INDUSTRY SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM

PROJECT REPORT

TEAM ID : PNT2022TMID52559

Submitted by

ASWIN KUMAR B	**CITC1905006**
MALSAWMTLUANGA V L	**CITC1905028**
MANOJ S	**CITC1905029**
RIYAZ AHAMED S	**CITC1905042**
SANDEEP T V	**CITC19 05 045**

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

COIMBATORE INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

CONTENTS

1. INTRODUCTION

1.1 PROJECT OVERVIEW

1.2 PURPOSE

LITERATURE SURVEY

2.1 EXISTING PROBLEM

Developed Intelligent Fire alarm system. [Hussam Elbehiery. J Am Sci 2012;8(8):1016-1024]. The primary purpose of fire alarm system is to provide an early warning of fire so that people can be evacuated & immediate action can be taken to stop or eliminate of the fire effect as soon as possible. Alarm can be triggered by using detectors or by manual call point (Remotely). To alert/evacuate the occupants siren are used. With the Intelligent Building of the rapid development of technology applications, commercial fire alarm market demand growth, the key is to use the bus system intelligent distributed computer system fire alarm system, although installation in the system much easier than in the past, but still cannot meet the modern needs, the installation costs of equipment costs about 33% ~ 70. The suggested technique in Fire alarm system used the addressable detectors units besides using the wireless connection between the detector in zones as a slave units and the main control unit as the master unit. The system shall include a control panel, alarm initiating devices, notification appliances, and the accessory equipment necessary for a complete functioning fire alarm system. In the wireless fire alarm, individual units are powered by primary & secondary batteries for the communication.

Research on Fire Alarm Computer Monitoring System in Fire Engineering Xiyang Feng and Chaofei Wang 2021 J. Phys.: Conf. Ser. 1915 042061 With the in-depth development and application of computer technology, the fire alarm computer monitoring system in fire protection engineering has become more and more essential equipment in modern life. With the support of network technology, the fire alarm monitoring system of fire protection engineering has formed a complete

system, including alarm monitoring, automatic fire control, fire linkage control, and fire data monitoring and analysis modules. This article mainly analyzes the fire alarm computer monitoring system in fire engineering.

Hamood Alqourabah,Amgad Muneer,Suliman Mohamed Fati in the paper titled"A Smart Fire Detrection using IoT Technology with Automatic Water Sprinkler", which employs different integrated detectors, such as heat, smoke, and flame. The signals from those detectors go through the system algorithm to check the fire's potentiality and then broadcast the predicted result to various parties using GSM modem associated with the system. To get real-life data without putting human lives in danger, an IoT technology has been implemented to provide the fire department with the necessary data. Finally, the main feature of the proposed system is to minimize false alarms, which, in turn, makes this system more reliable. The experimental results showed the superiority of our model in terms of affordability, effectiveness, and responsiveness as the system uses the Ubidots platform, which makes the data exchange faster and reliable. Poonam Sonsale, Rutika Gawas, Siddhi Pise, Anuj Kaldate in the paper" Intelligent Fire Extinguisher System"which proposes an adaptive fusion algorithm for fire detection,and uses a smoke sensor, flame sensor, and temperature sensor to detect fire incident. In reality, the phenomenon of the fire incident may have smoke, flame, and high temperature situations. However, these signals may happen simultaneously or sequentially. We develop an intelligent multi sensor based security system that contains a fire fighting system in our daily life. The security system can detect abnormal and dangerous situation and notify us. First, we design a firefighting system with extinguisher for the intelligent building. We design the fire detection system using sensors in the fire fighting system, and program the fire detection and fighting procedure using sensor

based method. Finally, we implement the fire detection system using fire fighting system.

GPS-based fire detection system (Global Positioning System) and SMS Gateway A Aryanti, I Mekongga and R S Dewi et al 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1108 012023 This research aims to produce a GPS-based fire detection system (Global Positioning System) and SMS Gateway. The benefits of this detection system can detect early fire occurrence based on the detection of temperature conditions by accommodating the nature of the fire and able to detect any rise in temperature caused by the existence of the fire. This detection system must also be able to read any smoke produced by a fire. To realize the system, required sensors capable of reading the temperature and smoke. The Arduino Uno microcontroller is the brain control system of the system. At a temperature of> 35 C, the system will activate the DHT 11 and MQ 2 sensors that detect smoke> 50 ppm from fire. The system will activate Buzzer as a warning in the form of the next alarm sound Global Positioning System (GPS) will provide information in the form of coordinates of the location of the point of fire through GSM SIM900 Module Short Message Service (SMS) to the user. The results obtained mq2 = 128 ppm and temperature value = 38 ° C and GPS data with latitude of -3.04798388 and a longitude of 104.78263092. From the data it is seen that the mq2 value reaches> 50ppm and the temperature value reaches> 35 ° C, and the detector outputs buzzer sound and warning notification of coordinate point in the form of SMS containing the message "FIRE available" with the coordinates of the location of the fire detected by GPS.

IoT Based Fire Detection System Using Machine Intelligence 4 authors, including Arun Rajesh DOI: 10.13140/RG.2.2.18979.99365 Fire alarms play an important role in residential safety work. While the Fire Services are the first line of

defence against fire accidents, they are heavily under-resourced and lack adequate manpower. After analysing the needs of the Indian Fire Department, this paper proposed a IoTarchitecture based fire alarm system that alerts the owner and fire station of a fire outbreak. This paper also uncovers the ideal conditions to set off the fire alarm based on the temperature, humidity and the nature of gases present in the environment using the decision tree algorithm. Several cases are recorded for experimentation and training. Results show 91.15% accuracy in detecting fire.

IOT Based Fire Detection System Rashmi Vinod Patil1, Sayali Fakira Jadhav, Kaveri Sitaram Kapse, Prof. M. B. Thombare, Prof. S. A. Talekar Article · July 2021 DOI: 10.48175/IJARSCT-1681 Fire Detection Systems are now widely used in various safety and security applications. The major amount of fire starts due to the electric short circuit. It leads to damage to property and also loss of life. To avoid that or to minimize the damage caused by fire outbreaks due to electric short circuits an IoT technology is used to control such a kind of risk. Traditional fire detection systems are not that effective and quick to alert the owner about fire, in case no one is present on the location. To overcome this problem in this paper we present the design and development of IoT based Fire Detection System. A system that combines qualities for fire, temperature and smoke detection, sending alert Text Message about the fire to the user along with onsite alarm(buzzer), updating temperature, humidity and smoke on ThingSpeak cloud every 15 seconds, and it also moves manually with the help of Android Application. The Fire Detection System consists of four main parts: Multiple sensors, communication system (Bluetooth, GSM, NodeMCU), motion planning (Manual patrolling), and Android application for manual patrolling of the system. This Fire Detection system can be used in college, school, office, and industry for safety purposes.

GSM based smart fire and high-temperature detection system Ravindra Koggalage, Manjula Welihindaand Hasitha Nuwan Article in ITEGAM- Journal of Engineering and Technology for Industrial Applications (ITEGAMJETIA) · January 2021 This research refers to an Arduino and Global System for Mobile (GSM) based system for efficient detection of fire hazards. This project's purpose is industrial and domestic safety, and the primary concern is to avoid the fire hazards that occur to the employees and the properties inside the buildings. As a solution, a smart fire and high-temperature detection system is design using GSM technology, smoke/temperature sensors, and Arduino technology. A smoke sensor is used to detect the smoke from the fire and a temperature sensor is used to detect temperature increase inside the building. In event of a fire, an alert message will be sent to the user via short message service (SMS) via the GSM module. Furthermore, when a fire is detected, a signal will be sent to the main power supply circuit breaker via a microcontroller and then the power supply of the particular building will shut down. Results from the test are documented and discussed in this paper. This system helps users to respond immediately to the situation and so improve their safety by protecting their lives and the properties from a disaster.

2.2 REFERENCES

1. Developed Intelligent Fire alarm system. [Hussam Elbehiery. J Am Sci 2012;8(8):1016- 1024].

2. Research on Fire Alarm Computer Monitoring System in Fire Engineering Xiyang Feng and Chaofei Wang 2021 J. Phys.: Conf. Ser. 1915 042061.

3. Hamood Alqourabah,Amgad Muneer,Suliman Mohamed Fati in the paper titled"A Smart Fire Detrection using IoT Technology with Automatic Water Sprinkler".

4. Poonam Sonsale, Rutika Gawas, Siddhi Pise, Anuj Kaldate in the paper" Intelligent Fire Extinguisher System".

5. GPS-based fire detection system (Global Positioning System) and SMS Gateway A Aryanti, I Mekongga and R S Dewi et al 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1108 012023.

6. IoT Based Fire Detection System Using Machine Intelligence 4 authors, including Arun Rajesh DOI: 10.13140/RG.2.2.18979.99365 .

7. IOT Based Fire Detection System Rashmi Vinod Patil1, Sayali Fakira Jadhav, Kaveri Sitaram Kapse, Prof. M. B. Thombare, Prof. S. A. Talekar Article · July 2021 DOI: 10.48175/IJARSCT-168.

8. GSM based smart fire and high-temperature detection system Ravindra Koggalage, Manjula Welihindaand Hasitha Nuwan Article in ITEGAM- Journal of Engineering and Technology for Industrial Applications (ITEGAM-JETIA) · January 2021

2.3 PROBLEM STATEMENT DEFINITION

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

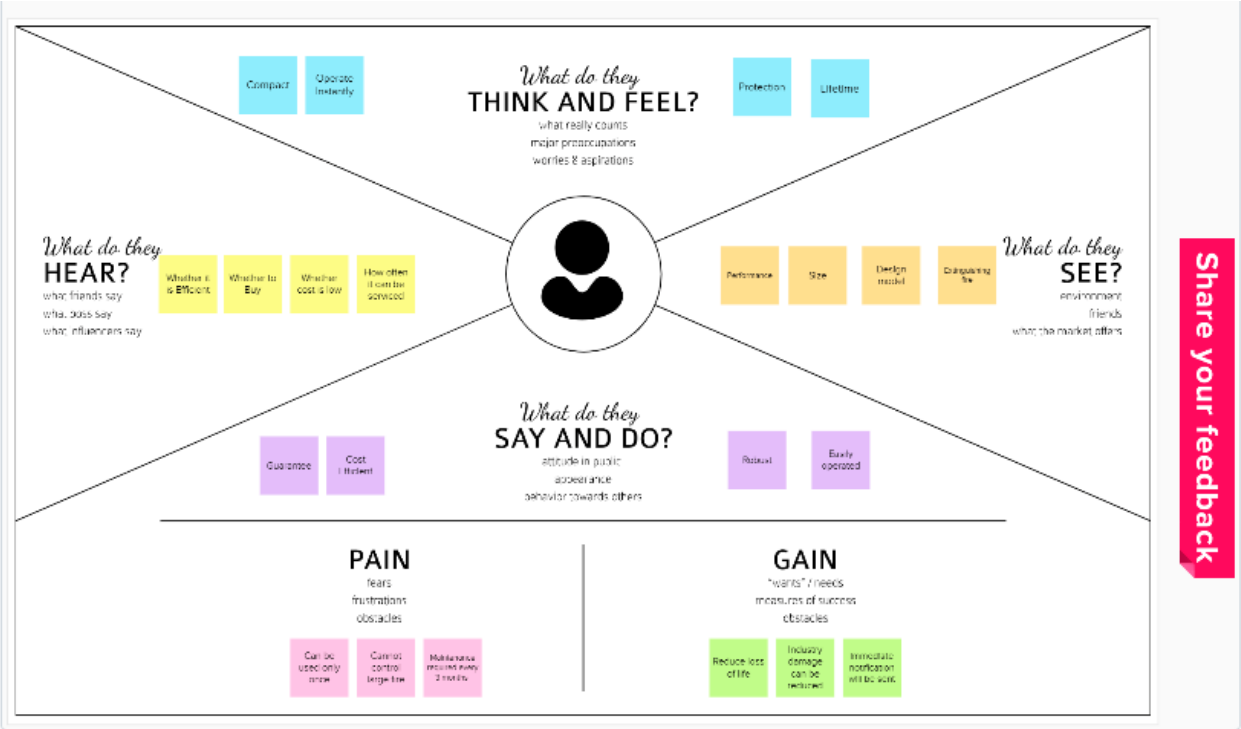
A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll

also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS-1	I am an industry owner	I'm trying to safeguard my employees and properties from fire	But I don't have a proper fire management system that detects fire immediately	Because employees, can't detect fire immediately, Meanwhile, disasters may happen	As a result, I need a fire management system that will detect the fire immediately, and the alarm and the sprinkler will be turned on simultaneously
PS-2	I am an industry owner	I'm trying to get a fire management system that monitors the temperature and gas values in my industry	But I don't have such a system that makes me aware of the temperature and gas values in my industry	Because I need to know the temperature and gas values in the surroundings to keep me more aware of the fire	As a result, I need a fire management system that keeps me more aware

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:



3.2 Ideation & Brainstorming:

3.3 Proposed Solution:

S.NO	PARAMETER	DESCRIPTION
1	Problem Statement (Problem to be solved)	On October 20, 2016 in Tamilnadu a major fire broke out in the huge cracker manufacturing hub in Sivakasi. Many people lost their lives in this accident. In cracker industries the chemicals used for manufacturing fireworks are highly sensitive to friction, impact, heat and static electricity. These friction and impact causes fire accidents. The fire accidents or blasts also occur in god owns either due to sparks from electrical fittings or from the impact stimuli generated during loading and unloading of boxes containing fireworks. These fire accidents cause great loss to the industry and also to the lives of the people working in the industries. So, to overcome this problem we have proposed a solution that uses sensors to detect the fire before it causes damage, sprinklers are used to control the fire and a fire alarm is used to alert the workers about the fire breakage.
2.	Idea / description	In the proposed model, a gas sensor, flame sensor and temperature sensors are used for the detection of fire. Gas Sensor Gas sensors (also known as gas detectors) are electronic devices that detect and identify different types of gasses. They are commonly used to detect toxic or explosive gasses and measure gas concentration.

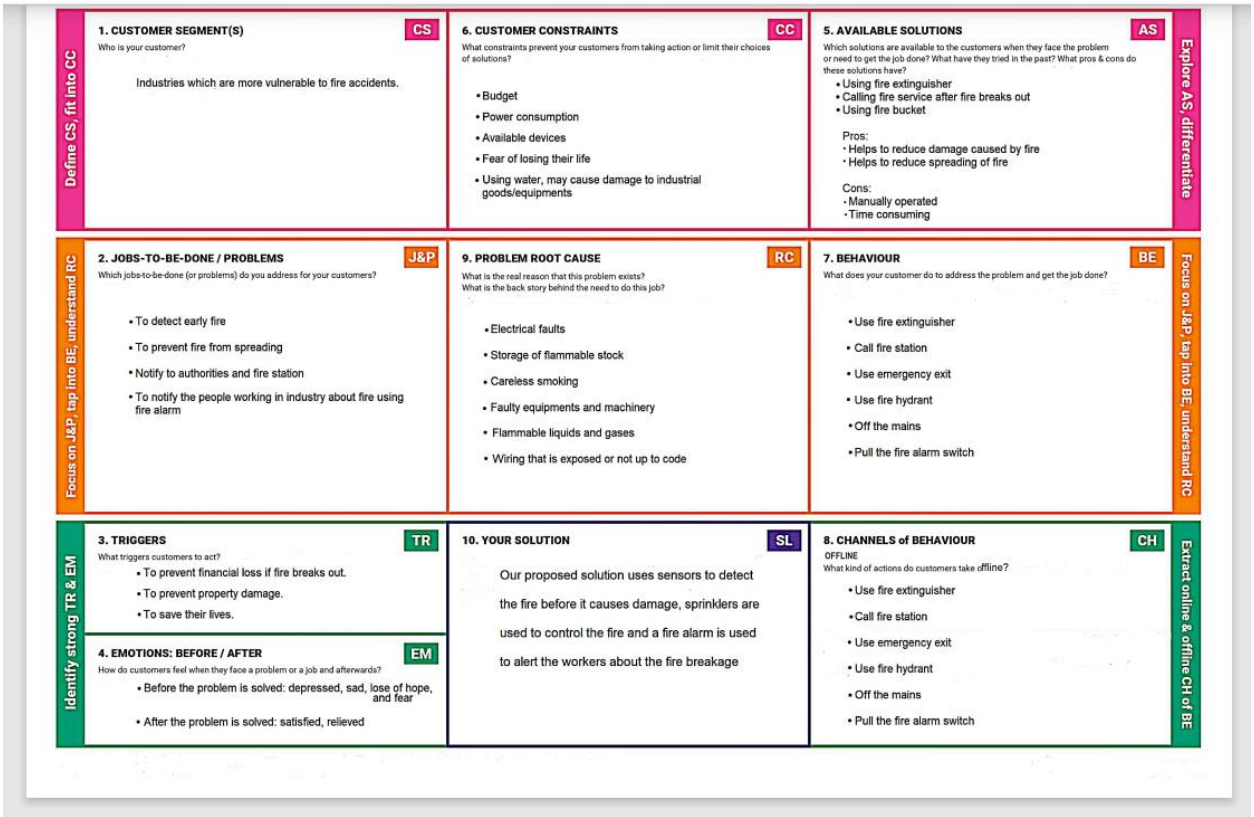
		Gas sensors are employed in factories and manufacturing facilities to identify gas leaks, and to detect smoke and carbon monoxide in homes. Gas sensors vary widely in size (portable and fixed), range, and sensing ability. They are often part of a large Embedded systems, such as hazmat and security systems, and they are normally connected to an audible alarm or interface. Because gas sensors are constantly interacting with air and other gasses, they have to be calibrated more often than many other types of sensors. In general gas sensors have the potential to detect all fires because every fire is emitting gas and an according fire detector is not dependent from the release of heat or smoke. Flame sensor. The flame sensor detects the presence of fire or flame based on the Infrared (IR) wavelength emitted by the flame. It gives logic 1 as output if a flame is detected; otherwise, it gives logic 0 as output.
3.	Novelty / Uniqueness	A fire detection system uses a smoke detector to detect a fire before it actually starts. An effective fire detection system eliminates damage by ensuring that a fire can be prevented before it even starts. A fire detector may also have a direct connection to an alarm monitoring centre. The

		smart fire management system includes a Gas sensor, Flame sensor and temperature sensors to detect any changes in the environment. Based on the temperature readings and if any gases are present then the alarm is triggered. If any flame is detected the sprinklers will be switched on automatically. Emergency alerts are notified to the authorities and fire station
4.	Social Impact / Customer Satisfaction	Fire management system provides an early warning of fire so that people can be evacuated and immediate action can be taken to stop or eliminate the fire effect as soon as possible. If fire is detected immediate notification will be sent to authorities and fire stations. The number one reason to install a fire alarm is to make the building safe for your employees, customers, and tenants. A combination of smoke and heat detectors, sirens and bells, and strobe lights detect fires and alert building occupants, giving them ample time to evacuate in an orderly fashion. Using automatic fire sprinklers protects the environment while further verifying that they reduce property damage and protect lives. It reduces financial loss in industries.
5.	Business Model (Revenue Model)	Customer segment This alarm system is designed for industries. Its purpose is industrial safety, and

		the primary concern is to avoid the fire hazards that occur to the employees and the properties inside the buildings. Industrial buildings shall include any building in which products or materials of all kinds and properties are fabricated, assembled, manufactured or processed, for example, assembly plants, industrial laboratories, dry cleaning plants, power plants, generating units, pumping stations, laundries, buildings or structures in gas plants, refineries, dairies and sawmills etc. Customer relationship The industry premises will be inspected and after a full assessment, recommendations will be made for the location specifically to ensure maximum safety without excess cost to the business. After installation the following will be provided in the premises. • Owner's manual and manufacturer's instructions covering all system equipment. • Operator instructions for basic system operations. • A detailed description of routine maintenance and testing as required and recommended, including: Listing of the individual system components that require periodic testing and maintenance. • Step-by-step instructions detailing the requisite testing and

		maintenance procedures, and the intervals at which these procedures need to be performed, for each type of device installed. • A testing and maintenance schedule. • Detailed troubleshooting instructions. • A service directory that includes a list of the names and telephone numbers of those who provide service for the system.
6.	Scalability of the Solution	The proposed model can be used in textile industries, paper industries, automobile industries, mining industries, cracker industries, cement industries etc.

3.4 Problem Solution fit:



4. REQUIREMENT ANALYSIS:

4.1 Functional requirement:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Rapid Detection of fire	The system must be able to detect fire rapidly
FR-2	Automatic, Accurate, Dynamic Aiming	The system must be able to quickly aim a large volume of water directly onto the flames, and it must be able to dynamically follow the flames if the fire grows or spreads
FR-3	3D location	The system must be able to accurately determine the three-dimensional position and volume of the flames in 3-dimensional space
FR-4	Automation and Autonomy	The system must be able to activate and function completely autonomously, without any external network or power and any human intervention
FR-5	Web server	The system must have a web server for system monitoring and allow for remote control by designated persons
FR-6	Cloud server	Cloud servers allows us to store information on the cloud and access this information using an internet connection. As the cloud provider is responsible for providing security, so they offer various backup recovery application for retrieving the lost data.

4.2 Non-Functional requirement:

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

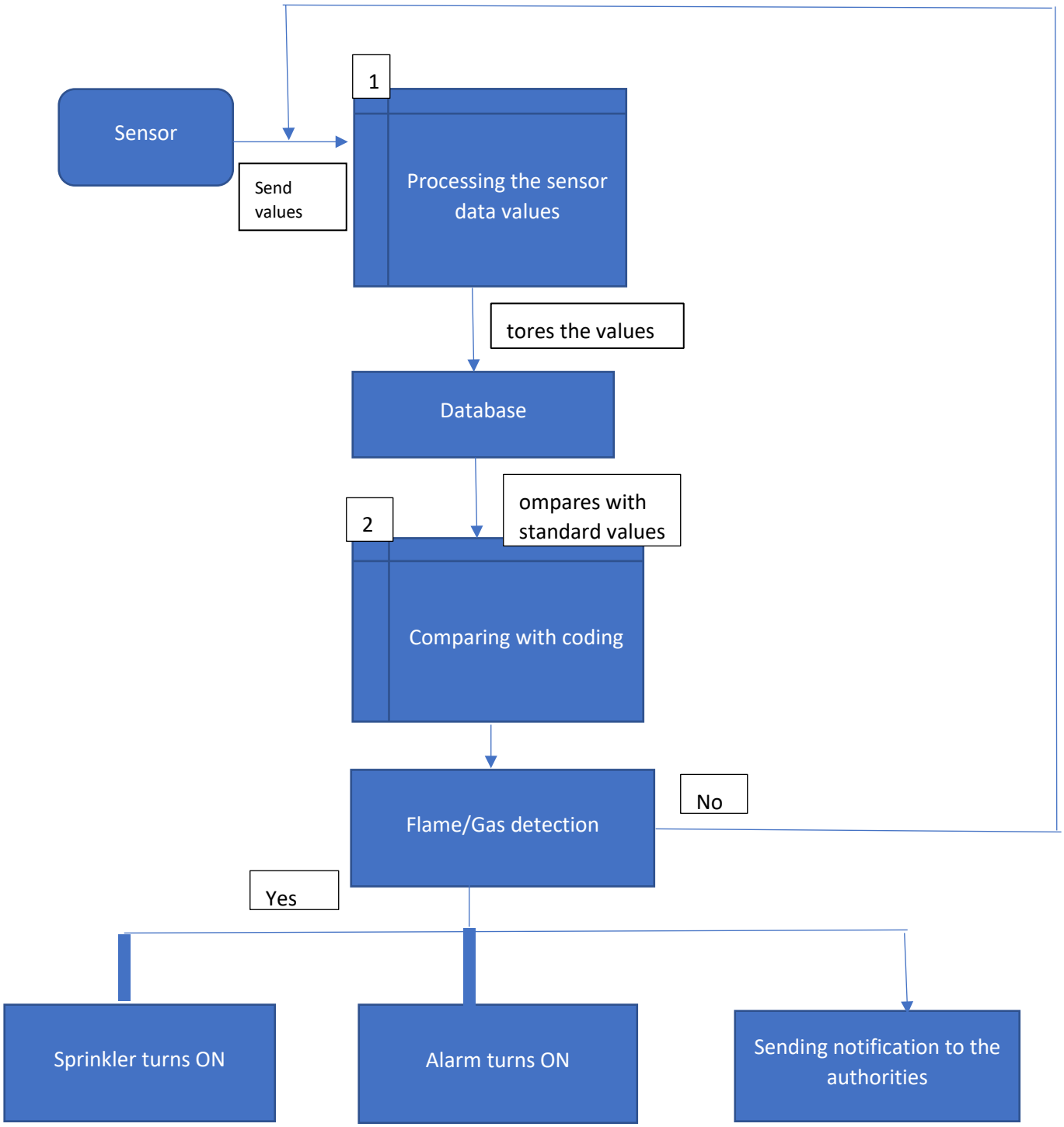
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It is completely automated. No need to manually remove any pin like a fire extinguisher. Instead, when the flame is detected, the sprinkler is turned on immediately and when a gas sensor detects any gases, an alarm is sent immediately and notifications are sent to the authorities. It is easier to use the fire management system.
NFR-2	**Security**	According to the testing and maintenance schedule, frequent tests are done to secure the fire management system. Fire management systems should be discharged, disassembled, and inspected annually. Mock drills should be conducted periodically. It should be checked whether it includes all the fire safety standards.
NFR-3	**Reliability**	This is the highest quality and most innovative fire sprinklers and special systems on the market;
		distributes a full line of best-in-class system components; and backs it up with premier customer service.

NFR-4	**Performance**	All the minimum durations of operations are here decided for every fire management system, according to the value of the flame sensor, gas, and temperature sensor. The emission of sprinklers shall start within a few seconds since the flame is detected and in case of any gas is detected, an alarm is turned on within a few seconds.
NFR-5	**Availability**	The fire management systems were effective in extinguishing fires 95% of the time. A new installation of the system shall be available for first-time use within 24 hours of the start of the installation.
NFR-6	**Scalability**	This model is not only used for small industries but it can also be used in large industries and buildings with proper infrastructure and technology.

5. PROJECT DESIGN:

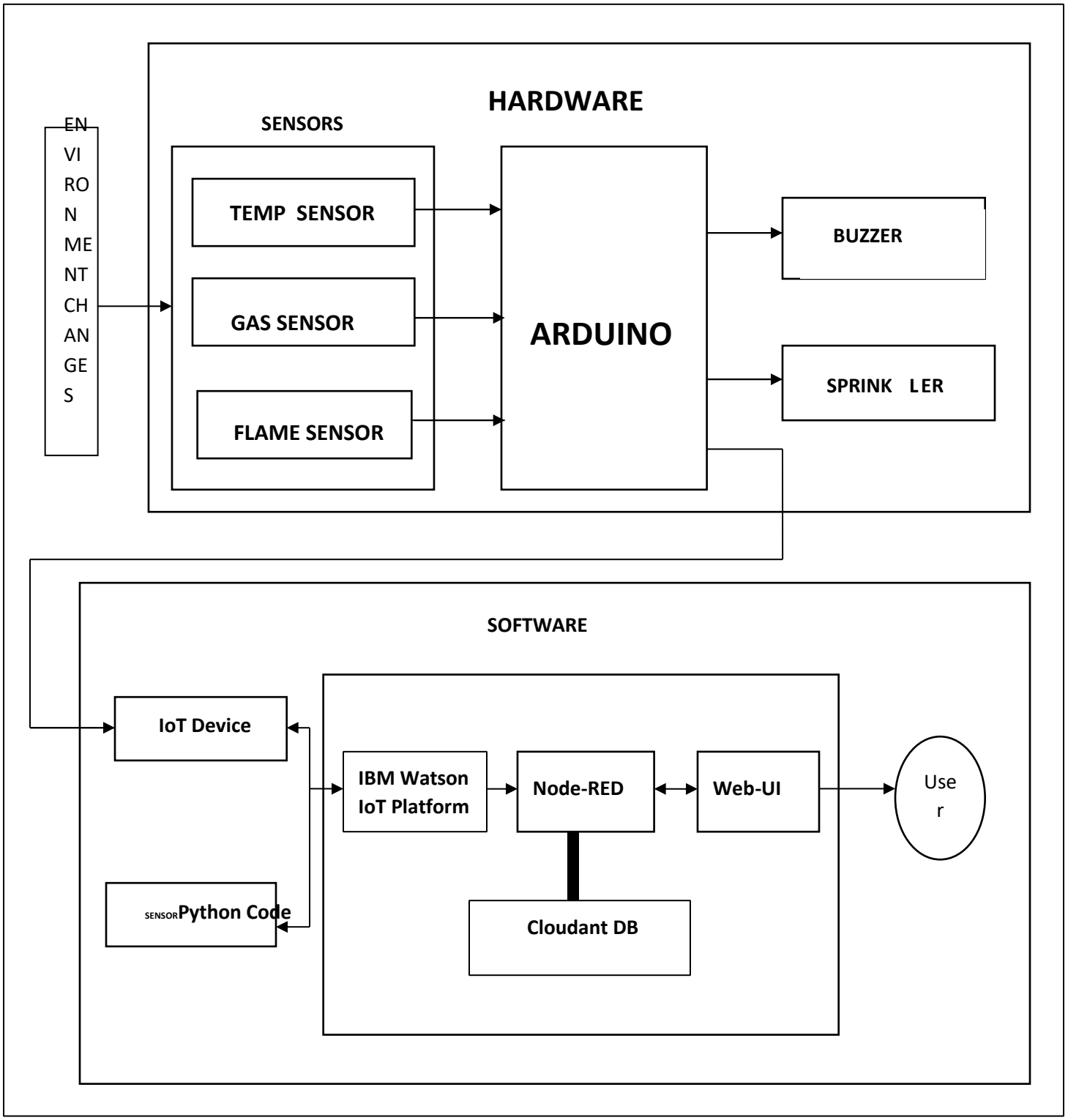
5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data centers and leaves the system, what changes the information, and where data is stored.

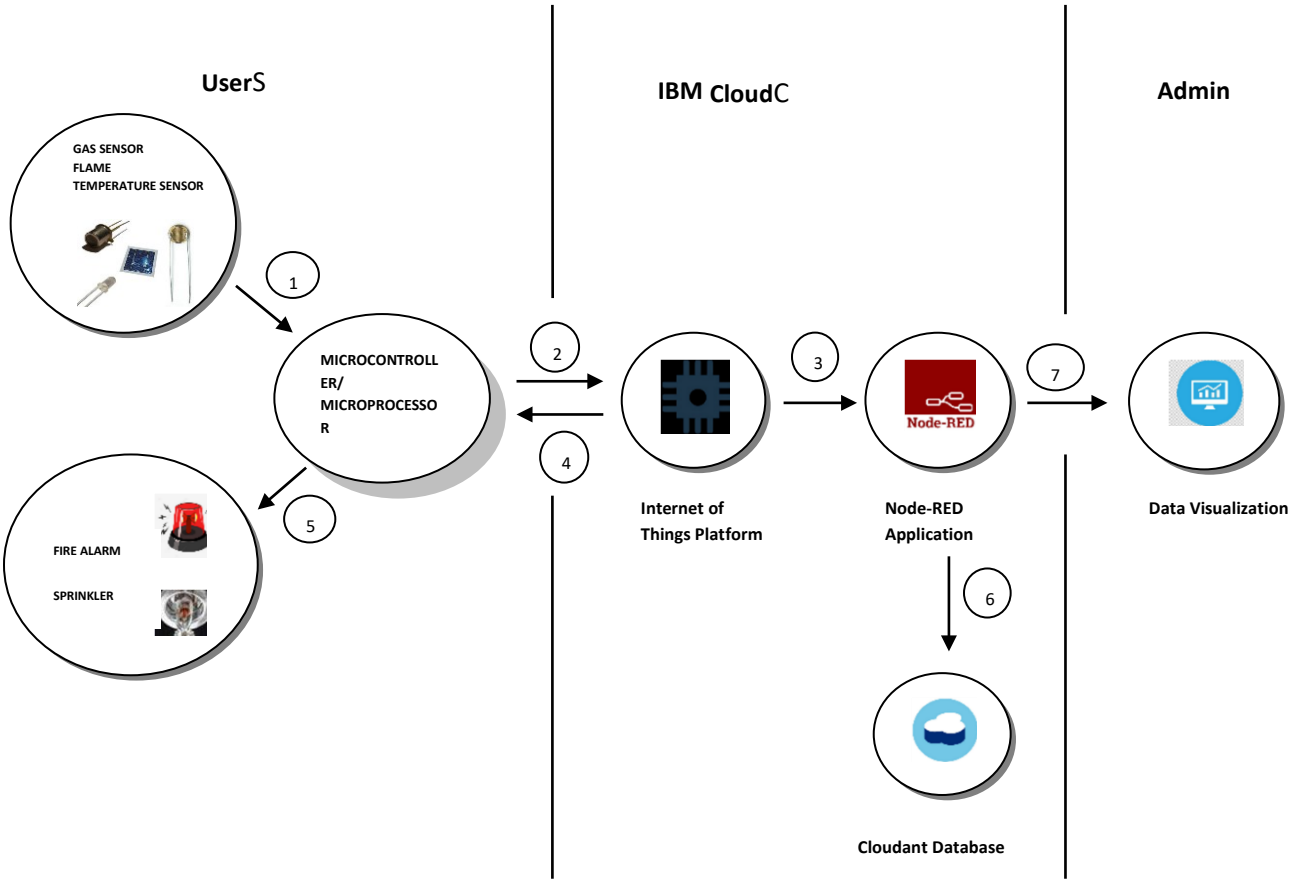

Sensor
Send values
1
Processing the sensor data values
tores the values
Database
ompares with standard values
2
Comparing with coding
Flame/Gas detection
No
Yes
Sprinkler turns ON
Alarm turns ON
Sending notification to the authorities

5.2 Solution & Technical Architecture:

Solution Architecture:



Technical Architecture:



5.3 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
Customer (Industrial user)	Rapid Detection of fire	USN-1	As a user, I need rapid detection of fire	I can safeguard my properties and employees	High	Sprint-1
Customer (Industrial user)	3D location	USN-2	As a user, I require a 3D location	Fire can be detected accurately	Medium	Sprint-1
Customer (Industrial user)	Automation and autonomy	USN-3	As a user, I need automation and autonomy	Human interaction can be avoided	High	Sprint-2
Customer (Industrial user)	Web server	USN-4	As a user, it's essential to have a web server	I can monitor and allow for remote control by designated persons	Medium	Sprint-4
Customer (Industrial user)	Automatic, Accurate, Dynamic Aiming	USN-5	As a user, I require automatic, accurate, and dynamic aiming	Aim a large volume of water directly at the flames, and dynamically follow the flames if the fire grows	High	Sprint-2
Customer (Industrial user)	Cloud server	USN-6	As a user, I need a cloud server	I can store the data securely	Low	Sprint-3
Customer (Industrial user)	Alarm	USN-7	As a user, I need an alarm	I can be safe before the fire spreads	High	Sprint-2
Customer (Fire station)	Notification	USN-8	As a user, I need a notification about the fire	I can know about the nearby fire breakage	Low	Sprint-3

6. PROJECT PLANNING & SCHEDULING:

6.1 Sprint Planning & Estimation:

6.2 Sprint Delivery Schedule:

6.3 Reports from JIRA:

7. CODING & SOLUTIONING:

8 TESTING

8. SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

8.1 TYPES OF TESTS

8.1.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at

component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

8.1.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

8.1.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

8.1.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

8.1.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

8.1.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

8.2 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

8.2.1 Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

8.2.2 Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

8.2.3 Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

8.3 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

8.4 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

9 RESULTS:

10 ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

- Photosensitivity is high
- Response time is fast
- Simple to use
- Sensitivity is adjustable
- Detection angle is 600,
- It is responsive to the flame range.
- Accuracy can be adjustable
- Operating voltage of this sensor is 3.3V to 5V

DISADVANTAGES:

- PCB occupied more complex process

APPLICATIONS

- Hydrogen stations
- Industrial heating
- Fire detection

- Fire alarm
- Fire fighting robot
- Drying systems
- Industrial gas turbines
- Domestic heating system
- Gas-powered cooking devices

11 CONCLUSIONS:

We have presented forest fire detection in IOT projects that use wireless sensor networks in their architectures to measure and transferring useful data. The role of a sensor node is to sense the environment, transferring and exchange sensory data with other nodes in the area. The industrial application of wireless sensor networks are in digital transmission to monitor temperature and humidity in the forest in a more timely and precise way, we pointed out unique advantages of safety in data transmission, flexibility in building the network, and low cost and energy requirements for a forest fire monitoring system based on a IOT wireless sensor technology that we designed.

12 FUTURE SCOPES:

Will use in real time for further Real time work.

13 APPENDIXES:

Source Code:

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

#include <PubSubClient.h>//library for MQtt

#include "DHT.h"// Library for dht11

#include <cstdlib>
```

```
#include <time.h>

#include <mjson.h>


#define DHTPIN 15     // what pin we're connected to

#define DHTTYPE DHT22   // define type of sensor DHT 11



DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected


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


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


#define ORG "3e40vw"

#define DEVICE_TYPE "sample-device"

#define DEVICE_ID "8007"

#define TOKEN "987654321"


String data3 = "";

String accidentstatus ="";
```

```
String sprinkstatus = "";

float temp =0;

bool isfanon = false;

bool issprinkon = false;

bool cansprinkoperate = true;

bool canfanoperate = true;

bool cansentalert = false;

int gas = 0;

int flame = 0;

int flow = 0;

long int cooldown= 600;









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

char publishTopic[] = "iot-2/evt/data/fmt/json";

char subscribetopic[] = "iot-2/cmd/command/fmt/String";

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

char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```
//---------------------------------------

WiFiClient wifiClient; // creating the instance for wificlient

PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like server id,portand wificredential



void setup()// configureing the ESP32

{

  Serial.begin(115200);

  dht.begin();

  //if real gas sensor is used make sure the senor is heated up for acurate readings

  /*

    - Here random values for readings and stdout were used to show the

      working  of the devices as physical or simulated devices are not

      available.

  */



  delay(10);
```

```
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()
{



  temp = dht.readTemperature();
  //setting a random seed (only for random values not in real life scenarios)
  srand(time(0));

  //initial variable activities like declaring , assigning
  gas = rand()%400;
  int flamereading = rand()%1024;
  flame = map(flamereading,0,1024,0,1024);
  int flow = ((rand()%100)>50?1:0);
```

```
//find the accident status 'cause fake alert may be caused by some mischief activities
if(temp < 45 ){
  if(flame > 650 ){
    accidentstatus = "Need Auditing";
    if(canfanoperate)
      isfanon = true;
    else
      isfanon = false;
    issprinkon = false;
  }
  else if(flame <= 10){
    accidentstatus = "nothing happened";
    isfanon = false;
    issprinkon = false;
  }

}else if(temp >= 45 && temp <= 55 ){
  if(flame <=650 && flame >100  ){

    if(cansprinkoperate)
```

```
    issprinkon = true;
   else
    issprinkon = false;
   accidentstatus = "moderate";
   if(gas > 160 && canfanoperate ){
    isfanon = true;
   }
   else{
    isfanon = false;
   }
  }
 else if(flame <= 100 && flame > 10){
  if(cansprinkoperate)
    issprinkon = true;
   else
    issprinkon = false;
   isfanon = false;
   accidentstatus = "moderate";
  }

}else if(temp > 55){
```

```
if(flame > 650){
  gas = 500 + rand()%500;
  accidentstatus = "severe";
  if(cansprinkoperate)
    issprinkon = true;
  else
    issprinkon = false;
  if(canfanoperate)
    isfanon = true;
  else
    isfanon = false;
}
else if(flame < 650 && flame > 400 ){
  gas = 300 + rand()%500;
  accidentstatus = "severe";
  if(cansprinkoperate)
    issprinkon = true;
  else
    issprinkon = false;

  if(canfanoperate)
```

```
      isfanon = true;
    else
      isfanon = false;

  }
}
else {
  accidentstatus = "Need moderate Auditing";
  isfanon = false;
  issprinkon = false;
}


if(issprinkon){
  if(flow){
    sprinkstatus = "working";
  }
  else{
    sprinkstatus = "not working";
  }
}
```

```
  else if(!issprinkon){
    sprinkstatus = "ready";
  }
  else {
    sprinkstatus = "something's wrong";
  }


  PublishData(temp,gas,flame,flow,isfanon,issprinkon);

  //a cooldown period is set as the values and situations are random in real life sceanarios the time can be reduced or neclected
  if(accidentstatus=="severe" && cooldown >= 600){
    cooldown = 0;
    sendalert();
    PublishData(temp,gas,flame,flow,isfanon,issprinkon);
    cansentalert = false;

  }

  if(cooldown > 999999){
```

```
    cooldown = 601;
  }




  delay(1000);
  ++cooldown;
  if (!client.loop()) {
    mqttconnect();
  }
}



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

void PublishData(float temp, int gas ,int flame ,int flow,bool isfanon,bool issprinkon) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
```

```
*/
String payload = "{\"temp\":";
payload += temp;
payload += "," "\"gas\":";
payload += gas;
payload += "," "\"flame\":";
payload += flame;
payload += "," "\"flow\":";
payload += ((flow)?"true":"false");
payload += "," "\"isfanon\":";
payload += ((isfanon)?"true":"false");
payload += "," "\"issprinkon\":";
payload += ((issprinkon)?"true":"false");
payload += "," "\"cansentalert\":";
payload += ((cansentalert)?"true":"false");
payload += "," "\"accidentstatus\":";
payload += "\""+accidentstatus+"\"";
payload += "," "\"sprinkstatus\":";
payload += "\""+sprinkstatus+"\"";
payload += "}";
```

```
  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish ok in Serial monitor or else it will print publish failed
  } 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);
    }
```

```
    initManagedDevice();
    Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);
  while (WiFi.status() != WL_CONNECTED) {
    delay(100);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
```

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

//handles commands from user side
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {

    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
```

```
const char *s =(char*) data3.c_str();
double pincode = 0;

if(mjson_get_number(s, strlen(s), "$.pin", &pincode)){
  if(((int)pincode)==67993){
     const char *buf;
     int len;

     if (mjson_find(s, strlen(s), "$.command", &buf, &len))  // And print it
     {

       String command(buf,len);
       if(command=="\"cantfan\""){
         //this works when there is gas sensor reads high value and if there should be a
         //manual trigger else it will be automate
         canfanoperate = !canfanoperate;
       }
       else if(command=="\"cantsprink\""){
         cansprinkoperate = !cansprinkoperate;
       }else if(command=="\"sentalert\""){
         //this works when there is accident status is severe and if there should be a
```

```
            //manual trigger else it will be automate
            resetcooldown();
          }
        }




    }
  }

  data3="";
}

void resetcooldown(){
  cooldown = 0;
}
//sent alert request to node-red
void sendalert(){
  cansentalert = true;
  cooldown = 0;
}
```