Project Design Phase-II
Data Flow Diagram & User Stories

Date	30 October 2022
Team ID	PNT2022TMID04334
Project Name	Gas leakage monitoring and alerting system for industries
Maximum Marks	2 Marks

Data Flow Diagrams:

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events. Upload document with wokwi share link and images of IBM cloud

Code:

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "oyi7sh"
#define DEVICE_TYPE "demo1"
#define DEVICE_ID "56789"
#define TOKEN "*7NByCMv-2-eQTpxPj"
String data3;
float dist;
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);
int LED = 4;
int trig = 5;
int echo = 18;
void setup()
{
  Serial.begin(115200);
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(LED, OUTPUT);
  delay(10);
  wificonnect();
  mqttconnect();
}
void loop()
{
  digitalWrite(trig, LOW);
  digitalWrite(trig,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  float dur = pulseIn(echo, HIGH);
  float dist = (dur*0.0343)/2;
  Serial.print ("Distancein cm");
  Serial.println(dist);
  PublishData(dist);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}
void PublishData(float dist)
{
```

```
  mqttconnect();
  String object;
  if (dist <100)
  {
    digitalWrite(LED, HIGH);
    Serial.println("object is near");
    object ="Near";
  }
  else
  {
    digitalWrite(LED, LOW);
    Serial.println("no object found");
    object = "No";
  }
  String payload = "{\"distance\":";
  payload += dist;
  payload += "," "\"object\":\"";
  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 client 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));
```

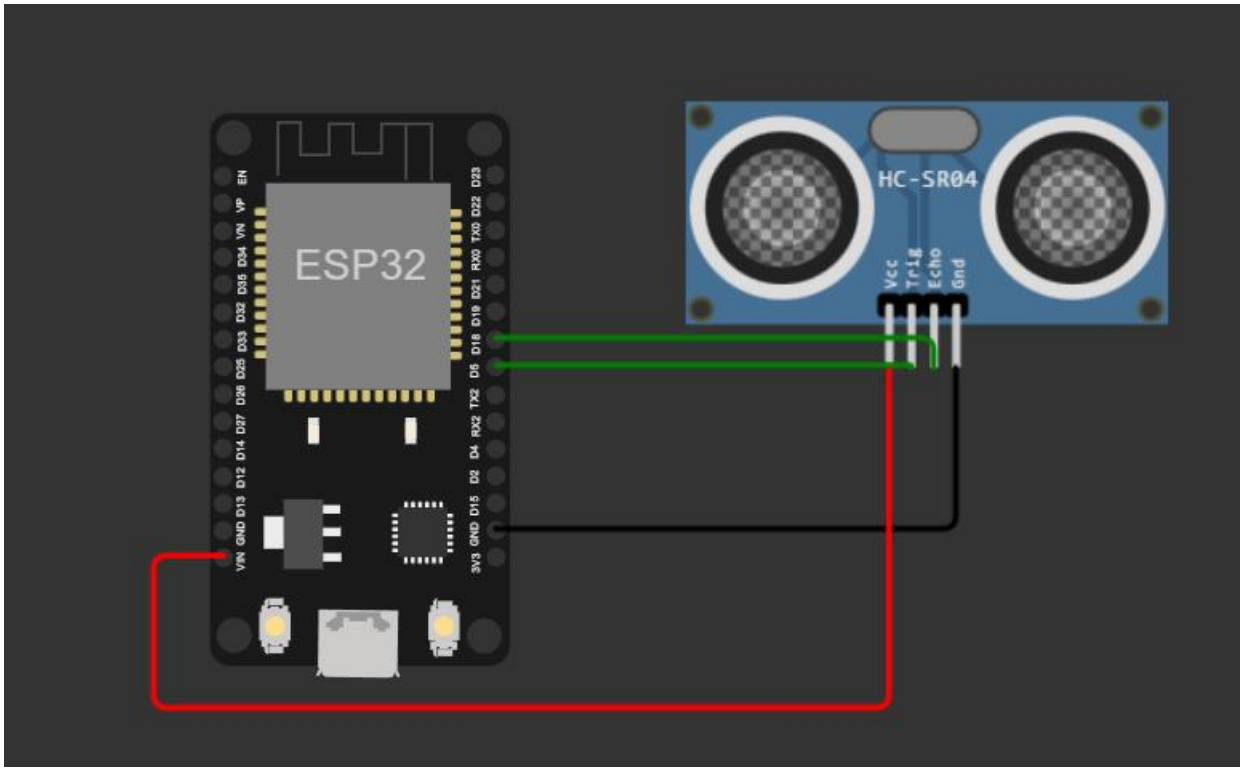
```
    Serial.println("subscribe to cmd OK");
  }
  else
  {
    Serial.println("subscribe to cmd FAILED");
  }
}
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++)
  {
    data3 += (char)payload[i];
  }
  data3="";
}
```

Diagram.json:

```
{
  "version": 1,
  "author": "HARISH M 19CSR055",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 0, "left": -90.67, "attrs": {} },
    { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -2.38, "left": 78.15, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
```

```
    [ "esp:GND.1", "ultrasonic1:GND", "black", [ "h0" ] ],
    [ "esp:VIN", "ultrasonic1:VCC", "red", [ "h-24.85", "v53.52", "h259.33" ] ],
    [ "ultrasonic1:ECHO", "esp:D18", "green", [ "v0" ] ],
    [ "ultrasonic1:TRIG", "esp:D5", "green", [ "v0" ] ]
  ]
}
```

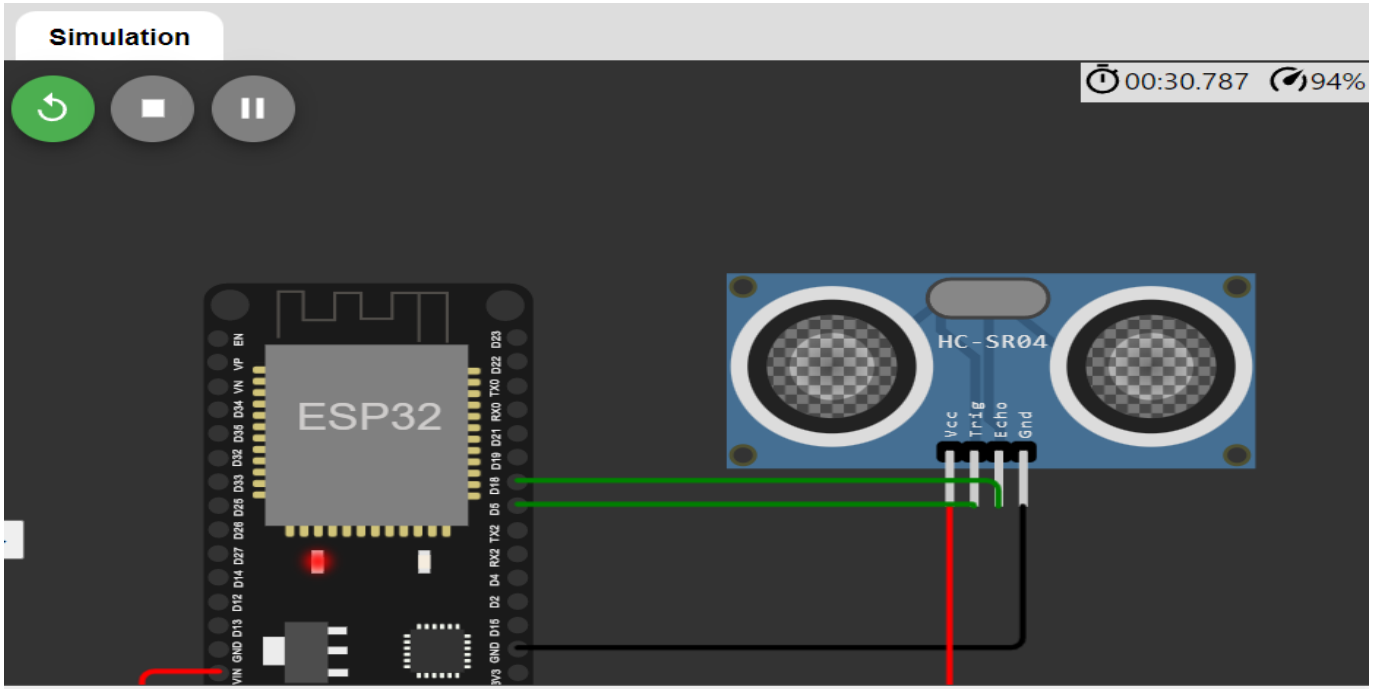
Circuit Diagram:



Link:

https://wokwi.com/projects/347145343196987988

Output:




```
object is near
Sending payload: {"distance":25.16,"object":""}
Publish ok
Distancein cm25.16
object is near
Sending payload: {"distance":25.16,"object":""}
Publish ok
```

IBM cloud output: