# LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya



**PRAKTIKUM SENSOR SISTEM KELEMBAHAN SUHU**

**Author :**

**Ivan AryaPutra Rachmadhani / 233140700111109**

**Fakultas Vokasi, Universitas Brawijaya**

**Email : ivanarya990@gmail.com**

**Abstract** (Abstrak)

Internet of Things (IoT) telah menjadi teknologi yang banyak diterapkan dalam berbagai bidang, termasuk pemantauan lingkungan. Salah satu aplikasi penting IoT adalah sistem pemantauan suhu yang dapat digunakan untuk mendeteksi perubahan suhu secara real-time dan memberikan respons otomatis jika suhu melebihi ambang batas tertentu. Pada praktikum ini, dirancang dan diuji sistem sensor suhu berbasis IoT yang menggunakan sensor suhu DHT22 yang terhubung dengan mikrokontroler ESP8266. Data suhu yang diperoleh dikirim ke platform IoT untuk dipantau secara online melalui antarmuka web atau aplikasi seluler. Jika suhu melebihi ambang batas yang telah ditentukan, sistem akan mengaktifkan peringatan atau tindakan otomatis, seperti menyalakan kipas atau mengirim notifikasi. Hasil pengujian menunjukkan bahwa sistem ini dapat memantau dan merespons perubahan suhu secara efisien dengan tingkat akurasi yang baik. Implementasi sistem ini dapat diterapkan pada berbagai bidang, seperti pemantauan suhu ruangan, pertanian pintar, dan sistem pendingin otomatis.

**1. Introduction** (Pendahuluan)

**1.1 Latar belakang**

Perkembangan teknologi di bidang Internet of Things (IoT) telah memberikan dampak yang signifikan dalam berbagai sektor, salah satunya adalah pemantauan lingkungan. IoT memungkinkan perangkat fisik untuk saling terhubung melalui jaringan internet dan melakukan pengambilan data secara real-time. Salah satu aplikasi IoT yang penting adalah pemantauan suhu, di mana sistem ini dapat mendeteksi perubahan suhu dalam suatu ruangan atau area dan memberikan respons otomatis apabila suhu melebihi batas yang telah ditentukan. Dalam praktikum ini, dirancang sebuah sistem pemantauan suhu berbasis IoT yang menggunakan sensor suhu DHT22 yang terhubung dengan mikrokontroler ESP8266. Data suhu yang dikumpulkan oleh sensor ini akan dikirim ke platform IoT, sehingga pengguna dapat memantau suhu tersebut melalui antarmuka web atau aplikasi seluler. Sistem ini juga dapat memberikan notifikasi atau mengaktifkan perangkat otomatis lainnya, seperti kipas, ketika suhu melebihi ambang batas yang telah ditentukan.

**1.2 Tujuan eksperimen**

Tujuan dari eksperimen ini adalah untuk mendesain dan menguji sistem pemantauan suhu berbasis IoT yang dapat mengirimkan data suhu secara real-time ke platform IoT untuk pemantauan jarak jauh. Selain itu, eksperimen ini bertujuan untuk menguji kemampuan sistem dalam merespons perubahan suhu secara otomatis, dengan memberikan peringatan atau mengaktifkan perangkat seperti kipas apabila suhu melebihi nilai ambang yang telah ditentukan. Dengan sistem ini, diharapkan dapat mempermudah pemantauan suhu dalam berbagai aplikasi, seperti di ruang kontrol, pertanian pintar, atau sistem pendinginan otomatis.

**2. Methodology (Metodologi)**

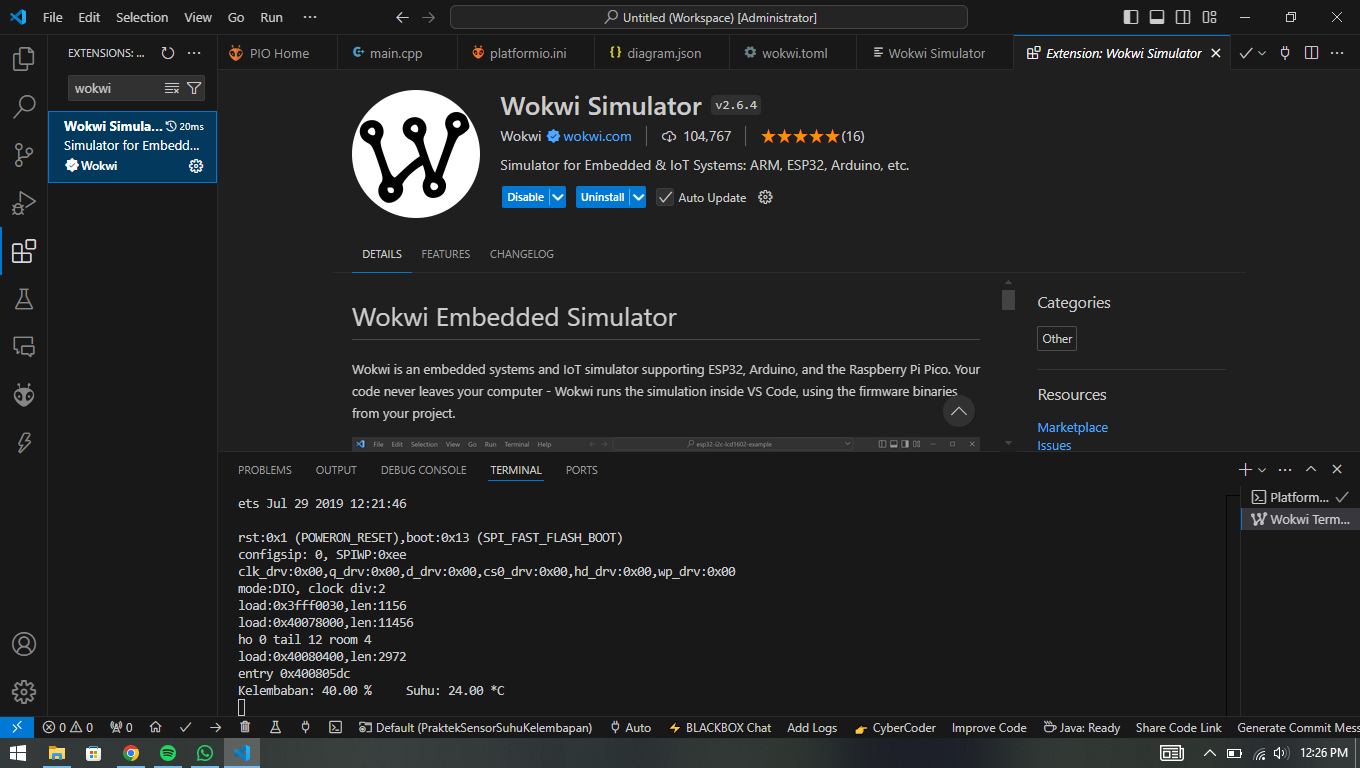
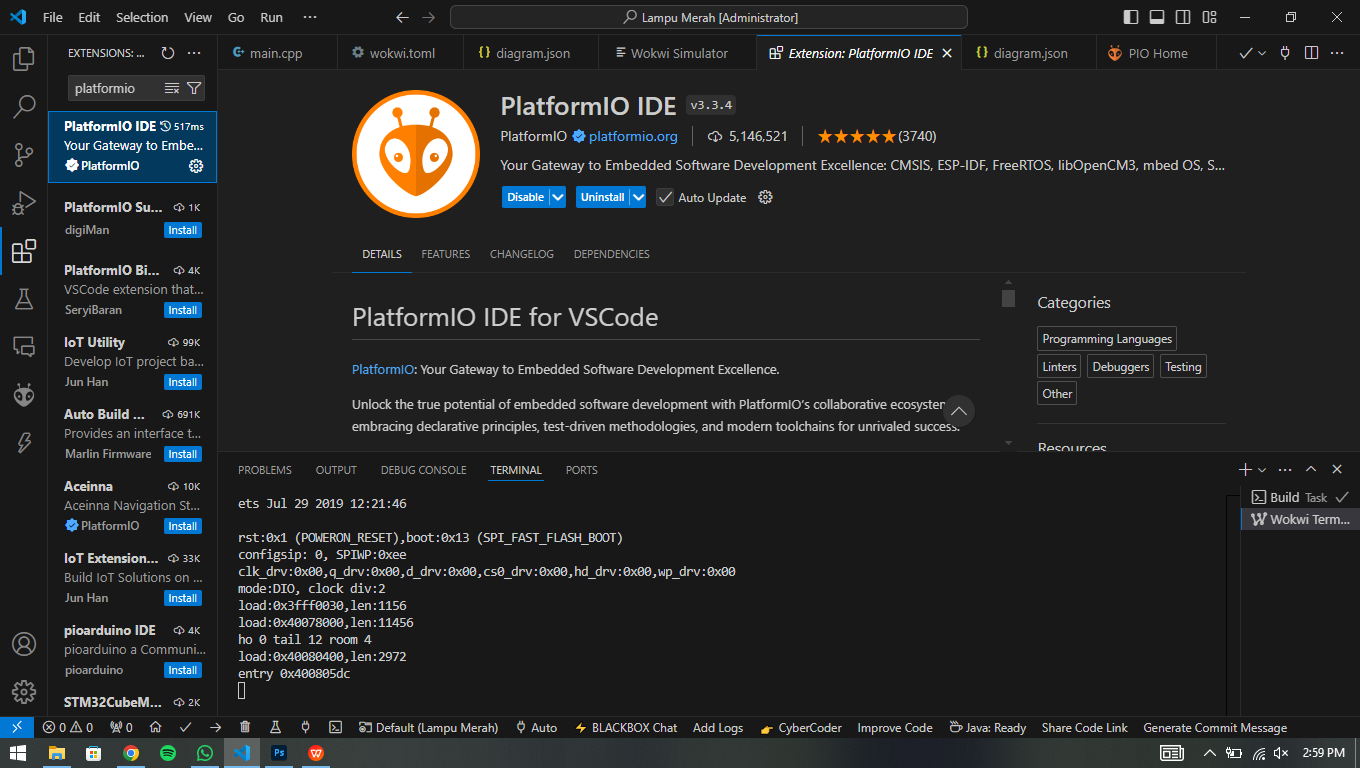
**2.1 Tools & Materials (Alat dan Bahan)**

Dalam eksperimen ini, beberapa alat dan bahan yang digunakan adalah sebagai berikut:

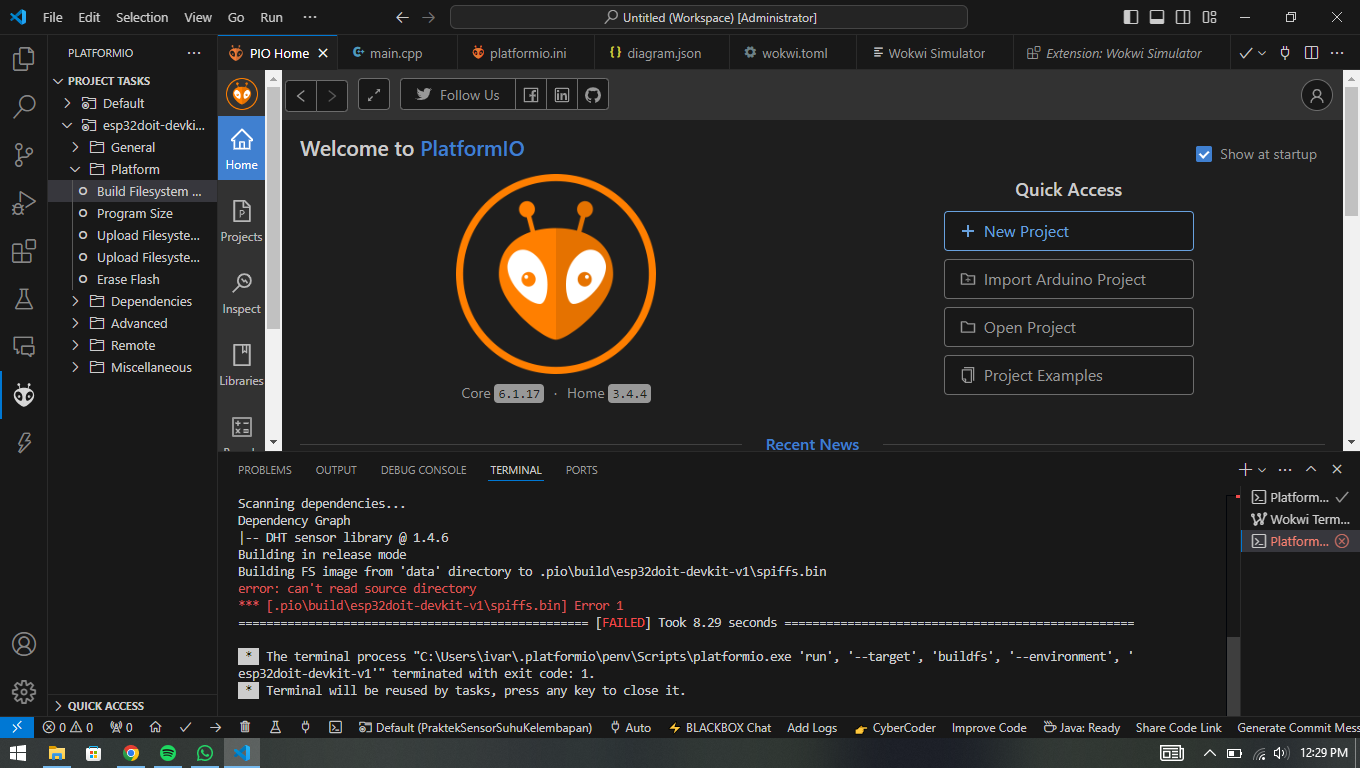
1. **Mikrokontroler**:  
   * **ESP8266**: Mikrokontroler berbasis Wi-Fi yang digunakan untuk menghubungkan sensor dengan platform IoT dan mengirimkan data suhu secara nirkabel. ESP8266 dipilih karena kemampuannya dalam koneksi internet dan kemudahan penggunaannya dalam aplikasi IoT.
2. **Sensor**:  
   * **Sensor Suhu DHT22**: Sensor yang digunakan untuk mengukur suhu dan kelembaban. DHT22 memiliki akurasi yang cukup baik dalam mengukur suhu pada rentang yang luas dan dapat memberikan hasil secara digital, yang memungkinkan integrasi langsung dengan mikrokontroler.
3. **Software**:  
   * **Arduino IDE**: Integrated Development Environment (IDE) yang digunakan untuk menulis dan meng-upload kode program ke mikrokontroler ESP8266. Arduino IDE mendukung berbagai bahasa pemrograman yang kompatibel dengan platform ESP8266.
   * **MQTT Broker**: Protokol yang digunakan untuk komunikasi antara mikrokontroler dan platform IoT. MQTT memungkinkan pengiriman data secara efisien dan real-time, dengan mengoptimalkan penggunaan bandwidth internet.

**2.2 Implementation Steps (Langkah Implementasi)**

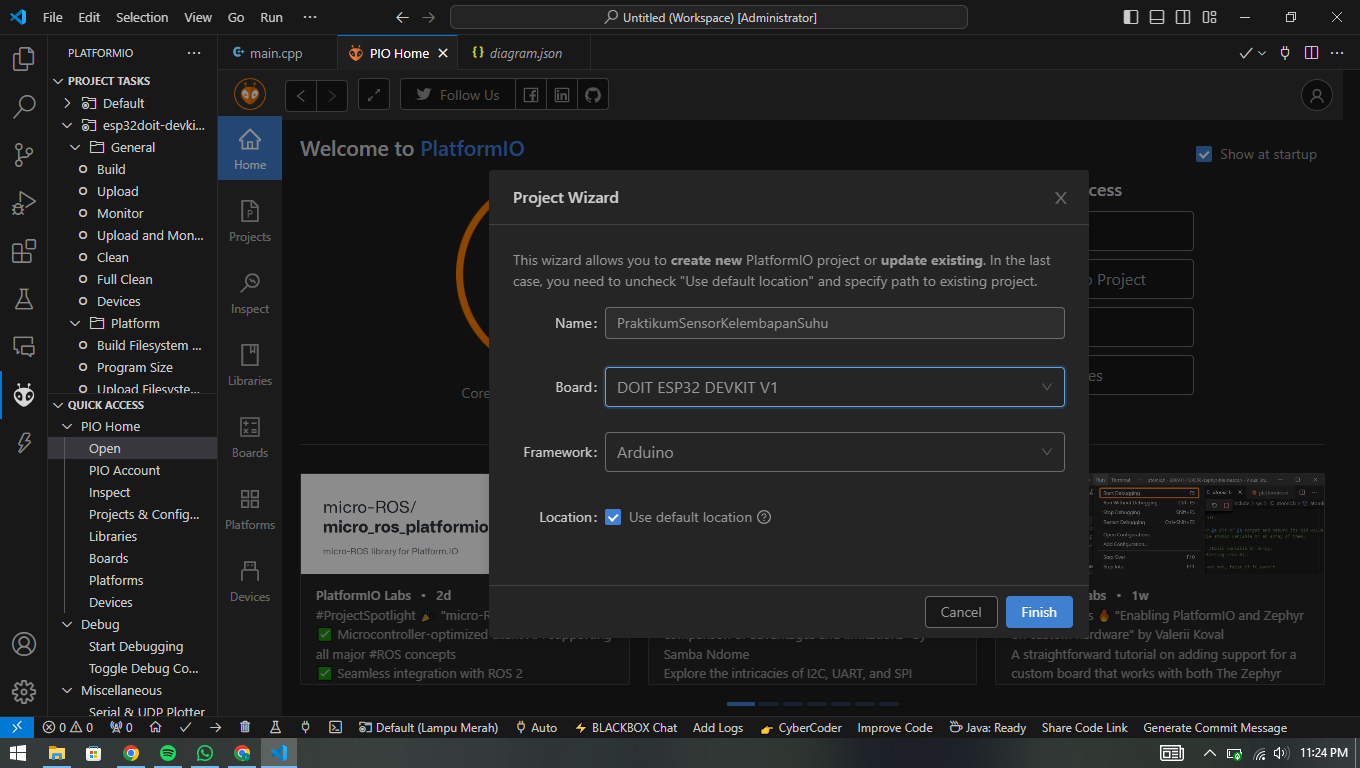
> Untuk langkah awal dalam menyusun sistem, pengkodean, dan pengujian adalah melakukan installasi ekstensi seperti **Wokwi Simulator, PlatformIO** pada V**isual Studio Code**.



> Setelah semua terinstall, klik PlatformIO lalu pilih **New Project**

****

**>** Beri nama project (PraktikumSensorKelembapanSuhu), kemudian pilih board menggunakan **Doit ESP32 Devkit V1**, dan framework **arduino.**

****

> Ubah syntax pada **src/main.cpp** seperti berikut,

**#include <Arduino.h>**

**#include <DHT.h>**

**#define DHTPIN 27 // Pin yang terhubung ke sensor DHT22**

**#define DHTTYPE DHT22 // Tipe sensor DHT**

**DHT dht(DHTPIN, DHTTYPE);**

**void setup() {**

**Serial.begin(115200);**

**dht.begin(); // Inisialisasi sensor**

**}**

**void loop() {**

**delay(2000); // Delay antar pembacaan**

**float humidity = dht.readHumidity();**

**float temperature = dht.readTemperature();**

**// Cek apakah pembacaan gagal**

**if (isnan(humidity) || isnan(temperature)) {**

**Serial.println("Gagal membaca sensor!");**

**return;**

**}**

**// Tampilkan hasil pembacaan**

**Serial.print("Kelembaban: ");**

**Serial.print(humidity);**

**Serial.print(" %\t");**

**Serial.print("Suhu: ");**

**Serial.print(temperature);**

**Serial.println(" \*C");**

**}**

> Setelah itu create file **diagram.json** dan masukkan syntax berikut melalui text editor,

**{**

**"version": 1,**

**"author": "Ivan AryaPutra Rachmadhani",**

**"editor": "wokwi",**

**"parts": [**

**{ "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },**

**{ "type": "wokwi-dht22", "id": "dht1", "top": -38.1, "left": -159, "attrs": {} }**

**],**

**"connections": [**

**[ "esp:TX", "$serialMonitor:RX", "", [] ],**

**[ "esp:RX", "$serialMonitor:TX", "", [] ],**

**[ "dht1:VCC", "esp:3V3", "red", [ "v38.4", "h67.2", "v-86.4" ] ],**

**[ "dht1:SDA", "esp:27", "green", [ "v0" ] ],**

**[ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]**

**],**

**"dependencies": {}**

**}**

> Jika sudah, lanjut create file **wokwi.toml** dan masukkan syntax berikut:

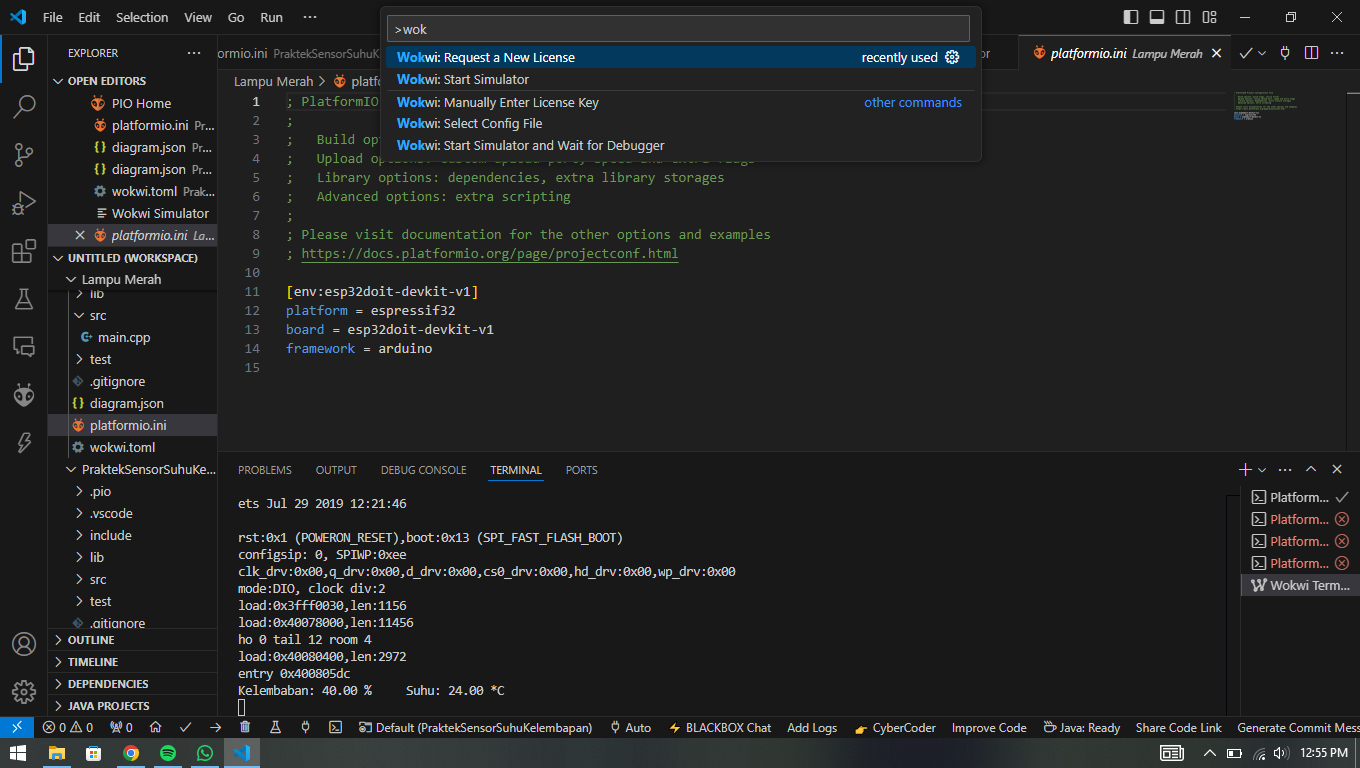
[wokwi]

version = 1

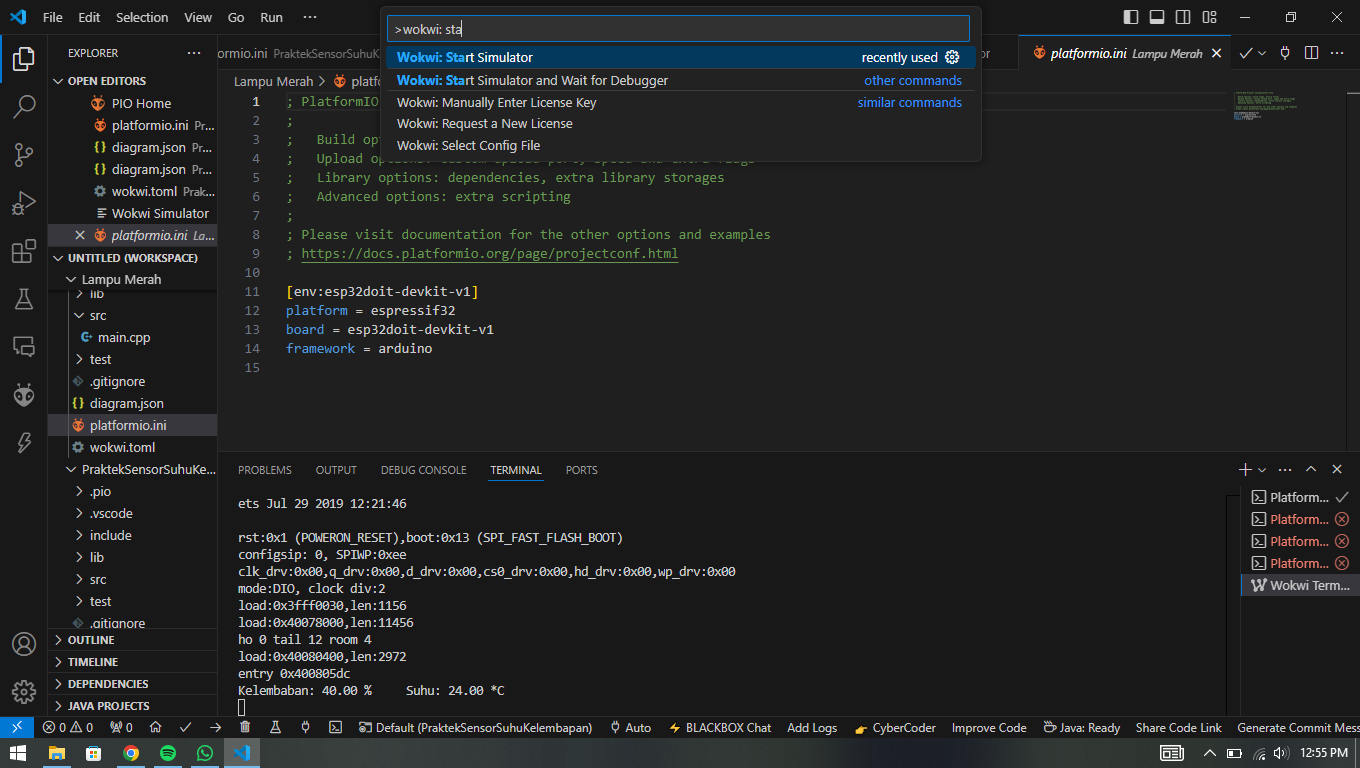
firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'

elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'

> Langkah selanjutnya adalah meminta lisensi wokwi untuk dapat menjalankan simulator sensornya **(Wokwi: Request a New License)**



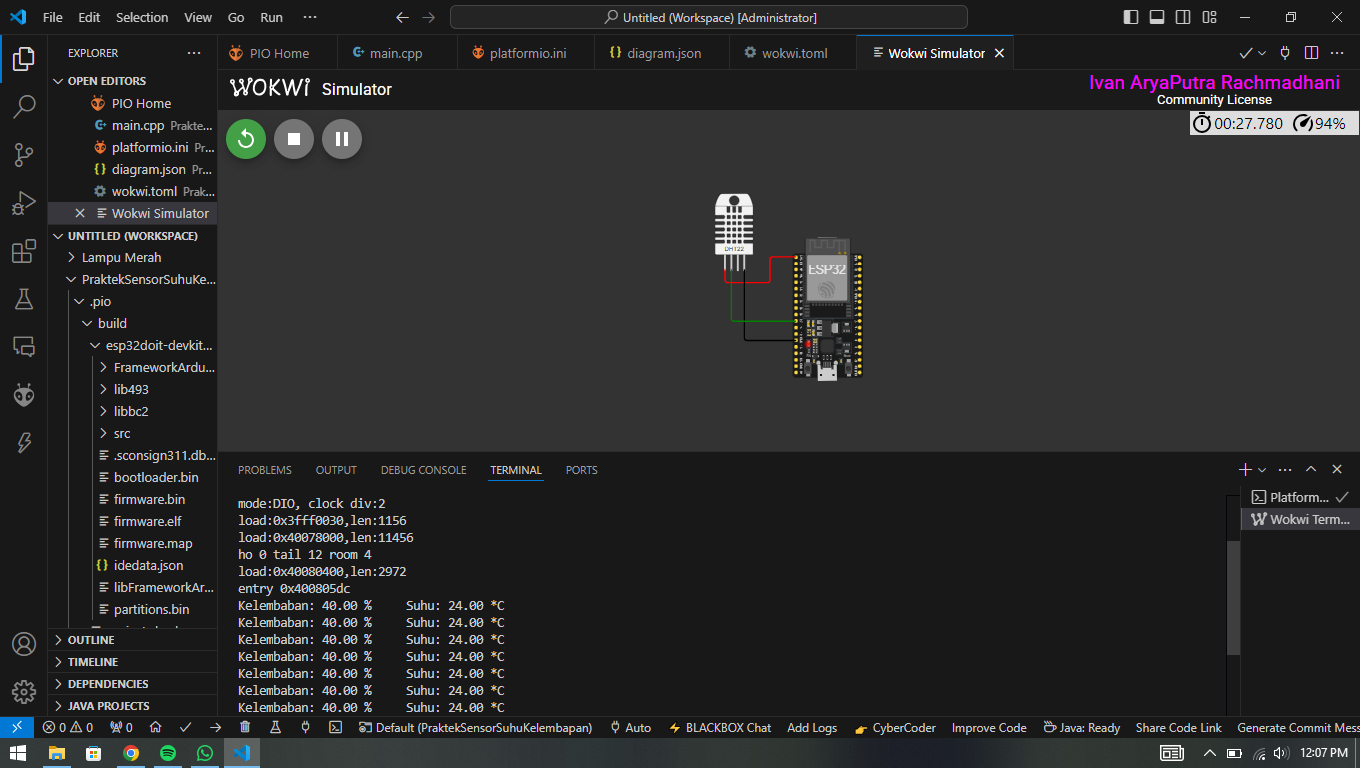
> Langkah terakhir lakukan **start simulator (Wokwi: Start Simulator)**



**3. Results and Discussion (Hasil dan Pembahasan)**

**3.1 Experimental Results (Hasil Eksperimen)**

> Data yang diperoleh



**4. Appendix (Lampiran, jika diperlukan)**

> Kode program cpp

#include <Arduino.h>

#include <DHT.h>

#define DHTPIN 27 // Pin yang terhubung ke sensor DHT22

#define DHTTYPE DHT22 // Tipe sensor DHT

DHT dht(DHTPIN, DHTTYPE);

void setup() {

Serial.begin(115200);

dht.begin(); // Inisialisasi sensor

}

void loop() {

delay(2000); // Delay antar pembacaan

float humidity = dht.readHumidity();

float temperature = dht.readTemperature();

// Cek apakah pembacaan gagal

if (isnan(humidity) || isnan(temperature)) {

Serial.println("Gagal membaca sensor!");

return;

}

// Tampilkan hasil pembacaan

Serial.print("Kelembaban: ");

Serial.print(humidity);

Serial.print(" %\t");

Serial.print("Suhu: ");

Serial.print(temperature);

Serial.println(" \*C");

}