PROJECT REPORT

1. INTRODUCTION

Project Overview

This is the IOT (internet of things) based intelligent fire monitoring and controlling system which not only gives the real time information about the situation on the monitorbut also takes the corrective action as per the need. In this system the sensorstransfer data wirelessly with thehelp of MQTT (message queuing telemetry transport) networking protocol which is designed for constrained with low-bandwidth. MQTT allows us to send commands to control output, read and publish data from sensorsnodes and much more. The first concept is the publish and subscribe system. 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 thisdata can be access by the Application program interface which we can display on themonitor and with the help of softwarethe operator can visualize the conditionat the time of fire accident.

In this project we will be discussing about Industry specific intelligent fire management system. Industry specific intelligent fire management system is a system which is specifically designed for the fire safety in industries.

This system uses various sensors and detectors to detect the fire and then it takes appropriate action to extinguish the fire. This system is very effective in extinguishing the fire and it also minimizes the damage caused by the fire.

Purpose

The purpose of the system is :

To prevent life losses, assests damageand uncontrollable spreadof fire.

To ensure the safetyof workers and alert the manager and fire department.

To not to recklessly endanger the life of the fire workers.This can be done by taking the control measures automatically.

2. LITERATURE SURVEY

Existing problems

Cost of ownership : The fire management system shoulb be cost effective. In average, the fire management is expected to last 10 years. The biggest problem is when the system cannont be maintained any longer due to componentnon-avaliabilty or due to beingunsupported by the manufacturer.

Structural changes : The structure of the hospital changes over time. The fire management system should be easilyable to upgrade and adaptable to the changingstructure.

Evaculation and fire stratergy : The alert and the control measuresare taken immediately, so that the building can be completely evacuated.

System performance changes within specificenvironments : The industry will have unique or specifiedcondition at some time. The major problemcaused is the false fire alarm.

References

1. Ahmed Imteaj, Tanveer Rahman, Muhammad Kamrul Hossain, MohammedShamsul Alam, Saad AhmadRahat, "An IoT based fire alarming and authentication system for workhouse using Raspberry Pi 3" , International Conference on Electrical, Computer and Communication Engineering (ECCE),

IEEE, 2017

2. Ondrej Krejcar, "Using of mobile devicelocalization for severaltypes ofapplications in intelligent crisis management",5th IEEE GCC Conference & Exhibition, IEEE, 2009

3.Karwan Muheden, Ebubekir Erdem, Sercan Vançin, "Design and implementation of the mobile fire alarm system using wireless sensor networks",17th International Symposiumon Computational Intelligence and Informatics (CINTI),IEEE, 2016

4. Azka Ihsan Nurrahman, Kusprasapta Mutijarsa, "Intelligent home management systemprototype design and development", International Conference on Information Technology Systems and Innovation (ICITSI), IEEE, 2015

5. Al Mamari, A. R. M. H., Al Mamari,H., Kazmi, S. I. A., Pandey, J., & AlHinai, S. (2019). IoT based Smart Parkingand Traffic Management System for MiddleEast College.

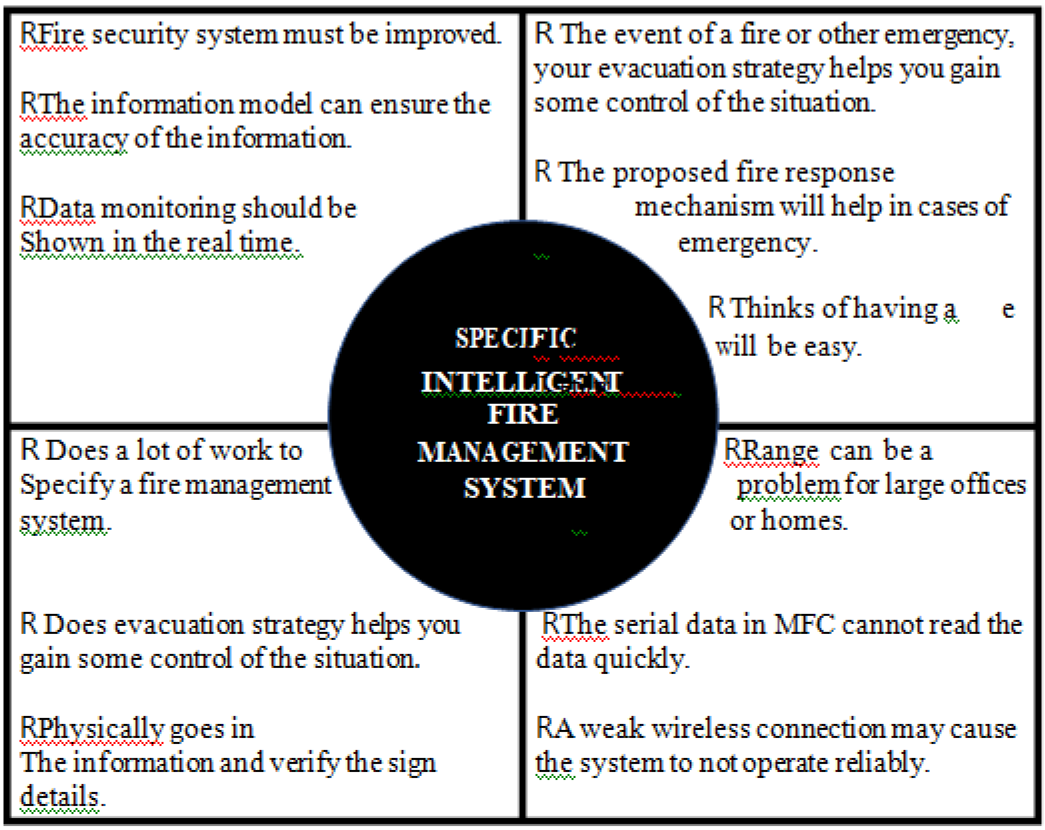
Problem Statement Definition

Fire is the rapid oxidation of a material in the exothermic chemical process of combustion, releasing heat, light and various reaction products. Although it’s a natural process, it can lead to great destruction. On average, everyday 35 people killed due to Fire-related accidents in the five years between 2016 and 2020, according to a report by Accidental Deaths and Suicides in India (ADSI),maintained by the National Crime Records Bureau. Fire is one of the major concerns when analyzing the potential risks on the building. Industrial Fires and Explosions cost companies and governments billions of Rupees every year apart from the loss of life, which can’t be described in monetary terms. These Fires not only results only in huge loss of Lives and Property but also disrupt production in the Industry. The Nilflisk says that the five major causes of industrial fires and explosions are Combustible dust, hot works, Flammable liquids and gasses, equipment and machinery and Electrical hazards.

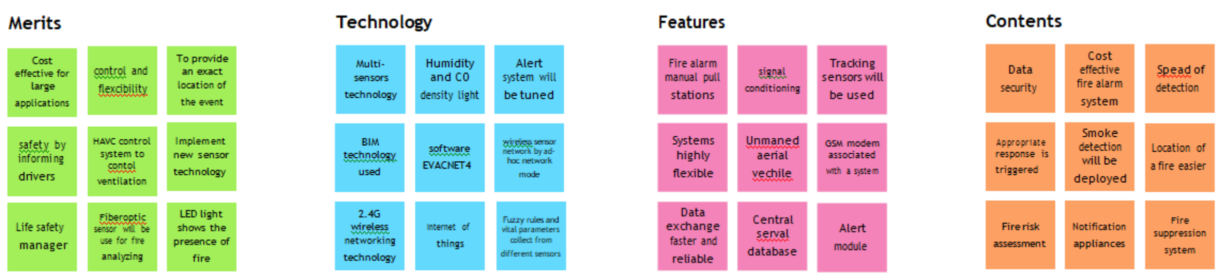
Objective: The objective of this Industry-Specific Intelligent Fire Management System is to detect any changesin environment like detecting hazardousgas, flame detectionand temperature that can lead to fire and exploitation incident. Based on the temperature readings and if any Gasses are present the exhaust fans should be powered ON automatically to replace contaminated and stale air with fresh, healthy air. If any flame is detected the sprinklers will be switched on automatically. Emergency alerts are notified to the authorities and Fire station. So that the authorities and Fire Fighters can control the situation.

3. IDEATION & PROPOSED SOLUTION:

Empathy Map Canvas:



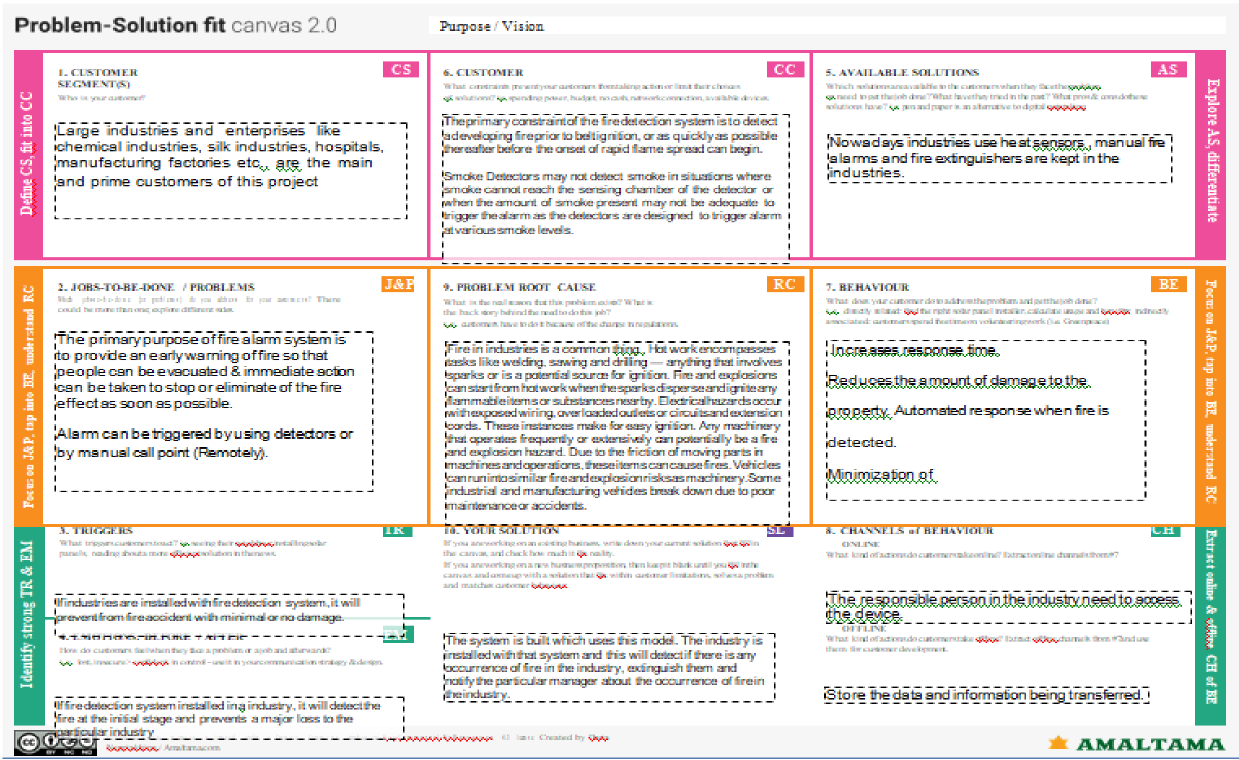
Ideation & Brainstorming



Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problemto be solved)	The existing fire alarm system on market nowadays is too complex in terms of its design and structure. Since the system is toocomplex, it needs regular maintenance to becarried out to make sure the systemoperateswell.
2.	Idea / Solution description	The main aim of the projectis the reaction orresponse time of fire alarm system, that is, the time betweenfire detection and extinguishing.
3.	Novelty / Uniqueness	In our Project we given more multi sensorstechnology and provide several database storage.
4.	Social Impact/ Customer Satisfaction	By solvingthis issue, more accidents can beprevent and exact location it will be shown if any fire occurs.
5.	Business Model(Revenue Model)	The proposed systemwill be systemhighly flexible and cost effective for largeapplications.
6.	Scalability of the Solution	The event of a fire or other emergency yourevacuation strategy helps you gain some control of the situation.

Problem Solution Fit



4. REQUIREMENT ANALYSIS:

Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	**User Requirements**	Workers and Product ProtectionAutomatic Sprinkler System Monitors Smoke ,Gas and Temperature
FR-2	**User Registration**	Manual Registration Registration throughwebpage Registration through FormRegistration through Gmail

FR-3	**User Confirmation**	Confirmation via PhoneConfirmation via Email Confirmation via OTP
FR-4	**Payment Options**	Cash on DeliveryNet Banking/UPI Credit/Debit/ATM Card
FR-5	**Product Delivery andInstallation**	Door Step delivery Take away Free Installation and 1 year Warranty
FR-6	**Product Feedback**	Through Webpage Through Phone calls Through Google forms

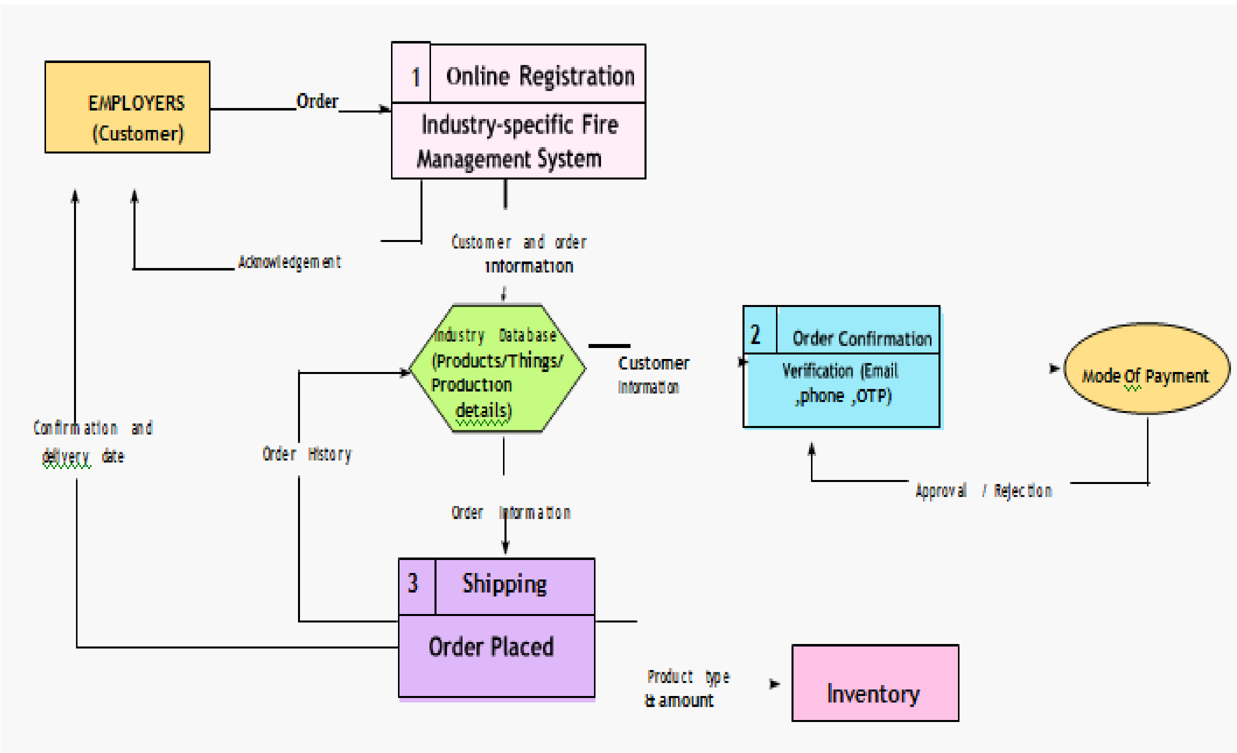
Non-Functional Requirements

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	Have a clear andself-explanatory manual.Easierto use. Easily accessible by everyone.
NFR-2	**Security**	Are inspected monthly by theFire Alarm Technician.Inspected and taggedby a contractor annually.
NFR-3	**Reliability**	Hardware requires a regular checking and service ..Software may be updated periodically. Immediate alertis provided in case of any system failure.
NFR-4	**Performance**	The equipment must have a gooduser interfaceIt should havea minimal energyrequirement It has to save lives of people and things

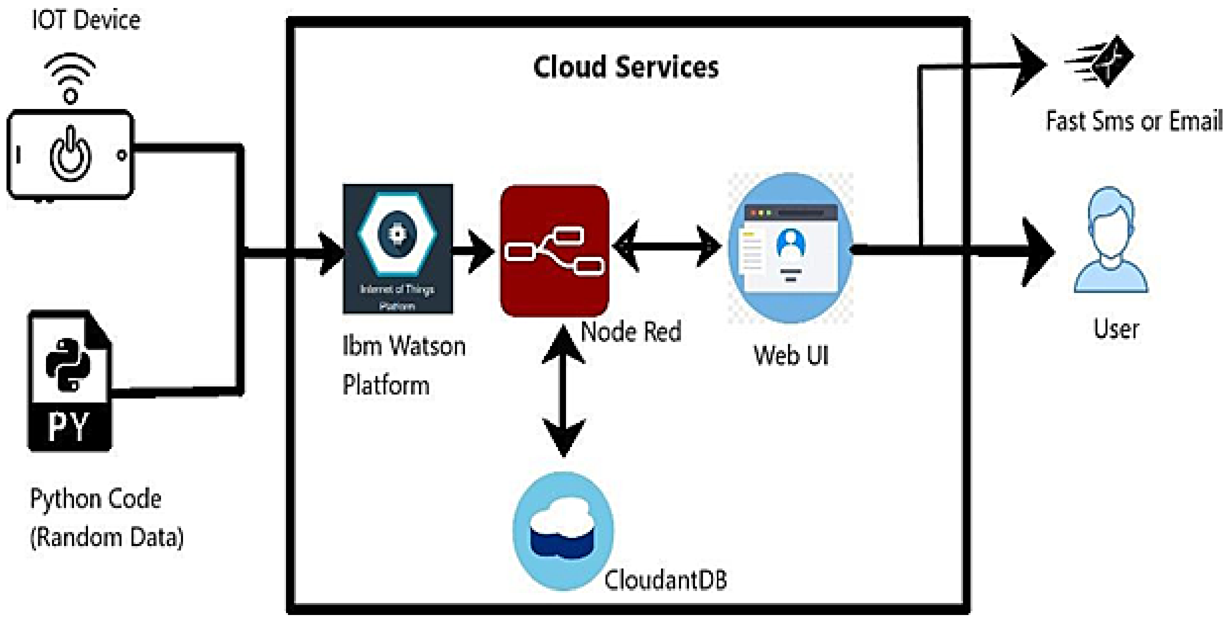
NFR-5	**Availability**	All the features will be available when the user requires. It depends on the need of the user and the customizationofthe user has done.
NFR-6	**Scalability**	The product has to cover all the space ofindustry irrespective of thesize or area.

5. PROJECT DESIGN:

Data Flow Diagrams



Solution & Technical Architecture:



User Stories

User Type	Functional requirement	User story number	User story/task	Acceptance criteria	Priority	Release
Customer (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 and password	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING:

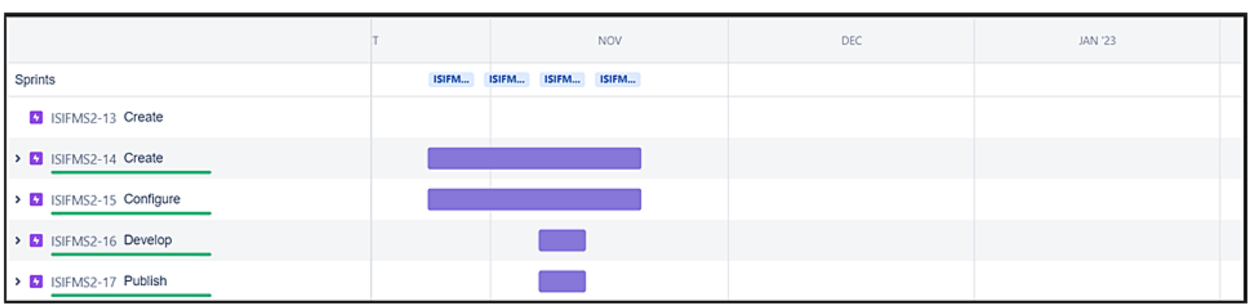
Sprint Planning& Estimation

Sprint	**Functional Requirement(Epic)**	**User Story Number**	**User Story/ Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Login	USN-1	As a customer , I might ensure login credential through gmail ease manner for the purpose of sending alert message to the owner.	2	High	Narmatha RCharan Sai Naveen R NaveenKumar J
Sprint-1	Registration	USN-2	As a user , I have to registered my details and tools details in a simple and easy manner in case of fire incident, this registered system sendsnotification to the industrialist.	2	High	Narmatha RCharan Sai Naveen R Naveen Kumar J
Sprint-2	Dashboard	USN-3	As a user, In case of Fire in the industry I need the sprinkler to spray water on the existing fire automatically.	2	Low	Narmatha RCharan Sai Naveen R Naveen Kumar J

Sprint-1	Dashboard	USN-4	As a user , I need to safeguard my properties as well as and it will be better to send alert messageto the firedepartment.	2	Medium	Narmatha RCharan Sai Naveen R Naveen KumarJ
Sprint-1	Dashboard	USN-5	As a user , Its good to have a IOT based system to extinguish the fire withouthuman presence.	2	High	Narmatha RCharan Sai Naveen R Naveen Kumar J

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

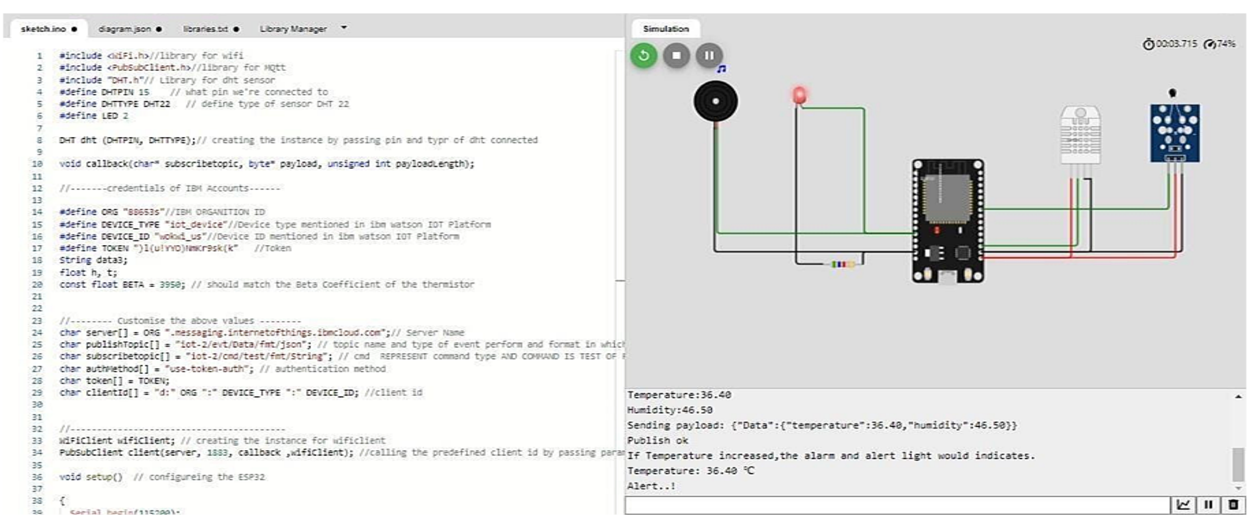
Reports From JIRA



BURNDOWN CHART

7. **CODING & SOLUTIONING:**

Code Explanation



Feature 1:

-Monitoring and detection of fire: The system can constantly monitor the environment for potential fire hazards and provide early warning in theevent of a fire.

-Automatic fire suppression: In the event of a fire, the system can automatically deployfire suppression systemssuch as sprinklers or fire extinguishers.

-Remote monitoring and control: The system can be monitored and controlled remotely, allowing for quick and effective response to fires.

-Integrated security: The system can be integrated with security systemsto provide additional protectionagainst fire hazards.

Feature 2:

-The cloud platform enables the iot based intelligent fire management system to remotely monitorand managefire safety devicesand systems in real-time. It also providesdata analysis and reporting capabilities to help improve fire safety.

-The fire detectionand suppression systemis fully automatedand cloud based. It uses advanced sensors to detect fire and notify the concerned personnel. The system is also equipped with intelligent video analytics that can identify the fire and its location. The fire management system is also equippedwith a fire suppression systemthat can automatically extinguish the fire.

MIT App Inventor is an intuitive, visual programming environment that allows everyone even childrento buildfully functional apps for smartphones and tablets. Those new to MIT App Inventor can have a simple first app up and running in less than 30 minutes. And what's more, our blocks-based tool facilitates the creation of complex, high-impact apps in significantly less time than traditional programming environments. The MIT App Inventor projectseeks to democratize software development by empowering all people, especially young people, to move from technology consumption to technology creation

Blocks-based coding programs inspire intellectual and creative empowerment. MIT App Inventor goes beyond this to provide real empowerment for kids to make a difference -- a way to achieve social impact of immeasurable value to their communities.

8. TESTING

Test Cases

The system is able to automatically detect and extinguish fires in the vicinity.

The system is able to automatically detectand report the presence of smoke in the area.

The system is able to automatically detectand report the presence of flames in the area.

The system is able to automatically shut off all gas and electrical suppliesin the event of a fire.

The system is able to automatically notifythe fire department in the event of a fire.

The system is able to automatically notifythe occupants of the building in the event of a fire.

The system is able to automatically evacuatethe building in the event of a fire.

User Acceptance Testing:

The IoT based Intelligent Fire Management System using ESP32 is tested by the user to check if it is working as expected. The user should be able to see the following:

-The system should be able to detect a fire and send an alert to the user.

-The systemshould be able to turn on the sprinklers automatically when a fire is detected.

-The systemshould be able to track the locationof the fire and provide updates to the user.

-The system should be able to provide information about the intensity of the fire.

9. RESULTS

Performance Metrics

There are many performance matricesthat can be used to evaluate the performance of an IoT-based intelligent fire management system. Some of the most important performance matrices include:

Response time: This is the time taken for the system to detect a fire and activate the fire suppression system.

Accuracy: This is the percentage of fires that are accurately detected by the system.

False positive rate: This is the percentage of times that the system incorrectly detects a fire.

False negative rate: This is the percentage of times that the system fails to detect a fire.

system availability: This is the percentage of time that the system is operational.

10. ADVANTAGES:

The Advantages of this Industry-Specific Intelligent Fire Managment systemare as follows

The user need not requireexpertise knowlege to control this system. This system is simple. The user can easily view thesensor values and take control actions.

The control actions are taken automatically.

If it is implemented in hardware, then the cost of implemention will be affordable.

As we are sensing the sensor values continously, any slight change in the environment is detected

This system is in User-Friendly format.

DISADVANTAGES:

The Disadvantage of this Industry-Specific Intelligent Fire Managment systemare as follows

This system will not be able to detect the orgin of fire.

This system will not providethe escape routeif there is fire outbreak.

If the industry has specific changesin the environment, then this system will gives falsealarm.

11. CONCLUSION:

An understanding and having Fire Managment system in the industry is of utmost importance.This project is a fire management system that can be user in the industry based on IOT.This system creates a simulation device cedentials in IBM WATSON IOT PLATFORM.In node- red,necessary nodes are installed and used.These nodes are installed and used.These nodes are deployed and the data is collected.In the event of fire, this system can issue sprinkleron,exhaust fan on.This remote user monitoring system can monitor the system status of each node in real time.Thissystem monitors the data continuously so that the any slight change in the environment can be easily detected.This ensures good control accracy .This Industry-Specific Intelligent Fire Managment ensures the protection of property, asset and the processes are cost eﬀective and the automatic measures are in control.

12. FUTURE SCOPE:

The future scope of Iot based intelligent fire management system is to develop a system that can automatically detect and extinguish fires. The system should be able to identify the type of fire and provide the appropriate response. It should also be able to send alerts to the authorities in case of a fire.There is no one-size- fits-all answer to this question, as the scope of an IoT-based intelligent fire management system will vary depending on the specific needs of the organization deploying it. However, some potential applications of such a system includeearly detection and notification of fires, automaticfire suppression, and remote monitoring and control of fire safety systems.

The future of IoT based intelligent fire management system looks very promising. With the help of IoT, the system will be able to monitor the fire situation in real time and take appropriate action accordingly. The system will also be able toautomatically detect the fire and send alerts to the concerned authorities.

13. APPENDIX:

Source Code:

```
//......Credentials For IBM ........


Organ
izati
on ID
7349
7z
De
vi
ce
Ty
pe
iot
_d
```

ev

ice

De

vi

ce

ID

12

34

Authenticati

on Method

use-token-

auth

Authenticati

on Token

12345678

//........Project SourceLink on Wokwi..........

Wokwi Link - https://wokwi.com/projects/347685130732569171

#include <WiFi.h>//library for wifi

#include <PubSubClient.h>//library

```
for MQtt#include "DHT.h"// Library

for dht sensor

#define DHTPIN 15   // what pin we'reconnected to

#define DHTTYPEDHT22 // define type of sensor

DHT 22#define LED 2


DHT dht (DHTPIN,DHTTYPE);// creatingthe instance by passing pin
and typrof dht connected


void callback(char* subscribetopic, byte* payload,
unsigned intpayloadLength);


//-------credentials of IBM Accounts------


#define ORG "88653s"//IBM ORGANITION ID

#define DEVICE_TYPE "iot_device"//Device type mentioned in
ibm watsonIOT Platform

#define DEVICE_ID "1234"//Device ID mentioned in ibm
watson IOTPlatform

#define TOKEN "12345678"

//TokenString data3;

float h, t;

const float BETA = 3950; // should match the Beta Coefficient of the thermistor
```

```
//-------- Customise the above values --------

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";// ServerName

char publishTopic[] = "iot-2/evt/Data/fmt/json"; // topic name and type of eventperform and format in whichdata to be send

char subscribetopic[] = "iot-2/cmd/test/fmt/String"; // cmd REPRESENTcommandtype AND COMMANDIS TEST OF FORMAT STRING

char authMethod[] = "use-token-auth"; // authentication

methodchar token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id




//_______________________________

WiFiClient wifiClient; // creating the instance for wificlient

PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefinedclient id by passing parameterlike server id,portand wificredential


void setup()  // configureing the ESP32


{

  Serial.begi

  n(115200);
```

```
  dht.begin();
  delay(10);
  Serial.print
  ln();
  wificonnec
  t();
  mqttconnec
  t();

  Serial.begin(960
  0);
  analogReadResol
  ution(10);

  pinMode(18,IN
  PUT);
  pinMode(14,O
  UTPUT);
  pinMode(12,O
  UTPUT);
}

void loop() // Recursive Function
{
```

```
h = dht.readHumidity();
t =
dht.readTemperatur
e();
Serial.print("Tempe
rature:");
Serial.println(t);
Serial.print("Humid
ity:");
Serial.println(h);

Publish
Data(t,
h);dela
y(1000
);
if
  (!client
  .loop())
  {
  mqttco
  nnect();
```

```
 }


//...............Analog Temperature Sensor...................


 int analogValue = analogRead(18);
 float celsius = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 /
298.15)+36.4;
 Serial.print("Tempe
 rature: ");
 Serial.print(celsius
 );

 Serial.println("
 ℃");
 Serial.print("Al
 ert..!");


 if(celsius >=
 35)
 digitalWrite(1
 4, HIGH);else
  digitalWrite(1
 4, LOW);
```

```
  delay(1000);

}


/*....................................retrieving to Cloud......................... */


void PublishData(float temp, float humid) {
  mqttconnect(); //function call for connecting to
  ibm


  /*
     creatingtheString in in form JSon to update the data to ibm cloud
  */


  String payload =
  "{\"Data\":{\"temperature\":";
  payload+= temp;
  payload+= ","
  "\"humidity\":";
  payload+= humid;
  payload += "}}";
```

```
  Serial.print("Sending
  payload: ");
  Serial.println(payload);



  if (client.publish(publishTopic, (char*)payload.c_str())) {
    Serial.println("Publish ok"); // ifit sucessfully uploaddata on the
cloud thenit will print publish ok in Serial monitoror else it will print
publish failed
    Serial.println("If Temperature increased,the alarm and alert light
wouldindicates. ");
  } else {
    Serial.println("Publish failed");
  }


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

```
    }


     initManaged
     Device();
     Serial.println
     ();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Conne
  cting to ");


  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to
establishthe connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi
```

```
  connected");
  Serial.println("IP
  address: ");
  Serial.println(WiFi.loc
  alIP());
}

void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
   // Serial.println((subscribetopic));
    Serial.println("subscribe to
    cmdOK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
  Serial.print("callback invokedfor topic:
  ");Serial.println(subscribetopic);
  for (int i = 0; i <
```

```
payloadLength; i++) {
Serial.print((char)payload[i]
);
  data3 += (char)payload[i];
}


Serial.println("data:
"+data3);
if(data3=="lighton")
{
Serial.println(dat
a3);
digitalWrite(LE
D,HIGH);


}


else
{
Serial.println(dat
a3);
digitalWrite(LE
```

```
D,LOW);

  }
data3="";

}
```

//..........Python Script for Random Outputs of Temperature and Humidity......

```
i
m
p
o
rt
ti
m
ei
m
p
o
rt
```

```
s
y
s
import
ibmiotf.applicati
onimport
ibmiotf.device
import random



#Provide your IBM Watson Device
Credentialsorganization = "bxobbs"
deviceType =
"b5ibm"
deviceId=
"b5device"
authMethod =
"token"
authToken =
"b55m1eibm"

# Initialize GPIO
```

```
def myCommandCallback(cmd):
    print("Command received: %s" %
    cmd.data['command'])
    status=cmd.data['command']
    if
        status=
        ="light
        on":
        print("l
        ed is
        on")
    else :
        print ("led is off")

    #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()


# Connectand send a datapoint "hello" with value "world" into the
cloud as anevent 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 }#printdata
        def myOnPublishCallback():
```

```
        print ("Published Temperature = %sC" % 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
clouddeviceCli.disconnect()


  "editor":
  "wokwi
  ",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 10, "left": -60.67,
"attrs": {} },
    {
      "type": "wokwi-led",
      "id": "led1",
```

"top": -109,
"left": -244.4,
"attrs": { "color": "red" }
},
{
"type": "wokwi-dht22",
"id": "dht1",
"top": -70.9,
"left": 157.2,
"attrs": { "temperature": "36.4", "humidity": "46.5" }
},
{
"type":"wokwi-ntc-
temperature-sensor","id":
"ntc1",
"top": -69.55,
"left": 253.55,
"rotate": 90,
"attrs": {}
},
{

"type":

"wokwi-

resistor","id":

"r1",

"top": 169.5,

"left": -190.59,

"attrs": { "value": "5600" }

},

{

"type": "wokwi-buzzer",

"id": "bz1",

"top": -118.83,

"left": -378.64,

"attrs": { "volume": "0.1" }

}

],

"connections": [

["esp:TX0", "$serialMonitor:RX", "", []],

["esp:RX0", "$serialMonitor:TX", "", []],

["dht1:GND", "esp:GND.1", "black", ["v0"]],

["dht1:SDA", "esp:D15", "green", ["v0"]],

["ntc1:GND", "esp:GND.1", "black", ["v0"]],

```
    [ "ntc1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "led1:C", "r1:1", "black", [ "v0" ] ],
    [ "r1:2", "esp:GND.2", "black", [ "v0" ] ],
    [ "led1:A", "esp:D14", "green", [ "v-0.86", "h89.56", "v199.46" ] ],
    [ "ntc1:OUT", "esp:D18", "green", [ "v0" ] ],
    [ "bz1:1", "esp:GND.2", "black", [ "v0" ] ],
    [ "bz1:2", "esp:D14", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:NC", "dht1:GND", "black", [ "v0" ] ]
  ]
}
```