SOURCE CODE

```
1. #define BLYNK_TEMPLATE_ID "TMPL6bDzj4qi4"
2. #define BLYNK_TEMPLATE_NAME "KANDANG"
3. #define BLYNK_AUTH_TOKEN "JsAjzn_pKmfnKls3Sag9kbVfGJO6dT-F"
4. #define BLYNK_PRINT Serial
5. #include <WiFi.h>
6. #include <WiFiClient.h>
7. #include <BlynkSimpleEsp32.h>
8. #include <Ultrasonic.h>
9. #include <DHT.h>
10.     char auth[] = BLYNK_AUTH_TOKEN;
11.     char ssid[] = "realme"; // Nama Wifi
12.     char pass[] = "12345678"; // Password Wifi
13.     BlynkTimer timer;
14.     #define TRIGGER_PIN 17 // PIN 17 trig terhubung
15.     #define ECHO_PIN 16 // PIN 16 echo terhubung
16.     #define buzzerPin 5  // PIN Buzzer
17.     #define GAS_SENSOR_PIN 33 // Pin sensor gas terhubung
18.     #define THRESHOLD 1500 // Ambang batas rendah deteksi
   gas metana
19.     #define DHTPIN 4 // Pin yang terhubung ke sensor DHT11
20.     #define RELAYPIN 25 // Pin yang terhubung ke relay
21.     #define DHTTYPE DHT11 // Tipe sensor DHT (DHT11, DHT21,
   DHT22)
22.     DHT dht (DHTPIN, DHTTYPE); // Inisialisasi objek sensor
   dht11
23.     Ultrasonic  ultrasonic(TRIGGER_PIN,  ECHO_PIN);     //
   Inisialisasi objek sensor ultrasonik
24.     void notifyOnGas() {
25.     }
26.     void setup() {
27.     Serial.begin(115200);
28.     Blynk.begin(auth, ssid, pass);
29.     dht.begin();
30.     pinMode(RELAYPIN, OUTPUT);
31.     pinMode(GAS_SENSOR_PIN, INPUT);
32.     pinMode(buzzerPin, OUTPUT);
33.     timer.setInterval(1000L, getMethaneLevel);
```

```
34.     }
35.     void loop()
36.     {
37.     delay(1000);  // Jeda 1 detik antara pembacaan
38.     float humidity = dht.readHumidity(); // Baca
   kelembapan
39.     float temperature = dht.readTemperature(); // Baca
   suhu dalam Celsius
40.     float methaneLevel = getMethaneLevel(); // Baca Gas
   Metana
41.     long distance = ultrasonic.distanceRead();
42.     Blynk.run();
43.     timer.run();
44.     // Kirim data ke Blynk
45.     Blynk.virtualWrite(V0, temperature);
46.     Blynk.virtualWrite(V1, humidity);
47.     Blynk.virtualWrite(V2, methaneLevel);
48.     Blynk.virtualWrite(V3, distance);
49.     if (temperature < 32) {
50.     digitalWrite(RELAYPIN, LOW);
51.     } else {
52.     digitalWrite(RELAYPIN, HIGH);
53.     }
54.     if (humidity < 60) {
55.     Serial.println("Kandang Normal");
56.     } else {
57.     Serial.println("Kandang Lembab, BERSIHKAN KANDANG
   SEKARANG!");
58.     Blynk.logEvent("kelembapan_alert","KANDANG LEMBAB,
   BERSIHKAN KANDANG SEKARANG !!!"); // Kirim notifikasi ke
   aplikasi Blynk
59.     }
60.     Serial.print("Kelembapan: "); // Gas Suhu Kelembapan
61.     Serial.print(humidity);
62.     Serial.print("%\t");
63.     Serial.print("Suhu: ");
64.     Serial.print(temperature);
65.     Serial.println("°C");
66.     Serial.print("Methane Level: "); // Gas Metana Level
```

```
67.     Serial.println(methaneLevel);
68.     if (methaneLevel < 1500) {
69.     Serial.println("Gas Normal");
70.       }
71.     else {
72.     Serial.println("Gas metana terdeteksi berbahaya,
   BERSIHKAN KANDANG SEKARANG!");
73.     Blynk.logEvent("metana_alert","Gas metana terdeteksi
   berbahaya, BERSIHKAN KANDANG SEKARANG!"); // Kirim
   notifikasi ke aplikasi Blynk
74.       }
75.       If (distance < 10) {
76.       digitalWrite(buzzerPin, LOW); // Buzzer aktif jika
   jarak di bawah 10cm
77.     } else {
78.     digitalWrite(buzzerPin, HIGH); // Buzzer non-aktif
   jika jarak di atas atau sama dengan 10cm
79.       }
80.     // Menampilkan jarak pada Serial Monitor
81.     Serial.print("Jarak: ");
82.     Serial.print(distance);
83.     Serial.println(" cm");
84.     }
85.     float getMethaneLevel()
86.     {
87.     int sensorValue = analogRead(GAS_SENSOR_PIN);
88.     float voltage = sensorValue * (3.3 / 4095.0); //
   Konversi nilai sensor menjadi tegangan
89.     float methaneConcentration = voltage * 3030;   //
   Konversi tegangan menjadi konsentrasi gas (ppm)
90.     return methaneConcentration;
91.     }
```