

INDUSTRY-SPECIFIC INTELLIGENT FIRE
MANAGEMENTSYSTEM

IBM PROJECT REPORT

submitted by

TEAM ID : PNT2022TMID18446

TEAM LEADER : R.RAM SURYA NARAYANAN

TEAM MEMBER 1 : S.SHREE SURYAA

TEAM MEMBER 2 : T.PRAVEEN RAJ

TEAM MEMBER 3 : P.MADASAMY

TEAM MEMBER 4 : M.M.VISHVESH

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

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

6.1 Feature 1

6.2 Feature 2

6.3 Database Schema (if Applicable)

7. TESTING

7.1 Test Cases

7.2 User Acceptance Testing

8. CONCLUSION

9. FUTURE SCOPE

CHAPTER 1

INTRODUCTION

1.1 Project Overview:

Under the rather broad heading of fire protection systems, this module will examine the main components of alerting, suppression, and containment features and systems. Consideration of these systems is a natural adjunct to a discussion of hazards and building construction features. The primary components we will examine are fire alarm systems, fire detection and notification systems, suppression agents and systems, water distribution systems, automatic sprinkler systems, standpipe and hose systems, and portable fire extinguishers. This module will cover a lot of basic material meant to provide the novice inspector a solid foundation on which to build. As was said in the earlier modules, it is only a beginning.

In the recent past, a few fire warning and alarm systems have been presented based on a combination of a smoke sensor and an alarm device to design a life-safety system. However, such fire alarm systems are sometimes error-prone and can react to non- actual indicators of fire presence classified as false warnings. There is a need for high-quality and intelligent fire alarm systems that use multiple sensor values (such as a signal from a flame detector, humidity, heat, and smoke sensors, etc.) to detect true incidents of fire. An Adaptive neuro-fuzzy Inference System (ANFIS) is used in this paper to calculate the maximum likelihood of the true presence of fire and generate fire alert. The novel idea proposed in this paper is to use ANFIS for the identification of a true fire incident by using change rate of smoke, the change rate of temperature, and humidity in the presence of fire. The model consists of sensors to collect vital data from sensor nodes where Fuzzy logic converts the raw data in a linguistic variable which is trained in ANFIS to get the probability of fire occurrence. The proposed idea also generates alerts with a message sent directly to the user's smartphone. Our system uses small size, cost-effective sensors and ensures that this solution is reproducible. MATLAB-based simulation is used for the experiments and the results show a satisfactory output.

Heat Detectors:

Heat detectors commonly are used to detect fires. They are not as prone to false alarms and are less expensive than smoke detectors. However, the response of heat detectors may not be adequate in many instances, which limits their usefulness. Heat detectors are slower to respond to fires than are smoke detectors because heat detectors cannot respond to smoke. Heat detectors typically are best suited for detecting fast-growing fires in small spaces. Heat detectors are also a means of fire detection in locations that smoke detectors cannot protect due to such environmental effects as mist, normally occurring smoke, and high humidity. Heat detectors have several different operating mechanisms.

Flame Detectors:

Another method of fire detection is detectors that are sensitive to the light waves emitted by fires. These typically operate by detecting ultraviolet (UV) or infrared (IR) energy. These detectors are extremely quick to operate and typically are used only in high hazard areas such as industrial process facilities, fuel-loading areas, and areas where explosions may occur. Explosion suppression systems protect them. One problem with IR detectors is that they will respond to sunlight, creating an unwanted alarm problem. Besides, both types of flame detectors must "see" the flame to detect it so they usually have to be pointed toward the locations where fires are likely to originate.

1.2 Purpose:

Conventional vs. Addressable (Intelligent/Smart):

Conventional systems have indeed been around a long time and have proven their reliability & credibility yet in today's digital world, people often think of analog devices or systems as old fashioned or with out-dated technology. Many businesses today still use them with. And, while newer technologies now exist, conventional systems still remains a good option. Conventional systems are highly reliable, cost-effective, and affordable for small buildings where just one or two zones could cover the entire area.

Reliability:

An addressable system is generally more reliable than a conventional system mainly due to how the different systems are wired. With a conventional system, if a device's wire is damaged or severed, its signal and the signal of other devices down the line cannot be transmitted to the control panel. With an addressable system, both ends of the wire connect to the control panel. Therefore, if one end becomes damaged or severed, signals can still reach the control panel through the other end of the loop. In addressable systems, a device can be removed or disabled and it will not affect the other devices in the loop.

Scalability:

Addressable systems provide a great deal of flexibility in comparison to conventional systems. While the number of devices either system can accommodate is determinant on the manufacturer of the alarm panels, every type of device added to a conventional system requires a new circuit. Because they require less wire, an addressable alarm control panel can accommodate far more devices than a conventional system. They can have anywhere from one to 30 loops, commonly referred to as a signalling line circuit (SLC), each of which can monitor and control several hundred devices.

Life Cycle Cost:

The equipment required for conventional systems usually comes with a much lower price tag than addressable systems, which is why conventional systems are still a popular choice for small businesses. However, it is wise to look beyond the initial costs. Even for small buildings, the lower initial costs to purchase the equipment needed for a conventional system are often offset by higher installation costs. Remember that conventional systems require a single circuit for each zone, one which can lead to much more complex wiring than an addressable system in which all the devices are wired into a single loop. This increase in complexity not only drives up the cost of installation but also introduces a greater risk of human error. While both types of systems require regular inspections and testing, trouble-shooting and maintenance are easier and less expensive with addressable systems. With independently wired zones in a conventional system, each device must be checked separately to find the problematic device.

CHAPTER 2

LITERATURE SURVEY

LITERATURE SURVEY:

1)Efficient fire detection for uncertain surveillance environment :

CNN technology is used to interconnect all the IOT devices present in the area.By capturing the video the features are detected and tests are done.Cost efficient.

2) Intelligent Multi-Sensor Detection system for monitoring indoor building fires :

The real fire data is collected from the developed sensing system is described.Then the collected sensor datas are used to analyze the performance of proposed fire detection algorithm in comparison to existing algorithm.

3)Smart performance-based design for building fire safety: Prediction of smoke motion via AI :

The datas are noted from various fire scenarios, atrium volumes, and ventilation conditions.After the conduction of test the results are used to develop the AI for accurate smoke visibility. Cost of making is more.

4)Event Classification and Intensity Discrimination for Forest Fire Inference With IoT :

Fire index is considered by using 256 fuzzy rules focusing on temperature,humidity,smoke,wind speed.

2.1 Problem Statement Definition:

Fire is one of the major concerns when analyzing the potential risks on buildings. To face undesired situations it is common to install fire safety systems in a way to prevent fire occurrence or to protect buildings against such events. In these cases it is usual to have well designed systems as well as good installation procedures. However the problem relies after this stage where it is needed some attention to issues related to test and maintenance that are not meet. This article states this problem presenting the risk of not identifying possible hidden failures that will prevent the safety barrier of having a successfully operation when it will be needed in a real fire situation. It is also shown the importance of having a test and maintenance planning and how to establish the frequency of those activities. Along the years some standards and regulations have been developed to assure a proper design, installation, maintenance and operation of such systems and avoid, limit or mitigate fire events. In this particular aspect a reference should be done to the National Fire Protection Association (NFPA) that has been for more than a century the leader on research and development of devices and on the publication of standards and regulation about fire safety. Usually these systems rely inactive for long periods or, in case of automatic systems, in a dormant mode. The so-called hidden failures are only often detected through periodic inspections or tests and thus it is very important to assure such procedures in a way to guarantee high fire safety levels. Concerning fire safety systems it is observed that equipment design and installation are not a problem. Based on this, it is important to pay attention to the remaining life time of such equipment and on their effective operation.

I'm trying to	<i>List their outcome or "job" the care about - what are they trying to achieve?</i>	List the thing they are trying to achieve here
but	<i>Describe what problems or barriers stand in the way – what bothers them most?</i>	Describe the problems or barriers that get in the way here
because	<i>Enter the "root cause" of why the problem or barrier exists – what needs to be solved?</i>	Describe the reason the problems or barriers exist
which makes me feel	<i>Describe the emotions from the customer's point of view – how does it impact them emotionally?</i>	Describe the emotions the result from experiencing the problems or barriers



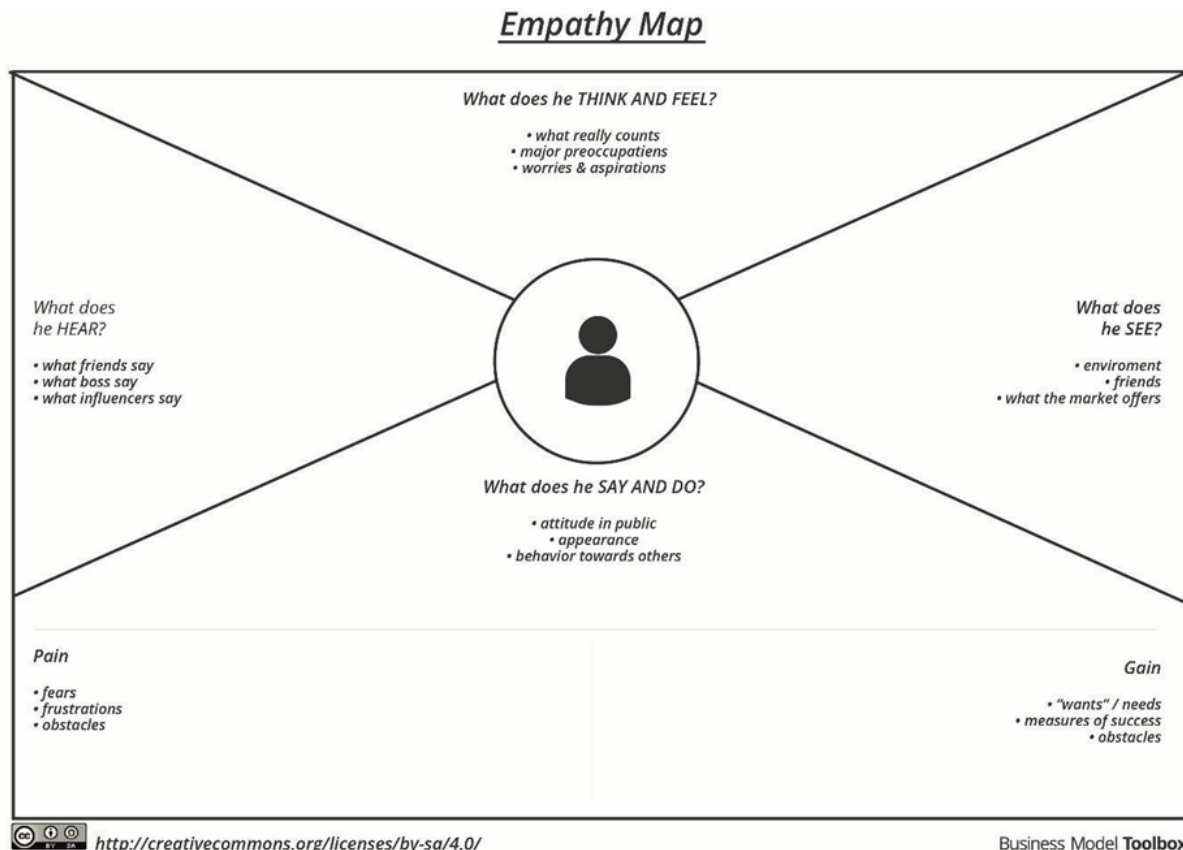
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	<p>An industrialist Who</p> <ol style="list-style-type: none"> care for the workers. Take high risk to improving the safety measures in the industry Have some basic knowledge about the fire management system in the industry 	To implement the fire detection and management system in industry	There is no sufficient technology to alert the fire accident	There is no available sensors and proper communication facilities	Frustrated

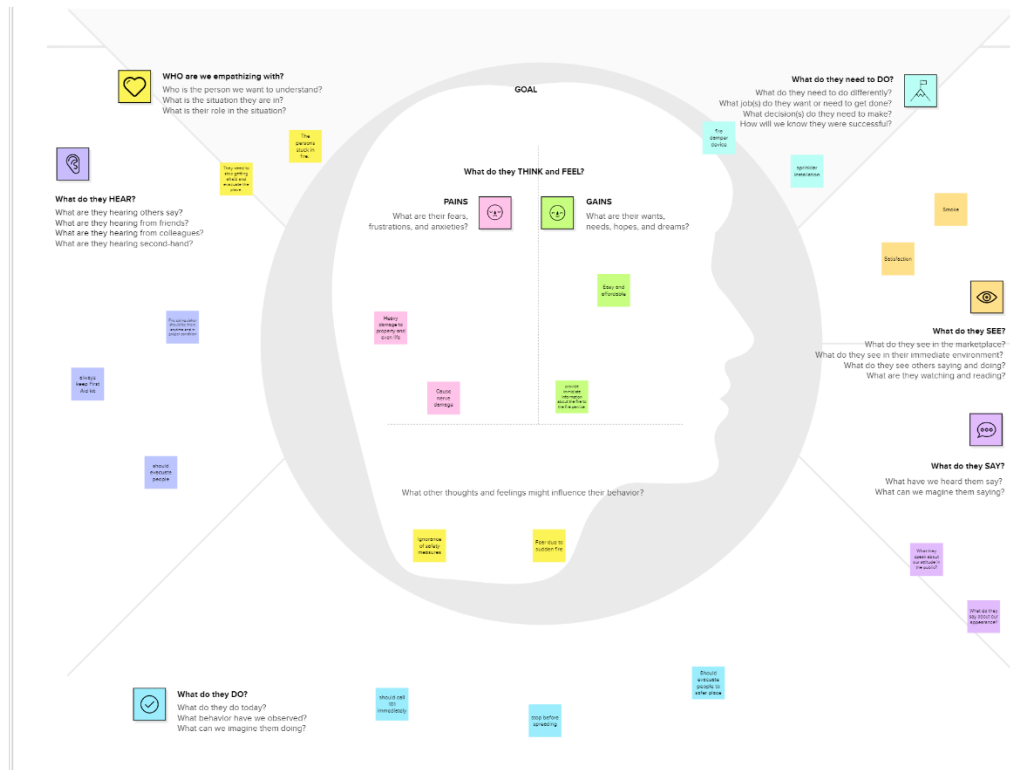
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to help teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.





Empathy map canvas

3.2 Ideation & Brainstorming:

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

- Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
- Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.
- Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) →

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

Fire accidents have become a major hazard to the industries and to many other domestic areas. The Fire should be extinguished as soon as it started

Key rules of brainstorming

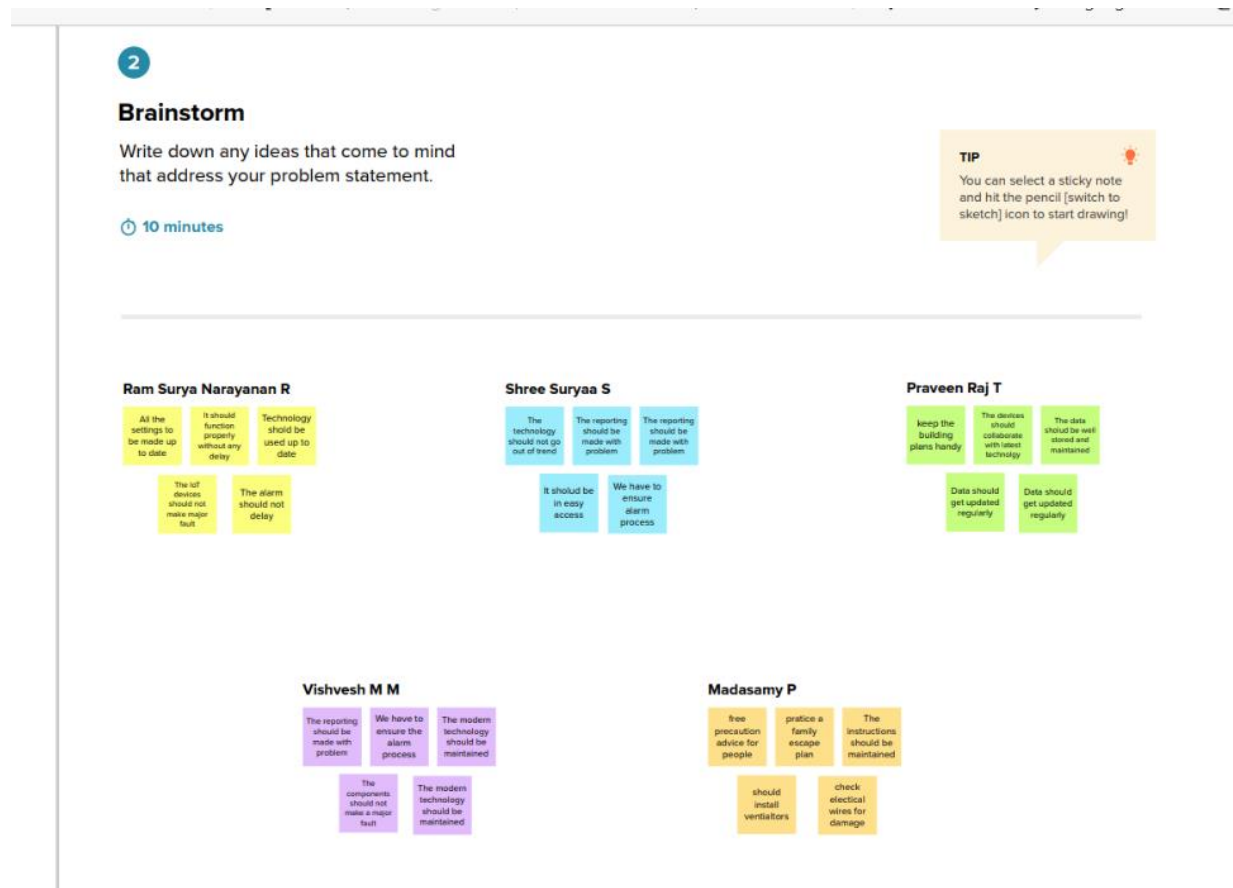
To run a smooth and productive session

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

Brainstorm

Brainstorming solution for industry specific intelligent fire management system is shown as diagram below.

Brainstorming Solution



3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To improve the safety management system in industries. Improving the safety management system against the fire incidents in industries.
2.	Idea / Solution description	To implement the fire safety management in industry based on IOT using Arduino uno board with fire detection and fire extinguisher system. And using some sensors (Humidity sensor, Flame sensor, smoke sensor) with GPS tracking system.
3.	Novelty / Uniqueness	An integrated system of temperature monitoring, gas monitoring, fire detection automatically fire extinguisher with accusation of information about locations and response through SMS notification and call.
4.	Social Impact/ Customer Satisfaction	<ul style="list-style-type: none"> ■ It early prevents the accident cost by fire in industries. ■ Nearby locations so maximum extend more accurate reliability. ■ Compatibility design integrated system.
5.	Business Model (Revenue Model)	<pre> graph TD A((Fire detection using fire detector)) --> B((Accurate information about location and response through SMS and call)) A --> C((Industry-specific intelligent fire management system)) C --> D((Fire extinguisher automatically (sprinkle the water))) C --> E((Buzzer give warning)) D --> C </pre>
6.	Scalability of the Solution	<ul style="list-style-type: none"> ■ This project can be used more efficiently with accurate information requiring. ■ Easy operability and maintenance.

		<ul style="list-style-type: none"> ■ Required lowtime for maintain ■ Cost is reasonable value.
--	--	--

3.4 Problem Solution fit:

Define CS, fit into CL	1. CUSTOMER SEGMENT(S) CS Who is your customer? eg. working parents of 0-5 y.o. kids	6. CUSTOMER LIMITATIONS EG. BUDGET, DEVICES CL What limits your customers to act when problem occurs? Spending power, budget, no cash in the pocket? Network connection? Available devices?	5. AVAILABLE SOLUTIONS PLUSES & MINUSES AS Which solutions are available to the customer when he/she is facing the problem? What had he/she tried in the past? Pluses & minuses?	Explore AS, differentiate
	2. PROBLEMS / PAINS + ITS FREQUENCY PR Which problem do you solve for your customer? There could be more than one, explore different sides. eg. existing solar solutions for private houses are not considered a good investment (1). How often does this problem occur?	9. PROBLEM ROOT / CAUSE RC What is the root of every problem from the list? eg. People think that solar panels are bad investment right now, because they are too expensive (1.1), and possible changes to the law might influence the return of investment significantly and diminish the benefits (1.2).	7. BEHAVIOR + ITS INTENSITY BE What does your customer do about / around / directly or indirectly related to the problem? eg. directly related: tries different "green energy" calculators in search for the best deal (1.1), usually chooses for 100% green provider (1.2). Indirectly related: volunteering work (Greenpeace etc) How often does this related behavior happen?	
Identify strong TR & EM	3. TRIGGERS TO ACT TR What triggers customer to act? eg. seeing their neighbor installing solar panels (1.1), reading about innovative, more beautiful and efficient solution (1.2)	10. YOUR SOLUTION SL If you are working on existing business - write down existing solution first, fill in the canvas and check how much does it fit reality. If you are working on a new business proposition then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.	8. CHANNELS of BEHAVIOR CH ONLINE Extract channels from Behavior block.	Extract online & offline CH of BE
	4. EMOTIONS BEFORE / AFTER EM Which emotions do people feel before/after this problem is solved? Use it in your communication strategy. eg. frustration, blocking (can't afford it) > boost, feeling smart, be an example for others (made a smart purchase)		OFFLINE Extract channels from Behavior block and use for customer development.	

Proposed Solution Diagram

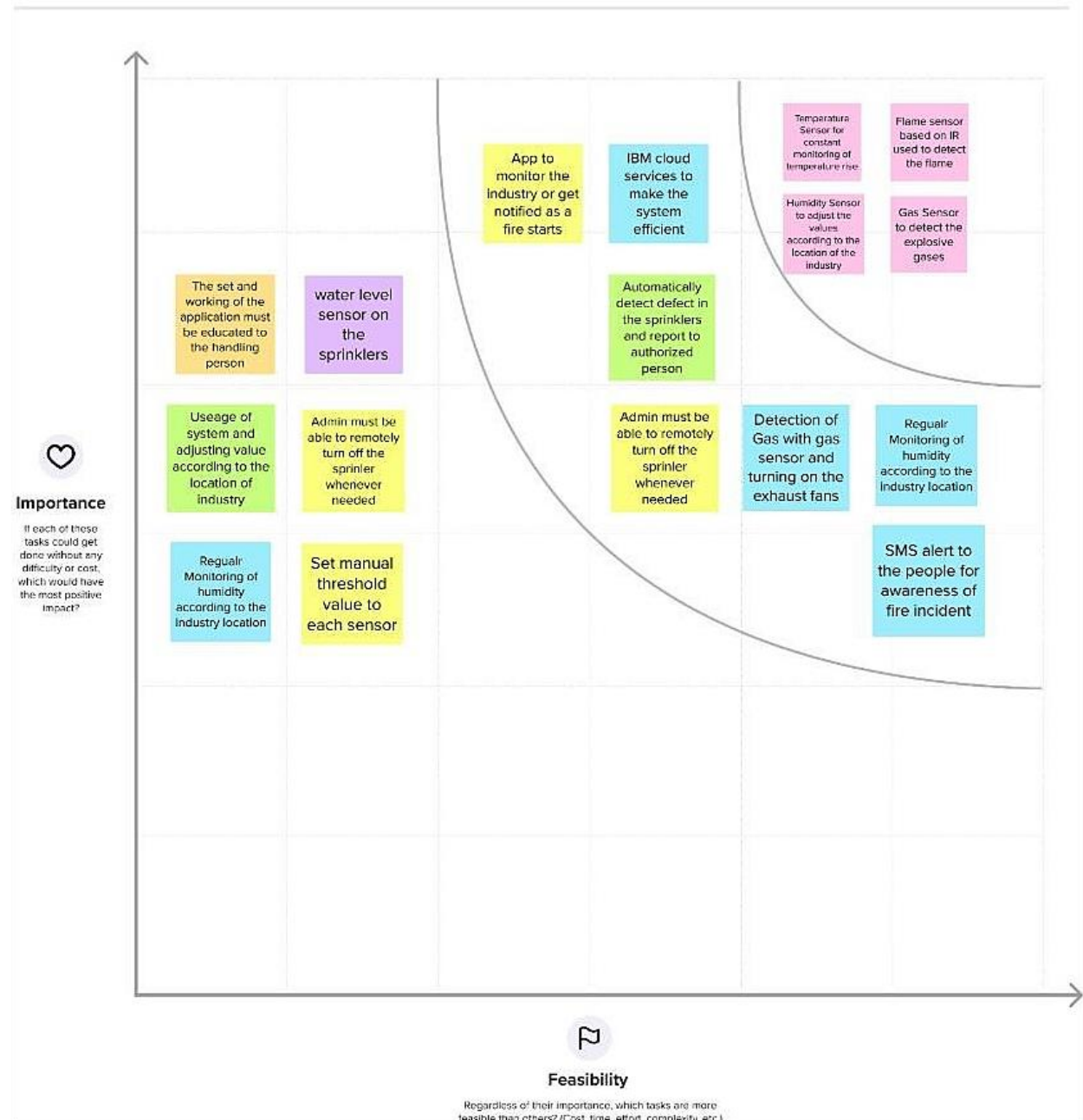
3.5 IDEA PRIORITIZATION:

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

⌚ 20 minutes



CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional requirement:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through website or application Registration through Social medias(like Instagram, Facebook) Registration through LinkedIN
FR-2	User Confirmation	Verification via Email Verification via OTP
FR-3	User Login	Login through website or App using the respective username and password
FR-4	User Access	Allows the app requirement
FR-5	User Guide	Guides the basic steps of using the application
FR-6	User Upload	User should be able to send the data
FR-7	User Solution	Data report should be generated and delivered to user for per every 24 hours
FR-8	User Data Sync	API interface to increase to invoice system

HARDWARE DEVICES:



Arduino UNO



5V Relay Supply



Buzzer



Flame Sensor



Mini Water Pump



Battery Holders

4.2 Non-Functional requirements:

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Usability requirements can consider language barriers and localization tasks. Usability can be assessed from the below functions. Efficiency of use. Low perceived workload. Easy and simple UI.
NFR-2	Security	Access permissions for the particular system information may only be changed by the system's data administrator.
NFR-3	Reliability	The database update process must roll back all related updates when any update fails.
NFR-4	Performance	The front-page load time must be no more than 2 seconds for users that access the website using an VoLTE mobile connection.
NFR-5	Availability	New module deployment mustn't impact front page, product pages, and check out pages availability and mustn't take longer than one hour. The rest of the pages that may experience problems must display a notification with a timer showing when the system is going to be up again.
NFR-6	Scalability	We can increase scalability by adding memory, servers, or disk space. On the other hand, we can compress data, use optimizing algorithms. The website attendance limit must be scalable enough to support 500,000 users at a time.

SOFTWARE APPLICATION:



The screenshot shows the Arduino IDE interface. The title bar indicates the file is named 'fire' and the IDE version is 1.8.19. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. The toolbar contains icons for opening, saving, and running the sketch. The main text area contains the following code:

```
fire
int flame=0; // select analog pin 0 for the sensor

int Beep=5; // select digital pin 5 for the buzzer

int val=0; // initialize variable

int relay= 13;

/* The setup() function is called when a sketch starts. It is used to initialize variables, pin modes, start using libraries, etc. This function will only run once, after each power up or reset of the Arduino b

void setup()
{
  pinMode(Beep,OUTPUT); // set buzzer pin as "output"
  pinMode(relay,OUTPUT); // set LED pin as "output"
  pinMode(flame,INPUT); // set flame pin as "input"
  Serial.begin(9600); // set baud rate at "9600"
}

/* The loop() function executes the program repeatedly until specified. */
```

The status bar at the bottom right indicates 'Arduino Uno on COM4'.

Arduino IDE

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Published Temperature = 88 C Humidity = 73 % to IBM Watson
Published Temperature = 59 C Humidity = 92 % to IBM Watson
Published Temperature = 9 C Humidity = 92 % to IBM Watson
Published Temperature = 76 C Humidity = 62 % to IBM Watson
Published Temperature = 95 C Humidity = 58 % to IBM Watson
Published Temperature = 14 C Humidity = 41 % to IBM Watson
Published Temperature = 73 C Humidity = 52 % to IBM Watson
Published Temperature = 10 C Humidity = 81 % to IBM Watson
Published Temperature = 69 C Humidity = 88 % to IBM Watson
Published Temperature = 45 C Humidity = 26 % to IBM Watson
Published Temperature = 76 C Humidity = 65 % to IBM Watson
Published Temperature = 17 C Humidity = 96 % to IBM Watson
Published Temperature = 51 C Humidity = 84 % to IBM Watson
Published Temperature = 84 C Humidity = 70 % to IBM Watson
Published Temperature = 34 C Humidity = 100 % to IBM Watson
Published Temperature = 84 C Humidity = 73 % to IBM Watson
Published Temperature = 42 C Humidity = 72 % to IBM Watson
Published Temperature = 10 C Humidity = 81 % to IBM Watson
Published Temperature = 97 C Humidity = 40 % to IBM Watson
Published Temperature = 13 C Humidity = 76 % to IBM Watson
Published Temperature = 44 C Humidity = 53 % to IBM Watson
Published Temperature = 65 C Humidity = 66 % to IBM Watson
Published Temperature = 62 C Humidity = 6 % to IBM Watson
Published Temperature = 93 C Humidity = 58 % to IBM Watson
Published Temperature = 49 C Humidity = 50 % to IBM Watson
Published Temperature = 6 C Humidity = 90 % to IBM Watson
Published Temperature = 4 C Humidity = 53 % to IBM Watson
Published Temperature = 95 C Humidity = 33 % to IBM Watson
Published Temperature = 80 C Humidity = 53 % to IBM Watson
Published Temperature = 96 C Humidity = 84 % to IBM Watson
Published Temperature = 91 C Humidity = 68 % to IBM Watson
Published Temperature = 70 C Humidity = 60 % to IBM Watson
Published Temperature = 79 C Humidity = 32 % to IBM Watson
Published Temperature = 41 C Humidity = 45 % to IBM Watson
Published Temperature = 42 C Humidity = 99 % to IBM Watson
Published Temperature = 18 C Humidity = 62 % to IBM Watson
Published Temperature = 97 C Humidity = 1 % to IBM Watson
Published Temperature = 75 C Humidity = 79 % to IBM Watson
```

WOKWI

SAVE

SHARE

esp32-dht22.ino

by urish

Docs

esp32-dht22.ino

diagram.json

libraries.txt

Library Manager

Simulation

▶

+

⋮

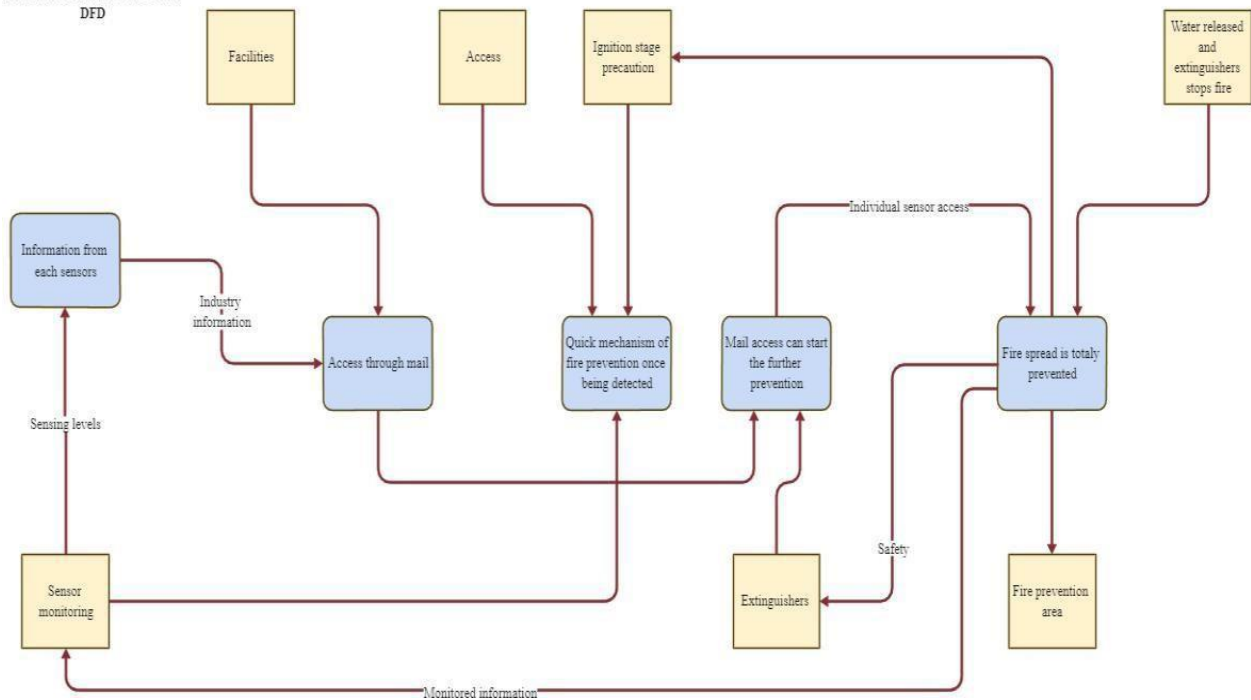
```
1 #include <time.h>
2 #include <WiFi.h>
3 #include <PubSubClient.h>
4 #define ORG "ksgtft"
5 #define DEVICE_TYPE "123"
6 #define DEVICE_ID "123_1"
7 #define TOKEN "12345678"
8 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
9 char publishTopic[] = "iot-2/evt/data/fmt/json"; char authMethod[]
10 = "use-token-auth";
11 char token[] = TOKEN;
12 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
13 WiFiClient wificlient;
14 PubSubClient client(server, 1883, wificlient);
15 float temperature = 0;
16 int gas = 0;
17 int flame = 0;
18 String flame_status = "";
19 String Gas_status = "";
20 String exhaust_fan_status = "";
21 String sprinkler_status = "";
22 void setup() {
23   Serial.begin(99900);
24   wifiConnect();
25   mqttConnect();
26 }
27 void loop() {
28   srand(time(0));
29   //initial variables and random generated data
30   temperature = random(-20,125);
31   gas = random(0,1000);
32   33
33   flame = map(flamereading,200,1024,0,2);
34   //flame = map(flame,0,1024,0,2);
35 }
```

CHAPTER 5

PROJECT DESIGN

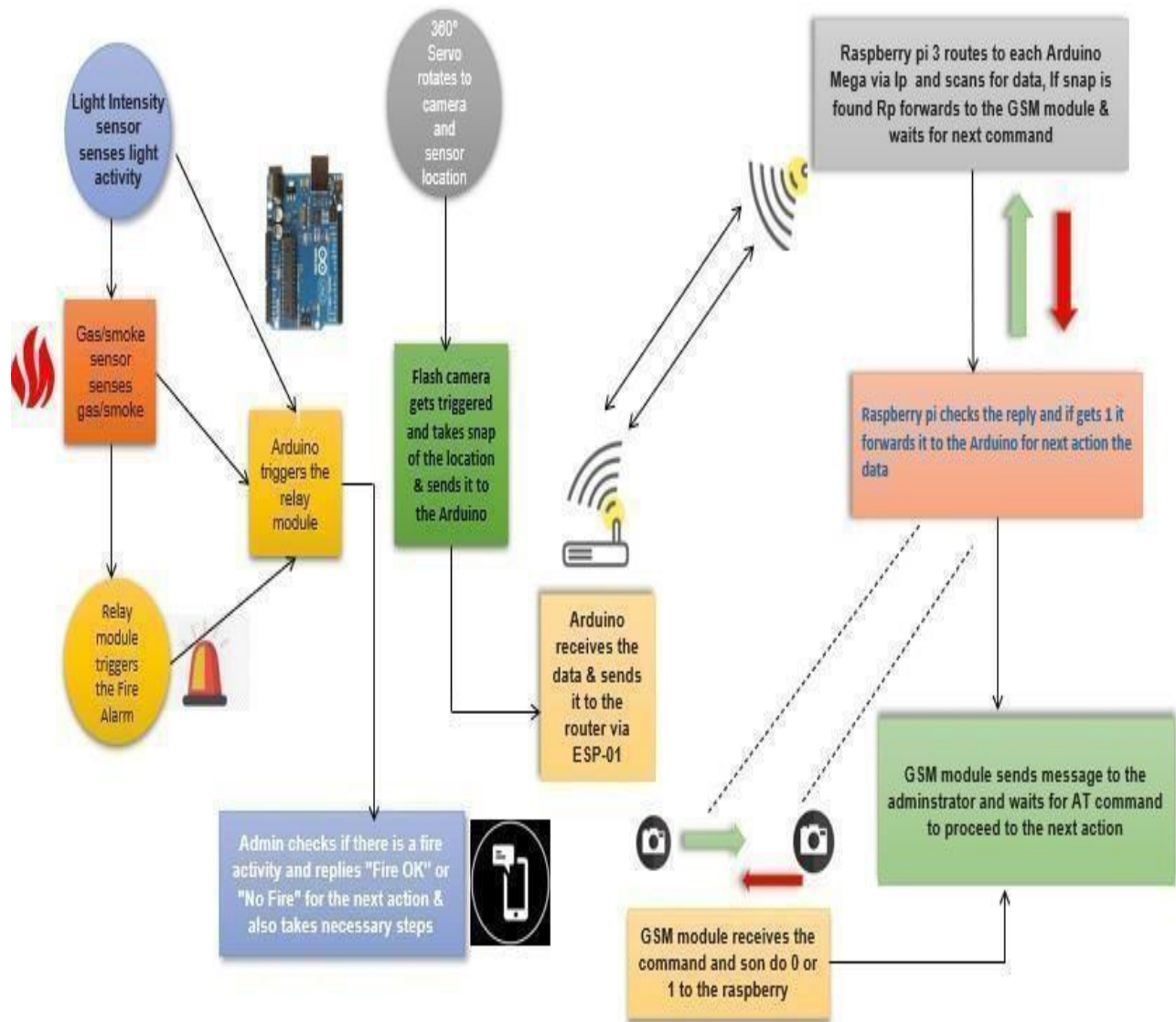
5.1 Data Flow Diagrams:

FIRE MANAGEMENT SYSTEM
DFD



Data Flow Diagram

5.2 Solution & Technical Architecture:



Solution Architecture

5.3 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	I can login with my id	High	Sprint-1

			the application by entering email & password	and password		
--	--	--	--	--------------	--	--

CHAPTER 6
CODING&SOLUTIONIN
G

CODING :

HARDWARE CODING:

```
int flame=0;/ select analog pin 0 for the sensorint
```

```
Beep=9;/ select digital pin 9 for the buzzer int
```

```
val=0;/ initialize variable
```

```
int relay= 13;
```

```
/* The setup() function is called when a sketch starts. It is used to initialize variables, pin modes, start using libraries, etc. This function will only run once, after each power up or reset ofthe Arduino board. */
```

```
void setup()
```

```
{
```

```
pinMode(Beep,OUTPUT);/ set buzzer pin as “output”
```

```
pinMode(relay,OUTPUT);/ set LED pin as “output”
```

```
pinMode(flame,INPUT);/ set flame pin as “input”
```

```
Serial.begin(9600);/ set baud rate at “9600”
```

```
}
```

```
/* The loop() function executes the program repeatedly until Specified. */
```

```
void loop()
```

```

{

val=analogRead(flame);/ read the analog value of the sensor

Serial.println(val);/ output and display the analog value

if(val>=500)/ when the analog value is larger than 600, the buzzer will buzz

{

digitalWrite(Beep,HIGH);

digitalWrite(relay,HIGH);

}else

{

digitalWrite(Beep,LOW);

digitalWrite(relay,LOW);

}

delay(500);

}

```

SOLUTIONS:

Done compiling

Sketch uses 2298 bytes (7%) of program storage space. Maximum is 32256 bytes.
Global variables use 100 bytes (0%) of dynamic memory, leaving 1858 bytes for local variables. Maximum is 2048 bytes.

CODING 2:

```
#include <time.h> #include
<WiFi.h> #include
<PubSubClient.h>#define
ORG "ksgtffi"
#define DEVICE_TYPE "123"
#define DEVICE_ID "123_1"
#define TOKEN "12345678"
char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; char
publishTopic[] = "iot-2/evt/data/fmt/json"; char authMethod[]
= "use-token-auth"; char
token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, wifiClient);
float temperature = 0;
int gas = 0;
int flame = 0;
String flame_status = "";
String Gas_status = "";
String exhaust_fan_status = "";
String sprinkler_status = ""; void
setup() { Serial.begin(99900);
wifiConnect();
mqttConnect();
}
void loop() {
srand(time(0));
/ initial variables and random generated data
temperature = random(-20,125);
gas = random(0,1000);
```

33

```
flame = map(flamereading,200,1024,0,2);
/ set a flame status
int flamereading = random(200,1024);
switch (flame) {
case 0:
flame_status = "No Fire";
break;
case 1:
flame_status = "Fire is Detected";break;
}
/ send the sprinkler status
if(flame==1){ sprinkler_status
= "Working";
}
else{
sprinkler_status = "Not Working";
}
/ toggle the fan according to gas reading
if(gas > 100){
Gas_status = "Gas Leakage is Detected";
exhaust_fan_status = "Working";
}
else{
Gas_status = "No Gas Leakage is Detected";
exhaust_fan_status = "Not Working";
}
/ json format for IBM Watson
String payload = "{";
payload+="\"gas\":";
payload+=gas;
payload+=",";
payload+="\"temperature\":";
```

```

payload+=(int)temperature;
payload+=",";
payload+="\flame\":";
payload+=flamereading;
payload+=",";
payload+="fire_status\":"+"+flame_status+"\";
payload+="sprinkler_status\":"+"+sprinkler_status+"\"; 34
payload+="Gas_status\":"+"+Gas_status+"\";
payload+="exhaust_fan_status\":"+"+exhaust_fan_status+"\"}";
if(client.publish(publishTopic, (char*) payload.c_str()))
{
Serial.println("Publish OK");
}
else{
Serial.println("Publish failed");
}
delay(1000);
if (!client.loop())
{
mqttConnect();
}
}
void wifiConnect()
{
Serial.print("Connecting to ");
Serial.print("Wifi");
WiFi.begin("Wokwi-GUEST", "",
6);
while (WiFi.status() != WL_CONNECTED)
{
delay(500);
Serial.print(".");
}
Serial.print("WiFi connected, IP address: ");

```

```

Serial.println(WiFi.localIP());
}

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

```

OUTPUT:

The screenshot displays the WOKWI simulation interface. On the left, the code for `esp32-dht22.ino` is shown, which includes headers for `time.h`, `WiFi.h`, and `PubSubClient.h`. It defines constants for the MQTT server, device type, ID, and token. The `setup` function initializes the serial port and the MQTT client. The `loop` function (partially visible) contains logic for reading sensor data and publishing it to the MQTT broker.

On the right, the simulation view shows an ESP32 microcontroller board connected to a DHT22 digital temperature and humidity sensor. The sensor is connected to the ESP32 via a 4-pin cable (VCC, GND, DATA, and a spare pin). The simulation status bar at the bottom indicates that the MQTT client successfully published data multiple times, with the message "Publish OK" repeated.

FEATURE 1:

- Monitoring and detection of fire: The system can constantly monitor the environment for potential fire hazards and provide early warning in the event of a fire.
- Automatic fire suppression: In the event of a fire, the system can automatically deploy fire suppression systems such 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 systems to provide additional protection against fire hazards.

Feature 2:

- The cloud platform enables the IoT-based intelligent fire management system to remotely monitor and manage fire safety devices and systems in real-time. It also provides data analysis and reporting capabilities to help improve fire safety.
- The fire detection and suppression system is fully automated and 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 equipped with a fire suppression system that can automatically extinguish the fire.

CHAPTER 7

TESTING

8.1 TEST CASES:

Defect Analysis:

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

8.2 User Acceptance Testing:

Test Case Analysis:

Section	Test Cases	Not Tested	Fail	Pass
Arduino UNO	7	0	0	7
Flame Sensor	5 1	0	0	51
Relay Services	2	0	0	2
Reproduced Factor	3	0	1	2
Exception Reporting	9	0	0	9
Fixed Report Output	4	0	0	4
Output	5	0	2	3

CHAPTER 8

CONCLUSIO N

CONCLUSION:

The fire detection systems proposed in the literature served fire stopping with no care of the responsiveness. Thus, this study considers the existing issues and build an efficient and effective fire detection system based on IoT technology, gas, temperature, and smoke sensors to collect the data accurately and rapidly. The continuous readings sent over WIFI modules to the central unit to analyze the data and trigger the water sprinkle. This system structure enhances the efficiency and effectiveness of fire detection. Moreover, using the Ubidots platform in this system made the data exchange faster and reliable. However, this study's proposed approach obtained an average response of 5 seconds to detect the fire and alert the property owner. Meanwhile, the water pump activated to suck water from the tank and release it into the water sprinkler to minimize the fire until the property owners and emergency services reached. Hence, the proposed system overcame the challenges of the issues of affordability, effectiveness, and responsiveness. The proposed system still needs further enhancements. Thus, one of the enhancement directions is integrating machine learning with the system to predict the potentiality of fire based on the collected data from different sources. Machine learning may help the operators find and overcome the vulnerabilities in their building to prevent fire instead of detection only.

Our system is a flexible one that offers the users all kinds of accessibility that would make the system a more viable than most other systems available in the market. But still we are aware of the limitations of our system. We look forward to improving the issues and make it an even better and cost-effective fire safety solution that can be helpful for mass people in the long run. Keeping in mind all kinds of sudden fire incidents the system will be optimized in future.

CHAPTER 9

FUTURE SCOPE

FUTURE SCOPE:

This project proposed an intelligent and smart fire warning system for smart buildings. This system not only analyses the fire presence, but also notifies the concerned people for severe fire chances in case of an emergency or critical situation. ANFIS architecture model makes the proposed system more efficient, robust and reliable; and reduces false alarms; the proposed system used easily available, lightweight and cost-effective sensors and is more reliable than conventional fire detection systems. This system can be used at the commercial level and results are reproducible. Further advancement in the proposed system can be achieved by researching more into precise and lightweight sensors that provide more accurate signals for analysis. Furthermore, the use of IoT (internet-of-things) can enhance the system by talking with various other devices and smart systems like sending the message to smart gas meters to stop the supply of gas in critical conditions, etc. This system is particularly designed for indoors, as the flame sensor is sensitive to sunlight and, secondly, the reading and training data may differ in open areas, but the minor change in training can overcome this problem.

In future, we want to adopt a more robust privacy measure, prevent data mining, implement a secondary medium of operation for the device, and implement a better debugging tool alongside a well flowing user interface that is easy to navigate and use. Some of the possible ways that the project could be improved is given below:

- **Robust privacy measures:**

As described before, the device has possibility of getting breached. So, the privacy measures to prevent such attacks have to be taken. Varied authentication solutions have to be implemented in this regard.

- **Implementation of secondary medium of operation:**

The device as of now is completely dependent on internet connection using either Wi-Fi or LAN connection. This poses a bit of a drawback while there is some kind of disrupt in the network connection.

