

BAB II

LANDASAN TEORI

2.1 Posisi Penelitian (State Of The Art)

State of the art sebelumnya dilakukan untuk mengidentifikasi perkembangan dalam penelitian terkait sistem pemantauan berbasis *Internet of Things (IoT)*, kontrol kipas otomatis, *data logging*, dan pemanfaatan energi panel surya. Melalui tinjauan ini, kesamaan, perbedaan, dan celah penelitian diidentifikasi untuk dijadikan dasar pengembangan studi ini, memastikan bahwa sistem yang diusulkan menawarkan kontribusi dan kebaruan dibandingkan dengan penelitian sebelumnya. Selain itu, ringkasan penelitian sebelumnya disajikan dalam bentuk tabel *State of the Art*

Tabel 2. 1 *State OF The Art*

No	Peneliti, (Tahun)	Judul Penelitian	Metode	Objek Penelitian	Perbandingan yang dijadikan alasan tinjauan penelitian	Link DOI
1	Adiprabowo et al. (2026)	<i>Eco-enzyme development based on IoT and computerization</i>	Pendekatan Partisipatif.	Pengolahan limbah organik	Tidak membahas sistem kontrol otomatis dan tidak menggunakan energi surya	DOI : 10.31763/iot.v6i2.1034
2	Sutikno et al., 2023)	<i>Monitoring Tegangan dan Arus Pada Panel Surya Menggunakan IoT</i>	Experimen, Studi Literatur.	<i>Monitoring</i> tegangan dan arus.	Hanya <i>monitoring</i> , belum ada kontrol multi level, notifikasi, dan <i>fail safe</i>	DOI : 10.24843/mite.2023.v22i01.p20
3	Aswadi (2025)	<i>Monitoring Kualitas Udara Berbasis IoT</i>	Research and Development	Kualitas udara	Tidak terdapat sistem	DOI: 10.70716/jocsit.v1i2.225

No	Peneliti, (Tahun)	Judul Penelitian	Metode	Objek Penelitian	Perbandingan yang dijadikan alasan tinjauan penelitian	Link DOI
			pment (R&D)	dalam ruangan	kontrol aktuator	
4	Engla Desita & Endah Fitriani (2025)	Pemanfaatan Panel Surya Berbasis IoT	Research and Development (R&D), <i>prototype</i>	Energi pemanas air listrik	Tidak mengintegrasikan kontrol multi level dan <i>Data logging</i>	DOI: 10.61132/uranus.v3i1.690
5	Gymnasdiar et al. (2026)	Smart Gate Guardian Berbasis IoT	Prototyping.	Gerbang otomatis	Fokus pada keamanan, tidak pada <i>monitoring</i> suhu dan energi	DOI: 10.56211/sudo.v5i1.1523
6	Hermandiana et al. (2025)	Pemanfaatan IoT untuk Energi Terbarukan	Studi literatur sistematis.	Sistem energi terbarukan	Tidak membahas kontrol bertingkat (<i>Multi level control</i>)	DOI: 10.28926/jtpdm.v5i4.2288
7	Irham Fadilah & Endah Fitriani (2025)	Protokol Alternatif IoT Saat Internet Terputus	Research and Development (R&D).	Sistem pertanian IoT	Tidak fokus pada <i>monitoring</i> suhu dan kontrol kipas	DOI: 10.61132/jupiter.v3i1.705
8	Mursidan et al. (2023)	<i>Monitoring Energi Listrik Berbasis IoT</i>	Rancang Bangun.	Energi listrik generator surya	Tidak terdapat sistem kontrol otomatis	DOI: 10.26740/jte.v12n2.p92-97
9	Pane et al. (n.d.)	<i>Automatic Monitoring System IoT Based Water Tanks</i>	<i>prototype</i> .	Tanki air	Tidak menggunakan energi surya dan tidak ada	DOI: 10.46729/ijsttm.v5i4.1151

No	Peneliti, (Tahun)	Judul Penelitian	Metode	Objek Penelitian	Perbandingan yang dijadikan alasan tinjauan penelitian	Link DOI
					kontrol multi level	
10	Paredes Cornejo et al. (2025)	<i>Automatic Control System and IoT Architecture</i>	prototipe.	Green house	Tidak menggunakan energi terbarukan dan tidak ada <i>Data logging</i> lokal	DOI: 10.14445/23488379/IJEE-E-V12I3P119
11	Rahaman (2022)	<i>Review IoT Architecture</i>	Studi literatur.	Arsitektur IoT	Bersifat konseptual, tidak implementasi sistem nyata	DOI: 10.5281/zenodo.7066810
12	Reddy et al. (2025)	<i>IoT Smart Mirror with News and Temperature</i>	Studi Literatur.	<i>Smart mirror</i> (suhu)	Hanya <i>monitoring</i> suhu tanpa kontrol sistem	DOI: 10.34218/IJVAR
13	Setiawan et al. (2025)	Desa Peternakan Mandiri Energi: Implementasi IOT Dengan Panel Surya dan Pit Biogas Sebagai Solusi Berkelanjutan.	Pendampingan Masyarakat.	Sistem energi desa	Tidak mengintegrasikan kontrol suhu.	DOI: 10.35446/pe-ngabdiankompetif.v4i2.2536
14	Sharma et al. (2025)	<i>Sensor Failure Mitigation in RTOS-Based IoT Systems Using Machine Learning</i>	Experiment.	Sistem IoT berbasis RTOS dengan sensor sebagai	Penelitian ini unggul dalam penanganan kegagalan sensor melalui pendekatan prediktif	DOI: 10.1016/j.jpdc.2025.105161

No	Peneliti, (Tahun)	Judul Penelitian	Metode	Objek Penelitian	Perbandingan yang dijadikan alasan tinjauan penelitian	Link DOI
				i sumber data utama	(<i>fail safe</i> berbasis AI), namun belum secara spesifik mengintegrasikan sistem notifikasi <i>realtime</i> yang kompleks	
15	(Zhang et al., 2025)	<i>Developing Realtime IoT-Based Public Safety Alert and Emergency Response Systems</i>	Rancangan bangun .	Sistem keselamatan publik berbasis IoT dengan multi-sensor	Penelitian ini fokus pada notifikasi <i>realtime</i> dan <i>Data logging</i> berbasis cloud dengan <i>fail safe</i> jaringan, namun belum membahas pemulihan data sensor.	DOI: 10.1038/s41598-025-13465-7

Kebaruan (*novelty*) dan kontribusi ilmiah yang membedakan suatu penelitian dari penelitian-penelitian sebelumnya. Pada bidang *Internet of Things* (IoT), berbagai penelitian telah dilakukan terkait sistem *monitoring* suhu, pengendalian kipas otomatis, maupun pemanfaatan energi surya sebagai sumber daya alternatif. Namun, sebagian besar penelitian terdahulu masih berfokus pada salah satu aspek tertentu, seperti *monitoring* suhu berbasis IoT, pengendalian kipas menggunakan metode *ON/OFF*, atau *monitoring* sistem panel surya secara terpisah. Selain itu, sebagian penelitian hanya menampilkan data secara *realtime* tanpa menyediakan

mekanisme penyimpanan data historis (*Data logging*) yang terintegrasi untuk kebutuhan analisis dan evaluasi sistem.

Penelitian lain telah mengembangkan sistem pengendalian kipas otomatis berbasis suhu, namun umumnya masih menggunakan metode kontrol sederhana berupa *ON/OFF* sehingga respons sistem kurang adaptif terhadap perubahan suhu lingkungan. Di sisi lain, penelitian mengenai *monitoring* panel surya berbasis IoT lebih banyak berfokus pada pengukuran parameter listrik seperti tegangan, arus, dan daya tanpa mengintegrasikan fungsi pengendalian suhu secara otomatis. Kondisi tersebut menunjukkan bahwa pengembangan sistem *monitoring*, pengendalian, dan pemanfaatan energi terbarukan masih banyak dilakukan secara parsial dan belum terintegrasi dalam satu *platform* yang komprehensif.

Berdasarkan telaah terhadap penelitian terdahulu, penelitian ini memiliki beberapa aspek kebaruan yang menjadi pembeda utama. Pertama, penelitian ini mengintegrasikan fungsi *monitoring* suhu dan parameter listrik (tegangan dan arus), pengendalian kipas otomatis berbasis *Multi level control*, *Data logging*, notifikasi kondisi kritis, serta pemanfaatan panel surya sebagai sumber energi dalam satu sistem yang saling terhubung. Integrasi tersebut memberikan kemampuan yang lebih luas dibandingkan dengan penelitian sebelumnya yang umumnya hanya berfokus pada satu atau dua fungsi tertentu.

Kedua, dari aspek metode pengendalian, penelitian ini menerapkan kombinasi metode *ON/OFF* dan *Multi level control* untuk mengatur kecepatan kipas berdasarkan rentang suhu yang telah ditentukan. Pendekatan ini memungkinkan sistem memberikan respons yang lebih adaptif terhadap perubahan suhu dibandingkan dengan metode *ON/OFF* konvensional yang hanya memiliki dua kondisi operasi. Dengan demikian, sistem yang dikembangkan berpotensi meningkatkan efisiensi penggunaan energi sekaligus menjaga stabilitas suhu secara lebih optimal.

Ketiga, penelitian ini tidak menggunakan *platform monitoring* pihak ketiga yang umum digunakan pada penelitian IoT, melainkan mengembangkan *website*

monitoring secara mandiri menggunakan teknologi web *modern*. *Website* yang dikembangkan berfungsi sebagai media *monitoring realtime*, pengelolaan *Data logging*, visualisasi data dalam bentuk grafik, serta penyampaian notifikasi kepada pengguna. Pendekatan ini memberikan fleksibilitas yang lebih tinggi dalam pengembangan sistem serta memungkinkan penyesuaian fitur sesuai kebutuhan penelitian.

Keempat, penelitian ini mengintegrasikan panel surya sebagai sumber energi utama yang mendukung operasional sistem *monitoring* dan pengendalian. Selain melakukan pemantauan suhu lingkungan, sistem juga memonitor parameter listrik yang dihasilkan oleh panel surya sehingga memungkinkan evaluasi kondisi sumber energi secara bersamaan. Integrasi antara sistem IoT, pengendalian suhu, dan energi terbarukan masih relatif terbatas ditemukan pada penelitian-penelitian sebelumnya, khususnya pada implementasi skala prototipe yang dikembangkan secara menyeluruh.

Dari sisi kontribusi ilmiah, penelitian ini diharapkan dapat memperkaya kajian mengenai integrasi teknologi IoT, sistem kontrol bertingkat (*Multi level control*), *data logging*, dan energi terbarukan dalam satu arsitektur sistem yang terpadu. Hasil penelitian juga dapat menjadi referensi bagi pengembangan sistem *monitoring* dan kontrol cerdas yang memanfaatkan sumber energi mandiri berbasis panel surya. Selain memberikan kontribusi terhadap pengembangan ilmu pengetahuan di bidang Teknik Informatika, penelitian ini juga memiliki nilai praktis sebagai alternatif solusi *monitoring* dan pengendalian suhu yang efisien, berkelanjutan, dan dapat diterapkan pada lingkungan pendidikan, pertanian, maupun industri.

Dengan demikian, penelitian ini penting dilakukan karena berupaya mengisi kesenjangan penelitian yang masih terdapat pada aspek integrasi sistem *monitoring* suhu, pengendalian kipas berbasis *Multi level control*, *Data logging*, notifikasi, dan pemanfaatan panel surya dalam satu *platform* IoT yang terhubung melalui *website*. Kesenjangan tersebut menunjukkan adanya kebutuhan akan sistem yang tidak hanya mampu melakukan *monitoring* secara *realtime*, tetapi juga mampu

melakukan pengendalian adaptif, penyimpanan data historis, serta memanfaatkan energi terbarukan untuk mendukung operasional sistem secara berkelanjutan.

2.1 *Internet of Things*

Internet of Things (IoT) adalah konsep di mana objek tertentu memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer. IoT adalah teknologi nirkabel yang sering kita temui. Ini menawarkan berbagai kemampuan, termasuk berbagi data, kontrol jarak jauh, dan banyak lainnya. Pada dasarnya, aplikasinya juga dapat diterapkan pada objek dunia nyata yang ada di sekitar kita (Mursidan et al., 2023).

Peneliti lain menyebutkan bahwa *Internet of Things* adalah sistem terintegrasi yang dirancang untuk terus memperluas konektivitas internet. Teknologi ini menghubungkan berbagai perangkat atau objek melalui internet. *Internet of Things* (IoT) adalah teknologi yang memungkinkan pengguna untuk mengontrol, berkomunikasi, dan berkolaborasi dengan berbagai perangkat keras. Teknologi IoT sangat relevan di berbagai bidang. (Ayu Syahfitri, 2025).

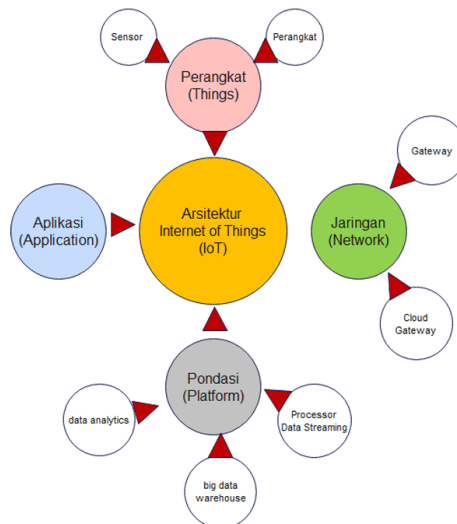
Internet of Things bukan hanya sebuah teknologi, tetapi juga merupakan faktor penting dalam mewujudkan sistem yang berkelanjutan. Penggunaan *Internet of Things* di bidang pendidikan dan energi tidak hanya memenuhi persyaratan modernisasi dan efisiensi, tetapi juga membantu menggerakkan perubahan menuju pembangunan berkelanjutan. Namun, infrastruktur yang terbatas, biaya implementasi yang tinggi, dan kurangnya pengetahuan teknologi, terutama di negara-negara berkembang, masih menjadi tantangan. (Hermadiana et al., 2025).

Sangat penting bagi teknologi *Internet of Things* (IoT) untuk memantau dan mengelola kinerja panel surya secara real-time. Teknologi IoT memungkinkan pengguna mengoptimalkan penggunaan energi terbarukan dengan memantau intensitas sinar matahari yang diterima panel dan status pengisian baterai. (Engla Desita & Endah Fitriani, 2025)

2.1.1 *Arsitektur Internet of Things (IoT)*

Arsitektur *Internet of Things* (IoT) terdiri atas empat lapisan utama, yaitu lapisan perangkat (*things*), lapisan jaringan (*network*), lapisan *platform* (*platform*),

dan lapisan aplikasi (*application*) yang saling terintegrasi dalam proses pengumpulan, pengiriman, pengolahan, hingga pemanfaatan data.



Gambar 2. 1 *Internet of Things*

Sumber: Adhitya W (2023 hlm. 202)

1. Lapisan Perangkat (*Things*)

Lapisan perangkat terdiri atas sensor, aktuator, dan berbagai perangkat pintar yang terhubung ke internet. Sensor berfungsi mengumpulkan data dari lingkungan, sedangkan aktuator menjalankan perintah yang diberikan sistem, seperti menyalakan atau mematikan perangkat. Contoh perangkat pada lapisan ini antara lain *smart home*, *smartwatch*, kendaraan pintar, dan perangkat industri (Adhitya W, 2023).

2. Lapisan Jaringan (*Network Layer*)

Lapisan jaringan berfungsi untuk menghubungkan perangkat IoT dengan internet menggunakan teknologi komunikasi seperti *Wi-Fi*, *Bluetooth*, *ZigBee*, dan *LoRaWAN*. Pada lapisan ini, gateway berperan menghubungkan perangkat dengan cloud, melakukan penyaringan data, serta memastikan komunikasi data berlangsung secara aman dan efisien (Adhitya W, 2023).

3. Lapisan *Platform* (*Platform Layer*)

Lapisan *platform* bertugas mengelola data yang diterima dari perangkat IoT melalui proses penyimpanan, pengolahan, dan analisis. Data yang tersimpan dapat dimanfaatkan untuk menghasilkan informasi, mendukung pengambilan keputusan, serta meningkatkan akurasi sistem melalui penerapan machine learning (Adhitya W, 2023).

4. Lapisan Aplikasi (*Application Layer*)

Lapisan aplikasi merupakan antarmuka yang memungkinkan pengguna memantau dan mengendalikan perangkat IoT melalui aplikasi web atau seluler. Selain itu, lapisan ini mendukung fungsi *monitoring*, otomatisasi, dan pemberian notifikasi secara *realtime* sesuai data yang diterima dari perangkat (Adhitya W, 2023).

2.1.2 Komponen *Internet of Things* (IoT)

Internet of Things (IoT) terdiri atas beberapa komponen utama yang saling terintegrasi untuk mendukung proses konektivitas dan pertukaran data antarperangkat. Komponen tersebut meliputi sensor, perangkat terhubung, jaringan komunikasi, *platform*, dan aplikasi (Erwin et al., 2023).

1. Sensor

Sensor merupakan komponen utama dalam sistem IoT yang berfungsi mendeteksi dan mengukur berbagai parameter fisik, seperti suhu, kelembapan, intensitas cahaya, dan gerakan. Data yang diperoleh dari sensor digunakan sebagai masukan bagi sistem sehingga perangkat dapat merespons kondisi lingkungan secara otomatis (Giaffreda et al., 2020).

2. Perangkat Terhubung (*Connected Devices*)

Perangkat terhubung atau *edge devices* merupakan perangkat fisik yang memiliki kemampuan komputasi dan komunikasi serta terhubung ke internet. Perangkat ini menerima dan memproses data dari sensor, kemudian mengambil keputusan atau menjalankan tindakan tertentu sebelum data diteruskan ke *platform* atau cloud untuk proses lebih lanjut (Romanelli, 2021).

3. Jaringan Komunikasi

Jaringan komunikasi berfungsi menghubungkan perangkat IoT dengan internet sehingga memungkinkan pertukaran data antara perangkat dan *platform*. Teknologi yang umum digunakan meliputi *WiFi*, *Bluetooth*, *ZigBee*, *LoRaWAN*, dan *NB-IoT*. Pemilihan teknologi jaringan yang tepat berpengaruh terhadap jangkauan, kecepatan, dan efisiensi komunikasi dalam sistem IoT (Al-Fagih et al., 2021).

4. Platform

Platform IoT merupakan lapisan perangkat lunak yang berfungsi mengelola, menyimpan, dan menganalisis data dari berbagai perangkat yang terhubung. Selain itu, *platform* menyediakan layanan manajemen perangkat dan mendukung pengembangan aplikasi sesuai kebutuhan pengguna (Erwin et al., 2023).

5. Aplikasi

Aplikasi merupakan antarmuka yang digunakan pengguna untuk memantau, mengendalikan, dan memperoleh informasi dari sistem IoT. Aplikasi dapat berupa aplikasi seluler, *dashboard* berbasis web, maupun sistem otomatisasi yang diterapkan pada berbagai bidang, seperti rumah pintar, kesehatan, dan transportasi (Erwin et al., 2023).

2.2 Energi Surya (Solar Panel)

Energi surya adalah sumber energi terbesar yang dapat dikonversi menjadi listrik, yang digunakan oleh manusia untuk memenuhi kebutuhan saat ini dan masa depan. (Dewi, Arfita Yuana, 2025). Metode kerjanya bergantung pada panel surya

atau solar sel yang mengubah energi matahari menjadi energi listrik. (Mursidan et al., 2023)

Dengan efisiensi teknologi photovoltaic saat ini, potensi energi surya di Indonesia mencapai lebih dari 200 GW, menurut peneliti lain. Namun, penggunaan energi surya masih kurang dari 100 MW, terutama di Kalimantan Barat (20 GW), Sumatera Selatan (17 GW), dan Kalimantan Timur (13 GW). (Ferdyson & Windarta, 2023)

No	Provinsi	Potensi	No	Provinsi	Potensi
1	Kalimantan Barat	20.113	18	Sumatera Barat	5.898
2	Sumatera Selatan	17.233	19	Kalimantan Utara	4.643
3	Kalimantan Timur	13.479	20	Sulawesi Tenggara	3.917
4	Sumatera Utara	11.851	21	Bengkulu	3.475
5	Jawa Timur	10.335	22	Maluku Utara	3.036
6	Nusa Tenggara Barat	9.931	23	Bangka Belitung	2.810
7	Jawa Barat	9.099	24	Banten	2.461
8	Jambi	8.847	25	Lampung	2.238
9	Jawa Tengah	8.753	26	Sulawesi Utara	2.113
10	Kalimantan Tengah	8.459	27	Papua	2.035
11	Aceh	7.881	28	Maluku	2.020
12	Kepulauan Riau	7.763	29	Sulawesi Barat	1.677
13	Sulawesi Selatan	7.588	30	Bali	1.254
14	Nusa Tenggara Timur	7.272	31	Gorontalo	1.218
15	Papua Barat	6.307	32	DI Yogyakarta	996
16	Sulawesi Tengah	6.187	33	Riau	753
17	Kalimantan Selatan	6.031	34	DKI Jakarta	225
				Total	207.898

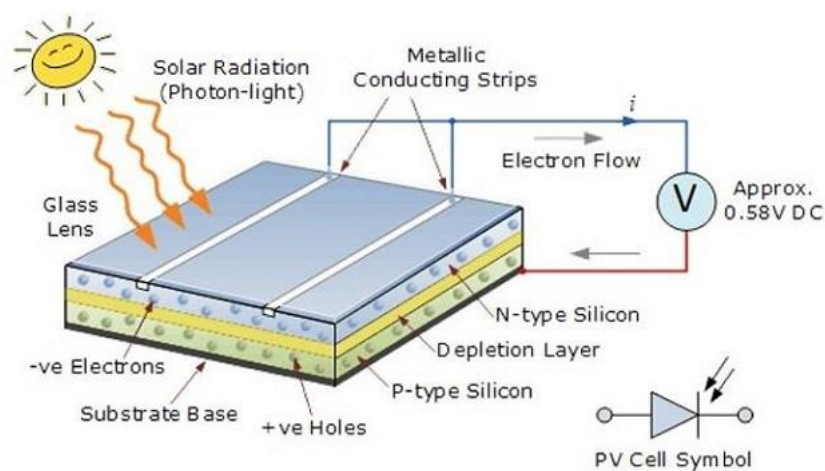
Gambar 2. 2 Potensi energi surya

Sumber : Kementrian Energi dan Sumber daya

Dengan daya rata-rata 4 kilowatt-jam per meter persegi, hampir semua wilayah Indonesia memiliki potensi untuk mengembangkan pembangkit listrik tenaga surya (PLTS), dengan variasi bulanan hingga 10%. Khususnya di wilayah barat Indonesia, potensi ini mencapai sekitar 4,5 kilowatt-jam per meter persegi per hari. Sementara wilayah timur Indonesia memiliki potensi tenaga surya sekitar 5,1 kilowatt jam per meter kubik per hari dengan variasi bulanan sekitar 9%, sedangkan potensi tenaga surya secara nasional mencapai 4,8 kilowatt jam per meter kubik per hari, atau 207.898 MW. Pemanfaatan energi surya di Indonesia saat ini hanya

mencapai sekitar 0,05%, atau 100 MW, sehingga diperlukan peningkatan sekitar 900 MW untuk mencapai target RUEN. Pemerintah membuat roadmap pemanfaatan energi surya dengan tujuan mencapai kapasitas PLTS terpasang sebesar 6,5 GW hingga tahun 2025. Jumlah ini menunjukkan potensi pasar yang sangat besar untuk pertumbuhan energi surya di masa mendatang. (Dewan Energi Nasional, 2020).

2.2.1 Prinsip Kerja Panel Surya



Gambar 2. 3 Prinsip kerja panel surya

Sumber: (Rudiyanto et al., 2023 hlm. 77)

Sel surya merupakan perangkat semikonduktor yang tersusun atas lapisan bermuatan positif dan negatif serta bekerja berdasarkan efek fotovoltaiik (*photovoltaic effect*), yaitu proses mengubah energi cahaya matahari secara langsung menjadi energi listrik. Material utama yang digunakan adalah silikon kristal yang memiliki struktur atom dengan elektron valensi yang memungkinkan terbentuknya ikatan kristal semikonduktor (Rudiyanto et al., 2023).

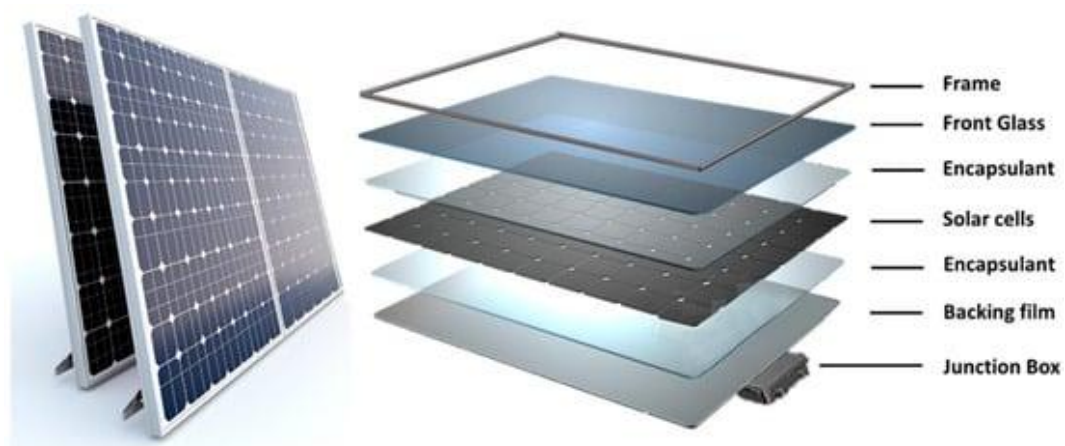
Ketika atom-atom silikon membentuk struktur kristal, setiap atom berbagi elektron valensi dengan atom di sekitarnya sehingga tercipta keseimbangan gaya tarik antara elektron dan inti atom. Struktur ini menghasilkan sifat semikonduktor yang mendukung terjadinya efek fotovoltaiik (Rudiyanto et al., 2023).

Pada saat cahaya matahari mengenai permukaan sel surya, energi foton akan membebaskan elektron pada material semikonduktor. Elektron yang bergerak dari semikonduktor tipe N menuju tipe P menghasilkan aliran elektron yang membentuk arus listrik searah (*Direct Current/DC*). Prinsip kerja inilah yang menjadi dasar pemanfaatan teknologi fotovoltaik pada sistem pembangkit listrik tenaga surya hingga saat ini (Rudiyanto et al., 2023).

2.2.2 Jenis – Jenis Panel Surya

Berdasarkan material penyusun dan teknologi pembuatannya, panel surya dibedakan menjadi beberapa jenis, yaitu *monokristalin*, *polikristalin*, *thin film*, dan *organic photovoltaic* (OPV).

1. Panel Surya *Monokristalin*



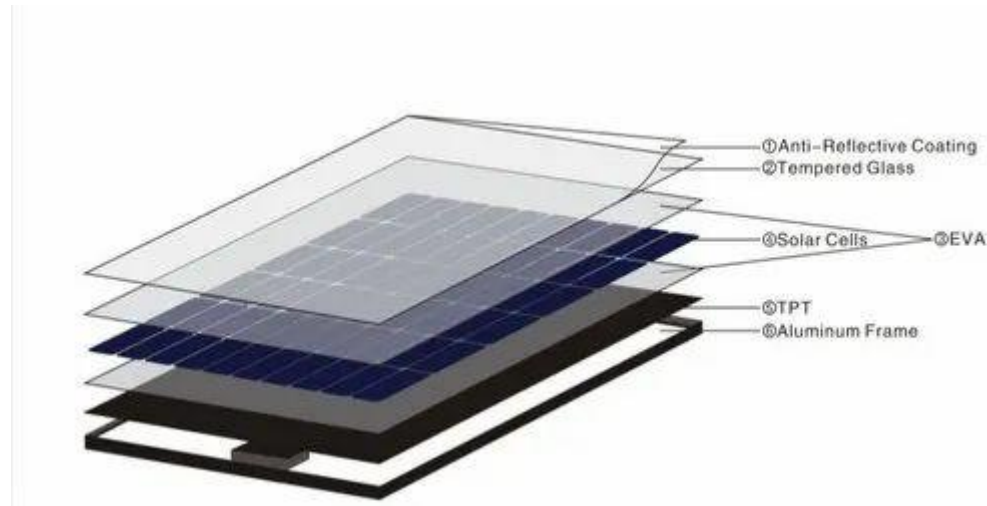
Gambar 2. 4 Panel Surya Monokristalin

Sumber: (Growingsilarmist.com)

Panel surya monokristalin dibuat dari kristal silikon tunggal dengan tingkat kemurnian yang tinggi. Proses pembuatannya menggunakan metode *Czochralski*, sehingga menghasilkan sel surya dengan struktur yang seragam dan efisiensi konversi energi yang tinggi, yaitu sekitar 15–22%. Panel ini memiliki warna hitam pekat, umur pakai yang panjang (sekitar 25 tahun), serta membutuhkan area pemasangan yang lebih kecil dibandingkan dengan jenis lainnya. Namun, biaya produksinya relatif lebih mahal karena proses manufakturnya lebih kompleks dan

menghasilkan limbah silikon yang cukup besar. Panel monokristalin banyak digunakan pada sistem pembangkit listrik tenaga surya yang memerlukan efisiensi tinggi dan memiliki keterbatasan ruang pemasangan (Rudiyanto et al., 2023).

2. Panel Surya *Polikristalin*



Gambar 2. 5 Panel Surya Polikristalin

Sumber: (indiaMART.com)

Panel surya polikristalin dibuat dari beberapa kristal silikon yang dilebur dan dicetak menjadi blok sebelum dipotong menjadi wafer. Teknologi ini menghasilkan biaya produksi yang lebih rendah dibandingkan dengan panel monokristalin. Efisiensi panel polikristalin berkisar antara 13–18%, sehingga memerlukan luas permukaan yang lebih besar untuk menghasilkan daya yang sama. Meskipun demikian, panel ini tetap mampu bekerja dengan baik pada kondisi cuaca mendung dan banyak digunakan karena harganya lebih ekonomis (Rudiyanto et al., 2023).

3. Panel Surya *Thin Film*

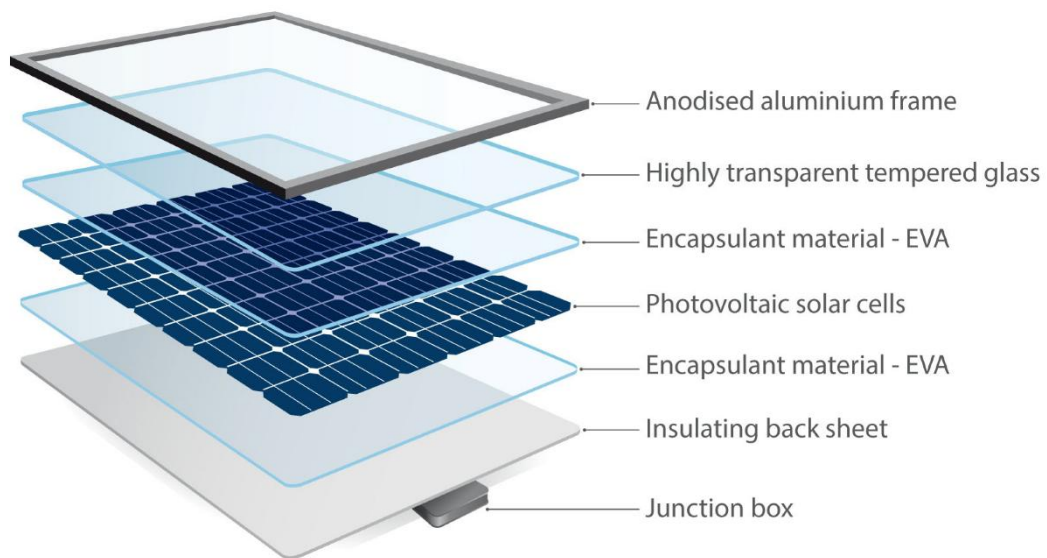


Gambar 2. 6 Panel Surya Thin Film

Sumber: (sanpower.com)

Panel surya *Thin Film* merupakan panel surya generasi kedua yang dibuat dengan menambahkan lapisan tipis material fotovoltaiik pada suatu substrat. Jenis ini memiliki karakteristik ringan, fleksibel, dan biaya produksi yang relatif rendah. Efisiensi panel Thin Film bervariasi tergantung pada material penyusunnya, umumnya berada pada kisaran 6–18%. Beberapa teknologi *thin film* yang banyak digunakan meliputi *Amorphous Silicon (a-Si)*, *Cadmium Telluride (CdTe)*, dan *Copper Indium Gallium Selenide (CIGS)*. Masing-masing memiliki keunggulan dan keterbatasan, seperti biaya produksi, efisiensi, fleksibilitas, serta penggunaan material tertentu yang bersifat beracun seperti kadmium pada teknologi CdTe dan sebagian CIGS (Rudiyanto et al., 2023).

4. *Organic Photovoltaic (OPV)*



Gambar 2. 7 Panel Surya *Organic Photovoltaic*

Sumber: (mdpi.com)

Organic Photovoltaic (OPV) merupakan teknologi sel surya generasi ketiga yang menggunakan material organik sebagai lapisan penyerap cahaya. Panel ini memiliki keunggulan berupa bobot yang ringan, fleksibel, biaya produksi yang rendah, serta dapat dibuat dalam berbagai bentuk dan warna. Namun, teknologi OPV masih memiliki efisiensi yang relatif rendah, sekitar 8–11%, dan umur operasional yang lebih pendek dibandingkan panel surya berbasis silikon. Pengembangan teknologi ini terus dilakukan, termasuk melalui pemanfaatan material perovskite yang mampu meningkatkan efisiensi konversi energi hingga lebih dari 20%, sehingga menjadi salah satu teknologi panel surya yang menjanjikan di masa depan (Rudiyanto et al., 2023).

2.2.3 Spesifikasi Panel Surya

Table 2. 2 Spesifikasi panel surya

Parameter	Perkiraan
Jenis	Monocrystalline
Daya	6W
Tegangan kerja	5V–6V
Arus maksimal	0.8A–1A
Penggunaan	IoT / charger / mini solar system

Panel surya yang digunakan pada penelitian ini merupakan panel surya jenis monocrystalline berdaya 6W. Panel surya ini digunakan sebagai sumber energi utama untuk menyuplai sistem *monitoring* suhu dan pengendalian kipas berbasis IoT. Jenis monocrystalline dipilih karena memiliki efisiensi konversi energi yang lebih tinggi dibandingkan dengan jenis lainnya, sehingga lebih cocok digunakan pada sistem berskala kecil dengan kebutuhan daya rendah hingga menengah. Selain itu, panel monocrystalline memiliki ukuran yang lebih ringkas dan performa yang lebih baik pada intensitas cahaya yang berubah-ubah (Rudiyanto et al., 2023).

2.3 Modul Pengisian dan Penyimpanan Energi

Modul pengisian dan penyimpanan energi merupakan bagian penting dalam sistem berbasis energi terbarukan yang berfungsi untuk mengelola energi listrik yang dihasilkan oleh panel surya sebelum digunakan oleh beban. Pada sistem *Internet of Things* (IoT), keberadaan modul ini sangat diperlukan untuk menjaga kontinuitas suplai daya sehingga perangkat dapat tetap beroperasi meskipun terjadi perubahan intensitas cahaya matahari. Sistem penyimpanan energi berperan penting dalam meningkatkan keandalan perangkat IoT yang memanfaatkan sumber energi terbarukan karena mampu menjaga kestabilan pasokan energi selama siklus operasi berlangsung (Orumwense & Abo-Al-Ez, 2022). Selain itu, penerapan teknologi *energy harvesting* yang dipadukan dengan sistem penyimpanan energi memungkinkan perangkat IoT beroperasi secara mandiri dan lebih efisien dalam memanfaatkan sumber energi terbarukan (Fadil et al., 2023).

Dalam penelitian ini, modul pengisian dan penyimpanan energi terdiri atas modul pengisian baterai TP4056, baterai *lithium-ion* tipe 18650, dan *buck converter* LM2596S. Ketiga komponen tersebut saling terintegrasi untuk membentuk sistem manajemen daya yang mampu menyimpan energi hasil konversi panel surya dan mendistribusikannya secara stabil ke seluruh perangkat yang digunakan. Integrasi sistem penyimpanan energi yang baik menjadi faktor penting dalam meningkatkan efisiensi dan keandalan sistem *monitoring* berbasis IoT yang bekerja secara *realtime*.

a. Modul TP4056

TP4056 merupakan modul pengisian daya yang dirancang khusus untuk baterai lithium-ion satu sel dengan tegangan nominal 3,7 volt. Modul ini menggunakan metode pengisian *Constant Current/Constant Voltage (CC/CV)* yang memungkinkan proses pengisian berlangsung secara aman dan efisien. Metode tersebut bekerja dengan memberikan arus konstan pada tahap awal pengisian, kemudian beralih ke tegangan konstan ketika baterai mendekati kapasitas penuh, sehingga dapat memperpanjang umur pakai baterai dan meningkatkan keamanan sistem.

Selain berfungsi sebagai pengisi daya, TP4056 umumnya dilengkapi dengan fitur proteksi terhadap kondisi *overcharge*, *over-discharge*, dan *short circuit*. Fitur proteksi tersebut sangat penting untuk menjaga keamanan baterai lithium-ion yang sensitif terhadap perubahan tegangan dan arus. Pada penelitian ini, TP4056 digunakan untuk mengatur proses pengisian baterai 18650 yang memperoleh sumber energi dari panel surya, sehingga energi yang tersimpan dapat dimanfaatkan secara optimal untuk mendukung operasional sistem *monitoring* dan pengendalian.

b. Baterai Lithium Ion 18650

Baterai *lithium-ion* tipe 18650 merupakan salah satu teknologi penyimpanan energi yang banyak digunakan pada perangkat elektronik portabel maupun sistem energi terbarukan. Baterai ini memiliki tegangan nominal sebesar 3,7 volt dengan kapasitas penyimpanan energi yang relatif tinggi dibandingkan dengan ukurannya. Menurut Nitta et al. (2021), baterai lithium-ion masih menjadi teknologi penyimpanan energi yang dominan karena memiliki densitas energi tinggi, efisiensi yang baik, serta kemampuan siklus pengisian ulang yang panjang.

Dalam penelitian ini, baterai 18650 berfungsi sebagai media penyimpanan energi yang berasal dari panel surya. Energi yang tersimpan kemudian digunakan untuk menyuplai kebutuhan daya ESP8266, sensor DHT22, sensor INA219, modul relay, dan kipas DC. Dengan adanya baterai, sistem dapat tetap beroperasi meskipun produksi energi dari panel surya mengalami penurunan akibat perubahan kondisi cuaca atau intensitas cahaya matahari. Sistem penyimpanan energi yang

baik juga mendukung pemantauan parameter listrik secara berkelanjutan, sehingga kondisi baterai dapat dipantau secara *realtime* untuk menjaga keandalan sistem (Setiawan et al., 2023).

c. Buck converter LM2596S

Merupakan konverter DC-DC yang berfungsi untuk menurunkan tegangan masukan menjadi tegangan keluaran yang lebih rendah dan stabil. Konverter ini bekerja menggunakan metode switching sehingga memiliki efisiensi yang lebih tinggi dibandingkan dengan regulator linier. Kemampuan tersebut menjadikan LM2596S banyak digunakan pada sistem elektronika yang membutuhkan kestabilan tegangan dengan konsumsi daya yang efisien.

Pada penelitian ini, LM2596S digunakan untuk mengatur tegangan keluaran dari baterai agar sesuai dengan kebutuhan operasional perangkat elektronik yang digunakan. Tegangan yang stabil sangat diperlukan untuk menjaga kinerja ESP8266 dan sensor agar dapat bekerja secara optimal. Selain meningkatkan kestabilan sistem, penggunaan *buck converter* juga membantu mengurangi kehilangan daya selama proses distribusi energi, sehingga efisiensi sistem dapat ditingkatkan.

d. Integrasi Modul Pengisian dan Penyimpanan Energi

Integrasi antara TP4056, baterai 18650, dan buck converter LM2596S membentuk suatu sistem manajemen energi yang mendukung operasional perangkat IoT secara mandiri. Energi listrik yang dihasilkan panel surya terlebih dahulu digunakan untuk mengisi baterai melalui TP4056, kemudian energi yang tersimpan disalurkan ke beban melalui buck converter untuk menghasilkan tegangan yang stabil sesuai kebutuhan sistem.

Melalui mekanisme tersebut, sistem *monitoring* suhu dan pengendalian kipas dapat tetap beroperasi secara berkelanjutan meskipun terjadi fluktuasi pada sumber energi utama. Selain meningkatkan kontinuitas suplai daya, integrasi sistem penyimpanan energi juga mendukung proses *monitoring* parameter tegangan dan arus secara *realtime* sehingga kondisi sumber daya dapat diketahui dengan lebih

akurat (Setiawan et al., 2023). Oleh karena itu, penerapan modul pengisian dan penyimpanan energi menjadi salah satu faktor penting dalam meningkatkan keandalan dan efisiensi sistem berbasis energi surya yang dikembangkan pada penelitian ini.

2.4 Mikrokontroler ESP8266

NodeMCU ESP8266 adalah modul *Wi-Fi* yang bertindak sebagai ekstensi mikrokontroler untuk membangun koneksi TCP/IP. Modul ini memiliki antarmuka USB ke serial dan dapat diprogram dengan LUA dan Arduino. Modul ini dilapisi dengan bahan logam untuk mengurangi interferensi dengan perangkat lain (Wantudi et al., 2023).

Peneliti lain menjelaskan bahwa meskipun ukurannya kecil (48 x 25,4 mm), perangkat ESP8266 memiliki beragam penerapan di bidang *Internet of Things*. Beberapa contohnya berkisar dari proyek-proyek kecil seperti sistem irigasi pintar, pengukuran konsumsi energi, atau alarm pendeteksi kebakaran (Santos et al., 2022).

Selain itu, perangkat ini memiliki memori *flash* sebesar 16 Mb dan antena *WiFi* terintegrasi yang beroperasi pada frekuensi 2,4 GHz. Selain itu, antena eksternal juga dapat dihubungkan ke perangkat ini untuk memperkuat sinyal. Dari segi pemrosesan, ESP8266 beroperasi pada frekuensi 80MHz atau 160 MHz, yang bisa di atur sesuai pilihan programmer dan perangkat ini dilengkapi dengan konektor baterai sehingga bisa beroperasi dalam mode mandiri (Santos et al., 2022).

Konfigurasi dan pemrograman perangkat ini dilakukan melalui Arduino IDE, perangkat ESP82266 dapat diprogram menggunakan lingkungan MicroPython atau NodeMCU (Santos et al., 2022). ESP8266 juga menggunakan koneksi internet, sehingga data yang dikirim langsung ke server, memastikan pembaruan secara *realtime* dan akses informasi yang segera.²⁹ Mikrokontroler ESP8266 NodeMCU, dengan modul *WiFi* terintegrasinya, memfasilitasi transmisi ini melalui protokol HTTP, menyediakan konektivitas yang stabil dan integrasi yang disederhanakan dengan server web dan pemrosesan data.(Gueye et al., 2025)

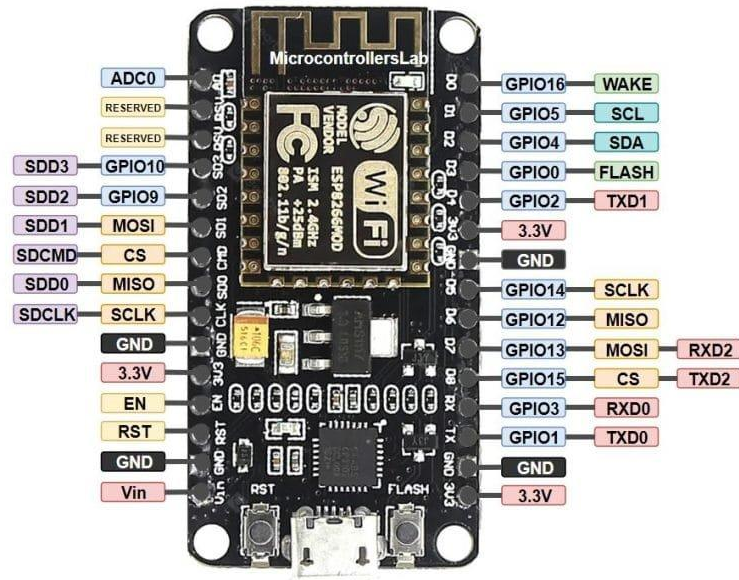
2.4.1 Arsitektur ESP8266

ESP8266 merupakan mikrokontroler yang dikembangkan oleh *Espressif Systems* untuk mendukung pengembangan perangkat *Internet of Things* (IoT). Mikrokontroler ini telah dilengkapi dengan modul *WiFi transceiver* sehingga mampu terhubung ke jaringan nirkabel maupun berfungsi sebagai *Access Point* (AP). Integrasi kemampuan pemrosesan data dan komunikasi nirkabel dalam satu chip menjadikan ESP8266 banyak digunakan sebagai pengendali utama pada berbagai aplikasi IoT. Selain itu, ESP8266 dapat diprogram menggunakan Arduino IDE, sehingga memudahkan proses pengembangan sistem dengan biaya yang relatif rendah (Kusuma et al., 2022).

2.4.2 Spesifikasi ESP8266

ESP8266 menggunakan prosesor Tensilica L106 32-bit dengan kecepatan kerja hingga 80 MHz. Mikrokontroler ini memiliki 17 pin, yang terdiri atas 11 pin Input/Output (I/O) dan 6 pin untuk komunikasi dengan memori *flash*. ESP8266 bekerja pada tegangan operasi 3,3 V serta mendukung beberapa antarmuka komunikasi, yaitu SPI, I²C, dan UART. Selain itu, perangkat ini juga dilengkapi dengan Analog-to-Digital Converter (ADC) 10-bit yang memungkinkan pembacaan data analog dari berbagai sensor, sehingga sesuai digunakan pada sistem *monitoring* dan kendali berbasis IoT (Kusuma et al., 2022).

2.4.3 Konfigurasi Pin ESP8266



Gambar 2. 8 ESP8266

Adapula berikut konfigurasi pin yang di pakai peneliti jabarkan menjadi seperti table berikut :

Table 2. 3 konfigurasi pin ESP8266

Pin ESP8266	Komponen Terhubung	Keterangan	Fungsi
VIN	Output LM2596S 5V	Supply utama NodeMCU	Menyuplai daya mikrokontroler
GND	Common Ground	Ground sistem	Referensi tegangan seluruh sistem
D1 (GPIO5)	SCL INA219	Jalur clock I2C	Sinkronisasi komunikasi sensor INA219
D2 (GPIO4)	SDA INA219 & DATA DHT22	Jalur data sensor	Komunikasi data sensor
D5 (GPIO14)	IN1 Relay 2 Channel	Kontrol relay	Mengaktifkan dan menonaktifkan relay
D7 (GPIO13)	Gate MOSFET IRLZ44N	Output PWM	Mengatur kecepatan kipas DC

Pin ESP8266	Komponen Terhubung	Keterangan	Fungsi
3V3	VCC DHT22 & INA219	Supply sensor	Menyuplai daya sensor
RX	Serial Monitor	Komunikasi serial	<i>Monitoring</i> dan debugging
TX	Serial Monitor	Komunikasi serial	Pengiriman data serial

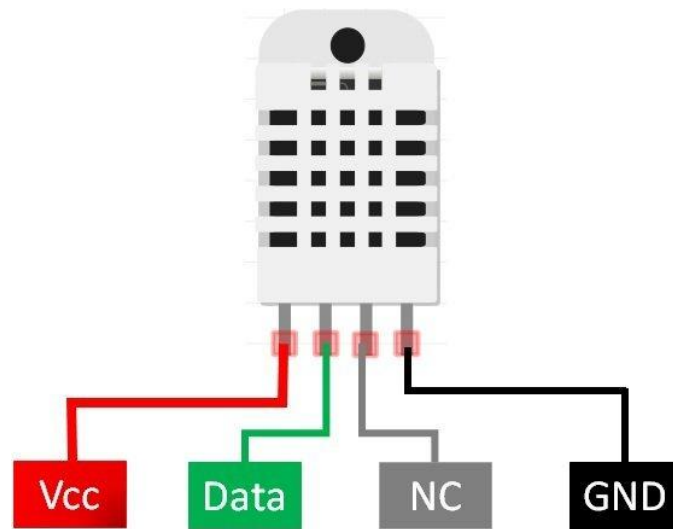
2.5 Sensor DHT22

Sensor DHT22 dilengkapi dengan konverter analog to digital bawaan dan dapat digunakan untuk mengukur suhu dan kelembapan. Ini dapat mendeteksi suhu dari -40 °C hingga 125 °C dengan *error* suhu $\pm 0,5^{\circ}\text{C}$, dan tingkat kesalahan untuk kelembapan adalah sekitar $\pm 2\%$ hingga $\pm 5\%$. Modul ini memiliki voltase 3-5V..(Hadi et al., 2024).

2.5.1 Prinsip Kerja DHT22

Sensor DHT22 digunakan sebagai sensor suhu untuk memantau kondisi temperatur pada sistem secara *realtime*. Sensor ini bekerja dengan membaca perubahan suhu kemudian mengirimkan data ke mikrokontroler untuk diproses. Pada penelitian ini, NodeMCU ESP8266 digunakan untuk mengontrol kipas DC melalui relay dan PWM berdasarkan nilai suhu yang terbaca oleh sensor DHT22. Apabila suhu meningkat melebihi batas yang telah ditentukan maka sistem akan mengaktifkan kipas secara otomatis, sedangkan ketika suhu menurun maka kecepatan kipas akan menyesuaikan menggunakan metode *Multi level control* sehingga sistem dapat bekerja lebih efektif dan efisien (Hariyono et al., 2024).

2.5.2 Konfigurasi Pin DHT22



Gambar 2. 9 DHT22

Sensor DHT22 memiliki empat pin, seperti DHT11, tetapi hanya tiga pin yang digunakan: VCC, GND, dan data. Ini karena sensor ini sering digunakan dalam berbagai situasi. Selain itu, kelebihan sensor DHT22 adalah akurasi pembacaan yang terjangkau dan memenuhi standar laboratorium (Hadi et al., 2024).

Table 2. 4 Konfigurasi pin DHT22

Pin DHT22	Koneksi NodeMCU ESP8266	Keterangan	Fungsi
VCC	3V3	Supply tegangan sensor	Menyuplai daya sensor
DATA	D2 (GPIO4)	Jalur data digital	Mengirim data suhu ke NodeMCU
GND	GND	Ground sistem	Referensi tegangan sensor

DHT22 digunakan untuk mengukur suhu dan kelembaban relatif. Sensor ini mampu berkomunikasi dengan mikrokontroler melalui port serial. Setelah dikalibrasi pabrik, sensor ini siap digunakan dan tidak memerlukan komponen tambahan untuk pengoperasiannya (Gueye et al., 2025).

2.6 Sensor INA219

INA219 adalah sensor arus dan tegangan yang dikembangkan oleh Texas Instruments yang dimaksudkan untuk mengukur arus dan tegangan dengan tingkat akurasi yang tinggi dalam sistem elektronik..(Mursidan et al., 2023).

2.6.1 Prinsip kerja INA219

Dalam sistem *monitoring* berbasis *Internet of Things* (IoT), INA219 banyak digunakan karena mampu melakukan pengukuran tegangan dan arus secara *realtime* dengan tingkat akurasi yang baik. Menurut (Mursidan et al., 2023) Sensor INA219 dapat dimanfaatkan untuk memantau kondisi baterai selama proses pengisian maupun penggunaan energi, sehingga mendukung proses *monitoring* daya secara berkelanjutan.

Beberapa penelitian menunjukkan bahwa sensor INA219 memiliki tingkat akurasi yang cukup tinggi untuk aplikasi *monitoring* energi. Penelitian Abidin et al. (2024) menunjukkan bahwa INA219 mampu mengukur tegangan, arus, dan daya pada sistem panel surya dengan tingkat kesalahan yang relatif kecil. Selain itu, Prasetyawati et al. (2023) menyatakan bahwa proses kalibrasi dapat meningkatkan akurasi sensor sehingga hasil pengukuran menjadi lebih mendekati alat ukur referensi.

2.6.2 Spesifikasi INA219

Table 2. 5 Spesifikasi INA219

No	Parameter	Spesifikasi
1.	Nama Sensor	INA219
2.	Produsen IC	Texas Instruments
3.	Tegangan Operasi	3,0 – 5,5 V DC
4.	Tegangan Bus Maksimum	0 – 26 V DC
5.	Rentang Pengukuran Arus	$\pm 3,2$ A (dengan resistor shunt $0,1 \Omega$ bawaan)
6.	Metode Pengukuran Arus	High-Side Current Sensing
7.	Resolusi ADC	12-bit
8.	Resolusi Tegangan Bus	4 mV/bit
9.	Resolusi Arus	Bergantung nilai kalibrasi (sekitar 0,1 mA)
10.	Akurasi Pengukuran	$\pm 1\%$ (Typical)
11.	Antarmuka Komunikasi	I ² C (Inter-Integrated Circuit)

No	Parameter	Spesifikasi
12.	Alamat I ² C Default	0x40
13.	Alamat I ² C yang Didukung	0x40–0x4F (bergantung konfigurasi pin A0 dan A1)
14.	Pengukuran	Tegangan, arus, dan daya

Sensor INA219 dapat mengukur arus hingga 3.2A dengan resolusi 0.8mA dan tegangan hingga 26V dengan resolusi 4mV. Ini juga memiliki proteksi arus berlebih, proteksi tegangan berlebih, dan deteksi tegangan input yang rendah. INA219 berkomunikasi dengan IC dan dapat digunakan untuk mengukur efisiensi konversi daya, memantau daya, dan mengukur arus panel surya. (Mursidan et al., 2023).

2.6.3 Konfigurasi Pin INA219

Table 2. 6 Konfigurasi pin INA219

Pin INA219	Koneksi NodeMCU ESP8266	Keterangan	Fungsi
VCC	3V3	Supply tegangan sensor	Menyuplai daya sensor
GND	GND	Ground sistem	Referensi tegangan sensor
SDA	D2 (GPIO4)	Jalur data I2C	Mengirim data sensor
SCL	D1 (GPIO5)	Jalur clock I2C	Sinkronisasi komunikasi I2C

Table 2. 7 Pin konfigurasi INA219

Terminal INA219	Koneksi	Keterangan
VIN+	Supply Positif	Input tegangan
VIN-	Beban Sistem	Output ke beban

Pada penelitian ini, sensor INA219 digunakan untuk mengukur tegangan dan arus pada sistem berbasis panel surya. Data hasil pengukuran kemudian dikirimkan ke ESP8266 untuk ditampilkan pada *website monitoring* secara *realtime* serta disimpan melalui fitur *Data logging*. Dengan demikian, sensor INA219 berperan sebagai perangkat utama dalam proses pemantauan kondisi energi pada sistem yang dikembangkan (Mursidan et al., 2023).

2.7 Sistem Monitoring

Sistem *monitoring* adalah sistem yang digunakan untuk memantau kondisi tertentu secara *realtime* menggunakan perangkat keras dan perangkat lunak yang saling terintegrasi. Pada penelitian ini, sistem *monitoring* digunakan untuk memantau suhu, arus, dan tegangan menggunakan sensor DHT22 dan INA219 yang terhubung dengan NodeMCU ESP8266. Data sensor diproses kemudian dikirimkan ke *website monitoring* melalui jaringan internet (Pratama et al., 2025).

Perancangan software pada penelitian ini dibagi menjadi dua bagian, yaitu pemrograman Arduino IDE pada alat *monitoring* dan perancangan *website monitoring*. Pemrograman Arduino IDE berfungsi untuk mengolah data sensor dan mengirimkannya ke *website monitoring*. Sementara itu, *website monitoring* digunakan untuk menampilkan data *monitoring* secara *realtime* dengan tampilan yang lebih menarik dan mudah diakses melalui internet (Pratama et al., 2025).

2.8 Sistem Kontrol

Sistem kontrol merupakan mekanisme yang digunakan untuk mengatur parameter sistem agar sesuai dengan nilai yang diinginkan. Sistem kontrol *modern* berbasis IoT memungkinkan proses *monitoring* dan pengendalian dilakukan secara bersamaan melalui komunikasi dua arah secara *realtime* (Paredes Cornejo et al., 2025). Parameter yang dikendalikan dalam penelitian ini adalah suhu lingkungan, yang diproses oleh mikrokontroler ESP8266 untuk mengatur kecepatan kipas sebagai aktuator. Sistem kontrol berperan dalam menjaga kestabilan serta memastikan sistem bekerja sesuai dengan kondisi yang telah ditentukan .

Kontrol *ON/OFF* merupakan metode pengendalian paling sederhana yang bekerja berdasarkan dua kondisi, yaitu aktif dan tidak aktif. Pada sistem ini, kipas akan menyala ketika suhu berada di bawah atau di atas batas tertentu dan akan mati ketika suhu telah mencapai nilai yang ditentukan . Metode ini mudah diterapkan karena tidak memerlukan perhitungan kompleks, namun menghasilkan respon yang kurang stabil akibat terjadinya perubahan berulang di sekitar nilai *set point*. Kondisi ini disebabkan oleh keluaran sistem yang hanya memiliki dua keadaan tanpa nilai perantara, sehingga kipas cenderung sering menyala dan mati dalam waktu singkat.

Multi level control merupakan metode pengendalian yang menggunakan lebih dari dua tingkat *output* dalam mengatur sistem. Pada penelitian ini, pengendalian kipas dilakukan dengan mengatur tingkat kecepatan menggunakan sinyal PWM yang dikendalikan oleh mikrokontroler melalui MOSFET. Nilai keluaran diatur dalam beberapa level berdasarkan rentang suhu yang terdeteksi. Pendekatan ini memungkinkan perubahan kecepatan kipas dilakukan secara bertahap sehingga menghasilkan respon yang lebih halus dan terkontrol.

Dibandingkan dengan kontrol *ON/OFF*, metode *Multi level control* memberikan kestabilan yang lebih baik karena mampu mengurangi perubahan berulang pada sistem. Pengaturan kecepatan kipas dilakukan secara bertingkat sehingga tidak terjadi perubahan drastis pada aktuator. Selain itu, penggunaan energi menjadi lebih efisien karena daya yang digunakan disesuaikan dengan kebutuhan aktual berdasarkan kondisi suhu yang terdeteksi.

2.9 Multi level control

Multi level control merupakan metode pengendalian yang mengatur keluaran sistem berdasarkan beberapa tingkat atau level kondisi yang telah ditentukan sebelumnya. Pada metode ini, setiap rentang nilai masukan akan menghasilkan aksi kontrol yang berbeda sesuai dengan level yang telah ditetapkan. Konsep tersebut banyak digunakan pada sistem pengendalian suhu karena mampu memberikan respons yang lebih adaptif dibandingkan metode *ON/OFF* konvensional yang hanya memiliki dua kondisi operasi, yaitu aktif dan tidak aktif. Penggunaan beberapa level pengendalian memungkinkan sistem bekerja lebih efisien serta mengurangi perubahan kondisi yang terjadi secara tiba-tiba pada aktuator.

Dalam sistem pengendalian suhu, *Multi level control* bekerja dengan membagi rentang suhu ke dalam beberapa kategori tertentu. Setiap kategori memiliki aksi pengendalian yang berbeda, misalnya kipas beroperasi pada kecepatan rendah, sedang, atau tinggi sesuai dengan peningkatan suhu yang terdeteksi sensor. Pendekatan ini memungkinkan proses pendinginan berlangsung secara bertahap sehingga suhu dapat dikendalikan dengan lebih stabil dibandingkan pengendalian dua kondisi (*binary control*).

Penerapan pengendalian bertingkat banyak digunakan pada sistem ventilasi, pendingin ruangan, dan berbagai aplikasi berbasis *Internet of Things* (IoT). Sunardi et al. (2023) menjelaskan bahwa pengendalian suhu yang terintegrasi dengan IoT memungkinkan proses *monitoring* dan kontrol dilakukan secara *realtime* sehingga meningkatkan efektivitas pengelolaan kondisi lingkungan. Selain itu, strategi pengendalian bertingkat juga berkontribusi dalam meningkatkan efisiensi energi karena aktuator hanya bekerja sesuai kebutuhan sistem.

Pada penelitian ini, metode *Multi level control* digunakan untuk mengendalikan kecepatan kipas berdasarkan nilai suhu yang diperoleh dari sensor DHT22. Sistem membagi suhu ke dalam beberapa level pengendalian yang telah ditentukan. Ketika suhu berada pada level normal, kipas akan bekerja pada kondisi minimum atau tidak aktif. Saat suhu meningkat hingga melewati batas tertentu, sistem akan mengaktifkan level pengendalian berikutnya dengan meningkatkan kecepatan kipas. Semakin tinggi suhu yang terdeteksi, semakin tinggi pula level kerja kipas yang diberikan oleh sistem. Dengan mekanisme tersebut, proses pendinginan dapat berlangsung secara lebih efektif dan efisien dibandingkan pengendalian *ON/OFF* biasa.

Penerapan *Multi level control* pada penelitian ini juga bertujuan untuk mengurangi konsumsi energi, memperpanjang umur operasional kipas, serta meningkatkan kestabilan suhu pada sistem *monitoring* berbasis panel surya. Oleh karena itu, metode ini dipilih sebagai solusi pengendalian yang sederhana, mudah diimplementasikan pada mikrokontroler ESP8266, dan sesuai dengan kebutuhan sistem yang dikembangkan.

2.10 Data logging

Data logging merupakan proses penyimpanan data hasil pengukuran sistem secara berkala pada media penyimpanan tertentu. Data yang dicatat mencakup parameter suhu, tegangan, dan arus yang diperoleh dari sensor dalam sistem *monitoring* berbasis *Internet of Things* (Putra & Fajar, 2023)

Proses *Data logging* dilakukan dengan mengumpulkan data dari sensor yang terhubung dengan mikrokontroler, kemudian menyimpannya dalam format terstruktur. Pada penelitian ini, penyimpanan dilakukan secara lokal menggunakan modul MicroSD serta ditampilkan secara *realtime* melalui *platform* IoT. Mekanisme ini memungkinkan data tersimpan secara kontinu dan dapat diakses untuk keperluan pemantauan sistem (Putra & Fajar, 2023)

Data logging digunakan untuk menganalisis kinerja sistem melalui pemanfaatan data historis. Data yang tersimpan digunakan untuk mengidentifikasi pola perubahan suhu serta respon sistem dalam mengendalikan kipas. Selain itu, data juga dapat digunakan untuk mendeteksi kondisi yang tidak normal selama sistem beroperasi

Penerapan *Data logging* mendukung peningkatan efisiensi dan keandalan sistem. Analisis data memungkinkan penyesuaian pengendalian berdasarkan kondisi aktual sehingga sistem dapat bekerja lebih stabil dan penggunaan energi menjadi lebih optimal

2.11 Notifikasi Sistem

Notifikasi sistem merupakan mekanisme pemberian peringatan terhadap kondisi tertentu pada sistem *monitoring* berbasis *Internet of Things*. Peringatan diberikan ketika parameter yang diamati, seperti suhu atau kondisi listrik, melewati batas nilai yang telah ditentukan sebagai ambang aman sistem (Alfian, 2023).

Mekanisme notifikasi dilakukan dengan memantau perubahan nilai parameter secara berkelanjutan melalui sensor yang terintegrasi dengan mikrokontroler. Ketika kondisi tidak normal terdeteksi, sistem akan mengirimkan peringatan secara otomatis dalam bentuk notifikasi melalui *platform* IoT. Notifikasi juga dapat dikirim melalui berbagai *platform* komunikasi seperti aplikasi mobile atau pesan instan untuk meningkatkan respons pengguna terhadap kondisi sistem (Pane et al., 2024). Pada penelitian ini, notifikasi dikirimkan melalui aplikasi *Blynk* dalam bentuk alert sehingga informasi dapat diterima pengguna secara langsung (Alfian, 2023).

Notifikasi sistem berperan dalam mendukung *monitoring realtime* karena memungkinkan pengguna memperoleh informasi kondisi sistem tanpa melakukan pengecekan manual. Mekanisme ini meningkatkan efisiensi pengelolaan sistem serta mempercepat proses pengambilan keputusan ketika terjadi perubahan kondisi operasional (Alfian, 2023).

Selain itu, notifikasi memungkinkan tindakan korektif dilakukan secara cepat ketika terjadi penyimpangan dari kondisi normal, seperti suhu yang melebihi batas aman. Deteksi dini terhadap kondisi tersebut dapat mengurangi risiko kerusakan komponen serta menjaga stabilitas sistem selama beroperasi (Alfian, 2023).

Penerapan notifikasi juga dapat dikombinasikan dengan fitur pengamanan sistem seperti auto shutdown, sehingga sistem tidak hanya memberikan peringatan, tetapi juga mampu melakukan tindakan otomatis untuk mencegah kerusakan yang lebih lanjut. Pendekatan ini meningkatkan keandalan serta kontinuitas operasi sistem secara keseluruhan.

2.12 Fail safe System

Fail safe system merupakan mekanisme pengamanan yang dirancang untuk memastikan sistem tetap berada pada kondisi aman ketika terjadi gangguan, kesalahan, atau kondisi tidak normal. Dalam sistem otomatis *modern*, konsep ini menjadi bagian dari strategi keselamatan yang bertujuan untuk mencegah kerusakan perangkat maupun risiko terhadap lingkungan sekitar. Pendekatan *fail safe* tidak hanya menghentikan operasi sistem, tetapi juga mengarahkan sistem menuju kondisi yang paling aman secara otomatis . (Gymnasdiar et al., 2026)

Pada implementasi sistem berbasis *Internet of Things* (IoT), mekanisme *fail safe* memiliki peran penting dalam menjaga keandalan operasional. Sistem IoT memiliki ketergantungan terhadap jaringan dan sensor, sehingga potensi gangguan seperti kegagalan komunikasi atau kesalahan pembacaan data dapat terjadi. Oleh karena itu, sistem perlu dirancang agar tetap mampu beroperasi atau beralih ke mode aman ketika terjadi gangguan tersebut. Penelitian menunjukkan bahwa penerapan mode *fail safe* memungkinkan sistem tetap berjalan dengan tingkat

keandalan yang tinggi meskipun terjadi gangguan konektivitas .(Irham Fadilah & Endah Fitriani, 2025)

Konsep *fail safe* juga diterapkan melalui mekanisme respons otomatis terhadap kondisi berbahaya. Pada sistem otomatis, sensor digunakan untuk mendeteksi kondisi abnormal, kemudian sistem akan memberikan respon berupa penghentian atau perubahan operasi. Sebagai contoh, pada sistem gerbang otomatis, ketika terdeteksi adanya objek yang menghalangi, sistem tidak hanya berhenti tetapi juga melakukan pembalikan arah (*auto reverse*) untuk menghindari potensi bahaya(Gymnasdiar et al., 2026) . Pendekatan ini menunjukkan bahwa *fail safe* tidak hanya bersifat pasif, tetapi juga adaptif terhadap kondisi yang terjadi.

Selain itu, *fail safe system* berkaitan erat dengan konsep resiliensi teknologi, yaitu kemampuan sistem untuk tetap berfungsi atau pulih dari gangguan. Sistem yang memiliki mekanisme *fail safe* mampu mempertahankan stabilitas operasional dan meminimalkan risiko kerusakan akibat kondisi yang tidak terduga. Hal ini menjadikan *fail safe* sebagai komponen penting dalam pengembangan sistem kontrol cerdas, terutama pada sistem yang bekerja secara otomatis dan *realtime*.

Dalam konteks penelitian ini, konsep *fail safe* diterapkan melalui mekanisme auto shutdown yang bekerja ketika parameter sistem melewati batas aman, seperti arus atau tegangan yang terlalu tinggi. Pendekatan ini memastikan bahwa sistem tidak beroperasi dalam kondisi berbahaya, sehingga dapat meningkatkan keamanan, keandalan, serta umur pakai perangkat secara keseluruhan.

Fail safe system dapat dibedakan dari sistem kontrol konvensional yang tidak memiliki mekanisme pengamanan otomatis. Pada sistem tanpa *fail safe*, kegagalan komponen atau kesalahan operasi dapat menyebabkan sistem tetap berjalan dalam kondisi berbahaya. Sebaliknya, sistem yang menerapkan *fail safe* akan secara otomatis menghentikan atau menyesuaikan operasi untuk mencapai kondisi aman. Perbedaan ini menunjukkan bahwa *fail safe* merupakan elemen penting dalam sistem pengaman *modern* yang menekankan aspek keamanan dan keandalan operasional.

2.13 Website monitoring

Website monitoring merupakan antarmuka berbasis web yang digunakan untuk menampilkan, mengelola, dan memantau data yang diperoleh dari perangkat *Internet of Things* (IoT) secara *realtime*. Dalam sistem IoT, *website monitoring* berfungsi sebagai media visualisasi data sehingga pengguna dapat mengakses informasi sistem melalui jaringan internet tanpa harus berada di lokasi perangkat. Melalui *website monitoring*, data yang diperoleh dari sensor dapat disajikan dalam bentuk tabel, grafik, maupun indikator kondisi sistem sehingga memudahkan proses pengawasan dan pengambilan keputusan.

Perkembangan teknologi web memungkinkan *website monitoring* tidak hanya berfungsi sebagai media tampilan data, tetapi juga sebagai pusat pengelolaan sistem. Data yang dikirim oleh perangkat IoT akan disimpan pada basis data server, kemudian ditampilkan secara dinamis melalui *Dashboard* berbasis web. Menurut Calderon et al. (2024), *platform monitoring* modern berperan penting dalam mengolah, menyimpan, dan menyajikan data IoT melalui antarmuka yang mudah digunakan sehingga pengguna dapat melakukan pemantauan secara efektif.

Salah satu komponen utama pada *website monitoring* adalah *Dashboard*. *Dashboard* merupakan tampilan visual yang menyajikan informasi penting secara ringkas dan terstruktur sehingga pengguna dapat memahami kondisi sistem dengan cepat. *Dashboard* umumnya menampilkan data *realtime*, grafik historis, status perangkat, serta notifikasi apabila terjadi kondisi tertentu. Penelitian Joanda et al. (2025) menunjukkan bahwa *Dashboard monitoring* berbasis web mampu meningkatkan efektivitas pemantauan sistem IoT karena pengguna dapat memperoleh informasi secara cepat, akurat, dan mudah dipahami.

Pada sistem *monitoring* berbasis web, fitur *Data logging* menjadi bagian yang sangat penting. Data yang diperoleh dari sensor tidak hanya ditampilkan secara langsung, tetapi juga disimpan ke dalam basis data sehingga dapat digunakan untuk analisis historis dan evaluasi kinerja sistem. Selain itu, *website monitoring* juga dapat dilengkapi dengan fitur notifikasi untuk memberikan peringatan kepada pengguna apabila terjadi kondisi yang melebihi batas yang telah ditentukan.

Keberadaan fitur tersebut memungkinkan proses pengawasan dilakukan secara lebih efektif dan responsif terhadap perubahan kondisi sistem.

Dalam penelitian ini, *website monitoring* dikembangkan menggunakan React JS sebagai *framework* frontend, Vite sebagai build tool, Tailwind CSS untuk perancangan antarmuka pengguna, serta Recharts untuk visualisasi grafik data *realtime*. *Website* terhubung dengan ESP8266 melalui REST API dan menggunakan basis data MySQL sebagai media penyimpanan data *monitoring*. *Website* ini berfungsi untuk menampilkan informasi suhu, tegangan, arus, status kipas, grafik historis, *Data logging*, serta notifikasi kondisi kritis secara *realtime*. Dengan adanya *website monitoring*, pengguna dapat melakukan pemantauan dan evaluasi kinerja sistem *monitoring* suhu dan pengendalian kipas berbasis panel surya secara lebih mudah, cepat, dan efisien.

2.14 REST API

Representational State Transfer *Application* Programming Interface (REST API) merupakan suatu arsitektur layanan web yang digunakan untuk memungkinkan komunikasi dan pertukaran data antara berbagai aplikasi melalui jaringan internet. REST API bekerja dengan memanfaatkan protokol Hypertext Transfer Protocol (HTTP) untuk mengirim, menerima, memperbarui, maupun menghapus data dari suatu server. Menurut Gupta dan Kumar (2023), REST API menjadi salah satu metode komunikasi yang paling banyak digunakan pada sistem *Internet of Things* (IoT) karena memiliki struktur yang sederhana, ringan, mudah diimplementasikan, serta kompatibel dengan berbagai *platform* dan bahasa pemrograman.

Dalam arsitektur REST API, metode HTTP seperti *GET*, *POST*, *PUT*, dan *DELETE* digunakan untuk berkomunikasi dengan data. Metode *GET* mengambil data dari server, *POST* mengirim data baru ke server, *PUT* memperbarui data yang sudah ada, dan *DELETE* menghapus data dari basis data. Data yang dipertukarkan umumnya menggunakan format *JavaScript Object Notation* (JSON) karena memiliki ukuran yang relatif kecil dan mudah diproses oleh berbagai perangkat maupun aplikasi berbasis web.

Dalam sistem berbasis IoT, REST API berperan sebagai penghubung antara perangkat sensor, server, *Database*, dan antarmuka pengguna. Data hasil pengukuran sensor yang diperoleh mikrokontroler akan dikirim ke server melalui REST API, kemudian disimpan ke dalam basis data untuk selanjutnya ditampilkan pada aplikasi *monitoring*. Menurut Fauzi dan Amarudin (2025), penggunaan REST API pada sistem *monitoring* berbasis IoT mampu meningkatkan efisiensi komunikasi data serta mempermudah integrasi antara perangkat keras dan perangkat lunak yang digunakan dalam sistem.

Keunggulan REST API terletak pada kemampuannya untuk mendukung komunikasi data secara *realtime*, fleksibilitas integrasi, serta kemudahan pengembangan sistem yang bersifat modular. Selain itu, REST API memungkinkan pengembang untuk memisahkan proses pengelolaan data pada sisi server (backend) dengan tampilan antarmuka pengguna (frontend), sehingga pengembangan dan pemeliharaan sistem dapat dilakukan dengan lebih terstruktur dan efisien.

Pada penelitian ini, REST API digunakan sebagai media komunikasi antara mikrokontroler ESP8266 dengan server *website monitoring*. Data suhu dari sensor DHT22 serta data tegangan dan arus dari sensor INA219 dikirimkan oleh ESP8266 ke server menggunakan metode HTTP POST. Selanjutnya, data tersebut disimpan ke dalam *Database MySQL* melalui mekanisme *Data logging* dan ditampilkan pada *website monitoring* berbasis React JS secara *realtime*. Selain mendukung proses *monitoring*, REST API juga digunakan untuk mengirimkan informasi status pengendalian kipas yang bekerja menggunakan metode *ON/OFF* dan *Multi level control*. Dengan demikian, REST API menjadi komponen penting yang menghubungkan seluruh bagian sistem sehingga proses *monitoring*, penyimpanan data, dan pengendalian dapat berjalan secara terintegrasi.

2.15 *Database MySQL*

MySQL adalah salah satu sistem manajemen basis data relasional (RDBMS), yang banyak digunakan dalam pengembangan aplikasi berbasis web dan *Internet of Things* (IoT). Fungsi MySQL termasuk menyimpan, mengelola, dan mengakses data secara terstruktur menggunakan bahasa kueri SQL. Menurut Widodo et al.

(2023), MySQL menjadi salah satu basis data yang populer karena memiliki kinerja yang baik, mudah diimplementasikan, bersifat *open source*, serta mampu menangani pengelolaan data dalam jumlah besar dengan efisien.

Pada sistem berbasis IoT, *Database* berperan sebagai media penyimpanan data hasil pengukuran sensor yang dikirimkan oleh perangkat melalui jaringan internet. Data yang tersimpan dapat digunakan untuk kebutuhan *monitoring*, analisis historis, pelaporan, maupun pengambilan keputusan. Penggunaan *Database* memungkinkan informasi yang diperoleh dari sensor tidak hanya ditampilkan secara *realtime*, tetapi juga dapat diakses kembali untuk melihat perubahan kondisi sistem dalam periode tertentu (Mulyawan et al., 2023).

MySQL mendukung berbagai operasi pengelolaan data seperti penyimpanan (insert), pembacaan (select), pembaruan (update), dan penghapusan (delete). Kemampuan tersebut menjadikan MySQL sangat sesuai untuk mendukung fitur *Data logging* pada sistem *monitoring*. Selain itu, MySQL dapat diintegrasikan dengan berbagai teknologi web melalui REST API sehingga proses pertukaran data antara perangkat IoT, server, dan antarmuka pengguna dapat berlangsung secara efektif dan terstruktur (Fauzi & Amarudin, 2025).

Dalam penelitian ini, MySQL digunakan sebagai media penyimpanan data *monitoring* yang diperoleh dari sensor DHT22 dan INA219. Data suhu, tegangan, arus, waktu pengukuran, serta status pengendalian kipas disimpan secara otomatis ke dalam *Database* melalui mekanisme *Data logging*. Selanjutnya, data tersebut diakses oleh *website monitoring* berbasis React JS untuk ditampilkan dalam bentuk tabel maupun grafik *realtime* menggunakan Recharts. Dengan demikian, penggunaan MySQL mendukung proses penyimpanan data historis, *monitoring* berkelanjutan, serta evaluasi kinerja sistem *monitoring* suhu dan pengendalian kipas berbasis panel surya yang dikembangkan pada penelitian ini.

2.16 React JS

React JS merupakan pustaka JavaScript (JavaScript Library) yang dikembangkan oleh Meta (Facebook) untuk membangun antarmuka pengguna

(User Interface/UI) pada aplikasi web. React JS menggunakan pendekatan berbasis komponen (component-based architecture), sehingga setiap bagian antarmuka dapat dikembangkan, dikelola, dan digunakan kembali secara terpisah. Pendekatan ini memungkinkan pengembangan aplikasi menjadi lebih terstruktur, efisien, dan mudah dipelihara dibandingkan metode pengembangan antarmuka web konvensional (Kumar & Singh, 2023).

Salah satu keunggulan utama React JS adalah penggunaan Virtual Document Object Model (Virtual DOM). Teknologi ini memungkinkan React melakukan pembaruan tampilan hanya pada bagian yang mengalami perubahan tanpa harus memuat ulang seluruh halaman web. Menurut Hussain et al. (2024), mekanisme Virtual DOM mampu meningkatkan performa aplikasi web, khususnya pada sistem yang memerlukan pembaruan data secara *realtime*. Oleh karena itu, React JS banyak digunakan dalam pengembangan *Dashboard monitoring*, sistem informasi, dan aplikasi berbasis *Internet of Things* (IoT).

Dalam sistem *monitoring* berbasis IoT, React JS berperan sebagai antarmuka yang menampilkan data hasil pengukuran sensor dalam bentuk tabel, indikator status, maupun grafik interaktif. Integrasi React JS dengan REST API memungkinkan data yang tersimpan pada server dapat ditampilkan secara dinamis kepada pengguna. Selain itu, React JS mendukung pengembangan Single Page Application (SPA), yaitu aplikasi web yang mampu menampilkan perubahan data tanpa melakukan proses reload halaman secara keseluruhan sehingga pengalaman pengguna menjadi lebih responsif dan interaktif (Sharma & Verma, 2022).

Pada penelitian ini, React JS digunakan sebagai *framework frontend* dalam pengembangan *website monitoring* suhu dan parameter listrik berbasis IoT. React JS berfungsi untuk menampilkan data suhu dari sensor DHT22 serta data tegangan dan arus dari sensor INA219 yang diperoleh melalui REST API. Data tersebut disajikan dalam bentuk *Dashboard monitoring*, grafik *realtime* menggunakan Recharts, status pengendalian kipas, serta informasi notifikasi sistem. Penggunaan React JS dipilih karena memiliki performa yang baik, mendukung pembaruan data

secara *realtime*, dan mampu menghasilkan antarmuka yang responsif sehingga sesuai dengan kebutuhan sistem *monitoring* yang dikembangkan.

2.17 Tailwind CSS

Tailwind CSS merupakan *framework* CSS yang menggunakan pendekatan *utility-first*, yaitu menyediakan kumpulan kelas utilitas yang dapat digunakan secara langsung untuk membangun tampilan antarmuka tanpa harus menuliskan banyak kode CSS secara manual. Pendekatan ini memungkinkan pengembang membuat desain yang fleksibel, konsisten, dan responsif dengan memanfaatkan kelas-kelas yang telah disediakan oleh *framework*. Tailwind CSS menjadi salah satu *framework* frontend yang banyak digunakan dalam pengembangan aplikasi web modern karena mampu mempercepat proses desain antarmuka serta meningkatkan produktivitas pengembang.

Berbeda dengan *framework* CSS konvensional yang menyediakan komponen dengan gaya bawaan tertentu, Tailwind CSS memberikan kebebasan kepada pengembang untuk membangun antarmuka sesuai kebutuhan melalui kombinasi kelas utilitas. Menurut Vargas Zermeño (2024), pendekatan ini membantu menjaga konsistensi desain dalam proyek berskala besar serta memudahkan kolaborasi antar pengembang karena aturan tampilan dapat diterapkan secara lebih terstruktur.

Selain fleksibel, Tailwind CSS juga mendukung pengembangan antarmuka yang responsif pada berbagai ukuran layar. *Framework* ini menyediakan utilitas untuk mengatur tata letak, warna, tipografi, jarak, dan berbagai elemen visual lainnya secara efisien. Penelitian Nandan et al. (2024) menunjukkan bahwa Tailwind CSS memiliki keunggulan dalam aspek fleksibilitas, kemudahan kustomisasi, serta performa yang baik ketika digunakan bersama mekanisme optimasi yang menghilangkan kelas CSS yang tidak digunakan.

Tailwind CSS banyak digunakan dalam pengembangan *Dashboard monitoring*, sistem informasi, dan aplikasi berbasis *Internet of Things* (IoT) karena mampu menghasilkan antarmuka yang modern dan mudah dikembangkan. Mardiana et al. (2024) menjelaskan bahwa penggunaan Tailwind CSS dapat

membantu menghasilkan tampilan aplikasi yang lebih responsif dan mempermudah proses pengembangan antarmuka pengguna pada aplikasi berbasis web.

Pada penelitian ini, Tailwind CSS digunakan sebagai *framework* untuk merancang antarmuka *website monitoring* yang terintegrasi dengan React JS. *Framework* ini digunakan untuk membangun *Dashboard monitoring* suhu, tegangan, arus, grafik *realtime*, *Data logging*, dan notifikasi sistem secara responsif. Penggunaan Tailwind CSS mendukung pengembangan *website monitoring* yang memiliki tampilan modern, mudah digunakan, serta mampu menyesuaikan tampilan pada berbagai perangkat, baik komputer maupun perangkat bergerak.

2.18 Recharts

Recharts merupakan pustaka (library) visualisasi data berbasis React JS yang digunakan untuk menampilkan informasi dalam bentuk grafik secara interaktif. Recharts dibangun menggunakan komponen React dan memanfaatkan teknologi SVG (Scalable Vector Graphics) sehingga mampu menghasilkan tampilan grafik yang responsif, dinamis, dan mudah diintegrasikan ke dalam aplikasi berbasis web. Menurut Kumar dan Patel (2023), penggunaan pustaka visualisasi data pada aplikasi web modern berperan penting dalam meningkatkan pemahaman pengguna terhadap data yang ditampilkan melalui representasi visual yang lebih informatif dibandingkan penyajian data dalam bentuk tabel.

Dalam sistem *monitoring* berbasis *Internet of Things* (IoT), visualisasi data menjadi salah satu komponen penting karena memungkinkan pengguna mengamati perubahan parameter sistem secara lebih mudah. Grafik yang ditampilkan secara *realtime* dapat membantu pengguna memahami pola perubahan data, mengidentifikasi kondisi abnormal, serta melakukan evaluasi terhadap kinerja sistem. Oleh karena itu, penggunaan library visualisasi seperti Recharts banyak diterapkan pada *Dashboard monitoring* berbasis React JS karena memiliki kemampuan integrasi yang baik dengan data yang diperoleh melalui REST API (Mardiana et al., 2024).

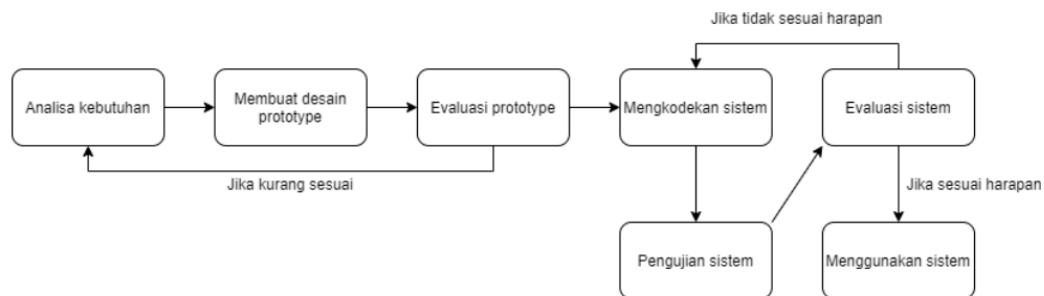
Recharts menyediakan berbagai jenis grafik seperti Line Chart, Bar Chart, Area Chart, Pie Chart, dan Composed Chart yang dapat digunakan sesuai kebutuhan sistem. Selain itu, Recharts mendukung pembaruan data secara dinamis sehingga grafik dapat diperbarui secara otomatis ketika data baru diterima dari server. Kemampuan tersebut sangat sesuai untuk aplikasi *monitoring realtime* yang memerlukan penyajian data secara kontinu dan interaktif. Menurut Sharma dan Verma (2022), visualisasi data *realtime* pada aplikasi web dapat meningkatkan efektivitas proses *monitoring* serta mempermudah pengguna dalam melakukan analisis terhadap kondisi sistem yang diamati.

Pada penelitian ini, Recharts digunakan sebagai library visualisasi data pada *website monitoring* yang dibangun menggunakan React JS. Data suhu yang diperoleh dari sensor DHT22 serta data tegangan dan arus dari sensor INA219 ditampilkan dalam bentuk grafik *realtime* sehingga pengguna dapat memantau perubahan kondisi sistem secara lebih mudah. Grafik yang disajikan juga terhubung dengan *Data logging* yang tersimpan pada *Database MySQL* melalui REST API. Dengan demikian, penggunaan Recharts mendukung penyajian informasi yang lebih informatif, interaktif, dan mudah dipahami dalam sistem *monitoring* suhu dan pengendalian kipas berbasis panel surya yang dikembangkan pada penelitian ini.

2.19 Metode Perancangan Perangkat Lunak

Dalam proses pengembangan sistem untuk memenuhi kebutuhan pengguna dan tujuan penelitian, pendekatan perancangan perangkat lunak digunakan. Pada penelitian ini, metode prototyping dipilih karena memungkinkan pengembangan sistem dilakukan secara bertahap dengan membuat prototipe awal yang dapat diuji dan dievaluasi sebelum menjadi sistem yang lengkap.

2.19.1 Metode *prototype*



Gambar 2. 10 Gambar Metode *prototype*

Sumber: (Kustanto et al., 2024)

Model *prototyping* merupakan metode pengembangan sistem yang digunakan untuk memperoleh informasi mengenai kebutuhan pengguna dengan menyajikan bentuk awal perangkat lunak yang dapat dilihat dan dievaluasi. Menurut Pricillia dan Zulfachmi (2021) dalam Kustanto et al. (2024), model *prototyping* merupakan metode yang digunakan untuk memperoleh informasi mengenai kebutuhan pengguna dengan berfokus pada penyajian elemen perangkat lunak yang dapat dilihat oleh pengguna sehingga membantu mengidentifikasi kebutuhan dalam pengembangan sistem.

Menurut Supiyandi et al. (2022) dalam Kustanto et al. (2024), metode *prototyping* menghasilkan *prototype* sistem yang memungkinkan pengembang melakukan interaksi dan evaluasi selama proses pengembangan. *prototype* dibangun berdasarkan kebutuhan awal yang telah ditentukan, kemudian dapat mengalami perubahan, penambahan, maupun pengurangan sesuai dengan hasil evaluasi dan pengujian.

Menurut Setiawan (2021) dalam Kustanto et al. (2024), tahapan penerapan metode *prototyping* meliputi analisis kebutuhan, desain *prototype*, evaluasi *prototype*, analisis dan perbaikan, iterasi, pengkodean sistem, serta implementasi sistem. Tahapan tersebut dilakukan secara bertahap dan berulang sehingga *prototype* dapat disesuaikan dengan kebutuhan sistem yang telah ditentukan.

Pada penelitian ini, metode *prototyping* diterapkan dalam rancang bangun sistem monitoring suhu dan pengendalian kipas berbasis *Internet of Things* (IoT) dengan *multi level control*, *data logging*, dan integrasi panel surya sebagai sumber energi. Metode ini digunakan untuk mengembangkan perangkat keras dan perangkat lunak secara bertahap melalui proses perancangan, evaluasi, perbaikan, pengujian, dan implementasi. Tahapan *prototyping* yang diterapkan dalam penelitian ini meliputi analisis kebutuhan, desain *prototype*, evaluasi *prototype*, analisis dan perbaikan, iterasi, pengkodean sistem, pengujian dan evaluasi sistem, serta implementasi sistem.

1. Analisis Kebutuhan

Tahap analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan perangkat keras dan perangkat lunak berdasarkan tujuan serta fungsi sistem yang akan dikembangkan. Analisis kebutuhan menjadi dasar dalam menentukan komponen dan teknologi yang digunakan dalam pembangunan sistem monitoring dan pengendalian kipas.

Kebutuhan perangkat keras dalam penelitian ini meliputi ESP8266 sebagai mikrokontroler, sensor DHT22 untuk mengukur suhu, sensor INA219 untuk mengukur tegangan dan arus listrik, MOSFET IRLZ44N sebagai komponen pengendali kipas, kipas DC sebagai aktuator, serta komponen pendukung sistem panel surya dan penyimpanan energi. Sementara itu, kebutuhan perangkat lunak meliputi Arduino IDE untuk pemrograman ESP8266, React JS untuk membangun antarmuka *website* monitoring, REST API sebagai media komunikasi antara ESP8266, *website*, dan database, serta MySQL sebagai basis data untuk menyimpan hasil monitoring.

Sistem dirancang memiliki tiga tingkat pengendalian kipas berdasarkan nilai suhu, yaitu Level 1 dengan PWM 30%, Level 2 dengan PWM 60%, dan Level 3 dengan PWM 100%. Selain itu, sistem memiliki fungsi monitoring suhu, tegangan, dan arus secara real-time serta *data logging* untuk menyimpan data hasil

pengukuran. Hasil analisis kebutuhan tersebut digunakan sebagai dasar dalam merancang *prototype* sistem.

2. Desain *prototype*

Tahap desain *prototype* dilakukan dengan membuat rancangan awal sistem berdasarkan hasil analisis kebutuhan. Desain *prototype* mencakup rancangan perangkat keras, hubungan antar-komponen, alur kerja sistem, serta rancangan perangkat lunak dan komunikasi data.

ESP8266 dirancang sebagai pusat pengendalian yang menerima data suhu dari sensor DHT22 dan data tegangan serta arus dari sensor INA219. Berdasarkan nilai suhu yang diperoleh, ESP8266 mengatur kecepatan kipas melalui MOSFET IRLZ44N menggunakan tiga tingkat PWM. Data hasil pembacaan sensor kemudian dikirimkan melalui jaringan ke REST API untuk diteruskan dan disimpan pada database MySQL.

Pada sisi perangkat lunak, *website* dirancang menggunakan React JS sebagai antarmuka monitoring. *Website* menampilkan data suhu, tegangan, dan arus serta menyediakan visualisasi data dan riwayat hasil pengukuran. MySQL digunakan sebagai media penyimpanan data hasil monitoring, sedangkan REST API digunakan untuk menghubungkan perangkat ESP8266 dengan sistem *website* dan database.

Pada bagian sumber energi, panel surya dirancang sebagai sumber energi *prototype* yang terintegrasi dengan rangkaian penyimpanan dan pengaturan tegangan untuk mendukung kebutuhan daya sistem. Hasil dari tahap desain berupa rancangan awal perangkat keras dan perangkat lunak yang digunakan sebagai acuan dalam pembuatan *prototype*.

3. Evaluasi *prototype*

Tahap evaluasi *prototype* dilakukan untuk mengetahui kesesuaian rancangan dan *prototype* awal dengan kebutuhan sistem yang telah ditentukan. Evaluasi

dilakukan terhadap fungsi utama perangkat keras dan perangkat lunak sebelum sistem dikembangkan lebih lanjut.

Pada penelitian ini, evaluasi *prototype* meliputi pembacaan suhu menggunakan DHT22, pengukuran tegangan dan arus menggunakan INA219, pengendalian kipas pada tiga tingkat PWM, komunikasi data antara ESP8266 dan REST API, penyimpanan data pada MySQL, serta tampilan data monitoring pada *website*.

Evaluasi dilakukan untuk mengetahui apakah setiap bagian *prototype* telah menjalankan fungsi yang dirancang. Apabila ditemukan ketidaksesuaian atau kendala pada *prototype*, hasil evaluasi digunakan sebagai dasar untuk menentukan perbaikan pada tahap berikutnya.

4. Analisis dan Perbaikan

Tahap analisis dan perbaikan dilakukan berdasarkan hasil evaluasi *prototype*. Setiap kekurangan atau ketidaksesuaian yang ditemukan dianalisis untuk menentukan bagian sistem yang perlu diperbaiki.

Perbaikan dapat dilakukan pada perangkat keras maupun perangkat lunak, seperti konfigurasi sensor, rangkaian pengendalian kipas, program pembacaan sensor, pengaturan PWM, komunikasi ESP8266 dengan REST API, proses penyimpanan data pada MySQL, maupun tampilan *website* monitoring.

Tahap ini dilakukan agar *prototype* semakin sesuai dengan kebutuhan sistem dan dapat menjalankan fungsi monitoring serta pengendalian sesuai dengan rancangan penelitian. Hasil perbaikan kemudian digunakan sebagai dasar untuk melakukan iterasi terhadap *prototype*.

5. Iterasi

Tahap iterasi dilakukan dengan mengulangi proses evaluasi, analisis, dan perbaikan terhadap *prototype* apabila masih ditemukan ketidaksesuaian. Proses tersebut memungkinkan setiap bagian sistem diperbaiki dan diuji kembali sebelum dilanjutkan ke tahap pengembangan berikutnya.

Pada penelitian ini, iterasi dapat dilakukan pada bagian pembacaan sensor DHT22 dan INA219, pengendalian kipas *multi-level*, komunikasi data melalui REST API, penyimpanan data pada MySQL, serta tampilan monitoring pada *website*. Iterasi dilakukan sampai *prototype* dapat menjalankan fungsi utama sesuai dengan kebutuhan sistem yang telah ditentukan.

6. Pengkodean Sistem

Setelah rancangan *prototype* dinyatakan sesuai dengan kebutuhan, dilakukan pengkodean sistem untuk mengimplementasikan rancangan ke dalam program yang dapat dijalankan.

Pada sisi mikrokontroler, program dikembangkan menggunakan Arduino IDE untuk ESP8266. Program mencakup pembacaan suhu menggunakan DHT22, pembacaan tegangan dan arus menggunakan INA219, pengendalian kipas menggunakan tiga tingkat PWM, serta pengiriman data hasil pengukuran ke REST API.

Pada sisi *website*, React JS digunakan untuk membangun antarmuka monitoring yang menampilkan data hasil pengukuran. REST API digunakan sebagai penghubung antara ESP8266, *website*, dan database MySQL. Data monitoring yang diterima sistem disimpan dalam database sebagai bagian dari fungsi *data logging* sehingga dapat ditampilkan kembali sebagai riwayat data pada *website*.

7. Pengujian dan Evaluasi Sistem

Setelah proses pengkodean selesai, dilakukan pengujian dan evaluasi terhadap sistem secara keseluruhan. Pengujian dilakukan untuk mengetahui apakah sistem telah berfungsi sesuai dengan kebutuhan dan rancangan penelitian.

Pengujian meliputi pembacaan suhu menggunakan DHT22, pengukuran tegangan dan arus menggunakan INA219, pengendalian kipas pada Level 1 dengan PWM 30%, Level 2 dengan PWM 60%, dan Level 3 dengan PWM 100%, serta pengujian fungsi monitoring dan *data logging*.

Selain itu, dilakukan pengujian komunikasi data antara ESP8266, REST API, dan database MySQL serta pengujian tampilan *website* monitoring. Hasil pengukuran sensor juga dibandingkan dengan alat ukur pembanding untuk mengetahui tingkat kesalahan dan akurasi hasil pengukuran. Hasil pengujian kemudian dianalisis secara kuantitatif untuk mengetahui kinerja sistem yang telah dibangun.

8. Implementasi Sistem

Tahap implementasi dilakukan setelah sistem dinyatakan sesuai berdasarkan hasil pengujian dan evaluasi. Pada tahap ini, seluruh perangkat keras dan perangkat lunak diintegrasikan untuk menjalankan sistem monitoring suhu dan pengendalian kipas secara keseluruhan.

Implementasi sistem meliputi penggunaan ESP8266, sensor DHT22, sensor INA219, MOSFET IRLZ44N, kipas DC, rangkaian panel surya, REST API, database MySQL, dan *website* monitoring berbasis React JS. Sistem diterapkan pada *prototype* untuk melakukan monitoring suhu, pengukuran tegangan dan arus, pengendalian kipas menggunakan tiga tingkat PWM, serta penyimpanan dan penyajian data hasil pengukuran.

Dengan penerapan metode *prototyping* tersebut, proses pengembangan sistem dilakukan secara bertahap dan iteratif. Setiap hasil evaluasi digunakan sebagai dasar untuk melakukan perbaikan sebelum sistem dilanjutkan ke tahap berikutnya. Metode ini memungkinkan pengembangan perangkat keras, perangkat lunak, komunikasi IoT, pengendalian kipas *multi-level*, *data logging*, dan integrasi panel surya dilakukan secara terstruktur hingga menghasilkan sistem yang sesuai dengan kebutuhan penelitian.

2.20 UML (Unified Modeling Language)

UML (Unified Modeling Language) adalah metode pemodelan visual yang digunakan untuk membangun sistem berorientasi objek. *Sequence Diagram*, *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *State Machine Diagram*, dan

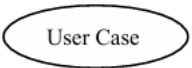
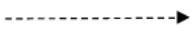
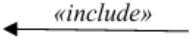
Component Diagram adalah contoh diagram UML yang sering digunakan (Noneng Marthiawati et al., 2024).

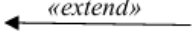
Dalam analisis kebutuhan, Unified Modeling Language (UML) digunakan untuk visualisasi, penentuan ruang lingkup, dan pendokumentasian artefak sistem. Ini sangat menguntungkan bagi berbagai pemangku kepentingan suatu aplikasi. Sebagian besar perancang sistem informasi saat ini memanfaatkan UML diagram untuk menggambarkan informasi dan membantu tim proyek berkomunikasi, mengeksplorasi potensi desain, dan melakukan validasi desain arsitektur perangkat lunak atau pembuatan program aplikasi. (Noneng Marthiawati et al., 2024).

2.20.1 Use case Diagram

Use case diagram merupakan gambaran interaksi antara pengguna dengan sistem yang digunakan untuk menunjukkan fungsi utama pada sistem. Oleh karena itu, pemilihan abstraksi yang sesuai sangat penting agar kebutuhan sistem dapat tergambarkan dengan baik. *Use case* diagram bekerja dengan mendeskripsikan interaksi antara aktor seperti admin maupun user terhadap sistem melalui sebuah diagram yang menunjukkan bagaimana sistem digunakan. Komponen utama pada *Use case* diagram terdiri dari aktor, *Use case*, asosiasi, dan *stereotype* (Ramdany et al., 2024). *Use case diagram* mempunyai simboisimbol dalam penggunaannya, yaitu:

Tabel 2. 8 UML *Use case*

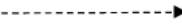
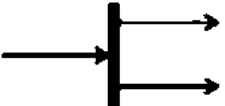
No	Simbol	Keterangan
1.		Aktor :Memerankan orang, sistem, atau alat yang berkomunikasi dengan <i>user case</i> .
2.		<i>Use case</i> :Abstraksi dan interaksi antara sistem dengan actor.
3.		<i>Association</i> :Penghubung antara actor dengan <i>Use case</i> .
4.		Generalisasi : Mununjukkan spesialisasi actor untuk dapat berpartisipasi dengan <i>Use case</i>
5.		<i>Include</i> : Menunjukkan bahwa suatu <i>Use case</i> seluruhnya merupakan fungsionalitas dari <i>Use case</i> lainnya

No	Simbol	Keterangan
6.		<i>Extend</i> : Menunjukkan bahwa suatu <i>Use case</i> merupakan tambahan fungsionalitas dari <i>Use case</i> lainnya jika suatu kondisi terpenuhi

2.20.2 Activity Diagram

Activity diagram merupakan diagram yang digunakan untuk menggambarkan alur aktivitas atau workflow pada suatu sistem. Diagram ini menunjukkan urutan proses mulai dari awal, aktivitas yang terjadi, hingga proses berakhir. Selain itu, activity diagram juga dapat digunakan untuk memodelkan proses yang berjalan secara paralel serta menggambarkan aliran aktivitas di dalam sistem secara keseluruhan (Ramdany et al., 2024). Activity diagram mempunyai simbol-simbol dalam penggunaannya, yaitu:

Tabel 2. 9 UML Activity Diagram

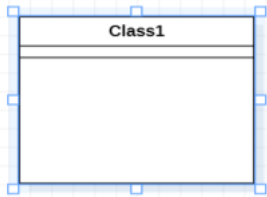
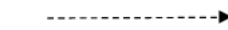
No	Simbol	Fungsi
1.		<i>Activity</i> : Menyatakan bagaimana kelas antarmuka saling berinteraksi satu sama lain.
2.		<i>Control Flow</i> : Menunjukkan urutan eksekusi.
3.		<i>Object Flow</i> : Menunjukkan aliran objek dari sebuah <i>action</i> atau <i>activity</i> ke <i>action</i> .
4.		<i>Start Point</i> : Menyatakan bahwa sebuah objek dibentuk atau diawali.
5.		<i>End Point</i> : Menyatakan bahwa sebuah objek dimulai atau diakhiri.
6.		<i>Join/Penggabungan</i> : Menyatakan untuk menggabungkan kembali <i>activity</i> atau <i>action</i> yang paralel.
7.		<i>Fork</i> : Menyatakan untuk memecah behavior menjadi <i>activity</i> atau <i>action</i> yang paralel.
8.		<i>Decision</i> : Menunjukkan penggambaran suatu keputusan/Tindakan yang harus diambil pada kondisi tertentu.

2.20.3 Class Diagram

Class diagram merupakan diagram yang digunakan untuk menggambarkan hubungan antar kelas serta struktur sistem yang akan dibangun. Diagram ini memperlihatkan atribut, metode, aturan, dan tanggung jawab setiap kelas dalam sistem. Selain itu, class diagram juga digunakan untuk memvisualisasikan struktur

Database dan relasi antar class pada sistem berbasis objek. Pada proses analisis dan perancangan, class diagram membantu menjelaskan struktur serta hubungan antar kelas yang membentuk arsitektur sistem (Ramdany et al., 2024). *Class diagram* mempunyai simbol-simbol dalam penggunaannya, yaitu:

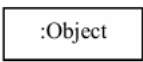
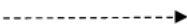
Tabel 2. 10 *Class diagram*

No	Simbol	Fungsi
1.		<i>Class</i> : Merepresentasikan objek atau entitas pada sistem.
2.		<i>Association</i> : Menunjukkan hubungan antar <i>class</i> .
3.		<i>Aggregation</i> : Hubungan dimana satu <i>class</i> memiliki <i>class</i> lain.
4.		<i>Composition</i> : Hubungan kuat antar <i>class</i> yang saling bergantung.
5.		<i>Generalization</i> : Menunjukkan hubungan pewarisan (<i>inheritance</i>).
6.		<i>Dependency</i> : Hubungan ketergantungan antara satu <i>class</i> dengan <i>class</i> lainnya.

2.20.4 Sequence Diagram

Sequence diagram merupakan diagram UML yang digunakan untuk menggambarkan interaksi antar objek dan pertukaran pesan dalam suatu sistem seiring waktu. Diagram ini menunjukkan hubungan antar objek dalam menjalankan fungsi sistem serta membantu menjelaskan alur proses sistem saat runtime (Wayahdi & Ruziq, 2023). *Sequence diagram* mempunyai simbol-simbol dalam penggunaannya, yaitu:

Tabel 2. 11 *Sequence diagram*

No	Simbol	Keterangan
1.		<i>Actor</i> : Pelaku yang berinteraksi dengan sistem.
2.		<i>Object</i> : Komponen berbentuk kotak menunjukkan bagaimana sebuah objek akan berperilaku dalam konteks sistem tertentu.
3.		<i>Lifelines</i> : Menunjukkan kejadian berurutan yang terjadi pada sebuah objek sepanjang proses pembuatan grafik.
4.		<i>Activation Box</i> : mewakili jumlah waktu yang dibutuhkan suatu objek untuk menyelesaikan tugas tertentu.
5.		<i>Return Message</i> : Simbol memberi isyarat bila pengirim harus menunggu respon pesan sebelum melanjutkan.
6.		<i>Synchronous Message</i> : Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
7.		<i>Control</i> : Bertugas untuk mengatur arus informasi dalam sebuah skenario sistem.

2.21 Pengujian Sistem

Pengujian perangkat lunak merupakan salah satu tahapan penting dalam proses pengembangan perangkat lunak. Pengujian dilakukan untuk mengetahui adanya kesalahan pada sistem serta memastikan perangkat lunak yang dihasilkan memiliki kualitas yang baik dan dapat berjalan sesuai fungsinya (Ghibran & Khamaeni, 2023).

2.21.1 Pengujian Perangkat Keras

Pengujian perangkat keras merupakan tahapan yang dilakukan untuk memastikan bahwa seluruh komponen yang digunakan pada sistem dapat bekerja sesuai dengan rancangan yang telah dibuat. Pengujian ini dilaksanakan secara bertahap terhadap masing-masing komponen, mulai dari sensor, mikrokontroler, sistem kendali, hingga catu daya berbasis panel surya. Tujuan dari pengujian ini adalah untuk mengetahui unjuk kerja, tingkat keandalan, serta kemampuan setiap perangkat dalam menjalankan fungsinya masing-masing sebelum seluruh sistem diintegrasikan secara utuh.

Pada penelitian ini, pengujian perangkat keras mencakup sensor DHT22 yang digunakan untuk mengukur suhu, sensor INA219 yang digunakan untuk mengukur tegangan dan arus, NodeMCU ESP8266 yang berperan sebagai pengendali utama, relay dan MOSFET IRLZ44N yang berfungsi sebagai pengendali kipas DC, serta sistem catu daya yang terdiri atas panel surya, modul TP4056, dan baterai rechargeable 18650. Hasil dari pengujian ini digunakan sebagai dasar untuk memastikan bahwa setiap komponen dapat bekerja secara optimal, menghasilkan data yang valid, serta mampu mendukung jalannya proses *monitoring* dan pengendalian kipas otomatis berbasis *Internet of Things* (IoT).

2.21.2 Pengujian Perangkat Lunak

Pengujian kotak putih dan kotak hitam adalah dua metode pengujian software. Pengujian kotak putih menguji aplikasi untuk mengetahui keberhasilan fungsi setiap fitur yang telah dibuat, dan pengujian kotak hitam memastikan bahwa tampilan halaman admin berjalan sesuai dengan persyaratan sistem. (Kurniawan, 2023; Uminingsih et al., 2022).

Pada penelitian ini, pengujian sistem menggunakan metode Black Box Testing, yang menguji fungsi sistem berdasarkan input dan output tanpa melihat struktur kode program. Pengujian dilakukan untuk memastikan bahwa setiap komponen sistem pengawasan dan pengendalian kipas otomatis berbasis IoT dapat berjalan sesuai dengan fungsi yang telah dirancang.

2.21.1 Black Box Testing

Black Box Testing merupakan metode pengujian yang digunakan untuk mengetahui kesalahan pada sistem aplikasi, seperti kesalahan fungsi, tampilan, maupun menu pada aplikasi. Pengujian dilakukan berdasarkan fungsionalitas sistem dengan menggunakan berbagai data masukan untuk memastikan sistem dapat menerima dan memproses data dengan benar. Jika data yang dimasukkan salah maka sistem akan menolak atau data tidak tersimpan pada *Database*, sedangkan jika data benar maka sistem akan menerima dan menyimpan data ke dalam *Database* (Uminingsih et al., 2022).

Pada penelitian ini metode *Black Box* dipakai untuk pengujian pada sensor DHT22, sensor INA219, koneksi NodeMCU ESP8266, REST API, *Database MySQL*, *website monitoring*, serta sistem *Multi level control* pada kipas DC. Selain itu, dilakukan pengujian akurasi sensor dengan membandingkan hasil pembacaan sensor terhadap alat ukur pembanding.