Industry-Specific Intelligent Fire Management System

PROJECT REPORT

TEAM ID -PNT2022TMID27352

SUBMITTED BY: (KCG COLLEGE OF TECHNOLOGY)

T.KAVYA-311019106029

S.KEERTHANA – 311019106031

I.KIRTHIKAA - 311019106031

M.LOGESHWARI-311019106033

Project Report Format

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. **PROJECT PLANNING & SCHEDULING**
   - 6.1 Sprint Planning & Estimation
   - 6.2 Sprint Delivery Schedule
   - 6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   - 7.1 Feature 1
   - 7.2 Feature 2
   - 7.3 Database Schema (if Applicable)
8. **TESTING**
   - 8.1 Test Cases
   - 8.2 User Acceptance Testing
9. **RESULTS**
   - 9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

    Source Code

    GitHub & Project Demo Link

1. INTRODUCTION

1.1 Project Overview

Industry specific fire management system is designed to protect industrial facilities from harmful fires through a combination of smart sensors and automatic fire suppression.

Automated ventilation: The system can automatically ventilate the area if it detects smoke or fumes. This helps reduce the risk of suffocation in case there is a fire, and it also reduces the risk of inhaling toxic fumes that may be produced by burning plastic or wood.

Smart sensors: The system uses advanced sensors that can detect heat and flames, as well as carbon monoxide. This allows you to rest easy knowing your home is protected against both common household fires and potentially deadly carbon monoxide leaks.

Automatic alerting: If there is a fire in your home, our smart system will automatically send out alerts via text message and email to let you know about it

First, it helps with the detection of a fire. Smart fire extinguishers are already equipped with sensors that detect heat and smoke. When a fire is detected, the smart fire extinguisher will automatically deploy itself and start spraying chemicals in the direction of the heat source. Since smart fire extinguishers are also connected to a smart security system, their activity is automatically reported to the system and alerts can be sent out to emergency services and other devices like an indoor sprinkler system or an automated ventilation system.

Second, it helps identify which parts of a building are protected from a given fire. Because smart fire extinguishers report their location when they deploy themselves, you can see on your smart security system’s map which parts of your building are protected (that is, which parts have had their fires put out by an active nearby smart fire extinguisher) and which have not had their fires put out yet.

Third, it helps automate inventory management for fire extinguishers. Smart security systems can keep track of all deployed smart fire extinguishers and send out alerts if any of them need to be refilled or replaced due.

1.2 Project Purpose

The primary purpose of fire alarm system is to provide an early warning of fire so that people can be evacuated & immediate action can be taken to stop or eliminate of the fire effect as soon as possible. Alarm can be triggered by using detectors or by manual call point (Remotely). To alert/evacuate the occupants siren are used. The suggested technique in Fire alarm system used the addressable detectors units besides using the wireless connection between the detector in zones as a slave units and the main control unit as the master unit. The system shall include a control panel, alarm initiating devices, notification appliances, and the accessory equipment necessary for a complete functioning fire alarm system.

A properly designed, installed, operated, and maintained fire alarm system can reduce the losses associated with an unwanted fire in any building. These losses include property and, more importantly, human life. The primary motivation for fire alarm system requirements in building and fire codes is to provide early notification to building occupants so they can exit the building, and to notify the fire service so it can respond to the fire. In settings such as hospitals the fire alarm system provides notification to staff so they can respond to the fire emergency (as opposed to evacuating the building). This module will explain the basic features of fire alarm systems and the inspection of these systems. It should be noted that fire alarm systems also are called "protective signaling systems," especially in NFPA documents and in other codes and standards.

2. LITERATURE SURVEY:

2.1 Existing problem :

Fire and smoke kill more people every year than many other forces. While controlled fire serves us in so many instances, uncontrolled fire can be of harm, however, the rapid detection of fire and its control can save lives and property damage worth millions. Conventional and addressable are two main types of fire alarm systems, but unfortunately, these fire alarm systems often generate false alarms. The ratio of false alarm is higher in conventional alarm systems compared to addressable, but addressable alarm fire systems are more expensive. The most likely cause of a false warning is different for distinct types of detection systems, such as a smoke sensor often being activated falsely due to an environmental effect. There were eight total

mistakes which can be concluded from these fire accidents. These included lack of installation of automatic systems, problems in enforcement of law, poor planning, maintenance and management in fire safety issues, poor performance of hospital staff, combustible construction material, poor post-fire accident management, problems caused by authority and conflicts of security and fire safety.

2.2 References:

Paper 1:

TITLE: Prototype of fire symptom detection system
AUTHORS: Oxsy Giandi ,Riyanarto
PUBISHED: 2018 International Conference on Information and Communications Technology (ICOIACT)

DESCRRIPTION:
One of smart home function is fire alert detection. The symptom detection of fire in the house is important action to prevent the mass fire and save many things. This research applies the new system of fire detection using gas leak concentration to predict the explosion and fire earlier called fire predictor and the fire appearance detector. The fire predictor just show the gas leak concentration and make an alarm rang. The fire detector use fuzzy system to make the fire detector classification. The output simulation system can send the data to MFC, but the MFC reader cannot parse it in real time.

Paper 2:

TITLE: Raspberry pi based smart fire management system employing sensor based automatic water sprinkler
AUTHORS: Jaspreet Singh; Ekambir Sidhu Noorinder
Published in: 2017 International Conference on Power and Embedded Drive Control (ICPEDC)

DESCRIPTION:
The smart fire management requires significant surveillance systems to detect and control the fire and fire causing agents automatically. This requirement has been accomplished in this proposed system by employing fire detection system and fire controlling system using Raspberry pi. The fire detection system entails flame

detectors along with temperature sensors which reduces the false fire detection rate. The system also notifies the user by emailing the video of fire affected area and gives the updates of room temperature from time to time. A gas leakage sensor has been employed to detect various types of gases like ethane, methane, LPG etc. This proposed system can be installed in a hall of maximum area 126ft × 21ft employing one Raspberry pi module.

Paper 3:

TITLE: Smart apparatus for fire evacuation -An IoT based fire emergency monitoring and evacuation system
AUTHORS: Swarnadeep Majumder;Sean O'Neil;Ryan Kennedy
PUBLISHED: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC)
DESCRIPTION:

According to the National Fire Protection Agency, 14,500 highrise fires occur every year causing 40 deaths and 520 injuries per year on average [1]. Many of these deaths and injuries can presumably be avoided if evacuees had knowledge of the location of the fire and a solid exit strategy. In an age of IoT devices, there are no such commercial products available to address this problem. We have developed a technology that can be used as a smart fire defense guidance system. The goal is to inform occupants and emergency services of the location of the fire and provide a real-time safe path of evacuation. It proposes to use a mesh network of smart fire alarms and path planning algorithms to provide these essential services.

Paper 4:

TITLE: Smart Fire Detection System in a Large Building Using Lora WAN
AUTHORS : K. Gokulakrishnan, A.Mohammed Ashim, J.Manoj Kumar

PUBLISHED :2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)

DESCRIPTION:

Nowadays, it has been an important issue to evacuate mass occupants for safety from large public buildings under emergency conditions. However, evacuation design in public buildings still remains in a static plan, which is difficult to deal with diverse

types of hazards and problems of uneven personnel distribution. Thus, a set of guidelines for safety design of large public buildings, or even a set of dynamical emergency evacuation systems, is necessary. In this project, an emergency evacuation system for large public buildings is proposed based on intelligent building by means of embedded system the main objective is to reduce the fire accident in the high rise building. This system is to guide the user and updating the current scenario of the place via android app.

Paper 5:

TITLE: Implementation of Intelligent Residential Fire Extinguisher System
AUTHORS: C. Bhuvaneswari,M. Kavitha,W.Abitha Memala,M. Pushpavalli
PUBLISHED: 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)

DESCRIPTION:

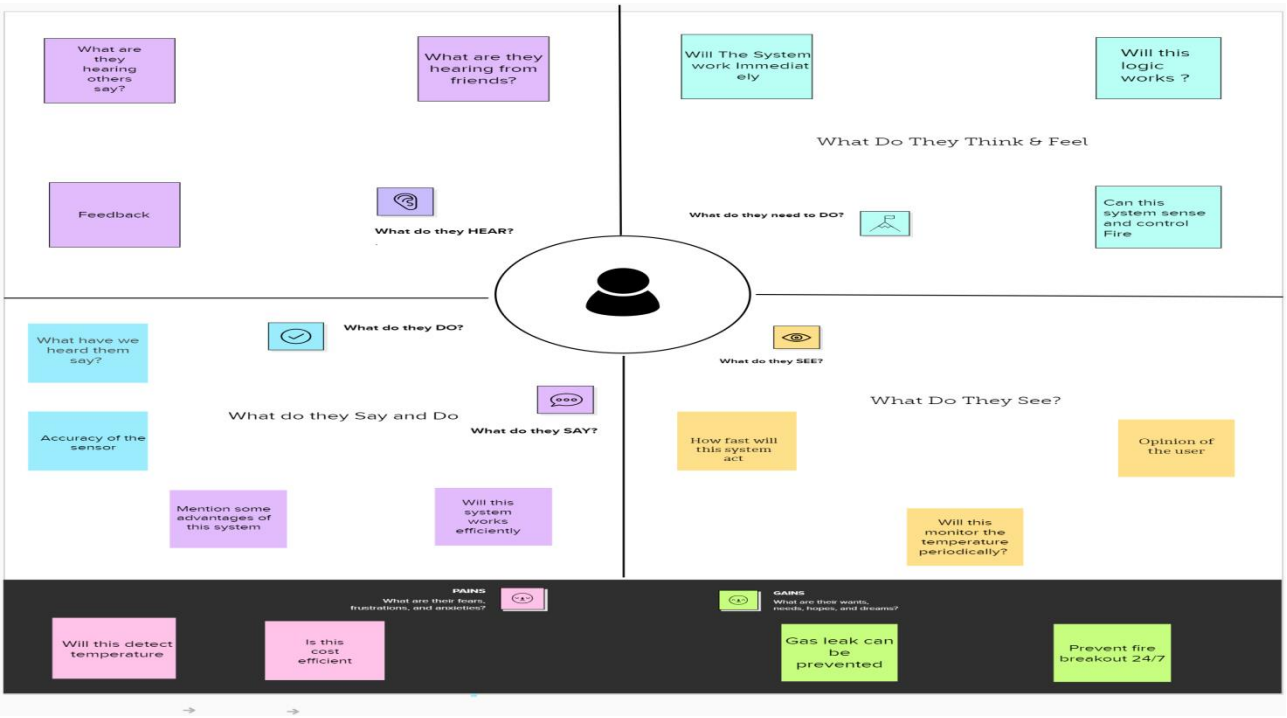
A new generation Intelligent fire extinguishing systems have been proposed to rescue humans and important documents which would be flamed. The Automatic Fire Extinguisher system can be controlled by Smartphones through Internet of things. It is a hardware-based model, accustomed automatically to extinguish the fireplace. Instructions may be given to the robot concerning its movement, turning on its water pump. It starts to manoeuvre within the direction with reference to fire intensity once when it senses the heat. The motor speed is adjusted with regard to the temperature range to pump the water. So that this robot pumps the accurate amount of water which is required to extinguish the fireplace. Proteus is used for Simulation of Fire Extinguisher and Python is used for coding in the Proteus.

2.3 Problem Statement Definition

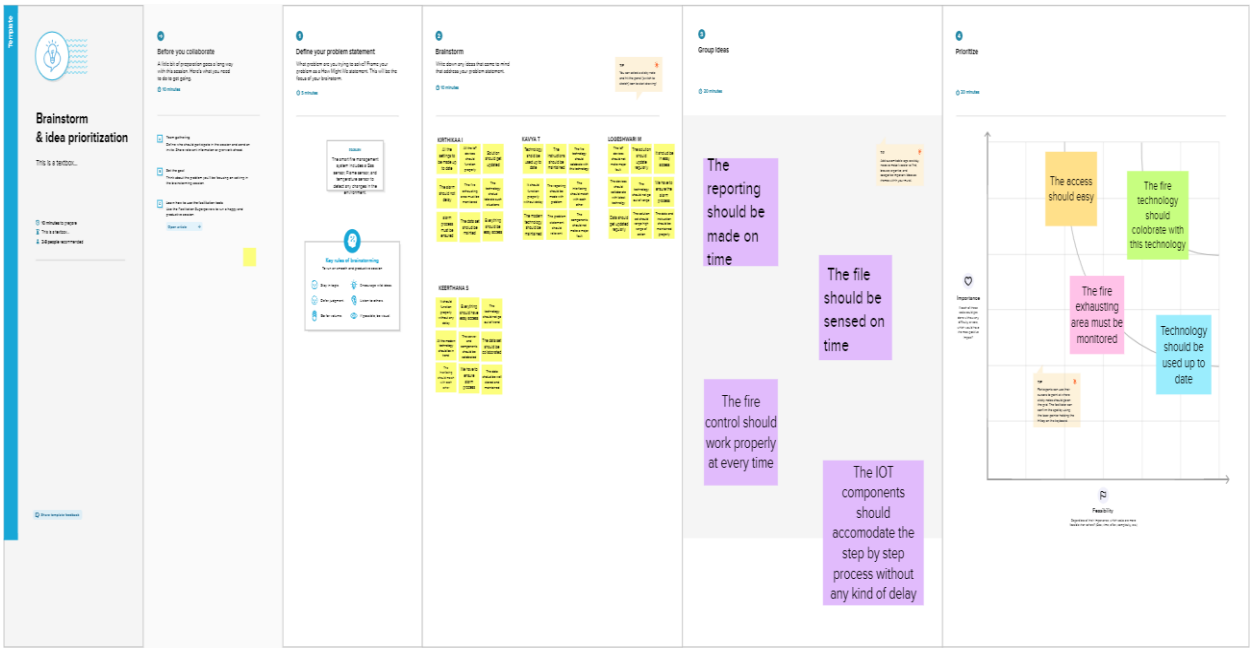
In emergency situations, such as building fires, the rescue team must quickly undertake the most appropriate intervention strategy aimed at preventing loss of human life. Emerging hardware and software technologies such as sensors, embedded systems, drones, Cloud, Artificial Intelligence and Building Information Modelling tools, offer exceptional opportunities in this context. In particular, the attempt to combine and integrate the Internet of Things technologies is very interesting in the context of future smart cities .

3. IDEATION & PROPOSED SOLUTION

3.1Empathy Map Canvas



3.2 Ideation & Brainstorming



3.3Proposed Solution

Proposed Solution Template:

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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	House combustion is one of the main concerns for builders, designers, and property residents. Singular sensors were used for a long time in the event of detection of a fire, but these sensors can not measure the amount of fire to alert the emergency response units.
2.	Idea / Solution description	This project proposes to establish a system that can detect fire and extinguish it in the shortest time subject to a few effective factors. In this case, the system aims to put out the fire before it spreads increasing the security of home, laboratory, office, factory and building that is important to human life.
3.	Novelty / Uniqueness	The uniqueness of our proposed project is the fire extinguisher system can find out the fire source by the proposed method and move to fire source to fight the fire using extinguisher.
4.	Social Impact / Customer Satisfaction	Our project helps save countless lives and has gone through a variety of changes to its design to further increase its efficiency and ability to put out a .
5.	Business Model (Revenue Model)	In order to gain a competitive market share, the companies operating in the market are involved in active innovation, by providing advanced fire detection systems.
6.	Scalability of the Solution	In compare to that fire extinguisher our fire fighting system is not too much costly. So it is more profitable.

3.4 Problem Solution fit

Project Title: Industry-specific intelligent fire management system **Project Design Phase-I - Solution Fit Template** **Team ID:** PNT2022TMID27352

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

- Industries
- Hospitals
- Schools and Colleges
- Public places

5. CUSTOMER CONSTRAINTS

- Installation cost
- Area required for installation
- No Awareness

8. AVAILABLE SOLUTIONS

- Foams
- Water types
- Wet chemical and water additives
- Dry powder and other agents for metal fires

Explore AS, differentiate

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

2. JOBS-TO-BE-DONE / PROBLEMS

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

6. PROBLEM ROOT CAUSE RC

- Welding, Cutting, Soldering and other Hot Work.
- Failure of Machinery and other equipment's.
- Smoking in Prohibited Areas.
- Chemical Reaction/Runway Reaction.

9. BEHAVIOUR BE

- Customer will try to contact the respective sector persons for this system installation
- They seek help from other social media and from their friends

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

3. TRIGGERS TR

Fire safety is a fundamental consideration in building design and management.

4. EMOTIONS: BEFORE / AFTER

Before:
Lack of Awareness.

Fear regarding Safety.

After:

Gained some knowledge.

7. YOUR SOLUTION

- Provide adequate means of escape.
- Install smoke detection systems.
- Conduct regular fire drills.

10.CHANNELS of BEHAVIOUR

ONLINE: Research on this topic

OFFLINE: Get educated on the topic

4. REQUIREMENT ANALYSIS

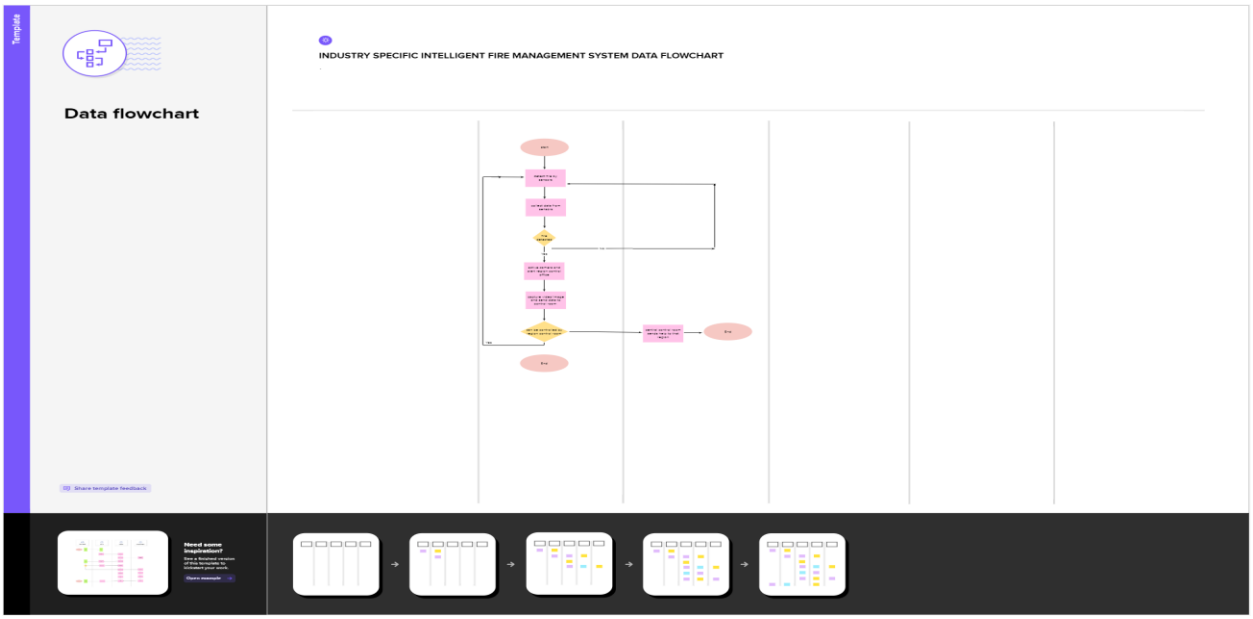
4.1 Functional Requirement:

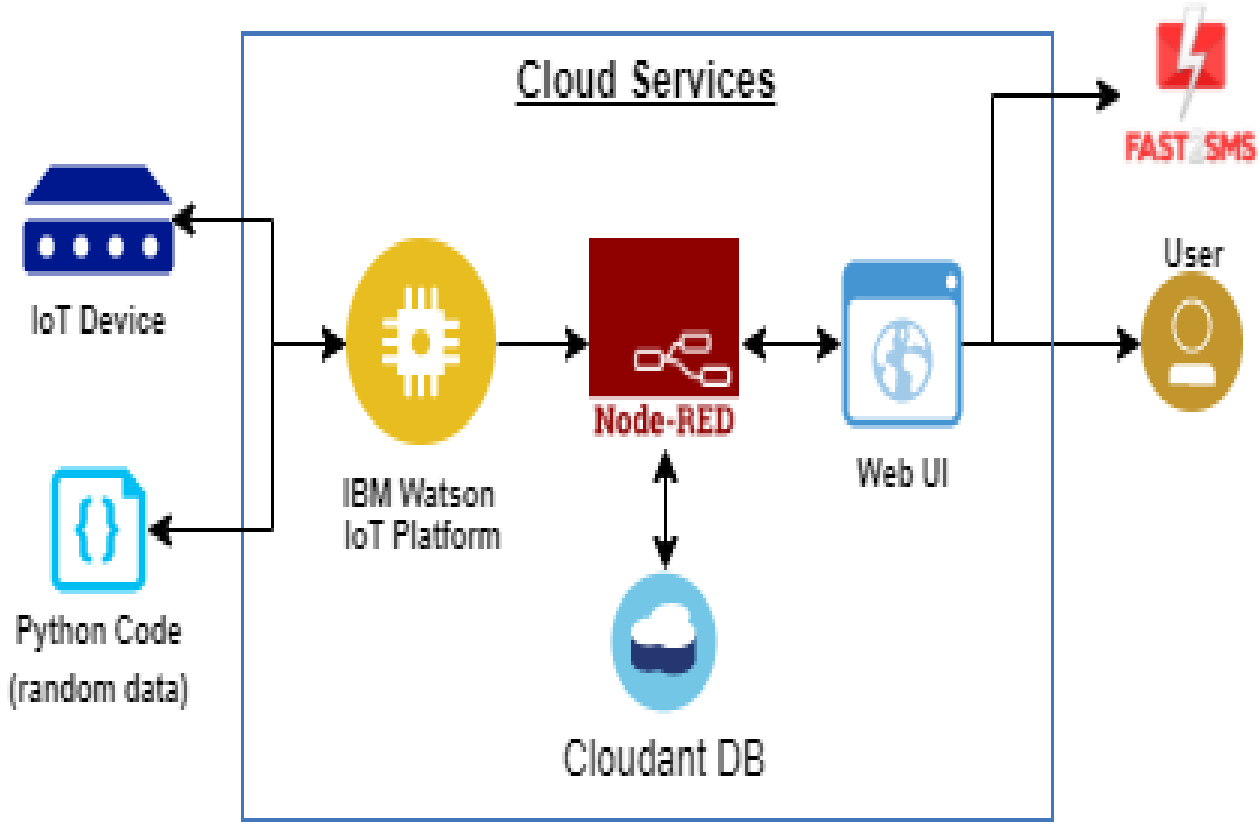
FR no	Functional Requirement(Epic)	Sub Requirement (Story / Sub-Task)
FR_1	User Registration	Registration through website or application Registration through Social medias Registration through Linkedin
FR_2	IJser Confirm action	Verification via Email or OTP
FR_3	User Login	Login through website or App using the respective username and password
FR_4	User Access	Access the app requirements
FR_5	User Upload	User should be able to upload the data
FR_6	User Solution	Data report should be generated and delivered to user for every 24 hours
FR_7	User Data Sync	API interface to increase to invoice system

4.2 Non-Functional requirement

NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	Usability requirements includes language barriers and localization tasks. Usability can be assessed by Efficiency of use.
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 a VoLTE mobile connection.
NFR-5	Availability	New module deployment must not impact front page, product pages, and check out pages availability and mustn't take longer than one hour.

5. PROJECT DESIGN
5.1 DATA FLOW





5.2 Solution & Technical Architecture

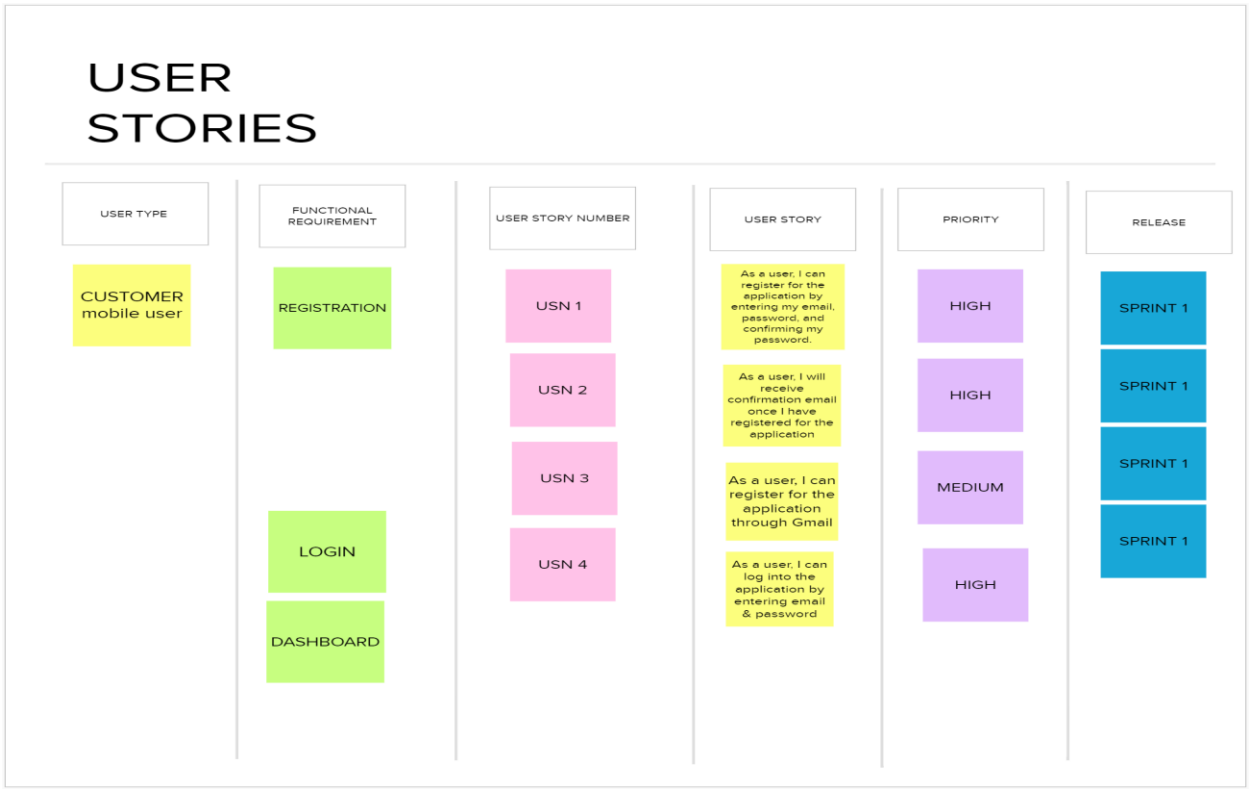
Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Node-RED, MIT app	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 cloud 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 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	MIT app
5.	Performance	Detects the Fire, gas leak, temperature	sensors

5.3User Stories



6.PROJECT PLANNING & SCHEDULING

6.1Sprint Planning & Estimation

Project Planning Phase

Milestone & Activity List

Date	**29 October 2022**
Team ID	**PNT2022TMID27352**
Project Name	**INDUSTRY SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM**

TITLE	DESCRIPTION	DATE
Literature Survey on The Selected Project and Information Gathering	A Literature Survey is a compilation summary of research done previously in the given topic. Literature survey can be taken from books, research paper online or from any source.	3rd September 2022
Prepare Empathy Map	Empathy Map provides easy way for team to initialize and better understand the target users.	10th September 2022
Ideation-Brainstorming	Brainstorming is a activity that will help to generate more innovative ideas. The process of coming up with new ideas.	17th September 2022
Define Problem Statement	A Problem Statement is a concise description of the problem or issue to be addressed or conditions to be improved. The problem statement identifies the gap between the problem state and desired state of the process.	18th September 2022
Problem Solution Fit	The problem solution fit helps us to understand the thoughts of the customer about certain job, pain, gain like behaviors and emotions.	1st October 2022
Proposed Solution	Proposed solution shows relate the current solution to desired results and describe the benefit that will accrue when the desired result is achieved.	24th September 2022
Solution Architecture	Solution Architecture is helps to ensure that a new system will fit the existing enterprise environment. It helps to understand that features to complete our project.	1st october 2022

Customer Journey	The customer journey map is a visual overview of how customer interact with and experience our website, services.	8th October 2022
Functional Requirement	Here functional and nonfunctional requirements are explained. It has specific features like usability, security, reliability, performance, availability and scalability.	15th October 2022
Data Flow Diagrams	Data Flow Diagram is a graphical representation of a Flow of data through an information system.	15th October 2022
Technology Architecture	Technology Architecture provides an overview of the various components of the system and how they work together. It is used for better decision making and understanding.	15th October 2022
Prepare Milestone & Activity List	It helps us to understand and evaluate our own progress and accuracy so far.	22nd October 2022
Spring Delivery Plan	A project sprint in scrum is a short period of time where in a development team works to complete specific tasks, milestones or deliverables.	In Progress

6.2 Sprint Delivery Schedule

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

Use the below template to create product backlog and sprint schedule

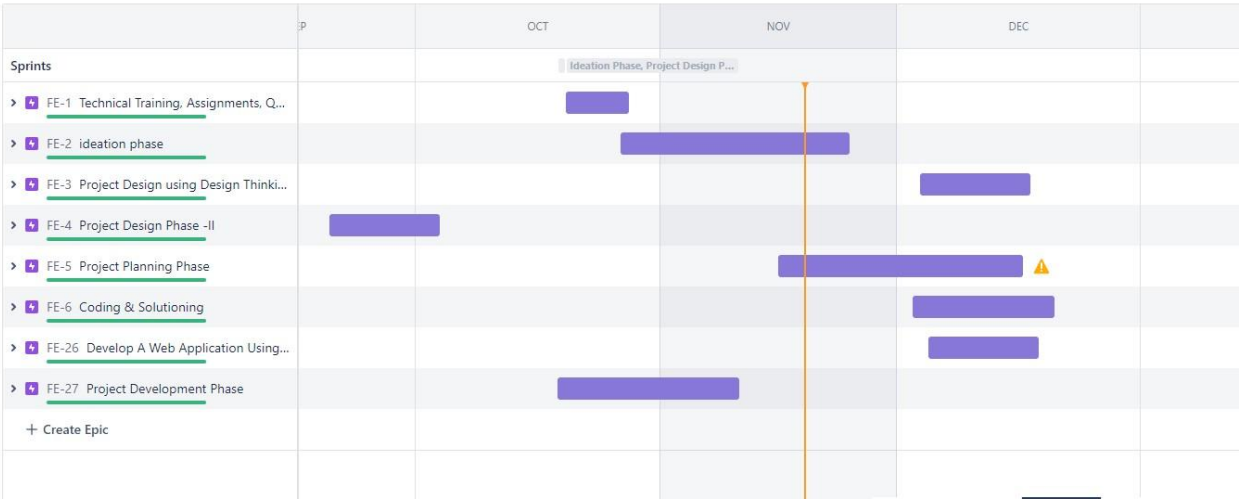
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Sensing	USN-3	Sensing the surrounding environment using the sensors	2	High	Kirthikaa , Keertha na
Sprint-1	Extinguish	USN-4	Turning on the exhaust fan as well as the fire sprinkler system in cause of fire	2	High	Logeshwari,Kavya
Sprint-2	Sending Data to the ibm Not platform	USN-5	Sending the data of the sensor form the microcontroller to the IBM Watson Dot platform	1	Medium	Kirthikaa , Keertha na
Sprint-3	Node-red	USN-6	Sending the data from the ibm Watson to the node-red for further process the data	3	High	Logeshwari, Kavya
	Storing of sensor data	USN-7	Storing the received sensor data in a cloud Database	1	Low	Kirthikaa , Keertha na
Sprint-4	Monitoring the environment	-USN 1	User can monitor the situation of the environment from a dashboard that displays sensor information about the environment	1	Medium	Logeshwari, Kavya

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Turn on/off the exhaust and sprinkler system	-USN 2	User can turn of the Exhaust fan as well as the sprinkler system if need in that situation	2	Medium	Logeshwari, Kavya
	Event Notification	-USN 8	Sending an alert SMS to the fire authority in case of fire	2	High	Logeshwari, Kavya

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

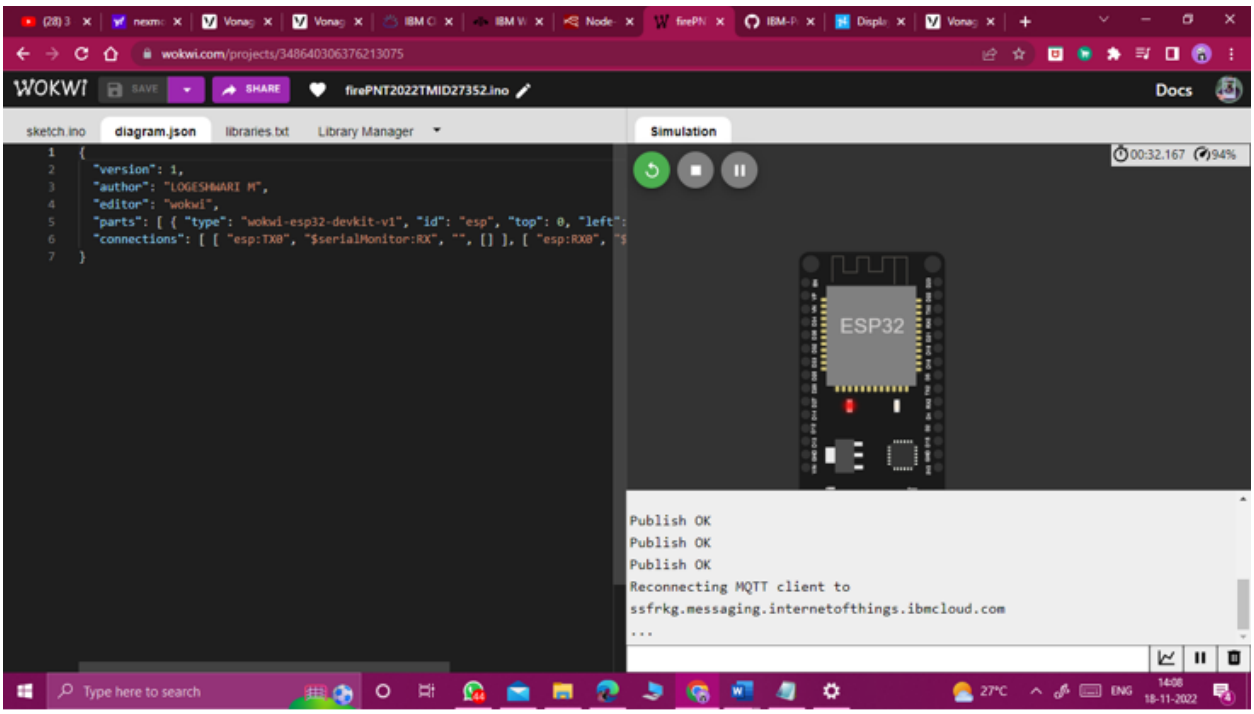
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	4	6 Days	24 Oct 2022	29 Oct 2022	4	29 Oct 2022
Sprint-2	1	6 Days	31 Oct 2022	05 Nov 2022	1	05 Nov 2022
Sprint-3	4	6 Days	07 Nov 2022	12 Nov 2022	4	12 Nov 2022
Sprint-4	5	6 Days	14 Nov 2022	19 Nov 2022	5	19 Nov 2022

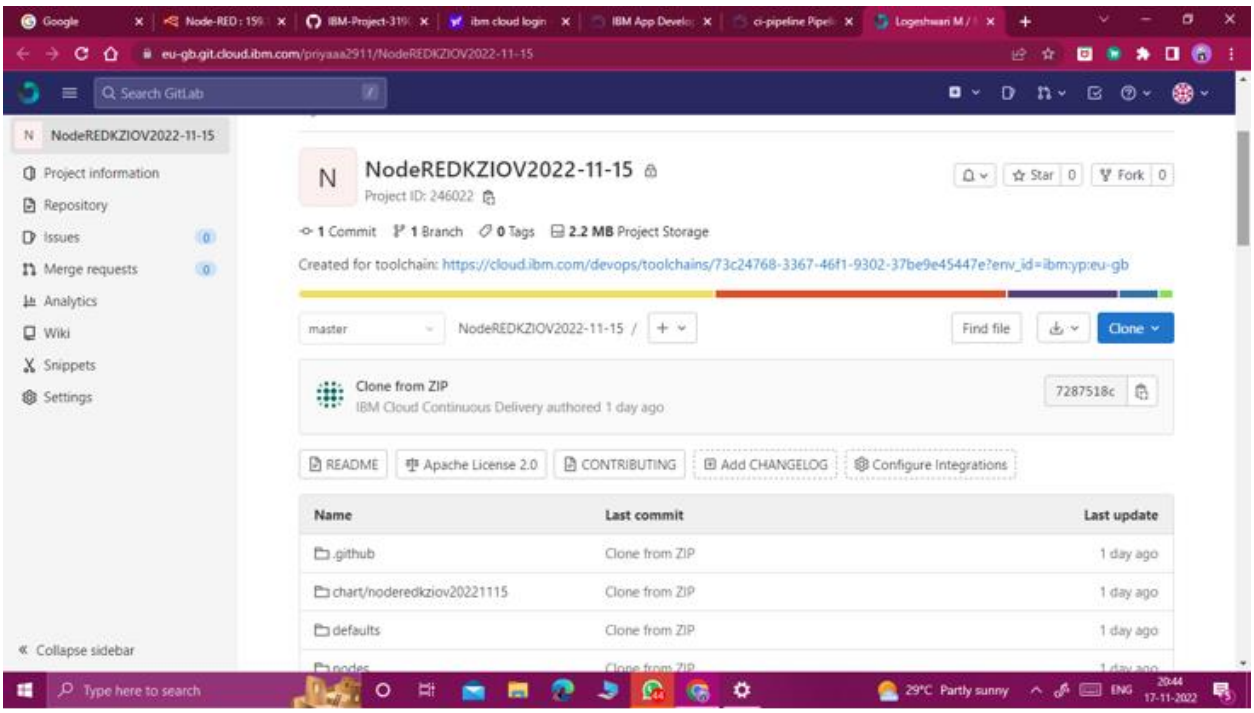
6.3 Reports from JIRA

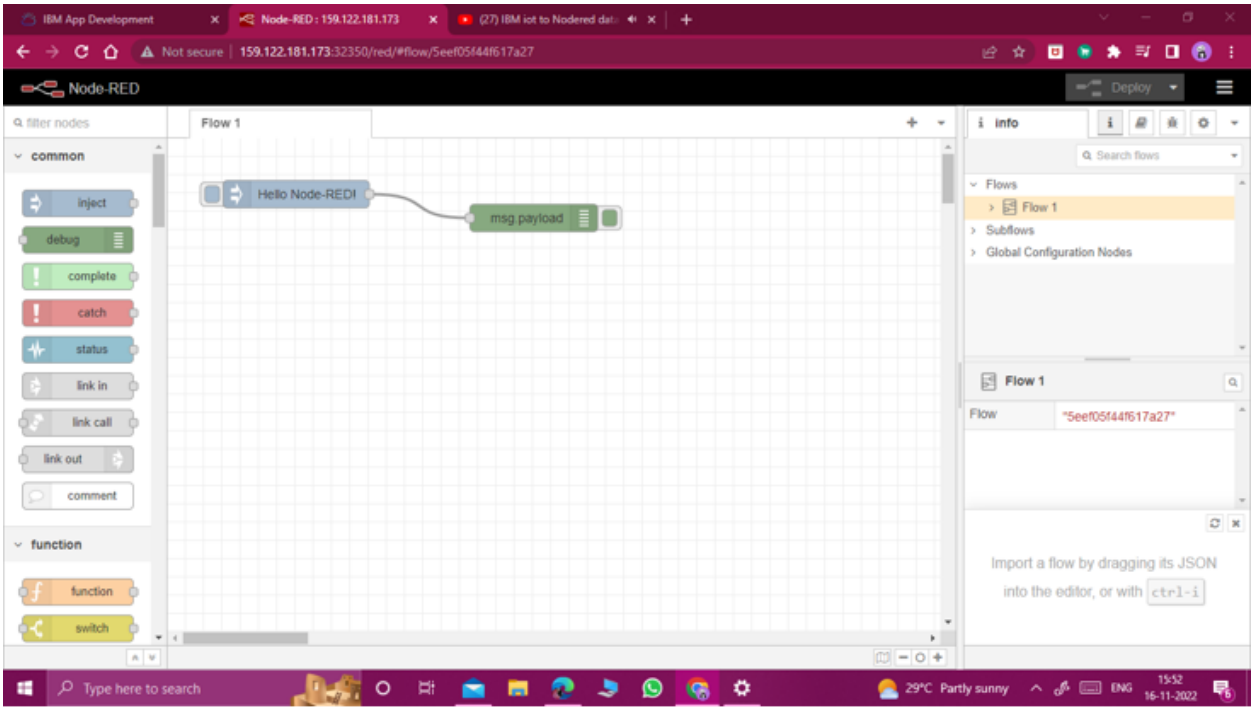


7 .CODING & SOLUTIONING (Explain the features added in the project along with code)

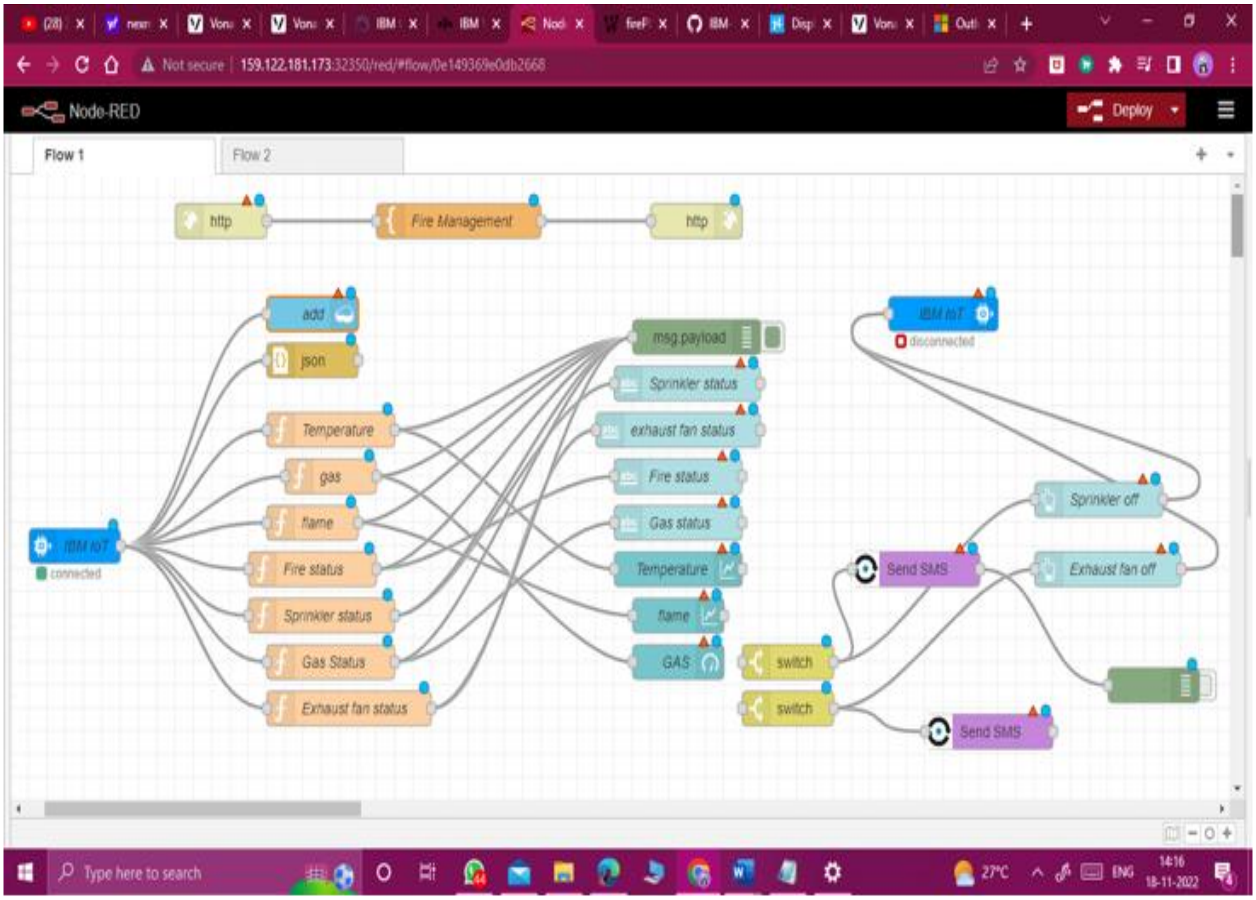
7.1 Feature 1



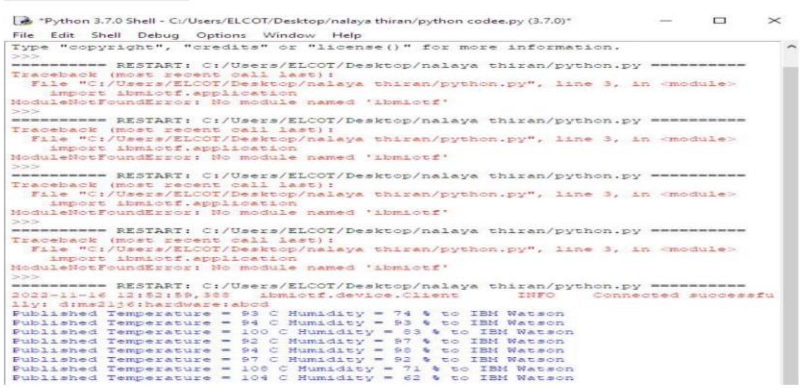

eu-gb.git.cloud.ibm.com/priyaaa2911/NodeREDKZIOV2022-11-15
Search GitLab
NodeREDKZIOV2022-11-15
Project information
Repository
Issues
Merge requests
Analytics
Wiki
Snippets
Settings
Collapse sidebar
NodeREDKZIOV2022-11-15
Project ID: 246022
Star 0
Fork 0
1 Commit
1 Branch
0 Tags
2.2 MB Project Storage
Created for toolchain: https://cloud.ibm.com/devops/toolchains/73c24768-3367-46f1-9302-37be9e45447e?env_id=ibm:yp:eu-gb
master
NodeREDKZIOV2022-11-15 /
Find file
Clone
Clone from ZIP
IBM Cloud Continuous Delivery authored 1 day ago
7287518c
README
Apache License 2.0
CONTRIBUTING
Add CHANGELOG
Configure Integrations
Name
Last commit
Last update
.github
Clone from ZIP
1 day ago
chart/noderedkziov20221115
Clone from ZIP
1 day ago
defaults
Clone from ZIP
1 day ago
Type here to search
29°C Partly sunny
ENG
20:44
17-11-2022

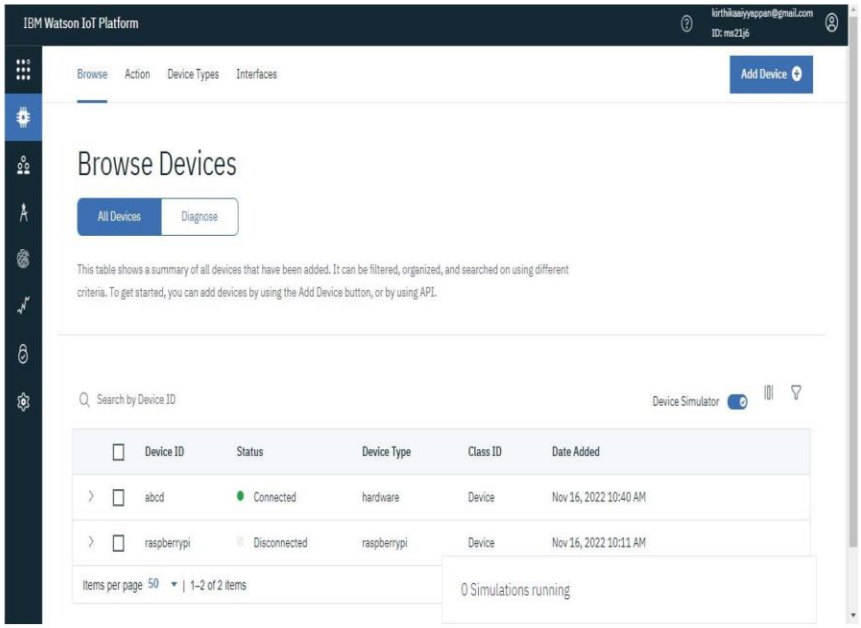

Not secure
159.122.181.173:32350/red/#flow/5eef05f44f617a27
Node-RED
Deploy
filter nodes
Flow 1
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
Hello Node-RED!
msg payload
info
Search flows
Flows
Flow 1
Subflows
Global Configuration Nodes
Flow 1
Flow
"5eef05f44f617a27"
Import a flow by dragging its JSON
into the editor, or with ctrl-i
Type here to search
29°C Partly sunny
ENG
15:52
16-11-2022

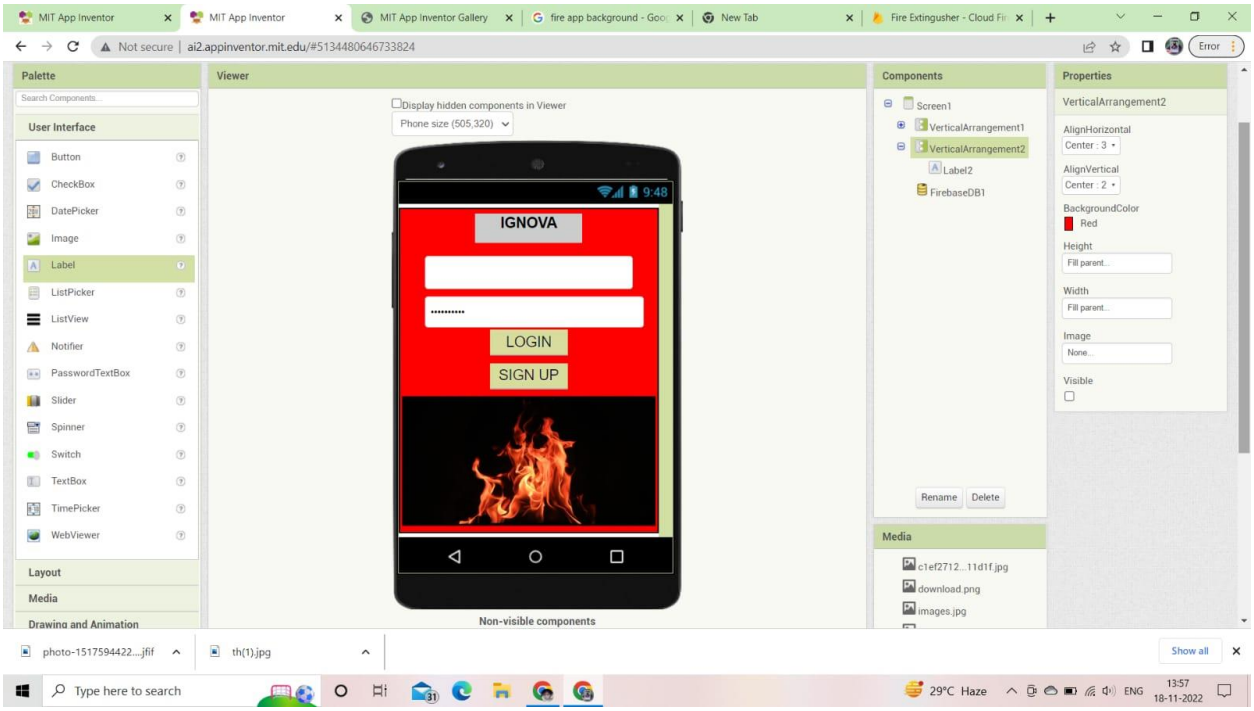
7.3 Feature 2

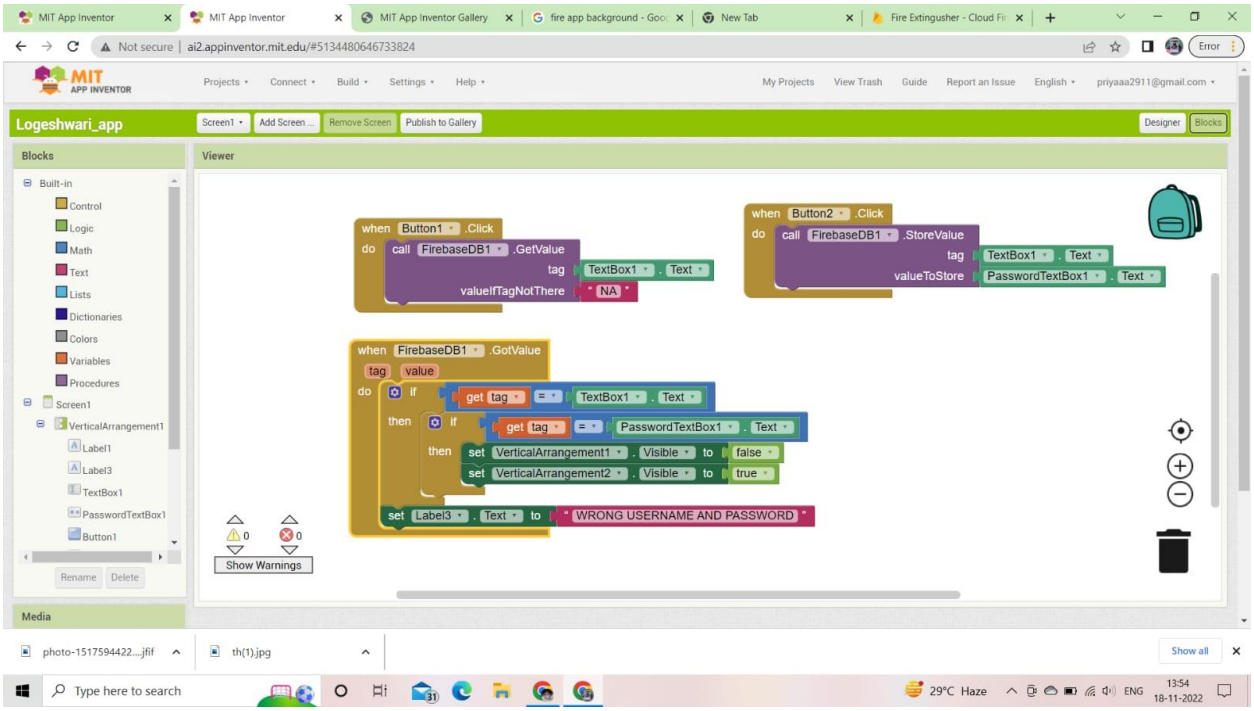


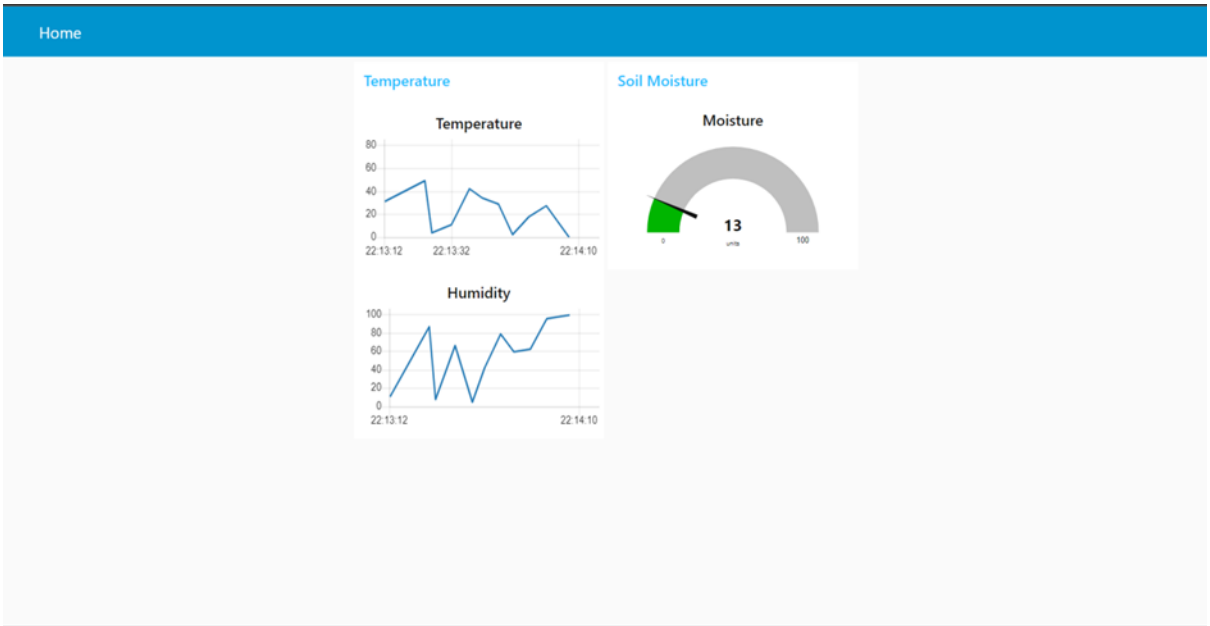
OUTPUT:




IBM Watson IoT Platform
kirthikasiyappan@gmail.com
ID: ms21j6
Browse
Action
Device Types
Interfaces
Add Device
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
abcd
Connected
hardware
Device
Nov 16, 2022 10:40 AM
raspberrypi
Disconnected
raspberrypi
Device
Nov 16, 2022 10:11 AM
Items per page 50
1–2 of 2 items
0 Simulations running


MIT App Inventor
Not secure | ai2.appinventor.mit.edu/#5134480646733824
Palette
Viewer
Components
Properties
User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
Display hidden components in Viewer
Phone size (505,320)
IGNOVA
LOGIN
SIGN UP
Non-visible components
Screen1
VerticalArrangement1
VerticalArrangement2
Label2
FirebaseDB1
Rename
Delete
VerticalArrangement2
AlignHorizontal
Center : 3
AlignVertical
Center : 2
BackgroundColor
Red
Height
Fill parent...
Width
Fill parent...
Image
None...
Visible
Media
c1ef2712...11d1f.jpg
download.png
images.jpg
photo-1517594422....jfif
th(1).jpg
Show all
Type here to search
29°C Haze
ENG
13:57
18-11-2022


MIT App Inventor
Logeshwari_app
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Viewer
when Button1 .Click
do call FirebaseDB1 .GetValue
tag TextBox1 . Text
valueIfTagNotThere " NA "
when Button2 .Click
do call FirebaseDB1 .StoreValue
tag TextBox1 . Text
valueToStore PasswordTextBox1 . Text
when FirebaseDB1 .GotValue
tag value
do if get tag = TextBox1 . Text
then if get tag = PasswordTextBox1 . Text
then set VerticalArrangement1 . Visible to false
set VerticalArrangement2 . Visible to true
set Label3 . Text to " WRONG USERNAME AND PASSWORD "
Show Warnings
Media


Home
Temperature
Temperature
80
60
40
20
0
22:13:12
22:13:32
22:14:10
Humidity
100
80
60
40
20
0
22:13:12
22:14:10
Soil Moisture
Moisture
13
0
100

9.RESULTS

9.1 Performance Metrics

1. Hours worked : 48 hours
2. Efficiency of the product:100%
3. Quality of the product:100%

10.ADVANTAGES

- It reduces the false warnings.
- The installation cost is low.
- This system monitors the surrounding 24/7.
- It improves security in industries and Offices.

11.DISADVANTAGES

1. This system cannot be implement in large scale industries.
2. The Control pannel need to be replaced, if it gets damaged.

12.CONCLUSION

➤ This system helps in reducing false warning.
➤ This system intimates the authorities at right time about the suitiation.
➤ As the system is cost effective it can be easily implemented in small scale industeries.

12. FUTURE SCOPE

- By using other upcoming technologies such AI can be used to make this system automated and efficient.

- Other drone models are used to provide aerial vision, among other things, to those directing the firefighting process. Providing unique insight to those who would typically require expensive helicopters to do the same work. Better yet, more advanced, and expensive, drones are being developed to fly up to 900 feet to spray water that would be typically unreachable by truck-mounted ladders.

13.APPENDIX

```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

#define ORG "ssfrkg"
#define DEVICE_TYPE "Logeshwari"
#define DEVICE_ID "311019106034"
#define TOKEN "6383715196"

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);
    int flamereading = random(200,1024);
    flame = map(flamereading,200,1024,0,2);

    //set a flame status

    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+"\",";
  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();
  }
}
```

CODE FOR NODE RED:

```
[
{
"id": "d5d1ac02e16a847e",
"type": "tab",
"label": "Flow 1",
"disabled": false,
"info": "",
"env": []
},
{
"id": "1533d345e1dc36a9",
"type": "ui_gauge",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "b0b2466424d96a85",
"order": 1,
"width": 0,
"height": 0,
"gtype": "gage",
"title": "Temperature",
"label": "Celsius",
"format": "{{value}}",
"min": 0,
"max": "100",
"colors": [
```

```
"#00b500",
"#e6e600",
"#ca3838"
],
"seg1": "40",
"seg2": "60",
"className": "",
"x": 910,
"y": 340,
"wires": []
},
{
"id": "cbd881198ad25821",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "Convert",
"func": "msg.payload=msg.payload.Temperature\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 460,
"y": 340,
"wires": [
[
"1533d345e1dc36a9",
"1cba1c03ecf32aa5",
"c883c624a3b90fc4",
"af83517ed28f54e9"
```

```
]
]
},
{
"id": "ea14ed512305484e",
"type": "ibmiot in",
"z": "d5d1ac02e16a847e",
"authentication": "apiKey",
"apiKey": "8d9dadc21a8b3d2f",
"inputType": "evt",
"logicalInterface": "",
"ruleId": "",
"deviceId": "T1",
"applicationId": "",
"deviceType": "Temp",
"eventType": "+",
"commandType": "",
"format": "json",
"name": "IBM IoT",
"service": "registered",
"allDevices": "",
"allApplications": "",
"allDeviceTypes": "",
"allLogicalInterfaces": "",
"allEvents": true,
"allCommands": "",
"allFormats": "",
"qos": 0,
"x": 190,
"y": 340,
```

```
"wires": [
[
"cbd881198ad25821"
]
]
},
{
"id": "08daf8c27126b608",
"type": "ibmiot in",
"z": "d5d1ac02e16a847e",
"authentication": "apiKey",
"apiKey": "8d9dadc21a8b3d2f",
"inputType": "evt",
"logicalInterface": "",
"ruleId": "",
"deviceId": "G1",
"applicationId": "",
"deviceType": "Gas_Sensor",
"eventType": "+",
"commandType": "",
"format": "json",
"name": "IBM IoT",
"service": "registered",
"allDevices": "",
"allApplications": "",
"allDeviceTypes": false,
"allLogicalInterfaces": "",
"allEvents": true,
"allCommands": "",
"allFormats": "",
```

```
"qos": 0,
"x": 190,
"y": 440,
"wires": [
[
"cce26a0bf1e99e72",
"d4be4869ed5d4f8c",
"9669c4a7a1ddc6de"
]
]
},
{
"id": "cce26a0bf1e99e72",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "",
"func": "msg.payload=msg.payload.NO2\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 480,
"y": 460,
"wires": [
[
"152e5e18a22ea640",
"c0b625de4d3f7730"
]
]
```

```
},
{
"id": "d4be4869ed5d4f8c",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "",
"func": "msg.payload=msg.payload.CO\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 400,
"y": 540,
"wires": [
[
"574dedc96208c0aa",
"f3dd2fc7f39bf5f7"
]
]
},
{
"id": "9669c4a7a1ddc6de",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "",
"func": "msg.payload=msg.payload.CO2\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
```

"finalize": "",
"libs": [],
"x": 340,
"y": 660,
"wires": [
[
"8dae8fa1a7ea7415",
"763a9ae51ae799ec"
]
]
},
{
"id": "152e5e18a22ea640",
"type": "ui_gauge",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "d2d6e984eb71a537",
"order": 1,
"width": 0,
"height": 0,
"gtype": "gage",
"title": "Nitrogen di Oxide",
"label": "Percentage",
"format": "{{value}}",
"min": 0,
"max": "100",
"colors": [
"#00b500",
"#e6e600",
"#ca3838"

```
],
"seg1": "20",
"seg2": "25",
"className": "",
"x": 770,
"y": 440,
"wires": []
},
{
"id": "574dedc96208c0aa",
"type": "ui_gauge",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "d2d6e984eb71a537",
"order": 2,
"width": 0,
"height": 0,
"gtype": "gage",
"title": "Carbon Mono Oxide",
"label": "percentage",
"format": "{{value}}",
"min": 0,
"max": "100",
"colors": [
"#00b500",
"#e6e600",
"#ca3838"
],
"seg1": "3",
"seg2": "5",
```

"className": "",
"x": 760,
"y": 540,
"wires": []
},
{
"id": "8dae8fa1a7ea7415",
"type": "ui_gauge",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "d2d6e984eb71a537",
"order": 3,
"width": 0,
"height": 0,
"gtype": "gage",
"title": "Carbon di Oxide",
"label": "Percentage",
"format": "{{value}}",
"min": 0,
"max": "100",
"colors": [
"#00b500",
"#e6e600",
"#ca3838"
],
"seg1": "20",
"seg2": "30",
"className": "",
"x": 700,
"y": 700,

"wires": []
},
{
"id": "8a41db8811f572d9",
"type": "ui_form",
"z": "d5d1ac02e16a847e",
"name": "",
"label": "",
"group": "c8492bdcd68680a2",
"order": 0,
"width": 0,
"height": 0,
"options": [
{
"label": "Username",
"value": "Username",
"type": "text",
"required": true,
"rows": null
},
{
"label": "Email",
"value": "Email",
"type": "email",
"required": true,
"rows": null
},
{
"label": "Phone Number",
"value": "Phonenumber",

```
"type": "number",
"required": true,
"rows": null
}
],
"formValue": {
"Username": "",
"Email": "",
"Phonenumber": ""
},
"payload": "",
"submit": "submit",
"cancel": "cancel",
"topic": "topic",
"topicType": "msg",
"splitLayout": "",
"className": "",
"x": 210,
"y": 760,
"wires": [
[
"40123f46207d3c68"
]
]
},
{
"id": "40123f46207d3c68",
"type": "cloudant out",
"z": "d5d1ac02e16a847e",
"name": "",
```

"cloudant": "3d10849a8821b6a7",
"database": "usercredentials",
"service": "ext",
"payonly": true,
"operation": "insert",
"x": 720,
"y": 760,
"wires": []
},
{
"id": "cec01e40a908c48e",
"type": "ui_led",
"z": "d5d1ac02e16a847e",
"order": 0,
"group": "b0b2466424d96a85",
"width": 0,
"height": 0,
"label": "Water Sprinkler",
"labelPlacement": "left",
"labelAlignment": "left",
"colorForValue": [
{
"color": "#ff0000",
"value": "false",
"valueType": "bool"
},
{
"color": "#008000",
"value": "true",
"valueType": "bool"

```
}
],
"allowColorForValueInMessage": false,
"shape": "circle",
"showGlow": true,
"name": "",
"x": 850,
"y": 120,
"wires": []
},
{
"id": "1cba1c03ecf32aa5",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "",
"func": "msg.payload=(msg.payload>45)? true: false;\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 240,
"y": 120,
"wires": [
[
"80a720021c12cce7"
]
]
},
```

```
{
"id": "80a720021c12cce7",
"type": "delay",
"z": "d5d1ac02e16a847e",
"name": "",
"pauseType": "delay",
"timeout": "1",
"timeoutUnits": "seconds",
"rate": "1",
"nbRateUnits": "1",
"rateUnits": "second",
"randomFirst": "1",
"randomLast": "5",
"randomUnits": "seconds",
"drop": false,
"allowrate": false,
"outputs": 1,
"x": 620,
"y": 120,
"wires": [
[
"cec01e40a908c48e"
]
]
},
{
"id": "cebd50b2ed69ccd3",
"type": "delay",
"z": "d5d1ac02e16a847e",
"name": "",
```

"pauseType": "delay",
"timeout": "1",
"timeoutUnits": "seconds",
"rate": "1",
"nbRateUnits": "1",
"rateUnits": "second",
"randomFirst": "1",
"randomLast": "5",
"randomUnits": "seconds",
"drop": false,
"allowrate": false,
"outputs": 1,
"x": 620,
"y": 180,
"wires": [
[
"1b0bc39e91d2d4dc"
]
]
},
{
"id": "56b6fd723b216743",
"type": "delay",
"z": "d5d1ac02e16a847e",
"name": "",
"pauseType": "delay",
"timeout": "1",
"timeoutUnits": "seconds",
"rate": "1",
"nbRateUnits": "1",

"rateUnits": "second",
"randomFirst": "1",
"randomLast": "5",
"randomUnits": "seconds",
"drop": false,
"allowrate": false,
"outputs": 1,
"x": 620,
"y": 240,
"wires": [
[
"dc9bb5498dd98dfc"
]
]
},
{
"id": "1b0bc39e91d2d4dc",
"type": "ui_led",
"z": "d5d1ac02e16a847e",
"order": 1,
"group": "b0b2466424d96a85",
"width": 0,
"height": 0,
"label": "Exhaust fan",
"labelPlacement": "left",
"labelAlignment": "left",
"colorForValue": [
{
"color": "#ff0000",
"value": "false",

"valueType": "bool"
},
{
"color": "#008000",
"value": "true",
"valueType": "bool"
}
],
"allowColorForValueInMessage": false,
"shape": "circle",
"showGlow": true,
"name": "",
"x": 850,
"y": 180,
"wires": []
},
{
"id": "dc9bb5498dd98dfc",
"type": "ui_led",
"z": "d5d1ac02e16a847e",
"order": 2,
"group": "b0b2466424d96a85",
"width": 0,
"height": 0,
"label": "Fire Alarm",
"labelPlacement": "left",
"labelAlignment": "left",
"colorForValue": [
{
"color": "#ff0000",

"value": "false",
"valueType": "bool"
},
{
"color": "#008000",
"value": "true",
"valueType": "bool"
}
],
"allowColorForValueInMessage": false,
"shape": "circle",
"showGlow": true,
"name": "",
"x": 850,
"y": 240,
"wires": []
},
{
"id": "c883c624a3b90fc4",
"type": "ui_chart",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "b0b2466424d96a85",
"order": 0,
"width": 0,
"height": 0,
"label": "Analytics",
"chartType": "line",
"legend": "false",
"xformat": "auto",

"interpolate": "linear",
"nodata": "Waiting for data",
"dot": false,
"ymin": "0",
"ymax": "100",
"removeOlder": "60",
"removeOlderPoints": "",
"removeOlderUnit": "1",
"cutout": 0,
"useOneColor": false,
"useUTC": false,
"colors": [
"#42b0ff",
"#aec7e8",
"#ff7f0e",
"#1adb1a",
"#98df8a",
"#d62728",
"#ff9896",
"#9467bd",
"#c5b0d5"
],
"outputs": 1,
"useDifferentColor": false,
"className": "",
"x": 940,
"y": 380,
"wires": [
[]
]

```
},
{
"id": "763a9ae51ae799ec",
"type": "ui_chart",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "5a710181faf582af",
"order": 5,
"width": 0,
"height": 0,
"label": "CO2 Analysis",
"chartType": "line",
"legend": "false",
"xformat": "HH:mm:ss",
"interpolate": "linear",
"nodata": "",
"dot": false,
"ymin": "0",
"ymax": "100",
"removeOlder": "60",
"removeOlderPoints": "",
"removeOlderUnit": "1",
"cutout": 0,
"useOneColor": false,
"useUTC": false,
"colors": [
"#1f77b4",
"#aec7e8",
"#ff7f0e",
"#2ca02c",
```

"#98df8a",
"#df2a2a",
"#ff9896",
"#9467bd",
"#c5b0d5"
],
"outputs": 1,
"useDifferentColor": false,
"className": "",
"x": 700,
"y": 640,
"wires": [
[]
]
},
{
"id": "c0b625de4d3f7730",
"type": "ui_chart",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "5a710181faf582af",
"order": 3,
"width": 0,
"height": 0,
"label": "NO2 Analysis",
"chartType": "line",
"legend": "false",
"xformat": "HH:mm:ss",
"interpolate": "linear",
"nodata": "",

"dot": false,
"ymin": "0",
"ymax": "100",
"removeOlder": "60",
"removeOlderPoints": "",
"removeOlderUnit": "1",
"cutout": 0,
"useOneColor": false,
"useUTC": false,
"colors": [
"#1f77b4",
"#aec7e8",
"#ff7f0e",
"#2ca02c",
"#98df8a",
"#d62728",
"#ff9896",
"#9467bd",
"#c5b0d5"
],
"outputs": 1,
"useDifferentColor": false,
"className": "",
"x": 740,
"y": 500,
"wires": [
[]
]
},
{

"id": "f3dd2fc7f39bf5f7",
"type": "ui_chart",
"z": "d5d1ac02e16a847e",
"name": "",
"group": "5a710181faf582af",
"order": 4,
"width": 0,
"height": 0,
"label": "CO Analysis",
"chartType": "line",
"legend": "false",
"xformat": "HH:mm:ss",
"interpolate": "linear",
"nodata": "",
"dot": false,
"ymin": "0",
"ymax": "100",
"removeOlder": "60",
"removeOlderPoints": "",
"removeOlderUnit": "1",
"cutout": 0,
"useOneColor": false,
"useUTC": false,
"colors": [
"#1f77b4",
"#aec7e8",
"#ff7f0e",
"#2ca52c",
"#98df8a",
"#d62728",

"#ff9896",
"#9467bd",
"#c5b0d5"
],
"outputs": 1,
"useDifferentColor": false,
"className": "",
"x": 750,
"y": 580,
"wires": [
[]
]
},
{
"id": "af83517ed28f54e9",
"type": "function",
"z": "d5d1ac02e16a847e",
"name": "",
"func": "msg.payload=(msg.payload>70)? true: false;\nreturn msg;",
"outputs": 1,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 360,
"y": 180,
"wires": [
[
"56b6fd723b216743",

```
"cebd50b2ed69ccd3"
]
]
},
{
"id": "5cd6db4b75445da3",
"type": "debug",
"z": "d5d1ac02e16a847e",
"name": "",
"active": true,
"tosidebar": true,
"console": false,
"tostatus": false,
"complete": "payload",
"targetType": "msg",
"statusVal": "",
"statusType": "auto",
"x": 690,
"y": 920,
"wires": []
},
{
"id": "ab2c14742c0e6cb4",
"type": "cloudant in",
"z": "d5d1ac02e16a847e",
"name": "",
"cloudant": "3d10849a8821b6a7",
"database": "usercredentials",
"service": "ext",
"search": "all",
```

"design": "Usercredentials",
"index": "Phonenumber",
"x": 460,
"y": 860,
"wires": [
[
"5cd6db4b75445da3"
]
]
},
{
"id": "7c787911c08c4fcb",
"type": "inject",
"z": "d5d1ac02e16a847e",
"name": "",
"props": [
{
"p": "payload"
},
{
"p": "topic",
"vt": "str"
}
],
"repeat": "",
"crontab": "",
"once": false,
"onceDelay": 0.1,
"topic": "",
"payload": "",

"payloadType": "date",
"x": 160,
"y": 860,
"wires": [
[
"ab2c14742c0e6cb4"
]
]
},
{
"id": "1d8c19444aee18bf",
"type": "debug",
"z": "d5d1ac02e16a847e",
"name": "",
"active": true,
"tosidebar": true,
"console": false,
"tostatus": false,
"complete": "false",
"statusVal": "",
"statusType": "auto",
"x": 850,
"y": 20,
"wires": []
},
{
"id": "3c36398fb917a7c9",
"type": "python-function",
"z": "d5d1ac02e16a847e",
"name": "",

"func": "import requests\nurl = \"https://www.fast2sms.com/dev/bulkV2\"\npayload = \"sender_id=FSTSMS&message=Alert!&language=english&route=p&numbers=9095057479\"\nheaders = {\n'authorization': \"o0azwVFNHOM5B3hrRxdenyU2cfZujqSpYEX7t8LAgJPb9kliWCugDvo1n0kcY8TGHOt3dIQwsKpLbAJU\",\n'Content-Type': \"application/x-www-form-urlencoded\",\n'Cache-Control': \"no-cache\",\n}\nresponse = requests.request(\"POST\", url, data=payload, headers=headers)\nprint(response.text)\n",

"outputs": 1,

"x": 580,

"y": 20,

"wires": [

[

"1d8c19444aee18bf"

]

]

},

{

"id": "057ef750bef3fc86",

"type": "inject",

"z": "d5d1ac02e16a847e",

"name": "",

"props": [

{

"p": "payload"

},

```
{
"p": "topic",
"vt": "str"
}
],
"repeat": "",
"crontab": "",
"once": false,
"onceDelay": 0.1,
"topic": "",
"payload": "",
"payloadType": "date",
"x": 180,
"y": 20,
"wires": [
[
"3c36398fb917a7c9"
]
]
},
{
"id": "b0b2466424d96a85",
"type": "ui_group",
"name": "Temperature",
"tab": "477494a5baf03dbe",
"order": 3,
"disp": true,
"width": "6",
"collapse": false,
"className": ""
```

```
},
{
"id": "8d9dadc21a8b3d2f",
"type": "ibmiot",
"name": "IBM_IOT",
"keepalive": "60",
"serverName": "lryrya.messaging.internetofthings.ibmcloud.com",
"cleansession": true,
"appId": "",
"shared": false
},
{
"id": "d2d6e984eb71a537",
"type": "ui_group",
"name": "Gas Sensor",
"tab": "e208207b5344b7ac",
"order": 2,
"disp": true,
"width": "6",
"collapse": false,
"className": ""
},
{
"id": "c8492bdcd68680a2",
"type": "ui_group",
"name": "Register Form",
"tab": "94803c79c48f4a14",
"order": 3,
"disp": true,
"width": "6",
```

"collapse": false,
"className": ""
},
{
"id": "3d10849a8821b6a7",
"type": "cloudant",
"host":
"https://apikey-v2-
2uxme1xe4ze15rnf5fh8l40eg4coz66pm8ateuwy93py:0172158934b6c
3a3e8b
39b0511e7af20@e8e96f53-33f3-4fd1-bb73-6fb98205f7c2-
bluemix.cloudantnosqldb.appdomain.
cloud",
"name": ""
},
{
"id": "5a710181faf582af",
"type": "ui_group",
"name": "Gas Analytics",
"tab": "e208207b5344b7ac",
"order": 3,
"disp": true,
"width": "6",
"collapse": false,
"className": ""
},
{
"id": "477494a5baf03dbe",
"type": "ui_tab",
"name": "Temperature",

"icon": "dashboard",
"disabled": false,
"hidden": false
},
{
"id": "e208207b5344b7ac",
"type": "ui_tab",
"name": "Gas Sensor",
"icon": "dashboard",
"disabled": false,
"hidden": false
},
{
"id": "94803c79c48f4a14",
"type": "ui_tab",
"name": "Register Form",
"icon": "dashboard",
"disabled": false,
"hidden": false
}
]

LINKS:

GIT REPO LINK:

https://github.com/IBM-EPBL/IBM-Project-22332-1659849421

video link:

:

https://clipchamp.com/watch/mH5XDaHrdIo