

SISTEM PEMANTAU KENYAMANAN RUANG KELAS MENGGUNAKAN PROTOKOL MQTT DAN HTTP DENGAN NOTIFIKASI TELEGRAM BERBASIS INTERNET OF THINGS

Aditya Aziz Fikhri^{*1}, Munirul Ula², Muhammad Sayuti³

^{1,3,2}Fakultas Teknik, Universitas Malikussaleh, Aceh

Email: ¹aditfreedom11@gmail.com, ²munirulula@unimal.ac.id, ³sayuti_m@unimal.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 10 Desember 2024, diterima untuk diterbitkan: 30 Oktober 2025)

Abstrak

Pendidikan berperan penting dalam pembangunan bangsa dan peningkatan kualitas hidup masyarakat. Sekolah sebagai institusi pendidikan formal membutuhkan lingkungan belajar yang sehat dan nyaman untuk mendukung proses belajar-mengajar. Salah satu faktor utama yang memengaruhi kenyamanan belajar adalah kualitas udara. Penelitian ini bertujuan untuk memantau kualitas udara di ruang kelas SD Sukma Bangsa Bireuen, Provinsi Aceh, menggunakan konsep Internet of Things (IoT) dengan membandingkan dua protokol komunikasi, yaitu MQTT dan HTTP. Parameter udara yang dipantau meliputi suhu, kelembapan, CO, CO₂, PM1, dan PM2.5. Hasil pengujian pada 10 sampel data menunjukkan bahwa protokol MQTT memiliki rata-rata waktu pengiriman data sebesar 6,2 milidetik, sedangkan protokol HTTP memerlukan rata-rata waktu 267 milidetik, menunjukkan bahwa MQTT sekitar 97,7% lebih cepat. Pada sesi pertama (pagi-siang), terjadi kenaikan suhu hingga 1,7°C, penurunan kelembapan sekitar 5%, sementara kadar CO dan CO₂, serta konsentrasi PM1 dan PM2.5 masih dalam batas aman. Sesi kedua (malam) menunjukkan penurunan suhu sekitar 2°C, kenaikan kelembapan sebesar 8%, dan peningkatan kadar CO₂ hingga 22,5% karena minimnya sirkulasi udara. Dashboard yang dibangun menggunakan Node-RED dapat menampilkan data secara real-time dari kedua protokol dengan lancar. Selain itu, sistem juga diintegrasikan dengan fitur notifikasi melalui Telegram, yang mampu mengirimkan peringatan otomatis setiap 10 menit jika parameter melebihi ambang batas, serta melayani permintaan data secara langsung dari pengguna. Dengan efisiensi pengiriman data, fleksibilitas arsitektur, dan kemampuan notifikasi real-time, sistem ini tidak hanya efektif untuk lingkungan kelas, tetapi juga berpotensi direplikasi pada ruang tertutup lainnya seperti laboratorium, ruang guru, dan ruang publik lainnya di lingkungan pendidikan.

Kata kunci: *IoT, HTTP, MQTT, kenyamanan, udara, telegram*

CLASSROOM COMFORT MONITORING SYSTEM USING MQTT AND HTTP PROTOCOLS WITH TELEGRAM NOTIFICATIONS BASED ON THE INTERNET OF THINGS

Abstract

Education plays an essential role in national development and improving the quality of life in society. Schools, as formal educational institutions, require a healthy and comfortable learning environment to support the teaching and learning process. One of the main factors affecting learning comfort is air quality. This study aims to monitor air quality in a classroom at SD Sukma Bangsa Bireuen, Aceh Province, using the Internet of Things (IoT) concept by comparing two communication protocols: MQTT and HTTP. The monitored air parameters include temperature, humidity, CO, CO₂, PM1, and PM2.5. Test results from 10 data samples show that the MQTT protocol achieved an average data transmission time of 6.2 milliseconds, while the HTTP protocol required an average of 267 milliseconds, indicating that MQTT is approximately 97.7% faster. During the first session (morning to afternoon), there was an increase in temperature of up to 1.7°C and a decrease in humidity of about 5%, while CO and CO₂ levels and PM1 and PM2.5 concentrations remained within safe limits. The second session (evening) showed a temperature drop of about 2°C, an increase in humidity by 8%, and a rise in CO₂ levels by up to 22.5% due to limited air circulation. The dashboard built using Node-RED successfully displayed real-time data from both protocols. Additionally, the system was integrated with a Telegram notification feature that could automatically send alerts every 10 minutes if any parameter exceeded the threshold, as well as respond to real-time data requests from users. With efficient data transmission, flexible architecture, and real-time notification capabilities, the system is not only effective for classroom environments

but also has the potential to be replicated in other enclosed spaces such as laboratories, teacher rooms, and public areas within educational institutions.

Keywords: *IoT, HTTP, MQTT, comfort, air, telegram*

1. PENDAHULUAN

Pendidikan memiliki peran yang sangat vital dalam pembangunan bangsa dan peningkatan kualitas hidup masyarakat. Sebagai institusi pendidikan formal, sekolah merupakan tempat utama di mana proses belajar-mengajar berlangsung. Lingkungan yang sehat dan nyaman adalah prasyarat bagi terciptanya kondisi belajar yang optimal. Salah satu aspek yang sangat mempengaruhi dalam menciptakan lingkungan belajar yang nyaman adalah kualitas udara pada ruangan (Setiati, Febrina dan Islami, 2022). Udara merupakan campuran berbagai jenis gas dengan perbandingan tak tentu tergantung suhu, tekanan udara, dan lingkungan (Hidayati, Zaini Rachman dan Arif Surya Rimbawan, 2020). Kualitas udara yang buruk dapat berdampak negatif terhadap kesehatan dan kenyamanan siswa dalam belajar yang pada akhirnya dapat mempengaruhi proses belajar-mengajar.

Dampak kualitas udara yang buruk dapat mengganggu proses belajar-mengajar bahkan jika anak-anak terpapar polusi udara dalam jangka panjang tentunya dapat mempengaruhi kesehatan hingga berpengaruh pada prestasi akademiknya. Hal ini disebabkan oleh beragam faktor, seperti gangguan konsentrasi, kelelahan, dan peningkatan penyakit yang berhubungan dengan polusi udara. Artinya kualitas udara kelas memegang peranan penting dan mempunyai hubungan yang kuat dengan hasil belajar siswa (Munjirin dan Iswinarti, 2023). Kualitas udara di lingkungan sekolah sering kali dipengaruhi oleh berbagai faktor, termasuk polusi dari kendaraan bermotor, kegiatan industri di sekitar sekolah, dan aktivitas internal sekolah seperti penggunaan bahan kimia dalam kegiatan laboratorium. Studi yang dilakukan oleh As'ari (2022) menunjukkan bahwa peningkatan paparan terhadap polutan seperti PM2.5 dan PM10, berkorelasi dengan meningkatnya keluhan *dyspnea* atau sesak nafas pada warga Desa Lakardowo, Kabupaten Mojokerto, Provinsi Jawa Timur. Sehingga hal ini membutuhkan perhatian khusus bagi pimpinan serta manajemen sekolah agar lingkungan sekolah tetap sehat.

Selain itu, temuan dari penelitian sebelumnya yang relevan dilakukan oleh Suratno et al. (2022) menunjukkan bahwa pemantauan kualitas udara berbasis IoT dengan Arduino pada ruang kelas di Sekolah Dasar dapat memberikan hasil yang signifikan terhadap konsentrasi CO₂ berdasarkan jumlah siswa di dalam kelas. Penelitian tersebut membuktikan bahwa semakin banyak jumlah siswa, semakin tinggi konsentrasi CO₂ di ruangan, namun

tidak secara langsung memengaruhi suhu udara. Suhu ruang kelas ditemukan masih belum ideal (31,3°C) dibandingkan standar kenyamanan (20–22°C). Dengan demikian, integrasi sistem pemantauan berbasis IoT sangat penting dalam memberikan gambaran nyata terkait kualitas udara di ruang kelas, yang menjadi dasar pengambilan keputusan untuk menciptakan lingkungan belajar yang sehat dan nyaman.

Salah satu upaya untuk pencegahan resiko kualitas udara yang buruk adalah dengan cara pemantauan kualitas udara yang dilakukan secara *real-time*. *Internet of Things* (IoT) adalah konsep yang mengacu pada jaringan perangkat fisik yang terhubung melalui internet, memungkinkan mereka untuk berkomunikasi dan bertukar data (Nugraha, Putra dan Setiawan, 2024). Konsep ini berawal dari perkembangan teknologi sensor dan komunikasi yang memungkinkan objek-objek fisik, seperti perangkat rumah tangga, kendaraan, dan mesin industri, untuk terhubung dan berbagi informasi (Adani dan Salsabil, 2020). Dengan memberikan beberapa sensor yang diperlukan serta adanya koneksi internet, pengguna dapat bertukar informasi terkait dengan kualitas udara sehingga dapat mengetahui tingkat kenyamanan ruang kelas sesuai dengan parameter yang telah disebutkan sebelumnya. Dalam mengirimkan data dari sensor melalui perangkat IoT ke server dibutuhkan sebuah protokol komunikasi agar data dapat terkirim.

Berdasarkan survei IoT & Edge Eclipse Foundation tahun 2023, protokol komunikasi populer yang digunakan dalam IoT adalah MQTT (*Message Queue Telemetry Transport*) dan HTTP (*Hypertext Transfer Protocol*) (Nugraha, Putra dan Setiawan, 2024). MQTT adalah protokol *publish-subscribe* yang digunakan untuk komunikasi IoT (Bender et al., 2021). Protokol MQTT dirancang untuk fokus pada meminimalkan penggunaan *bandwidth* jaringan dan sumber daya perangkat untuk memastikan pengiriman yang handal (Mishra dan Kertesz, 2020). Sedangkan HTTP merupakan protokol yang menjadi dasar model *client-server* yang umum digunakan pada website (Nikolov, 2020). Penelitian yang dilakukan oleh Fikhri, et al (2025) yang bertujuan untuk membandingkan kinerja protokol MQTT dan HTTP menggunakan server *virtual machine* dengan spesifikasi identik, adapun parameter yang diuji seperti penggunaan CPU, waktu pengiriman pesan, dan tingkat kehilangan data. Protokol MQTT, yang mengadopsi model *publish-subscribe* dan dikenal ringan, menunjukkan kecepatan pengiriman pesan yang jauh lebih cepat dibandingkan HTTP, terutama saat

menangani jumlah pesan yang besar. Namun, peningkatan jumlah pesan pada MQTT menyebabkan lonjakan penggunaan CPU dan tingkat kehilangan data yang cukup tinggi mencapai 33,8% pada 600.000 pesan. Sebaliknya, protokol HTTP yang menggunakan model *request-response* dengan pendekatan multi proses mampu menjaga integritas data dengan tingkat keberhasilan pengiriman hingga 100%, meskipun waktu pengirimannya lebih lambat. Berdasarkan hasil ini, MQTT lebih cocok diterapkan pada sistem yang membutuhkan pengiriman cepat dan dapat mentoleransi sebagian kehilangan data, sedangkan HTTP lebih ideal untuk aplikasi yang memerlukan akurasi tinggi dan keandalan dalam pengiriman data.

Penelitian ini bertujuan untuk merancang bangun sistem IoT yang dilakukan untuk melakukan pemantauan kualitas udara pada ruang kelas di SD Sukma Bangsa Bireuen, Provinsi Aceh, adapun parameter udara yang dilakukan pemantauan yaitu Suhu, Kelembapan, CO, CO₂, PM₁, dan PM_{2.5}. Metode penelitian yang dilakukan meliputi perancangan perangkat IoT, perancangan infrastruktur server meliputi spesifikasi dan instalasi IP *public*, pengujian langsung pada salah satu ruang kelas, dan pengujian notifikasi pada aplikasi Telegram. Proses pengiriman data hasil pembacaan sensor kualitas udara dikirimkan melalui protokol MQTT dan HTTP, kemudian melalui protokol tersebut akan dilakukan perbandingan dari aspek kecepatan waktu pengiriman data ke server. Server yang digunakan dalam penelitian ini yaitu *dedicated server* yang merupakan keunggulan dari penelitian ini, sehingga server dapat dikontrol sepenuhnya. Broker yang digunakan pada protokol MQTT yaitu Mosquitto, penelitian ini menggunakan skema QoS 0 pada protokol MQTT. Hasil data yang terkirim akan ditampilkan pada *dashboard* yang dibangun dengan menggunakan node-red, kemudian data juga dapat dilihat melalui aplikasi Telegram secara *real-time*. Notifikasi akan dikirimkan kepada pengguna Telegram ketika salah satu parameter berada diatas ambang batas standar yang telah ditentukan.

2. TINJAUAN PUSTAKA

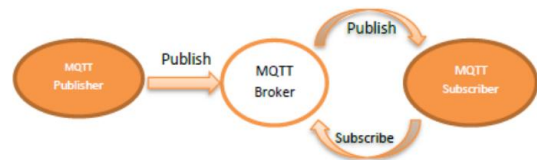
2.1 Protokol MQTT

Protokol MQTT (*Message Queue Telemetry Transport*) adalah protokol *publish-subscribe* yang dirancang khusus untuk komunikasi dalam konteks Internet of Things (IoT). Protokol ini sangat efisien dalam hal penggunaan *bandwidth* dan sumber daya perangkat, yang membuatnya ideal untuk perangkat dengan keterbatasan sumber daya dan kondisi jaringan yang tidak stabil (Bender et al., 2021).

MQTT beroperasi berdasarkan model komunikasi *publish-subscribe* yang memungkinkan perangkat untuk berkomunikasi secara asinkron. Dalam model ini, perangkat yang bertindak sebagai publisher mengirimkan pesan ke topik tertentu,

sedangkan perangkat yang bertindak sebagai subscriber menerima pesan dari *topic* yang sama. Broker MQTT berperan sebagai perantara yang mengelola distribusi pesan antara publisher dan subscriber. Broker ini memastikan bahwa pesan disampaikan dengan cara yang efisien dan dapat diandalkan (Mishra dan Kertesz, 2020).

Dalam protokol MQTT, terdapat mekanisme seperti *quality of service* (QoS) dalam mengirimkan data, yaitu QoS 0 (kirim sekali, tidak perlu konfirmasi), QoS 1 (kirim sekali, harus ada konfirmasi), dan QoS 2 (kirim sekali, dengan proses *handshaking* dua arah untuk memastikan pesan diterima hanya sekali). Dengan mekanisme ini, MQTT dapat menyesuaikan tingkat keandalan pesan berdasarkan kebutuhan aplikasi dan kondisi jaringan (Krunianto et al., 2022). Dalam penelitian yang dilakukan oleh Nikolov (2020), protokol MQTT dapat bekerja dalam kondisi jaringan yang terbatas dapat menggunakan sumber daya yang rendah dibandingkan dengan protokol HTTP.



Gambar 1. protokol MQTT

2.2 Protokol HTTP

HTTP atau *Hypertext Transfer Protocol* awalnya dikembangkan dengan tujuan untuk mempublikasikan dan mengunduh halaman web dengan HTML. Saat ini HTTP tidak hanya digunakan sebagai protokol komunikasi untuk menangani halaman web melainkan dimanfaatkan untuk mentransfer data antara klien dan server. HTTP menentukan protokol dalam melakukan permintaan (*request*) dan tanggapan (*response*) di antara klien dan server (Hanif dan Amri, 2023).

Dalam protokol HTTP, terdapat beberapa metode utama yang digunakan untuk interaksi antara klien dan server yaitu GET, POST, PUT, dan DELETE. Metode GET digunakan oleh klien untuk meminta data dari server, seperti halaman web atau informasi dari API (*Application Program Interface*). Metode POST dan PUT digunakan untuk mengunggah data ke server; POST umumnya digunakan untuk mengirimkan data baru, sedangkan PUT digunakan untuk memperbarui data yang sudah ada di server. Sementara itu, metode DELETE digunakan untuk menghapus data dari server (Kurniawan dan Warlina, 2020). Penelitian yang dilakukan oleh Nugraha, dkk. (2024), protokol HTTP menggunakan sumber daya dan delay yang lebih besar dibandingkan dengan MQTT, namun protokol HTTP dapat menerima semua pesan yang dikirimkan saat dilakukan uji beban, artinya protokol HTTP sangat sedikit *data loss* dibandingkan MQTT.

2.3 Kualitas Udara

Pada Peraturan Menteri Kesehatan (Permenkes) nomor (1077/MENKES/PER/V/2011) tentang Pedoman Penyehatan Udara Dalam Ruang menyebutkan terdapat tiga persyaratan kualitas udara di dalam ruangan yaitu Kualitas Fisik, Kualitas Kimia, dan Kualitas Biologi. Sementara itu, salah satu aspek yang sangat mempengaruhi dalam menciptakan lingkungan belajar yang nyaman adalah kualitas udara pada ruangan kelas (Setiati, Febrina dan Islami, 2022). Terdapat sebelas rekomendasi parameter beserta ambang batas kualitas udara yang dapat mempengaruhi kenyamanan kelas dapat dilihat pada Tabel 1.

Tabel 1. Rekomendasi ambang batas kualitas udara pada kelas

jenis parameter	rekomendasi ambang batas	standar referensi
Suhu	26-29 °C	Permenkes Tahun 2011
Kelembapan	60-65%	Permenkes Tahun 2011
CO	Max 30 ppm	Permenkes Tahun 2011
CO ₂	800-1000 ppm	CIBS Guide A (2016)
Laju Ventilasi	Min 10 L/s	CIBS AM 10 (2005)
Kecepatan Udara	0.2 m/c	ASHRA Standard 62.1 (2019)
NO ₂	Max 100 ppb/jam	Permenkes Tahun 2011
O ₃	Max 0.3 ppm	Permenkes Tahun 2011
PM ₁	Max 150 u/m ³	USEPA (2013)
PM _{2.5}	Max 35 u/m ³	Permenkes Tahun 2011
Formaldehida	Max 0.1 ppm	Permenkes Tahun 2011

3. METODE PENELITIAN

3.1. Gambaran Umum Sistem

Proses pemantauan kualitas udara pada ruang kelas dilakukan dengan menggunakan perangkat IoT yang untuk mendeteksi parameter udara yaitu suhu, kelembapan, CO, CO₂, PM₁, dan PM_{2.5}. Pengambilan sampel data kualitas udara dilakukan di kelas 1 SD Sukma Bangsa, Kabupaten Bireuen, Provinsi Aceh. Setelah dilakukan deteksi parameter udara, data dikirimkan ke server MQTT dan HTTP secara sekuensial dan akan dicatat lama waktu proses pengiriman untuk dilakukan perbandingan. Server yang dibangun yaitu server tanpa pihak ketiga atau dedicated server, diharapkan dapat mengirimkan data hasil deteksi dapat berjalan lebih cepat. Broker pada MQTT pada penelitian ini menggunakan Mosquitto. Data hasil pengiriman dapat dilihat pada dashboard yang dibangun dengan menggunakan platform Node-RED, selain itu pengguna juga dapat melakukan pengiriman data secara *real-time* dengan menggunakan aplikasi Telegram. Notifikasi sistem akan dikirimkan melalui Telegram apabila terdapat salah satu parameter kualitas udara yang melebihi ambang batas seperti yang ada pada Tabel 1.

Keunikan dari penelitian ini dibandingkan penelitian sebelumnya terletak pada kombinasi penggunaan dua protokol komunikasi (MQTT dan HTTP) yang diuji secara langsung pada lingkungan yang sama, serta pada integrasi fitur notifikasi ganda berbasis Telegram. Sistem notifikasi mencakup dua mekanisme utama, yaitu: notifikasi otomatis yang dikirim setiap 10 menit apabila terdeteksi parameter di atas ambang batas, dan notifikasi berdasarkan permintaan pengguna yang dapat diakses secara *real-time* melalui bot Telegram.

Selain itu, sistem dirancang dengan server yang dapat dikontrol secara penuh (*dedicated server*) dengan perantara IP *public* yang memungkinkan replikasi di ruang kelas lain tanpa ketergantungan pada infrastruktur *cloud* pihak ketiga. Kombinasi pendekatan ini menjadikan sistem tidak hanya efisien dalam pengiriman data, tetapi juga adaptif, hemat biaya, dan dapat diimplementasikan secara luas dalam konteks pemantauan kualitas udara di lingkungan pendidikan.

3.2. Spesifikasi Sistem

Pada bagian ini, akan dibahas secara rinci mengenai spesifikasi perangkat keras dan perangkat lunak yang digunakan pada server, mencakup topik dan *endpoint* pengiriman data, serta struktur jaringan dan arsitektur server yang akan digunakan untuk mendukung penelitian ini. Struktur jaringan yang digunakan juga akan dijabarkan untuk memberikan gambaran bagaimana alur data berjalan dari perangkat sensor hingga ke server yang bertugas memproses data tersebut. Sebagai pendukung, Tabel 2 menyajikan daftar lengkap spesifikasi perangkat keras dan perangkat lunak pada kedua server yang digunakan dalam penelitian ini.

Tabel 2. Spesifikasi perangkat keras dan lunak server

komponen	server MQTT	server HTTP
Sistem Operasi	Ubuntu 22.04 LTS	Ubuntu 22.04 LTS
CPU Core	1	1
RAM	6 GB	6 GB
SSD	60 GB	60 GB
Dashboard Builder	Node-RED	Node-RED
Database	MariaDB	MariaDB
Broker/Webserver	Mosquitto	Apache2

Kedua server diberikan spesifikasi yang sama agar dapat menguji perbandingan kedua protokol dalam aspek waktu pengiriman. Tabel 3(a) merupakan perancangan *topic* yang digunakan pada protokol MQTT dan Tabel 3(b) merupakan *endpoint* yang digunakan untuk protokol HTTP yang digunakan pada penelitian ini.

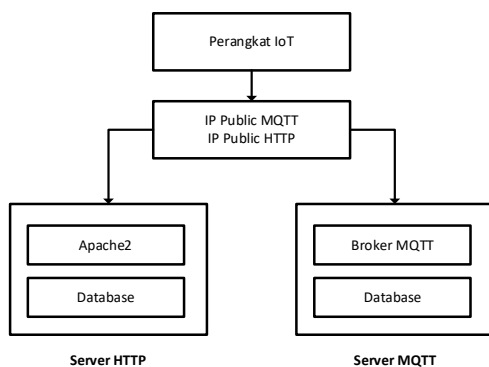
Tabel 3(a). *Topic* pada protokol MQTT

topic	parameter data
/kenyamanan/suhu	suhu
/kenyamanan/kelembapan	kelembapan
/kenyamanan/co	CO
/kenyamanan/co2	CO ₂
/kenyamanan/pm1	PM ₁
/kenyamanan/pm25	PM _{2.5}

Tabel 3(b). Endpoint pada protokol HTTP

endpoint	parameter data	method
suhu	suhu	GET
kelembapan	kelembapan	GET
co	CO	GET
co2	CO ₂	GET
pm1	PM ₁	GET
pm25	PM _{2.5}	GET

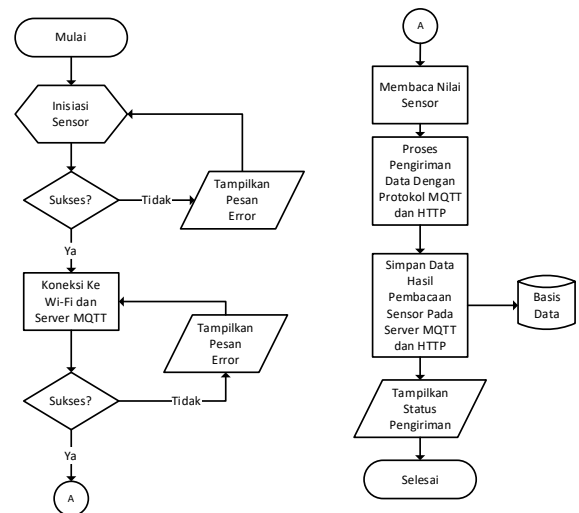
Karena server untuk menyimpan data pada penelitian ini menggunakan *dedicated server*, maka dibutuhkan sebuah alamat IP public yang digunakan agar server dapat diakses secara *online*, IP public digunakan untuk melakukan *redirect* ke server lokal untuk dilakukan penyimpanan data hasil pembacaan sensor, Gambar 2 merupakan struktur jaringan dan server yang digunakan dalam penelitian ini.



Gambar 2. Struktur jaringan dan server

3.3. Diagram Alir dan Konteks IoT

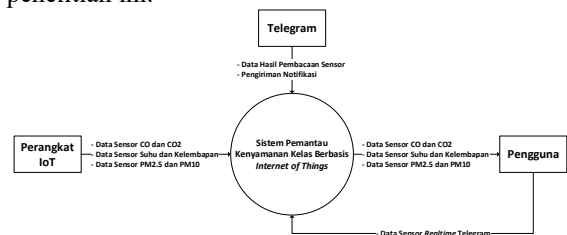
Pada bagian ini, akan dijelaskan secara mendalam mengenai diagram alir dan diagram konteks perangkat IoT yang dirancang dalam penelitian ini. Diagram alir memberikan gambaran terstruktur mengenai alur kerja perangkat IoT, mulai dari proses inisialisasi perangkat hingga pengiriman data ke server. Diagram ini bertujuan untuk memvisualisasikan langkah-langkah yang terjadi secara berurutan ketika perangkat IoT diaktifkan dan mulai menjalankan fungsinya. Sementara itu, diagram konteks menggambarkan hubungan antara perangkat IoT dengan sistem lain seperti server dan pengguna. Gambar 3(a) secara khusus menampilkan diagram alir yang memetakan proses yang berjalan saat perangkat IoT mulai aktif.



Gambar 3(a). diagram alir perangkat IoT

Ketika perangkat IoT diaktifkan, maka akan melakukan inisialisasi semua sensor, jika salah satu sensor tidak berhasil diaktifkan, maka akan menampilkan pesan error pada layar. Setelah berhasil melakukan inisialisasi sensor, maka perangkat akan melakukan koneksi ke Wi-Fi dan server MQTT melalui IP Public server MQTT. Jika sukses maka akan melakukan proses pembacaan sensor, jika gagal maka akan menampilkan pesan error. Setelah melakukan pembacaan sensor, selanjutnya akan melakukan pengiriman data hasil pembacaan sensor pada kedua server dengan IP Public pada kedua protokol. Setelah berhasil, maka akan dilakukan penyimpanan data pada database kedua server. Terakhir yang dilakukan yaitu menampilkan pesan bahwa data berhasil terkirim.

Karena penelitian ini menggunakan sebuah sistem untuk pemantauannya, maka dibutuhkan diagram konteks yang bertujuan untuk mendefinisikan entitas dan tugasnya pada sistem, Gambar 3(b) merupakan diagram konteks pada penelitian ini.

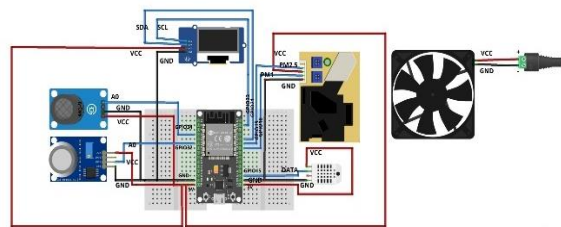


Gambar 3(b). diagram konteks sistem

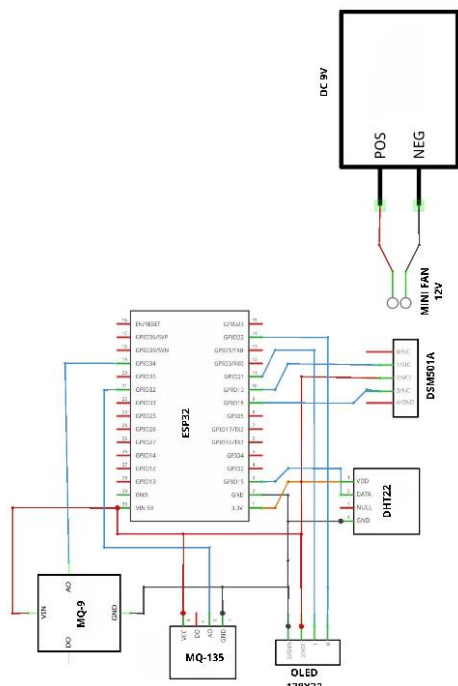
Terdapat tiga entitas pada penelitian ini yaitu Perangkat IoT, Telegram, dan Pengguna. Perangkat IoT mengirimkan data kualitas udara pada pengguna, Telegram mengirimkan notifikasi dan notifikasi kepada pengguna. Pengguna dapat menerima data hasil pembacaan sensor dan melakukan permintaan data secara *real-time* pada aplikasi Telegram.

3.4. Perancangan Perangkat IoT

Pada bagian ini akan menjelaskan proses perancangan perangkat IoT beserta *wiring diagram*. Gambar 4(a) merupakan perancangan perangkat IoT dan Gambar 4(b) merupakan *wiring diagram* perancangan perangkat IoT pada penelitian ini.



Gambar 4(a). perancangan perangkat IoT



Gambar 4(b). *wiring diagram* perangkat IoT

Tabel 4. Perincian perangkat dan pin yang digunakan

sensor	pin pada sensor	pin pada ESP32
DHT22	VDD	3.3V
	GND	GND
	Data	GPIO15
MQ-135	VCC	5V/VIN
	GND	GND
	AOUT	GPIO32
MQ-9	VCC	5V/VIN
	GND	GND
	AOUT	GPIO34
DSM501A	GND	GND
	PM1	GPIO19
	PM2.5	GPIO18
OLED 128X32	VCC	5V
	GND	GND
	SCL	GPIO22
	SDA	GPIO21

Perangkat yang digunakan dalam penelitian ini yaitu ESP32 digunakan sebagai mikrokontroler, kemudian untuk deteksi suhu dan kelembapan digunakan sensor DHT22, untuk deteksi CO digunakan sensor MQ-9, sedangkan CO₂ dengan menggunakan MQ-135, terakhir yang digunakan untuk deteksi PM₁, dan PM_{2.5} menggunakan sensor DSM501A. Terdapat perangkat tambahan seperti OLED 128x32 untuk menampilkan output teks ke layar, kemudian *mini fan* agar membantu debu masuk ke sensor DSM501A untuk dilakukan pembacaan data. Tabel 4 merupakan rincian dari perangkat dan sensor dan pin yang digunakan.

3.5. Perancangan Basis Data

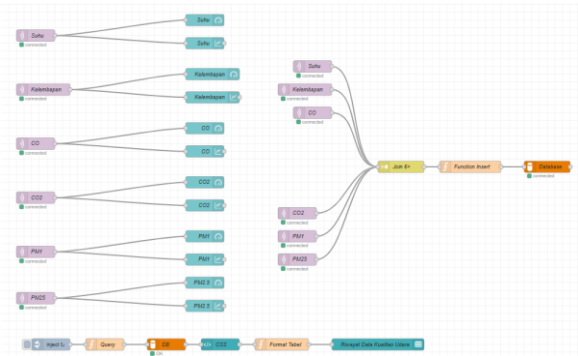
Basis data yang digunakan dalam sistem pada penelitian ini sebanyak 8 atribut yaitu id, suhu, kelembapan, co, co2, pm1, dan pm2.5 dengan masing-masing atribut memiliki tipe data dan jumlah data yang berbeda, Tabel 5 merupakan perincian basis data yang digunakan pada penelitian ini.

Tabel 5. Perancangan basis data

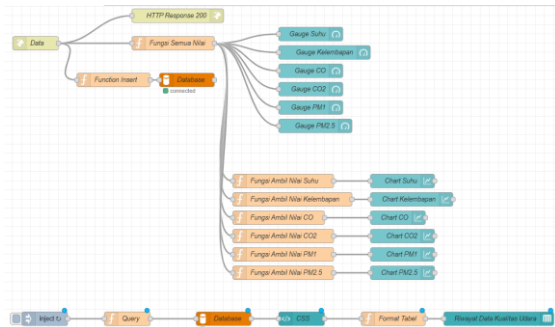
atribut	tipe data
id (PK)	id (11)
suhu	varchar (10)
kelembapan	varchar (10)
co	varchar (10)
co2	varchar (10)
pm1	varchar (10)
pm25	varchar (10)
timestamp	timestamp

3.6. Perancangan Dashboard IoT

Dashboard IoT pada penelitian ini dibangun dengan aplikasi node-red. Terdapat perancangan dashboard yang berbeda antara server MQTT dan HTTP. Gambar 5(a) merupakan perancangan dashboard yang digunakan pada server MQTT dan Gambar 5(b) merupakan perancangan *dashboard* yang digunakan pada server HTTP.



Gambar 5(a). perancangan *dashboard* server MQTT



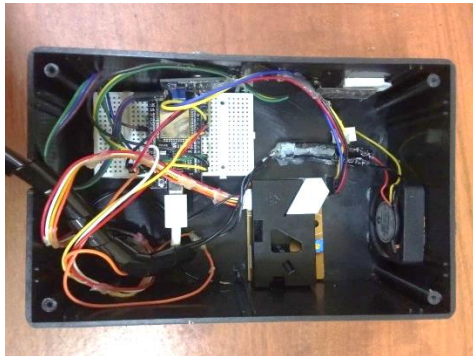
Gambar 5(b). Perancangan dashboard server HTTP

Pada perancangan dashboard MQTT, terdapat *node* MQTT in yang digunakan untuk mendapatkan data hasil pembacaan sensor kemudian dilakukan subscribe dengan *topic* yang telah dirincikan pada Tabel 3(a). Pada perancangan dashboard HTTP, terdapat *node* HTTP in yang digunakan untuk mendapatkan data dari endpoint dari protokol HTTP yang telah dijelaskan pada Tabel 3(a). Untuk menampilkan data dalam bentuk grafik, gauge, dan tabel, kedua protokol menggunakan *node* yang sama.

4. HASIL DAN PEMBAHASAN

4.1. Implementasi Perangkat IoT

Perangkat yang dilakukan perancangan dilakukan perakitan dan penyisipan kode program dengan Arduino IDE. Gambar 6(a) dan Gambar 6(b) merupakan hasil dari perancangan perangkat IoT.



Gambar 6(a). instalasi sensor pada case perangkat



Gambar 6(b). hasil perancangan perangkat IoT

Perangkat yang telah dilakukan perancangan dilakukan pengujian mulai dari inisialisasi sensor, koneksi pada Wi-Fi, pembacaan sensor, hingga pengiriman data. Gambar 7(a) hingga Gambar 7(d) merupakan pengujian perangkat IoT dengan menggunakan serial monitor pada Arduino IDE dengan baud rate 115200.

```
Output  Serial Monitor x
Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on 'COM3')

Inisialisasi Sensor :
Berhasil!

Mengkoneksikan Ke Wi-Fi Redmi Note 11
..
Terkoneksi ke Wi-Fi
SSID : Redmi Note 11
IP address: 192.168.1.9
Connecting to MQTT...
Connected to MQTT
```

Gambar 7(a). proses inisialisasi sensor dan koneksi Wi-Fi pada serial monitor

```
Output  Serial Monitor x
Message (Enter to send message to 'DOIT ESP32

-----
Hasil Pembacaan Sensor :
-----

PM2.5: 56.45 ug/m3
PM1: 0.00 ug/m3
Temp: 30.60 C
Humidity: 67.50 %
CO : 0.00 PPM
CO2 : 359.19 PPM
```

Gambar 7(b). proses pembacaan nilai sensor pada serial monitor

```
Output  Serial Monitor x
Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on 'COM3')

-----
Proses Pengiriman Data Dengan Protokol MQTT :
-----

Suhu : Publish Berhasil
Kelembapan : Publish Berhasil
CO : Publish Berhasil
CO2 : Publish Berhasil
PM25 : Publish Berhasil
PM1 : Publish Berhasil
Total Waktu Pengiriman MQTT : 6 ms
```

Gambar 7(c). proses pengiriman data pada dengan menggunakan protokol MQTT

```
Output  Serial Monitor x
Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on '

-----
Proses Pengiriman Data Dengan Protokol HTTP :
-----

Data HTTP Terkirim!
Total Waktu Pengiriman HTTP : 6025 ms
```

Gambar 7(d). proses pengiriman data sensor dengan menggunakan protokol HTTP

Perangkat IoT yang telah melalui tahap uji coba kemudian diimplementasikan pada ruang kelas untuk memantau dan mendapatkan data kondisi kualitas udara secara *real-time*. Implementasi ini bertujuan untuk mengevaluasi apakah perangkat telah berfungsi sesuai dengan rancangan dalam menyampaikan informasi terkait parameter kualitas udara di lingkungan tersebut. Perangkat IoT ini terhubung dengan server yang memiliki jarak fisik yang cukup jauh dan berada pada jaringan yang berbeda, sehingga untuk memastikan komunikasi antar perangkat dan server berjalan lancar, digunakan IP *public* pada kedua server, baik untuk protokol MQTT maupun HTTP. IP *public* ini berfungsi sebagai penghubung utama yang memungkinkan perangkat IoT mengirimkan data secara langsung ke server melalui kedua protokol komunikasi yang diterapkan. Gambar 8 menggambarkan implementasi perangkat IoT di ruang kelas, menunjukkan bagaimana perangkat ditempatkan dan terintegrasi dengan infrastruktur jaringan untuk mendukung proses pemantauan kualitas udara.



Gambar 8. implementasi perangkat IoT pada ruang kelas

4.2. Pengujian Pengiriman Data

Pengujian pengiriman data pada penelitian ini diambil masing-masing 10 data pada server MQTT dan HTTP. Tabel 6(a) merupakan data hasil pengiriman data pada server MQTT.

Tabel 6(a). Pengiriman data server MQTT						
suhu (°C)	Kelembapan (%)	CO (ppm)	CO ₂ (ppm)	PM ₁ (ug/m ³)	PM _{2.5} (ug/m ³)	waktu (ms)
30.70	65.20	0.00	380.81	0.00	57.75	6
30.70	65.40	0.00	400	88.24	59.34	6
30.70	65.00	0.00	392.24	96.68	53.17	6
30.70	64.90	0.00	392.24	92.30	44.73	7
30.70	64.90	0.00	369.63	86.53	15.46	6
30.50	63.90	0.00	384.59	111.37	1.55	5
30.20	62.40	0.00	384.59	101.50	30.11	6
29.90	61.86	0.00	384.59	86.05	37.96	7
29.40	60.30	0.00	369.63	91.94	35.71	6
29.00	60.30	0.00	355.10	182.96	15.57	7
rata-rata						6.2

Dari 10 data pengiriman yang dilakukan, data yang diterima tidak ada gangguan sehingga 100%

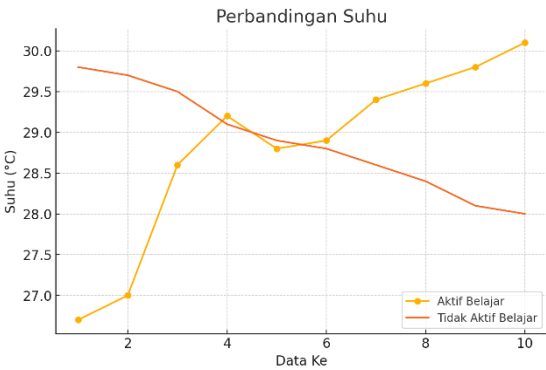
data diterima oleh server. Waktu yang dibutuhkan untuk mengirimkan data ke server MQTT rata-rata yaitu 6.2ms, hal ini merupakan waktu yang sangat cepat untuk tingkat pengiriman data pada server. Tabel 7 merupakan data hasil pengiriman data pada server HTTP.

Tabel 7. Pengiriman data server HTTP						
suhu (°C)	Kelembapan (%)	CO (ppm)	CO ₂ (ppm)	PM ₁ (ug/m ³)	PM _{2.5} (ug/m ³)	waktu (ms)
30.70	65.20	0.00	380.81	0.00	57.75	542
30.70	65.40	0.00	400	88.24	59.34	124
30.70	65.00	0.00	392.24	96.68	53.17	191
30.70	64.90	0.00	392.24	92.30	44.73	325
30.70	64.90	0.00	369.63	86.53	15.46	151
30.50	63.90	0.00	384.59	111.37	1.55	122
30.20	62.40	0.00	384.59	101.50	30.11	102
29.90	61.86	0.00	384.59	86.05	37.96	327
29.40	60.30	0.00	369.63	91.94	35.71	129
29.00	60.30	0.00	355.10	182.96	15.57	542
rata-rata						267

Hal yang sama dilakukan pada server HTTP, 10 data pengiriman yang dilakukan, data yang diterima juga tidak mengalami gangguan sehingga 100% data diterima oleh server. Waktu yang dibutuhkan untuk mengirimkan data ke server HTTP rata-rata yaitu 267ms. Pengiriman data dengan protokol HTTP membutuhkan waktu lebih lama dari protokol MQTT, namun keduanya dapat terbilang cepat karena proses pengiriman tidak lebih dari 1 detik.

4.3. Pengujian Pemantauan Ruang Kelas

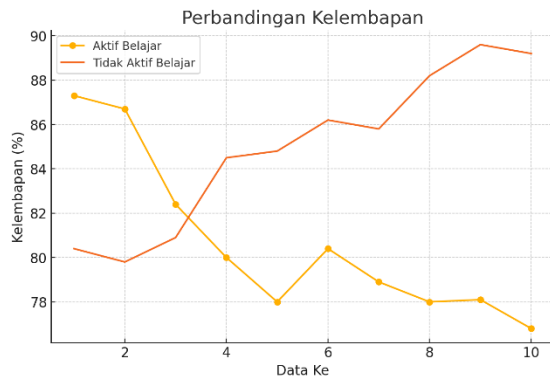
Pengujian pemantauan pada ruang kelas dilakukan pada dua sesi, sesi pertama yaitu pada pukul 08.00 WIB hingga 12.30 WIB yang mana rentang tersebut merupakan waktu aktif belajar di kelas, kemudian sesi kedua dilakukan pada pukul 17.00 WIB hingga 23.59 WIB yang mana waktu tersebut tidak terjadi aktivitas pembelajaran di kelas. Data yang diambil untuk hasil pemantauan yaitu 10 data pada waktu yang berbeda. Gambar 8 merupakan hasil pengujian pemantauan ruang kelas pada saat aktif belajar.



Gambar 8. grafik perbandingan suhu

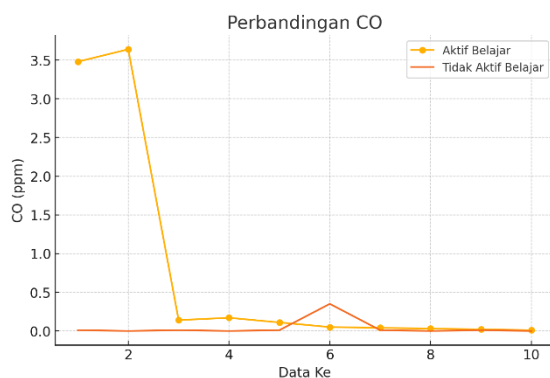
Perbandingan suhu ruangan kelas pada Gambar 8 menjelaskan bahwa sesi pertama atribut suhu pada

waktu aktif belajar naik secara berangsur hingga pukul 12.30 WIB, namun ketika pengambilan data pada sesi kedua pada pukul 17.00 WIB hingga 23.59, suhu menurun secara berangsur. Gambar 9 merupakan grafik perbandingan kelembapan ruang kelas.



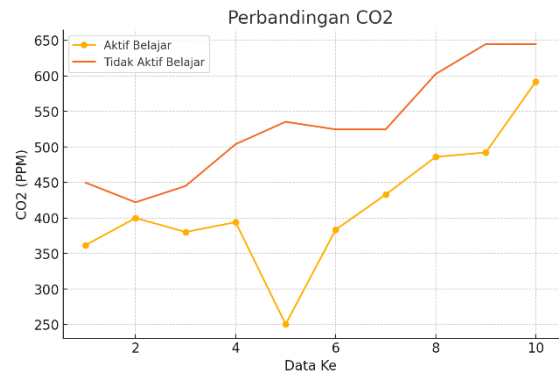
Gambar 9. grafik perbandingan kelembapan

Pada Gambar 9 menjelaskan bahwa sesi pertama atribut kelembapan pada waktu aktif belajar menurun secara berangsur hingga pukul 12.30 WIB, namun ketika pengambilan data pada sesi kedua pada pukul 17.00 WIB hingga 23.59, kelembapan naik secara berangsur. Namun pada kedua sesi, kelembapan menunjukkan angka yang tinggi dan melewati ambang batas rekomendasi, Gambar 10 merupakan grafik perbandingan CO pada ruang kelas.



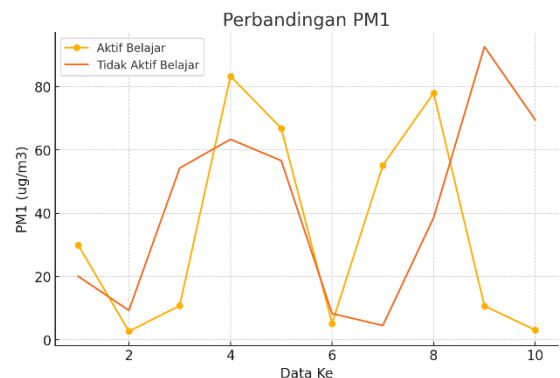
Gambar 10. grafik perbandingan CO

Perbandingan CO pada ruangan kelas pada Gambar 10 menjelaskan bahwa sesi pertama CO pada waktu aktif belajar berada pada nilai 3.5 ppm karena ada aktivitas pembakaran di area sekitar kelas, namun turun dan stabil dibawah 0.5 ppm hingga pukul 12.30 WIB, pengambilan data pada sesi kedua pada pukul 17.00 WIB hingga 23.59, nilai CO stabil karena tidak ada aktivitas pembakaran di area sekitar ruang kelas. Gambar 11 merupakan grafik perbandingan CO₂ ruang kelas.



Gambar 11. grafik perbandingan CO₂

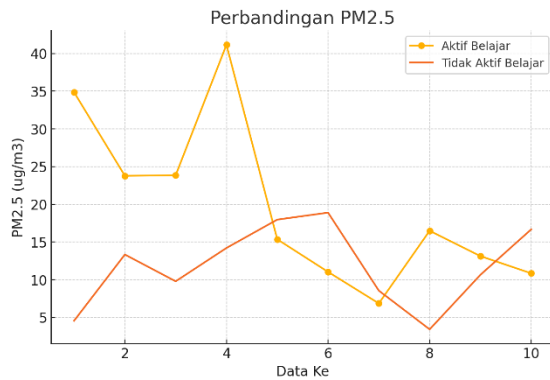
Perbandingan CO₂ pada ruangan kelas pada Gambar 8 menjelaskan bahwa sesi pertama CO₂ pada waktu aktif belajar diuji dalam keadaan pintu kelas terbuka dan menghidupkan kipas angin, sehingga menghasilkan nilai diatas 350 ppm dan dibawah 600 ppm, hal tersebut disebabkan karena aktivitas pembelajaran yang memungkinkan siswa mengeluarkan CO₂. Pengambilan data pada sesi kedua pada pukul 17.00 WIB hingga 23.59, nilai CO₂ relatif lebih tinggi karena pintu cenderung tertutup dan memungkinkan pertukaran udara melalui ventilasi yang kurang. Gambar 12 merupakan grafik perbandingan PM₁ pada ruang kelas.



Gambar 12. grafik perbandingan PM₁

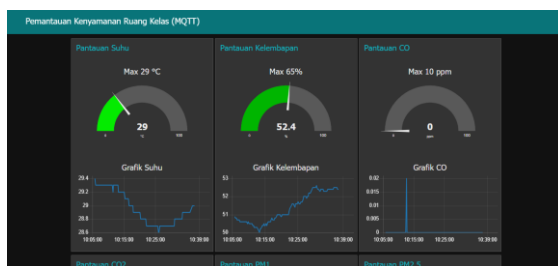
Perbandingan PM₁ pada ruangan kelas pada Gambar 12 menjelaskan bahwa secara umum, PM₁ yang diukur pada kedua sesi menunjukkan perubahan yang tidak begitu signifikan dan tidak tetap. Hal ini disebabkan karena kelas cenderung berdebu, namun hasil pembacaan PM₁ menunjukkan berada pada angka yang aman. Gambar 13 merupakan grafik perbandingan PM_{2.5} pada ruang kelas.

Pada sesi pertama, konsentrasi PM_{2.5} menunjukkan melewati nilai ambang batas rekomendasi namun selebihnya berada dalam zona aman. Pada sesi kedua, konsentrasi terpantau aman dan secara umum kondisi PM_{2.5} berada di zona aman.

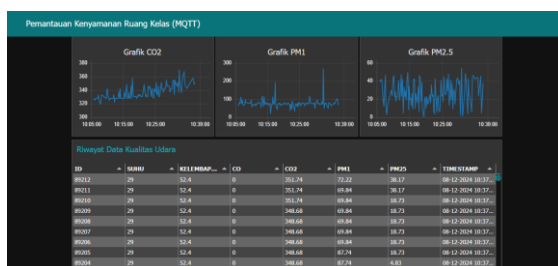
Gambar 13. grafik perbandingan PM_{2.5}

4.4. Pengujian Dashboard

Hasil pengiriman data dari perangkat IoT akan ditampilkan pada *dashboard* dengan tujuan agar mempermudah pengguna untuk melakukan pemantauan dari jarak jauh. Gambar 14(a) dan Gambar 14(b) merupakan hasil pengujian *dashboard* dengan menampilkan data yang dikirimkan dengan menggunakan protokol MQTT.

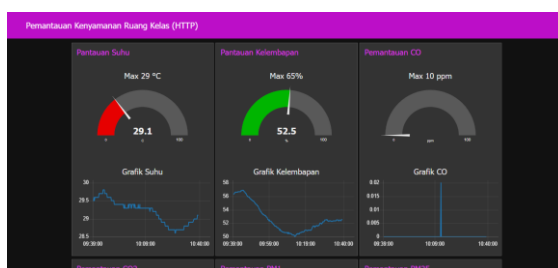


Gambar 14(a). dashboard pada server MQTT

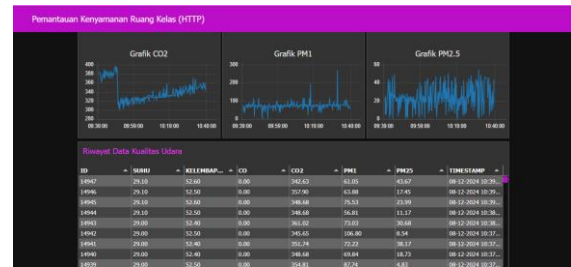


Gambar 14(b). data riwayat kualitas udara pada server MQTT

Gambar 15(a) dan Gambar 15(b) merupakan hasil pengujian *dashboard* dengan menampilkan data yang dikirimkan dengan menggunakan protokol HTTP.



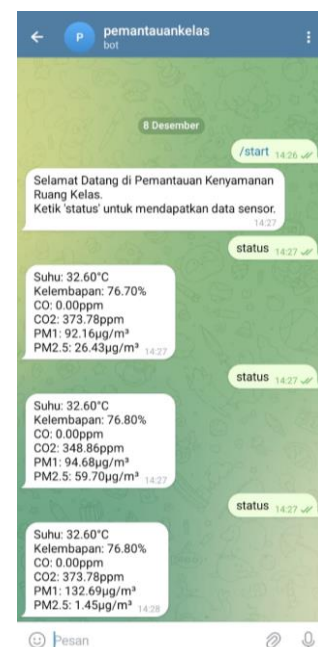
Gambar 15(a). dashboard pada server HTTP



Gambar 15(b). data riwayat kualitas udara pada server HTTP

4.5. Pengujian Notifikasi Pada Telegram

Pengujian notifikasi pada telegram pada penelitian ini mencakup dua hal, yaitu pada saat pengguna melakukan *request* data *real-time* kualitas udara dan pengujian notifikasi ketika salah satu parameter kualitas udara berada diatas ambang batas. Gambar 16 merupakan proses *request* data *real-time* kualitas udara.

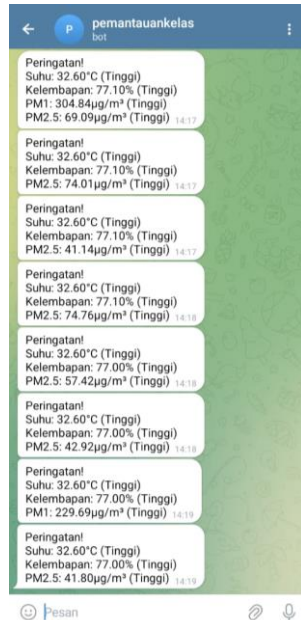


Gambar 16. request status data kualitas udara dari Telegram

Pada Gambar 16 menunjukkan hasil pengujian fitur request data *real-time* melalui bot Telegram dengan mengirimkan kata kunci 'status' sebanyak tiga kali oleh pengguna. Dalam setiap percobaan, bot berhasil merespons dengan cepat dan menampilkan data terkini dari hasil pembacaan sensor kualitas udara, seperti suhu, kelembapan, CO, CO₂, PM₁, dan PM_{2.5}. Tidak ditemukan keterlambatan maupun kegagalan dalam pengiriman pesan balasan, yang menunjukkan bahwa konektivitas antara sistem IoT dan platform Telegram berjalan stabil dan responsif.

Keberhasilan ini membuktikan bahwa sistem mampu mendukung pemantauan berbasis permintaan (*user-triggered monitoring*), yang sangat berguna dalam skenario ketika pengguna ingin memperoleh informasi secara langsung tanpa bergantung pada visualisasi dashboard. Respon *real-time* yang cepat juga menunjukkan bahwa beban

pemrosesan dan pengambilan data dari server cukup ringan, menjadikan fitur ini layak untuk diimplementasikan di lingkungan sekolah yang membutuhkan akses cepat dan mudah terhadap kondisi udara dalam kelas.



Gambar 17. notifikasi peringatan parameter kualitas udara tinggi

Gambar 17 menunjukkan proses pengiriman notifikasi otomatis oleh bot Telegram ketika salah satu parameter kualitas udara melebihi ambang batas yang telah ditentukan. Notifikasi dikirimkan secara berkala setiap 10 menit ketika parameter berada pada kondisi tidak ideal. Fitur ini menandakan bahwa sistem memiliki fungsi monitoring pasif dan aktif secara terus-menerus, sehingga pengguna tidak perlu melakukan pemantauan secara manual setiap saat. Dengan pendekatan ini, sistem dapat meningkatkan kesadaran pengguna terhadap kondisi lingkungan secara preventif. Hal ini sangat relevan untuk lingkungan kelas, di mana paparan udara yang buruk bisa berdampak langsung pada kenyamanan serta kesehatan siswa dan efektivitas proses belajar.

5. KESIMPULAN DAN SARAN

Sistem pemantauan kenyamanan kelas yang dilakukan pengukuran dari aspek kualitas udara berbasis Internet of Things berjalan dengan baik. Implementasi dilakukan pada ruang kelas yang dilakukan pada sesi pertama dengan mengambil 10 sampel data pada waktu yang berbeda menghasilkan kenaikan suhu, penurunan tingkat kelembapan, dari tingkat CO dan CO₂ masih dalam kategori aman. Konsentrasi PM1 dan PM2.5 juga terpantau pada zona aman. Pada sesi kedua, terjadi penurunan suhu namun terjadi kenaikan kelembapan yang tinggi, kadar CO tidak ada perubahan yang signifikan. Namun kadar CO₂ mengalami peningkatan karena seluruh ruangan kelas ditutup dan tingkat sirkulasi

udara yang kecil membuat deteksi CO₂ melalui sensor MQ-135 mengalami kenaikan.

Pada pengujian pengiriman data dengan protokol MQTT pada 10 sampel data percobaan, didapatkan hasil rata-rata waktu pengiriman yaitu 6.2 ms. Namun pada pengiriman data dengan protokol HTTP didapatkan hasil rata-rata 267 ms yang mana lebih lama dibandingkan protokol MQTT, namun kedua protokol dapat mengirimkan data kurang dari 1 detik. Pengujian dashboard baik pada server MQTT dan HTTP berjalan dengan baik. Pengujian notifikasi melalui Telegram mencakup dua hal, yaitu pengujian pengiriman request keadaan kualitas udara secara real-time, percobaan dilakukan sebanyak tiga kali dan tidak ada kendala yang didapatkan. Dari sisi notifikasi terkait peringatan parameter kualitas udara yang nilainya berada di atas ambang dilakukan pengujian dan berjalan dengan baik, notifikasi dikirimkan setiap 10 menit sekali. Perangkat IoT yang dirancang dapat berjalan dengan baik dan berjalan sesuai yang dirancang.

Berdasarkan hasil pengujian, sistem ini memiliki potensi untuk direplikasi di sekolah-sekolah lain, terutama yang berada di wilayah dengan infrastruktur jaringan terbatas, mengingat efisiensi pengiriman data yang ditunjukkan oleh protokol MQTT. Sistem ini juga dapat diadaptasi untuk digunakan pada berbagai ruang tertutup lainnya, seperti ruang guru, laboratorium, perpustakaan, atau ruang pelayanan publik yang membutuhkan pemantauan kualitas udara secara *real-time*. Dengan fleksibilitas sistem dan efisiensi komunikasi data, penelitian ini tidak hanya berkontribusi pada aspek teknis pengembangan IoT, tetapi juga membuka peluang penerapan yang lebih luas dalam konteks peningkatan kenyamanan dan kesehatan lingkungan belajar maupun kerja.

Terdapat saran agar penelitian ini dapat dikembangkan yaitu, karena sedikitnya alat yang dilakukan untuk mengukur tingkat akurasi, perlu dilakukan penelitian lebih lanjut tentang akurasi sensor yang digunakan. Kemudian penelitian ini dapat dikembangkan dengan membandingkan kinerja kedua protokol, terlebih dengan menggunakan dedicated server dengan tujuan agar mendapatkan hasil yang lebih akurat.

DAFTAR PUSTAKA

- 1077/MENKES/PER/V/2011, 2011. *Peraturan Menteri Kesehatan Republik Indonesia*.
- ADANI, F. AND SALSABIL, S., 2020. Internet of Things: Sejarah Teknologi Dan Penerapannya. *Jurnal Isu Teknologi*, 14(2), pp.92–99.
- AS'AR, R.M., 2022. Hubungan Kadar PM2.5 dan PM10 Terhadap Keluhan Dyspnea Warga Desa Lakardowo, Kabupaten Mojokerto, Provinsi Jawa Timur. *Environmental Pollution Journal*, 2(2), pp.419–425.

- AZIZ FIKHRI, A., ULA, M. AND SAYUTI, M., 2025. *Perbandingan Kinerja Protokol MQTT dan HTTP Dalam Komunikasi Data Internet of Things Artikel Penelitian*. <https://doi.org/http://dx.doi.org/10.30811/jim.v10i1.6733>.
- BENDER, M., KIRDAN, E., PAHL, M.O. AND CARLE, G., 2021. Open-source MQTT evaluation. In: *2021 IEEE 18th Annual Consumer Communications and Networking Conference, CCNC 2021*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/CCNC49032.2021.9369499>.
- HANIF, A. AND AMRI, R., 2023. Implementasi Internet Of Things Pada Protokol MQTT Dan HTTP Dalam Sistem Pendeteksi Banjir. *Jurnal Invotek Polbeng - Seri Informatika*, 8(2), pp.489–501.
- HIDAYAYI, Q., ZAINI RACHMAN, F. AND ARIF SURYA RIMBAWAN, M., 2020. Sistem Monitoring Kualitas Udara Berbasis Fuzzy Logic. *ISAS Publishing*, 6(1), pp.260–267.
- KRUNIANTO, A., DEDY IRAWAN, J., ARIWIBISONO, F.X. AND WARDHANA, A., 2022. Penerapan IoT (Internet Of Things) Untuk Controlling Lampu Menggunakan Protokol MQTT Berbasis Web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(2), pp.1153–1161.
- KURNIAWAN, A.R. and WARLINA, L., 2020. Web Service for Academic Information Systems. In: *IOP Conference Series: Materials Science and Engineering*. IOP Publishing Ltd. pp.1–7.
- MISHRA, B. AND KERTESZ, A., 2020. The use of MQTT in M2M and IoT systems: A survey. *IEEE Access*, 8, pp.201071–201086. <https://doi.org/10.1109/ACCESS.2020.3035849>.
- MUNJIRIN, A. AND ISWINARTI, 2023. Faktor-Faktor yang Mempengaruhi Prestasi Akademik Remaja. *Cognicia*, 11(2), pp.106–111.
- NIKOLOV, N., 2020. Research of MQTT, CoAP, HTTP and XMPP IoT Communication protocols for Embedded Systems. In: *2020 29th International Scientific Conference Electronics, ET 2020 - Proceedings*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ET50336.2020.9238208>.
- NUGRAHA, I.R., PUTRA, W.H.N. AND SETIAWAN, E., 2024. A Comparative Study of HTTP and MQTT for IoT Applications in Hydroponics. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 8(1), pp.119–126.
- SETIATI, T.W., FEBRINA, S.E. AND ISLAMI, F.S., 2022. Investigasi Kualitas Udara Ruang Kelas dengan Perubahan Ventilasi Aktif Menjadi Alami Pasca Pandemi di Daerah Tropis Lembab. *Jurnal Arsir Universitas Muhammadiyah Palembang*, 6(2), pp.126–135.
- SURATNO, T., SAPUTRA, E., ABIDIN, Z., ARSA, D. AND SYARIEF, N., 2022. Internet of Things (IoT) Arduino-Based Classroom Monitoring Utilizes Temperature Sensors And CO2 Sensors. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 16(3), p.313. <https://doi.org/10.22146/ijccs.76241>.