PROJECT DOCUMENTATION REPORT

Date	24-11-22
Team ID	PNT2022TMID00080
Project Name	Project - Hazardous Area Monitoring for Industrial Plant powered by IoT

INDEX

1. **INTRODUCTION** **4**
    1.1 Project Overview
    1.2 Purpose

2. **LITERATURE SURVEY** **5**
    2.1 Existing problem
    2.2 References
    2.3 Problem Statement Definition

3. **IDEATION & PROPOSED SOLUTION** **9**
    3.1 Empathy Map Canvas
    3.2 Ideation & Brainstorming
    3.3 Proposed Solution
    3.4 Problem Solution fit

4. **REQUIREMENT ANALYSIS** **14**
    4.1 Functional requirement
    4.2 Non-Functional requirements

5. **PROJECT DESIGN** **17**
    5.1 Data Flow Diagrams
    5.2 User Stories
    5.3 Solution & Technical Architecture

6. **PROJECT PLANNING & SCHEDULING** **19**
    6.1 Sprint Planning & Estimation
    6.2 Sprint Delivery Schedule
    6.3 Reports from JIRA

7. **CODING & SOLUTIONING** **20**
    7.1 Feature 1
    7.2 Feature 2

8. **TESTING** **26**
    8.1 Test Cases
    8.2 User Acceptance Testing

9. **RESULTS** **29**

9.1 Performance Metrics

10. **ADVANTAGES & DISADVANTAGES** **31**

11. **CONCLUSION** **32**

12. **FUTURE SCOPE** **32**

13. **APPENDIX** **33**

Source Code

GitHub & Project Demo Link.

1. INTRODUCTION:

1.1 Project Overview

- In Industrial plants there are some hazardous areas which contains potentially life-threatening levels of temperature parameters.

- This project is mainly for detecting, monitoring these temperature parameters and displaying it to the industrial workers and providing timely alert to the industrial workers and admins via various resources like smart wearable device, mobile application and web application.

- Smart wearable device is used for displaying all the temperature related parameters to the worker whenever he reaches near the beacon sensor, which is used for gathering the current temp. levels, air quality and humidity.

- Web application is used for displaying all the temperature related parameters constantly with the help of beacon sensor.

- Mobile application is used for displaying all the temperature related parameters and in sending alert (or) warning messages whenever harmful level temperature is detected.

- Thereby avoiding any risks (or) injuries and in saving the lives of the industrial workers working on that place.

1.2 Purpose

- The main purpose of the project is to save the lives of the workers working on hazardous areas in industries.

- More modern and cost-efficient approach of detecting and monitoring temperature parameters in industrial plants.

- An efficient way for administrators working on industrial plants in detecting temperature parameters and in monitoring them at all times.

- Providing more than one way of detecting and monitoring temperature parameters in industrial plants.

- Easier and methodical way of alerting the workers of toxic level gases present in that area.

2. LITERATURE SURVEY:

2.1 Existing problem

A strong system of safety data collection, analysis, and sharing will assist the industry to understand the root causes of an event, explore existing and potential hazards, and continuously improving existing safety programs. Different countries and industries have conducted multiple reporting systems to collect, analyze, and share information with the public.

Safety training through computer-aided technologies
The development of virtual reality (VR), augmented reality (AR), and mixed reality (MR) have embedded worker training systems and become significant cost-effective and safer ways to educate workers. The immersive VR/AR/MR environments within computer-generated simulations have also gained popularity in safety training to identify the potential hazards as well as educate moving vehicle operators on the job site. Hazardous construction scenarios can be simulated interactively with the working environment, workers' behavior, high-risk equipment, and working sequence

Integrating BIM and safety
Numerous studies and industrial applications evidenced that safety and BIM integration can assist in safety planning and execution of projects, for example to automatic checking of construction models and schedules for preventing fall-related accidents; automated scaffolding-related safety hazard identification,visualization & prevention ,blind spots identification and mapping, etc.

Proximity detection devices
Many proximity avoidance systems have been developed by utilizing various technologies, such as an ultrasonic-based sensor, radio-frequency identification (RFID) sensing technology, radar, GPS, and magnetic field generators, to prevent contact accidents, particularly for accidents due to being struck by equipment.

Wearable sensing devices
A wide range of wearable devices has been applied across different industrial sectors including health care, manufacturing, etc. Some of these devices have proven to be very useful and beneficial to these industries and efforts are being made by both researchers and industry practitioners to improve on these technologies and learn from their initial implementation. With the attention being gained by wearable devices worldwide, mobile devices are becoming part of everyday life and the number, types, and forms of wearable devices are increasing exponentially in recent years.

Problems:
1. Cost of various types of sensors used and number of devices connected to it.
2. Using devices in an In-efficient approch fro temperature detection & monitoring.
3. Utilization of VR, AR, MR systems require a large sum of money for the industry.
4. No easier (or) more modern approach for alerting the workers.
5. Multiple ways for displaying the temperature for the workers and admins to monitor.

2.2 References

S NO	TITLE	Authors	Abstract	Drawbacks
1	IoT-Based Data Logger for Weather Monitoring Using Arduino-Based Wireless Sensor Networks with Remote Graphical Application and Alerts	Jamal Mabrouki , Mourade Azrour, Driss Dhiba, Yousef Farhaoui, and Souad El Hajjaji	In recent years, monitoring systems play significant roles in our life. So, in this paper, we propose an automatic weather monitoring system that allows having dynamic and real-time climate data of a given area. The proposed system is based on the internet of things technology and embedded system. The system also includes electronic devices, sensors, and wireless technology. The main objective of this system is sensing the climate parameters, such as temperature, humidity, and existence of some gases, based on the sensors. The captured values can then be sent to remote applications or databases. Afterwards, the stored data can be visualized in graphics and tables form.	No information about where we can implement this, just the monitoring thing is explained and done.
2	Design and Validation of a Multifunctional Android-Based Smart Home Control and Monitoring System	LUN-DE LIAO (Member, IEEE), YUHLING WANG YUNG-CHUNG TSAO, I-JAN WANG, DE-FU JHANG, TSUNG-SHENG CHU, CHIA-HUI TSAO, CHIH-NING TSAI, SHENG-FU CHEN, CHIUNG-CHENG CHUANG,	Users often need to control and monitor the environmental variables of their homes, even when they are not at home. In this paper, we present a multifunctional, low-cost, and flexible system for smart home control and environmental monitoring. This system employs an embedded micro web server based on an Arduino Yún microcontroller with Internet connectivity that allows remote device control. The proposed system can be controlled via the Internet through an Android-based mobile app. To guarantee access regardless of Internet availability, the proposed system can also be controlled via standalone manual operation using a touch display. The proposed system transmits sensor data to a cloud platform and can receive commands from the server, allowing many devices to be automatically controlled. To demonstrate the feasibility and effectiveness of this system, devices	Bounded only to mobile application and there is no web application or SMS for fast notification as we may not have our Internet connections on always.

		AND TZONG-RONG GER	such as light switches, power plugs, and various sensors, including temperature, gas, 2.5-µm particulate matter (PM2.5) and motion sensors, were integrated into a prototype of the proposed home control system.	
3	Micraspis: A Computer-Aided Proposal Toward Programming and Architecting Smart IoT Wearables	LONG-PHUOC TÔN, LAM-SON LÊ, (Member, IEEE), AND MINH-SON NGUYEN	A wearable is a lightweight body-worn device that relies on data-driven communications to keep people connected purposefully, for instance, for fire-fighting, prompting fast-food clients, and medical treatment. With the rise of wearable computing in the era of IoT-driven smart applications, programmers now expect the time to market for these devices to be shortened. While support for IoT programming in general has gathered traction, tool proposals that automate the development of smart solutions based on the Internet of Wearable Things, though of paramount importance, still stay on the sidelines. We propose a code generation tool called Micraspis that allows a wearable to be described both functionally and architecturally – as if they are two sides of the same coin. The tool has an underlyingmodel-to-code transformation mechanism to generate source code that is executable on a specific IoT programming platform such as Arduino. Our experiments demonstrate that programming code generated by Micraspis amounts to at least 60% of the source code needed to fulfill the business logic of ordinary wearable devices. programmers' assessment on how Micraspis assists them in programming and architecting Iot.	Sole usage of Wearable device only. This can cause limitations as we may not be able to monitor through other means.

4	A Privacy-Preserving Iot-Based Fire Detector	ABDULLAH H. ALTOWAIJRI, MOHAMMED S. ALFAIFI, TARIQ A. ALSHAWI, (Member, IEEE), AHMED B. IBRAHIM, AND SALEH A. ALSHEBEILI	Fire detection has been an issue of interest to researchers due to its significant damage to lives and property within a very short time. One of the recent solutions developed to detect fire is to use Internet of Things (IoT) devices equipped with cameras for surveillance. The captured videos of surroundings may be processed by the IoT devices themselves or at the cloud. The latter case is required if the detection algorithm is computationally demanding. However, the use of clouds has a flaw. In fact, using the cloud could pose the threat of having the privacy of a place violated, either through hacking or unauthorized access to the footage of the place where the cloud is installed. In this paper, a fire detection system that preserves the privacy of surroundings, while maintaining a high level of accuracy for fire detection is proposed. The proposed system makes use of the cloud for fire detection; and that is achieved by sending to the cloud features extracted from the video captured by the IoT device. Binary video descriptors and Convolutional Neural Network (CNN) have been used to develop the fire detection algorithm. The video descriptors are used to extract features, while CNN is used for classification. Videos with real fire and non-fire scenes have been used in this development. Results show that the performance of proposed fire detection algorithm can achieve 97.5% classification accuracy.	No online web app or mobile applications where we can see the current situation of the monitored environment.

2.3 Problem Statement Definition

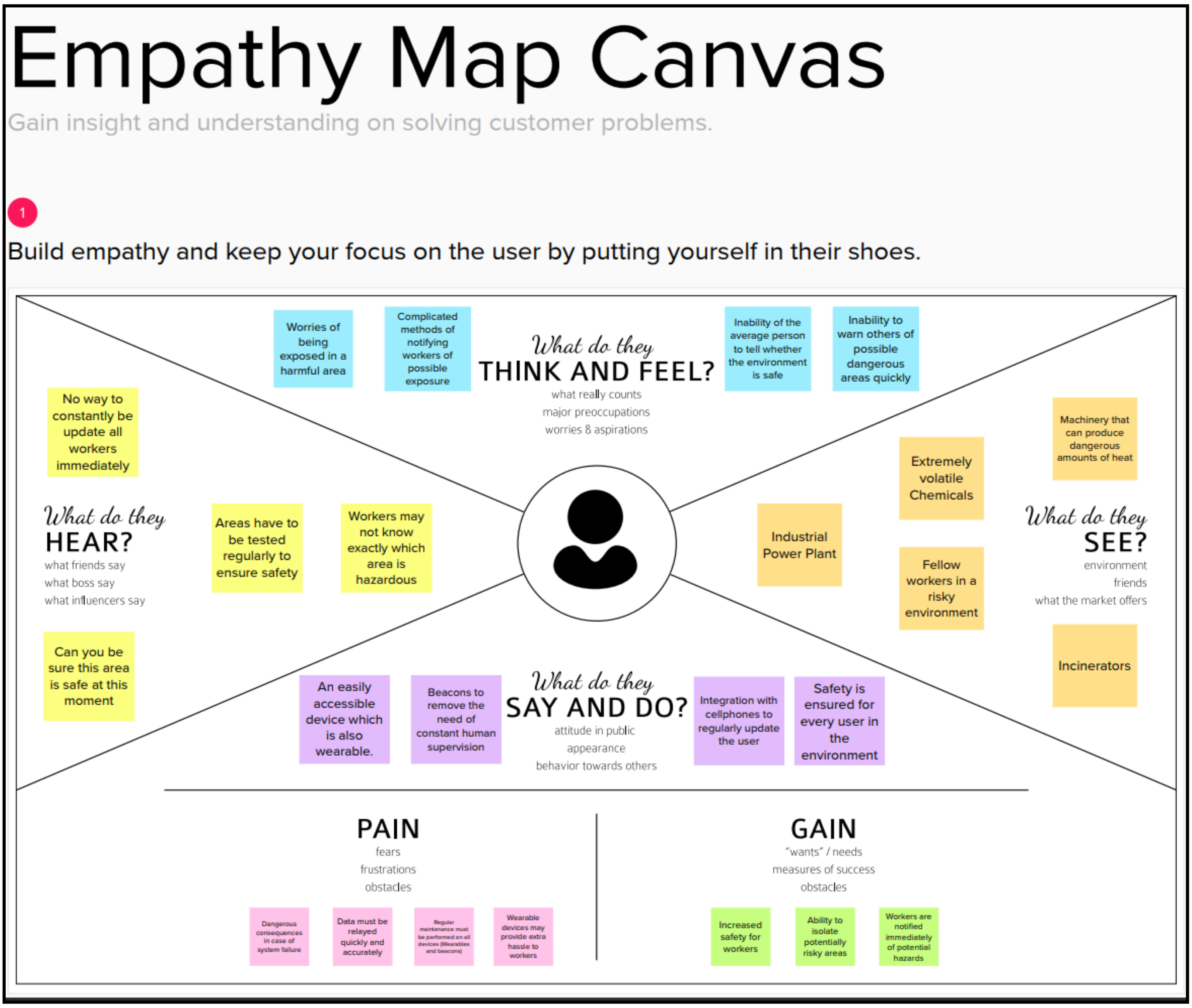
1. Rahul is a Plant Manager who needs to be informed of possible hazardous areas because it could pose a risk to the lives of the workers in the facility.

2. Christopher is a technician who needs to be informed when he is entering a hazardous area because it can lead to dangerous health complications.

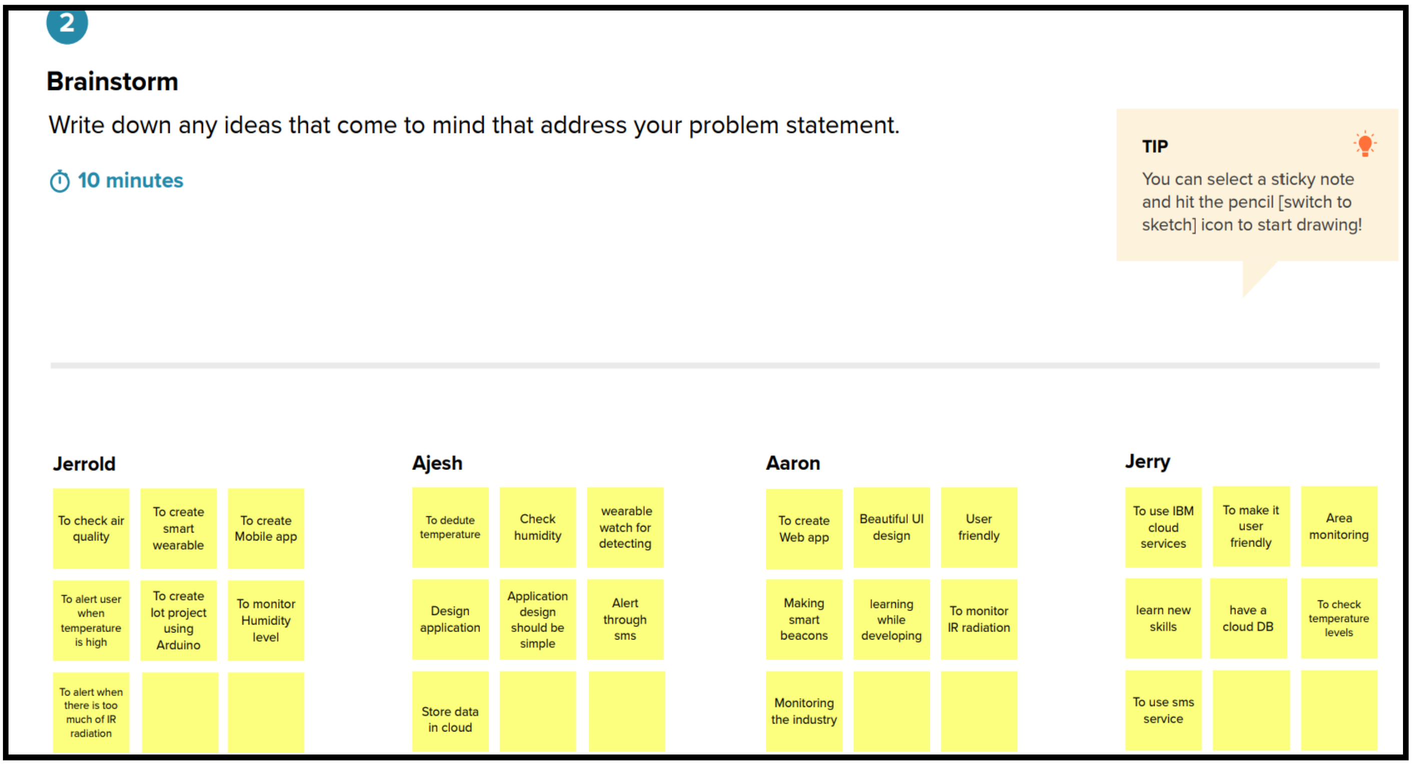
3. Markus is the Safety Inspector who needs to easily ascertain whether the various areas of the plant are hazardous or not because he needs to provide accurate reports to ensure safety.

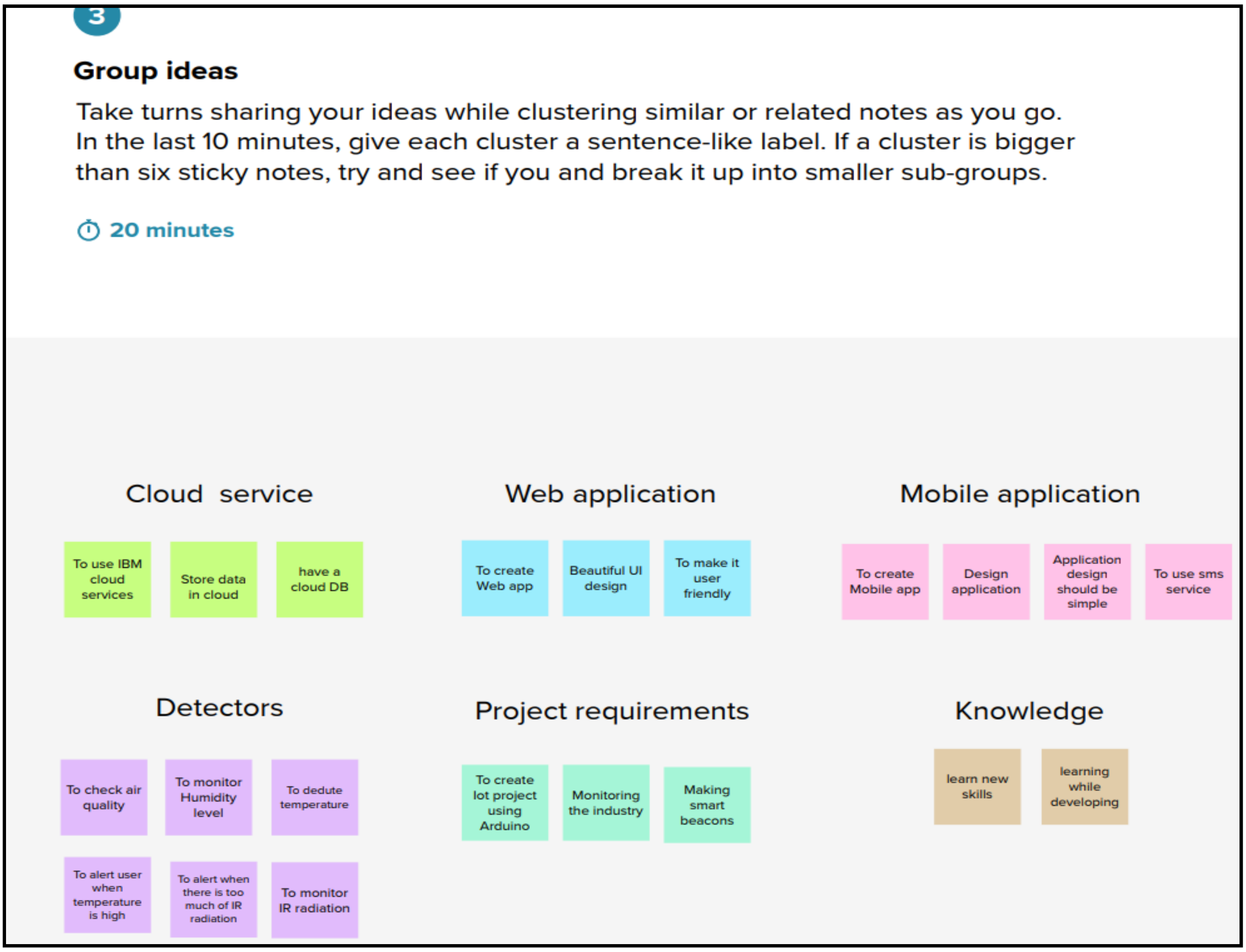
3. IDEATION & PROPOSED SOLUTION:

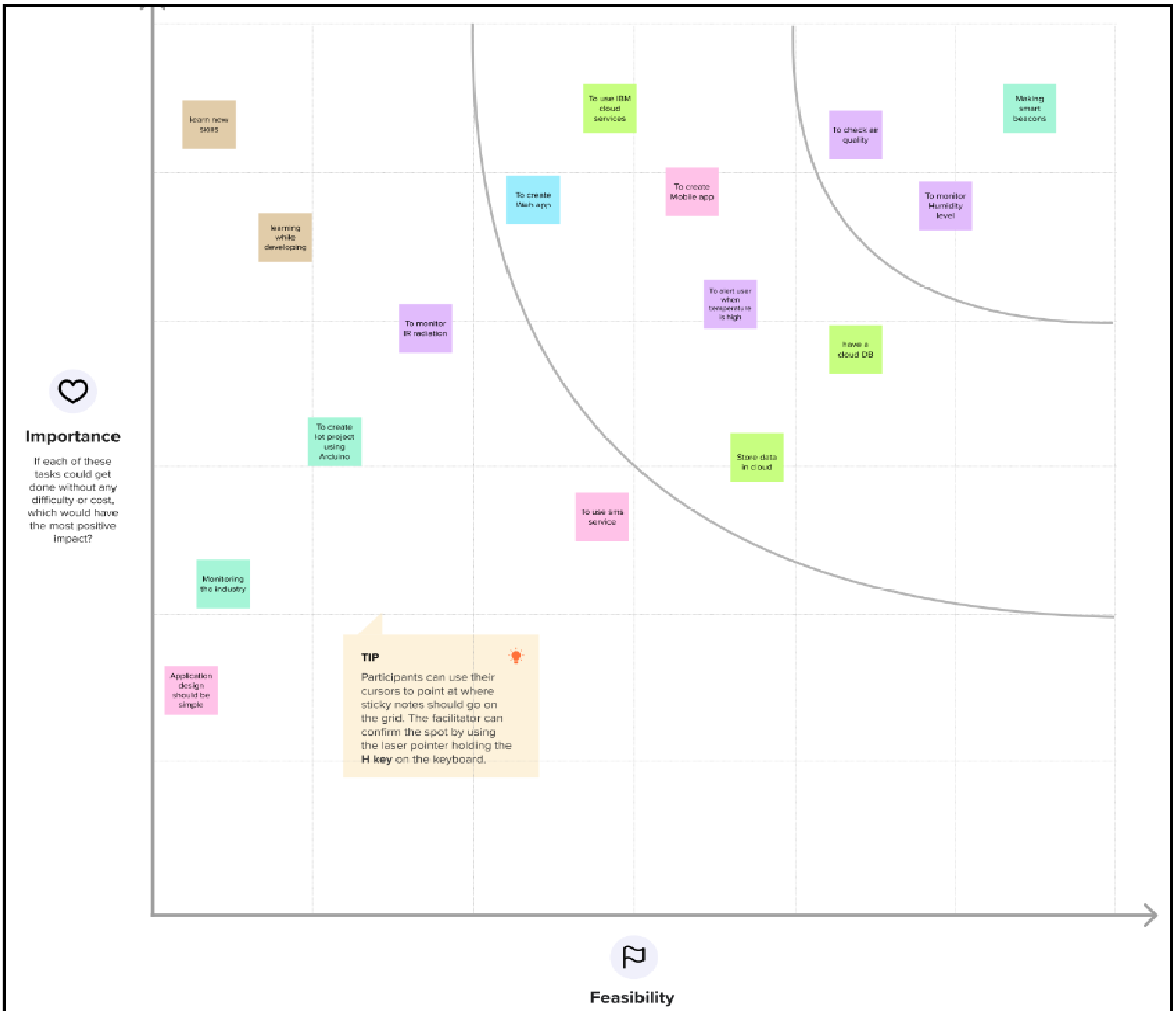
3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming







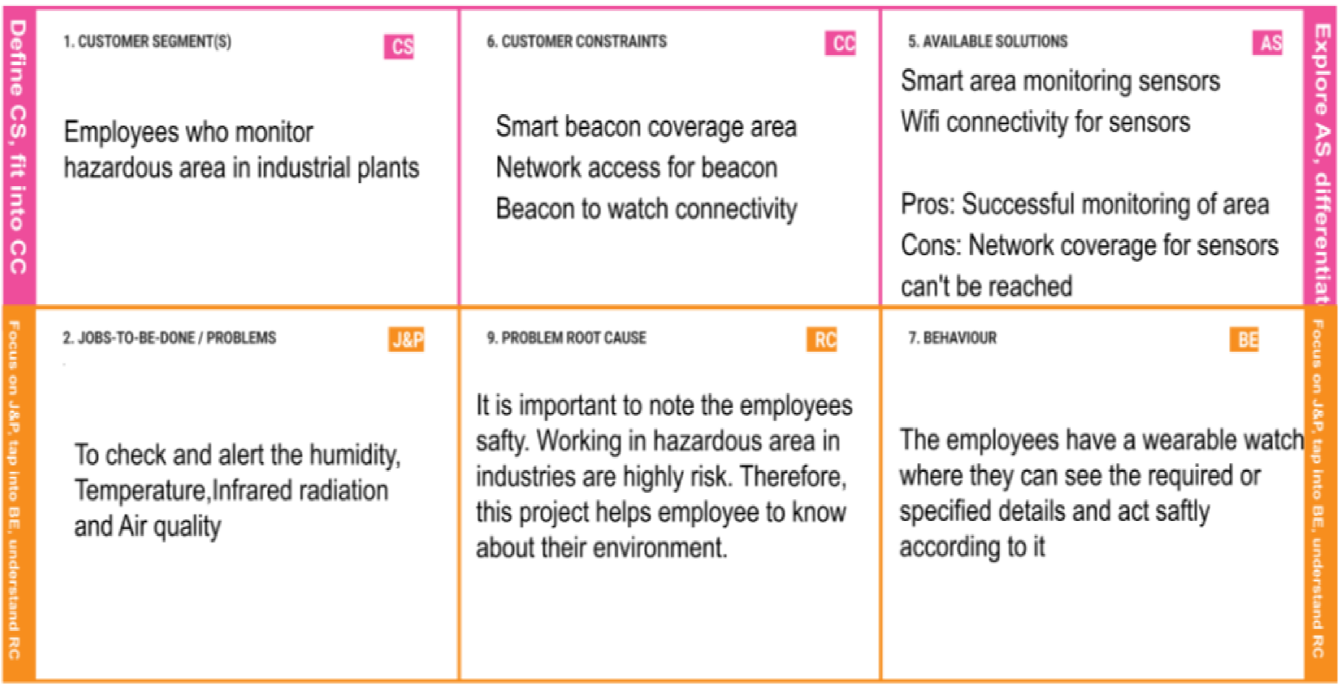
3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To monitor and alert the industrial workers the risk of toxic or hazardous gases present within the area of an industry, ensuring the safety of the workers.
2.	Idea / Solution description	Providing a wearable device which collects the data(temperature) via beacon sensors and displays it. An alert message is also sent to mobile whenever high temperature (or) toxic gases are detected within the area through SMS

		using API. Ensuring precautions and safety of the workers.
3.	Novelty / Uniqueness	● Makes it easier to know the temperature (or) any hazardous gases present in the area without the worker having to constantly doing manual checks. ● Provides different solution to ensure the safety of the workers. ● Wearable devices display the current temperature present in the area all the time. ● Alerts via SMS to mobiles of the workers when high temperature is detected. ● Alerts on both the wearable device and mobile application occurs simultaneously to prevent the worker from entering into hazardous areas.
4.	Social Impact / Customer Satisfaction	● Ensures safety. ● Saves lives of workers. ● Comfortable & User-friendly. ● Simple and reliable. ● Helps in taking necessary precautions to avoid the risk of endangering human lives. ● Necessary updates and more functions can be added to the mobile application to make it easier to use.
5.	Business Model (Revenue Model)	● Through our mobile application the revenue can be made in the form of pop-up advertisements, overlay ads from third party services. ● Wearable devices can be priced and sold by the industry to the workers.

6.	Scalability of the Solution	• Large no. of people can be supplied with the wearable devices to ensure their safety. • Beacon sensors cover large amount of area and supplies data accurately and more readily. • Multiple users can receive alert messages and notifications simultaneously regarding hazardous gases without any delay. • Each user has individual wearable device and mobile devices which provide information accordingly. • It ensures the safety of each and every worker working in harmful gases and high temperature environment.

3.4 Problem Solution Fit



Identify strong TR & EM	3. TRIGGERS TR	10. YOUR SOLUTION SL	8.CHANNELS of BEHAVIOUR CH	Identify strong TR & EM
	Successful execution of our solution will make even other industry to implement this solution	We are going to monitor the area using suitable sensors in the beacons. We will connect our wearable to the beacons. We will send updates to online cloud from the beacon. From the cloud we will be accessing the reading and using that we will have a web page and a mobile application to display them. We will have sms service to alert abnormal readings	**8.1 ONLINE** All the informations will be stored in cloud. so the employees can see the cloud storage or mobile application for referring the details of surroundings. **8.2 OFFLINE** Employees used to wear a watch which captures the information of the surroundings.	
	4. EMOTIONS: BEFORE / AFTER EM			
	It will be easy for employees to identify or to know about their environment			

4. REQUIREMENT ANALYSIS:

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Data Gathering	The smart beacon must be able to detect and the temperature of a particular area in real.
FR-2	Location Detection	The smart beacon must be able to detect whena wearable devicehas entered an areanearit.
FR-3	Beacon DataSyncing	The smart beacon must be able to share its stored data with both the wearable device and admindashboard through thecloud.
FR-4	Wearable DeviceDisplay	The wearable device must be able to display thetemperature of the area wherethe worker is currently present.

FR-5	SMS Notification	If the temperature of the area is found to reach dangerous levels, the worker should be informed via SMS to their phoneinstructing them to leavethe area.
FR-6	Admin Dashboard	If the temperature of the area is found to reach dangerous levels the adminis informed via the dashboard and must take the necessary precautions.

4.2 Non-Functional requirement

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	The wearable device should be slim and not annoy or disturb the workers who are wearing them. They should also reliably display the temperature without large delays and notifications should be clear in cases of detected danger.
NFR-2	**Security**	The connection of the beacons to the cloud and wearable devices should be secure. The security of the database housing all the temperature data should also be bolstered.

NFR-3	**Reliability**	The wearable device should be able to function without any faults even at dangerous temperatures. If a fault is detected it should notify the user and the admin to be immediately repaired and replaced. The beacons should also be regularly maintained to ensure reliability.
NFR-4	**Performance**	The device should update temperature readings in real time and requires high end sensors and processors to do so. The time to send data to the cloud and other devicesshould also be made as small as possible.

NFR-5	**Availability**	The user should be able to check the temperature of the area no matter where or at what time they are in the plant. The dashboard should be constantly active so as to ensure safetyprecautions can be executed whenever dangeris detected.
NFR-6	**Scalability**	If the area that needs to be monitored needs to be increased all one has to do is install new smart beacon devices and connect them to the same system as the previous beacons. It can also be replicated in different plants with different factors to be monitored giving it highlyscalability.

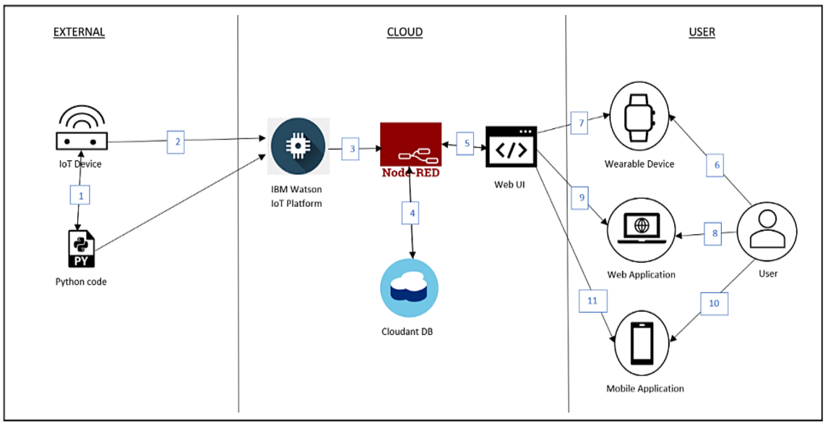
5. PROJECT DESIGN:

5.1 Data Flow Diagram

Data Flow Diagrams:

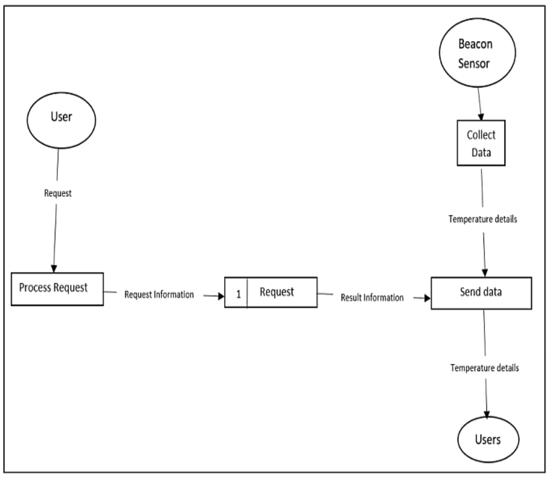
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

FLOW:



1. Necessary Python code for collecting temp. details from IoT device is written.
2. IoT device is connected with the IBM Watson IoT platform for gathering data.
3. Next step uses Node-Red services after IoT platform is all set.
4. Cloudant DB is used for storing and retrieving data.
5. Node-Red services are used to create Web application and UI designs.
6. (6,7,8,9,10,11) The user uses Smartwatch, Web and mobile app to receive various information and alerts.

DFD Level 0 (Industry Standard)

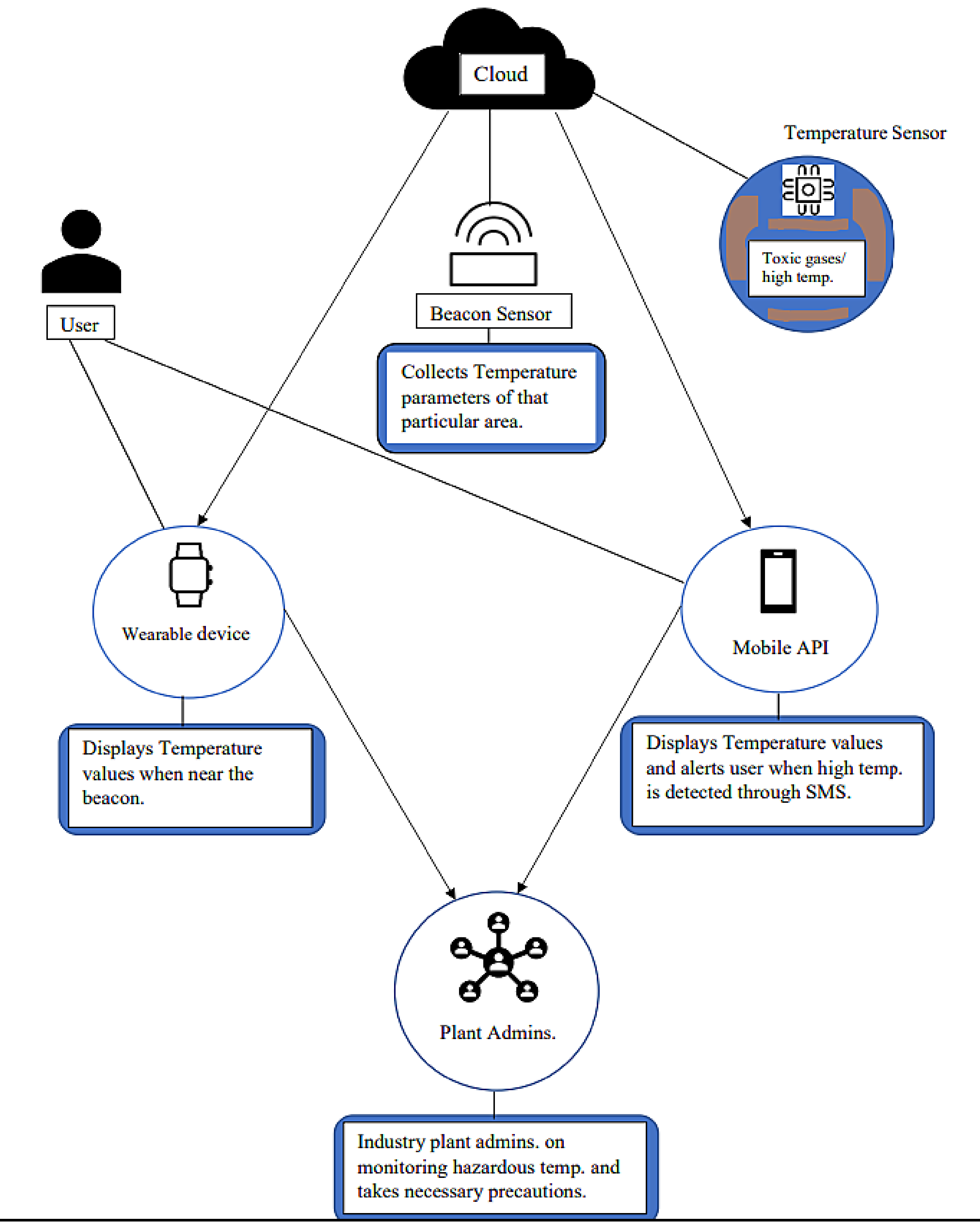


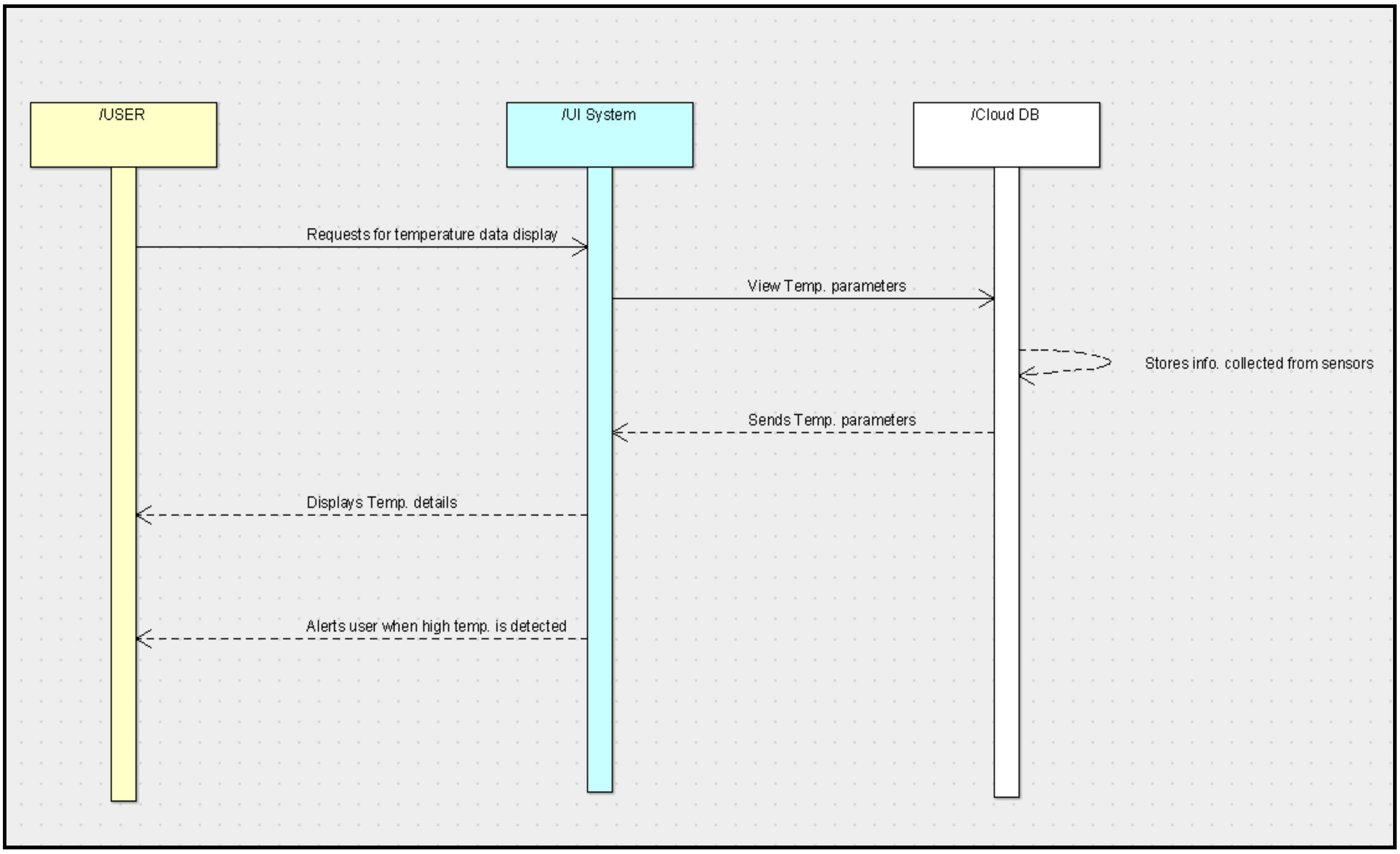
5.2 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Technician	Installation	USN-1	The technician must install the smart beacons at points to ensure the entire area of the plant is covered.	A beacon can be found in every area of the plant.	High	Sprint-1
	Data Gathering	USN-2	The beacons obtain the temperature of their respective area using sensors.	The temperature of areas within the plant is obtained.	High	Sprint-1
	Data Sync	USN-3	The beacons send their data to the cloud in the real time which is in turn sent to nearby wearable devices and the administrators dashboard.	Data is sent to the cloud successfully and synced with other devices.	High	Sprint-1
Worker	Wearable device display	USN-4	The wearable devices should display the data sent by beacons within the area.	The user can see the temperature of the area on their device.	High	Sprint-1
	Wearable device adjustments	USN-5	The user can adjust the size of the wearable device to better suit them.	The user can make adjustments to the device to make working with it more comfortable.	Low	Sprint-2
	Wearable display customization	USN-6	The user can adjust the device display to suit their needs on the device itself.	The user can modify the display of the device to increase readability.	Medium	Sprint-2
	SMS Notifications	USN-7	The user is sent a notification to their phone from the wearable device through an API when the area they are in reaches dangerous temperatures.	The user is informed of potential danger via SMS as soon as it is detected by the beacons.	High	Sprint-1
Administrator	Admin Dashboard	USN-8	The beacons send the data through the cloud to a dashboard which is run by the administrator.	The data of all the beacons can be viewed by the administrator of the plant.	High	Sprint-1
	Dashboard Customization	USN-9	The dashboard can be customized by the admin to suit their personal requirements and priorities.	The admin can customize the UI for their dashboard.	Medium	Sprint-2

5.3 Solution & Technical Architecture

SYSTEM ARCHITECTURE:



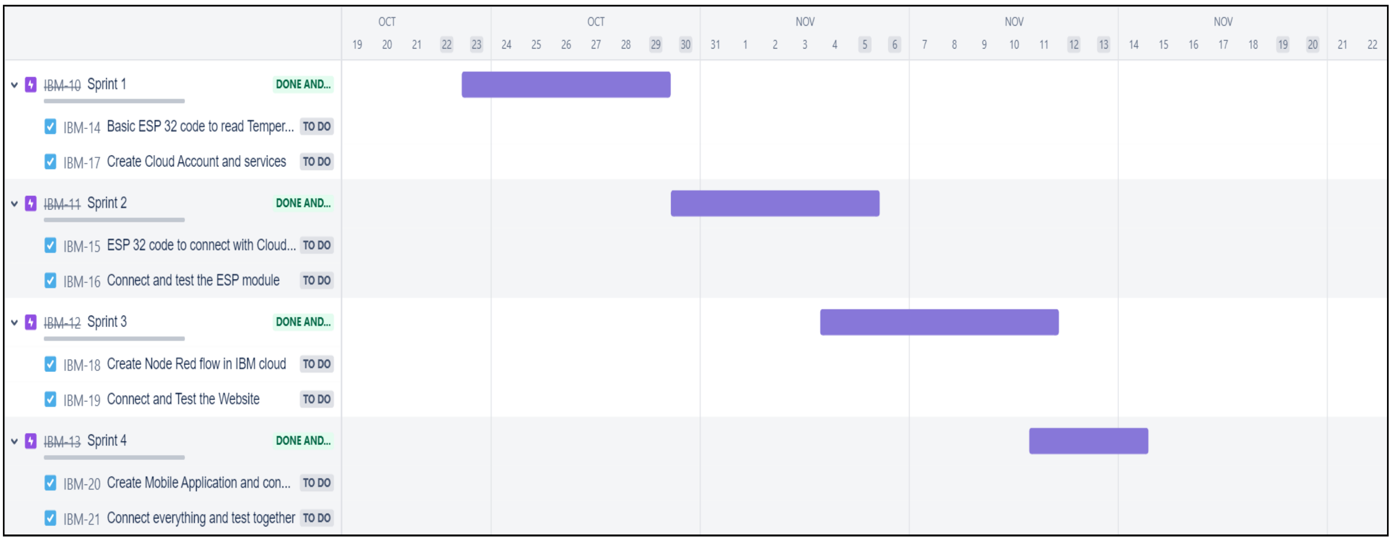


6. PROJECT PLANNING & SCHEDULING:

6.1 Sprint Planning & Estimation, 6.2 Sprint Delivery Schedule

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		29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		19 Nov 2022

6.3 Reports from JIRA



7. CODING & SOLUTIONING:

7.1 Feature 1

Displaying all temperature parameters, Humudity, Air quality using arduino IoT device and beacon sensors.

- Arduino device & Beacon sensor -To gether data live and display it and also to send it to the cloud database.
- Temperature sensor - Used for gathering temperaure related parameters.
- Cloud DB - Stores all the information received from the arduino device & Beacon.

Watson IoT platform is used here to display the gathered data from the sensor.

Arduino code:

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <DHT.h>
WiFiClient wifiClient;
String data3;
#define DHTTYPE DHT11
#define DHTPIN 15
#define MQTPIN 34
DHT dht(DHTPIN, DHTTYPE);
```

```
#define ORG "v6wg8x"
#define DEVICE_TYPE "projectFinal"
#define DEVICE_ID "FinalDeliverable"
#define TOKEN "A1ymH))p*JB&iMWNpY"
#define speed 0.034
void callback(char* topic, byte* playload, unsigned int payloadLength);
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char topic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, callback , wifiClient);
void publishData();
String command;
String data = "";
long duration;
float dist;
const int motorPin = 2;
void setup()
{
	Serial.begin(115200);
	dht.begin();
	pinMode(motorPin, OUTPUT);
	wifiConnect();
	mqttConnect();
}
void loop() {
	publishData();
	delay(2000);
	if (!client.loop()) {
		mqttConnect();
	}
}
```

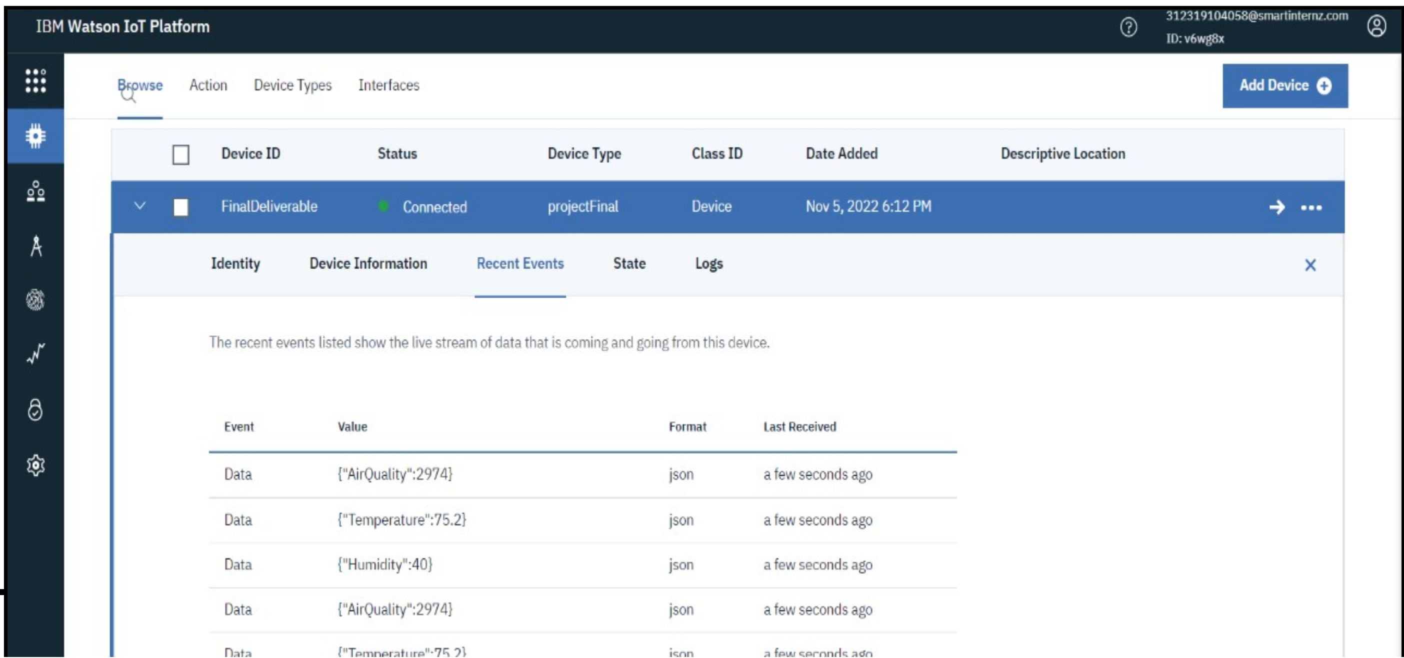
```
void wifiConnect() {
        Serial.print("Connecting to "); Serial.print("Wifi");
        WiFi.begin("JerroldWi-Fi","75779901");
        while (WiFi.status() != WL_CONNECTED) {
                delay(500);
                Serial.print(".");
        }
        Serial.print("WiFi connected, IP address: ");
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);
        }
        initManagedDevice();
        Serial.println();
        }
}
void initManagedDevice() {
        if (client.subscribe(topic)) {
                Serial.println("IBM subscribe to cmd OK");
        } else {
                Serial.println("subscribe to cmd FAILED");
        }
}
void publishData(){
```

```
	int sensorValue = analogRead(MQTPIN); //MQT 135 connected to GPIO 34 (Analog ADC1_CH6)
	int humid = dht.readHumidity();
	float temp = dht.readTemperature();
	String payload = "{\"Humidity\":0";
	payload += "}";
	if (client.publish(publishTopic, (char*) payload.c_str())) {

	Serial.println("Publish OK ");
	Serial.println(payload);
	}
	delay(500);
	payload = "{\"Temperature\":0";
	payload += "}";
	if (client.publish(publishTopic, (char*) payload.c_str())) {
		Serial.print("Publish OK ");
		Serial.println(payload);
	}
	delay(500);
	payload = "{\"AirQuality\":0";
	payload += "}";
	if (client.publish(publishTopic, (char*) payload.c_str())) {
		Serial.print("Publish OK");
		Serial.println(payload);
	}
}
void callback(char* subscribeTopic, byte* payload, unsigned int payloadLength) {
	Serial.print("callback invoked for topic:");
	Serial.println(subscribeTopic);
	for (int i = 0; i < payloadLength; i++) {
		data3 += (char)payload[i];
	}
```

```
if (data3 == "motoron") {
        Serial.println(data3);
        digitalWrite(motorPin,HIGH);
}
if(data3 == "motoroff"){
        Serial.println(data3);
        digitalWrite(motorPin,LOW);
        }
data3 = "";
delay(500);
}
```

Output:

```
Connecting to Wifi..Reconnecting MQTT client to
v6wg8x.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
```

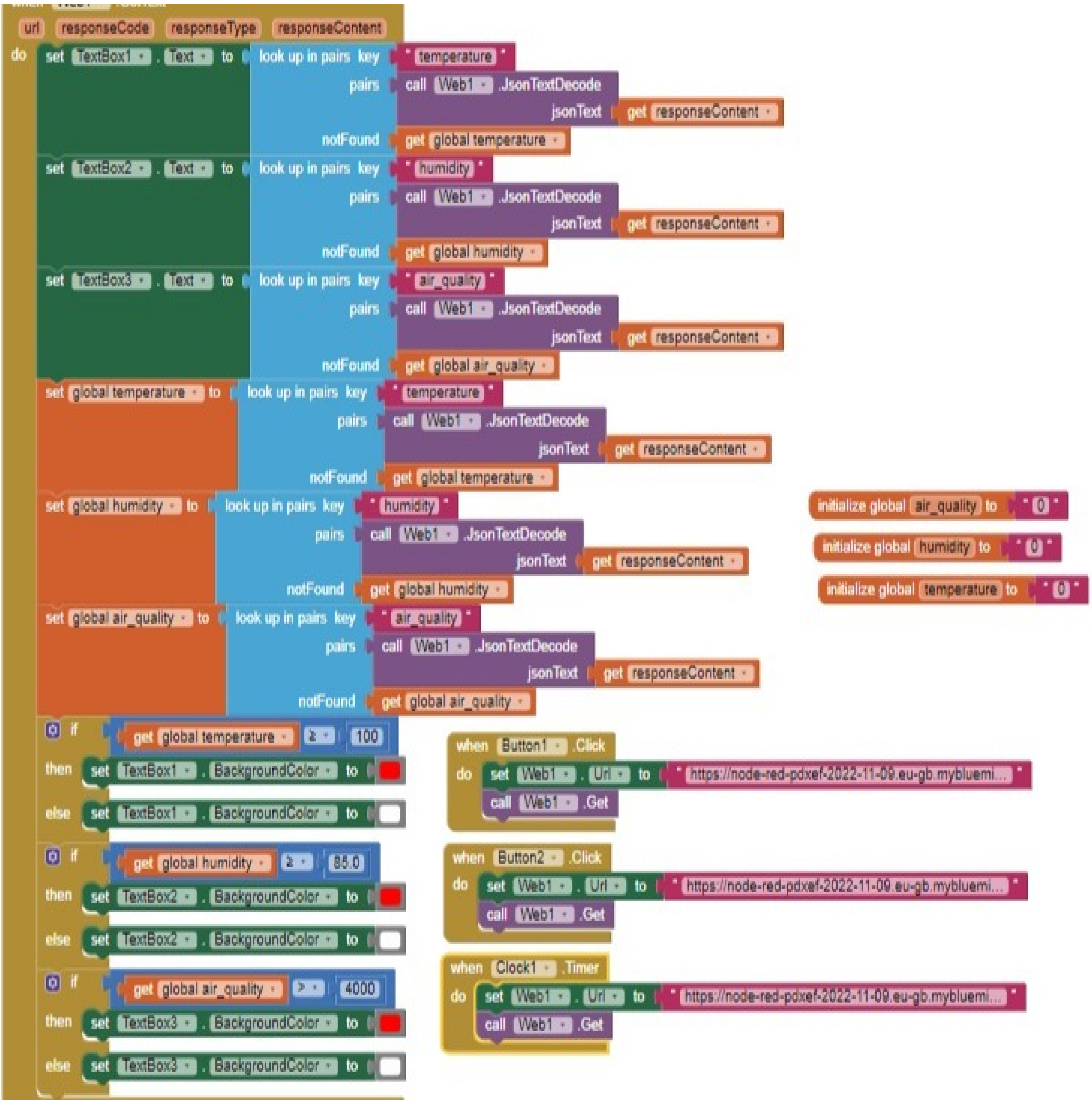


7.2 <u>Feature 2</u>

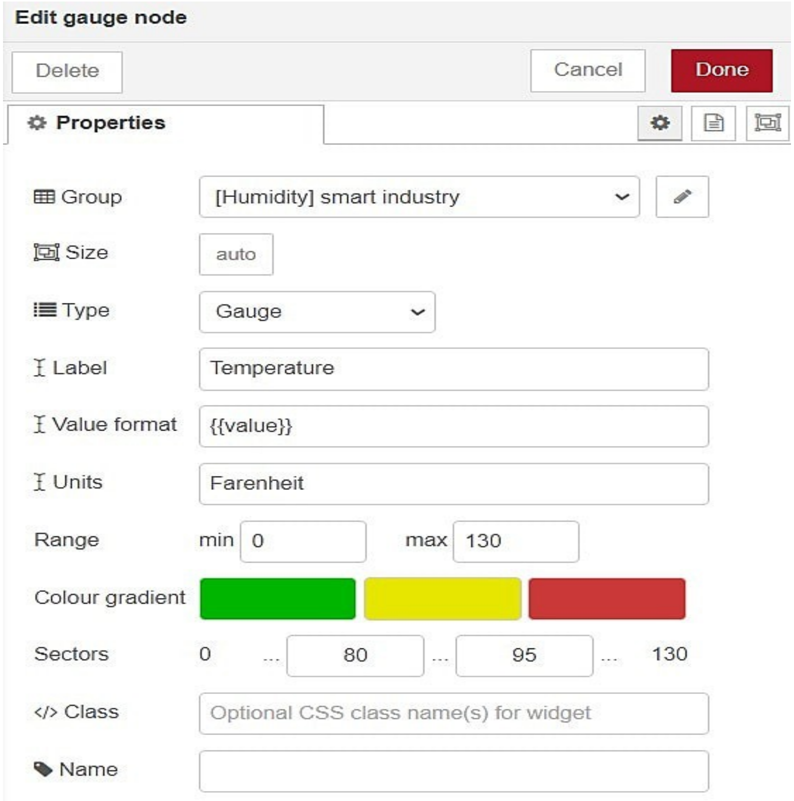
Providing alert (or) warning message to the user whenever toxic level of temperature, humidity and air quality is detected on that area via mobile application and web application.

Both Mobile App and Web App is used for displaying the temperature parameters and also for issuing warnings. A Red colour is displayed over the data fields to indicate that the current temperature is harmful.

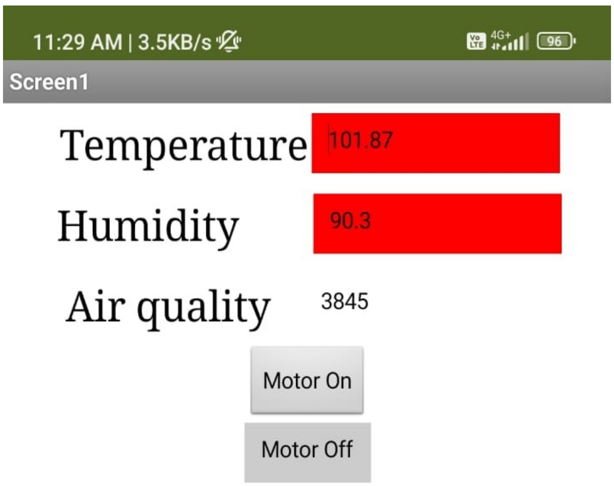
Mobile App code:



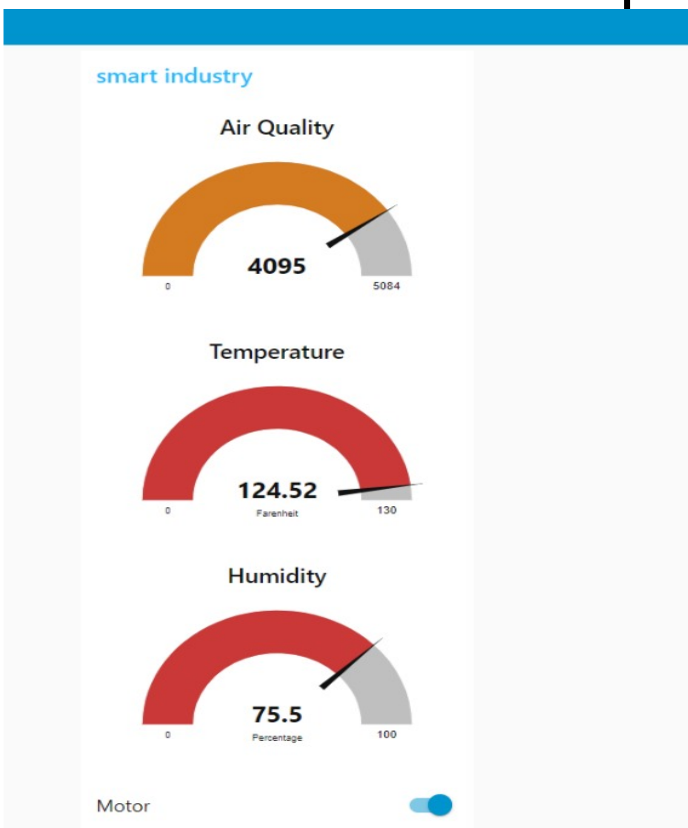
Web App configurations for alert message:



Output:



Mobile app



8. TESTING:

8.1 Test Cases

TESTCASE ID	FEATURE TYPE	COMPONENT	TEST SCENARIO	STEPS TO EXECUTE	TEST DATA	EXPECTED RESULT	ACTUAL RESULT	STATUS	TC FOR AUTOMATION (Y/N)	EXECUTED BY
DEV-TC-001	Functional	IoT Device	Verify whether the sensors are connected to the IoT device	1. Connect the sensor and boards to create the device. 2. Write a program to obtain the required values from the sensors.	Temperature, Air Quality, Humidity of environment	Obtain values using sensors from environment	Working as Expected	PASS	Y	JERROLD GIDEON S
DEV-TC-002	Functional	IoT Device	Verify whether the values are actually being sent from the IoT device.	1. Write a program to send the values to the selected cloud service. 2. Implement the program into the device and run it.	Temperature, Air Quality, Humidity of environment	Values should be sent to the chosen cloud service	Working as Expected	PASS	Y	JERROLD GIDEON S
CLD-TC-001	Functional	IBM Cloud	Verify the presence of a cloud database to store data.	1. Create an IBM cloud account to create a cloud database. 2. Create a database using cloudant and store some values.	Random python values/environmental values	A database should be present that can send, receive and store data.	Working as Expected	PASS	Y	JERRY BRITTO J
CLD-TC-002	Functional	IBM Cloud	Verify whether the values are sent in realtime to the database.	1. Connect the database to the IoT device using keys. 2. Activate the IoT device. 3. Check "recent events" to check inputs sent from the IoT device	Random python values/environmental values	Values should be received and stored from the IoT Device	Working as Expected	PASS	Y	JERRY BRITTO J
UI-TC-001	Functional	Node Red Flow	Verify whether data is received from the cloud database and displayed in the debug menu.	1. View the node red flow editor for the cloud database. 2. Connect the appropriate nodes to obtain values from the IoT device. 3. Deploy the node red flow.	Data values from cloud in the debug menu	Values obtained from the IoT device through the cloud is obtained.	Working as Expected	PASS	Y	AARON DANARAJ F

UI-TC-002	UI	Node Red Web UI	Verify whether the datacan be viewedon the web UI	1. Enter nodesto display thedata values using gauges in the web UI. 2. Enterthe appropriate urlto view the web UI.	Datavalues from cloudin the web UI	Obtained values are displayed using gauges in theUI	Working as Expected	PASS	Y	AARON DANARAJ F
MB-TC-001	Functional	Node Red	Verify whether responses can be made from the web and mobile applicatio n.	1. Add nodesin the nodeflow editor to allow for http responses. 2. Createbuttons to view on the app andweb UI forrequ ests.	Turning on and off of light usinga button	Http responses betweenthe App andnode red flow occour based on inputs.	Working as Expected	PASS	Y	AJESH T M
MB-TC-002	UI	MIT App Inventor	Verify whether data is received in real time and can be viewed in themobile application.	1. Using the MIT App inventorbuild the userinterface for the mobile appication. 2. Connect the mobile app to the cloudservice to obtain and display the data values.	Data valuesfrom cloud	Allvalues can be viewed on the mobile application at any time.	Working as Expected	PASS	Y	AJESH T M

8.2 User Acceptance Testing

1. Defect Analysis:

This report shows the numberof resolved or closedbugs at each severity level, and howthey were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	3	1	2	11
Duplicate	2	0	2	0	4
External	2	1	1	2	6
Fixed	6	0	1	5	12
Not Reproduced	2	0	1	1	4

Skipped	0	2	0	0	2
Won't Fix	0	3	2	0	5
Totals	17	9	8	10	44

2. Test Case Analysis:

This report shows the numberof test cases that have passed, failed,and untested.

Section	Total Cases	Not Tested	Fail	Pass
Physical Device	15	0	0	15
Connectivity	11	0	0	11
Web User Interface	6	0	0	6
Mobile Application	6	0	0	6
Final Report Output	9	0	0	9

9. RESULTS:

9.1 Performance Metrics

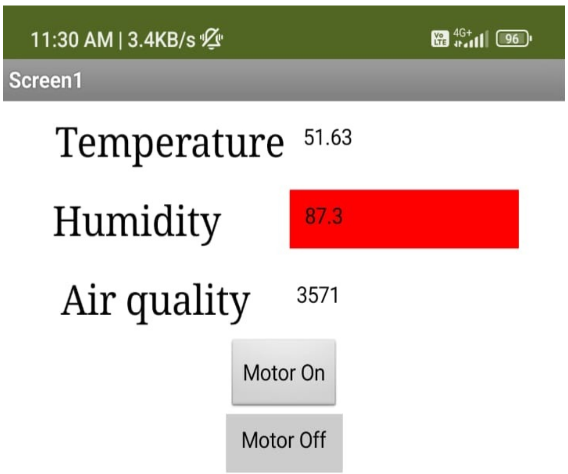
For Temperature Humidity sensor calculation the temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy.

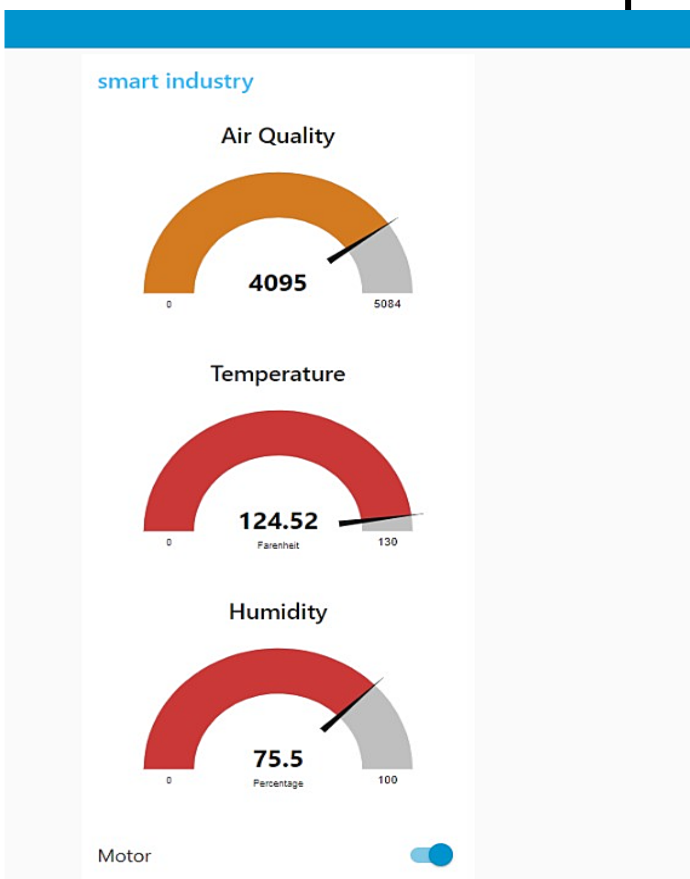
For Air Quality sensor calculation the MQ-135 gas sensor needs a load resistor. The load resistor serves to adjust the sensor’s sensitivity and accuracy. The value can range anywhere from 10KΩ to 47KΩ (the higher the resistance, the more sensitive the sensor becomes). So as per the load resistor the values reading should be tuned.

Based on these readings performance of the devices will be checked.

Output:

```
Connecting to Wifi..Reconnecting MQTT client to
v6wg8x.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
Publish OK
{"Humidity":40.00}
Publish OK {"Temperature":75.20}
Publish OK{"AirQuality":0}
```





10. ADVANTAGES & DISADVANTAGES:

Advantages:

1. Easier and more modern approach to detect and display Temperature parameters, Air quality and Humidity.

2. Multiple ways to display temperature parameters.

3. Easy to understand, comfortable and user-friendly.

4. Alert messages feature to warn the user on the mobile application about high temperatures.

5. Smart wearable device which makes it handy and easily accessible to know information about the temperature in real time at all times.

6. Large number of people can use this device simultaneously which makes it more scalable.

7. It ensures the safety of the workers in industries working in hazardous areas.

8. Useful for the administrators to identify and resolve problems regarding the temperatures in hazardous areas.

Disadvantages:

1. Cost of the device depends on the amount of device purchased for usage by the industry.

2. Data must be relayed accurately because false or misinformation may cause delay for the workers.

3. Regular maintenance of the devices and the systems must be performed.

4. System failures may cause dangerous consequences to the workers.

5. Wearable devices sometimes gets in the way of the working and causes hassle to workers.

11. CONCLUSION:

We hope to gain hands-on experience with the trending technologies of "Embedded System" and "Internet of Things" through this project. IoT-enabled industrial monitoring systems have become increasingly popular in a variety of industries because they improve safety standards by providing real-time monitoring of critical parameters such as temperature, humidity, and smoke, as well as alerting officials and workers regularly. The implementation is not only for safety reasons, but it also has the potential to increase industry yields. In our project, the Internet of Things (IoT) is used to collect data and communicate through the internet. We hope that our project will be beneficial enough to be implemented in industries across India, saving lives and property from accidents and risks that are often overlooked by industry personnel and users. Companies in the industrial and logistics sectors can better meet the new era of instant needs by utilizing the Industrial Internet of Things (IoT). IoT technologies are used in manufacturing processes and across supply chains in the Industrial Internet of Things.

12. FUTURE SCOPE:

Industrial IoT strategy should include machine learning and big data technology in addition to data from devices and sensors, harnessing the combination of existing sensor data, machine to machine (M2M) connectivity, and automation technologies to deliver greater insight back to the business. We are planning to make this project more cost-efficient and more reliable in the future. We are also planning on making the devices more effective by adding new features which makes it more faster and accessible by all the users. More new informations regarding the temperature parameters and various other details are planned to be implemeted in the future. Security features are planned to be made more robust and reliable to avoid system failure. New and various other methods for sending out alert and warnings to the user are planned to be added in the future.

13. APPENDIX:

Source code link: https://github.com/IBM-EPBL/IBM-Project-10538-1659185995/tree/main/Final%20Deliverables/Source%20Code

Github Page link: https://github.com/IBM-EPBL/IBM-Project-10538-1659185995

Project Demo link: https://github.com/IBM-EPBL/IBM-Project-10538-1659185995/blob/main/Final%20Deliverables/Project_video.mp4