SRI VENKATESWARA COLLEGE OF ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

EC18099-PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

INDUSTRY-SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM

NALAIYA THIRAN PROJECT REPORT 2022

Submitted by

SIVASUBRAMANIAN R	**2127190701102**
UGENDRAN R	**2127190701119**
SUJITHA R	**2127190701111**
SAKTHI M	**2127190701093**

TEAM ID: PNT2022TMID53651

NOVEMBER 2022

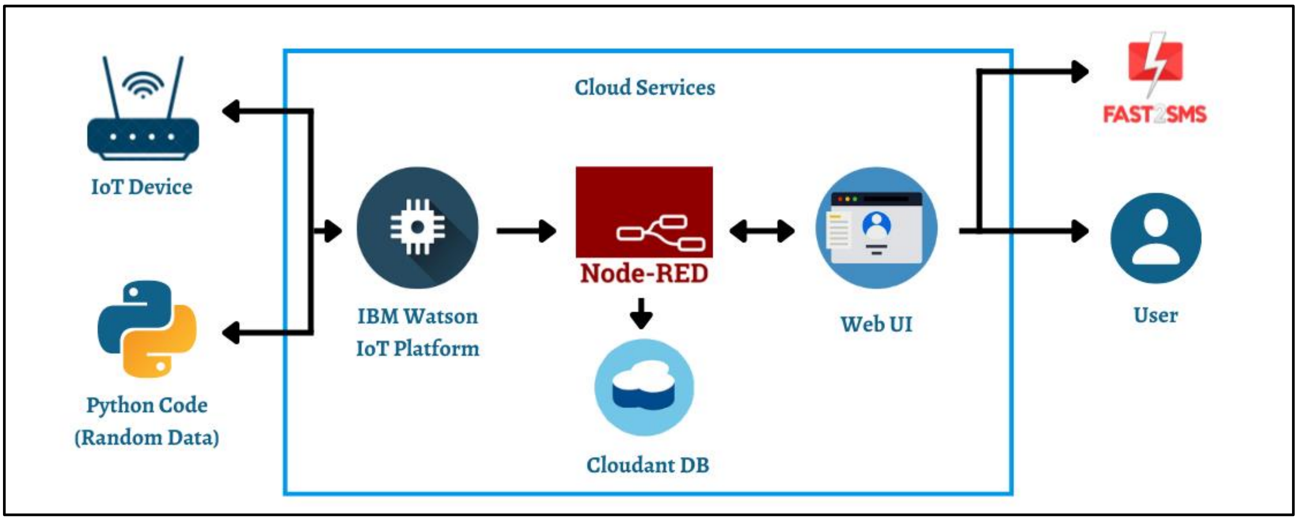
PROJECT REPORT FORMAT

1. INTRODUCTION

1.1 Project Overview

Industry-Specific Intelligent Fire Management System

- *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 main goal of a fire alarm system is to give people advance notice of a fire so they can escape and take swift action to reduce or completely extinguish the fire's effects as soon as feasible.

- *The exhaust fans are turned on based on the temperature readings and if any gases are present.*
- *Sprinklers will be activated automatically if a flame is detected.*
- *The authorities and the fire station are notified of any emergency alerts.*

2. LITERATURE SURVEY

2.1 Existing Problem

Existing analysis based on our project

https://www.nbmcw.com/article-report/others/intelligent-fire-alarm-systems-for-new-age-smart-buildings.html

2.2 References

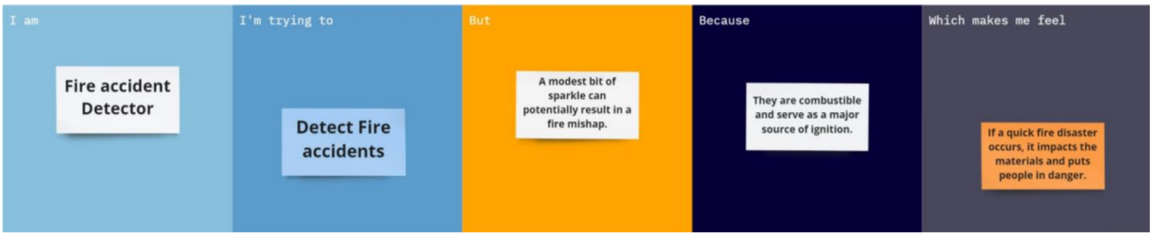
Book based on our project

Introduction to Fire Alarm System by NIYAJ B

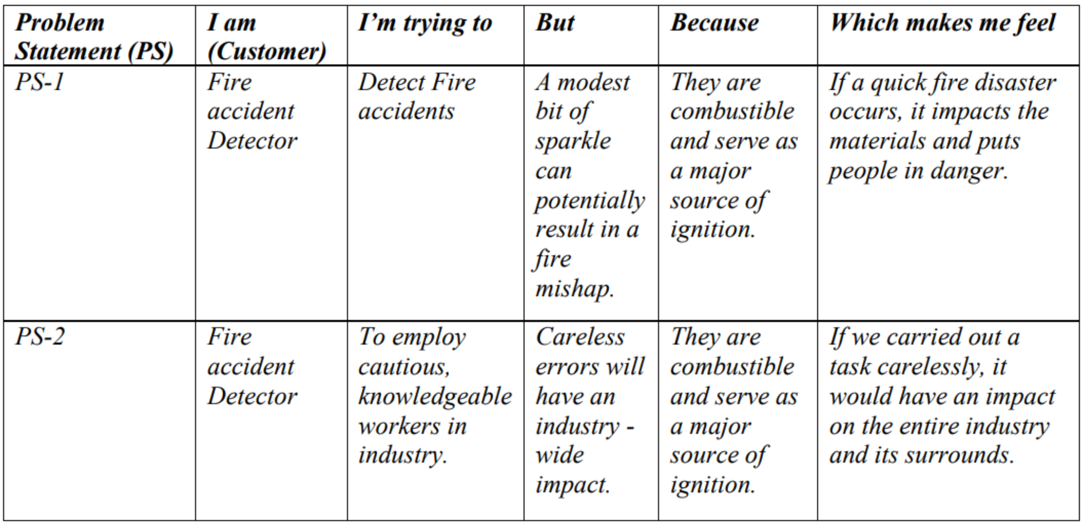
Research papers based on our project

- *https://www.researchgate.net/publication/280620907_Developed_Intelligent_Fire_alarm_system*
- *https://www.researchgate.net/publication/333538169_Intelligent_fire_detection_and_alert_system_using_labVIEW*

2.3 Problem Statement Definition



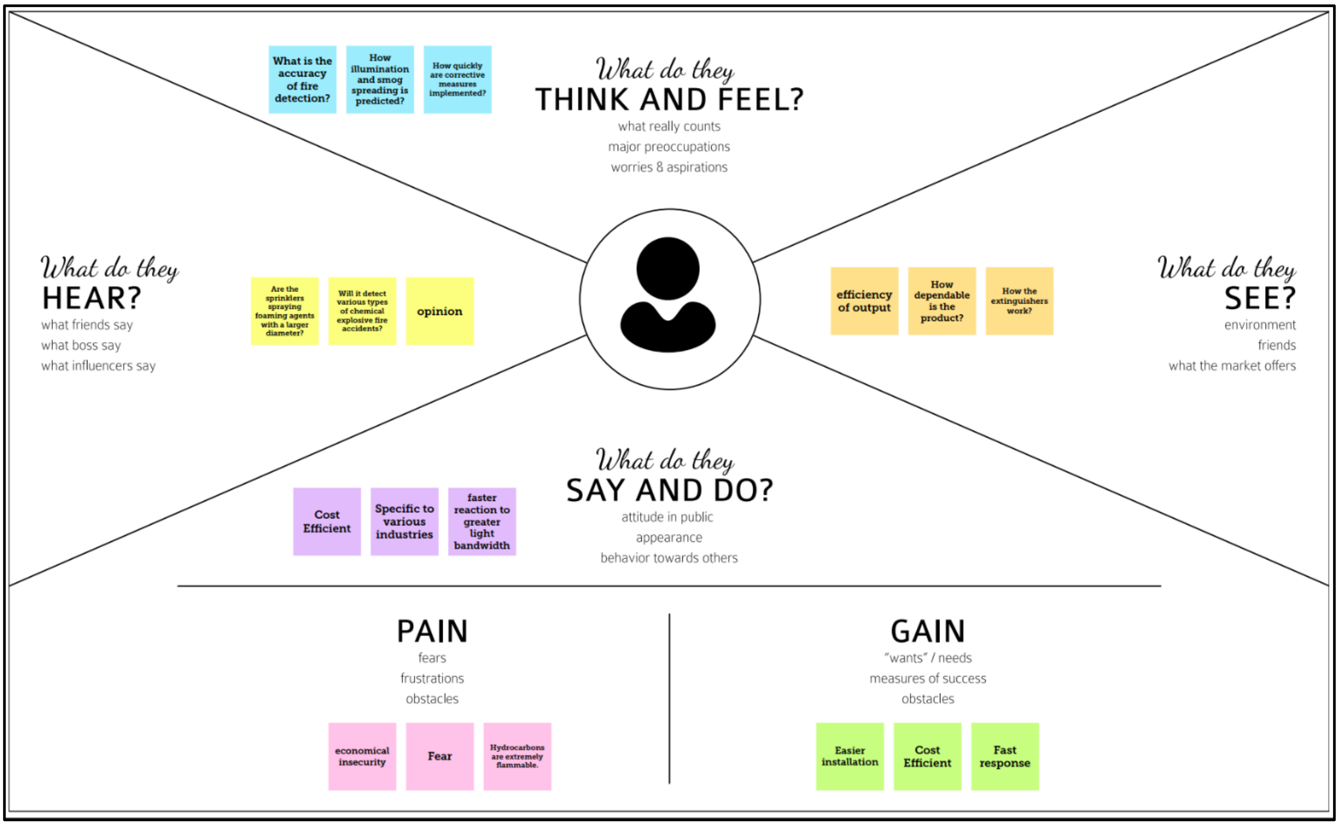
I am	I'm trying to	But	Because	Which makes me feel
Fire accident Detector	Detect Fire accidents	A modest bit of sparkle can potentially result in a fire mishap.	They are combustible and serve as a major source of ignition.	If a quick fire disaster occurs, it impacts the materials and puts people in danger.



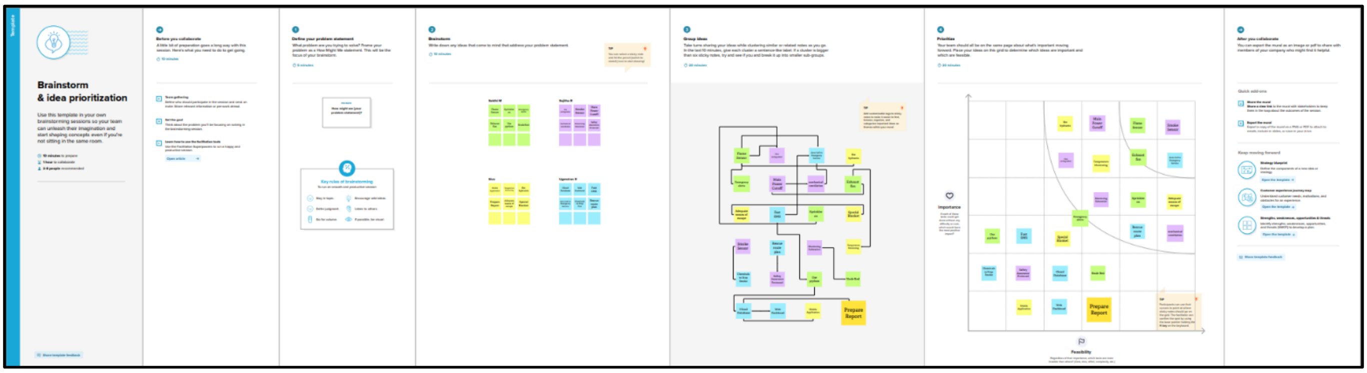
Problem Statement (PS)	***I am (Customer)***	***I'm trying to***	***But***	***Because***	***Which makes me feel***
PS-1	*Fire accident Detector*	*Detect Fire accidents*	*A modest bit of sparkle can potentially result in a fire mishap.*	*They are combustible and serve as a major source of ignition.*	*If a quick fire disaster occurs, it impacts the materials and puts people in danger.*
PS-2	*Fire accident Detector*	*To employ cautious, knowledgeable workers in industry.*	*Careless errors will have an industry - wide impact.*	*They are combustible and serve as a major source of ignition.*	*If we carried out a task carelessly, it would have an impact on the entire industry and its surrounds.*

3. IDEATION & PROPOSED SOLUTION

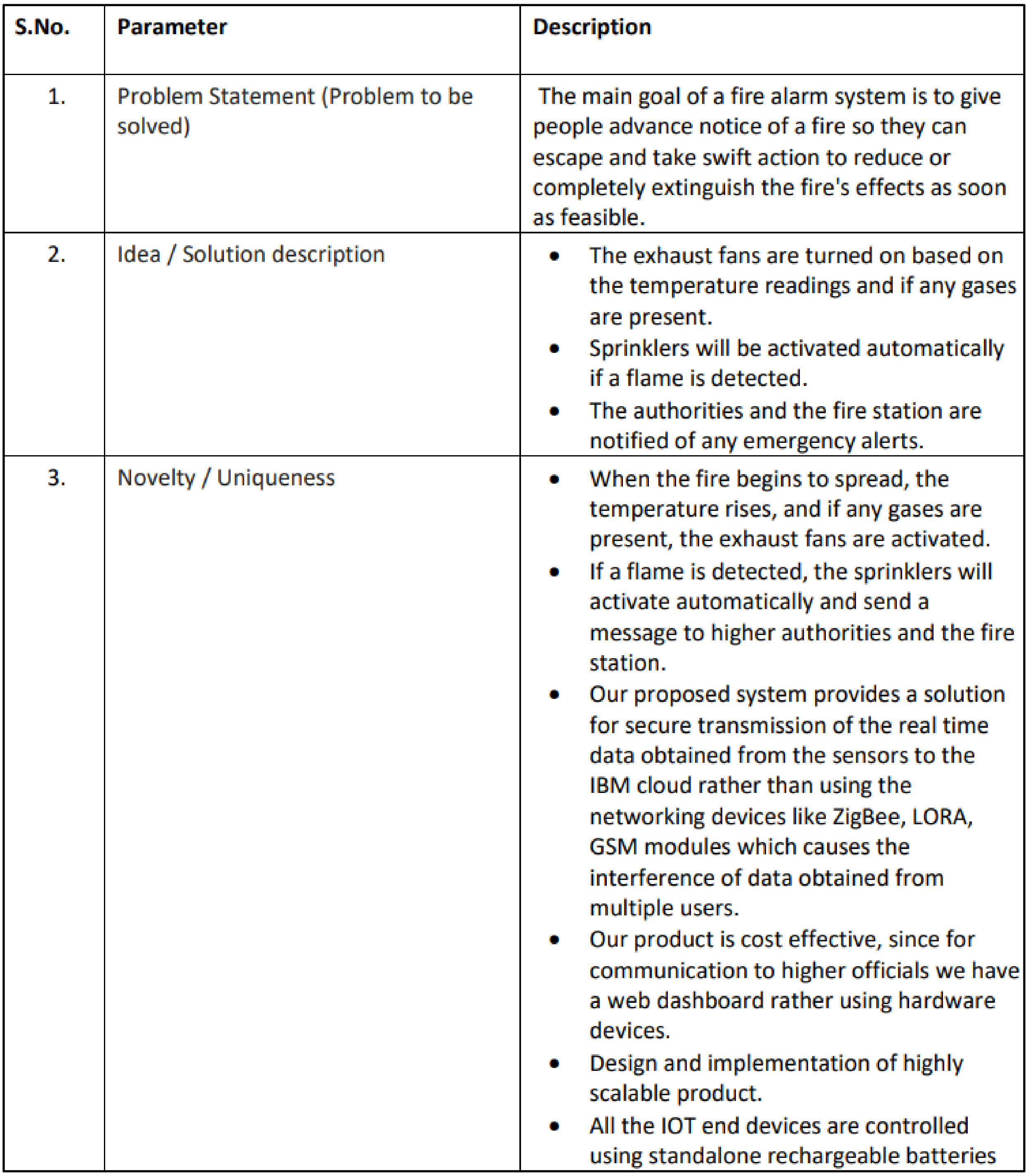
3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming


Template
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.
Prepare Report
Importance
Feasibility

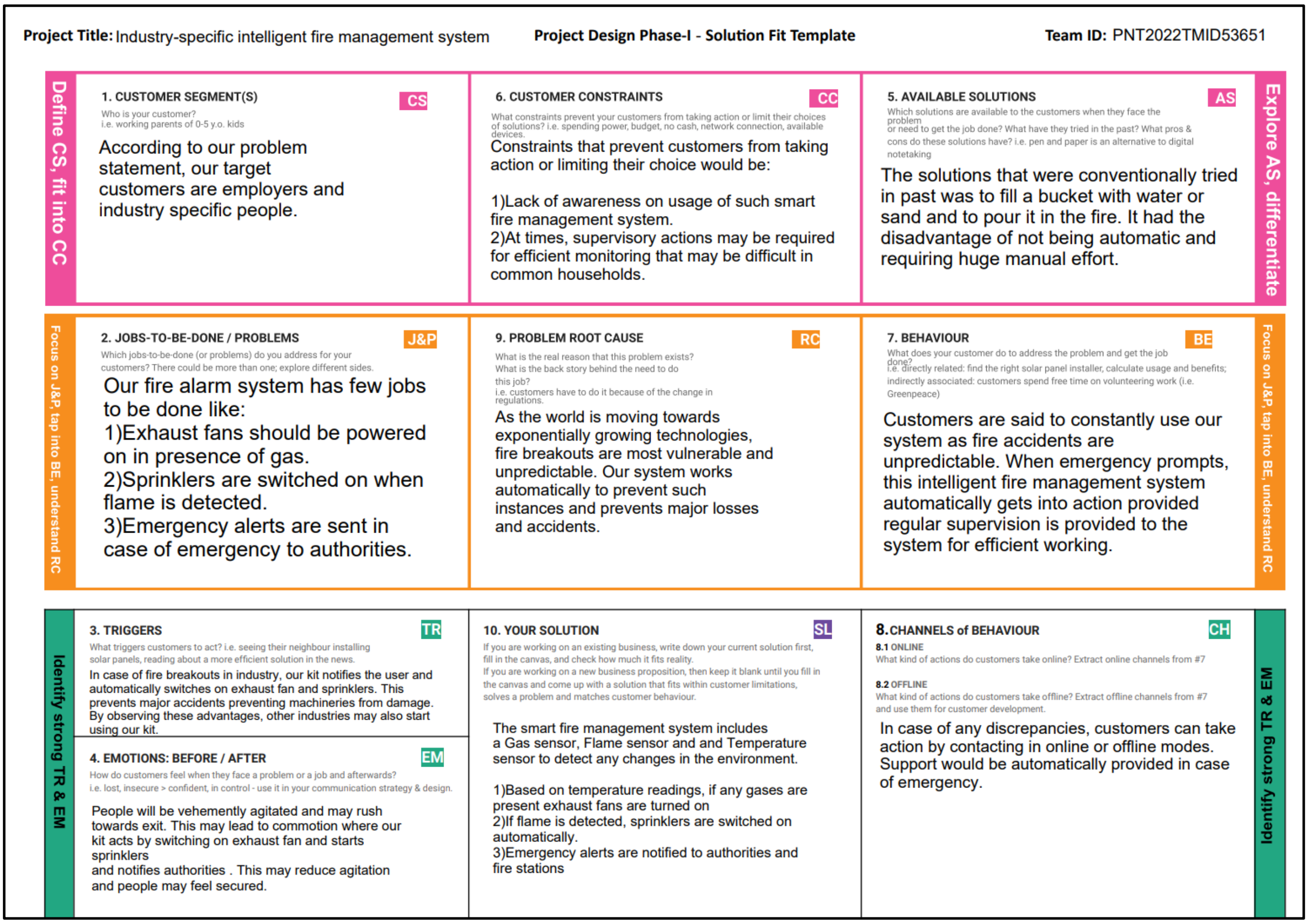
3.3 Proposed Solution



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The main goal of a fire alarm system is to give people advance notice of a fire so they can escape and take swift action to reduce or completely extinguish the fire's effects as soon as feasible.
2.	Idea / Solution description	• The exhaust fans are turned on based on the temperature readings and if any gases are present. • Sprinklers will be activated automatically if a flame is detected. • The authorities and the fire station are notified of any emergency alerts.
3.	Novelty / Uniqueness	• When the fire begins to spread, the temperature rises, and if any gases are present, the exhaust fans are activated. • If a flame is detected, the sprinklers will activate automatically and send a message to higher authorities and the fire station. • Our proposed system provides a solution for secure transmission of the real time data obtained from the sensors to the IBM cloud rather than using the networking devices like ZigBee, LORA, GSM modules which causes the interference of data obtained from multiple users. • Our product is cost effective, since for communication to higher officials we have a web dashboard rather using hardware devices. • Design and implementation of highly scalable product. • All the IOT end devices are controlled using standalone rechargeable batteries

<table>
<tr><td></td><td></td><td>so that the product would last for a long span.</td></tr>
<tr><td>4.</td><td>Social Impact / Customer Satisfaction</td><td>• Customer experience can be identified through client feedback provided by customers who use our kit and provide feedback.
• Prevents Pollution.</td></tr>
<tr><td>5.</td><td>Business Model (Revenue Model)</td><td>
<table>
<tr><th>COMPONENTS</th><th>SPECIFICATION</th><th>COST</th></tr>
<tr><td>Raspberry Pi4</td><td>4GB RAM</td><td>₹ 4969</td></tr>
<tr><td>DHT11</td><td>0 °C to 50 °C</td><td>₹ 175</td></tr>
<tr><td>Gas Sensor (MQ2)</td><td>300 – 10000 ppm</td><td>₹120</td></tr>
<tr><td>Flame Sensor</td><td>760nm-1100nm</td><td>₹ 45</td></tr>
<tr><td>Buzzer</td><td>95 dB</td><td>₹ 75</td></tr>
<tr><td>Exhaust fan</td><td>1900 RPM Speed, 150 mm Sweep and 22 W Power Consumption</td><td>₹ 500</td></tr>
<tr><td>Battery</td><td>9 V / 3.7 V</td><td>₹ 225</td></tr>
<tr><td>Sprinkler</td><td>2 inch</td><td>₹ 460</td></tr>
</table>
<b>TOTAL COST</b>: ₹ 6569</td></tr>
<tr><td>6.</td><td>Scalability of the Solution</td><td>With the help of our software, automated real-time decision-making is possible in a setting where hundreds of thousands of sensors are constantly providing data through a web dashboard without interfering with each other.</td></tr>
</table>

3.3 Problem Solution fit



Project Title: Industry-specific intelligent fire management system

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID53651

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

According to our problem statement, our target customers are employers and industry specific people.

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Constraints that prevent customers from taking action or limiting their choice would be:

1)Lack of awareness on usage of such smart fire management system.
2)At times, supervisory actions may be required for efficient monitoring that may be difficult in common households.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

The solutions that were conventionally tried in past was to fill a bucket with water or sand and to pour it in the fire. It had the disadvantage of not being automatic and requiring huge manual effort.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

Our fire alarm system has few jobs to be done like:
1)Exhaust fans should be powered on in presence of gas.
2)Sprinklers are switched on when flame is detected.
3)Emergency alerts are sent in case of emergency to authorities.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

As the world is moving towards exponentially growing technologies, fire breakouts are most vulnerable and unpredictable. Our system works automatically to prevent such instances and prevents major losses and accidents.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Customers are said to constantly use our system as fire accidents are unpredictable. When emergency prompts, this intelligent fire management system automatically gets into action provided regular supervision is provided to the system for efficient working.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

In case of fire breakouts in industry, our kit notifies the user and automatically switches on exhaust fan and sprinklers. This prevents major accidents preventing machineries from damage. By observing these advantages, other industries may also start using our kit.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

People will be vehemently agitated and may rush towards exit. This may lead to commotion where our kit acts by switching on exhaust fan and starts sprinklers
and notifies authorities . This may reduce agitation and people may feel secured.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits 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.

The smart fire management system includes a Gas sensor, Flame sensor and and Temperature sensor to detect any changes in the environment.

1)Based on temperature readings, if any gases are present exhaust fans are turned on
2)If flame is detected, sprinklers are switched on automatically.
3)Emergency alerts are notified to authorities and fire stations

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

In case of any discrepancies, customers can take action by contacting in online or offline modes. Support would be automatically provided in case of emergency.

Identify strong TR & EM

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	Sensing	Fire breakouts are sensed by smoke detectors. Gas leakages are sensed by gas sensors.
FR-2	Alert	Buzzing of vital alarms.
FR-3	Notification	Transmitting SMS with location to the Fire station. Transmitting SMS to the concerned authorities.
FR-4	Actuation	Turning ON exhaust fans. Activating sprinklers.
FR-5	Automation	Activation of the system immediately in case of fire breakouts

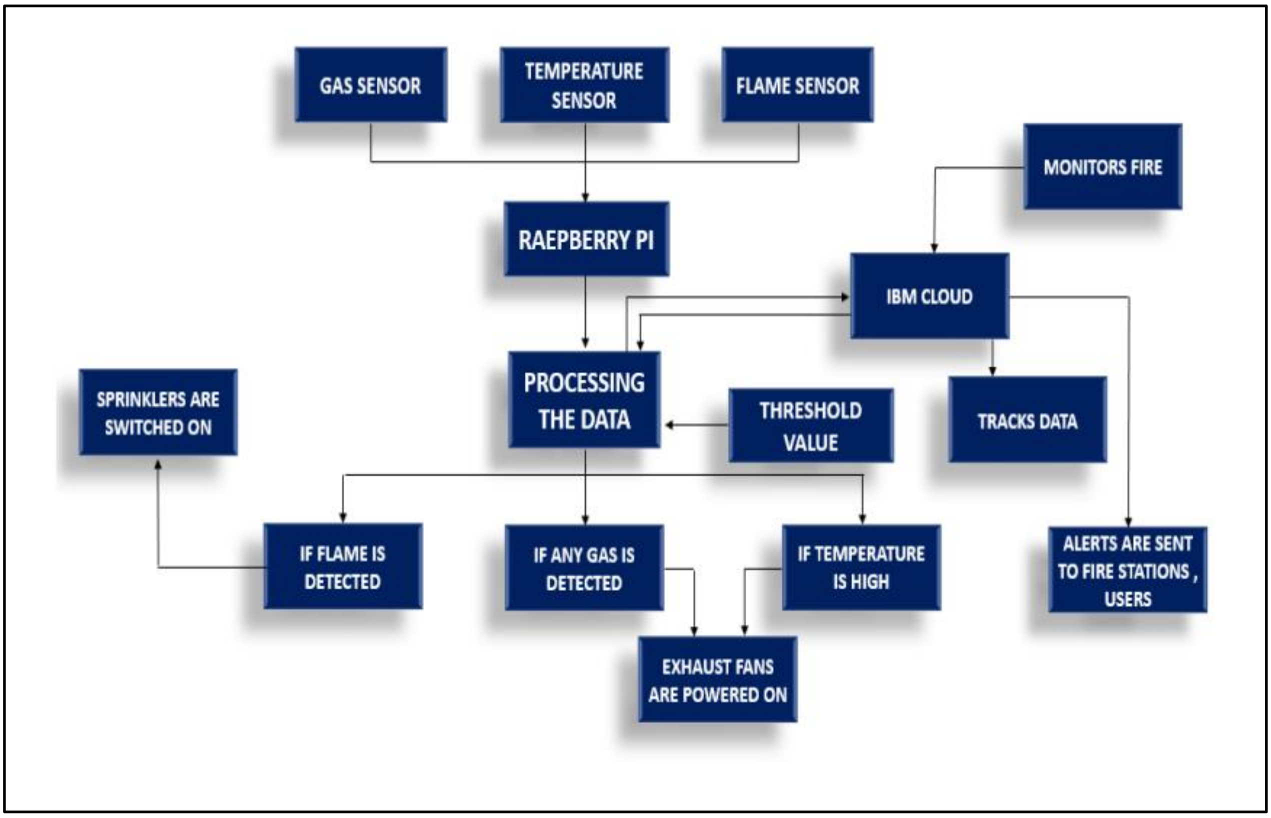
4.2 Non - Functional requirement

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Ease of usage and durability of the system.
NFR-2	**Security**	Software being prone to attacks is low. Hence it proves to be secured.
NFR-3	**Reliability**	It deals with high accuracy.
NFR-4	**Performance**	Performance is enhanced with faster response.
NFR-5	**Availability**	This system can be implemented in institutions, restaurants, schools, offices and other public places.
NFR-6	**Scalability**	It can be upgraded regularly with modifications for various requirements.

5. PROJECT DESIGN

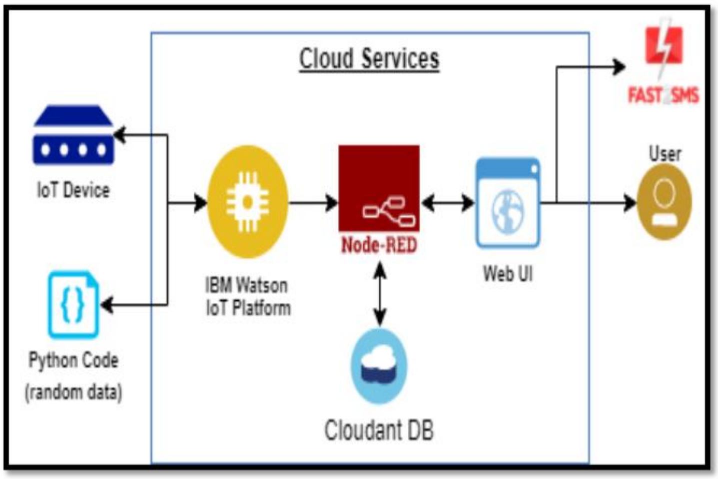
5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

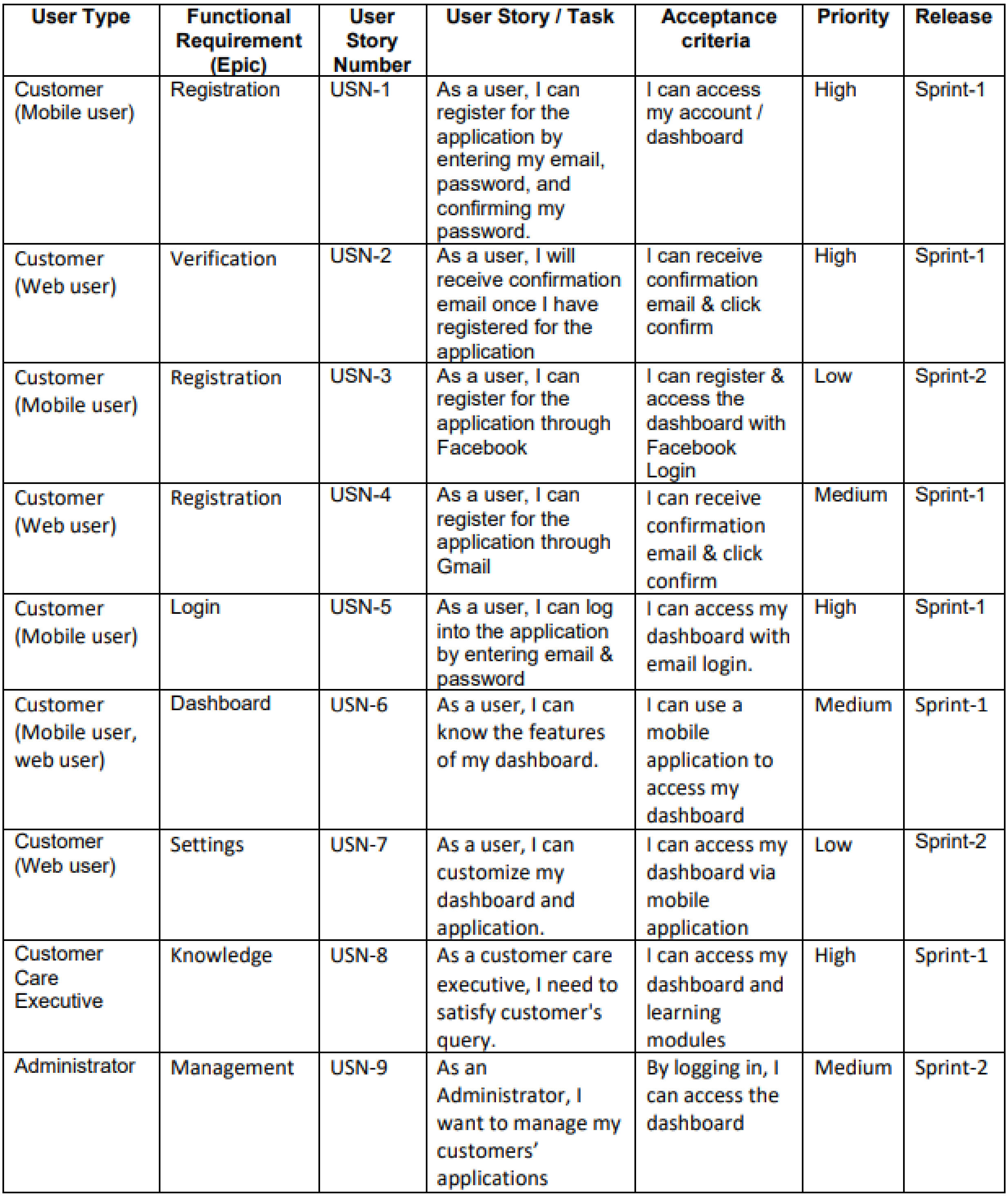
Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Node-RED, MIT app, Web UI.	IBM IoT Platform, IBM Node red, IBM Cloud.
2.	Application Logic-1	Create IBM Watson IoT platform and create node-red service	IBM Watson, IBM cloudant service, IBM node-red.
3.	Application Logic-2	Develop python script to publish and subscribe to IBM IoT Platform	python
4.	Application Logic-3	Build a web application using node red service	IBM Node-red
5.	Database	Data Type, Configurations etc.	MySQL
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant.
7.	File Storage	Developing mobile application to store and receive the sensors information and to react accordingly	Web UI, python.
8.	External API-1	Using this IBM fire management API we can track the temperature of the incident place and where the fire had been attacked.	IBM fire management API.
9.	External API-2	Using this IBM Sensors it detects the fire, gas leaks , temperature and provides the activation of sprinklers to web UI	IBM Sensors
10.	Machine Learning Model	Using this we can derive the object recognition model	Object Recognition Model
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Cloud Server Configuration	IBM cloudant, IBM IoT Platform

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	MIT App Inventor	MIT License
2.	Security Implementations	IBM Services	Encryptions, IBM Controls
3.	Scalable Architecture	Sensor-IoT Cloud based architecture	cloud computing and AI
4.	Availability	Mobile, laptop, desktop, etc.	MIT app
5.	Performance	Detects the Fire, gas leak and temperature threshold.	Sensors

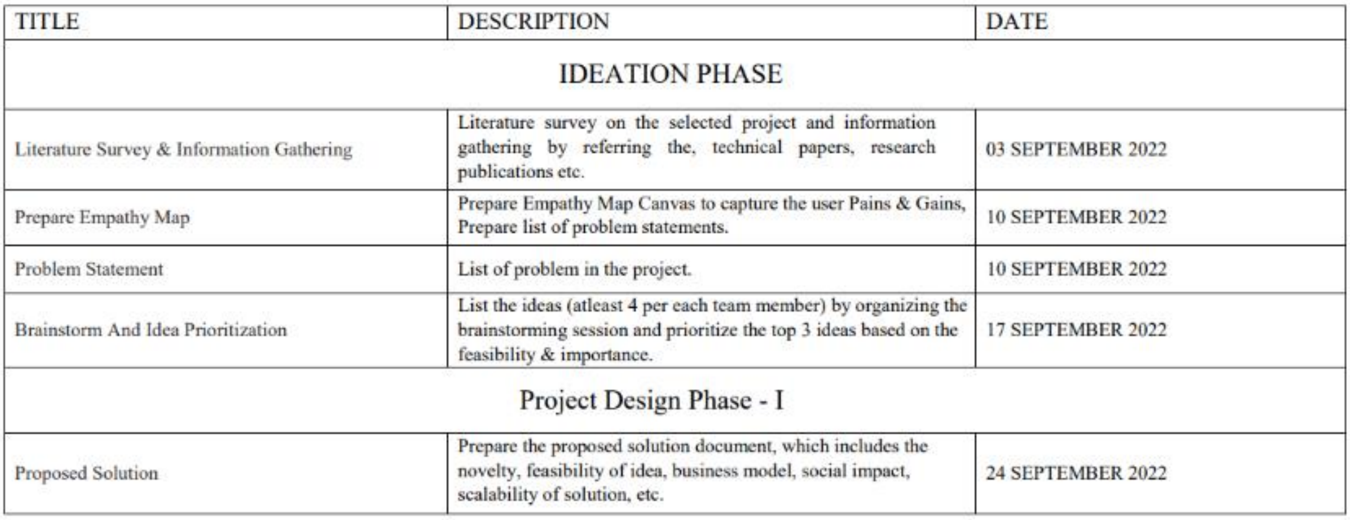
5.3 User Stories



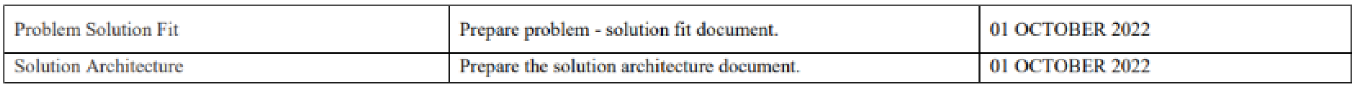
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Customer (Web user)	Verification	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
Customer (Mobile user)	Registration	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
Customer (Web user)	Registration	USN-4	As a user, I can register for the application through Gmail	I can receive confirmation email & click confirm	Medium	Sprint-1
Customer (Mobile user)	Login	USN-5	As a user, I can log into the application by entering email & password	I can access my dashboard with email login.	High	Sprint-1
Customer (Mobile user, web user)	Dashboard	USN-6	As a user, I can know the features of my dashboard.	I can use a mobile application to access my dashboard	Medium	Sprint-1
Customer (Web user)	Settings	USN-7	As a user, I can customize my dashboard and application.	I can access my dashboard via mobile application	Low	Sprint-2
Customer Care Executive	Knowledge	USN-8	As a customer care executive, I need to satisfy customer's query.	I can access my dashboard and learning modules	High	Sprint-1
Administrator	Management	USN-9	As an Administrator, I want to manage my customers' applications	By logging in, I can access the dashboard	Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

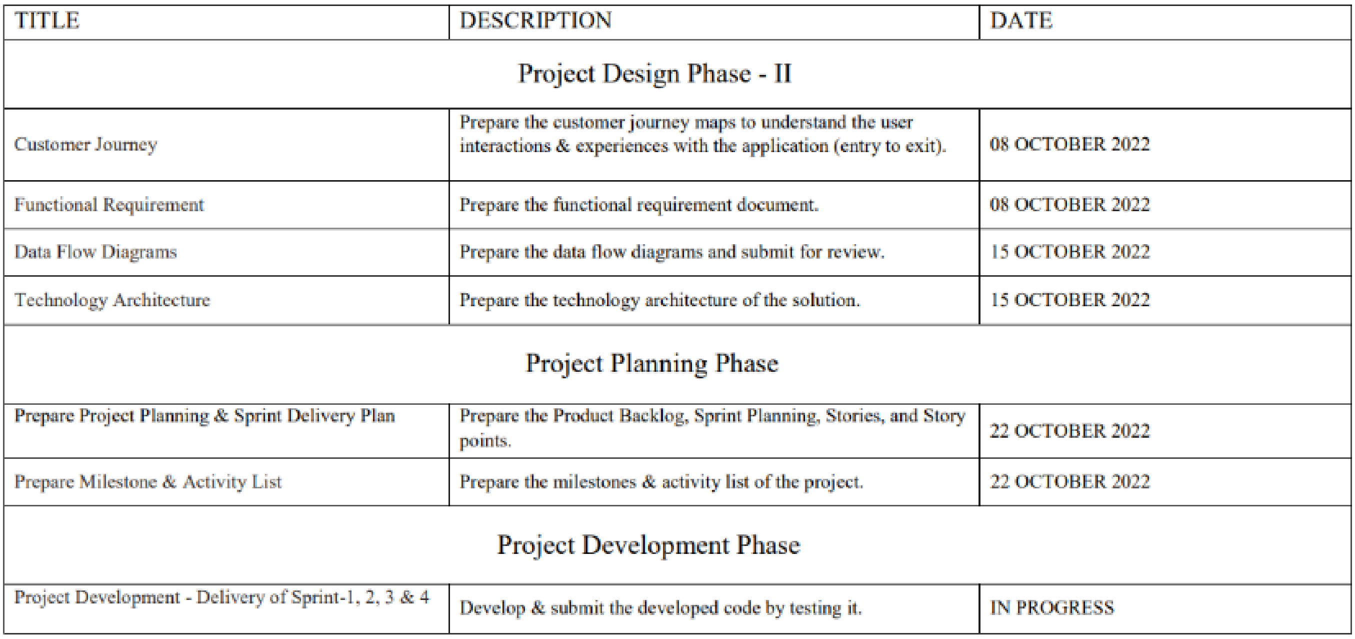
6.1 Sprint Planning & Estimation



TITLE	DESCRIPTION	DATE
IDEATION PHASE		
Literature Survey & Information Gathering	Literature survey on the selected project and information gathering by referring the, technical papers, research publications etc.	03 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements.	10 SEPTEMBER 2022
Problem Statement	List of problem in the project.	10 SEPTEMBER 2022
Brainstorm And Idea Prioritization	List the ideas (atleast 4 per each team member) by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	17 SEPTEMBER 2022
Project Design Phase - I		
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	24 SEPTEMBER 2022

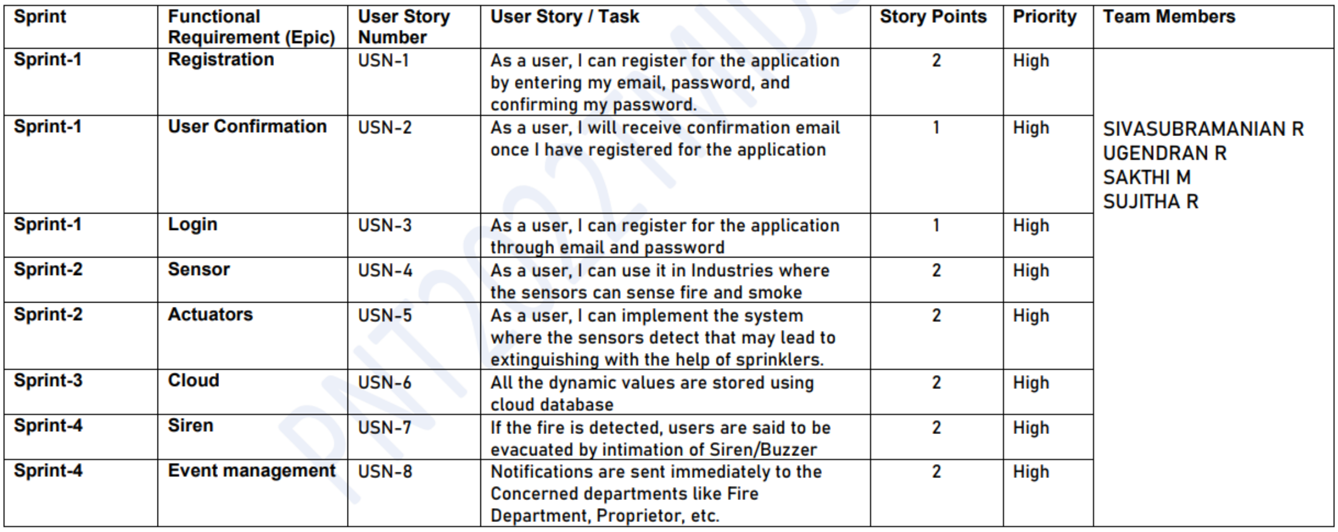


Problem Solution Fit	Prepare problem - solution fit document.	01 OCTOBER 2022
Solution Architecture	Prepare the solution architecture document.	01 OCTOBER 2022

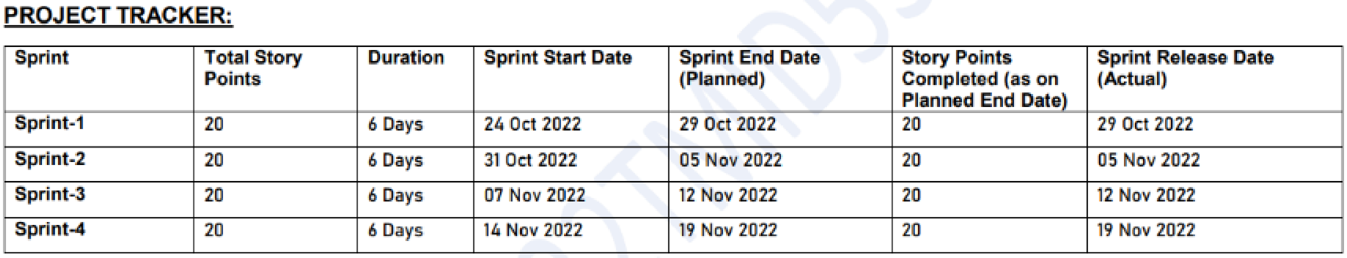


TITLE	DESCRIPTION	DATE
Project Design Phase - II		
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	08 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	08 OCTOBER 2022
Data Flow Diagrams	Prepare the data flow diagrams and submit for review.	15 OCTOBER 2022
Technology Architecture	Prepare the technology architecture of the solution.	15 OCTOBER 2022
Project Planning Phase		
Prepare Project Planning & Sprint Delivery Plan	Prepare the Product Backlog, Sprint Planning, Stories, and Story points.	22 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	22 OCTOBER 2022
Project Development Phase		
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS

6.2 Sprint Delivery Schedule



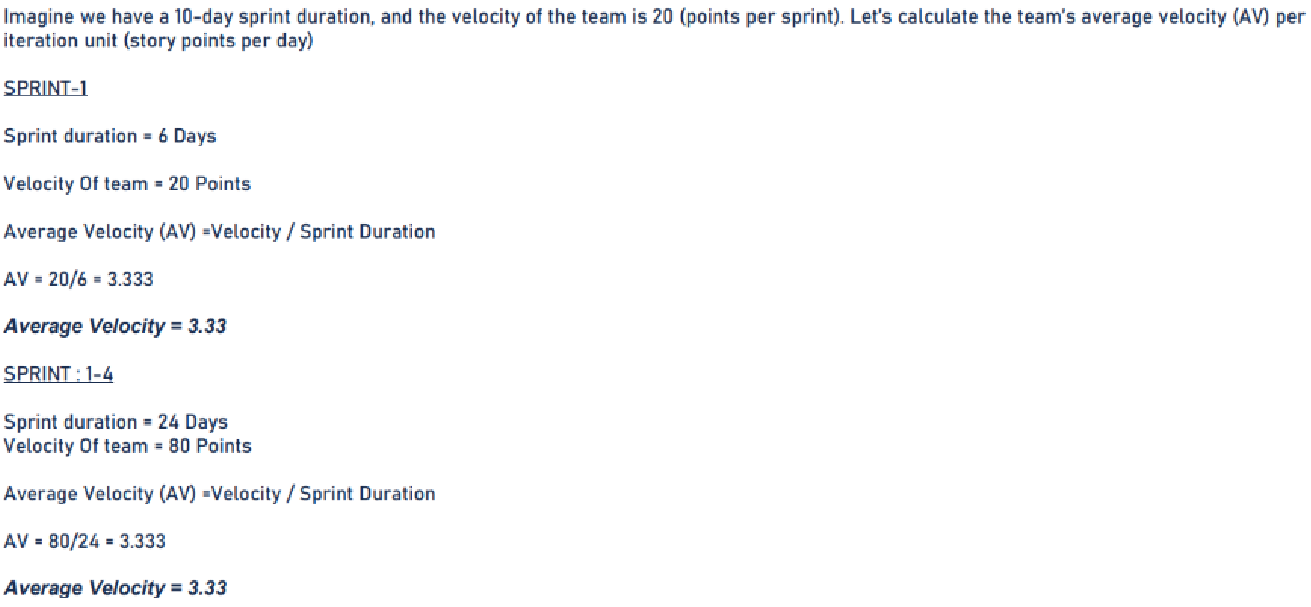
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	SIVASUBRAMANIAN R UGENDRAN R SAKTHI M SUJITHA R
Sprint-1	User Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	
Sprint-1	Login	USN-3	As a user, I can register for the application through email and password	1	High	
Sprint-2	Sensor	USN-4	As a user, I can use it in Industries where the sensors can sense fire and smoke	2	High	
Sprint-2	Actuators	USN-5	As a user, I can implement the system where the sensors detect that may lead to extinguishing with the help of sprinklers.	2	High	
Sprint-3	Cloud	USN-6	All the dynamic values are stored using cloud database	2	High	
Sprint-4	Siren	USN-7	If the fire is detected, users are said to be evacuated by intimation of Siren/Buzzer	2	High	
Sprint-4	Event management	USN-8	Notifications are sent immediately to the Concerned departments like Fire Department, Proprietor, etc.	2	High	



PROJECT TRACKER:

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

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)

SPRINT-1

Sprint duration = 6 Days

Velocity Of team = 20 Points

Average Velocity (AV) =Velocity / Sprint Duration

AV = 20/6 = 3.333

Average Velocity = 3.33

SPRINT : 1-4

Sprint duration = 24 Days
Velocity Of team = 80 Points

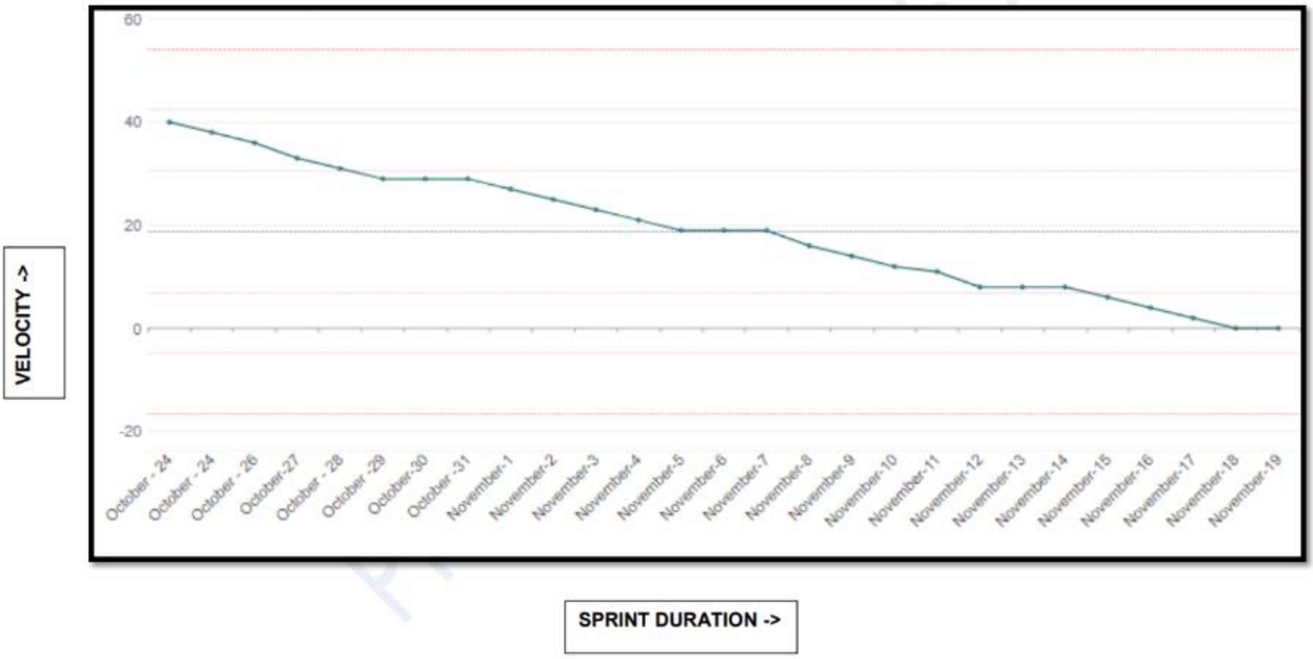
Average Velocity (AV) =Velocity / Sprint Duration

AV = 80/24 = 3.333

Average Velocity = 3.33

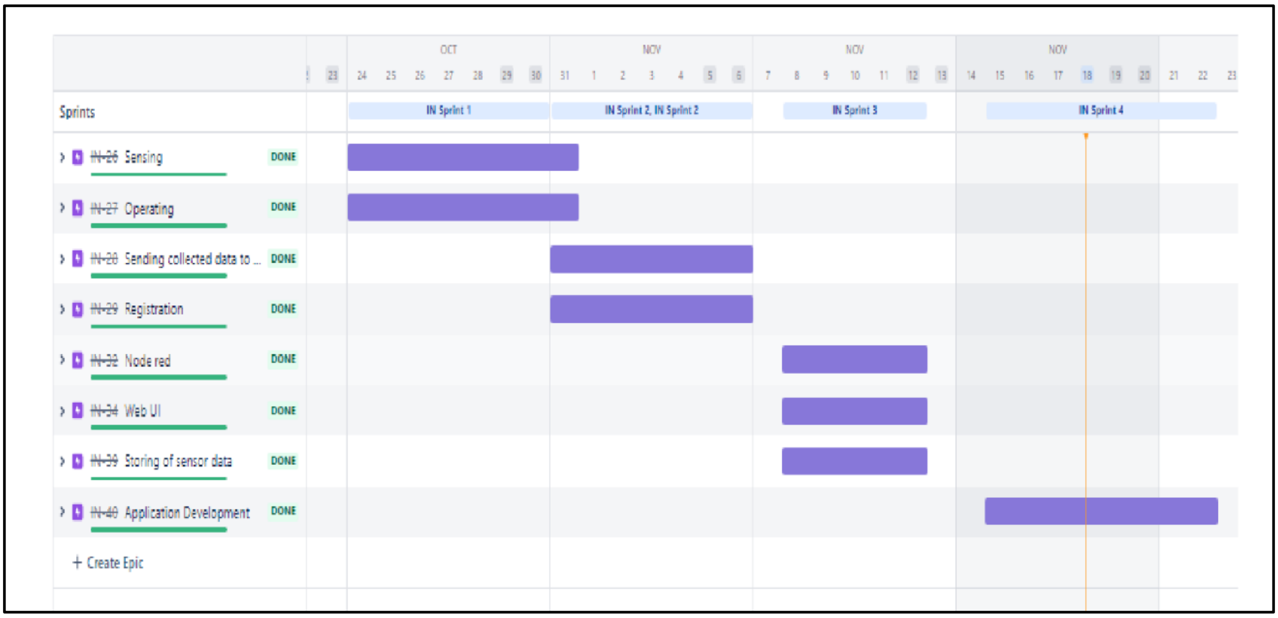
BURNDOWN CHART

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



6.3 Reports from JIIRA

Agile software development, customer service, start-ups, and businesses are just a few of the things that Jira brings teams together for. Teams can plan, assign, track, report, and manage their work with Jira's help.



Sprints

~~IN-26~~ Sensing DONE

~~IN-27~~ Operating DONE

~~IN-28~~ Sending collected data to ... DONE

~~IN-29~~ Registration DONE

~~IN-32~~ Node red DONE

~~IN-34~~ Web UI DONE

~~IN-39~~ Storing of sensor data DONE

~~IN-40~~ Application Development DONE

+ Create Epic

7. CODING & SOLUTIONING

7.1 Feature 1

IBM Cloud offers stories and dashboards for you to share your ideas and analyses. It is possible to construct a view that incorporates visual representations of the data, such as graphs, charts, plots, tables, maps, and other visual representations.

7.2 Feature 2

Investigate advanced visualisations of your data on the IBM cloud to find trends and correlations that affect your business. A dashboard enables you to keep track of events or actions at a glance by delivering crucial insights and analysis about your data on one or more pages or displays.

The modules in our work are as follows:

1. *Working with the data set*
2. *Creating charts for data visualisation*
3. *Building the dashboard*

8. TESTING

8.1 Test Cases

Testing involves looking for flaws or weaknesses in a piece of work. It offers a technique to examine the operation of separate parts, sub-assemblies, assemblies, and/or finished goods. Software testing makes ensuring that software adheres to specifications without deviating in an unacceptable way and meets user expectations. Different test types exist. Each type of test responds to the testing requirement in a unique way.

8.2 User Acceptance Testing

Users' acceptance: Any project's testing phase is vital and demands active participation from the end user. It also ensures that the system meets the functional requirements. At this point, all test cases are executed to make sure the software is accurate and complete.

Prior to the client accepting the programme, the test must be successfully completed. Once customer personnel have verified the accuracy of the preliminary production statistics load and the successful completion of the test suite, the customer formally approves delivery of this system.

9. RESULTS

9.1 Performance Metrics

Configuring ESP32 Using Wokwi Projects

```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>
const int DHT_PIN = 15;
bool is_exhaust_fan_on = false;
bool is_sprinkler_on = false;
float temperature  = 0;
int gas_ppm = 0;
int flame = 0;
int flow = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
DHTesp dhtSensor;
void setup() {
  Serial.begin(99900);
  /** sensor pin setups **/
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  //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.
  */
}
void loop() {
  TempAndHumidity  data = dhtSensor.getTempAndHumidity();
  //setting a random seed
  srand(time(0));
  //initial variable activities like declaring , assigning
  temperature  = data.temperature;
  gas_ppm = rand()%1000;
  int flamereading = rand()%1024;
  flame = map(flamereading,0,1024,0,1024);
```

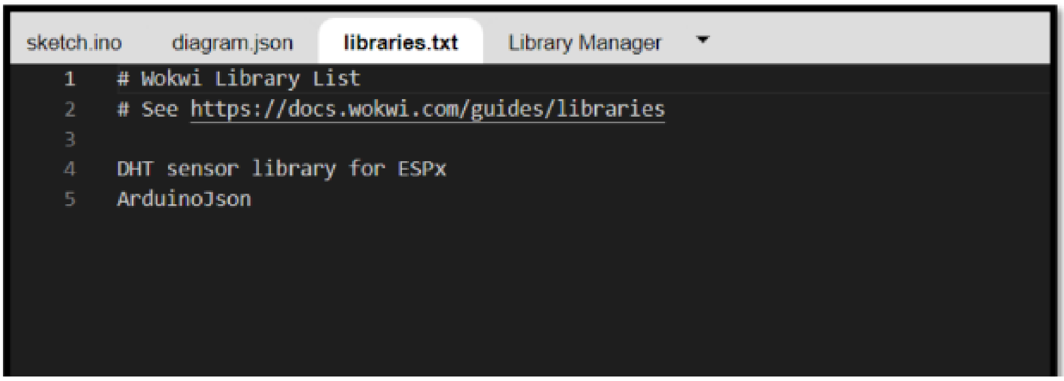
```
int flamerange = map(flamereading,0,1024,0,3);
int flow = ((rand()%100)>50?1:0);
//set a flame status based on how close it is.....
switch (flamerange) {
case 2:    // A fire closer than 1.5 feet away.
  flame_status = "Close Fire";
  break;
case 1:    // A fire between 1-3 feet away.
  flame_status = "Distant Fire";
  break;
case 0:    // No fire detected.
  flame_status = "No Fire";
  break;
}
//toggle the fan according to gas in ppm in the room
if(gas_ppm > 100){
  is_exhaust_fan_on = true;
}
else{
  is_exhaust_fan_on = false;
}
//find the accident status 'cause fake alert may be caused by some mischief activities
if(temperature < 40 && flamerange ==2){
  accident_status = "need auditing";
  is_sprinkler_on = false;
}
else if(temperature < 40 && flamerange ==0){
  accident_status = "not found";
  is_sprinkler_on = false;
}
else if(temperature > 50 && flamerange == 1){
  is_sprinkler_on = true;
  accident_status = "moderate";
}
else if(temperature > 55 && flamerange == 2){
  is_sprinkler_on = true;
  accident_status = "severe";
}else{
  is_sprinkler_on = false;
  accident_status = "none";
}
//send the sprinkler status
if(is_sprinkler_on){
  if(flow){
    sprinkler_status = "working";
  }
```

```
    else{
      sprinkler_status = "not working";
    }
  }
  else if(is_sprinkler_on == false){
    sprinkler_status = "it should not!";
  }
  else{
    sprinkler_status = "Error!!";
  }
  //Obivously the output.It is like json format 'cause it will help us for future sprints
  String out = "{";
  out+="Temperature\":"+String(temperature,2)+",";
  out+="CarbonMonoOxide\":"+String(gas_ppm)+",";
  out+="flame\":"+String(flame)+",";
  out+="is_exhaust_fan_on\":"+String((is_exhaust_fan_on)?"true":"false")+",";
  out+="is_sprinkler_on\":"+String((is_sprinkler_on)?"true":"false")+"}";

  Serial.println(out);

  delay(2000);
}
```

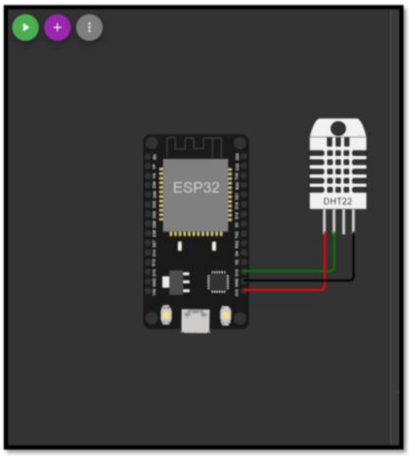
LIBRARIES

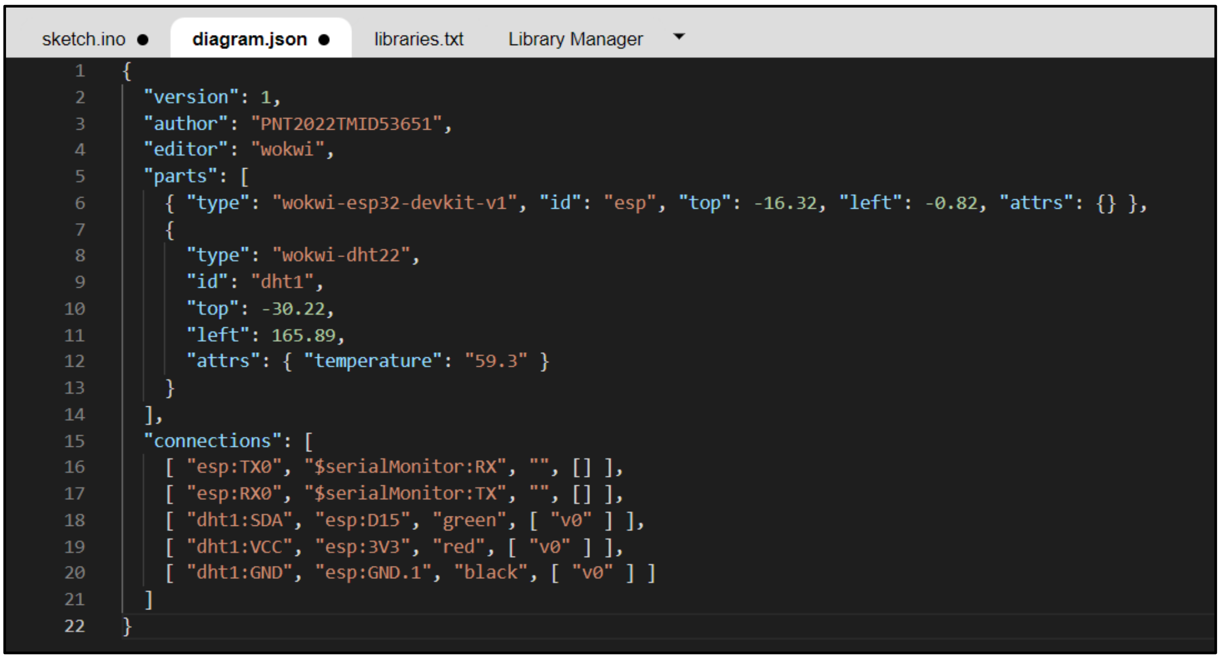


```
# Wokwi Library List
# See https://docs.wokwi.com/guides/libraries

DHT sensor library for ESPx
ArduinoJson
```

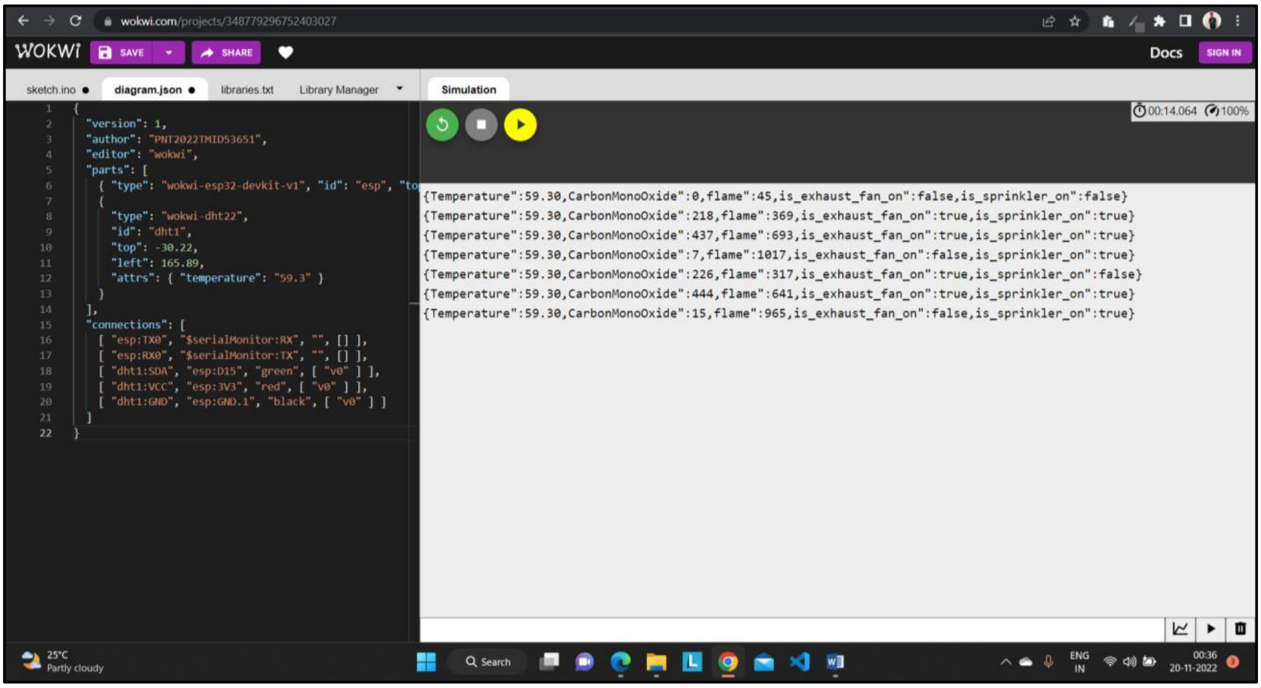
Circuit





```
{
  "version": 1,
  "author": "PNT2022TMID53651",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -16.32, "left": -0.82, "attrs": {} },
    {
      "type": "wokwi-dht22",
      "id": "dht1",
      "top": -30.22,
      "left": 165.89,
      "attrs": { "temperature": "59.3" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]
  ]
}
```

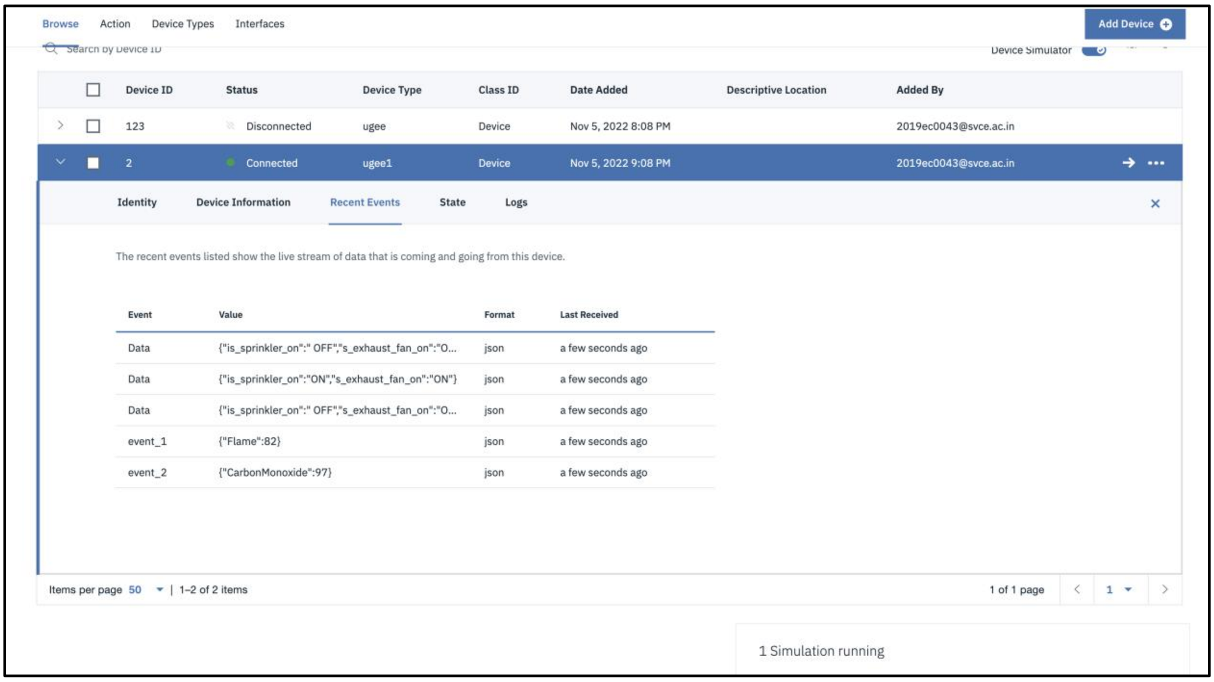
Output



```
{
  "version": 1,
  "author": "PNT2022TMID53651",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "to
    {
      "type": "wokwi-dht22",
      "id": "dht1",
      "top": -30.22,
      "left": 165.89,
      "attrs": { "temperature": "59.3" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]
  ]
}
```

```
{Temperature":59.30,CarbonMonoOxide":0,flame":45,is_exhaust_fan_on":false,is_sprinkler_on":false}
{Temperature":59.30,CarbonMonoOxide":218,flame":369,is_exhaust_fan_on":true,is_sprinkler_on":true}
{Temperature":59.30,CarbonMonoOxide":437,flame":693,is_exhaust_fan_on":true,is_sprinkler_on":true}
{Temperature":59.30,CarbonMonoOxide":7,flame":1017,is_exhaust_fan_on":false,is_sprinkler_on":true}
{Temperature":59.30,CarbonMonoOxide":226,flame":317,is_exhaust_fan_on":true,is_sprinkler_on":false}
{Temperature":59.30,CarbonMonoOxide":444,flame":641,is_exhaust_fan_on":true,is_sprinkler_on":true}
{Temperature":59.30,CarbonMonoOxide":15,flame":965,is_exhaust_fan_on":false,is_sprinkler_on":true}
```

IBM Watson Output:



	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
□	123	Disconnected	ugee	Device	Nov 5, 2022 8:08 PM		2019ec0043@svce.ac.in
■	2	● Connected	ugee1	Device	Nov 5, 2022 9:08 PM		2019ec0043@svce.ac.in

The recent events listed show the live stream of data that is coming and going from this device.

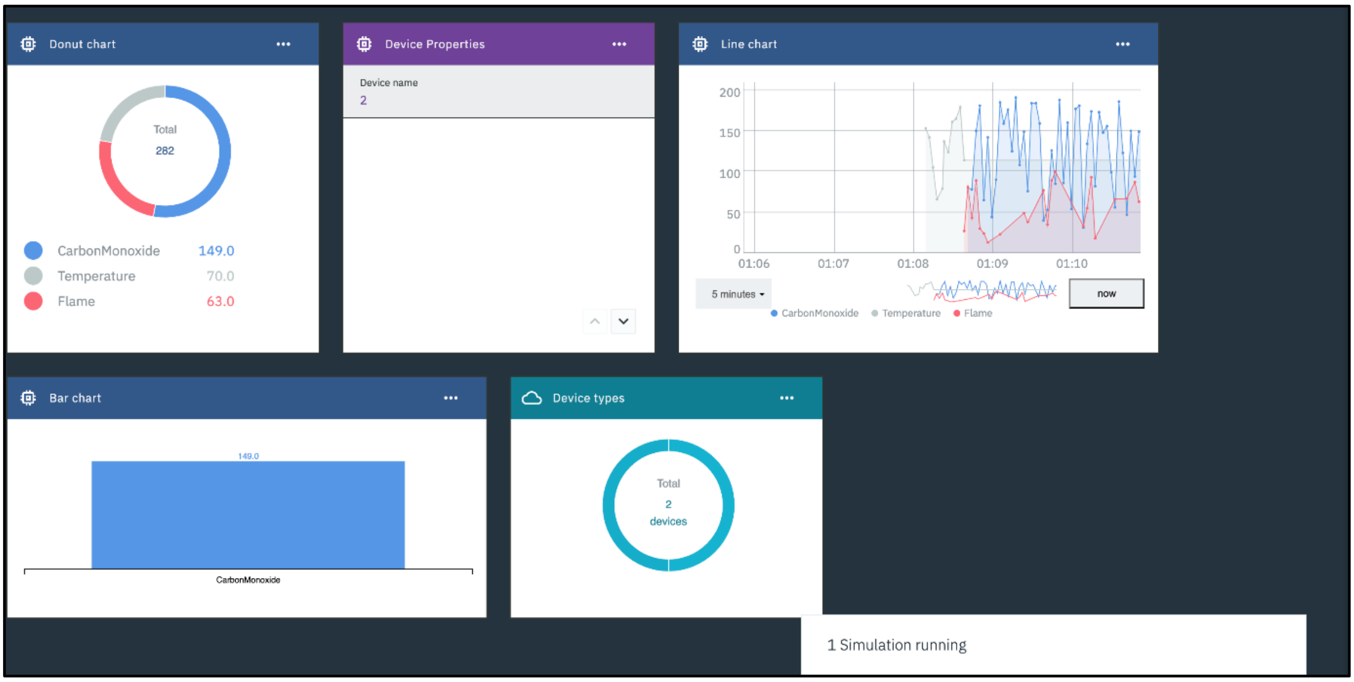
Event	Value	Format	Last Received
Data	{"is_sprinkler_on":" OFF","s_exhaust_fan_on":"O...	json	a few seconds ago
Data	{"is_sprinkler_on":"ON","s_exhaust_fan_on":"ON"}	json	a few seconds ago
Data	{"is_sprinkler_on":" OFF","s_exhaust_fan_on":"O...	json	a few seconds ago
event_1	{"Flame":82}	json	a few seconds ago
event_2	{"CarbonMonoxide":97}	json	a few seconds ago

Items per page 50 | 1–2 of 2 items

1 of 1 page 1

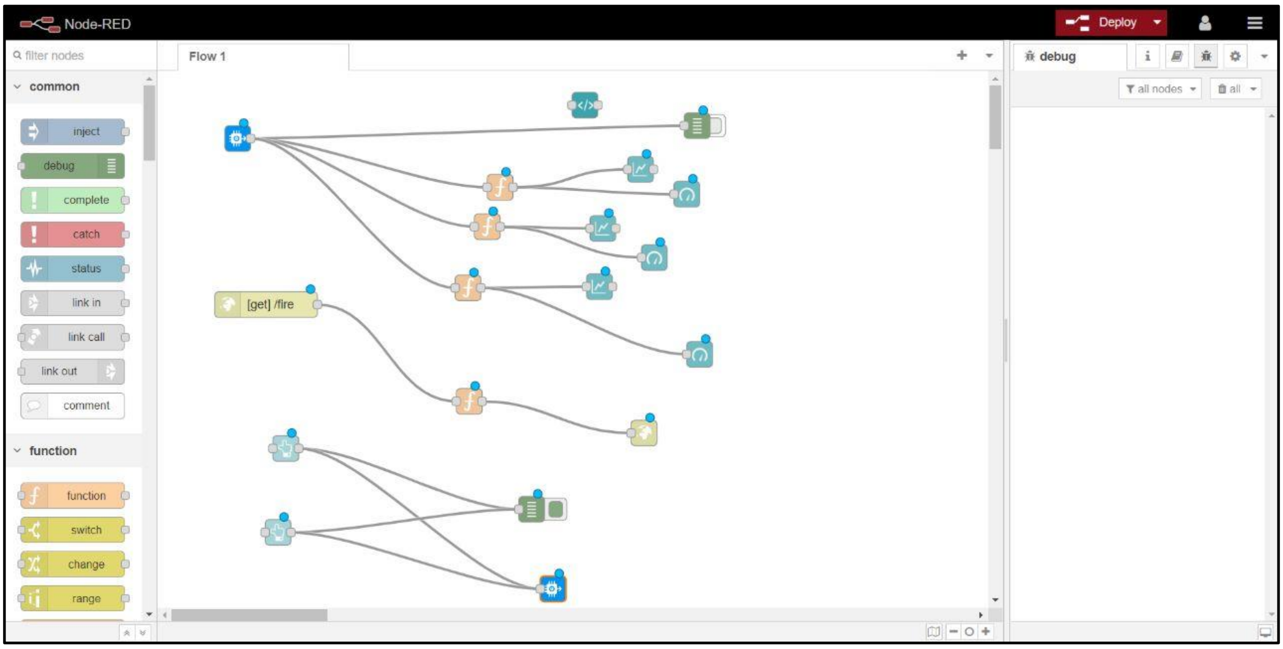
1 Simulation running

IBM Watson Dashboard:

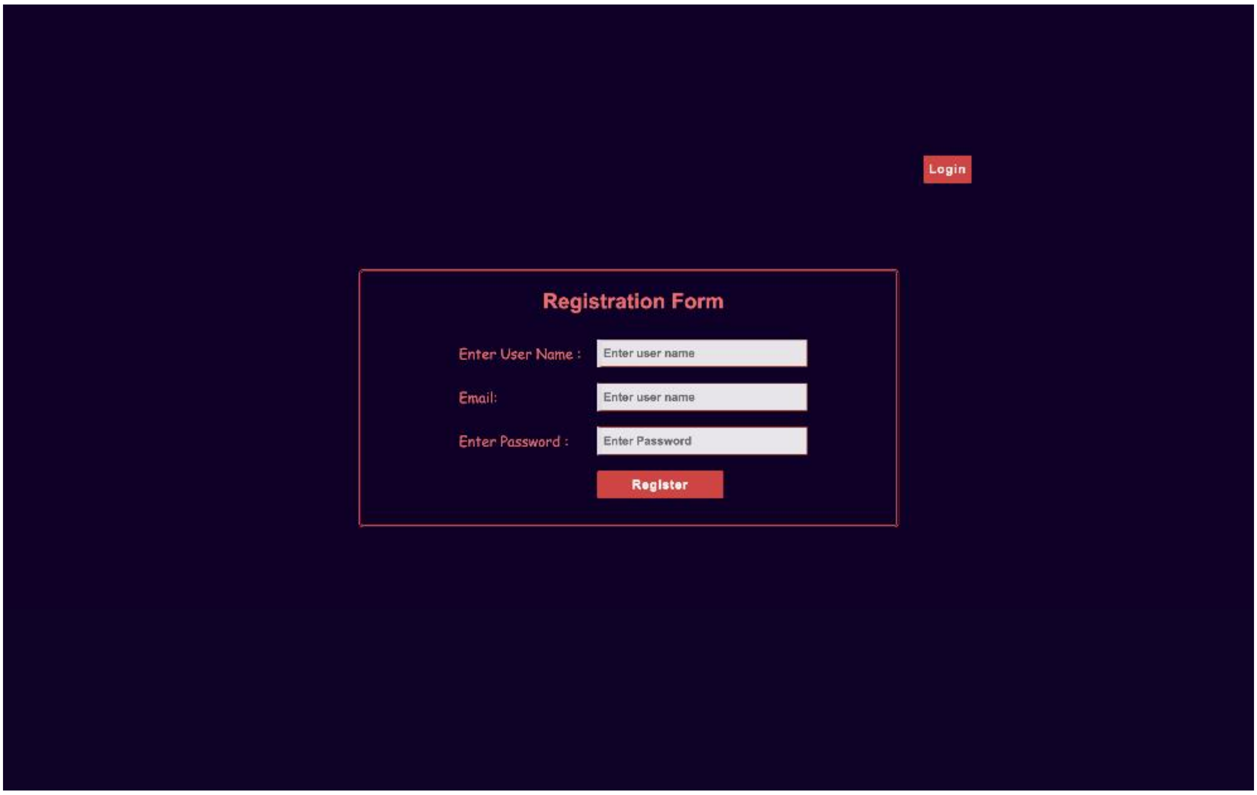


1 Simulation running

Node red



Website



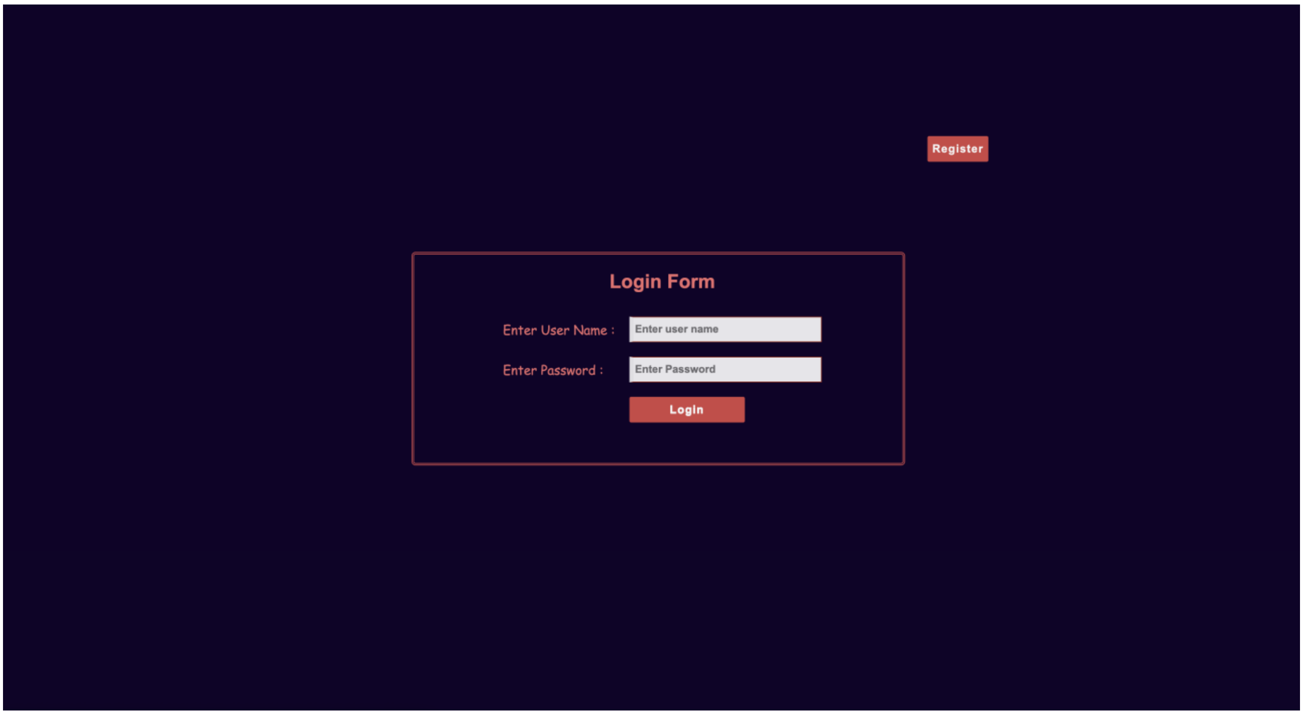
Registration Form

Enter User Name : Enter user name

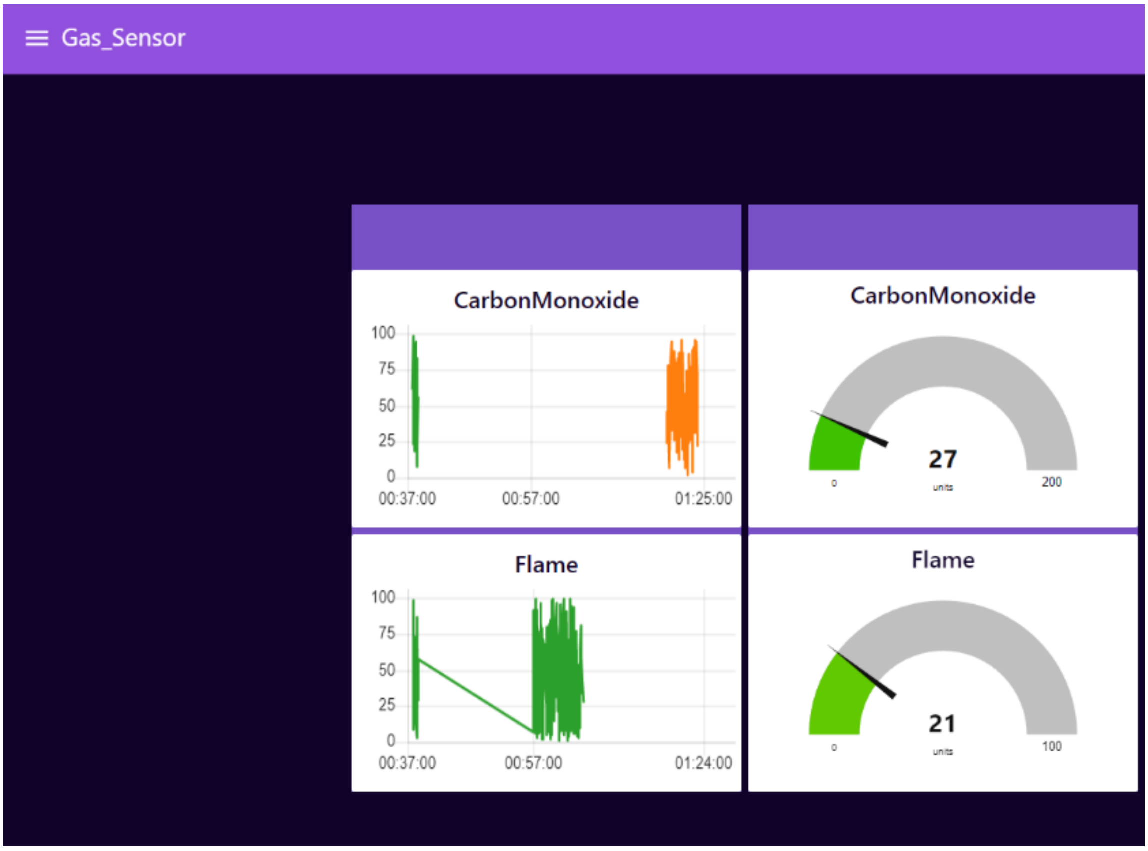
Email: Enter user name

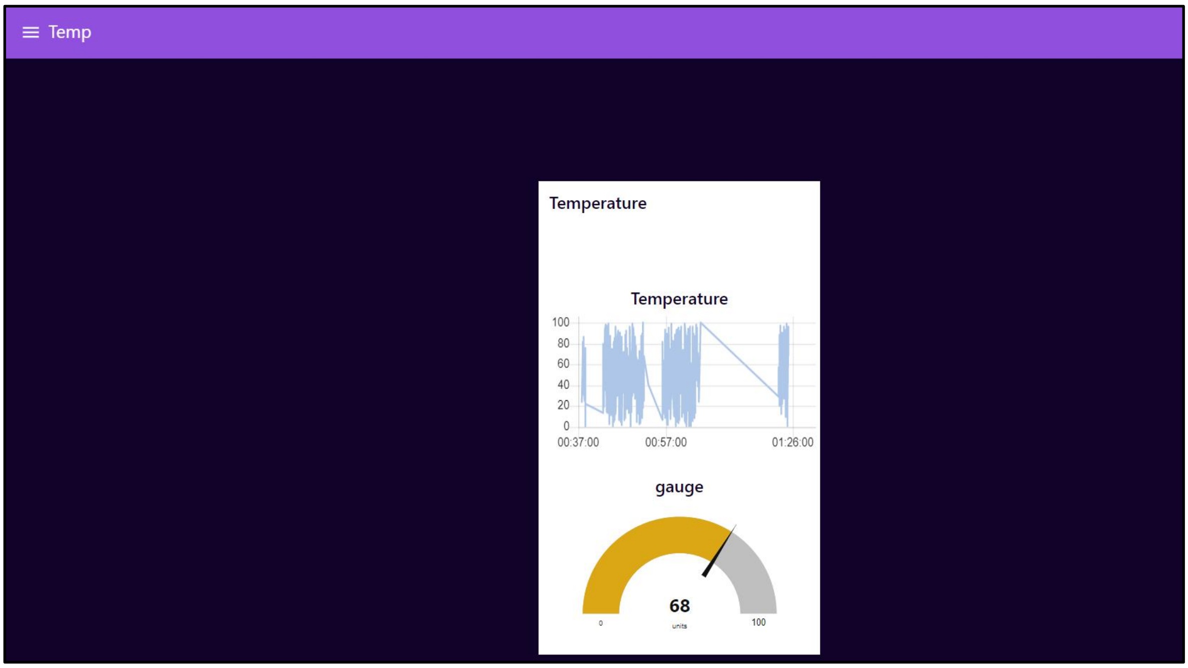
Enter Password : Enter Password

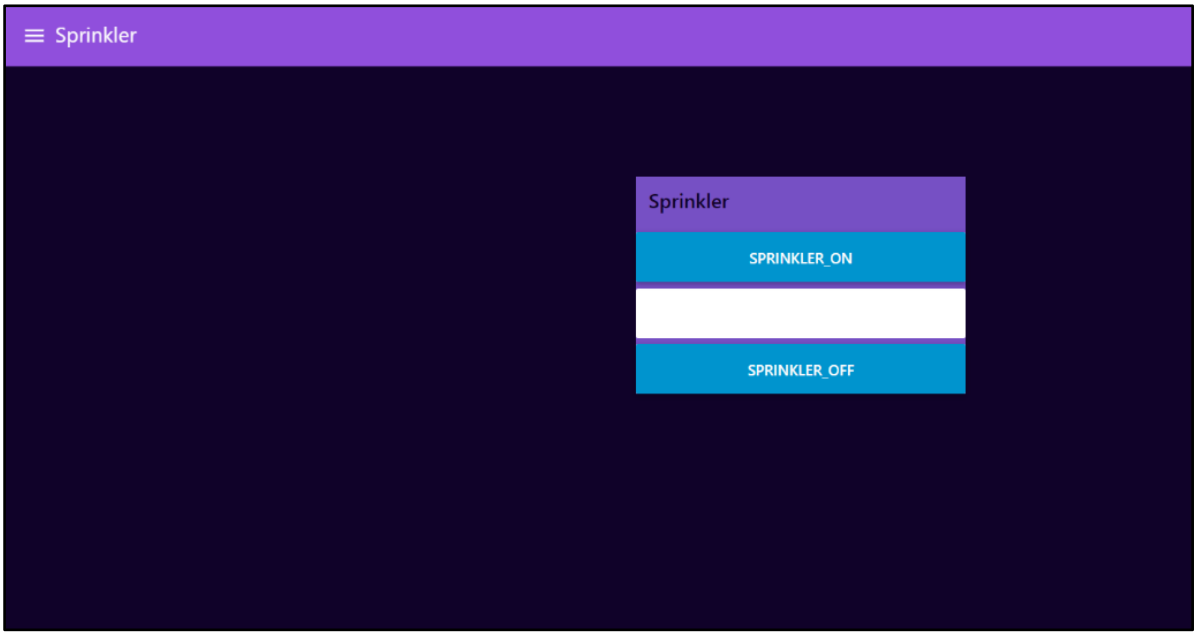
Register



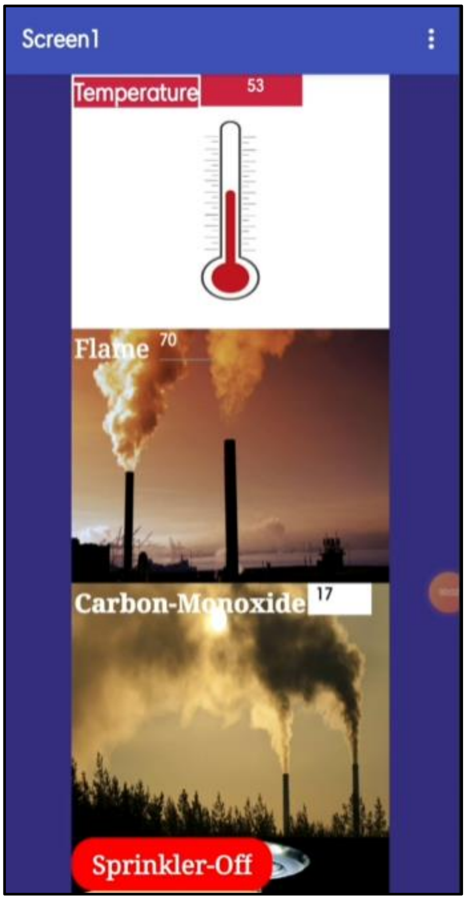
Node red dashboard




Temp
Temperature
Temperature
100
80
60
40
20
0
00:37:00
00:57:00
01:26:00
gauge
68
units
0
100


Sprinkler
Sprinkler
SPRINKLER_ON
SPRINKLER_OFF

MIT App Inventor



10. ADVANTAGES & DIS-ADVANTAGES

10.1 Advantages:

1. It is used to preventing the industries before fire accident occurred.

2. It is used to alert and warning everyone and prevent everyone against the fire detection.

3. It is connected with various components of fire sprinkler, exhaust fan to take speedy actions.

4. It reduce the risk of dying.

5. It provide early warning and it help us to make a preventive actions and it is less expensive.

10.2 Dis-Advantage:

1. It possible to detect false detection, because of industry always become heavy temperature, so it is certain difficult task.

11. CONCLUSION

The project's results suggest that the system has remarkable fire and smoke detection capabilities useful for both domestic and professional uses. This method enables us to save lives in perilous circumstances. An alert is indicated by the GSM module.

A sensor node can identify gases like propane, CO2, and others. Estimates are produced for transmission distance and power consumption. Simple approaches were used to build the sensor, and an Arduino UNO Micro controller was used.

Gas detectors can detect oxygen depletion as well as toxic, flammable, and combustible gases. To monitor manufacturing processes and cutting-edge technologies like photovoltaics, this type of equipment is often used in industry and can be found in areas like oil rigs. They might be used to put out flames.

The Techamor Y201 is a good option if you want a portable, palm-sized fire and smoke detector. This method is a cheap way to find out whether there are any gas or smoke leaks in your house or place of business.

12. FUTURE SCOPE

In this study, we leverage IOT technologies to improve current safety regulations.

The goal in developing this prototype was to completely eliminate any big or minor hazards that might have been brought on by hazardous and dangerous gas leaks into the environment.

Using IOT technology, which can perform predictive analytics on sensors and Smart Alerting protocols that involve text messaging the appropriate authority, we have developed a fire and smoke detector for society.

This system will be able to recognise smoke and gas in the immediate environment using gas sensors. This will provide protection from the most dangerous problem. This simple smoke and fire detector offers the benefits of being user-friendly and providing fire accident alerts.

Using gas sensors, this system will be able to identify smoke and gas in the surrounding area. This will shield against the main dangerous issue.

This straightforward smoke and fire detector has the advantage of being easy to use and of giving fire accident warnings.

With the ability to travel along a track that runs the length of the pipeline, a mobile gas and smoke sensing robot may be built to detect the leaking of smoke and gas for the detection of fires.

The key benefit of this method is that it turns off the cylinder's regulator knob automatically when gas and smoke leaks are discovered.

This system employs GSM technology to send alarm messages to the appropriate person if there is a gas leak and no one is home.

The GSM module is there to give the appropriate person an immediate message reporting the gas & smoke leak.

13. APPENDIX

13.1 Source Code:

```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>
const int DHT_PIN = 15;
bool is_exhaust_fan_on = false;
bool is_sprinkler_on = false;
float temperature  = 0;
int gas_ppm = 0;
int flame = 0;
int flow = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
DHTesp dhtSensor;
void setup() {
  Serial.begin(99900);
  /** sensor pin setups **/
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  //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.
  */
}
void loop() {
  TempAndHumidity  data = dhtSensor.getTempAndHumidity();
  //setting a random seed
  srand(time(0));
  //initial variable activities like declaring , assigning
  temperature  = data.temperature;
```

```
gas_ppm = rand()%1000;
int flamereading = rand()%1024;
flame = map(flamereading,0,1024,0,1024);
int flamerange = map(flamereading,0,1024,0,3);
int flow = ((rand()%100)>50?1:0);
//set a flame status based on how close it is.....
switch (flamerange) {
case 2:    // A fire closer than 1.5 feet away.
  flame_status = "Close Fire";
  break;
case 1:    // A fire between 1-3 feet away.
  flame_status = "Distant Fire";
  break;
case 0:    // No fire detected.
  flame_status = "No Fire";
  break;
}
//toggle the fan according to gas in ppm in the room
if(gas_ppm > 100){
  is_exhaust_fan_on = true;
}
else{
  is_exhaust_fan_on = false;
}
//find the accident status 'cause fake alert may be caused by some mischief activities
if(temperature < 40 && flamerange ==2){
  accident_status = "need auditing";
  is_sprinkler_on = false;
}
else if(temperature < 40 && flamerange ==0){
  accident_status = "not found";
  is_sprinkler_on = false;
}
else if(temperature > 50 && flamerange == 1){
  is_sprinkler_on = true;
  accident_status = "moderate";
}
else if(temperature > 55 && flamerange == 2){
  is_sprinkler_on = true;
  accident_status = "severe";
}else{
  is_sprinkler_on = false;
  accident_status = "none";
}
//send the sprinkler status
if(is_sprinkler_on){
    if(flow){
      sprinkler_status = "working";
    }
    else{
      sprinkler_status = "not working";
    }
  }
  else if(is_sprinkler_on == false){
    sprinkler_status = "it should not!";
  }
  else{
    sprinkler_status = "Error!!";
  }
  //Obivously the output.It is like json format 'cause it will help us for future sprints
  String out = "{";
  out+="Temperature\":"+String(temperature,2)+",";
  out+="CarbonMonoOxide\":"+String(gas_ppm)+",";
  out+="flame\":"+String(flame)+",";
  out+="is_exhaust_fan_on\":"+String((is_exhaust_fan_on)?"true":"false")+",";
  out+="is_sprinkler_on\":"+String((is_sprinkler_on)?"true":"false")+"}";
  Serial.println(out);
  delay(2000);
}
```

13.2. GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-3308-1658545235