# HAZARDOUS AREA MONITORING FOR INDUSTRIAL PLANT POWERED BY IOT

## TEAM ID: PNT2022TMID01115

**TEAM MEMBERS**
**PRINCEY .T**
**DHANALAKSHMI.R**
**VISHNU PRIYA.S**
**RUVANTHIKA.P.A**

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

## 1. INTRODUCTION :

Technology advancement is a never-ending process; thus, we must be well-equipped and informed about new developments. Day-to-Day human life has gotten more convenient as a result of these technological improvements. Automation has evolved into a must need. The internet today provides access to all data and systems, and web technology is continually expanding. A network interface enables remote management and control of embedded devices using a web-based embedded system. Controlling Internet of Things (IoT) devices is

done through web controllers, often

known as E -controllers. A web controller, often known as an Econtroller, is a set of embedded systems and software stacks that is the most extensively used method of web development in the world. Instead of employing large server systems for monitoring, administering, and handling data, remote login and monitoring using a distributed web control system produced using web pages generated in web applications are increasingly used instead of big server systems for monitoring, administering, and processing data. Web control systems that leverage IoT has three characteristics: energy savings, comfort, and efficiency. Our main objective is to adapt

the Internet control system to the Internet of Things, allowing users to access the application over the Internet from anywhere in the globe. IoT monitoring allows youto analyze dynamic systems and analyze billions of events and alerts. IoT monitoring also enables you to bridge the gap between devices and businesses by collecting and analyzing a wide range of IoT data at a web scale across connected devices, consumers, and apps. The industrial monitoring system connects itself with the open-source app Blynk. Blynk connects itself with esp8266 for virtual control of the devices along with getting updates. The Arduino Mega is the brain of the project connected to

the component and operates them with the code embedded in it. Sensors like smoke sensors, humidity, and temperature sensors are used to monitor the surroundings of the machine

## 1.1. PROJECT OVEREVIEW :

IN an industry ,there are many area which deal with temperature ,pressure ,humidity etc. These areas are called as hazardous area since they are highly prone to risks and hence theymonitoring intensely and with great precision .To monitor the various parameter like temperature, pressure ,humidity etc, there are various sensor that are installed inside the premises of the industry ,especially inside these hazardous area. These hazardous areas

.These sensor work with high precision and can sense and detect even the smallest fluctuation (change) .In the temperature pressure and humidity. To detect the change in temperature , high quality sensor like thermistor ,thermocouple ,resistance temperature detectors(RTDs) and other semiconductor based ICs are used .To detect change in pressure

high quality  sensor like gauge MMs pressure sensor variable capacitance pressure sensors etcare used .To detect the change in humidity levels a, high quality capacitve humidity sensor is used .If in case there parameter ,there is an alarm that goes off, there by notifying and precautioning the worker ,employers and the people in the vinicity if in case a fire break out
,there are water and or sand based on the cause of the fire .If the fire is because of a short circuit, then putting it out using water may become fatal, Hence sand is used. Sometime evengases like carbondioxide is used extinguish the fire .To avoid all tht need to be taken .Hence both the employees should be notified of the increase or decrease in any o the aforementioned parameter .This can be done in many ways . Like foe instance ,any person

who enter the hazardous area should waer proper PPE kit and a sensor should be installed inany part of the uniform of the employer like im thr ID card or jacket . Also the employer should receive an alert in his mobile phone via a notification about the change in the parameters .Just the way the employer /worker Is cautioned of the change in the parameter .the emolyers and also the other workers are entited to know the change in the proper precaution can be taken before any mishappening the information that the employer receivershould be boardcasted to all the workers for the safety .It should also be stored in both the database and in the cloud for future reference .Also necessary step should be taken to avoid such calamities in future

## **1.2.** PURPOSE:

The proposed system aims at reducing the risk of fires and explosions, thus increasing the safety of workers engaged in maintenance or inspection of gas storages. The monitoring system is based on compact battery-powered wearable sensor nodes containingsensors for LPG flammable compounds, toxic gases, and oxygen.

## 2. LITERATURE:

### 1. IoT Based Data Logger for Weather Monitoring UsingArduino Based Wireless Sensor Networks with

### Remote Graphical Application and Alerts

ABSTRACT:

In recent years, monitoring systems play significant roles in our life. So, inthis paper, we propose an automatic weather monitoring system that allowshaving dynamic and real-time climate data of a given area. The proposed system is based on the internet of things technology and embedded system.The system also includes electronic devices, sensors, and wireless technology. The main objective of this system is sensing the climate parameters, such as temperature, humidity, and existence of some gases,

based on the sensors. The captured values can then be sent to remote applications or databases. Afterwards, the stored data can be visualized ingraphics and tables form**.**

## 2.Design and Validation of a Multifunctional Android-BasedSmart Home Control and Monitoring System

ual operation using a touch display. The proposed system transmits sensordata to a

cloud platform and can receive commands from the server, allowing manydevices to be automatically

controlled. To demonstrate the feasibility and effectiveness of this system,devices such as light

switches, power plugs, and various sensors, including temperature, gas, 2.5-μm particulate matter

(PM2.5) and motion sensors, were integrated into a prototype of theproposed home control system.

Finally, we implemented the prototype in a model home to validate theflexibility, scalability, usability,

and reliability of the system

**3.** Micraspis : A Compute r-Aided Proposal Toward Programming andArchitecting Smart IoT

Weara

bles

ABST

RACT:

A wearable is a lightweight body-worn device that relies on

data-driven communications to keep people connected purposefully, for instance, for fire-fighting, prompting fast-food clients, and medical treatment. With the

rise of wearable computing in the era of IoT-driven smart applications, programmers now expect the time to market for these devices to be shortened.  While support for IoT programming in general has gatheredtraction, tool proposals that

automate the development of smart solutions based on the Internet of Wearable Things, though of paramount importance, still stay on the sidelines. We propose a code generation tool called Micraspis that allows a

wearable to be described both functionally and architecturally – as if theyare two sides of the same coin. The tool has an underlying model-to-codetransformation mechanism to generate source code that is executable on aspecific IoT programming platform such as Arduino. Our experiments demonstrate that programming code generated by Micraspis

amounts to at least 60% of the source code needed to fulfill the businesslogic of ordinary wearable devices. We conducted an interview to meticulously collect programmers' assessment on how Micraspis assists them in programming and architecting smart IoTwearables. A total of 161 programmers responded to a Likert scale questionnaire, with which at least 65% of them

either agree or strongly agree. Overall, the results show that Micraspis haspromising applicability in supporting IoWT-enabled smart solutions.

## 2.1. **EXISTING PROBLEM**:

- NO information about where can we implement this ,just themonitoring thing is explained and done.

- Bounded only to mobile application and there is no
- web application or SMS for fast notification as we may not have ourInternet connections on always
- This can cause limitations as we may not be able to monitor throughother means

## 2.2. REFERENCE:U

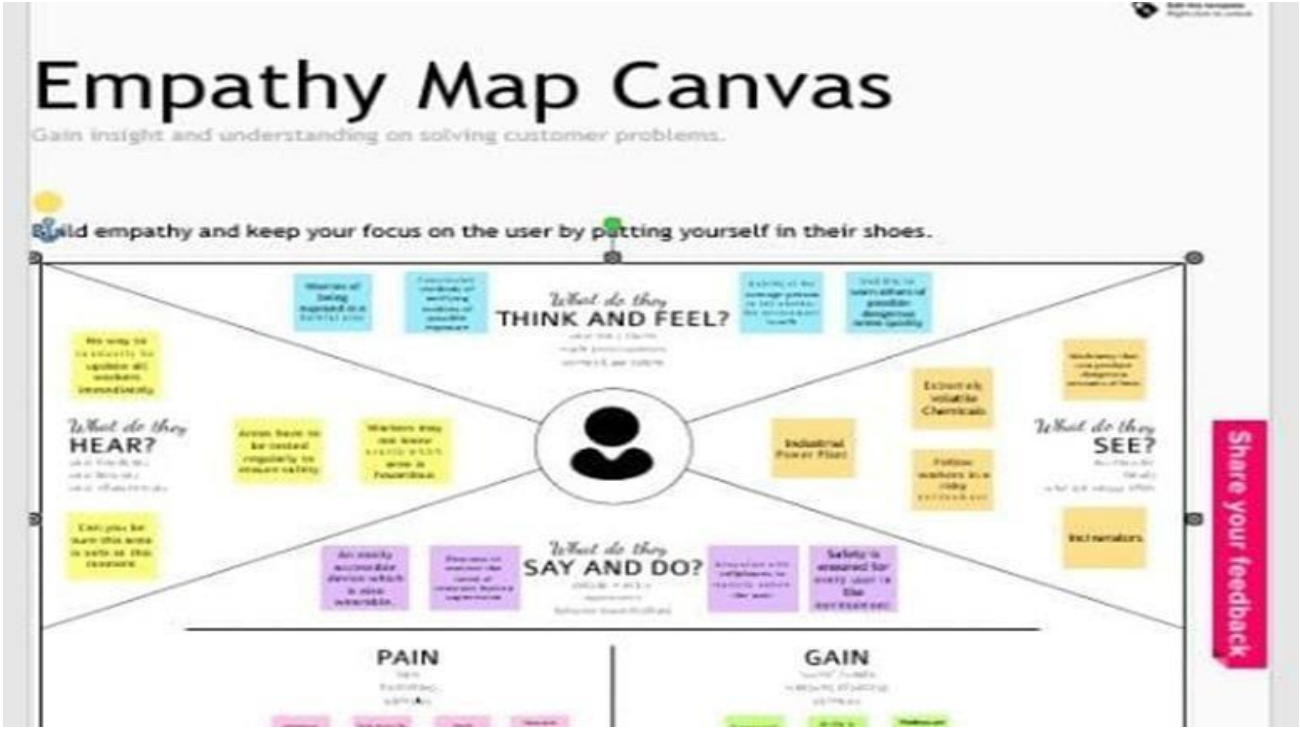
Jamal Mabrouki , Mourade Azrour, Driss Dhiba, Yousef Farhaoui, andSouad El Hajjaji

LUN-DE LIAO (Member, IEEE), YUHLING WANG YUNGCHUNGTSAO, IJAN WANG, DE-F

***JHANG, TSUNGSHENG CHU, CHIA-HUI TSAO, CHIHNING TSAI, SHENG FU , CHIUNGCHENGCHUANG,***

AND TZONGRONG GER LONG-PHUOC TÔN, LAM-SON LÊ, (Member, IEEE), ANDMINH-SON NGU**CHEN**Y

3.1

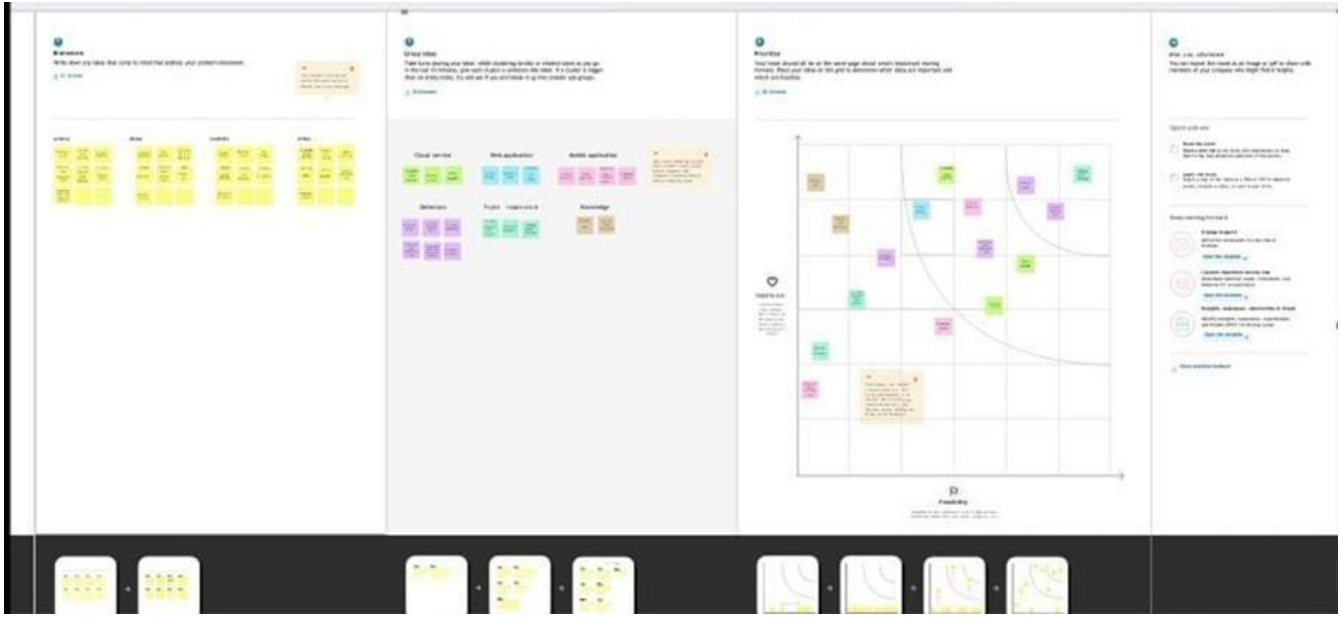

Empathy Map Canvas
Gain insight and understanding on solving customer problems.
Build empathy and keep your focus on the user by putting yourself in their shoes.
What do they THINK AND FEEL?
What do they HEAR?
What do they SEE?
What do they SAY AND DO?
PAIN
GAIN
Share your feedback

HAZARDOUS AREA MONITORING FOR INDUSTRIAL PLANT POWERED BY IOT

**CATEGORY: INTERNET OF THINGS**

| PROBLEM STATEMENTS | IAM | I'M TRYING TO | BECAUSE | WHICH MAKES ME FEEL |
|---|---|---|---|---|
| PS-1 | RAM, FACTORY OWNER | BE NOTIFIED OF THE HAZARDOUS MAYHEM | IT MAY CREATE CHAOS TO LIVELIHOOD OF PEOPLE IN VICINITY | RESPONSIBLE FOR THE LIVES OF MY WORKERS |
| PS-2 | JAMES, MANAGER | BE INFORMED OF THE EMPLOYEES ENTERING THE HAZARDOUS AREA | IT CAN CAUSE SERIOUS HEALTH ISSUES | ACCOUNTABLE FOR THEIR LIVES |
| PS-3 | NANDHINI, SAFETY INSPECTOR | ANALYSE WHETHER THE AREA IS HAZARDOUS OR NOT | I NEED TO PROVIDE ACCURATE REPORTS | WHICH MAKES ME ANSWERABLE TO THE HIGHER AUTHORITIES |
| | | | | |

3.2



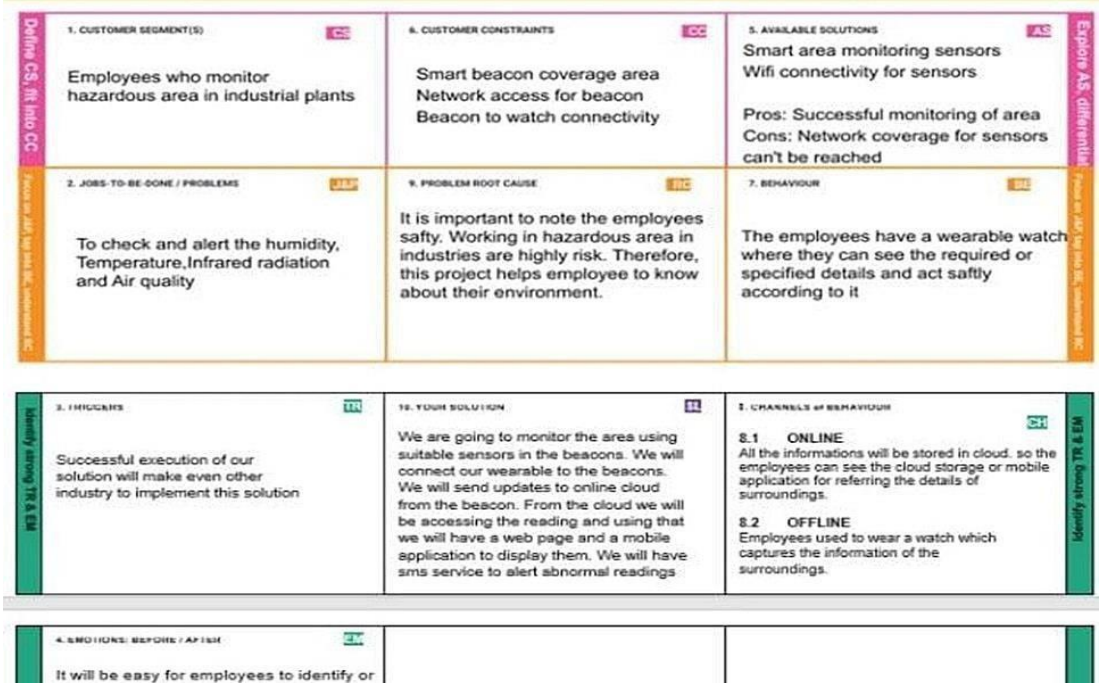
## 3.3 PROPOSED SOLUTION

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | To detect and warn the employees of the danger due to fire that may breakout within the vicinity and to safeguard the employees |
| 2. | Idea / Solution description | By installing modern-day temperature sensors like thermocouples ,RTD 's (Resistance temperature detector) thermistors to detect change in temperature ,if any . An alarm goes off and lights start blinking to show the change in temperature. By constructing big and long pipes on the ceiling which spray water or sand (According to the cause of fire) |
| 3. | Novelty / Uniqueness | • Makes it easier to know the temperature (or) any hazardous gases present in the area without the worker having to constantly doing manual checks.<br>•Provides different solution to ensure the safety of the workers.<br>•Wearable devices display the current temperature present in the area all the time.<br>• Alerts via SMS to mobiles of the workers when high temperature is detected. |

| | | |
|---|---|---|
| 4. | Social Impact / Customer Satisfaction | • Very safe<br>• Cost effective<br>• Easy installation<br>• User friendly<br>• Easy detection and avoid accident |
| 5. | Business Model (Revenue Model) | • By introducing non skip advertisements in social platforms |

| | | |
|---|---|---|
| | | • High quality sensors can be brought from good reputed companies through tenders approved by government |
| 6. | Scalability of the Solution | • Good quality sensors detect the temperature very accurately before the fire explosion<br>• Water and sand are always kept handy in large amounts to stop the fire from spreading<br>• Every user is immediately alerted and comes to know of the difference in temperature through the chip that is present in his safety jacket<br>• It ensures the safety of each and every worker |

## 3.3 PROBLEM SOLUTION:



## **4.**REQURIMENT ANALYSIS:

4.1

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | Data Gathering | The temperature sensors like thermistors, thermocouples and RTD's must be able to sense the change in temperature. |
| FR-2 | Location Detection | The GPS must be able to detect the location when the employees enter the hazardous area. |
| FR-3 | Data Syncing | The I-Net control gas detection software must be able share the data to both the employees and the admin through the cloud. |
| FR-4 | Thermistor Device Display | The thermistor device must be able to sense and display both the temperature and pressure where the worker is currently present. |
| FR-5 | SMS Notification | If the temperature changes and reaches to dangerous levels, then the employees must be notified with an SMS and should be able to leave the place immediately. |
| FR-6 | Admin Dashboard | If there is a difference in the temperature and before it reaches the dangerous level, the admin must be informed and he must take necessary precautions. |

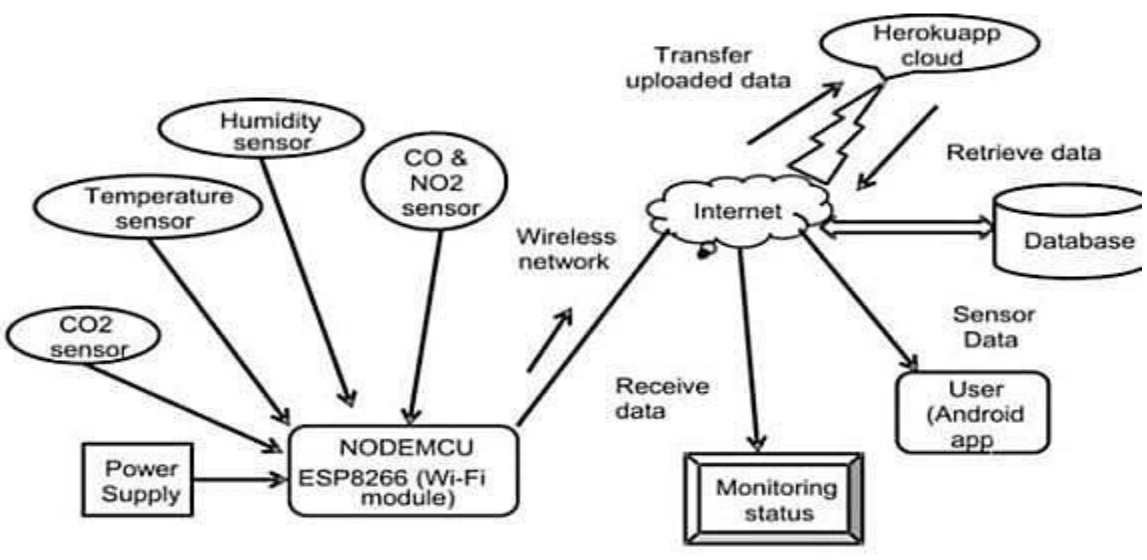
4.2

**Non-functional Requirements:**

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

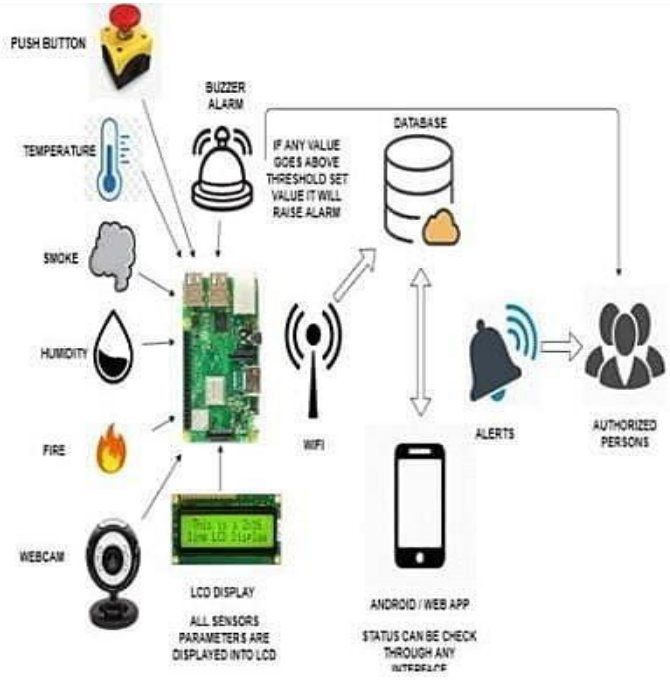
| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | The wearable device (RF ID card) must be smart, accurate and light weight. It should display the difference in temperature accurately and immediately without any delays. |
| NFR-2 | **Security** | The connection of the temperature sensors to the wearable device and the cloud must be very safe and secure.<br>The security of the database housing all the data containing the temperature must be kept secure and non hackable. |
| NFR-3 | **Reliability** | The RF ID card should be able to perform accurately without any issues even during calamities. |
| | | overall efficient monitoring.<br>This can be done in different plants in different ways, thereby increasing the scalability to an higher extent. |

5.1.DATA FLOW MODEL:



## 5.1 SOLUTION ARCHITECTURE:

The node mcu is connected with temperature sensor and gas detecting sensors and other carbon di o
in every employees phone and when hazardhappens, this sensor sends notification to the employees mobile through cloud applications at the time employees are notified about the high pressure or high temperature about the hazardous gases at that particular place and they wont enter into that region. After sometime when the condition turns to normal, then again te notification is sent to all the employees regarding the condition of the place and then the employees enter into their workspace and continue with their work. This is done with the help of node mcu temperature, humidity, pressure, gas sensors and with a help of cloud and internet and then an app.


PUSH BUTTON
BUZZER ALARM
TEMPERATURE
IF ANY VALUE GOES ABOVE THRESHOLD SET VALUE IT WILL RAISE ALARM
DATABASE
SMOKE
HUMIDITY
FIRE
WEBCAM
WIFI
ALERTS
AUTHORIZED PERSONS
LCD DISPLAY
ALL SENSORS PARAMETERS ARE DISPLAYED INTO LCD
ANDROID / WEB APP
STATUS CAN BE CHECK THROUGH ANY

## 5.2 User stories

**User Stories**

Use the below template to list all the user stories for the product.

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Technician | Installation | USN-1 | The technician must install the thermistor at points to ensure the entire area of the plant is covered. | A sensor can be found in every area of the plant. | High | Sprint-1 |
| | Data Gathering | USN-2 | The thermistor and thermocouple obtain the temperature of theirrespective area using sensors. | The temperature of areas within the plant is obtained. | High | Sprint-1 |
| | Data Sync | USN-3 | The sensors send their data to the cloud in the real time which is in turn sent to nearby wearable devices and the administrators dashboard. | Data is sent to the cloud successfully and synced with other devices. | High | Sprint-1 |
| Worker | Wearable device display | USN-4 | The wearable devices should display the data sent by thermistor within the area. | The user can see the Temperature , pressure of the area on their device. | High | Sprint-1 |
| | Wearable device adjustments | USN-5 | The user can vary the length and shape of id card | The user can make adjustments to the device according to there comfort | Low | Sprint-2 |
| | Wearable display customization | USN-6 | The user can adjust software setup to suit there needs in mobile app itself | The user can modify the display of the app to increase readability. | Medium | Sprint-2 |
| | SMS Notifications | USN-7 | The user is sent a notification to their phone from the wearable device through an API when the area they are in reaches dangerous temperatures. | The user is informed of potential danger via SMS as soon as it is detected by The sensor | High | Sprint-1 |
| Administrator | Admin Dashboard | USN-8 | send the data through the cloud toa dashboard which is run by the administrator. | The data of all the sensor can be viewed by the administrator of the plant | High | Sprint-1 |
| | Dashboard Customization | USN-9 | The dashboard can be customized by the admin to suit their personal requirements and priorities. | The admin can customize the UI for their dashboard. | Medium | Sprint-2 |

# PROJECT PLANNING & SCHEDULING

## 6.1 Sprint planning & Estimation

**Product Backlog, Sprint Schedule, and Estimation :**

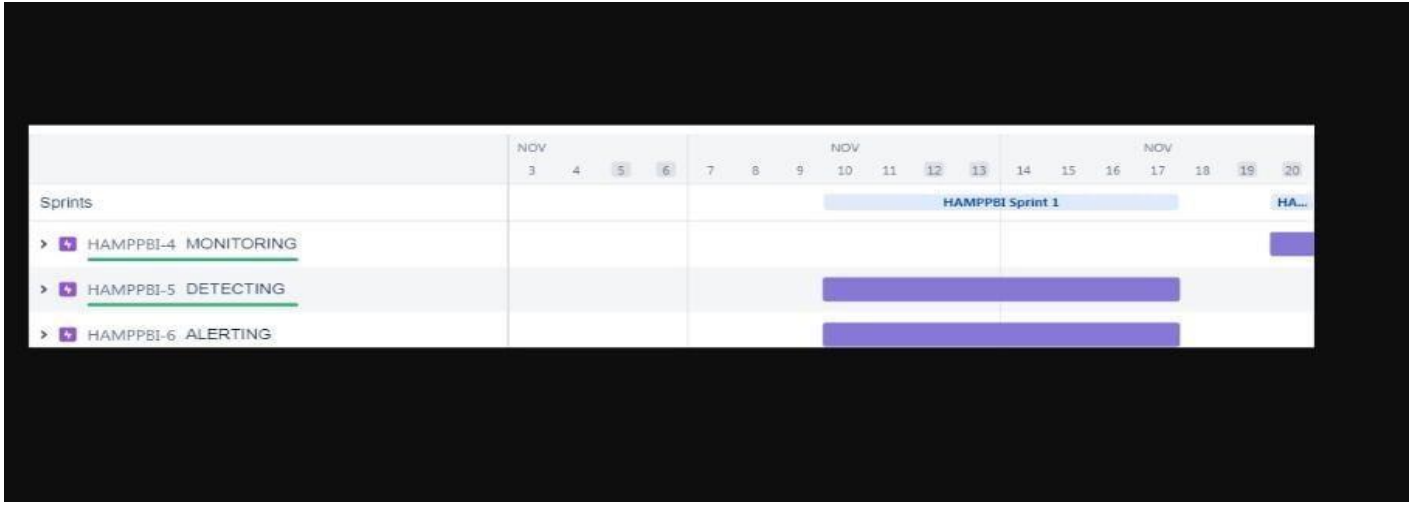
| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Detecting | USN-1 | A plant manager needs to be informed of possible hazardous areas | 5 | High | 4 |
| Sprint-1 | Alerting | USN-2 | A technician needs to be informed when he is entering a hazardous area because it can lead to health complications | 3 | High | 4 |
| Sprint-2 | Monitoring | USN-3 | A company owner must be aware of all the issues in the industrial plant | 2 | Medium | 4 |
| Sprint-2 | Monitoring | USN-4 | A safety inspector needs to be easily ascertained whether the various areas of the plant or not | 1 | Low | 4 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

## 6.2 Sprint delivery schedule:

**Project Tracker, Velocity & Burndown Chart :**

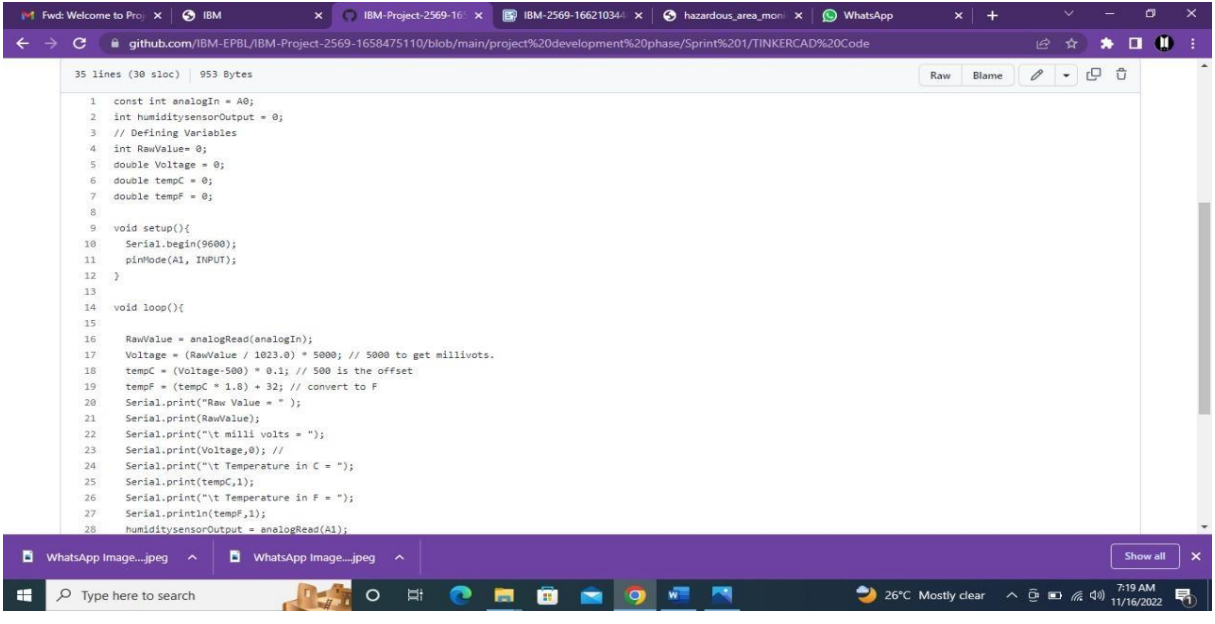
| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint Release Date (Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 7 Days | 10 Oct 2022 | 17 Oct 2022 | 20 | 18 Oct 2022 |
| Sprint-2 | 20 | 7 Days | 20 Oct 2022 | 27 Nov 2022 | 20 | 28 Nov 2022 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

6.3 Reports from JIRA:



# 7. CODING&SOLUTION:

## 7.1 Feature1



:

```
    Serial.print("Humidity: "); // Printing out Humidity Percentage
    Serial.print(map(humiditysensorOutput, 0, 1023, 10, 70));
    Serial.println("%");

    delay(5000);  //iterate every 5 seconds

}
```



**7.1: Features**

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <DHT.h>
WiFiClient wifiClient;
String data3;
#define DHTTYPE DHT11
```

```
#def
ine
DHTPIN 4
#def
ine
MQTPIN 34
DHT
dht(DHTPI
N,
DHTTYPE);


#def
ine ORG
"v6wg8x"
#def
ine
DEVICE_T
YPE
"projectFin
al"
#def
ine
DEVICE_ID
```

```
"FinalDeliv
erable"
        #def
ine TOKEN
"A1ymH))p
*JB&iMWN
pY"
        #def
ine speed
0.034


        void
callback(c
har* topic,
byte*
playload,
unsigned
int
payloadLe
ngth);


        char
server[] =
```

```
ORG
".messagin
g.interneto
fthings.ibm
cloud.com
";
        char
publishTop
ic[] = "iot-
2/evt/Data/f
mt/json";
        char
topic[] =
"iot-
2/cmd/test/
fmt/String"
;
        char
authMetho
d[] = "use-
token-
auth";
        char
token[] =
```

```
TOKEN;
        char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
        PubSubClient client(server, 1883, callback , wifiClient);
        void publishData();

        String command;
        Stri
```

```
ng data =
"";

        long
duration;
        float
dist;




        void
setup()
        {

Serial.begi
n(115200);

dht.begin()
;

wifiConnec
```

```
t();

mqttConne
ct();
        }


        void
loop() {

publishDat
a();

delay(500);


          if
(!client.loo
p()) {

mqttConne
ct();
          }
```

```
        }

        void
wifiConnec
t() {

Serial.print
("Connecti
ng to ");
Serial.print
("Wifi");

WiFi.begin(
"JerroldWi-
Fi","75779
901");

while
(WiFi.statu
s() !=
WL_CONN
ECTED) {
```

```
delay(500);

Serial.print
(".");
            }

Serial.print
("WiFi
connected,
IP address:
");
Serial.print
ln(WiFi.loc
allP());
          }

          void
mqttConne
ct() {
            if
(!client.con
nected()) {
```

```
Serial.print
("Reconne
cting
MQTT
client to ");
Serial.print
ln(server);

while
(!client.con
nect(clientI
d,
authMetho
d, token)) {

Serial.print
(".");

delay(500);
            }

initManage
```

```
dDevice();

Serial.print
ln();
            }
          }


          void
initManage
dDevice() {
             if
(client.sub
scribe(topi
c)) {

Serial.print
ln("IBM
subscribe
to cmd
OK");
            }
else {
```

```
Serial.println("subscribe to cmd FAILED");
      }
    }

    void publishData()
    {
      int sensorValue = analogRead(MQTPIN); //MQT 135 connected to GPIO 34 (Analog ADC1_CH6)
```

```
Serial.print
("AirQua="
);

Serial.print
(sensorVal
ue, DEC);

Serial.print
ln(" PPM");

float humid
=
dht.readHu
midity();

float temp
=
dht.readTe
mperature(
true);
```

```
String
payload =
"{\"Humidit
y\":";

payload +=
humid;

payload +=
"}";
            if
(client.publ
ish(publish
Topic,
(char*)
payload.c_
str())) {

Serial.print
ln("Publish
OK");
            }
```

```
payload =
"{\"Temper
ature\":";

payload +=
temp;

payload +=
"}";
            if
(client.publ
ish(publish
Topic,
(char*)
payload.c_
str())) {

Serial.print
ln("Publish
OK");
            }

payload =
```

```
"{\"AirQual
ity\":";

payload +=
String(sen
sorValue);

payload +=
"}";
            if
(client.publ
ish(publish
Topic,
(char*)
payload.c_
str())) {

Serial.print
ln("Publish
OK");
             }
           }
```

```
        void
callback(c
har*
subscribeT
opic, byte*
payload,
unsigned
int
payloadLe
ngth) {

Serial.print
("callback
invoked for
topic:");

Serial.print
ln(subscrib
eTopic);
          for
(int i = 0; i
<
payloadLe
ngth; i++) {
```

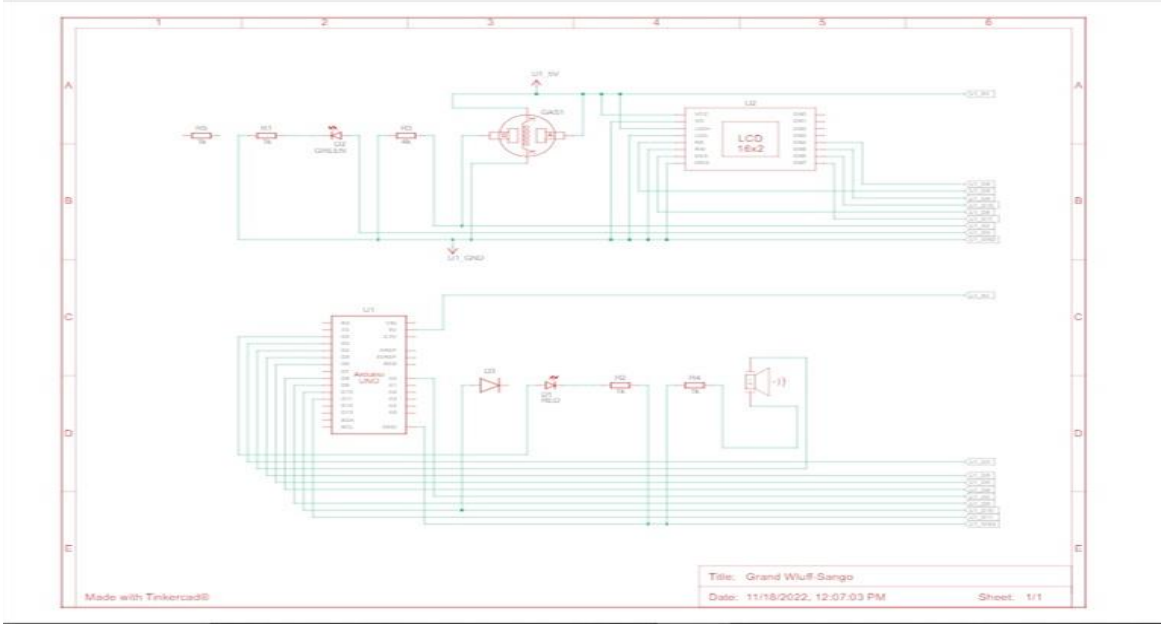
```
dist +=
(char)paylo
ad[i];
            }

Serial.print
ln("data:" +
data3);
            if
(data3 ==
"lighton") {

Serial.print
ln(data3);
            }

data3 = "";
          }
```

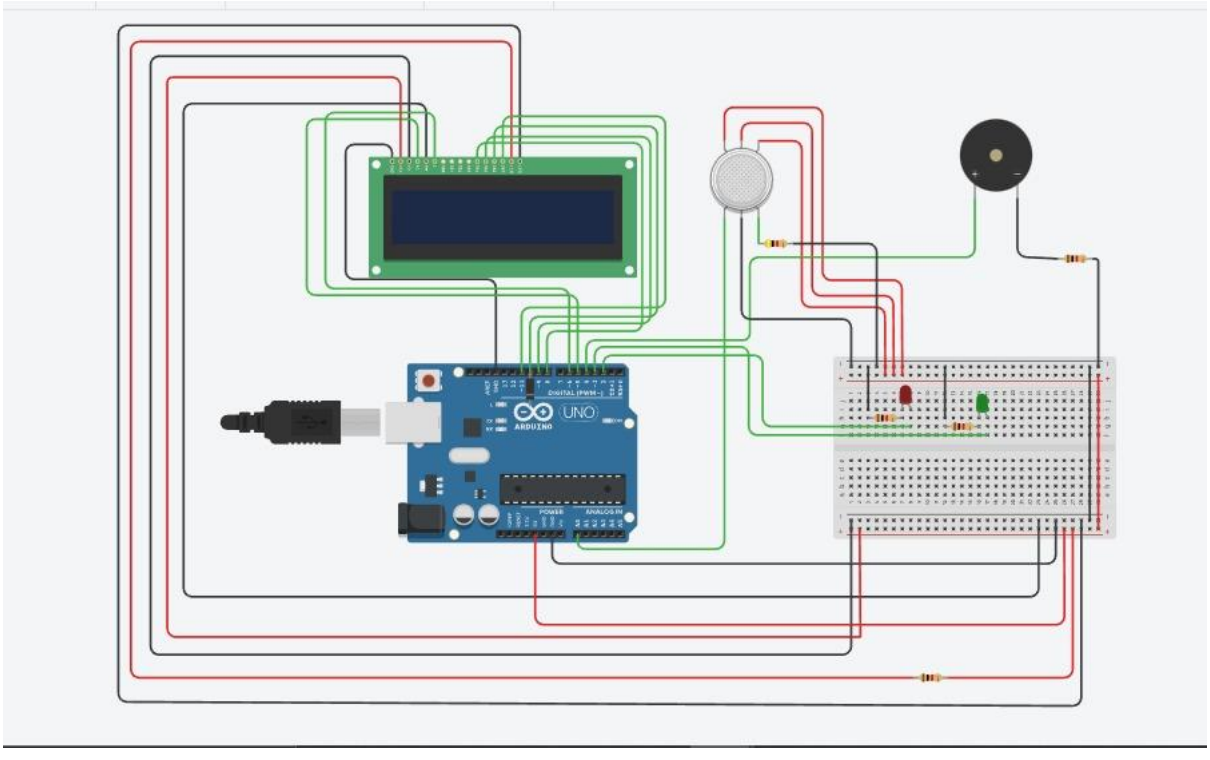
## Components

The design of a sensor-based automatic gas leakage detector with an alert and control system. The components are

| S. No. | Name of the Component | Quantity |
|---|---|---|
| 1. | Arduino UNO R3 | 1 |
| 2. | Breadboard | 1 |
| 3. | LED | 2 |
| 4. | Resistor | 5 |
| 5. | Piezo | 1 |
| 6. | Gas Sensor | 1 |
| 7. | LCD (16x2) | 1 |

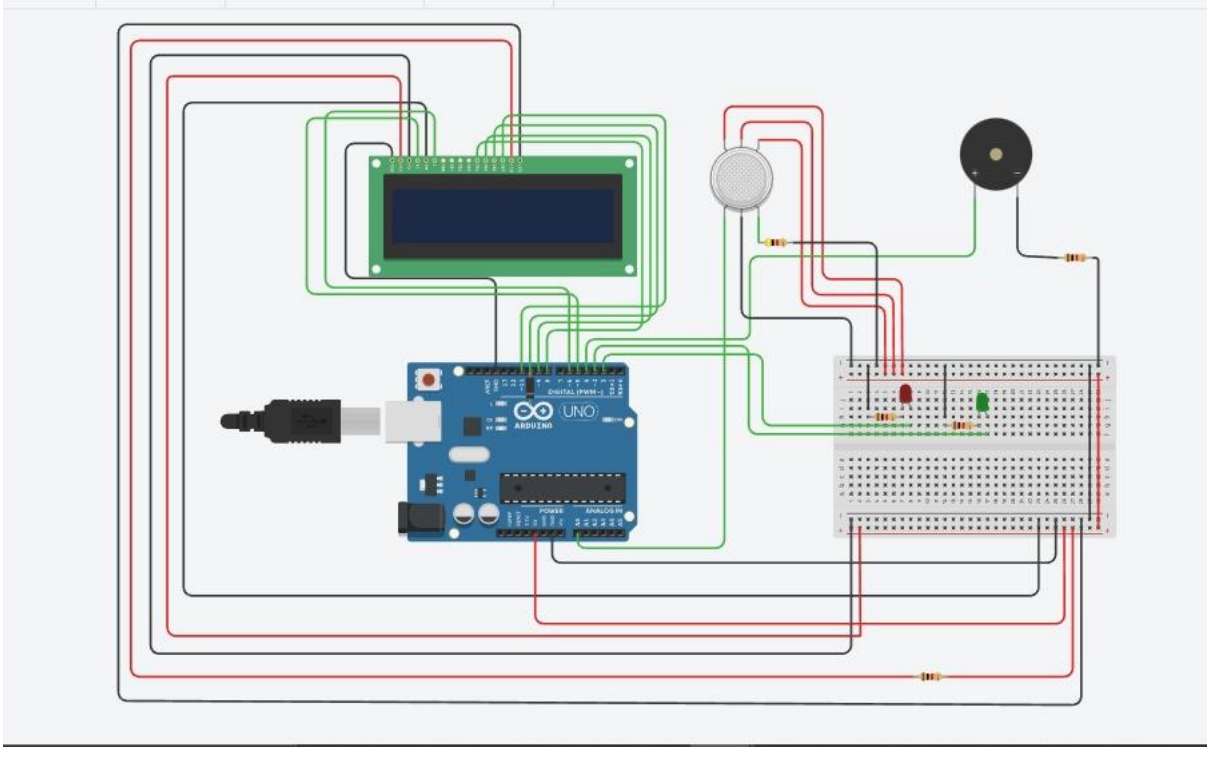
CIRCUIT DIGRAM:




DIGITAL (PWM~)
UNO
ARDUINO
POWER
ANALOG IN

## TESTING

As all connections are made and verified, the entire module is simulated to check its working. The sensor keeps on detecting any gas leakage is there. If there is no leakage happens no alert signals are sent.


UNO
ARDUINO

# ADVANTAGES

- Detect the concentration of the gases
- The sensor-enabled solution helps prevent the high risk of gas explosions and affecting any casualties within and outside the premises.
- Get real-time alerts about the gaseous presence in the atmosphere
- Prevent fire hazards and explosions.
- Supervise gas concentration levels
- Ensure worker’s health

If any of the sensor output will be high, Voice module will produce the sound for

- intimating the condition to others.
- To detect fire in the disaster-prone area.
- Also extinguishes the fire on detection.
- Reduces the level of destruction.
- Simple and low cost technology.
- Measures fl Reduce human effort.
- Reliable and economical.
- ammability of gases.
- It has robust and simple construction.
- Automation of sensors leads to better monitoring of devices

## DISADVANTAGE :

• Only one gas can be measured with each instrument.

• When heavy dust, steam or fog blocks the laser beam, the system will not be able to take measurements. This is also the case when a person or vehicle blocks the path.

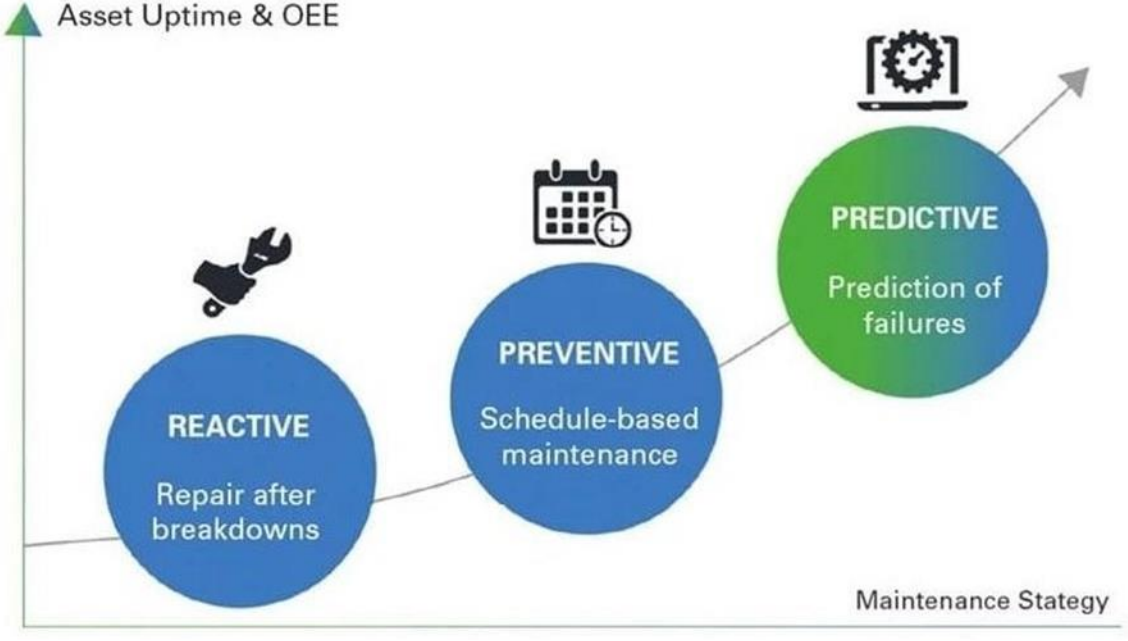
RESULT: The IoT-based study can be enhanced further by offering extra functionality to industry personnelthrough the

use of an Android app to improve industry control and monitoring. Smoke and gas sensors can also be

connected to the system to safeguard the safety of workers and commodities in the event of a fire orpoisonous

gas leak. Data can be used to minimise industrial dangers in high-profile factories, track yield in powerplants,

assure safety in fast-paced industries, and assess nuclear safety levels, among other things. Time can besaved if


Asset Uptime & OEE
REACTIVE
Repair after breakdowns
PREVENTIVE
Schedule-based maintenance
PREDICTIVE
Prediction of failures
Maintenance Stategy

CONCLUSION :

This project specifically aims to make use IOT to actively monitor and analyse various factors in a typical heavy industrial zone like temperature and levels of gases in the environment. .If the above parameter exceedthe recommended safe values, the system can track the same and issuse alerts .Also the data generated in realtime can provide important information about how smoothly the work is going on in different zones .this system can be deployed in many industrial area like mining ,underground factories ,metal refineries, automatic welding factories and even heavy parts production lines .it will help to provide a safe andefficient working

## FUTURE SCOPE:

**Major cities in India are pushing the Smart Home application, and the gas monitoring system is a part of the Smart Home application. Enhancing Industrial Safety using IoT. This system can be implemented in Industries, Hotels, and wherever the gas cylinders are used. This system can be used in industries involving applications such as Furnaces, Boilers, Gas welding, Gas cutting, Steel Plants, Metallurgical industries, Food processing Industries, Glass Industries, Plastic industries, Pharmaceuticals, and Aerosol manufacturing. Our system uses the gas sensor to detect any gas leakages. The gas sensor sends out a signal to the microcontroller as soon as it encounters a gas leakage. The microcontroller processes this signal, and a message is displayed on the LCD to alert the user**

**APPENDIX:**

## Source Code:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(6, 7, 8, 9, 10, 11);
float gasPin = A0;
float gasLevel;
int ledPin = 2;
int buttonPin = 3;
int buzzPin = 4;
int buttonState;
int fan = 5;

void setup(){
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
  pinMode(gasPin,INPUT);
  pinMode(fan,OUTPUT);
  Serial.begin(9600);
```

```
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print(" Welcome");
  lcd.setCursor(0,2);
  lcd.print("GAS LEAKAGE SYSTEM");
  delay(500);
  lcd.clear();
}

void loop(){
  // Read the value from gas sensor and button
  gasLevel = analogRead(gasPin);
  buttonState = digitalRead(buttonPin);

  // call the function for gas detection and button work
  gasDetected(gasLevel);
  buzzer(gasLevel);
  exhaustFanOn(buttonState);
}

// Gas Leakage Detection & Automatic Alarm and Fan ON
void gasDetected(float gasLevel){
  if(gasLevel >= 300){
   digitalWrite(buzzPin,HIGH);
    digitalWrite(ledPin,HIGH);
```

```
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN ON");
   delay(1000);
   lcd.clear();
  }else{
   digitalWrite(ledPin,LOW);
    digitalWrite(buzzPin,LOW);
    digitalWrite(fan,LOW);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN OFF");
   delay(1000);
   lcd.clear();
  }
}
//BUZZER
void buzzer(float gasLevel){
if(gasLevel>=300)
  {
```

```
  for(int i=0; i<=30; i=i+10)
  {
  tone(4,i);
  delay(400);
  noTone(4);
  delay(400);
  }
  }
}
// Manually Exhaust FAN ON
void exhaustFanOn(int buttonState){
  if(buttonState == HIGH){
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("Button State:");
    lcd.print(buttonState);
    lcd.setCursor(0,2);
    lcd.print("FAN ON");
    delay(10000);
    lcd.clear();
  }
}
```

- .

GITHUB LINK:

**https://github.com/IBM-EPBL/IBM-Project-2569-1658475110**

**DEMO LINK:**

- https://drive.google.com/file/d/1Bj3DLAVIk7QDTKamNXPFsVyo9yut-t2K/view?usp=drivesdk