JP COLLEGE OF ENGINEERING

Nalaiya Thiran

executed by

Industry-specific intelligent fire management system

Project ID : PNT2022TMID47460

Team Lead : Sneha K

Team Members : Thanalakshmi M

Ramya R

Santhiya V

CONTENTS

1.INTRODUCTION

1.1 Project Overview

- 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 the exhaust fans are powered ON.
- If any flame is detected the sprinklers will be switched on automatically.Emergency alerts are notified to the authorities and Fire station.

1.2 Purpose

The purpose of the system is :

● To prevent life losses , assests damage and uncontrollable spread of fire. ● To ensure the safety of 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

2.1 Existing problem

The existing problems of the system are:

- 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 component non-avaliabilty or due to being unsupported by the manufacturer.

- Structural changes : The structure of the hospital changes over time. The fire management system should be easily able to upgrade and adaptable to the changing structure.
- Evaculation and fire stratergy : The alert and the control measures are taken immediately, so that the building can be completely evaculated.
- System performance changes within specific environments : The industry will have unique or specified condition at some time. The major problem caused is the false fire alarm.

2.2 References

[1] Gazi weldesyase, Bahta G/meskel, Mekonen Abreha, Solomon Baynes, “GSM Based Fire and Smoke Detection and Prevention System”, on 08/10/2010, Adigrat, Tigray, Ethiopia. [2] May Zaw Tun, Htay Myint, “Arduino based Fire Detection and Alarm System Using Smoke Sensor”, Volume 6, Issue 4, on April – 2020, Myanmar.

[3] Nitin Galugade, Mahesh Jakka, Devika Nair, Madhur Gawas, “Fire Monitoring and Controlling System based on Iot”, 2020, Mumbai, India.

2.3 Problem Statement Definition

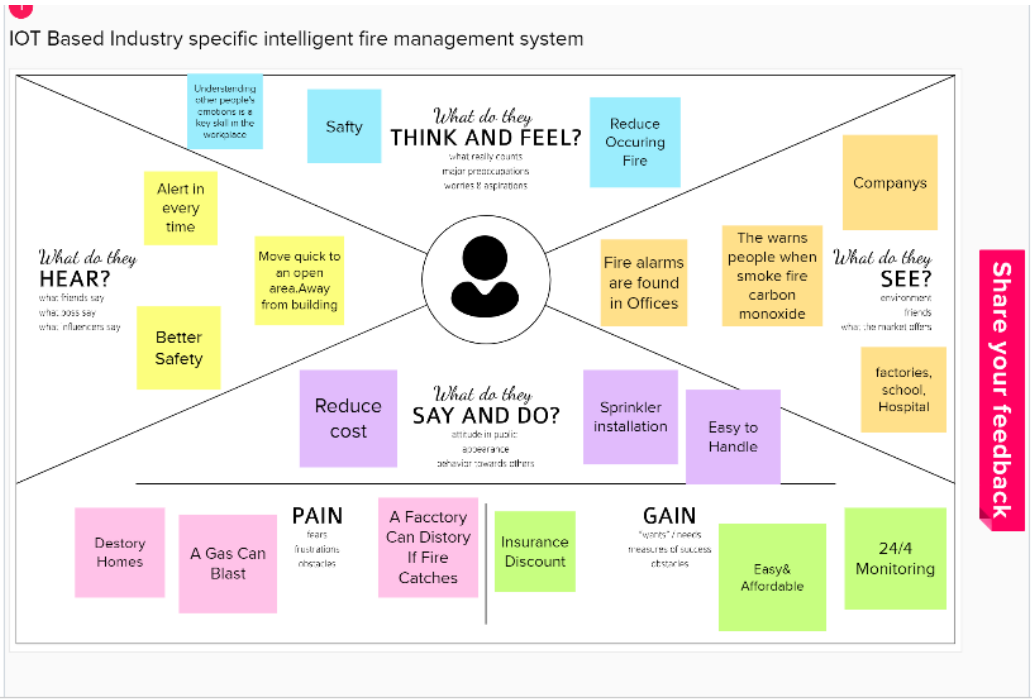
Background: Fire is the rapid oxidation of a material in the exothermic chemical process of combustion, releasing heat, light and various reaction products. Although it’s a natural process, it can lead to great destruction. On average, everyday 35 people killed due to Fire-related accidents in the five 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 changes in environment like detecting hazardous gas, flame detection and 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

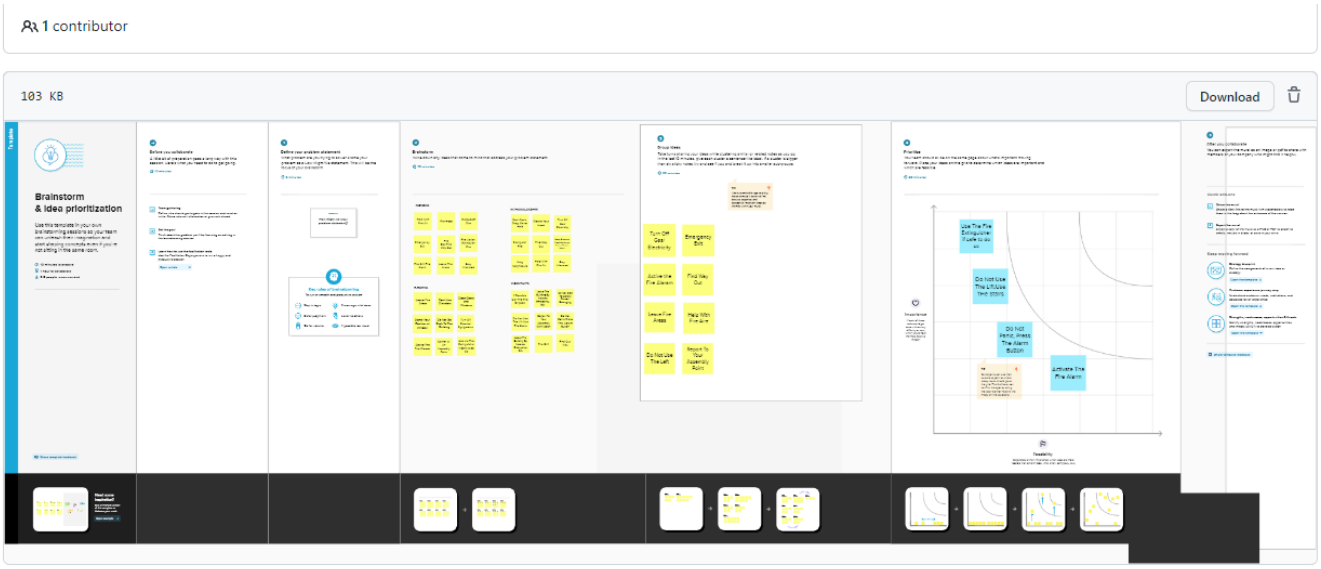
any flame is detected the sprinklers will be switched on automatically. Emergency alerts are notified to the authorities and Fire station. So that the authorities and Fire Fighters can control the situation.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

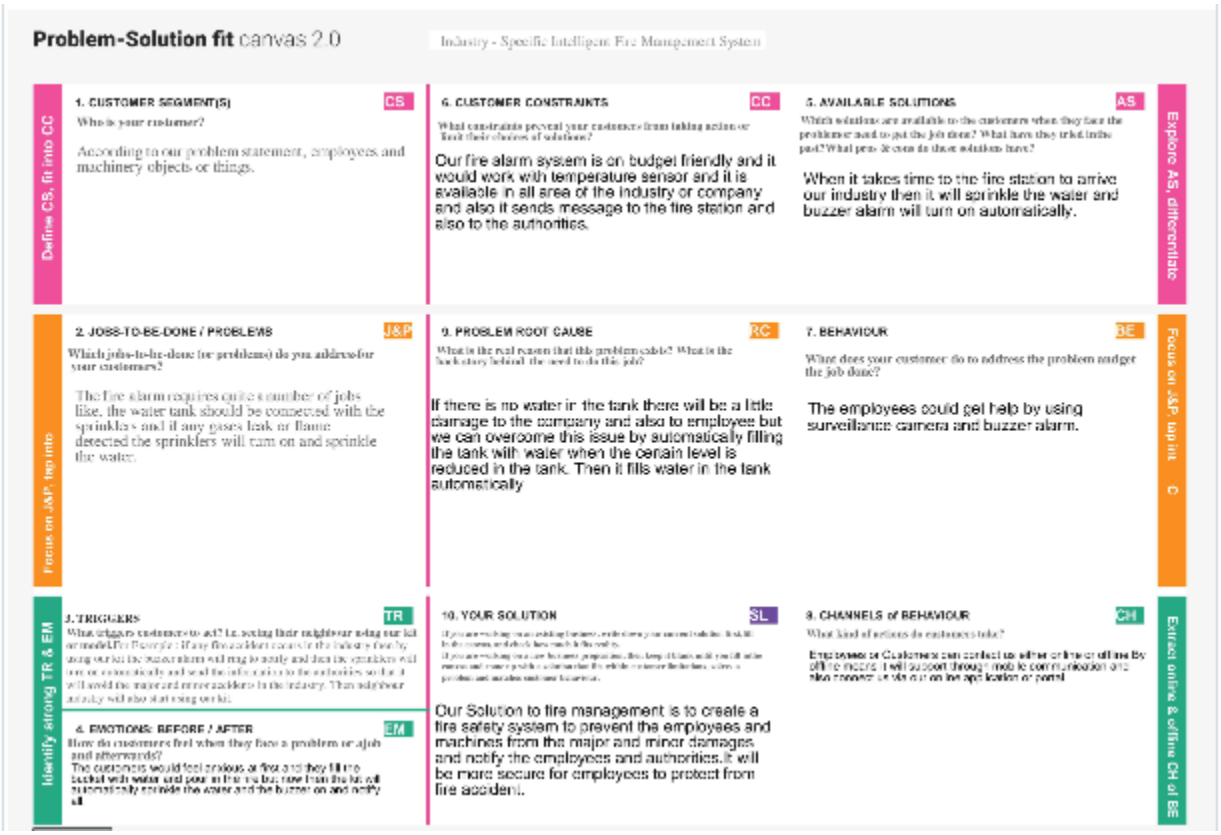


3.3 Proposed Solution

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Digital control equipment connected directly to fire alarm system via network connection. Equipment connected to building automation management system,which is connected to fire alarm by network connection. Mechanical equipment connected directly to fire alarm system via contact/relays
2.	Idea / Solution description	Alarm and warning systems,Fire service facilities,Management of fire safety
3.	Novelty / Uniqueness	Evacuation strategy Means of escape First aid fire fighting Fire service facilitiesxpenses.
4.	Social Impact / Customer Satisfaction	providing friendly, flexible and fair service is at the heart of everything we do, we measure it very carefully.
5.	Business Model (Revenue Model)	Fire safety management protects against fire and prevents business losses. As well as threatening safety and the environment, fire can cause both financial loss and significant damage to a company's image and public relations.
6.	Scalability of the Solution	automatic fire alarm system activated smoke detectors and manual call points,Automatic systems, Automatic suppression, detection.

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through e-mail id&Mobile Number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Web Applications	Node Service
FR-4	Configure to Device	IBM Wastson IoT Platform
FR-5	Data base	Cloudant DB
FR-6	Python Script	IBM IoT Platform

Non-functional Requirements:

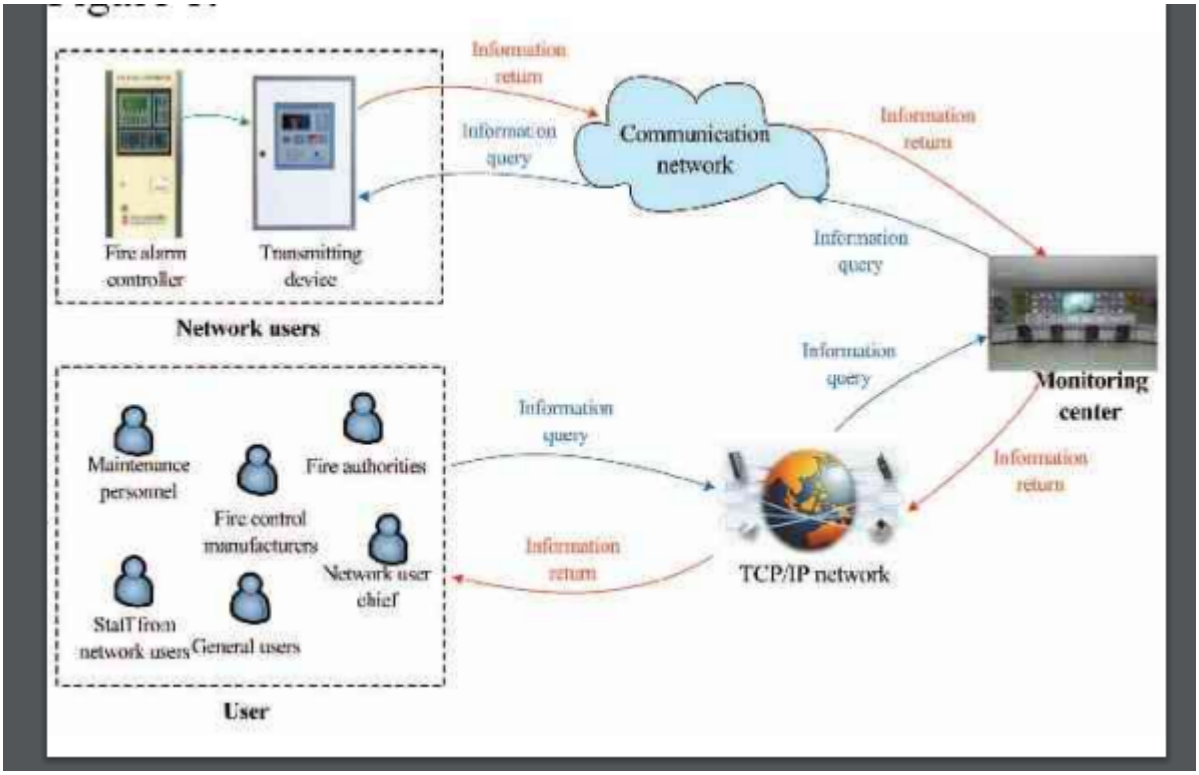
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	used to protect the damage from fire
NFR-2	**Security**	Information about increase in changes in situation is secured
NFR-3	**Reliability**	More Consistency and Dependability
NFR-4	**Performance**	Easy to moniter and measure the Temperature
NFR-5	**Availability**	Available IN defined Geofence

NFR-6	**Scalability**	To exceed future Demand

5 PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture

Solution Architecture:

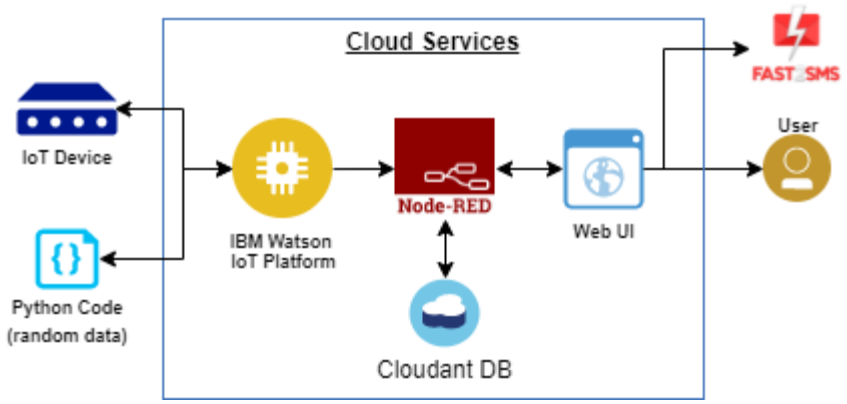
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

● Find the best tech solution to solve existing business problems.

● Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.

● Define features, development phases, and solution requirements.

● Provide specifications according to which the solution is defined, managed, and



delivered

5.3 User Stories

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2	Application Logic-1	Logic for a process in the application	Java / Python
3	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local

			Filesystem
8	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9	External API-2	Purpose of External API used in the application	Aadhar API, etc.
1	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
1	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used

4	**Availability**	**Justify the availability of application (e.g. use of load balancers, distributed servers etc.)**	**Fire Alarm,smoke sensor**
5	**Performance**	**Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.**	**Monitor the fire&** **Monitor the Alarm**

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint-2	US-1	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	10	High	Sneha K, Thanalakshmi M
Sprint-2	US-2	Create a Node-RED service.	10	High	Ramya R,Santhiya V

Sprint-4	US-1	Create Web UI in Node- Red 10 High	Sneha K Thanalakshmi M Ramya R Santhiya V
Sprint-4	US-2	Configure the Node-RED flow to 10 High receive data from the IBM IoT platform and also use Cloudant DB nodes to store the received sensor data in the cloudant DB	Sneha K Thanalakshmi M Ramya R Santhiya V

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3		US-1	Develop a python script to publish random sensor data such as temperature, Flame level and Gas level to the IBM IoT platform	7	High	Sneha K Thanalakshmi M Ramya R Santhiya V
Sprint-3		US-2	After developing python code, commands are received just print the statements which represent the control of the devices.	5	Medium	Sneha K Thanalakshmi M Ramya R Santhiya V
Sprint-3		US-3	Publish Data To The IBM Cloud	8	High	Sneha K Thanalakshmi M Ramya R Santhiya V

6.2 Sprint Delivery Schedule

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1 Sensing	USN-1	Sensing the environment using the sensors.	3	High	Sneha k
Operating	USN-2	Turning on the exhaust fan as well as the fire sprinkler system in case of fire and gas leakage.	3	Medium	Ramya R

Sprint 2 Sending collected data to the IBM Watson	USN 3	Sending the data of the Sensors to the IBM Watson.	3	High	Santhiya V

Sprint Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Registration	USN 4	Entering my email and password to verify authentication process.	3	High	Thanalakshmi M
Sprint-3 Storing of sensor data	USN-5	Storing in Cloud and database.	2	Medium	Sneha k
Node red	USN 6	Sending the data from the IBM Watson to the Node red.	3	High	Vibin T
Web UI	USN-7	Monitors the situation of the environment which displays sensor information.	1	Low	Santhiya V

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Velocity:

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

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

6.3 Reports from JIRA

https://pnt2022tmid47460.atlassian.net/jira/software/projects/ISIFMS2/boards/2/roadmap

7. CODING & SOLUTIONING

7.1 Feature 1

Python script for generating the random sensor values - Temperature, Flame Level and Gas Level to the IBM Watson IoT Platform.

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


#Provide your IBM Watson Device Credentials

organization = "4aqwut"
deviceType = "12345678dt"
deviceId = "12345678did"
authMethod = "token"
authToken = "*PrtsGAO?B@_tTPEKT"


# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="sprinkleron":
        print ("Sprinkler is on")
    elif status == "sprinkleroff":
        print ("Sprinkler is off")
    elif status == "exhaustfanon":
        print ("Exhaust Fan ON")
    elif status == "exhaustfanoff":
        print ("Exhaust Fan 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))
```

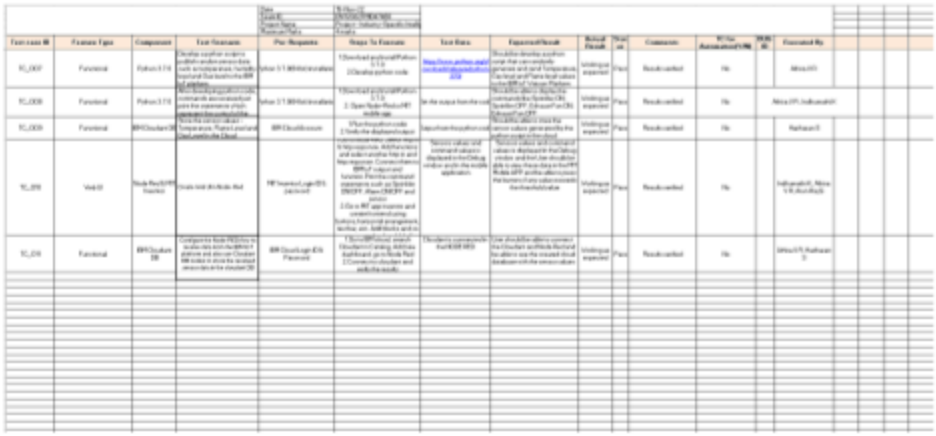
7.2 Feature 2

Output :

```
Published Temperature = 3 C Flame_Level = 88 % Gas_Level = 30 % to IBM Watson
Published Temperature = 22 C Flame_Level = 51 % Gas_Level = 16 % to IBM Watson
Published Temperature = 80 C Flame_Level = 32 % Gas_Level = 88 % to IBM Watson
Published Temperature = 98 C Flame_Level = 81 % Gas_Level = 34 % to IBM Watson
Command received: sprinkleroff
Sprinkler is off
Command received: exhaustfanoff
Exhaust Fan OFF
Command received: sprinkleron
Sprinkler is on
Published Temperature = 93 C Flame_Level = 77 % Gas_Level = 43 % to IBM Watson
Command received: exhaustfanon
Exhaust Fan ON
Published Temperature = 18 C Flame_Level = 37 % Gas_Level = 88 % to IBM Watson
Published Temperature = 61 C Flame_Level = 53 % Gas_Level = 65 % to IBM Watson
Published Temperature = 95 C Flame_Level = 76 % Gas_Level = 90 % to IBM Watson
Published Temperature = 56 C Flame_Level = 14 % Gas_Level = 27 % to IBM Watson
Published Temperature = 34 C Flame_Level = 33 % Gas_Level = 51 % to IBM Watson
Published Temperature = 9 C Flame_Level = 56 % Gas_Level = 80 % to IBM Watson
Published Temperature = 42 C Flame_Level = 51 % Gas_Level = 18 % to IBM Watson
```

8. TESTING

8.1 Test Cases



8.2 User Acceptance Testing

Purpose of Document : The purpose of this document is to briefly explain the test coverage and open issues of the Industry-specific intelligent fire management system project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis :

Section	Pass 7
Print the Sensor values	51
Client Mobile Application	2
Security	

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Severity 1	Severity 2	Severity 3	Severity 4
10	4	2	3
1	0	3	0

2	3	0	1
11	2	4	20
0	0	1	0
0	0	1	1
0	0	0	1

Resolution Subtotal By Design 20 Duplicate 4 External 6 Fixed 37 Not Reproduced 1 Skipped

2 Won't Fix 8

Totals 24 14 13 26 70 **Test Case Analysis**

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

3	0	0
9	0	0
4	0	0

Outsource Shipping 3 Exception Reporting 9 Final Report Output 4

Version Control 2 0 0 2

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

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

- The user need not require expertise knowlege to control this system. This system is simple. The user can easily view the sensor 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.

The Disadvantage of this Industry-Specific Intelligent Fire Managment system are as follows ● This system will not be able to detect the orgin of fire.

- This system will not provide the escape route if there is fire outbreak. ● If the industry has specific changes in the environment, then this system will gives false alarm.

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 sprinkler on,exhaust fan on.This remote user monitoring system can monitor the system status of each node in real time.This system monitors the data continuously so that the any slight change in the environment can be easily detected.This ensures good control accracy .ThisIndustry-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 this project is to add additonal featureslike triggering the extinguisher automatically, predict the escape route if the fire outbreaks and to implement this system in real time using hardware.

13.APPENDIX

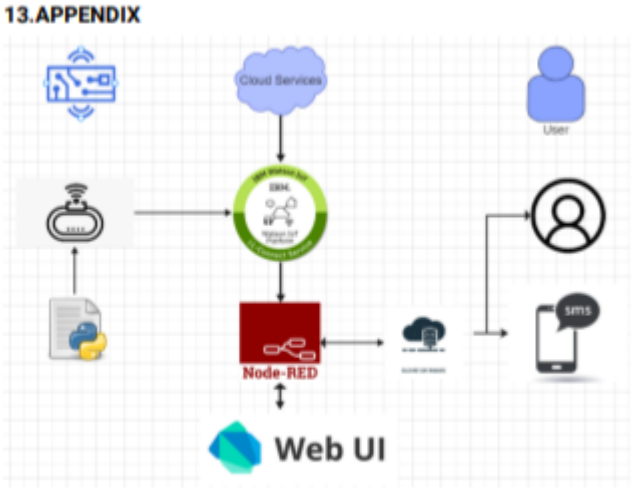


Fig : Technology architecture of our project

Source Code

https://github.com/IBM-EPBL/IBM-Project-5300-1658756270/tree/main/Develop%20a%20Python%20Script

GitHub & Project Demo Link

https://github.com/IBM-EPBL/IBM-Project-5300-1658756270