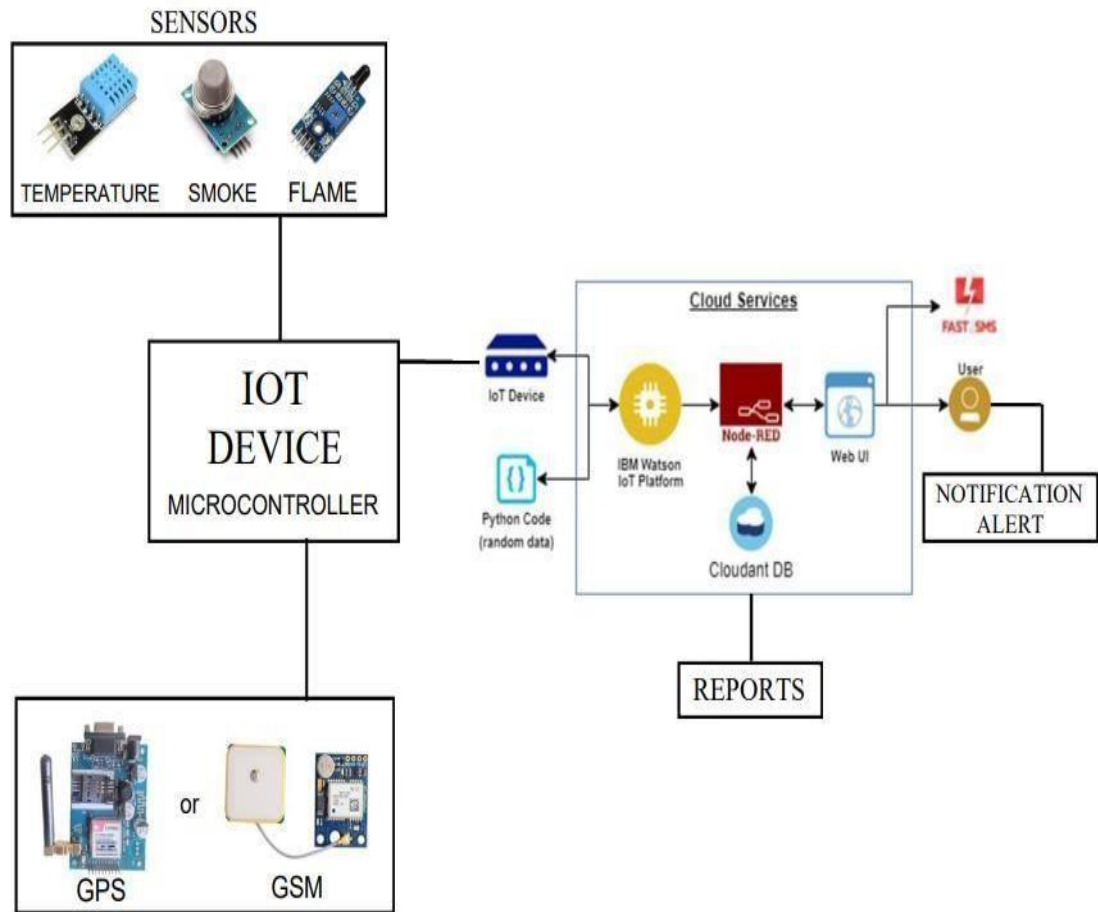
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Have a clear and self-explanatory manual. Easier to use. Easily accessible by everyone.
NFR-2	Security	Are inspected monthly by the Fire Alarm Technician. Inspected and tagged by a contractor annually.
NFR-3	Reliability	Hardware requires a regular checking and service. Software may be updated periodically. Immediate alert is provided in case of any system failure.
NFR-4	Performance	The equipment must have a good user interface It should have a minimal energy requirement It has to save lives of people and things.
NFR-5	Availability	All the features will be available when the user requires. It depends on the need of the user and the customization of the user has done.
NFR-6	Scalability	The product has to cover all the space of industry irrespective of the size or area.

5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



5.3 User Stories

User Type	Functional Requirement	User Story Number	User Story/Task	Acceptance Criteria	Priority	Release
Customers (Mobile user, web user, care executive, Administrator)	Registration	USN - 1	As a user, I can register for the application by entering my mail, password, and confirming my password	I can access my account/ dashboard	High	Sprint - 1
		USN - 2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint - 1
	Dashboard	USN - 3	As a user, I can register for the application through internet	I can register & access the dashboard with Internet login	Low	Sprint - 2
		USN - 4	As a user, I can register for the application through Gmail	I can confirm the registration in Gmail	Medium	Sprint - 1
	Login	USN - 5	As a user, I can log into the application by entering email & password	I can login with my ID & password	High	Sprint - 1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

S.No	Milestones	Activities	Timeline
1.	Literature Survey & Information Gathering	Prepare Literature Survey for the selected project & gather necessary information.	29 Aug - 3rd Sept 2022
2.	Empathy Map	Prepare Empathy Map Canvas to capture the user gains and pains.	5 - 10th Sept 2022
3.	Brainstorm	Ideas are listed and top 3 ideas are prioritized.	12 - 17 Sept 2022
4.	Proposed Solution	Proposed solution document is prepared which includes novelty, feasibility of idea, social impact, scalability of solution.	19 - 24 Sept 2022
5.	Problem Solution Fit	Includes customer segments and customer constraints, the problem root cause and jobs to be done.	26 Sept - 01 Oct 2022
6.	Solution Architecture	From receiving data to notifying user are represented in architectural diagrams.	26 Sept - 01 Oct 2022
7.	Customer Journey	Prepare customer journey maps to understand user interactions and experiences with the applications.	3 - 8 Oct 2022
8.	Functional Requirements	Functional Requirements and non functional requirements like scalability, and accuracy are described.	10 - 15 Oct 2022
9.	Data Flow Diagrams and User Stories	Data flow diagrams and user stories are prepared and four sprint phases are described.	10 - 15 Oct 2022
10.	Technology Architecture	Technical flow graphs are created and the functions of technical stacks are defined.	10 - 15 Oct 2022

11.	Prepare Milestones & Activities List	Prepare the list for milestones & activities.	17 - 22 Oct 2022
12.	Sprint Delivery Plan	Prepare the sprint delivery plan of the project.	17 - 22 Oct 2022
13.	Project Development – Delivery of Sprint 1,2,3& 4	Develop and submit the developed code by testing it.	29 Oct 2022 - 19 Nov 2022

6.2 Sprint Delivery Schedule

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration (Mobile user)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	1	Low	Logesh K Gowtham J Gowthamraj V
Sprint-1	Login (Mobile user)	USN-2	As a user, I can log into the application by entering email & password	3	High	Logesh K Gopalakrishnan N Gowtham J
Sprint-2	Dashboard (Mobile user)	USN-3	By entering correct password, I could access the dashboard.	13	Medium	Gowtham J Gopalakrishnan N Gowthamraj V
Sprint-3	Alert message (Mobile user)	USN-4	As a user, I can get alert messages regarding smoke and temperature parameters.	13	High	Logesh K Gopalakrishnan N Gowthamraj V
Sprint-4	Data storage (Mobile user)	USN-5	As a user, I will able to store parameter values.	2	High	Gowtham J Logesh K
Sprint-4	Checking (Mobile & web user)	USN-6	As a user I can Test the system performance, for an emergency case, it is deployed and I can use the system 24/7.	8	High	Gopalakrishnan N Gowtham J Logesh K
Sprint-1	Login (web user)	USN-7	As a user, I can log into the application by entering email & password	13	High	Gowthamraj V Gowtham J

Sprint-1	Dashboard (web user)	USN-8	I could access the dashboard.	3	Medium	Gopalakrishnan N Gowtham J
Sprint-3	Alert message (web user)	USN-9	As a user, I can get alert messages regarding smoke and temperature parameters.	5	High	Logesh K Gowthamraj V
Sprint-4	Data Storage message (web user)	USN-10	As a user, I will able to store parameter values.	2	High	Logesh K Gopalakrishnan N
Sprint-4	Checking (Mobile & web user)	USN-11	As a user, I can check whether the system correctly detects the fire and gas, and does it alerts the user, also whether the fire or smoke has been put down or not.	8	High	Gopalakrishnan N Gowtham J

Project Tracker, Velocity & Burndown Chart:

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	17 Nov 2022

Velocity:

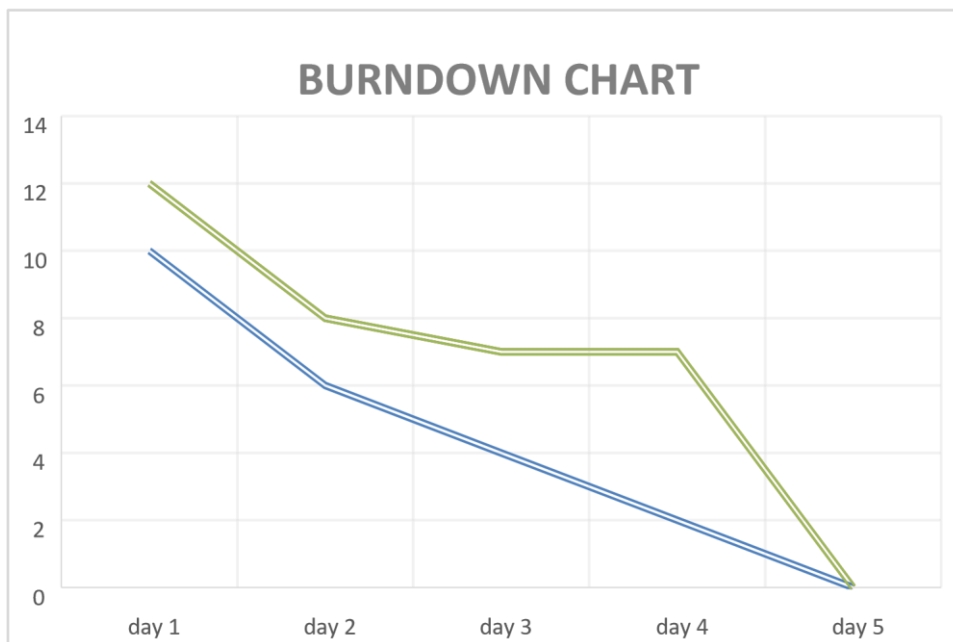
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

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

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

<https://www.visual-paradigm.com/scrum/scrum-burndown-chart/>
<https://www.atlassian.com/agile/tutorials/burndown-charts>



Reference:

<https://www.atlassian.com/agile/projectmanagement>
<https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software>
<https://www.atlassian.com/agile/tutorials/epics> <https://www.atlassian.com/agile/tutorials/sprints>
<https://www.atlassian.com/agile/project-management/estimation>
<https://www.atlassian.com/agile/tutorials/burndown-charts>

7. CODING AND SOLUTIONING (Explain the features added to the project along with code) :

CODE :

```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>
bool exhaust_fan_on =
false; bool sprinkler_on =
false; float temperature =
0; int gas_level = 0; int
flame = 0;
```

```

String flame_status = "";
String accident_status = "";
String sprinkler_status = ""; void
setup() {
  Serial.begin(99900);
}
void loop() {
  //setting a random seed
  srand(time(0)); //initial
  variable temperature =
  random(-20,125); gas_level =
  random(0,1000); int
  flamereading =
  random(200,1024);
  flame = map(flamereading,0,1024,0,2);
  //set a flame status
  Serial.print("Temperature : ");
  Serial.println(temperature);

  Serial.print("Gas_level : ");
  Serial.println(gas_level);

  Serial.print("Flame : ");
  Serial.println(flame);
  switch (flame)
  { case 0:
    flame_status = "No Fire";
    Serial.println("Flame Status : "+flame_status);
    break; case 1:
    flame_status = "Fire is Detected";
    Serial.println("Flame Status : "+flame_status);
    break; }
  //Gas Detection if(gas_level
  > 100){
  Serial.println("Gas Status : Gas leakage Detected");
  } else{
  exhaust_fan_on = false;
  Serial.println("Gas Status : No Gas leakage Detected"); }
  //send the sprinkler status if(flame){
  sprinkler_status = "Sprinkler ON";
  Serial.println("Sprinkler Status : "+sprinkler_status);
  } else{
  sprinkler_status = "Sprinkler OFF";
  Serial.println("Sprinkler Status : "+sprinkler_status);
  }
  //toggle the fan according to gas
  if(gas_level > 100){ exhaust_fan_on
  = true;
  Serial.println("Exhaust fan Status : Fan ON");
  } else{

```



```

exhaust_fan_on = false;
Serial.println("Exhaust fan Status : Fan OFF");
}
Serial.println("");
Serial.println("");
Serial.println(" -----#####-----
-----");
Serial.println("");
Serial.println(""); delay(3000);
}

```

7.1 Feature 1

In the above code, we have added a feature that when the fire is detected in the industry, then the Sprinkler will be ON.

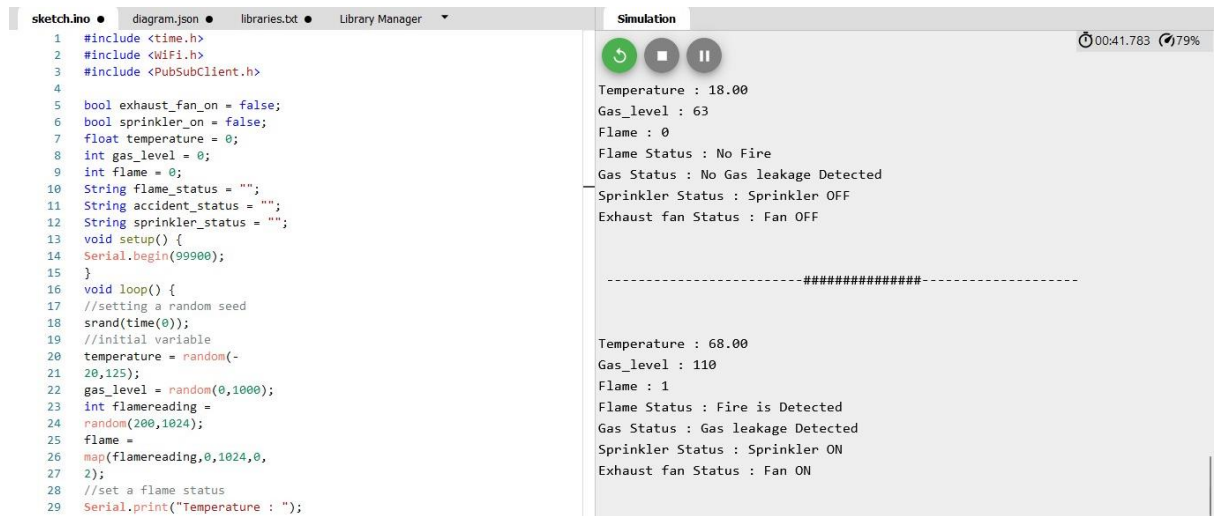
Solutioning of this feature:



7.2 Feature 2:

In the above code, we have added another feature is that when the “Gas leakage is detected” in the industry, then the Exhaust fan will be ON.

Solutioning of this feature:



8. TESTING :

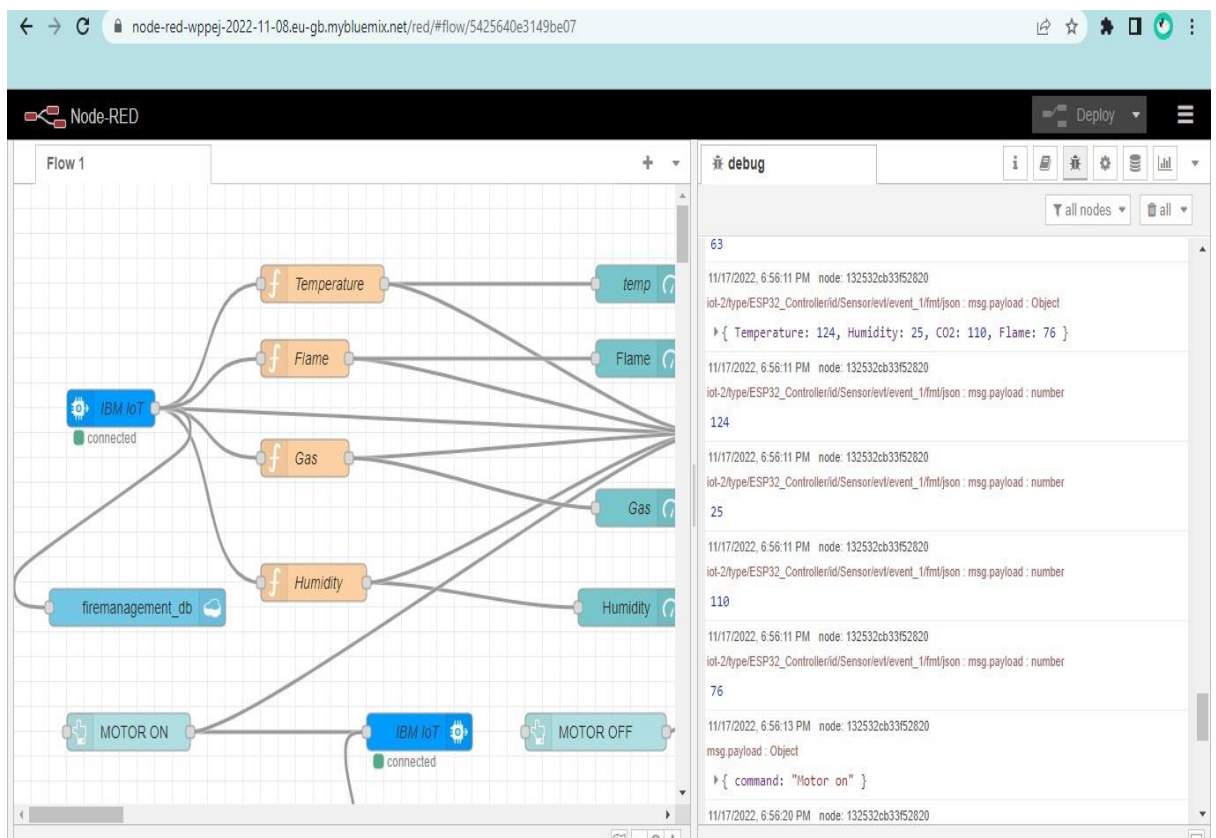
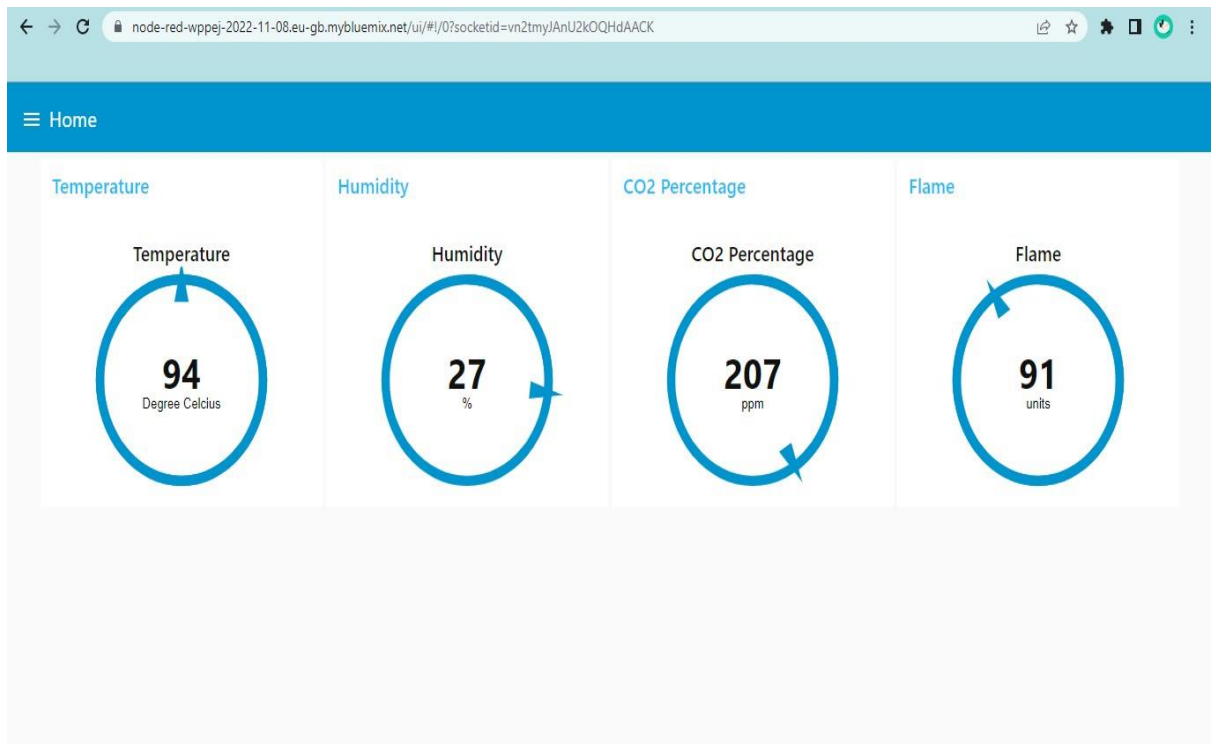
8.1 Test Cases

s.no	Inputs	Outputs	Results
1	Temperature : 113.00 Gas_level : 945 Flame : 0	Flame Status : No Fire Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler OFF Exhaust fan Status : Fan ON	Passed
2	Temperature : 122.00 Gas_level : 41 Flame : 0	Flame Status : No Fire Gas Status : No Gas leakage Detected Sprinkler Status : Sprinkler OFF Exhaust fan Status : Fan OFF	Passed
3	Temperature : 39.00 Gas_level : 639 Flame : 1	Flame Status : Fire is Detected Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler ON Exhaust fan Status : Fan ON	Passed

4	Temperature : 40.00 Gas_level : 761 Flame : 0	Flame Status : No Fire Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler OFF Exhaust fan Status : Fan ON	Passed
5	Temperature : 67.00 Gas_level : 21 Flame : 1	Flame Status : Fire is Detected Gas Status : No Gas leakage Detected Sprinkler Status : Sprinkler ON Exhaust fan Status : Fan OFF	Passed
6	Temperature : 107.00 Gas_level : 592 Flame : 1	Flame Status : Fire is Detected Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler ON	Passed
		Exhaust fan Status : Fan ON	
7	Temperature : 8.00 Gas_level : 990 Flame : 0	Flame Status : No Fire Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler OFF Exhaust fan Status : Fan ON	Passed
8	Temperature : 84.00 Gas_level : 532 Flame : 0	Flame Status : No Fire Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler OFF Exhaust fan Status : Fan ON	Passed

9	Temperature : 123.00 Gas_level : 272 Flame : 1	Flame Status : Fire is Detected Gas Status : Gas leakage Detected Sprinkler Status : Sprinkler ON Exhaust fan Status : Fan ON	Passed
10	Temperature : 163.00 Gas_level : 94 Flame : 1	Flame Status : Fire is Detected Gas Status : No Gas leakage Detected Sprinkler Status : Sprinkler ON Exhaust fan Status : Fan OFF	Passed

9. RESULTS :



10. ADVANTAGES :

The main advantage and function of a fire alarm system is to ensure ultimate safety. The system warn and keep people safe and reduce the amount of destruction to a building. Proactively gave protection to secure business processes and valuable goods from fires. Protect from multiple hazards with just one system.

DISADVANTAGES

The system is essentially useless if the batteries aren't charged, since it won't work properly. It must be borne in mind that a major disadvantage of heat detectors is the fact they rely on heat to actuate. For large buildings, they can be expensive to install because of the extensive amounts of wire that are necessary to accurately monitor initiating devices.

11. CONCLUSION

It is important to implement the fire protection system in places like industries, colleges and so on as a part of building safety as well as prevention of life loss. This system recommends the use of fire alarms system, sprinkler system and smoke control system, these are helpful in notify the actions like fire and smoke conditions. It can also helps to slow down the growth of fire. The followence of guidelines make the place safe to work and in case of fire the building will include protection that will results in minimum loss.

12. FUTURE SCOPE

There haven't really been any momentous industry standards which will keep up with technological advances a lot faster, those responsible for benefit fire detection manufacturers and the marketplace. Interconnection between multiple system has been implemented in upcoming period. Different Gas Sensor can be used to identify the accuracy of the gas while fire occurs. Making the Devices more and more capable, and their application, so that they all are evolving to connect to the internet and cooperate with other devices. The future will be with the detectors. Detector will be more of a sensor, with the detection capability for the products of combustion, such as carbon monoxide, carbon dioxide, sulfur dioxide, nitrogen oxides in addition to heat and and other subordinates matters.

13. APPENDIX

Source Code :

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "0bm892",
        "typeId": "NodeMcu",
        "deviceId": "1234"
    },
    "auth": {
        "token": "12345678"
    }
}

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:
    temperature=random.randint(-20,125)
    humidity=random.randint(0,100)
    gas_level=random.randint(0,500)
    flame=random.randint(0,200)

    if (flame>80):
        print("Sprinkler ON")
    else:
        print("Sprinkler OFF")

    if (gas_level>100):
        print("Exhaust Fan ON")
    else:
        print("Exhaust Fan OFF")

    myData={'temperature':temperature, 'humidity':humidity, 'gas_level':
gas_level, 'flame':flame}
    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()
```

OUTPUT :

The screenshot displays a web application interface for managing devices. The top navigation bar includes links for 'Browse', 'Action', 'Device Types', and 'Interfaces', along with an 'Add Device' button. The main content area is divided into tabs: 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, showing a table of events. The table has four columns: 'Event', 'Value', 'Format', and 'Last Received'. It lists five 'status' events with JSON data and a 'a few seconds ago' timestamp. A status box at the bottom right indicates '1 Simulation running'.

Event	Value	Format	Last Received
status	{"temperature":87,"humidity":9,"gas_level":418,...	json	a few seconds ago
status	{"temperature":-5,"humidity":73,"gas_level":335...	json	a few seconds ago
status	{"temperature":72,"humidity":68,"gas_level":42...	json	a few seconds ago
status	{"temperature":87,"humidity":63,"gas_level":11...	json	a few seconds ago
status	{"temperature":44,"humidity":14,"gas_level":49...	json	a few seconds ago

1 Simulation running

The screenshot displays the PyCharm IDE interface. The left sidebar shows the project structure with 'pythonProject1' and its subdirectories 'venv' and 'library root'. The main editor window shows a file named 'main.py' with the following code:

```
1 #IBM Watson IoT Platform
2 #pip install wiotp-sdk
3 import wiotp.sdk.device
4 import time
5 import random
6 myConfig = {
    # ... (code is partially obscured)
    while True
```

The bottom panel shows the 'Run' output, which includes the command used to execute the script and its output:

```
"C:\Users\Praveen Kumar\PycharmProjects\pythonProject1\venv\Scripts\python.exe" "C:/Users/Praveen Kumar/PycharmProjects/pythonProject1/main.py"
Sprinkler ON
Exhaust Fan OFF
Published data Successfully: %s {'temperature': -18, 'humidity': 20, 'gas_level': 32, 'flame': 131}
2022-11-18 14:18:34,019 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:0bm892:NodeMcu:1234
Sprinkler OFF
Exhaust Fan ON
Published data Successfully: %s {'temperature': 69, 'humidity': 48, 'gas_level': 448, 'flame': 35}
Sprinkler ON
Exhaust Fan ON
Published data Successfully: %s {'temperature': 28, 'humidity': 32, 'gas_level': 377, 'flame': 97}
Sprinkler OFF
Exhaust Fan ON
Published data Successfully: %s {'temperature': 113, 'humidity': 25, 'gas_level': 278, 'flame': 20}
Sprinkler OFF
Exhaust Fan OFF
Published data Successfully: %s {'temperature': 113, 'humidity': 42, 'gas_level': 29, 'flame': 2}
```

The bottom status bar indicates the file encoding is UTF-8, the font size is 14 spaces, and the Python version is 3.6 (pythonProject1).

Github Link :-

<https://github.com/IBM-EPBL/IBM-Project-26661-1660032629>