

# Aplikasi Simulasi IoT Untuk Smart Sistem Monitoring dan Data Logging Real Time Sistem Peringatan Kebakaran

Muhammad Iqbal

Universitas Telkom  
Jl. Telekomunikasi No.1, Bandung, Indonesia  
e-mail: miqbal@telkomuniversity.ac.id

**Abstrak** - Kebakaran merupakan ancaman serius yang memerlukan deteksi dini dan respons cepat untuk meminimalisir kerugian. Penelitian ini mengusulkan sistem peringatan kebakaran berbasis Internet of Things (IoT) yang menggabungkan sensor DHT22 untuk memantau suhu dan kelembapan secara real-time, LED sebagai indikator visual peringatan, serta platform cloud ThingSpeak untuk logging data dan visualisasi. Sistem ini dirancang menggunakan simulator Wokwi untuk memvalidasi prototipe sebelum implementasi fisik. Ketika sensor DHT22 mendeteksi kenaikan suhu melebihi ambang batas yang ditentukan, sistem akan mengaktifkan LED sebagai alarm lokal dan mengirimkan data ke ThingSpeak melalui antarmuka WiFi. Hasil pengujian menunjukkan bahwa sistem mampu memberikan respons cepat (dalam hitungan detik) terhadap perubahan suhu ekstrem, dengan akurasi pembacaan sensor mencapai  $\pm 2$  °C. ThingSpeak berhasil menampilkan data secara konsisten.

Kata Kunci: IoT, kebakaran, real-time, monitoring

## PENDAHULUAN

### Latar Belakang

Sistem peringatan kebakaran merupakan salah satu elemen penting dalam menjaga keselamatan baik bagi individu maupun aset yang ada di suatu tempat. Kebakaran dapat terjadi secara tiba-tiba dan dapat menyebabkan kerusakan yang sangat besar jika tidak ditangani dengan cepat. Oleh karena itu, sistem peringatan dini yang dapat memberikan informasi secara cepat dan akurat sangat diperlukan. Sistem tradisional yang menggunakan alarm lokal sering kali memiliki keterbatasan dalam hal efisiensi waktu, keterjangkauan, dan pengawasan jarak jauh. Dalam perkembangan teknologi yang semakin pesat, terutama dalam bidang Internet of Things (IoT), solusi berbasis cloud computing menjadi semakin relevan untuk meningkatkan sistem peringatan kebakaran.

Sistem peringatan kebakaran berbasis cloud memanfaatkan sensor-sensor seperti sensor suhu (DHT22) dan sensor gas untuk mendeteksi kondisi yang dapat mengindikasikan kebakaran. Data yang diperoleh dari sensor ini dapat dikirimkan secara real-time ke platform cloud, seperti ThingSpeak, yang memungkinkan pemantauan dan pengelolaan data secara jarak jauh. Teknologi ini memungkinkan sistem peringatan kebakaran tidak hanya berfungsi secara lokal, tetapi juga memungkinkan pemantauan secara global, dengan memberikan notifikasi kepada pengguna melalui aplikasi berbasis web atau mobile. Selain itu, penggunaan ESP32 sebagai Micro Controller Unit (MCU) untuk menghubungkan perangkat IoT juga memberikan keuntungan dalam hal konsumsi daya rendah dan kemampuan komunikasi nirkabel yang handal.

### Studi Literatur

Berdasarkan penelitian ini (Octavianti dkk., 2024), Prototype Internet of Things (IoT) ini berhasil dalam memantau kelembapan kebun dan ketinggian air sawah menggunakan sensor DHT dan HC-SR04. Data dari sensor dikirim ke mikrokontroler ESP32, yang memprosesnya dan meneruskannya ke perangkat seperti LCD, Relay Module, dan platform Blynk melalui cloud. Sistem ini juga memungkinkan pengguna untuk mengontrol pompa air melalui motor yang terhubung ke mikrokontroler ESP32 dari cloud. Rangkaian ini diharapkan dapat menjadi dasar untuk pengembangan simulator pendeteksi ketinggian air dan kelembapan tanah di dunia nyata. Namun, pengujian langsung dengan perangkat fisik diperlukan untuk memvalidasi implementasi sistem ini.

Penelitian ini (Arif F & Indra K, 2025) membandingkan kinerja transmisi data antara ESP32 dan Raspberry Pi Pico menggunakan simulasi Wokwi. Hasilnya menunjukkan bahwa kedua mikrokontroler dapat mengelola transmisi data dengan baik, tetapi Raspberry Pi Pico memiliki waktu tunda transmisi yang lebih cepat daripada ESP32. Penelitian ini (Anggie M dkk., 2024) mensimulasikan sistem pemantauan suhu dan kelembapan menggunakan sensor DHT22 dan Arduino Uno pada platform Wokwi. Hasil simulasi menunjukkan bahwa sistem dapat mengukur suhu dan kelembapan dengan akurat, menampilkan data pada LCD, dan mengaktifkan relay dan LED jika suhu melebihi 25°C. Penelitian-penelitian terkait dengan Wokwi (Adhari Adiguna & Wisnu Widagdo, 2024; Rizki Prasetyo Tulodo dkk., 2023).



Penelitian ini membuat simulasi sistem pendeteksi kebakaran menggunakan sensor DHT22, LDR, dan potensiometer. Sistem ini dirancang untuk mendeteksi tanda awal kebakaran dan memberikan peringatan melalui buzzer dan notifikasi real-time di aplikasi Blynk. Hasil simulasi menunjukkan bahwa sistem bekerja dengan baik dalam mendeteksi potensi kebakaran. (Wibowo dkk., 2024). Simulasi sistem ini pendeteksi kebakaran menggunakan 3 sensor: DHT22, LDR, dan potensiometer. Sistem memberikan peringatan jika suhu di atas 55°C, nilai lux sensor LDR kurang dari 281, dan nilai potensiometer di atas 500. Beberapa penelitian sejenis dengan menggunakan ESP32 (Kautsar Permana dkk., 2025). Penelitian ini (Adam dkk., 2024) berfokus pada pengembangan sistem pelacakan kendaraan berbasis IoT yang terintegrasi dengan aplikasi Android. Sistem ini dirancang untuk memantau aktivitas kendaraan secara real-time, yang meliputi posisi, jarak tempuh, kecepatan, dan durasi perjalanan. Pengembangan penelitian yang dikaryakan menjadi pengabdian kepada masyarakat (Emerensiana Ngaga dkk., 2024; Khakim dkk., 2024). Penelitian ini membahas implementasi RESTful API (Ardhana dkk., 2023) menggunakan Laravel dan simulator IoT Wokwi untuk memonitor suhu dan kelembaban. Untuk mengatur langkah-langkah pengembangan program pengukuran suhu dan kelembaban, penelitian ini menggunakan metode waterfall. Diharapkan, hasil penelitian ini dapat memberikan panduan praktis kepada pengembang dalam membangun sistem pengukuran suhu dan kelembaban yang terintegrasi secara efektif dengan teknologi web dan IoT.

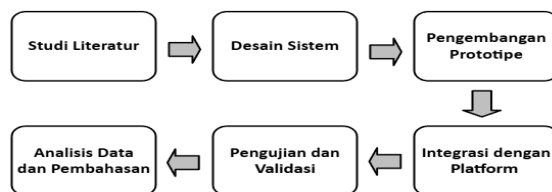
**Tujuan Penelitian**

Penelitian ini bertujuan untuk mengembangkan dan mengimplementasikan smart sistem monitoring dan data logging real-time untuk peringatan kebakaran berbasis cloud dengan menggunakan sensor DHT22, ESP32, LED, Simulator Wokwi dan platform cloud ThingSpeak. Sistem ini diharapkan dapat memberikan solusi yang lebih efektif dan efisien dalam pemantauan kebakaran, serta memungkinkan interaksi dan pemantauan jarak jauh, baik di rumah, kantor, maupun pabrik.

**METODE PENELITIAN**

Pada metode penelitian ini akan terbagi menjadi beberapa tahapan pelaksanaan. Mulai dari studi literatur, desain sistem, pengembangan prototipe, integrasi dengan platform, pengujian dan validasi serta analisis data dan pembahasan.

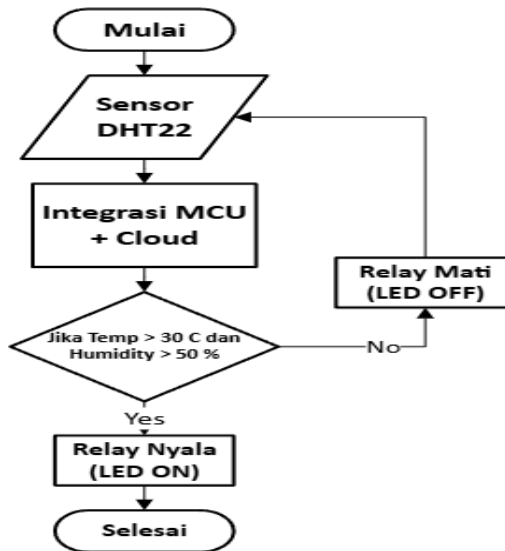
**Perancangan Sistem**



**Gambar 1. Diagram Metode Penelitian**

1. Studi Literatur: Melakukan penelusuran dan pemahaman terhadap teori dan penelitian terkait.
2. Perencanaan dan Perancangan Sistem: Merumuskan arsitektur sistem, pemilihan komponen, dan alur data.
3. Pengembangan Prototipe (Simulasi Wokwi): Membuat dan menguji prototipe virtual menggunakan platform Wokwi.
4. Integrasi dengan Platform ThinkSpeak: Menghubungkan prototipe Wokwi dengan platform IoT ThinkSpeak.
5. Pengujian dan Validasi: Melakukan serangkaian pengujian untuk memastikan sistem berfungsi dengan baik.
6. Analisis Data dan Pembahasan: Menganalisis data yang diperoleh dan membahas hasil pengujian.

**Flowchart Sistem**



**Gambar 2. Flowchart Sistem**

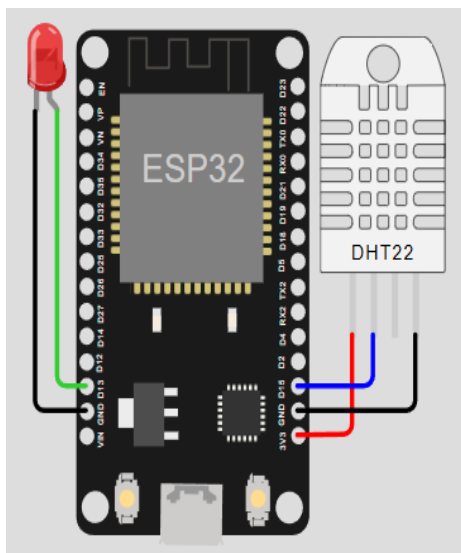
Flowchart ini menggambarkan alur kerja sebuah sistem yang menggunakan sensor DHT22 (kemungkinan besar untuk mengukur suhu dan kelembaban) untuk mengontrol status sebuah relay, yang dalam konteks ini diasumsikan terhubung dengan sebuah LED.

Sensor DHT22: Sistem akan membaca data dari sensor DHT22. Sensor ini bertugas mengukur suhu dan kelembaban lingkungan. DHT22 yang merupakan versi lain dari sensor DHT11 suhu dan kelembaban dengan spesifikasi yang sedikit berbeda

(biasanya rentang dan akurasi yang lebih baik). Integrasi MCU + Cloud: Data yang diperoleh dari sensor DHT22 kemudian diolah dan diintegrasikan dengan Mikrokontroler Unit (MCU) dan platform Cloud. Langkah ini mengindikasikan bahwa data suhu dan kelembapan tidak hanya dibaca tetapi juga kemungkinan dikirimkan dan diproses lebih lanjut di cloud untuk keperluan monitoring, analisis, atau kontrol jarak jauh. Jika Temp > 30 °C dan Humidity > 50 %: Ini adalah kondisi percabangan (decision). Sistem akan mengevaluasi apakah suhu yang terukur lebih besar dari 30 °C dan kelembapan yang terukur lebih besar dari 50 %. Yes: Jika kondisi pada langkah 4 terpenuhi (suhu di atas 30 °C dan kelembapan di atas 50 %), maka alur akan menuju ke langkah selanjutnya: Relay Nyala (LED ON): Relay akan aktif (ON), dan sebagai hasilnya, LED yang terhubung dengan relay akan menyala. No: Jika kondisi pada langkah 4 tidak terpenuhi (salah satu atau kedua kondisi tidak benar), maka alur akan menuju ke langkah Relay Mati (LED OFF): Relay akan non-aktif (OFF), dan sebagai hasilnya, LED yang terhubung dengan relay akan mati. Selesai: Ini adalah titik akhir dari alur program atau sistem setelah keputusan dan aksi (menyalakan atau mematikan relay/LED) telah dilakukan. Setelah mencapai "Selesai", sistem akan kembali ke langkah "Sensor DHT22" untuk terus memantau suhu dan kelembapan secara berkala (looping).

### Rangkaian Sistem Kerja

Pada gambar 3. Terkait dengan rangkaian sistem kerja, berikut merupakan penjelasan terkait dengan perangkat yang digunakan pada sistem :



Gambar 3. Rangkaian Sistem Kerja

LED jika kutub Positif (Anoda) yaitu Kabel berwarna hijau terhubung dari pin D13 pada ESP32 ke kaki yang lebih panjang dari LED (anoda). Pin D13 adalah salah satu pin Digital Input/Output (GPIO) pada ESP32. Sementara pada kutub Negatif (Katoda) yaitu Kaki yang lebih pendek dari LED (katoda) terhubung melalui kabel berwarna hitam ke pin GND (Ground)

pada ESP32. Ini adalah jalur tegangan negatif atau referensi nol volt untuk LED.

Sensor DHT22 memiliki tiga pin yang terhubung ke ESP32, VCC (Power) yaitu Kabel berwarna merah terhubung dari pin 3V3 (3.3 Volt) pada ESP32 ke pin VCC pada DHT22. Ini memberikan sumber daya tegangan 3.3V yang dibutuhkan oleh sensor untuk beroperasi. Data: Kabel berwarna biru terhubung dari pin D4 pada ESP32 ke pin data pada DHT22. Pin D4 adalah pin GPIO pada ESP32 yang digunakan untuk komunikasi data serial antara ESP32 dan sensor DHT22. GND (Ground): Kabel berwarna hitam terhubung dari pin GND (Ground) pada ESP32 ke pin GND pada DHT22. Ini adalah jalur tegangan negatif atau referensi nol volt untuk sensor.

### Cara Kerja Sensor DHT22

Sensor DHT22 bekerja berdasarkan prinsip fisika dan elektronika yang melibatkan perubahan kapasitansi (untuk kelembapan) dan resistansi (untuk suhu). Berikut adalah penjelasan lebih rinci mengenai cara kerja sensor ini:

Sensor kelembapan pada DHT22 menggunakan prinsip kapasitansi untuk mendeteksi tingkat kelembapan udara. Kapasitansi adalah kemampuan bahan untuk menyimpan muatan listrik, yang dapat berubah sesuai dengan kelembapan yang ada di udara. Sensor DHT22 memiliki bahan elektrostatis yang dapat menyerap kelembapan, yang akan menyebabkan perubahan pada nilai kapasitansi.

Kapasitansi (C): dipengaruhi oleh kelembapan dan dapat dihitung menggunakan rumus kapasitansi dasar:

$$C = \epsilon \times \frac{A}{d} \quad (1)$$

Di mana:

C = Kapasitansi (Farad)

$\epsilon$  = Permittivitas bahan (farad per meter)

A = Luas permukaan elektroda (meter persegi)

d = Jarak antara elektroda (meter)

Pada sensor DHT22, nilai kapasitansi berubah seiring dengan perubahan kelembapan. Sensor ini mengukur kapasitansi yang berubah untuk menghasilkan data yang dapat digunakan untuk menentukan tingkat kelembapan udara. Kelembapan dapat dihitung dengan menggunakan data kapasitansi yang diukur dan membandingkannya dengan referensi yang telah dikalibrasi. Kelembapan relatif (RH) dapat dihitung sebagai berikut:

$$RH = \frac{C_{diukur}}{C_{Maks}} \times 100 \quad (2)$$

Di mana:

RH = Kelembapan relatif dalam persen (%)

$C_{diukur}$  = Kapasitansi yang diukur oleh sensor

$C_{Maks}$  = Kapasitansi maksimum yang bisa dicapai oleh sensor pada kelembapan 100%.

Sensor suhu DHT22 bekerja dengan prinsip resistansi termistor (NTC – Negative Temperature Coefficient). Resistor termistor ini berubah resistansinya seiring dengan perubahan suhu di

sekitar sensor.

Hukum Ohm (untuk menghitung resistansi):

$$V = I \times R \quad (3)$$

Di mana:

V = Tegangan (Volt)

I = Arus (Ampere)

R = Resistansi (Ohm)

Sensor DHT22 menggunakan NTC thermistor, yang artinya resistansi elemen sensor akan menurun seiring dengan peningkatan suhu. Ketika suhu meningkat, nilai resistansi elemen termistor berkurang, dan ini menyebabkan arus yang mengalir dalam sirkuit juga berubah. Perubahan arus ini digunakan untuk menghitung suhu.

Untuk menghitung suhu, sensor DHT22 menggunakan rumus Steinhart-Hart untuk sensor termistor:

$$\frac{1}{T} = A + B \ln(R) + C (\ln(R))^3 \quad (4)$$

Di mana:

T = Suhu dalam Kelvin (K)

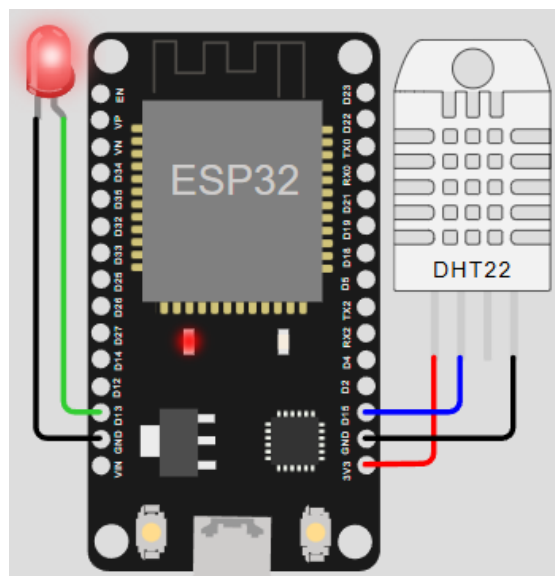
R = Resistansi termistor pada suhu tertentu

A,B,C = Parameter kalibrasi spesifik untuk sensor termistor

Setelah pengukuran kelembapan dan suhu dilakukan, hasilnya dikonversi menjadi nilai digital melalui konversi analog-ke-digital (ADC) dan dikirim ke mikrokontroler (ESP32) untuk diproses. Nilai digital ini kemudian dikirimkan ke sistem yang lebih besar, seperti platform cloud (misalnya ThingSpeak) atau perangkat output lain seperti LCD atau LED.

## HASIL DAN PEMBAHASAN

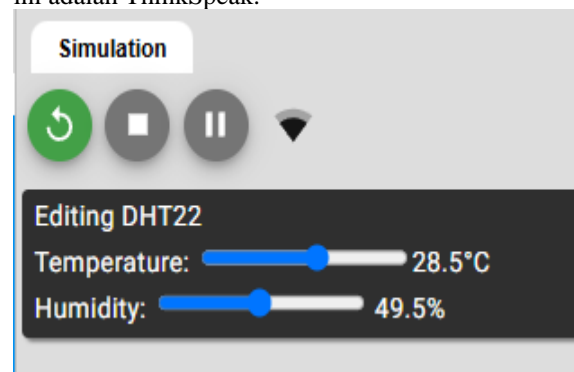
Setelah melakukan perancangan perangkat lunak dari pengaturan dan pemilihan komponen, penyusunan skematik rangkaian, langkah selanjutnya yaitu menjalankan simulasi rangkaiannya dengan mengatur parameter suhu dan kelembapan secara virtual pada sensor DHT22 untuk menguji respons sistem. Lalu amati apakah relay atau LED aktif ketika suhu atau kelembapan melebihi batas yang telah ditentukan yaitu tidak boleh melebihi  $temperature > 30^\circ C$  dan  $humidity > 50\%$ . Dengan akurasi pembacaan sensor hingga  $\pm 2^\circ C$ , sistem memberikan respons cepat (dalam hitungan detik) terhadap perubahan suhu ekstrem.



**Gambar 4. Hasil Rancangan Penelitian**

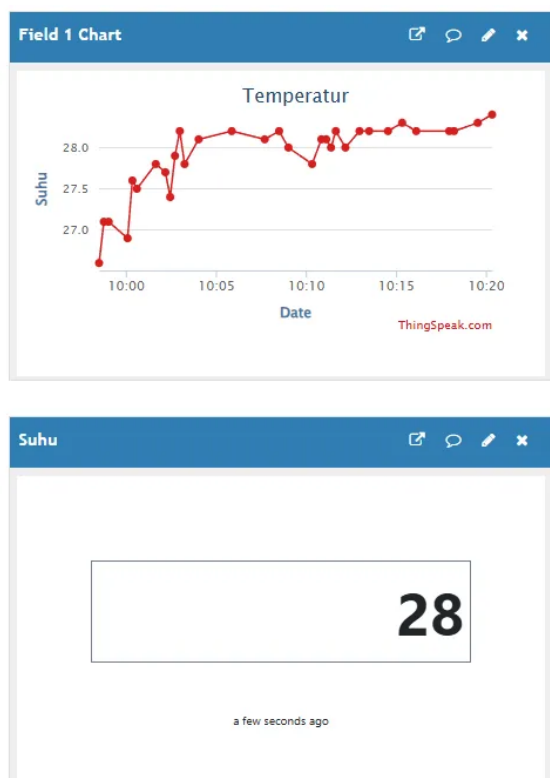
Pada Gambar 4. Skema ini bekerja dari *microcontroller unit* (MCU) khususnya ESP32, mengirimkan instruksi kepada sensor DHT22 dalam pengecekan berupa kelembapan dan suhu. Kondisi dikatakan aman apabila suhu  $< 30^\circ C$  dan kelembapan  $< 50\%$ , sehingga relay/LED tidak memberikan notifikasi atau warning. Sebaliknya, jika suhu dan kelembapan melebihi ambang batas, maka akan timbul alarm atau dalam hal ini LED akan menyala. Dalam kondisi lebih lanjut, relay ini dapat dihubungkan dalam memberikan notifikasi kepada pihak berwajib.

Dalam simulasi yang digunakan, terdapat fitur kontrol pada sensor DHT22 (**Gambar 5**) yaitu suhu dan kelembapan. Kontrol ini berguna dalam luaran yang dihasilkan pada jaringan cloud dalam hal ini adalah ThinkSpeak.



**Gambar 5. Kontrol Suhu dan Kelembapan**

Peneliti dapat menentukan nilai temperatur dan humidity pada aplikasi Wokwi yang digunakan pada prototipe ini, sehingga jika berjalan sesuai dengan kontrol dari peneliti sehingga dapat perubahan secara real time pada ThinkSpeak berjalan dengan baik.



Gambar 6. Hasil Pengukuran ThingSpeak

Suhu cenderung stabil dengan sedikit fluktuasi Gambar 6: Grafik "Temperatur" menunjukkan bahwa suhu berada di kisaran 27 hingga 28 °C selama periode waktu yang ditampilkan. Meskipun ada beberapa kenaikan dan penurunan kecil, tren secara keseluruhan relatif stabil. Pembacaan terakhir menunjukkan suhu di angka 28 °C. Kelembaban menunjukkan variasi yang lebih signifikan: Grafik "Humidity" memperlihatkan perubahan kelembaban yang lebih besar dibandingkan suhu. Kelembaban tampak menurun di awal periode, kemudian berfluktuasi antara 49% dan sekitar 50-51%. Pembacaan terakhir menunjukkan kelembaban di angka 49%. Pembacaan suhu dan kelembaban menunjukkan bahwa data ini merupakan pembacaan sensor secara *real-time* atau baru saja diperbarui.

Secara keseluruhan, data ini memberikan gambaran tentang kondisi lingkungan (suhu dan kelembaban) yang sedang dipantau oleh sistem IoT atau simulasi tersebut. Suhu relatif stabil, sementara kelembaban menunjukkan variasi yang lebih dinamis dalam periode waktu pengamatan.

#### KESIMPULAN

Penelitian ini mengembangkan sistem peringatan kebakaran berbasis cloud yang memanfaatkan teknologi Internet of Things (IoT) untuk monitoring suhu dan kelembapan secara real-time. Sistem ini mengintegrasikan sensor DHT22, ESP32, LED, dan platform ThingSpeak untuk mendeteksi potensi kebakaran dan memberikan notifikasi kepada pengguna. Melalui implementasi

IoT dan cloud computing, sistem ini mampu memberikan solusi yang lebih efisien, efektif, dan responsif dibandingkan dengan sistem tradisional. Hasil pengujian menunjukkan bahwa sistem peringatan kebakaran berbasis cloud dapat memberikan peringatan yang akurat dan cepat dalam mendeteksi perubahan suhu dan gas berbahaya. Selain itu, pengiriman data secara real-time ke platform cloud memungkinkan pemantauan jarak jauh, yang meningkatkan tingkat keselamatan di lokasi yang terhubung dengan sistem. Penggunaan LED sebagai indikator visual juga memberikan tanda yang jelas di tempat kejadian.

Sistem ini juga menunjukkan keandalan dalam komunikasi nirkabel menggunakan ESP32 dan cloud-based data logging, yang memastikan data dapat disimpan dan diakses kapan saja oleh pengguna. Dengan demikian, sistem peringatan kebakaran berbasis cloud ini dapat memberikan pengawasan lebih baik, notifikasi lebih cepat, dan penanganan lebih efisien terhadap kebakaran. Secara keseluruhan, penelitian ini membuktikan bahwa penerapan IoT dan cloud computing dalam sistem peringatan kebakaran tidak hanya meningkatkan efektivitas dan efisiensi sistem, tetapi juga membuka peluang untuk pengembangan lebih lanjut dalam berbagai sektor, termasuk rumah tinggal, gedung perkantoran, dan industri lainnya. Sistem ini sangat relevan dengan perkembangan teknologi saat ini yang mengutamakan kenyamanan, keamanan, dan pengawasan yang lebih baik melalui platform berbasis cloud.

#### REFERENSI

- Adam, M. F., Informatika, Y., Karawang, S., Ronggowaluyo, J. H., Timur, T., & Karawang, K. (2024). SIMULASI SISTEM DATA LOGGING BERBASIS INTERNET OF THINGS UNTUK MELACAK AKTIVITAS KENDARAAN DENGAN APLIKASI ANDROID. Dalam *Jurnal Mahasiswa Teknik Informatika* (Vol. 8, Nomor 5).
- Adhari Adiguna, M., & Wisnu Widagdo, B. (2024). Implementasi Media Belajar Internet of Things (IoT) untuk Deteksi Suhu dengan Buzzer Menggunakan Wokwi. *Journal of Informatics and Communications Technology*, 6(1), 86–97. <https://doi.org/10.52661>
- Anggie M, Naif B, Nur Fajri Faiz, Muhammad Rizki, & Didik Aribowo. (2024). Simulasi Pengukuran Suhu dan Kelembaban Ruangan Menggunakan Arduino Uno dan DHT22 pada Wokwi. *Jurnal Teknik Mesin, Industri, Elektro dan Informatika*, 3(4), 200–213. <https://doi.org/10.55606/jtmei.v3i4.4544>
- Ardhana, V. Y. P., Hidayat, M. T., Jannah, M., Sumiati, S., Rini, P., & Sari, N. (2023). Implementasi RESTful API Pada Laravel dan Simulator IoT Wokwi Untuk Pengukuran Suhu

- dan Kelembaban Menggunakan Metode Waterfall. *Arcitech: Journal of Computer Science and Artificial Intelligence*, 3(2), 93. <https://doi.org/10.29240/arcitech.v3i2.9334>
- Arif F, & Indra K. (2025). Studi Kinerja Transmisi Data Menggunakan ESP32 dan Raspberry Pi Pico Berbasis Simulasi Wokwi. *JOURNAL ZETROEM*.
- Emerensiana Ngaga, Yovinia Carmeneja Hoar Siki, Donatus Manehat, Yulianti Paula Bria, & Paskalis Andrianus Nani. (2024). PENGENALAN DAN PELATIHAN PEMROGRAMAN DASAR IOT DENGAN SIMULASI WOKWI BAGI SISWA TKJ SMKN NOEMUTI. *Jurnal Abdimas Bina Bangsa*.
- Kautsar Permana, A., Satria, A., Qurrata, H., Nurhadi, ain, Farras Belmino Tyto Putra, A., & Studi Teknologi Rekayasa Otomotif, P. (2025). SIMULASI CONDITION MONITORING MESIN BERBASIS IOT MENGGUNAKAN ESP32. *HEXAGON (Jurnal Teknik dan Sains)*, 6(1), 2721–3714.
- Khakim, L., Budihartono, E., Rakhman, A., & Sutanto, A. (2024). Pemanfaatan Aplikasi Wokwi sebagai Media Pembelajaran Mikrokontroler Berbasis Simulator di SMK Dinamika Kota Tegal. Dalam *Jurnal Abdimas PHB* (Vol. 7, Nomor 2).
- Octavianti, N. S., Rizky, M., Sulistiono, W. A., & Kusumantara, M. (2024). DESIGN OF WHEAT FIELD HUMIDITY TRACKING TOOLS AND PLANTATION WATER HEIGHT MEASURING, BASED ON SENSOR. Dalam *Jurnal Sistem Informasi Dan Bisnis Cerdas (SIBC)* (Vol. 17, Nomor 1).
- Rizki Prasetyo Tulodo, Nur Tulus Ujjianto, Eko Budiraharjo, & Yustia Hapsari. (2023). Pengembangan Sistem Pendeteksi Hujan Berbasis Internet of Things (IoT) dengan Simulasi Wokwi. *Jurnal Engineering*.
- Wibowo, S. A., Setiawan, A. A., Alvino, K., Nova, D., Kom, P. S., & Kom, M. (2024). Sistem Simulasi Pendeteksi Kebakaran Berbasis IoT Menggunakan Blynk. *Seminar Nasional Teknologi Informasi dan Bisnis (SENATIB)*, 2024.
- Ayuningtyas, A. A. (2022). *PENERAPAN INTERNET OF THINGS (IoT) DALAM UPAYA MEWUJUDKAN PERPUSTAKAAN DIGITAL DI ERA SOCIETY 5.0*.
- Kurniawan Ade, & Wilianto. (2018). *SEJARAH, CARA KERJA DAN MANFAAT INTERNET OF THINGS*.
- KURNIAWAN, M. I., SUNARYA, U., & TULLOH, R. (2018). Internet of Things : Sistem Keamanan Rumah berbasis Raspberry Pi dan Telegram Messenger. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 6(1), 1. <https://doi.org/10.26760/elkomika.v6i1.1>
- Sandi, G. H., & Fatma, Y. (2023). PEMANFAATAN TEKNOLOGI INTERNET OF THINGS (IOT) PADA BIDANG PERTANIAN. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 7, Issue 1).
- SURYANI, T. A., LINDAWATI, L., & FADHLI, M. (2024). Pengembangan Prototipe Sistem Smart Parking dengan Integrasi Parking Lock berbasis Internet of Things. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 12(3), 731. <https://doi.org/10.26760/elkomika.v12i3.731>