LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Akses API Melalui Simulasi WOKWI**

*Eunike Febianti*

Fakultas Vokasi, Universitas Brawijaya

[eunikefebianti93@gmail.com](mailto:eunikefebianti93@gmail.com)

**Abstract (Abstrak)**

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

IoT semakin berkembang dalam berbagai bidang, termasuk pemantauan lingkungan yang memanfaatkan sensor untuk mengirimkan data ke sistem berbasis web. ESP32 adalah salah satu mikrokontroler yang sering digunakan untuk proyek IoT karena mendukung konektivitas WiFi. Namun, pengujian perangkat keras dapat menjadi tantangan karena keterbatasan alat dan lingkungan. Oleh karena itu, WOKWI sebagai simulator ESP32 menjadi solusi untuk menguji komunikasi antara sensor dan server secara virtual sebelum implementasi pada perangkat fisik.

* 1. **Tujuan Eksperimen**

a) Memahami konsep komunikasi IoT menggunakan ESP32 dan API.

b) Mempelajari cara mengirimkan data sensor ke server melalui HTTP POST.

c) Menggunakan simulator WOKWI untuk menguji pengiriman data secara virtual.

d) Memverifikasi keberhasilan pengiriman data dengan melihat respon dari server.

1. **Methodology (Metodologi)**

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

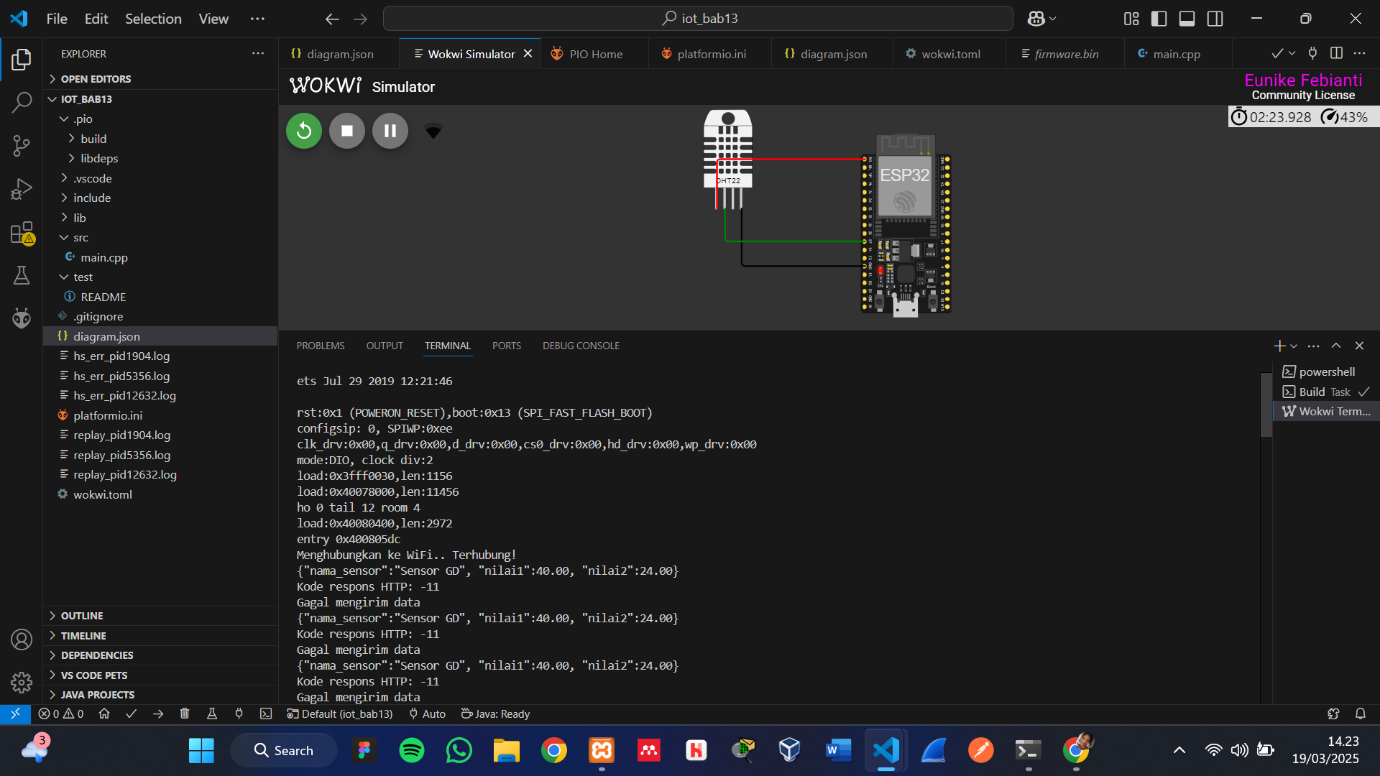
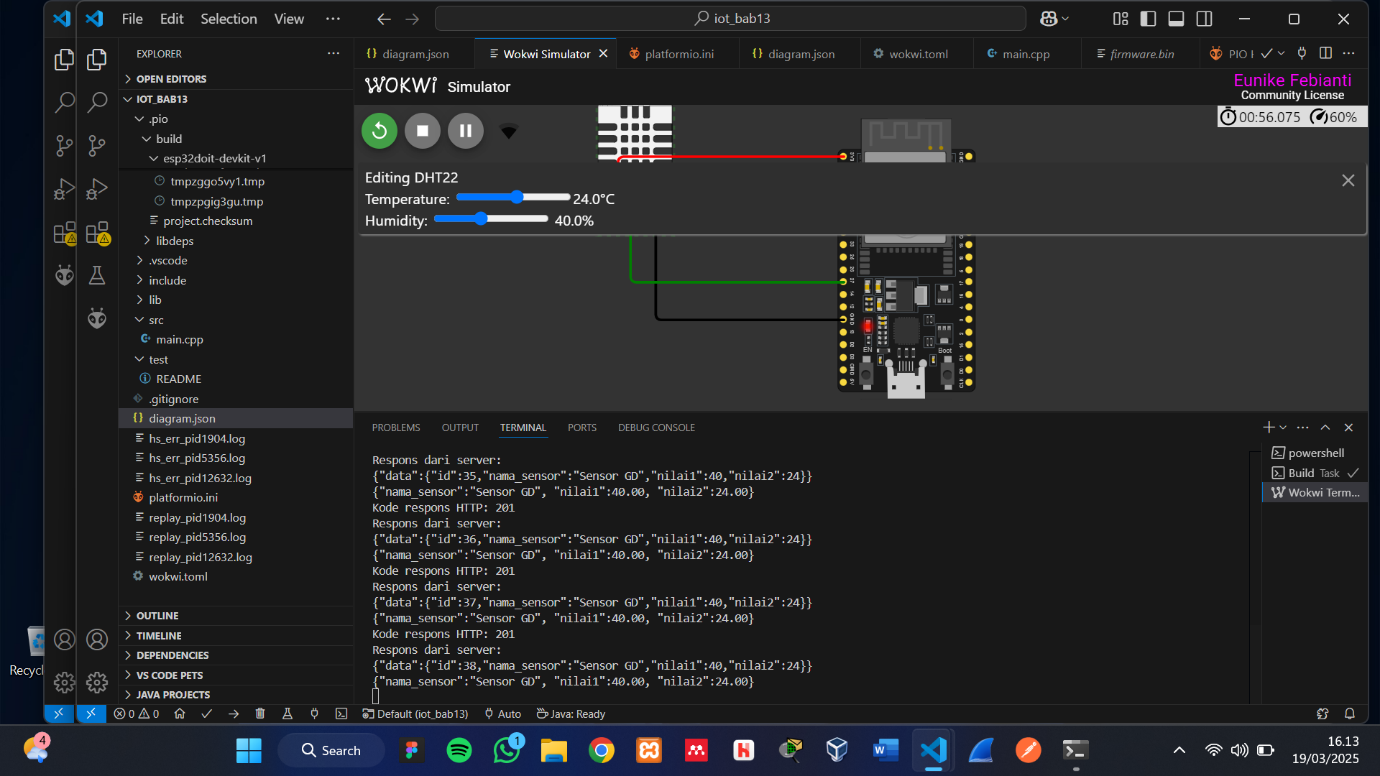
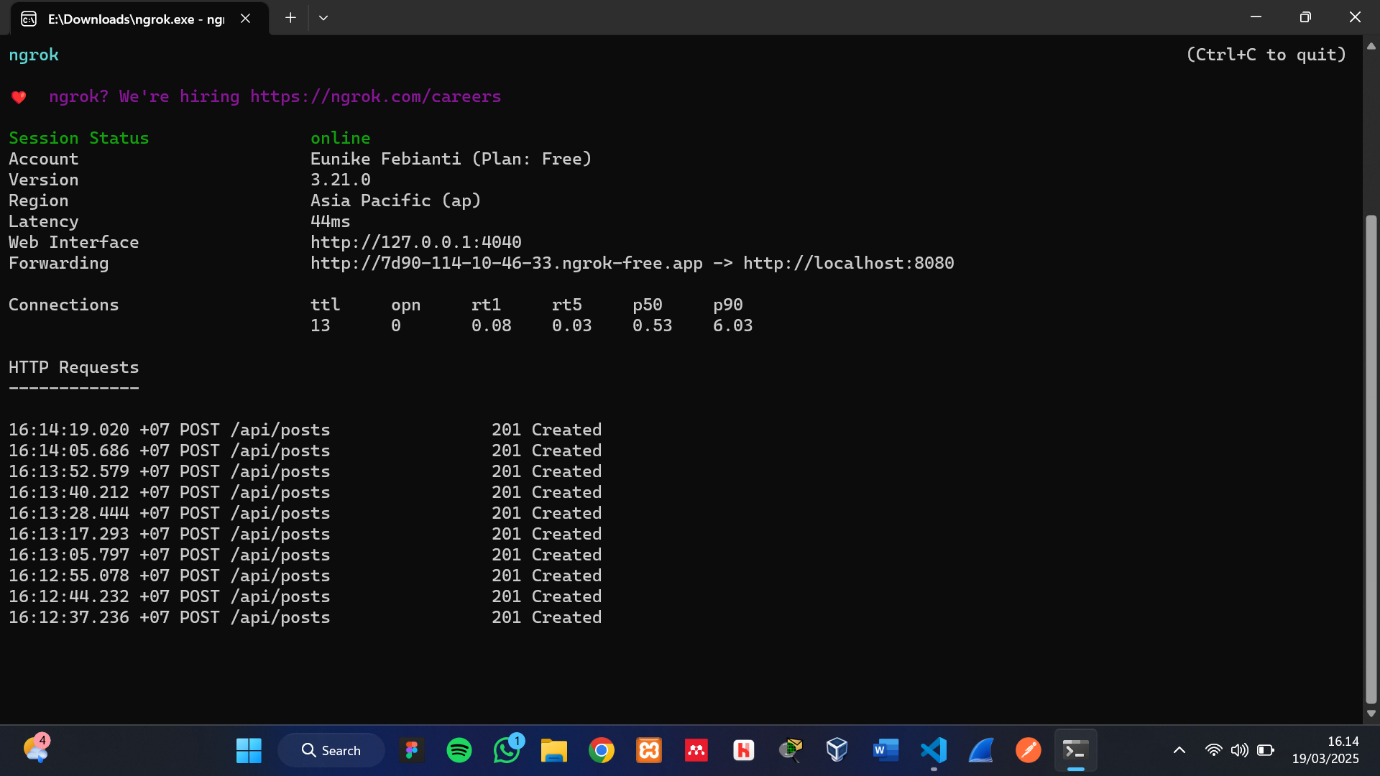
Laptop, VS Code, XAMPP, phpMyAdmin, Herd, Laravel 11, Postman, Ngrok, dan koneksi internet.

**2.2 Implementation Steps (Langkah Implementasi)**

* 1. Dari tugas bab 12 sebelumnya, jalankan API terlebih dahulu di terminal dengan **php artisan serve --host=0.0.0.0 --port=8080**
  2. Buat project baru wokwi simulator di platform.io dengan nama iot\_bab13 dengan board DOIT ESP32 DEVKIT V1
  3. Isikan main.cpp dengann kodingan yang tertera di modul (codingan terdapat di lapiran)
  4. isikan file wokwi.toml (codingan terdapat di lampiran)
  5. isikan file diagram.json (codingan terdapat di lampiran)
  6. tambahkan library **lib\_deps = adafruit/DHT sensor** [**library@^1.4.6**](mailto:library@%5e1.4.6) pada file platformio.ini
  7. Kemudian buka Ngrok yang sudah terinstall pada bab sebelumnya klik 2x, lalu akan diarahkan ke cmd lalu jalankan perintah konfigurasi yang ada pada akun Ngrok di cmd, lalu ketikkan perintah ini: **ngrok http --scheme=http 8000**.
  8. Copy link forwarding yang terdapat di aplikasi ngrok dan pastekan ke dalam file main.cpp (pada line //inisiasi HTTP client)
  9. Start wokwi simulator nya

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

**3.1 Experimental Results (Hasil Eksperimen)**

****

**A screenshot of a computer

AI-generated content may be incorrect.**

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

**Wokwi.toml**

[wokwi]

version = 1

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

elf = 'C:\Users\ASUS\Documents\PlatformIO\Projects\iot\_bab13\.pio\build\esp32doit-devkit-v1\firmware.elf'

**diagram.json**

{

"version": 1,

"author": "Eunike Febianti",

"editor": "wokwi",

"parts": [

{ "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 28.8, "left": 139.24, "attrs": {} },

{ "type": "wokwi-dht22", "id": "dht1", "top": 0.3, "left": -43.8, "attrs": {} }

],

"connections": [

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

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

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

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

[ "dht1:VCC", "esp:3V3", "red", [ "v-57.6", "h134.4", "v38.4" ] ]

],

"dependencies": {}

}

**Main.cpp**

#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 = "Wokwi-GUEST";

const char\* password = "";

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://7d90-114-10-46-33.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();

}

}

**Platformio.ini**

[env:esp32doit-devkit-v1]

platform = espressif32

board = esp32doit-devkit-v1

framework = arduino

lib\_deps = adafruit/DHT sensor library@^1.4.6