

HAZARDOUS AREA MONITORING FOR INDUSTRIAL PLANT POWERED BY IOT

TEAM ID-PNT2022TMID14654

PROJECT SUBMITTED BY:

MANOJKUMAR M

KARTHICK S

KARTHIKEYAN M

MANOBHARATH M

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 6.3 Reports from JIRA

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

7.1 Feature 1 7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code GitHub & Project Demo Link

PROJECT REPORT

1. INTRODUCTION

1.1 Project Overview

Industrial hazard is the danger or chance of accidental injury to the worker while engaged at his accustomed work in the plant. In industrial areas fire accidents can prevent by fire detection using temperature, gas and flame sensors with automatic water sprinkler, harmful or toxic gas leakages can identify by monitoring harmful gases and intimate alert message to safety control board of industry, Machine overheating can reduce by cooling machine with compressor, control the humidity and temperature instantaneously in order to have stable, controllable atmospheric conditions. Identifying industrial hazards allows employers to protect their worker from accidents, injuries, and fatalities.

1.2 Purpose

Safety stands as the foremost priority in most of today's technology-aided industrial processes. By using continuous monitoring and detecting of hazardous signs, major problems will be prevented at the initial stages. Thus, minor problems can prevent easily in order to prevent major problems. For preventing

the kind of accidents resulting in the seizing of work and loss of the production. By controlling the hazard to the minimal level, accidents in the industry can be avoided.

2. LITERATURE SURVEY

2.1 Existing Problem

2.1.1 Problem-1

A Smart Fire Detection System using IOT Technology With Automatic Water Sprinkler.

Hamood Alqourabah, Amgad Muneer, Suliman Mohammed Fati et al.(2020) proposed a “A Smart Fire Detection System using IOT Technology With Automatic Water Sprinkler”. This paper has been developed to solve the slow response issue of fire accidents.

Temperature, gas, and flame sensors are inputs. Output like LED and Buzzer indicate a fire. Water system launched with 12V water pump powered by Arduino and control by a 5v relay. The sprinkler head is the outer of the water output. An ultrasonic sensor is used to measure the tank level and inform the need for refilling. By analyzing the result, we can easily identify the sense of fire accident.

2.1.2 Problem-2

Gas monitoring and power cut-off system for underground mines.

A. Kumar, H. Kumar, V. N. Pandey, D. K. P. Singh and S. K.Chaulya et al. (2012) proposed a "Gas monitoring and power cut-off system for underground mines".This mechanism detect several hazardous gases from burning smoke, fumes (soldering) in the field of mining domain. This work includes designing and implementation of a system that continuously monitors the concentration of methane (CH₄) and carbon-monoxide (CO) gas in the underground mines, which automatically cuts off the power supply of the particular zone in an underground mine, when the concentration of CH₄ exceeds more than the permissible limit decided by the user. By analyzing this paper, The proposed gas monitoring system place a vital role in detecting the gases in mines.

2.1.3 Problem-3

Monitoring temperature, humidity and Controlling system in industrial fixed room storage based on IoT.

M.T.A.Seman, M.N.Abdullah, M.K.Ishak et al.(2020) proposed a “MONITORING TEMPERATURE, HUMIDITY AND CONTROLLING SYSTEM IN INDUSTRIAL FIXED ROOM STORAGE BASED ON

IOT”. this paper was proposed new features that can detect, notify, record and control the humidity and temperature instantaneously in order to have stable, controllable atmospheric conditions. The temperature and humidity sensor is the input used to get the surrounding temperature and humidity inside the room. The NodeMCU works as the brain of the system that

receives data from DHT- 11 sensor and upload them to the database. The data and information from DHT-11 sensor will be analysed graphically on Blynk platform using NodeMCU. Humidifier triggered when the reading of temperature and humidity is less than 20% and more than 80%.

The system is capable of obtaining the temperature and humidity of a space, thus providing a real time temperature and humidity monitoring either using an application in a smartphone via WIFI over IoT, a LCD display embedded in the system or a computer interface.

2.1.4 Problem-4

Toxic gas detection and monitoring utilizing internet of things.

Dr. Chalasani Srinivas, Mohan Kumar.Ch et al. (2017) proposed a “Toxic gas detection and monitoring utilizing internet of things”. This project intended to avoid industrial accidents and to monitor harmful gases and to intimate alert message to safety control board of industry using ArduinoUno R3 and internet of things. Ardunio Uno R3 board is used as central microcontroller which is connected with sensor which can continuously monitor respective environment parameters. An alarm is produced instantly if the level of the gases goes above the normal level means indication through the internet specific receiver section. Data received by sensor is stored in internet which can be used for further processing and it can be analyzed for improving safety regulations. This model can be future extended for providing better living environment for people in and around industries with a pollution controlled environment.

2.2 References

2.2.1 Problem-1

A Smart Fire Detection System using IOT Technology With Automatic Water Sprinkler.

Hamood Alqourabah, Amgad Muneer, Suliman Mohammed Fati et al, 2020.

2.2.2 Problem-2

Gas monitoring and power cut-off system for underground mines.

A. Kumar, H. Kumar, V. N. Pandey, D. K. P. Singh and S. K.Chaulya et al, 2012.

2.2.3 Problem-3

Monitoring temperature, humidity and Controlling system in industrial fixed room storage based on IoT.

M.T.A.Seman, M.N.Abdulllah, M.K.Ishak et al, 2020.

2.2.4 Problem-4

Toxic gas detection and monitoring utilizing internet of things.

Dr. Chalasani Srinivas, Mohan Kumar.Ch et al, 2017.

2.3 Problem Statement Definition

2.3.1 Person1 (Industrial Worker)

I'm trying to manage to control catastrophic condition, but it causes disorders in human health and cause to death. Because of leakages in toxic gases which makes me feel that don't have safety while working in an industries.

2.3.2 Person2 (Industrial Worker)

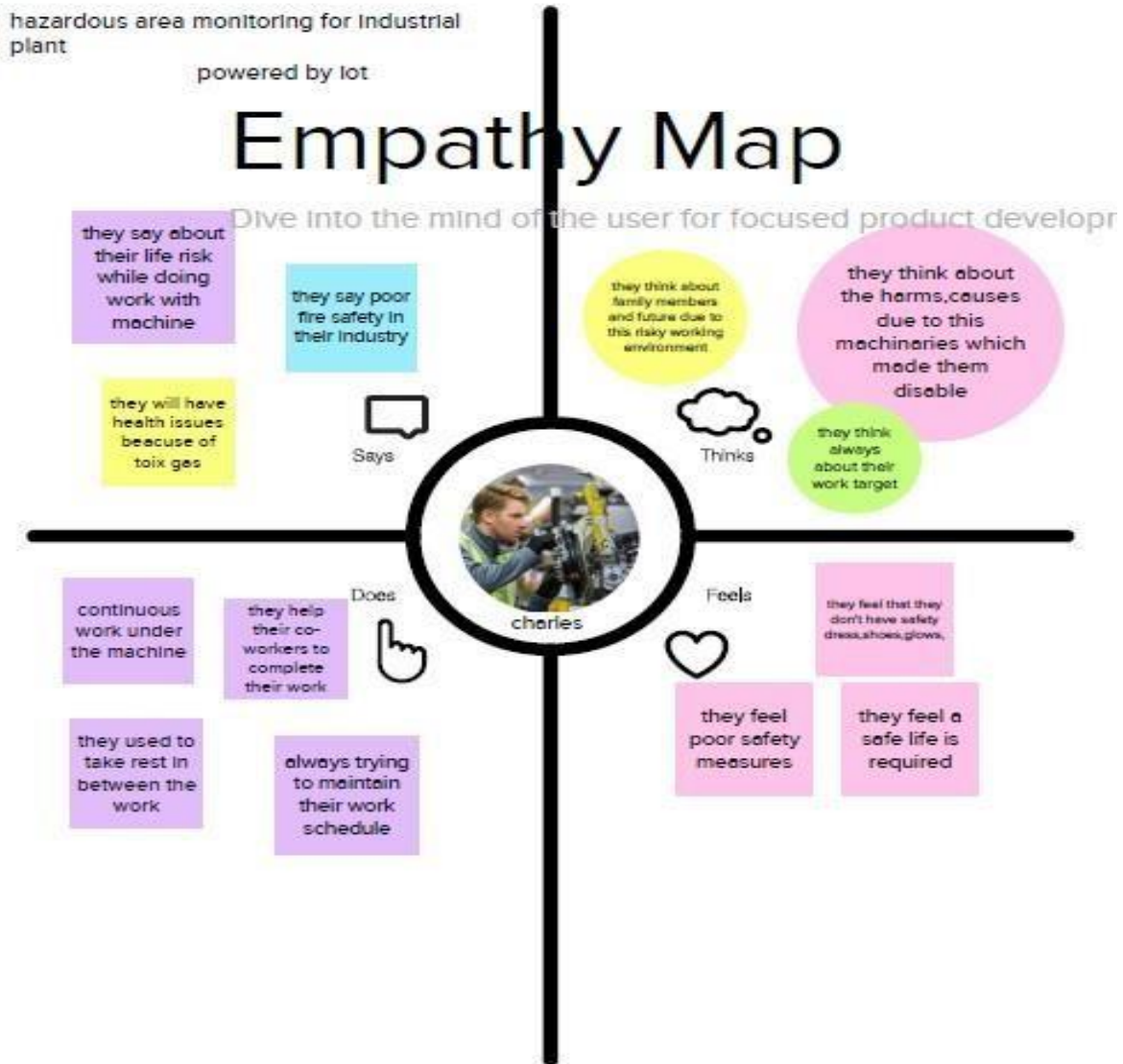
I'm trying to control the initial stage of fire hazard, but it causes extreme danger for humans and machines in industries. Because of Defective in products, Flammable Liquids, Gas explosions.

2.3.3 Person3 (Industrial Worker)


I'm trying to reduce the heat by using smart compressor, but it creates more faults in machinaries. Because of improper maintainence which make me feel storage of production.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy map canvas



3.2 Ideation & Brainstorming



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.

[Open article](#) ➔

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

Industrial hazard in the danger or chance of accidental injury to the worker with in the plant. Identifying industrial hazards allows employers to protect their workers from accidents, injuries and fatalities. Safety in simple terms means freedom from the occurrence of risk on injury or loss. Industrial safety refers to the protection of workers from the danger of industrial accidents when they work in the factories.

Key rules of brainstorming

To run a smooth and productive session

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

2

Brainstorm

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

⌚ 10 minutes

TIP
You can select a sticky note and tap the pencil button to start drawing!

Person 1

- detection of overheating in machines
- smart switch board control
- using a lone worker alarm to protect solo employees
- industrial iot bot

Person 2

- Emergency shutdown system
- fluid level indicators and controller
- Fire detection alarm and automatic water flushing
- End-effectors and control logic for robot

Person 3

- Toxic gas detection in industrial areas
- Temperature monitoring at tanks in industries
- Detecting minor faults in machines
- Continuous monitoring of valves at tanks in industries

Person 4

- Human entry detection in dangerous areas
- Smart glove that control remote devices using the sensor and internet
- Feed the heap with Arduino and camera
- Industrial laser detector for supplying materials

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

⌚ 20 minutes

1. Detection of toxic gases in industry environmental places

2. Fire detection alarm and automatic water flushing system.

3. Human entry detection in dangerous area

4. detection of overheating in machines

5. Smart switch board control.

6. Temperature monitoring at tanks in industries

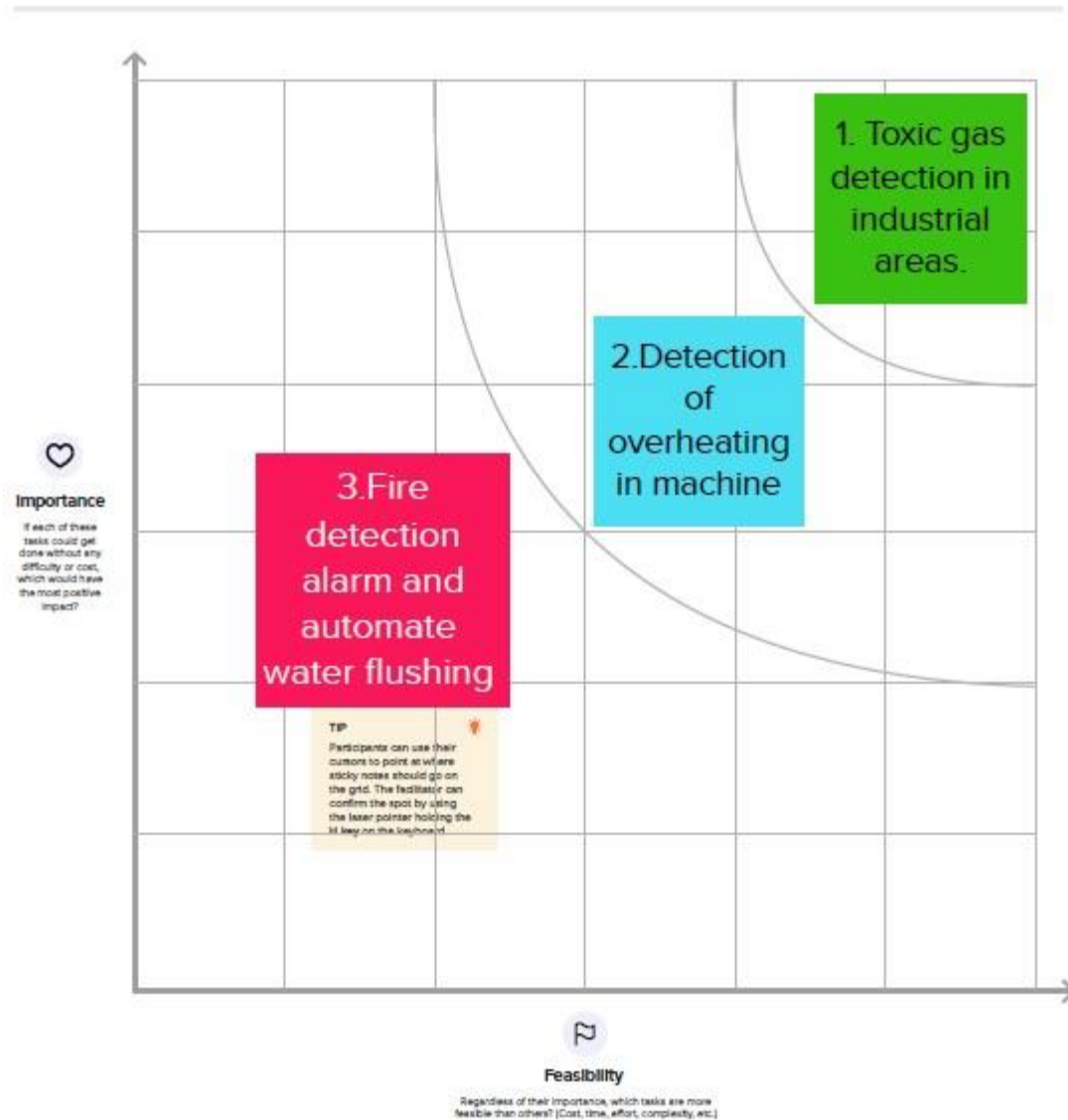
TIP
Add customisable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as they arise within your mind.

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

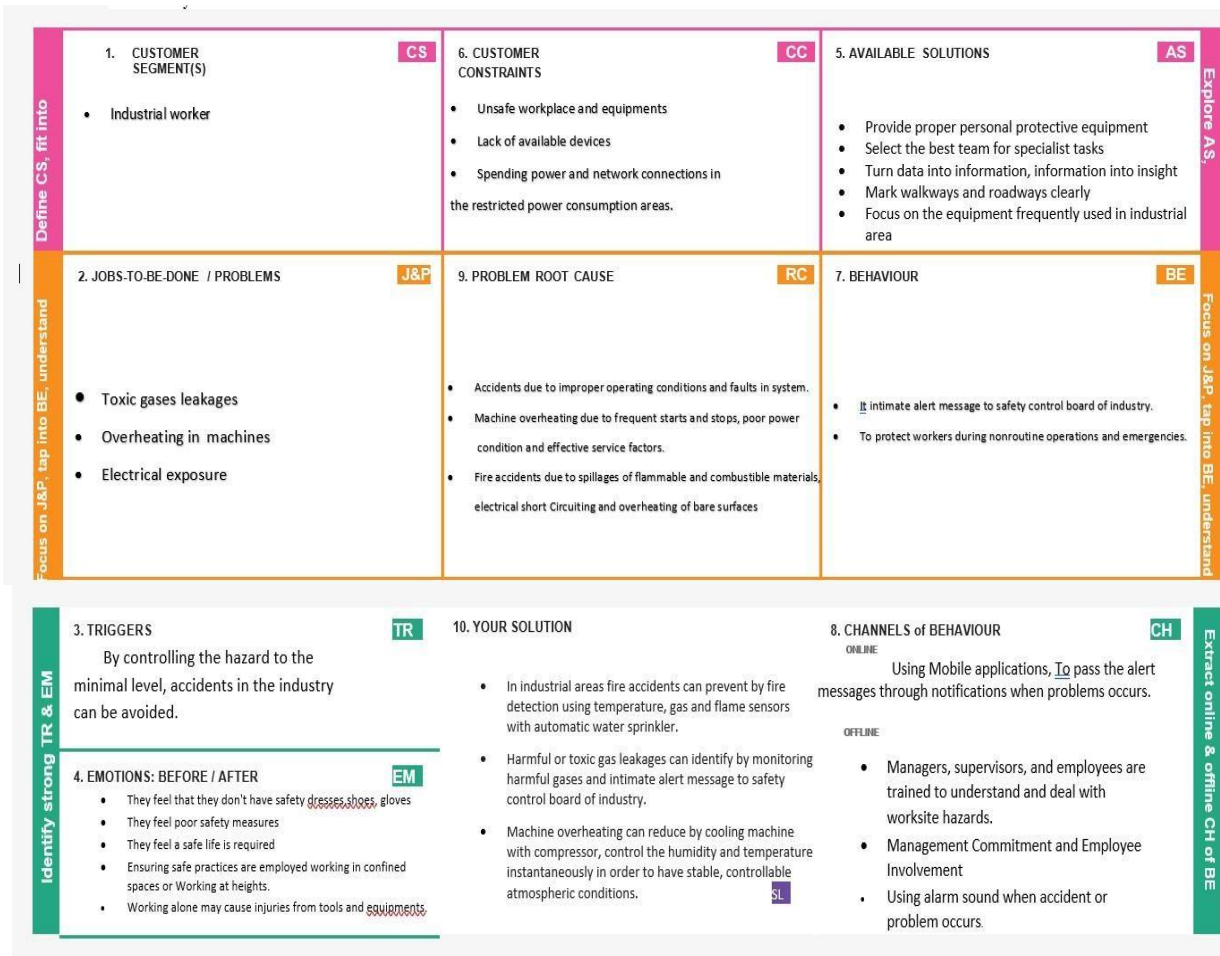


3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Industrial hazard is the danger or chance of accidental injury to the worker while engaged at his accustomed work in the plant. The most occurring problems in industries are fire accidents, harmful gas leakages, overheating in machines, electrical explosions, accidents due to improper operating conditions and faults in systems. Identifying industrial hazards allows employers to protect their worker from accidents, injuries, and fatalities.
2.	Idea / Solution description	In industrial areas fire accidents can prevent by fire detection using temperature, gas and flame sensors with automatic water sprinkler, Harmful or toxic gas leakages can identify by monitoring harmful gases and intimate alert message to safety control board of industry, Machine overheating can reduce by cooling machine with compressor, control the humidity and temperature instantaneously in order to have stable, controllable atmospheric conditions.
3.	Novelty / Uniqueness	Safety stands as the foremost priority in most of today's technology-aided industrial processes. By using continuous monitoring and detecting of hazardous signs, major problems will be prevented at the initial stages. Thus, minor problems can prevent easily in order to prevent major problems. For preventing the kind of accidents resulting in the seizing of work and loss of the production. By controlling the hazard to the minimal level, accidents in the industry can be avoided.
4.	Social Impact / Customer Satisfaction	With the help of employee's safety, a working environment having greater motivation and satisfaction of the manpower can be created. Broadly, having a safer working environment for performing the operations is expected by a worker from their employer. There are some specific advantages that can be obtained from an accident-free plant. Huge cost savings, high legal and moral grounds, improved productivity are some of the important benefits. Greater job satisfaction and higher motivation can be obtained by the workers if their expectations of safety from hazards are fulfilled by the industrial working environment.

5.	Business Model (Revenue Model)	To meet industrialists explaining about the problem-solving solutions for the hazards in industries by proposing the ideas that using IoT technology. For the further development of problem solutions into products collecting funds from the industries as per the need cost. There are some specific advantages that can be obtained from an accident-free plant. Huge cost savings, high legal and moral grounds, improved productivity are some of the important benefits.
6.	Scalability of the Solution	By taken the investigations based on the solutions found there will be more possibilities to make the ideas in to applicable. The ideas proposed are possible by locating IoT sensors and sensing hazards by continuously monitoring the stability conditions in industrial environment. This mechanism is feasible because it is easy to setup, user friendly, wireless detector, predicts the risk, in beforehand, affordable etc., in today's technology application of these kind of sensor are very successful, trustworthy and easily accessed in many fields of work.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	Detection of Hazard	Reporting the hazard to control node. Control the humidity and temperature instantaneously in order to have stable environment.
FR-2	Air monitoring	Identifying harmful gas leakages by Air monitoring sensors and transmitting message to user (industrial workers).
FR-3	Emergency Shutdown system	Shut downing the entire system in case of peak situations for example: Machine over heating with colling compressor.
FR-4	Fire and smoke detection	Fire detection using temperature, gas and flame sensors with automatic water sprinkler.
FR-5	End Users (Industrial workers)	Using cloud Service (Audio and visual notifications).

--	--	--

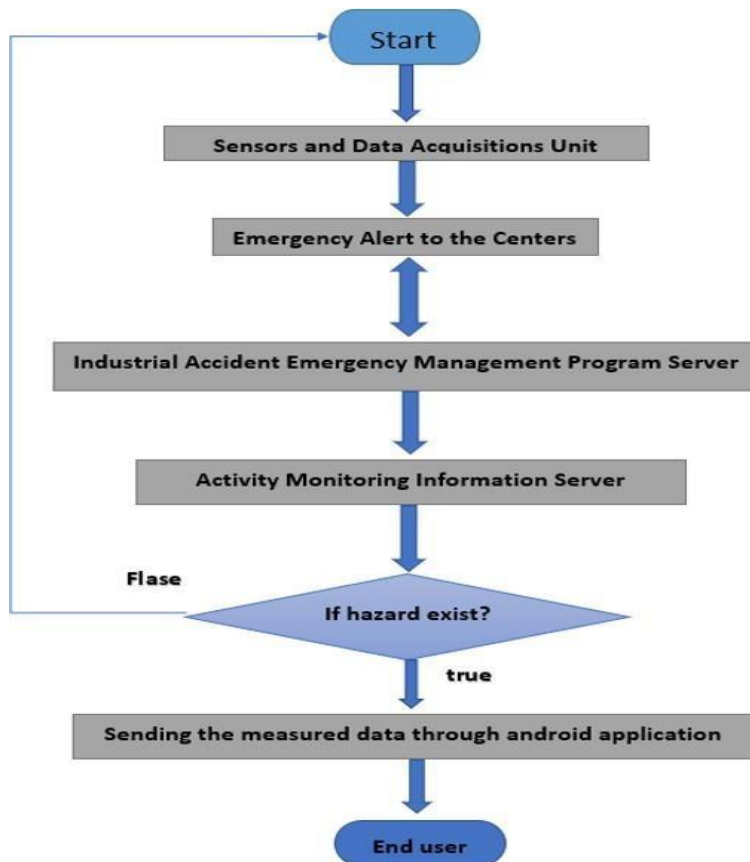
4.2 Non-Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Huge cost savings, Greater job satisfaction and higher motivation can be obtained by the workers if their expectations of safety from hazards are fulfilled by the industrial working environment.
NFR-2	Security	By using continuous monitoring and detecting of hazardous signs, major problems will be prevented at the initial stages.
NFR-3	Reliability	High speed, not cost effective and controlling the hazard at minimal level.

NFR-4	Performance	When problem occurs, the investigations based on the problems found there will be more possibilities to make the ideas in to applicable.
NFR-5	Availability	This mechanism is feasible and it is easy to set up. Trust worthy and easy to access in many fields.
NFR-6	Scalability	This mechanism is feasible because it is easy to set up, user friendly, wireless detector, predicts the risk, in beforehand, affordable

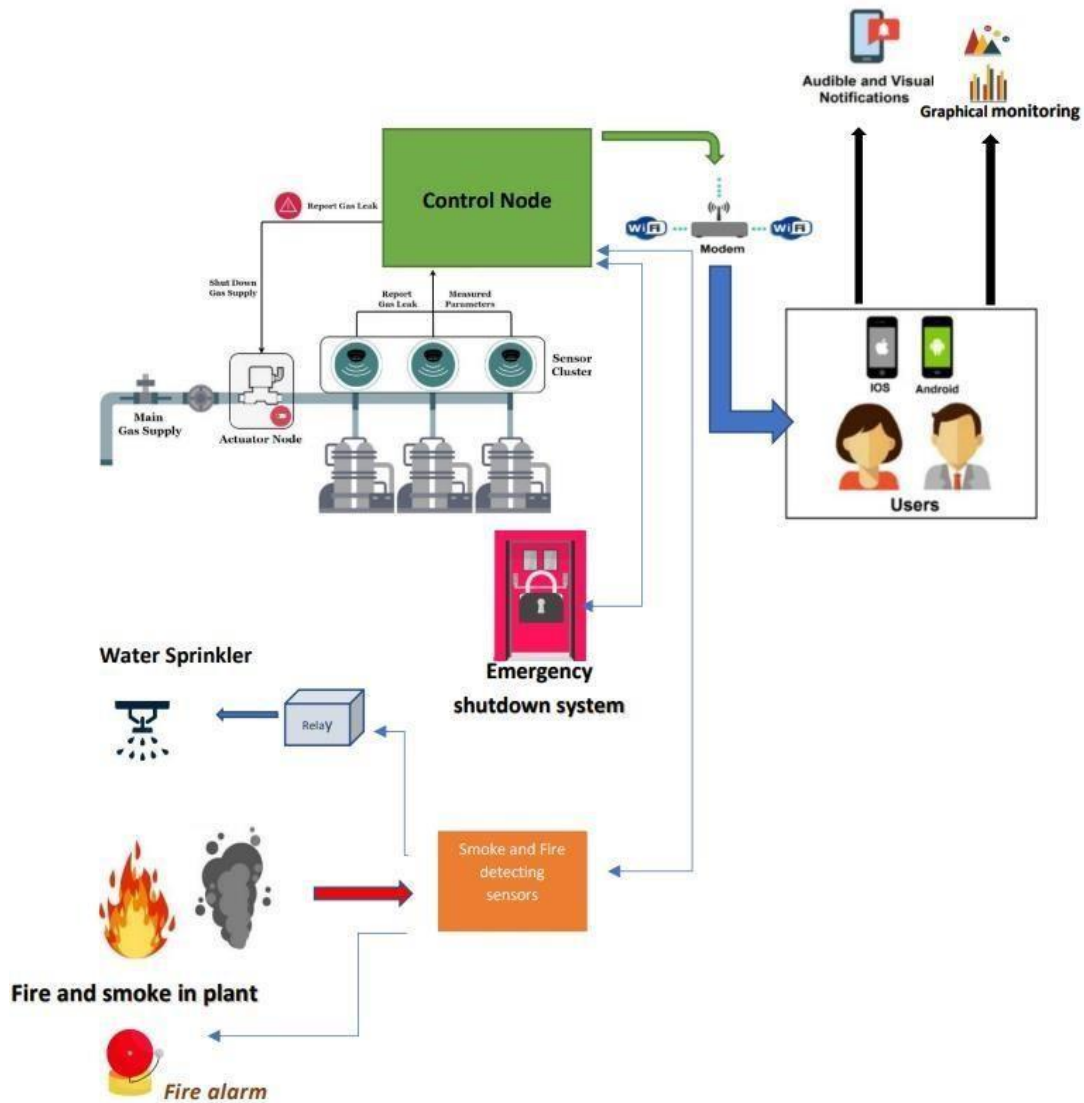
5. PROJECT DESIGN

5.1 Data Flow Diagrams

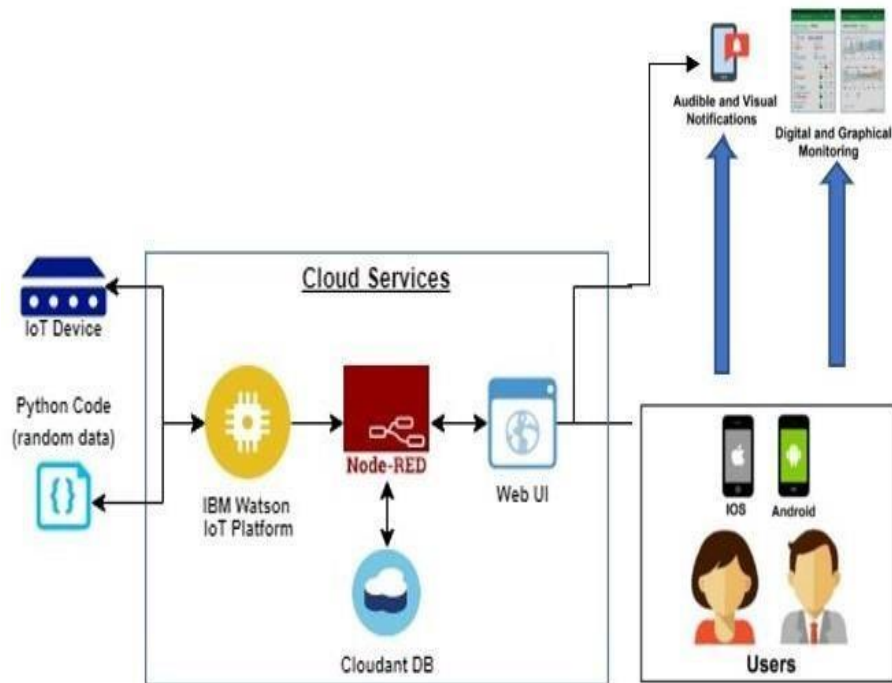


5.2 Solution & Technical Architecture

5.2.1 Solution Architecture



5.2.2 Technology Architecture



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Industrial worker)	Detection of hazard	USN-1	As a worker, I will receive information from control node when hazard reported	I can aware myself by receiving information from control node	High	Sprint-1

		USN-2	As a worker, I can monitor the humidity and temperature to have stable environment	I can notify the environmental changes via mobile application	Medium	Sprint-1
	Air monitoring	USN-3	As a worker, I can identify harmful gas leakages through mobile notifications	I can identify the harmful gas leakages in the field	High	Sprint-2
	Emergency shutdown system	USN-4	As a worker, I can measure the machine condition and shut down the entire system in the case of peak condition	I can shutdown the system in emergency situation	Medium	Sprint-3
	Fire and smoke detection	USN-5	As a worker, I can detect fire using temperature, smoke and flame sensors with automatic water sprinkler	I can detect fire by using sensors with water sprinkler	High	Sprint-4

End user	Alerting through message	USN-6	As a worker, I can receive the message in the form of audio, visual notification and graphical notification	I can detect the hazard and get notification	Low	Sprint-5
----------	--------------------------	-------	---	--	-----	----------

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint1	Detection of hazard	USN-1	As a worker, I will receive information from control node when hazard reported.	4	High	Manojkumar M Karthick S Karthikeyan M Manobharath M
		USN-2	As a worker, I can monitor the humidity and temperature to have stable environment	3	Medium	Manojkumar M Karthick S Karthikeyan M Manobharath M
Sprint2	Air monitoring	USN-3	As a worker, I can identify harmful gas leakages through mobile notifications	4	High	Manojkumar M Karthick S Karthikeyan M Manobharath M

Sprint3	Emergency shutdown system	USN-4	As a worker, I can measure the machine condition and shut down the entire system in the case of peak condition	3	Medium	Manojkumar M Karthick S Karthikeyan M Manobharath M
---------	---------------------------	-------	--	---	--------	--

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Fire and smoke detection	USN-5	As a worker, I can detect fire using temperature, smoke and flame sensors with automatic water sprinkler	4	High	Manojkumar M Karthick S Karthikeyan M Manobharath M
	Alerting through message	USN-6	As a worker, I can receive the message in the form of audio, visual notification and graphical notification.	3	Low	Manojkumar M Karthick S Karthikeyan M Manobharath M

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	3 Days	05 Nov 2022	07 Nov 2022	20	07 Nov 2022
Sprint-2	20	3 Days	08 Nov 2022	10 Nov 2022	20	10 Nov 2022
Sprint-3	20	3 Days	11 Nov 2022	13 Nov 2022	20	13 Nov 2022
Sprint-4	20	3 Days	14 Nov 2022	17 Nov 2022	20	17 Nov 2022

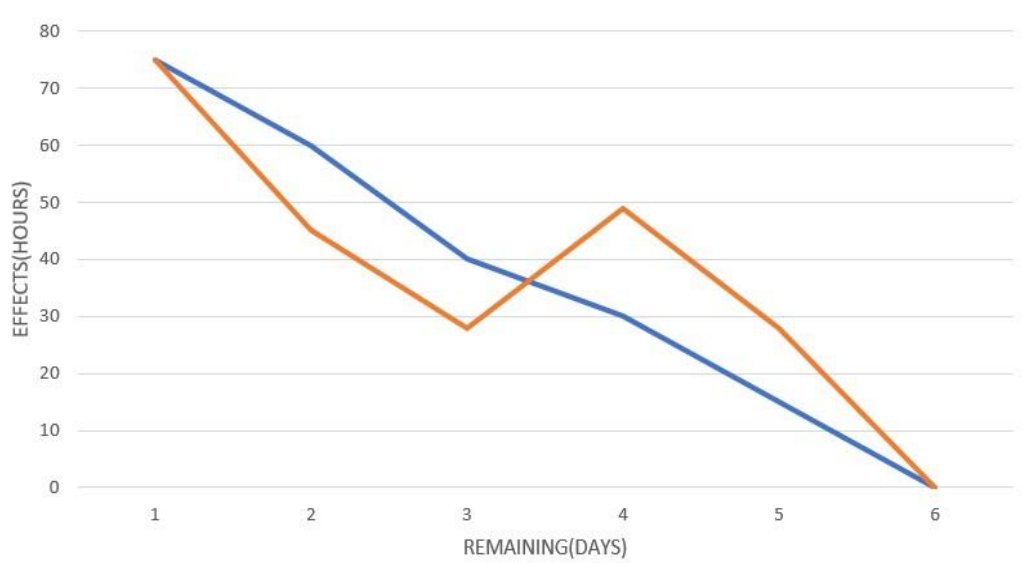
Velocity:

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)

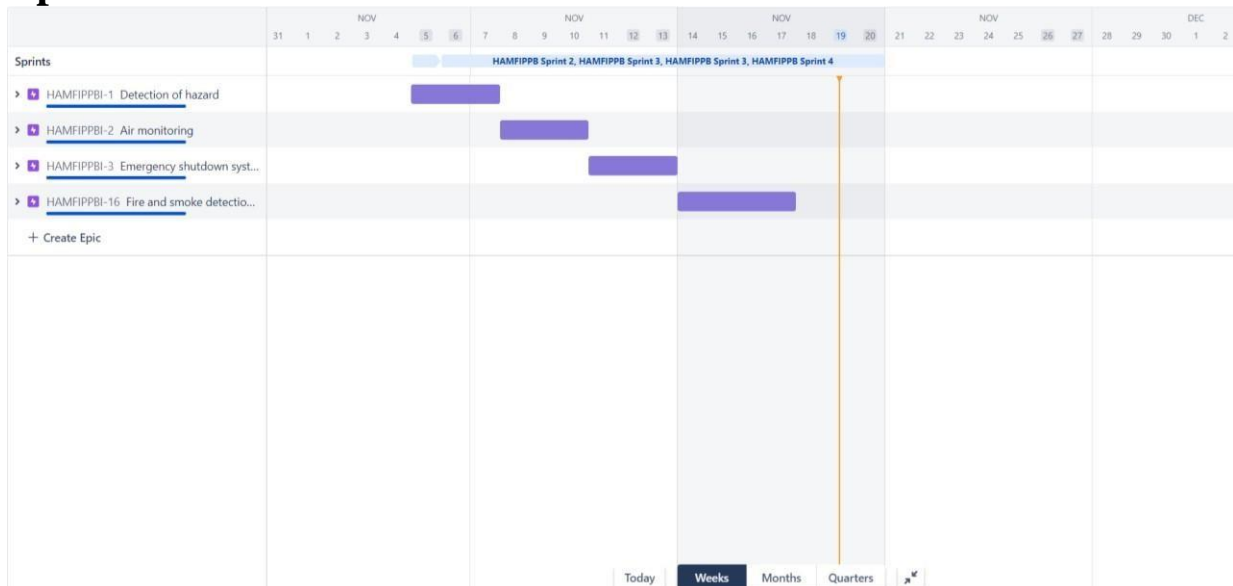
$$AV = \text{Sprint Duration} / \text{Velocity}$$

$$= 20 / 3 = 6.66$$

Burndown chart:



6.3 Reports from JIRA Road map:



Backlog:

Jira Software

Your work

Projects

Filters

Dashboards

People

Apps

Create

Q Search

Hazardous Area Monit...

Software project

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

Does your team need more from Jira? Get a free trial of our Standard plan.

X

Projects / Hazardous Area Monitoring for Industrial Plant Powered by IoT

Backlog

...

Q

KD Y HB A

+

Epic

▼

Insights

▼ HAMFIPB Sprint 1 5 Nov – 7 Nov (1 issue)

0 0 0 Complete sprint ...

To Detect the hazards using sensing devices

HAMFIPBI-5 As a worker, I will receive information from control node when hazard reported, As a worker, I can monitor the hum...

DETECTION OF HAZARD

IN REVIEW

Y

+ Create issue

▼ HAMFIPB Sprint 2 8 Nov – 10 Nov (1 issue)

0 0 0 Complete sprint ...

To monitor the air in environment

HAMFIPBI-6 As a worker, I can identify harmful gas leakages through mobile notifications

AIR MONITORING

IN REVIEW

+ Create issue

Does your team need more from Jira? Get a free trial of our Standard plan.

X

Projects / Hazardous Area Monitoring for Industrial Plant Powered by IoT

Backlog

...

Q

KD Y HB A

+

Epic

▼

Insights

▼ HAMFIPB Sprint 3 11 Nov – 13 Nov (1 issue)

0 1 0 Complete sprint ...

To shutdown the entire system in emergency situations

HAMFIPBI-14 As a worker, I can measure the machine condition and shut down the entire system in the case of peak co...

EMERGENCY SHUTDOWN SYSTEM

IN REVIEW

+ Create issue

▼ HAMFIPB Sprint 4 14 Nov – 17 Nov (1 issue)

0 1 0 Complete sprint ...

To detect fire and smoke and to alert through messages or notifications

HAMFIPBI-17 As a worker, I can detect fire using temperature, smoke and flame sensors with automatic water sprinkl...

FIRE AND SMOKE DETECTION, ALE...

IN REVIEW

KD

+ Create issue

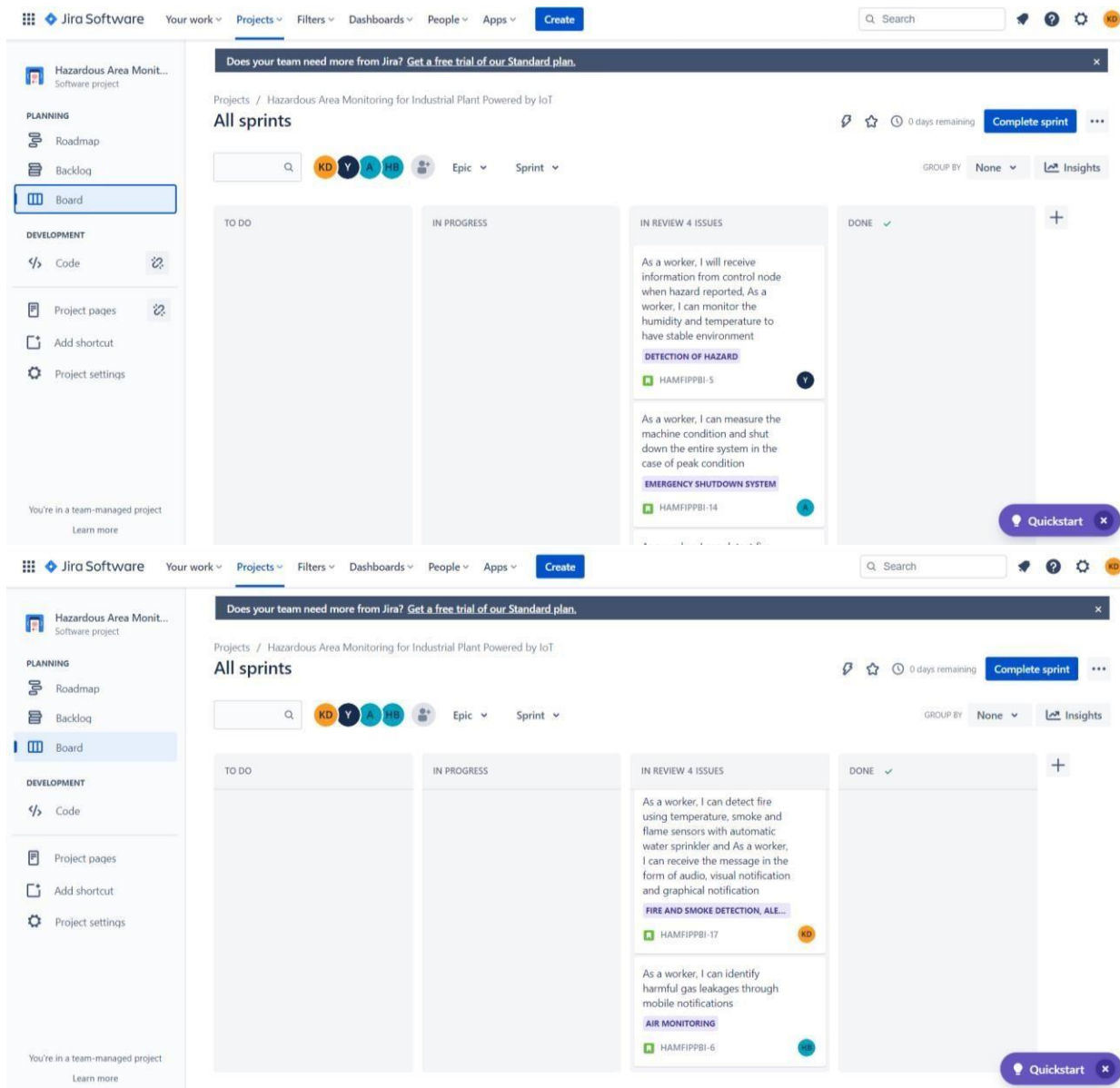
▼ Backlog (0 issues)

Quickstart

X

0 1 0 Create sprint

Board:



7. CODING & SOLUTIONING

7.1 Feature-1

7.1.1 Python script to connect ibm watson IoT platform to get data from Cloud.

```
import time
import sys
import ibmiotf.application
```

```

import ibmiotf.device import
random
#Provide your IBM Watson Device Credentials
organization = "f7gdph"
deviceType    =    "Abcd"
deviceId      =    "204527"
authMethod    =    "token"
authToken     = "8778766443"
# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command'] if status=="shutdown": print ("machine
                                                off") else :
        print ("machine
on") #print(cmd)
try:
    deviceOptions = {"org": organization, "type": deviceType, "id":
deviceId, "auth-method": authMethod, "auth-token": authToken}
    deviceCli =
ibmiotf.device.Client(deviceOptions)
    #..... except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud
as an event of type "greeting" 10 times deviceCli.connect()

while True:
    #Get Sensor Data from
    DHT11
    temp=random.randint(0,100)
    Humid=random.randint(0,100)

```

```

data = { 'temp' : temp, 'Humid':
Humid } #print data def
myOnPublishCallback():
    print ("Published Temperature = %s C" % temp, "Humidity =
%s %%" % Humid, "to IBM Watson") success =
deviceCli.publishEvent("IoTSensor", "json", data,
qos=0, on_publish=myOnPublishCallback) if not
success: print("Not connected to
IoTF") time.sleep(1)
deviceCli.commandCallback =
myCommandCallback # Disconnect the device and application from the
cloud deviceCli.disconnect()

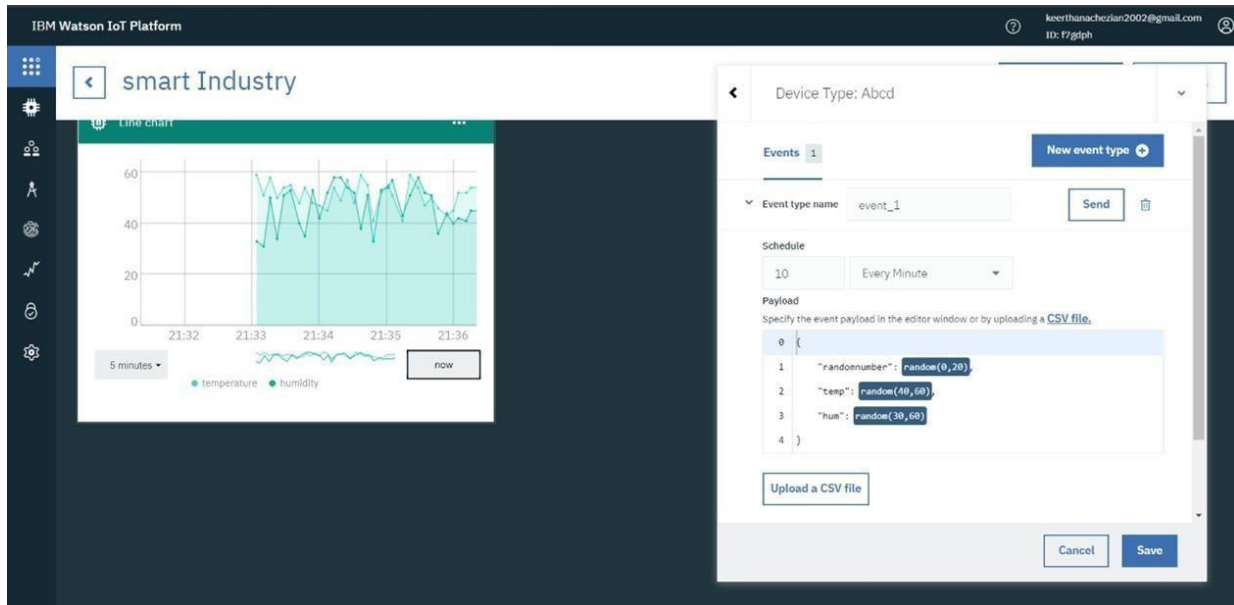
```

7.1.2 IBM Simulation

The screenshot shows the IBM Watson IoT Platform interface. The main panel displays a table of recent events for device 204527, which is disconnected. The simulation panel on the right shows 1/50 simulations running for device type 'Abcd'. The simulation panel includes a table of events and a status bar at the bottom indicating 221 events sent and 9.62 KB sent.

Event	Value	Format	Last Received
event_1	{"randomnumber":15,"temp":47,"hum":53}	json	a few seconds ago
event_1	{"randomnumber":4,"temp":44,"hum":57}	json	a minute ago
event_1	{"randomnumber":1,"temp":57,"hum":53}	json	a minute ago
event_1	{"randomnumber":15,"temp":43,"hum":39}	json	a minute ago

7.1.3 Line chart IBM watson IoT platform board



7.1.4 Installing IBM IoT using Command prompt window

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\DHARANIDHANA>pip install ibmiotf
Requirement already satisfied: ibmiotf in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site-packages
(0.4.0)
Requirement already satisfied: iso8601>=0.1.12 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site-
packages (from ibmiotf) (1.1.0)
Requirement already satisfied: pytz>=2017.3 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site-pac
kages (from ibmiotf) (2022.6)
Requirement already satisfied: paho-mqtt>=1.3.1 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site
-packages (from ibmiotf) (1.6.1)
Requirement already satisfied: requests>=2.18.4 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site
-packages (from ibmiotf) (2.28.1)
Requirement already satisfied: requests_toolbelt>=0.8.0 in c:\users\dharanidhana\appdata\local\programs\python\python37\
lib\site-packages (from ibmiotf) (0.10.1)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib
\site-packages (from requests>=2.18.4->ibmiotf) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\site-pac
kages (from requests>=2.18.4->ibmiotf) (3.4)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\dharanidhana\appdata\local\programs\python\python37\
lib\site-packages (from requests>=2.18.4->ibmiotf) (2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\dharanidhana\appdata\local\programs\python\python37\lib\si
te-packages (from requests>=2.18.4->ibmiotf) (2022.9.24)
You are using pip version 10.0.1, however version 22.3.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Users\DHARANIDHANA>

```

7.1.5 Node-RED on IBM Cloud

Node-RED on IBM Cloud

Node-RED

Flow-based programming for the Internet of Things

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

This instance is running as an IBM Cloud application, giving it access to the wide range of services available on the platform.

More information about Node-RED, including documentation, can be found at [nodered.org](#).

Go to your Node-RED flow editor

[Learn how to customise Node-RED](#)

7.1.6 Run Python code with connection of IBM Cloud and Node-RED.

```
ibmiotfpublish.py - C:\Users\DHARANIDHANA\AppData\Local\Programs\Python\Python370\...
File Edit Format Run Options Window Help
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "f7gdph"
deviceType = "Abcd"
deviceId = "204527"
authMethod = "token"
authToken = "8778766443"

# Initialize GPIO

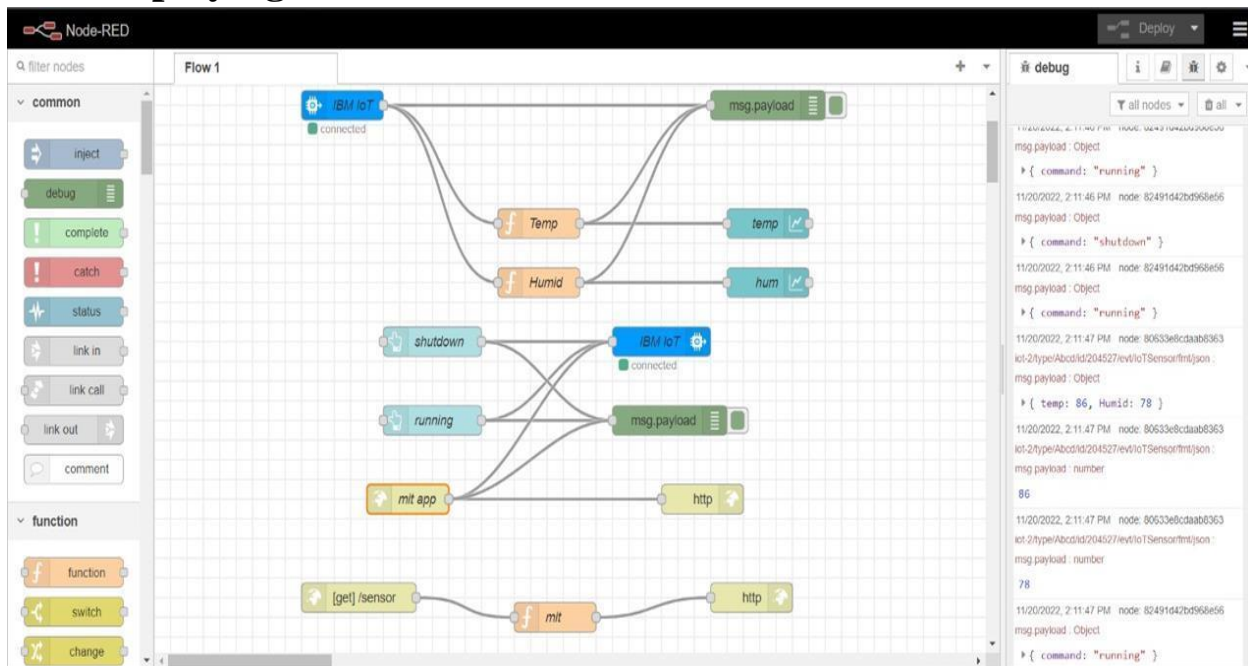
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="shutdown":
        print ("machine off")
    else :
        print ("machine on")
    #print(cmd)

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId}
    deviceCli = ibmiotf.device.Client(deviceOptions)
    #.....
except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

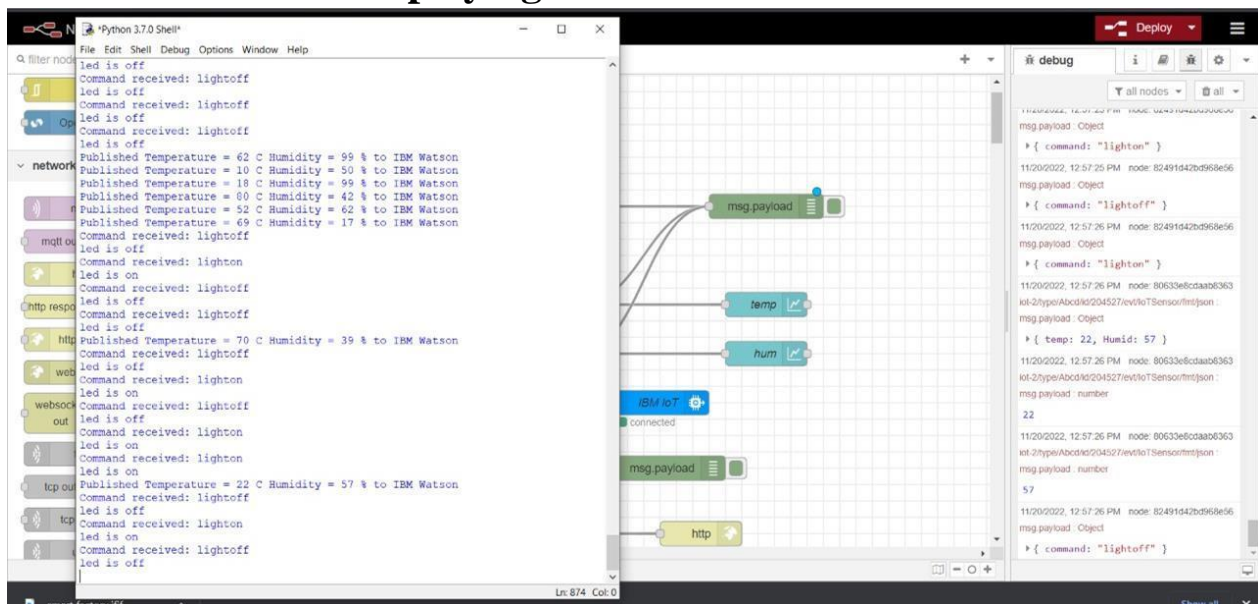
# Connect and send a datapoint "hello" with value "world" into the cloud as an e

Python 37.0 Shell
File Edit Shell Debug Options Window Help
Published Temperature = 26 C Humidity = 6 % to IBM Watson
Published Temperature = 44 C Humidity = 18 % to IBM Watson
Published Temperature = 45 C Humidity = 51 % to IBM Watson
Command received: shutdown
machine off
Published Temperature = 88 C Humidity = 81 % to IBM Watson
Command received: shutdown
machine off
Command received: shutdown
machine off
Published Temperature = 81 C Humidity = 33 % to IBM Watson
Command received: shutdown
machine off
Command received: shutdown
machine off
Published Temperature = 83 C Humidity = 44 % to IBM Watson
Command received: running
machine on
Command received: shutdown
machine off
Command received: shutdown
machine off
Published Temperature = 3 C Humidity = 15 % to IBM Watson
Command received: shutdown
machine off
Command received: running
machine on
Command received: running
machine on
Command received: running
machine on
Published Temperature = 16 C Humidity = 58 % to IBM Watson
Published Temperature = 50 C Humidity = 82 % to IBM Watson
Ln: 5 Col: 0
```

7.1.7 Deploying in Node-RED

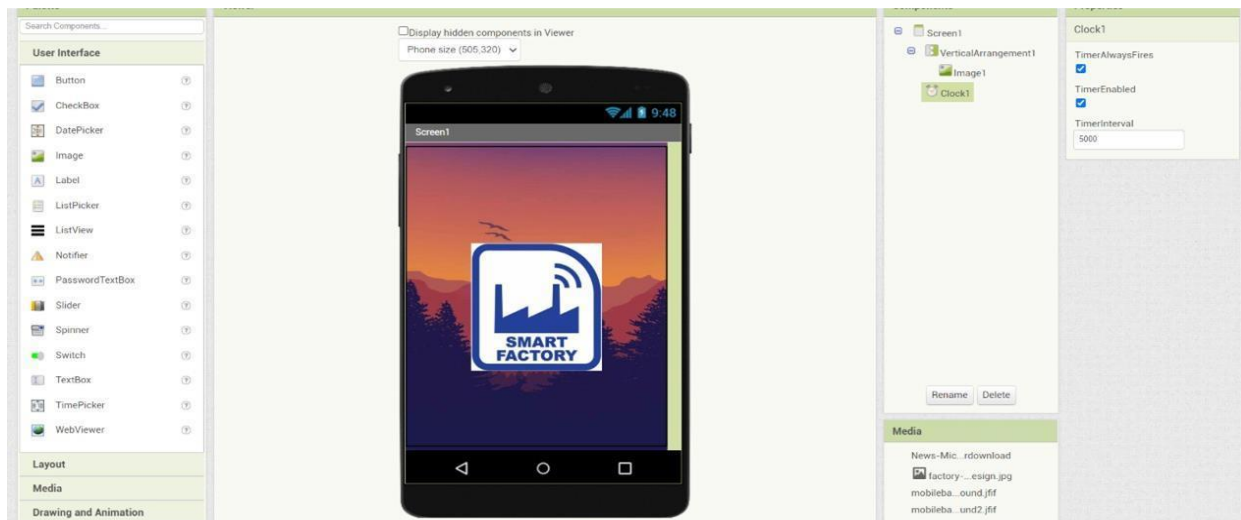


7.1.8 Simulation of Deploying Node-RED on IBM Cloud.

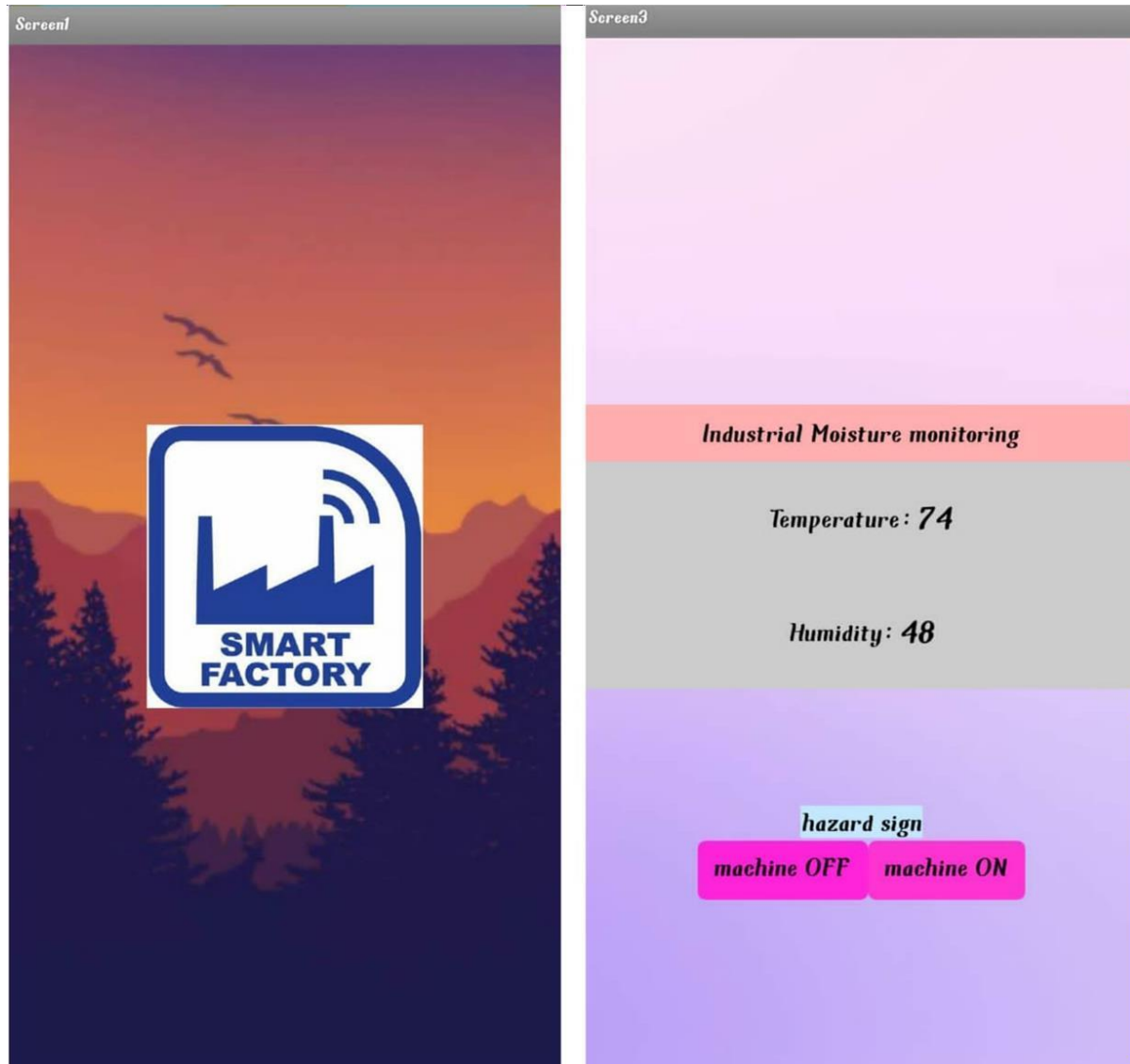


7.2 Feature-2

7.2.1 Displaying Temperature and Humidity values in Mobile Application.

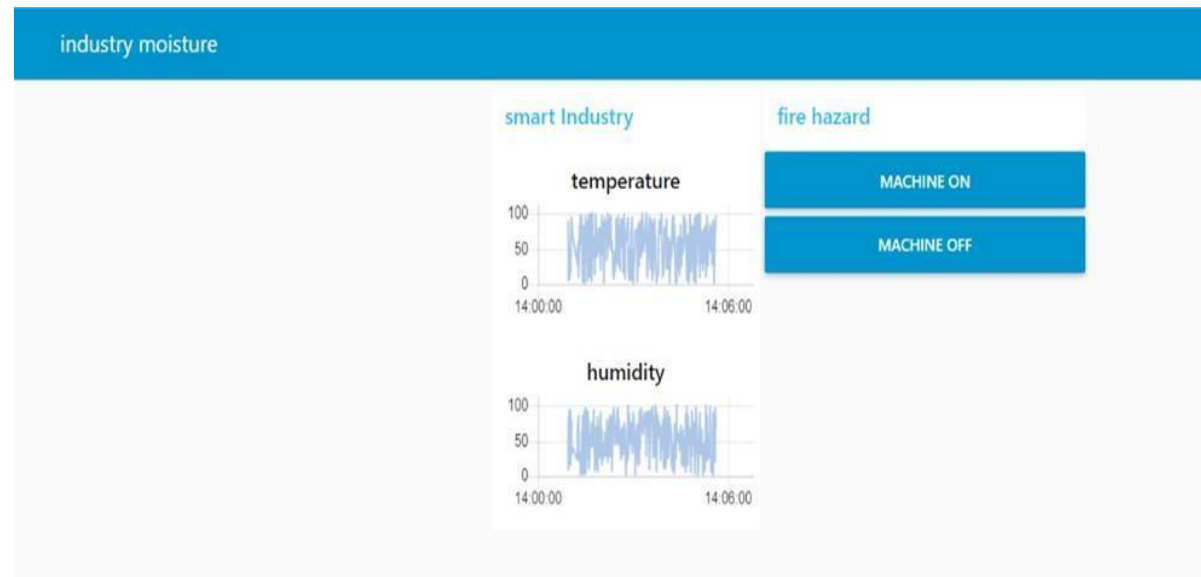


7.2.2 Detecting Temperature and Humidity through Mobile Application.



7.3 Database Schema

7.3.1 Node-RED dashboard



7.3.2 Wokwi Connection using DHT22 Sensor

WOKWI SAVE SHARE sketch.ino Docs SIGN UP

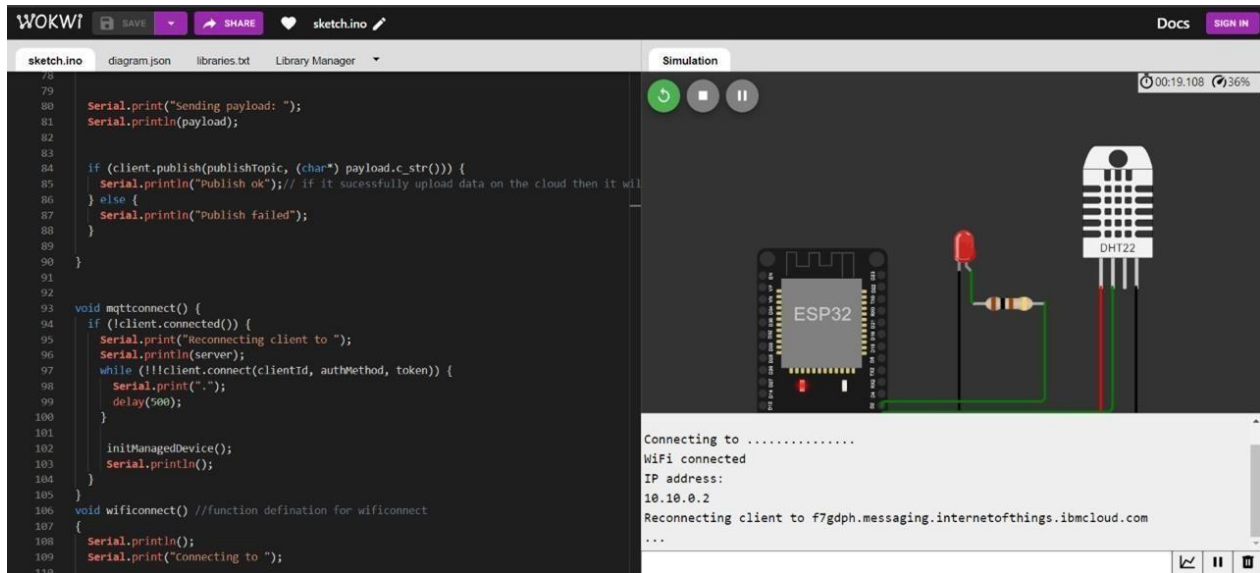
sketch.ino diagram.json libraries.txt Library Manager

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 #include "DHT.h" // Library for dht11
4 #define DHTPIN 15 // what pin we're connected to
5 #define DHTTYPE DHT22 // define type of sensor DHT 11
6 #define LED 2
7
8 DHT dht (DHTPIN, DHTTYPE); // creating the instance by passing pin and type of dht connect
9
10 void callback(char* subscribtopic, byte* payload, unsigned int payloadlength);
11
12 //-----credentials of IBM Accounts-----
13
14 #define ORG "f2gph" //IBM ORGANIZATION ID
15 #define DEVICE_TYPE "Abcd" //Device type mentioned in ibm watson IOT Platform
16 #define DEVICE_ID "204322" //Device ID mentioned in ibm watson IOT Platform
17 #define TOKEN "0770766443" //token
18 String data3;
19 float h, t;
20
21 //----- Customise the above values -----
22
23 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
24 char publishTopic[] = "iot-2/evt/Data/fmt/json"; // topic name and type of event perform a
25 char subscribtopic[] = "iot-2/cmd/command/fmt/String"; // cmd REPRESENT command type AND
26 char authMethod[] = "use-token-auth"; // authentication method
27 char token[] = TOKEN;
28 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
29
30 //-----
31
32 WiFiClient wificlient; // creating the instance for wificlient
```

Simulation

The simulation shows an ESP32 microcontroller board connected to a DHT22 digital temperature and humidity sensor. The sensor is connected to the ESP32 via a 4-pin cable: red to VCC, green to GND, yellow to VCC, and blue to GND. A red LED is also connected to the ESP32's GND pin.

7.3.3 Wokwi Simulation



8. TESTING

8.1 Test Cases-1

NFT Test approach	load test						
step 1	Python script to connect IBM watson IoTplatform to get data from cloud						
step 2	IBM simulation						
step 3	line chart IBM watson IoT platform board						
step 4	Installing ibm iot using command prompt window and node-red on ibm cloud						
step 5	Run python code with connection of ibm cloud and deploying in node-red						
step 6	Simulation of deploying node-red on ibm cloud and then displaying temperature and humidity values in mobile application						
step 7	Detecting temperature and humidity through mobile application						
step 8	database schema node-red dashboard						

8.1.1 Test Cases-2

TEAM ID - PNT2022TMD39426									
NFT - Risk Assessment									
S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Volume Changes	Risk Score	Justification
1	Hardware from Working for Industrial Plant Powered by IoT	Unfamiliar order	Low	No Changes	Moderate	NIL	>5 to 10%	YELLOW	Reporting the hazard to control made up by using Cloud Service
NFT - Detailed Test Plan									
S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff					
1	From hazard assessment to control from hazard fulfilled by the industrial working environment	High speed, not cost effective and controlling the hazard in industrial plant	to stop industrial working and identify of hazardous gaps, when problem arises generated at the next stage	S. SETHURAMAN VSAGAR, S. SETHURAMAN, LAKSHMI					
End Of Test Report									
S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff	
1	If there is no expectation of safety from hazard assessment by the industrial working environment	controlling the hazard at industrial level	This mechanism is feasible because it is easy to setup, low trouble, when defect arises, provides the risk, is beforehand, affordable	This mechanism is feasible and it is easy to set up. Your security and easy to access in every field.	Industrial hazard is the danger or chance of potential harm to the worker, while engaged in his occupational work in the plant. The most occurring problems in industries are to maintain hazardous gas leakage, overloading in machines, electrical equipment, accidents due to improper operating conditions and faults in systems. Identifying industrial hazards, these employees to protect their worker from accidents, injuries, and fatalities.	High cost savings, Greater job satisfaction and higher motivation under leadership by the workers if their expectations of safety from hazards are fulfilled by the industrial working environment.	When problem occurs, the investigation is based on the problems found there will be more possibilities to make the issue to be applicable.	S. SETHURAMAN VSAGAR, S. SETHURAMAN, LAKSHMI	

8.2.2 Test Cases-3

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Client Application	35	0	0	35
Security	2	0	0	2
Outsource Shipping	4	0	0	4
Exception Reporting	7	0	0	7
Final Report Output	8	0	0	8
Version Control	1	0	0	1

8.2 User Acceptance Testing

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	8	5	4	1	18
Duplicate	0	2	1	0	3
External	1	2	3	0	6
Fixed	12	4	2	18	36
Not Reproduced	1	0	0	0	1
Skipped	1	0	1	0	2
Won't Fix	3	5	0	1	9
Totals	20	12	14	25	71

9. RESULTS

9.1 Performance Metrics

S.NO	PARAMETERS	PERFORMANCE
1.	Response Time	0.2s (Average of 10 trials)
2.	Workload	500 users (Calculated based on Cloud Space)
3.	Revenue	Individual users while hazards occurs in an Industrial plant or Industry
4.	Efficiency	User friendly, low cost, Simple and straightforward workflow, which makes the process efficient
5.	Down Time	Almost no down time due to IBM Cloud enabled solution.

10. ADVANTAGES

- Immediate response of fire accident by detecting temperature and humidity in the work station.
- Implementation is simpler. Maintain the controllable atmospheric condition.
- Electronic Sensors have got enormous contribution towards the research in the field of mining domain.
- Prevent the workers and hazardous occurring in industries earlier by using this method.

DISADVANTAGES

- Periodic adjustment.
- Complexity in designing process.
- Continuous monitoring is required.

11. CONCLUSION

Hence, The project offers that by using continuous monitoring and detecting of hazardous signs, major problems will be prevented at the initial stage itself. Huge cost savings, Greater job satisfaction and higher motivation can be obtained by the workers if their expectations of safety from hazards are fulfilled by the industria working environment. This mechanism is feasible because it is easy to setup, user friendly, wireless detector, predicts the risk, in beforehand, affordable.

12. FUTURE SCOPE

Fire accidents can be controlled to a great extent in a places such as forests, homes, colleges, industries, trains and some other public places. Fire accidents leads to death of excess of people, by using this technique we can solve those lifes easily.

The emergency shutdown systems market is anticipated to grow at 7.2% CAGR during the forecast period. The emergency shutdown system is largely deployed in the oil & gas industry, power generating sector, manufacturing industry, and various other industries.

13. **APPENDIX** **Source Code**

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

```
#Provide your IBM Watson Device Credentials
organization = "f7gdph"
deviceType = "Abcd"
deviceId = "204527"
authMethod = "token"
authToken =
"8778766443"
```

```
# Initialize GPIO
```

```
def myCommandCallback(cmd): print("Command received: %s" %
cmd.data['command']) status=cmd.data['command'] if
status=="shutdown ":
print ("machine
off")
else :
print ("machine
on") #print(cmd)
```

```

try:
    deviceOptions = {"org": organization, "type": deviceType, "id":
deviceId, "auth-method": authMethod, "auth-token": authToken} deviceCli =
    ibmiotf.device.Client(deviceOptions)
    #.....

except Exception as e: print("Caught exception connecting device:
    %s" % str(e)) sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as
an event of type "greeting" 10 times deviceCli.connect()

while True:
    #Get Sensor Data from
    DHT11
    temp=random.randint(0,100)
    Humid=random.randint(0,100)
    data = { 'temp' : temp, 'Humid':
    Humid } #print data def
    myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s
%%" % Humid, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback) if not
    success:
        print("Not connected to IoTf")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback

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

```

GitHub Link- <https://github.com/IBM-EPBL/IBM-Project-15506-1659599467>

Project Demo Link-