

IMPLEMENTASI SISTEM PEMANTAUAN KUALITAS UDARA *REAL-TIME* BERBASIS *INTERNET OF THINGS* (IoT)

Studi Kasus: Implementasi ESP32 dengan Sensor DHT22 dan MQ-135 serta Visualisasi Data
pada Platform Blynk dan ThingSpeak

Disusun Oleh:

Muhammad Zaky Tabrani / 105222025 / zakitabrani1004@gmail.com

Felix Joshua Paulus / 105222032 / felixjoshua27@gmail.com



Laporan penelitian ini adalah sebagai bentuk
Ujian Akhir Semester (UAS)
untuk mata kuliah Mikrokontroller dan Internet of Things (IoT)

July 2025

Daftar Isi

Daftar Isi	ii
I. ABSTRAK	1
II. PENDAHULUAN.....	1
a. Latar Belakang.....	1
b. Tujuan dan Manfaat	2
III. METODE PENGEMBANGAN	3
a. Diagram Blok Sistem.....	3
b. Komponen Perangkat Keras dan Perangkat Lunak.....	4
c. Skema Rangkaian Elektronik.....	5
d. Alur Kerja Sistem	6
IV. HASIL DAN ANALISIS	7
V. KESIMPULAN DAN SARAN	13
a. Kesimpulan.....	13
b. Saran	13
VI. REFERENSI	13
VII. LAMPIRAN	15

I. ABSTRAK

Polusi udara menjadi tantangan serius bagi kesehatan masyarakat global dan lokal, sehingga menuntut adanya solusi pemantauan yang efektif dan dapat diakses. Proyek ini bertujuan untuk merancang dan membangun sebuah prototipe sistem pemantauan kualitas udara berbasis *Internet of Things* (IoT) yang mampu memberikan data secara *real-time*. Sistem ini menggunakan mikrokontroler ESP32 sebagai unit pemrosesan utama yang terhubung dengan sensor DHT22 untuk mengukur suhu dan kelembapan, serta sensor MQ-135 untuk mendeteksi tingkat polutan di udara. Data yang telah diakuisisi oleh sensor ditampilkan secara lokal pada layar LCD 16x2 dan secara bersamaan dikirimkan ke platform *cloud* ThingSpeak dan Blynk melalui koneksi WiFi untuk visualisasi dan pemantauan jarak jauh. Sebuah tantangan teknis pada ESP32, di mana modul WiFi dapat berkonflik dengan *Analog-to-Digital Converter* (ADC) yang digunakan oleh sensor MQ-135, diatasi dengan mengimplementasikan strategi manajemen koneksi, yaitu menonaktifkan WiFi sementara saat pembacaan sensor gas untuk memastikan akurasi data. Hasil pengujian menunjukkan bahwa sistem berhasil membaca, menampilkan, dan mengirimkan data secara konsisten. Prototipe ini membuktikan kelayakan sebuah sistem pemantau kualitas udara yang berbiaya rendah, otomatis, dan terhubung, serta memberikan fondasi untuk pengembangan lebih lanjut seperti penambahan notifikasi peringatan dan kalibrasi sensor yang lebih akurat.

Kata Kunci: *Internet of Things*, ESP32, Kualitas Udara, DHT22, MQ-135, Blynk, ThingSpeak.

II. PENDAHULUAN

a. Latar Belakang

Kualitas udara yang buruk merupakan masalah global yang serius. Menurut data WHO, hampir seluruh populasi dunia (99%) tinggal di wilayah dengan polusi udara yang melebihi batas aman, dengan sekitar 6,7 juta kematian pada tahun 2019 yang dihubungkan dengan paparan polusi udara [1]. Tidak hanya berdampak secara global, kondisi serupa terjadi secara lokal: misalnya pada Agustus 2024, Indeks Standar Pencemar Udara (ISPU) DKI Jakarta sempat mencapai 110 (kategori “tidak sehat”), menunjukkan tingginya konsentrasi partikel dan gas berbahaya di udara [3].

Peningkatan polusi udara di lingkungan perkotaan menimbulkan risiko kesehatan serius seperti penyakit pernapasan dan jantung [2]. Untuk melindungi kesehatan masyarakat, diperlukan sistem pemantauan kualitas udara yang *real-time* dan terjangkau [2]. Masalah utamanya adalah minimnya alat monitoring yang dapat mengukur parameter lingkungan secara otomatis, inilah yang ingin diatasi dengan sistem IoT. IoT memungkinkan penggunaan jaringan sensor cerdas yang terhubung

ke internet untuk mengumpulkan dan mengirim data lingkungan secara terus-menerus [4]. Sistem yang diusulkan menggunakan mikrokontroler (ESP32) dengan sensor DHT22 (suhu dan kelembapan) dan MQ-135 (konsentrasi gas polutan), mengirim data ke platform IoT (misalnya Blynk) untuk ditampilkan secara *real-time*. Dengan cara ini, perubahan kualitas udara dapat dipantau jarak jauh dan instan.

Sensor MQ-135 merupakan sensor gas semikonduktor berbasis SnO_2 yang umum digunakan dalam pemantauan kualitas udara. Sensor ini dapat mengenali berbagai gas berbahaya seperti amonia (NH_3), nitrogen oksida (NO_x), benzena, asap, dan alkohol [5]. Kemampuan deteksi yang luas ini membuat MQ-135 ideal untuk aplikasi pemantauan lingkungan, baik di dalam maupun luar ruangan. Sensor ini memiliki cakupan deteksi gas yang lebar serta respons yang cepat dan sensitif [6], sehingga sangat berguna untuk mendeteksi perubahan polutan di udara. Sensor MQ-135 bersama ESP32 berfungsi pada lapisan persepsi, mengumpulkan data polutan udara secara langsung [7]. Selanjutnya, pada lapisan komputasi tepi (*edge computing/gateway*) data dikumpulkan dan diolah. Setelah itu data yang sudah diproses akan dikirim ke platform *cloud* (lapisan aplikasi) untuk penyimpanan, visualisasi, serta analisis lebih lanjut.

b. Tujuan dan Manfaat

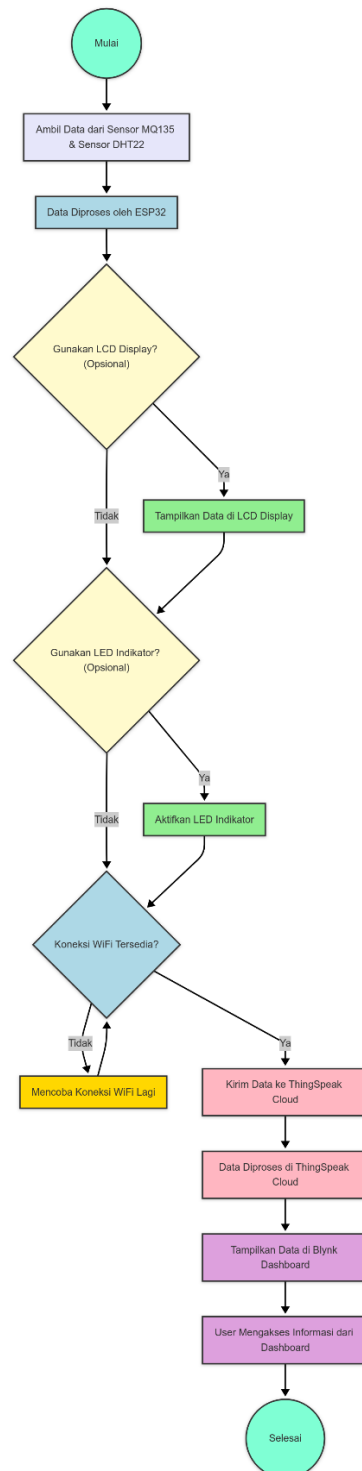
Tujuan utama proyek ini adalah merancang dan membangun prototipe sistem pemantauan kualitas udara berbasis IoT yang dapat bekerja secara otomatis dan *real-time*. Sistem ini diharapkan mampu memonitor berbagai parameter lingkungan (konsentrasi gas berbahaya, suhu, kelembapan) secara kontinu dan menampilkan hasilnya secara langsung pada aplikasi. Dengan prototipe tersebut, pengguna dapat dengan mudah mengetahui kondisi kualitas udara di sekitar mereka, misalnya di dalam ruangan rumah, atau ruang kantor, tanpa harus melakukan pengukuran manual. Demikian pula di lingkungan sekolah, pemantauan kualitas udara dalam ruang kelas dapat memastikan sirkulasi udara yang baik dan mencegah potensi gangguan kesehatan bagi siswa [8]. Seluruh rangkaian perangkat (sensor, mikrokontroler, dan aplikasi) diintegrasikan sehingga sistem bekerja *end-to-end* mulai dari pengambilan data hingga visualisasi hasil.

Manfaat dari alat ini bersifat langsung dan jangka panjang. Secara langsung, pengguna pribadi dapat menjaga kesehatan mereka dengan lebih proaktif, karena mereka mendapatkan informasi kualitas udara instan yang membantu menghindari paparan polutan berbahaya. Dengan adanya data kontinu, otoritas setempat dan pemangku kebijakan dapat mengambil langkah pengendalian polusi yang lebih efektif [4].

III. METODE PENGEMBANGAN

Bagian ini menjelaskan tahapan perancangan, komponen yang digunakan, serta alur kerja sistem pemantauan kualitas udara.

a. Diagram Blok Sistem



Sistem ini dirancang dengan arsitektur IoT tiga lapis. Penjelasan ini memperjelas gambar diagram alir yang telah dibuat dengan menunjukkan alur data paralel pada lapisan implementasi.

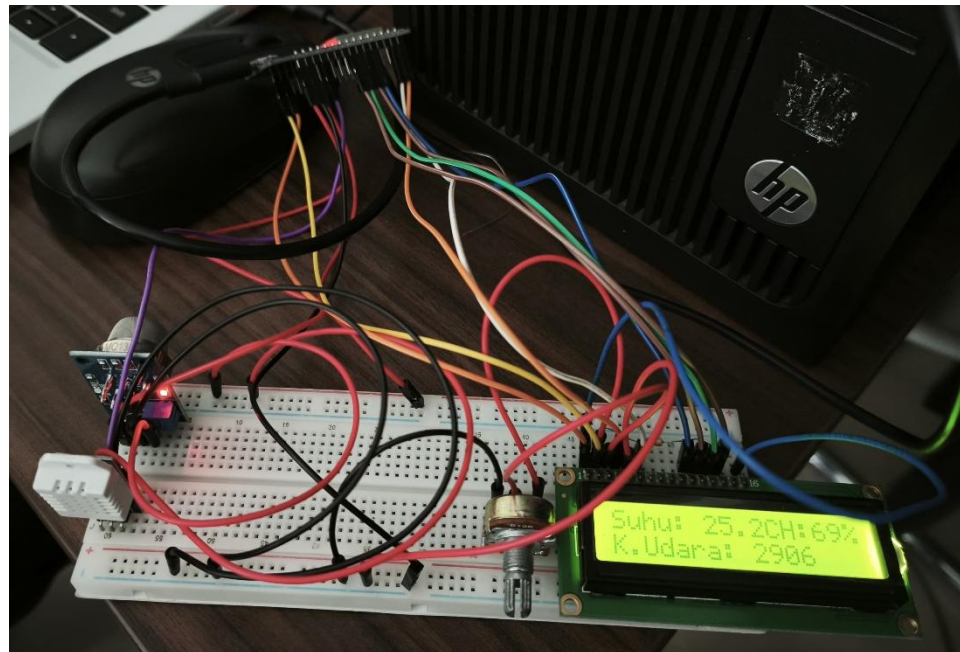
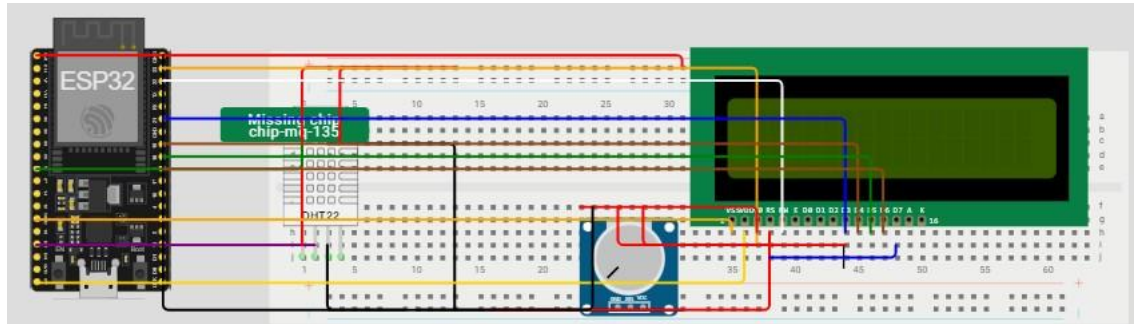
- **Lapisan Persepsi (*Input*):** Data lingkungan diakuisisi menggunakan dua sensor utama: Sensor DHT22 (suhu dan kelembapan) dan Sensor MQ-135 (kualitas udara).
- **Lapisan Jaringan & Gateway (*Processing*):**
Mikrokontroler ESP32: Berfungsi sebagai unit pemrosesan pusat. ESP32 mengolah data dari sensor, mengontrol tampilan pada LCD, dan mengelola konektivitas internet untuk mengirimkan data.
- **Lapisan Output:** Berbeda dengan alur sekuensial yang mungkin tersirat pada gambar diagram alir, pada implementasinya ESP32 mengirimkan data **secara paralel** ke beberapa media *output*:
 - **Tampilan Lokal (LCD 16x2):** Menampilkan data sensor secara langsung pada perangkat.
 - **Platform Cloud (Blynk & ThingSpeak):** Data dikirim secara terpisah dan bersamaan ke server Blynk dan ThingSpeak melalui koneksi WiFi untuk pemantauan jarak jauh dan visualisasi.

b. Komponen Perangkat Keras dan Perangkat Lunak

- **Perangkat Keras:**
 - **ESP32:** Mikrokontroler dengan WiFi dan Bluetooth internal yang berfungsi sebagai unit pemroses pusat.
 - **Sensor DHT22:** Sensor digital untuk mengukur suhu dan kelembapan relatif dengan akurasi tinggi.
 - **Sensor Gas MQ-135:** Sensor semikonduktor yang sensitif terhadap berbagai jenis gas seperti amonia (NH₃), NO_x, alkohol, benzena, dan asap. Nilai outputnya bersifat analog dan merepresentasikan kualitas udara secara umum.
 - **LCD 16x2:** Layar karakter alfanumerik untuk menampilkan informasi secara lokal.
 - **Breadboard dan Kabel Jumper:** Untuk merangkai semua komponen menjadi satu prototipe fungsional.
- **Perangkat Lunak:**
 - **MicroPython:** Firmware yang digunakan pada ESP32 untuk mempermudah pemrograman.
 - **Thonny IDE:** Lingkungan pengembangan terintegrasi (*Integrated Development Environment*) untuk menulis dan mengunggah kode MicroPython ke ESP32.

- **Platform Blynk:** Platform IoT untuk membuat antarmuka aplikasi seluler dengan *widget* visualisasi data seperti *gauge*.
- **Platform ThingSpeak:** Layanan *cloud* dari MathWorks untuk agregasi, visualisasi, dan analisis data IoT dalam bentuk grafik *time-series*.

c. Skema Rangkaian Elektronik



Komponen-komponen dihubungkan ke pin GPIO serta pin daya pada ESP32 sebagai berikut.

Komponen	Pin pada Komponen	Terhubung ke Pin ESP32	Keterangan
Sensor DHT22	VCC	3V3	Daya (+)
	DATA	GPIO 2	Pin Data Digital
	GND	GND	Ground (-)
Sensor MQ-135	VCC	3V3	Daya (+)
	A0 (Analog Out)	GPIO 26	Pin Output Analog
	GND	GND	Ground (-)

LCD 16x2	VCC	VIN	Daya (+)
	GND	GND	Ground (-)
	V0	(Pin tengah Potensiometer)	Pengatur Kontras Layar
	RS (Register Select)	GPIO 23	Kontrol Pin
	E (Enable)	GPIO 22	Kontrol Pin
	D4	GPIO 21	Pin Data 4
	D5	GPIO 19	Pin Data 5
	D6	GPIO 18	Pin Data 6
	D7	GPIO 5	Pin Data 7
	A (Anode)	VIN (melalui Resistor)	<i>Backlight</i> LED (+)
	K (Cathode)	GND	<i>Backlight</i> LED (-)

d. Alur Kerja Sistem

Berikut adalah alur kerja sistem yang berjalan pada perangkat:

- a. **Inisialisasi:** Saat dinyalakan(di-*run*), ESP32 menginisialisasi semua komponen (sensor dan LCD) dan mencoba terhubung ke jaringan WiFi. Status koneksi ditampilkan di LCD.
- b. **Akuisisi Data Sebagian:** Sistem membaca data suhu dan kelembapan dari sensor DHT22 terlebih dahulu.
- c. **Manajemen Koneksi untuk Akurasi Sensor:**
 - i. Untuk memastikan pembacaan sensor gas akurat, sistem **menonaktifkan koneksi WiFi untuk sementara**. Langkah ini krusial karena ADC yang digunakan sensor MQ-135 pada ESP32 dapat berkonflik dengan modul WiFi.
 - ii. Setelah WiFi mati, sistem membaca nilai analog dari sensor MQ-135.
- d. **Tampilan Data Lokal:** Semua data yang terkumpul (suhu, kelembapan, dan kualitas udara) ditampilkan secara lengkap pada layar LCD. Sistem memberikan jeda agar data mudah dibaca oleh pengguna.
- e. **Pengiriman Data ke Cloud:**
 - i. ESP32 mengaktifkan dan menyambungkan kembali WiFi.
 - ii. Jika koneksi berhasil, data dikirim secara **paralel** ke *endpoint* API ThingSpeak dan Blynk.
 - iii. Status pengiriman data (sukses atau gagal) akan diindikasikan pada layar LCD.
- f. **Siklus Berikutnya:** Sistem akan menunggu selama beberapa detik sebelum mengulangi seluruh siklus dari langkah nomor 2.

IV. HASIL DAN ANALISIS

- **Output dari Terminal (Thonny)**

```
..
Terhubung ke WiFi!
Alamat IP: 192.168.127.198
Koneksi WiFi stabil, mengirim data ke cloud...
Mengirim data ke Blynk...
☑ Data terkirim ke Blynk
Mengirim data ke ThingSpeak...
ThingSpeak Server Response: 200 - 873

Siklus selesai. Menunggu 10 detik...
Membaca sensor DHT22...
Menonaktifkan WiFi untuk membaca MQ135...
Hasil Baca Sensor -> Suhu: 23.3, Lembab: 68.1, Udara: 3053
Menahan tampilan data di LCD selama 10 detik...
Mengaktifkan & menghubungkan kembali WiFi...
Mengaktifkan antarmuka WiFi...
Menghubungkan ke WiFi...
..
```

- Koneksi Jaringan:
Sistem berhasil terhubung ke jaringan WiFi dengan alamat IP lokal 192.168.127.198. Pesan “Koneksi WiFi stabil” menandakan bahwa tidak terjadi gangguan jaringan saat pengiriman data ke *cloud*.

- Pengiriman Data ke Cloud:
Data sensor berhasil dikirim ke dua platform berbeda:

Blynk: Terdapat konfirmasi “☑ Data terkirim ke Blynk” yang menandakan pengiriman ke virtual pin berhasil tanpa error.

ThingSpeak: Sistem menerima respons server HTTP 200–873, yang mengindikasikan keberhasilan pengiriman data dengan response code 200 dan ID entri (873).

- Pembacaan Sensor:
Sistem membaca data dari sensor DHT22 dan MQ135, menghasilkan output sebagai berikut:
Suhu: 23.3°C
Kelembaban: 68.1%
Kualitas udara (nilai MQ135): 3053

Nilai tersebut menunjukkan kondisi lingkungan yang cukup lembab, dengan suhu normal. Nilai kualitas udara 3053 tergolong tinggi dan berpotensi menunjukkan kondisi yang kurang sehat tergantung pada kalibrasi sensor MQ135.

- Manajemen Energi dan WiFi:
Setelah proses pembacaan, sistem menonaktifkan koneksi WiFi untuk menghemat

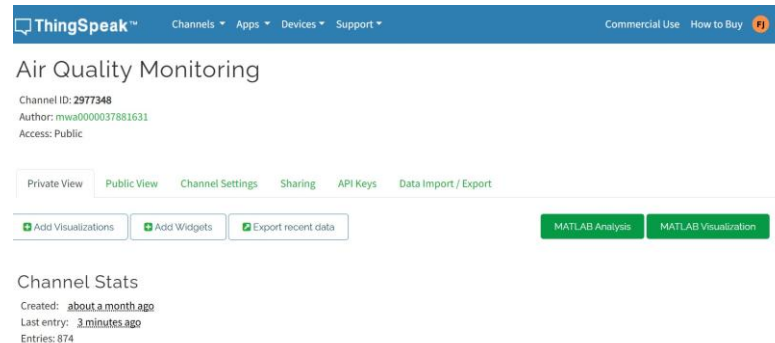
energi dan hanya mengaktifkannya kembali saat diperlukan. Hal ini tercermin dari output seperti “Menonaktifkan WiFi untuk membaca MQ135...” dan “Mengaktifkan & menghubungkan kembali WiFi...”.

- **Tampilan LCD:**



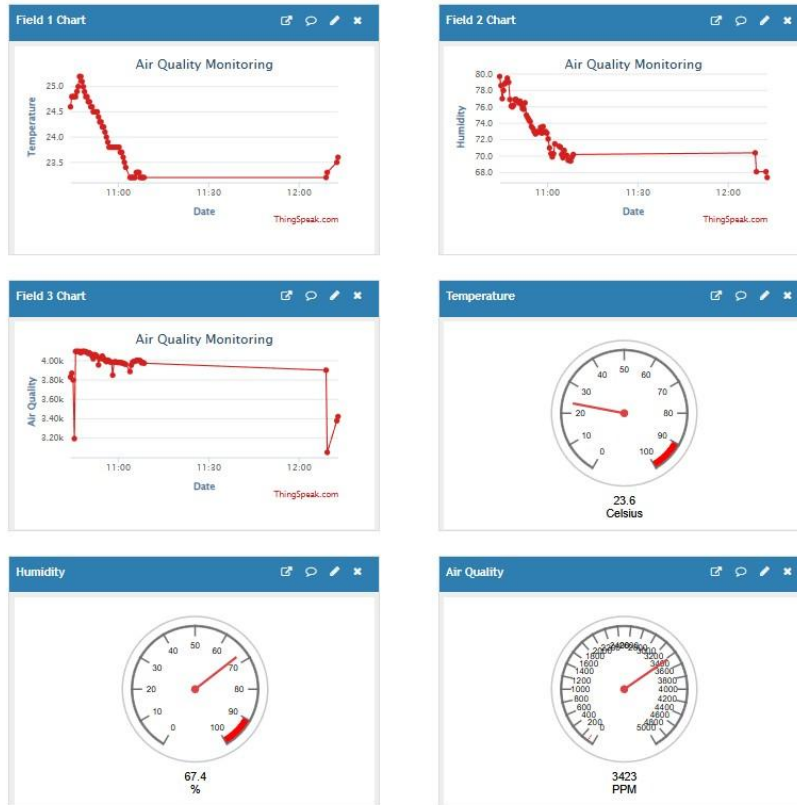
Informasi pembacaan sensor ditampilkan pada layar LCD selama 10 detik sebelum sistem masuk ke siklus berikutnya. Hal ini menunjukkan integrasi antarmuka pengguna secara lokal yang berjalan dengan baik.

- **Visualisasi Data di ThingSpeak**



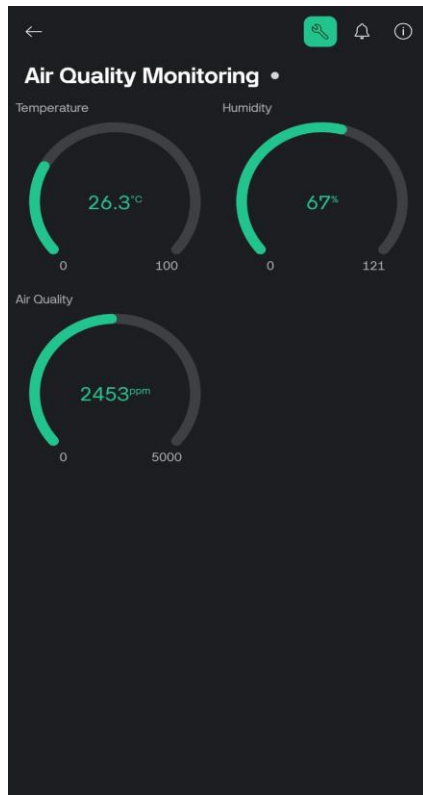
Platform ThingSpeak digunakan untuk memantau dan memvisualisasikan data sensor secara real-time melalui channel ID 2977348 yang bersifat publik. Dari informasi statistik channel, sistem telah berhasil mengirimkan sebanyak 874 entri dalam kurun waktu sekitar satu bulan. Data terakhir masuk 3 menit sebelum pengambilan tangkapan layar, menunjukkan bahwa sistem berjalan secara kontinu dan stabil.

Visualisasi data ditampilkan dalam bentuk grafik time-series dan gauge, dengan masing-masing field merepresentasikan parameter lingkungan berikut:



- Field 1 – Suhu (Temperature): Grafik menunjukkan tren suhu udara yang cenderung menurun dari sekitar 25°C menjadi sekitar 23.5°C, sebelum akhirnya naik kembali. Gauge menunjukkan nilai saat ini 23.6°C, yang masih tergolong dalam kisaran suhu ruangan normal.
- Field 2 – Kelembaban (Humidity): Grafik kelembaban memperlihatkan penurunan bertahap dari 78% menjadi sekitar 67–68%, kemudian stagnan pada titik tersebut. Gauge menunjukkan nilai terakhir yaitu 67.4%, yang masih dalam batas kelembaban normal di lingkungan dalam ruangan.
- Field 3 – Kualitas Udara (Air Quality/MQ135): Nilai kualitas udara bervariasi antara 4800 hingga turun mendekati 3200 PPM. Gauge menunjukkan nilai terakhir 3423 PPM.

- **Visualisasi Data di Blynk**



Selain ThingSpeak, sistem juga mengirimkan data secara real-time ke aplikasi Blynk melalui koneksi internet dan virtual pin. Antarmuka Blynk menampilkan tiga indikator utama dalam bentuk gauge digital:

Suhu: 26.3°C

Kelembaban: 67%

Kualitas Udara: 2453 ppm

Perbedaan nilai dengan ThingSpeak (misalnya 2453 PPM di Blynk vs. 3423 PPM di ThingSpeak) disebabkan oleh jeda waktu pengambilan data. Namun, nilai-nilai tersebut tetap menunjukkan konsistensi data dan performa sistem dalam mengirimkan pembacaan sensor ke dua platform secara bersamaan.

- **Interpretasi Parameter Lingkungan**

Sistem ini memantau tiga parameter utama kualitas udara, yaitu suhu, kelembaban, dan kualitas udara berbasis sensor. Setiap data dikirim melalui virtual pin (VPin) yang sesuai di Blynk, Thingspeak dan dipetakan sebagai berikut:

Parameter	Sensor	Satuan	Virtual (Blynk & ThingSpeak)	Pin
Suhu	DHT22	°C (Celsius)	V0	
Kelembaban	DHT22	% (Percentage)	V1	
Kualitas Udara	MQ135	ppm (parts per million)	V2	

1. Suhu (Temperature)

- Sensor: DHT22
- Satuan: Derajat Celsius (°C)
- Penjelasan: Suhu menunjukkan seberapa panas atau dingin lingkungan sekitar.

2. Kelembaban (Humidity)

- Sensor: DHT22
- Satuan: Persentase kelembaban relatif (% RH)
- Penjelasan: Menggambarkan banyaknya uap air di udara.

3. Kualitas Udara (Air Quality)

- Sensor: MQ135
- Satuan: ppm (parts per million)
- Penjelasan: Sensor ini mendeteksi konsentrasi berbagai gas. Nilai PPM yang tinggi (>3000) menunjukkan kualitas udara menurun. Karena saat dilakukan uji percobaan di ruangan ber-AC kualitas udara stabil di 2000-2500 dan saat dilakukan tes menggunakan korek api kualitas udara berada di angka > 3000.

- **Riwayat Kualitas Udara (Web Lokal)**

Selain visualisasi melalui ThingSpeak dan Blynk, sistem ini juga menampilkan data secara lokal melalui halaman web berbasis HTML yang terhubung langsung ke perangkat ESP32. Halaman ini menampilkan tabel dengan judul “Riwayat Kualitas Udara” yang berfungsi sebagai log data sensor.

Riwayat Kualitas Udara				
Waktu	Suhu (°C)	Kelembaban (%)	Kualitas Udara (ppm)	Status
10.30.07	24.5	77.2	3319	Kualitas udara buruk
10.29.42	24.6	77.4	3331	Kualitas udara buruk
10.29.15	24.7	76.6	3376	Kualitas udara buruk
10.28.49	24.8	75.9	3603	Kualitas udara buruk
10.28.24	25.0	75.3	3552	Kualitas udara buruk
10.27.59	25.2	74.6	3570	Kualitas udara buruk
10.27.31	25.4	74.4	3472	Kualitas udara buruk
10.26.41	26.0	71.9	3611	Kualitas udara buruk
10.26.17	26.4	70.6	3599	Kualitas udara buruk

Karakteristik Tabel Riwayat:

- Menampilkan 20 entri data terakhir dalam bentuk tabel.
- Data terbaru selalu ditampilkan di baris teratas.
- Ketika data baru masuk, entri paling bawah akan dihapus secara otomatis, sehingga hanya data terkini yang ditampilkan.
- Interval pengiriman data adalah setiap 20–30 detik, sesuai dengan jeda pembacaan dan pengiriman dari ESP32.

Isi Kolom:

Kolom	Keterangan
Waktu	Menunjukkan jam:menit:detik saat data dicatat.
Suhu (°C)	Diambil dari sensor DHT22, satuan derajat Celsius.
Kelembaban (%)	Dihasilkan oleh DHT22, satuan kelembaban relatif.
Kualitas Udara (ppm)	Dihasilkan oleh sensor MQ135, menunjukkan kadar gas/polutan.
Status	Otomatis diklasifikasi menjadi “Kualitas udara buruk” jika > 3000 ppm dan “Normal” jika < 3000 ppm

Kelebihan Visualisasi Lokal:

- Tidak memerlukan internet.
- Responsif dan real-time.

- Cocok digunakan di lingkungan terbatas (laboratorium, ruang kerja, atau kelas).
- Interval 20–30 detik memberikan cukup frekuensi untuk pemantauan lingkungan secara dinamis tanpa membebani sistem.

V. KESIMPULAN DAN SARAN

a. Kesimpulan

Berdasarkan hasil dan analisis, dapat ditarik beberapa kesimpulan utama. Pertama, prototipe sistem pemantauan kualitas udara berbasis IoT telah berhasil dibangun dan berfungsi sesuai dengan rancangan. Sistem mampu mengakuisisi data suhu, kelembapan, dan kualitas udara secara otomatis dan kontinu. Kedua, integrasi sistem berjalan dengan baik, mencakup pembacaan data sensor, penampilan informasi secara lokal pada layar LCD, serta pengiriman data secara simultan ke dua platform *cloud* (Blynk dan ThingSpeak) untuk visualisasi dan pemantauan jarak jauh. Terakhir, sebuah pengetahuan penting yang diperoleh adalah keberhasilan implementasi strategi manajemen koneksi, dengan menonaktifkan modul WiFi sementara saat membaca sensor analog, untuk mengatasi konflik perangkat keras pada ESP32. Strategi ini terbukti krusial dalam memastikan stabilitas dan akurasi data sensor MQ-135.

b. Saran

Untuk pengembangan sistem di masa depan, beberapa peningkatan dapat dilakukan untuk meningkatkan performa dan fungsionalitas. Disarankan untuk melakukan kalibrasi pada sensor MQ-135 terhadap gas referensi standar agar nilai output yang bersifat kualitatif dapat dikonversi menjadi kuantitatif dengan satuan yang jelas (misalnya ppm CO₂). Selain itu, sistem dapat disempurnakan dengan menambahkan indikator LED fisik, seperti yang diilustrasikan pada diagram alir awal. LED ini dapat memberikan status visual secara cepat (misal, hijau untuk udara baik, kuning untuk sedang, merah untuk buruk), sehingga lebih intuitif bagi pengguna.

VI. REFERENSI

- [1] World Health Organization, “Types of pollutants,” *Air Quality and Health*, WHO, Jun. 2025. [Online]. Available: <https://www.who.int/teams/environment-climate-change-and-health/air-quality-and-health/health-impacts/types-of->

[pollutants#:~:text=%2A%20More%20than%2099,ambient%20air%20pollution%20each%20year](#). [Accessed: 2-Jul-2025].

[2] M. G. Arkhan and Z. R. S. Elsi, “Air Quality Monitoring System Based Internet of Things,” *Brilliance*, vol. 4, no. 2, pp. 669–673, Nov. 2024. <https://doi.org/10.47709/brilliance.v4i2.4924>

[3] I. Fadhlurrahman, “Sore ini polusi udara DKI Jakarta tercatat paling buruk di Indonesia,” *Databoks Katadata*, 13 Aug. 2024. [Online]. Available: <https://databoks.katadata.co.id/layanan-konsumen-kesehatan/statistik/79fe3f377fcc83b/sore-ini-polusi-udara-dki-jakarta-tercatat-paling-buruk-di-indonesia#:~:text=Kualitas%20Udara%20di%20DKI%20Jakarta,di%20DKI%20Jakarta%20sebesar%20110>. [Accessed: 2-Jul-2025].

[4] M. Alam, M. M. Islam, N. M. Nayan, and J. Uddin, “An IoT Based Real-Time Environmental Monitoring System for Developing Areas,” *J. Adv. Res. Appl. Sci. Eng. Technol.*, vol. 52, no. 1, pp. 106–121, Oct. 2024, <https://doi.org/10.37934/araset.52.1.106121>

[5] Cirkuit Designer, “How to Use MQ-135 SENSOR AIR QUALITY – Examples, Pinouts, and Specs,” *Cirkuit Designer Docs*, [Online]. Available: <https://docs.cirkuitdesigner.com/component/aa757c75-ab00-4a76-af06-df8e80c6b816/mq-135-sensor-air-quality>. [Accessed: 2-Jul-2025].

[6] Olimex Ltd., “SNS-MQ135 Gas Sensor Technical Data,” *Olimex Ltd.*, Rev. A, Jan. 2015. [Online]. Available: <https://www.olimex.com/Products/Components/Sensors/Gas/SNS-MQ135/resources/SNS-MQ135.pdf>. [Accessed: 2-Jul-2025].

[7] Z. Idrees, Z. Zou, and L. Zheng, “Edge computing based IoT architecture for low cost air pollution monitoring systems: A comprehensive system analysis, design considerations & development,” *Sensors*, vol. 18, no. 9, p. 3021, Sep. 2018. <https://doi.org/10.3390/s18093021>

[8] D. Antar, “IoT Smart Sensors Monitor School Indoor Air Quality to Keep Students Safe,” *Campus Safety Magazine*, Apr. 3, 2024. [Online]. Available: <https://www.campussafetymagazine.com/news/iot-smart-sensors-monitor-school-indoor-air-quality-to-keep-students-safe/132665/>. [Accessed: 2-Jul-2025].

VII. LAMPIRAN

Link Github (Keseluruhan) : <https://github.com/felixjoshua/air-quality-monitoring-system>

Link Source Code : <https://github.com/felixjoshua/air-quality-monitoring-system/tree/main/Kode>

Link Youtube (Video Demo) : https://youtu.be/zROQavme17w?si=qixW2itEGXcX_rjX