

Turnitin LLCC

2213020164_M. Rizal Umami

 TI-15

Document Details

Submission ID

trn:oid::2945:394823561

Submission Date

Jun 23, 2026, 2:04 PM GMT+7

Download Date

Jun 23, 2026, 2:15 PM GMT+7

File Name

2213020164_M. Rizal Umami.pdf

File Size

2.4 MB

66 Pages

11,422 Words

69,319 Characters

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Top Sources

- 9%  Internet sources
- 5%  Publications
- 11%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Top Sources

- 9% Internet sources
- 5% Publications
- 11% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	repository.unpkediri.ac.id	<1%
2	Internet	123dok.com	<1%
3	Student papers	Universitas Siliwangi on 2026-02-13	<1%
4	Student papers	Universitas Islam Riau on 2026-06-15	<1%
5	Internet	repository.its.ac.id	<1%
6	Internet	docplayer.info	<1%
7	Internet	ojs.unpkediri.ac.id	<1%
8	Student papers	Universitas Merdeka Malang on 2026-05-05	<1%
9	Internet	ojs.uma.ac.id	<1%
10	Student papers	Lambung Mangkurat University on 2025-07-11	<1%
11	Internet	repository.ittelkom-pwt.ac.id	<1%

12	Internet	repository.umsu.ac.id	<1%
13	Internet	digilib.uns.ac.id	<1%
14	Student papers	Universitas Negeri Semarang on 2018-03-25	<1%
15	Student papers	Universitas Sebelas Maret on 2025-07-08	<1%
16	Internet	www.trijurnal.lemlit.trisakti.ac.id	<1%
17	Internet	qdoc.tips	<1%
18	Student papers	Fakultas Teknik on 2025-10-28	<1%
19	Student papers	Universitas Sumatera Utara on 2026-06-15	<1%
20	Internet	openjournal.unpam.ac.id	<1%
21	Internet	repository.umy.ac.id	<1%
22	Internet	scholar.unand.ac.id	<1%
23	Internet	suhendar.blog.upi.edu	<1%
24	Student papers	Perpustakaan on 2025-08-07	<1%
25	Student papers	Telkom University on 2025-06-30	<1%

26	Student papers	Telkom University on 2025-07-08	<1%
27	Internet	digilib.unila.ac.id	<1%
28	Internet	repository.polman-babel.ac.id	<1%
29	Student papers	Saddleback High School on 2026-01-26	<1%
30	Student papers	Universitas Brawijaya on 2016-12-21	<1%
31	Student papers	Universitas Muhammadiyah Purwokerto on 2025-12-11	<1%
32	Internet	repository.trisakti.ac.id	<1%
33	Internet	repository.unri.ac.id	<1%
34	Student papers	Departement of Informatics Engineering on 2025-08-25	<1%
35	Student papers	Sekolah Teknik Elektro & Informatika on 2025-07-29	<1%
36	Student papers	Universitas Brawijaya on 2019-12-22	<1%
37	Internet	www.researchgate.net	<1%
38	Student papers	Durban University of Technology on 2026-06-18	<1%
39	Publication	Fajar Ponco Adi Wibowo, Hamid Muhammad Jumasa, Murhadi Murhadi. "Manaje...	<1%

40	Publication	Liberius Harsen Nggala, Nachrowie Nachrowie, Subairi Subairi. "Implementasi M...	<1%
41	Student papers	Universitas Brawijaya on 2017-08-04	<1%
42	Student papers	Universitas Negeri Semarang on 2026-05-03	<1%
43	Internet	digilib.uinsby.ac.id	<1%
44	Internet	informatika.unpkediri.ac.id	<1%
45	Internet	repository.umb.ac.id	<1%
46	Student papers	Universitas Putera Batam on 2018-11-30	<1%
47	Internet	digilib.iain-palangkaraya.ac.id	<1%
48	Internet	library.binus.ac.id	<1%
49	Internet	repositori.uin-alaudidin.ac.id	<1%
50	Internet	repository.poltekbangjayapura.ac.id	<1%
51	Student papers	Politeknik Negeri Bandung on 2020-09-24	<1%
52	Student papers	Telkom University on 2025-07-03	<1%
53	Student papers	Universitas Negeri Jakarta on 2017-08-21	<1%

54	Internet	eprints.utm.my	<1%
55	Internet	etheses.uin-malang.ac.id	<1%
56	Internet	jurnal.unpad.ac.id	<1%
57	Internet	repository.ub.ac.id	<1%
58	Internet	studentpasca.budiluhur.ac.id	<1%
59	Internet	tedytirta.files.wordpress.com	<1%
60	Publication	Jiyan Suhada, Hari Setyawan. "Analisis Perbandingan Kinerja Metode Load Balan...	<1%
61	Student papers	Saddleback High School on 2025-09-18	<1%
62	Publication	Supiana Doni. "Pendidikan Matematika II", Open Science Framework, 2022	<1%
63	Publication	Suseno Suseno, Satyo Hega Wibowo. "PENENTUAN KAPASITAS PRODUKSI MENGG...	<1%
64	Student papers	Telkom University on 2025-07-03	<1%
65	Student papers	Universitas Bengkulu on 2024-07-23	<1%
66	Student papers	Universitas Brawijaya on 2017-01-13	<1%
67	Student papers	Universitas Putera Batam on 2018-11-30	<1%

68	Student papers	Universitas Putera Batam on 2019-05-23	<1%
69	Internet	dspace.uui.ac.id	<1%
70	Internet	pauzan.com	<1%
71	Student papers	Telkom University on 2025-06-24	<1%
72	Student papers	UPN Veteran Yogyakarta on 2026-05-04	<1%
73	Student papers	Universitas Brawijaya on 2017-06-20	<1%
74	Student papers	Universitas Brawijaya on 2018-07-14	<1%
75	Student papers	Universitas Brawijaya on 2018-07-18	<1%
76	Student papers	Universitas Brawijaya on 2021-06-14	<1%
77	Student papers	Universitas Brawijaya on 2023-12-29	<1%
78	Student papers	Universitas Islam Negeri Sumatera Utara on 2025-08-29	<1%
79	Student papers	Universitas Putera Batam on 2020-12-02	<1%
80	Internet	digilib.uinsgd.ac.id	<1%
81	Internet	eprints.polsri.ac.id	<1%

82	Internet	idjpcr.usu.ac.id	<1%
83	Internet	repositori.kemdikbud.go.id	<1%
84	Internet	repository.upi.edu	<1%
85	Internet	www.scribd.com	<1%
86	Publication	Jumawal, Harianto, Mahpuz, Muhammad Wasil. "Perbandingan Performa Senso...	<1%
87	Student papers	Universitas Brawijaya on 2018-03-07	<1%
88	Student papers	Universitas Brawijaya on 2018-07-03	<1%
89	Student papers	Universitas Brawijaya on 2021-06-17	<1%
90	Student papers	unas on 2025-04-16	<1%
91	Student papers	Universitas Bengkulu on 2026-01-15	<1%
92	Student papers	Universitas Brawijaya on 2022-01-13	<1%
93	Student papers	Universitas Dian Nuswantoro on 2016-06-27	<1%
94	Student papers	Universitas Putera Batam on 2026-01-31	<1%

BAB I

PENDAHULUAN

A. Latar Belakang

Panel surya sebagai salah satu sumber energi terbarukan memegang peranan penting dalam upaya transformasi energi menuju energi bersih dan berkelanjutan di Indonesia. Sistem Pembangkit Listrik Tenaga Surya (PLTS) telah menjadi solusi energi utama untuk daerah terpencil maupun sebagai penunjang sistem energi yang sudah ada (Afif & Martin, 2022). Namun, kinerja panel surya dipengaruhi oleh berbagai faktor seperti kerusakan fisik, penumpukan debu, kotoran hewan, dan faktor lingkungan lainnya yang dapat menurunkan efisiensi konversi energi secara signifikan (Meilany *et al.*, 2024).

Berbagai inovasi telah dikembangkan untuk mengatasi penurunan efisiensi panel surya akibat kotoran dan debu, seperti sistem pembersih mekanis berbasis *rolling brush* dan *wiper* (Sari *et al.*, 2022). Namun, sistem tersebut masih bergantung pada *timer*, sehingga pembersihan dilakukan tanpa mempertimbangkan tingkat kekotoran aktual. Penelitian (Cahyanti *et al.*, 2025) sudah mampu mendeteksi kekotoran menggunakan sensor warna, tetapi masih terbatas pada fungsi pembersihan tanpa didukung monitoring performa panel dan penyesuaian pembersihan terhadap kondisi lingkungan.

Berdasarkan permasalahan tersebut, telah dikembangkan sebuah prototipe yang dipresentasikan dalam berbagai kompetisi, di nasional maupun internasional pada tahun 2025. Prototipe tersebut berjudul “Deteksi dan Monitoring Kebersihan Panel Surya Menggunakan YOLOv8 Berbasis IoT untuk Optimasi Efisiensi” yang dirancang oleh Dosen Julian Sahertian, S.Pd., M.T. bersama tim mahasiswa M. Rizal Umami, Ilham Khefi Ramadhanu, dan M. Choirul Anwar.

Prototipe (*Smart Solar Panel*) berhasil meningkatkan efisiensi penyerapan energi melalui deteksi dan monitoring kebersihan panel secara *real-time* menggunakan YOLOv8 berbasis IoT. Sistem ini mengidentifikasi kotoran dan menampilkan performa panel kepada pengguna. Namun, prototipe masih

terbatas karena pembersihan hanya dapat diaktifkan pada tombol *cleaning* pada aplikasi *desktop*, sehingga tidak bekerja otomatis meskipun terjadi penurunan daya. Firebase berfungsi sebagai pelaporan data tanpa komunikasi dua arah untuk kendali perangkat secara langsung (Padmaja *et al.*, 2021).

Maka dari itu, penelitian ini berfokus pada pengembangan sistem pembersihan otomatis pada *Smart Solar Panel* melalui penerapan *fuzzy logic* untuk menentukan kebutuhan pembersihan berdasarkan data pembacaan sensor BH-1750, DHT-11, dan *Voltage* Sensor. Output sistem berupa aktivasi motor pembersih yang bekerja secara otomatis sesuai kondisi aktual panel. Selain itu, protokol (MQTT) diterapkan sebagai pengganti firebase karena (MQTT) mendukung komunikasi dua arah yang lebih andal dan berlatensi rendah untuk pengendalian perangkat IoT secara *real-time*.

B. Identifikasi Masalah

Merujuk pada latar belakang, prototipe *Smart Solar Panel* belum dapat menentukan pembersihan secara otomatis, pengguna masih menyimpulkan sendiri data sensor (*intensitas cahaya, suhu, dan tegangan*) dan menekan tombol *cleaning* secara manual meskipun performansi panel sudah menurun.

C. Rumusan Masalah

Berdasarkan masalah yang diidentifikasi, muncul sejumlah pertanyaan yang menjadi fokus utama penelitian ini, antara lain:

1. Bagaimana menerapkan *Fuzzy Logic* pada prototipe *Smart Solar Panel*?
2. Bagaimana menerapkan *Internet of Things* dengan protokol MQTT pada prototype *Smart Solar Panel*?

D. Batasan masalah

Guna memastikan fokus yang jelas dalam penelitian yang akan dilakukan, diperlukan batasan-batasan masalah yang jelas. Adapun batasan-batasan masalah sebagai berikut:

1. Objek penelitian merupakan prototipe “*Smart Solar Panel*” yang telah dikembangkan dan dilombakan pada kompetisi tingkat nasional tahun 2025.

2. Fokus penelitian ini adalah implementasi *fuzzy logic* untuk menentukan kebutuhan pembersihan panel surya berdasarkan pembacaan sensor.
3. Mikrokontroler yang digunakan dalam penelitian ini adalah ESP-32.
4. Protokol komunikasi IoT yang digunakan adalah MQTT (Message Queue Telemetry Transport).
5. Sensor yang digunakan meliputi BH-1750 (intensitas cahaya), Voltage Sensor (tegangan), dan DHT-11 (suhu).
6. Sistem kendali yang dikembangkan bertujuan untuk mengaktifkan motor pembersih secara otomatis berdasarkan hasil keputusan *fuzzy logic*.
7. Penelitian ini tidak membahas analisis desain mekanik sistem pembersih maupun perhitungan efisiensi daya panel surya secara keseluruhan.

E. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, tujuan dari penelitian ini adalah:

1. Menerapkan *fuzzy logic* pada prototipe *Smart Solar Panel* untuk menentukan kebutuhan pembersihan berdasarkan data sensor.
2. Menerapkan sistem *Internet of Things* dengan protokol MQTT sebagai komunikasi dua arah pada prototipe *Smart Solar Panel* untuk mendukung pembersihan otomatis *real-time*.

F. Manfaat Penelitian

Adapun manfaat dan kegunaan penelitian dapat dirasakan oleh beberapa pihak terkait dalam penelitian sebagai berikut:

1. Bagi peneliti, penelitian ini menjadi sarana penerapan ilmu pada bidang *Internet of Things* dan *Artificial Intelligence*, khususnya pada penerapan *fuzzy logic*.
2. Prototipe “*Smart Solar Panel*” memiliki fitur pembersihan otomatis yang dapat membantu pengguna tanpa perlu melakukan analisis manual terhadap data sensor.
3. Mengurangi penurunan efisiensi panel surya akibat penumpukan kotoran melalui pembersihan yang tepat waktu sesuai kondisi lingkungan.

4. Menghadirkan pengembangan sistem panel surya berbasis IoT yang lebih adaptif dan dapat melakukan monitoring sekaligus pengendalian secara mandiri.

44

BAB II

LANDASAN TEORI

A. Teori dan Penelitian Terdahulu

1. Landasan Teori

a. Prototipe *Smart Solar Panel*



Gambar 2. 1 Prototipe *Smart Solar Panel*

Prototipe *Smart Solar Panel* merupakan sistem pemantauan dan pembersih otomatis untuk panel surya yang menggabungkan teknologi deteksi berbasis AI dengan aktuator mekanis. Prototipe dirancang oleh Dosen Julian Sahertian, S.Pd., M.T. bersama tim mahasiswa M. Rizal Umami, Ilham Khefi Ramadhanu, dan M. Choirul Anwar. Sistem ini menggunakan model *object detection You Only Look Once (YOLO) versi 8 (YOLOv8)* untuk mendeteksi kotoran dan kerusakan pada permukaan panel.

Untuk proses deteksi, gambar dari panel surya ditangkap menggunakan ESP32 CAM, sebuah mikrokontroler dengan kamera bawaan yang berfungsi sebagai IP camera. Data visual yang diperoleh kemudian diproses oleh model YOLOv8 yang mampu mengidentifikasi dan mengklasifikasikan objek yang terdeteksi secara cepat dan akurat (Hayati *et al.*, 2023)

Untuk mempermudah interaksi pengguna, sistem ini juga dilengkapi dengan *Graphical User Interface (GUI)* berbasis *desktop*

84

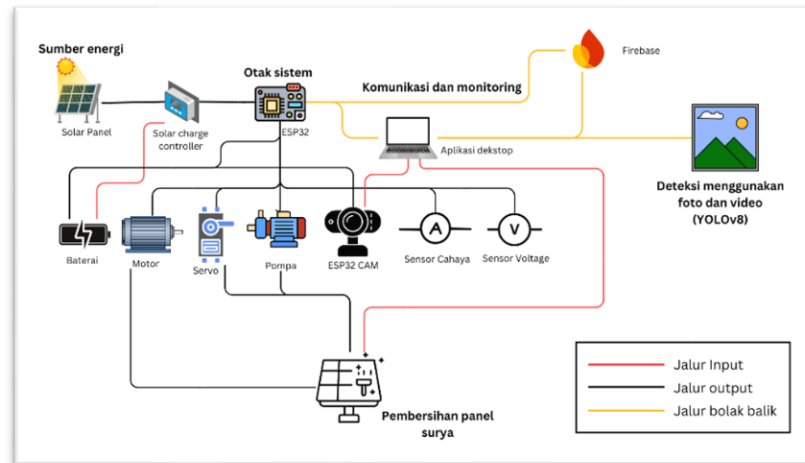
54

yang dikembangkan menggunakan *library* Python Tkinter, sebuah pustaka GUI standar yang ringan dan telah tersedia secara *default* di sebagian besar distribusi Python (Ilham & Avorizano, 2025).

Untuk proses pembersihan, sistem menggunakan dua motor servo 360° yang dihubungkan dengan sikat akuarium sebagai alat pembersih mekanis yang bergerak melintasi permukaan panel. Proses ini didukung oleh pompa air DC, yang menyemburkan air ke permukaan panel untuk membantu mengangkat debu dan kotoran sebelum disikat. Keseluruhan sistem dikendalikan oleh ESP32, yang mengatur kerja motor servo, pompa air, serta pergerakan *pan-tilt* kamera. Pengoperasian alat dapat dilakukan secara jarak jauh melalui aplikasi, yang memungkinkan pengguna memulai proses pembersihan, mengatur sudut pandang kamera, serta memantau hasil deteksi secara *real-time*. Sistem ini menggunakan Firebase sebagai *real-time* database, yang memungkinkan komunikasi antara alat dan aplikasi.

Implementasi dari prototipe ini telah dilakukan dalam penelitian berjudul “Pemanfaatan Sistem Panel Surya Berbasis IoT Sebagai Solusi Alternatif Penghematan Listrik Di Green House Kokonot Plant” (Helilintar *et al.*, 2025). Dalam penelitian tersebut, prototipe ini digunakan untuk menjaga efisiensi penyerapan energi surya di lingkungan perkebunan, yang membuktikan bahwa mekanisme pembersihan berbasis IoT ini efektif dalam mempertahankan stabilitas tegangan listrik dibandingkan sistem panel surya konvensional tanpa perawatan otomatis.

Berdasarkan perancangan mekanisme kerja sistem yang diterapkan pada penelitian terdahulu dapat diilustrasikan melalui gambar 2.2 berikut.



Gambar 2. 2 Arsitektur Sistem *Smart Solar Panel*

b. *Internet of Things*

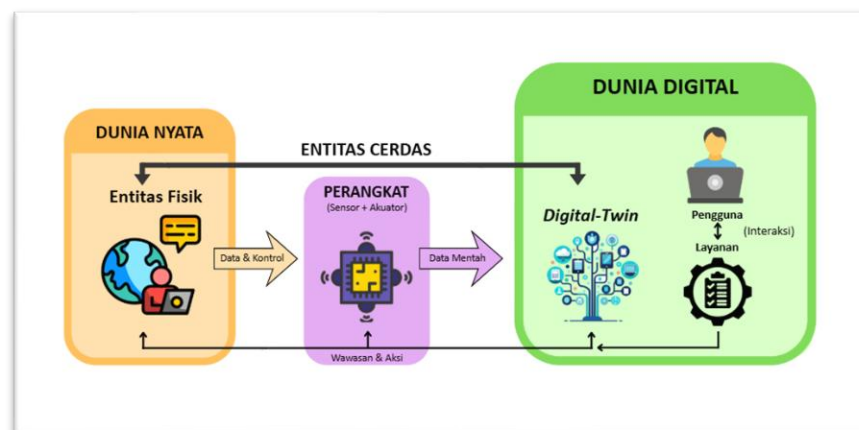
Internet of Things (IoT) merupakan konsep yang menghubungkan berbagai perangkat fisik ke jaringan internet sehingga mampu saling berkomunikasi, bertukar data, dan melakukan tindakan secara otomatis tanpa intervensi manusia secara langsung. Teknologi ini memungkinkan objek fisik seperti sensor, aktuator, dan mikrokontroler berfungsi sebagai entitas cerdas yang dapat melakukan identifikasi, pemantauan, dan pengendalian dari jarak jauh melalui jaringan komunikasi berbasis internet. Penelitian (Hidayati, 2024), IoT berperan sebagai jembatan antara dunia fisik dan dunia digital melalui integrasi perangkat, jaringan, serta platform komputasi yang memungkinkan data terkirim secara *real-time* untuk mendukung pengambilan keputusan.

Sedangkan Penelitian (Witczak & Szymoniak, 2024) menegaskan bahwa IoT tidak hanya berfungsi sebagai sistem monitoring, tetapi juga mampu mewujudkan otomatisasi tindakan, sehingga IoT memiliki nilai signifikan dalam peningkatan efisiensi pemanfaatan perangkat dan sumber daya. Berdasarkan penelitian (Suhardi *et al.*, 2024) perkembangan IoT semakin masif karena memadukan teknologi *cloud*, *wireless sensor network* (WSN), dan protokol komunikasi ringan yang menjadikan transfer data lebih cepat serta konsumsi daya lebih efisien.

1) Karakteristik *Internet of Things*

Karakteristik IoT pada model *Digital Twin* yang dijelaskan oleh (Schweiger & Barth, 2023), memiliki 3 karakteristik utama:

- Perangkat IoT menciptakan tiruan virtual (*Digital Twin*) di dunia digital yang mencerminkan kondisi terkini dari entitas fisik di dunia nyata secara *real-time*.
- IoT menghubungkan kedua dunia melalui perangkat keras, sensor bertugas mengirim data mentah ke sistem digital, dan aktuator bertugas menerima perintah digital untuk melakukan aksi fisik.
- Sistem IoT tidak hanya mengolah data, tetapi mengubahnya menjadi layanan (*services*) yang dapat diakses dan digunakan oleh pengguna (*user*) untuk berinteraksi atau mengambil keputusan.



Gambar 2. 3 Karakteristik *Internet of Things*

2) Arsitektur *Internet of Things*

Sistem *Internet of Things* (IoT) dapat bekerja secara optimal dalam menghubungkan dunia fisik dan digital, diperlukan struktur arsitektur yang tersusun secara hierarkis. Dalam penelitian (Dauda *et al.*, 2024), arsitektur IoT dibagi menjadi 4 lapisan utama:

- Perception Layer* (Lapisan Persepsi)

Lapisan ini, yang sering disebut sebagai lapisan penginderaan, berfungsi sebagai fondasi utama dari sistem IoT. Tugas utamanya adalah mengenali objek dan mengumpulkan informasi dari lingkungan fisik melalui perangkat keras seperti sensor, RFID, dan barcode. Sensor pada lapisan ini bertindak layaknya panca indera manusia, menangkap fenomena fisik (seperti suhu, getaran, lokasi, atau detak jantung) dan mengubahnya menjadi sinyal digital yang dapat diproses lebih lanjut oleh sistem.

b) *Network Layer* (Lapisan Jaringan)

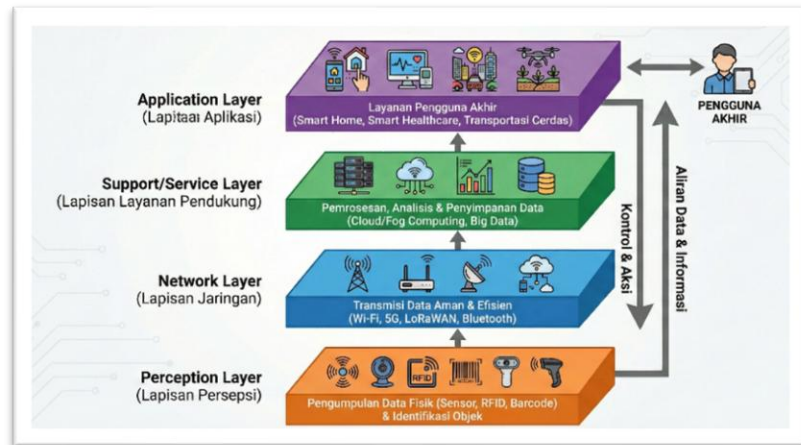
Setelah data berhasil dikumpulkan oleh lapisan persepsi, data tersebut harus ditransmisikan ke pusat pemrosesan. Lapisan jaringan berfungsi untuk menjamin konektivitas dan pengiriman data yang aman dan efisien dari perangkat ke server atau cloud. Lapisan ini memanfaatkan berbagai standar komunikasi, baik nirkabel maupun kabel, seperti Wi-Fi, Bluetooth, ZigBee, 4G/5G, hingga LoRaWAN, tergantung pada kebutuhan jarak dan konsumsi daya perangkat.

c) *Support/ Service Layer* (Lapisan Layanan Pendukung)

Lapisan ini berperan sebagai penghubung cerdas antara jaringan dan aplikasi. Fungsinya adalah memproses data dalam jumlah besar yang diterima dari lapisan jaringan. Lapisan ini memastikan bahwa data mentah diubah menjadi informasi yang terstruktur sehingga siap digunakan oleh lapisan aplikasi di atasnya untuk pengambilan keputusan.

d) *Application Layer* (Lapisan Aplikasi)

Lapisan teratas yang berinteraksi langsung dengan pengguna akhir (*end-user*). Lapisan aplikasi menyajikan hasil olahan data dalam bentuk antarmuka atau layanan yang spesifik sesuai dengan kebutuhan industri.



Gambar 2. 4 Arsitektur *Internet of Things*

c. *Fuzzy Logic*

Logika *Fuzzy* adalah metodologi sistem kendali pemecahan masalah yang meniru cara manusia mengambil keputusan. Berbeda dengan logika *Boolean* konvensional yang hanya mengenal dua nilai kebenaran mutlak yaitu “Benar” (1) atau “Salah” (0), logika *fuzzy* memperkenalkan konsep “derajat kebenaran” yang bernilai antara 0 hingga 1. Penelitian (Muttaqi *et al.*, 2024) memungkinkan penanganan ketidakpastian dan ketidaktepatan data yang sering ditemui pada sistem fisik, seperti pembacaan sensor suhu yang tidak selalu stabil atau perubahan cahaya yang gradual. Dalam sistem kendali IoT, logika *fuzzy* digunakan untuk memetakan ruang input menjadi ruang output melalui proses penalaran yang menyerupai bahasa alami manusia, seperti “Jika suhu sedikit panas, maka kipas berputar pelan”.

1) *Membership Function*

Elemen dasar dari logika *fuzzy* adalah himpunan *fuzzy*. Jika pada himpunan tegas (*crisp set*) sebuah elemen hanya memiliki dua kemungkinan (menjadi anggota atau tidak), pada himpunan *fuzzy* setiap elemen memiliki derajat keanggotaan (μ). Kurva yang merepresentasikan pemetaan titik input data ke dalam nilai keanggotaannya (antara 0 sampai 1) disebut Fungsi Keanggotaan (*Membership Function*).

Secara matematis, derajat keanggotaan $\mu(x)$ untuk kurva segitiga dirumuskan sebagai berikut:

$$\mu(x) = \begin{cases} 0, & \text{Jika } x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}, & \text{Jika } a < x \leq b \\ \frac{c-x}{c-b}, & \text{Jika } b < x < c \end{cases} \quad \dots\dots\dots (2.1)$$

Di mana:

x = Nilai masukan dari sensor (input tegas).

$\mu(x)$ = Nilai derajat keanggotaan.

2) Arsitektur Sistem Inferensi Fuzzy (FIS)

Proses pengolahan data dalam logika fuzzy dilakukan melalui Sistem Inferensi Fuzzy (*Fuzzy Inference System/ FIS*). Menurut (Muttaqi *et al.*, 2024), terdapat 4 tahapan dalam alur kerja FIS metode Mamdani dalam sistem kendali:

a) Fuzzifikasi (*Fuzzification*)

Fuzzifikasi adalah tahap proses konversi data masukan tegas (*crisp input*) yang diperoleh dari sensor (misalnya: suhu 32°C) menjadi nilai fuzzy (*fuzzy input*) menggunakan fungsi keanggotaan yang telah ditetapkan.

b) Basis Aturan (*Rule Base*)

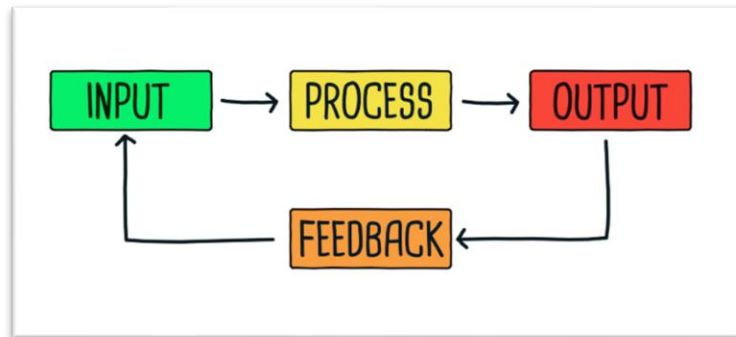
Sistem ini berisi sekumpulan aturan logika "JIKA-MAKA" (*IF-THEN rules*) yang mendefinisikan hubungan antara input dan output.

c) Mesin Inferensi (*Inference Engine*)

Mesin Inferensi adalah tahap derajat keanggotaan dari input dievaluasi terhadap aturan-aturan yang ada. Dalam metode Mamdani, fungsi implikasi yang sering digunakan adalah fungsi MIN (mengambil nilai minimum dari premis aturan).

d) Defuzzifikasi (*Defuzzification*):

Defuzzifikasi adalah tahap akhir proses pengubahan kembali besaran fuzzy (hasil inferensi) menjadi nilai tegas (*crisp output*) yang dapat dimengerti oleh aktuator.



Gambar 2. 5 Alur Proses *Fuzzy Inference System*

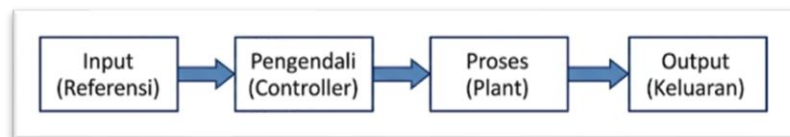
d. Sistem Kendali

Sistem kendali merupakan cabang ilmu rekayasa yang mempelajari hubungan antara perangkat masukan, pemrosesan, dan keluaran untuk mengatur perilaku sistem dinamis agar sesuai dengan nilai yang diinginkan. Dalam Penelitian (Reddy *et al.*, 2024) mendefinisikan sistem kendali sebagai susunan komponen fisik yang terhubung sedemikian rupa untuk memerintahkan, mengarahkan, atau mengatur sistem itu sendiri atau sistem lain. Seiring dengan berkembangnya *Internet of Things* (IoT), sistem kendali konvensional telah bertransformasi menjadi sistem kendali berbasis jaringan, di mana aliran informasi antara kontroler dan perangkat fisik terjadi melalui protokol komunikasi data. Berdasarkan keberadaan jalur umpan balik (*feedback*), sistem kendali diklasifikasikan secara fundamental menjadi dua kategori utama, yaitu sistem kendali loop terbuka (*open-loop*) dan sistem kendali loop tertutup (*closed-loop*)

1) Sistem Kendali Loop Terbuka (*Open-Loop Control*)

Sistem kendali loop terbuka adalah jenis arsitektur pengendalian yang paling mendasar di mana aksi kontrol yang dihasilkan oleh pengendali bersifat independen dan tidak dipengaruhi oleh keluaran proses. Penelitian (Khairunnisah *et al.*,

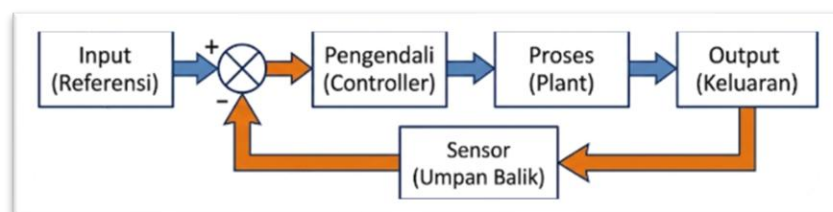
2024), karakteristik dari ketiadaan mekanisme umpan balik untuk memverifikasi apakah keluaran aktual telah sesuai dengan perintah yang diberikan. Dalam operasionalnya, sinyal masukan akan diproses oleh kontroler untuk menghasilkan perintah langsung kepada aktuator berdasarkan basis waktu atau pemicu manual, tanpa memperhitungkan gangguan eksternal yang mungkin mempengaruhi hasil akhir.



Gambar 2. 6 Alur Open-Loop Control

2) Sistem Kendali Loop Tertutup (*Closed-Loop Control*)

Berbeda dengan sistem loop terbuka, sistem kendali loop tertutup atau sering disebut sebagai sistem kendali umpan balik (*feedback control system*) memiliki mekanisme koreksi otomatis untuk menjaga kestabilan sistem. (Khairunnisah *et al.*, 2024) menjelaskan bahwa sistem ini memanfaatkan elemen penginderaan (sensor) untuk memantau variabel proses secara terus-menerus dan mengirimkan data tersebut kembali ke unit pengendali. Prinsip kerja utamanya didasarkan pada perhitungan nilai kesalahan (*error*), yaitu selisih antara nilai acuan yang diinginkan (*set point*) dengan nilai aktual yang terbaca oleh sensor.



Gambar 2. 7 Alur Closed-Loop Control

e. ESP32

ESP32 merupakan sebuah sistem mikrokontroler berjenis *System on Chip* (SoC) yang dikembangkan oleh Espressif Systems

sebagai penerus dari modul ESP8266. Menurut penelitian yang dilakukan oleh (Sri Surani *et al.*, 2025) keunggulan utama ESP32 terletak pada arsitekturnya yang menggabungkan modul Wi-Fi dan *Bluetooth Low Energy* (BLE) dalam satu keping *chip*, sehingga memungkinkan perangkat ini untuk melakukan komunikasi data secara nirkabel dengan efisiensi daya yang tinggi.

Selain kemampuan pemrosesan yang cepat, (Widyatmika *et al.*, 2021) menjelaskan bahwa ESP32 memiliki kapasitas memori yang mumpuni, yaitu 520 KB SRAM internal dan dukungan memori flash eksternal hingga 16 MB. Fitur ini sangat krusial untuk menyimpan program yang besar dan menangani *buffer* data saat pengiriman informasi ke server. Kelebihan lain yang ditekankan dalam literatur adalah ketersediaan pin *Input/Output* (GPIO) yang multifungsi, termasuk fitur *ADC (Analog to Digital Converter)* beresolusi 12-bit yang memberikan tingkat presisi pembacaan sensor analog yang jauh lebih baik dibandingkan mikrokontroler generasi sebelumnya yang hanya memiliki resolusi 10-bit.



Gambar 2. 8 ESP32

f. ***Message Queuing Telemetry Transport* (MQTT)**

MQTT adalah protokol komunikasi ringan berbasis standar ISO (ISO/IEC PRF 20922) yang bekerja pada lapisan aplikasi (*Application Layer*) dalam model OSI. Menurut studi komparatif

3 protokol IoT yang dilakukan oleh (Mishra & Kertesz, 2020), MQTT dirancang khusus untuk komunikasi mesin-ke-mesin (M2M) pada kondisi jaringan yang memiliki *bandwidth* terbatas, *latensi tinggi*, atau *koneksi yang tidak stabil*. Berbeda dengan protokol HTTP yang menggunakan konsep *request-response* yang berat, MQTT beroperasi di atas protokol TCP/IP namun dengan *overhead* paket data yang sangat kecil (sekitar 2 byte untuk header tetap), menjadikannya sangat efisien untuk perangkat IoT bertenaga baterai seperti ESP32.

1) *Arsitektur Publish-Subscribe*

75 Karakteristik utama yang membedakan MQTT dari protokol konvensional adalah penggunaan arsitektur *Publish-Subscribe*. (Pratama *et al.*, 2023) dalam penelitiannya menjelaskan bahwa model ini memisahkan (*decoupling*) antara *pengirim pesan (Publisher)* dan *penerima pesan (Subscriber)*. Dalam mekanisme ini, *pengirim dan penerima tidak perlu mengetahui keberadaan satu sama lain dan tidak harus terhubung pada waktu yang bersamaan*. Komunikasi keduanya dijembatani oleh komponen pusat yang disebut MQTT Broker.

41 41 Broker bertindak sebagai “kantor pos” digital yang bertugas menerima semua pesan dari *Publisher*, memfilter pesan tersebut, dan mendistribusikannya kepada *Subscriber* yang telah mendaftarkan diri pada topik tertentu. (Mishra & Kertesz, 2020) menambahkan bahwa struktur ini memungkinkan skalabilitas yang tinggi, di mana satu perangkat sensor dapat mengirim data ke ribuan perangkat penerima (seperti *dashboard* laptop atau aplikasi seluler) secara bersamaan tanpa membebani kinerja sensor itu sendiri.

2) *Topic Message*

Dalam protokol MQTT, data ditransmisikan melalui kanal logis yang disebut sebagai “Topik” (*Topic*). Dalam Penelitian (Wardihani *et al.*, 2024), topik dalam MQTT bersifat hierarkis dan

menyerupai struktur folder pada sistem operasi komputer yang dipisahkan oleh tanda garis miring (*forward slash*).

3) *Quality of Service/ QoS*

Salah satu fitur krusial dalam protokol MQTT untuk menjamin keandalan data adalah *Quality of Service* (QoS). Berdasarkan analisis performa jaringan oleh (Wardihani et al., 2024) MQTT menyediakan tiga level pengiriman data antara *Publish-Subscribe*, yaitu:

a) QoS Level 0 (*At most once*)

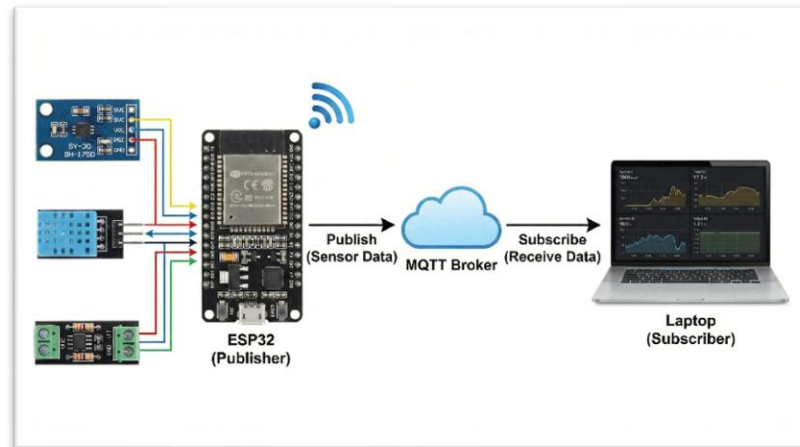
Sebagai metode "*fire and forget*". Pesan dikirimkan satu kali tanpa jaminan bahwa pesan tersebut sampai ke penerima. Tidak ada konfirmasi (*acknowledgement*), sehingga ini adalah level pengiriman tercepat namun paling tidak andal jika jaringan buruk.

b) QoS Level 1 (*At least once*)

Protokol menjamin bahwa pesan pasti sampai ke penerima setidaknya satu kali. Pengirim akan terus mengirim ulang pesan sampai menerima konfirmasi (PUBACK) dari penerima. Namun, terdapat kemungkinan duplikasi pesan yang diterima.

c) QoS Level 2 (*Exactly once*)

Level ini menawarkan jaminan pengiriman tertinggi di mana pesan dipastikan sampai tepat satu kali tanpa duplikasi. Mekanisme ini menggunakan proses *handshake* empat langkah yang lebih kompleks dan memakan *bandwidth* lebih besar, namun sangat penting untuk aplikasi kritis di mana data tidak boleh hilang atau ganda.



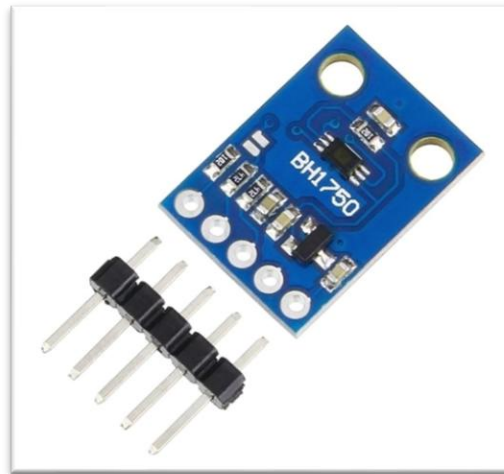
Gambar 2. 9 Arsitektur MQTT

g. Sensor BH-1750

Sensor BH-1750 adalah modul sensor cahaya digital yang dirancang untuk mengukur intensitas iluminasi lingkungan dalam satuan Lux. Berbeda dengan sensor cahaya berbasis resistansi seperti *Light Dependent Resistor* (LDR) yang memerlukan kalibrasi analog yang rumit, (Rahma *et al.*, 2025) menyatakan bahwa BH-1750 menggunakan antarmuka komunikasi serial I2C (*Inter-Integrated Circuit*) yang langsung menghasilkan data keluaran digital. Hal ini dimungkinkan karena di dalam sensor ini terdapat konverter ADC 16-bit internal yang mampu mengubah intensitas cahaya analog menjadi sinyal digital secara presisi, sehingga data yang dihasilkan lebih stabil dan tidak mudah terpengaruh oleh *noise* atau gangguan sinyal elektronik.

Dalam penerapannya pada sistem pemantauan lingkungan, (Rahma *et al.*, 2025) menemukan bahwa sensor BH-1750 memiliki rentang pengukuran yang sangat luas, mulai dari 1 hingga 65.535 Lux, dengan resolusi pengukuran yang tinggi. Kemampuan ini memungkinkan sensor untuk mendeteksi perubahan cahaya yang halus sekalipun, mulai dari kondisi gelap gulita hingga di bawah terik matahari langsung. Karakteristik respons spektral sensor ini juga dirancang mendekati sensitivitas mata manusia, menjadikannya

komponen yang ideal untuk aplikasi pertanian cerdas (*smart farming*) guna memantau kebutuhan cahaya pada tanaman secara akurat.



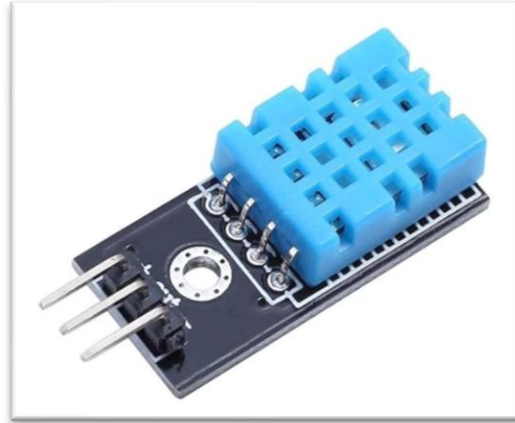
Gambar 2. 10 Sensor BH-1750

h. Sensor DHT-11

DHT-11 adalah sensor gabungan yang berfungsi untuk mendeteksi parameter suhu dan kelembaban udara secara bersamaan dalam satu paket modul. Penelitian (Putri et al., 2025), prinsip kerja sensor ini memanfaatkan komponen resistif untuk mengukur kelembaban dan termistor NTC (*Negative Temperature Coefficient*) untuk mengukur suhu. Data dari kedua elemen tersebut kemudian diolah oleh mikrokontroler 8-bit yang tertanam di dalam sensor untuk dikirimkan melalui protokol komunikasi *single-wire* (satu kabel data). Desain ini menjadikan DHT-11 sebagai solusi yang sangat efisien dari segi penggunaan pin pada mikrokontroler utama.

Meskipun dikenal ekonomis, (Setiawati & Haryanti, 2025) dalam penelitiannya bahwa DHT-11 memiliki karakteristik keterlambatan (*delay*) pengambilan data sekitar dua detik, yang berarti sensor ini lebih cocok digunakan untuk pemantauan lingkungan yang perubahannya tidak terlalu drastis. Sensor ini mampu mengukur kelembaban dengan rentang 20% hingga 90% RH dan suhu antara 0°C hingga 50°C. Walaupun akurasinya tidak setinggi seri DHT-22, tingkat

presisi yang ditawarkan DHT-11 dinilai sudah cukup memadai untuk aplikasi pemantauan suhu ruangan standar atau prototipe sistem kendali iklim mikro sederhana.



Gambar 2. 11 Sensor DHT-11

i. Voltage Sensor

Modul sensor tegangan yang umum digunakan dalam sistem IoT bekerja berdasarkan prinsip pembagi tegangan (*voltage divider*) menggunakan rangkaian resistor. (Achlison *et al.*, 2023) menjelaskan bahwa modul ini berfungsi untuk menurunkan tegangan input yang tinggi dari sumber daya (seperti baterai atau panel surya) menjadi level tegangan rendah yang aman untuk dibaca oleh pin analog mikrokontroler. Mengingat ESP32 beroperasi pada logika tegangan maksimal 3,3 Volt, penggunaan sensor ini menjadi mutlak diperlukan untuk mencegah kerusakan permanen pada pin GPIO akibat kelebihan tegangan (*over-voltage*).

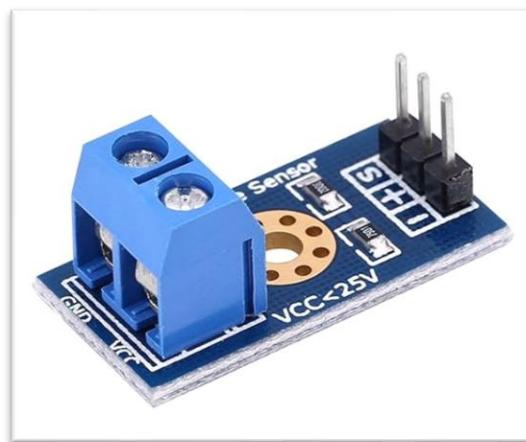
Secara teknis, (Achlison *et al.*, 2023) memaparkan bahwa modul sensor tegangan standar memiliki rasio pembagian 5:1, yang artinya mampu mendeteksi tegangan input hingga 25 Volt jika menggunakan referensi 5 Volt, atau sekitar 16,5 Volt jika digunakan pada sistem 3,3 Volt seperti ESP32. Sinyal keluaran analog dari sensor ini kemudian dibaca oleh fitur ADC pada mikrokontroler dan dikonversi kembali melalui rumus pemrograman untuk mendapatkan nilai tegangan

asli yang akurat. Tingkat akurasi pembacaan sensor ini sangat bergantung pada presisi resistor yang digunakan serta kestabilan tegangan referensi pada mikrokontroler.

Prinsip kerja dari *Voltage Sensor* didasarkan pada rumus pembagi tegangan:

$$V_{out} = V_{in} \times \frac{R2}{R1 + R2} \dots\dots\dots (2.1)$$

Dimana V_{out} adalah tegangan yang masuk ke ESP32, dan V_{in} adalah tegangan sumber yang diukur. Nilai digital yang terbaca oleh ADC ESP32 kemudian dikonversi kembali menjadi nilai tegangan asli melalui pemrograman.



Gambar 2. 12 Voltage Sensor

2. Kajian Pustaka

Penelitian (Afif & Martin, 2022) dalam jurnal berjudul "*Tinjauan Potensi dan Kebijakan Energi Surya di Indonesia*" menegaskan bahwa Sistem Pembangkit Listrik Tenaga Surya (PLTS) memegang peranan vital dalam transformasi energi bersih di Indonesia, baik untuk daerah terpencil maupun penunjang sistem yang ada. Namun, penelitian tersebut menekankan bahwa tantangan utama implementasi PLTS adalah menjaga efisiensi konversi energi agar tetap optimal dalam jangka panjang di tengah faktor lingkungan yang dinamis.

Menanggapi masalah efisiensi tersebut, penelitian (Meilany *et al.*, 2024) dengan judul “*Analisis Output Daya pada Sistem Pembersih Debu Berbasis ESP32 terhadap Panel Surya*”. Penelitian ini berhasil merancang sistem pembersih otomatis menggunakan mikrokontroler ESP32 untuk mengatasi penumpukan debu yang menurunkan kinerja panel. Hasil analisis mereka menunjukkan adanya peningkatan output daya yang signifikan setelah proses pembersihan. Namun, fokus utama penelitian adalah pada analisis dampak daya dan mekanisme pembersihan dasar. Sistem tersebut belum menerapkan algoritma cerdas yang kompleks untuk memutuskan kapan waktu terbaik melakukan pembersihan berdasarkan multivariabel kondisi lingkungan.

Terkait mekanisme pembersihan fisik, penelitian (Sari *et al.*, 2022) dengan judul “*Sistem Kendali Pembersih Panel Surya Menggunakan Rolling Brush dan Wiper Berbasis Arduino*”. Penelitian ini merancang mekanisme pembersihan fisik menggunakan sikat gulung dan *wiper* untuk mengatasi debu pada panel surya. Meskipun alat ini mampu membersihkan permukaan panel secara efektif, sistem kendalinya masih bekerja secara sederhana berbasis Arduino tanpa adanya algoritma cerdas untuk menentukan kapan waktu yang tepat untuk pembersihan. Operasional alat umumnya bergantung pada jadwal waktu (*timer*) atau sakelar manual. Perbedaan dengan penelitian yang sedang dilakukan saat ini adalah penerapan algoritma *Fuzzy Logic* yang tidak lagi bergantung pada *timer*, melainkan pada data *real-time* dari sensor (intensitas cahaya, suhu, dan tegangan) untuk memutuskan tindakan pembersihan secara otomatis hanya ketika kondisi lingkungan memerlukannya.

Penelitian (Cahyanti *et al.*, 2025) dalam jurnal berjudul “*Rancang Bangun Robot Pembersih Panel Surya Menggunakan Metode Dry Cleaning Berbasis Fuzzy Logic Controller*”. Penelitian ini berhasil menerapkan logika *fuzzy* pada robot pembersih debu kering (*dry cleaning*). Persamaan dengan penelitian ini adalah penggunaan kendali *Fuzzy Logic* sebagai otak pengambil keputusan. Namun, perbedaannya terletak pada mekanisme dan

integrasi sistem. Penelitian tersebut fokus pada metode *dry cleaning* (tanpa air), sedangkan penelitian ini menggunakan mekanisme *wet cleaning* (dengan pompa air) untuk hasil yang lebih maksimal pada kotoran membandel. Selain itu, penelitian ini melengkapi sistem dengan fitur monitoring performa panel secara komprehensif berbasis IoT yang terintegrasi dengan sensor cahaya, tegangan, dan suhu, yang belum menjadi fokus utama pada penelitian robot *dry cleaning* tersebut.

Penelitian yang menjadi dasar pengembangan prototipe dirancang oleh dosen Julian Sahertian, S.Pd., M.T. bersama tim mahasiswa M. Rizal Umami, Ilham Khefi Ramadhanu, dan M. Choirul Anwar, (2025). dengan judul “*Deteksi dan Monitoring Kebersihan Panel Surya Menggunakan YOLOv8 Berbasis IoT untuk Optimasi Efisiensi*”. Penelitian ini berhasil mengembangkan sistem deteksi visual menggunakan AI (YOLOv8) untuk memantau jenis kotoran pada panel. Kekurangan utama dari prototipe ini adalah mekanisme eksekusinya yang masih manual, pengguna harus menekan tombol *cleaning* secara manual pada aplikasi *desktop* untuk memulai pembersihan. Penelitian saat ini bertujuan menyempurnakan prototipe tersebut dengan mengganti pemicu manual menjadi otomatis penuh berdasarkan keputusan *Fuzzy Logic*.

Arsitektur komunikasi IoT pada penelitian (Padmaja *et al.*, 2021) yang berjudul “*Google Firebase based Modern IoT System Architecture*”. Penelitian ini membahas arsitektur sistem IoT modern berbasis *Firebase* sebagai *real-time database*. Meskipun *Firebase* sangat memadai untuk penyimpanan dan pelaporan data, penelitian tersebut juga menyoroti karakteristiknya yang lebih condong sebagai basis data ketimbang protokol kendali *real-time* dua arah yang responsif untuk perangkat keras (aktuator). Berdasarkan analisis tersebut, penelitian yang sedang dilakukan ini melakukan pembaruan protokol komunikasi dari *Firebase* menjadi MQTT (*Message Queuing Telemetry Transport*). Protokol MQTT dipilih karena kemampuannya mendukung komunikasi dua arah (*publish-subscribe*) dengan latensi yang jauh lebih rendah, sehingga sangat tepat untuk

mendukung sistem kendali otomatis pada *Smart Solar Panel* yang membutuhkan respons cepat antara sensor, kontroler, dan aktuator.

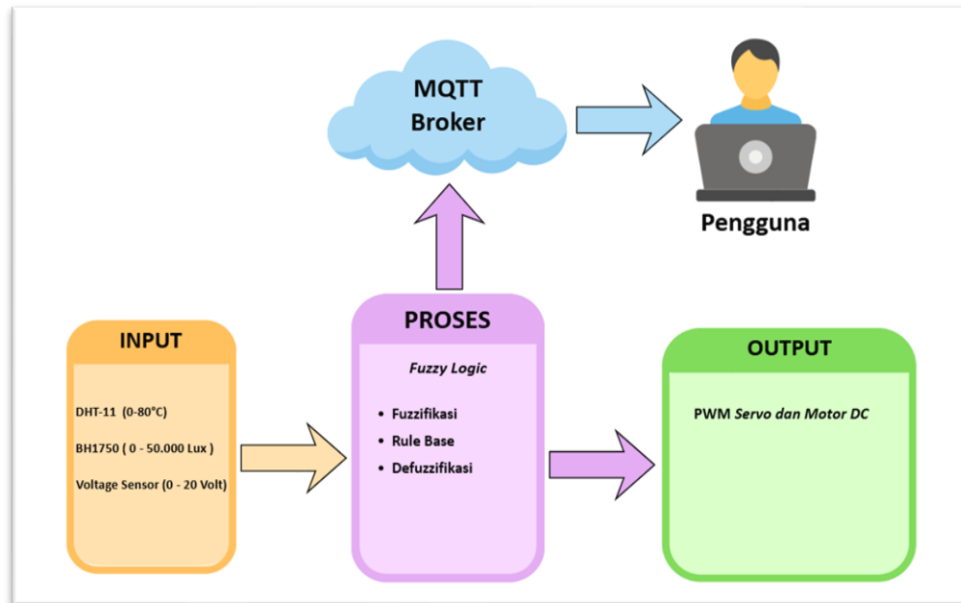
B. Kerangka Berpikir

Penelitian ini dibangun berdasarkan permasalahan penurunan efisiensi panel surya akibat akumulasi debu dan kotoran yang menghambat penyerapan cahaya matahari. Meskipun prototipe *Smart Solar Panel* sebelumnya yang dikembangkan oleh dosen Julian Sahertian, S.Pd., M.T. bersama tim mahasiswa M. Rizal Umami, Ilham Khefi Ramadhanu, dan M. Choirul Anwar, (2025). telah berhasil mendeteksi jenis kotoran menggunakan AI (YOLOv8), sistem tersebut masih memiliki kelemahan pada mekanisme eksekusi pembersihan yang bersifat semi-otomatis. Pengguna masih harus memantau data dan menekan tombol *cleaning* secara manual pada aplikasi *desktop* berbasis *Firebase* yang memiliki latensi komunikasi tinggi untuk kendali *real-time*.

Untuk mengatasi keterbatasan tersebut, fokus penelitian ini mengusulkan pengembangan sistem kendali otomatis penuh (*fully automatic*) dengan menerapkan algoritma *Fuzzy Logic* pada mikrokontroler ESP32. Sistem akan bekerja dengan mengolah data masukan dari tiga sensor utama: sensor BH-1750 (intensitas cahaya), DHT-11 (suhu), dan *Voltage Sensor* (tegangan). Ketiga variabel ini akan diproses melalui tahapan *fuzzifikasi*, evaluasi *rule base*, dan *defuzzifikasi* untuk menghasilkan keputusan cerdas apakah panel surya dalam kondisi "Bersih", "Sedang", atau "Kotor".

Hasil keputusan *Fuzzy Logic* menunjukkan kondisi yang memerlukan pembersihan, mikrokontroler akan secara otomatis mengaktifkan aktuator berupa motor pembersih (*wet cleaning*). Seluruh proses pemantauan data sensor dan status pembersihan akan ditransmisikan menggunakan protokol komunikasi MQTT (*Message Queuing Telemetry Transport*). Protokol ini dipilih untuk menggantikan *Firebase* karena kemampuannya dalam mendukung komunikasi dua arah (*publish-subscribe*) dengan latensi rendah, memastikan sinkronisasi data antara perangkat keras dan antarmuka pengguna terjadi secara *real-time* dan responsif.

Dengan integrasi metode cerdas dan protokol komunikasi yang handal ini, diharapkan efisiensi panel surya dapat terjaga secara optimal tanpa memerlukan intervensi manual dari pengguna.



Gambar 2. 13 Kerangka Berpikir Penelitian

12

BAB III

METODE PENELITIAN

A. Desain Penelitian

1. Jenis Penelitian

Jenis penelitian yang digunakan adalah *Research and Development* (R&D). Merujuk pada penelitian (Sugiyono, 2022) metode ini diterapkan untuk mengembangkan dan menguji efektivitas suatu produk yang sudah ada agar lebih berdaya guna. Dalam konteks penelitian ini, objek pengembangan adalah prototipe *Smart Solar Panel*, namun memerlukan penyempurnaan pada sistem kendalinya.

Fokus pengembangan dilakukan dengan memodifikasi sistem yang semula berjalan semi-manual menjadi otomatis penuh (*fully automatic*). Melalui algoritma *Fuzzy Logic* dan integrasi protokol MQTT pada prototipe *Smart Solar Panel*, penelitian ini bertujuan meningkatkan kecerdasan prototipe dalam melakukan pembersihan mandiri secara *real-time*.

2. Variabel Penelitian

Variabel yang diukur pada penelitian ini adalah intensitas cahaya, suhu, dan tegangan panel surya. Intensitas cahaya didapat dari sensor BH-1750 dengan satuan Lux, suhu didapat dari sensor DHT-11 dengan satuan Derajat Celcius (°C), dan tegangan didapat dari pembacaan *Voltage Sensor* dengan satuan Volt.

Keluaran sistem berupa status kondisi kebersihan panel yang diklasifikasikan menjadi tiga tingkatan (Bersih, Sedang, Kotor) berdasarkan logika *fuzzy*. Keputusan ini kemudian dikonversi menjadi sinyal kendali (High/Low) untuk mengaktifkan motor pembersih, sikat pembersih, dan pompa air.

3. Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan yaitu dengan melakukan eksperimen langsung pada prototipe *Smart Solar Panel* yang telah dikembangkan. Eksperimen dilakukan untuk mendapatkan data

83

56

49

kondisi aktual sensor (cahaya, suhu, dan tegangan) dan menguji ketepatan respon algoritma *Fuzzy Logic* dalam menghasilkan keputusan status panel (Bersih, Sedang, atau Kotor). Selain itu, dilakukan pengambilan data latensi pengiriman informasi untuk memvalidasi kinerja komunikasi protokol MQTT.

B. Instrumen Penelitian

1. Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk membangun sistem kendali otomatis *Smart Solar Panel* dijabarkan pada Tabel 3.1 berikut:

Tabel 3. 1 Spesifikasi Perangkat Keras

No	Nama Komponen	Fungsi	Jumlah
1	ESP32	Sebagai otak suatu alat untuk menyimpan program yang memproses algoritma <i>Fuzzy</i> , membaca sensor, dan mengirim data dengan modul Wi-Fi.	1
2	Sensor BH-1750	Sebagai pengukur tingkat iluminasi matahari (Lux)	1
3	Sensor DHT-11	Sebagai pengukur suhu	1
4	<i>Voltage Sensor</i>	Sebagai modul pembagi tegangan	1
5	Motor Servo	Sebagai aktuator mekanis untuk menggerakkan pembersih (<i>wiper</i>)	2
6	Motor DC	Sebagai aktuator penyemprot air	2
7	Modul Relay	Sakelar elektronik untuk memutus atau menyambung arus	2
8	Stepdown	Sebagai modul untuk menurunkan tegangan	1

2. Perangkat Lunak

Daftar perangkat lunak yang digunakan dalam perancangan kode program, simulasi, dan antarmuka *dashboard* tercantum pada Tabel 3.2 berikut:

Tabel 3. 2 Spesifikasi Perangkat Lunak

No	Nama <i>Software</i>	Fungsi
1	Arduino IDE	Sebagai lingkungan pengembangan untuk menulis, kompilasi, dan <i>upload</i> program ke ESP32.
2	Python & Tkinter	Sebagai bahasa pemrograman dan pustaka grafis untuk membangun aplikasi monitoring <i>desktop</i> .
3	MQTT Broker	Sebagai layanan server untuk perantara pertukaran pesan data IoT.
4	Visual Studio Code	Sebagai editor teks untuk pengembangan <i>script</i> Python pada sisi komputer pengguna.
5	MATLAB	Sebagai <i>software</i> untuk memvalidasi kurva himpunan dan aturan <i>fuzzy</i> sebelum diterapkan.
6	EasyEDA	Sebagai <i>software</i> untuk desain skematik perangkat keras.

3. Dataset

Dataset yang digunakan dalam penelitian ini berupa himpunan aturan logika *fuzzy* (*Rule Base*) yang menghubungkan input sensor cahaya (BH-1750), suhu (DHT-11), dan tegangan (Voltage Sensor) dengan keputusan akhir sistem. Format aturan disusun mengikuti pola "JIKA-MAKA" (*IF-THEN*) seperti yang ditampilkan pada Tabel 3.3 berikut:

Tabel 3. 3 Basis Aturan (*Rule Base*)

No	<i>Rule Base</i>	Keputusan Sistem
1	<i>IF</i> Cahaya = Gelap <i>AND</i> Suhu = Dingin <i>AND</i> Volt = Rendah	BERSIH
2	<i>IF</i> Cahaya = Redup <i>AND</i> Suhu=Normal <i>AND</i> Volt = Sedang	BERSIH
3	<i>IF</i> Cahaya = Terik <i>AND</i> Suhu = Normal <i>AND</i> Volt = Tinggi	BERSIH
4	<i>IF</i> Cahaya = Terik <i>AND</i> Suhu = Normal <i>AND</i> Volt = Rendah	KOTOR
5	<i>IF</i> Cahaya = Terik <i>AND</i> Suhu = Panas <i>AND</i> Volt = Sedang	SEDANG
6	<i>IF</i> Cahaya = Terik <i>AND</i> Suhu = Panas <i>AND</i> Volt = Rendah	KOTOR
7	<i>IF</i> Cahaya = Redup <i>AND</i> Suhu = Normal <i>AND</i> Volt = Rendah	SEDANG
8	<i>IF</i> Cahaya = Gelap <i>AND</i> Suhu = Normal <i>AND</i> Volt = Rendah	BERSIH
9	<i>IF</i> Cahaya = Terik <i>AND</i> Suhu = Panas <i>AND</i> Volt = Tinggi	BERSIH

4. Analisis Hasil

a. Pengujian Fungsional

Tahap ini dilaksanakan untuk memverifikasi kinerja operasional dari setiap komponen sistem untuk memastikan bahwa fungsi pembacaan sensor, pemrosesan logika, dan eksekusi aktuator berjalan tanpa kendala. Daftar komponen yang diuji beserta indikator keberhasilan fungsi masing-masing item disajikan secara rinci pada Tabel 3.4.

Tabel 3. 4 Pengujian Fungsional

No	Komponen yang Diuji	Hasil yang Diharapkan
1	Sensor BH-1750	Nilai Lux berubah linier terhadap intensitas cahaya.
2	Sensor DHT-11	Suhu yang terbaca sesuai dengan kondisi termal ruangan/lingkungan.
3	Voltage Sensor	Mampu membaca nilai tegangan DC yang dihasilkan panel surya.
4	Mekanisme Pembersih	Motor servo berputar menyapu panel dan pompa menyembrotkan air.
5	Konektivitas WiFi	ESP32 berhasil terhubung ke <i>Access Point</i> yang ditentukan.
6	Transmisi MQTT	Data sensor berhasil terkirim (<i>Publish</i>) dan perintah diterima (<i>Subscribe</i>).

b. Pengujian Non-Fungsional

Pengujian non-fungsional pada penelitian ini berfokus pada dua aspek utama, yaitu akurasi keputusan sistem cerdas dan performansi pengiriman data melalui jaringan.

1) Pengujian Akurasi Sistem (*Fuzzy Logic Accuracy*)

Pengujian akurasi bertujuan untuk mengetahui tingkat keberhasilan sistem dalam mengambil keputusan pembersihan secara otomatis. Akurasi dihitung dengan membandingkan output sistem *Fuzzy Logic* terhadap kondisi riil panel surya yang diamati secara manual. Penelitian (Duta, 2025), pengujian akurasi pada sistem berbasis *fuzzy* sangat penting untuk memastikan bahwa respon aktuator sesuai dengan variasi kondisi lingkungan yang dinamis.

Persamaan yang digunakan untuk menghitung akurasi adalah sebagai berikut:

$$Akurasi(\%) = \left(\frac{Jumlah\ Keputusan\ Benar}{Total\ Sampel\ Pengujian} \right) \times 100\% \dots\dots\dots (3.1)$$

2) Pengujian Performansi Jaringan (QoS)

Pengujian ini dilakukan untuk mengevaluasi kinerja protokol MQTT dalam mengirimkan data sensor dari mikrokontroler ESP32 ke aplikasi *monitoring* secara *real-time*. Parameter yang diukur mengacu pada standar *Quality of Service* (QoS) yang ditetapkan oleh TIPHON (*Telecommunications and Internet Protocol Harmonization Over Networks*). Berdasarkan penelitian (Mishra & Kertesz, 2020), analisis QoS pada sistem monitoring berbasis IoT diperlukan untuk menjamin data yang diterima pengguna akurat dan minim gangguan. Parameter utama yang diukur meliputi *Throughput* dan *Latency*.

a) *Throughput*

Throughput didefinisikan sebagai jumlah total data yang berhasil ditransfer dari pengirim ke penerima dalam satuan waktu tertentu. Semakin tinggi nilai *throughput*, semakin baik kinerja jaringan tersebut. Pengukuran *throughput* menjadi indikator utama dalam menentukan seberapa efisien bandwidth jaringan digunakan untuk pengiriman data sensor.

Persamaan untuk menghitung *throughput* adalah:

$$Throughput = \left(\frac{Jumlah\ Data\ Diterima\ (bit)}{Durasi\ Pengamatan\ (s)} \right) \dots\dots\dots (3.2)$$

Kategori kualitas *throughput* diklasifikasikan berdasarkan standar TIPHON seperti yang ditunjukkan pada Tabel 3.5:

Tabel 3. 5 Kategori Indeks *Throughput*

Kategori	Persentase Capaian	Indeks
Sangat Bagus	>2,1 Mbps	4
Bagus	700 s.d. 1200 Kbps	3
Sedang	338 s.d. 700 Kbps	2
Buruk	0 s.d. 338 Kbps	1

b) Latency (Delay)

Latency atau *delay* adalah waktu tunda yang dibutuhkan sebuah paket data untuk merambat dari titik pengirim ke titik penerima. Dalam sistem kendali IoT, *latency* yang rendah sangat diharapkan agar respon aktuator tidak terlambat. Nilai *latency* yang kecil (di bawah 150 ms) menunjukkan kualitas jaringan yang "Sangat Bagus" dan ideal untuk aplikasi *real-time monitoring*.

Persamaan untuk menghitung *latency* adalah:

$$Latency = Waktu Terima Paket - Waktu Terima Paket(3.3)$$

Kategori kualitas *latency* menurut standar TIPHON ditunjukkan pada Tabel 3.6 berikut:

Tabel 3. 6 Kategori Indeks *Latency*

Kategori	Rentang Waktu (ms)	Indeks
Sangat Bagus	< 150 ms	4
Bagus	150 s.d. 300 ms	3
Sedang	300 s.d. 450 ms	2
Buruk	0 s.d. 450 ms	1

D. Tempat dan Jadwal Penelitian

1. Tempat Penelitian

Pelaksanaan penelitian ini, mulai dari tahap perancangan, perakitan perangkat keras, hingga pemrograman sistem, dilaksanakan di Ruang Robot Kota Kediri. Pemilihan lokasi ini didasarkan pada ketersediaan fasilitas yang memadai.

Selain itu, untuk tahap pengujian pengambilan data sensor (intensitas cahaya dan suhu) serta validasi kinerja panel surya, penelitian dilakukan di area terbuka (*outdoor*) pada lokasi Ruang Robot Kota Kediri. Lokasi ini dipilih untuk mendapatkan paparan sinar matahari langsung guna menguji respon sistem kendali otomatis secara *real-time*.

2. Jadwal Penelitian

Penelitian ini direncanakan berlangsung selama 6 bulan, tahapan kegiatan disusun secara sistematis mengikuti alur metode *Research and Development* (R&D) sebagaimana tercantum pada Tabel 3.7, berikut:

Tabel 3. 7 Jadwal Penelitian

No	Kegiatan	Waktu Pelaksanaan																							
		Bulan ke 1				Bulan ke 2				Bulan ke 3				Bulan ke 4				Bulan ke 5				Bulan ke 6			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Identifikasi Masalah																								
2	Pengumpulan Data																								
3	Desain																								
4	Pengembangan																								
5	Implementasi																								
6	Evaluasi																								
7	Penulisan Laporan																								

E. Objek Penelitian

1. Analisis Kebutuhan Sistem

Kebutuhan dalam pengembangan sistem ini adalah kemampuan mengakuisisi data parameter lingkungan, yaitu intensitas cahaya, suhu, dan tegangan panel secara *real-time*. Selain itu, jalur komunikasi data yang stabil untuk menghubungkan perangkat keras dengan aplikasi *monitoring*. Oleh karena itu, diterapkan protokol MQTT sebagai jembatan pertukaran data, sehingga status kebersihan panel dapat dipantau dan mekanisme pembersihan dapat dikendalikan dari jarak jauh oleh pengguna.

2. Subjek Penelitian

Subjek utama yang dikaji dalam penelitian ini adalah (*prototype*) alat pembersih panel surya otomatis. Fokus pengujian performansi terletak pada ketepatan implementasi algoritma *Fuzzy Logic* dalam mengolah tiga variabel masukan (sensor cahaya, suhu, dan tegangan) menjadi keputusan gerak. Evaluasi juga dilakukan terhadap kinerja pengiriman data, khususnya pada aspek kecepatan transfer (*throughput*) dan waktu tunda (*latency*) untuk memastikan keberhasilan sistem komunikasi IoT yang dibangun.

F. Prosedur Penelitian

Prosedur penelitian ini dilaksanakan secara bertahap mengikuti alur model pengembangan sistematis seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3. 1 Konsep Prosedur Penelitian

1. Identifikasi Masalah

Identifikasi masalah dilaksanakan melalui analisis mendalam terhadap kendala utama yang menyebabkan penurunan kinerja panel surya, yaitu faktor debu dan kotoran. Langkah ini diikuti dengan perumusan solusi teknologi yang relevan, yaitu penerapan sistem kendali otomatis guna menggantikan metode pembersihan manual.

2. Pengumpulan Data

Proses pengumpulan data dilakukan dengan mengkaji berbagai literatur dan penelitian terdahulu (periode 2020-2025) yang berfokus pada penerapan IoT dan logika *fuzzy*. Selain itu, observasi sebelumnya secara mandiri guna mempelajari karakteristik sensor BH-1750, sensor DHT-11, *Voltage Sensor*, dan protokol komunikasi MQTT yang akan digunakan pada prototipe *Smart Solar Panel*.

3. Desain

Desain perancangan arsitektur sistem baik dari sisi mekanik maupun elektronik. Kegiatan ini meliputi pembuatan skema rangkaian elektronik untuk ESP32, desain mekanis lengan pembersih pada panel, serta perancangan antarmuka (*User Interface*) aplikasi *dashboard* berbasis Python yang disesuaikan dengan kebutuhan pemantauan pengguna.

4. Pengembangan

Tahap pengembangan dari proses realisasi teknis baik rancangan yang telah dibuat sebelumnya. Pada fase ini, seluruh komponen perangkat keras dirakit menjadi satu kesatuan alat, dan kode program (algoritma *fuzzy* dengan protokol MQTT) diprogram serta diunggah ke dalam mikrokontroler (ESP32).

5. Implementasi

Implementasi dari proses penerapan prototipe yang telah selesai dibangun ke objek penelitian yang sesungguhnya. Alat dipasang pada panel surya di lingkungan terbuka untuk melihat respon sistem terhadap kondisi panas, mendung, maupun berdebu secara *real-time* sesuai skenario penggunaan di lapangan.

6. Evaluasi

Tahap evaluasi berfokus pada pengujian validasi sistem untuk meminimalisir kegagalan fungsi. Evaluasi tidak hanya menggunakan metode *blackbox* untuk fungsionalitas, tetapi juga mencakup pengukuran performansi jaringan (QoS) dan akurasi keputusan logika *fuzzy* guna memastikan alat bekerja presisi sesuai standar yang diharapkan.

7. Penulisan Laporan

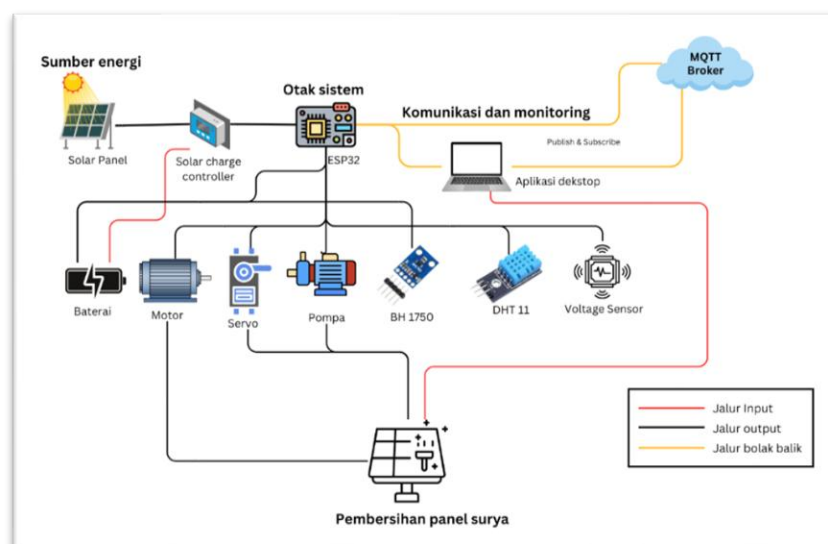
Penulisan laporan dilakukan sebagai tahap akhir setelah seluruh data pengujian terkumpul dan tujuan penelitian tercapai. Seluruh rangkaian proses, mulai dari perancangan hingga analisis hasil, didokumentasikan secara sistematis menjadi sebuah karya tulis ilmiah.

G. Teknik Pengumpulan Data

1. Desain Sistem

a. Perangkat Keras

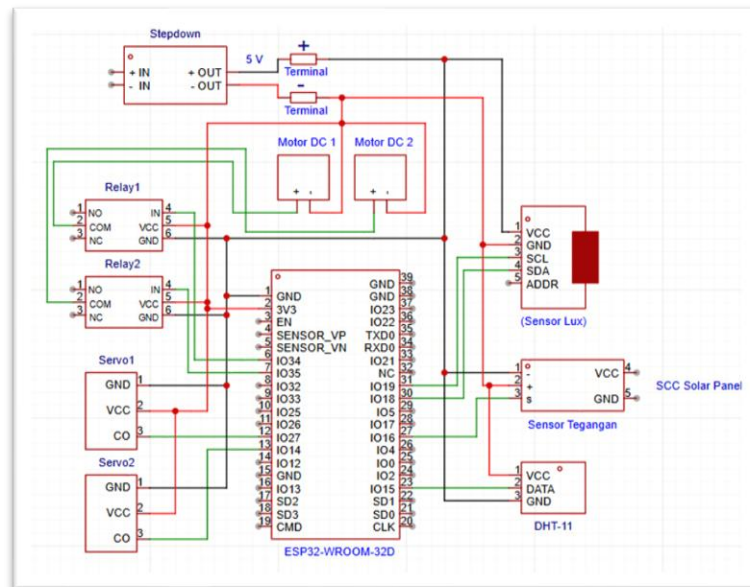
Perancangan perangkat keras berpusat pada mikrokontroler ESP32 yang berfungsi sebagai kendali utama. Ditunjukkan pada Gambar 3.2, sistem mengintegrasikan blok input berupa sensor (BH-1750, DHT-11, dan Sensor Tegangan) untuk mengambil data, serta blok output (Motor Servo dan Pompa Air) untuk mekanisme pembersihan.



Gambar 3. 2 Arsitektur Perangkat Keras

36

Perancangan elektronik divisualisasikan melalui diagram skematik pada Gambar 3.3 menggunakan *software* EasyEDA. Skema ini memetakan jalur koneksi dengan ESP32-WROOM-32D sebagai pusat pemrosesan, sedangkan rincian alokasi pin GPIO untuk sensor dan aktuator dijabarkan pada Tabel 3.8.

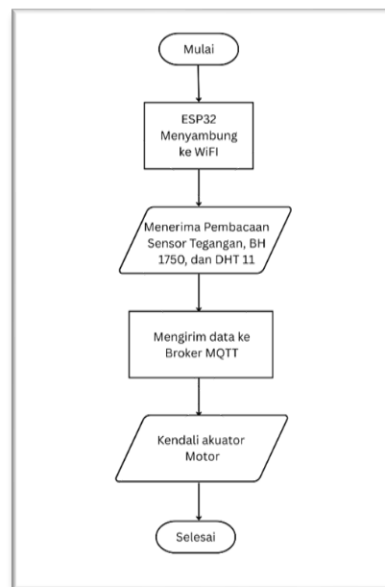


Gambar 3. 3 Skematik Rangkaian Perangkat Keras

Tabel 3. 8 Konfigurasi Pin

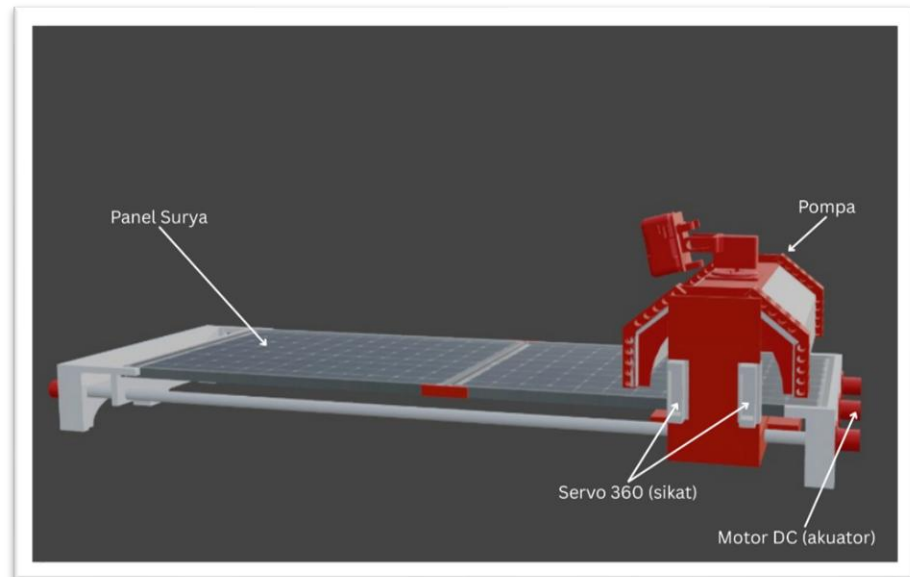
No	Nama Komponen	Pin ESP32	Fungsi
1	Sensor DHT-11	GPIO 15	Input Data Suhu & Kelembaban
2	Sensor Tegangan	GPIO 34	Input Analog (ADC) Pembacaan Voltase
3	Sensor BH-1750	GPIO 22	Jalur Clock I2C (Serial Clock)
		GPIO 21	Jalur Data I2C (Serial Data)
4	Servo 1	GPIO 14	Output PWM Kendali Wiper 1
5	Servo 2	GPIO 12	Output PWM Kendali Wiper 2
6	Relay 1	GPIO 26	Output Digital Motor DC 1
7	Relay 2	GPIO 27	Output Digital Motor 2

Alur kerja perangkat keras dirancang berjalan secara linier dan sistematis sebagaimana divisualisasikan pada Gambar 3.4. Proses operasional dimulai dengan tahap inisialisasi koneksi jaringan Wi-Fi untuk memastikan sistem terhubung ke internet. Setelah koneksi stabil, mikrokontroler melakukan akuisisi data sensor secara *real-time* guna mendapatkan nilai parameter lingkungan terkini. Data tersebut selanjutnya ditransmisikan ke *Broker* MQTT sebagai laporan pemantauan, dan siklus diakhiri dengan eksekusi gerak mekanik pada aktuator motor yang merespons keputusan logika sistem.



Gambar 3. 4 Diagram Alir Perangkat Keras

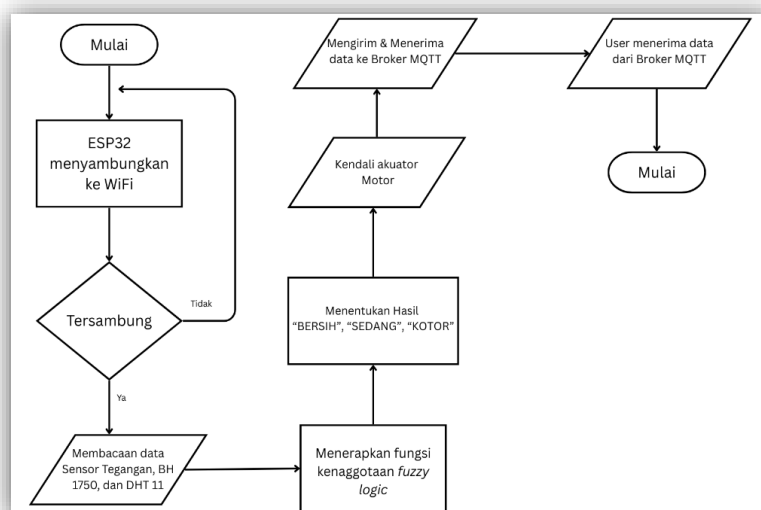
Rancangan mekanik yang divisualisasikan dalam bentuk pemodelan 3D. Desain ini bertujuan untuk memberikan gambaran presisi mengenai integrasi komponen fisik sebelum proses perakitan dilakukan. Sebagaimana ditunjukkan pada Gambar 3.5, desain 3D menampilkan struktur rangka utama yang menopang dua buah panel surya yang dipasang secara sejajar.



Gambar 3. 5 Desain Perangkat Keras

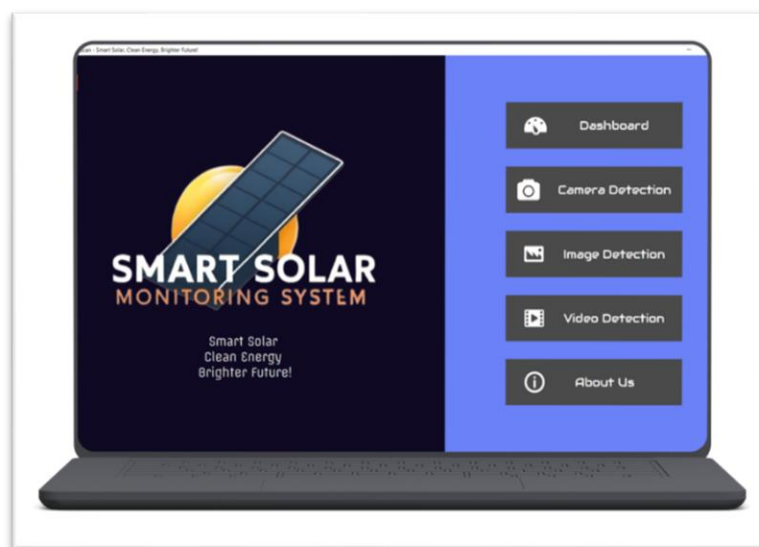
b. Perangkat Lunak

Perangkat lunak dirancang untuk memproses logika dan komunikasi data. Pada Gambar 3.6, alur program diawali dengan koneksi jaringan dan data sensor. Data tersebut kemudian diolah melalui tahapan *Fuzzy Logic* (Fuzzifikasi, Inferensi, Defuzzifikasi) untuk menghasilkan keputusan ("BERSIH", "SEDANG", atau "KOTOR").



Gambar 3. 6 Diagram Alir Perangkat Lunak

Hasil keputusan logika tersebut memiliki dua cabang eksekusi: pertama, menginstruksikan motor untuk bergerak sesuai kondisi; dan kedua, mengirimkan (*publish*) data monitoring ke aplikasi pengguna melalui protokol MQTT. Data yang diterima kemudian ditampilkan pada aplikasi *desktop* seperti rancangan pada Gambar 3.7, yang memungkinkan pengguna memantau status kebersihan dan nilai sensor secara *real-time*.



Gambar 3. 7 Rancang Tampilan *Desktop*

2. Proses Pengolahan Data

Proses pengolahan data menggunakan metode *Fuzzy Logic* diawali dengan pembentukan himpunan fuzzy. Variabel input terdiri dari Suhu, Cahaya, dan Tegangan, sedangkan variabel output adalah Kecepatan Motor. Berikut adalah rincian himpunan keanggotaan untuk setiap variabel.

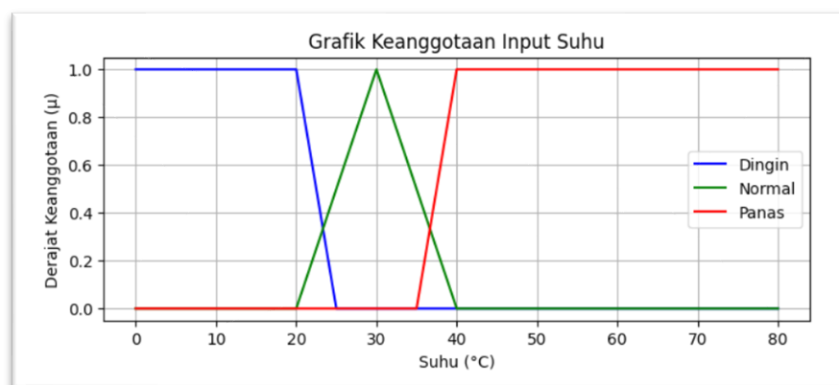
a. Himpunan Keanggotaan Input Suhu

Variabel suhu memiliki semesta pembicaraan pada rentang 0°C hingga 80°C. Variabel ini dibagi menjadi tiga himpunan linguistik: Dingin, Normal, dan Panas.

Tabel 3. 9 Himpunan Keanggotaan Input Suhu

Variabel Linguistik	Semesta Pembicara (Unit)	Rentang Domain (°C)
Dingin	[0 - 80]	0 – 25
Normal		20 – 40
Panas		35 – 80

Representasi visual grafik fungsi keanggotaan suhu ditunjukkan pada gambar 3.8, berikut.



Gambar 3. 8 Grafik Keanggotaan Suhu

Persamaan matematis untuk himpunan suhu adalah:

$$\begin{aligned}
 \mu_{DINGIN}(x) & \begin{cases} 1; & x \leq 20 \\ \frac{25 - x}{5}; & 20 < x < 25 \\ 0; & x \geq 25 \end{cases} \\
 \mu_{NORMAL}(x) & \begin{cases} 0; & x \leq 20 \vee x \geq 40 \\ \frac{x - 20}{10}; & 20 < x < 30 \\ \frac{40 - x}{10}; & 30 \leq x < 40 \end{cases} \\
 \mu_{PANAS}(x) & \begin{cases} 0; & x \leq 35 \\ \frac{x - 35}{5}; & 35 < x < 40 \\ 1; & x \geq 40 \end{cases} \dots\dots\dots (3.4)
 \end{aligned}$$

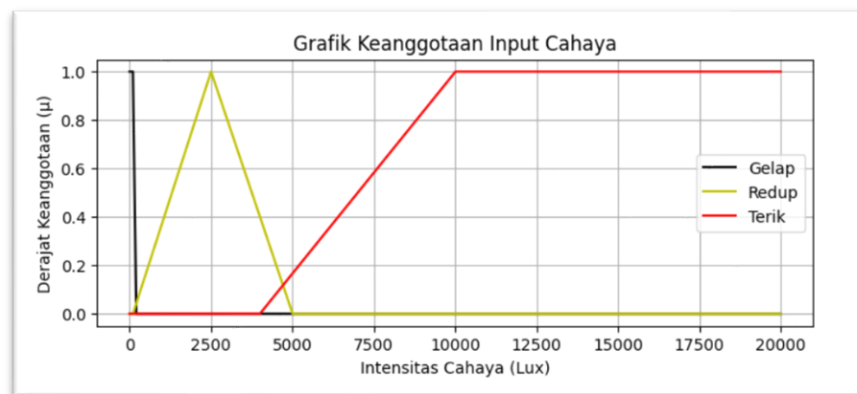
b. Himpunan Keanggotaan Input Cahaya

Variabel intensitas cahaya memiliki semesta pembicaraan 0 hingga 20.000 Lux. Variabel ini dibagi menjadi tiga himpunan: Gelap, Redup, dan Terik.

Tabel 3. 10 Himpunan Keanggotaan Input Cahaya

Variabel Linguistik	Semesta Pembicara (Unit)	Rentang Domain (Lux)
Gelap	[0 – 20.000]	0 – 200
Redup		100 – 5.000
Terik		4000 – 20.000

Representasi visual grafik fungsi keanggotaan cahaya ditunjukkan pada gambar 3.9.



Gambar 3. 9 Grafik Keanggotaan Cahaya

Persamaan matematis untuk himpunan cahaya adalah:

$$\mu_{GELAP}(x) \begin{cases} 1; & x \leq 100 \\ \frac{200 - x}{100}; & 100 < x < 200 \\ 0; & x \geq 200 \end{cases}$$

$$\mu_{REDUP}(x) \begin{cases} 0; & x \leq 100 \vee x \geq 5000 \\ \frac{x - 100}{2400}; & 100 < x < 2500 \\ \frac{5000 - x}{2500}; & 2500 \leq x < 5000 \end{cases}$$

$$\mu_{TERIK}(x) \begin{cases} 0; & x \leq 4000 \\ \frac{x - 4000}{6000}; & 4000 < x < 10000 \\ 1; & x \geq 10000 \end{cases} \dots\dots\dots (3.5)$$

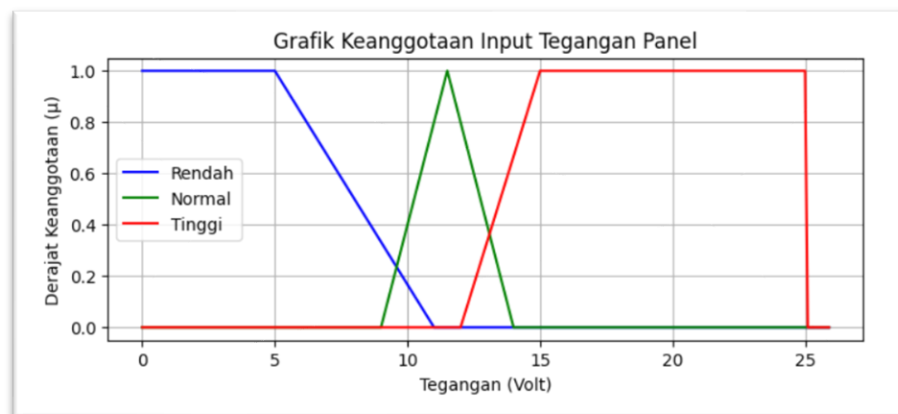
c. Himpunan Keanggotaan Input Tegangan

Variabel tegangan panel memiliki semesta pembicaraan 0 hingga 25 Volt. Variabel ini dibagi menjadi tiga himpunan: Rendah, Normal, dan Tinggi.

Tabel 3. 11 Himpunan Keanggotaan Input Tegangan

Variabel Linguistik	Semesta Pembicara (Unit)	Rentang Domain (Volt)
Rendah	[0 - 25]	0 – 11
Normal		9 – 14
Tinggi		12 – 25

Representasi visual grafik fungsi keanggotaan tegangan ditunjukkan pada gambar 3.10.



Gambar 3. 10 Grafik Keanggotaan Tegangan

Persamaan matematis untuk himpunan tegangan adalah:

$$\mu_{RENDAH}(x) \begin{cases} 1; & x \leq 5 \\ \frac{11 - x}{6}; & 5 < x < 11 \\ 0; & x \geq 11 \end{cases}$$

$$\mu_{NORMAL}(x) = \begin{cases} 0; & x \leq 9 \vee x \geq 14 \\ \frac{x - 9}{2,5}; & 9 < x < 11,5 \\ \frac{14 - x}{2,5}; & 11,5 \leq x < 14 \end{cases}$$

$$\mu_{TINGGI}(x) = \begin{cases} 0; & x \leq 12 \\ \frac{x - 12}{3}; & 12 < x < 15 \\ 1; & x \geq 15 \end{cases} \dots\dots\dots (3.6)$$

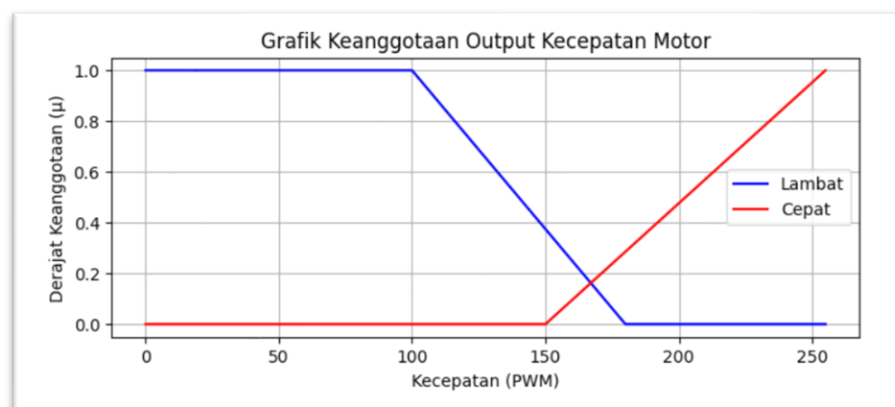
d. Himpunan Keanggotaan Output Motor

Variabel output kecepatan motor menggunakan nilai PWM dengan semesta pembicaraan 0 hingga 255. Variabel ini dibagi menjadi dua himpunan: Lambat dan Cepat.

Tabel 3. 12 Himpunan Keanggotaan Output Motor

Variabel Linguistik	Semesta Pembicara (Unit)	Rentang Domain (PWM)
Lambat	[0 - 255]	0 – 180
Cepat		150 – 255

Representasi visual grafik fungsi keanggotaan motor ditunjukkan pada gambar 3.11.



Gambar 3. 11 Grafik Keanggotaan Motor

Persamaan matematis untuk himpunan motor adalah:

$$\mu_{LAMBAT}(x) \begin{cases} 1; & x \leq 100 \\ \frac{180 - x}{80}; & 100 < x < 180 \\ 0; & x \geq 180 \end{cases}$$

$$\mu_{CEPAT}(x) \begin{cases} 0; & x \leq 12 \\ \frac{x - 150}{105}; & 12 < x < 15 \\ 1; & x \geq 180 \end{cases} \dots\dots\dots (3.7)$$

3. Simulasi Penyelesain Masalah

a. Data Input

Pada pukul 12.00 siang, matahari bersinar terik. Namun, sistem mendeteksi tegangan yang dihasilkan panel surya tidak optimal (drop) dibandingkan intensitas cahaya yang diterima. Hal ini mengindikasikan adanya debu/kotoran yang menutupi permukaan panel.

- 1) Sensor Suhu (DHT11): 36° (Kondisi: Panas)
- 2) Sensor Cahaya (BH1750): 18.000 Lux (Kondisi: Sangat Terang)
- 3) Sensor Tegangan (Voltage): 11,2 Volt (Kondisi: Drop/ Rendah)

b. Pengolahan Data

1) Tahap Fuzzifikasi

Sistem mengubah data angka asli menjadi Derajat Keanggotaan (μ) bernilai 0 s.d 1.

a) Sensor Suhu (DHT-11)

Himpunan “PANAS” (35° C – 80° C) dengan penyelesaian:

$$\mu_{Panas} [36] = \frac{55 - 35}{80 - 35} = \frac{20}{45} = 0,44$$

b) Sensor Cahaya (BH 1750)

Himpunan “TERIK” (5.000 – 20.000 Lux) dengan penyelesaian:

$$\begin{aligned} \mu_{Terang} [18.000] &= \frac{18.0000 - 10.000}{20.000 - 10.000} = \frac{8.000}{10.000} \\ &= 0,8 \end{aligned}$$

c) *Voltage Sensor*

Himpunan “RENDAH” (10 – 12 Volt) dengan penyelesaian:

$$\mu_{Rendah} [12,2] = \frac{12 - 11,2}{12 - 10} = \frac{0,8}{2} = 0,4$$

2) Tahap Inferensi

Rule: IF Suhu PANAS AND Cahaya TERANG AND Tegangan RENDAH THEN Motor CEPAT.

Dari hasil $\mu_{Panas} = 0,44$; $\mu_{Terang} = 0,8$; $\mu_{Rendah} = 0,4$. Maka untuk menentukan *rule* nilai MIN (Terkecil) dari ketiga input:

$$\alpha_{predikat} = MIN(0,44; 0,8; 0,4)$$

$$\text{Nilai Kekuatan Aturan} = 0,4$$

3) Tahap Defuzzifikasi

Sistem menghitung kecepatan motor (PWM) menggunakan metode Centroid (Center of Gravity) dengan menghitung Rumus Garis Miring.

$$\mu(z) = \frac{z - 150}{255 - 150} = \frac{z - 150}{105}$$

Karena kekuatan aturannya 0,4, langkah berikutnya mencari di angka PWM garis miring tersebut mencapai ketinggian 0,4

$$0,4 = \frac{z_1 - 150}{105}$$

$$0,4 \times 105 = z_1 - 150$$

$$42 = z_1 - 150$$

$$z_1 = 150 + 42$$

$$z_1 = 192$$

Artinya:

Dari PWM 150 s.d 192, bentuk grafiknya adalah Segitiga (naik). Sedangkan dari PWM 192 s.d 255, bentuk grafiknya adalah Persegi Panjang (rata di tinggi 0,4), kemudian menghitung titik berat dengan langkah;

62

Rumus Luas Segitiga:

$$\text{Luas } (L_1) = \frac{1}{2} \times \text{Alas} \times \text{Tinggi}$$

$$L_1 = \frac{1}{2} \times 0,5 \times 0,4 = 8,4$$

Titik Pusat C_1 Untuk segitiga siku-siku, titik beratnya ada di $\frac{2}{3}$ dari alas (dihitung dari titik awal).

$$C_1 = \left(\frac{2}{3} \times 42\right) = 150 + 28 = 178$$

Momen (M_1) = Luas x Titik Pusat

$$M_1 = 8,4 \times 178 = 1.492,2$$

Rumus Persegi Panjang:

$$\text{Luas } (L_2) = \text{Panjang} \times \text{Tinggi}$$

$$(L_2) = 63 \times 0,4 = 25,2$$

Titik Pusat C_2 untuk persegi panjang

$$C_1 = 192 + \left(\frac{63}{2}\right) = 192 + 31,5 = 223,5$$

Momen (M_2) = Luas x Titik Pusat

$$M_2 = 25,2 \times 223,5 = 5.632,2$$

Rumus Centroid adalah Total Momen dibagi Total Luas.

$$Z^* = \frac{\sum \text{Momen}}{\sum \text{Luas}} = \frac{M_1 + M_2}{L_1 + L_2}$$

$$Z^* = \frac{1.495,2}{8,4} + \frac{5.632,2}{25,2} = \frac{7.127,4}{33,6}$$

$$Z^* \approx 212,12$$

c. Data Output

1) Data Kinerja Sistem (*Fuzzy Output*)

Tabel 3. 13, menunjukkan rekam jejak (*log*) alat saat mendeteksi panel kotor hingga melakukan pembersihan.

Tabel 3. 13 Simulasi *Fuzzy Output*

No	Waktu (WIB)	Suhu (°C)	Cahaya (Lux)	Tegangan (V)	Output PWM	Status Sistem	Keterangan
1	13:00	55.0	18.000	11.2	212	CLEANING	Deteksi Kotor
2	13:00	55.1	18.100	11.2	212	CLEANING	Proses Pembersihan Berjalan
3	13:02	54.8	18.500	13.2	0	STANDBY	Pembersihan Selesai

Pada data nomor 1, terlihat tegangan *drop* ke 11,2 V saat cahaya terang 18.000 Lux. Sistem merespon dengan *output* PWM 212 (Motor Cepat), sesuai dengan perhitungan teori.

2) Data Kualitas Jaringan (*QoS Output*)

Tabel ini menunjukkan performa pengiriman data JSON yang berisi angka “212” tadi ke server MQTT.

Tabel 3. 14 Simulasi *Qos Output*

No	ID Paket	Ukuran Data	Waktu Kirim (ms)	Waktu Terima (ms)	Latency (ms)	Status Pengiriman
1	Pkt_101	128	13:00:00.100	13:00:00.180	80 ms	Berhasil
2	Pkt_102	128	13:00:05.100	13:00:00.195	95 ms	Berhasil
3	Pkt_103	128	13:02:00.100	13:02:00.185	85 ms	Berhasil

1

BAB IV

HASIL DAN PEMBAHASAN

A. Implementasi Lembar Kerja

Implementasi sistem kendali otomatis pada prototipe *Smart Solar Panel* terintegrasi *Internet of Things* (IoT) ini membutuhkan beberapa lembar kerja perangkat lunak yang dirancang untuk dapat saling berkomunikasi secara berkesinambungan. Lembar kerja tersebut dibagi menjadi tiga komponen utama sebagai berikut:

1. Lembar Kerja Aplikasi Desktop Utama

Lembar kerja ini merupakan antarmuka grafis atau *Graphical User Interface* (GUI) berbasis *desktop* yang dikembangkan menggunakan bahasa pemrograman Python dan pustaka Tkinter. Lembar kerja ini berfungsi sebagai pusat visualisasi data pantauan pengguna, yang bertugas menampilkan fluktuasi parameter tegangan panel, intensitas cahaya matahari, serta nilai suhu lingkungan dalam bentuk komponen grafik dinamis. Selain itu, lembar kerja ini menyediakan tombol kendali operasional aktuator apabila sistem dialihkan ke mode pemrosesan manual.

2. Lembar Kerja Perhitungan Fuzzy

Lembar kerja ini memuat blok kode program algoritma *Fuzzy Inference System* (FIS) metode Mamdani yang ditanamkan secara internal pada memori mikrokontroler ESP32. Tahapan proses yang dijalankan meliputi konversi nilai masukan tegas (*crisp input*) sensor menjadi nilai keanggotaan *fuzzy*, proses komparasi aturan logika (JIKA-MAKA) pada komponen *rule base*, hingga kalkulasi *defuzzifikasi* menggunakan metode titik berat (*centroid*) untuk menghasilkan nilai tegas (*crisp output*) berupa besaran sinyal *Pulse Width Modulation* (PWM) penggerak motor.

3. Lembar Kerja Dashboard MQTT

Lembar kerja ini berupa konfigurasi panel manajemen pada server *cloud* MQTT Broker (HiveMQ Cloud) yang difungsikan sebagai gerbang

perantara komunikasi nirkabel jarak jauh. Pada lembar kerja ini, diatur parameter kanal logis (*topic*), alamat URL server host, nomor port jaringan, serta daftar nama pengguna (*username*) dan kata sandi (*password*) enkripsi keamanan data untuk mengontrol aktivitas publikasi (*publish*) dan berlangganan (*subscribe*) data dari perangkat pintar.

B. Keterkaitan Lembar Kerja

Setiap komponen lembar kerja perangkat lunak yang dibangun memiliki ikatan fungsional yang erat dalam membentuk arsitektur jaringan kendali loop tertutup (*closed-loop control system*). Hubungan timbal balik dan mekanisme aliran data antar-lembar kerja tersebut dijabarkan melalui poin operasional berikut:

1. Aliran Masukan Sensor

Mikrokontroler ESP32 secara periodik melakukan akuisisi data fisik dari sensor BH-1750, DHT-11, dan sensor tegangan. Data mentah tersebut diproses langsung oleh Lembar Kerja Perhitungan *Fuzzy* internal untuk menentukan status kondisi kebersihan panel surya.

2. Publikasi Data melalui MQTT

Hasil akhir pengolahan logika berupa status kondisi (BERSIH, SEDANG, atau KOTOR) beserta besaran parameter sinyal PWM motor dikemas ke dalam format struktur teks JSON berukuran 128 Byte. Paket data ini dilempar (*publish*) oleh ESP32 menuju server MQTT Broker melalui topik logis *smartsolar/panel/data*.

3. Visualisasi Antarmuka Pengguna

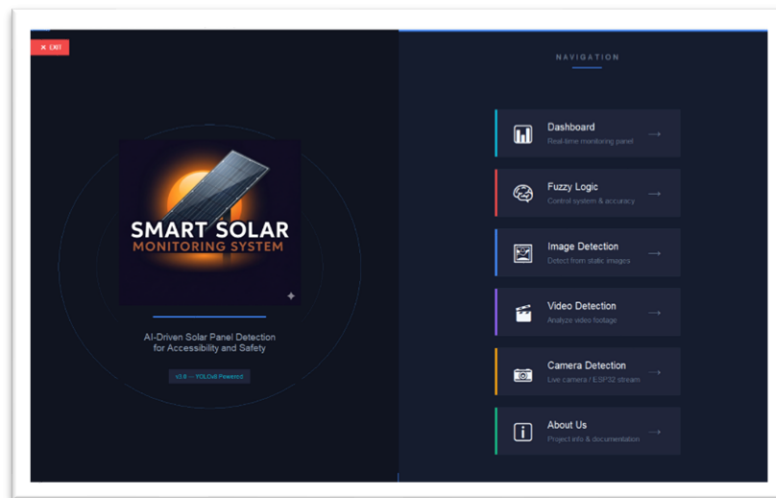
Jendela Aplikasi Desktop Utama yang bertindak selaku *subscriber* menangkap pesan data JSON dari server broker MQTT secara instan. Setelah pesan dibongkar, aplikasi langsung memperbarui grafik fluktuasi parameter pada layar monitor pengguna secara *real-time*.

4. Mekanisme Umpan Balik Manual

Apabila pengguna mengubah mode operasional menjadi manual melalui tombol pada Aplikasi Desktop Utama, sinyal instruksi gerak aktuator akan dipublikasikan balik menuju topik kontrol MQTT Broker.

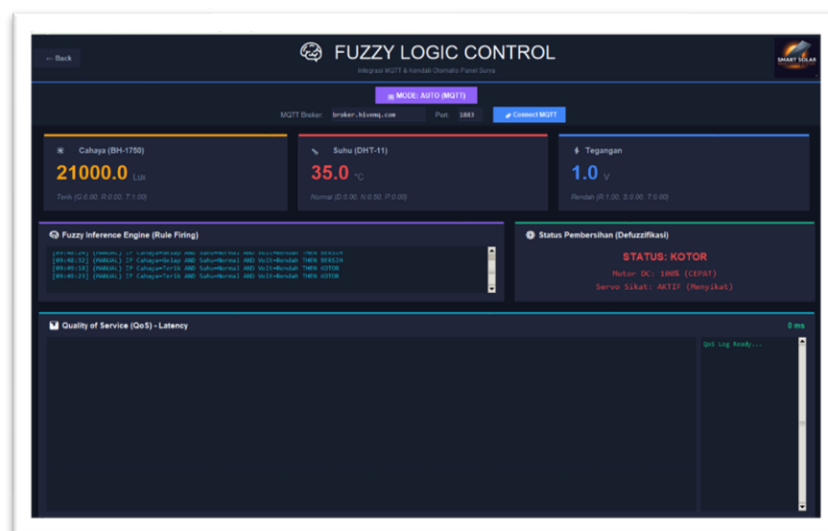
C. Implementasi Desain Sistem

1. Halaman Utama Monitoring Desktop



Gambar 4. 1 GUI Desktop

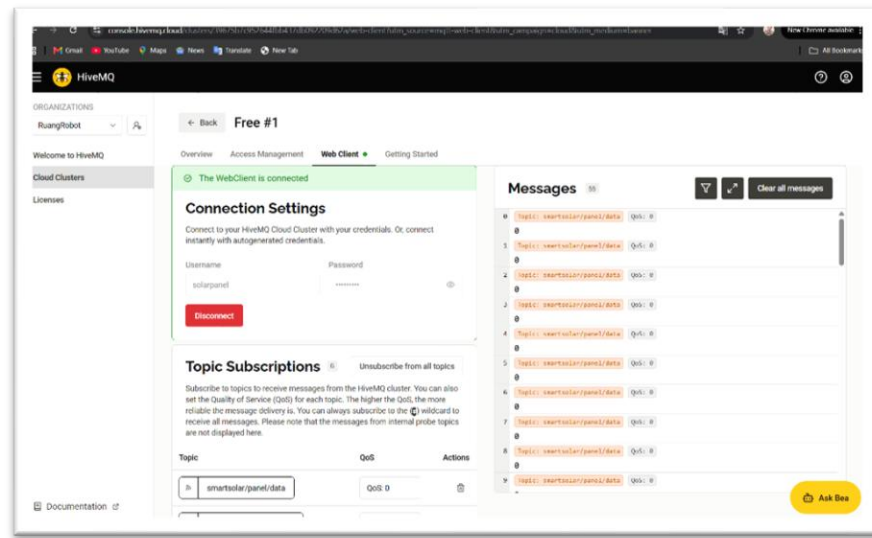
Antarmuka GUI komputer menampilkan desain visual yang terbagi menjadi dua panel utama. Panel sebelah kanan memuat deretan tombol aksi navigasi, antara lain tombol menuju menu *Dashboard* utama, halaman deteksi visual kamera (*Camera Detection*), menu analisis citra foto statis (*Image Detection*), fuzzy logic (*Fuzzy Logic*), pemroses cuplikan rekaman (*Video Detection*), serta menu profil pengembang (*About Us*).



Gambar 4. 2 GUI Desktop Menu *Fuzzy Logic*

Ketika menu *Fuzzy Logic* diakses, layar berganti menampilkan dua diagram grafik garis yang memperlihatkan kurva pergerakan parameter nilai intensitas cahaya dan suhu lingkungan, serta indikator teks status mesin dalam kondisi *STANDBY* atau *CLEANING*.

2. Dashboard HiveMQ Cloud Broker

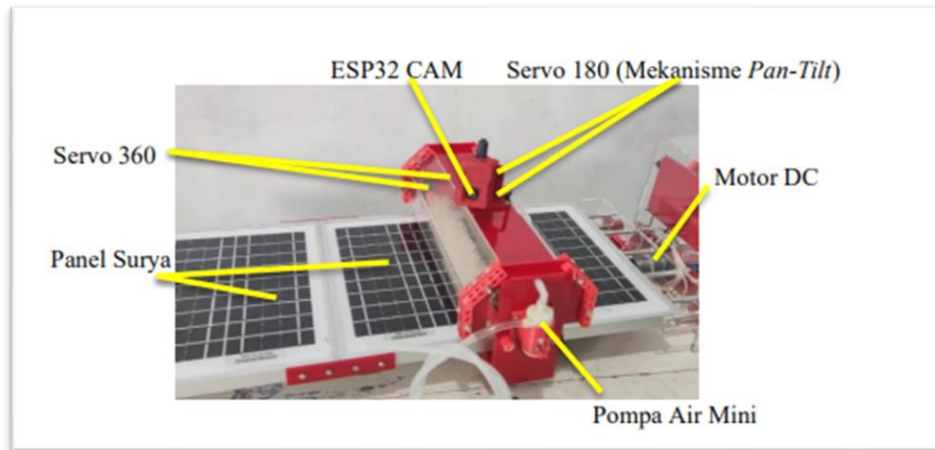


Gambar 4.3 WebClient HiveMQ

Implementasi server perantara memanfaatkan platform HiveMQ Cloud versi non-berbayar (*free cluster*). Panel kontrol web menampilkan status kesehatan klaster nirkabel secara berkala, lengkap dengan informasi alamat URL host, enkripsi port TLS, serta statistik volume paket data yang melintas. Melalui fitur menu bawaan *WebClient*, pengembang dapat melakukan simulasi pelacakan kebenaran isi data teks payload yang dikirimkan oleh ESP32 pada kanal topik monitoring tanpa perlu menyalakan aplikasi komputer utama.

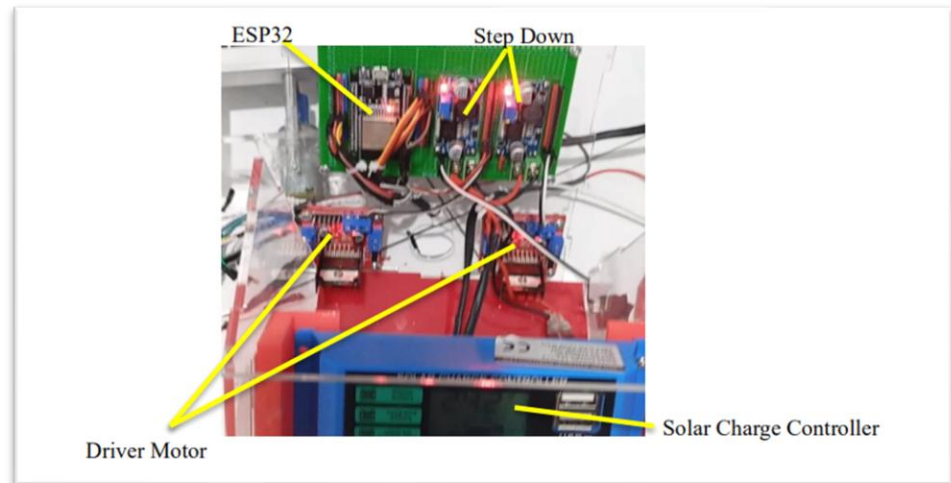
3. Desain Fisik dan Integrasi Perangkat Keras

Prototipe fisik *Smart Solar Panel* dirancang menggunakan struktur rangka berbahan akrilik transparan berdimensi dan juga slider yang sesuai dengan *design 3D Printing*. Komponen utama terdiri dari dua bilah panel surya yang diletakkan sejajar pada bagian atas rangka.



Gambar 4. 4 Spesifikasi Alat 1

- a. ESP32 CAM
Digunakan untuk menangkap gambar maupun video kondisi panel surya.
- b. Panel Surya
Berperan sebagai penghasil listrik alternatif yang didapat dari sinar matahari.
- c. Servo 360
Digunakan sebagai penggerak atau pemutar sikat untuk membersihkan permukaan panel surya.
- d. Servo 180
Digunakan sebagai mekanisme pan-tilt untuk mengatur sudut pandang dan posisi kamera.
- e. Pompa Air Mini
Digunakan untuk mengeluarkan air ketika sikat berputar membersihkan permukaan panel surya
- f. Motor DC
Berperan sebagai aktuator utama dalam alat untuk menggerakkan maju mundur alat untuk membersihkan permukaan panel surya.



Gambar 4. 5 Spesifikasi Alat 2

g. ESP32

Berperan sebagai mikrokontroler dan otak dalam mengatur kinerja seluruh alat meliputi motor DC sebagai aktuator utama, sistem pembersih yaitu servo 360 yang terhubung dengan sikat dan pompa air mini, serta berperan dalam komunikasi firebase untuk kontrol alat melalui aplikasi.

h. Stepdown

Digunakan untuk menurunkan tegangan dari baterai menjadi 5V agar sesuai dengan kebutuhan power supply dari ESP32 dan ESP32 CAM.

i. Driver Motor L298N

Digunakan untuk mengatur kecepatan motor DC sebagai aktuator utama dan mengatur kecepatan pompa air mini.

j. Solar Charge Control

Sebagai regulator atau penghubung serta mengatur aliran listrik antara panel surya dengan baterai.

k. Limit Switch

Berperan sebagai trigger atau pemicu alat jika telah berada diujung panel surya untuk mengatur aktuator utama berhenti maupun berganti arah gerak ketika membersihkan permukaan panel surya.

D. Pengujian Sistem

1. Pengujian Fungsional

Pengujian fungsional dilakukan dengan metode pengujian kotak hitam (*black-box testing*) untuk menguji kelayakan operasional elektrik dari setiap komponen perangkat keras prototipe. Rekapitulasi hasil uji fungsionalitas disajikan pada Tabel 4.1.

Tabel 4. 1 Hasil Pengujian Fungsional Komponen Perangkat Keras

No	Komponen yang Diuji	Hasil yang Diharapkan	Hasil yang Diperoleh	Keterangan
1	Sensor BH-1750	Nilai Lux berubah linier terhadap intensitas cahaya	Berhasil	Valid
2	Sensor DHT-11	Suhu yang terbaca sesuai dengan kondisi lingkungan.	Berhasil	Valid
3	Sensor Tegangan	Mampu membaca nilai tegangan DC dari luaran panel surya.	Berhasil	Valid
4	Mekanisme Pembersih	servo berputar menyapu panel dan pompa mengalirkan air.	Berhasil	Valid
5	Konektivitas WiFi	Modul internal ESP32 terhubung ke jalur internet.	Berhasil	Valid
6	Transmisi MQTT	Perangkat sukses melakukan aksi <i>Publish</i> dan <i>Subscribe</i> topik.	Berhasil	Valid

2. Pengujian Non-Fungsional

a. Pengujian Kestabilan Respon Aturan Kontrol

Pengujian fungsionalitas sistem kontrol bertujuan memverifikasi apakah logika pemrograman yang ditanamkan mampu mengeluarkan keputusan instruksi yang selaras dengan rancangan basis aturan skenario

rule base. Rekapitulasi hasil pengujian kesesuaian tindakan disajikan pada Tabel 4.2.

Tabel 4. 2 Matriks Kesesuaian Eksekusi Keputusan Sistem Kontrol

No Skenario	Input			Hasil Target	Hasil Eksekusi	Tindakan
	Cahaya	Suhu	Tegangan			
1	Gelap	Dingin	Rendah	BERSIH	BERSIH	Benar
2	Redup	Normal	Sedang	BERSIH	BERSIH	Benar
3	Terik	Normal	Tinggi	BERSIH	BERSIH	Benar
4	Terik	Normal	Rendah	KOTOR	KOTOR	Benar
5	Terik	Panas	Sedang	SEDANG	KOTOR	Salah
6	Terik	Panas	Rendah	KOTOR	KOTOR	Benar
7	Redup	Normal	Rendah	SEDANG	BERSIH	Salah
8	Gelap	Normal	Rendah	BERSIH	BERSIH	Benar
9	Terik	Panas	Tinggi	BERSIH	BERSIH	Benar

Berdasarkan data observasi pada Tabel 4.2, terdapat deviasi pada skenario nomor 5 dan 7 yang disebabkan oleh karakteristik tumpang tindih (*overlapping*) grafik fungsi keanggotaan variabel masukan pada program. Untuk menilai kelayakan naskah secara matematis, dilakukan perhitungan nilai performansi kontrol menggunakan parameter indeks pengujian kuantitatif (Sensitivitas, Spesifisitas, dan Akurasi total). Hasil kalkulasi performa kendali tersebut dirangkum pada Tabel 4.3.

Tabel 4. 3 Hasil Evaluasi Parameter Kinerja Sistem Kontrol

No	Parameter Kinerja	Nilai Capaian Pengujian	Hasil
1	Sensitivitas (<i>True Positive Rate</i>)	77,7%	Baik
2	Spesifisitas (<i>True Negative Rate</i>)	66,6%	Cukup
3	Akurasi	77,7%	Baik

b. Pengujian Performansi Pengiriman Data (QoS)

Pengujian aspek non-fungsional jaringan dilakukan guna menilai keandalan protokol komunikasi data MQTT dalam memindahkan muatan informasi dari prototipe menuju aplikasi monitoring. Pengujian *Throughput* dilakukan dengan memvariasikan besaran beban kapasitas data (*payload data*), sedangkan pengujian *Latency* dilakukan pada rentang kelajuan koneksi internet yang berbeda-beda. Rekapitulasi pengujian parameter jaringan disajikan pada Tabel 4.4 dan Tabel 4.5.

Tabel 4. 4 Hasil Pengujian Nilai *Throughput* Jaringan

Payload (Bytes)	2000	1500	1024	750	500	128	32
Perulangan Uji	<i>Throughput (bps)</i>						
1	310.200	290.110	512.000	240.125	187.500	128.000	32.000
2	275.400	255.300	204.800	201.200	187.500	64.000	16.000
3	275.400	255.300	204.800	201.200	187.500	64.000	16.000
4	281.140	261.100	256.000	204.150	166.666	64.000	32.000
5	270.557	254.261	256.000	200.200	166.666	64.000	16.000
Rata-rata	282.539	263.214	287.520	209.375	183.333	76.800	22.400

Tabel 4. 5 Hasil Pengujian Nilai *Latency* Transmisi

Kecepatan (Mbps)	42.28	37.90	21.99	10.85	3.55	1.63	0.48	0.21
Perulangan Uji	<i>Latency (ms)</i>							
1	307.21	289.45	306.13	274.15	294.10	315.42	338.50	352.84
2	228.22	312.18	327.37	301.88	324.61	298.15	315.22	342.48
3	355.62	268.52	247.45	255.42	236.39	332.64	412.65	578.83
4	283.46	295.34	271.15	282.60	243.63	258.90	274.18	260.25
5	297.38	279.11	294.55	269.75	275.39	310.22	299.40	280.85
Rata-rata	294.37	288.92	289.33	276.76	274.82	303.07	327.99	363.05

E. Uji Coba Keseluruhan

1. Pengujian Respon Sensor Fisik

Pengujian respon sensor ditujukan untuk membuktikan kepekaan pembacaan modul sensor BH-1750, DHT-11, dan sensor tegangan terhadap stimulasi kondisi riil di lingkungan luar ruangan (*outdoor*) area Ruang Robot Kota Kediri. Hasil rekam data pengujian sensor pada berbagai skenario kondisi lapangan disajikan pada Tabel 4.6.

Tabel 4. 6 Hasil Log Data Respon Pembacaan Blok Sensor Masukan

No	Waktu	Kondisi Lapangan	Sensor Cahaya (lux)	Sensor Suhu (°C)	Sensor Tegangan (v)	Keterangan
1	02:25:51	Malam	3.3	25.9	1.00	Normal
2	08:15:22	Pagi	18.654.2	35.8	20.24	Normal
3	12:26:48	Siang	20.938.3	35.4	20.32	Normal
4	15:45:06	Siang	919.2	ERR!	1.00	System Error
5	15:45:25	Siang	2100.0	35.0	1.00	Cleaning Active

2. Pengujian Parameter Kecepatan Putaran Motor

Pengujian tingkat kecepatan mekanis motor servo penarik sikat dilakukan dengan memberikan variasi input pulsa nilai PWM dari rentang batas 0 sampai 255. Pengukuran kelajuan putar dihitung menggunakan satuan *Round per Minute* (RPM) untuk melihat pengaruh friksi statis komponen sikat, seperti yang tertera pada Tabel 4.7.

Tabel 4. 7 Parameter Nilai PWM Terhadap Kelajuan Putar Motor

No	Nilai Input (PWM)	Kecepatan (RPM)	Keterangan
1	0	0	Motor Diam Sempurna
2	212	1600	Motor Berputar Kelajuan Kategori Tinggi

3. Pengujian Open Loop (Sistem Kendali Terbuka)

Pengujian sistem kendali loop terbuka (*open loop*) ditujukan untuk memverifikasi keandalan transmisi dua arah saat sistem monitoring diposisikan pada mode manual lewat tombol interaksi aplikasi. Hasil pencatatan unjuk kerja fungsi manual dirangkum pada Tabel 4.8.

Tabel 4. 8 Hasil Pengujian Fungsi Kontrol Manual (*Open Loop*)

No	Aksi GUI Desktop	Respon	Status Pengujian	Keterangan
1	Menekan tombol aksi <i>Cleaning On</i>	Pompa air DC aktif menyemprot dan servo berputar menyapu panel.	Berhasil	Jalur data sukses
2	Menekan tombol aksi <i>Cleaning Off</i>	Seluruh motor penggerak dan pompa air berhenti seketika.	Berhasil	Jalur data sukses
3	Mengubah opsi mode ke pilihan <i>Auto</i>	Sistem mengunci fungsi manual dan mengaktifkan algoritma cerdas.	Berhasil	Jalur data sukses

4. Pengujian Close Loop (Sistem Kendali Cerdas Tertutup)

Pengujian sistem kendali loop tertutup cerdas (*closed loop*) dilaksanakan secara kontinu dengan mengalirkan 527 baris data hasil rekaman aktual di lapangan pada saat mode sistem diposisikan pada pilihan *Auto*. Parameter keluaran nilai tegas (*crisp output*) PWM hasil komputasi mikrokontroler ESP32 yang kemudian di proses oleh MQTT berdasarkan inputan pada aplikasi *desktop* , kemudian dikorelasikan langsung dengan perhitungan rumus teoretis manual, seperti dijabarkan pada sampel pengujian Tabel 4.9.

Tabel 4. 9 Hasil Pengujian Fungsi Kontrol Otomatis (*Open Loop*)

No	No Paket Data	Input Sensor	PWM Manual	PWM ESP32	Selisih	Ouput Motor
1	Pkt_102	Tegangan = 1,0 V; Suhu = 25,9°C; Cahaya = 3,3 Lux	0	0.00	0	STANDBY
2	Pkt_305	Tegangan = 20,2 V; Suhu = 35,8°C; Cahaya = 18.654 Lux	0	0.00	0	STANDBY
3	Pkt_334	Tegangan = 1,0 V; Suhu = 35,0°C; Cahaya = 21,0 Lux	212	212.12	0.12	CLEANING
4	Pkt_338	Tegangan = 1,0 V; Suhu = 35,0°C; Cahaya = 21,0 Lux	212	212.12	0.12	CLEANING

F. Evaluasi Hasil Pengujian

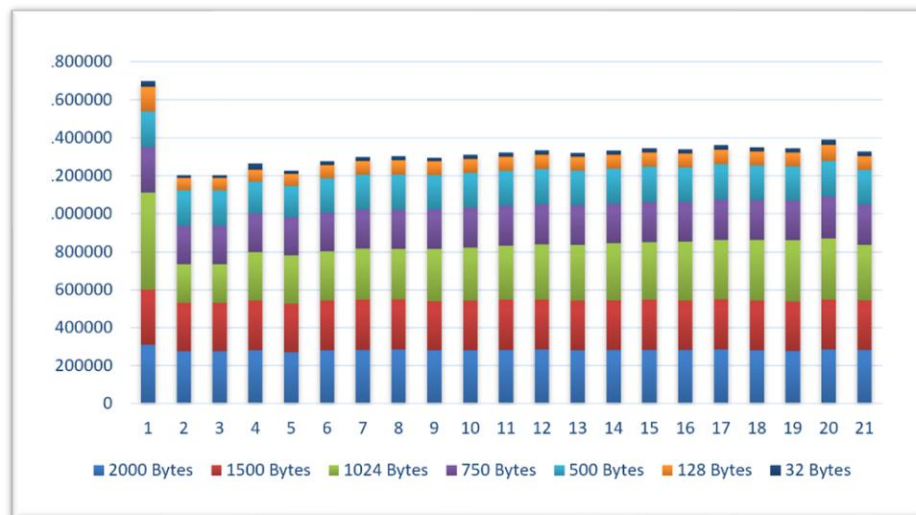
1. Respon Fungsional dan Non-Fungsional

Berdasarkan hasil pengujian fungsional, integrasi perangkat keras loop tertutup (*closed-loop*) terbukti mampu beroperasi secara mandiri tanpa memerlukan intervensi manual dari pengguna. Data fisik dari sensor BH-1750, DHT-11, dan *Voltage Sensor* berhasil ditransmisikan secara konsisten menuju algoritma fuzzy internal ESP32.

Namun, hasil pengujian non-fungsional pada kestabilan respon aturan menunjukkan adanya deviasi keputusan pada skenario 5 dan 7, sehingga akurasi sistem tertahan di angka 77,7%. Analisis mendalam

menunjukkan bahwa penyebab utama hasil belum sesuai harapan ini adalah terjadinya fenomena tumpang tindih (*overlapping*) yang terlalu lebar pada kurva grafik fungsi keanggotaan variabel masukan di dalam kode program. Batas-batas linguistik antara domain “Redup” ke “Terik” pada sensor cahaya, serta domain “Rendah” ke “Sedang” pada sensor tegangan memiliki irisan nilai derajat keanggotaan (μ) yang ambigu. Akibatnya, mesin inferensi FIS Mamdani mengeksekusi *rule* yang kurang tepat saat sensor membaca nilai-nilai yang berada di area transisi tersebut.

Dari aspek performansi komunikasi data, pengujian komparatif metrik *throughput* membuktikan adanya pengaruh nyata dari fragmentasi paket pada lapisan transport. Penurunan nilai transfer bit bersih yang terjadi secara spesifik pada titik pembebanan *payload* sebesar 1500 Bytes disebabkan oleh dimensi ukuran paket yang mulai melampaui ambang batas *Maximum Transmission Unit* (MTU) pada arsitektur jaringan nirkabel lokal. Hal ini memaksa lapisan transport untuk memecah bungkusan data menjadi beberapa fragmen kecil, sehingga memicu *overhead* komputasi dan mereduksi kelajuan transfer efektif.

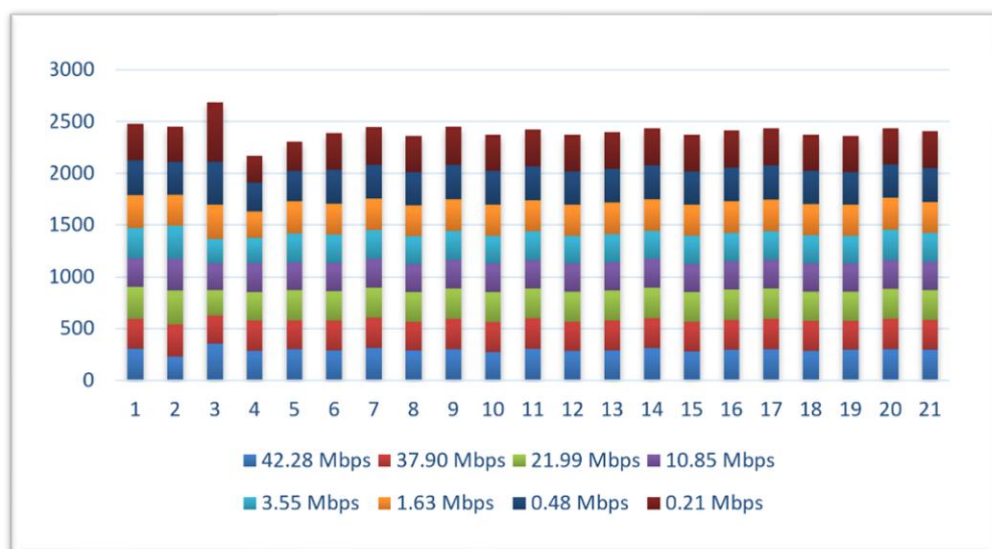


Gambar 4.6. Grafik Rata-Rata *Throughput*

Pada Gambar 4.6 memperlihatkan tren kenaikan yang linier dari payload 32 Bytes (22.400 bps), 128 Bytes (76.800 bps), 500 Bytes (183.333 bps), hingga 750 Bytes (209.375 bps), dan mencapai titik puncak pada 1024

Bytes sebesar 287.520 bps. Terjadi penurunan *throughput* menjadi 263.214 bps saat muatan dinaikkan ke 1500 Bytes, sebelum kembali naik ke angka 282.539 bps pada pembebanan 2000 Bytes. Penurunan di titik 1500 Bytes ini membuktikan adanya pengaruh nyata dari fragmentasi paket pada lapisan transport akibat ukuran data yang mulai melampaui batas *Maximum Transmission Unit* (MTU) nirkabel lokal, sehingga mereduksi kecepatan transfer bersih namun tetap menjaga stabilitas pengiriman informasi.

Pada parameter waktu tunda operasional berkala, sistem mencatatkan rata-rata *latency* transmisi sebesar 89,80 ms. Merujuk pada acuan internasional TIPHON, seluruh kualitas koneksi nirkabel dengan delay di bawah 150 ms diklasifikasikan ke dalam peringkat tertinggi, yaitu Kategori Sangat Bagus (Indeks 4). Keberhasilan latensi rendah ini sangat krusial bagi sistem kontrol IoT karena menjamin respon eksekusi data yang instan (*real-time execution*), sehingga aplikasi *desktop* dapat memperbarui visualisasi parameter fisik dalam hitungan milidetik dan menggerakkan aktuator sikat tanpa kendala jeda waktu.



Gambar 4.7. Grafik Rata-Rata *Latency*

Performansi keandalan respon ini diuji secara komparatif melalui variasi kecepatan internet yang disajikan pada Gambar 4.7. Pada spektrum lebar pita internet tinggi hingga menengah antara 42,28 Mbps hingga 3,55 Mbps, nilai

rata-rata *latency* mampu bertahan sangat stabil di kisaran 274,82 ms hingga 294,38 ms, membuktikan bahwa pembatasan koneksi tingkat medium tidak mendegradasi performa protokol MQTT. Sebaliknya, grafik menunjukkan lonjakan tajam saat kecepatan diturunkan ke bawah 1 Mbps, di mana kecepatan 1,63 Mbps menghasilkan latensi 303,07 ms, naik menjadi 327,99 ms pada 0,48 Mbps, dan mencapai titik terlambat 363,05 ms pada kecepatan ekstrem 0,21 Mbps. Kenaikan kurva pada kondisi internet lambat ini mengonfirmasi adanya gejala akumulasi antrean paket data (*packet queuing delay*) pada memori buffer ESP32 akibat menyempitnya jalur pipa digital di lapangan.

2. Perbandingan Capaian Sebelum dan Sesudah Implementasi

a. Sebelum Implementasi

Arsitektur sistem kendali pembersihan masih bersifat loop terbuka (*open-loop*) dan semi-manual. Pengguna diwajibkan memantau pergerakan angka sensor pada layar secara manual, lalu mengambil kesimpulan subyektif untuk menekan tombol *cleaning* pada aplikasi. Apabila terjadi penurunan daya panel akibat tumpukan debu di siang hari namun pengguna tidak memeriksa aplikasi, mekanisme pembersihan aktuator tidak akan pernah berjalan. Dari sisi komunikasi, penggunaan arsitektur Firebase memakan kapasitas lebar pita (*bandwidth*) yang lebih besar akibat beban *header* paket *request-response* HTTP yang berat untuk kendali interaktif perangkat keras

b. Sesudah Implementasi

Arsitektur sistem kendali bertransformasi sepenuhnya menjadi loop tertutup (*closed-loop*) otomatis yang adaptif. Melalui inteligensi *fuzzy logic*, mikrokontroler secara mandiri mampu membedakan penyebab jatuhnya nilai tegangan keluaran panel surya. Jika tegangan jatuh akibat faktor alamiah waktu malam hari (intensitas cahaya rendah), sistem secara otomatis mengunci diri dalam status *standby*. Namun, jika tegangan jatuh pada saat kondisi siang hari terik (intensitas cahaya tinggi), sistem langsung mengidentifikasi adanya hambatan debu kotoran dan secara instan mengaktifkan aktuator pompa air mini serta motor sikat pada pulsa PWM

tinggi bernilai 212 tanpa membutuhkan instruksi manusia. Pergantian protokol komunikasi menjadi MQTT memotong beban *overhead header* tetap hingga tersisa 2 bytes saja, sehingga menghasilkan komunikasi dua arah yang jauh lebih responsif, hemat kuota data, dan ramah terhadap kapasitas *bandwidth* jaringan lokal.

3. Perbandingan dengan Penelitian Terdahulu

a. (Sari *et al.*, 2022)

Mengembangkan mekanisme sikat gulung (*rolling brush*) dan *wiper* berbasis Arduino yang operasional alatnya masih bergantung pada jadwal waktu (*timer* kaku) atau sakelar manual. Produk pengembangan dalam penelitian skripsi ini jauh lebih adaptif karena keputusan pembersihan tidak lagi bergantung pada *timer*, melainkan pada data *real-time* multivariabel kondisi lingkungan.

b. (Cahyanti *et al.*, 2025)

Berhasil menerapkan kendali *Fuzzy Logic* pada robot pembersih panel surya, namun terbatas menggunakan metode *dry cleaning* (pembersihan kering tanpa air). Penelitian ini menawarkan keunggulan fungsional lewat metode *wet cleaning* (dengan integrasi pompa air mini) untuk hasil pengangkatan kotoran membandel yang lebih maksimal, sekaligus melengkapinya dengan arsitektur monitoring komprehensif berbasis IoT.

c. (Helilintar *et al.*, 2025)

Pada Jurnal Dimastara (Sinta 5): Telah melakukan validasi empiris bahwa sistem panel surya berbasis IoT efektif mempertahankan stabilitas tegangan listrik di Greenhouse perkebunan. Namun, prototipe pada proyek Dimastara tersebut masih memiliki kelemahan fatal pada mekanisme eksekusi yang bersifat semi-manual berbasis database Firebase. Penelitian skripsi ini berhasil menyempurnakan kelemahan tersebut secara tuntas melalui implementasi algoritma cerdas fuzzy logic Mamdani untuk otomatisasi penuh (*fully automatic*) dan melakukan migrasi total ke protokol

komunikasi MQTT yang memiliki latensi komunikasi jauh lebih rendah dan responsif untuk kendali dua arah secara *real-time*.

2

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan seluruh rangkaian tahapan perancangan, implementasi, dan pengujian secara langsung yang telah dilaksanakan terhadap prototipe *Smart Solar Panel* terintegrasi *Internet of Things (IoT)* menggunakan protokol MQTT, maka dapat diambil beberapa kesimpulan utama sebagai berikut:

1. Penerapan algoritma *Fuzzy Logic Controller* pada mikrokontroler ESP32 terbukti beroperasi dengan sempurna. Melalui analisis eksperimental terhadap 527 paket data di lapangan, seluruh tahapan *fuzzy* mulai dari fuzzifikasi, evaluasi aturan (*rule base*), hingga defuzzifikasi *centroid* mampu mengidentifikasi kondisi permukaan panel secara akurat. Sistem ini sangat adaptif dalam membedakan penurunan tegangan akibat waktu malam hari (yang mengunci sistem pada status *standby*) dengan penurunan daya akibat hambatan debu pada siang hari terik (yang secara otomatis mengaktifkan sikat mekanis pada pulsa PWM tinggi bernilai 212). Selain itu, tingkat keamanan fisik alat telah terpenuhi secara optimal melalui integrasi *fail-safe system* yang responsif menghentikan total arus penggerak seketika saat mendeteksi kegagalan data atau sensor eror (ERR!), sehingga komponen mekanik terhindar dari kerusakan struktural dan beban panas berlebih (*overheating*).
2. Penerapan teknologi IoT dengan protokol MQTT menunjukkan hasil performansi *Quality of Service (QoS)* yang bervariasi namun tetap mendukung fungsionalitas alat secara penuh. Pengujian parameter *latency* (waktu tunda pengiriman paket) mencatatkan nilai rata-rata keseluruhan sebesar 89,80 ms, yang secara mutlak masuk ke dalam kategori Sangat Bagus (Indeks 4) berdasarkan standar internasional TIPHON karena berada jauh di bawah ambang batas maksimal 150 ms, sehingga menjamin pertukaran data monitoring berjalan secara *real-time*. Di sisi lain, parameter *throughput* menghasilkan rata-rata laju data konvergen yang rendah sebesar

11

52

64

133,89 bps, sehingga masuk dalam kategori Jelek (Indeks 1) menurut ukuran telekomunikasi umum. Meskipun demikian, nilai kelajuan yang kecil ini justru menjadi bukti ilmiah keberhasilan efisiensi protokol MQTT yang sukses memangkas beban tambahan data (*overhead header*) untuk pengiriman bungkusan teks JSON berkapasitas 128 Byte, sehingga sistem monitoring terbukti sangat hemat konsumsi kuota data internet dan ramah terhadap kapasitas lebar pita (*bandwidth*) jaringan lokal.

B. Saran

4 Beberapa saran yang dapat diajukan untuk pengembangan, penyempurnaan, dan keberlanjutan sistem pada penelitian-penelitian selanjutnya meliputi penambahan fungsi filter pembacaan parameter data sensor pada algoritma pemrograman mikrokontroler untuk mereduksi efek fluktuasi data yang ekstrem saat terjadi momen transisi perubahan cuaca mendadak dari cerah ke mendung, sehingga eksekusi status aktuator pembersih di lapangan menjadi lebih stabil dan presisi. Selain itu, disarankan pula untuk menerapkan metode distribusi komputasi berbasis *Edge Computing* pada pengembangan berikutnya, di mana proses pengolahan data sensor tingkat lanjut dan kalkulasi pengambilan keputusan dapat dipisahkan atau dialihkan ke sisi server *cloud* guna menjaga kestabilan memori kerja serta meringankan beban performa komputasi pada keping *chip* mikrokontroler ESP32 di lapangan. Terakhir, sejalan dengan batasan masalah penelitian awal yang telah ditetapkan, diharapkan pada riset masa depan dapat dilakukan pengujian kuantitatif yang lebih mendalam mengenai besaran efisiensi peningkatan daya konversi listrik dalam satuan Watt yang berhasil diserap oleh panel surya secara komparatif untuk membandingkan performa sebelum dan sesudah dibersihkan oleh lengan sikat mekanis otomatis secara berkala.