

Project Development - Delivery Of Sprint - 2

Date	14 November 2022
Team ID	PNT2022TMID39165
Project Name	IoT Based Safety Gadget for Child Safety Monitoring & Notification

NOTIFICATION:

This coding will make connection between IoT Device & Parent's application. When the child cross across the geofence message will be notified on parent's application.

CODING:

```
#include<WiFi.h>//library      for      wifi
#include<PubSubClient.h>//library      for
MQTT
void callback(char* subscribe topic, byte* payload, unsigned int payload length);
//-----credentials of IBM Account-----
#define ORG "frpi8s"// IBM ORGANIZATION ID
#define DEVICE_TYPE "NodeMCU"//DEVICE TYPE MENTIONED IN IOT
WATSON
PLATFORM #define DEVICE_ID "12345"//DEVICE ID MENTIONED IN
IOTWATSONPLATEFORM
#define TOKEN "12345678"//Token String data3;floatdist;
//-----customize the above value-----
```

```

char server [] =ORG ".messaging.internetofthings.ibmcloud.com";//servername
char publish topic[]="ultrasonic/evt/Data/fmt/json";/*topic name and typeof
event perform and format in which data to be send*/
char subscribetopic[]="ultrasonic/cmd/test/fmt/String";/*cmd REPRESENT
Command tupe 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 an instance for wificlient
PubSubClient client(server, 1883 , callback , wifiClient);/*calling the predefined
client id by passing parameter like server id,portand wificredential*/
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); Serial.println(); 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("distance incm");
Serial.println(dist); PublishData(dist);
delay(1000);
if(!client.loop()){ mqttconnect();
}
}
/*.....retriving to cloud. ....

```

```

*/

void PublishData(float dist){ mqttconnect();//function call for connecting to ibm
/*creating the string in form of JSON to update the data to ibm cloud*/ Stringobject;
if(dist<100)
    {
        digitalWrite(LED,HIGH); Serial.println("no object is near");
object="Near";
    }
    else
    {
        digitalWrite(LED,LOW); Serial.println("no object found"); object="No";
    }
    String payload="{\"distance\":"; payload +=dist;
    payload +=",\" \"object\":\":"; payload += object;
    payload += "\"}";
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if(client.publish(publishtopic, (char*) payload.c_str())){
        Serial.println("Publish ok");/* if its 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");
    }
}

void
mqttconnect(){ if(!client.connected()){ Serial.print("Reconnectin
g client to ");
Serial.println(server);while(!!!client.connect(clientid,authMethod,

```

```

token)){ Serial.print("."); delay(500);
}
initManagedDevice();
Serial.println();
}
}
void wificonnect()//function defenition for wificonnect
{
    Serial.println();
    Serial.print("Connecting to ");
    WiFi.begin("vivo 1816", "taetae95",6);//PASSING THE WIFI
    CREDIDENTIALS TOESTABLISH 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(subscribeto
    pic)){Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
    }else
    { Serial.println("subscribe to cmd failed");
    }
}

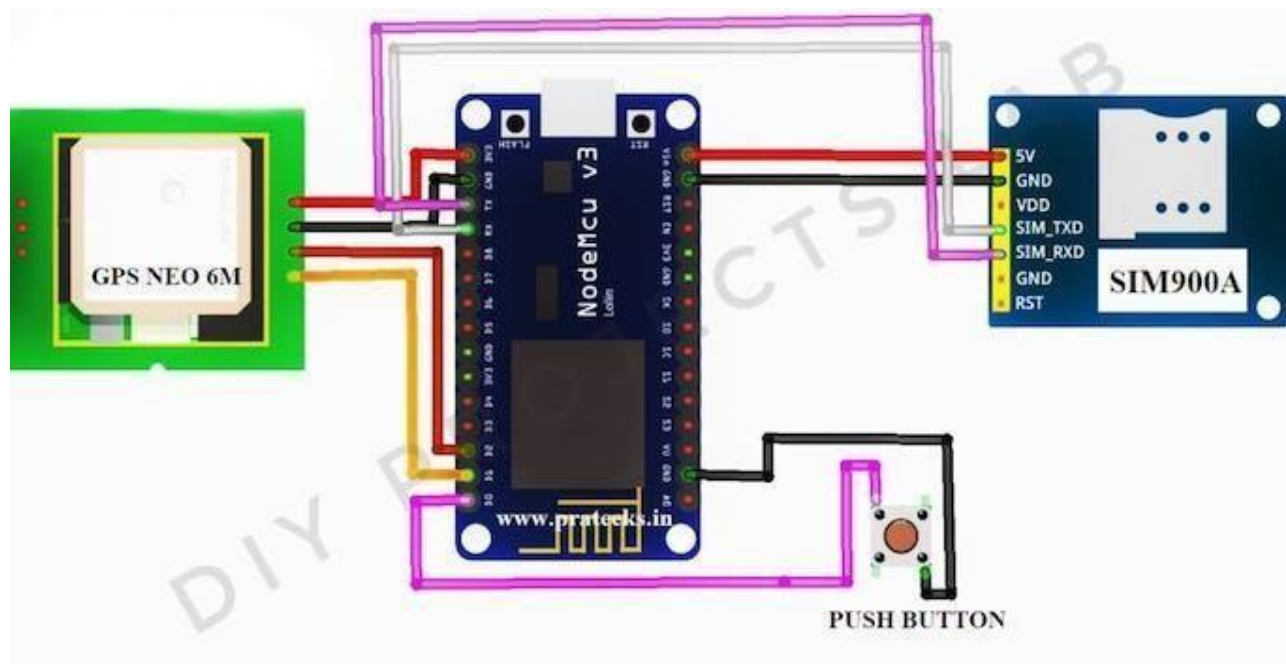
```

```

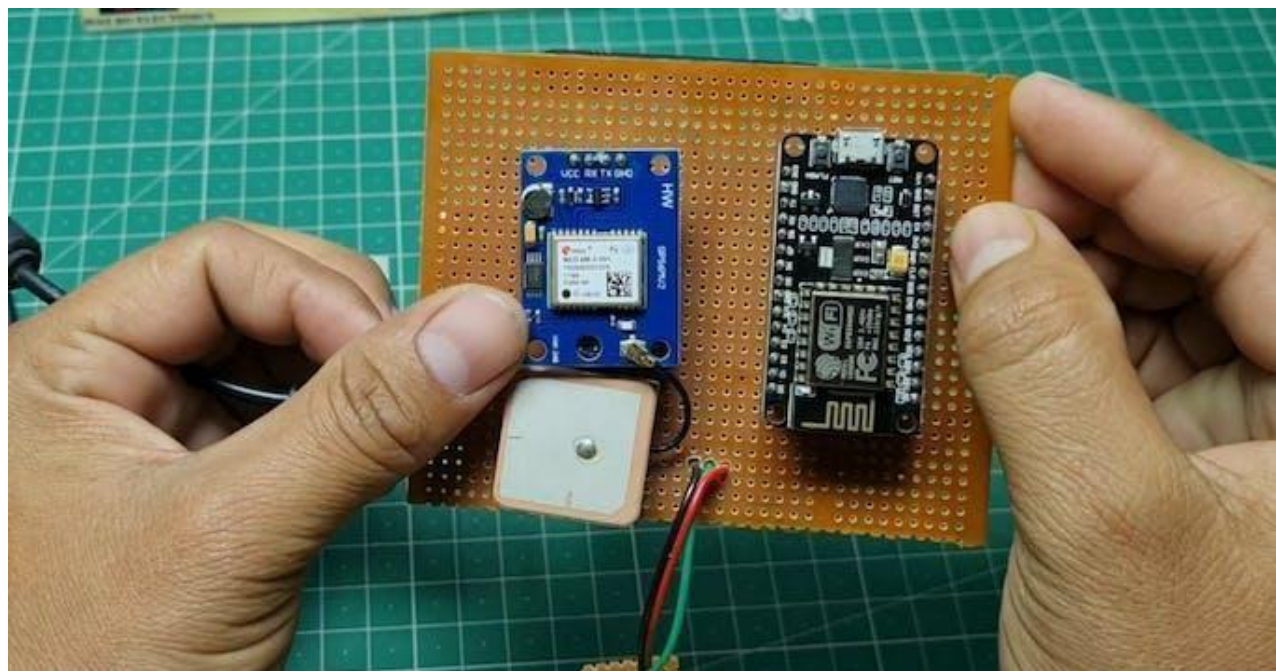
    }
void callback(char* subscribetopic,byte*payload,unsigned int payloadLength)
{
    Serial.print("callback invoked for topic: ");
    Serial.println(subscribetopic); for(int i=0; i< payloadLength; i++){
        //Serial.print((char)payload[i]); data3 +=(char)payload[i];
    }
    //Serial.println("dta: "+ data3);
    //if(data3=="Near")
    //{
    //Serial.println(data3);
    //digitalWrite(LED,HIGH);
    //}
    //else //{
    //Serial.println(data3);
    //digitalWrite(LED,LOW);//} data3="";
}

```

SCHEMATIC DIAGRAM:



OUTPUT:



NOTIFY TO THIS DEVICE IBM WATSON CLOUD COMMUNICATION:

← Back

Device Drilldown - 12345

Device Credentials

Connection Information

Recent Events

State

Device Information

Metadata

Diagnostics

Connection Logs

Device Actions

Device Credentials

You registered your device to the organization. Add these credentials to the device to connect it to the platform. After the device is connected, you can navigate to view connection and event details.

Organization ID	frpi8s
Device Type	NodeMCU
Device ID	12345
Authentication Method	use-token-auth
Authentication Token	12345678

!

Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the device to generate a new authentication token.

Find out how to add these credentials to your device ↗

Browse Action Device Types Interfaces
Add Device +

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

Event	Value	Format	Last Received
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":11.03}	json	a minute ago