SPRINT -4

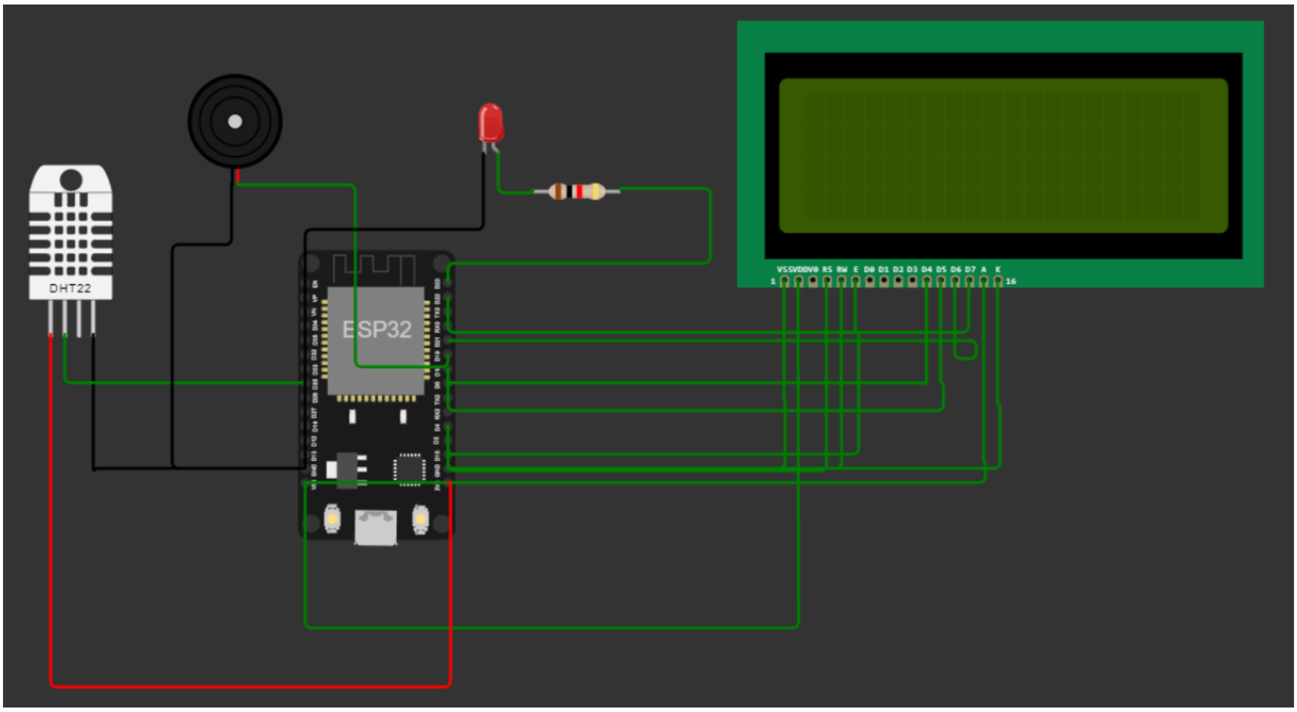
Date	13 NOV 2022
Team ID	PNT2022TMID35909
Project Name	GAS LEAKAGE MONITORING AND ALERTING SYSTEMS FOR INDUSTRIES

UPDATED CODE:

```
#include <LiquidCrystal.h>
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHTesp.h"
#define LED 23
#define BUZZER_PIN 19  // define type of sensor DHT 11
String command;
void callback(char* subscribetopic, byte* payload,
unsigned int payloadLength);

#define ORG "ckdbr5"//IBM ORGANITION ID
#define DEVICE_TYPE "123"//Device type mentioned in ibm
watson IOT Platform
#define DEVICE_ID "252725"//Device ID mentioned in ibm
watson IOT Platform
#define TOKEN "27252527"     //Token
String data3;
int trigger;
float h, t;


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

const int DHT_PIN = 25;
DHTesp dhtSensor;
LiquidCrystal lcd(4,15,5,18,21,22);
int ThreshHold = 60;

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

void setup() {
  Serial.begin(9600);
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  lcd.begin(16,2);
  pinMode(LED,OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  wificonnect();
  mqttconnect();
}

void loop() {

  delay(5000);
  digitalWrite(LED, HIGH);
  TempAndHumidity  data = dhtSensor.getTempAndHumidity();
  Serial.println("Temperature: " +
String(data.temperature, 2) + "°C");
  Serial.println("Humidity: " + String(data.humidity, 1)
+ "%");

  float gassensor=random(0,100);
  Serial.print(F("Gas Concentration: "));
  Serial.println(gassensor);

  if (gassensor>ThreshHold)
  {
    trigger=1;
    Serial.println(F("GAS LEAKED ALERT!"));
    Serial.println();
    lcd.clear();
    lcd.print ("GAS LEAKAGE :(");
    tone(BUZZER_PIN,31);
    delay (1000);
    lcd.clear();
    lcd.print ("ALERT!!!");
    delay(1000);
    noTone(BUZZER_PIN);

  }

  else
  {
    trigger=0;
    Serial.println(F("SAFE!"));
    Serial.println();
    lcd.clear();
    lcd.print ("ALL GOOD :)");
    delay(1000);
    lcd.clear();
    lcd.print ("SAFE!");
    delay(1000);
  }

  PublishData(data.temperature,data.humidity,gassensor,tr
igger);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}

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

void PublishData(float temp, float humid , float
sensorvalue ,int trigger) {
  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 += "," "\"Humidity\":";
  payload += humid;
  payload += "," "\"GasConcentration\":";
  payload += sensorvalue;
  payload += "," "\"Status\":";
  payload += trigger;
  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
  } else {
    Serial.println("Publish failed");
    delay(2000);
  }
}
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)
{
  delay(3000);
  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=="shutdown")
  {
    Serial.println(data3);
    digitalWrite(LED,LOW);
  }
  data3="";
}
```

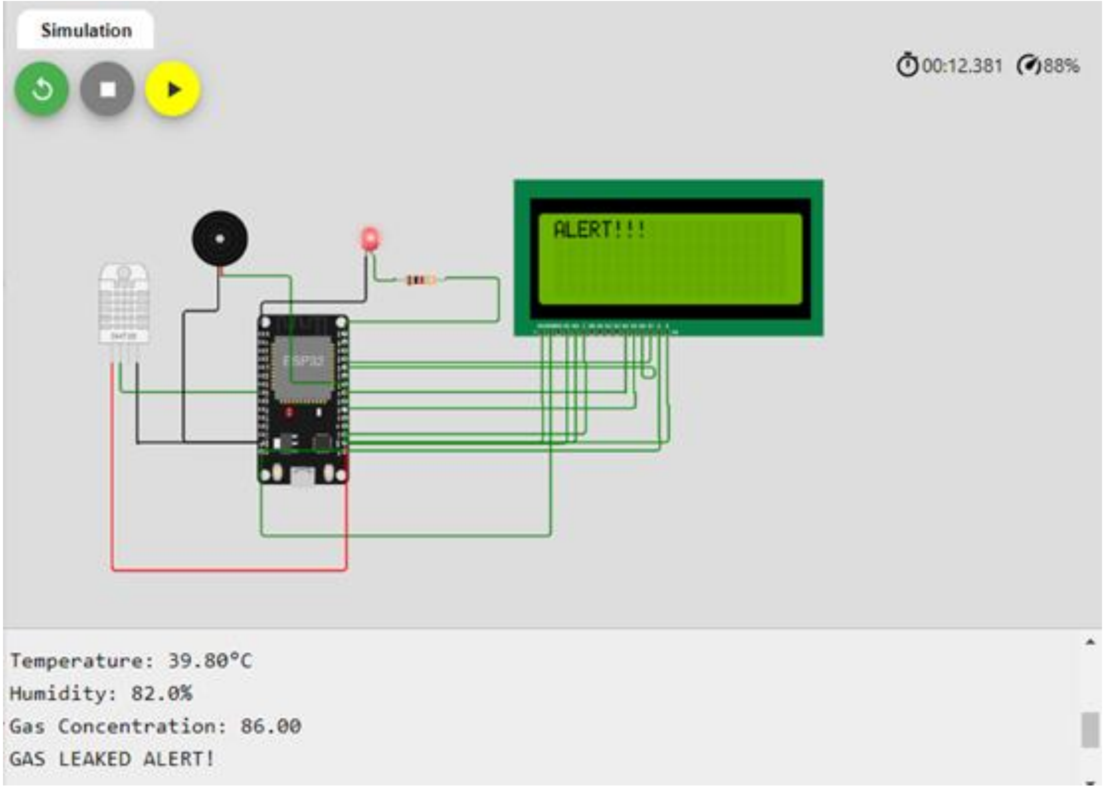
CIRCUIT DIAGRAM:



OUTPUTS:

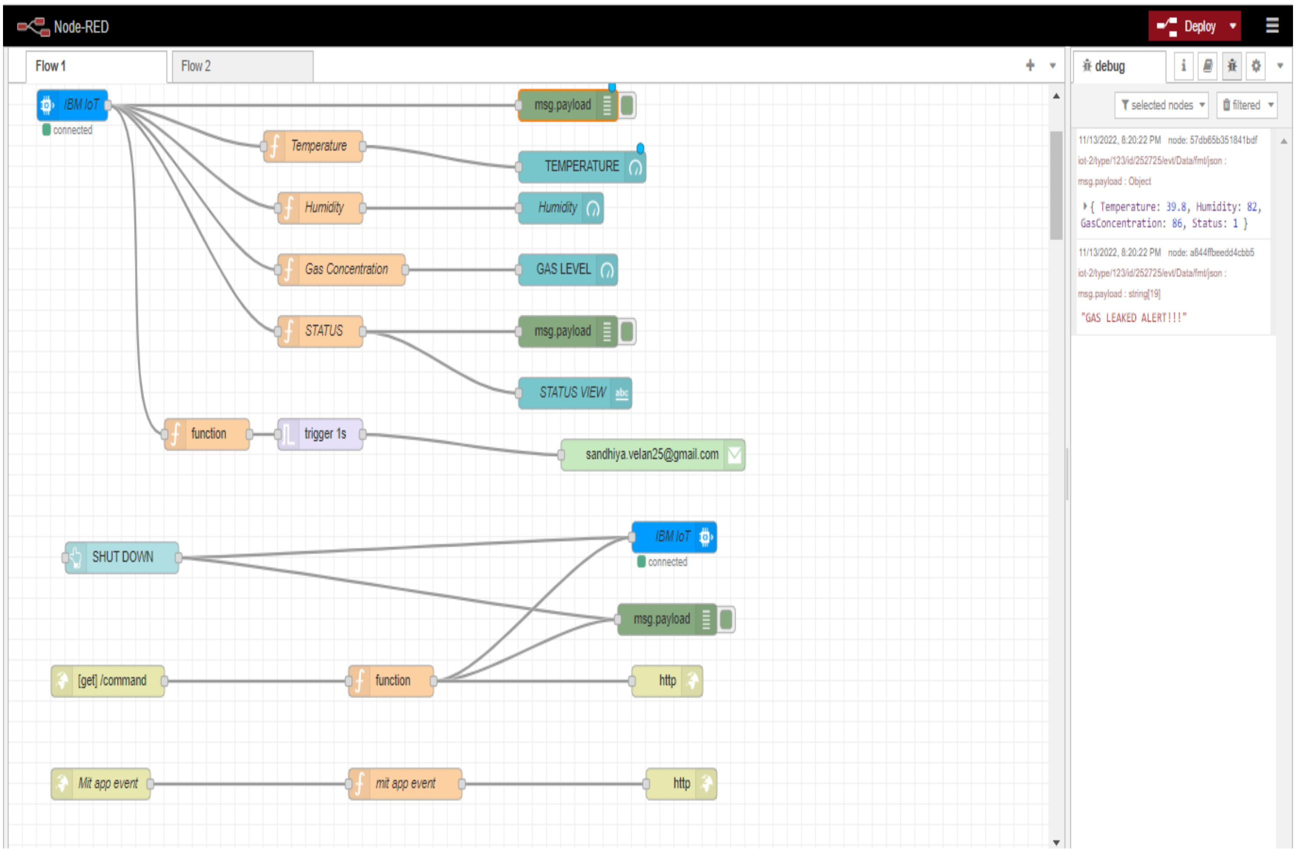
(i)When gas leakage occurs:

WOKWI OUTPUT:

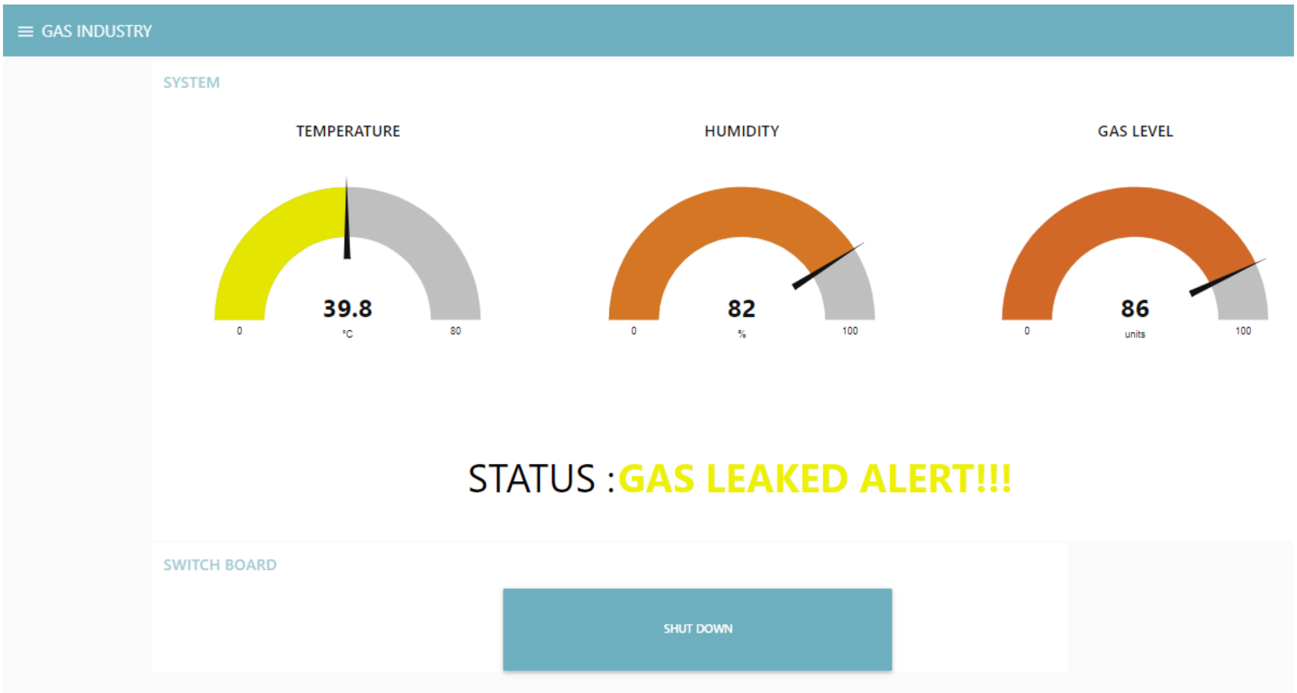


```
Temperature: 39.80°C
Humidity: 82.0%
Gas Concentration: 86.00
GAS LEAKED ALERT!
```

NODE RED OUTPUT:

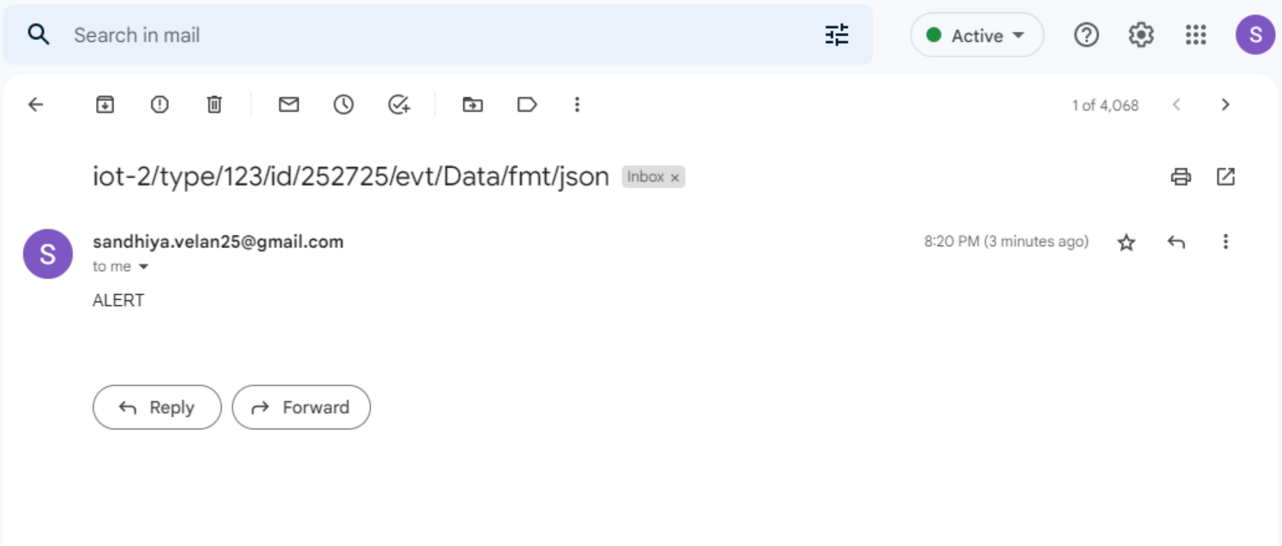


```
11/13/2022, 8:20:22 PM   node: 57db65b351841bdf
iot-2/type/123/id/252725/evt/Data/fmt/json :
msg.payload : Object
▸{ Temperature: 39.8, Humidity: 82,
GasConcentration: 86, Status: 1 }

11/13/2022, 8:20:22 PM   node: a844ffbeedd4cbb5
iot-2/type/123/id/252725/evt/Data/fmt/json :
msg.payload : string[19]
"GAS LEAKED ALERT!!!"
```



STATUS :GAS LEAKED ALERT!!!

NOTIFICATION:



iot-2/type/123/id/252725/evt/Data/fmt/json

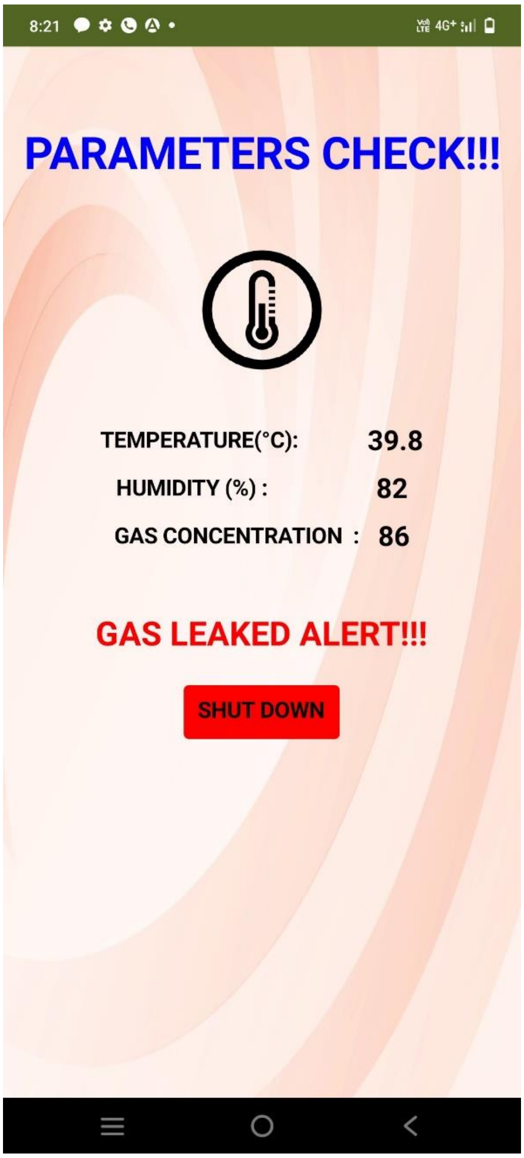
sandhiya.velan25@gmail.com

to me

8:20 PM (3 minutes ago)

ALERT

MITAPP OUTPUT:



PARAMETERS CHECK!!!

TEMPERATURE(°C): 39.8

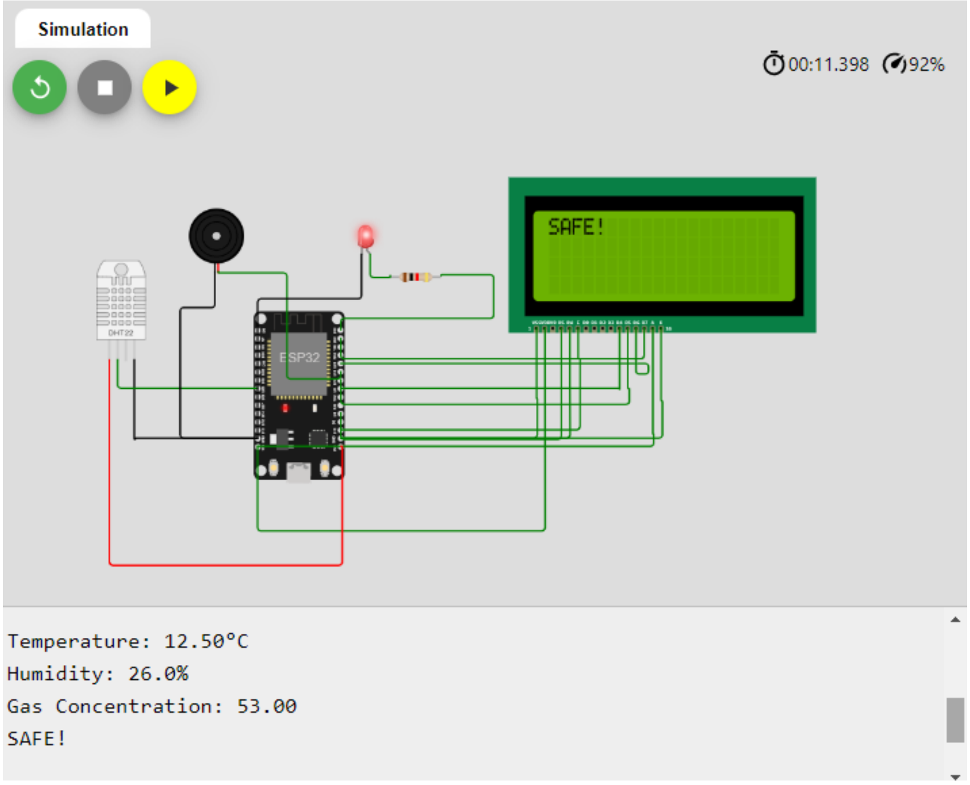
HUMIDITY (%) : 82

GAS CONCENTRATION : 86

GAS LEAKED ALERT!!!

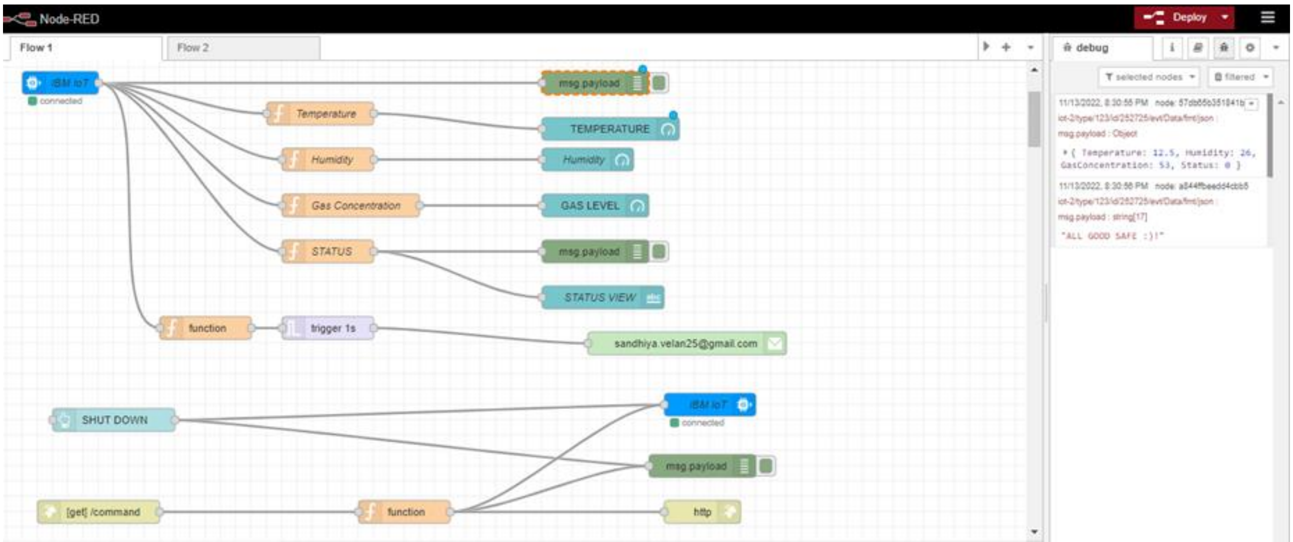
(ii)When nothing happens:

WOKWI OUTPUT:

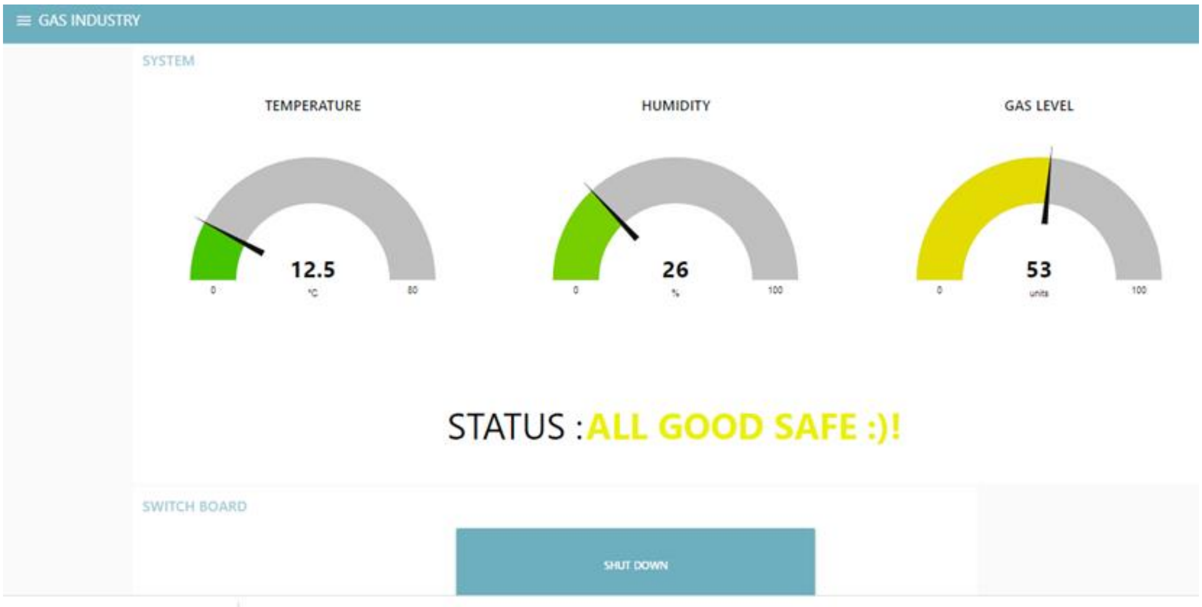


```
Temperature: 12.50°C
Humidity: 26.0%
Gas Concentration: 53.00
SAFE!
```

NODE RED OUTPUT:



```
11/13/2022, 8:30:56 PM  node: 57db66b351841b
iot-2/type/123/id/252725/evt/Data/fmt/json :
msg.payload : Object
▸{ Temperature: 12.5, Humidity: 26,
GasConcentration: 53, Status: 0 }
11/13/2022, 8:30:56 PM  node: a844fbeedd4cbb5
iot-2/type/123/id/252725/evt/Data/fmt/json :
msg.payload : string[17]
"ALL GOOD SAFE :)!"
```



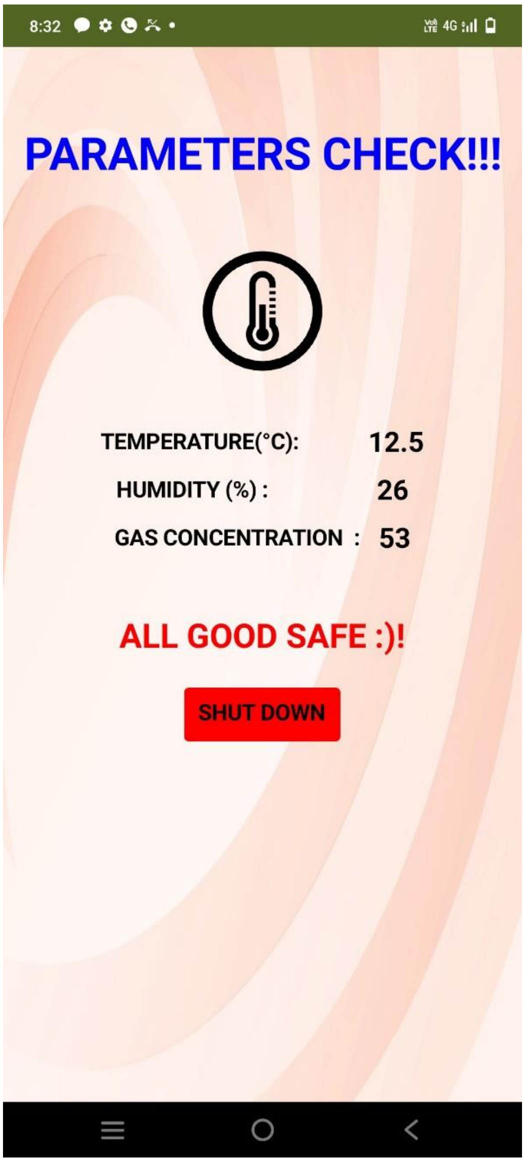
TEMPERATURE 12.5 °C

HUMIDITY 26 %

GAS LEVEL 53 units

STATUS : ALL GOOD SAFE :)!

MITAPP OUTPUT:



PARAMETERS CHECK!!!

TEMPERATURE(°C): 12.5

HUMIDITY (%) : 26

GAS CONCENTRATION : 53

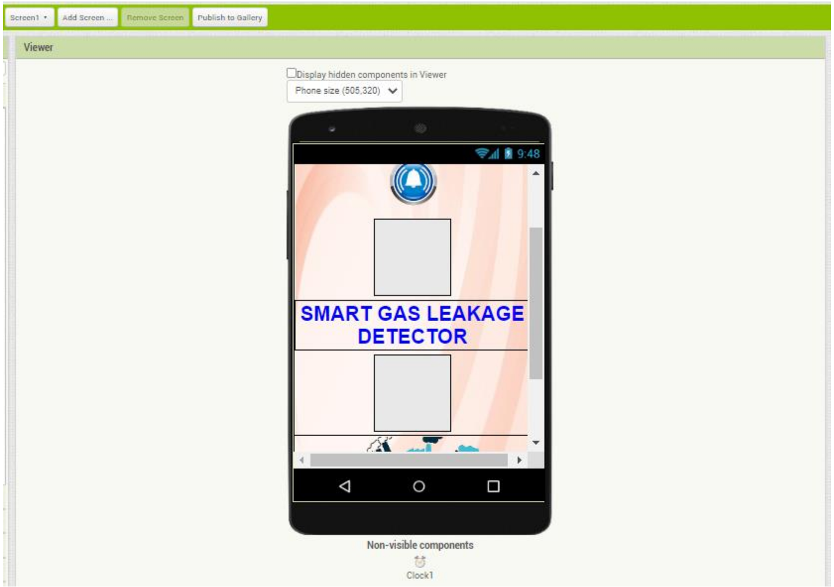
ALL GOOD SAFE :)!

LINK:

https://wokwi.com/projects/347920709920490066

MTAPP-FRONTEND:

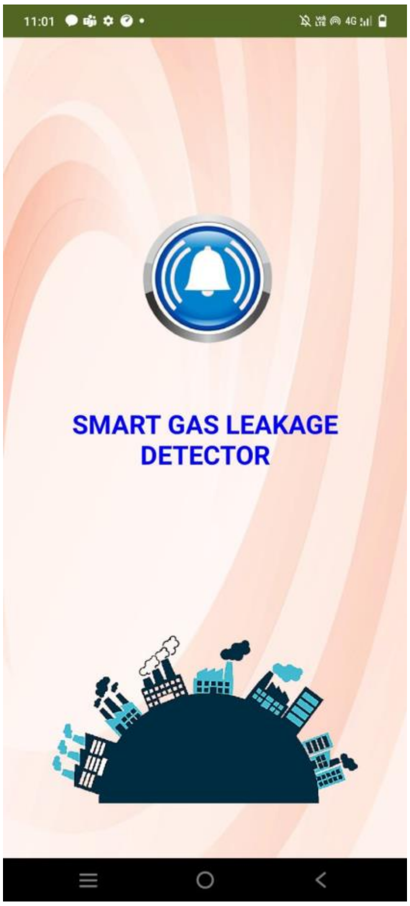
SCREEN-1:



□ Display hidden components in Viewer

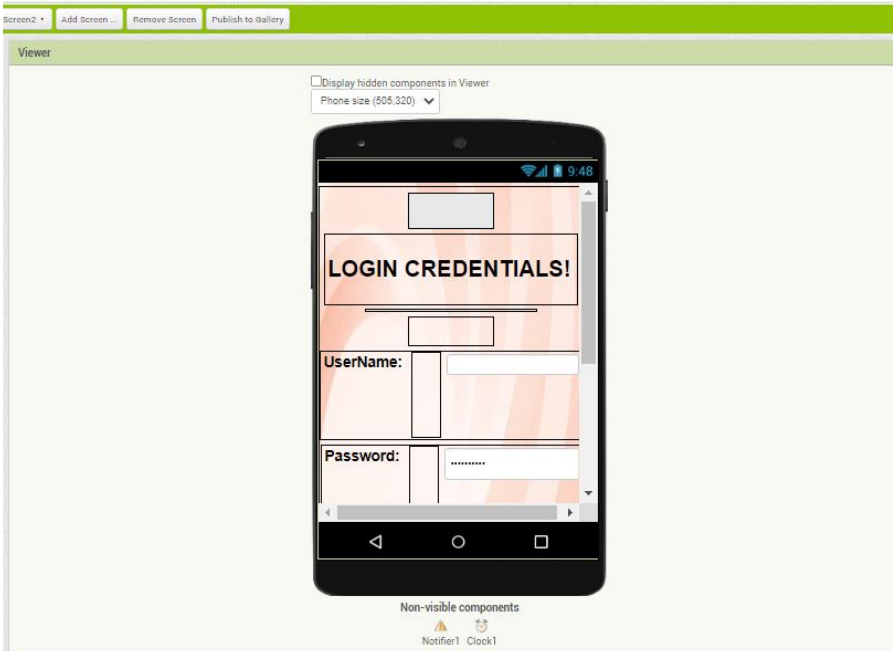
Phone size (505,320)

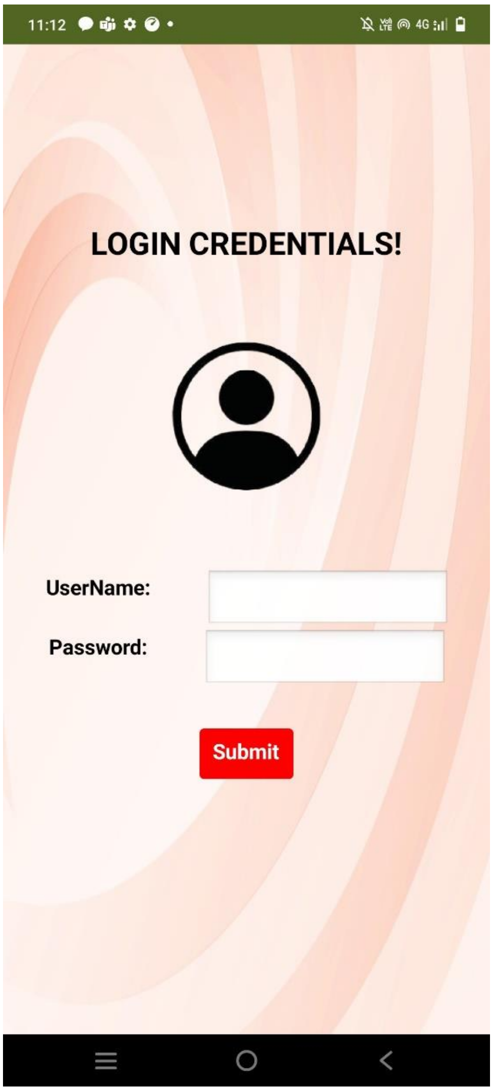
SMART GAS LEAKAGE DETECTOR



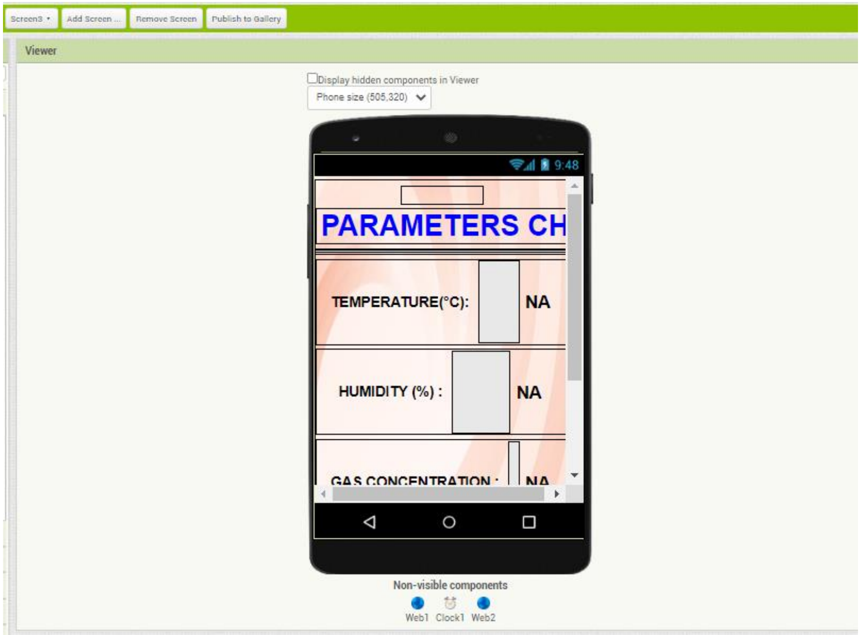
SMART GAS LEAKAGE DETECTOR

SCREEN-2:


Screen2 ▾
Add Screen ...
Remove Screen
Publish to Gallery
Viewer
□Display hidden components in Viewer
Phone size (505,320)
9:48
LOGIN CREDENTIALS!
UserName:
Password:
Non-visible components
Notifier1
Clock1


11:12
LOGIN CREDENTIALS!
UserName:
Password:
Submit

SCREEN-3:

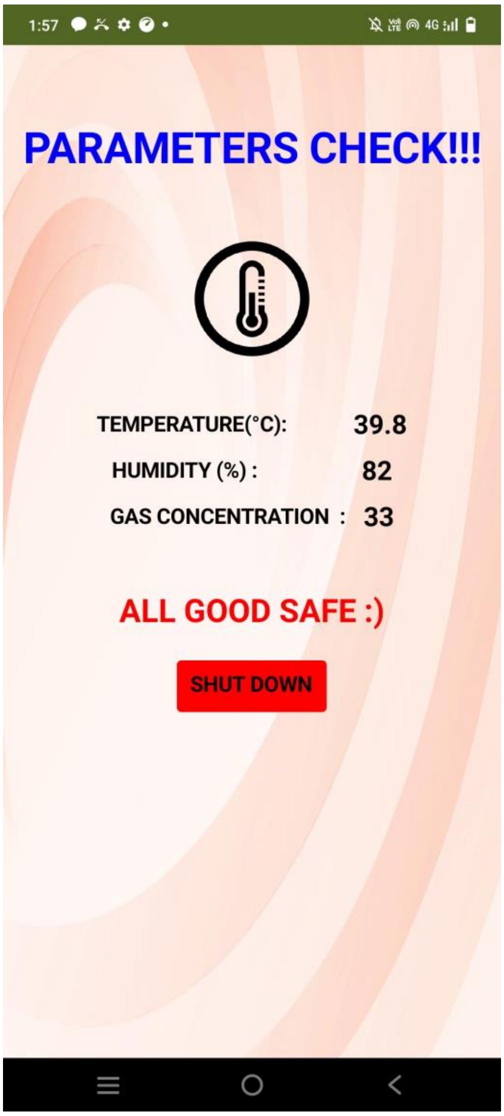


PARAMETERS CH

TEMPERATURE(°C): NA

HUMIDITY (%) : NA

GAS CONCENTRATION : NA



PARAMETERS CHECK!!!

TEMPERATURE(°C): 39.8

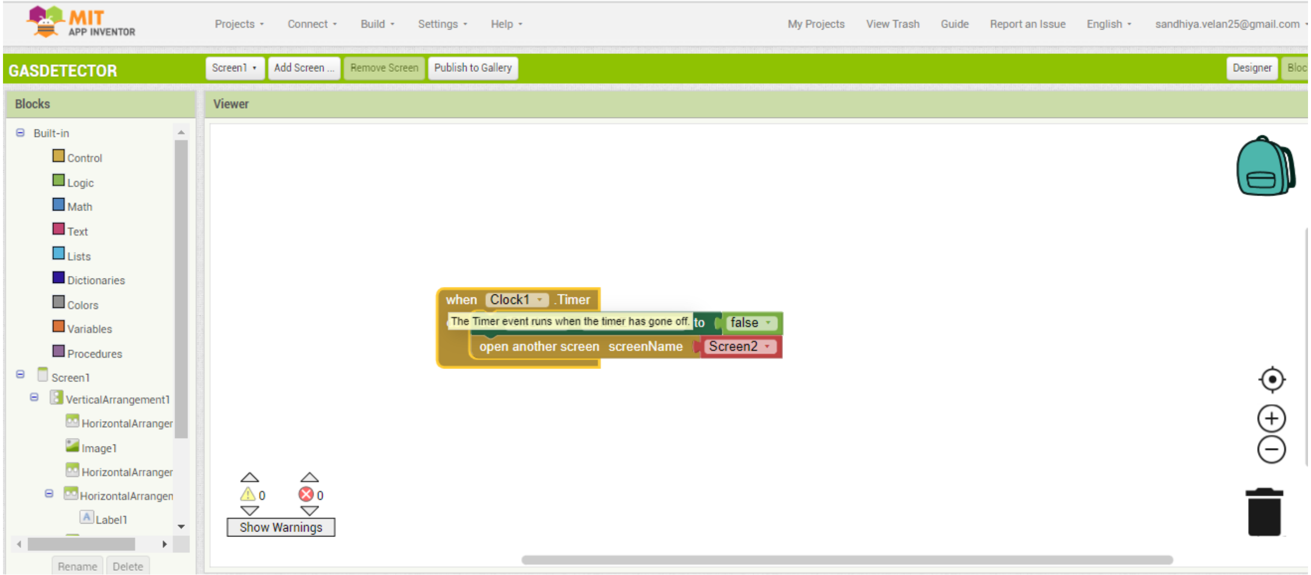
HUMIDITY (%) : 82

GAS CONCENTRATION : 33

ALL GOOD SAFE :)

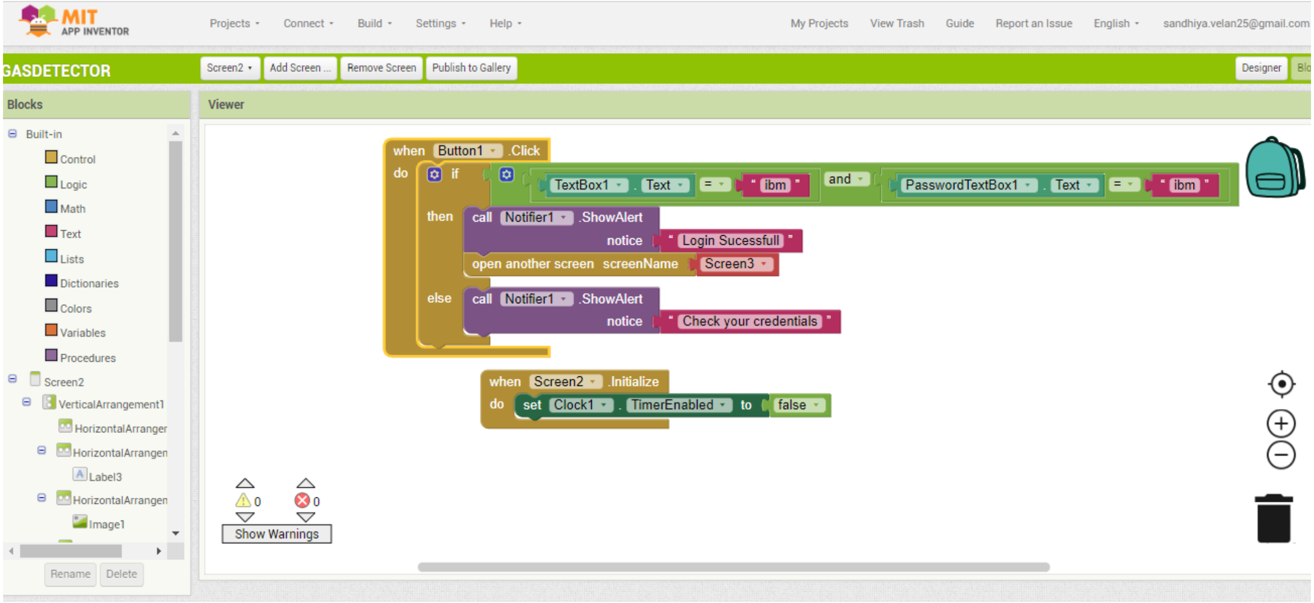
MTAPP-BACKEND:

SCREEN-1:



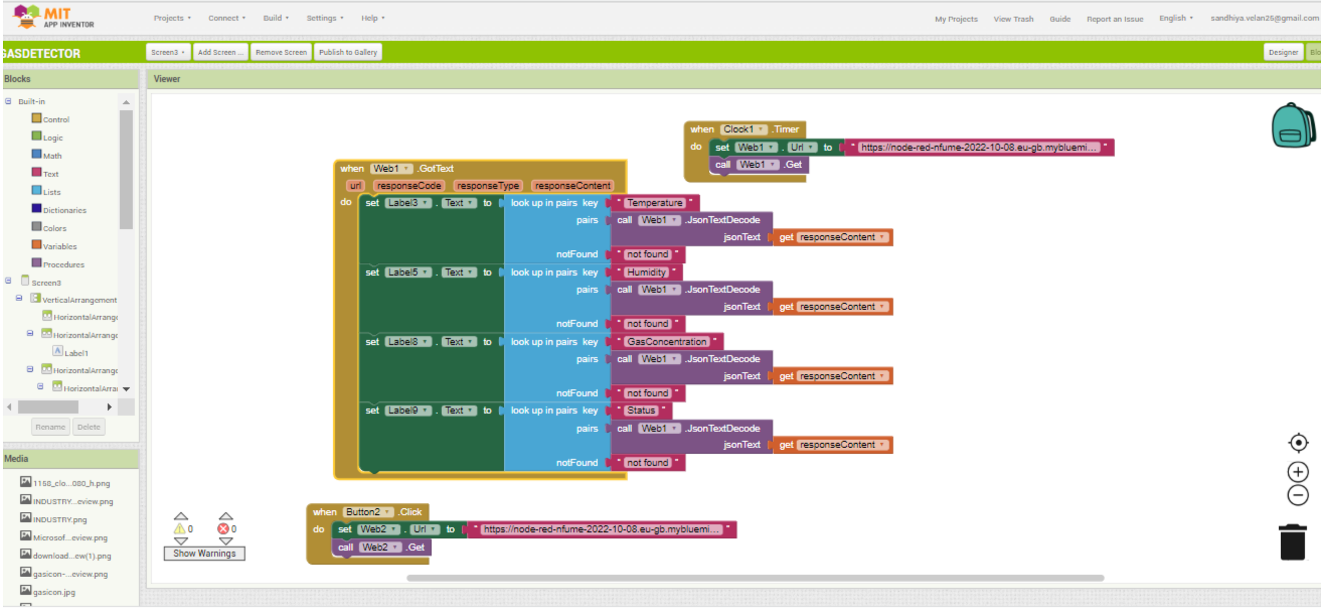
```
when Clock1 .Timer
  The Timer event runs when the timer has gone off. to false
  open another screen screenName "Screen2"
```

SCREEN-2:



```
when Button1 .Click
do  if  TextBox1 . Text = "ibm" and PasswordTextBox1 . Text = "ibm"
    then  call Notifier1 .ShowAlert
              notice "Login Sucessfull"
          open another screen screenName "Screen3"
    else  call Notifier1 .ShowAlert
              notice "Check your credentials"

when Screen2 .Initialize
do  set Clock1 . TimerEnabled to false
```

SCREEN-3:



```
when Clock1 .Timer
do  set Web1 . Url to " https://node-red-nfume-2022-10-08.eu-gb.mybluemi... "
    call Web1 .Get

when Web1 .GotText
  url  responseCode  responseType  responseContent
do  set Label3 . Text to  look up in pairs  key  " Temperature "
                                            pairs  call Web1 .JsonTextDecode
                                                              jsonText  get responseContent
                                            notFound  " not found "
    set Label5 . Text to  look up in pairs  key  " Humidity "
                                            pairs  call Web1 .JsonTextDecode
                                                              jsonText  get responseContent
                                            notFound  " not found "
    set Label8 . Text to  look up in pairs  key  " GasConcentration "
                                            pairs  call Web1 .JsonTextDecode
                                                              jsonText  get responseContent
                                            notFound  " not found "
    set Label9 . Text to  look up in pairs  key  " Status "
                                            pairs  call Web1 .JsonTextDecode
                                                              jsonText  get responseContent
                                            notFound  " not found "

when Button2 .Click
do  set Web2 . Url to " https://node-red-nfume-2022-10-08.eu-gb.mybluemi... "
    call Web2 .Get
```