Project Development Phase
Sprint-4

Date	12-November 2022
Team ID	PNT2022TMID35932
Project Name	Hazardous Area Monitoring for Industrial Plant powered by IoT

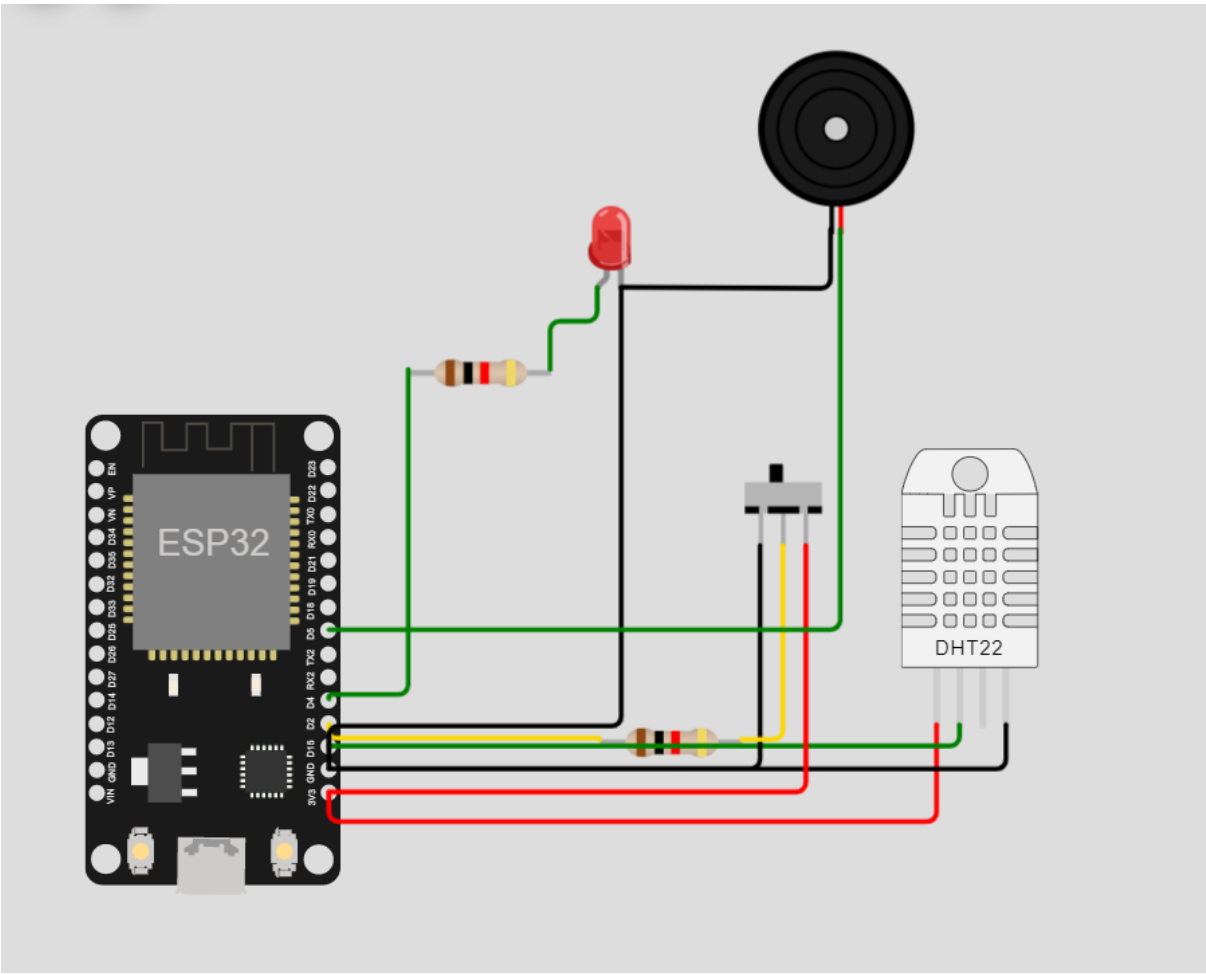
Updated IoT End device ESP32:

```
#include "DHT.h"// Library for dht22
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include <HTTPClient.h>//library for HTTP requests


#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
const int LED = 4;
//GAS SENSOR MQ-02
#define GAS_SENSOR 2
String alarmon = "{\"Alar\":1}";

//Your Domain name with URL path or IP address with path
String serverName = "http://169.51.205.238:32312/command";

unsigned long lastTime = 0;
unsigned long timerDelay = 1000;

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "bxddo9"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "Assign4"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "45625689713"     //Token
String data3;
float h, t;
int val;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform and
format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND
COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
```

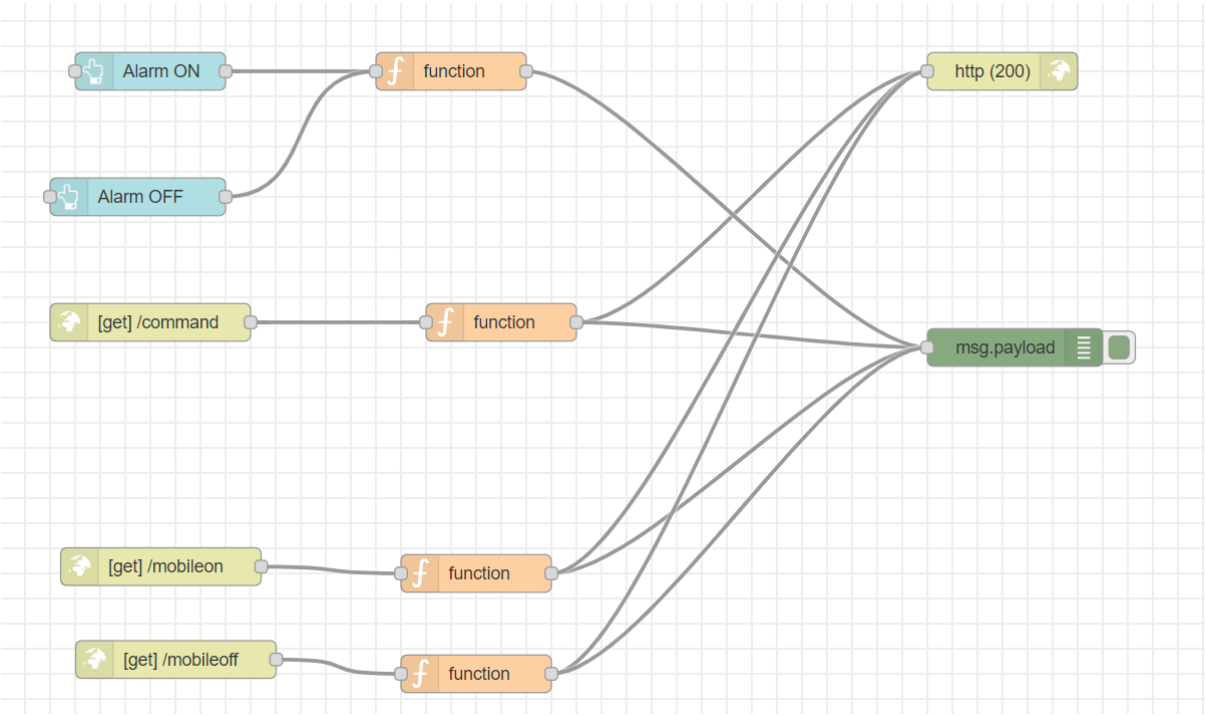
```
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by
passing parameter like server id,portand wificredential

const int DHT_PIN = 15;
bool al;

void setup() {
  Serial.begin(115200);
  Serial.println();
  pinMode(LED,OUTPUT);
  //digitalWrite(LED,HIGH);
  delay(10);
  wificonnect();
  mqttconnect();
}

void loop() {
  val = digitalRead(GAS_SENSOR);

  h = dht.readHumidity();
  t = dht.readTemperature();
  Serial.print("Temperature:");
  Serial.println(t);
  Serial.print("Humid:");
  Serial.println(h);
  Serial.print("Gas Sensor:");
  Serial.println(val);
  if(t > 45 || val == 1)
  {  al = 1;
  }
  else
  {  al = 0;
  }
  PublishData(t, h, val, al);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
  if ((millis() - lastTime) > timerDelay) {
    //Check WiFi connection status
    if(WiFi.status()== WL_CONNECTED){
      HTTPClient http;

      String serverPath = serverName + "?temperature=24.37";

      // Your Domain name with URL path or IP address with path
      http.begin(serverPath.c_str());

      // Send HTTP GET request
      int httpResponseCode = http.GET();

      if (httpResponseCode>0) {
       // Serial.print("HTTP Response code: ");
        //Serial.println(httpResponseCode);
        String payload = http.getString();
        //Serial.println(payload);

        if(payload == alarmon)
        {
          digitalWrite(LED,HIGH);
          tone(5,262,2000);
        }
        else
```

```
        {
          digitalWrite(LED,LOW);
          digitalWrite(5,LOW);
        }
      }
      else {
        Serial.print("Error code: ");
        Serial.println(httpResponseCode);
      }
      // Free resources
      http.end();
    }
    else {
      Serial.println("WiFi Disconnected");
    }
    lastTime = millis();
  }
}

/*.....................................retrieving to Cloud.............................*/

void PublishData(float temp, float humid, int vol,int alarm) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"Temperature\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
  payload += ","\"Gas_Sensor\":";
  payload += val;
  payload += ","\"Alarm\":";
  payload += al;
  payload += "}";


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


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will
print publish ok in Serial monitor or else it will print publish failed
    Serial.println("");
  } else {
    Serial.println("Publish failed");
  }

}


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

     initManagedDevice();
     Serial.println();
```

```
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  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++) {
    Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
  if(data3=="lighton")
  {
Serial.println(data3);
digitalWrite(LED,HIGH);
  }
  else
  {
Serial.println(data3);
digitalWrite(LED,LOW);
}
data3="";
}
```

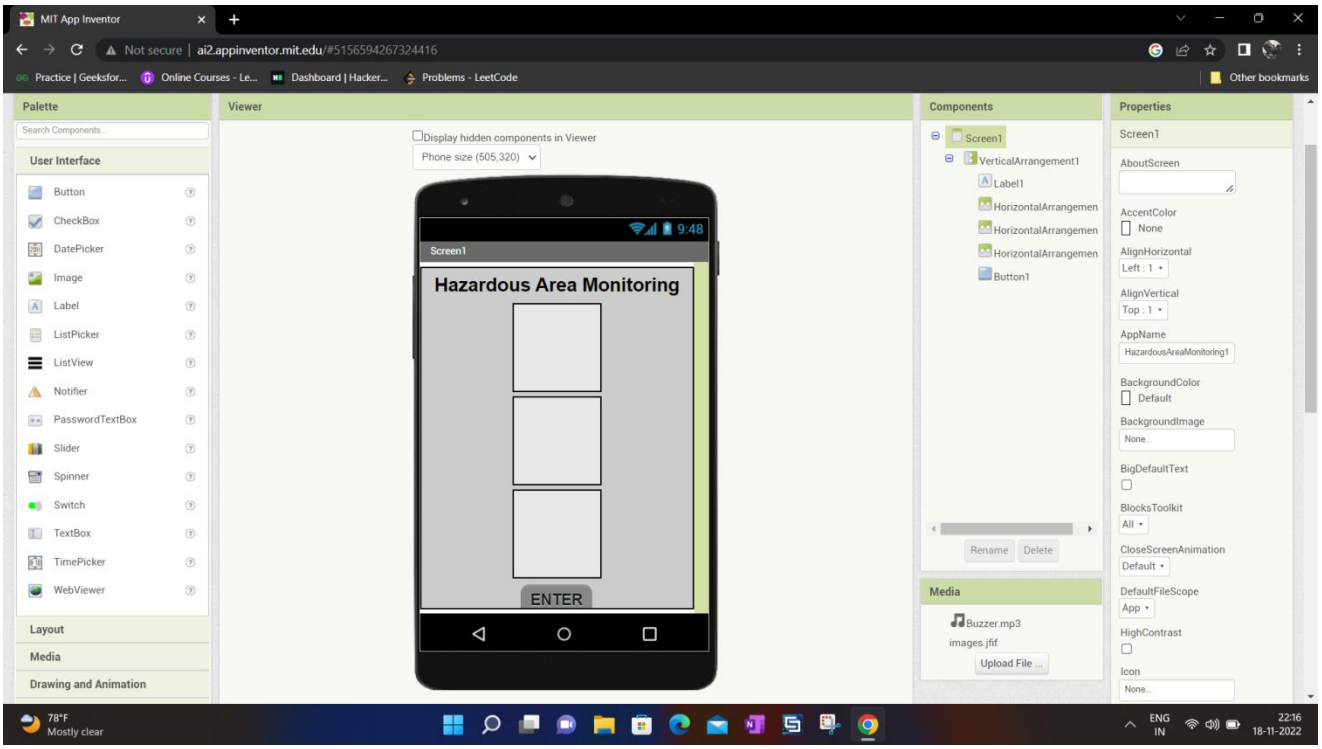


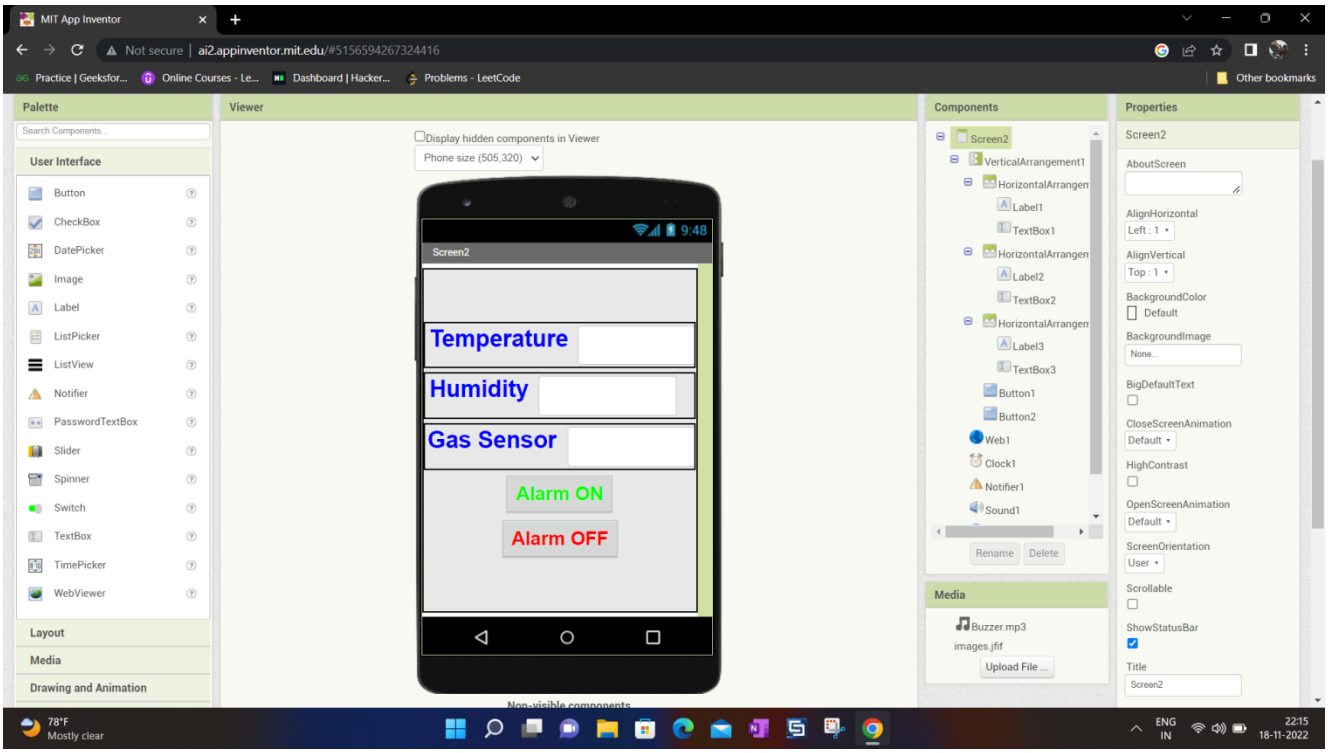
Node Red flow-2:

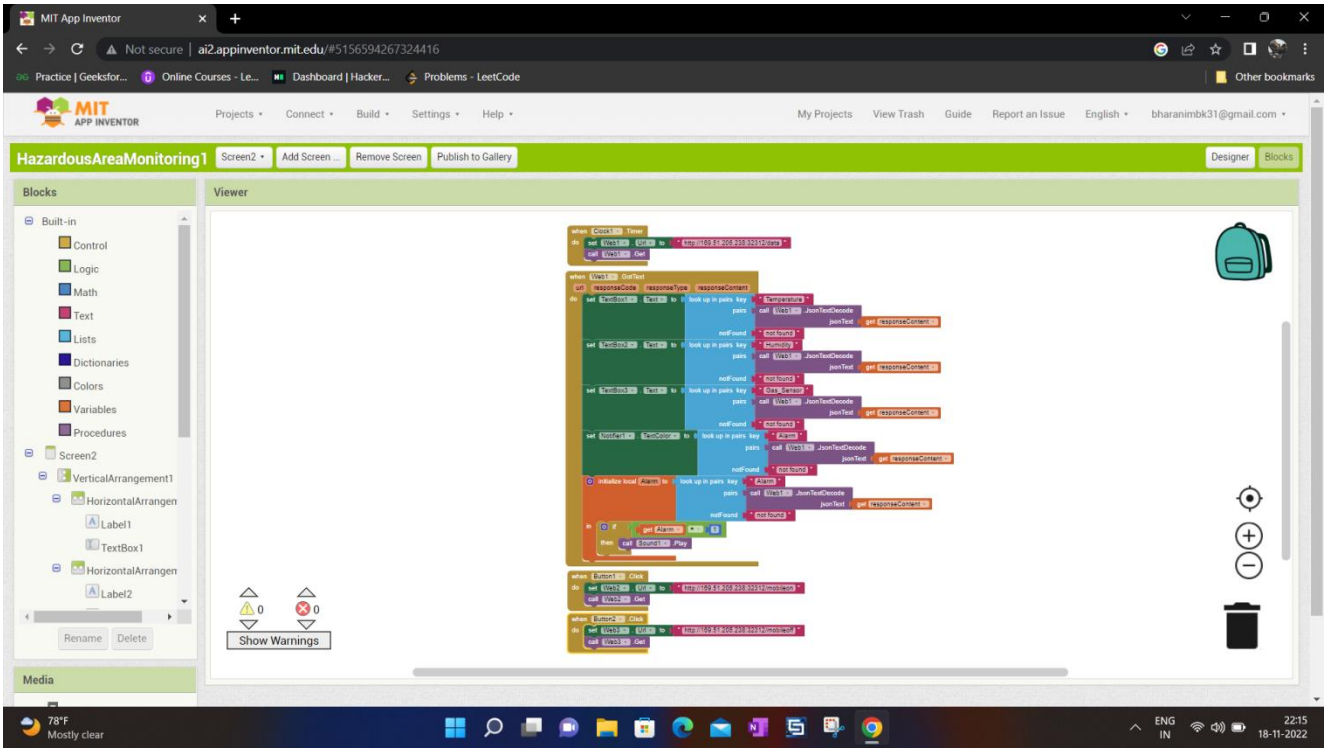


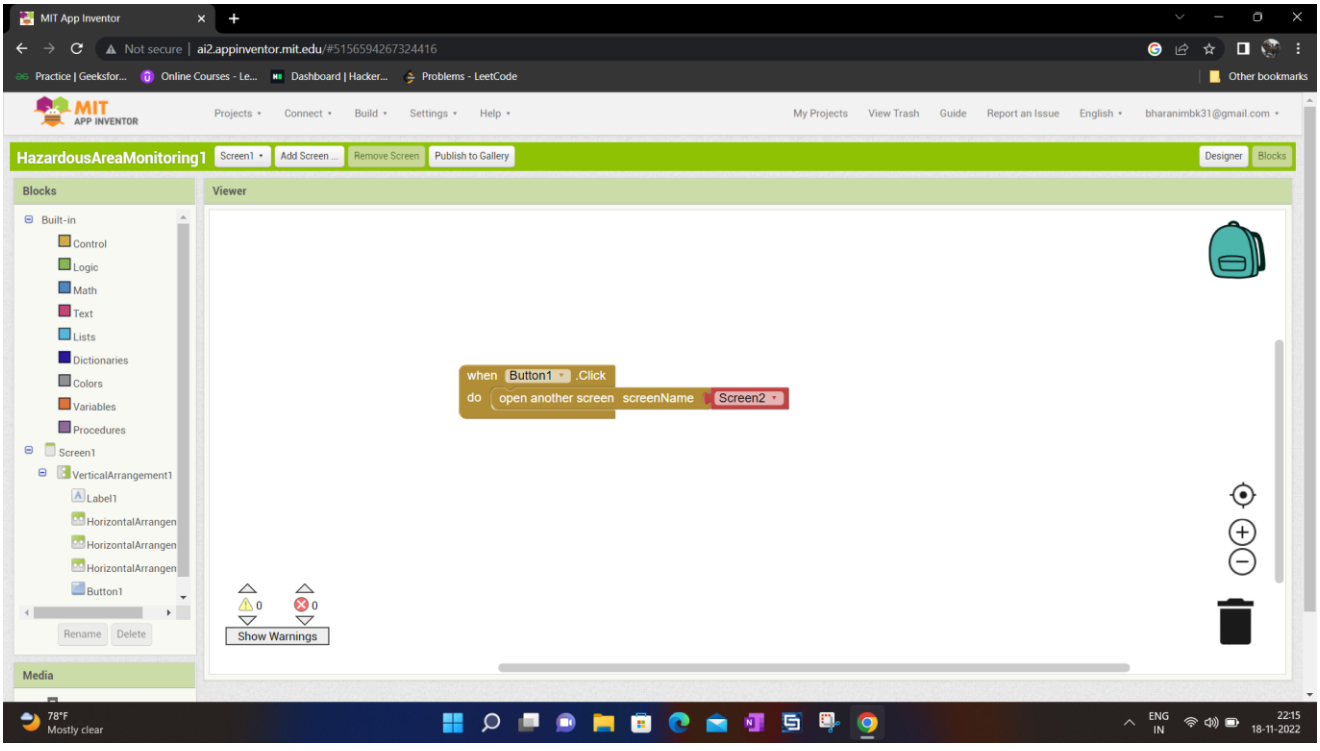
[{"id":"1819418ffaee3ee5","type":"tab","label":"Flow
2","disabled":false,"info":"","env":[]},{"id":"7f5cf345.63f56c","type":"http
response","z":"1819418ffaee3ee5","name":"","statusCode":"200","headers":{},"x":840,"y":
200,"wires":[]},{"id":"e71c7a7d.e7c598","type":"debug","z":"1819418ffaee3ee5","name":"",
"active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","x":850,"y"
:420,"wires":[]},{"id":"c7807102.3f433","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"command","method":"get","upload":false,"sw
aggerDoc":"","x":160,"y":400,"wires":[["60410cde.562a34"]]},{"id":"60410cde.562a34","typ
e":"function","z":"1819418ffaee3ee5","name":"","func":"msg.payload =
{'Alar':global.get('Alar')}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":440,"y":400,"wires":[["7f
5cf345.63f56c","e71c7a7d.e7c598"]]},{"id":"698da80c7836cea4","type":"ui_button","z":"18
19418ffaee3ee5","name":"","group":"66bce42692eed6b1","order":3,"width":0,"height":0,"
passthru":false,"label":"Alarm
ON","tooltip":"","color":"","bgcolor":"","icon":"","payload":"1","payloadType":"num","topic
":"topic","topicType":"msg","x":160,"y":200,"wires":[["6878957e52c678f6"]]},{"id":"687895
7e52c678f6","type":"function","z":"1819418ffaee3ee5","name":"","func":"if(msg.payload
== 1)\n{\n   msg.payload=1;\n}\nelse\n{\n
msg.payload=0;\n}\nglobal.set('Alar',msg.payload)\nmsg.payload =
{'Alar':global.get('Alar')}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":400,"y":200,"wires":[["e7
1c7a7d.e7c598"]]},{"id":"9747945c8b11b89f","type":"ui_button","z":"1819418ffaee3ee5","
name":"","group":"66bce42692eed6b1","order":3,"width":0,"height":0,"passthru":false,"lab
el":"Alarm
OFF","tooltip":"","color":"","bgcolor":"","icon":"","payload":"0","payloadType":"num","topi
c":"topic","topicType":"msg","x":150,"y":300,"wires":[["6878957e52c678f6"]]},{"id":"25c01
3b66f1577be","type":"function","z":"1819418ffaee3ee5","name":"","func":"global.set('Alar'
,1)\nmsg.payload = {'Alar':1}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":420,"y":600,"wires":[["e7
1c7a7d.e7c598","7f5cf345.63f56c"]]},{"id":"31722af4312e8a88","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"mobileon","method":"get","upload":false,"sw
aggerDoc":"","x":168.39999389648438,"y":594.5999755859375,"wires":[["25c013b66f1577
be"]]},{"id":"f1fe5338e83a445a","type":"http
in","z":"1819418ffaee3ee5","name":"","url":"mobileoff","method":"get","upload":false,"sw
aggerDoc":"","x":180.39999389648438,"y":668.5999755859375,"wires":[["0dbffb65ea00a5
7f"]]},{"id":"0dbffb65ea00a57f","type":"function","z":"1819418ffaee3ee5","name":"","func"
:"global.set('Alar',0)\nmsg.payload = {'Alar':0}\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","libs":[],"x":420,"y":680,"wires":[["7f
5cf345.63f56c","e71c7a7d.e7c598"]]},{"id":"66bce42692eed6b1","type":"ui_group","name"
:"ALARM","tab":"921bcc13fd3771df","order":4,"disp":true,"width":7,"collapse":false},{"id":
"921bcc13fd3771df","type":"ui_tab","name":"Hazardous Area
Monitoring","icon":"dashboard","disabled":false,"hidden":false}]

Mobile App:


MIT App Inventor
Not secure | ai2.appinventor.mit.edu/#5156594267324416
Palette
Viewer
Components
Properties
Screen1
Hazardous Area Monitoring
ENTER
Media
Buzzer.mp3
images.jfif
Upload File ...


MIT App Inventor
Not secure | ai2.appinventor.mit.edu/#5156594267324416
Palette
Viewer
Components
Properties
Screen2
Temperature
Humidity
Gas Sensor
Alarm ON
Alarm OFF
Media
Buzzer.mp3
images.jfif
Upload File ...


MIT App Inventor
Not secure | ai2.appinvertor.mit.edu/#5156594267324416
Practice | Geeksfor...
Online Courses - Le...
Dashboard | Hacker...
Problems - LeetCode
Other bookmarks
MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
bharanimbk31@gmail.com
HazardousAreaMonitoring1
Screen2
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen2
VerticalArrangement1
HorizontalArrangen
Label1
TextBox1
HorizontalArrangen
Label2
Rename
Delete
Media
Viewer
Show Warnings
78°F
Mostly clear
ENG
IN
22:15
18-11-2022


MIT App Inventor
Not secure | ai2.appinvertor.mit.edu/#5156594267324416
Practice | Geeksfor...
Online Courses - Le...
Dashboard | Hacker...
Problems - LeetCode
Other bookmarks
MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
bharanimbk31@gmail.com
HazardousAreaMonitoring1
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen1
VerticalArrangement1
Label1
HorizontalArrangen
HorizontalArrangen
HorizontalArrangen
Button1
Rename
Delete
Media
Viewer
when Button1 .Click
do open another screen screenName Screen2
Show Warnings
78°F
Mostly clear
ENG
IN
22:15
18-11-2022