LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik** **Real Hardware LED**

*Emilda Citra Diva*

Fakultas Vokasi, Universitas Brawijaya

[Citradiva1010@student.ub.ac.id](mailto:Citradiva1010@student.ub.ac.id)

**Abstract (Abstrak)**

1. **Introduction (Pendahuluan)**
   1. **Latar Belakang**

Praktikum ini bertujuan untuk mempelajari dasar pemrograman mikrokontroler ESP32 dengan implementasi pengendalian LED, pemindaian jaringan WiFi, serta integrasi sensor suhu dan kelembaban (DHT22) ke API berbasis Laravel menggunakan tunneling NGROK. Proses dilakukan secara bertahap dimulai dari upload program LED berkedip, scanning WiFi di sekitar, hingga pengiriman data aktual dari ESP32 ke server Laravel menggunakan HTTP POST. Hasil praktik menunjukkan ESP32 dapat terhubung ke WiFi, membaca data suhu/kelembaban, dan mengirimkannya ke database melalui API dengan sukses.

**1.2 Tujuan Eksperimen**

Mempelajari pengoperasian ESP32 dari pengendalian output digital (LED), scanning WiFi, hingga komunikasi data sensor ke server API menggunakan NGROK

**Methodology (Metodologi)**

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

1. Mikrokontroler ESP32
2. DHT11
3. XAMPP/MySQL
4. Wokwi IoT Simulator
5. Arduino IDE
6. VS Code
7. BreadBoard
8. Kabel USB
9. LED
10. Kabel Jumper

**2.2 Implementation Steps (Langkah Implementasi)**

1. Melakukan scanning WiFi menggunakan ESP32.
2. Menyambungkan sensor DHT22 ke pin 27 pada ESP32.
3. Menulis dan memodifikasi kode main.cpp agar membaca data suhu dan kelembaban lalu mengirimkannya ke API.
4. Menjalankan Laravel menggunakan php artisan serve --host=0.0.0.0 --port=8080.
5. Menjalankan NGROK dengan perintah ngrok http --scheme=http 8080.
6. Melakukan pengujian POST request melalui ESP32 ke URL NGROK yang tersedia.
7. Mengamati hasil melalui serial monitor.

**Main ccp dht22**

#include <Arduino.h>

#include <WiFi.h>

#include <HTTPClient.h>

#include "DHT.h"

#define DHTPIN 27

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

// Ganti dengan kredensial WiFi Anda

const char\* ssid = "diva's phone";

const char\* password = "Citradiva10";

unsigned long previousMillis = 0;

const long interval = 5000;  // Interval 5 detik (5000 ms)

void setup() {

  Serial.begin(115200);

  // Hubungkan ke WiFi

  WiFi.begin(ssid, password);

  Serial.print("Menghubungkan ke WiFi");

  while (WiFi.status() != WL\_CONNECTED) {

    delay(500);

    Serial.print(".");

  }

  Serial.println(" Terhubung!");

  dht.begin();

  // Tunggu sebentar agar koneksi stabil

  delay(1000);

}

void loop() {

  unsigned long currentMillis = millis();

  // Lakukan POST setiap interval yang telah ditentukan

  if (currentMillis - previousMillis >= interval) {

    previousMillis = currentMillis;

    float h = round(dht.readHumidity());

    // Read temperature as Celsius (the default)

    float t = round(dht.readTemperature());

    // Check if any reads failed and exit early (to try again).

    if (isnan(h) || isnan(t)) {

      Serial.println(F("Failed to read from DHT sensor!"));

      return;

    }

    // Compute heat index in Celsius (isFahreheit = false)

    float hic = dht.computeHeatIndex(t, h, false);

    // Inisialisasi HTTPClient

    HTTPClient http;

    String url = "http://a710-175-45-190-3.ngrok-free.app/api/posts"; // Ganti dengan URL ngrok yang benar

    http.begin(url);  // Menggunakan HTTP, bukan HTTPS

    http.addHeader("Content-Type", "application/json");

String payload = "{\"nama\_sensor\":\"Sensor GD\", \"nilai1\":" + String(h) + ", \"nilai2\":" + String(t) + "}";

Serial.println(payload);  // Untuk melihat apakah payload sudah terbentuk dengan benar

    // Kirim POST request

    int httpResponseCode = http.POST(payload);

    // Tampilkan kode respons HTTP

    Serial.print("Kode respons HTTP: ");

    Serial.println(httpResponseCode);

    // Tampilkan respons dari server jika request berhasil

    if (httpResponseCode == 200 || httpResponseCode == 201) {

      String response = http.getString();

      Serial.println("Respons dari server:");

      Serial.println(response);

    } else {

      Serial.println("Gagal mengirim data");

    }

    // Tutup koneksi HTTP

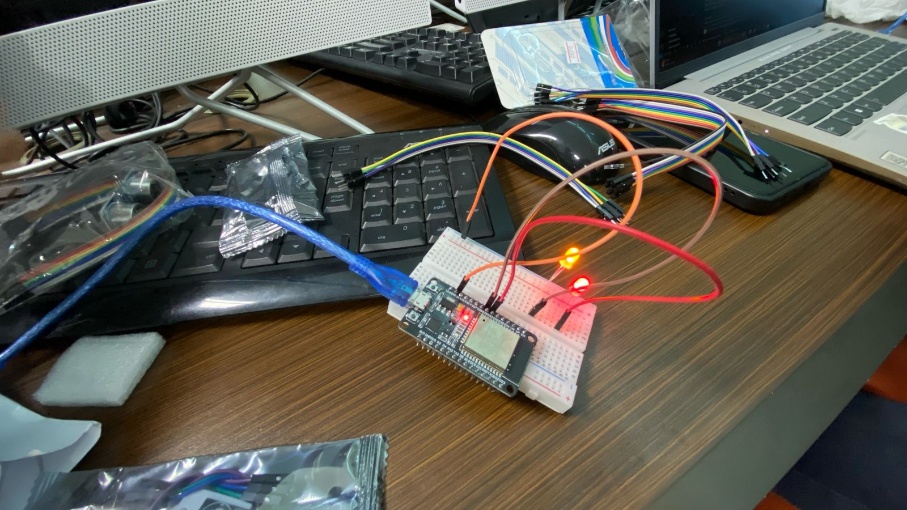
    http.end();

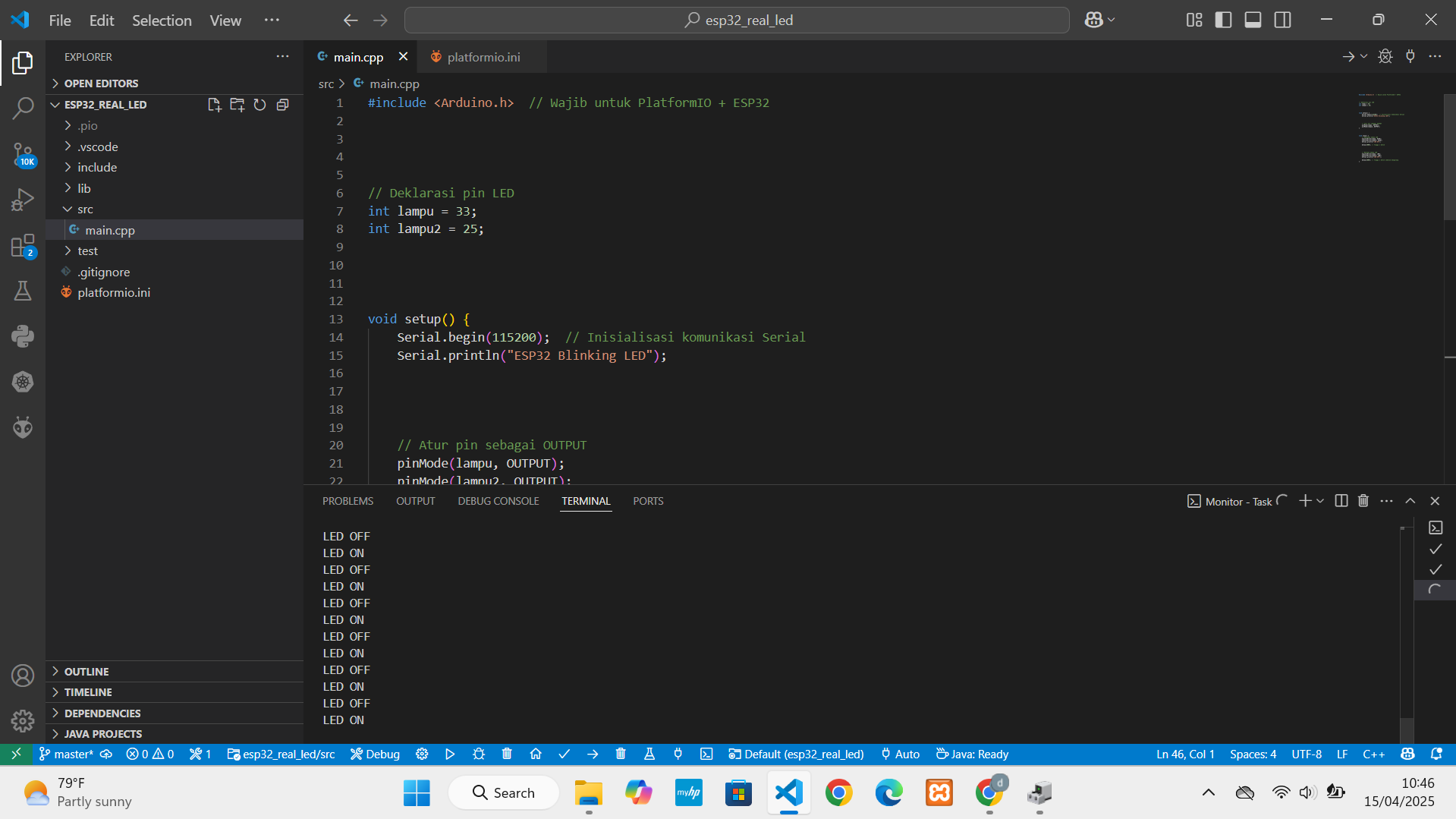
  }

}

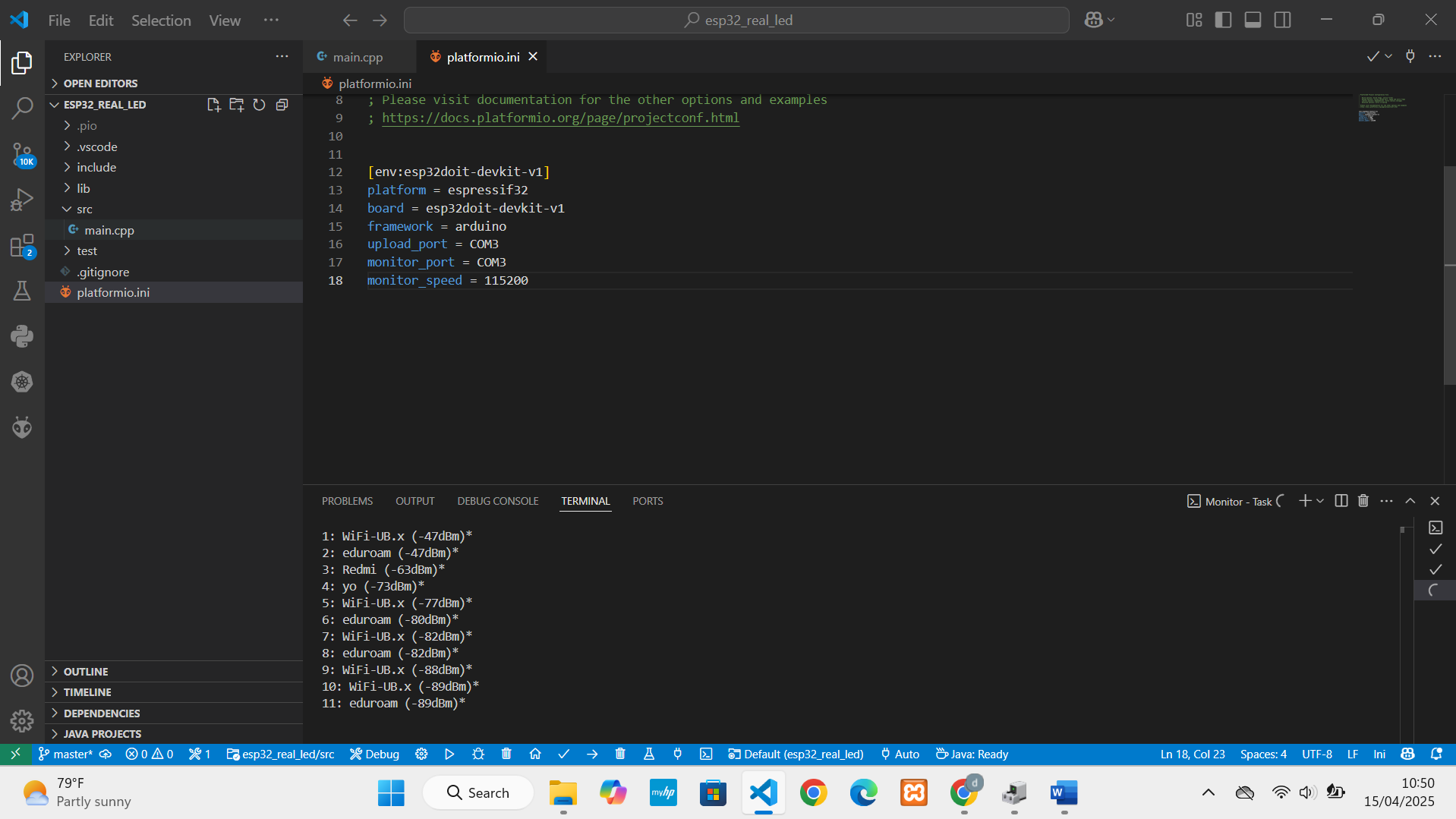
1. **Results and Discussion (Hasil dan Pembahasan)**

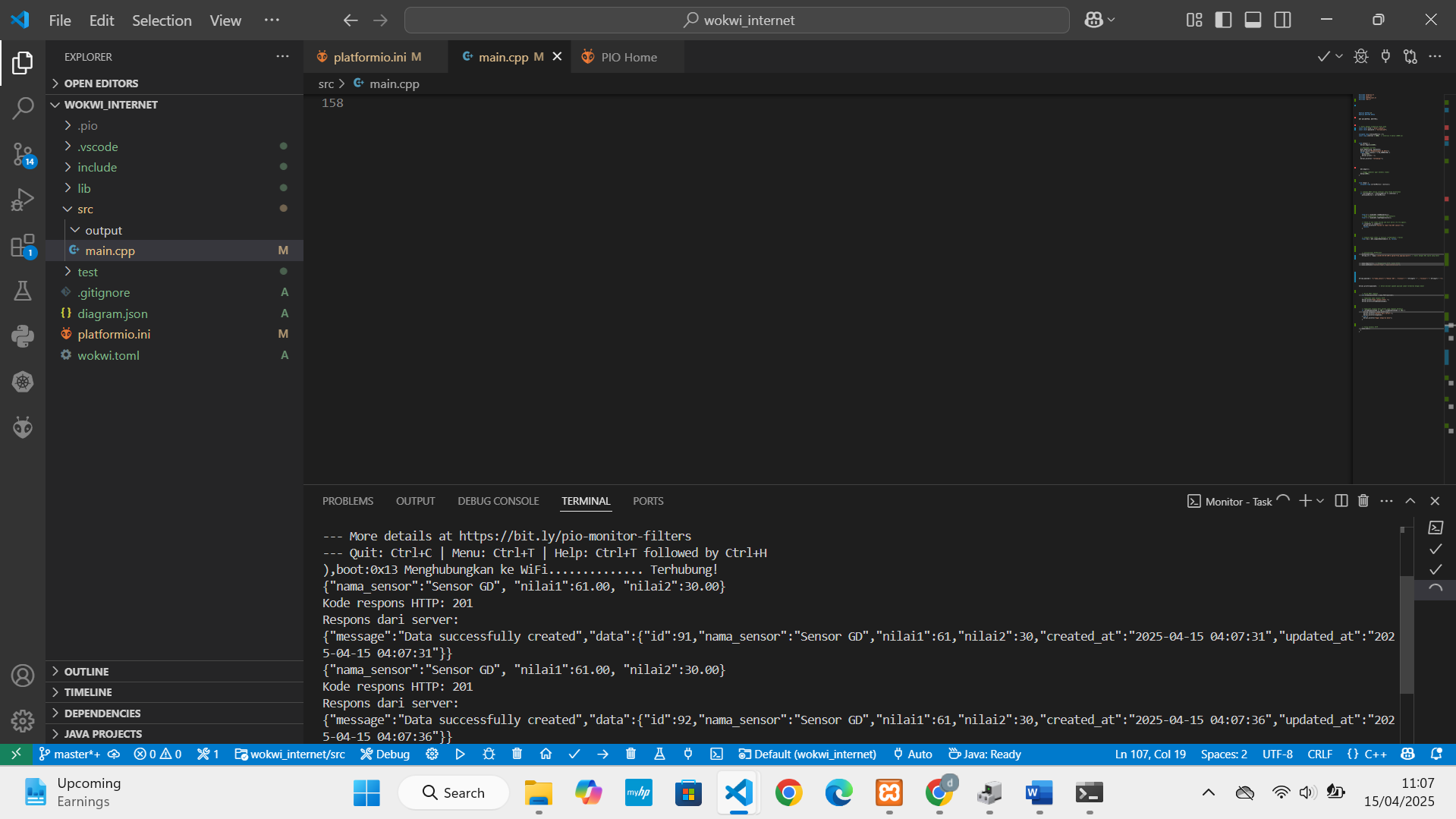
**3.1 Experimental Results (Hasil Eksperimen)**

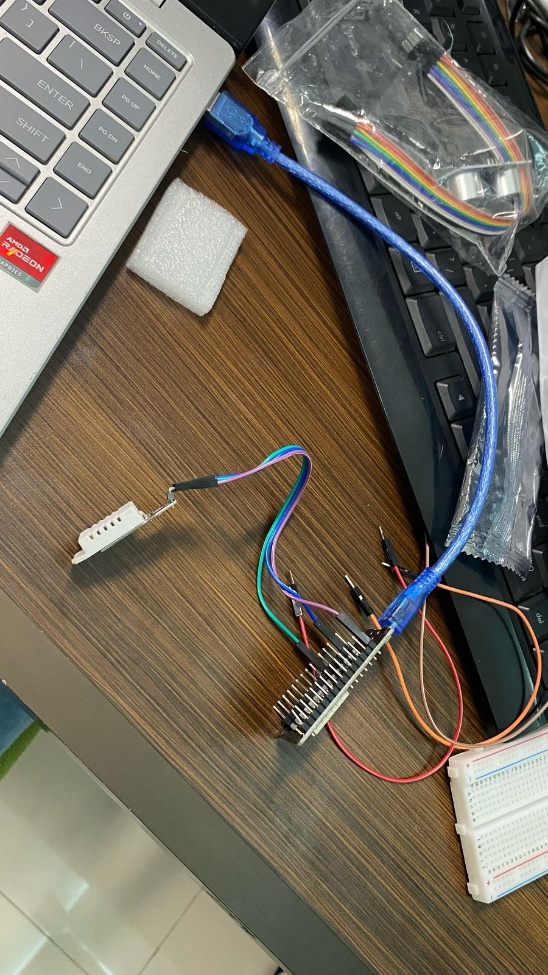
****

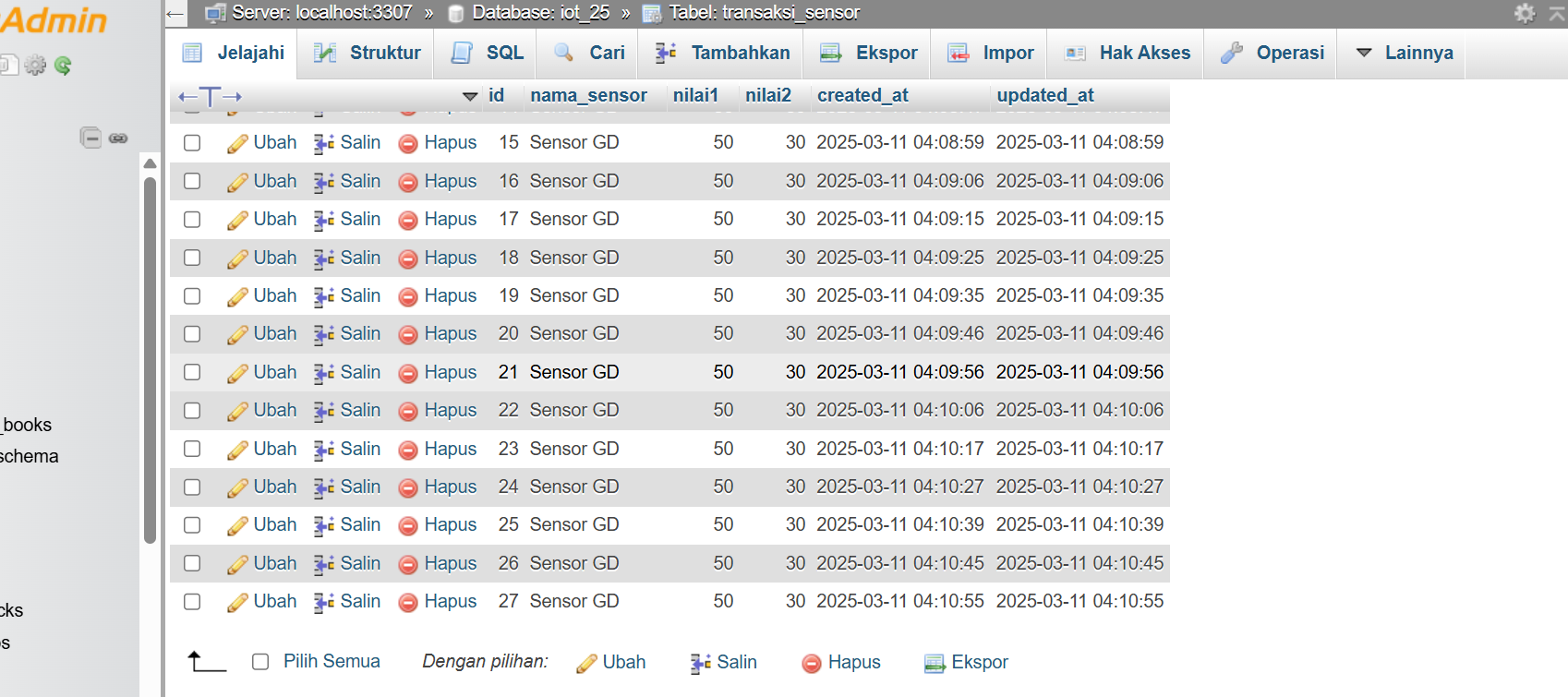
****

Scan wifi



****

****

****