

BAB II DASAR TEORI

2.1 Tinjauan Pustaka

Berdasarkan studi literatur yang sudah dilakukan melalui konferensi internasional maupun jurnal dan paper, didapatkan beberapa penelitian pada Tabel 2.1 yang menjadi referensi dan memiliki topik serta metodologi serupa dengan penelitian ini

Tabel 2. 1 Tinjauan Pustaka

No.	Judul Penelitian	Keterangan	Target Perbaikan
1	<i>Weather Station Design Using IoT Platform Based On Arduino Mega.</i> (Kusriyanto & Putra, 2018)	Sistem stasiun cuaca ini menggunakan Arduino Mega 2560 sebagai unit pemroses utama untuk mengolah data dari sensor DHT-22, BMP-180, dan FC-37, serta dilengkapi fitur data logging dan transmisi data melalui modul Wi-Fi ESP8266 [8].	Meningkatkan akurasi sistem dengan mengganti sensor tersebut menggunakan sensor industri Honde HD-WSM-U-07 yang memiliki presisi lebih tinggi dan <i>error</i> lebih rendah, dengan akuisisi data melalui protokol RS-485 Modbus dan LoRa32.
2	<i>Weather Station Using Raspberry Pi</i> (Mathur, dkk, 2021)	Sistem Stasiun cuaca ini menggunakan Raspberry Pi dan arduino Uno yang dilengkapi dengan SparkFun Weather Shield untuk meremam kondisi iklim, serta memanfaatkan antarmuka	Sistem berbasis Raspberry Pi memiliki konsumsi daya tinggi, jangkauan Wi-Fi terbatas, dan menggunakan komponen hobi yang kurang tahan terhadap cuaca ekstrem, sehingga

		platform cloud untuk analisis data [9].	dikembangkan sistem berbasis LoRa32 dan sensor Honde HD-WSM-U-07
3	<i>Development of Hybrid Dynamical and Machine Learning Based Weather Forecasting using a Network of Dense Weather Station</i> (Kaushik, dkk, 2025)	Penelitian ini mengembangkan model hybrid berbasis Numerical Weather Prediction (WRF) dan machine learning untuk meningkatkan akurasi prediksi numerik, namun masih memiliki kompleksitas komputasi tinggi dan belum menghasilkan klasifikasi cuaca yang mudah dipahami pengguna. [10].	Mengembangkan sistem klasifikasi cuaca menggunakan algoritma MLP yang lebih efisien secara komputasi dan mandiri dari ketergantungan model fisik.
4	<i>A Neuro Model for Weather Forecasting.</i> (Preduna, dkk, 2020)	Penelitian ini menunjukkan bahwa neural network mampu melakukan prediksi cuaca berbasis time series dengan akurasi yang baik akan tetapi belum menerapkan teknik penanganan data tidak seimbang serta validasi silang, sehingga potensi	Mengembangkan klasifikasi kondisi cuaca serta penerapan penanganan data tidak seimbang dan validasi silang untuk meningkatkan keandalan dan generalisasi model.

		bias model masih cukup besar. [11].	
5	<i>Weather Data Prediction using the XGBoost Algorithm on LoRa-Based Weather Station System</i> (Mohammad, dkk, 2025)	Sistem stasiun cuaca ini menggunakan LoRa dan mengimplementasikan algoritma XGBoost sebagai metode utama untuk melakukan prediksi parameter cuaca, dengan memanfaatkan pendekatan model regresi dalam proses pelatihannya [7].	Sistem tersebut masih terbatas pada prediksi numerik, sehingga diperlukan penerapan algoritma <i>Multilayer Perceptron (MLP)</i> untuk melakukan klasifikasi cuaca secara otomatis sebagai pelengkap metode regresi sebelumnya.

2.2 Cuaca dan Parameter Meteorologi

2.2.1 Iklim dan Cuaca

Iklim terbentuk dari integrasi berbagai unsur fisika atmosfer yang disebut unsur-unsur iklim (*climatic elements*). Rotasi bumi menyebabkan perubahan cuaca dan iklim secara harian (diurnal), sedangkan revolusi bumi menimbulkan perubahan musiman dalam skala tahunan. Persebaran iklim di berbagai wilayah dunia sangat beragam dan dipengaruhi oleh faktor-faktor pengendali iklim (*climatic controls*), seperti intensitas radiasi matahari, letak lintang, ketinggian tempat, kondisi topografi, pusat tekanan udara, kedudukan terhadap samudera, pergerakan massa udara, serta arus laut [12]. Iklim merupakan gambaran statistik kondisi atmosfer seperti suhu, tekanan udara, angin, dan kelembaban yang diamati dalam jangka waktu panjang, umumnya minimal 30 tahun [13]. Oleh karena itu, iklim bersifat relatif stabil dibandingkan dengan cuaca yang bersifat sesaat dan cepat berubah. Iklim dapat dipahami sebagai hasil rata-rata kondisi cuaca yang terjadi dari hari ke hari dalam periode waktu yang lama. Setiap wilayah memiliki karakteristik iklim yang berbeda karena bentuk bumi yang bulat menyebabkan distribusi radiasi matahari tidak merata. Faktor lintang geografis, ketinggian

wilayah, jarak dari perairan, serta pengaruh arus laut turut menentukan perbedaan iklim antar daerah.

Berdasarkan klasifikasi iklim matahari, Indonesia termasuk wilayah beriklim tropis yang terletak di antara $23,5^{\circ}$ LU hingga $23,5^{\circ}$ LS [14]. Indonesia juga dipengaruhi oleh sistem angin muson yang berubah arah setiap setengah tahun akibat perbedaan tekanan udara antara Benua Asia dan Australia. Pada periode Desember – Februari, angin barat membawa massa udara lembap yang menyebabkan musim hujan, sedangkan pada bulan Juni–Agustus, angin timur yang cenderung kering menyebabkan musim kemarau. Selain itu, pada masa peralihan atau pancaroba, kondisi cuaca menjadi tidak menentu [12]. Unsur-unsur utama pembentuk iklim meliputi curah hujan, intensitas radiasi matahari, angin, suhu udara, dan kelembaban. Secara umum, iklim dibedakan dari cuaca karena berlaku dalam jangka waktu lama, mencakup wilayah luas, dan merupakan hasil rata-rata dari kondisi cuaca.

2.2.2 Klasifikasi Cuaca dan Hubungan Parameternya

Secara teknis, cuaca merepresentasikan parameter udara pada titik waktu dan lokasi spesifik yang cakupannya relatif sempit. Unsur-unsur cuaca meliputi suhu udara, kelembaban, tekanan udara, angin, dan curah hujan. Berbeda dengan iklim yang bersifat jangka panjang dan relatif stabil, cuaca bersifat dinamis dan mudah berubah. Oleh karena itu, sistem pemantauan dan prediksi cuaca memerlukan pengamatan data secara kontinu dan real-time agar dapat menggambarkan kondisi atmosfer secara akurat.

Tabel 2. 2 Nilai Parameter Cuaca Kompleks Universitas Diponegoro

Kategori Cuaca	Suhu ($^{\circ}$C)	Kelembapan (%)	Curah Hujan (mm/h)	Iradiasi (W/m^2)
Cerah	27,5 – 34,7	32,5 – 74,5	0	>254
Berawan	26,9 – 35	76,4 – 98,7	0 – 0,9	100 – 500
Hujan Sedang	24 – 30,3	92,3 – 98,1	1 – 20	<100
Hujan Lebat	21,6 – 26,6	92,3 – 98,1	>20	<50

Rentang nilai parameter cuaca pada Tabel 2.2 diperoleh dari hasil analisis data observasi AWS Universitas Diponegoro pada periode 1 Januari 2025 hingga 1 Februari 2026. Data yang digunakan telah melalui tahap pembersihan, meliputi penghapusan nilai kosong dan data anomali menggunakan pendekatan *rule-based*. Selanjutnya, analisis dilakukan menggunakan statistik deskriptif yang meliputi nilai rata-rata, minimum, median, maksimum, dan standar deviasi pada setiap kategori cuaca. Analisis ini bertujuan untuk memperoleh gambaran sebaran data yang representatif serta meminimalkan pengaruh nilai ekstrem. Hasil analisis parameter lokal digunakan sebagai dasar penentuan batas keputusan (*decision boundary*) pada sistem klasifikasi cuaca berbasis MLP. Seluruh satuan parameter mengikuti standar pengukuran sistem AWS.

2.2.3 Data Cuaca Kompleks Universitas Diponegoro

Data cuaca yang disajikan pada Tabel 2.3 merupakan data cuaca pada lingkungan Universitas Diponegoro yang diperoleh dari hasil pemodelan regresi menggunakan algoritma XGBoost dalam bentuk data numerik dengan 4 parameter cuaca yaitu, iradiasi matahari, suhu, curah hujan, dan kelembapan udara.

Tabel 2. 3 Data Hasil Prediksi XGBoost pada Sistem Stasiun Cuaca Berbasis LoRa

Tanggal	Suhu (°C)	Iradiasi (W/m²)	Kelembapan (%)	Curah Hujan (mm/h)
28-Jul-2025	29.06	658.83	85.57	0.16
29-Jul-2025	29.48	636.38	86.52	0.1
30-Jul-2025	28.23	628.29	87.81	0.1
31-Jul-2025	29.51	658.98	87.17	0.2
1-Aug-2025	29.28	635.12	87.07	0.14
2-Aug-2025	29.48	634.31	81.02	0.08
3-Aug-2025	31.26	758.30	83.34	0.11
4-Aug-2025	30.00	690.07	84.68	0.12
5-Aug-2025	28.97	627.33	85.01	0.16
6-Aug-2025	29.84	788.06	81.54	0.14

Berdasarkan hasil pengolahan data prediksi XGBoost pada rentang 28 Juli 2025 hingga 6 Agustus 2025, kondisi cuaca di kompleks Universitas Diponegoro berada pada kategori Berawan dan Cerah. Hal ini sejalan dengan kondisi klimatologis wilayah pada periode tersebut yang termasuk dalam musim kemarau, ditandai dengan intensitas radiasi matahari yang relatif tinggi, suhu udara yang hangat, serta curah hujan yang sangat rendah ($< 0,5$ mm/jam). Dengan demikian, proses klasifikasi ini digunakan sebagai tahap transformasi data dari bentuk numerik menjadi bentuk kategorikal yang selanjutnya dimanfaatkan sebagai data masukan pada pemodelan *Multilayer Perceptron*.

2.3 Sistem Stasiun Cuaca dan Komunikasi Data

Subbab ini membahas komponen utama yang mendukung sistem stasiun cuaca, mulai dari proses akuisisi data lingkungan, komunikasi antarperangkat, hingga pengiriman data menuju sistem pemantauan. Integrasi perangkat keras, teknologi komunikasi, dan infrastruktur jaringan diperlukan untuk memastikan data cuaca dapat dikumpulkan, ditransmisikan, dan diolah secara andal sebagai dasar pengambilan keputusan.

2.3.1 IoT

Internet of Things (IoT) merupakan teknologi yang memungkinkan perangkat saling terhubung untuk melakukan penginderaan lingkungan, berbagi informasi, dan mendukung pengambilan keputusan secara otomatis. IoT berpotensi membantu mengatasi berbagai permasalahan global, seperti pertumbuhan populasi, krisis energi, keterbatasan sumber daya, dan pencemaran lingkungan. Dengan demikian, IoT berperan sebagai ekosistem cerdas yang mendukung peningkatan kualitas hidup manusia [15].

Berbeda dengan internet konvensional, perangkat IoT memiliki keterbatasan sumber daya, seperti kapasitas memori, kemampuan pemrosesan, *bandwidth*, dan konsumsi energi. Keterbatasan tersebut dipengaruhi oleh penggunaan baterai dan jumlah perangkat yang besar sehingga efisiensi energi menjadi aspek yang penting. Oleh karena itu, teknologi jaringan konvensional, seperti jaringan seluler atau WiFi, kurang sesuai diterapkan pada beberapa sistem IoT karena kebutuhan energi yang tinggi dan keterbatasan skalabilitasnya [16].

Sebagai solusi atas keterbatasan sumber daya pada perangkat IoT, berkembang berbagai teknologi komunikasi yang dirancang khusus untuk konsumsi daya rendah dan jangkauan luas, yang dikenal sebagai *Low Power Wide Area Network* (LPWAN). Teknologi LPWAN memungkinkan perangkat IoT berkomunikasi dalam jarak yang sangat jauh dengan kebutuhan energi yang minimal, sehingga cocok untuk aplikasi berskala besar. Umumnya, jaringan LPWAN menggunakan topologi bintang, di mana perangkat berkomunikasi langsung dengan node pusat, sehingga arsitektur jaringan menjadi lebih sederhana dan mudah dikelola dibandingkan jaringan multi-hop yang kompleks [15], [16]. Untuk memberikan gambaran karakteristik berbagai teknologi komunikasi yang umum digunakan dalam sistem IoT, Tabel 2.4 menyajikan perbandingan antara Bluetooth, ZigBee, Wi-Fi, dan LoRa berdasarkan beberapa parameter utama.

Tabel 2. 4 Perbandingan Teknologi Komunikasi pada Sistem IoT

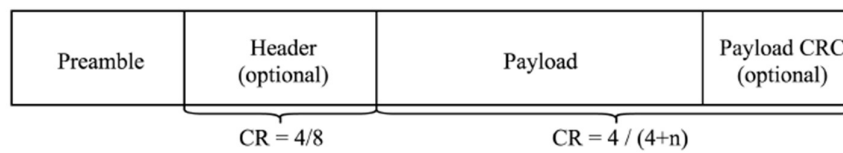
Spesifikasi	Bluetooth	ZigBee	Wi-Fi	LoRa
Arus Max.	30 mA	30 mA	100 mA	17 mA
Jangkauan	10 m	10-100 m	100 m	>15 km
Kecepatan Data	1 Mbps	250 Kbps	11 Mbps & 54 Mbps	290bps sampai 50 Kbps
Topologi	Star	Star & Mesh	Star & Point-to-point	Star
Biaya	Rendah	Rendah	Sedang	Rendah

2.3.2 LoRa

LoRa (*Long Range*) merupakan teknologi komunikasi nirkabel yang dirancang untuk memenuhi kebutuhan transmisi data jarak jauh dengan konsumsi daya yang rendah, sehingga dapat diterapkan pada aplikasi *Internet of Things* (IoT) yang melibatkan banyak perangkat dengan keterbatasan sumber energi. Dengan karakteristik tersebut, LoRa memungkinkan perangkat sensor atau aktuator beroperasi dalam jangka waktu lama menggunakan sumber daya baterai, sehingga cocok untuk sistem monitoring dan akuisisi data berskala besar [16], [17].

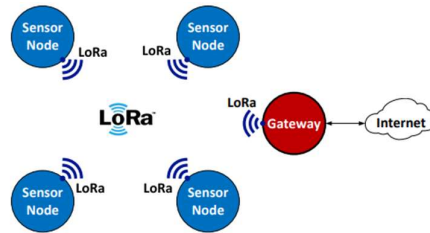
LoRa beroperasi pada pita frekuensi ISM (*Industrial, Scientific, and Medical*) yang tidak berlisensi, seperti 433 MHz, 868 MHz, dan 915 MHz,

tergantung regulasi di masing-masing wilayah [18]. Lapisan fisik LoRa dikembangkan oleh Semtech dan mendukung komunikasi berdaya rendah, berkecepatan data rendah, namun dengan jangkauan yang luas. Ukuran payload dalam satu transmisi berkisar antara 2 hingga 255 byte, dengan laju data maksimum mencapai sekitar 50 kbps. Dalam kondisi ideal, LoRa mampu menjangkau hingga sekitar 5 km di wilayah perkotaan dan lebih dari 15 km di wilayah pedesaan dengan kondisi line of sight. Konsumsi daya yang sangat rendah memungkinkan perangkat berbasis LoRa dapat beroperasi menggunakan baterai hingga lebih dari 10 tahun [17].



Gambar 2.1 *Frame LoRa*

Setiap transmisi LoRa dikemas dalam bentuk frame yang ditunjukkan pada Gambar 2.1, frame ini terdiri dari beberapa bagian utama, yaitu preamble, header, payload, dan CRC [16]. Untuk mengatur kinerja komunikasi dan konsumsi energi, LoRa menyediakan beberapa parameter pada lapisan fisik (*physical layer*) yang dapat dikonfigurasi sesuai kebutuhan aplikasi. Parameter utama tersebut meliputi *Transmission Power* (TP), *Carrier Frequency* (CF), *Spreading Factor* (SF), *Bandwidth* (BW), dan *Coding Rate* (CR). Pengaturan parameter-parameter ini akan memengaruhi jangkauan komunikasi, laju data, ketahanan terhadap gangguan, serta efisiensi energi perangkat [19]. Dalam implementasi jaringan LoRa, performa sistem tidak hanya ditentukan oleh parameter fisik, tetapi juga oleh parameter kualitas transmisi yang menggambarkan kondisi kanal komunikasi, yaitu RSSI, SNR, dan *packet loss* [20].

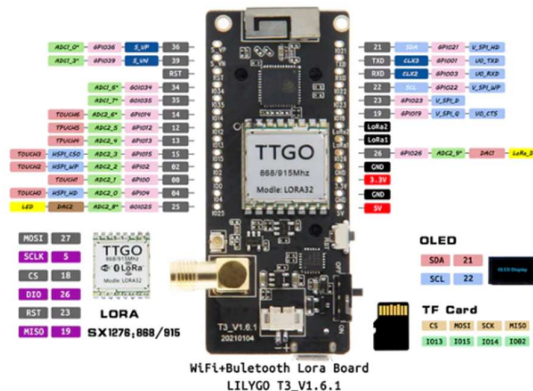


Gambar 2. 2 Arsitektur LoRa P2P

Arsitektur jaringan LoRa umumnya menggunakan topologi *star-of-stars* yang terdiri dari tiga komponen utama, yaitu *end-device*, *gateway*, dan *network server*. Akan tetapi dalam penelitian ini menggunakan LoRa point-to-point yang berlangsung secara langsung antar perangkat end node. Peer-to-Peer (P2P) adalah model jaringan di mana setiap node memiliki peran yang setara tanpa adanya server pusat. Setiap node dapat saling berkomunikasi secara langsung maupun melalui node lain sebagai perantara [21]. Secara regulasi, alokasi frekuensi LoRa di Indonesia berada pada rentang spektrum 920–923 MHz yang termasuk dalam regional parameter AS923 untuk wilayah Asia Tenggara [22]. Rentang AS923 mencakup frekuensi 915–928 MHz dengan fokus saluran operasional pada sub-band 923–923,5 MHz, sehingga penggunaan frekuensi 915 MHz masih sesuai dengan standar operasional LoRa dan LoRaWAN di Indonesia. Oleh karena itu, pada penelitian ini sistem komunikasi diimplementasikan menggunakan pita frekuensi 915 MHz dengan memanfaatkan modul MCU LoRa32 V2.1_1.6 sebagai perangkat utama [23].

2.3.3 LoRa 32 V2.1_1.6

LoRa32 V2.1_1.6 merupakan modul mikrokontroler berbasis ESP32 yang telah terintegrasi dengan transceiver LoRa SX1276 dan dirancang untuk beroperasi pada pita frekuensi 915 MHz. Modul ini mendukung beberapa jenis komunikasi nirkabel, yaitu Wi-Fi, Bluetooth, dan LoRa, sehingga sesuai untuk implementasi sistem IoT yang memerlukan jangkauan komunikasi luas, konsumsi daya rendah, serta pengiriman data yang efisien. LoRa32 V2.1_1.6 bekerja pada rentang tegangan 1,8 V hingga 3,7 V dan dilengkapi dengan berbagai fitur pendukung, seperti layar OLED berukuran 0,96 inci, slot kartu microSD, serta konektor antena eksternal tipe SMA [24].



Gambar 2. 3 Pin LoRa

Pada modul LoRa32 V2.1_1.6, komunikasi antara mikrokontroler dan chip LoRa SX1276 dilakukan menggunakan protokol Serial Peripheral Interface (SPI). Sedangkan dalam proses akuisisi data dari sensor cuaca HONDE HD-WSM-U-07, modul memanfaatkan antarmuka UART yang dikonversi ke protokol RS-485. Dalam penelitian ini, LoRa32 V2.1_1.6 difungsikan sebagai pusat kendali sistem (MCU utama) yang mengatur proses komunikasi LoRa antara perangkat pengirim dan penerima pada sistem stasiun cuaca berbasis LoRa. Gambar 2.3 menyajikan pin yan terdapat pada Pada modul LoRa32 V2.1_1.6. Rincian konfigurasi pin yang mencakup nomor pin, nama pin, serta fungsi masing-masing pin disajikan pada Tabel 2.5.

Tabel 2. 5 Spesifikasi LoRa32

Spesifikasi	LoRa32 V2.1_1.6
MCU	ESP32
Tegangan Kerja	1.8V – 3.7V
Konsumsi Arus	10 – 14mA
Frekuensi <i>Error</i>	±15KHz
Suhu Operasi	-40°C sampai +85°C
LoRa Modul	SX1278 (433), SX1276 (868 & 915)
Chip Serial	CH9102
Memori	4 MB Flash
Protokol Nirkabel	Wi-Fi & Bluetooth 4.2
Antarmuka Digital	SPI, I2C, UART, PWM

Tabel 2.5 Lanjutan

Spesifikasi	LoRa32 V2.1_1.6
Slot Kartu	Micro SD
Sensitivitas	-i49dBm
Diameter	64.47mm × 27mm
Power Supply	USB Mikro (3.3V – 5V) JST GH 2pin 1.25mm

2.3.4 Sensor Cuaca Honde HD-WSM-U-07

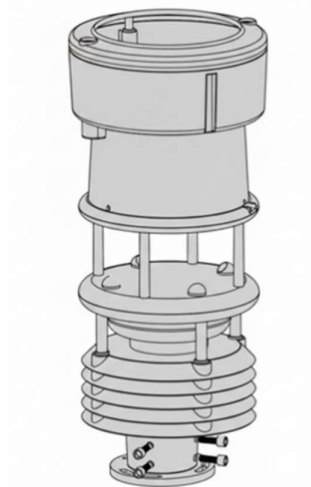
Sensor cuaca HONDE HD-WSM-U-07 merupakan perangkat pengukuran lingkungan yang diproduksi oleh Honde Technology Co., Ltd dan digunakan untuk memperoleh data kondisi cuaca secara real-time. Sensor ini bekerja dengan catu daya 12–24 V dan menggunakan komunikasi Modbus RTU yang ditransmisikan melalui media RS-485. Parameter komunikasi yang digunakan meliputi kecepatan transmisi 9600 bps, panjang data 8 bit, 1 stop bit, serta tanpa penggunaan parity bit. Selain sebagai sarana pertukaran data, protokol RS-485 juga menyediakan mekanisme pemeriksaan kesalahan menggunakan *Cyclic Redundancy Check* (CRC) serta sistem pengalamatan register yang memungkinkan setiap parameter cuaca dapat diakses secara terstruktur. Sensor ini termasuk dalam kategori sensor cuaca terintegrasi. Parameter meliputi suhu, kelembapan udara, tekanan atmosfer, kecepatan angin, arah angin, curah hujan, serta radiasi matahari. Tabel 2.6 menunjukkan spesifikasi teknik yang siliki oleh sensor Honde HD-WSM-U-07.

Tabel 2. 6 Spesifikasi Teknik Sensor Honde HD-WSM-U-07

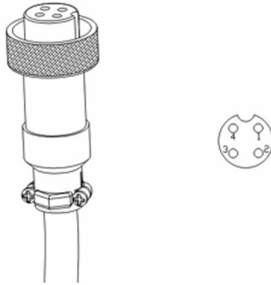
Parameter	Rentang	Resolusi	Akurasi
Suhu	-40°C – 60°C	0.01°C	± 0.3°C (25°C)
Kelembapan	0-100%RH	0.01%RH	± 3%RH (0-90%RH)
Tekanan Udara	300-1100hpa	0.1hpa	± 0.5hpa (0-30°C)
Kecepatan Angin	0-60m/s	0.01m/s	(0-30m/s) ± 0.3m/s (30-60m/s) ± 0.5m/s
Arah Angin	0-360°	0.1°	± 2°
Curah Hujan	0-200mm/h	0.2mm	Error <5%
Iradiasi Matahari	0-2000W/m ²	1W	3%

Dari sisi fisik seperti pada Gambar 2.4, sensor HONDE HD-WSM-U-07 tersusun atas beberapa komponen utama yang terintegrasi dalam satu unit, diantaranya adalah

1. *Illumination* atau *Solar Radiation*, berfungsi untuk mengukur intensitas radiasi matahari yang diterima oleh sensor.
2. *Tipping Bucket*, berfungsi untuk mengukur curah hujan dengan mekanisme ember jungkit dimana setiap perpindahan merepresentasikan jumlah hujan tertentu.
3. *Ultrasonic Wind Speed*, berfungsi untuk mengukur kecepatan angin menggunakan gelombang ultrasonik tanpa komponen mekanik bergerak.
4. *Ultrasonic Wind Direction*, berfungsi untuk mengukur arah angin berdasarkan perbedaan waktu tempuh sinyal ultrasonik.
5. *Control Circuit*, merupakan rangkaian kendali yang berfungsi sebagai pusat pemrosesan sinyal dari seluruh sensor.
6. *Louver Box*, berfungsi sebagai tempat pengukuran suhu, kelembapan, dan tekanan udara, sekaligus melindungi sensor dari kondisi eksternal.
7. *Bottom Fixed Flange*, berfungsi sebagaiudukan mekanis untuk menopang dan memasang sensor di lokasi pengukuran.



Gambar 2. 4 Sensor Cuaca Honde



Gambar 2. 5 Kabel Komunikasi RS-485 pada Sensor Cuaca Honde

Pada koneksi fisik yang ditunjukkan pada Gambar 2.5, sensor ini dilengkapi dengan 4 pin utama, yaitu 2 pin jalur suplai daya positif dan negatif dan 2 pin komunikasi RS-485 A serta RS-485 B. Dalam penelitian ini, suplai daya sensor diperoleh dari sumber SMPS 12 V–5 A, sedangkan jalur komunikasi RS-485 dihubungkan ke mikrokontroler melalui konverter UART TTL ke RS-485. Melalui konfigurasi alamat komunikasi RS-485 yang disajikan pada Tabel 2.7 dan fungsi kode pada modbus RTU yang disajikan pada Tabel 2.8, data sensor dapat dikirimkan ke MCU menggunakan protokol Modbus RTU dengan akses parameter dilakukan menggunakan *function code 03* untuk membaca data (*read holding register*) dan *function code 06* untuk pengaturan alamat (*write single register*) [25].

Tabel 2. 7 Alamat Komunikasi RS-485

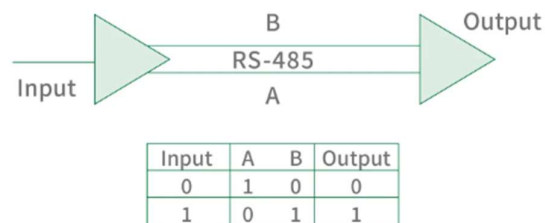
Address	Operasi	Isi
0x0009	Read Only	Suhu
0x000A	Read Only	Kelembapan Udara
0x000B	Read Only	Tekanan Atmosfer
0x000C	Read Only	Kecepatan Angin
0x000D	Read Only	Arah Angin
0x000E	Read Only	Curah Hujan 10 Menit
0x000F	Read Only	Iradiasi
0x0000	Read Write	Alamat Sensor 0x01 – 0xFF

Tabel 2. 8 Fungsi Kode pada Modbus RTU

Kode	Fungsi
1 = 01H	Membaca status coil
2 = 02H	Membaca <i>input</i> diskrit
3 = 03H	Membaca register holding
4 = 04H	Membaca register <i>input</i>
5 = 05H	Mengaktifkan satu coil
6 = 06 H	Mengaktifkan satu register
15 = 0FH	Mengaktifkan beberapa coil
16 = 10H	Menetapkan beberapa register

2.3.5 Protokol RS-485

RS-485 merupakan standar komunikasi serial yang menggunakan metode transmisi diferensial, dimana data dikirim berdasarkan selisih tegangan antara dua jalur sinyal. Skema diferensial ini membuat RS-485 lebih tahan terhadap gangguan noise dibandingkan komunikasi *single-ended*. Selain itu, protokol ini memungkinkan banyak perangkat terhubung dalam satu jalur komunikasi yang sama hanya dengan menggunakan sepasang kabel *twisted pair*, sehingga sangat cocok untuk sistem jaringan jarak jauh dan lingkungan dengan interferensi tinggi [26]. Skema transmisi RS-485 secara umum ditunjukkan pada Gambar 2.6.



Gambar 2. 6 Transmisi Data Half-Duplex

Pada penelitian ini digunakan mode komunikasi *half-duplex* diferensial dimana proses pengiriman dan penerimaan data dilakukan secara bergantian melalui sepasang kabel yang sama. Oleh karena itu, jalur keluaran pemancar dan masukan penerima digabungkan pada satu pasangan *twisted pair*. Pada penelitian

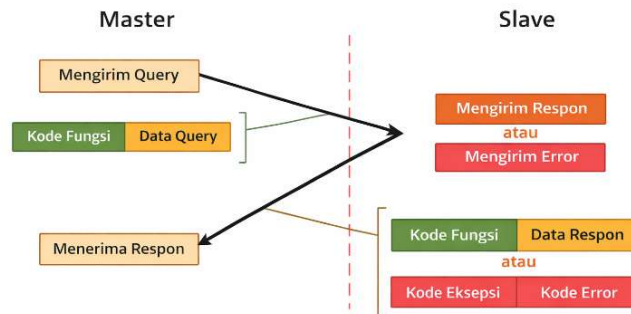
ini, RS-485 digunakan sebagai media komunikasi utama antara sensor cuaca HONDE HD-WSM-U-07 dan sistem mikrokontroler. Jalur RS-485 A dan B dihubungkan ke konverter UART TTL ke RS-485 yang berfungsi sebagai antarmuka agar data Modbus RTU dari sensor dapat dibaca dan diproses oleh MCU berbasis TTL. Tabel 2.9 menunjukkan spesifikasi RS-485.

Tabel 2. 9 Spesifikasi RS-485

Spesifikasi	RS-485
Mode Operasi	Half-duplex, Differential
Tegangan Operasi	-7V sampai 12V
Maksimal Laju Data	10Mbps
Tegangan Keluaran Driver	1,5V – 5V
Arus Keluaran Driver	$\pm 55\text{mA}$
Batas Arus Hubung Singkat	250mA
Impedansi Karakteristik	120 ohm
Topologi Jaringan	Bus

2.3.6 Modbus dan Modbus RTU

Modbus merupakan protokol komunikasi terbuka yang banyak digunakan pada sistem industri dan *Internet of Things* (IoT) karena strukturnya yang sederhana, mudah diimplementasikan, serta kompatibel dengan berbagai jenis perangkat. Protokol ini dapat beroperasi pada berbagai media komunikasi, baik serial seperti RS-232 dan RS-485 maupun jaringan berbasis Ethernet (TCP/IP). Secara umum, Modbus menerapkan arsitektur *master-slave* seperti yang ditunjukkan pada gambar 2.7, dimana perangkat master berperan aktif mengirimkan permintaan data, sedangkan perangkat *slave* bertugas memberikan respons sesuai perintah yang diterima [27].



Gambar 2. 7 Master-Slave pada Modbus

Salah satu jenis Modbus yang digunakan dalam komunikasi serial adalah Modbus RTU (*Remote Terminal Unit*). Modbus RTU bekerja pada lapisan Data Link OSI dan menggunakan media fisik seperti RS-485 atau RS-232. Data dikirim dalam bentuk frame yang terdiri atas beberapa bagian, yaitu penanda awal, alamat *slave*, *function code*, data, pemeriksaan kesalahan menggunakan *Cyclic Redundancy Check* (CRC), serta penanda akhir, seperti yang ditunjukkan pada gambar 2.8. Alamat slave memiliki rentang nilai 1 hingga 247, sedangkan alamat 0 digunakan untuk fungsi broadcast. Dalam satu byte Modbus RTU umumnya terdapat start bit, bit data, bit paritas, dan stop bit, sehingga total terdiri dari beberapa bit karakter yang membentuk satu frame komunikasi [28].

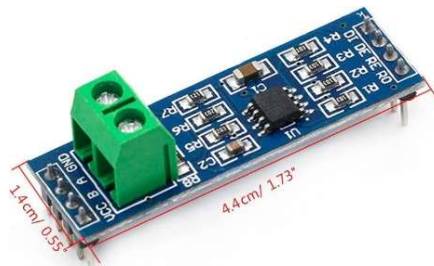
Protocol Data Unit					
Start	Address (Slave ID)	Function Code	Data	CRC Error Check	End
3,5 bytes	1 byte	1 byte	n bytes	2 bytes	3,5 bytes
Application Data Unit					

Gambar 2. 8 Frame pada Modbus RTU

Pada penelitian ini, Modbus RTU digunakan sebagai protokol komunikasi serial antara sensor HONDE HD-WSM-U-07 dan MCU melalui jalur RS-485 dengan bantuan konverter UART TTL to RS-485. MCU LoRa32 berperan sebagai master yang secara berkala mengirimkan permintaan data, sedangkan sensor bertindak sebagai slave yang merespons dengan data parameter cuaca. Data yang diterima kemudian diproses oleh MCU, dikonversi ke format JSON, dan dikirimkan ke modul receiver melalui jaringan LoRa untuk diteruskan ke server. Adapun fungsi kode pada Modbus RTU disajikan dalam Tabel 2.8.

2.3.7 UART TTL to RS-485 Converter

UART TTL to RS-485 Converter merupakan perangkat antarmuka yang berfungsi untuk mengubah sinyal komunikasi serial berbasis UART TTL menjadi sinyal komunikasi RS-485. Konversi ini memungkinkan perangkat yang hanya mendukung level tegangan TTL, seperti mikrokontroler, dapat berkomunikasi dengan perangkat lain yang menggunakan standar RS-485. Secara prinsip, konverter ini bekerja dengan mengubah level tegangan TTL yang berada pada kisaran 3,3 V hingga 5 V menjadi sinyal diferensial RS-485 yang ditransmisikan melalui dua jalur utama, yaitu A dan B. Mekanisme diferensial ini memungkinkan data dikirim dalam bentuk selisih tegangan antar kedua jalur, sehingga lebih tahan terhadap noise dibandingkan komunikasi serial konvensional seperti RS-232. Dengan karakteristik tersebut, komunikasi RS-485 mampu menjangkau jarak hingga sekitar 1200 meter tanpa mengalami penurunan kualitas sinyal yang signifikan.

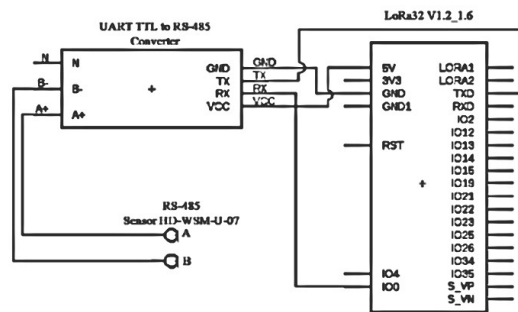


Gambar 2. 9 UART/TTL to RS-485 Converter

Pada penelitian ini, UART TTL to RS-485 Converter digunakan sebagai penghubung antara sensor cuaca HONDE HD-WSM-U-07 dengan mikrokontroler LoRa32 V2.1_1.6 ditunjukkan pada Gambar 2.10. Jalur A dan B dari konverter dihubungkan ke jalur RS-485 pada sensor, sedangkan pin TX dan RX konverter dihubungkan ke port UART pada MCU. Konfigurasi ini memungkinkan data hasil pengukuran sensor dapat diterima, dibaca, dan diproses oleh MCU secara andal meskipun berada pada jarak yang relatif jauh dan lingkungan dengan gangguan tinggi, spesifikasi UART/TTL to RS-485 Converter ditunjukkan pada Tabel 2.10.

Tabel 2. 10 Spesifikasi UART/TTL to RS-485 Converter

Spesifikasi	UART/TTL to RS-485 Converter
Tegangan <i>Input</i>	3.3V dan 5V
Jarak Transmisi	1200 m
Suhu Operasi	-40°C - 80°C
Dimensi	14mm × 44mm



Gambar 2. 10 Rangkaian UART/TTL Converter dengan LoRa32

2.3.8 SMPS 12V-5A

Switch Mode Power Supply (SMPS) merupakan jenis catu daya yang berfungsi untuk mengonversi tegangan listrik bolak-balik (AC) menjadi tegangan searah (DC) dengan tingkat efisiensi yang tinggi. SMPS dirancang untuk menghasilkan keluaran tegangan dan arus yang relatif stabil meskipun terjadi variasi pada beban maupun sumber listrik. Karakteristik utama dari SMPS adalah penggunaan teknik *high-frequency switching* yang memungkinkan proses konversi energi berlangsung dengan kehilangan daya yang minimal.



Gambar 2. 11 SMPS

SMPS 12V-5A memiliki masukan berupa tegangan 220 VAC yang terdiri dari terminal fasa (L), netral (N), dan *ground* serta menghasilkan keluaran berupa dua terminal positif 12 VDC dan dua terminal negatif 12 VDC. Secara umum,

SMPS bekerja dengan menerima tegangan AC dari sumber listrik, kemudian menyearahkannya menggunakan rangkaian penyearah jembatan (*bridge rectifier*). Spesifikasi Tegangan hasil penyearahan tersebut selanjutnya difilter menggunakan kapasitor untuk mengurangi ripple dan gangguan sinyal. Setelah itu, tegangan DC masuk ke rangkaian switching yang beroperasi pada frekuensi tinggi untuk menghasilkan tegangan keluaran yang lebih stabil, efisien, dan sesuai dengan kebutuhan sistem. Spesifikasi SMPS ditunjukkan pada Tabel 2.11.

Tabel 2. 11 Spesifikasi SMPS 12V-5A

Spesifikasi	SMPS 12V-5A
Tegangan Keluaran	11V – 13.8V
Arus Keluaran	5A
Daya Maksimum	60 Watt
Dimensi	159 x 98 x 42 mm

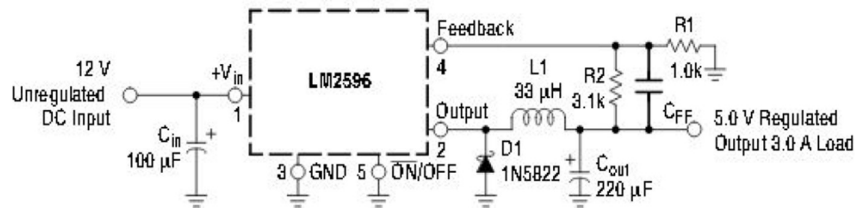
Dalam penelitian ini, SMPS 12V–5A digunakan sebagai sumber catu daya utama yang berfungsi mengubah tegangan PLN menjadi tegangan 12 VDC untuk menyuplai daya pada modul komunikasi serta sensor cuaca. Arus SMPS yang dibutuhkan pada sistem ini dapat dihitung menggunakan persamaan matematis 2.1

$$I_{SMPS} = \frac{P_{Total}}{V_{Input}} \quad (2.1)$$

2.3.9 Buck Converter 5V LM2596

Buck converter merupakan rangkaian konversi daya DC ke DC yang berfungsi untuk menurunkan nilai tegangan masukan menjadi tegangan keluaran yang lebih rendah dengan efisiensi tinggi. Tabel 2.12 menyajikan informasi mengenai spesifikasi Buck Converter LM2596. Buck Converter memiliki tegangan *input* DC 7-24 V dengan arus maksimal 3A. Rangkaian ini banyak digunakan pada sistem elektronika karena mampu menghasilkan tegangan yang stabil tanpa kehilangan daya yang besar seperti pada regulator linear. Gambar 2.12 menunjukkan rangkaian buck converter yang terdiri dari beberapa komponen. Prinsip kerja buck converter didasarkan pada proses switching yang dilakukan oleh transistor di dalam IC LM2596 yang bekerja bersama dengan komponen dioda, induktor, dan kapasitor. Ketika transistor berada dalam kondisi aktif, energi listrik

akan disimpan di dalam induktor. Sebaliknya, saat transistor tidak aktif, induktor akan melepaskan energi tersebut melalui dioda menuju beban. Kombinasi induktor dan kapasitor berfungsi sebagai filter untuk meredam riak tegangan sehingga keluaran menjadi lebih halus dan stabil.



Gambar 2. 12 Rangkaian Buck Converter

Dalam penelitian ini, buck converter LM2596 digunakan untuk menurunkan tegangan 12 VDC dari SMPS menjadi 5 VDC agar sesuai dengan kebutuhan operasional modul LoRa32 V2.1_1.6, Spesifikasi komponen ini ditunjukkan pada Tabel 2.11.

Tabel 2. 12 Spesifikasi Buck Converter LM2596

Spesifikasi	Buck Converter LM2596
Input DC	7V – 24 V
Arus Maksimal	3A
Efisiensi	80 – 90%
Suhu Operasi	-45 °C sampai ± 125°C
Dioda	Dioda SS510
Dioda Zener	Dioda Zener 5.1V
Induktor	33uH SMD
Kapasitor	ElcoSMD 330uF 25V

2.3.10 DIP Switch 2 pin

DIP (Dual In-line Package) switch merupakan komponen saklar mekanis yang digunakan untuk melakukan pengaturan atau konfigurasi perangkat elektronik secara manual. DIP switch terdiri dari beberapa saklar kecil yang dapat diubah posisinya secara individual, sehingga setiap saklar dapat mengaktifkan atau menonaktifkan jalur rangkaian tertentu sesuai kebutuhan sistem. Pada tipe 2-pin,

DIP switch umumnya bekerja dalam mode SPST (*Single Pole Single Throw*) dimana setiap saklar hanya memiliki satu jalur yang dapat dibuka atau ditutup.



Gambar 2. 13 DIP Switch 2 Pin

Dalam penelitian ini, DIP switch dimanfaatkan sebagai media pengaturan mode kerja sistem, seperti pemilihan fungsi transmitter, receiver berbasis Wi-Fi, maupun receiver berbasis LAN. Pengaturan dilakukan melalui perubahan posisi saklar sebelum sistem dinyalakan. Apabila perubahan perlu dilakukan saat sistem aktif, konfigurasi tetap dapat diperbarui dengan mengatur posisi DIP switch lalu menekan tombol reset.

2.3.11 Push Button 4 Kaki

Push button merupakan jenis saklar yang digunakan untuk menghubungkan atau memutus aliran listrik secara sementara melalui mekanisme tekan. Pada umumnya, push button memiliki dua kondisi logika, yaitu *Normally Open* (NO) dan *Normally Closed* (NC). Push button yang digunakan dalam sistem ini bertipe *Normally Open*, sehingga dalam kondisi normal (tidak ditekan) rangkaian berada dalam keadaan terbuka, sedangkan saat tombol ditekan rangkaian akan terhubung.

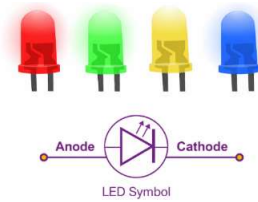


Gambar 2. 14 Push Button 4 Kaki

Dalam penelitian ini, push button digunakan sebagai antarmuka kontrol pengguna yang berfungsi sebagai tombol select dan reset. Tombol select digunakan untuk menjalankan modul ketika media penyimpanan micro SD tidak tersedia, sedangkan tombol reset digunakan untuk melakukan konfigurasi ulang mode sistem

ketika perangkat sedang menyala, yaitu dengan mengubah posisi DIP switch kemudian menekan tombol reset agar konfigurasi baru dapat diterapkan.

2.3.12 LED



Gambar 2. 15 LED

Light Emitting Diode (LED) merupakan komponen elektronika berbasis semikonduktor yang berfungsi sebagai sumber cahaya ketika dialiri arus listrik. LED termasuk jenis dioda yang hanya dapat menghantarkan arus dalam satu arah, yaitu dari anoda ke katoda. Cara kerja LED didasarkan pada prinsip elektroluminesensi, yaitu pelepasan energi dalam bentuk cahaya akibat perpindahan elektron di dalam material semikonduktor. Warna cahaya yang dihasilkan bergantung pada jenis bahan semikonduktor yang digunakan, seperti merah, hijau, biru, atau putih. LED banyak digunakan karena memiliki konsumsi daya rendah, umur pakai yang panjang, serta respons yang cepat [29]. Pada penelitian ini LED digunakan sebagai indikator status penerimaan data LoRa dan keberhasilan pengiriman data menuju server atau database melalui MQTT.

2.4 Perangkat Lunak

Subbab ini membahas perangkat lunak yang digunakan untuk mendukung proses akuisisi, pengolahan, komunikasi, dan visualisasi data pada sistem yang dikembangkan.

2.4.1 Python

Python merupakan bahasa pemrograman tingkat tinggi yang bersifat interpretatif dan dapat dijalankan pada berbagai sistem operasi serta dapat digunakan untuk berbagai keperluan, termasuk analisis data dan machine learning. Python juga memiliki banyak library pendukung, salah satunya adalah scikit-learn yang digunakan dalam penelitian ini untuk membangun dan mengimplementasikan model klasifikasi. Oleh karena fleksibilitas dan kemudahan tersebut, Python menjadi salah satu bahasa pemrograman yang paling banyak digunakan secara

global [30]. Pada penelitian ini, Python dimanfaatkan sebagai bahasa utama dalam pengembangan sistem stasiun cuaca berbasis LoRa, terutama untuk pembuatan model machine learning, pengolahan data sensor, serta integrasi dengan sistem back-end berbasis Laravel. Selain itu, Python menyediakan banyak library yang mendukung pemrosesan data, pengembangan model pembelajaran mesin, serta kemudahan koneksi dengan basis data MySQL sebagai media penyimpanan.

2.4.2 MQTT

MQTT (*Message Queuing Telemetry Transport*) merupakan protokol komunikasi yang dirancang khusus untuk kebutuhan komunikasi machine-to-machine dan *Internet of Things* (IoT). Protokol ini menggunakan model publish-subscribe yang memungkinkan pertukaran data berlangsung secara ringan dan efisien, terutama pada perangkat dengan sumber daya terbatas. MQTT terdiri dari beberapa komponen utama, yaitu publisher sebagai pengirim data, subscriber sebagai penerima data, broker sebagai pengelola dan penghubung komunikasi, serta topik sebagai jalur pengelompokan pesan. Pada mekanismenya, publisher mengirimkan pesan ke broker melalui topik tertentu, kemudian subscriber akan menerima pesan tersebut dengan melakukan subscribe pada topik yang sama. Dengan konsep ini, publisher dan subscriber tidak perlu saling mengetahui identitas satu sama lain, sehingga komunikasi dapat dilakukan secara fleksibel dan skalabel [31]. Dalam penelitian ini, MQTT digunakan sebagai protokol komunikasi antara modul receiver dan server dengan skema yang ditunjukkan pada Gambar 2.16. Modul receiver berperan sebagai publisher yang mengirimkan data sensor ke broker, sedangkan server bertindak sebagai subscriber yang menerima data tersebut untuk selanjutnya diproses dan ditampilkan.

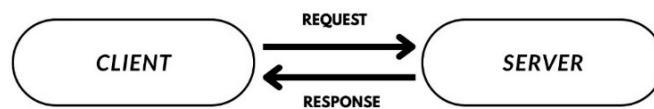


Gambar 2. 16 Skema Pengiriman Data pada MQTT

2.4.3 Client-Server

Server merupakan sistem komputer yang berfungsi menyediakan layanan data kepada klien, mengelola permintaan komunikasi, menyimpan informasi, serta

mengatur lalu lintas data dalam suatu jaringan. Server bekerja berdasarkan arsitektur *client-server*, di mana client berperan sebagai pihak yang mengirimkan permintaan data atau layanan, sedangkan server memproses permintaan tersebut dan memberikan respon sesuai kebutuhan. Selain itu, server juga memungkinkan penyimpanan data serta pengaturan hak akses terhadap informasi yang tersedia [32]. Dalam sistem ini, modul receiver dan aplikasi pengguna bertindak sebagai client, sementara PC digunakan sebagai server lokal yang berperan sebagai pusat pengelolaan data stasiun cuaca. Server bertugas menerima data dari modul receiver, mengatur endpoint, mengelola basis data, menjalankan broker MQTT, serta mengatur tampilan sistem melalui *back-end* dan *front-end* hingga data dapat disajikan dalam bentuk dashboard.



Gambar 2. 17 *Client-Server*

2.4.4 Laravel

Laravel merupakan framework pengembangan aplikasi web berbasis PHP yang menerapkan arsitektur *Model-View-Controller* (MVC). Pada konsep MVC, model berfungsi untuk mengelola data dan relasi antar tabel, view digunakan untuk menampilkan data kepada pengguna, sedangkan controller bertugas menghubungkan model dan view serta mengatur alur logika aplikasi [33]. Laravel juga menyediakan fitur *routing* yang berperan sebagai penghubung antara permintaan (*request*) dari pengguna dengan controller yang sesuai, sehingga setiap permintaan harus melalui proses routing terlebih dahulu sebelum diproses lebih lanjut. Selain itu, Laravel memiliki berbagai fitur pendukung seperti sistem migrasi basis data, template engine blade, serta dokumentasi yang lengkap [34].

Dalam penelitian ini, Laravel digunakan sebagai framework utama untuk membangun dashboard monitoring stasiun cuaca serta menampilkan hasil prediksi kondisi cuaca dalam beberapa hari ke depan.

2.4.5 Laragon

Laragon merupakan sebuah development server lokal yang digunakan untuk menyediakan lingkungan server dalam proses pengembangan dan pengujian sistem berbasis web. Laragon berfungsi sebagai pengelola berbagai layanan server seperti web server dan database server dalam satu platform terintegrasi, sehingga memudahkan proses konfigurasi dan manajemen sistem. Dibandingkan dengan software sejenis seperti XAMPP atau WAMP, Laragon memiliki keunggulan dalam hal kemudahan konfigurasi, kecepatan proses, serta fleksibilitas dalam pengelolaan layanan server tanpa memerlukan pengaturan manual yang kompleks. Antarmuka yang sederhana juga memungkinkan pengguna untuk menjalankan, menghentikan, serta mengatur layanan server secara cepat [35].

Dalam penelitian ini, Laragon digunakan sebagai server lokal untuk menjalankan layanan basis data PostgreSQL sebagai sistem manajemen database utama. PostgreSQL berperan dalam menyimpan data hasil pengukuran sensor stasiun cuaca serta data pendukung lainnya secara terstruktur. Pemanfaatan Laragon membantu proses pengembangan sistem menjadi lebih efisien karena seluruh layanan seperti back-end, database PostgreSQL, dan pengujian dashboard dapat dijalankan dalam satu lingkungan terintegrasi sebelum sistem diimplementasikan secara penuh.

2.4.6 Visual Studio Code

Visual Studio Code (VS Code) merupakan editor kode open source yang dikembangkan oleh Microsoft dan dapat dijalankan pada berbagai sistem operasi, seperti Windows, Linux, dan macOS. VS Code mendukung banyak bahasa pemrograman, antara lain C++, Java, Python, PHP, dan lainnya. Editor ini mampu mengenali jenis bahasa pemrograman yang digunakan serta menyediakan fitur pewarnaan sintaks untuk mempermudah proses penulisan kode. Selain itu, VS Code memiliki integrasi dengan GitHub serta menyediakan berbagai ekstensi yang dapat ditambahkan sesuai kebutuhan pengembang [36].

Dalam penelitian ini, VS Code digunakan sebagai editor utama dalam pengembangan sistem berbasis Laravel. VS Code dipilih karena mendukung integrasi dengan database, Python, serta berbagai library machine learning. Selain

itu, fitur debugging dan integrasi dengan Git membantu dalam pengelolaan kode agar lebih terstruktur dan efisien.

2.4.7 Arduino IDE

Arduino IDE merupakan perangkat lunak yang digunakan untuk menulis, mengompilasi, serta mengunggah program ke papan mikrokontroler seperti Arduino dan ESP32. Melalui Arduino IDE, pengguna dapat melakukan penulisan dan modifikasi kode secara langsung sebelum dikirimkan ke perangkat yang digunakan. Arduino IDE terdiri dari beberapa komponen utama, yaitu editor program sebagai tempat penulisan kode, compiler untuk mengubah kode sumber menjadi kode biner, serta uploader yang berfungsi mengirimkan kode biner ke memori mikrokontroler [37]. Dalam penelitian ini, Arduino IDE digunakan sebagai lingkungan pengembangan utama untuk memprogram modul LoRa32 V2.1_1.6 baik sebagai transmitter maupun receiver. Arduino IDE mendukung berbagai library tambahan yang dibutuhkan dalam sistem stasiun cuaca berbasis LoRa, serta menyediakan fitur serial monitor yang berguna untuk proses *debugging*, pengujian komunikasi, dan pemantauan data secara langsung.

2.4.8 XGBoost

XGBoost (*Extreme Gradient Boosting*) merupakan salah satu algoritma *machine learning* modern yang banyak digunakan karena memiliki kemampuan prediksi yang tinggi serta tingkat interpretabilitas yang cukup baik, terutama pada data bertipe tabular. Berbeda dengan model *black-box* yang sulit dijelaskan, XGBoost menggunakan sistem keputusan berbasis pohon yang bersifat rekursif, sehingga proses pengambilan keputusan dapat dianalisis melalui struktur cabang yang terbentuk. Tingkat kepentingan setiap fitur pada XGBoost juga dapat dievaluasi berdasarkan seberapa sering fitur tersebut digunakan dalam pembentukan node pada pohon keputusan [38]. Secara konsep, XGBoost termasuk ke dalam metode ensemble learning berbasis strategi *boosting*, dimana beberapa decision tree digunakan sebagai weak learner yang kemudian digabungkan untuk membentuk satu model yang lebih kuat (*strong learner*) [39]. Proses pembentukan pohon dilakukan secara bertahap dengan tujuan memperbaiki kesalahan dari model sebelumnya, sehingga performa prediksi meningkat secara iteratif. Selain itu,

XGBoost menerapkan mekanisme regularisasi untuk mengontrol kompleksitas model, sehingga mampu mengurangi risiko terjadinya overfitting [7].

Pada penelitian ini, XGBoost digunakan sebagai algoritma untuk melakukan prediksi data cuaca dalam bentuk nilai numerik, yang selanjutnya hasil prediksi tersebut dimanfaatkan sebagai *input* pada proses klasifikasi kondisi cuaca.

2.4.9 PostgreSQL

PostgreSQL adalah sistem manajemen basis data yang awalnya dikembangkan oleh Berkeley Computer Science Department, University of California. PostgreSQL termasuk ke dalam jenis *Object-Relational Database Management System* (ORDBMS), yaitu basis data yang menggabungkan konsep relasional dengan objek serta mendukung penggunaan bahasa SQL dan PostgreSQL (Procedural Language) [40]. PostgreSQL menerapkan arsitektur *client-server*, di mana client berperan sebagai pihak yang mengirimkan permintaan data, sedangkan server bertugas melayani permintaan tersebut. Setiap permintaan yang diterima akan diproses oleh server melalui mekanisme terpisah, sehingga aktivitas antar pengguna tidak saling mengganggu. Server utama atau *postmaster* berfungsi mengatur koneksi, sementara proses back-end menangani eksekusi kueri dan pemrosesan data. Dengan mekanisme ini, PostgreSQL mampu melayani banyak pengguna secara bersamaan dengan tetap menjaga keamanan dan konsistensi data. Pada sistem ini, PostgreSQL dipilih sebagai basis data karena bersifat open source, stabil, serta memiliki performa yang baik untuk kebutuhan pencatatan data secara real-time.

2.5 Statistik Deskriptif

Statistik deskriptif merupakan metode statistik yang digunakan untuk menganalisis data dengan cara mendeskripsikan atau menggambarkan data yang telah terkumpul tanpa bermaksud melakukan generalisasi terhadap populasi yang lebih luas. Analisis ini bertujuan untuk menyajikan serta menggambarkan karakteristik data sehingga pola dan kecenderungan nilai dapat dipahami dengan lebih jelas. Penyajian statistik deskriptif dapat berupa ukuran pemusatan dan ukuran penyebaran data seperti mean, median, nilai minimum, nilai maksimum, serta standar deviasi. Dalam penelitian ini, analisis statistik deskriptif digunakan untuk

menjelaskan karakteristik sebaran data parameter cuaca sebelum dilakukan tahapan analisis selanjutnya. Melalui analisis ini, pola kecenderungan dan variasi data dapat dipahami secara kuantitatif tanpa melakukan generalisasi terhadap populasi yang lebih luas. Statistik deskriptif dimanfaatkan untuk menganalisis data AWS Undip guna memperoleh rentang nilai parameter lokal yang dijadikan sebagai acuan dalam proses pelatihan model klasifikasi cuaca. Parameter statistik yang digunakan dalam penelitian ini meliputi mean, nilai minimum, nilai maksimum, median, dan standar deviasi.

Mean merupakan nilai rata-rata dari seluruh data yang diperoleh dan digunakan untuk menggambarkan kecenderungan pusat data. Secara matematis Mean dapat ditulis dalam Persamaan (2.2).

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.2)$$

Nilai minimum merupakan nilai terkecil dalam suatu kumpulan data, sedangkan nilai maksimum merupakan nilai terbesar. Kedua nilai ini digunakan untuk mengetahui rentang variasi data secara keseluruhan. Secara matematis nilai min dan max masing-masing dapat ditulis dalam Persamaan (2.3) dan Persamaan (2.4).

$$x_{min} = \min(x_1, x_2, \dots, x_n) \quad (2.3)$$

$$x_{max} = \max(x_1, x_2, \dots, x_n) \quad (2.4)$$

Median adalah nilai tengah dari data yang telah diurutkan dari yang terkecil hingga terbesar. Median digunakan untuk menggambarkan pusat data yang tidak terlalu dipengaruhi oleh nilai ekstrem (outlier). Jika jumlah data ganjil, median adalah nilai di posisi tengah. Jika jumlah data genap, maka median dihitung dengan persamaan (2.5).

$$Median = \frac{x_{n+1}}{2} \quad (2.5)$$

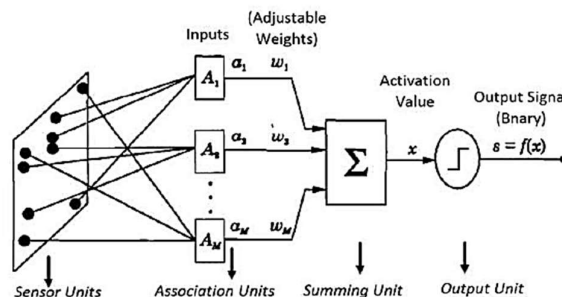
Standar deviasi merupakan ukuran penyebaran data yang menunjukkan seberapa jauh nilai-nilai data menyimpang dari rata-ratanya. Standar deviasi digunakan untuk mengetahui tingkat variasi atau dispersi data terhadap nilai rata-ratanya. Secara matematis Standar Deviasi dapat ditulis dalam Persamaan (2.6).

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.6)$$

2.6 Artificial Neural Network - Multilayer Perceptron

Jaringan Syaraf Tiruan atau *Artificial Neural Network* (ANN) merupakan suatu metode komputasi yang dibangun berdasarkan sejumlah unit pemrosesan yang saling terhubung satu sama lain. Jaringan ini terdiri dari kumpulan neuron yang menghubungkan data masukan (*input*) dengan keluaran (*output*) dan dirancang untuk meniru cara kerja otak manusia dalam menganalisis serta memproses informasi. ANN termasuk ke dalam sistem komputasi cerdas yang memiliki kemampuan pembelajaran dari data sehingga banyak digunakan untuk menyelesaikan permasalahan seperti klasifikasi, prediksi, dan pengenalan pola [41].

Struktur ANN tersusun atas neuron-neuron yang saling terhubung melalui hubungan yang disebut sinaps. Gambar 2.18 menunjukkan diagram model neuron. Setiap neuron menerima sinyal masukan yang kemudian diproses menggunakan bobot (*weight*), bias atau nilai ambang (*threshold*), serta fungsi aktivasi untuk menghasilkan keluaran tertentu. Bobot menentukan seberapa besar pengaruh suatu masukan terhadap neuron, sedangkan fungsi aktivasi menentukan apakah sinyal akan diteruskan atau tidak berdasarkan nilai aktivasi yang dihasilkan [41].

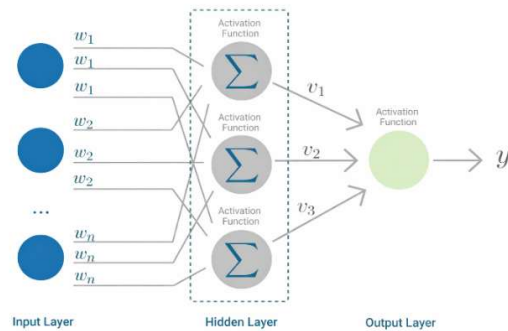


Gambar 2. 18 Diagram Model Neuron

Secara umum, neuron-neuron dalam ANN diorganisasikan ke dalam beberapa lapisan (*layer*), yaitu *input layer*, *hidden layer*, dan *output layer*. *Input layer* berfungsi menerima data awal, *hidden layer* berperan dalam melakukan

pemrosesan dan ekstraksi pola dari data, sedangkan *output layer* menghasilkan keluaran akhir dari jaringan. Keberadaan *hidden layer* memungkinkan jaringan memodelkan hubungan data yang bersifat kompleks dan tidak linier [41].

Salah satu bentuk ANN yang paling banyak digunakan adalah *Multilayer Perceptron* (MLP). MLP merupakan jaringan syaraf tiruan dengan arsitektur *feedforward* yang terdiri dari satu *input layer*, satu atau lebih *hidden layer*, dan satu *output layer*. Aliran data pada MLP hanya bergerak satu arah dari *input* menuju *output* tanpa adanya umpan balik langsung, sehingga termasuk ke dalam jaringan *feedforward*. Model ini sangat efektif untuk menangani permasalahan klasifikasi dan prediksi yang tidak dapat dipisahkan secara linier [42].



Gambar 2. 19 Diagram *Multi Layer Perceptron*

Gambar 2.19 menunjukkan pada *Multilayer Perceptron*, setiap neuron pada *hidden layer* menerima *input* dari seluruh neuron pada *layer* sebelumnya. Nilai *input* tersebut dikalikan dengan bobot masing-masing koneksi dan dijumlahkan dengan bias untuk menghasilkan nilai aktivasi. Proses MLP dapat dinyatakan secara matematis pada Persamaan (2.7).

$$z = \sum_{i=1}^n w_i x_i + b \quad (2.7)$$

di mana x_i merupakan nilai *input*, w_i merupakan bobot, b adalah bias, dan z merupakan hasil penjumlahan berbobot yang selanjutnya diproses oleh fungsi aktivasi untuk menghasilkan *output* neuron. Fungsi aktivasi memberikan sifat nonlinier pada jaringan sehingga MLP mampu mempelajari hubungan data yang kompleks [42]. Pada lapisan *output* digunakan fungsi aktivasi softmax untuk

menghasilkan nilai probabilitas pada setiap kelas keluaran. Fungsi ini memungkinkan model menentukan kelas dengan peluang tertinggi sebagai hasil klasifikasi. Softmax umum digunakan pada permasalahan klasifikasi multi-kelas dan dapat diterapkan pada berbagai arsitektur jaringan saraf, termasuk MLP. Secara matematis fungsi softmax dapat dinyatakan dengan Persamaan (2.8).

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (2.8)$$

Proses pembelajaran pada MLP bertujuan untuk memperoleh bobot dan bias yang menghasilkan kesalahan paling kecil antara keluaran jaringan dan target sebenarnya. Oleh karena itu, MLP umumnya dilatih menggunakan algoritma pembelajaran terawasi dengan metode *feedforward* dan *Backpropagation*. *Feedforward* digunakan untuk menghitung keluaran jaringan, sedangkan *Backpropagation* digunakan untuk memperbarui bobot berdasarkan nilai *error* agar performa model semakin optimal [43]. Dengan struktur *multilayer* yang dimilikinya, MLP menjadi salah satu model ANN yang paling praktis dan banyak digunakan dalam berbagai aplikasi seperti prediksi cuaca, klasifikasi citra, pengenalan pola, serta sistem pendukung keputusan karena kemampuannya dalam memodelkan hubungan data yang kompleks secara nonlinier [44].

2.6.1 *Feed forward Back Propagation*

Feed forward Back Propagation merupakan metode pembelajaran yang umum digunakan pada jaringan saraf tiruan, khususnya pada model MLP. Metode ini menggabungkan proses *feed forward* untuk menghitung keluaran jaringan dan *Backpropagation* untuk memperbarui bobot berdasarkan kesalahan yang dihasilkan. *Backpropagation* termasuk algoritma *supervised learning* karena proses pelatihannya menggunakan pasangan data *input* dan target keluaran. Tujuan utama algoritma *Backpropagation* adalah memperoleh kemampuan jaringan dalam mengenali pola data latih sekaligus melakukan generalisasi terhadap data baru. Dengan demikian, jaringan tidak hanya menghafal data, tetapi mampu memberikan prediksi yang tepat pada data yang belum pernah dilihat sebelumnya [45].

Secara umum, algoritma *feed forward Backpropagation* terdiri dari inisialisasi parameter, proses *feed forward*, perhitungan *error*, proses

Backpropagation, dan iterasi hingga diperoleh *error* minimum. Pada tahap *feed forward*, data mengalir dari *input layer* menuju *output layer* melalui *hidden layer*. Pada tahap *Backpropagation*, *error* disebarkan kembali untuk memperbarui bobot menggunakan learning rate agar kesalahan semakin kecil [45], [46]. Secara matematis, proses *feed forward* dirumuskan pada Persamaan (2.9) dan (2.10).

$$z_j = \sum_{i=1}^n w_{ij} x_i + b_j \quad (2.9)$$

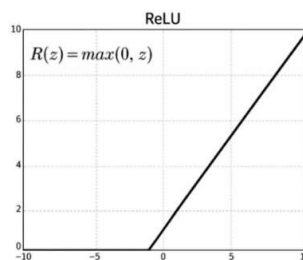
Secara matematis, proses pembaruan bobot pada *Backpropagation* dirumuskan seperti Persamaan (2.10)

$$w_{new} = w_{old} - \alpha \frac{\partial E}{\partial w} \quad (2.10)$$

Pada proses ini *feed forward* dan *Backpropagation* dilakukan secara iteratif hingga jaringan mencapai kondisi konvergen dengan *error* minimum [42], [43].

2.6.2 Fungsi Aktivasi Rectified Linear Unit (ReLU)

Rectified Linear Unit (ReLU) merupakan salah satu fungsi aktivasi yang paling banyak digunakan dalam jaringan syaraf tiruan modern karena memiliki bentuk matematis yang sederhana dan efisien secara komputasi. Fungsi ReLU berperan penting dalam memperkenalkan sifat non-linear ke dalam jaringan sehingga jaringan tidak hanya mampu mempelajari hubungan linier, tetapi juga pola yang lebih kompleks dalam data. Gambar 2.20 menunjukkan bahwa ReLU bekerja dengan cara meneruskan nilai *input* secara langsung apabila bernilai positif, dan mengubahnya menjadi nol apabila bernilai negatif, sehingga neuron hanya akan aktif ketika menerima sinyal di atas ambang tertentu [47], [48].



Gambar 2. 20 Fungsi Aktivasi ReLU

Secara matematis, fungsi ReLU dapat dirumuskan dengan persamaan (2.11).

$$f(x) = \max(0, x) \quad (2.11)$$

yang berarti bahwa jika nilai *input* x bernilai positif maka *output* akan sama dengan nilai *input* tersebut, sedangkan jika nilai *input* x bernilai negatif atau sama dengan nol maka *output* akan berubah menjadi nol. Sifat ini menyebabkan ReLU memiliki mekanisme pemotongan (*thresholding*) yang sederhana namun efektif dalam menentukan neuron mana yang aktif dan mana yang tidak aktif dalam proses propagasi sinyal [47], [48]. Dalam jaringan saraf, keluaran setiap neuron dengan ReLU dihitung menggunakan persamaan pada Persamaan (2.12).

$$y_j = f(z_j) \quad (2.12)$$

di mana x_i merupakan nilai *input*, w_i adalah bobot sinaptik, b adalah bias, dan f adalah fungsi ReLU. Hasil perhitungan ini menunjukkan bahwa ReLU tidak berdiri sendiri, tetapi bekerja sebagai bagian dari keseluruhan proses *feed forward*, di mana keluaran dari suatu *layer* akan menjadi masukan bagi *layer* berikutnya sampai menghasilkan keluaran akhir pada *output layer* [47].

ReLU memiliki keunggulan dibandingkan fungsi aktivasi konvensional seperti sigmoid dan tanh, terutama dari sisi efisiensi komputasi karena tidak melibatkan operasi eksponensial, serta kemampuannya dalam mengurangi permasalahan vanishing gradient. Selain itu, ReLU menghasilkan representasi sparse karena banyak neuron dapat bernilai nol, yang berkontribusi pada stabilitas pembelajaran dan percepatan konvergensi proses pelatihan pada jaringan saraf modern [49].

2.6.3 Adam Optimizer

Optimizer merupakan komponen penting dalam proses pelatihan jaringan saraf tiruan karena berfungsi memperbarui bobot jaringan agar nilai kesalahan (*loss*) dapat diminimalkan. Berbagai algoritma *optimizer* telah dikembangkan, seperti Stochastic Gradient Descent (SGD), AdaGrad, RMSProp, Adam, AdamW, dan AdaMax, yang masing-masing memiliki strategi berbeda dalam pengaturan *learning rate* dan pembaruan bobot. Di antara metode tersebut, Adam menjadi salah satu *optimizer* yang paling banyak digunakan karena menggabungkan keunggulan metode momentum dan *adaptive learning rate*, sehingga mampu meningkatkan stabilitas dan efisiensi proses pelatihan. [50], [51].

Adam (*Adaptive Moment Estimation*) bekerja dengan memanfaatkan dua momen statistik gradien, yaitu momen pertama dan momen kedua, yang dihitung menggunakan Persamaan (2.13) dan (2.14).

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (2.13)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (2.14)$$

di mana g_t merupakan gradien pada iterasi ke- t , sedangkan β_1 dan β_2 adalah parameter peluruhan momen. Karena nilai awal momen diinisialisasi nol, dilakukan proses bias correction untuk menghilangkan kecenderungan nilai menuju nol sehingga estimasi momen menjadi lebih akurat sebelum digunakan dalam pembaruan bobot [52]. Untuk mengatasi bias pada iterasi awal, dilakukan *bias correction* yang dirumuskan pada Persamaan (2.15).

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.15)$$

Selanjutnya, parameter diperbarui menggunakan aturan pembaruan yang ditunjukkan pada Persamaan (2.16).

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (2.16)$$

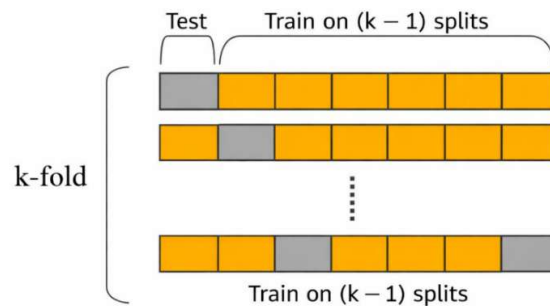
Pendekatan ini memungkinkan Adam menyesuaikan pembaruan bobot secara adaptif, mengurangi osilasi gradien, serta mempercepat konvergensi, sehingga cocok digunakan pada berbagai permasalahan klasifikasi dan prediksi berbasis jaringan saraf [51], [52].

2.7 Evaluasi Model Klasifikasi

Evaluasi model klasifikasi merupakan tahap penting dalam penelitian *machine learning* untuk menilai kemampuan model dalam melakukan prediksi pada data yang belum pernah dilihat sebelumnya. Evaluasi bertujuan tidak hanya mengukur tingkat akurasi, tetapi juga menganalisis kesalahan prediksi yang terjadi. Pada penelitian klasifikasi, evaluasi umumnya dilakukan menggunakan teknik *cross-validation* serta metrik berbasis *Confusion Matrix* seperti *accuracy*, *precision*, *recall*, dan *F1-score*, sehingga performa model dapat dinilai secara menyeluruh baik secara global maupun per kelas. [53], [54].

2.7.1 K-Fold Cross Validation

K-Fold Cross Validation merupakan metode resampling yang paling umum digunakan untuk mengevaluasi performa model klasifikasi, khususnya ketika jumlah data relatif terbatas. Pada metode ini, dataset dibagi secara acak menjadi k bagian (fold) yang kurang lebih memiliki ukuran yang sama. Setiap fold secara bergantian digunakan sebagai data uji (*testing set*), sedangkan $k-1$ fold lainnya digunakan sebagai data latih (*training set*). Gambar 2.21 menunjukkan proses yang dilakukan sebanyak k kali hingga seluruh data pernah digunakan sebagai data uji, kemudian hasil evaluasi dari seluruh iterasi dirata-ratakan untuk memperoleh estimasi performa model yang lebih stabil dan representatif [53], [54].



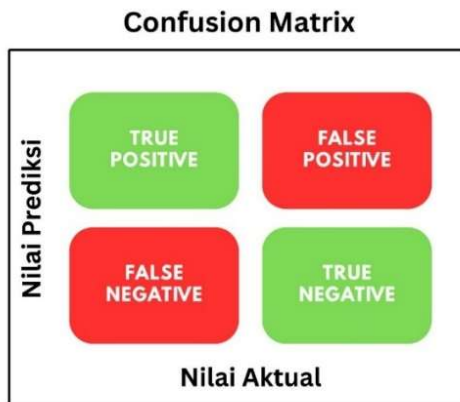
Gambar 2. 21 *K-Fold Cross Validation*

K-Fold Cross Validation memiliki keunggulan dibandingkan metode hold-out karena mampu memanfaatkan seluruh data secara optimal serta memberikan estimasi performa yang lebih akurat. Namun demikian, pada dataset berukuran kecil, hasil estimasi k -fold dapat memiliki variansi yang cukup tinggi karena bergantung pada proses pembagian data secara acak. Oleh karena itu, beberapa penelitian menyarankan pengulangan proses k -fold beberapa kali untuk mengurangi variansi, meskipun hal ini meningkatkan beban komputasi. Secara umum, nilai k yang paling sering digunakan dalam praktik adalah 5 atau 10 karena memberikan keseimbangan antara akurasi estimasi dan efisiensi komputasi [55], [56].

2.7.2 Confusion Matrix

Confusion Matrix merupakan alat utama dalam evaluasi model klasifikasi terawasi (supervised classification) yang digunakan untuk membandingkan label

aktual dengan hasil prediksi model. Matriks ini menyajikan informasi secara rinci mengenai jumlah prediksi benar dan salah pada setiap kelas, sehingga memungkinkan analisis kesalahan model secara lebih mendalam. Dengan *Confusion Matrix*, peneliti dapat mengetahui jenis kesalahan yang paling sering terjadi, misalnya apakah model cenderung salah mengklasifikasikan suatu kelas sebagai kelas lain. Informasi ini sangat penting dalam proses perbaikan desain dan optimasi model klasifikasi [57], [58].



Gambar 2. 22 *Confusion Matrix*

Gambar 2.22 menunjukkan empat komponen umum *Confusion Matrix* yang terdiri dari *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). TP menunjukkan jumlah data yang diklasifikasikan dengan benar sebagai kelas positif, TN menunjukkan jumlah data yang diklasifikasikan dengan benar sebagai kelas negatif, FP menunjukkan kesalahan prediksi positif, sedangkan FN menunjukkan kesalahan prediksi negatif. Keempat komponen ini menjadi dasar perhitungan metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *F1-score* [57], [59].

2.7.3 Metrik Evaluasi

Accuracy merupakan metrik evaluasi yang mengukur proporsi prediksi yang benar terhadap seluruh jumlah data. Metrik ini memberikan gambaran umum mengenai performa model, namun pada kondisi data tidak seimbang (*imbalanced data*) *accuracy* sering kali kurang representatif karena model dapat terlihat akurat

hanya dengan memprediksi kelas mayoritas [43], [58]. *Accuracy* dapat dirumuskan secara matematis dan dihitung menggunakan Persamaan (2.17)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.17)$$

Precision merupakan metrik yang mengukur ketepatan prediksi kelas positif, yaitu perbandingan antara jumlah prediksi benar positif terhadap seluruh prediksi positif yang dilakukan oleh model. *Precision* sangat penting ketika kesalahan false positive harus diminimalkan, misalnya pada sistem diagnosis atau sistem peringatan dini [42], [58]. *Precision* dirumuskan dengan persamaan (2.18).

$$Precision = \frac{TP}{TP + FP} \quad (2.18)$$

Recall merupakan metrik yang mengukur kemampuan model dalam menemukan seluruh data positif yang sebenarnya. *Recall* sering disebut sebagai sensitivitas dan sangat penting ketika kesalahan *false negative* lebih berbahaya dibandingkan kesalahan lainnya, misalnya pada deteksi penyakit atau bencana [42], [58]. *Recall* dapat ditulis dalam bentuk matematis pada Persamaan (2.19).

$$Recall = \frac{TP}{TP + FN} \quad (2.19)$$

F1-Score merupakan metrik gabungan yang mengombinasikan *precision* dan *recall* dalam satu nilai harmonik. Metrik ini digunakan untuk memberikan penilaian yang lebih seimbang terhadap performa model, terutama pada dataset dengan distribusi kelas tidak merata. Dengan menggunakan F1-Score, model tidak hanya dinilai dari seberapa banyak prediksi yang benar, tetapi juga dari keseimbangan antara ketepatan dan kelengkapan prediksi [57]. *F1-Score* dirumuskan dengan persamaan (2.20).

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.20)$$