PROJECT DEVELOPMENT PHASE

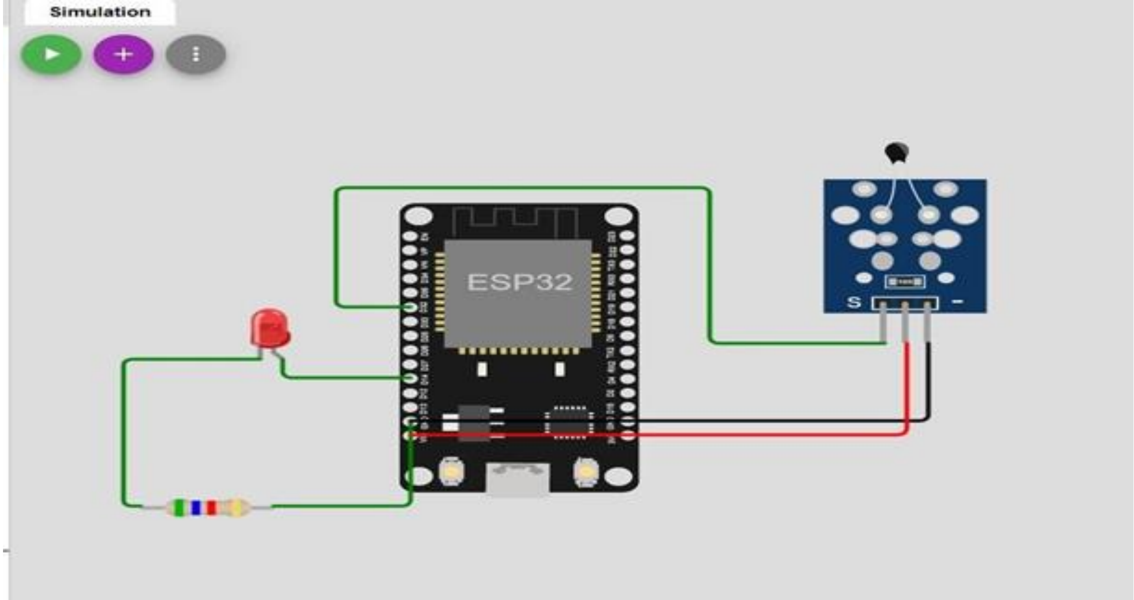
DELIVERY OF SPRINT-1

Date	29 October 2022
Team ID	PNT2022TMID20420
Project Name	Industry Specific Intelligence Fire Management System

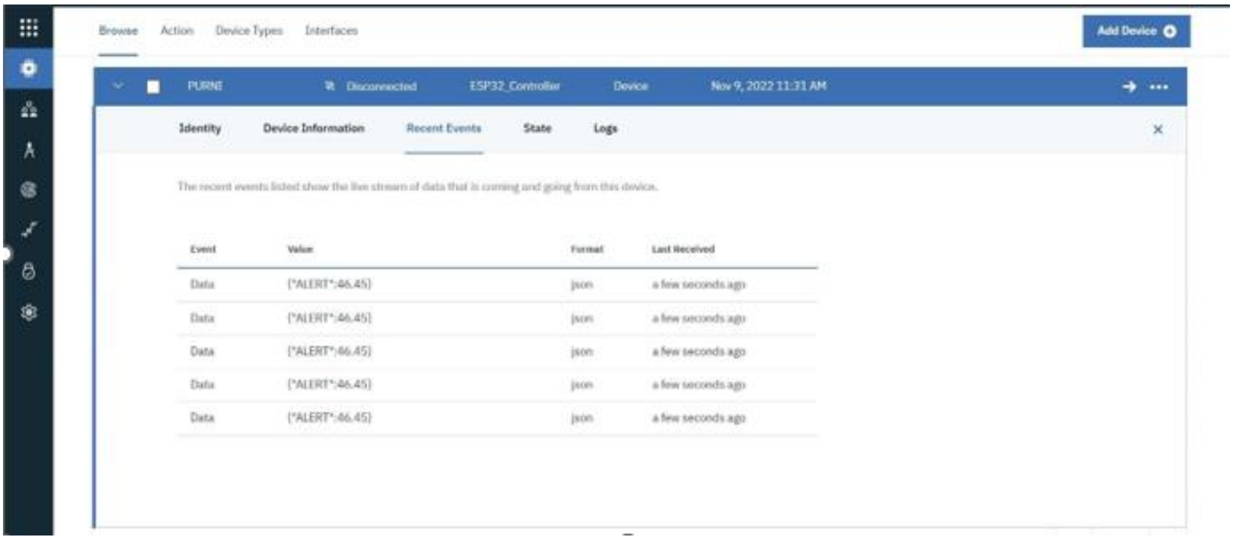
```
#include <WiFi.h>

#include <PubSubClient.h>

#define temp_pin 15 void callback(char* subscribetopic,byte* payload, unsigned int payloadLength);
#define ORG "jesccj"

#define DEVICE_TYPE "ESP32_Controller" #define DEVICE_ID "PURNI"

#define TOKEN "*Vzh&EwwgbRpqohJd+" String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com"; char publishTopic[]="iot-2/evt/Data/fmt/json"; char subscribeTopic[]="iot-2/cmd/test/fmt/String"; char authMethod[]="use-token-auth"; char token[]=TOKEN; char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

WiFiClient wifiClient;

PubSubClient client(server,1883,callback,wifiClient);

// should match the Beta Coefficient of the thermistor void setup() {

Serial.begin(9600); analogReadResolution(10); pinMode(32,INPUT); pinMode(14,OUTPUT);


wificonnect(); mqttconnect();

} void loop() {     const float BETA = 3950; // should match the Beta Coefficient of the thermistor int analogValue = analogRead(A4); float temp = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 / 298.15) - 273.15;           //float temp = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 / 298.15) - 273.15;

Serial.print("Temperature: ");


Serial.print(temp); Serial.println(" °C"); if(temp>=35){ PublishData2(temp); digitalWrite(14, HIGH);

}else{

digitalWrite(14, LOW); PublishData1(temp);

} delay(1000); if(!client.loop()){ mqttconnect();
```

```
}


//delay(2000);
} void PublishData1(float tem){   mqttconnect();
String payload= "{\"temp\":"; payload += tem;    payload+="}";


Serial.print("Sending payload:");
Serial.println(payload);


if(client.publish(publishTopic,(char*)payload.c_str())){
Serial.println("publish ok");
} else{
Serial.println("publish failed");
} }
void PublishData2(float tem){ mqttconnect();
String payload= "{\"ALERT\":"; payload += tem;   payload+="}";


Serial.print("Sending payload:");
Serial.println(payload);


if(client.publish(publishTopic,(char*)payload.c_str())){
Serial.println("publish ok");
} else{
Serial.println("publish failed");
}
}
void mqttconnect(){ if(!client.connected()){
Serial.print("Reconnecting to");
Serial.println(server);
while(!!!client.connect(clientID, authMethod, token)){
Serial.print("."); delay(500);
}
initManagedDevice();

Serial.println();
}
}
void wificonnect(){
Serial.println(); Serial.print("Connecting to");

WiFi.begin("Wokwi-GUEST","",6); while(WiFi.status()!=WL_CONNECTED){ delay(500);
Serial.print(".");
}
Serial.println(""); Serial.println("WIFI CONNECTED"); Serial.println("IP address:");
Serial.println(WiFi.localIP());
} void initManagedDevice(){
if(client.subscribe(subscribeTopic)){
Serial.println((subscribeTopic)); Serial.println("subscribe to cmd ok");
}else{
Serial.println("subscribe to cmd failed");
}
}
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];
}
Serial.println("data:"+ data3); if(data3=="lighton"){ Serial.println(data3); digitalWrite(14,HIGH);
}else{
Serial.println(data3); digitalWrite(14,LOW);
}
data3="";
}
```

DIAGRAM



OUTPUT



PURNE Disconnected ESP32_Controller Device Nov 9, 2022 11:31 AM

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

Event	Value	Format	Last Received
Data	{"ALERT":46.45}	json	a few seconds ago
Data	{"ALERT":46.45}	json	a few seconds ago
Data	{"ALERT":46.45}	json	a few seconds ago
Data	{"ALERT":46.45}	json	a few seconds ago
Data	{"ALERT":46.45}	json	a few seconds ago