

## JAWABAN KNOWLEDGE TEST

1. Platform yang sering digunakan: Thingspeak
2. Alasan: gratis, mudah, tempat/field untuk sensor cukup banyak, bisa dihubungkan dengan aplikasi HP dan website, dan bisa realtime.
3. Project yang pernah dikerjakan:

No.	Nama Project	
1	Prototype Smarthome with MIT APP Inventor	Mengkode Arduino
		Membuat aplikasi dengan MIT App Inventor
		Membuat rangkaian elektronis untuk Smarthome
2	Smarthome Implementation with Laravel	Mengkode Arduino dan ESP
		Membuat website berbasis Laravel yang diintegrasikan dengan hardware
		<a href="https://jurnal.ugm.ac.id/juliet/article/view/53820">https://jurnal.ugm.ac.id/juliet/article/view/53820</a>
3	Design of Control System for Smartfarm Using MIT APP Inventor	Mendesain purwa rupa menggunakan 3D Inventor
		Mendesain rangkaian elektronis menggunakan Eagle
		Mengkode Arduino dan ESP
		Membuat interface menggunakan MIT App Inventor

4. Pemahaman mengenai:
  - a. IoT atau Internet of Things adalah dimana perangkat (hardware) dapat terhubung melalui internet dan dapat diakses dari jarak jauh
  - b. Mikrokontroler adalah papan pengendali kecil (computer kecil) yang berbentuk chip dan dapat digunakan untuk berbagai hal.
  - c. Bluetooth Low Energy adalah Bluetooth dengan teknologi tambahan yaitu penghematan energy
  - d. MQTT adalah titik untuk mengirimkan data antar perangkat dengan metode publish dan subscribe
  - e. Mod bus adalah prosedur pengiriman data antar perangkat pada jaringan yang sama secara serial
  - f. CAN bus
  - g. OBDII

- h. Realtime operating sistem adalah sistem yang bekerja secara real time atau diwaktu yang sama (bisa paralel)
  - i. Single board computer adalah computer mini yang didalamnya sudah terdapat sistem operasi(OS)
  - j. Armbian adalah OS berbasis Linux
  - k. UV4L Streaming
  - l. webRTC Streaming adalah aplikasi untuk komunikasi secara realtime menggunakan website
5. Program
- a. Modul GPS dengan I2C, Modul cellular dengan UART, Data ADC (100 dps) dengan RTOS
6. Mengirim data Vsumber, Suhu ESP, dan sensor suhu secara realtime dari ESP ke MQTT
- a. Untuk data suhu internal esp, saya buat terpisah menggunakan ESP32, karena saat mencoba menggunakan ESP8266(ESP yang sama miliki) tidak dapat berjalan

```
#include <WiFi.h>
#include <PubSubClient.h>

char* ssid = "HOMY";
const char* password = "Cats2021";
const char* mqttServer = "192.168.1.10";
const int mqttPort = 1800;
const char* mqttUser = "";
const char* mqttPassword = "";

WiFiClient espClient;
PubSubClient client(espClient);

void callback(char* topic, byte* payload, unsigned int length) {
    Serial.print("Message arrived in topic: ");
    Serial.println(topic);
    Serial.print("Message:");
    for (int i = 0; i < length; i++) {
        Serial.print((char)payload[i]);
    }
}
```

```

#ifdef __cplusplus
extern "C" {
#endif
uint8_t temprature_sens_read();
#ifdef __cplusplus }
#endif
uint8_t temprature_sens_read();

void setup() {
    Serial.begin(115200);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.println("Connecting to WiFi..");
    }
    Serial.println("Connected to the WiFi network");
    client.setServer(mqttServer, mqttPort);
    client.setCallback(callback);
    while (!client.connected()) {
        Serial.println("Connecting to MQTT...");
        if (client.connect("ESP32Client", mqttUser, mqttPassword)) {
            Serial.println("connected");
        } else {
            Serial.print("failed with state ");
            Serial.print(client.state());
            delay(2000);
        }
    }
}

void loop() {
    float suhu_in=(temprature_sens_read() - 32) / 1.8;
    Serial.println(suhu_in);
    Serial.println(" C");
    client.subscribe("suhu_in", suhu_in);
    client.loop();
    delay(10);
}

```

b. Program Vsumber dan sensor suhu realtime

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 5

const char* ssid = "HOMY";
const char* password = "Cats2021";
const char* mqttServer = "192.168.1.10";
const int mqttPort = 1883;
const char* mqttUser = "";
const char* mqttPassword = "";

WiFiClient espClient;
PubSubClient client(espClient);

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
char msg_out2[20];

ADC_MODE(ADC_VCC);
char msg_out[20];

void setup() {
    Serial.begin(115200);
    sensors.begin();
    pinMode(5, INPUT);

    WiFi.begin(ssid, password);
```

```
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("Connecting to WiFi..");
}
Serial.println("Connected to the WiFi network");

client.setServer(mqttServer, mqttPort);
client.setCallback(callback);

while (!client.connected()) {
    Serial.println("Connecting to MQTT...");
    if (client.connect("ESP8266Client", mqttUser, mqttPassword ))
    {
        Serial.println("connected");
    } else {
        Serial.print("failed with state ");
        Serial.print(client.state());
        delay(2000);
    }
}

void callback(char* topic, byte* payload, unsigned int length) {
    Serial.print("Message arrived in topic: ");
    Serial.println(topic);
    Serial.print("Message:");
    for (int i = 0; i < length; i++) {
        Serial.print((char)payload[i]);
    }
}
```

```
void loop() {  
    float vdd = ESP.getVcc() / 1024.0;  
    Serial.print(vdd);  
    dtostrf(vdd, 2, 2, msg_out);  
    client.publish("vin", msg_out);  
    client.loop();  
    delay (1000);  
  
    sensors.requestTemperatures();  
    Serial.print(sensors.getTempCByIndex(0));  
    dtostrf(sensors.getTempCByIndex(0), 2, 2, msg_out2);  
    client.publish("temp", msg_out2);  
    client.loop();  
    delay (1000);  
}
```

## 7. Urutan proses streaming video dengan SBC