

# SISTEM PEMANTAU KUALITAS UDARA DALAM RUANG MENGGUNAKAN RASPBERRY PI DAN TELEGRAM

Manggala Fahreza<sup>1</sup>, Henry Candra<sup>2</sup>

<sup>1, 2</sup>Jurusan Teknik Elektro, Universitas Trisakti, Jakarta, Indonesia, 11440

<sup>1</sup>manggalafahreza@gmail.com, <sup>2</sup>henrycandra@trisakti.ac.id,

---

## Abstrak

Polusi tidak hanya terjadi di luar ruangan namun juga di dalam ruangan. Faktor terbesar penyebab buruknya kualitas udara dalam ruangan adalah ketidak lancarannya sirkulasi udara dan adanya zat polutan. Buruknya kualitas udara dalam ruangan dapat menimbulkan *Sick Building Syndrome* dan *Building Related Illness*. Agar menjaga kualitas udara dalam ruangan perlu adanya sistem *monitoring* untuk memantau zat-zat yang berbahaya di dalam ruangan khususnya dalam ruangan kerja. Dalam penelitian ini dilakukan perancangan sistem *monitoring* suhu, kelembapan, karbon monoksida, karbon dioksida dan partikel debu. Sistem menggunakan 4 sensor, 3 mikrokontroler NodeMCU dan 1 Raspberry Pi. NodeMCU mengirimkan notifikasi via Telegram saat nilai pembacaan sensor melebihi batas normal. Raspberry Pi berfungsi menerima data pembacaan sensor dari NodeMCU, menyimpan pada server *database* dan menampilkannya menggunakan aplikasi Grafana. Notifikasi via Telegram dapat membantu *user* untuk pemberitahuan dini status kualitas udara. Hasil pengamatan didapatkan kondisi zat karbon monoksida, karbon dioksida dan partikel debu memenuhi persyaratan sedangkan suhu dan kelembapan pada ruangan tersebut perlu disesuaikan kembali. Parameter *delay* pengiriman data yang diperoleh sistem adalah kurang dari 1 detik - 1 detik untuk NodeMCU 1 dan NodeMCU 2. Sedangkan *delay* NodeMCU 3 adalah 3 detik – 3,7 detik. Dari hasil yang diperoleh, sistem dapat digunakan untuk mengamati kualitas udara di dalam ruangan.

**Kata Kunci:** Polusi, Indoor, Raspberry Pi, Telegram

---

## Abstract

Pollution does not occur only outdoors, but also indoors. The biggest factor causing poor air quality indoor is lack of air circulation and the pollutants inside. Poor indoor air quality will cause Sick Building Syndrome or Building Related Illness. It is necessary to have an air quality monitoring system, especially in workspace. In this research, we design a monitoring system for getting information of temperature, humidity, concentrate of carbon monoxide, carbon dioxide and dust. The system uses 4 sensors, 3 microcontroller NodeMCU and 1 Raspberry Pi. NodeMCU will send data notification via Telegram when the sensor reading exceeds the threshold and send sensor data to Raspberry Pi and store it on the database server and display using Grafana. The result of the observations showed that the conditions of carbon monoxide, carbon dioxide and dust particles met the requirements while the temperature and humidity not required and needed to be readjusted in the room. Transmission system delay for NodeMCU 1 and NodeMCU 2 is less than 1 second - 1 second. Transmission system delay for NodeMCU 3 is 3 second – 3.7 second. From the results, the system can be used to observe indoor air quality.

**Key Words:** Pollution, Indoor, Raspberry Pi, Telegram

---

## 1. Pendahuluan

Saat ini, polusi udara sangat sulit dihindari. Terlebih pada kota-kota besar atau daerah industri. Polusi tidak hanya terjadi di luar ruangan saja (*outdoor*) namun juga di dalam ruangan (*indoor*). Menurut data *World Health Organization* (WHO), angka kematian akibat polusi udara mencapai 7 juta orang. Sekitar 3,3 juta diantaranya meninggal karena pengaruh udara yang buruk di dalam ruangan.

Faktor terbesar penyebab buruknya kualitas udara dalam ruangan adalah ketidak lancarannya sirkulasi udara

dan adanya zat polutan [1]. Kurangnya ventilasi membuat udara segar yang masuk ke dalam ruangan jadi lebih sedikit. Selain itu, jumlah karbon dioksida di sebuah gedung biasanya berhubungan dengan berapa banyak udara segar yang masuk ke dalam ruangan tersebut. *Air Conditioner* (AC) yang kotor dan penggunaan karpet akan menimbulkan debu dan partikel halus yang dapat mengganggu saluran pernapasan. Kelembapan dalam ruangan juga faktor yang dapat menimbulkan jamur, bakteri, mikroba hingga virus di dalam ruangan.

Buruknya kualitas udara dalam ruangan akan menimbulkan *Sick Bulding Syndrome* (SBS) [2], yang timbul dikarenakan ketidak-nyamanan penghuni terhadap keadaan bangunan yang ditempati. *Building Related Illness* (BRI). BRI merupakan situasi yang terjadi ketika penghuni gedung menderita suara penyakit disebabkan karena eksposur tertentu yang terjadi di dalam Gedung.

Penelitian [3] telah berhasil membuat sistem monitoring kualitas udara dalam ruangan dengan menggunakan platform IoT Blynk apps. Blynk apps terhubung secara tidak langsung ke prototipe alat pendeteksi kualitas udara melalui internet. Nilai yang terbaca pada sensor diproses dan jika memenuhi level tertentu maka sistem akan memberikan notifikasi kepada user melalui Blynk apps. Parameter sensor MQ135 yang diamati sebagai detektor kualitas udara meliputi Ammonia (NH<sub>3</sub>), Nitrogen Oxide (NO<sub>x</sub>), Alcohol, Benzena, Carbon Monoxide (CO), dan Carbon Dioxide (CO<sub>2</sub>). Sistem pemantauan yang dihasilkan baik dan sudah berbasis internet, kekurangannya tidak ada sensor debu, suhu dan kelembapan. Hasil pengujian hanya mengambil satu sample data dan kurang untuk grafik pemantauan harian, bulanan maupun tahunan, yang dapat memberikan rekomendasi mengenai kualitas udara.

Penelitian [4] telah berhasil melakukan pembuatan sistem monitoring kualitas udara dalam ruangan dan kontrol menggunakan fuzzy logic. Sistem menggunakan konsep IoT dengan mengamati konsentrasi karbon dioksida dan PM10 hasil dari pembacaan sensor akan ditampilkan secara grafik melalui dashboard. Mikrokontroler yang digunakan Arduino dan Raspberry Pi. Sistem fuzzy control digunakan untuk dapat mengatur interval kerja exhaust fan dengan otomatis tergantung dengan kondisi konsentrasi polutan sehingga dapat mengurangi level PM10 dan CO<sub>2</sub> serta membuat Air Quality Index (AQI) dalam level aman. Dari hasil yang diteliti, penelitian sudah dapat memonitor kondisi polutan secara grafik namun belum ada fitur notifikasi langsung kepada user terkait kondisi polutan.

Oleh karena itu, penelitian ini akan melakukan perancangan sistem monitoring kualitas udara dalam ruangan dengan menggunakan Raspberry pi dan Telegram Bot yang akan diimplementasikan pada ruangan Network Monitoring Center di PT. XXX. Ruang Network Monitoring Center merupakan ruangan yang selalu aktif 24 jam maka dari itu kualitas udara di dalam ruangan sangat perlu dijaga yang akan berkaitan dengan kesehatan pekerja dan produktivitas.

Sistem ini memonitor beberapa zat yaitu karbon monoksida (CO), karbon dioksida (CO<sub>2</sub>), partikel debu, temperatur dan kelembapan. Hasil dari sensor diproses pada mikrokontroler kemudian diamati secara grafik pada aplikasi Grafana dan notifikasi dari bot Telegram.

**Tabel 1. Ketrerangan NodeMCU Dengan Sensor**

Mikrokontroler	Jenis Sensor	Keterangan
NodeMCU 1	DHT11	Suhu dan Kelembapan (°C) dan (%)
	MQ-7	Konsentrasi CO (ppm)
NodeMCU 2	MQ-135	Konsentrasi CO <sub>2</sub> (ppm)
NodeMCU 3	GP2Y1010A U0F	Konsentrasi Partikel Debu (mg/m <sup>3</sup> )

Penggunaan Raspberry sebagai mini PC yang akan memproses data hasil pengukuran sensor ke aplikasi Grafana dan dijadikan sebagai acuan status sistem *online* atau *offline* dengan parameter ping. Diharapkan sistem ini dapat diimplementasikan untuk mengukur tingkat kualitas udara dalam ruangan kerja dan dapat dijadikan sebagai data analisis untuk mengembangkan konsep *green building*.

## 2. METODE PENELITIAN

Tahapan penelitian dibagi menjadi tiga tahap, tahap pertama menghubungkan mikrokontroler NodeMCU dengan sensor dan melihat pembacaan nilai data dari masing-masing sensor. Skema pembagian NodeMCU dengan sensor terdapat pada Tabel 1.

Tahap kedua, dilakukan pengiriman data sensor tersebut dari NodeMCU ke Raspberry Pi yang menjalankan tugas sebagai server. Pengiriman data dari Mikrokontroler NodeMCU ke Raspberry membutuhkan protokol pengiriman data yaitu MQTT (*Message Queuing Telemetry Transport*).

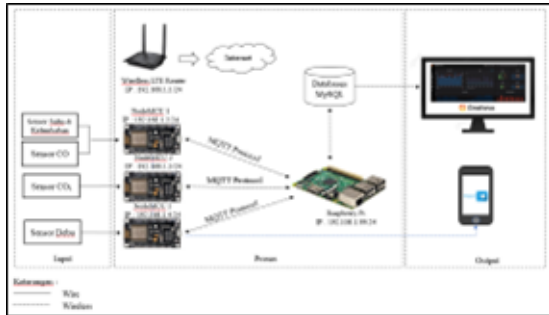
Tahap ketiga adalah pembuatan *server database*, Grafana dan notifikasi bot Telegram. Data yang telah diterima oleh Raspberry Pi dikumpulkan di dalam *database* MySQL untuk kemudian divisualisasikan menggunakan aplikasi Grafana. Jika ditemukan nilai pembacaan sensor yang melebihi nilai batas sesuai persyaratan dari Kementerian Kesehatan, maka NodeMCU akan mengirimkan notifikasi peringatan kepada user menggunakan Telegram bot.

Nilai batas normal zat yang ditentukan mengacu kepada keputusan Menteri Kesehatan Nomor 1405/MENKES/SK/XI/2002 [14] yang menetapkan persyaratan kesehatan lingkungan kerja perkantoran dan industri. Nilai batas normal atau yang direkomendasikan menyesuaikan dengan jenis zat pada penelitian ini seperti pada Tabel 2.

Telegram bot merupakan salah satu fitur *open source* yang disediakan oleh aplikasi Telegram yang kompatibel digunakan dalam pengembangan sistem *Internet of Things* seperti notifikasi peringatan ataupun menjalankan perintah otomatis dari *user*.

**Tabel 2. Nilai Batas Normal Zat**

Jenis Zat	Nilai Batas Normal
CO	< 25 <i>part per million</i> (ppm)
CO <sub>2</sub>	< 1000 <i>part per million</i> (ppm)
Partikel Debu	< 0,15 mg/m <sup>3</sup>
Suhu	18-28 °C
Kelembapan	40-60 %



**Gambar 1. Blok Diagram Sistem**

**3. PERANCANGAN SISTEM**

Rancangan sistem terdiri dari beberapa susunan blok yang terhubung dan saling terintegrasi. Blok diagram pada gambar di bawah memberikan gambaran rancangan sistem yang akan dibuat sesuai dengan tujuan penelitian.

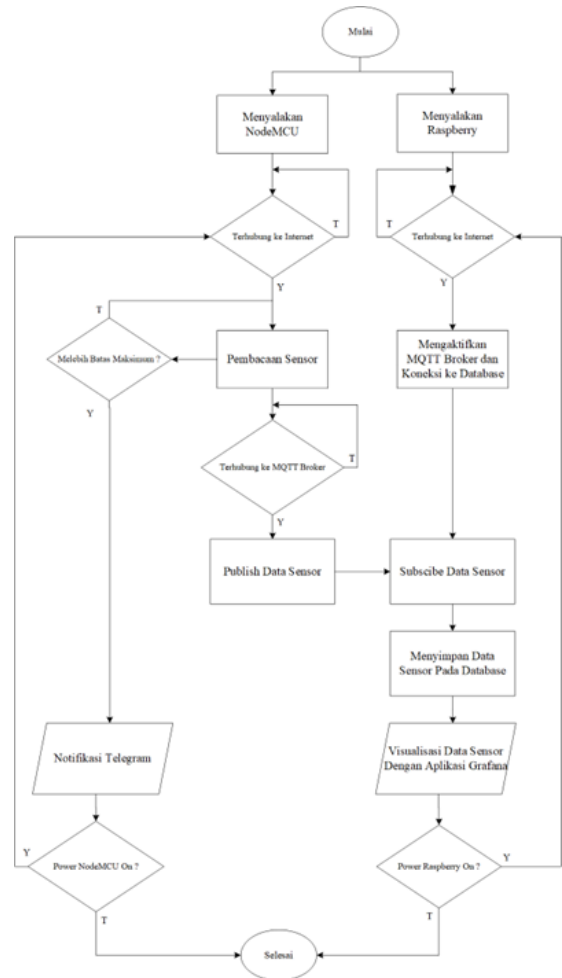
Gambar 1 menunjukkan rancangan blok diagram sistem yang akan dibuat. Selanjutnya dirancang diagram alir sistem yang berfungsi menggambarkan alur dari sistem yang bekerja sehingga dapat dipahami dan dijadikan acuan dalam penelitian.

Masing-masing sensor yang digunakan berfungsi mengukur besaran konsentrasi nilai zat yang ada di dalam ruangan. Perubahan nilai tegangan *input* dari sensor akan dikonversi oleh NodeMCU sehingga pembacaan dapat menggunakan aplikasi Arduino IDE pada serial monitor.

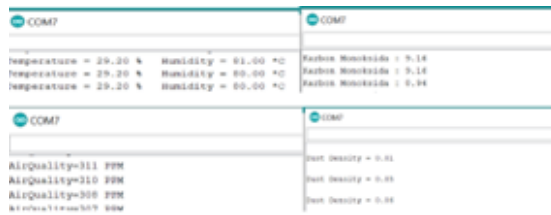
Setelah data zat didapatkan, selanjutnya data nilai zat tersebut dikirimkan oleh NodeMCU ke Raspberry Pi menggunakan protokol MQTT (*Message Queuing Telemetry Transport*). MQTT bekerja dengan pola publish dan *subscribe* [15]. NodeMCU bertindak sebagai *publisher* dan Raspberry Pi sebagai *subscriber*. MQTT memerlukan *broker* sebagai perantara pengiriman dan topik sebagai identitas data.

Data sensor yang sudah diterima oleh Raspberry Pi disimpan pada *database* MySQL. *Database* yang dibuat memiliki tabel penyimpanan untuk data sensor pada masing-masing NodeMCU dan penyimpanan data ping alamat IP NodeMCU.

Telegram bot dibuat dengan menggunakan bot



**Gambar 2. Diagram Alir Sistem**



**Gambar 3. Pembacaan Data Sensor Pada Serial Monitor**

**Tabel 3. Nilai Rata-Rata Pengamatan Pada NodeMCU 1**

Waktu	Suhu (°C)	Kelembapan (%)	CO (ppm)	Delay (detik)
Pagi	27,66	76,8	12,43	1
Siang	32,57	52,3	13,92	<1
Malam	30,49	65,7	14,7	1

father dengan perintah `/newbot`. Setelah itu, langkah selanjutnya adalah menentukan `username` bot dan mendapatkan token API yang digunakan untuk mengakses bot dari NodeMCU.

Penelitian ini melakukan pemantauan terhadap status koneksi NodeMCU. Raspberry Pi melakukan cek terhadap status koneksi NodeMCU dengan menjalankan program `request respon` ping ke alamat IP dari NodeMCU 1 NodeMCU 2 dan NodeMCU 3.

Integrasi Grafana dengan `database` dilakukan dengan cara memilih `database` yang digunakan yaitu MySQL dan memilih tabel yang ingin ditampilkan. Setelah terintegrasi, `user` dapat mengatur tampilan seperti warna, bentuk grafik dan penyajian data lainnya yang diinginkan.

**4. Pengujian Sistem**

Proses pengujian dilakukan dalam empat tahap, yaitu tahap pertama menampilkan data pembacaan nilai sensor. Tahap kedua adalah pengujian sistem dan pengamatan `delay`. Tahap ketiga pengujian visualisasi dengan grafana. Tahap keempat adalah pengujian pengiriman notifikasi via Telegram.

**4.1 Pengujian Pembacaan Sensor Dengan Serial Monitor**

Pada tahap pertama pengujian dimulai dengan menampilkan data pembacaan nilai sensor menggunakan `software` Arduino IDE. Gambar 3 menunjukkan proses pembacaan sensor suhu, kelembapan, konsentrasi karbon monoksida, karbon dioksida dan partikel debu.

**4.2 Pengujian sistem dan Pengamatan Delay**

Pengamatan bertujuan untuk mengetahui perbandingan hasil pembacaan sensor pada masing-masing zat dan delay pengiriman data sensor saat dikirimkan dari NodeMCU sampai dengan penerimaan pada `database`. Pengamatan dilakukan pada tanggal 16 Agustus 2021 sampai dengan 17 Agustus 2021 dengan tiga kondisi waktu yang berbeda, yaitu pada pagi, siang dan malam hari. Hasil pengamatan tertera pada Tabel 3, Tabel 4 dan Tabel 5.

Pada Tabel 3 merupakan hasil rata-rata pengamatan konsentrasi zat pada mikrokontroler NodeMCU 1.

**Tabel 4. Nilai Rata-Rata Pengamatan Pada NodeMCU 2**

Waktu	CO <sub>2</sub> (ppm)	Delay (detik)
Pagi	247,6	1
Siang	198,8	<1
Malam	14,7	1

Pada Tabel 4 merupakan hasil pengamatan rata-rata nilai konsentrasi karbon dioksida.

Tabel 5 menunjukkan hasil pengamatan rata-rata nilai konsentrasi partikel debu pada NodeMCU 3.

**4.3 Pengujian Visualisasi Grafana**

Pengujian visualisasi Grafana dengan membandingkan hasil data yang diperoleh pada `database` dan hasil visualisasi pada Grafana.

Pada Tabel 6 didapatkan sampel pembacaan nilai sensor pada tanggal 22 Juli 2021 untuk dibandingkan dengan hasil visualisasi pada Grafana.

Pada Gambar 4 terlihat hasil visualisasi untuk nilai suhu dan kelembapan sesuai data terakhir pada pukul 10:52:01 WIB.

Gambar 5 menunjukkan hasil visualisasi konsentrasi karbon monoksida yang disajikan berupa grafik realtime sesuai dengan waktu pada tabel mulai pukul 10:47:14 WIB sampai dengan 10:52:01 WIB.

Selanjutnya untuk pembacaan sensor karbon dioksida pada NodeMCU 2.

Pada Tabel 7 merupakan sampel hasil pembacaan sensor untuk karbon dioksida pada `database` tanggal 22 Juli 2021.

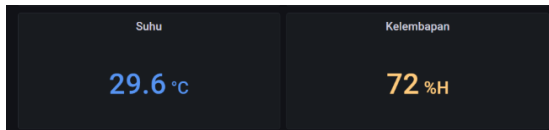
Pada Gambar 6 menunjukkan visualisasi pada Grafana seperti pada sampel `database` mulai pukul 13:44:32 WIB sampai dengan 13:49:00 WIB.

**Tabel 5. Nilai Rata-Rata Pengamatan NodeMCU 3**

Waktu	Debu (mg/m <sup>3</sup> )	Delay (detik)
Pagi	0,005	3
Siang	0,05	3,05
Malam	0,01	3,7

**Tabel 6. Database NodeMCU 1**

Suhu (°C)	Kelembapan (%)	CO (ppm)	Waktu (jam:menit:det)
29,2	73	8,62	10:47:14
29,2	73	8,97	10:48:32
29,4	73	8,72	10:49:37
29,6	73	8,53	10:50:46
29,6	72	8,40	10:52:01



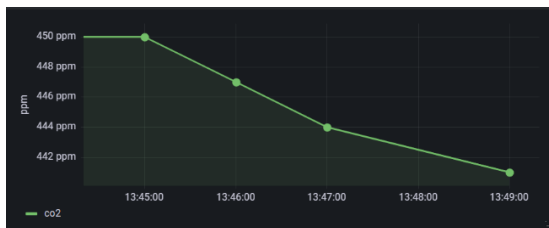
**Gambar 4. Visualisasi Grafana Suhu dan Kelembapan**



**Gambar 5. Visualisasi Grafana Karbon Monoksida**

**Tabel 7. Database NodeMCU 2**

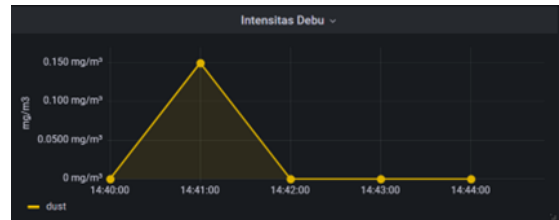
CO <sup>2</sup> (ppm)	Waktu (jam:menit:detik)
450	13:44:32
450	13:45:38
447	13:46:43
444	13:47:54
441	13:49:00



**Gambar 6. Visualisasi Grafana Karbon Dioksida**

**Tabel 8. Database NodeMCU 3**

Debu (mg/m <sup>3</sup> )	Waktu (jam:menit:detik)
0	14:40:22
0,15	14:41:26
0	14:42:29
0	14:43:33
0	13:44:37



**Gambar 7. Visualisasi Grafana Intensitas Debu**

Selanjutnya untuk pengujian pada NodeMCU 3 yang memiliki data sensor konsentrasi partikel debu.

Pada Tabel 8 merupakan sampel pengambilan data konsentrasi partikel debu pada *database* tanggal 22 Juli 2021.

Gambar 7 menunjukkan hasil visualisasi Grafana dengan sampel data yang diambil mulai pukul 10:40:22 WIB sampai dengan 13:44:37 WIB.

Pada Gambar 8 menunjukkan tampilan keseluruhan data sensor dan status NodeMCU pada dashboard Grafana.

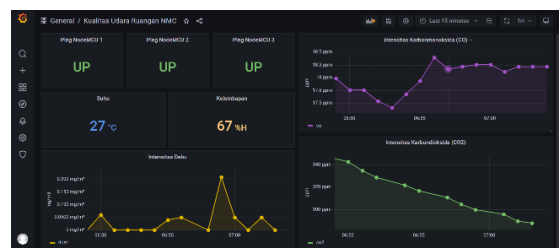
#### 4.4 Pengujian Notifikasi Telegram

Pengujian dilakukan dengan melakukan *adjustment* nilai pembacaan sensor dari nilai batas normal yang telah ditentukan pada Tabel 2. Jika nilai melebihi batas maka Telegram Bot mengirimkan notifikasi kepada *user* terkait status peringatan kualitas udara.

Gambar 9 menunjukkan hasil notifikasi Telegram saat pembacaan suhu melebihi batas yaitu 29 °C.

Gambar 10 menunjukkan hasil notifikasi Telegram saat pembacaan kelembapan melebihi batas yaitu 69%.

Gambar 11 menunjukkan hasil notifikasi Telegram saat pembacaan konsentrasi karbon monoksida melebihi batas normal yaitu 27 ppm.



**Gambar 8. Dashboard Monitoring Grafana**



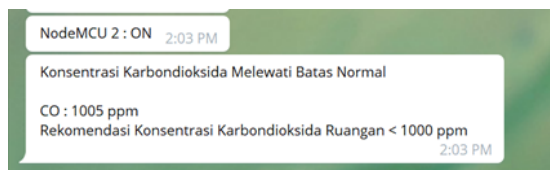
**Gambar 9. Informasi Telegram Bot Suhu Melebihi Batas Normal**



**Gambar 10. Informasi Telegram Bot Kelembapan Melebihi Batas Normal**



**Gambar 11. Informasi Telegram Bot Karbon Monoksida Melebihi Batas Normal**



**Gambar 12. Informasi Telegram Bot Karbon Dioksida Melebihi Batas Normal**



**Gambar 13. Informasi Telegram Bot Intensitas Debu Melebihi Batas Normal**

Gambar 12 menunjukkan hasil notifikasi Telegram saat pembacaan konsentrasi karbon dioksida melebihi batas normal yaitu 1005 ppm.

Gambar 13 menunjukkan hasil notifikasi Telegram saat pembacaan intensitas debu melebihi batas normal yaitu 0,25 mg/m<sup>3</sup>.

**5. ANALISA HASIL SIMULASI**

Setelah dilakukan pengujian pada tahap sebelumnya, dapat dianalisa hasil dari pembuatan dan pengujian sistem yang telah dilakukan. Pembacaan keempat sensor yang meliputi parameter zat suhu, kelembapan, konsentrasi, karbon monoksida, karbon dioksida dan partikel debu didapatkan hasil yang sesuai dengan nilai kondisi ruangan yang diamati tanpa *Air Conditioner* dan *Humidifier*.

Pada Pengujian 4.2 didapatkan hasil pengamatan parameter nilai sensor dan delay pada kondisi waktu pagi, siang dan malam hari. Hasil pengamatan pagi hari didapatkan rata-rata nilai suhu 27,6 °C dan kelembapan 76,8%. Mengacu kepada Tabel 2, nilai suhu masuk ke dalam *range* normal yang direkomendasikan antara 18-28 °C sedangkan kelembapan melebihi batas rekomendasi yaitu 40-60%. Selanjutnya konsentrasi karbon monoksida yang didapatkan rata-rata adalah 12,43 ppm dan masih dalam *range* normal yaitu di bawah 25 ppm.

Pengamatan pada siang hari didapatkan rata-rata suhu ruangan yang naik menjadi 32,57 °C. Kelembapan turun menjadi 52,3%. Pada kondisi siang hari ini, nilai suhu dan kelembapan tidak direkomendasikan karena nilai melebihi batas normal mengacu kepada Tabel 2. Sedangkan untuk nilai konsentrasi karbon monoksida rata-rata naik menjadi 13,92 ppm dan nilai tersebut masih dalam *range* normal yang direkomendasikan di bawah 25 ppm

Pengamatan pada saat malam hari didapatkan suhu ruangan turun dibandingkan siang hari dengan rata-rata 30,49 °C dan kelembapan naik menjadi 65,7%. Nilai tersebut juga tidak masuk ke dalam *range* yang direkomendasikan sedangkan untuk nilai konsentrasi karbon monoksida rata-rata saat malam hari adalah 14,7 ppm. Nilai tersebut masih dalam *range* normal yaitu di bawah 25 ppm.

Pengamatan konsentrasi karbon dioksida pada pagi hari rata-rata adalah 247,6 ppm, pada siang hari 198,8

ppm dan malam hari 222,8 ppm. Berdasarkan Tabel 2, nilai tersebut masuk ke dalam batas normal yaitu kurang dari 1000 ppm.

Pengamatan konsentrasi intensitas debu rata-rata pada pagi hari yaitu  $0,005 \text{ mg/m}^3$ , siang hari  $0,05 \text{ mg/m}^3$ , malam hari  $0,01 \text{ mg/m}^3$ . Nilai tersebut masih masuk ke dalam *range* normal mengacu kepada Tabel 2. Dari perbandingan waktu didapatkan nilai intensitas debu yang lebih baik pada pagi hari.

Selanjutnya hasil pengamatan *delay* pengiriman data dari NodeMCU sampai ke Raspberry Pi pada tiga kondisi waktu didapatkan untuk NodeMCU 1 dan NodeMCU 2 memiliki rata-rata *delay* kurang dari 1 detik sampai dengan 1 detik. Namun untuk NodeMCU 3 memiliki *delay* yang lebih besar yaitu rata-rata 3 detik sampai dengan 3,7 detik. Hal ini dapat disebabkan oleh beberapa faktor antara lain pemberian *delay* pada program dan kondisi *hardware* NodeMCU 3. Dalam pengamatan *delay* ini tidak ada pengaruh waktu dalam pengiriman data.

Pengujian visualisasi Grafana seperti pada Sub 4.3 didapatkan hasil *monitoring real time* dan sesuai dengan waktu yang tertera pada *database*. Pengujian 4.4 notifikasi Telegram berhasil dikirimkan saat pembacaan sensor melebihi batas normal dengan mengacu kepada parameter Tabel 2.

## 6. KESIMPULAN

Sistem *monitoring* yang dibuat menggunakan Raspberry Pi yang berfungsi sebagai server *database* dan memvisualisasikan data sensor menggunakan aplikasi Grafana. Telegram bot mengirimkan notifikasi saat pembacaan sensor melebihi batas normal. Dari hasil pengujian didapatkan hasil nilai zat karbon monoksida, karbon dioksida dan intensitas debu memenuhi persyaratan sedangkan kondisi suhu dan kelembapan pada ruangan tidak memenuhi dan perlu dilakukan penyesuaian kembali terhadap kondisi ruangan. *Delay* pengiriman data yang dihasilkan untuk NodeMCU 1 dan NodeMCU 2 adalah kurang dari dan hingga 1 detik. Sedangkan NodeMCU 3 membutuhkan *delay* sekitar 3 sampai dengan 3,7 detik. Dari hasil yang didapat, sistem *monitoring* ini dapat digunakan untuk membantu *user* mengukur dan memantau kualitas udara di dalam ruangan.

## Daftar Pustaka

- [1] R. Prihatmanti, "Kualitas Udara Dalam Ruang – UniversitasCiputra." <https://www.uc.ac.id/library/kualitas-udara-dalam-ruang/>, February. 8, 2017 (accessed Oct. 24, 2020).
- [2] Safety Sign Indonesia , "Buruknya Indoor Air Quality, Akar Munculnya Sick Building Syndrome di Tempat Kerja • Safety Sign Indonesia - Rambu K3, Lalu Lintas, Exit & Emergency , Label B3." <https://safetysign.co.id/news/227/Buruknya-Indoor-Air-Quality-Akar-Munculnya-Sick-Building-Syndrome-di-Tempat-Kerja>, May. 26, 2016 (accessed Oct. 24, 2020).
- [3] J. M. S. Waworundeng and O. Lengkong, "Sistem Monitoring dan Notifikasi Kualitas Udara dalam Ruang dengan Platform IoT," *CogITO Smart J.*, vol. 4, no. 1, p. 94, Jun. 2018, doi: 10.31154/cogito.v4i1.105.94-103.
- [4] F. Pradityo and N. Surantha, "Indoor Air Quality Monitoring and Controlling System based on IoT and Fuzzy Logic," in *2019 7th International Conference on Information and Communication Technology (ICoICT)*, Kuala Lumpur, Malaysia, Jul. 2019, pp. 1–6. doi: 10.1109/ICoICT.2019.8835246.
- [5] R. N. Lesmana and Y. Rahayu, "Membangun Sistem Pemantau Kualitas Udara Dalam Ruang Dengan Mengaplikasikan Sensor CO, O3, PM10 Berbasis LabVIEW," vol. 3, p. 6, 2016.
- [6] H. Shidki, I. Chandra, and E. Djunaedy, "Analisis Kualitas Udara Dalam Ruang Pada Kantor Terbuka Di Universitas Telkom," p. 9.
- [7] N. Hema and J. Yadav, "Secure Home Entry Using Raspberry Pi with Notification via Telegram," in *2020 6th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, Mar. 2020, pp. 211–215. doi: 10.1109/ICSC48311.2020.9182778.
- [8] "DHT11 Sensor Pinout, Features, Equivalents & Datasheet." <https://components101.com/sensors/dht11-temperature-sensor> (accessed Aug. 18, 2021).
- [9] "MQ7datasheet." <https://datasheetpdf.com/pdf-file/694312/Hanwei/MQ7/1> (accessed Aug. 18, 2021)
- [10] "MQ-135 Gas Sensor Pinout, Features, Alternatives, Datasheet & Uses Guide." <https://components101.com/sensors/mq135-gas-sensor-for-air-quality> (accessed Aug. 18, 2021).
- [11] "gp2y1010au.e.pdf." <https://www.sparkfun.com/datasheets/Sensors/gp2y1010au.e.pdf> (accessed Aug. 18, 2021).
- [12] "Introduction to NodeMCU V3 – The Engineering Projects." <https://www.theengineeringprojects.com/2018/10/introduction-to-nodemcu-v3.html> (accessed Aug. 18, 2021)

- [13] M. N. Al-Azam, D. Rizaludin, Y. S. Raharjo, and A. Nugroho, "Message Queuing Telemetry Transport dalam Internet of Things menggunakan ESP-32," *J. MEDIA Inform. BUDIDARMA*, vol. 3, no. 3, p. 159, Jul. 2019, doi: 10.30865/mib.v3i3.1160.
- [14] Kementerian Kesehatan Republik Indonesia, "Persyaratan Kesehatan Lingkungan Kerja Perkantoran dan Industri", "kepmenkes\_no\_1405\_tahun\_2002.pdf."
- [15] "Send data from ESP8266 or ESP32 to Raspberry Pi via MQTT." <https://diy10t.com/microcontroller-to-raspberry-pi-wifi-mqtt-communication/> (accessed Jul. 22, 2021).