# LAPORAN PRAKTIKUM

# IMPLEMENTASI WIDGET MENGGUNAKAN BLYNK

Disusun untuk memenuhi tugas mata kuliah Internet of Things

Dosen Pengampu:

Ir. Subairi, ST., MT., IPM



Disusun Oleh:

Putri Zamzami Rizkiani

233140707111119

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**DEPARTEMEN INDUSTRI KREATIF DAN DIGITAL**

**FAKULTAS VOKASI**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2025**

# DAFTAR ISI

[DAFTAR ISI ii](#_Toc201458866)

[ABSTRAK 3](#_Toc201458867)

[BAB I 4](#_Toc201458868)

[PENDAHULUAN 4](#_Toc201458869)

[1.1 Latar Belakang 4](#_Toc201458870)

[1.2 Tujuan Praktikum 4](#_Toc201458874)

[BAB II 5](#_Toc201458875)

[METODOLOGI 5](#_Toc201458876)

[2.1 Alat dan Bahan 5](#_Toc201458877)

[2.2 Langkah Implementasi 5](#_Toc201458878)

[BAB III 7](#_Toc201458879)

[HASIL DAN PEMBAHASAN 7](#_Toc201458880)

[3.1 Hasil Eksperimen 7](#_Toc201458881)

[3.2 Kode Program 8](#_Toc201458882)

[3.2.1 Main.cpp 8](#_Toc201458883)

[3.2.2 Diagram.json 10](#_Toc201458884)

[BAB IV 11](#_Toc201458885)

[KESIMPULAN 11](#_Toc201458886)

[4.1 Kesimpulan 11](#_Toc201458887)

# ABSTRAK

Perkembangan teknologi Internet of Things (IoT) telah membuka peluang besar dalam menciptakan sistem monitoring berbasis sensor yang dapat diakses secara real-time. Dalam proyek ini, dirancang sebuah sistem pemantauan suhu dan kelembaban menggunakan mikrokontroler ESP32 dan sensor DHT22, yang terhubung ke platform Blynk sebagai antarmuka pengguna. Proses pengembangan dimulai dari simulasi awal menggunakan Wokwi untuk memastikan koneksi perangkat dan validasi kode, kemudian dilanjutkan dengan implementasi nyata menggunakan PlatformIO di Visual Studio Code. Data suhu dan kelembaban ditampilkan melalui widget interaktif di aplikasi Blynk dan diperbarui secara berkala melalui koneksi WiFi. Proyek ini membuktikan bahwa sistem pemantauan berbasis IoT dapat dikembangkan dengan cara yang efisien, fleksibel, dan terintegrasi.

*Kata kunci: IoT, ESP32, DHT22, Blynk, Sensor Suhu, Sensor Kelembaban, PlatformIO*

The advancement of Internet of Things (IoT) technology has enabled the development of sensor-based monitoring systems that can be accessed in real time. In this project, a temperature and humidity monitoring system is designed using an ESP32 microcontroller and a DHT22 sensor, connected to the Blynk platform as a user interface. The development process begins with an initial simulation using Wokwi to validate device connections and code functionality, followed by real implementation using PlatformIO in Visual Studio Code. The temperature and humidity data are displayed through interactive widgets in the Blynk application and are updated periodically via a WiFi connection. This project demonstrates that IoT-based monitoring systems can be developed efficiently, flexibly, and in an integrated manner.

*Keyword: IoT, ESP32, DHT22, Blynk, Temperature Sensor, Humidity Sensor, PlatformIO*

# BAB I

# PENDAHULUAN

## Latar Belakang

## Perkembangan teknologi Internet of Things (IoT) telah mengubah cara manusia berinteraksi dengan lingkungan fisik melalui koneksi internet. IoT memungkinkan perangkat elektronik seperti sensor, mikrokontroler, dan aktuator untuk saling berkomunikasi dan bertukar data secara otomatis tanpa campur tangan manusia secara langsung. Salah satu implementasi praktis dari teknologi ini adalah dalam bidang pemantauan lingkungan, seperti monitoring suhu dan kelembaban udara, yang sangat dibutuhkan dalam berbagai sektor seperti pertanian, industri, hingga kesehatan.

## Dalam proyek ini, sistem monitoring suhu dan kelembaban dibangun menggunakan mikrokontroler ESP32 dan sensor DHT22. ESP32 dipilih karena memiliki konektivitas WiFi bawaan, performa tinggi, serta mudah diintegrasikan dengan berbagai platform. Sensor DHT22 digunakan karena mampu membaca suhu dan kelembaban dengan akurasi yang cukup baik. Data yang dihasilkan oleh sensor akan dikirim ke aplikasi Blynk melalui jaringan WiFi, sehingga dapat dipantau secara real-time melalui smartphone.

## Proyek ini juga memanfaatkan simulasi awal menggunakan platform Wokwi untuk memudahkan proses perancangan dan pengujian sebelum diimplementasikan secara nyata menggunakan Visual Studio Code dengan PlatformIO. Dengan integrasi ke aplikasi Blynk, pengguna dapat melihat pembacaan sensor melalui widget serta mengontrol LED secara jarak jauh. Dengan demikian, proyek ini tidak hanya menjadi contoh implementasi IoT sederhana, tetapi juga memperkenalkan alur kerja yang efisien dari tahap simulasi hingga deployment sistem monitoring berbasis cloud.

## Tujuan Praktikum

1. Menerapkan konsep IoT untuk monitoring suhu dan kelembapan secara real time.
2. Menghubungkan mikrokontroler ESP32 dengan Blynk Cloud untuk pengiriman data ke aplikasi Blynk.
3. Menampilkan data sensor dalam bentuk widget interaktif di aplikasi Blynk.

# BAB II

# METODOLOGI

## Alat dan Bahan

1. ESP32 sebagai mikrokontroler utama
2. Sensor DHT22
3. LED indicator
4. Resistor
5. Platform Wokwi
6. VSCode + PlatformIO
7. Platform Blynk

## Langkah Implementasi

1. Simulasi Awal di Wokwi

* Buka situs <https://wokwi.com> dan buat proyek baru dengan board ESP32 DevKit V1.
* Tambahkan sensor DHT22, LED (merah), dan resistor 1kΩ ke dalam diagram.
* Sambungkan LED merah ke GPIO26 melalui resistor dan GND.
* Salin dan tempelkan kode main.cpp ke editor Wokwi.
* Jalankan simulasi dan pastikan serial monitor menampilkan data suhu dan kelembapan.

1. Memindahkan hasil simulasi ke VSCode + PlatformIO

* Buka VSCode, lalu install ekstensi PlatformIO IDE melalui Extensionss
* Buka PlatformIO lalu klik New Project
* Pilih board esp32doit-devkit-v1, frameword Arduino, dan beri project name
* Setelah proyek dibuat, buka file src/main.cpp dan tempelkan isi kode dari simulasi Wokwi.

1. Menambahkan library dan menyambungkan ke Blynk

* Buka file platformio.ini, lalu tambahkan bagian berikut:

[env:esp32doit-devkit-v1]

platform = espressif32

board = esp32doit-devkit-v1

framework = arduino

lib\_deps =

    blynkkk/Blynk@^1.3.2

    beegee-tokyo/DHT sensor library for ESPx@^1.19

* Tekan Ctrl + S untuk menyimpan. PlatformIO akan otomatis mengunduh semua library tersebut.
* Pastikan token Blynk (BLYNK\_AUTH\_TOKEN) sudah sesuai dengan yang didapat dari Blynk.
* Hubungkan board ESP32 ke komputer melalui kabel USB.
* Klik ikon centang (✓) untuk Build, lalu klik ikon panah kanan (→) untuk Upload ke board
* Buka website Blynk <https://blynk.io/>
* Buat template dan simpan Template ID, Device Name, serta Auth tokennya.
* Tambahkan widget dengan datastreams V0 untuk suhu (gunakan Gauge atau Display), V1 untuk kelembapan, V2 untuk tombol LED, dan V3 untuk indikator LED

1. Jalankan aplikasi Blynk, pastikan:

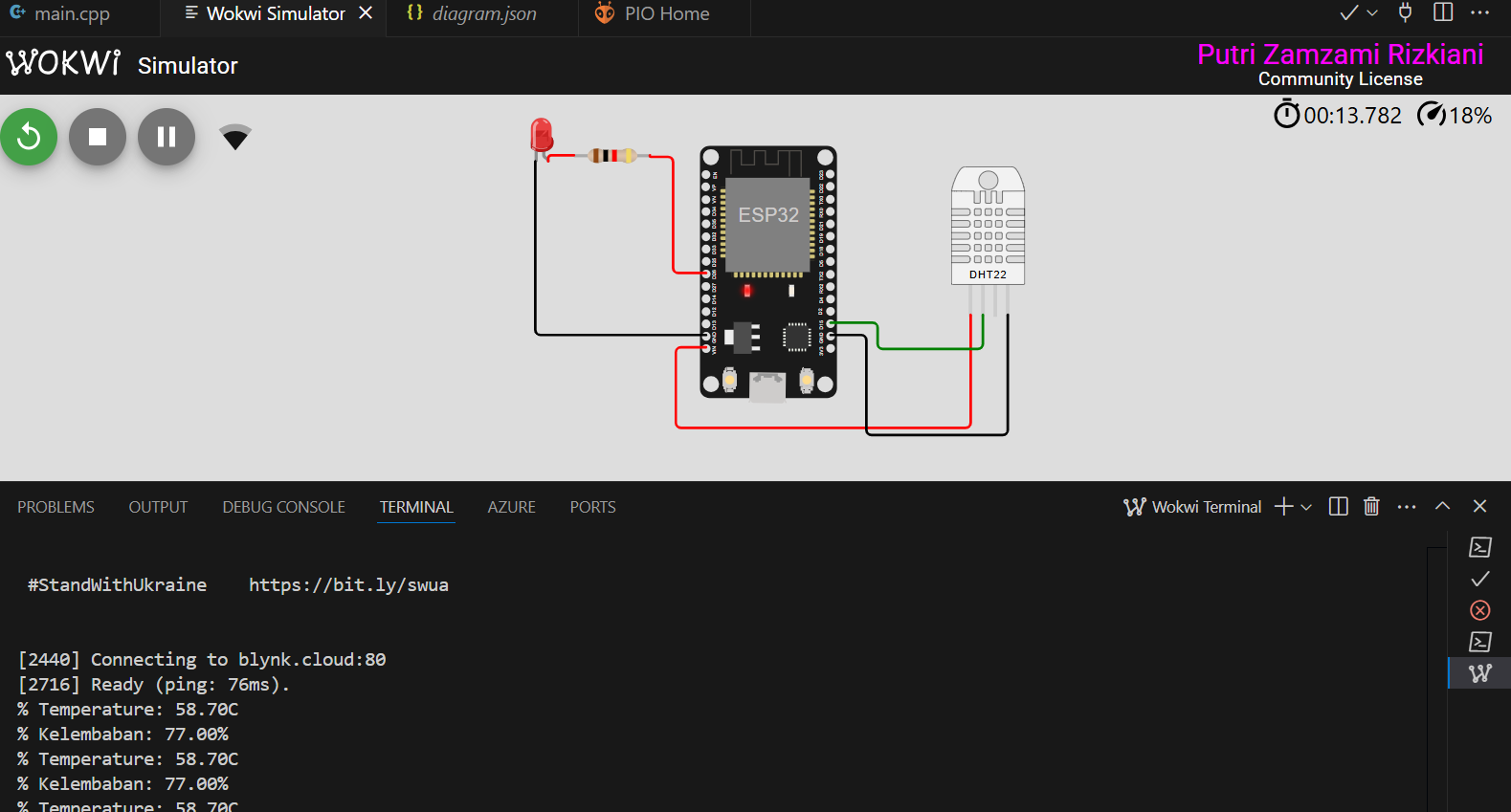
* Suhu dan kelembaban tampil di V0 dan V1 secara real-time.
* Tombol (V2) bisa menyalakan dan mematikan LED fisik di ESP32.
* Data juga tampil di Serial Monitor VSCode.

# BAB III

# HASIL DAN PEMBAHASAN

## Hasil Eksperimen

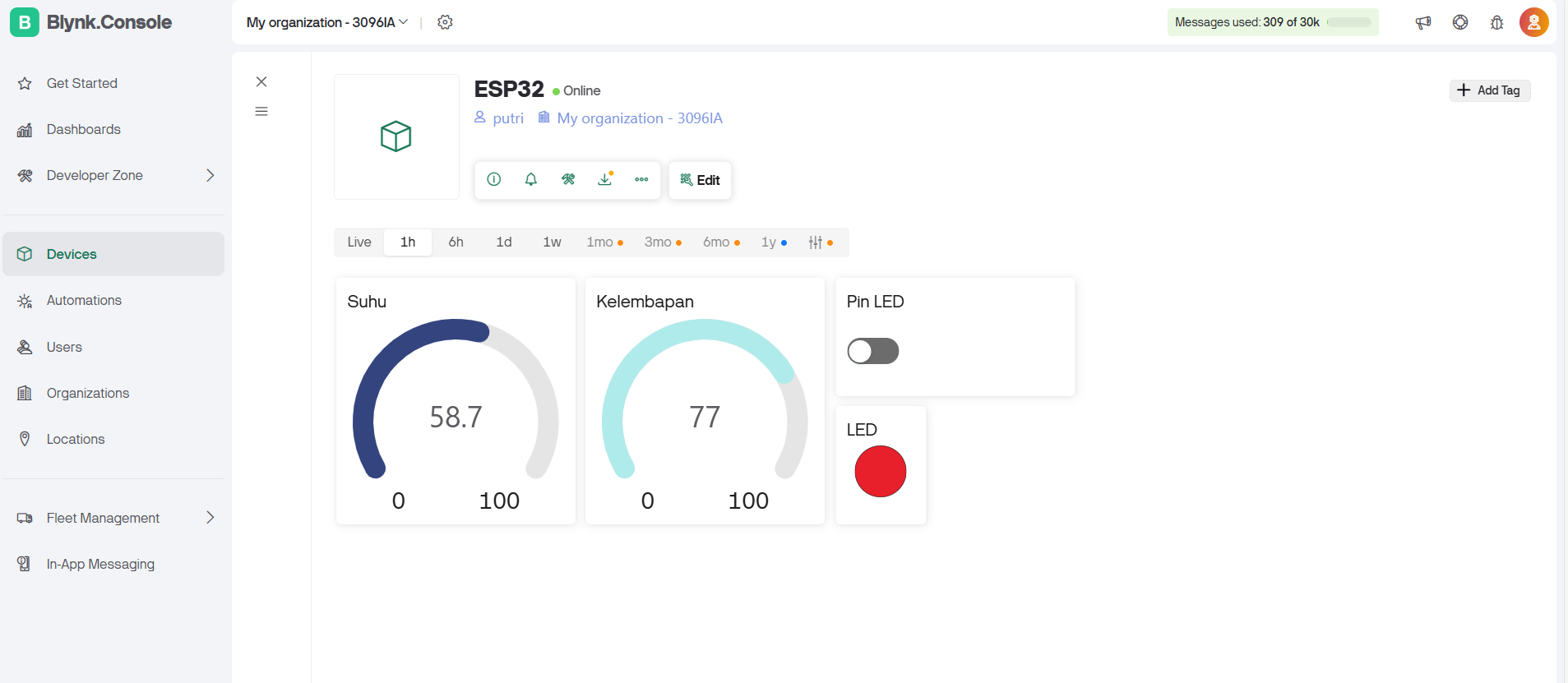
Berdasarkan hasil simulasi yang telah dilakukan di VSCode, dapat dilihat pada Gambar 3.1 menunjukkan tampilan simulasi sistem monitoring pada platform Wokwi. Rangkaian terdiri dari mikrokontroler ESP32 yang terhubung dengan sensor DHT22 dan sebuah LED merah melalui resistor. Sensor DHT22 terhubung ke pin GPIO15 untuk pengiriman data suhu dan kelembaban, sedangkan LED dikendalikan melalui GPIO26. Pada terminal simulator, terlihat bahwa ESP32 berhasil terkoneksi dengan server Blynk (ping: 76ms) dan mampu membaca serta menampilkan data suhu sebesar 58.78°C dan kelembaban sebesar 77.00%. Nilai tersebut dikirimkan secara berkala dan dicetak melalui Serial Monitor sebagai validasi fungsi sensor. Simulasi ini memastikan bahwa sistem telah berjalan sesuai skenario, baik dari sisi pembacaan sensor maupun pengendalian output.



(Gambar 3.1 Simulasi menggunakan VSCode)

Gambar 3.2 merupakan tampilan antarmuka Blynk Console setelah ESP32 berhasil dihubungkan dengan Blynk Cloud. Terdapat tiga widget utama pada dashboard: Gauge "Suhu", Gauge "Kelembapan", dan tombol "Pin LED". Data suhu ditampilkan pada pin virtual V0 dan menunjukkan nilai sebesar 58.7°C, sementara kelembaban ditampilkan pada V1 dengan nilai 77%. Widget tombol (V2) berfungsi untuk mengendalikan LED yang terhubung ke ESP32. Selain itu, terdapat indikator visual (V3) berbentuk lingkaran merah yang menampilkan status LED secara real-time, di mana kondisi "off" ditunjukkan dengan warna tetap merah namun tidak aktif.

Dari kedua gambar tersebut dapat disimpulkan bahwa sistem telah berhasil menjalankan fungsi utama yaitu pembacaan dan pengiriman data sensor ke cloud secara real-time, serta mampu menerima perintah dari pengguna untuk mengendalikan perangkat output. Integrasi antara simulasi awal (Wokwi) dan implementasi cloud (Blynk) berjalan dengan baik, dan sistem mampu merepresentasikan alur kerja sederhana dari perangkat IoT monitoring.



(Gambar 3.2 Tampilan Monitoring di Blynk)

## Kode Program

### 3.2.1 Main.cpp

#define BLYNK\_DEVICE\_NAME "ESP32"

#define BLYNK\_PRINT Serial

#define BLYNK\_TEMPLATE\_ID "TMPL6\_GHw-7xw"

#define BLYNK\_TEMPLATE\_NAME "ESP32"

#define BLYNK\_AUTH\_TOKEN "brf7B2SqUu5FTLuz\_2XMxVCkmm-b6O9J"

#include <WiFi.h>

#include <BlynkSimpleEsp32.h>

#include <DHTesp.h> //Library untuk DHT

char auth[] = BLYNK\_AUTH\_TOKEN ; //Auth Token

char ssid[] = "Wokwi-GUEST"; //nama hotspot yang digunakan

char pass[] = ""; //password hotspot yang digunakan

const int DHT\_PIN = 15;

int value0, value1, value2, value3, value6;

byte LED\_R = 26;

byte LED\_Y = 27;

byte LED\_G = 14;

byte LED\_B = 12;

DHTesp dht;

BlynkTimer timer;

//function untuk pengiriman sensor

void sendSensor()

{

 TempAndHumidity  data = dht.getTempAndHumidity();

//menampilkan temperature pada Serial monitor

Serial.print("% Temperature: ");

Serial.print(data.temperature);

Serial.println("C ");

Serial.print("% Kelembaban: ");

Serial.print(data.humidity);

Serial.println("% ");

Blynk.virtualWrite(V0, data.temperature); //mengirimkan data temperatur ke Virtual pin VO di Blynk Cloud

Blynk.virtualWrite(V1, data.humidity); //mengirimkan data kelembaban ke Virtual pin V1 di Blynk Cloud

}

BLYNK\_WRITE(V2)

{

  int nilaiBacaIO =param.asInt();

   digitalWrite(LED\_R, nilaiBacaIO);

  Blynk.virtualWrite(V3, nilaiBacaIO);

}

void setup()

{

// Debug console

Serial.begin(115200); //serial monitor menggunakan bautrate 9600

dht.setup(DHT\_PIN, DHTesp::DHT22);

pinMode(LED\_R, OUTPUT);

Blynk.begin(auth, ssid, pass); //memulai Blynk

timer.setInterval(1000, sendSensor); //Mengaktifkan timer untuk pengiriman data 1000ms

}

void loop()

{

Blynk.run(); //menjalankan blynk

timer.run(); //menjalankan timer

}

### 3.2.2 Diagram.json

  {

  "version": 1,

  "author": "Anonymous maker",

  "editor": "wokwi",

  "parts": [

    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -278.9, "left": 52.76, "attrs": {} },

    {

      "type": "wokwi-led",

      "id": "led1",

      "top": -306.4,

      "left": -89.47,

      "attrs": { "color": "red" }

    },

    {

      "type": "wokwi-resistor",

      "id": "r5",

      "top": -274.74,

      "left": -44.52,

      "attrs": { "value": "1000" }

    },

    {

      "type": "wokwi-dht22",

      "id": "dht1",

      "top": -260.42,

      "left": 247.56,

      "attrs": { "temperature": "58.7", "humidity": "77" }

    }

  ],

  "connections": [

    [ "esp:TX0", "$serialMonitor:RX", "", [] ],

    [ "esp:RX0", "$serialMonitor:TX", "", [] ],

    [ "led1:A", "r5:1", "red", [ "v0" ] ],

    [ "r5:2", "esp:D26", "red", [ "v1.2", "h17.93", "v81.46" ] ],

    [ "dht1:VCC", "esp:VIN", "red", [ "v87.6", "h-228.22", "v-54.65" ] ],

    [ "dht1:GND", "esp:GND.1", "black", [ "v93.06", "h-109.48", "v-76.5" ] ],

    [ "dht1:SDA", "esp:D15", "green", [ "v26.39", "h-81.44", "v-19.67" ] ],

    [ "led1:C", "esp:GND.2", "black", [ "v0" ] ]

  ],

  "dependencies": {}

}

# BAB IV

# KESIMPULAN

## Kesimpulan

## Berdasarkan hasil implementasi dan eksperimen yang telah dilakukan, ddapat disimpulkan bahwa proyek ini berhasil membangun sistem monitoring suhu dan kelembaban berbasis IoT menggunakan ESP32 dan sensor DHT22. Data berhasil dikirim ke aplikasi Blynk secara real-time dan ditampilkan melalui widget, serta dapat dikontrol melalui tombol virtual.

## Simulasi awal di Wokwi membantu memastikan sistem bekerja dengan baik sebelum diimplementasikan secara nyata di PlatformIO. Seluruh proses berjalan lancar, mulai dari pembacaan sensor, pengiriman data, hingga kontrol LED.

## 