LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT):

**Praktik Akses API Melalui Simulasi WOKWI**



**Micko Junior Putra Pratma**

**233140700111076 (T4C)**

**Email : mickojunior1004@gmail.com**

Fakultas Vokasi

Program studi Teknologi Informasi

Universitas Brawijaya

2025

**Abstrak**

Wokwi merupakan platform simulasi berbasis web yang memungkinkan pengujian mikrokontroler seperti ESP32. Sementara itu, Ngrok adalah alat yang digunakan untuk membuat terowongan, sehingga server lokal dapat diakses melalui internet.

**1.Pendahuluan**

* 1. **Latar belakang**

API berfungsi sebagai penghubung yang memungkinkan perangkat IoT mengirim serta menerima data ke backend, seperti database atau aplikasi web. Dalam lingkungan pengembangan lokal, API biasanya berjalan di localhost, sehingga tidak dapat diakses oleh perangkat di luar jaringan lokal. Untuk mengatasi keterbatasan ini, Ngrok digunakan sebagai alat tunneling yang memberikan akses ke server lokal melalui internet dengan menyediakan URL publik yang dapat diakses oleh Wokwi atau perangkat IoT lainnya.

**1.2 Tujuan eksperimen**

 Menganalisis penggunaan Wokwi untuk mensimulasikan perangkat IoT dalam mengakses API secara virtual menggunakan PlatformIO di VS Code.

 Menguji konektivitas antara API lokal berbasis Laravel dengan simulasi perangkat di Wokwi melalui Ngrok sebagai layanan tunneling.

**2. Methodology (Metodologi)**

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

* ESP32 (simulasi)
* Wokwi Simulator
* PlatformIO di VS Code
* PHP

**2.2 Implementation Steps (Langkah Implementasi)**

 Membuat database **iot\_25** di phpMyAdmin.

 Menginstal dan mengonfigurasi Laravel 11 di VS Code.

 Membuat model **TransaksiSensor** dan melakukan migrasi database.

 Membuat **TransaksiSensorController** untuk mengelola data sensor.

 Mengonfigurasi route API di Laravel.

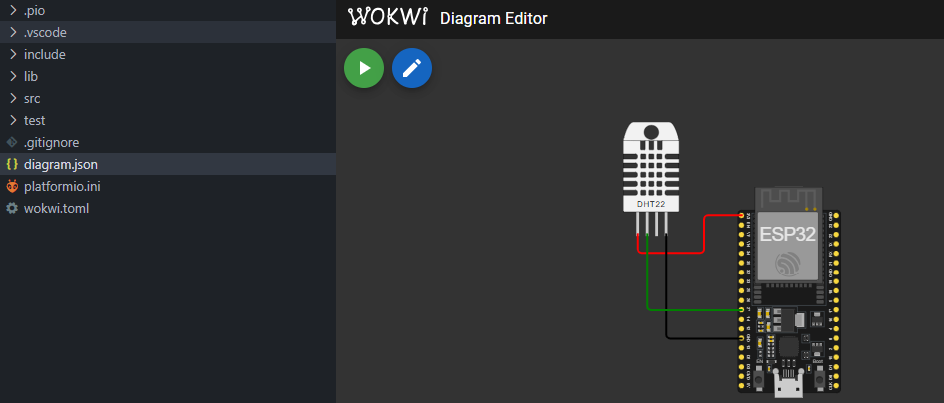
 Menjalankan server Laravel dengan php artisan serve.

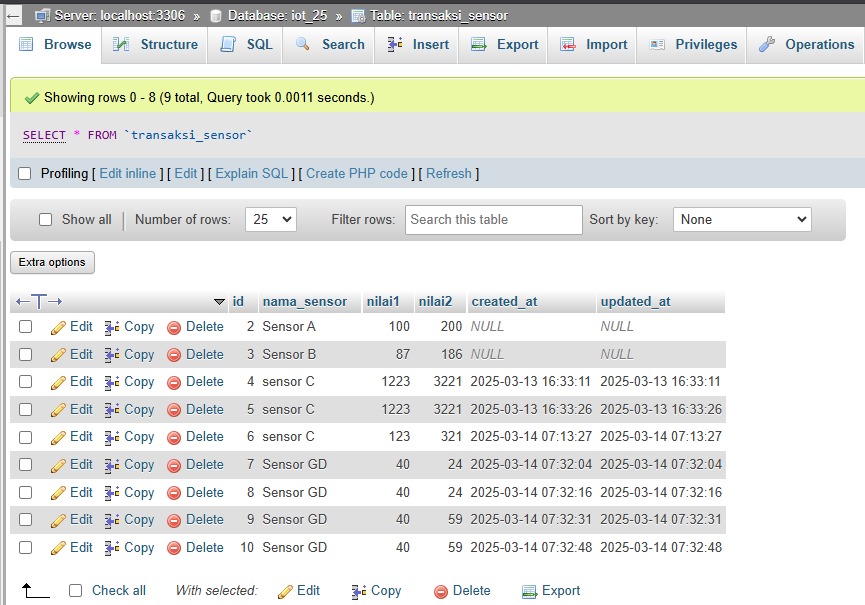
 Mengonline-kan API menggunakan Ngrok.

 Mengakses API dari Wokwi untuk mengirim dan menerima data sensor

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

**3.1 Experimental Results**





**4. Appendix**

#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://1c04-175-45-191-252.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();

  }

}