PPNS POLITEKNIK PERKAPALAN NEGERI SURABAYA

TUGAS AKHIR (AE43250)

SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST

Ajiib Dzaki Al Khairi
NRP. 0922040003

DOSEN PEMBIMBING:
MUSTIKA KURNIA MAYANGSARI, S.Kom., M.Tr.Kom.
ADIANTO, S.T., M.T.

PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA
2026

TUGAS AKHIR (AE43250)

SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST

Ajiib Dzaki Al Khairi
NRP. 0922040003

DOSEN PEMBIMBING:
MUSTIKA KURNIA MAYANGSARI, S.Kom., M.Tr.Kom.
ADIANTO, S.T., M.T.

PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA
2026

"Halaman Sengaja Dikosongkan"

LEMBAR PENGESAHAN
TUGAS AKHIR

SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST

Disusun Oleh:
Ajiib Dzaki Al Khairi
0922040003

Diajukan Untuk Memenuhi Salah Satu Syarat Kelulusan
Program Studi Teknik Otomasi
Jurusan Teknik Kelistrikan Kapal
POLITEKNIK PERKAPALAN NEGERI SURABAYA

Disetujui oleh Tim penguji Tugas Akhir Tanggal Ujian : 27 Juli 2026
Periode Wisuda : Oktober 2026

Menyetujui,

Dosen Penguji	NIDN/NUPTK	Tanda Tangan
1. Mustika Kurnia Mayangsari, S.Kom., M.Tr.Kom.	(0835774675230272)	(.....)
2. Ir. Joko Endrasmono, M.T.	(0009096402)	(.....)
3. Zindhu Maulana Ahmad Putra, S.ST., M.Tr.T.	(0023099403)	(.....)
4. Novia Anggraini, S.Pd., M.Pd.	(5755771672230352)	(.....)

Dosen Pembimbing	NIDN/NUPTK	Tanda Tangan
1. Mustika Kurnia Mayangsari, S.Kom., M.Tr.Kom.	(0835774675230272)	(.....)
2. Adianto, S.T., M.T.	(0002077704)	(.....)

Menyetujui
Ketua Jurusan,

KEMENTERIAN PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI
POLITEKNIK PERKAPALAN NEGERI SURABAYA

Isa Rachman, S.T., M.T.
NIP. 198008162008121001

Mengetahui
Koordinator Program Studi,

Agus Khumaidi, S.ST., M.T.
NIP. 199308172020121004

“Halaman ini sengaja dikosongkan”

PPNS	**PERNYATAAN BEBAS PLAGIAT**	No. : F.WD I. 021 Date : 3 Nopember 2015 Rev. : 01 Page : 1 dari 1

Yang bertandatangan dibawah ini :

Nama : Ajiib Dzaki Al Khairi

NRP. : 0922040003

Jurusan/Prodi : Teknik Kelistrikan Kapal / D4 Teknik Otomasi

Dengan ini menyatakan dengan sesungguhnya bahwa :

Tugas Akhir yang akan saya kerjakan dengan judul :

SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST

Adalah **benar karya saya sendiri** dan **bukan plagiat dari karya orang lain.**

Apabila dikemudian hari terbukti terdapat plagiat dalam karya ilmiah tersebut, maka saya bersedia menerima **sanksi** sesuai ketentuan peraturan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Surabaya, 10 Agustus 2026

Yang membuat pernyataan,

(Ajiib Dzaki Al Khairi)

NRP. 0922040003

“Halaman ini sengaja dikosongkan”

KATA PENGANTAR

Dengan memanjatkan puji syukur ke hadirat Allah SWT atas segala rahmat, hidayah, serta karunia-Nya, penulis dapat menyelesaikan Tugas Akhir yang berjudul **"SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST"**. Tugas Akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Terapan pada Program Studi D4 Teknik Otomasi, Jurusan Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya.

Dalam proses penyusunan Tugas Akhir ini, penulis menghadapi berbagai tantangan dan kendala yang memerlukan ketekunan serta usaha yang maksimal. Penulis menyadari bahwa Tugas Akhir ini masih memiliki keterbatasan dan belum sepenuhnya sempurna. Namun, keberhasilan penyelesaian Tugas Akhir ini tidak lepas dari doa, dukungan, bimbingan, dan bantuan dari berbagai pihak. Melalui kesempatan ini penulis menyampaikan rasa hormat dan ucapan terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah memberikan kontribusi, khususnya kepada:

1. Allah SWT yang senantiasa melimpahkan rahmat, hidayah dan pertolongan-Nya sehingga penulis diberikan kemudahan, kesehatan, serta kelancaran dalam menyelesaikan penyusunan Tugas Akhir ini.
2. Kedua orang tua tercinta, Bapak Moh. Imam Syafi'i dan Ibu Helmi Fitriati, yang selalu memberikan kasih sayang, doa, dan dukungan. Terima kasih atas setiap doa yang menembus langit, setiap rupiah yang dikeluarkan demi pendidikan penulis, setiap pengorbanan yang dilakukan tanpa pernah mengharap balasan, serta kasih sayang yang selalu mengiringi setiap langkah. Seluruh pengorbanan tersebut menjadi sumber semangat dan motivasi terbesar bagi penulis dalam menyelesaikan pendidikan hingga memperoleh gelar sarjana. Semoga Allah SWT senantiasa melimpahkan kesehatan, kebahagiaan, dan umur yang penuh keberkahan kepada Bapak dan Ibu.
3. Kakak dan kedua adik, Mas Ari Agustan, Mbak Amanina Azizah Qurrota A'yun, Muh. Kayyis Rosyidul Arkan, dan Zain Ahmad Ayyasy Ulwan.

Terima kasih atas segala dukungan, bantuan, dan doa yang selalu diberikan sehingga penulis mampu menyelesaikan studi dengan baik.

4. Bapak Rachmad Tri Soelistijono, S.T., M.T. selaku Direktur Politeknik Perkapalan Negeri Surabaya.
5. Bapak Isa Rachman, S.T., M.T. selaku Ketua Jurusan Teknik Kelistrikan Kapal Politeknik Perkapalan Negeri Surabaya.
6. Bapak Agus Khumaidi, S.ST., M.T. selaku Koordinator Program Studi D4 Teknik Otomasi Politeknik Perkapalan Negeri Surabaya.
7. Ibu Mustika Kurnia Mayangsari, S.Kom., M.Tr.Kom. selaku Dosen Pembimbing I yang dengan penuh kesabaran telah memberikan bimbingan, arahan, motivasi, solusi, serta berbagai ilmu dan pengalaman selama proses penyusunan Tugas Akhir ini.
8. Bapak Adianto, S.T., M.T. selaku Dosen Pembimbing II yang telah memberikan bimbingan, masukan, koreksi, motivasi, serta dukungan sehingga penulis dapat menyelesaikan penelitian dan penyusunan Tugas Akhir ini.
9. Bapak Ryan Yudha Adhitya, S.ST., M.T. selaku Koordinator Tugas Akhir Program Studi Teknik Otomasi, yang telah memberikan arahan selama proses pelaksanaan Tugas Akhir.
10. Seluruh dosen, staf, dan tenaga kependidikan Politeknik Perkapalan Negeri Surabaya yang telah memberikan pelayanan dan dukungan selama penulis menjalani masa studi.
11. Daffa Rizqi Nanda Firmansyah dan Lailatul Robiyah yang senantiasa memberikan bantuan, dukungan, dan semangat, serta menemani penulis dalam menghadapi berbagai proses hingga penyelesaian Tugas Akhir ini.
12. Seluruh pihak yang tidak dapat penulis sebutkan satu per satu, atas kontribusi dan peran yang telah diberikan dalam mendukung kelancaran penyusunan Tugas Akhir ini.

Semoga Allah SWT senantiasa melimpahkan rahmat-Nya dan membalas segala kebaikan seluruh pihak yang telah memberikan bantuan, dukungan, serta doa dengan pahala yang berlipat ganda. Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna dan memiliki beberapa kekurangan. Oleh karena itu,

kritik dan saran yang bersifat membangun sangat diharapkan sebagai bahan evaluasi dan perbaikan guna menyempurnakan karya ini di masa mendatang.

Penulis

(Ajiib Dzaki Al Khairi)

“Halaman ini sengaja dikosongkan”

SISTEM PERINGATAN DINI RISIKO KUALITAS AIR TAMBAK UDANG VANAME BERDASARKAN *DISSOLVED OXYGEN* MENGGUNAKAN RANDOM FOREST

Ajiib Dzaki Al Khairi

ABSTRAK

Budidaya udang vaname merupakan sektor perikanan strategis yang berkontribusi besar terhadap perekonomian dan ekspor di Indonesia. Udang tercatat sebagai komoditas penyumbang ekspor perikanan terbesar, dengan kontribusi mencapai 34,5% dari total ekspor perikanan nasional. Data Badan Pusat Statistik (BPS) menunjukkan bahwa nilai ekspor udang Indonesia tumbuh rata-rata sebesar 4,61% per tahun selama periode 2017–2022. Namun, sebagian besar tambak masih dikelola secara tradisional sehingga hasil produksi sulit diprediksi dan rentan mengalami kegagalan akibat perubahan kualitas air. Parameter kualitas air seperti suhu, pH, *Total Dissolved Solids* (TDS), dan terutama *Dissolved Oxygen* (DO) memiliki peran penting dalam menentukan keberhasilan budidaya. Meskipun data kualitas air dapat diperoleh melalui teknologi *Internet of Things* (IoT), pemanfaatannya sering kali belum optimal sehingga informasi kondisi tambak tidak tersaji secara komprehensif. Sistem ini dikembangkan untuk mengklasifikasikan kualitas air ke dalam tingkat risiko tertentu berdasarkan nilai *Dissolved Oxygen* (DO) virtual, sedangkan parameter suhu dan pH digunakan sebagai fitur masukan pada model Random Forest. Pengujian menunjukkan model Random Forest memperoleh nilai akurasi sebesar 97,42% dengan nilai *macro average precision* sebesar 96,72%, *recall* 96,41%, dan *F1-score* 96,55%. Model diimplementasikan pada sistem *backend* untuk klasifikasi otomatis, *dashboard website*, dan notifikasi Telegram, namun masih memerlukan penyempurnaan untuk meningkatkan kemampuan generalisasi pada data baru.

Kata kunci: Budidaya Udang Vaname, *Internet of Things* (IoT), Klasifikasi Kualitas Air, Peringatan Dini, Random Forest.

“Halaman ini sengaja dikosongkan”

EARLY WARNING SYSTEM FOR WATER QUALITY RISK IN VANNAMEI SHRIMP PONDS BASED ON DISSOLVED OXYGEN USING RANDOM FOREST

Ajiib Dzaki Al Khairi

ABSTRACT

Vannamei shrimp farming is a strategic sector of the fisheries industry that contributes significantly to Indonesia's economy and exports. Shrimp is the largest contributor to fisheries exports, accounting for 34.5% of total national fisheries exports. Data from the Central Bureau of Statistics (BPS) show that the export value of Indonesian shrimp grew at an average rate of 4.61% per year during the 2017–2022 period. However, most shrimp ponds are still managed traditionally, making production outcomes difficult to predict and vulnerable to failure due to changes in water quality. Water quality parameters such as temperature, pH, Total Dissolved Solids (TDS), and especially Dissolved Oxygen (DO) play an important role in determining the success of shrimp farming. Although water quality data can be obtained through Internet of Things (IoT) technology, its use is often suboptimal, resulting in pond condition information not being presented comprehensively. This system was developed to classify water quality into specific risk levels based on virtual Dissolved Oxygen (DO) values. At the same time, temperature and pH were used as input features for the Random Forest model. The test results showed that the Random Forest model achieved an accuracy of 97.42%, with a macro-average precision of 96.72%, a recall of 96.41%, and an F1-score of 96.55%. The model was implemented in the back-end system for automatic classification, dashboards website, and Telegram notifications. However, it still requires further improvement to enhance its generalization capability on new data.

Keywords: *Classification Water Quality, Early Warning System, Internet of Things (IoT), Random Forest, Vaname Shrimp Farming.*

“Halaman ini sengaja dikosongkan”

DAFTAR ISI

HALAMAN JUDUL .....Error! Bookmark not defined.

LEMBAR PENGESAHAN ..... **iii**

LEMBAR BEBAS PLAGIAT ..... **v**

KATA PENGANTAR ..... **vii**

ABSTRAK ..... **xi**

ABSTRACT ..... **xiii**

DAFTAR ISI ..... **xv**

DAFTAR TABEL ..... **xxi**

DAFTAR GAMBAR ..... **xxiii**

DAFTAR NOTASI ..... **xxvii**

BAB 1 PENDAHULUAN **1**

1.1 Latar Belakang 1

1.2 Rumusan Masalah 3

1.3 Tujuan Penelitian 4

1.4 Manfaat Penelitian 4

1.5 Batasan Masalah 5

BAB 2 TINJAUAN PUSTAKA **7**

2.1 Kajian Penelitian Terdahulu 7

2.2 Kajian Pustaka 10

2.2.1 Tambak Udang Vaname 10

2.2.2 Kualitas Air Tambak Udang 11

2.2.3 Parameter Kualitas Air Tambak Udang 12

2.2.4 Sistem Peringatan Dini 15

2.2.5 Penilaian Risiko Kualitas Air 15

2.2.6 Konsep Sensor *Dissolved Oxygen* Virtual 17

2.2.7 *Internet of Things* (IoT) 17
2.2.8 Random Forest 19
2.2.9 Evaluasi Ketetapan Klasifikasi 21
2.3 Komponen *Hardware* 23
2.3.1 ESP32 23
2.3.2 Sensor Suhu DS18B20 24
2.3.3 Sensor pH DFRobot V2 25
2.3.4 Sensor *Dissolved Oxygen* DFRobot 26
2.3.5 Sensor *Total Dissolved Solids* DFRobot 27
2.3.6 *Analog Signal Isolator* 28
2.3.7 Aerator 29
2.3.8 *Power Supply* 30
2.3.9 Kontaktor 31
2.3.10 LCD 20x4 31
2.3.11 *Solid State Relay* (SSR) 32
2.3.12 *Buzzer* 33
2.3.13 *Selector Switch* 33
2.3.14 *Pushbutton* 34
2.3.15 *Miniature Circuit Breaker* (MCB) 35
2.4 Komponen *Software* 36
2.4.1 Arduino IDE 36
2.4.2 Visual Studio Code 37
2.4.3 Google Colab 37
2.4.4 Next.js 38
2.4.5 PostgreSQL 39
2.4.6 Supabase 39

2.4.7 Python 40
2.4.8 Scikit-learn 41
2.4.9 Eagle 42
BAB 3 METODOLOGI PENELITIAN 43
3.1 Konsep Penelitian 43
3.1.1 Diagram Blok Sistem 43
3.1.2 Arsitektur Perencanaan Sistem 45
3.1.3 Konsep Sistem Peringatan Dini dan Klasifikasi Kualitas Air 46
3.2 Tahap Penelitian 47
3.2.1 Tahap Identifikasi Masalah 49
3.2.2 Analisa Kebutuhan Sistem 49
3.2.3 Desain dan Perancangan Sistem 50
3.2.4 Implementasi dan Pengujian 52
3.2.5 Analisa Data 52
3.2.6 Pembuatan Laporan 54
3.3 Perencanaan dan Desain *Hardware* 54
3.3.1 Perancangan *Hardware* Elektrik 54
3.3.2 Perancangan *Hardware* Mekanik 56
3.4 Perencanaan dan Desain *Software* 59
3.4.1 Perancangan Komunikasi ESP-NOW dan MQTT 59
3.4.2 Perancangan *Database* 60
3.4.3 Perancangan *Backend* 62
3.4.4 Perancangan *Dashboard Monitoring* 63
3.4.5 Konversi dan Pengolahan Data Sensor 63
3.5 Penerapan Model Klasifikasi Random Forest 68
3.5.1 Pembersihan Data dan Penanganan *Outlier* 69

3.5.2 Inisialisasi Model 71
3.5.3 Pelatihan Model 72
3.5.4 Proses Klasifikasi Kualitas Air 72
3.5.5 Implementasi Model Random Forest pada *Website* 73
3.6 Evaluasi Performa Model Klasifikasi 74
3.6.1 Evaluasi Akurasi Sensor 74
3.6.2 *Confusion Matrix* 74
3.6.3 *Accuracy, Precision, Recall, dan F1-Score* 75
BAB 4 HASIL DAN PEMBAHASAN 77
4.1 Pengujian Sensor dan Aktuator 77
4.1.1 Hasil Pengujian Sensor Suhu 77
4.1.2 Hasil Pengujian Sensor TDS 81
4.1.3 Hasil Pengujian Sensor pH 86
4.1.4 Hasil Pengujian Sensor DO 91
4.1.5 Hasil Pengujian LCD 20x4 95
4.1.6 Hasil Pengujian *Buzzer* 97
4.1.7 Hasil Pengujian Relay 98
4.1.8 Hasil Pengujian Aerator 100
4.1.9 Hasil Pengujian Kontaktor 102
4.2 Hasil Perancangan *Hardware* dan *Software* 104
4.2.1 Hasil Perancangan Elektrik 104
4.2.2 Hasil Perancangan Mekanik 108
4.2.3 Hasil Perancangan *Software* 111
4.3 Pengujian Metode Random Forest 115
4.3.1 Pengambilan *Dataset* Pengujian 115
4.3.2 Analisis Parameter TDS 118

4.3.3 Preprocessing Data 118
4.3.4 Pembentukan Kelas Risiko 121
4.3.5 Pembagian Data 122
4.3.6 Penerapan *Synthetic Minority Oversampling Technique* 123
4.3.7 Pembentukan Model Random Forest 124
4.3.8 Evaluasi Model 126
4.4 Implementasi Model Random Forest pada *Back-end* 128
4.4.1 Integrasi Model Random Forest 129
4.4.2 Pengujian Prediksi Model 130
4.4.3 Pengujian Implementasi Model pada *Back-end* 131
4.4.4 Pengujian Sistem *Monitoring* 133
4.4.5 Pengujian Notifikasi Peringatan Dini 135
BAB 5 KESIMPULAN DAN SARAN 137
5.1 Kesimpulan 137
5.2 Saran 138
DAFTAR PUSTAKA 139
LAMPIRAN 145
Lampiran 1. Surat Izin Pengambilan Data Penelitian 145
Lampiran 2. Data Hasil Klasifikasi 147
Lampiran 3. Biodata Mahasiswa 155

"Halaman ini sengaja dikosongkan"

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu 7
Tabel 2. 2 Tabel spesifikasi mikrokontroler ESP32..... 24
Tabel 2. 3 Tabel spesifikasi Sensor suhu DS18B20 25
Tabel 2. 4 Tabel spesifikasi sensor pH meter SKU SEN0169..... 26
Tabel 2. 5 Tabel spesifikasi sensor *Dissolved Oxygen* DFRobot..... 27
Tabel 2. 6 Tabel spesifikasi sensor TDS meter..... 28
Tabel 3. 1 Kategori Risiko kualitas Air 47
Tabel 3. 2 Struktur Tabel Data Sensor..... 61
Tabel 3. 3 Struktur Tabel Telegram *User* 61
Tabel 3. 4 Rentang Optimal Parameter Kualitas Air 70
Tabel 3. 5 Ilustrasi *Confusion Matrix* Tiga Kelas..... 75
Tabel 4. 1 Konfigurasi ESP32 dengan Sensor Suhu DS18B20 78
Tabel 4. 2 Data Hasil Pengujian Sensor Suhu DS18B20..... 78
Tabel 4. 3 Konfigurasi ESP32 dengan Sensor TDS..... 82
Tabel 4. 4 Data Hasil Pengujian Sensor TDS 83
Tabel 4. 5 Konfigurasi ESP32 dengan Sensor pH 87
Tabel 4. 6 Data Hasil Pengujian Sensor pH..... 88
Tabel 4. 7 Konfigurasi ESP32 dengan Sensor DO 92
Tabel 4. 8 Data Hasil Pengukuran Sensor DO..... 93
Tabel 4. 9 Konfigurasi ESP32 dengan LCD 20x4 96
Tabel 4. 10 Data Hasil Pengujian LCD 20x4..... 96
Tabel 4. 11 Konfigurasi ESP32 dengan *Buzzer* 97
Tabel 4. 12 Data Hasil Pengujian *Buzzer*..... 98
Tabel 4. 13 Konfigurasi ESP dengan SSR..... 99
Tabel 4. 14 Data Hasil Pengujian SSR..... 99
Tabel 4. 15 Data Hasil Pengujian Aerator 101
Tabel 4. 16 Data Hasil Pengujian Kontaktor 103
Tabel 4. 17 Data Hasil Pengujian *Hardware Minimum System*..... 106
Tabel 4. 18 Hasil Pengambilan Data Tambak Udang Vaname..... 116
Tabel 4. 19 Informasi Pengambilan Data Penelitian..... 119
Tabel 4. 20 Hasil Pembentukan Label Kualitas Air..... 121

Tabel 4. 21 Distribusi Data Hasil Pelabelan 122
Tabel 4. 22 Hasil Pembagian Data 122
Tabel 4. 23 Komposisi Data *Training* dan *Testing* 123
Tabel 4. 24 Hasil Evaluasi Model Random Forest 128
Tabel 4. 25 Hasil Pengujian Implementasi 132

DAFTAR GAMBAR

Gambar 2. 1 Tambak Udang Vaname 11
Gambar 2. 2 Arsitektur *Layer* IoT 18
Gambar 2. 3 Random Forest 19
Gambar 2. 4 Pinout ESP32 23
Gambar 2. 5 *Waterproof* DS18B20 *Temperature* Sensor 24
Gambar 2. 6 *Gravity Analog* pH Sensor V2 25
Gambar 2. 7 *Gravity Analog Dissolved Oxygen* Sensor 26
Gambar 2. 8 *Gravity Analog* TDS Sensor 27
Gambar 2. 9 *Analog Signal Isolator* 28
Gambar 2. 10 Aerator 29
Gambar 2. 11 *Power Supply* 30
Gambar 2. 12 Kontaktor 31
Gambar 2. 13 LCD 20x4 31
Gambar 2. 14 *Solid State Relay* 32
Gambar 2. 15 *Buzzer* 33
Gambar 2. 16 *Selector switch* 34
Gambar 2. 17 *Pushbutton* 34
Gambar 2. 18 *Miniature Circuit Breaker* 35
Gambar 2. 19 Arduino IDE 36
Gambar 2. 20 Visual Studio Code 37
Gambar 2. 21 Google Colab 38
Gambar 2. 22 Next.js 38
Gambar 2. 23 PostgreSQL 39
Gambar 2. 24 Supabase 40
Gambar 2. 25 Python 40
Gambar 2. 26 Scikit-learn 41
Gambar 2. 27 Eagle 42
Gambar 3. 1 Diagram Blok Sistem 43
Gambar 3. 2 Arsitektur Sistem Klasifikasi Kualitas Air 45
Gambar 3. 3 *Flowchart* Tahapan Penelitian 48
Gambar 3. 4 *Flowchart* Sistem Klasifikasi Kualitas Air 51

Gambar 3. 5 *Flowchart* Proses *Machine Learning* 53
Gambar 3. 6 Desain Rangkaian dan *Wiring* Sistem 55
Gambar 3. 7 Skematik *Minimum System* ESP32 55
Gambar 3. 8 Rancangan *Plant* Sistem Tambak Udang 56
Gambar 3. 9 Rancangan Desain Mekanik pada *Box* Sensor 57
Gambar 3. 10 Rancangan Desain Mekanik Panel Kontrol 57
Gambar 3. 11 Rancangan *Plant* Pengujian Sistem 58
Gambar 3. 12 Arsitektur Komunikasi ESP-NOW dan MQTT 59
Gambar 3. 13 Rancangan Desain *Dashboard* Web Sistem 63
Gambar 4. 1 *Wiring* ESP32 dengan Sensor Suhu DS18B20 78
Gambar 4. 2 *Wiring* ESP32 dengan Sensor TDS 82
Gambar 4. 3 *Wiring* ESP32 dengan Sensor pH 87
Gambar 4. 4 *Wiring* ESP32 dengan Sensor DO 92
Gambar 4. 5 *Wiring* ESP32 dengan LCD 20x4 95
Gambar 4. 6 *Wiring* ESP32 dengan *Buzzer* 97
Gambar 4. 7 *Wiring* ESP32 dengan SSR 99
Gambar 4. 8 *Wiring* pada Aerator 101
Gambar 4. 9 *Wiring* Kontaktor 103
Gambar 4. 10 Hasil Perancangan *Board Layout PCB* 105
Gambar 4. 11 Hasil Implementasi Rangkaian PCB 106
Gambar 4. 12 Modul dan Probe Sensor 108
Gambar 4. 13 *Wiring* Panel *Monitoring* Kualitas Air 109
Gambar 4. 14 *Wiring* Panel Kontrol Aerator 110
Gambar 4. 15 Prototipe *Plant* Tambak 111
Gambar 4. 16 Antarmuka *Profile* 112
Gambar 4. 17 *Dashboard Monitoring* 112
Gambar 4. 18 Tampilan Riwayat Data Sensor 113
Gambar 4. 19 Tampilan Visualisasi *Chart* 114
Gambar 4. 20 Pengujian Notifikasi pada Bot Telegram 114
Gambar 4. 21 Pengambilan Data Kualitas Air Tambak 116
Gambar 4. 22 Hubungan Parameter Kualitas Air dengan DO Virtual 120
Gambar 4. 23 Distribusi Data *Training* Sebelum dan Sesudah SMOTE 124

Gambar 4. 24 Visualisasi Salah Satu Hasil Pohon Keputusan..... 125
Gambar 4. 25 Hasil *Confusion Matrix* 126
Gambar 4. 26 Proses Pemuatan Model Random Forest..... 129
Gambar 4. 27 Hasil Pengujian Prediksi Model dengan API..... 130
Gambar 4. 28 Hasil Respon API pada Proses Prediksi..... 131
Gambar 4. 29 Pengujian Metode Random Forest di Tambak..... 133
Gambar 4. 30 Data Tersimpan pada *Database* 134
Gambar 4. 31 Tampilan Kualitas Air pada *Dashboard* 134
Gambar 4. 32 Tampilan Pesan Notifikasi 136

“Halaman ini sengaja dikosongkan”

DAFTAR NOTASI

Gini (S)	: Himpunan *dataset*
C	: Jumlah kelas
pi	: Proporsi sampel kelas ke-i
$Gini_{split}$	: Nilai *Gini* setelah pemisahan
ni	: Jumlah sampel pada *child node* ke-i
n	: Total jumlah sampel pada *node* induk
Gini (S_i)	: Nilai *Gini impurity* dari *child node* ke-*i*
H(x)	: Hasil klasifikasi akhir Random Forest
h_i	: Hasil prediksi dari pohon keputusan ke-i
x	: Sampel data yang diklasifikasi
T	: Jumlah pohon keputusan
TP	: *True Positive*
TN	: *True Negative*
FP	: *False Positive*
FN	: *False Negative*

"Halaman ini sengaja dikosongkan"

BAB 1
PENDAHULUAN

Bab ini menguraikan gambaran umum mengenai penelitian yang dilakukan, meliputi latar belakang penelitian, rumusan masalah, batasan penelitian, tujuan penelitian, dan manfaat penelitian. Masing-masing bagian diuraikan sebagai berikut.

1.1 Latar Belakang

Permintaan pasar terhadap produk perikanan yang terus meningkat mendorong pengembangan sistem produksi yang berkelanjutan. Dalam konteks tersebut, budidaya udang merupakan salah satu sektor perikanan yang memiliki peran strategis dalam mendukung perekonomian dan ketahanan pangan, baik di tingkat nasional maupun global. Kondisi tersebut mendorong berkembangnya berbagai peluang usaha di sektor perikanan, khususnya usaha tambak udang yang memiliki potensi produksi dan nilai ekonomi yang tinggi. Pada tahun 2012 pemerintah menetapkan target revitalisasi 82.870 hektar tambak udang di 22 kabupaten pantai utara Jawa, sebagai wujud industrialisasi budidaya perikanan yang diproyeksikan hasilkan 102.000 ton udang bernilai Rp5,89 triliun (Nadhif, 2016). Udang menjadi penyumbang ekspor perikanan terbesar nasional (34,5%). Data BPS (2023) dari Bea Cukai, KKP, dan BI menunjukkan nilai ekspor udang naik positif 4,61% per tahun selama 2017–2022 (Kementerian Kelautan dan Perikanan, 2023). Budidaya udang di Indonesia masih didominasi metode tradisional, mayoritas petambak mempertahankan pengelolaan manual sederhana yang bergantung pada siklus alam (Boa *et al*., 2023).

Berdasarkan hasil survey yang dilakukan dengan mitra yaitu di IBAP Banjar Kemuning, diketahui bahwa sistem pengelolaan kualitas air masih dilakukan dengan pencatatan manual dan tradisional. Pemantauan parameter kualitas air, khususnya *Dissolved Oxygen* (DO) masih bergantung pada pengukuran berkala, sehingga perubahan kondisi air sering terlambat terdeteksi. Kondisi tersebut sejalan dengan temuan Ananta *et al.*, (2025), tambak udang tradisional memiliki hasil produksi yang sulit diprediksi karena dipengaruhi banyak faktor yang tidak sepenuhnya terkontrol. Beragam faktor penyebabnya antara lain serangan penyakit,

keberadaan predator, dan perubahan kondisi lingkungan. Tingginya padat tebar pun belum tentu sejalan dengan peningkatan produksi, sehingga dalam pengelolaan yang sama hasil panen dapat sangat baik, sedang, bahkan hingga gagal. Kondisi ini diduga berkaitan dengan masih banyaknya petambak tradisional yang tidak melakukan penanganan kualitas air dan manajemen budidaya secara terkontrol dan hanya bergantung pada kondisi alam. Berdasarkan permasalahan tersebut, penelitian yang sedang dikembangkan dalam bidang sistem *monitoring* dan pengendalian kualitas air tambak dengan fokus pada pengembangan sistem peringatan dini risiko kualitas air berdasarkan parameter DO virtual berperan sebagai acuan dalam pembentukan label kualitas air, sedangkan parameter lainnya digunakan sebagai variabel prediktor (fitur) untuk membangun model klasifikasi.

Penelitian Caesar *et al.*, (2025) sebelumnya telah menerapkan pendekatan modern melalui manajemen kualitas air dan teknologi yang lebih terkontrol dengan memanfaatkan *Internet of Things* (IoT), pendekatan ini bertujuan untuk meningkatkan produktivitas tambak, menekan angka kematian udang, serta mengoptimalkan efisiensi operasional. Parameter kualitas air yang umum digunakan dalam budidaya udang vaname meliputi suhu, pH, dan DO. Parameter tersebut harus berada pada kisaran optimal agar pertumbuhan udang berlangsung maksimal (Gunawan *et al*., 2025). Data hasil pengukuran dari sensor-sensor tersebut dikirimkan secara *real-time* ke aplikasi berbasis web atau *mobile*, sehingga pembudidaya dapat mengetahui kondisi tambak dari mana saja dan mengambil keputusan serta tindakan pengelolaan dengan lebih cepat.

Meskipun sistem pemantauan kualitas air berbasis *Internet of Things* (IoT) telah mampu menyediakan data kualitas air tambak secara *real-time*, penelitian Caesar *et al.,* (2025) berfokus pada fungsi *monitoring* dan visualisasi data. Pendekatan tersebut belum memanfaatkan data historis untuk melakukan klasifikasi kualitas air maupun pengambilan keputusan secara otomatis. Oleh karena itu, diperlukan pengembangan sistem yang mengintegrasikan pendekatan *machine learning* sebagai metode klasifikasi kualitas air untuk menentukan status kondisi air secara otomatis guna menjaga parameter kualitas air tetap berada pada kondisi optimal dan meningkatkan efisiensi serta keberhasilan budidaya udang vaname.

Pengembangan sistem *monitoring* kualitas air tambak udang yang dilengkapi dengan klasifikasi otomatis memerlukan integrasi antara sensor, pemrosesan data dan metode *machine learning* dalam suatu arsitektur berbasis *Internet of Things* (IoT) yang dapat mengakuisisi, pengiriman, dan pengolahan data secara *real-time*. Kualitas air tambak dapat dinilai berdasarkan parameter fisika dan kimia seperti pH, suhu, TDS dan DO. Selanjutnya data hasil pengukuran digunakan untuk menghasilkan nilai DO virtual, yang kemudian dapat dimanfaatkan sebagai parameter utama oleh model *machine learning* untuk melakukan klasifikasi kualitas air secara otomatis, sedangkan parameter kualitas air lainnya digunakan sebagai variabel prediktor (fitur) untuk membangun model. Integrasi hasil klasifikasi dengan sistem notifikasi dan kontrol otomatis aerator diharapkan mampu meningkatkan pengambilan keputusan terhadap perubahan kualitas air dalam pengelolaan tambak udang.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang tersebut, maka rumusan masalah yang akan dibahas dalam tugas akhir adalah sebagai berikut.

1. Bagaimana merancang sistem *monitoring* yang dapat mengelola data parameter tambak udang untuk menggantikan sistem pencatatan manual dalam mendukung risiko kualitas air?
2. Bagaimana proses penerapan Random Forest menggunakan data sensor *Dissolved Oxygen* (DO) virtual sebagai parameter utama untuk menghasilkan klasifikasi kualitas air tambak udang yang mampu melakukan penilaian risiko kualitas air secara otomatis dan konsisten?
3. Bagaimana performa penerapan model klasifikasi menggunakan metode Random Forest dalam mengevaluasi kualitas air tambak udang secara otomatis?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dijelaskan, maka tujuan penelitian yang akan dibahas dalam tugas akhir adalah sebagai berikut.

1. Merancang sistem *monitoring* yang dapat mengelola data parameter tambak udang secara terstruktur sebagai pengganti sistem pencatatan manual dalam mendukung penilaian risiko kualitas air.
2. Menerapkan metode Random Forest dengan memanfaatkan data sensor DO virtual sebagai parameter utama untuk menghasilkan klasifikasi kualitas air tambak udang yang mampu melakukan penilaian risiko kualitas air secara otomatis dan konsisten.
3. Mengevaluasi dan menganalisis performa model klasifikasi yang digunakan pada tambak udang, meliputi aspek akurasi, efisiensi, dan kemampuan dalam pengambilan keputusan.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan kontribusi dan manfaat bagi penulis serta pihak-pihak terkait, antara lain sebagai berikut.

1. Meningkatkan pengetahuan dan pemahaman mengenai sistem klasifikasi kualitas air tambak udang yang mampu menilai mutu air secara otomatis.
2. Mengimplementasikan ilmu pengetahuan perkuliahan dalam pengembangan sistem klasifikasi kualitas air berbasis *machine learning*.
3. Memberikan inovasi teknologi untuk membantu pembudidaya udang dalam pemantauan kualitas air yang dapat digunakan untuk pengambilan keputusan pengelolaan tambak.
4. Menjadi solusi alternatif berupa sistem pendukung keputusan mutu air tambak melalui pemanfaatan data historis dan DO virtual.
5. Mendorong kreativitas mahasiswa dalam mengembangkan teknologi akuatik dan sistem pendukung keputusan.

1.5 Batasan Masalah

Berdasarkan ruang lingkup yang telah ditetapkan, tugas akhir ini dibatasi sebagai berikut.

1. Penelitian ini dilaksanakan pada satu lokasi tambak udang, yaitu di IBAP yang terletak di Desa Banjar Kemuning, Kecamatan Sedati, Kabupaten Sidoarjo.
2. Data kualitas air yang digunakan berupa parameter pH, suhu, *Total Dissolved Solids* (TDS), dan *Dissolved Oxygen* (DO).
3. Klasifikasi kualitas air dilakukan berdasarkan DO virtual sebagai dasar pembentukan kelas risiko yang terdiri atas tercemar ringan, tercemar sedang, tercemar berat.
4. Sistem kontrol otomatis hanya diterapkan pada aerator dan notifikasi peringatan dini.
5. Acuan baku mutu kualitas air hanya menggunakan PP Nomor 22 Tahun 2021.
6. Penelitian hanya menerapkan algoritma Random Forest pada sistem *monitoring* berbasis web yang diakses melalui jaringan lokal.

“Halaman ini sengaja dikosongkan”

BAB 2
TINJAUAN PUSTAKA

Bab ini menampilkan rangkuman konsep yang berkaitan dengan topik penelitian, seperti teori dasar, penelitian terdahulu, serta pendekatan yang relevan dengan penelitian mengenai kualitas air budidaya tambak udang. Pembahasan mencakup literatur ilmiah, metode, serta *software* dan *hardware* yang digunakan dalam penyelesaian Tugas Akhir ini.

2.1 Kajian Penelitian Terdahulu

Berikut ini merupakan kajian terhadap penelitian terdahulu yang relevan dan berkaitan dengan topik dalam tugas akhir ini.

Tabel 2. 1 Penelitian Terdahulu

No	Keterangan		
1.	**Judul**	:	Sistem Kontrol Kualitas Air Tambak Udang berbasis Fuzzy Logic (Akbar & Irawan, 2023).
	Metode	:	Penelitian mengembangkan sistem kontrol kualitas air tambak udang berbasis Fuzzy Logic untuk mengendalikan kondisi air secara otomatis berdasarkan parameter suhu, kekeruhan, dan pH. Sistem diuji menggunakan prototipe media wadah.
	Hasil Pembahasan	:	Sistem mampu melakukan pengendalian kualitas air tambak secara otomatis berdasarkan parameter suhu, kekeruhan dan pH. Metode Fuzzy Logic efektif digunakan sebagai sistem kontrol otomatis untuk menjaga kualitas air, namun pengujian hanya dilakukan pada prototipe sehingga belum merepresentasikan kondisi tambak udang yang sebenarnya.
	Keterkaitan	:	Penelitian ini sama-sama membahas sistem pemantauan dan pengendalian kualitas air tambak udang berbasis mikrokontroler.
	Perbedaan	:	Penelitian ini menggunakan Fuzzy Logic sebagai metode pengambilan keputusan dengan parameter yang diukur yaitu suhu, pH dan kekeruhan menggunakan prototype media wadah, sedangkan pada tugas akhir ini menggunakan Random Forest untuk klasifikasi risiko kualitas air berdasarkan parameter suhu, pH sebagai fitur dan DO virtual sebagai parameter utama dengan fokus sistem peringatan dini (*Early Warning Sistem*).

No	Keterangan		
2.	**Judul**	:	Implementasi IoT untuk *Early Warning System* (EWS) Menggunakan Metode DES Model Holt pada Tambak Udang Vaname (Alfiansyah *et al*., 2021).
	Metode	:	Penelitian mengembangkan sistem *Internet of Things* (IoT) berbasis *Early Warning System* (EWS) dengan metode *Double Exponential Smooting* (DES) Model Holt untuk memprediksi nilai suhu dan pH air tambak. Hasil prediksi digunakan sebagai dasar pemberian peringatan dini melalui notifikasi telegram ketika parameter diluar ambang batas.
	Hasil Pembahasan	:	Sistem mampu memprediksi suhu dan pH air tambak serta mengirimkan notifikasi secara otomatis ketika hasil prediksi menunjukkan diluar ambang batas aman. Model menghasilkan nilai MSE suhu sebesar 0.01954, MAPE suhu 9.0859%, MSE pH 0.000155, dan MAPE pH 0.828%. Penggunaan metode DES Model Holt cukup efektif untuk melakukan *forecasting* parameter kualitas air dan mendukung sistem peringatan dini. Namun, sistem hanya melakukan prediksi terhadap parameter suhu dan pH, tanpa melakukan klasifikasi tingkat risiko kualitas air secara keseluruhan.
	Keterkaitan	:	Penelitian ini memiliki kesamaan dengan tugas akhir yang dilakukan, yaitu mengembangkan *Early Warning System* (EWS) berbasis IoT untuk membantu pemantauan kualitas air tambak udang secara otomatis.
	Perbedaan	:	Penelitian ini menggunakan DES Model Holt untuk melakukan *forecasting* suhu dan pH sebagai dasar pemberian notifikasi, sedangkan tugas akhir ini menggunakan regresi linear untuk memperoleh DO virtual dan Random Forest untuk mengklasifikasikan tingkat risiko kualitas air dan peringatan dini didasarkan pada hasil DO virtual.
3.	**Judul**	:	Sistem *monitoring* dan Rekomendasi Kualitas Budidaya Ikan Nila dengan algoritma Random Forest (Wijaya *et al*., 2021).
	Metode	:	Penelitian mengembangkan sistem *monitoring* kualitas budidaya ikan nila secara otomatis dengan memanfaatkan sensor pH, suhu, dan kekeruhan yang dipadukan dengan algoritma Random Forest untuk menghasilkan rekomendasi kualitas air.
	Hasil Pembahasan	:	Algoritma Random Forest menghasilkan tingkat akurasi sebesar 94,44%, sehingga sistem mampu memberikan rekomendasi kualitas air secara andal dan mendukung pengambilan keputusan dalam

No	Keterangan		
			pengelolaan budidaya ikan nila. Ini menunjukkan bahwa algoritma Random Forest memiliki performa yang baik dalam mengklasifikasikan kondisi kualitas air berdasarkan data sensor.
	Keterkaitan	:	Penelitian memiliki kesamaan dengan tugas akhir ini, yaitu sama-sama menerapkan algoritma Random Forest untuk menganalisis kualitas air berdasarkan data historis pada sistem *monitoring* kualitas air tambak udang.
	Perbedaan	:	Penelitian ini berfokus pada budidaya ikan nila dengan parameter suhu, pH, dan kekeruhan dengan menghasilkan rekomendasi kualitas air, sedangkan tugas akhir ini diterapkan pada budidaya udang vaname dengan memanfaatkan parameter DO virtual sebagai dasar labeling atau parameter utama klasifikasi menggunakan Random Forest dan menghasilkan sistem peringatan dini.
4.	**Judul**	:	Pengukuran Kejenuhan Oksigen Terlarut pada Air menggunakan *Dissolved Oxygen* Sensor (Yuliantari *et al*., 2021).
	Metode		Penelitian melakukan pengukuran kadar *Dissolved Oxygen* (DO) menggunakan metode elektrokimia dengan Gravity Analog *Dissolved Oxygen* Sensor SKU SEN0237 yang dihubungkan dengan Arduino Nano. Pengukuran dilakukan dengan memperhatikan suhu dan salinitas untuk menentukan tingkat kejenuhan oksigen terlarut dalam air.
	Hasil Pembahasan	:	Sensor mampu mengukur kadar DO hingga mencapai kondisi jenuh. Nilai DO meningkat dari 8.25 mg/L pada awal pengukuran menjadi stabil di sekitar 14.21 mg/L, menunjukkan bahwa sensor dapat digunakan untuk pemantauan kualitas air pada aplikasi akuakultur maupun akuaponik. Selain itu, menunjukkan bahwa DO merupakan parameter penting untuk menentukan kualitas air karena berperan dalam proses respirasi, metabolisme, pertumbuhan dan reproduksi organisme akuatik.
	Keterkaitan	:	Penelitian ini memiliki kesamaan dengan tugas akhir yang dilakukan, yaitu sama-sama menggunakan parameter *Dissolved Oxygen* (DO) sebagai indikator utama dalam penilaian kualitas air pada sistem budidaya.
	Perbedaan	:	Penelitian ini berfokus pada pengukuran DO secara langsung menggunakan sensor fisik untuk mengetahui tingkat kejenuhan oksigen terlarut, sedangkan tugas akhir ini mengembangkan sensor virtual DO yang didapat menggunakan regresi linear yang

No	Keterangan		
			kemudian dimanfaatkan sebagai masukan Random Forest untuk klasifikasi risiko kualitas air dan sistem peringatan dini, sehingga dapat mengurangi ketergantungan pada sensor DO fisik.

(Sumber : Penulis)

Berdasarkan Tabel 2. 1, dapat diketahui bahwa penelitian terdahulu telah banyak membahas sistem *monitoring* kualitas air berbasis *Internet of Things* (IoT), metode Fuzzy Logic, *Double Exponential Smoothing* (DES), maupun Random Forest. Namun, belum ditemukan penelitian yang mengintegrasikan sensor DO virtual sebagai parameter utama dengan algoritma Random Forest untuk mengklasifikasikan risiko kualitas budidaya udang vaname dalam sistem peringatan dini. Pengembangan ini diharapkan dapat mendukung klasifikasi risiko kualitas air secara efektif serta membantu pengambilan keputusan penanganan secara tepat dengan sistem peringatan dini.

2.2 Kajian Pustaka

Kajian pustaka digunakan sebagai landasan teori yang mendukung penelitian tentang budidaya udang vaname dan penjelasan konsep dasar klasifikasi data yang digunakan untuk mengelompokkan kondisi kualitas air ke dalam kategori tertentu sebagai dasar perancangan dan implementasi sistem klasifikasi, serta komponen *hardware* dan *software* yang digunakan pada tugas akhir ini.

2.2.1 Tambak Udang Vaname

Dalam kegiatan budidaya perikanan, salah satu sistem yang banyak dikembangkan adalah tambak udang vaname. Tambak udang vaname adalah sistem budidaya akuatik yang digunakan untuk memelihara udang putih (*Litopenaeus vannamei*) dengan pengelolaan kualitas air sebagai faktor utama yang berdampak pada pertumbuhan, tingkat kelangsungan hidup, dan produktivitas udang tersebut.

Udang vaname merupakan jenis udang introduksi yang mulai dibudidayakan di Indonesia sejak awal tahun 2000-an sebagai komoditas ekonomis alternatif untuk menggantikan udang windu yang produksinya menurun akibat serangan penyakit serta perubahan kualitas lingkungan. Pengembangan udang vaname dinilai mampu membangkitkan kembali usaha pertambakan nasional

karena memiliki tingkat produksi yang tinggi dan prospek pasar yang luas, baik untuk kebutuhan dalam negeri maupun ekspor (Mansyur & Rangka, 2008).

Gambar 2. 1 Tambak Udang Vaname
(Sumber : Penulis)

Desain tambak seperti Gambar 2. 1 membantu petambak dalam mengendalikan kualitas air secara lebih efektif, karena pada tambak dengan volume yang relatif terbatas pengaturan parameter utama yaitu suhu, DO, pH, dan TDS dapat dilakukan dengan lebih terkendali. Pengelolaan kualitas air yang baik pada tambak udang vaname memiliki peran penting dalam mendukung laju pertumbuhan, meningkatkan tingkat kelangsungan hidup, serta memperkuat ketahanan udang terhadap serangan penyakit, sehingga diperlukan pemantauan kondisi kualitas air secara berkala dan berkelanjutan.

2.2.2 Kualitas Air Tambak Udang

Kondisi kualitas air menjadi salah satu faktor utama yang memengaruhi keberhasilan budidaya udang vaname. Selama fase pembesaran, kualitas air berperan penting dalam menyediakan lingkungan yang ideal yang mendukung pertumbuhan dan kesehatan udang. Kualitas air tambak udang ditentukan oleh kondisi kimia, fisika, dan biologi perairan. Parameter kualitas air tersebut harus dijaga dalam batas optimal agar dapat mendukung pertumbuhan dan kelangsungan hidup udang vaname (Shilman *et al*., 2025).

Hasil penelitian sebelumnya menunjukkan laju pertumbuhan harian udang vaname pada sistem budidaya intensif sangat dipengaruhi oleh kestabilan kondisi

kualitas air. Meskipun parameter fisika, kimia, dan biologi perairan mengalami variasi, pengelolaan dan pemantauan kualitas air yang dilakukan secara kontinu mampu menjaga lingkungan budidaya tetap mendukung pertumbuhan udang secara optimal (Halim *et al*., 2022). Perubahan kualitas air yang terjadi secara mendadak dan tidak terpantau dengan baik dapat menyebabkan stres pada udang, menurunkan nafsu makan, serta meningkatkan risiko terkena penyakit dan kematian massal, sehingga diperlukan sistem pemantauan kualitas air yang mampu mendeteksi perubahan kondisi air secara cepat dan akurat.

2.2.3 Parameter Kualitas Air Tambak Udang

Keberhasilan budidaya udang sangat dipengaruhi oleh kondisi lingkungan perairan tempat pembesaran udang. Parameter kualitas air tambak udang merupakan indikator penting yang digunakan untuk menilai kondisi lingkungan perairan dalam kelangsungan hidup udang. Setiap parameter kualitas air yang terjaga dalam kondisi stabil dan optimal secara tidak langsung dapat meningkatkan produktivitas hasil panen. Salah satu permasalahan utama dalam kegiatan pembesaran udang vaname adalah menurunnya produktivitas yang disebabkan oleh memburuknya kualitas air yang dapat mengganggu proses budidaya. Oleh karena itu, pengelolaan kualitas air yang optimal harus mengacu pada standar yang telah ditetapkan, seperti Standar Nasional Indonesia (Halim *et al*., 2022).

1. **Suhu**

   Suhu pada budidaya tambak udang akan memengaruhi pertumbuhan, *survival rate* (SR), konsumsi oksigen dan imunitas udang. Pada baku mutu air sungai kelas II disebutkan suhu diperbolehkan menyimpang maksimal pada *deviation* ±3°C dari suhu udara diatas permukaan air (Pemerintah Republik Indonesia, 2021). Mahmudi *et al*., (2020) mengatakan jika suhu terlalu rendah akan menurunkan metabolisme dan nafsu makan yang akan menyebabkan imunitas menurun dan rentan terkena penyakit, jika terlalu tinggi maka metabolisme cepat dan nafsu makan naik yang akan menyebabkan FCR. Hasil penelitian menunjukkan bahwa suhu perairan tambak udang vaname berkisar antara 29,5–31,7°C sehingga masih berada pada kisaran optimal untuk mendukung kehidupan dan pertumbuhan udang (Caesar *et al*., 2025).

Parameter suhu dengan *Dissolved Oxygen* (DO) berkorelasi negatif atau terbalik yang artinya peningkatan suhu air menyebabkan penurunan konsentrasi DO, karena kemampuan air dalam melarutkan oksigen akan semakin berkurang seiring dengan meningkatnya suhu (Sihite *et al*., 2023).

2. Potential of Hydrogen (pH)

pH berpengaruh pada reaksi kimia yang terjadi ditambak sehingga nilai pH perlu dikendalikan dalam rentang tertentu untuk menjaga kestabilan metabolisme dan kondisi fisiologi udang. Berdasarkan baku mutu air sungai kelas II disebutkan bahwa nilai pH air diperbolehkan pada rentang 6 sampai 9 (Pemerintah Republik Indonesia, 2021). Sukendar *et al*., (2025) menjelaskan bahwa perubahan nilai pH diketahui berdampak signifikan terhadap tingkat kelangsungan hidup, pertumbuhan, serta kemampuan reproduksi berbagai organisme perairan. Selain itu, studi sebelumnya melaporkan bahwa perubahan pH yang melebihi batas normal berpotensi mampu menghambat pertumbuhan dan dapat memicu stres pada udang (Yunarty *et al*., 2022). Nilai pH pada tambak udang vaname semi-intensif berkisar antara 7,71–7,89, yang masih berada dalam kisaran optimal untuk mendukung pertumbuhan dan kelangsungan hidup udang vaname (Putra *et al*., 2024).

Parameter pH dengan *Dissolved Oxygen* (DO) berkorelasi positif atau berbanding lurus yang artinya peningkatan konsentrasi DO diikuti oleh peningkatan nilai pH. Hal ini menunjukkan perairan yang memiliki kadar oksigen yang tinggi umumnya memiliki nilai pH yang lebih basa (Sihite *et al*., 2023).

3. *Dissolved Oxygen* (DO)

Dissolved Oxygen (DO) berperan pada respirasi atau pernapasan organisme air seperti udang, plankton dan mikroorganisme lainnya. Ideal DO pada tambak udang adalah tidak kurang dari 4 mg/l. Jika terlalu rendah dapat menyebabkan stres fisiologis, pertumbuhan lambat dikarenakan nafsu makan menurun dan dapat menyebabkan kematian massal. Berdasarkan baku mutu air sungai Kelas II, konsentrasi DO ditetapkan memiliki batas minimal sebesar 4

mg/L, yang menunjukkan bahwa nilai DO di bawah batas tersebut tidak memenuhi standar kualitas lingkungan perairan (Pemerintah Republik Indonesia, 2021).

Patkaew *et al*., (2024) menyatakan bahwa kadar DO merupakan salah satu parameter paling kritis dalam budidaya udang karena berpengaruh langsung pada kelangsungan hidup, pertumbuhan, dan konsumsi pakan organisme akuatik. Hasil penelitian menunjukkan nilai DO yang diukur pada tambak udang vaname berkisar antara 4,7–6,97 mg/L, yang masih mendekati kisaran optimal untuk mendukung kehidupan dan pertumbuhan udang vaname (Caesar *et al*., 2025).

4. Total Dissolved Solid (TDS)

Total Dissolved Solids (TDS) menunjukkan tingkat konsentrasi total zat padat terlarut dalam air, yang mencakup mineral, garam, logam berat, serta senyawa organik yang larut sepenuhnya. Jika terlalu rendah dapat menyebabkan gangguan pada proses osmoregulasi udang akibat pengaruhnya terhadap salinitas perairan tambak, sehingga rentan terinfeksi bakteri. kadar TDS yang terlalu tinggi dapat memicu stres osmotik pada udang, menurunkan konsentrasi oksigen terlarut, serta mendorong terjadinya blooming alga akibat peningkatan nutrien. Berdasarkan baku mutu air sungai Kelas II, nilai TDS yang diperbolehkan adalah kurang dari 1.000 mg/L, sehingga konsentrasi TDS yang melebihi batas tersebut menunjukkan kondisi perairan yang tidak memenuhi standar kualitas lingkungan (Pemerintah Republik Indonesia, 2021). Nilai TDS pada penelitian tambak udang vaname sebelumnya berkisar antara 582,83–587,17 ppm, yang menunjukkan bahwa kandungan padatan terlarut masih berada pada kisaran yang sesuai untuk budidaya udang vaname (Putra *et al*., 2024).

Parameter *Total Dissolved Solid* (TDS) dengan *Dissolved Oxygen* (DO) menunjukkan korelasi negatif atau berbanding terbalik yang artinya peningkatan konsentrasi TDS dalam perairan cenderung diikuti oleh penurunan kadar DO begitu pula sebaliknya, kadar DO yang tinggi dalam air dapat berkontribusi dalam menekan TDS pada perairan tersebut (Sa'idi, 2020).

2.2.4 Sistem Peringatan Dini

Dalam upaya mencegah terjadinya risiko akibat perubahan kondisi lingkungan perairan, diperlukan suatu sistem yang mampu memberikan informasi peringatan secara cepat dan akurat. Sistem peringatan dini (*Early Warning System/EWS*) merupakan suatu sistem yang dirancang untuk mendeteksi secara dini perubahan kondisi lingkungan yang berpotensi menimbulkan risiko. Pendekatan tersebut kini semakin banyak digunakan sebagai alternatif untuk sistem peringatan apabila terjadi penyimpangan parameter kualitas air dari kondisi optimal (Alfiansyah *et al*., 2021). Peringatan tersebut memungkinkan pembudidaya untuk melakukan tindakan secara cepat dan tepat, seperti penambahan aerasi, pengaturan pemberian pakan, atau pergantian air. Sistem peringatan dini terdiri dari proses pengambilan data, analisis kondisi, dan penyampaian peringatan kepada pengguna dengan demikian, penerapan sistem peringatan dini diharapkan dapat meningkatkan efektivitas pengelolaan tambak serta meminimalkan risiko kegagalan produksi.

2.2.5 Penilaian Risiko Kualitas Air

Penilaian terhadap potensi dampak lingkungan menjadi salah satu langkah penting dalam pengelolaan kualitas air pada sistem budidaya. Sistem penilaian risiko kualitas air merupakan proses untuk mengidentifikasi dan mengklasifikasi potensi dampak negatif kualitas air terhadap kesehatan dan kelangsungan hidup budidaya. Risiko diartikan sebagai peluang terjadinya gangguan fisiologis, penurunan pertumbuhan, hingga kematian udang akibat kondisi kualitas air yang tidak optimal.

Penilaian risiko dilakukan dengan menggunakan parameter kualitas air utama yaitu *Dissolved Oxygen* (DO) virtual, karena DO berperan penting dalam proses respirasi dan metabolisme udang, sedangkan parameter kualitas air lainnya digunakan sebagai variabel prediktor (fitur) dalam model karena berperan dalam memengaruhi ketersediaan DO dan merepresentasi kondisi lingkungan perairan. Integrasi ini memungkinkan model menangkap hubungan antarvariabel yang berkontribusi terhadap risiko bagi tambak udang vaname.

Berdasarkan pendekatan tersebut, nilai DO virtual digunakan sebagai dasar dalam pembentukan kategori risiko kualitas air. Penentuan kategori mengacu pada literatur budidaya udang vaname dan baku mutu kualitas air sehingga setiap kategori merepresentasikan tingkat risiko yang berbeda terhadap kondisi fisiologis dan kelangsungan hidup udang vaname.

1. **Kategori tercemar ringan (DO ≥ 5,0 mg/L)**

   Nilai DO ≥ 5 menunjukkan kondisi udang mampu beradaptasi dengan baik serta proses respirasi dan metabolisme berlangsung optimal. Pada kondisi ini, parameter kualitas air lainnya berada dalam kisaran optimal atau hanya mengalami sedikit penyimpangan, sehingga risiko terhadap gangguan fisiologis relatif rendah. Namun, perubahan parameter kualitas air lainnya tetap perlu dipantau karena dapat memengaruhi kondisi perairan sepenuhnya (Patkaew *et al*., 2024).

2. **Kategori tercemar sedang (DO ≥ 4,0 − < 5,0 mg/L)**

   Kondisi ini masih memungkinkan udang untuk hidup tetapi mengalami adanya penurunan efisiensi metabolisme dan peningkatan kerentanan terhadap penyakit. Pada kondisi ini, udang masih bertahan hidup, namun diperlukan pengelolaan kualitas air untuk mencegah penurunan menjadi kondisi yang lebih buruk.

3. **Kategori tercemar berat (DO < 4,0 mg/L)**

   Kondisi ini menunjukkan hipoksia berat yang berpotensi menyebabkan gangguan fisiologis serius pada udang, ditandai dengan stres akut, penurunan fungsi metabolisme dan sistem imun, serta peningkatan risiko kematian, terutama jika berlangsung dalam jangka waktu lama atau terjadi pada perairan yang terkontaminasi (Telaumbanua, 2025).

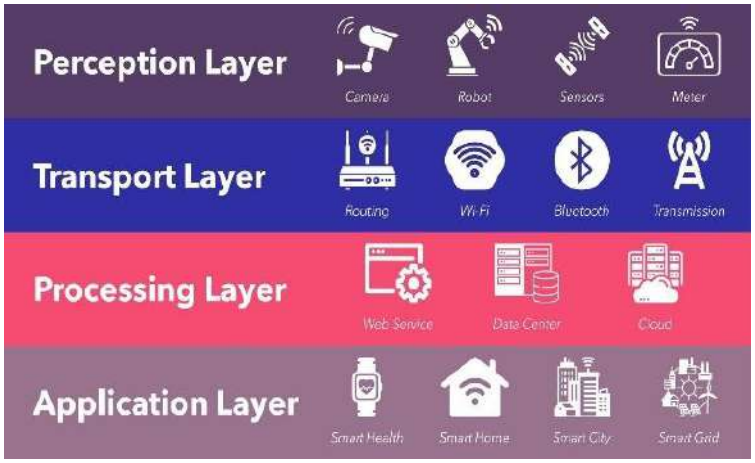
2.2.6 Konsep Sensor *Dissolved Oxygen* Virtual

Perkembangan teknologi pengolahan data dan pembelajaran mesin membuka peluang baru dalam pemantauan kualitas air secara lebih efisien. Sensor virtual merupakan model prediktif berbasis data yang memanfaatkan parameter sekunder atau parameter yang berkorelasi dan diukur dengan memperkirakan parameter yang sulit atau mahal diukur secara langsung, dengan memanfaatkan model *machine learning.* Penerapan sensor virtual dapat mengurangi biaya pengukuran, ketergantungan pada proses labolatorium yang tidak efisien, serta mendukung pemantauan kualitas air secara *real-time* atau mendekati *real-time,* khususnya untuk parameter yang memiliki hubungan nonlinier dengan parameter lainnya. Dalam konteks kualitas air, parameter *Dissolved Oxygen* (DO) dapat diprediksi menggunakan sensor virtual dengan memanfaatkan parameter kualitas air lain seperti suhu dan pH yang memiliki keterkaitan fisika dan kimia terhadap parameter *Dissolved Oxygen* (DO). Pendekatan ini menjadi solusi alternatif untuk menggantikan sensor DO fisik yang relatif mahal dan memerlukan perawatan khusus (Paepae *et al*., 2021).

Dari pemanfaatan data sensor virtual dapat digunakan langsung sebagai dasar klasifikasi kualitas air, dan pendekatan ini terbukti mampu membedakan kondisi kualitas air secara efektif ketika dipadukan dengan metode *machine learning* (Rozek & Lagod, 2025).

2.2.7 *Internet of Things* (IoT)

Konsep *Internet of Things* (IoT) memungkinkan integrasi antar perangkat fisik yang terhubung ke internet sehingga mampu melakukan pertukaran data serta pemrosesan informasi secara otomatis. Konsep ini relevan dalam pengembangan teknologi untuk meningkatkan efisiensi dan produktivitas dalam berbagai bidang, termasuk bidang perikanan. *Internet of Things* (IoT) digunakan sebagai teknologi pendukung utama dalam membangun sistem peringatan dini dan klasifikasi kualitas air pada tambak udang vaname. IoT berfungsi sebagai penghubung antara kondisi fisik lingkungan tambak dengan sistem pengolahan data berbasis digital secara *real-time* dan berkelanjutan.



Gambar 2. 2 Arsitektur *Layer* IoT
(Sumber : https://www.kaashivinfotech.com/blog/iot-architecture/)

Perception layer adalah lapisan awal pada arsitektur IoT yang berfungsi sebagai penangkap data dari lingkungan fisik. Lapisan ini terdiri dari sensor dan aktuator seperti suhu, kamera, RFID atau perangkat lainnya. Data yang dikumpulkan masih berupa data mentah dan menjadi dasar utama bagi seluruh proses IoT.

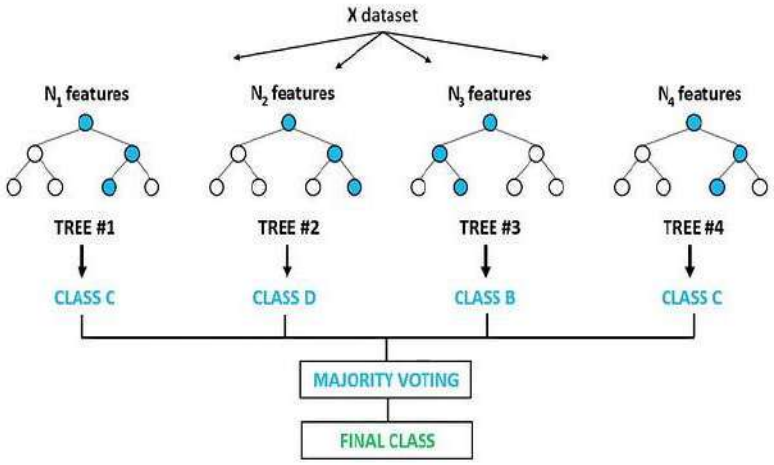
Transport Layer memiliki peran mengirimkan data yang telah dikumpulkan oleh *perception layer* ke *processing layer*. Lapisan ini menggunakan berbagai teknologi komunikasi dan jaringan seperti *Wi-Fi, Bluetooth, ZigBee, LoRa*, seluler, atau jaringan internet. Keandalan dan kecepatan pengiriman data pada *layer* ini sangat penting agar data dapat diterima secara *real-time* dan minim gangguan.

Processing Layer berfungsi untuk mengolah, menyimpan, dan menganalisis data yang diterima dari *transport layer.* Pada lapisan ini biasanya menggunakan *server*, *data center*, *cloud computing* dan layanan web untuk melakukan pemrosesan data, analisis, serta pengambilan keputusan otomatis. Hasil pengolahan data inilah yang akan menjadi informasi penting pagi pengguna atau sistem.

Application Layer merupakan lapisan akhir yang menyediakan layanan dan antarmuka langsung. Lapisan ini menampilkan hasil dari pemrosesan data dalam bentuk aplikasi atau sistem *dashboard* web. *Application Layer* membantu pengguna dapat memantau kondisi, menerima notifikasi, serta mengontrol perangkat IoT sesuai dengan kebutuhan dan tujuan penggunaan.

2.2.8 Random Forest

Dalam pengembangan model klasifikasi berbasis *machine learning*, pemilihan algoritma yang tepat menjadi faktor penting untuk memperoleh kinerja yang optimal. Algoritma hutan acak (Random Forest) dikategorikan sebagai metode *ensemble* pada pembelajaran mesin yang bekerja dengan menggabungkan beberapa pohon keputusan untuk menghasilkan model klasifikasi dan regresi yang lebih akurat. yang dilatih pada sampel data dan subset fitur yang berbeda secara acak. Proses klasifikasi pada Random Forest dilakukan dengan terlebih dahulu membagi data, kemudian memilih sampel secara acak untuk membangun setiap pohon keputusan. Setelah seluruh pohon terbentuk, masing-masing memberikan prediksi kelas untuk sampel tersebut, lalu dilakukan pemilihan berdasarkan suara terbanyak untuk menentukan kelas akhir yang diprediksi (G. A. D. S. A. Ningsih & Pramartha, 2024).



Gambar 2. 3 Random Forest
(Sumber : medium.com)

Metode Random Forest membentuk sekumpulan pohon keputusan yang dibangun secara acak. Penentuan kelas akhir dilakukan melalui proses klasifikasi berdasarkan kelas yang paling sering muncul (modus) dari seluruh pohon keputusan yang dihasilkan. Algoritma atau prosedur penyusunan Random Forest pada suatu himpunan data yang terdiri atas *n* observasi dan *p* variabel penjelas (prediktor) melibatkan beberapa tahapan dalam proses pembentukan model dan pendugaan (Rachmi, 2020).

Tahapan awal pada algoritma Random Forest adalah membentuk subset dari data pelatihan untuk setiap pohon keputusan. Dari *dataset* awal yang terdiri atas N sampel, akan dibangun sebanyak T pohon keputusan. Setiap pohon keputusan dikonstruksi menggunakan subset data yang dipilih secara acak, di mana satu sampel dapat terpilih lebih dari satu kali.

Setelah subset data diperoleh, setiap pohon keputusan kemudian dibangun untuk melakukan proses klasifikasi menggunakan algoritma Random Forest dengan kriteria *Gini Impurity*. Nilai *Gini Impurity* dihitung untuk suatu himpunan S yang memiliki c kelas.

$$Gini\ (S) = 1 - \sum_{i=1}^{c} p_i^2 \quad (2.1)$$

Dengan

S = Himpunan *dataset*

C = Jumlah kelas

pi = Proporsi sampel dari kelas ke-i

Nilai *Gini* mendekati 0 jika semua sampel dalam *node* termasuk dalam satu kelas (*node* murni atau *leaf node*), dan mendekati 0.5 atau lebih tinggi jika sampel dari berbagai kelas terdistribusi secara merata di dalam *node.* Setelah dilakukan *split* terhadap suatu *node* berdasarkan fitur tertentu, maka nilai rata-rata setelah *split* dapat dihitung menggunakan *Gini Split.*

$$Gini_{Split} = \sum_{i=1}^{c} \left(\frac{n_i}{n}\right) x\ Gini(Si) \quad (2.2)$$

Dengan

$Gini_{split}$ = Nilai Gini setelah pemisahan

n_i = jumlah sampel pada *child node* ke-i

n = Total jumlah sampel pada *node* induk

$Gini(S_i)$ = nilai *Gini impurity* dari *child node* ke-i

Atribut dengan nilai *Gini Split* paling kecil dipilih sebagai dasar pemisahan *node* karena menghasilkan pemisahan yang paling optimal dalam proses klasifikasi. Setelah seluruh pohon keputusan pada Random Forest terbentuk, hasil prediksi dari

masing-masing pohon digabungkan menggunakan mekanisme *majority voting*. Setiap pohon memberikan prediksi kelas terhadap suatu sampel, dan kelas yang memperoleh jumlah suara terbanyak dari keseluruhan pohon ditetapkan sebagai hasil prediksi akhir. Mekanisme *majority voting* tersebut dapat dirumuskan untuk memperjelas proses agregasi prediksi dari setiap pohon keputusan.

Secara matematis, apabila terdapat T pohon dalam Random Forest dan setiap pohon ke-i menghasilkan prediksi $h_i(x)$ untuk sampel x, maka prediksi akhir $H(x)$ ditentukan sebagai berikut.

$$H(x) = mode\ \{h_1(x), h_2(x), \dots, h_T(x)\} \quad (2.3)$$

Dengan

H(x) = Hasil klasifikasi akhir Random Forest

h_i = Hasil prediksi dari pohon keputusan ke-i

x = Sampel data yang diklasifikasi

T = Jumlah pohon keputusan

Mekanisme *majority voting* berperan dalam meningkatkan kinerja model dengan mengintegrasikan kontribusi dari sejumlah pohon keputusan yang dilatih menggunakan variasi data yang berbeda. Melalui kombinasi teknik *bagging*, pemilihan fitur secara acak, penggunaan *Gini Index*, serta penerapan *majority voting*, Random Forest membentuk suatu model klasifikasi yang memiliki tingkat akurasi tinggi dan kestabilan yang baik (Cholil, 2025).

Berdasarkan hasil pengujian sebelumnya, algoritma Random Forest menunjukkan performa paling optimal dibandingkan metode lainnya dengan capaian akurasi yang tinggi. Hal ini menunjukkan bahwa Random Forest merupakan metode yang sesuai untuk digunakan dalam klasifikasi kualitas air untuk menentukan risiko mutu perairan (L. Ningsih *et al*., 2024).

2.2.9 Evaluasi Ketetapan Klasifikasi

Evaluasi ketepatan klasifikasi dilakukan untuk mengetahui sejauh mana model yang dikembangkan mampu mengklasifikasikan data sesuai dengan kelas yang sebenarnya. *Confusion Matrix* merupakan teknik evaluasi dilakukan untuk menilai kinerja model klasifikasi dengan cara membandingkan hasil prediksi model

terhadap kelas aktual. Matriks ini menunjukkan jumlah prediksi benar dan salah yang diklasifikasikan ke dalam kategori *True Positive, True Negative, False Positive*, serta *False Negative*, sehingga memberikan gambaran yang jelas mengenai kemampuan model dalam melakukan klasifikasi. Berdasarkan *Confusion Matrix* memungkinkan perhitungan berbagai metrik evaluasi, termasuk *accuracy, precision, recall,* dan *F1-score*, guna menilai performa model secara komprehensif.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.4)$$

$$Precision = \frac{TP}{TP + FP} \quad (2.5)$$

$$Recall = \frac{TP}{TP + FN} \quad (2.6)$$

$$F1\text{-}Score = 2\,x\,\frac{Precision\ x\ Recall}{Precision + Recall} \quad (2.7)$$

Accuracy merupakan ukuran evaluasi yang digunakan untuk menilai tingkat ketepatan model klasifikasi secara umum. Nilai *accuracy* dihitung berdasarkan perbandingan antara jumlah prediksi yang sesuai dengan kelas sebenarnya terhadap keseluruhan data pengujian. Metrik ini menunjukkan seberapa baik model dalam mengklasifikasikan data secara umum. Namun, *accuracy* kurang merepresentasikan kinerja model apabila data yang digunakan memiliki distribusi kelas yang tidak seimbang.

Precision digunakan untuk mengukur tingkat ketepatan prediksi pada kelas positif. Metrik ini menggambarkan persentase data yang diklasifikasikan sebagai kelas positif dan benar-benar sesuai dengan kelas aktual. Nilai *precision* yang tinggi menunjukkan bahwa model mampu meminimalkan kesalahan berupa prediksi positif palsu (*false positive*) sehingga cocok digunakan pada sistem yang menuntut keakuratan tinggi dalam memberikan peringatan.

Recall merupakan metrik evaluasi yang digunakan untuk mengukur kemampuan model dalam mengenali seluruh data yang termasuk ke dalam kelas positif. Nilai *recall* yang tinggi menunjukkan bahwa model mampu mengurangi kesalahan berupa data positif yang gagal terdeteksi (*false negative*).

F1-score merupakan metrik evaluasi yang menggabungkan nilai *precision* dan *recall* dalam bentuk rata-rata. Metrik ini digunakan untuk menilai keseimbangan antara ketepatan prediksi dan kemampuan deteksi model. *F1-score* sangat sesuai digunakan pada permasalahan klasifikasi dengan data yang tidak seimbang, karena mampu memberikan gambaran kinerja model secara lebih komprehensif dibandingkan *accuracy* saja.

Berdasarkan metrik evaluasi tersebut, kinerja model klasifikasi kualitas air menggunakan *confusion matrix, accuracy, precision, recall, dan F1-score* untuk mengevaluasi kinerja model klasifikasi kualitas air. Pada sistem peringatan dini, metrik *recall dan F1-score* digunakan sebagai indikator utama karena merepresentasikan kemampuan model dalam mendeteksi kondisi berisiko secara akurat dan seimbang.

2.3 Komponen *Hardware*

Kebutuhan *hardware* dalam pengembangan dirancang sebagai sistem yang terintegrasi untuk mendukung proses pemantauan dan analisis kualitas air secara otomatis. *Hardware* yang digunakan berfungsi untuk melakukan *monitoring* data parameter kualitas air tambak secara *real-time*, pemrosesan data, serta mendukung proses klasifikasi kualitas air kedalam risiko tertentu.

2.3.1 ESP32

ESP32 memiliki sejumlah pin *General Purpose Input Output* (GPIO) yang dapat digunakan untuk menghubungkan sensor, aktuator, maupun modul komunikasi sesuai kebutuhan sistem. Berikut merupakan Gambar 2. 4 yang menunjukkan konfigurasi pin (*pinout*) dari chip ESP32.

Gambar 2. 4 Pinout ESP32
(Sumber : https://forum.arduino.cc/t/is-this-esp32-pinout-diagram-wrong-d1-mini)

Espressif Systems mengembangkan sebuah mikrokontroller yaitu ESP32 sebagai pengganti ESP8266. ESP32 memiliki kelebihan yaitu sudah dilengkapi *WiFi* dan *Bluetooth*, serta konsumsi daya yang rendah sehingga sangat cocok untuk mendukung pengembangan sistem *Internet of Things* (IoT) yang membutuhkan koneksi *wireless* (Pratama & Kiswantono, 2022). Adapun tabel beserta penjelasan spesifikasinya sebagai berikut.

Tabel 2. 2 Tabel spesifikasi mikrokontroler ESP32

No	Parameter	Spesifikasi
1	*Frequency Clock*	80 MHz hingga 240 MHz
2	Mikroprosesor	Xtensa Dual-Core 32 Bit LX6
3	Kapasitas SRAM	520 KB
4	Fitur Analog & PWM	ADC 12-bit (hingga 18 channel), DAC 2 channel, PWM hingga 16 channel
5	Kapasitas *memory flask*	4 MB
6	*Wi-Fi Transceiver*	11b/g/n
7	*Bluetooth*	Versi 4.2/BLE (Bluetooth Low Energy)
8	Tegangan Kerja	3.0 ~ 3.6 V
9	*Integrated Crystal*	40 MHz Crystal
10	GPIO	34 GPIO

(Sumber : Espressif Systems, 2026)

2.3.2 Sensor Suhu DS18B20

Sensor DS18B20 merupakan sensor suhu digital yang banyak digunakan karena memiliki tingkat akurasi yang baik, tahan terhadap air (*waterproof*), serta mudah diintegrasikan dengan mikrokontroler. Berikut merupakan Gambar 2. 5 yang menunjukkan bentuk fisik sensor suhu DS18B20.

Gambar 2. 5 *Waterproof* DS18B20 *Temperature* Sensor
(Sumber : https://www.dfrobot.com/product-689.html)

Dalam sistem *monitoring* kualitas air, sensor suhu berperan penting untuk mengukur suhu perairan sebagai salah satu parameter fisik yang memengaruhi kondisi lingkungan akuatik. Tingkat kecepatan dan keakuratan dalam mengukur suhu sensor ini memiliki kestabilan yang lebih tinggi dibandingkan dengan sensor suhu lainnya, sehingga sensor tersebut sangat sesuai dengan kebutuhan pada sistem (Ibrahim *et al*., 2023). Sensor DS18B20 menggunakan antarmuka *One-Wire* yang memungkinkan pengiriman data stabil dengan gangguan sinyal minimal. Adapun tabel beserta penjelasan spesifikasinya sebagai berikut.

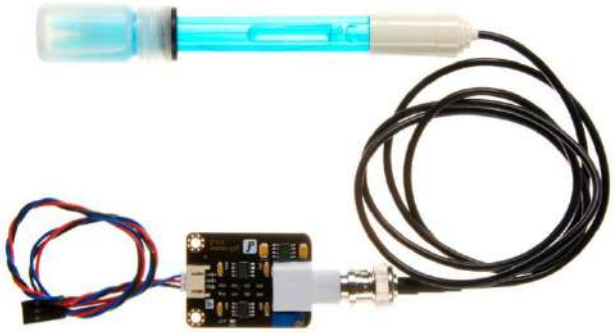
Tabel 2. 3 Tabel spesifikasi Sensor suhu DS18B20

No.	Parameter	Spesifikasi
1	*Usable power/data*	3.0V – 5.5V
2	*Accuracy*	±0.5°C *from* -10°C *to* +85°C
3	*Usable temperature range*	-55 to 125°C (-67°F *to* +257°F)
4	*Selectable resolution*	9 *to* 12bit
5	*Query time*	*less than* 750ms
6	*Diameter cable*	4mm
7	*Length*	90cm

(Sumber : DFRobot, 2025)

2.3.3 Sensor pH DFRobot V2

Sensor pH DFRobot V2 digunakan sebagai salah satu perangkat untuk mengukur derajat keasaman air pada sistem *monitoring* kualitas air. Berikut merupakan Gambar 2. 6 yang menunjukkan bentuk fisik sensor analog pH.



Gambar 2. 6 *Gravity Analog* pH Sensor V2
(Sumber : https://shopee.co.id/DFRobot-Analog-pH-Sensor-Meter-Kit-V2-i.39683264.6365749123)

Parameter pH merupakan salah satu indikator penting dalam menentukan kualitas air tambak. Sensor pH DFRobot Gravity pH Sensor V2 digunakan untuk

mengukur tingkat keasaman atau kebasaan air secara *real-time*. Sensor ini menghasilkan keluaran analog yang dapat diolah oleh mikrokontroler serta memiliki kestabilan dan akurasi yang baik setelah dilakukan kalibrasi. Oleh karena itu, sensor pH DFRobot V2 banyak digunakan pada sistem *monitoring* kualitas air karena mampu memberikan hasil pengukuran yang konsisten. Adapun spesifikasi sensor pH DFRobot V2 ditunjukkan pada tabel berikut.

Tabel 2. 4 Tabel spesifikasi sensor pH meter SKU SEN0169

No	Parameter	Spesifikasi
1	*Modul power*	3.3V – 5.5V
2	*Output Voltage*	0 – 3.0V
2	*Measuring Range*	0-14PH
3	*Measuring Temperature*	0-60 °C
4	*Respon time*	≤ 1min
5	*Measuring Accuracy*	±0.1@25℃
6	*Signal Connector*	PH2.0-3P

(Sumber : DFRobot, 2025)

2.3.4 Sensor *Dissolved Oxygen* DFRobot

Sensor *Dissolved Oxygen* (DO) DFRobot merupakan salah satu sensor yang digunakan dalam sistem *monitoring* kualitas air. Berikut merupakan Gambar 2. 7 yang menunjukkan bentuk fisik sensor Analog *Dissolved Oxygen* (DO).

Gambar 2. 7 *Gravity Analog Dissolved Oxygen* Sensor
(Sumber : https://www.dfrobot.com/product-1628.html)

Salah satu sensor yang digunakan dalam sistem ini adalah Sensor *Dissolved Oxygen* (DO) DFRobot. Sensor DO adalah sensor analog yang digunakan untuk mengukur kadar DO di dalam air (Forhad *et al*., 2024). Sensor ini berperan penting dalam menilai kelayakan air bagi organisme hidup karena kadar oksigen sangat mempengaruhi kehidupan akuatik. Sensor DO DFRobot mampu memberikan data pengukuran secara *real-time* dan mendukung sistem *monitoring* kualitas air secara

otomatis. Dengan karakteristik tersebut, sensor ini mampu mendukung proses pengukuran dalam sistem secara optimal. Adapun tabel beserta penjelasan spesifikasinya sebagai berikut.

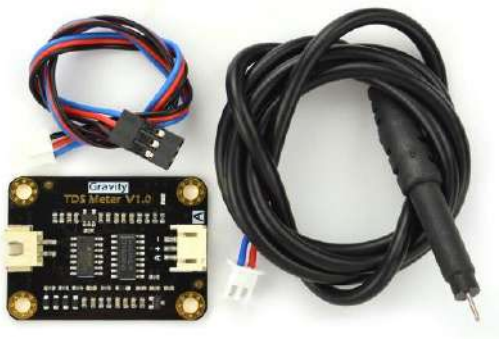
Tabel 2. 5 Tabel spesifikasi sensor *Dissolved Oxygen* DFRobot

No	Parameter	Spesifikasi
1	*Operating Voltage*	3.3V – 5.5V
2	*Output Voltage*	0 - 3V
3	*Detection range*	0~20mg/L
4	*Response Time*	Up to 98% *full response, within 90 seconds* (25℃)
5	*Pressure Range*	0-50PSI
6	*Cable Connector*	BNC
7	*Signal Connector*	Gravity Analog Interface (PH2.0-3P)
8	*Dimension*	42mm * 32mm

(Sumber : DFRobot, 2025)

2.3.5 Sensor *Total Dissolved Solids* DFRobot

Sensor *Total Dissolved Solids* (TDS) DFRobot digunakan untuk mengukur konsentrasi zat padat terlarut sebagai salah satu parameter kualitas air. Berikut merupakan Gambar 2. 8 yang menunjukkan bentuk fisik sensor Analog TDS Meter.



Gambar 2. 8 *Gravity Analog* TDS Sensor
(Sumber : https://www.dfrobot.com/product-1662.html)

Salah satu sensor yang dipakai dalam sistem ini adalah sensor TDS. Sensor *Total Dissolved Solids* (TDS) DFRobot merupakan sensor analog yang digunakan untuk mengetahui tingkat kandungan zat padat terlarut di dalam air (Forhad *et al*., 2024). Sensor ini menilai kualitas air berdasarkan nilai konduktivitas listrik yang dipengaruhi oleh jumlah mineral dan ion terlarut.

Dengan kemampuan memberikan hasil pengukuran yang stabil setelah dilakukannya kalibrasi, sensor TDS DFRobot berperan penting dalam pemantauan

kualitas air terutama pada sistem pengolahan air dan hidroponik. Adapun tabel beserta penjelasan spesifikasinya sebagai berikut.

Tabel 2. 6 Tabel spesifikasi sensor TDS meter

No	Parameter	Spesifikasi
1	*Input Voltage*	3.0V – 5.5V
2	*Output Voltage*	0 - 2.3V
3	*Current*	3 - 6mA
4	*TDS Measurement Range*	0 - 1000ppm
5	*TDS Measurement Accuracy*	± 10% F.S. (25 ℃)
6	*Module Size*	PH2.0-3P
7	*Electrode Interface*	XH2.54-2P

(Sumber : DFRobot, 2025)

2.3.6 *Analog Signal Isolator*

Analog signal isolator digunakan sebagai perangkat pendukung untuk mengisolasi sinyal analog sehingga mengurangi gangguan antar sensor pada sistem. Berikut merupakan Gambar 2. 9 yang menunjukkan bentuk fisik analog signal isolator.

Gambar 2. 9 *Analog Signal Isolator*
(Sumber : https://www.dfrobot.com/product-1621.html)

Dalam pembacaan sensor kualitas air tambak, beberapa sensor digunakan secara bersamaan untuk mengukur parameter. Penggunaan sensor dalam satu media air dapat menimbulkan gangguan antar sensor, terutama pada sensor yang menghasilkan atau menerima sinyal analog. Gangguan tersebut dapat menyebabkan noise, perbedaan potensial tegangan, maupun interfensi sinyal yang berdampak pada menurunnya akurasi hasil pengukuran. Perangkat pendukung berupa analog sinyal isolator dapat digunakan untuk meminimalkan gangguan yang terjadi selama proses pengukuran.

Analog signal isolator bekerja dengan mengisolasi jalur sinyal antara sensor dan mikrokontroler sehingga tidak terjadi hubungan listrik secara langsung antara kedua rangkaian. Dengan adanya isolasi tersebut, pengaruh interferensi dari sensor lain dapat dikurangi sehingga sinyal yang diterima oleh sistem menjadi lebih stabil. Penggunaan analog signal isolator membantu meningkatkan keakuratan dan konsistensi pembacaan sensor ketika beberapa sensor dioperasikan secara bersamaan dalam satu media air, sehingga data yang diperoleh lebih representatif terhadap kondisi kualitas air yang sebenarnya.

2.3.7 Aerator

Aerator digunakan sebagai perangkat untuk meningkatkan kadar *Dissolved Oxygen* (DO) di dalam air sehingga kondisi perairan tetap mendukung kehidupan udang. Berikut merupakan Gambar 2. 10 yang menunjukkan bentuk fisik Aerator.

Gambar 2. 10 Aerator
(Sumber : Penulis)

Dalam sistem budidaya perairan, diperlukan perangkat pendukung yang mampu menjaga kondisi lingkungan tetap optimal bagi organisme akuatik. Aerator adalah perangkat yang digunakan untuk meningkatkan kadar DO di dalam air (Mohan *et al*., 2022). Alat ini bekerja dengan cara mengaduk dan memercikkan air sehingga terjadi kontak antara air dan udara secara lebih intensif. Penggunaan aerator sangat penting dalam sistem perairan seperti tambak dan kolam budidaya karena mampu menjaga kualitas air, mendukung kehidupan organisme akuatik,

serta mencegah terjadinya kekurangan oksigen yang dapat menyebabkan kematian biota air.

Selain meningkatkan kadar DO, aerator juga berfungsi untuk menjaga sirkulasi air di tambak. Sirkulasi yang baik dapat mencegah terbentuknya zona air yang stagnan yang berpotensi menumpuk bahan organik dan limbah metabolisme organisme. Sirkulasi yang merata menciptakan lingkungan perairan yang lebih stabil dan mendukung kesehatan organisme akuatik.

2.3.8 *Power Supply*

Power supply digunakan sebagai sumber catu daya untuk menyuplai kebutuhan listrik pada seluruh komponen dalam sistem. Berikut merupakan Gambar 2. 11 yang menunjukkan bentuk fisik *Power supply*.

Gambar 2. 11 *Power Supply*
(Sumber : https://www.globalsources.com/product/waterproof-power-supply_1149937678f.htm)

Keandalan sistem elektronik sangat bergantung pada ketersediaan sumber daya listrik yang stabil dan sesuai dengan kebutuhan perangkat. *Power supply* merupakan perangkat yang berperan sebagai penyedia sumber daya listrik untuk menyuplai tegangan listrik ke seluruh rangkaian sistem (Sitohang *et al*., 2018). *Power supply* mengubah tegangan listrik AC dari sumber utama menjadi tegangan DC yang sesuai dengan kebutuhan komponen seperti mikrokontroler, sensor, dan aktuator. Oleh karena itu, pemilihan *power supply* disesuaikan dengan kebutuhan daya seluruh komponen agar sistem *monitoring* dan peringatan dini dapat beroperasi secara stabil dan berkelanjutan.

2.3.9 Kontaktor

Kontaktor digunakan sebagai perangkat saklar elektromagnetik untuk mengendalikan beban listrik berdaya besar pada sistem. Berikut merupakan Gambar 2. 12 yang menunjukkan bentuk fisik kontaktor.

Gambar 2. 12 Kontaktor
(Sumber : https://atstekno.com/cara-kerja-kontaktor-dan-jenis-jenisnya/)

Dalam sistem otomasi berbasis elektronik, kontaktor digunakan sebagai saklar elektromagnetik untuk mengendalikan beban listrik berarus dan bertegangan besar. Kontaktor bekerja menggunakan kumparan (*coil*) elektromagnet yang menghasilkan medan magnet saat diberi tegangan, sehingga kontak dapat menghubungkan atau memutus aliran listrik ke beban. Kontaktor memiliki keunggulan berupa kemampuan menangani arus tinggi, konstruksi yang kuat, serta dilengkapi kontak bantu untuk fungsi kontrol dan pengaman (Nurshofa *et al*., 2024).

2.3.10 LCD 20x4

LCD 20x4 digunakan untuk menampilkan nilai parameter kualitas air dan status sistem secara *real-time*. Berikut merupakan Gambar 2. 13 yang menunjukkan bentuk fisik LCD 20x4.

Gambar 2. 13 LCD 20x4
(Sumber : quartzcomponents.com)

Penyajian informasi hasil pemrosesan data secara visual menjadi aspek penting dalam sistem *monitoring* berbasis elektronik. LCD 20x4 *display* adalah perangkat keluaran (*output*) yang digunakan untuk menampilkan hasil pemrosesan data pada sistem yang dirancang. Layar LCD 20x4 mampu menyajikan informasi berupa nilai sensor, dan status sistem secara *real-time* dengan tampilan yang jelas dan kontras tinggi (Safrilly & Badarudin, 2024). Penggunaan LCD 20x4 ini memudahkan pengguna dalam melakukan *monitoring* kondisi sistem secara langsung tanpa memerlukan perangkat tambahan, sehingga meningkatkan efektivitas dan kemudahan operasional sistem.

2.3.11 *Solid State Relay* (SSR)

Solid State Relay (SSR) digunakan sebagai saklar elektronik untuk mengendalikan beban listrik secara otomatis tanpa menggunakan kontak mekanis. Berikut merupakan Gambar 2. 14 yang menunjukkan bentuk fisik SSR.

Gambar 2. 14 *Solid State Relay*
(Sumber : https://www.edukasielektronika.com/2023/05/pengertian-dan-penjelasan-tentang-solid-state-relay.html)

Dalam sistem otomasi elektronik, diperlukan pengendalian arus listrik yang cepat dan stabil. *Solid State Relay* (SSR) merupakan perangkat yang berfungsi sebagai saklar otomatis dengan memanfaatkan komponen semikonduktor untuk menghubungkan atau memutus aliran listrik pada beban bertegangan atau berarus besar (Mude *et al*., 2024). Dibandingkan relay elektromekanis, SSR lebih unggul karena tidak memiliki bagian mekanik, sehingga lebih tahan lama, tidak menimbulkan percikan, dan memiliki kecepatan *switching* yang lebih tinggi.

2.3.12 *Buzzer*

Buzzer digunakan sebagai perangkat keluaran (*output*) untuk memberikan peringatan suara ketika sistem mendeteksi kondisi tertentu. Berikut merupakan Gambar 2. 15 yang menunjukkan bentuk fisik *buzzer*.

Gambar 2. 15 *Buzzer*
(Sumber : https://www.blibli.com/p/alarm-buzzer-sfm-27-dc-3-24v-95-db-3105a-continuous/ps--ALA-47809-00466)

Pada sistem peringatan dini, penyampaian informasi secara cepat kepada pengguna menjadi aspek yang sangat penting. *Buzzer* merupakan perangkat keluaran (*output*) yang berfungsi sebagai indikator audio pada sistem elektronik untuk memberikan notifikasi berupa bunyi ketika terjadi kondisi atau peristiwa tertentu, seperti peringatan, kesalahan, atau perubahan status sistem (Sinuhaji, 2023). Dalam sistem ini, *buzzer* digunakan sebagai umpan balik suara agar pengguna dapat mengetahui respons sistem tanpa harus selalu memperhatikan tampilan visual. Selain itu, penggunaan *buzzer* meningkatkan efektivitas komunikasi, terutama dalam kondisi darurat atau visibilitas terbatas, dengan pola bunyi tertentu yang membedakan jenis peringatan sehingga pengguna dapat segera mengambil tindakan.

2.3.13 *Selector Switch*

Selector switch berfungsi sebagai saklar pemilih yang memungkinkan pengguna mengubah mode operasi sistem secara manual. Berikut merupakan Gambar 2. 16 yang menunjukkan bentuk fisik *selector switch*.

Gambar 2. 16 *Selector switch*

(Sumber : https://www.tokopedia.com/anugerahserbaser/3-pole-selector-switch-axc-20-3-3p-putih?utm_source=google&utm_medium=organic&utm_campaign=pdp)

Dalam sistem *monitoring* berbasis elekronik, *selector switch* digunakan sebagai antarmuka pengguna untuk mengendalikan operasi sistem, seperti memilih mode manual atau otomatis serta mengaktifkan maupun menonaktifkan fungsi tertentu. Komponen ini bekerja dengan menghubungkan atau memutus kontak listrik sesuai posisi sakelar yang dipilih. Penggunaan *selector switch* memberikan kemudahan dalam pengoperasian sistem karena mudah digunakan, memiliki keandalan yang baik, tahan terhadap gangguan, dan tidak memerlukan pemrograman tambahan tambahan.

2.3.14 *Pushbutton*

Pushbutton berfungsi sebagai saklar tekan yang digunakan untuk mengaktifkan atau menghentikan fungsi tertentu pada sistem. Berikut merupakan Gambar 2. 17 yang menunjukkan bentuk fisik *pushbutton*.

Gambar 2. 17 *Pushbutton*

(Sumber : https://www.ubuy.hn/en/product/MNPJX8JI-lictop-ac-660v-red-green-momentary-start-stop-no-nc-push-button-switch)

Dalam sistem *monitoring*, *pushbutton* merupakan *input* berbasis saklar tekan yang digunakan untuk memberikan perintah sesaat pada sistem elekronik. Komponen bekerja ketika ditekan, sehingga menghubungkan atau memutuskan rangkaian listrik dan akan kembali ke posisi semula ketika dilepas. *Pushbutton* berperan sebagai antarmuka pengguna untuk memberikan perintah langsung seperti start atau stop. Penggunaan *pushbutton* memberikan kemudahan dalam sistem karena pengoperasian yang memiliki respon yang cepat, desain yang sederhana dan keandalan dalam berbagai kondisi penggunaan.

2.3.15 *Miniature Circuit Breaker* (MCB)

Miniature Circuit Breaker (MCB) digunakan sebagai perangkat pengaman utama untuk melindungi sistem dari gangguan arus lebih dan hubung singkat. Berikut merupakan Gambar 2. 18 yang menunjukkan bentuk fisik *Miniature Circuit Breaker.*

Gambar 2. 18 *Miniature Circuit Breaker*

(Sumber : https://www.lazada.co.id/products/mcb-chint-1p-1-phase-1-pas-mcb-chint-nxb-63-6ka-6a-10a-16a-20a-25a-32a-40a-i7307338687.html)

Dalam sistem *monitoring*, *Miniature Circuit Breaker* (MCB) merupakan perangkat proteksi listrik yang digunakan untuk melindungi rangkaian dari arus lebih (*overload*) dan hubung singkat (*short circuit*). Komponen ini bekerja secara otomatis dengan memutus aliran listrik ketika terjadi gangguan, sehingga mencegah kerusakan pada peralatan maupun risiko kebakaran. MCB dapat dioperasikan kembali secara manual setelah kondisi sistem kembali normal. MCB berperan penting sebagai pengaman utama dalam sistem sekaligus sebagai kontrol untuk menghubungkan atau memutus sumber daya listrik.

2.4 Komponen *Software*

Dalam pengembangan sistem yang terintegrasi, diperlukan *software* untuk mendukung proses pengolahan, analisis, dan visualisasi data secara otomatis. *Software* yang digunakan berfungsi untuk melakukan manajemen data hasil *monitoring*, pra-pemrosesan data, dan implementasi model klasifikasi. Selain itu, *software* ini juga mengatur alur komunikasi data antar komponen sistem agar proses berjalan dengan terstruktur.

2.4.1 Arduino IDE

Arduino IDE merupakan lingkungan pengembangan (*Integrated Development Environment*) yang digunakan untuk mengembangkan program pada mikrokontroler. Berikut merupakan Gambar 2. 19 yang menampilkan logo *software* Arduino IDE.

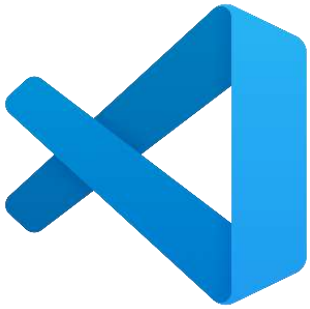
Gambar 2. 19 Arduino IDE
(Sumber : https://www.arduino.cc/en/software/)

Perangkat lunak yang digunakan salah satunya adalah Arduino IDE *(Integrated Development Environment)*, sedangkan mikrokontroler yang digunakan yaitu ESP32. Arduino IDE berfungsi sebagai lingkungan pengembangan untuk menulis dan mengunggah program (*sketch*) yang berisi logika dan algoritma sistem ke dalam mikrokontroler ESP32 (Harianto *et al*., 2021). Arduino IDE menyediakan berbagai pustaka pendukung serta fitur pemantauan serial yang memudahkan proses pengujian, pemantauan data dan debugging sistem selama tahap pengembangan. Arduino IDE memudahkan proses integrasi berbagai sensor dan aktuator, dengan dukungan berbagai pustaka tersebut proses pengembangan sistem menjadi lebih efisien karena tidak harus menulis kode program dari awal. Selain itu, Arduino IDE dilengkapi dengan *serial monitor* yang dapat menampilkan hasil

program sensor atau aktuator melalui komunikasi serial yang berperan penting dalam proses pengujian dan debugging, khususnya untuk memastikan keakuratan data sensor dan format pesan yang dikirim.

2.4.2 Visual Studio Code

Visual Studio Code digunakan untuk mengembangkan program pada sisi *server* dan antarmuka sistem *monitoring*. Berikut merupakan Gambar 2. 20 yang menampilkan logo *software* Visual Studio Code.



Gambar 2. 20 Visual Studio Code
(Sumber : https://apps.microsoft.com/detail/xp9khm4bk9fz7q?hl=en-US&gl=US)

Visual Studio Code adalah perangkat lunak editor kode sumber (*source code editor*) yang digunakan dalam proses pengembangan perangkat lunak. Aplikasi ini mendukung berbagai bahasa pemrograman serta menyediakan fitur-fitur pendukung seperti *debugging*, ekstensi yang memudahkan dalam menulis, mengedit dan mengelola kode program (Rizkyellyasa, 2023). Dukungan ekstensi yang luas memungkinkan integrasi dengan pustaka dan *framework* yang digunakan dalam pengembangan sistem, seperti pemrograman Python dan pengelolaan data berbasis *machine learning*.

2.4.3 Google Colab

Google Colab digunakan untuk melakukan pra-pemrosesan data, pelatihan model Random Forest, serta evaluasi hasil model. Berikut merupakan Gambar 2. 21 yang menampilkan logo *software* Google Colab.

Gambar 2. 21 Google Colab
(Sumber : https://en.wikipedia.org/wiki/Google_Colab)

Google Colab (*Colaboratory*) merupakan lingkungan pengembangan berbasis *cloud* yang digunakan dalam pengujian program. Platform ini memungkinkan pengguna dapat menulis dan menjalankan program tanpa memerlukan instalasi *software* tambahan. Google colab mendukung berbagai bahasa termasuk Python, serta menyediakan fitur pendukung seperti eksekusi sel interaktif, integrasi dengan Google Drive, dan dukungan pustaka komputasi yang mempermudah proses analisis dan pengolahan data (Bisong, 2019).

2.4.4 Next.js

Next.js digunakan untuk mengembangkan *dashboard* yang menampilkan hasil *monitoring* dan klasifikasi kualitas air secara *real-time*. Berikut merupakan Gambar 2. 22 yang menampilkan logo *software* Next.js.

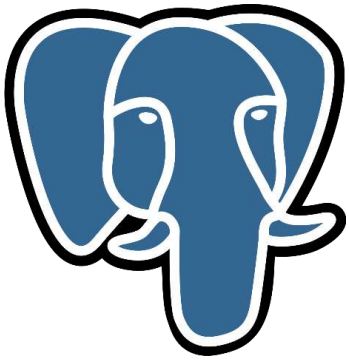
Gambar 2. 22 Next.js
(Sumber : https://www.drupal.org/project/next)

Next.js adalah *framework* berbasis JavaScript yang digunakan untuk mengembangkan aplikasi web interaktif dengan cepat dan terstruktur, terutama untuk kebutuhan visualisasi data, antarmuka interaktif serta integrasi dengan berbagai layanan *backend*. Next.js memungkinkan pengembang menampilkan hasil

pengolahan data, grafik, serta *output* model klasifikasi secara *real-time* tanpa memerlukan pemahaman mendalam mengenai pengembangan web. Next.js digunakan sebagai antarmuka (*user interface*) untuk menampilkan hasil pemrosesan dan klasifikasi data.

2.4.5 PostgreSQL

PostgreSQL digunakan sebagai media penyimpanan data hasil pembacaan sensor dan hasil klasifikasi kualitas air. Berikut merupakan Gambar 2. 23 yang menampilkan logo *software* postgreSQL.



Gambar 2. 23 PostgreSQL
(Sumber : https://en.wikipedia.org/wiki/PostgreSQL)

PostgreSQL adalah sistem manajemen basis data relasional (*Relational Database Management System*/RDBMS) yang berfungsi untuk menyimpan, mengatur, dan memproses data secara terstruktur. PostgreSQL menggunakan bahasa *Structured Query Language* (SQL) sebagai bahasa standar dalam pengelolaan basis data, seperti proses penyimpanan, pengambilan, pembaruan, dan penghapusan data. PostgreSQL digunakan sebagai media penyimpanan data kualitas air tambak udang vaname.

2.4.6 Supabase

Supabase digunakan sebagai platform *backend* untuk mengelola basis data dan mendukung komunikasi data pada sistem *monitoring*. Berikut merupakan Gambar 2. 24 yang menampilkan logo *software* Supabase.

Gambar 2. 24 Supabase
(Sumber : https://eaglestatus.io/services/supabase)

Supabase adalah sebuah platform *backend-as-a-service* (BaaS) yang bersifat *open-source* dan digunakan untuk mendukung pengembangan aplikasi berbasis web maupun *mobile* secara efisien. Supabase menyediakan berbagai layanan terintegrasi seperti *database*, autentikasi, API otomatis, dan penyimpanan data dalam satu sistem. Platform ini dibangun di atas PostgreSQL yang berfungsi sebagai sistem manajemen basis data relasional yang andal. Supabase digunakan sebagai *backend* untuk menjalankan sistem *monitoring* tambak udang vaname serta mengelola basis data yang digunakan. Melalui fitur API otomatis, Supabase memungkinkan aplikasi untuk mengakses dan menampilkan data secara *real-time* pada *dashboard monitoring*.

2.4.7 Python

Python digunakan untuk melakukan *preprocessing* data, pengembangan model Random Forest, serta evaluasi hasil klasifikasi. Berikut merupakan Gambar 2. 25 yang menampilkan logo *software* Python.

Gambar 2. 25 Python
(Sumber : https://en.wikipedia.org/wiki/Python_(programming_language))

Python merupakan bahasa pemrograman tingkat tinggi yang banyak digunakan dalam bidang analisis data dan penelitian ilmiah karena memiliki sintaks yang sederhana serta didukung oleh berbagai pustaka pendukung (Bisong, 2019). Python digunakan sebagai alat utama untuk melakukan pengolahan data kualitas air tambak udang, mulai dari tahap *preprocessing* data, pelatihan dan pengujian model, hingga evaluasi hasil menggunakan algoritma Random Forest. Selain itu, Python juga berperan dalam menghasilkan hasil klasifikasi atau prediksi kondisi kualitas air yang selanjutnya disimpan ke dalam basis data dan ditampilkan pada sistem *monitoring* berbasis web. Dengan demikian, Python berfungsi sebagai komponen penting dalam proses analisis dan pemodelan data.

2.4.8 Scikit-learn

Scikit-learn digunakan untuk mengimplementasikan model Random Forest, serta melakukan proses pelatihan, pengujian, dan evaluasi model. Berikut merupakan Gambar 2. 26 yang menampilkan logo *software* Scikit-learn.

Gambar 2. 26 Scikit-learn
(Sumber : https://en.wikipedia.org/wiki/Scikit-learn)

Untuk melakukan komputasi *machine learning*, Scikit-learn merupakan salah satu pustaka Python yang dirancang untuk menyediakan berbagai algoritma pembelajaran mesin dengan antarmuka yang sederhana dan konsisten. Pustaka ini banyak digunakan dalam pengolahan data. Scikit-learn mendukung berbagai metode pembelajaran seperti klasifikasi, regresi, *clustering*, dan reduksi dimensi, sehingga sering dimanfaatkan dalam penelitian dan pengembangan sistem berbasis data. Selain menyediakan algoritma, Scikit-learn juga mendukung pra-pemrosesan data normalisasi, standarisasi, dan pembagian *dataset* menjadi data latih dan data uji. Fitur evaluasi model seperti *confusion matrix* dan perhitungan akurasi turut

membantu dalam menganalisis performa model yang di bangun. Dengan dukungan algoritma *ensemble* seperti Random Forest, Scikit-learn menjadi salah satu pustaka yang banyak digunakan dalam pengembangan sistem klasifikasi dan prediksi berbasis *machine learning.*

2.4.9 Eagle

Eagle digunakan sebagai perangkat lunak untuk merancang skematik dan tata letak *Printed Circuit Board* (PCB). Berikut merupakan Gambar 2. 27 yang menampilkan logo *software* Eagle.

Gambar 2. 27 Eagle
(Sumber : https://heti.com.mx/autodesk-software/autodesk-eagle/)

Dalam perancangan *hardware*, diperlukan *software* untuk mendukung pembuatan skema rangkaian dan *layout* PCB secara akurat. Eagle (*Easily Applicable Graphical Layout Editor*) merupakan perangkat lunak yang digunakan untuk merancang skema rangkaian elektronik (*schematic*) dan tata letak papan sirkuit tercetak (*Printed Circuit Board*/PCB) (Sasnita *et al*., 2024). Eagle menyediakan pustaka komponen elektronik yang lengkap dan fleksibel, sehingga memudahkan pengguna dalam memilih serta menyesuaikan komponen sesuai kebutuhan sistem. Selain itu, Eagle mendukung proses pengecekan kesalahan (*Electrical Rule Check* dan *Design Rule Check*) untuk memastikan rancangan PCB sesuai dengan standar desain dan meminimalkan kesalahan sebelum proses fabrikasi. Eagle digunakan suntuk merancang skematik dan *layout* PCB sistem *monitoring* dan peringatan dini kualitas air agar rangkaian yang dibuat lebih rapi, terstruktur, dan mudah direalisasikan.

BAB 3
METODOLOGI PENELITIAN

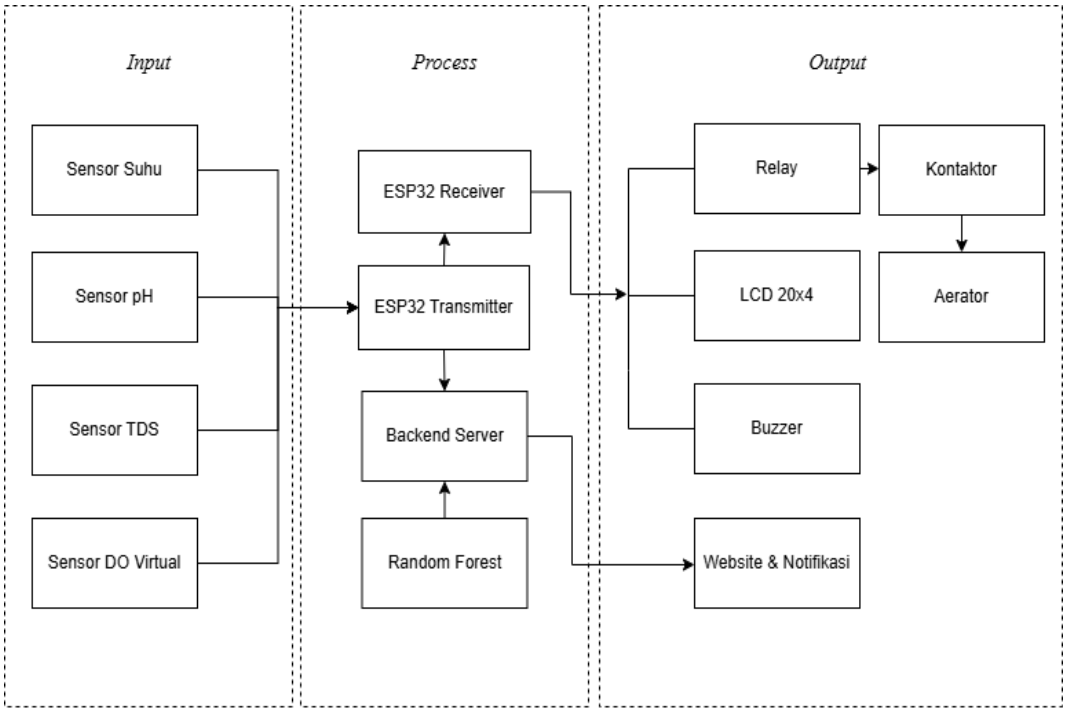
Pada Bab 3 dilakukan pembahasan mendalam terkait perancangan sistem, yang mencakup tahapan perumusan konsep, rangkaian tahapan penelitian, desain rinci *hardware* dan *software,* penerapan model klasifikasi, dan evaluasi performa model.

3.1 Konsep Penelitian

Pada subbab ini dibahas konsep penelitian yang menjadi landasan utama dalam memahami konteks dan tujuan penelitian. Konsep ini menjelaskan alur kerja sistem klasifikasi kualitas air tambak udang.

3.1.1 Diagram Blok Sistem

Berikut merupakan Gambar 3. 1 yang menunjukkan alur kerja sistem dari *input*, *process*, hingga *output*.



Gambar 3. 1 Diagram Blok Sistem
(Sumber : Penulis)

Pada bagian *input,* sistem terdiri dari beberapa sensor yang berfungsi untuk mengumpulkan data dari lingkungan tambak udang. Sensor yang digunakan

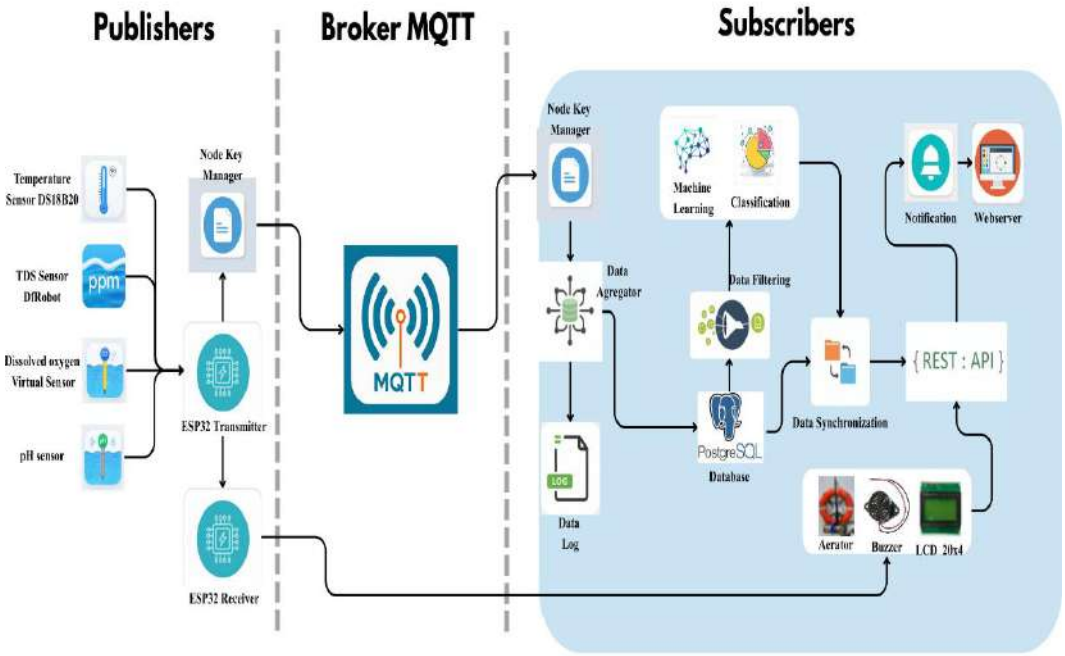
meliputi sensor suhu, sensor pH, sensor TDS, dan sensor DO virtual. Sensor suhu berfungsi untuk mengukur suhu air, sedangkan sensor pH digunakan untuk mengetahui tingkat keasaman air. Sensor TDS berperan dalam mengukur kadar total zat terlarut dalam air, dan sensor DO virtual digunakan untuk mengukur kadar *Dissolved Oxygen*. Seluruh data yang diperoleh dari sensor-sensor tersebut kemudian dikirimkan ke ESP32 *transmitter* untuk diproses lebih lanjut.

Pada bagian *process*, data yang diperoleh dari seluruh sensor diolah oleh ESP32 *transmitter* sebagai pusat pengumpulan dan pengiriman data sensor. Data tersebut kemudian diteruskan secara nirkabel menggunakan protokol ESP-NOW, yaitu protokol komunikasi *peer-to-peer* yang memungkinkan pertukaran data secara langsung antar perangkat ESP32 tanpa memerlukan jaringan *Wi-Fi* atau *access point,* menuju ESP32 *receiver* kemudian diteruskan ke *backend.* Pada ESP32 *receiver* bertugas untuk menjalankan aktuator seperti aerator, *buzzer* dan LCD 20x4 sedangkan pada *backend server*, data sensor diproses menggunakan metode Random Forest untuk melakukan analisis dan pengambilan keputusan berdasarkan parameter yang telah ditentukan. Hasil dari proses tersebut mengevaluasi kondisi berdasarkan kombinasi data sensor untuk menentukan status sistem apabila membutuhkan tindakan. Hasil analisis tersebut kemudian dikirim kembali sebagai dasar pengendalian *output* sistem.

Pada bagian *output*, sistem menghasilkan keluaran berdasarkan hasil pengolahan data pada tahap proses. Keluaran sistem berupa pengendalian perangkat dan memberikan informasi kepada pengguna. Perangkat keluaran yang digunakan meliputi relay, kontaktor dan aerator yang berfungsi untuk membantu menjaga kadar DO dan menjaga sirkulasi air, LCD 20x4 sebagai media penampil informasi hasil pembacaan sensor, *buzzer* sebagai indikator peringatan apabila kondisi berada di luar parameter yang ditentukan. Selain itu, sistem juga menampilkan data dan mengirimkan notifikasi melalui telegram dan *website* sebagai *dashboard monitoring* kondisi tambak secara *online*. Seluruh keluaran tersebut bekerja sesuai dengan keputusan yang dihasilkan oleh proses analisis sistem.

3.1.2 Arsitektur Perencanaan Sistem

Berikut arsitektur perencanaan sistem dirancang menggunakan konsep *publish–subscribe* dengan protokol MQTT. Sistem terdiri dari tiga komponen utama, yaitu *Publishers*, Broker MQTT, dan *Subscribers*, sebagaimana ditunjukkan pada Gambar 3. 2.



Gambar 3. 2 Arsitektur Sistem Klasifikasi Kualitas Air
(Sumber : Penulis)

Arsitektur sistem penelitian menjelaskan rancangan sistem *monitoring*, klasifikasi, dan peringatan dini kualitas air tambak udang berbasis *Internet of Things* (IoT). Sistem ini dirancang menggunakan arsitektur *publish–subscribe* dengan protokol MQTT untuk memastikan pengiriman data sensor berlangsung secara *real-time* dan efisien.

Pada sisi *publisher*, sistem terdiri dari sensor suhu DS18B20, sensor pH, sensor TDS yang terhubung ke ESP32 *transmitter*. Data sensor tersebut kemudian digunakan dalam proses prediksi nilai DO virtual pada *server*. ESP32 *transmitter* berfungsi sebagai perangkat akuisisi data yang membaca nilai sensor secara periodik. Data hasil pembacaan sensor kemudian dikirimkan ke ESP32 *Receiver* menggunakan protokol ESP-NOW dan digunakan untuk mengendalikan perangkat keluaran seperti LCD 20×4, *buzzer*, dan aerator. Protokol ini dipilih karena

memiliki waktu transmisi yang relatif cepat, konsumsi daya yang rendah, serta tidak memerlukan infrastruktur jaringan seperti *wireless access point*. Selain itu, komunikasi dilakukan berdasarkan alamat MAC (*Media Access Control*) sehingga proses pertukaran data antarperangkat dapat berlangsung secara langsung (*peer-to-peer*) dengan latensi yang rendah. Selain itu, data sensor juga dikirimkan ke broker MQTT yaitu HiveMQ melalui jaringan internet untuk proses menyimpan dan pengolahan data pada *server*.

Broker MQTT berperan sebagai perantara komunikasi antara perangkat IoT dan *server*. Broker menerima data dari ESP32 *transmitter* dan mendistribusikannya ke *subscriber* tanpa keterikatan langsung antara pengirim dan penerima data. Mekanisme ini memungkinkan sistem bekerja secara *asynchronous* serta meningkatkan keandalan komunikasi data.

Pada sisi *subscriber*, data sensor yang diterima dari broker MQTT diproses oleh *server backend*. Data terlebih dahulu melalui proses agregasi dan penyaringan untuk memastikan validitas data sebelum disimpan ke dalam *database* PostgreSQL. Data yang tersimpan berfungsi sebagai data historis dan menjadi *input* dalam proses klasifikasi kualitas air.

Proses klasifikasi kualitas air dilakukan menggunakan metode Random Forest dengan parameter utama *Dissolved Oxygen* (DO) virtual. Hasil klasifikasi digunakan sebagai dasar sistem peringatan dini untuk mendeteksi kondisi air yang tidak optimal. Ketika kondisi berisiko terdeteksi, sistem mengirimkan notifikasi kepada pengguna melalui telegram, kemudian diteruskan ke ESP32 *Receiver* untuk mengendalikan aerator, *buzzer*, serta menampilkan informasi kondisi kualitas air pada LCD 20×4. Selain itu, sistem menyediakan REST API untuk mendukung integrasi dengan aplikasi *monitoring* berbasis web.

3.1.3 Konsep Sistem Peringatan Dini dan Klasifikasi Kualitas Air

Konsep sistem peringatan dini dirancang untuk mendeteksi secara cepat kondisi kualitas air yang berada di luar batas normal sehingga tindakan pencegahan dapat dilakukan. Sistem bekerja secara *real-time* dengan memanfaatkan data hasil pembacaan sensor yang selanjutnya dikirimkan ke *server* melalui protokol komunikasi untuk diproses lebih lanjut.

Selain sebagai sistem peringatan dini, sistem ini dilengkapi dengan mekanisme klasifikasi kualitas air menggunakan algoritma Random Forest. Parameter pH dan suhu digunakan sebagai atribut (*feature*) dalam proses pembentukan model klasifikasi. Sementara itu, nilai *Dissolved Oxygen* (DO) virtual digunakan sebagai dasar dalam proses pelabelan data sehingga menghasilkan kelas kualitas air yang menjadi target (*label*) pada proses pelatihan model. Dengan demikian, algoritma Random Forest mempelajari hubungan antara parameter pH dan suhu terhadap kelas kualitas air yang dibentuk berdasarkan nilai DO virtual.

Proses pelabelan dilakukan dengan mengelompokkan nilai virtual DO ke dalam tiga kategori kualitas air, yaitu tercemar ringan, tercemar sedang, dan tercemar berat berdasarkan rentang nilai DO yang telah ditentukan. Kategori kualitas air yang digunakan ditunjukkan pada Tabel 3. 1.

Tabel 3. 1 Kategori Risiko kualitas Air

Kelas Risiko	Kategori	Kriteria
Kelas 1	Tercemar Ringan	DO Virtual ≥ 5 mg/L
Kelas 2	Tercemar Sedang	4 ≤ DO Virtual < 5 mg/L
Kelas 3	Tercemar Berat	DO Virtual < 4 mg/L,

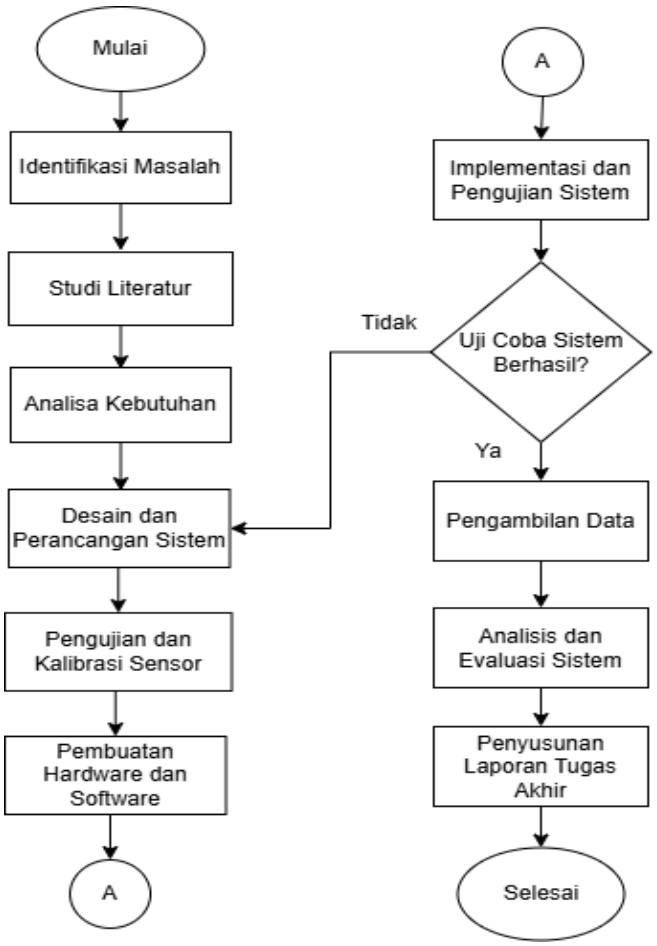
(Sumber : Penulis)

Sistem akan menampilkan hasil klasifikasi kualitas air pada *dashboard* pemantauan melalui model yang telah dilatih kemudian digunakan untuk memprediksi kategori kualitas air berdasarkan nilai parameter pH dan suhu yang diterima secara *real-time*. Apabila nilai DO virtual dibawah ambang batas maka akan otomatis mengirimkan notifikasi melalui aplikasi Telegram agar pembudidaya dapat segera melakukan tindakan penanganan terhadap kondisi air tambak udang vaname.

3.2 Tahap Penelitian

Pada tahap ini dibahas mengenai tahapan-tahapan yang dilakukan dalam perancangan dan pengembangan sistem. Sistem yang dibuat akan diuji menggunakan data nyata untuk melihat kinerja metode klasifikasi. Tahapan penelitian disusun secara sistematis agar proses pengembangan sistem dapat berjalan secara terstruktur dan terarah.

Penyusunan tahapan ini bertujuan untuk memastikan setiap proses, mulai dari perencanaan hingga evaluasi sistem dilakukan secara berurutan dan saling berkaitan dan dengan adanya tahapan penelitian yang jelas, penelitian dapat dilakukan sesuai dengan tujuan yang ditetapkan serta menghasilkan sistem yang mampu bekerja secara optimal dalam melakukan *monitoring* dan klasifikasi kualitas air. Alur penelitian ditunjukkan dalam *flowchart* pada Gambar 3. 3 sebagai panduan penelitian.



Gambar 3. 3 *Flowchart* Tahapan Penelitian
(Sumber : Penulis)

Flowchart tersebut menunjukkan alur tahapan penelitian yang dilakukan secara sistematis, mulai dari identifikasi masalah hingga penyusunan laporan tugas akhir. Setiap tahapan saling berkaitan dan disusun secara berurutan untuk memastikan proses perancangan, pengembangan, serta pengujian sistem berjalan dengan baik.

3.2.1 Tahap Identifikasi Masalah

Pada tahap identifikasi masalah dilakukan analisis awal untuk memahami permasalahan yang terjadi pada kualitas air tambak udang vaname. Kualitas air merupakan salah satu faktor penting yang memengaruhi pertumbuhan, kesehatan, dan tingkat kelangsungan hidup udang. Perubahan kualitas air yang tidak terpantau dengan baik dapat menyebabkan penurunan produktivitas budidaya bahkan meningkatkan risiko kematian udang. Oleh karena itu, diperlukan sistem yang mampu memantau kondisi kualitas air secara berkelanjutan dan memberikan informasi yang mudah dipahami oleh pengguna.

Pemantauan kualitas air yang dilakukan saat ini umumnya hanya menampilkan nilai parameter sensor seperti suhu, pH, *Total Dissolved Solid* (TDS), dan *Dissolved Oxygen* (DO). Informasi yang diperoleh masih berupa data mentah sehingga pengguna harus melakukan interpretasi secara mandiri untuk menentukan kondisi kualitas air tambak. Kondisi tersebut dapat menyebabkan perbedaan penilaian dan keterlambatan dalam pengambilan keputusan ketika terjadi penurunan kualitas air.

Berdasarkan permasalahan tersebut, diperlukan suatu sistem yang tidak hanya mampu melakukan pemantauan kualitas air secara *real-time*, tetapi juga dapat mengklasifikasikan kondisi kualitas air secara otomatis berdasarkan data yang diperoleh dari sensor. Pemanfaatan data historis dan metode machine learning diharapkan dapat membantu mengidentifikasi tingkat risiko kualitas air secara lebih cepat dan konsisten, sehingga dapat mendukung sistem peringatan dini serta pengambilan keputusan dalam pengelolaan tambak udang vaname.

3.2.2 Analisa Kebutuhan Sistem

Tahap analisa kebutuhan sistem bertujuan untuk mengidentifikasi seluruh kebutuhan yang diperlukan dalam pengembangan sistem yang akan dibangun, baik dari *hardware* maupun *software.* Hasil analisa kebutuhan digunakan sebagai dasar dalam proses perancangan dan implementasi sistem sehingga sistem yang dibangun dapat berfungsi sesuai dengan tujuan penelitian.

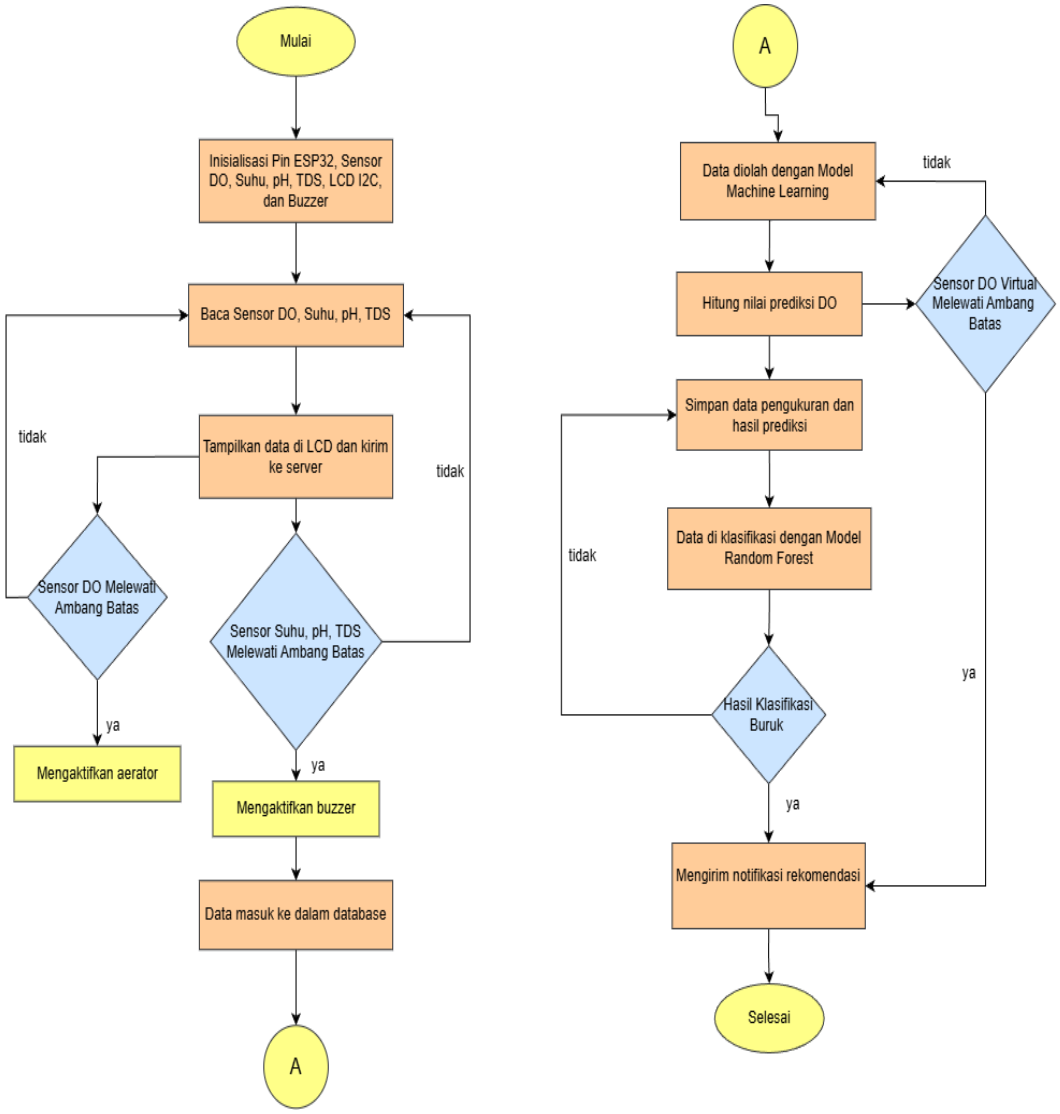
Pada sisi *hardware*, analisis meliputi penggunaan mikrokontroler ESP32 sebagai pusat pengolahan data, sensor suhu, sensor pH, sensor TDS, dan sensor DO

virtual sebagai parameter kualitas air tambak, serta perangkat aktuator dan antarmuka berupa kontaktor, relay, aerator, layar LCD 20x4, *buzzer* sebagai sistem peringatan dini, dan catu daya sebagai sumber energi sistem.

Pada sisi *software*, analisis mencakup penggunaan broker MQTT sebagai media komunikasi data antara perangkat IoT dan *server*, *backend* digunakan untuk pengolahan data dan integrasi sistem, basis data untuk penyimpanan data kualitas air, model klasifikasi Random Forest untuk mengklasifikasikan kondisi kualitas air tambak, serta *dashboard* berbasis web sebagai sarana *monitoring* dan visualisasi data secara *real-time*. Hasil dari tahap analisis kebutuhan sistem ini menjadi acuan utama dalam proses desain dan perancangan sistem secara keseluruhan.

3.2.3 Desain dan Perancangan Sistem

Pada tahap desain perancangan sistem dilakukan perancangan keseluruhan sistem peringatan dini kualitas air tambak udang berdasarkan hasil analisis kebutuhan sistem. Tahap ini berfokus pada perancangan integrasi antara *hardware* dan *software*, termasuk alur pengambilan data dari sensor kualitas air, pemrosesan data, serta penerapan sistem klasifikasi kualitas air menggunakan metode Random Forest. Perancangan ini bertujuan untuk memastikan seluruh komponen sistem dapat saling terhubung dan berfungsi secara terpadu sesuai dengan kebutuhan yang telah ditetapkan.



Gambar 3. 4 *Flowchart* Sistem Klasifikasi Kualitas Air
(Sumber : Penulis)

Berdasarkan Gambar 3. 4, alur kerja sistem dimulai dari inisialisasi pin ESP32 dan juga sensor beserta aktuatornya. Setelah dilakukan inisialisasi, sistem melakukan pembacaan data sensor secara periodik, kemudian data tersebut ditampilkan pada LCD dan dikirim ke *server* untuk disimpan dalam *database.* Ketika sensor DO melewati ambang batas maka akan mengaktifkan aerator secara otomatis dan ketika sensor suhu, TDS, pH melewati ambang batas maka akan mengaktifkan *buzzer*. Data yang diperoleh kemudian diproses menggunakan model prediksi yang telah dibangun untuk menentukan nilai DO virtual dan hasil tersebut disimpan bersama data pengukuran aktual. Tahap berikutnya adalah proses klasifikasi kualitas air menggunakan metode Random Forest untuk menentukan

kondisi tambak. Jika nilai DO virtual melewati ambang batas maka sistem akan mengirimkan notifikasi rekomendasi kepada pengguna. Seluruh proses ini berlangsung secara berulang dan *real-time* selama sistem beroperasi guna memastikan kualitas air tambak tetap berada dalam kondisi optimal.

3.2.4 Implementasi dan Pengujian

Pada tahap implementasi dan pengujian, desain sistem yang telah dirancang direalisasikan dalam bentuk *hardware* dan *software*. Implementasi meliputi pemasangan dan kalibrasi sensor kualitas air, mikrokontroler ESP32, serta pengembangan berbasis MQTT untuk pengiriman data ke *server.*

Tahap pengujian sistem dilakukan secara menyeluruh untuk memastikan setiap komponen berfungsi dengan baik. Pengujian meliputi pengujian fungsional sensor, pengujian komunikasi data antara ESP32 dan *server*, serta pengujian akurasi pembacaan sensor.

Hasil pengujian digunakan untuk mengevaluasi kinerja sistem secara keseluruhan, baik dari sisi *hardware, software,* maupun model klasifikasi yang digunakan. Apabila ditemukan ketidaksesuaian antara hasil implementasi dan perancangan sistem, maka dilakukan perbaikan dan penyesuaian hingga sistem dapat beroperasi sesuai dengan tujuan penelitian.

3.2.5 Analisa Data

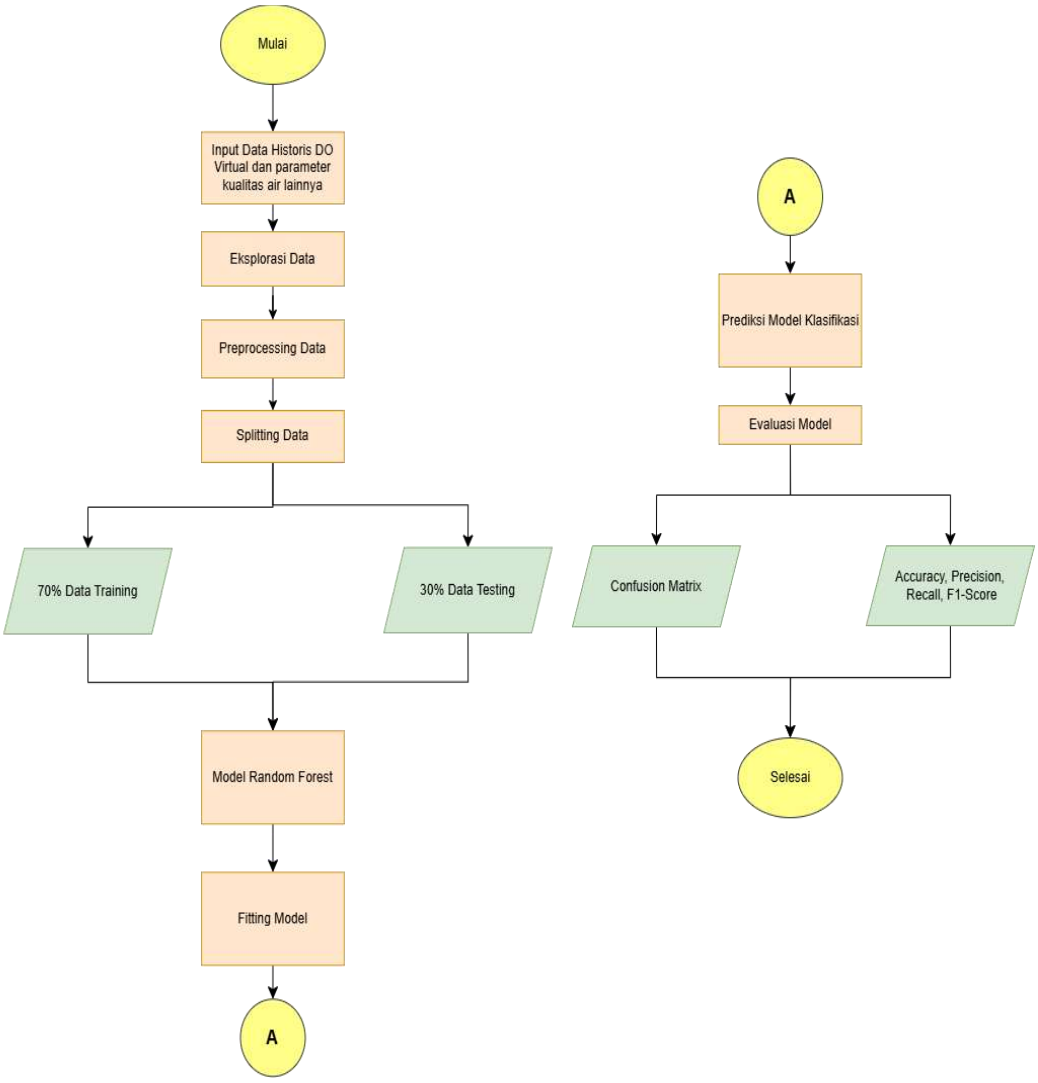
Pada tahap analisa data dilakukan pengolahan dan analisis data yang digunakan untuk proses klasifikasi kualitas air tambak udang.. Proses meliputi pengelolaan data sensor, pemilihan fitur, serta penerapan metode klasifikasi yang sesuai.

Tahap ini memastikan bahwa data yang diperoleh dari sistem dapat dimanfaatkan secara optimal dalam proses pengambilan keputusan terkait kualitas air. Tahapan-tahapan tersebut disusun secara sistematis agar proses klasifikasi dapat menghasilkan keluaran yang akurat dan konsisten.

Berdasarkan hasil penelitian terdahulu, algoritma Random Forest diketahui memiliki performa yang lebih baik dibandingkan metode lainnya, khususnya dalam menangani data kompleks, tahan terhadap noise dan bersifat nonlinear.

Perancangan model bertujuan untuk menerapkan algoritma Random Forest dalam analisis data guna mendukung penggunaan sensor *Dissolved Oxygen* (DO) virtual, sehingga dapat meningkatkan akurasi proses klasifikasi kualitas air pada tambak udang.

Data yang digunakan merupakan data nyata yang diperoleh dari sistem, kemudian diolah melalui beberapa tahapan *machine learning* untuk menghasilkan model klasifikasi yang optimal. Alur proses analisa data ditunjukkan dalam *flowchart* pada Gambar 3. 5 sebagai panduan dalam pelaksanaan analisa data.



Gambar 3. 5 *Flowchart* Proses *Machine Learning*
(Sumber : Penulis)

Proses analisa data diawali dengan pengumpulan dan eksplorasi data untuk memahami karakteristik *dataset* yang digunakan. Selanjutnya dilakukan

preprocessing data untuk mempersiapkan data sebelum digunakan pada proses pelatihan model. *Dataset* kemudian dibagi menjadi data latih dan data uji untuk membangun serta mengevaluasi model Random Forest. Model yang telah dilatih digunakan untuk melakukan klasifikasi kualitas air, kemudian hasil klasifikasi dievaluasi untuk mengetahui tingkat performa model.

3.2.6 Pembuatan Laporan

Setelah seluruh tahapan penelitian selesai dilakukan, tahap pembuatan laporan dilaksanakan dengan menyusun seluruh hasil penelitian secara sistematis dan terstruktur. Laporan ini mencakup proses perancangan, implementasi sistem, analisis data, serta hasil pengujian dan evaluasi sistem. Tahap ini bertujuan untuk mendokumentasikan seluruh proses dan hasil penelitian sebagai bentuk pertanggungjawaban ilmiah, serta sebagai referensi bagi pengembangan penelitian selanjutnya.

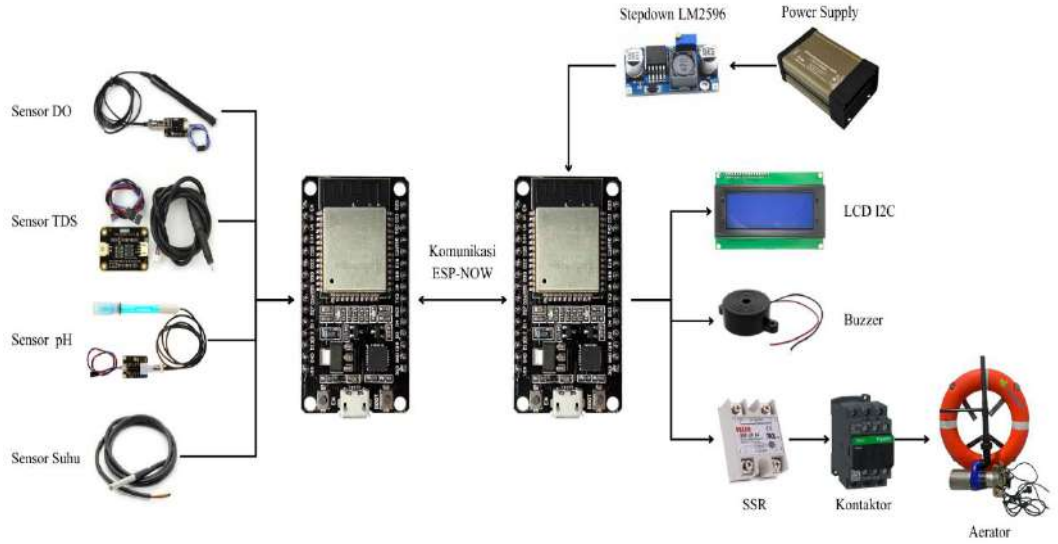
3.3 Perencanaan dan Desain *Hardware*

Pada subbab ini menjelaskan perencanaan dan desain *hardware* yang digunakan dalam sistem pemantauan kualitas air tambak udang. Mencakup *wiring* komponen, skematik *minimum system* ESP32 dan desain alat beserta *plant* tambak udang.

3.3.1 Perancangan *Hardware* Elektrik

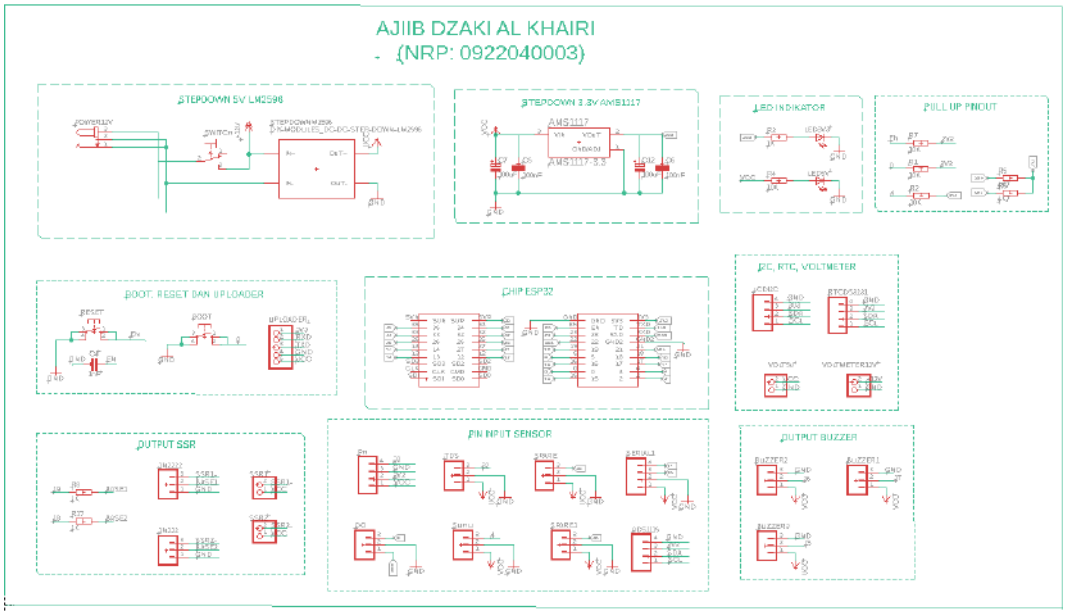
Perancangan *hardware* elektrik dilakukan untuk menentukan konfigurasi rangkaian dan pengkabelan (*wiring*) sistem secara keseluruhan. Perancangan tersebut bertujuan untuk memastikan seluruh komponen dapat bekerja secara stabil dan terintegrasi serta menjadi dasar dalam tahap implementasi fisik sistem, sehingga kesesuaian antara desain dan realisasi dapat terjaga.

Konfigurasi rangkaian juga memperhatikan distribusi daya dan komunikasi antar komponen agar sistem dapat beroperasi sesuai dengan rancangan. Desain rangkaian dan *wiring* ditampilkan pada Gambar 3. 6 sebagai panduan dalam pelaksanaan implementasi sistem.



Gambar 3. 6 Desain Rangkaian dan *Wiring* Sistem
(Sumber : Penulis)

Seluruh komponen dihubungkan sesuai dengan kebutuhan sistem agar komunikasi data dan distribusi daya dapat berjalan dengan baik. Desain rangkaian ini digunakan sebagai acuan dalam proses implementasi dan pengujian sistem guna memastikan sistem dapat beroperasi secara stabil dan sesuai dengan perancangan. Selain itu, skematik *minimum system* ESP32 ditampilkan pada Gambar 3. 7 sebagai dasar perancangan rangkaian pengendali sistem.

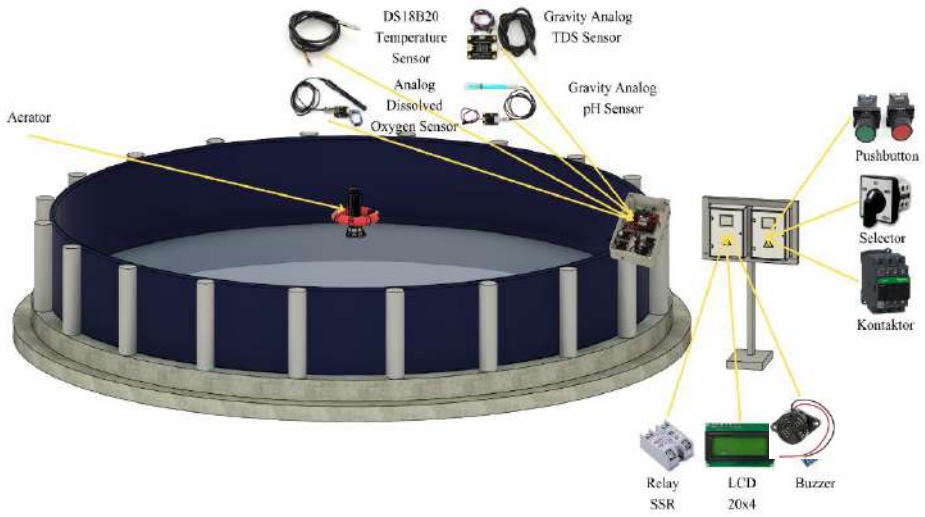


Gambar 3. 7 Skematik *Minimum System* ESP32
(Sumber : Penulis)

Gambar tersebut menunjukkan skematik minimum sistem mikrokontroler ESP32 yang digunakan sebagai pengendali utama pada sistem *monitoring* kualitas air. Skematik ini menampilkan konfigurasi pin ESP32 beserta rangkaian catu daya yang dirancang untuk memastikan kestabilan tegangan selama sistem beroperasi. Rangkaian catu daya terdiri dari *step-down* LM2596 untuk menghasilkan tegangan 5V dan regulator AMS1117 yang menurunkan tegangan menjadi 3,3V sebagai catu utama ESP32. Konfigurasi *minimum system* ini digunakan sebagai acuan dalam perancangan PCB dan implementasi perangkat keras agar ESP32 dapat bekerja secara stabil dalam pengolahan dan pengiriman data sistem *monitoring*.

3.3.2 Perancangan *Hardware* Mekanik

Perancangan *hardware* mekanik dilakukan untuk menentukan bentuk fisik dan tata letak perangkat sistem pada lingkungan tambak udang. Rancangan *Plant* pada Tambak Udang ditunjukkan pada Gambar 3. 8, yang menggambarkan penempatan perangkat *monitoring* dan komponen pada area tambak. Perancangan ini bertujuan untuk memastikan perangkat dapat terpasang dengan baik, terlindungi dari kondisi lingkungan tambak, serta mendukung kinerja sistem *monitoring* kualitas air secara optimal.



Gambar 3. 8 Rancangan *Plant* Sistem Tambak Udang
(Sumber : Penulis)

Mekanik *plant* tambak udang ini dirancang dengan tujuan untuk memastikan bahwa semua komponen bekerja secara maksimal dan mendukung

operasional sistem *monitoring* dalam jangka panjang. Perancangan mekanik dilakukan dengan mempertimbangkan kondisi lingkungan tambak yang lembap dan korosif, sehingga struktur dan penempatan komponen diatur agar tetap aman dan terlindungi.

Gambar 3. 9 Rancangan Desain Mekanik pada *Box* Sensor
(Sumber: Penulis)

Pada Gambar 3. 9 ditunjukkan rancangan desain mekanik *box* sensor yang digunakan sebagai wadah rangkaian elektronika pada sistem *monitoring* kualitas air tambak udang. *Box* sensor dirancang untuk melindungi komponen elektronika dari paparan air, debu, dan kondisi lingkungan luar, serta menjaga susunan komponen tetap rapi sehingga memudahkan proses instalasi dan pemeliharaan sistem.

Gambar 3. 10 Rancangan Desain Mekanik Panel Kontrol
(Sumber: Penulis)

Pada Gambar 3. 10 ditunjukkan rancangan desain mekanik panel kontrol yang digunakan sebagai pusat pengendali pada sistem *monitoring* kualitas air tambak udang. Panel kontrol dirancang menggunakan *box* panel berbahan logam

yang berfungsi melindungi komponen elektronika dari gangguan lingkungan luar. Panel ini juga memudahkan proses instalasi, pengoperasian, dan pemeliharaan sistem.

Panel kontrol dilengkapi dengan tombol *ON* dan *OFF* sebagai sistem manual aerator, lampu indikator sebagai penanda kondisi operasi, LCD untuk menampilkan informasi hasil pembacaan sensor secara langsung, *selector switch* yang digunakan untuk memilih mode operasi sistem, serta *buzzer* yang berfungsi sebagai alarm ketika nilai parameter kualitas air melebihi batas yang telah ditentukan. Tata letak setiap komponen dirancang secara ergonomis sehingga memudahkan pengguna dalam mengoperasikan dan memantau kondisi sistem.

Gambar 3. 11 Rancangan *Plant* Pengujian Sistem
(Sumber: Penulis)

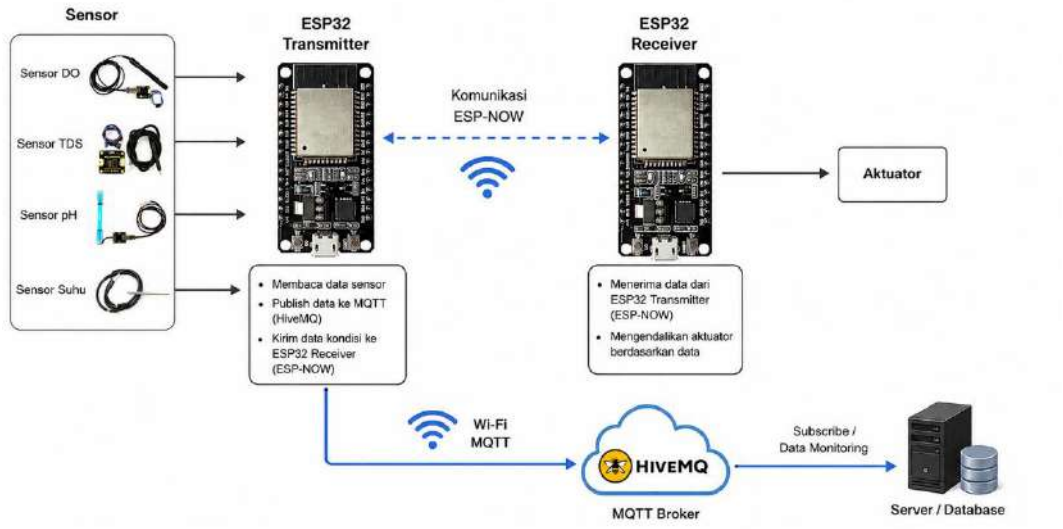
Pada Gambar 3. 11 tersebut menunjukkan rancangan *plant* desain pengujian sistem *monitoring* kualitas air yang dilakukan di lingkungan kampus. *Plant* pengujian terdiri dari wadah air sebagai media uji, *enclosure* sensor yang ditempatkan di dalam media air, serta *box* panel yang berisi rangkaian elektronik seperti mikrokontroler dan LCD. Rancangan ini memungkinkan pengujian fungsi sensor, akurasi pembacaan data, serta kestabilan sistem *monitoring* secara *real-time* dalam kondisi terkontrol.

3.4 Perencanaan dan Desain *Software*

Pada subbab ini dijelaskan perencanaan dan desain perangkat lunak yang digunakan dalam sistem pemantauan kualitas air tambak udang. Pembahasan mencakup perancangan komunikasi data, pengiriman data, perancangan *database*, perancangan *backend,* serta desain *monitoring* berbasis web.

3.4.1 Perancangan Komunikasi ESP-NOW dan MQTT

Komunikasi antarperangkat dirancang menggunakan kombinasi protokol ESP-NOW dan MQTT. ESP-NOW digunakan sebagai media komunikasi nirkabel antar modul ESP32 tanpa memerlukan koneksi *access point* (AP), sedangkan MQTT digunakan untuk mengirimkan data hasil pengukuran ke *server* sehingga dapat dipantau secara *real-time*.



Gambar 3. 12 Arsitektur Komunikasi ESP-NOW dan MQTT
(Sumber: Penulis)

Berdasarkan Gambar 3. 12, sistem terdiri atas dua buah ESP32 yang memiliki fungsi berbeda, yaitu ESP32 *Transmitter* dan ESP32 *Receiver*. ESP32 *Transmitter* bertugas membaca data dari sensor kualitas air yang meliputi suhu, pH, TDS, dan DO. Setelah data sensor diperoleh, ESP32 *Transmitter* mengirimkan data tersebut ke broker MQTT HiveMQ menggunakan protokol MQTT melalui koneksi *Wi-Fi.* ESP32 *Transmitter* bertindak sebagai *publisher*, sedangkan HiveMQ berfungsi sebagai *message* broker yang menerima dan meneruskan data kepada aplikasi atau *server* yang berlangganan (*subscriber*). Data yang diterima

selanjutnya disimpan dalam *database* dan digunakan sebagai masukan pada proses klasifikasi menggunakan algoritma Random Forest serta sistem peringatan dini.

Selain mengirimkan data hasil pembacaan sensor ke broker MQTT, ESP32 *Transmitter* juga mengirim data yang dikemas dalam bentuk struktur data (*data packet*) dan dikirimkan secara nirkabel menggunakan protokol ESP-NOW menuju ESP32 *Receiver*. ESP32 *Receiver* berfungsi menerima data yang dikirimkan dan menggunakannya sebagai dasar dalam mengendalikan aktuator, seperti aerator, LCD dan *buzzer*. Dengan mekanisme ini, proses pemantauan kualitas air dan pengendalian aktuator dapat berlangsung secara bersamaan, di mana data kualitas air tetap dikirimkan ke *server* untuk keperluan *monitoring*, sementara aktuator dapat merespons perubahan kondisi kualitas air secara cepat melalui komunikasi ESP-NOW.

3.4.2 Perancangan *Database*

Perancangan *database* digunakan untuk mengelola penyimpanan data yang digunakan pada sistem peringatan dini dan klasifikasi kualitas air. *Database* berfungsi sebagai media penyimpanan data sensor, hasil prediksi DO virtual, hasil klasifikasi kualitas air serta user_id telegram pengguna untuk notifikasi.

Database yang digunakan adalah postgreSQL yang diimplementasikan melalui platform supabase. Pemilihan PostgreSQL didasarkan pada kemampuannya dalam mengelola data secara terstruktur, mendukung penyimpanan data dalam jumlah besar, serta memiliki integrasi yang baik dengan *backend server* dan aplikasi berbasis web. Seluruh data yang diterima dari sensor akan disimpan sebagai data historis sehingga dapat digunakan untuk proses *monitoring*, prediksi DO virtual, klasifikasi kualitas air, serta pengiriman notifikasi kepada pengguna.

Struktur *database* dirancang untuk memastikan proses penyimpanan, pengelolaan, dan pengambilan data dapat dilakukan secara efisien. Hubungan antar tabel berfungsi untuk menjaga konsistensi data serta memudahkan proses pengolahan dan penyajian informasi pada *dashboard monitoring*.

Tabel 3. 2 Struktur Tabel Data Sensor

Name	*Type*	Keterangan
Id	*Int8*	Identitas setiap data pengukuran
Temp	*Float4*	Menyimpan nilai suhu air dalam °C.
Ph	*Float4*	Menyimpan nilai pH air.
Tds	*Float4*	Menyimpan nilai TDS dalam ppm.
Do_real	*Float4*	Menyimpan nilai DO dalam mg/L.
Do_virtual	*Float4*	Menyimpan nilai DO hasil prediksi
Aerator	*Bool*	Menyimpan status aktuator aerator
Kualitas_air	*Text*	Menyimpan hasil klasifikasi kualitas air
Created_at	*Timestamp*	Menyimpan tanggal dan waktu saat data direkam ke dalam *database.*

Berdasarkan Tabel 3. 2, struktur *database* pada sistem menggunakan tabel data_sensor yang berfungsi untuk menyimpan seluruh data hasil pemantauan kualitas air tambak udang vaname. Data yang disimpan meliputi hasil pembacaan sensor suhu (*temp*), derajat keasaman (ph), *Total Dissolved Solids* (tds), dan *Dissolved Oxygen* aktual (*do_real*). Selain itu, tabel ini juga menyimpan nilai Do virtual (do_virtual) yang merupakan hasil prediksi dan digunakan sebagai dasar dalam proses klasifikasi kualitas air.

Tabel data_sensor juga menyimpan hasil klasifikasi kualitas air pada atribut kualitas_air, status aktuator aerator pada atribut aerator, serta waktu pencatatan data pada atribut *created_at*. Penyimpanan informasi tersebut memungkinkan sistem melakukan pemantauan kondisi kualitas air secara *real-time*, menyimpan riwayat pengukuran, serta mendukung proses analisis dan evaluasi terhadap kinerja sistem klasifikasi.

Tabel 3. 3 Struktur Tabel Telegram *User*

Name	*Type*	Keterangan
Id	*Int8*	Identitas setiap pengguna telegram yang terdaftar
Created_at	*Timestamp*	Menyimpan tanggal dan waktu saat pengguna terdaftar pada sistem
Chat_id	*text*	Menyimpan *chat id* telegram yang digunakan sebagai tujuan pengiriman notifikasi

Berdasarkan Tabel 3. 3, tabel *telegram_users* digunakan untuk menyimpan data pengguna yang menjadi tujuan pengiriman notifikasi melalui Telegram. Tabel ini mendukung sistem peringatan dini dengan menyediakan daftar penerima notifikasi secara otomatis. Notifikasi dikirim ketika nilai DO virtual melewati ambang batas dengan menunjukkan hasil pembacaan semua sensor sehingga pengguna dapat segera melakukan tindakan penanganan.

3.4.3 Perancangan *Backend*

Backend dirancang sebagai pusat pengelolaan data yang menghubungkan perangkat *monitoring*, layanan komunikasi data, *database,* model *machine learning, dashboard* web, dan sistem notifikasi Telegram. *Backend* dibangun menggunakan beberapa layanan (*service*) yang saling terintegrasi, yaitu Node.js, Python, dan Supabase PostgreSQL.

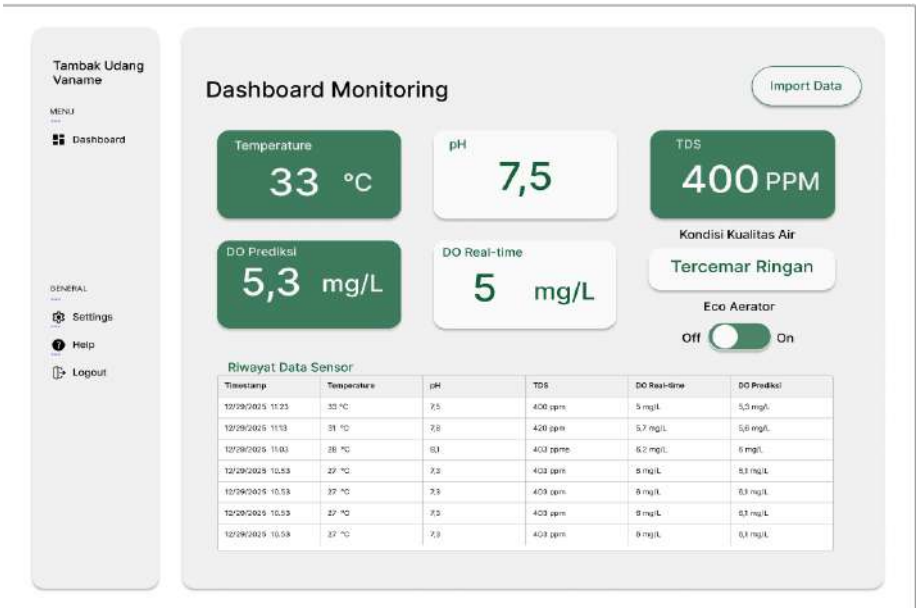
Data hasil pembacaan sensor yang dikirim oleh ESP32 *Transmitter* melalui protokol MQTT terlebih dahulu diterima oleh layanan *mqtt-server* yang dikembangkan menggunakan Node.js. Layanan ini berfungsi sebagai *subscriber* pada broker HiveMQ untuk menerima data sensor secara *real-time*. Setelah data diterima, sistem melakukan validasi sederhana terhadap format data sebelum menyimpannya ke dalam *database* PostgreSQL yang dikelola melalui Supabase.

Data sensor yang telah tersimpan pada *database* selanjutnya digunakan oleh *AI Server* yang dikembangkan menggunakan Python untuk melakukan proses prediksi nilai DO virtual menggunakan model *machine learning* yang telah dilatih sebelumnya. Nilai pH dan suhu yang dihasilkan kemudian digunakan sebagai dasar dalam proses klasifikasi kualitas air berdasarkan aturan yang telah ditetapkan. Hasil prediksi dan klasifikasi selanjutnya diperbarui kembali ke dalam *database* sehingga dapat diakses oleh komponen sistem lainnya.

Backend juga menyediakan layanan bagi *dashboard* web yang dibangun menggunakan Next.js. *Dashboard* mengambil data secara langsung dari Supabase sehingga pengguna dapat memantau kondisi kualitas air, hasil prediksi DO virtual, status aktuator, serta kategori kualitas air secara *real-time*.

3.4.4 Perancangan *Dashboard Monitoring*

Perancangan *software* sistem ditunjukkan pada Gambar 3. 13, yang menggambarkan perancangan dan integrasi komponen *software* pada sistem *monitoring* kualitas air tambak udang.



Gambar 3. 13 Rancangan Desain *Dashboard* Web Sistem
(Sumber: Penulis)

Berdasarkan gambar tersebut, perancangan sistem *monitoring* dan klasifikasi kualitas air tambak udang vaname di implementasikan dalam bentuk antarmuka *dashboard* berbasis web. *Dashboard* ini berfungsi sebagai media untuk menampilkan informasi hasil pengukuran sensor dan hasil pemrosesan data secara *real-time*.

Dengan adanya perancangan *software* ini, pengguna dapat memantau kondisi kualitas air tambak secara mudah. serta memperoleh informasi peringatan dini apabila terjadi penurunan kualitas air yang berpotensi membahayakan organisme budidaya.

3.4.5 Konversi dan Pengolahan Data Sensor

Proses konversi data sensor dilakukan untuk mengubah hasil pembacaan sensor menjadi nilai parameter kualitas air yang sesuai dengan satuan pengukuran masing-masing. Setiap sensor memiliki karakteristik keluaran yang berbeda sehingga diperlukan persamaan konversi agar data yang diperoleh dapat digunakan

pada tahap pengolahan data selanjutnya. Persamaan konversi yang digunakan meliputi sensor suhu, pH, *Total Dissolved Solids* (TDS), dan *Dissolved Oxygen* (DO).

1. Konversi Sensor Suhu DS18B20

Sensor Suhu DS18B20 menghasilkan data dalam bentuk digital yang dikirimkan melalui protokol komunikasi *One-Wire.* Data tersebut dibaca oleh mikrokontroler ESP32 menggunakan *library DallasTemperature*, kemudian dikonversi menjadi nilai suhu dalam satuan derajat Celsius (°C) dengan tipe data *float*. Proses konversi dilakukan secara otomatis melalui fungsi *getTempCByIndex(0)*, sehingga nilai keluaran sensor dapat langsung digunakan pada proses pengolahan data sistem. Secara matematis, konversi nilai suhu mengikuti persamaan berikut.

$$T = \frac{D}{16} \tag{3.1}$$

Dengan

T = Suhu dalam derajat Celcius (°C)

D = nilai digital (*raw data*) yang dibaca oleh *register* sensor suhu.

Proses pembacaan suhu diawali dengan pemanggilan fungsi *sensors.requestTemperatures();* untuk meminta sensor melakukan konversi suhu. Selanjutnya, nilai suhu hasil konversi dibaca dan disimpan ke dalam variabel suhuDS18B20 menggunakan perintah suhuDS18B20 = *sensors.getTempCByIndex(0);*

Pada implementasinya, konversi tersebut dilakukan secara otomatis melalui fungsi *getTempCByIndex(0)*, sehingga ESP32 langsung memperoleh nilai suhu dalam satuan derajat Celsius tanpa perlu melakukan perhitungan secara manual.

2. Konversi Sensor TDS DFRobot

Sensor TDS DFRobot menghasilkan keluaran berupa nilai analog yang dibaca oleh ADC pada mikrokontroler ESP32. Nilai ADC tersebut kemudian dikonversi menjadi tegangan, selanjutnya dihitung menjadi nilai *Electrical*

Conductivity (EC), dikompensasi terhadap suhu referensi, dan dikonversi menjadi nilai *Total Dissolved Solids* (TDS). Persamaan yang digunakan pada proses konversi sensor TDS ditunjukkan sebagai berikut.

$$V = \frac{ADC}{4095} \times V_{ref} \tag{3.2}$$

Dengan

V = Tegangan hasil konversi (Volt)

ADC = nilai digital hasil pembacaan ADC

V_{ref} = Tegangan referensi ADC sebesar 3,3 V.

Setelah diperoleh nilai tegangan, program fungsi *readTDS()* menghitungnya menjadi nilai *Electrical Conductivity* (EC) menggunakan persamaan polinomial hasil kalibrasi sensor sebagai berikut.

$$EC = (133.42V^3 - 255.86V^2 + 857.39V) \times K \tag{3.3}$$

Dengan

EC = Nilai konduktivitas listrik

V = Tegangan keluaran sensor

K = Faktor kalibrasi

Nilai EC kemudian dikompensasi terhadap suhu referensi 25°C menggunakan persamaan berikut.

$$EC_{25} = \frac{EC}{1 + 0.02(T - 25)} \tag{3.4}$$

Dengan

EC_{25} = Nilai konduktivitas pada suhu referensi 25°C

T = Suhu air

Tahap terakhir adalah mengonversi nilai konduktivitas yang telah dikompensasi menjadi nilai *Total Dissolved Solids* (TDS) menggunakan persamaan berikut.

$$TDS = EC_{25} \times 0.5 \tag{3.5}$$

Dengan

TDS = Nilai *Total Dissolved Solids*

EC_{25} = Nilai konduktivitas pada suhu referensi 25°C

3. Konversi Sensor pH DFRobot

Sensor pH menghasilkan keluaran berupa tegangan analog yang dibaca oleh ADC mikrokontroler ESP32. Nilai ADC yang diperoleh kemudian dikonversi menjadi tegangan keluaran sensor. Kemudian, nilai tegangan digunakan untuk menghitung nilai pH berdasarkan persamaan regresi linear hasil kalibrasi menggunakan larutan buffer pH 7 dan pH 4. Persamaan yang digunakan pada proses konversi sensor pH dijelaskan sebagai berikut.

$$V = \frac{ADC}{4095} \times V_{ref} \tag{3.6}$$

Dengan

V = Tegangan keluaran sensor (mV)

ADC = Nilai rata-rata hasil pembacaan ADC

V_{ref} = Tegangan referensi ADC sebesar 3300 mV

Nilai tegangan hasil pembacaan kemudian digunakan untuk menghitung nilai kemiringan garis (*slope*) menggunakan persamaan berikut.

$$Slope = \frac{7 - 4}{\left(\frac{V_7 - 1500}{3}\right) - \left(\frac{V_4 - 1500}{3}\right)} \tag{3.7}$$

Dengan

Slope = Gradien hubungan tegangan terhadap pH

V_7 = Tegangan hasil kalibrasi pada larutan buffer pH 7 (mV)

V_4 = Tegangan hasil kalibrasi pada larutan buffer pH 4 (mV)

Selanjutnya, nilai konstanta (*intercept*) dihitung menggunakan persamaan berikut.

$$Intercept = 7 - Slope\left(\frac{V_7 - 1500}{3}\right) \tag{3.8}$$

Dengan

Intercept = konstanta persamaan regresi linear

Setelah nilai *slope* dan *intercept* diperoleh, nilai pH dihitung menggunakan persamaan berikut.

$$\text{pH} = Slope\left(\frac{V-1500}{3}\right) + Intercept \tag{3.9}$$

Setelah nilai pH diperoleh, program tidak langsung menggunakannya sebagai hasil akhir. Nilai tersebut terlebih dahulu disimpan ke dalam *buffer* sebanyak 10 data, kemudian dihitung nilai rata-ratanya menggunakan metode *Moving Average* untuk mengurangi fluktuasi pembacaan akibat gangguan (*noise*). Selanjutnya, setiap nilai rata-rata tersebut diakumulasikan selama interval pengiriman data, kemudian dihitung kembali rata-ratanya sehingga diperoleh nilai pH *Final* yang lebih stabil sebelum dikirim melalui komunikasi ESP-NOW dan MQTT.

4. Konversi Sensor DO DFRobot

Sensor DO DFRobot menghasilkan keluaran berupa tegangan analog yang dibaca oleh ADC pada mikrokontroler ESP32. Nilai ADC yang diperoleh kemudian dikonversi menjadi tegangan keluaran sensor menggunakan tegangan referensi ADC sebesar 3300 mV. Persamaan konversi ADC menjadi tegangan ditunjukkan pada persamaan berikut.

$$V = \frac{ADC}{4095} \times V_{ref} \tag{3.10}$$

Dengan

V = Tegangan keluaran sensor (mV)

ADC = Nilai rata-rata hasil pembacaan ADC

V_{ref} = Tegangan referensi ADC sebesar 3300 mV

Nilai tegangan yang diperoleh selanjutnya digunakan untuk menghitung tegangan saturasi sensor (*saturation voltage*) menggunakan persamaan berikut.

$$V_{sat} = \mathrm{CAL1}_V + 35(T - \mathrm{CAL1}_T) \quad (3.11)$$

Dengan

V_{sat} = Tegangan saturasi sensor (mV)

$\mathrm{CAL1}_V$ = Tegangan hasil kalibrasi sensor

$\mathrm{CAL1}_T$ = Suhu saat proses kalibrasi

T = Suhu air (°C)

Selanjutnya, nilai *Dissolved Oxygen* (DO) dihitung menggunakan nilai tegangan sensor, tegangan saturasi, serta nilai DO jenuh berdasarkan suhu yang diperoleh dari *DO_Table*, sebagaimana ditunjukkan pada persamaan berikut.

$$DO = \frac{V \times DO_{Table}}{V_{sat}} \quad (3.12)$$

Dengan

DO = Konsentrasi oksigen terlarut

V = Tegangan keluaran sensor

DO_{Table} = Nilai DO jenuh berdasarkan suhu

V_{sat} = Tegangan saturasi sensor

Nilai DO yang diperoleh selanjutnya dibagi dengan faktor skala 1000 sehingga diperoleh nilai *Dissolved Oxygen* dalam satuan mg/L sebelum digunakan pada proses *monitoring* dan pengolahan data. Setelah nilai *Dissolved Oxygen* (DO) diperoleh, program tidak langsung menggunakannya sebagai hasil akhir. Nilai DO dari setiap proses pembacaan terlebih dahulu diakumulasikan ke dalam variabel *doSum*, kemudian jumlah pembacaan dicatat pada variabel *avgCount*. Selanjutnya, pada setiap interval pengiriman data, program menghitung nilai rata-rata seluruh hasil pembacaan DO sehingga diperoleh nilai DO Final yang lebih stabil sebelum dikirim melalui komunikasi ESP-NOW dan MQTT.

3.5 Penerapan Model Klasifikasi Random Forest

Pada Subbab ini menjelaskan penerapan model klasifikasi Random Forest yang digunakan pada sistem. Pembahasan mencakup proses pengolahan data

sensor, tahapan inisialisasi model, pelatihan model, proses klasifikasi kualitas air, serta implementasi model random forest pada sistem kualitas air berbasis web.

3.5.1 Pembersihan Data dan Penanganan *Outlier*

Pengolahan data sensor merupakan tahap awal sebelum model klasifikasi Random Forest diterapkan. Data yang digunakan diperoleh dari sensor kualitas air yang mengukur empat parameter, yaitu suhu, pH, dan DO.

Proses *preprocessing* meliputi penanganan *outlier* menggunakan metode *Interquartile Range* (IQR) serta pelabelan data berdasarkan nilai *Dissolved Oxygen* (DO) virtual. Tahap *preprocessing* bertujuan untuk meningkatkan kualitas *dataset* sehingga data yang digunakan pada proses pelatihan mampu menghasilkan model klasifikasi dengan performa yang lebih baik.

Penanganan *outlier* dilakukan menggunakan metode *Interquartile Range* (IQR) karena metode ini mampu mengidentifikasi data yang memiliki nilai ekstrem berdasarkan sebaran data tanpa mengharuskan data mengikuti distribusi tertentu. Pada metode IQR, terlebih dahulu dihitung nilai kuartil pertama (Q1) dan kuartil ketiga (Q3), kemudian nilai *Interquartile Range* (IQR) diperoleh menggunakan persamaan berikut.

$$IQR = Q3 - Q1 \tag{3.13}$$

Dengan

IQR = *Interquartile Range*

Q1 = Kuartil pertama (25%)

Q3 = Kuartil ketiga (75%)

Selanjutnya dihitung batas bawah data normal (*lower bound*) dan batas atas data normal (*upper bound*) menggunakan persamaan berikut.

$$Lower = Q_1 - 1.5 \times IQR \tag{3.14}$$

$$Upper = Q_3 + 1.5 \times IQR \tag{3.15}$$

Data yang berada pada rentang nilai *lower bound* hingga *upper bound* dipertahankan sebagai data normal. Sebaliknya, data yang berada di luar rentang tersebut dikategorikan sebagai *outlier* dan ditangani sebelum digunakan pada proses pelatihan model Random Forest.

Metode IQR diterapkan pada setiap parameter yang digunakan dalam penelitian, yaitu suhu (*temperature*), pH, dan DO virtual. *Dataset* yang telah melalui proses penanganan *outlier* selanjutnya digunakan pada tahap pelabelan data berdasarkan nilai DO virtual.

Setelah proses penanganan *outlier* selesai dilakukan, data kualitas air diproses lebih lanjut dengan melakukan pelabelan berdasarkan nilai *Dissolved Oxygen* (DO) virtual. Pelabelan dilakukan dengan mengelompokkan nilai DO virtual ke dalam tiga kategori kualitas air, yaitu tercemar ringan, tercemar sedang, dan tercemar berat, sesuai dengan rentang nilai DO yang telah ditentukan. Label yang dihasilkan selanjutnya digunakan sebagai target (label) dalam proses pelatihan algoritma Random Forest.

Selanjutnya, parameter pH dan suhu digunakan sebagai atribut (*feature*) dalam pembentukan model klasifikasi. Algoritma Random Forest mempelajari hubungan antara kedua parameter tersebut dengan label kualitas air yang telah dibentuk, sehingga model dapat digunakan untuk memprediksi kategori kualitas air berdasarkan data sensor yang diperoleh.

Tabel 3. 4 Rentang Optimal Parameter Kualitas Air

Parameter	Rentang Optimal	Keterangan
pH	6 - 9	Rentang pH yang direkomendasikan untuk budidaya udang vaname.
TDS	< 1000 ppm	Rentang TDS yang direkomendasikan untuk budidaya udang vaname.
Suhu	27 –33°C	Rentang suhu optimal untuk budidaya udang vaname.

Berdasarkan Tabel 3. 4, parameter pH dan suhu digunakan sebagai atribut (*feature*) dalam pembentukan model klasifikasi. Sementara itu, nilai *Dissolved Oxygen* (DO) virtual digunakan sebagai target (*label*) yang diperoleh melalui proses pelabelan berdasarkan kategori kualitas air. *Dataset* yang telah memiliki atribut dan label selanjutnya digunakan dalam proses pelatihan algoritma Random Forest.

Setelah seluruh proses pengolahan data dan pelabelan selesai dilakukan, *dataset* dibagi menjadi data latih (*training set*) dan data uji (*testing set*) dengan perbandingan 70:30, yaitu 70% data digunakan sebagai data latih dan 30% sisanya digunakan sebagai data uji. Data latih digunakan untuk membangun model Random Forest sehingga model dapat mempelajari pola hubungan antara parameter pH dan

suhu terhadap kategori kualitas air. Sementara itu, data uji digunakan untuk mengevaluasi performa model yang telah dilatih, sehingga dapat diketahui tingkat akurasi serta kemampuan model dalam melakukan prediksi terhadap data yang belum pernah digunakan selama proses pelatihan.

Metode *Synthetic Minority Over-sampling Technique* (SMOTE) diterapkan untuk mengatasi ketidakseimbangan jumlah data antar kelas. Metode ini menghasilkan sampel sintetis pada kelas minoritas yang memanfaatkan data yang sudah ada agar distribusi jumlah data pada setiap kelas menjadi seimbang. Proses penerapan SMOTE hanya dilakukan pada data latih agar tidak terjadi *data leakage* yang menyebabkan hasil evaluasi menjadi bias. Data latih yang telah diseimbangkan kemudian digunakan dalam proses pelatihan algoritma Random Forest.

3.5.2 Inisialisasi Model

Inisialisasi model merupakan tahap penentuan konfigurasi parameter sebelum model klasifikasi Random Forest dibangun dan dilatih. Pada tahap ini, parameter model ditetapkan agar proses pelatihan dapat berjalan secara konsisten dan menghasilkan model yang mampu melakukan klasifikasi kualitas air.

Algoritma Random Forest dibangun menggunakan 100 pohon keputusan (*n_estimators* = 100). Setiap pohon keputusan dibangun dari sampel data latih yang dipilih secara acak (*bootstrap sampling*), kemudian hasil prediksi seluruh pohon digabungkan menggunakan mekanisme *majority voting* untuk menentukan kelas akhir. Penggunaan sejumlah pohon keputusan bertujuan untuk meningkatkan stabilitas model dan mengurangi variasi hasil prediksi.

Selain itu, parameter *random_state* = 42 digunakan untuk mengontrol proses pembangkitan bilangan acak selama pembentukan model. Penetapan parameter tersebut bertujuan agar proses pelatihan dapat direproduksi sehingga menghasilkan keluaran yang konsisten pada setiap proses pelatihan maupun pengujian.

3.5.3 Pelatihan Model

Pelatihan model merupakan tahap pembentukan model Random Forest dengan mempelajari hubungan antara parameter kualitas air dan kelas kualitas air. Proses pelatihan dilakukan menggunakan data latih yang telah diseimbangkan menggunakan metode *Synthetic Minority Over-sampling Technique* (SMOTE) sehingga distribusi data pada setiap kelas menjadi lebih seimbang.

Pada proses pelatihan, algoritma Random Forest membangun sejumlah pohon keputusan menggunakan teknik *bootstrap aggregating* (*bagging*). Setiap pohon keputusan dibangun dari subset data latih yang dipilih secara acak dengan pengembalian (*sampling with replacement*), sehingga setiap pohon memiliki komposisi data yang berbeda. Selanjutnya, pada setiap simpul pohon dilakukan pemilihan atribut terbaik berdasarkan kriteria *Gini Impurity* untuk menghasilkan pemisahan data yang optimal. Pendekatan ini menghasilkan kumpulan pohon keputusan yang beragam sehingga mampu meningkatkan kemampuan generalisasi model dalam melakukan klasifikasi terhadap data baru.

Setelah seluruh pohon keputusan terbentuk, dilakukan proses *fitting* menggunakan data latih yang telah disiapkan. Pada tahap ini, setiap pohon keputusan dilatih menggunakan subset data hasil *bootstrap sampling* sehingga mampu mempelajari pola dari data latih.

3.5.4 Proses Klasifikasi Kualitas Air

Proses klasifikasi kualitas air merupakan tahap penerapan model Random Forest yang telah dilatih untuk mengklasifikasikan kondisi kualitas air tambak secara *real-time*. Pada tahap ini, model menerima data masukan (*input*) berupa nilai parameter pH dan suhu yang diperoleh dari sistem pemantauan kualitas air.

Pada proses klasifikasi, setiap data masukan diproses oleh seluruh pohon keputusan yang telah dibangun selama proses pelatihan. Setiap pohon keputusan menghasilkan prediksi kelas secara independen berdasarkan pola hubungan antara parameter pH dan suhu dengan kategori kualitas air yang telah dipelajari sebelumnya. Hasil prediksi dari seluruh pohon keputusan digabungkan menggunakan mekanisme *majority voting*, yaitu kelas yang memperoleh jumlah suara terbanyak ditetapkan sebagai hasil klasifikasi akhir.

Hasil klasifikasi yang dihasilkan oleh model terdiri atas tiga kategori kualitas air, yaitu tercemar ringan, tercemar sedang, dan tercemar berat. Kategori tercemar ringan menunjukkan kondisi kualitas air yang masih berada pada tingkat risiko rendah, tercemar sedang menunjukkan adanya penurunan kualitas air yang memerlukan perhatian, sedangkan tercemar berat menunjukkan kondisi kualitas air yang memerlukan penanganan segera. Hasil klasifikasi kemudian ditampilkan secara *real-time* melalui sistem berbasis web sehingga pengguna dapat memantau kondisi kualitas air secara langsung.

3.5.5 Implementasi Model Random Forest pada *Website*

Implementasi Implementasi model Random Forest berbasis web merupakan tahap penerapan model klasifikasi yang telah dilatih ke dalam sistem *monitoring* kualitas air tambak berbasis web. Sistem web dibangun menggunakan *framework* Next.js yang memungkinkan penyajian antarmuka yang responsif dan interaktif sehingga pengguna dapat memantau kondisi kualitas air secara mudah.

Model Random Forest yang telah selesai dilatih disimpan dalam format *pickle* (.pkl) pada sisi *backend*. Penyimpanan model dalam format tersebut bertujuan agar model dapat digunakan kembali tanpa perlu melalui proses pelatihan ulang setiap kali sistem dijalankan. Ketika sistem menerima data baru, *backend* memuat model yang telah disimpan dan melakukan proses klasifikasi berdasarkan nilai parameter pH dan suhu yang diterima. Hasil klasifikasi kemudian berupa kategori tercemar ringan, tercemar sedang, atau tercemar berat.

Selanjutnya, data sensor beserta hasil klasifikasi disimpan ke dalam *database* menggunakan Supabase dengan PostgreSQL sebagai sistem manajemen basis data. Penyimpanan data dilakukan agar riwayat hasil klasifikasi dapat diakses kembali untuk keperluan pemantauan maupun analisis. Seluruh data tersebut kemudian ditampilkan melalui antarmuka web yang mencakup *dashboard monitoring*, grafik perkembangan kualitas air, riwayat data sensor, serta hasil klasifikasi kualitas air sehingga pengguna dapat memantau kondisi tambak secara *real-time*.

3.6 Evaluasi Performa Model Klasifikasi

Pada subbab ini menjelaskan evaluasi performa model klasifikasi Random Forest yang digunakan pada sistem. Pembahasan mencakup pengukuran akurasi sensor, kinerja model menggunakan *confusion matrix* serta perhitungan metrik evaluasi yang meliputi *accuracy, precision, recall, dan F1-score* untuk mengetahui tingkat keberhasilan model dalam mengklasifikasikan kualitas air tambak.

3.6.1 Evaluasi Akurasi Sensor

Tahap pengujian sensor dilakukan dengan cara membandingkan hasil pembacaan sensor terhadap alat ukur standar sebagai nilai acuan, sehingga tingkat akurasi sensor dapat dianalisis pada berbagai kondisi pengukuran. Proses evaluasi ini diperlukan agar data yang dihasilkan sensor bersifat valid dan sesuai dengan standar pengukuran yang berlaku. Selain itu, selisih nilai sensor dan nilai referensi dihitung dalam bentuk persentase *error* untuk mengetahui besarnya penyimpangan yang terjadi. Persentase *error* tersebut diperoleh menggunakan rumus berikut.

$$\%Error = \left|\frac{\text{Nilai Referensi} - \text{Nilai Sensor}}{\text{Nilai Referensi}}\right| \text{ x } 100\% \qquad (3.16)$$

Melalui perhitungan tersebut, dapat ditentukan tingkat deviasi pembacaan sensor terhadap nilai sebenarnya. Persentase *error* yang diperoleh digunakan sebagai indikator untuk mengevaluasi tingkat akurasi pembacaan sensor terhadap nilai referensi. Nilai tersebut selanjutnya digunakan sebagai dasar dalam menilai keandalan sensor pada tahap pengujian sistem.

3.6.2 *Confusion Matrix*

Confusion matrix merupakan metode evaluasi yang digunakan untuk mengukur performa model klasifikasi dengan membandingkan hasil prediksi model terhadap kelas yang sebenarnya. *Confusion matrix* disusun dalam bentuk matriks yang terdiri atas tiga kelas, yaitu tercemar ringan, tercemar sedang, dan tercemar berat. Nilai pada diagonal utama menunjukkan jumlah data yang berhasil diklasifikasikan dengan benar, sedangkan nilai di luar diagonal menunjukkan jumlah data yang mengalami kesalahan klasifikasi.

Tabel 3. 5 Ilustrasi *Confusion Matrix* Tiga Kelas

Aktual	Prediksi ringan	Prediksi Sedang	Prediksi Berat
Ringan	n_{11}	n_{12}	n_{13}
Sedang	n_{21}	n_{22}	n_{23}
Berat	n_{31}	n_{32}	n_{33}

Berdasarkan Tabel 3. 5, elemen pada diagonal utama (n_{11}, n_{22}, dan n_{33}) menunjukkan jumlah data yang berhasil diklasifikasikan dengan benar pada masing-masing kelas. Sementara itu, elemen di luar diagonal menunjukkan jumlah data yang mengalami kesalahan klasifikasi ke kelas lainnya.

Pada klasifikasi multikelas, nilai *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), dan *True Negative* (TN) dihitung untuk masing-masing kelas menggunakan pendekatan *one-vs-rest*. Nilai-nilai tersebut selanjutnya digunakan sebagai dasar dalam perhitungan metrik evaluasi, yaitu *accuracy, precision, recall,* dan *F1-score*, untuk menilai performa model secara menyeluruh.

3.6.3 *Accuracy, Precision, Recall, dan F1-Score*

Evaluasi performa model klasifikasi Random Forest dilakukan menggunakan metrik *accuracy, precision, recall,* dan *F1-score*. Seluruh metrik tersebut dihitung berdasarkan nilai *confusion matrix* yang diperoleh dari hasil pengujian model terhadap data uji.

Nilai *accuracy* digunakan untuk mengukur tingkat ketepatan model secara keseluruhan, sedangkan *precision* digunakan untuk mengevaluasi ketepatan model dalam melakukan prediksi pada setiap kelas. Selanjutnya, *recall* digunakan untuk mengukur kemampuan model dalam mendeteksi seluruh data pada masing-masing kelas, sementara *F1-score* digunakan untuk memberikan evaluasi yang lebih seimbang dengan mempertimbangkan nilai *precision* dan *recall.*

Nilai *accuracy*, *precision*, *recall*, dan *F1-score* dihitung menggunakan fungsi *accuracy_score()* dan *classification_report()* dari pustaka Scikit-learn. Hasil evaluasi digunakan untuk menilai kinerja model Random Forest dalam mengklasifikasikan kualitas air.

“Halaman ini sengaja dikosongkan”

BAB 4
HASIL DAN PEMBAHASAN

Bab 4 menyajikan hasil dan pembahasan mengenai perancangan, implementasi, dan pengujian sistem *monitoring* kualitas air tambak udang. Pembahasan meliputi perancangan sistem *monitoring*, penerapan metode Random Forest menggunakan data sensor dengan DO virtual, serta evaluasi performa model dalam mengklasifikasikan kualitas air tambak secara otomatis.

4.1 Pengujian Sensor dan Aktuator

Pada subbab ini membahas hasil pengujian sensor dan aktuator yang digunakan pada sistem. Pengujian dilakukan untuk mengetahui akurasi masing-masing komponen berdasarkan data hasil pengukuran yang diperoleh. Hasil pengujian ini selanjutnya digunakan sebagai dasar dalam menilai keandalan sistem secara keseluruhan serta sebagai pertimbangan dalam melakukan kalibrasi atau perbaikan apabila diperlukan.

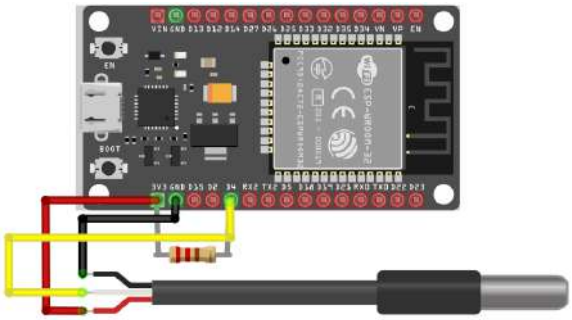
4.1.1 Hasil Pengujian Sensor Suhu

Sensor suhu DS18B20 digunakan untuk mengukur suhu pada sistem secara akurat. Data hasil pengukuran kemudian dikirimkan ke ESP32 melalui protokol komunikasi *One-Wire* untuk diproses, sehingga sistem dapat memantau kondisi suhu secara *real-time.*

Sensor DS18B20 memiliki konverter analog-ke-digital (ADC) terintegrasi sehingga data suhu yang dihasilkan langsung berupa data digital tanpa memerlukan rangkaian konversi analog ke digital eksternal. Selain itu, protokol *One-Wire* memungkinkan beberapa sensor dihubungkan hanya menggunakan satu jalur data. Sebagai contoh, apabila nilai digital yang diterima adalah 480, maka suhu yang diperoleh dapat dihitung menggunakan persamaan (3.1).

$$T = \frac{480}{16} = 30°\text{C}$$

Konfigurasi pin pada rangkaian sensor suhu DS18B20 dapat dilihat pada Gambar 4. 1, dengan penjelasan terkait alokasi pin yang digunakan dijelaskan dalam Tabel 4. 1.



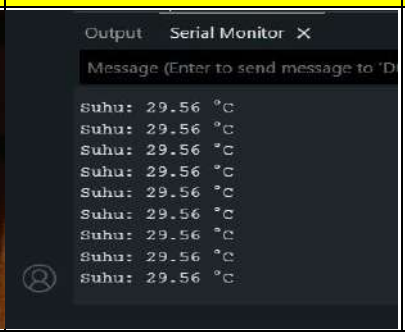
Gambar 4. 1 *Wiring* ESP32 dengan Sensor Suhu DS18B20

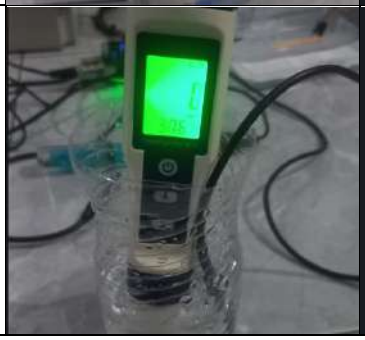
Tabel 4. 1 Konfigurasi ESP32 dengan Sensor Suhu DS18B20

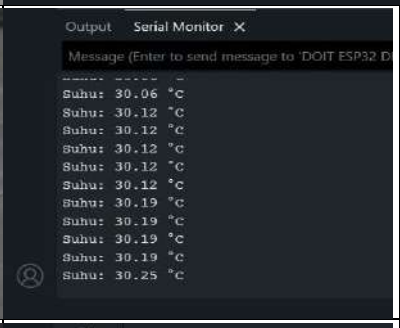
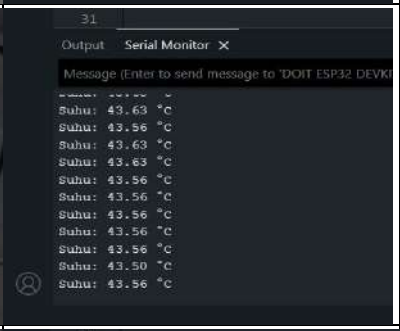
No.	ESP32	Suhu DS18B20
1	3.3V	VCC
2	GND	GND
3	GPIO4 (D4)	Data

Pengujian sensor dilakukan dengan menggunakan sepuluh sampel pengujian. Nilai suhu setiap sampel diukur terlebih dahulu menggunakan termometer digital sebagai acuan. Selanjutnya, hasil pembacaan dari sensor DS18B20 dibandingkan dengan data dari termometer digital untuk menilai tingkat akurasi sensor tersebut.

Tabel 4. 2 Data Hasil Pengujian Sensor Suhu DS18B20

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
1			Lingkungan	0.203

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
2		Message (Enter to send message t Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.19 °C Suhu: 30.12 °C	Air Normal	1.339
3		Output Serial Monitor × Message (Enter to send message to 'DOIT ESP Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C Suhu: 22.12 °C	Air dingin	0.545
4		Output Serial Monitor × Message (Enter to send message to 'DOI Suhu: 42.94 °C Suhu: 42.94 °C Suhu: 42.94 °C Suhu: 42.88 °C Suhu: 42.94 °C Suhu: 42.94 °C Suhu: 42.94 °C Suhu: 42.94 °C Suhu: 42.88 °C	Air hangat	5.418
5		Output Serial Monitor × Message (Enter to send message to 'DOIT ESP Suhu: 28.37 °C Suhu: 28.31 °C Suhu: 28.31 °C Suhu: 28.37 °C Suhu: 28.31 °C Suhu: 28.31 °C Suhu: 28.37 °C Suhu: 28.31 °C Suhu: 28.31 °C Suhu: 28.31 °C Suhu: 28.31 °C	Air hujan	2.379
6		Output Serial Monitor × Message (Enter to send message to 'DOIT ESP32 DI Suhu: 30.62 °C Suhu: 30.56 °C Suhu: 30.56 °C Suhu: 30.56 °C Suhu: 30.56 °C Suhu: 30.56 °C Suhu: 30.50 °C Suhu: 30.56 °C Suhu: 30.50 °C Suhu: 30.50 °C Suhu: 30.50 °C	Air PDAM	0.130

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
7			Air panas	6.128
8			Air garam	0.333
9			Air hangat	4.052
10			Air normal	5.468
Rata-Rata Persentase *Error*				2.599

Tabel 4. 2 merupakan data hasil pengujian sensor suhu DS18B20 dan alat ukur termometer digital. Hasil dari pengujian, diperoleh nilai rata-rata *error* sebesar 2.599%. Hal ini menunjukkan bahwa meskipun ada sedikit perbedaan antara hasil pengukuran sensor suhu dan alat ukur digital, tingkat akurasi sensor tetap cukup baik dan dapat diterima untuk penggunaan dalam pengukuran suhu dalam sistem.

Secara keseluruhan, dapat disimpulkan bahwa sensor suhu DS18B20 berfungsi dengan baik, meskipun terdapat *error* kecil namun sensor masih memberikan pengukuran suhu yang memadai untuk digunakan pada sistem yang membutuhkan suhu secara *real-time*.

4.1.2 Hasil Pengujian Sensor TDS

Sensor TDS digunakan untuk mengukur tingkat kekeruhan atau kandungan zat terlarut dalam air. Data yang diperoleh dari sensor diproses oleh ESP32, sehingga sistem dapat memantau kualitas air secara *real-time*.

Sensor TDS bekerja berdasarkan prinsip pengukuran konduktivitas listrik dalam larutan, semakin tinggi kandungan zat terlarut, maka semakin besar nilai konduktivitas yang terdeteksi. Sensor ini menghasilkan sinyal analog yang kemudian diubah menjadi nilai digital oleh ADC pada ESP32 dan selanjutnya diolah menjadi nilai TDS (ppm). Sebagai contoh, apabila nilai ADC yang terbaca sebesar 1240, maka tegangan keluaran sensor diperoleh menggunakan persamaan (3.2).

$$V = \frac{1240}{4095} \times 3.3$$

$$V = 0.999 \text{ Volt}$$

Selanjutnya setelah diperoleh nilai tegangan, program fungsi readTDS() menghitungnya menjadi nilai Electrical Conductivity (EC) menggunakan persamaan polinomial hasil kalibrasi diperoleh menggunakan persamaan (3.3).

$$EC = (133.42(0.999)^3 - 255.86(0.999)^2 + 857.39(0.999)) \times 1.26$$

$$EC = 930.0 \text{ µS/cm.}$$

Faktor kalibrasi (K) merupakan faktor pengali yang digunakan untuk menyesuaikan hasil pembacaan sensor agar mendekati nilai referensi. Nilai K diperoleh melalui proses kalibrasi awal sensor dengan membandingkan hasil pengukuran sensor TDS terhadap alat ukur standar atau larutan standar yang memiliki nilai TDS atau EC diketahui. Berdasarkan hasil kalibrasi tersebut, digunakan faktor kalibrasi sebesar 1,26 sehingga hasil pengukuran sensor mendekati nilai referensi.

Nilai EC kemudian dikompensasi terhadap suhu referensi 25°C sehingga diperoleh menggunakan persamaan (3.4).

$$EC_{25} = \frac{930.0}{1 + 0.02(30 - 25)}$$

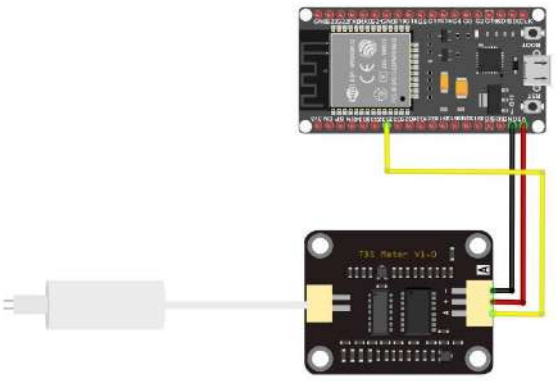
$$EC_{25} = 846.0\ \mu S/cm$$

Tahap terakhir adalah mengonversi nilai konduktivitas yang telah dikompensasi menjadi nilai *Total Dissolved Solids* (TDS) dihitung menggunakan persamaan (3.5).

$$TDS = 846.0 \times 0.5$$

$$TDS = 423.0\ ppm$$

Konfigurasi pin pada rangkaian sensor TDS dapat dilihat pada Gambar 4. 2, dengan penjelasan terkait alokasi pin yang digunakan dijelaskan dalam Tabel 4. 3.



Gambar 4. 2 *Wiring* ESP32 dengan Sensor TDS

Tabel 4. 3 Konfigurasi ESP32 dengan Sensor TDS

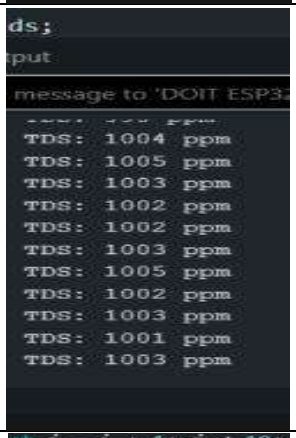
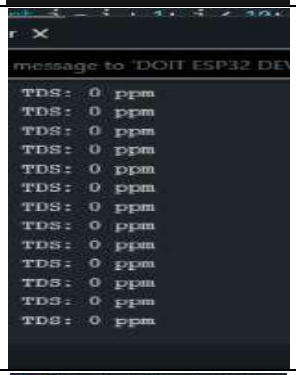
No.	ESP32	TDS
1	5V	VCC
2	GND	GND
3	GPIO32 (D32)	Data

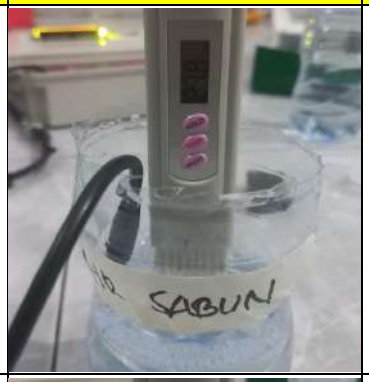
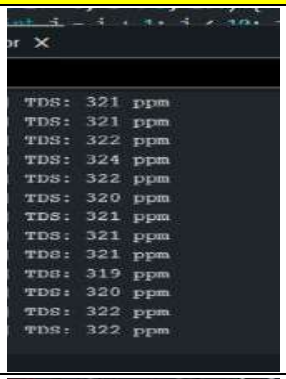
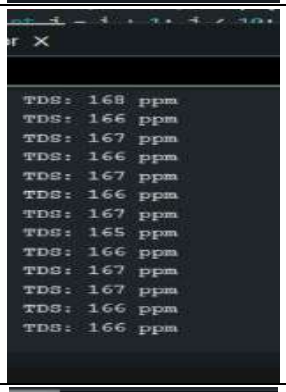
Pengujian sensor dilakukan dengan melakukan pengukuran TDS pada 10 jenis air yang berbeda, setiap jenis air memiliki karakteristik yang berbeda, yang meliputi tingkat salinitas dan kandungan mineral yang dapat memengaruhi hasil pengukuran TDS. Pengukuran dilakukan dengan menggunakan sensor TDS yang

telah dikalibrasi, kemudian hasil pengukuran tersebut dibandingkan dengan pengukuran alat ukur standart yaitu TDS meter sebagai acuan.

Tabel 4. 4 Data Hasil Pengujian Sensor TDS

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
1			Air Minum	1.785
2			Air sumur	3.846
3			Air buffer TDS 359 ppm	1.949

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
4			Air PDAM	1.932
5			Air buffer TDS 1000 ppm	0.3
6			Aquades	1
7			Air PDAM	1.287

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
8			Air sabun	8.053
9			Air gula	6.41
10			Air garam	*Error* tinggi karena batas sensor
Rata-Rata Persentase *Error*				2.951

Tabel 4. 4 merupakan data hasil pengujian sensor TDS dengan berbagai jenis air. Dari hasil pengujian, diperoleh nilai rata-rata *error* sebesar 2.951%. Hal ini menunjukkan bahwa meskipun ada beberapa perbedaan antara hasil pengukuran sensor TDS dengan alat ukur TDS meter, tingkat akurasi sensor masih cukup baik dan dapat diterima untuk pengukuran dalam pengukuran kualitas air. Sensor TDS memiliki keterbatasan dalam hal kapasitas pengukurannya. Berdasarkan *datasheet*, sensor hanya mampu mengukur kadar TDS hingga kurang lebih 1200 ppm. Pengukuran pada sampel dengan kadar TDS yang melebihi batas ini akan menghasilkan pembacaan tidak akurat atau *error* yang tinggi, seperti yang terlihat pada sampel air garam, yang melebihi kapasitas pengukuran sensor.

Dapat disimpulkan bahwa sensor TDS berfungsi dengan baik dalam pengukuran kadar padatan terlarut pada berbagai jenis air. Meskipun terdapat yang sedikit lebih tinggi pada beberapa sampel, terutama pada air dengan kandungan padatan tinggi seperti air garam, sensor tetap memberikan pengukuran yang memadai untuk sampel air lainnya.

4.1.3 Hasil Pengujian Sensor pH

Sensor pH digunakan untuk mengukur tingkat keasaman atau kebasaan pada sistem. Data yang diperoleh dari sensor pH diproses oleh ESP32, sehingga memungkinkan sistem untuk memantau kondisi pH secara *real-time*.

Sensor pH bekerja berdasarkan prinsip pengukuran potensi elektrokimia yang terjadi antara elektroda referensi dan elektroda pH. Sensor ini menghasilkan tegangan yang sebanding dengan nilai pH, yang kemudian diubah menjadi data digital oleh oleh ADC pada ESP32 dan selanjutnya diolah menjadi nilai pH. Sensor pH menggunakan koneksi analog untuk mengirimkan sinyal hasil pengukuran ke ESP32. Sebagai contoh, apabila nilai ADC yang terbaca sebesar 1469, maka tegangan keluaran sensor diperoleh menggunakan persamaan (3.6).

$$V = \frac{1469}{4095} \times 3300$$

V = 1184 mV

Setelah diperoleh nilai tegangan, fungsi *readPH()* menghitung nilai pH menggunakan persamaan regresi linear berdasarkan dua titik kalibrasi, yaitu larutan buffer pH 7 dan larutan buffer pH 4. Nilai tegangan hasil kalibrasi disimpan pada variabel *neutralVoltage* dan *acidVoltage*, kemudian dibaca dari memori EEPROM melalui fungsi *beginPH()* sehingga tetap digunakan meskipun sistem dimatikan, menggunakan persamaan (3.7).

$$Slope = \frac{7 - 4}{\left(\frac{1320 - 1500}{3}\right) - \left(\frac{1829 - 1500}{3}\right)}$$

$$Slope = \frac{3}{-60 - 109.67}$$

$$Slope = -0.01768$$

Selanjutnya dihitung nilai konstanta (*intercept*) menggunakan persamaan (3.8).

$$Intercept = 7 - (-0.01768\,x\ -\ 60)$$

$$Intercept = 7 - 1.0608$$

$$Intercept =\ 5.9392$$

Setelah nilai *slope* dan *intercept* diperoleh, nilai pH dihitung menggunakan persamaan (3.9).

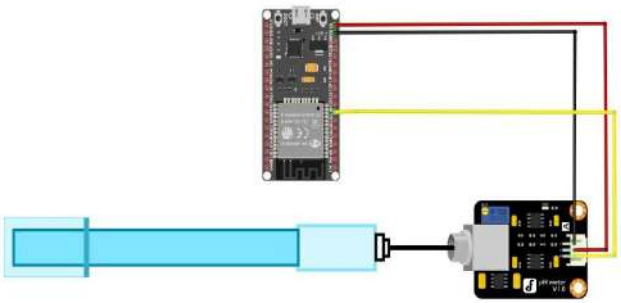
$$\text{pH} = -0.01768\left(\frac{1184 - 1500}{3}\right) + 5.9392$$

$$\text{pH} = -0.01768\,x\,(-105.33) + 5.9392$$

$$\text{pH} = 1.862 + 5.9392$$

$$\text{pH} =\ 7.8$$

Nilai pH yang diperoleh selanjutnya diproses menggunakan metode *Moving Average* sehingga dihasilkan nilai pembacaan yang lebih stabil sebelum digunakan pada proses *monitoring*. Konfigurasi pin pada rangkaian sensor pH dapat dilihat pada Gambar 4. 3, dengan penjelasan terkait alokasi pin yang digunakan dijelaskan dalam Tabel 4. 5.



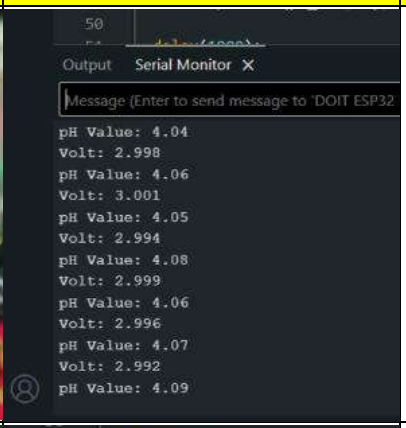
Gambar 4. 3 *Wiring* ESP32 dengan Sensor pH

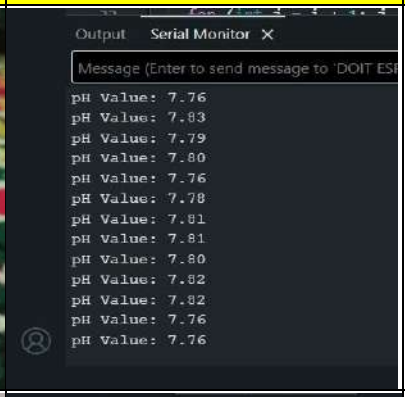
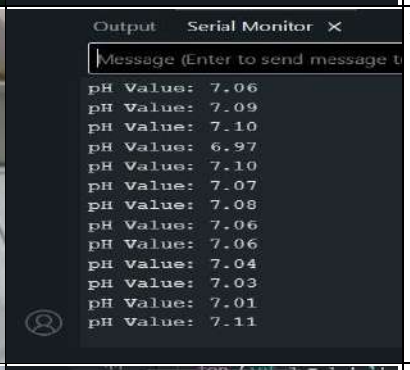
Tabel 4. 5 Konfigurasi ESP32 dengan Sensor pH

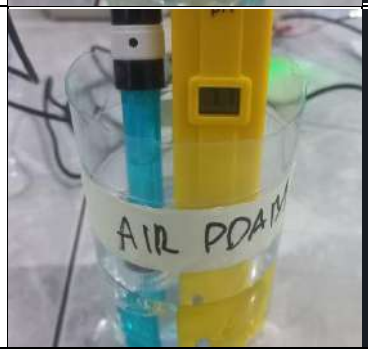
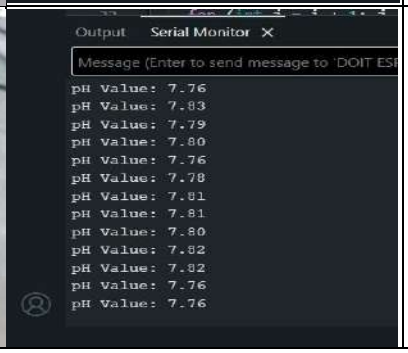
No.	ESP32	Ph
1	5V	VCC
2	GND	GND
3	GPIO36 (D36)	Data

Pengujian sensor pH dilakukan dengan melakukan pengukuran 10 jenis sampel air yang berbeda. Setiap jenis air memiliki karakteristik yang berbeda, seperti tingkat keasaman atau kebasaan, yang dapat memengaruhi hasil pembacaan sensor. Pengukuran dilakukan dengan menggunakan sensor pH yang telah dikalibrasi, kemudian hasil pengukuran tersebut dibandingkan dengan pengukuran dari alat ukur yaitu pH meter sebagai acuan.

Tabel 4. 6 Data Hasil Pengujian Sensor pH

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
1			Buffer 4.01	1.5
2			Buffer 6.86	1.044
3			Air sabun	1.111

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
4			Air sumur	1.75
5			Aquades	0.428
6			Buffer 4.01	0.487
7			Buffer 6.86	3.03

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
8			Air garam	1.764
9			Air gula	0.666
10			Air PDAM	0.779
Rata-Rata Persentase *Error*				1.255

Tabel 4. 6 merupakan data hasil pengujian sensor pH dengan berbagai jenis air. Dari hasil pengujian diperoleh nilai rata-rata *error* sebesar 1.255%. Hal ini menunjukkan bahwa ada sedikit perbedaan antara hasil pengukuran sensor pH dengan alat ukur pH meter, tetapi tingkat akurasi sensor masih cukup baik untuk pengukuran kualitas air.

Secara keseluruhan, dapat disimpulkan bahwa sensor pH berfungsi dengan baik dalam pengukuran tingkat keasaman dan kebasaan pada berbagai jenis air. Meskipun terdapat sedikit perbedaan, sensor tetap memberikan pengukuran yang memadai.

4.1.4 Hasil Pengujian Sensor DO

Sensor DO digunakan untuk mengukur kadar oksigen terlarut dalam sistem. Data yang diperoleh dari sensor DO diproses oleh ESP32, yang memungkinkan sistem memantau kadar oksigen terlarut secara *real-time*.

Sensor DO bekerja berdasarkan prinsip pengukuran perubahan resistansi atau tegangan yang dihasilkan akibat reaksi elektrokimia antara oksigen terlarut dan elektroda. Sensor ini menghasilkan sinyal yang sebanding dengan kadar oksigen terlarut, yang kemudian dikonversi menjadi data digital oleh ADC pada ESP32 dan selanjutnya diproses untuk menghasilkan nilai kadar oksigen terlarut. Sensor DO menggunakan koneksi analog untuk mengirimkan sinyal hasil pengukuran ke ESP32. Sebagai contoh, apabila nilai ADC yang terbaca sebesar 1220, maka tegangan keluaran sensor diperoleh menggunakan persamaan (3.10).

$$V = \frac{1220}{4095} \times 3300$$

$$\text{V} = 983 \text{ mV}$$

Nilai tegangan tersebut kemudian dikirim ke fungsi *readDO(ADC_Voltage, suhuDS18B20)* untuk dihitung nilai DO. Program menggunakan metode *Single Point Calibration*, sehingga terlebih dahulu menghitung tegangan saturasi sensor menggunakan persamaan (3.11).

$$V_{sat} = 983 + 35(30 - 30)$$

$$V_{sat} = 983 + 0$$

$$V_{sat} = 983$$

Selanjutnya nilai tersebut digunakan pada persamaan DO. Program kemudian mengambil nilai DO jenuh berdasarkan suhu dari tabel *DO_Table*. Misalkan tegangan sensor yang terbaca tetap 983 mV (hasil konversi ADC = 1220), maka pada suhu 30°C, tabel DO pada program adalah *DO_Table*[30] = 7560 Nilai DO kemudian dihitung menggunakan persamaan (3.12).

$$DO = \frac{983 \times 7560}{983}$$

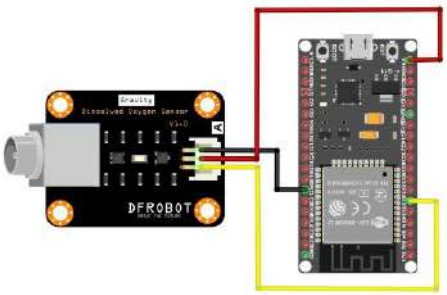
$$DO = 7560$$

Karena hasil pada fungsi *readDO()* masih dikalikan 1000, maka diperoleh nilai DO sebagai berikut.

$$DO = \frac{7560}{1000}$$

$$DO = 7.56 \text{ mg/L}$$

Nilai *Dissolved Oxygen* (DO) yang diperoleh selanjutnya diproses menggunakan metode rata-rata sehingga dihasilkan nilai pembacaan yang lebih stabil sebelum digunakan pada proses *monitoring*. Konfigurasi pin pada rangkaian sensor DO dapat dilihat pada Gambar 4. 4, dengan penjelasan terkait alokasi pin yang digunakan dijelaskan dalam Tabel 4. 7.



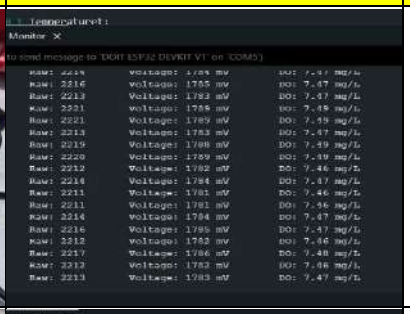
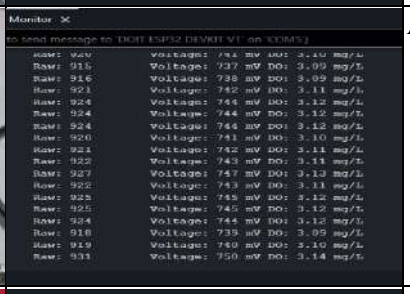
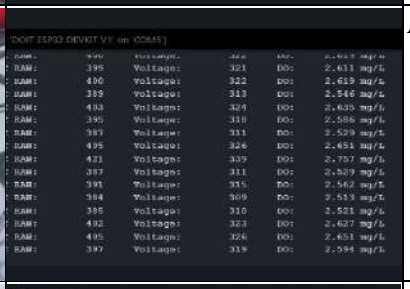
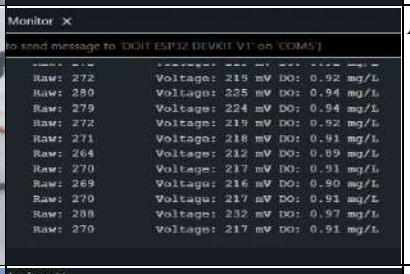
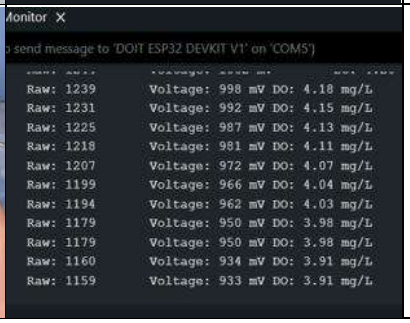
Gambar 4. 4 *Wiring* ESP32 dengan Sensor DO

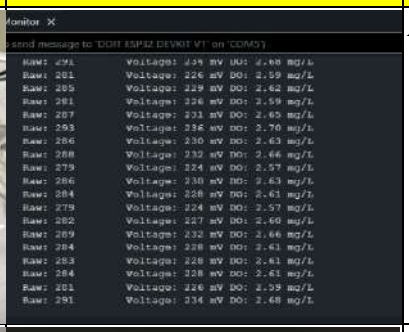
Tabel 4. 7 Konfigurasi ESP32 dengan Sensor DO

No.	ESP32	DO
1	5V	VCC
2	GND	GND
3	GPIO34 (D34)	Data

Pengujian dilakukan untuk memastikan akurasi dan keandalan sensor DO dalam mengukur kadar oksigen terlarut dalam air. Sensor diuji dengan menggunakan 10 sampel air yang berbeda yang memiliki karakteristik yang dapat memengaruhi hasil pembacaan sensor. Pengukuran dilakukan dengan menggunakan sensor DO yang telah dikalibrasi, kemudian hasil pengukuran tersebut dibandingkan dengan pengukuran dari alat ukur yaitu DO meter sebagai acuan.

Tabel 4. 8 Data Hasil Pengukuran Sensor DO

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
1			Udara	0.268
2			Air 3.23 mg/L	2.786
3			Air 2.65 mg/L	0.754
4			Air 0.86 mg/L	4.651
5			Air aerasi 3.95 mg/L	1.012

No.	Alat Ukur dan Sensor	Serial Monitor	Jenis Air	*Error* (%)
6			Air 2.74 mg/L	2.189
7			Air 3.34 mg/L	7.185
8			Udara	2.643
9			Air 2.09 mg/L	4.306
10			Air 1.70 mg/L	5.882
Rata-Rata Persentase *Error*				3.167

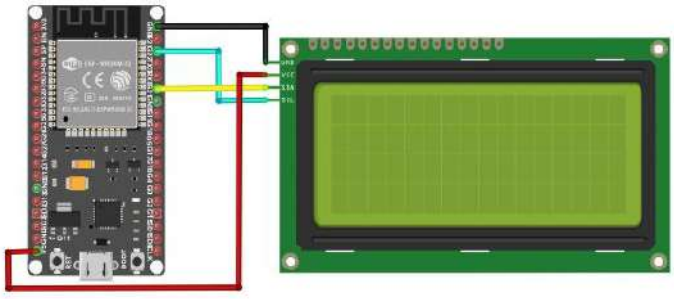
Tabel 4. 8 merupakan data hasil pengujian sensor DO (*Dissolved Oxygen*) dengan berbagai jenis air. Dari hasil pengujian diperoleh rata rata persentase *error* sebesar 3.167%. Hal ini menunjukkan bahwa terdapat perbedaan antara hasil pengukuran sensor DO dengan alat ukur DO meter, namun selisih tersebut masih dalam batas yang dapat diterima.

Dapat disimpulkan bahwa sensor DO mampu bekerja dengan cukup baik dalam mengukur kadar oksigen terlarut dalam air. Meskipun terdapat sedikit penyimpangan hasil pengukuran, sensor tetap memberikan data yang cukup akurat dan layak digunakan untuk *monitoring* kualitas air.

4.1.5 Hasil Pengujian LCD 20x4

LCD (*Liquid Crystal Display*) 20x4 digunakan sebagai monitor untuk memantau kondisi secara *real-time*. Layar ini mempermudah peneliti untuk melihat berbagai parameter seperti suhu, TDS, pH, DO pada sistem. Dengan adanya informasi yang ditampilkan secara *real-time*, peneliti dapat dengan mudah memantau dan menganalisis kondisi sistem dengan cepat dan akurat.

LCD ini terhubung dengan ESP32 menggunakan protokol komunikasi 12C, yang memungkinkan pengiriman data antar perangkat melalui jalur SDA dan SCL secara efisien. Konfigurasi pin pada rangkaian LCD dapat dilihat pada Gambar 4. 5, dengan penjelasan mengenai alokasi pin yang digunakan dijelaskan dalam Tabel 4. 9



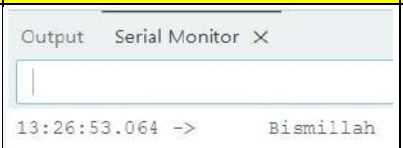
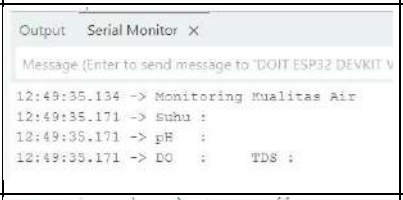
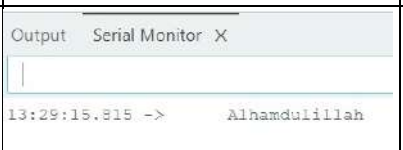
Gambar 4. 5 *Wiring* ESP32 dengan LCD 20x4

Tabel 4. 9 Konfigurasi ESP32 dengan LCD 20x4

No.	ESP32	LCD 20x4
1	5V	VCC
2	GND	GND
3	D21	SDA
4	D22	SCL

Pengujian LCD dilakukan untuk memastikan keakuratan tampilan data yang ditampilkan di layar. Pengujian ini dilakukan dengan membandingkan data yang dikirim dari ESP32 dengan data yang tampil pada LCD.

Tabel 4. 10 Data Hasil Pengujian LCD 20x4

No.	LCD 20x4	Serial Monitor	Validasi
1			Benar
2			Benar
3			Benar
4			Benar
5			Benar

Dari Tabel 4. 10 dapat dilihat bahwa hasil pengujian LCD 20x4 menunjukkan bahwa setiap karakter dan pesan yang ditampilkan pada LCD sesuai dengan data yang dikirim melalui serial monitor. Pengujian dilakukan pada beberapa kondisi pesan, termasuk teks tunggal, multi-baris, dan kombinasi teks dan

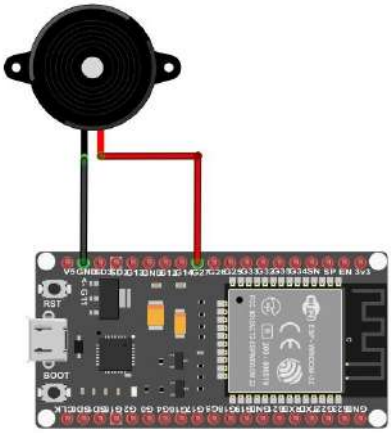
angka. Semua data yang ditampilkan pada LCD 20x4 sama dengan *output* pada serial monitor.

Secara keseluruhan, dapat disimpulkan bahwa LCD 20x4 berfungsi dengan baik dalam menampilkan informasi yang dikirim oleh ESP32. Tidak terdapat kesalahan tampilan atau karakter yang hilang, sehingga tingkat akurasi dan keandalan LCD dalam menampilkan informasi digital dapat dikatakan cukup tinggi untuk keperluan *monitoring* sistem.

4.1.6 Hasil Pengujian *Buzzer*

Buzzer digunakan sebagai indikator suara untuk memberikan notifikasi atau peringatan secara *real-time*. *Buzzer* mempermudah peneliti untuk mengetahui kondisi tertentu dalam sistem tanpa harus terus menerus melihat layar LCD. Dalam sistem *buzzer* digunakan ketika parameter kualitas air melewati ambang batas yang telah ditentukan.

Buzzer terhubung dengan ESP32 melalui pin digital, yang memungkinkan pengendalian nyala atau bunyi secara langsung melalui program mikrokontroller. Konfigurasi pin pada rangkaian *buzzer* dapat dilihat pada gambar Gambar 4. 6, dengan penjelasan mengenai alokasi pin dijelaskan pada tabel Tabel 4. 11.



Gambar 4. 6 *Wiring* ESP32 dengan *Buzzer*

Tabel 4. 11 Konfigurasi ESP32 dengan *Buzzer*

No.	ESP32	*Buzzer*
1	D13, D14, D23	Data
2	GND	Negatif

Pengujian *buzzer* dilakukan untuk memastikan respon dan keakuratan bunyi suara saat sistem bekerja. Pengujian dilakukan dengan mengaktifkan kondisi tertentu pada ESP32 dan memeriksa apakah *buzzer* berbunyi sesuai dengan perintah yang diberikan oleh ESP32.

Tabel 4. 12 Data Hasil Pengujian *Buzzer*

No.	*Buzzer*	Serial Monitor	Validasi
1			Benar
2			Benar

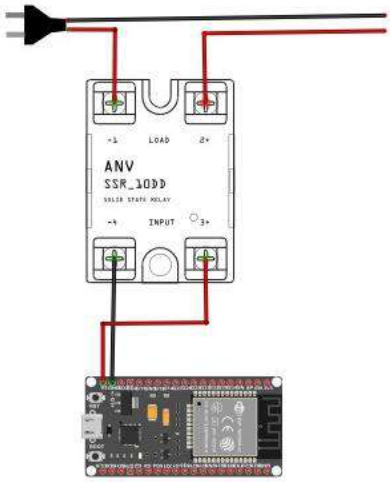
Dari Tabel 4. 12 dapat dilihat bahwa hasil pengujian *buzzer* menunjukkan bahwa perangkat memberikan respon bunyi sesuai dengan perintah yang dikirim melalui ESP32. Pengujian dilakuksan pada kondisi *buzzer ON* dan *OFF*, serta dicocokkan dengan *output* tampilan serial monitor. Semua respon *buzzer* sesuai dengan perintah, baik ketika aktif maupun nonaktif.

Secara keseluruhan, dapat disimpulkan bahwa *buzzer* berfungsi dengan baik sebagai indikator suara dalam sistem. Tidak terdapat keterlambatan atau kegagalan bunyi, sehingga tingkat keandalan *buzzer* dalam memberikan notifikasi dapat dikatakan cukup tinggi untuk mendukung *monitoring* sistem.

4.1.7 Hasil Pengujian Relay

SSR (*Solid State Relay*) digunakan sebagai saklar elektronik untuk mengendalikan perangkat daya tinggi secara otomatis. SSR mempermudah peneliti dalam mengontrol beban listrik tanpa harus mengoperasikan saklar manual, sehingga sistem dapat bekerja secara otomatis berdasarkan kondisi program. Dalam sistem, SSR digunakan untuk mengaktifkan atau menonaktifkan aerator ketika parameter kualitas air melewati ambang batas yang telah ditentukan.

SSR terhubung dengan ESP32 melalui pin digital, yang memungkinkan pengendalian *ON/OFF* secara langsung melalui program ESP32. Konfigurasi pin pada rangkaian SSR dapat dilihat pada Gambar 4. 7, dengan penjelasan alokasi pin dijelaskan pada Tabel 4. 13.



Gambar 4. 7 *Wiring* ESP32 dengan SSR

Tabel 4. 13 Konfigurasi ESP dengan SSR

No.	ESP32	Relay 1 dan 2
1	D18	Input +
2	GND	Input -

Pengujian SSR *(Solid State Relay)* dilakukan untuk memastikan respon dan keandalan saklar elektronik dalam mengendalikan perangkat beban. Pengujian dilakukan dengan memberikan perintah dari ESP32 dan memeriksa apakah SSR dapat mengaktifkan atau memutus aliran listrik ke beban sesuai dengan intruksi yang diberikan ESP32.

Tabel 4. 14 Data Hasil Pengujian SSR

No.	Solid State Relay	Serial Monitor	Validasi
1		Output Serial Monitor × Message (Enter to send message to 'DOIT ESI Solid State Relay 1 OFF Solid State Relay 1 OFF Solid State Relay 1 OFF Solid State Relay 1 OFF	Benar

No.	Solid State Relay	Serial Monitor	Validasi
2		Output Serial Monitor × Message (Enter to send message to 'DOIT E Solid State Relay 1 ON Solid State Relay 1 ON Solid State Relay 1 ON Solid State Relay 1 ON	Benar

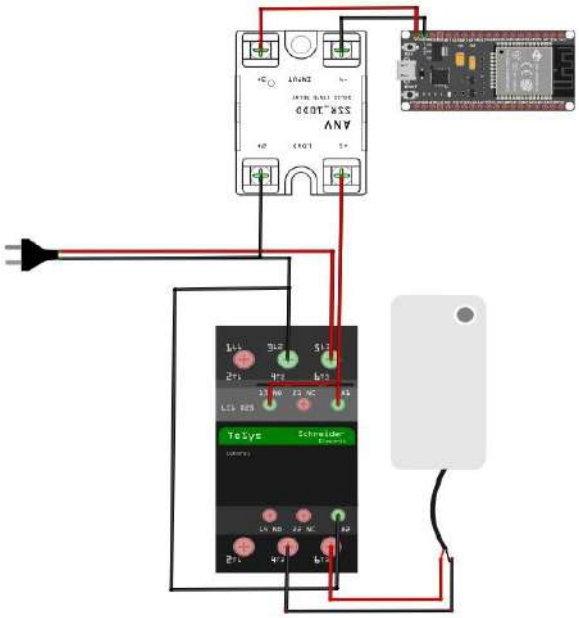
Pada Tabel 4. 14 dapat dilihat bahwa hasil pengujian SSR menunjukkan bahwa perangkat mengaktifkan dan memutus aliran listrik ke beban sesuai dengan perintah yang dikirim melalui ESP32. Pengujian dilakukan pada kondisi SSR *ON* dan *OFF*, serta dicocokkan dengan *output* yang tampil pada serial monitor. Semua respon SSR sesuai dengan perintah, baik ketika aktif atau nonaktif.

Secara keseluruhan, dapat disimpulkan bahwa SSR berfungsi dengan baik sebagai saklar elektronik dalam sistem. Tidak terdapat keterlambatan atau kegagalan pengendalian arus, sehingga tingkat keandalan SSR dalam mengatur perangkat beban dapat dikatakan cukup tinggi untuk mendukung sistem.

4.1.8 Hasil Pengujian Aerator

Aerator digunakan sebagai perangkat untuk meningkatkan kadar oksigen terlarut dalam air secara otomatis. Aerator mempermudah peneliti dalam menjaga kualitas air tanpa harus melakukan pengadukan atau aerasi manual, sehingga sistem dapat bekerja secara otomatis berdasarkan kondisi program. Dalam sistem, aerator diaktifkan atau dinonaktifkan melalui ketika parameter kualitas air melewati ambang batas yang telah ditentukan.

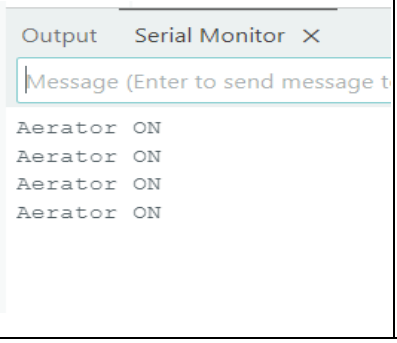
Aerator menerima pasokan daya melalui kontaktor, yang dikendalikan oleh ESP32 melalui SSR. Konfigurasi pengendalian aerator melalui SSR dapat dilihat pada Gambar 4. 8.



Gambar 4. 8 *Wiring* pada Aerator

Pengujian aerator dilakukan untuk memastikan keandalan perangkat dalam meningkatkan kadar oksigen terlarut didalam air secara otomatis. Pengujian dilakukan untuk mengaktifkan aerator melalui perintah dari ESP32 dan memeriksa apakah aerator bekerja sesuai dengan intruksi yang diberikan. Yaitu menyala ketika parameter kualitas air melewati ambang batas yang telah ditentukan dan mati ketika kondisi kembali normal.

Tabel 4. 15 Data Hasil Pengujian Aerator

No.	Aerator	Serial Monitor	Validasi
1			Benar

No.	Aerator	Serial Monitor	Validasi
2		Output Serial Monitor × Message (Enter to send messa Aerator OFF Aerator OFF Aerator OFF	Benar

Dari Tabel 4. 15 dapat dilihat bahwa hasil pengujian aerator menunjukkan bahwa perangkat bekerja sesuai dengan perintah yang dikirim ESP32. Pengujian dilakukan pada kondisi aerator *ON* dan *OFF*, serta dicocokkan dengan tampilan *output* pada serial monitor. Semua respon aerator sesuai dengan intruksi, baik ketika aktif maupun nonaktif.

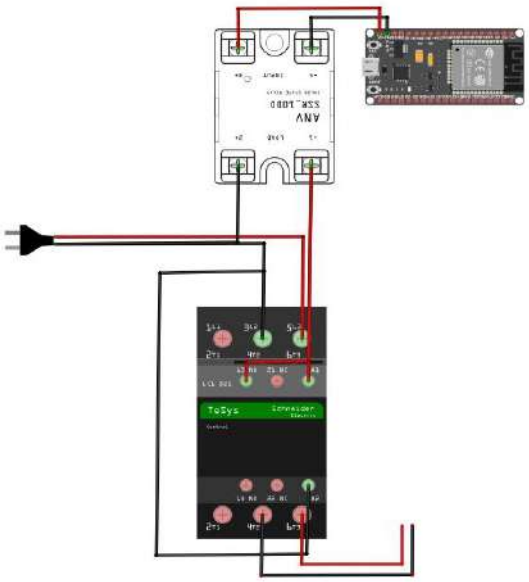
Secara keseluruhan, dapat disimpulkan bahwa aerator berfungsi dengan baik sebagai perangkat pengatur kadar oksigen terlarut dalam air. Tidak ada keterlambatan atau kegagalan kerja, sehingga tingkat keandalan aerator dalam menjalankan fungsinya dapat dikatakan cukup tinggi untuk mendukung sistem.

4.1.9 Hasil Pengujian Kontaktor

Kontaktor digunakan sebagai perangkat yang pengendali beban listrik tergantung mode manual atau otomatis yang digunakan. Pada mode manual, pengoperasian kontaktor dilakukan menggunakan *pushbutton* sehingga pengguna dapat secara langsung menghubungkan atau memutus aliran listrik ke beban sesuai kebutuhan.

Pada mode otomatis, kontaktor dikendalikan melalui relay yang terhubung dengan ESP32. Relay berfungsi sebagai perantara untuk mengaktifkan atau menonaktifkan *coil* kontaktor, sehingga beban listrik dapat dikendalikan secara langsung tanpa intervensi manual. Dengan demikian, sistem mampu bekerja secara mandiri sesuai dengan kondisi yang telah ditentukan.

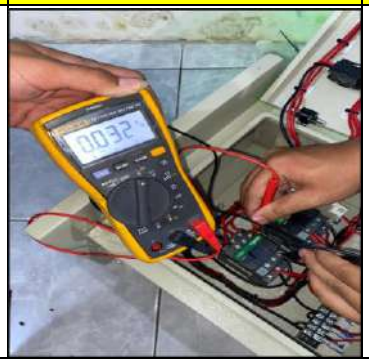
Saat coil kontaktor aktif, kontak utama menutup sehingga arus mengalir ke beban dan saat tidak aktif, kontak terbuka. Konfigurasi ditunjukkan pada Gambar 4. 9.



Gambar 4. 9 *Wiring* Kontaktor

Pengujian kontaktor dilakukan untuk memastikan keandalan perangkat dalam mengendalikan beban secara otomatis. Pengujian dilakukan dengan mengaktifkan kontaktor melalui perintah dari ESP32 dan memeriksa apakah kontaktor bekerja sesuai instruksi yang diberikan, yaitu menutup (*ON*) ketika sinyal kontrol aktif dan membuka (*OFF*) ketika sinyal dihentikan.

Tabel 4. 16 Data Hasil Pengujian Kontaktor

No.	Kontaktor	Kondisi	Voltase
1		Kontaktor Nonaktif	0.032

No.	Kontaktor	Kondisi	Voltase
2		Kontaktor Aktif	226.6

Pada Tabel 4. 16 dapat dilihat bahwa hasil pengujian kontaktor menunjukkan bahwa perangkat bekerja sesuai dengan kondisi yang diharapkan. Pengujian dilakukan pada kondisi kontaktor aktif dan nonaktif dengan melihat nilai tegangan yang dihasilkan pada beban. Pada saat kontaktor nonaktif, tegangan yang terukur sangat kecil yaitu sebesar 0,032 V, sedangkan pada saat kontaktor aktif, tegangan yang terukur mencapai 226,6 V. Hal ini menunjukkan bahwa kontaktor mampu memutus dan menghubungkan aliran listrik dengan baik.

Dapat disimpulkan bahwa kontaktor berfungsi dengan baik sebagai perangkat pengendali beban listrik. Tidak ditemukan adanya gangguan atau kegagalan dalam proses pengujian, sehingga tingkat keandalan kontaktor dalam menjalankan fungsinya dapat dikatakan cukup tinggi untuk mendukung sistem.

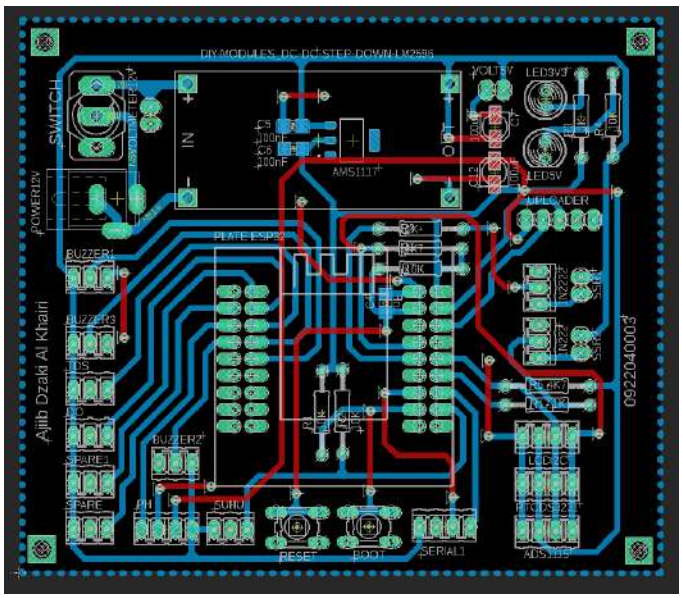
4.2 Hasil Perancangan *Hardware* dan *Software*

Pada subbab ini membahas hasil perancangan sistem secara keseluruhan yang meliputi aspek *hardware* dan *software* yang digunakan pada sistem. Perancangan dilakukan untuk memastikan bahwa seluruh komponen dapat bekerja sesuai dengan fungsi yang telah direncanakan. Perancangan *Hardware* meliputi aspek elektrik dan mekanik, sementara pada sisi *software*, dilakukan perancangan *website* berbasis Next.js untuk *monitoring dashboard* sistem.

4.2.1 Hasil Perancangan Elektrik

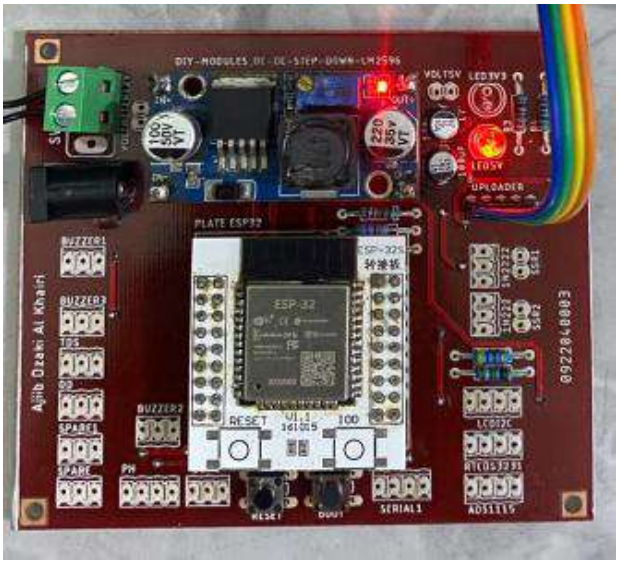
Pada tahap perancangan elektrik, dilakukan proses perancangan dan pengembangan rangkaian elektronika yang berfungsi untuk menunjang kinerja sistem *monitoring* dan kontrol sistem secara keseluruhan. Rangkaian tersebut

meliputi pembuatan desain PCB sebagai pusat integrasi berbagai komponen, serta perancangan panel kontrol yang berfungsi menempatkan komponen elektronik agar tersusun rapi. Tujuan dari perancangan adalah agar komponen dapat beroperasi secara optimal, mudah dalam proses pemasangan dan perawatan, serta terlindungi dari potensi gangguan eksternal.



Gambar 4. 10 Hasil Perancangan *Board Layout PCB*

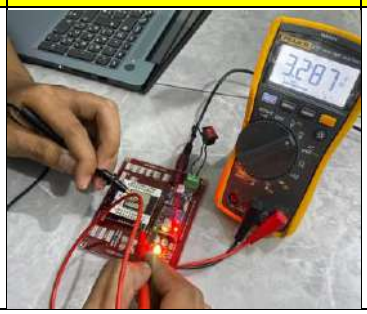
Gambar 4. 10 merupakan hasil perancangan layout PCB yang dibuat menggunakan aplikasi Eagle. Pada desain tersebut menunjukkan tata letak jalur penghubung antar komponen, meliputi penempatan mikrokontroler, sensor, terminal konektor, serta berbagai komponen pendukung lainnya. PCB ini dirancang menggunakan dua layer dengan tujuan mengoptimalkan pemanfaatan ruang serta mengurangi kemungkinan gangguan sinyal antar jalur. Penggunaan dua *layer* juga memungkinkan jalur catu daya (*power*) dan jalur sinyal dipisahkan dengan lebih baik sehingga tata letak rangkaian menjadi lebih rapi, memudahkan proses manufaktur, serta meningkatkan stabilitas dan keandalan sistem secara keseluruhan.

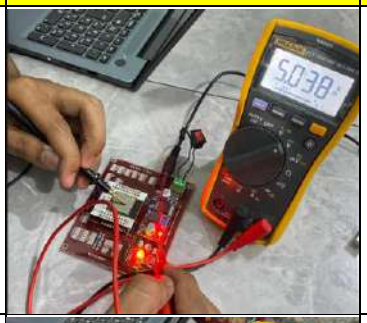


Gambar 4. 11 Hasil Implementasi Rangkaian PCB

Gambar 4. 11 menunjukkan realisasi fisik dari PCB yang telah dirakit. Berbagai komponen seperti sensor, mikrokontroller ESP32, konektor terminal, serta *buck converter* telah terpasang mengikuti desain layout yang telah dibuat sebelumnya. Penyusunan komponen dilakukan secara teratur, kemudian dilakukan pengecekan koneksi untuk memastikan tidak terdapat kesalahan penyolderan maupun potensi hubungan pendek. Setelah itu, PCB diuji untuk memastikan seluruh komponen dapat berfungsi dengan baik sebelum diintegrasikan ke dalam sistem secara keseluruhan.

Tabel 4. 17 Data Hasil Pengujian *Hardware Minimum System*

No.	ESP32	Power Input	Validasi
1		3.3V	Benar

No.	ESP32	Power Input	Validasi
2		5V	Benar
3		12V	Benar
4		Upload Program	Benar

Pada Tabel 4. 17 dapat dilihat bahwa hasil pengujian *hardware minimum system* berbasis ESP32 menunjukkan bahwa perangkat dapat beroperasi dengan baik pada berbagai variasi tegangan *input*. Tegangan 3.3V digunakan untuk mensuplai inti mikrokontroler, memberikan daya pada pin GPIO dan sensor atau komponen yang bekerja pada level logika 3.3V. Tegangan 5V menjadi sumber input untuk regulator 3.3V, sehingga tegangan tersebut dapat diturunkan menjadi tegangan kerja yang sesuai untuk ESP32.

Selain itu, tegangan 12V digunakan sebagai sumber daya utama sistem yang kemudian didistribusikan ke berbagai bagian rangkaian sesuai kebutuhan. Tegangan ini umumnya digunakan untuk mensuplai beban atau perangkat eksternal yang memerlukan daya lebih besar, serta sebagai input awal yang kemudian diturunkan menjadi 5V dan 3,3V melalui rangkaian regulator.

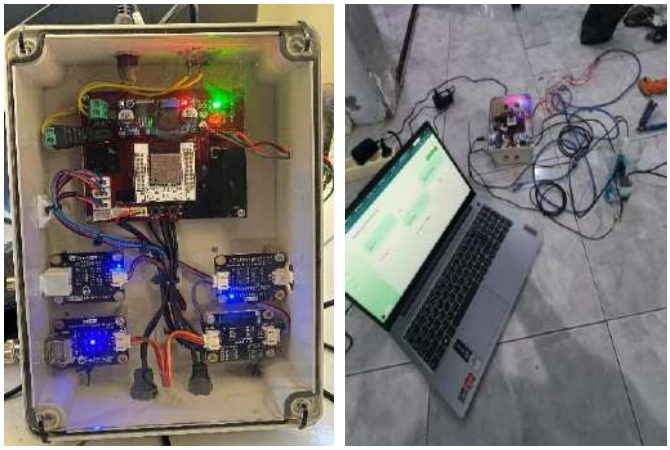
Setelah pengujian tegangan selesai dilakukan, tahap berikutnya adalah pengujian proses *upload* program ke *minimum system* ESP32. Pengujian ini bertujuan untuk memastikan bahwa sistem dapat terhubung dengan perangkat serta mampu menerima dan menjalankan program dengan baik tanpa mengalami kendala selama proses pemrograman.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem catu daya pada *minimum system* ESP32 bekerja dengan baik dan stabil. Setiap variasi tegangan dapat dimanfaatkan sesuai dengan fungsinya masing-masing tanpa mengalami gangguan, sehingga sistem mampu mendukung kinerja perangkat secara optimal.

4.2.2 Hasil Perancangan Mekanik

Pada tahap perancangan mekanik, dilakukan proses perancangan dan pembuatan struktur fisik sistem yang berfungsi untuk menunjang kinerja sistem *monitoring* dan kontrol secara keseluruhan. Perancangan ini meliputi pembuatan rangka atau *casing* sebagai tempat pemasangan seluruh komponen, serta penataan posisi perangkat seperti sensor, aktuator, dan panel kontrol agar tersusun secara rapi dan terorganisir.

Tujuan dari perancangan mekanik adalah untuk memastikan setiap komponen dapat terpasang dengan kuat dan aman, memudahkan dalam proses instalasi serta perawatan, serta memberikan perlindungan terhadap komponen dari pengaruh lingkungan seperti debu, air, dan benturan.



Gambar 4. 12 Modul dan Probe Sensor

Pada Gambar 4. 12 menunjukkan modul dan probe sensor yang digunakan dalam sistem. Pada bagian luar terlihat beberapa probe sensor yang terdiri dari sensor suhu, sensor TDS, sensor pH, dan sensor DO yang dirancang untuk ditempatkan langsung pada media pengukuran. Probe-probe tersebut disusun dalam satu wadah pelindung berbentuk tabung untuk memudahkan proses instalasi serta menjaga kestabilan posisi sensor saat digunakan.

Pada bagian dalam tabung terdapat modul sensor yang berfungsi sebagai pengolah sinyal dari masing-masing probe. Modul sensor TDS, DO, dan pH terhubung langsung dengan probe terkait untuk mengubah data hasil pengukuran menjadi sinyal yang dapat dibaca oleh sistem utama. Penempatan modul di dalam casing bertujuan untuk melindungi rangkaian dari gangguan eksternal seperti air, debu, dan kerusakan fisik.

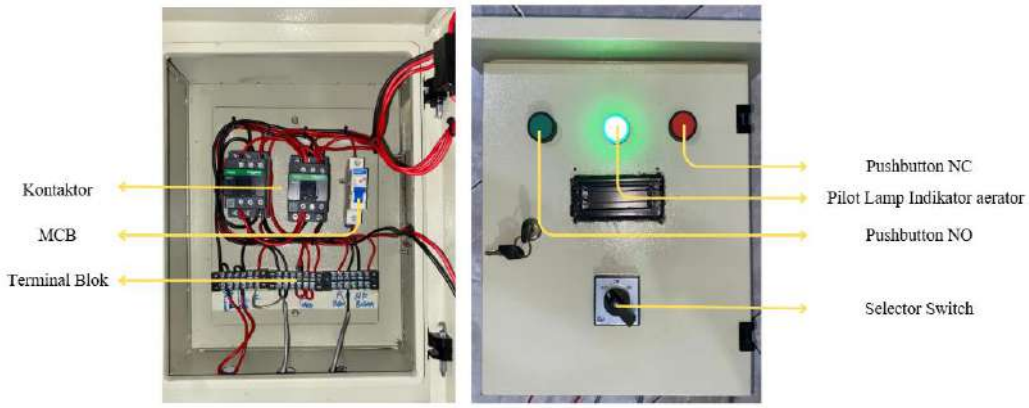
Gambar 4. 13 *Wiring* Panel *Monitoring* Kualitas Air

Pada Gambar 4. 13 menunjukkan *wiring* panel yang menjadi pusat pengendali dan distribusi daya sistem. Pada bagian dalam panel terdapat beberapa komponen utama seperti *power supply* 12V yang berfungsi sebagai sumber tegangan, mikrokontroler ESP32 sebagai pengolah data, serta MCB yang digunakan sebagai pengaman rangkaian dari arus lebih atau hubung singkat.

Selain itu, terdapat juga terminal blok yang digunakan untuk merapikan dan mempermudah koneksi antar kabel, serta stopkontak sebagai sumber daya tambahan untuk perangkat lain. Kipas DC dipasang di dalam panel untuk menjaga suhu komponen tetap stabil agar kinerja sistem tetap optimal. Komponen lain

seperti *buzzer* berfungsi sebagai indikator suara ketika terjadi kondisi tertentu pada sistem.

Pada bagian depan panel, terdapat *pilot lamp* indikator 220V yang berfungsi sebagai penanda bahwa sistem dalam kondisi aktif, serta LCD 20x4 yang digunakan untuk menampilkan informasi parameter kualitas air secara *real-time*. Seluruh komponen disusun secara rapi di dalam panel untuk memudahkan proses instalasi, pemeliharaan, serta meningkatkan keamanan sistem dari gangguan eksternal.



Gambar 4. 14 *Wiring* Panel Kontrol Aerator

Pada Gambar 4. 14 menunjukkan *wiring* panel kontrol aerator yang berfungsi sebagai sistem pengendali utama dalam mengoperasikan beban aerator. Pada bagian dalam panel terdapat beberapa komponen utama seperti kontaktor yang berfungsi sebagai saklar elektromagnetik untuk menghubungkan dan memutus aliran listrik ke aerator, MCB sebagai pengaman rangkaian dari gangguan arus lebih dan hubung singkat, serta terminal blok yang digunakan untuk mempermudah distribusi dan pengkabelan antar komponen.

Pada bagian depan panel terdapat beberapa komponen antarmuka pengguna, yaitu *push button* NO (*Normally Open*) dan *push button* NC (*Normally Closed*) yang digunakan untuk mengontrol operasi aerator secara manual. Selain itu, terdapat *selector switch* yang berfungsi untuk memilih mode operasi, seperti mode manual atau otomatis. *Pilot lamp* indikator aerator digunakan sebagai penanda visual yang menunjukkan kondisi sistem, apakah aerator dalam keadaan aktif atau tidak.

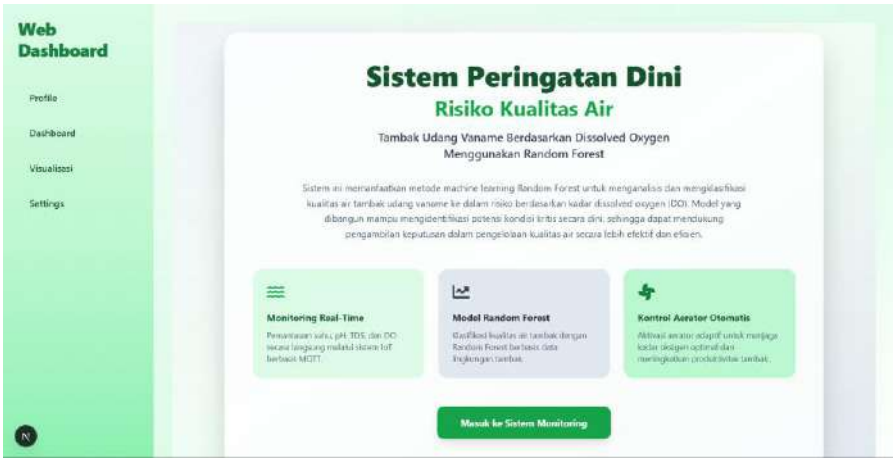
Gambar 4. 15 Prototipe *Plant* Tambak

Berdasarkan Gambar 4. 15, prototipe *plant* tambak yang digunakan merupakan wadah penampung air yang berfungsi sebagai media simulasi sistem pemantauan dan pengendalian kualitas air. Prototipe ini digunakan untuk merepresentasikan kondisi tambak secara nyata dalam skala kecil sehingga memungkinkan proses pengujian sensor, aktuator, serta sistem kontrol berbasis ESP32 dapat dilakukan secara terintegrasi.

Pada prototipe ini, seluruh komponen seperti sensor, aktuator, dan sistem kontrol ditempatkan untuk mendukung proses pengambilan data serta pengujian kinerja sistem. Dengan adanya prototipe plant tambak, proses pengujian dapat dilakukan secara terkontrol sebelum sistem diimplementasikan pada kondisi tambak sebenarnya.

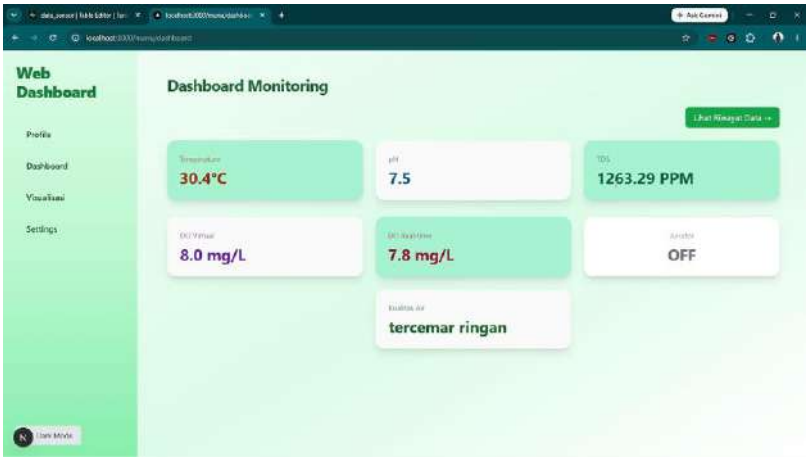
4.2.3 Hasil Perancangan *Software*

Pada tahap perancangan *software* dilakukan pengembangan sistem *dashboard website* yang berfungsi untuk mengolah data, menampilkan informasi, serta mengendalikan sistem secara keseluruhan. *Website* dirancang untuk memantau kualitas air secara *real-time*. Antarmuka ini menampilkan data penting yang diperoleh dari sensor-sensor yang terpasang, seperti sensor suhu, sensor TDS, sensor pH, dan sensor DO yang digunakan untuk mengukur parameter kualitas air.



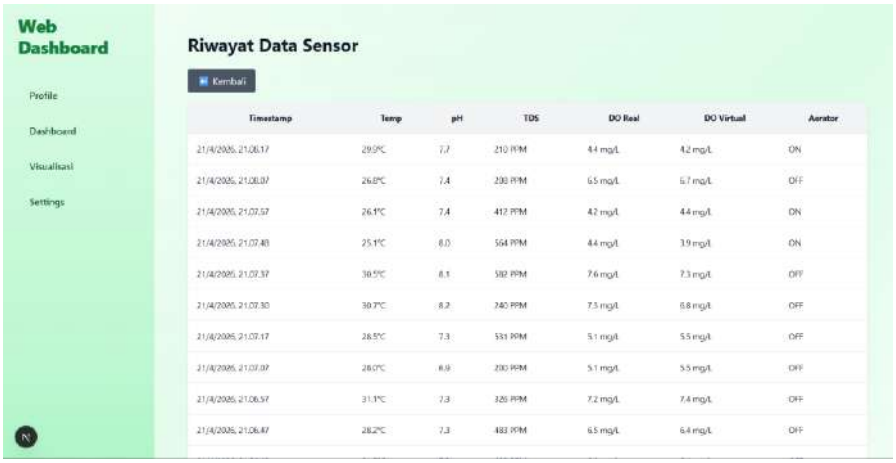
Gambar 4. 16 Antarmuka *Profile*

Pada Gambar 4. 16 menunjukkan tampilan antarmuka halaman *profile* pada *web dashboard* sistem *monitoring* kualitas air yang berfungsi sebagai halaman utama dalam memberikan informasi umum terkait sistem. Pada halaman ini ditampilkan deskripsi mengenai tujuan sistem sebagai sistem peringatan dini risiko kualitas air, serta informasi singkat terkait parameter yang digunakan. Selain itu, terdapat beberapa fitur utama yang disajikan dalam bentuk menu, seperti *monitoring real-time*, analisis data, dan kontrol otomatis, serta tombol navigasi untuk mengakses halaman *monitoring*. Secara keseluruhan, antarmuka ini dirancang agar informatif, mudah dipahami, dan memudahkan pengguna dalam mengakses fitur-fitur utama sistem.



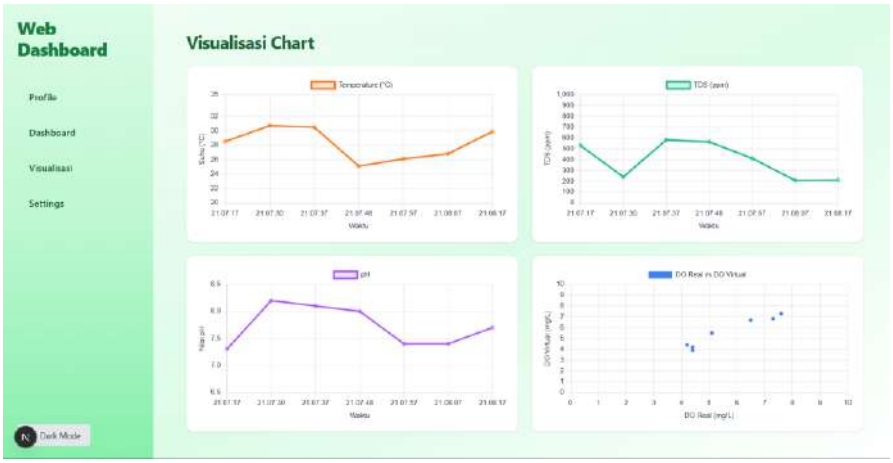
Gambar 4. 17 *Dashboard Monitoring*

Pada Gambar 4. 17 menunjukkan tampilan *dashboard monitoring* pada *web* yang digunakan untuk menampilkan kondisi kualitas air secara *real-time*. Pada halaman ini ditampilkan beberapa parameter utama hasil pembacaan sensor, seperti suhu, pH, TDS, dan *Dissolved Oxygen* (DO), baik nilai aktual maupun nilai rata-rata. Selain itu, terdapat indikator status aerator yang menunjukkan kondisi aktif atau nonaktif, serta informasi kualitas air secara keseluruhan dalam bentuk kategori. Data ditampilkan dalam bentuk kartu yang terstruktur sehingga mudah dibaca dan dipahami oleh pengguna. *Dashboard* ini juga dilengkapi dengan fitur akses ke riwayat data untuk melihat hasil pengukuran sebelumnya, sehingga memudahkan pengguna dalam melakukan pemantauan dan analisis kondisi air secara menyeluruh.



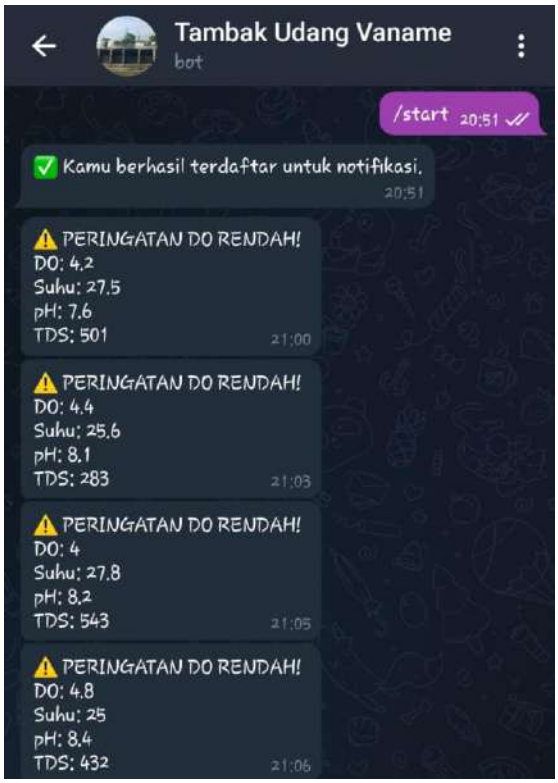
Gambar 4. 18 Tampilan Riwayat Data Sensor

Pada Gambar 4. 18 menunjukkan tampilan halaman riwayat data sensor pada *web dashboard* yang digunakan untuk menyimpan dan menampilkan data hasil pengukuran secara berkala. Pada halaman ini ditampilkan data dalam bentuk tabel yang memuat informasi seperti waktu pengambilan data, nilai suhu, pH, TDS, DO, serta status aerator. Data riwayat ini memungkinkan pengguna untuk melihat perubahan kondisi kualitas air dari waktu ke waktu secara lebih detail. Dengan adanya fitur ini, pengguna dapat melakukan analisis terhadap tren data, mengevaluasi kinerja sistem, serta membantu dalam pengambilan keputusan berdasarkan data yang telah tersimpan.



Gambar 4. 19 Tampilan Visualisasi *Chart*

Pada Gambar 4. 19 menunjukkan tampilan visualisasi *chart* pada web *dashboard* yang digunakan untuk menampilkan data sensor dalam bentuk grafik. Grafik ini memudahkan pengguna dalam melihat tren perubahan parameter seperti suhu, pH, TDS, dan DO secara visual dari waktu ke waktu, sehingga analisis kondisi kualitas air dapat dilakukan dengan lebih cepat dan efektif.



Gambar 4. 20 Pengujian Notifikasi pada Bot Telegram

Pada Gambar 4. 20 menunjukkan bahwa notifikasi pada bot Telegram yang digunakan sebagai sistem peringatan dini dapat memberikan peringatan secara otomatis dikirim ketika nilai parameter, khususnya DO berada di bawah ambang batas yang telah ditentukan. Notifikasi juga menampilkan informasi parameter lainnya yang memungkinkan pengguna mengetahui kondisi aktual air secara langsung tanpa harus langsung ke lokasi tambak atau melihat *dashboard*.

4.3 Pengujian Metode Random Forest

Pada tahap ini dilakukan pengujian metode Random Forest yang digunakan untuk menganalisis dan memprediksi risiko kualitas air tambak udang vaname. Pengujian bertujuan untuk mengetahui tingkat akurasi dan kinerja model dalam mengklasifikasikan tingkat risiko kualitas air, seperti tercemar ringan, sedang, dan berat, dengan menjadikan parameter *Dissolved Oxygen* (DO) sebagai indikator utama dan parameter lainnya digunakan sebagai variabel prediktor (fitur) untuk meningkatkan akurasi prediksi. Data yang digunakan diperoleh dari hasil pengukuran langsung di tambak, kemudian disimpan dalam bentuk *dataset* menggunakan Microsoft Excel.

4.3.1 Pengambilan *Dataset* Pengujian

Pengambilan data dilakukan secara langsung pada tambak untuk memperoleh data kualitas air yang merepresentasikan kondisi lingkungan budidaya secara aktual. Proses ini bertujuan untuk memastikan sistem *monitoring* mampu melakukan pembacaan parameter kualitas air secara stabil, akurat, dan berkesinambungan pada kondisi operasional sebenarnya. Parameter yang diukur meliputi suhu, *Total Dissolved Solids* (TDS), pH, dan *Dissolved Oxygen* (DO).

Selama proses pengambilan data, sensor dipasang dan dioperasikan secara langsung pada media air tambak sehingga setiap parameter dapat diukur secara *real-time*. Data hasil pembacaan sensor kemudian dikirimkan ke sistem *monitoring* untuk disimpan sebagai *dataset* penelitian. Pengambilan data secara langsung pada tambak dilakukan agar data yang diperoleh mampu merepresentasikan kondisi kualitas air sebenarnya sehingga dapat digunakan sebagai dasar dalam proses pelabelan data serta pembentukan model klasifikasi Random Forest.

Gambar 4. 21 Pengambilan Data Kualitas Air Tambak

Berdasarkan proses pengambilan data di lapangan, perangkat *monitoring* ditempatkan pada lokasi tambak sehingga sensor dapat melakukan pembacaan parameter kualitas air secara langsung. Sensor mengukur nilai pH, suhu, *Total Dissolved Solids* (TDS), dan *Dissolved Oxygen* (DO) secara *real-time*. Hasil pembacaan kemudian dikirimkan ke sistem *monitoring* dan disimpan sebagai *dataset* penelitian untuk proses analisis lebih lanjut.

Tabel 4. 18 Hasil Pengambilan Data Tambak Udang Vaname

Created_at	temp	ph	tds	do virtual
2026-06-08 16:59	28.1875	7.77118	1229.37	6.517805
2026-06-08 17:04	28	7.6785	1280.67	7.43013
2026-06-08 17:09	26.875	7.71866	1237.84	6.576476
2026-06-08 17:14	26.125	7.70136	1265.88	7.722643
2026-06-08 17:19	27	7.67912	1280.55	7.559275
2026-06-08 17:24	27.375	7.70198	1227.46	7.364606
2026-06-08 17:29	27.625	7.6612	1271.06	7.275043
2026-06-08 17:35	27.625	7.72051	1255.45	7.147963
2026-06-08 17:40	27.5625	7.75079	1241.36	7.077303
2026-06-08 17:45	28.0625	7.71804	1244.06	7.178837
2026-06-08 17:50	28.1875	7.71557	1274.17	7.014654
2026-06-08 17:55	28.1875	7.71619	1265.65	6.924958
2026-06-08 18:00	28.25	7.70631	1295.82	6.881889
2026-06-08 18:05	28.1875	7.91081	1237.3	6.790034
2026-06-08 18:10	28.125	7.50057	1230.77	6.770603

Created_at	temp	ph	tds	do virtual
2026-06-08 18:15	28.1875	7.72051	1244.83	6.780113
2026-06-08 18:20	28.1875	7.48636	1289.71	6.594594
2026-06-08 18:25	28.125	7.88547	1270.47	6.454306
2026-06-08 18:31	28.0625	7.86509	1265.3	6.321543
2026-06-08 18:36	28.125	7.44311	1225.73	6.210237

Berdasarkan Tabel 4. 18, diperoleh *dataset* hasil pengambilan data kualitas air secara langsung pada tambak udang vaname. *Dataset* yang diperoleh terdiri atas lima atribut, yaitu timestamp, temperature, pH, TDS, dan DO virtual. Kolom *timestamp* menunjukkan waktu pengambilan data, sedangkan kolom temp, pH, TDS, dan DO virtual merupakan hasil pembacaan sensor pada setiap interval pengukuran. Data tersebut selanjutnya digunakan sebagai *dataset* awal dalam proses pengolahan data sebelum dilakukan pelabelan dan pembentukan model klasifikasi Random Forest.

Berdasarkan hasil pengukuran, nilai suhu air tambak berada pada kisaran 26–28°C, sedangkan nilai pH berada pada kisaran 7,4–7,9. Nilai DO virtual menunjukkan variasi antara sekitar 6,0–7,7 mg/L, yang menggambarkan perubahan kondisi oksigen terlarut selama proses pengambilan data. Variasi nilai pada setiap parameter menunjukkan bahwa kondisi kualitas air tambak berubah secara dinamis sesuai dengan kondisi lingkungan dan waktu pengukuran.

Selama proses pengambilan data, diperoleh bahwa nilai TDS cenderung berada pada lebih dari 1000 ppm, bahkan beberapa data menunjukkan nilai melebihi 1200 ppm. Kondisi tersebut menunjukkan bahwa air tambak memiliki kandungan padatan terlarut yang relatif tinggi. Sensor TDS yang digunakan memiliki keterbatasan rentang pengukuran sehingga pembacaan nilai TDS berada pada kisaran maksimum kemampuan sensor. Oleh karena itu data TDS tidak digunakan sebagai salah satu atribut (*feature*) dalam pembentukan model Random Forest karena tidak dapat merepresentasikan karakteristik kualitas air tambak berdasarkan kemampuan pengukuran sensor yang digunakan.

Dataset hasil pengukuran tersebut selanjutnya digunakan sebagai dasar dalam proses pengolahan data, yang meliputi penanganan *outlier*, pelabelan berdasarkan nilai DO virtual, pembagian data latih dan data uji, serta proses pelatihan model Random Forest.

4.3.2 Analisis Parameter TDS

Selama proses pengambilan data, diperoleh bahwa nilai TDS pada air tambak cenderung berada di atas 1000 ppm, bahkan beberapa pengukuran mencapai batas maksimum pembacaan sensor. Kondisi tersebut menunjukkan bahwa sensor TDS yang digunakan telah mencapai batas rentang pengukurannya (saturasi), sehingga variasi nilai yang dihasilkan tidak lagi mampu merepresentasikan perubahan kondisi kualitas air secara akurat.

Kondisi tersebut menyebabkan sebagian besar data TDS memiliki nilai yang relatif konstan meskipun terjadi perubahan kondisi air tambak. Akibatnya, parameter TDS tidak mampu memberikan informasi yang cukup untuk membedakan karakteristik kualitas air pada setiap kondisi pengamatan. Berdasarkan hasil tersebut, parameter TDS dinilai kurang sesuai untuk digunakan sebagai fitur (*feature*) pada proses pembentukan model Random Forest.

Meskipun demikian, sensor TDS tetap digunakan sebagai bagian dari sistem *monitoring* kualitas air. Nilai hasil pembacaan sensor tetap ditampilkan pada LCD, *dashboard* web, dan notifikasi Telegram sebagai informasi pendukung kondisi kualitas air tambak. Dengan demikian, parameter TDS tetap berfungsi sebagai parameter *monitoring*, namun tidak digunakan sebagai variabel masukan dalam proses klasifikasi menggunakan algoritma Random Forest.

4.3.3 Preprocessing Data

Sebelum dilakukan proses *preprocessing*, data hasil pengukuran kualitas air terlebih dahulu dikumpulkan melalui proses pengambilan data secara langsung pada tambak udang vaname serta menggunakan sampel air tambak. Data yang diperoleh merupakan hasil pembacaan sensor terhadap parameter suhu *(temperature*), pH, *Total Dissolved Solids* (TDS), dan *Dissolved Oxygen* (DO) virtual yang direkam secara berkala selama periode pengamatan. Informasi mengenai waktu pengambilan data, jumlah data, serta periode pengamatan ditunjukkan pada Tabel 4. 19.

Tabel 4. 19 Informasi Pengambilan Data Penelitian

No	Tanggal	Tempat Pengambilan	Interval	Jumlah Data
1	08-06-2026	Tambak	5 menit	64
2	09-06-2026	Tambak	5 menit	46
3	10-06-2026	Tambak	5 menit	54
4	11-06-2026	Prototype	5 menit	61
5	15-06-2026	Prototype	1 menit	7
6	16-06-2026	Prototype	2 menit	364
7	17-06-2026	Prototype	2 menit	360
8	03-07-2026	Tambak Tradisional	1 menit	182
9	04-07-2026	Tambak	1 menit	999
Total		9 Hari		2.137

Proses pengambilan data dilakukan selama sembilan hari dengan dua lokasi pengambilan, yaitu secara langsung pada tambak dan menggunakan sampel air tambak. Interval pengambilan data bervariasi antara 1 menit, 2 menit, dan 5 menit sesuai dengan kebutuhan pengujian. Dari keseluruhan proses pengambilan data diperoleh sebanyak 2.137 data yang selanjutnya digunakan sebagai *dataset* pada tahap *preprocessing* sebelum dilakukan pelatihan model Random Forest.

Tahap *preprocessing* diawali dengan penanganan *outlier* menggunakan metode *Interquartile Range* (IQR). Berdasarkan hasil perhitungan pada parameter pH, diperoleh nilai kuartil pertama (Q1) sebesar 7,668815 dan kuartil ketiga (Q3) sebesar 7,764990. Sebagai contoh, nilai *Interquartile Range* (IQR) dihitung menggunakan persamaan (3.13).

$$IQR = 7{,}76499 - 7{,}668815$$

$$IQR = 0{,}096175$$

Selanjutnya dihitung batas bawah (*lower bound*) menggunakan persamaan (3.14).

$$Lower = 7{,}668815 - (1{,}5 \times 0{,}096175)$$

$$Lower = 7{,}524553$$

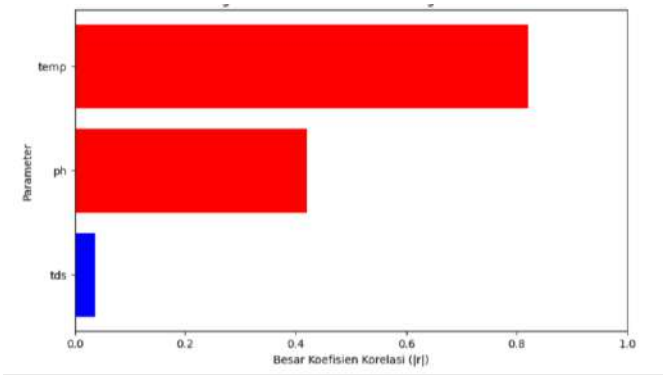
Kemudian batas atas (*upper bound*) dihitung menggunakan persamaan (3.15).

$$Upper = 7{,}764990 + (1{,}5 \times 0{,}096175)$$

$$Upper = 7{,}909253$$

Berdasarkan hasil perhitungan tersebut, data dengan nilai pH yang berada pada rentang 7,524553 ≤ pH ≤ 7,909253 dipertahankan sebagai data normal. Sebaliknya, data yang memiliki nilai di luar rentang tersebut dikategorikan sebagai *outlier* dan dihapus dari *dataset* sebelum digunakan pada proses pelatihan model Random Forest.

Proses yang sama diterapkan pada parameter suhu (*temp*) dan DO virtual. Setelah seluruh proses penanganan *outlier* selesai dilakukan, jumlah data berkurang dari 2.137 data menjadi 1.939 data. *Dataset* hasil *preprocessing* tersebut selanjutnya digunakan pada tahap pelabelan data berdasarkan nilai DO virtual.



Gambar 4. 22 Hubungan Parameter Kualitas Air dengan DO Virtual

Berdasarkan Gambar 4. 22, hasil analisis korelasi menunjukkan bahwa setiap parameter memiliki tingkat hubungan yang berbeda terhadap nilai DO Virtual. Parameter suhu memiliki nilai koefisien korelasi terbesar, yaitu sekitar 0,82, sehingga menunjukkan hubungan yang kuat dengan DO Virtual. Hal ini mengindikasikan bahwa perubahan nilai suhu memberikan pengaruh yang lebih besar terhadap perubahan nilai DO Virtual dibandingkan parameter lainnya.

Sementara itu, parameter pH memiliki nilai koefisien korelasi sekitar 0,43, yang menunjukkan hubungan pada kategori sedang. Nilai tersebut mengindikasikan bahwa pH memiliki keterkaitan terhadap DO Virtual, meskipun pengaruhnya tidak sebesar suhu.

Di sisi lain, parameter TDS memiliki nilai koefisien korelasi yang sangat rendah, yaitu sekitar 0,03. Hal ini menunjukkan bahwa hubungan antara TDS dan DO Virtual sangat lemah, sehingga parameter tersebut memberikan kontribusi yang relatif kecil terhadap perubahan nilai DO Virtual pada *dataset* yang digunakan.

4.3.4 Pembentukan Kelas Risiko

Proses pembentukan kelas risiko dilakukan melalui pelabelan (*labeling*) berdasarkan nilai *Dissolved Oxygen* (DO) virtual. Pelabelan bertujuan untuk menghasilkan kelas kualitas air yang digunakan sebagai target (label) pada proses pelatihan algoritma Random Forest.

Pelabelan dilakukan dengan mengelompokkan setiap data ke dalam tiga kategori kualitas air berdasarkan rentang nilai DO virtual. Data dengan nilai DO ≥ 5 mg/L diberi label tercemar ringan, data dengan nilai 4 ≤ DO < 5 mg/L diberi label tercemar sedang, sedangkan data dengan nilai DO < 4 mg/L diberi label tercemar berat. Proses pelabelan dilakukan secara otomatis terhadap seluruh data menggunakan fungsi yang telah diimplementasikan pada program.

Setelah proses pelabelan selesai dilakukan, setiap data memiliki satu kelas kualitas air yang digunakan sebagai target (label) pada proses klasifikasi. Sementara itu, parameter suhu (*temperature*) dan pH digunakan sebagai fitur (*feature*) dalam pembentukan model Random Forest. Dengan demikian, model mempelajari hubungan antara kedua parameter tersebut terhadap kelas kualitas air yang dibentuk berdasarkan nilai DO virtual.

Tabel 4. 20 Hasil Pembentukan Label Kualitas Air

Created_at	temp	ph	do virtual	label
2026-06-09 22:24	27.625	7.509	6.83125	tercemar ringan
2026-06-09 22:34	27.625	7.54607	6.804339	tercemar ringan
2026-06-09 22:39	27.625	7.50159	6.76197	tercemar ringan
2026-06-09 22:44	27.625	7.70917	6.650741	tercemar ringan
2026-06-09 22:49	27.625	7.64616	6.563952	tercemar ringan
2026-06-09 23:51	27.625	7.97217	6.86387	tercemar ringan
2026-06-10 00:14	27.4375	7.69075	6.535396	tercemar ringan
2026-06-10 01:25	27.3125	7.68745	4.32285	tercemar sedang
2026-06-11 13:08	28.4375	7.70721	6.002428	tercemar ringan
2026-06-11 17:12	28.5625	7.9494	2.583372	tercemar berat
2026-06-11 17:22	27.875	7.83757	2.3266	tercemar berat
2026-06-11 17:27	28	7.96485	2.430409	tercemar berat
2026-06-11 17:32	28	7.85796	2.326342	tercemar berat
2026-06-11 17:37	28	7.64296	2.281347	tercemar berat
2026-06-11 18:18	28.3125	7.98091	2.025453	tercemar berat
2026-06-15 23:42	29.3125	7.74718	2.436276	tercemar berat
2026-06-15 23:45	29.3125	7.74384	2.385669	tercemar berat
2026-06-15 23:46	29.3125	7.75183	2.183743	tercemar berat
2026-06-15 23:50	29.375	7.75483	2.218352	tercemar berat
2026-06-15 23:52	29.375	7.7529	2.223803	tercemar berat

Berdasarkan Tabel 4. 20, hasil proses pelabelan menunjukkan bahwa setiap data kualitas air telah memiliki atribut label yang merepresentasikan kelas kualitas air, yaitu tercemar ringan, tercemar sedang, dan tercemar berat. Atribut label tersebut selanjutnya digunakan sebagai variabel target (*target*) pada proses pembentukan model klasifikasi menggunakan algoritma Random Forest, sedangkan parameter suhu (*temperature*) dan pH digunakan sebagai variabel masukan (*feature*) dalam proses pelatihan model.

Tabel 4. 21 Distribusi Data Hasil Pelabelan

No	Kelas Kualitas Air	Jumlah Data	Persentase (%)
1	Tercemar Ringan	1005	51,83
2	Tercemar Sedang	457	23,57
3	Tercemar Berat	477	24,60
Total		1939	100

Berdasarkan Tabel 4. 21, kelas tercemar ringan memiliki jumlah data paling banyak, yaitu sebanyak 1005 data (51,83%), sedangkan kelas tercemar berat dan tercemar sedang masing-masing berjumlah 478 data (24,60%) dan 454 data (23,57%). Perbedaan jumlah data antar kelas menunjukkan bahwa distribusi data belum sepenuhnya seimbang, meskipun selisihnya tidak terlalu besar.

4.3.5 Pembagian Data

Pada tahap ini dilakukan pembagian *dataset* menjadi data latih (*training data*) dan data uji (*testing data*) menggunakan metode *train-test split* dengan perbandingan 70:30. Sebanyak 70% data digunakan sebagai data latih untuk membangun model Random Forest, sedangkan 30% sisanya digunakan sebagai data uji untuk mengevaluasi performa model. Proses pembagian data juga menggunakan parameter *stratify* sehingga proporsi setiap kelas pada data latih dan data uji tetap terjaga sesuai dengan distribusi data awal.

Tabel 4. 22 Hasil Pembagian Data

No	*Dataset*	Jumlah Data	Persentase (%)
1	Data Training	1357	70
2	Data Testing	582	30
Total		1939	100

Berdasarkan Tabel 4. 22, sebanyak 1357 data (70%) digunakan sebagai data latih untuk membangun model Random Forest, sedangkan 582 data (30%) digunakan sebagai data uji untuk mengevaluasi performa model. Penggunaan parameter *stratify* bertujuan untuk menjaga proporsi setiap kelas pada data latih dan data uji sehingga distribusi kelas tetap konsisten dengan *dataset* hasil pelabelan.

Tabel 4. 23 Komposisi Data *Training* dan *Testing*

Kelas	Training	Testing
Tercemar Ringan	703	302
Tercemar Sedang	320	137
Tercemar Berat	334	143
Total	1357	582

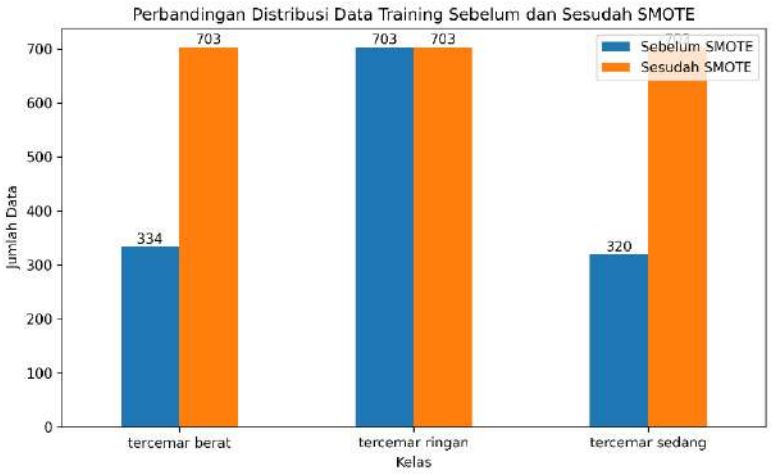
Tabel 4. 23 menunjukkan bahwa distribusi setiap kelas pada data *training* dan data *testing* tetap terjaga. Kondisi ini menunjukkan bahwa penggunaan parameter *stratify* pada proses *train-test split* berhasil mempertahankan proporsi masing-masing kelas sehingga tidak terjadi perubahan distribusi data setelah proses pembagian.

4.3.6 Penerapan *Synthetic Minority Oversampling Technique*

Setelah proses pembagian data selesai dilakukan, tahap selanjutnya adalah penerapan *Synthetic Minority Oversampling Technique* (SMOTE) pada data latih (*training data*). SMOTE diterapkan untuk menghasilkan data sintetis pada kelas yang memiliki jumlah data lebih sedikit sehingga distribusi data latih menjadi lebih seimbang. Penerapan SMOTE hanya dilakukan pada data latih, sedangkan data uji tetap menggunakan data asli agar proses evaluasi model tidak mengalami *data leakage*.

SMOTE diterapkan setelah proses *train-test split* dan sebelum pembentukan model Random Forest. Dengan demikian, data hasil *oversampling* hanya digunakan pada proses pelatihan model, sedangkan proses pengujian tetap menggunakan data asli untuk memperoleh hasil evaluasi yang objektif.

Untuk melihat pengaruh penerapan SMOTE terhadap distribusi data latih, dilakukan perbandingan jumlah data pada setiap kelas sebelum dan sesudah proses *oversampling.* Hasil perbandingan tersebut ditunjukkan pada Gambar 4. 23.



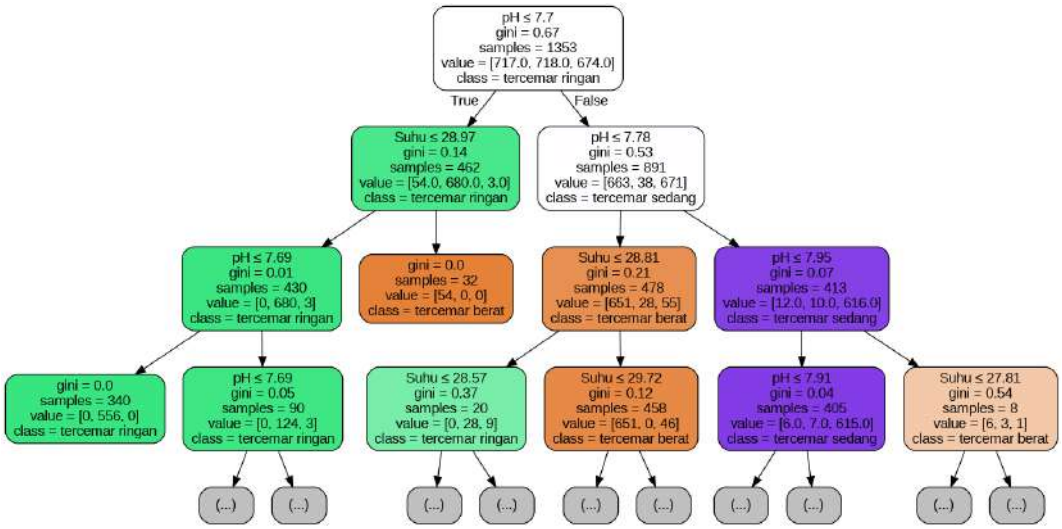
Gambar 4. 23 Distribusi Data *Training* Sebelum dan Sesudah SMOTE

Setelah penerapan SMOTE, jumlah data pada setiap kelas menjadi 703 data. Hal ini menunjukkan bahwa metode SMOTE berhasil menghasilkan data sintetis pada kelas minoritas sehingga distribusi data *training* menjadi seimbang. Data hasil *oversampling* tersebut selanjutnya digunakan sebagai data latih pada proses pembentukan model Random Forest.

4.3.7 Pembentukan Model Random Forest

Pada tahap ini, setelah data training diseimbangkan menggunakan SMOTE, tahap selanjutnya adalah pembentukan model klasifikasi menggunakan algoritma Random Forest. Model dibangun menggunakan data training hasil *oversampling*, sedangkan proses evaluasi dilakukan menggunakan data testing yang tidak mengalami perubahan. Pendekatan ini bertujuan untuk memperoleh hasil evaluasi yang objektif terhadap kemampuan model dalam mengklasifikasikan kualitas air.

Model Random Forest dibangun menggunakan algoritma Random Forest *Classifier* dengan jumlah pohon (*n_estimators*) sebanyak 100 dan nilai *random_state* sebesar 42. Parameter *random_state* digunakan untuk memastikan proses pelatihan model menghasilkan hasil yang konsisten setiap kali dijalankan. Proses pelatihan dilakukan menggunakan data latih hasil penerapan SMOTE, dengan parameter suhu dan pH sebagai fitur (*feature*) serta kelas kualitas air sebagai target (*label*). Pada tahap ini, model mempelajari pola hubungan antara fitur masukan dan kelas kualitas air sehingga dapat digunakan untuk melakukan prediksi pada data uji (*testing data*).



Gambar 4. 24 Visualisasi Salah Satu Hasil Pohon Keputusan

Pada Gambar 4. 24 menunjukkan salah satu pohon keputusan (*decision tree*) yang terbentuk pada model Random Forest. Pohon keputusan tersebut dibangun menggunakan kriteria pemisahan *Gini impurity*, yaitu ukuran yang digunakan untuk menentukan tingkat kemurnian data pada setiap simpul (*node*). Semakin kecil nilai *Gini impurity*, maka semakin baik proses pemisahan data karena data pada simpul tersebut semakin didominasi oleh satu kelas.

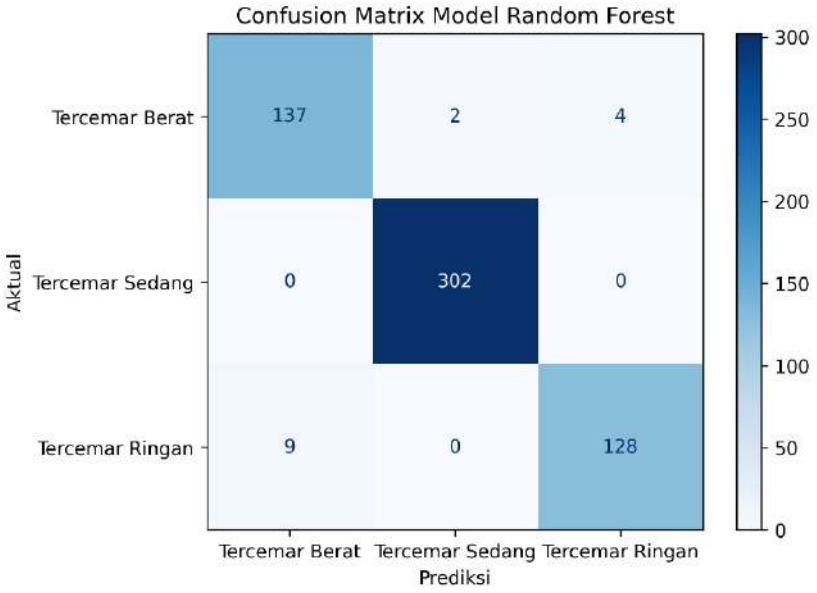
Pada setiap simpul, model memilih salah satu fitur, yaitu suhu (*temperature*) atau pH, kemudian membandingkan nilai fitur tersebut dengan suatu nilai ambang (*threshold*). Sebagai contoh, pada simpul akar dilakukan pemisahan berdasarkan kondisi pH ≤ 7,78. Apabila kondisi tersebut terpenuhi, maka data akan mengikuti cabang kiri (*True*), sedangkan jika tidak terpenuhi, data akan mengikuti cabang kanan (*False*). Proses pemisahan selanjutnya dilakukan berdasarkan nilai suhu atau pH hingga mencapai simpul daun (*leaf node*).

Setiap simpul pada pohon juga menampilkan informasi berupa nilai *Gini impurity*, jumlah data (*samples*), distribusi jumlah data pada masing-masing kelas (*value*), serta kelas mayoritas (*class*) yang menjadi hasil prediksi pada simpul tersebut. Proses pembentukan pohon dilakukan secara berulang hingga diperoleh aturan keputusan yang digunakan untuk mengklasifikasikan data ke dalam kelas tercemar ringan, tercemar sedang, atau tercemar berat.

4.3.8 Evaluasi Model

Pada tahap ini dilakukan evaluasi terhadap model klasifikasi yang telah dibangun menggunakan algoritma Random Forest. Evaluasi dilakukan terhadap data uji untuk mengetahui kemampuan model dalam melakukan klasifikasi.

Hasil evaluasi ditampilkan dalam bentuk *confusion matrix* yang menunjukkan jumlah prediksi benar dan salah pada masing-masing kelas, sehingga dapat digunakan untuk menganalisis performa model secara lebih detail.



Gambar 4. 25 Hasil *Confusion Matrix*

Berdasarkan Gambar 4. 25, hasil *confusion matrix* menunjukkan bahwa model Random Forest mampu mengklasifikasikan sebagian besar data dengan benar. Hal ini terlihat dari dominannya nilai pada diagonal utama *confusion matrix*, yang menunjukkan jumlah data yang berhasil diprediksi sesuai dengan kelas sebenarnya.

Pada kelas tercemar berat, sebanyak 137 data berhasil diklasifikasikan dengan benar. Sementara itu, terdapat 2 data yang salah diklasifikasikan sebagai tercemar sedang dan 4 data yang salah diklasifikasikan sebagai tercemar ringan.

Pada kelas tercemar sedang, model berhasil mengklasifikasikan seluruh 302 data dengan benar. Tidak terdapat data yang salah diprediksi sebagai tercemar berat maupun tercemar ringan.

Sementara itu, pada kelas tercemar ringan, sebanyak 128 data berhasil diprediksi dengan benar. Adapun 9 data lainnya salah diklasifikasikan sebagai tercemar berat, sedangkan tidak terdapat data yang salah diprediksi sebagai tercemar sedang.

Hasil *confusion matrix* menunjukkan bahwa model Random Forest memiliki kemampuan klasifikasi yang baik karena sebagian besar data berhasil diprediksi sesuai dengan kelas sebenarnya. Kesalahan klasifikasi yang terjadi relatif sedikit dan hanya ditemukan pada beberapa data yang memiliki karakteristik yang saling berdekatan.

$$Accuracy = \frac{137 + 302 + 128}{582} = \frac{567}{582} = 0.9742 = 97.42\%$$

1. Tercemar Berat

$$Precision = \frac{137}{137 + 0 + 9} = \frac{137}{146} = 0.9384$$

$$Recall = \frac{137}{137 + 2 + 4} = \frac{137}{143} = 0.9580$$

$$F1\text{-}Score = 2 \text{ x } \frac{0.93 \; x \; 0.95}{0.93 + 0.95} = 0.9481$$

2. Tercemar Ringan

$$Precision = \frac{302}{302 + 2 + 0} = \frac{302}{304} = 0.9934$$

$$Recall = \frac{302}{302 + 0 + 0} = \frac{302}{302} = 1.00$$

$$F1\text{-}Score = 2 \text{ x } \frac{0.99 \; x \; 1.00}{0.99 + 1.00} = 0.9967$$

3. Tercemar Sedang

$$Precision = \frac{128}{128 + 4 + 0} = \frac{128}{132} = 0.9672$$

$$Recall = \frac{128}{128 + 9 + 0} = \frac{128}{137} = 0.9343$$

$$F1\text{-}Score = 2 \text{ x } \frac{0.96 \; x \; 0.93}{0.96 + 0.93} = 0.9517$$

Tabel 4. 24 Hasil Evaluasi Model Random Forest

	Precision	*Recall*	*F1-Score*	*Support*
Tercemar Berat	0.9384	0.9580	0.9481	143
Tercemar Ringan	0.9934	1.00	0.9967	302
Tercemar Sedang	0.9697	0.9343	0.9517	137
Accuracy			0.9742	582
Macro avg	0.9672	0.9641	0.9655	582
Weighted avg	0.9743	0.9742	0.9742	582

Berdasarkan Tabel 4. 24, model Random Forest menunjukkan performa yang sangat baik dalam mengklasifikasikan kualitas air. Hal ini ditunjukkan dengan nilai *accuracy* sebesar 0,97, yang menunjukkan bahwa sebagian besar data uji berhasil diklasifikasikan dengan benar.

Pada kelas tercemar ringan, model memperoleh nilai *Precision* sebesar 0,99, *Recall* sebesar 1,00, dan *F1-Score* sebesar 1,00. Nilai tersebut menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mengenali seluruh data pada kelas tercemar ringan, dengan hampir tidak terdapat kesalahan klasifikasi.

Pada kelas tercemar berat, model memperoleh nilai *Precision* sebesar 0,94, *Recall* sebesar 0,96, dan *F1-Score* sebesar 0,95. Hasil tersebut menunjukkan bahwa model mampu mengidentifikasi sebagian besar data pada kelas tercemar berat dengan tingkat ketepatan yang tinggi.

Sementara itu, pada kelas tercemar sedang, model memperoleh nilai *Precision* sebesar 0,97, *Recall* sebesar 0,93, dan *F1-Score* sebesar 0,95. Nilai tersebut menunjukkan bahwa model juga memiliki performa yang baik dalam mengklasifikasikan kelas tercemar sedang, meskipun masih terdapat sejumlah kecil data yang mengalami kesalahan klasifikasi.

4.4 Implementasi Model Random Forest pada *Back-end*

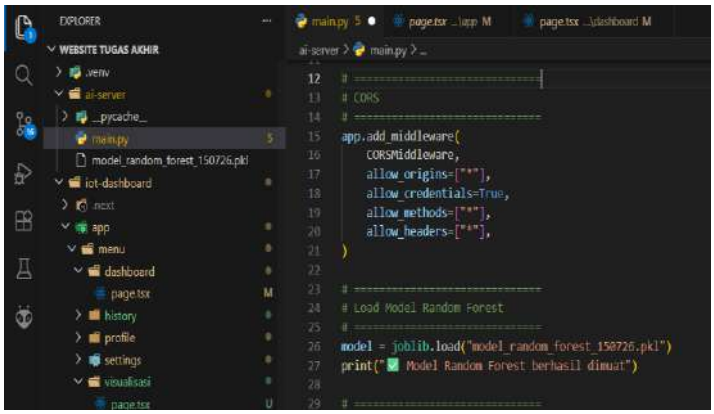
Pada tahap ini dilakukan implementasi model Random Forest pada sistem *back-end* untuk mengklasifikasikan kualitas air tambak udang. Model yang telah melalui proses pelatihan dan evaluasi kemudian diintegrasikan ke dalam aplikasi *back-end* menggunakan *framework* Flask sehingga dapat digunakan untuk melakukan prediksi kualitas air berdasarkan data masukan yang diterima. Hasil

prediksi selanjutnya dikirimkan melalui *Application Programming Interface* (API) dalam format JSON dan ditampilkan pada sistem *dashboard monitoring* secara *real-time*.

4.4.1 Integrasi Model Random Forest

Setelah model Random Forest selesai melalui proses pelatihan dan evaluasi, langkah selanjutnya adalah mengintegrasikan model tersebut ke dalam sistem *back-end*. Model yang telah dilatih disimpan dalam format *pickle* (.pkl) menggunakan *library joblib,* sehingga dapat digunakan kembali tanpa perlu melakukan proses pelatihan ulang setiap kali sistem dijalankan.

Pada saat aplikasi *back-end* dijalankan, model Random Forest dimuat (*load*) menggunakan fungsi *joblib.load().* Proses ini memungkinkan *back-end* memanfaatkan model yang telah dibangun untuk melakukan prediksi kualitas air berdasarkan data masukan yang diterima. Potongan kode proses pemuatan model ditunjukkan pada Gambar 4. 26.



Gambar 4. 26 Proses Pemuatan Model Random Forest

Setelah model berhasil dimuat, *back-end* menyediakan layanan *Application Programming Interface* (API) sebagai media komunikasi antara sistem *monitoring* dengan model Random Forest. Data parameter kualitas air, yaitu suhu (*temperature*) dan pH, dikirimkan ke *back-end* melalui API, kemudian diproses oleh model untuk menghasilkan prediksi kategori kualitas air berupa tercemar ringan, tercemar sedang, atau tercemar berat.

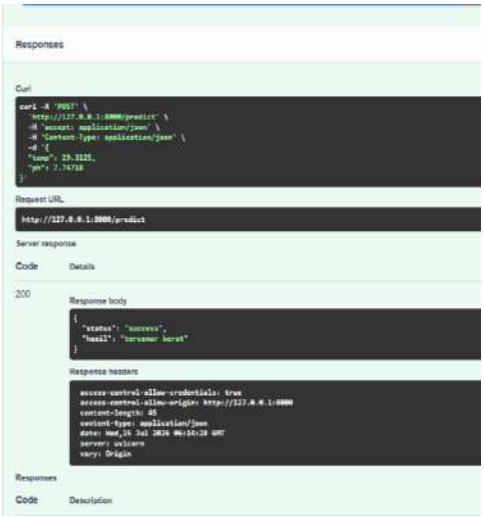
Hasil prediksi selanjutnya dikembalikan dalam format JSON melalui API sehingga dapat diterima oleh sistem *dashboard monitoring* untuk ditampilkan

kepada pengguna secara *real-time*. Dengan mekanisme tersebut, proses klasifikasi dapat dilakukan secara otomatis setiap kali sistem menerima data parameter kualitas air.

4.4.2 Pengujian Prediksi Model

Pengujian prediksi model dilakukan untuk memastikan bahwa model Random Forest yang telah diintegrasikan pada *back-end* dapat menerima data masukan, melakukan proses klasifikasi, serta mengembalikan hasil prediksi dengan benar. Pengujian dilakukan menggunakan layanan Swagger UI yang disediakan oleh *framework* FastAPI melalui *endpoint /predict.*

Pada proses pengujian, data masukan berupa parameter suhu (*temperature*) dan pH dikirimkan dalam format JSON ke *back-end*. Selanjutnya, data tersebut diproses oleh model Random Forest yang telah dimuat menggunakan *joblib.load()*. Setelah proses prediksi selesai dilakukan, sistem mengembalikan hasil klasifikasi dalam format JSON yang berisi status proses dan kategori kualitas air.



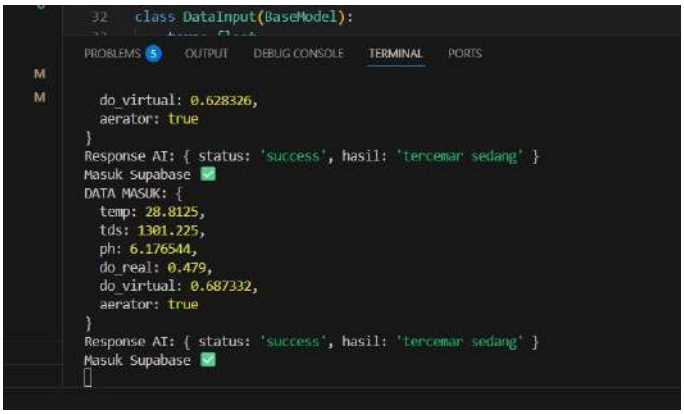
Gambar 4. 27 Hasil Pengujian Prediksi Model dengan API

Berdasarkan Gambar 4. 27, data masukan berupa suhu sebesar 29,3125°C dan nilai pH sebesar 7,74718 berhasil diterima oleh *back-end* melalui *endpoint /predict.* Selanjutnya, data tersebut diproses oleh model Random Forest yang telah diintegrasikan ke dalam sistem sehingga menghasilkan prediksi kategori kualitas air sesuai dengan data masukan yang diberikan.

Selain menghasilkan prediksi yang sesuai, sistem juga mengembalikan respons dengan status *success* dan kode HTTP 200 (OK). Hal ini menunjukkan bahwa proses komunikasi antara layanan API dengan model Random Forest berjalan dengan baik, sehingga model dapat digunakan untuk melakukan klasifikasi kualitas air secara otomatis berdasarkan data masukan yang diberikan.

4.4.3 Pengujian Implementasi Model pada *Back-end*

Setelah model Random Forest berhasil diintegrasikan ke dalam sistem *back-end*, tahap selanjutnya adalah melakukan pengujian implementasi model. Pengujian ini bertujuan untuk memastikan bahwa model dapat menerima data masukan melalui API, memproses data tersebut menggunakan model Random Forest, serta mengembalikan hasil prediksi kualitas air sesuai dengan data yang diberikan.



Gambar 4. 28 Hasil Respon API pada Proses Prediksi

Berdasarkan Gambar 4.28, hasil implementasi pada backend menunjukkan bahwa sistem berhasil menerima data masukan berupa nilai temperatur (*temp*), TDS, pH, dan DO real dari perangkat. Selanjutnya, sistem melakukan proses perhitungan nilai DO Virtual menggunakan model yang telah dibangun, kemudian mengirimkan nilai tersebut ke model Random Forest melalui API untuk dilakukan proses klasifikasi kualitas air.

Hasil pengujian menunjukkan bahwa model Random Forest berhasil mengembalikan respons berupa status *success* beserta hasil prediksi kualitas air, seperti kategori tercemar sedang. Setelah proses prediksi selesai, data hasil klasifikasi kemudian berhasil disimpan ke dalam *database* Supabase, yang ditunjukkan dengan munculnya informasi "Masuk Supabase" pada terminal.

Tabel 4. 25 Hasil Pengujian Implementasi

No	temp	ph	Do_virtual	Kualitas Air
1	30.125	6.07451	6.235	tercemar berat
2	30.125	6.08465	7.43502	tercemar berat
3	30.125	4.5933	7.95383	tercemar berat
4	30.0625	1.05949	3.65276	tercemar berat
5	30.125	1.05974	3.63053	tercemar berat
6	30.125	1.07809	3.6297	tercemar berat
7	30.1875	6.23691	7.43678	tercemar berat
8	28.375	6.76788	8.11669	tercemar ringan
9	28.375	6.93538	4.8411	tercemar ringan
10	28.4375	6.98984	4.89984	tercemar ringan
11	29.875	2.53027	7.48291	tercemar berat
12	29.9375	1.15294	7.02735	tercemar berat
13	28.5625	5.25163	1.42808	tercemar sedang
14	28.6875	6.31412	1.06664	tercemar sedang
15	28.6875	6.57041	9.85304	tercemar sedang
16	28.6875	6.58552	14.0627	tercemar sedang
17	28.5625	6.62774	9.30603	tercemar sedang
18	28.625	6.74916	2.3207	tercemar sedang
19	28.6875	1.83137	1.54735	tercemar sedang
20	29.5625	1.11995	1.24827	tercemar berat

Berdasarkan Tabel 4. 25, Hasil pengujian menunjukkan bahwa sistem dapat menerima data masukan dan mengembalikan hasil prediksi dalam bentuk kategori kualitas air, yaitu tercemar ringan, tercemar sedang, dan tercemar berat. Berdasarkan hasil tersebut, proses komunikasi antara API dan model Random Forest telah berjalan sesuai dengan mekanisme yang dirancang.

Namun, pada beberapa data uji masih ditemukan hasil klasifikasi yang belum sepenuhnya sesuai dengan kondisi yang diharapkan berdasarkan nilai parameter kualitas air. Perbedaan tersebut menunjukkan bahwa meskipun model memiliki performa evaluasi yang baik pada tahap pengujian menggunakan data uji, hasil prediksi pada tahap implementasi terhadap data baru masih dipengaruhi oleh karakteristik data masukan yang berbeda dengan data pelatihan. Oleh karena itu, model masih memiliki keterbatasan dalam melakukan generalisasi terhadap seluruh variasi data yang ditemui pada kondisi implementasi.

Hasil ini menunjukkan bahwa evaluasi model yang tinggi belum menjamin seluruh prediksi pada tahap implementasi akan sesuai dengan kondisi lapangan, sehingga diperlukan penambahan data pelatihan yang lebih beragam serta

penyempurnaan model agar kemampuan generalisasi terhadap data baru dapat ditingkatkan.

4.4.4 Pengujian Sistem *Monitoring*

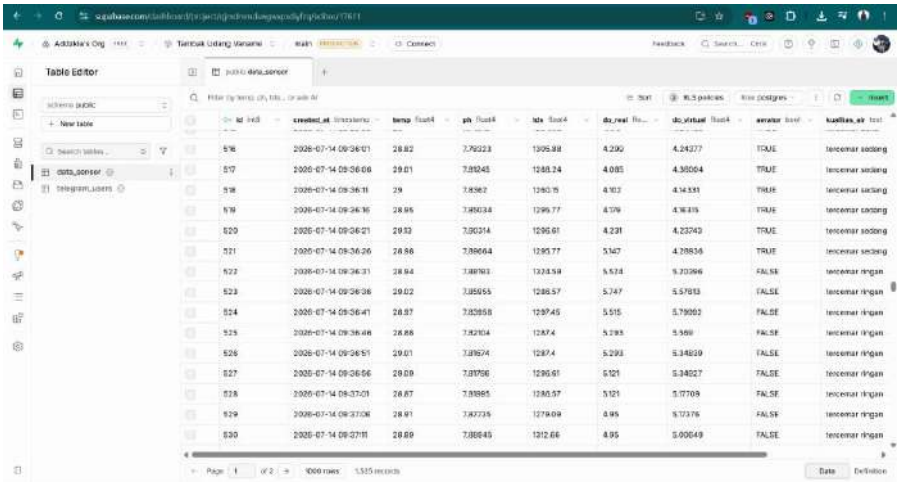
Setelah implementasi model Random Forest pada *back-end* selesai dilakukan, tahap selanjutnya adalah pengujian sistem *monitoring* secara keseluruhan. Pengujian ini bertujuan untuk memastikan bahwa setiap komponen sistem dapat bekerja secara terintegrasi, mulai dari proses pengiriman data sensor hingga penampilan hasil klasifikasi pada *dashboard monitoring*.

Gambar 4. 29 Pengujian Metode Random Forest di Tambak.

Gambar 4. 29 menunjukkan proses pengujian metode Random Forest di lokasi tambak. Pada tahap ini dilakukan pengambilan data menggunakan perangkat yang telah terpasang, kemudian data dikirim ke *backend* untuk diproses menggunakan model Random Forest. Selanjutnya, hasil klasifikasi kualitas air disimpan ke dalam basis data dan ditampilkan pada *dashboard monitoring* sehingga dapat digunakan sebagai informasi kondisi kualitas air tambak secara *real-time*.

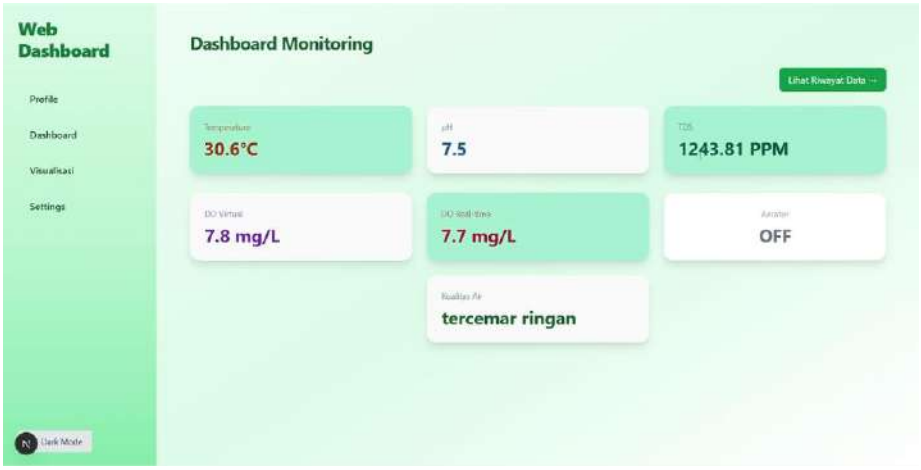
Pengujian dilakukan dengan mengamati alur pengiriman data dari perangkat ESP32 melalui protokol MQTT menuju *database*. Data yang telah tersimpan selanjutnya diproses oleh *back-end* menggunakan model Random Forest untuk menghasilkan klasifikasi kualitas air. Hasil klasifikasi tersebut kemudian

ditampilkan pada *dashboard monitoring* sehingga dapat dipantau oleh pengguna secara *real-time*.



Gambar 4. 30 Data Tersimpan pada *Database*

Pada Gambar 4. 30 menunjukkan data hasil pengiriman dari perangkat ESP32 yang telah berhasil diterima dan disimpan ke dalam *database*. Data yang tersimpan meliputi parameter kualitas air. Data tersebut selanjutnya digunakan sebagai sumber informasi yang ditampilkan pada *dashboard monitoring* sehingga pengguna dapat memantau kondisi kualitas air secara *real-time*. Hal ini menunjukkan bahwa proses pengiriman data dari perangkat ESP32 hingga penyimpanan ke dalam *database* telah berjalan sesuai dengan alur sistem yang dirancang.



Gambar 4. 31 Tampilan Kualitas Air pada *Dashboard*

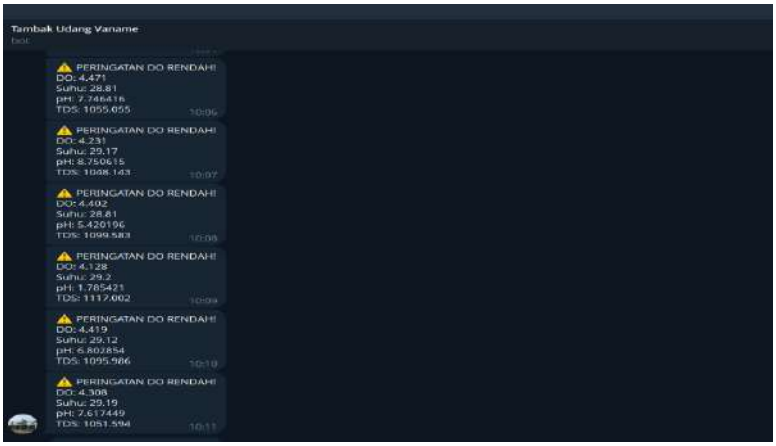
Pada Gambar 4. 31, *dashboard monitoring* menampilkan informasi kualitas air yang diperoleh dari hasil pembacaan sensor dan proses klasifikasi menggunakan model Random Forest. Informasi yang ditampilkan meliputi nilai suhu (*temperature*), pH, *Total Dissolved Solids* (TDS), *Dissolved Oxygen* (DO) virtual, *Dissolved Oxygen* (DO) *real-time*, status aerator, serta hasil klasifikasi kualitas air. Dengan adanya tampilan tersebut, pengguna dapat memantau kondisi kualitas air tambak secara *real-time* melalui antarmuka *dashboard*.

Berdasarkan hasil pengujian sistem *monitoring*, seluruh komponen sistem telah bekerja secara terintegrasi mulai dari proses pengiriman data oleh perangkat ESP32, penyimpanan data ke dalam *database*, proses klasifikasi menggunakan model Random Forest pada *back-end*, hingga penampilan hasil pada *dashboard monitoring*. Integrasi tersebut memungkinkan informasi kualitas air ditampilkan secara otomatis sehingga memudahkan pengguna dalam memantau kondisi kualitas air tambak. Parameter TDS tetap ditampilkan sebagai informasi hasil *monitoring* lapangan. Namun nilai tersebut tidak digunakan sebagai fitur pada model Random Forest karena sensor telah mencapai batas rentang pengukurannya pada kondisi air tambak yang memiliki kadar padatan terlarut tinggi.

4.4.5 Pengujian Notifikasi Peringatan Dini

Pengujian notifikasi peringatan dini dilakukan untuk memastikan bahwa sistem mampu memberikan informasi kepada pengguna secara otomatis berdasarkan hasil klasifikasi kualitas air yang dihasilkan oleh model Random Forest. Pengujian dilakukan dengan memberikan data masukan yang menghasilkan kondisi kualitas air tertentu, kemudian mengamati apakah sistem dapat menampilkan notifikasi sesuai dengan hasil klasifikasi yang diperoleh.

Pada proses pengujian, data hasil klasifikasi yang diperoleh dari *back-end* dikirimkan ke sistem *dashboard* untuk memicu tampilan notifikasi.



Gambar 4. 32 Tampilan Pesan Notifikasi

Berdasarkan Gambar 4. 32, sistem berhasil mengirimkan notifikasi peringatan dini ke aplikasi Telegram ketika nilai *Dissolved Oxygen* (DO) berada di bawah batas yang telah ditentukan. Notifikasi tersebut memuat informasi mengenai nilai suhu, pH, TDS serta waktu pengiriman sehingga pengguna dapat mengetahui kondisi kualitas air secara langsung. Notifikasi Telegram tetap menampilkan nilai TDS sebagai informasi *monitoring* kualitas air secara menyeluruh. Nilai tersebut tidak digunakan dalam proses prediksi Random Forest, tetapi tetap disampaikan kepada pengguna sebagai informasi kondisi aktual yang dibaca sensor di lapangan.

Notifikasi dikirimkan secara otomatis setiap kali sistem mendeteksi nilai DO berada pada kondisi rendah. Dengan adanya fitur ini, pengguna dapat segera mengetahui adanya penurunan kadar oksigen terlarut pada air tambak tanpa harus memantau *dashboard monitoring* secara terus-menerus. Informasi tersebut diharapkan dapat membantu pengguna dalam mengambil tindakan yang diperlukan untuk menjaga kualitas air tambak.

BAB 5
KESIMPULAN DAN SARAN

Bab ini menyajikan kesimpulan dan saran dari seluruh rangkaian penelitian yang telah dilakukan. Kesimpulan disusun berdasarkan hasil analisis, implementasi, dan pengujian sistem peringatan dini risiko kualitas air tambak udang menggunakan metode Random Forest, sedangkan saran diberikan sebagai bahan pertimbangan untuk pengembangan sistem dan penyempurnaan penelitian pada masa yang akan datang.

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem peringatan dini risiko kualitas air tambak udang menggunakan metode Random Forest, diperoleh beberapa kesimpulan sebagai berikut.

1. Sistem *monitoring* kualitas air tambak udang berhasil dirancang dan dibangun untuk mengelola data kualitas air secara terintegrasi. Keberhasilan sistem ini ditunjukkan melalui kemampuannya dalam menerima data hasil pembacaan sensor dari perangkat ESP32, mengirimkan data ke *database* melalui protokol MQTT, serta menampilkan informasi kualitas air pada *dashboard monitoring* secara *real-time*. Sistem juga dilengkapi dengan *back-end* yang mengintegrasikan model Random Forest untuk melakukan klasifikasi kualitas air secara otomatis, serta fitur notifikasi peringatan dini melalui Telegram sehingga pengguna dapat memperoleh informasi kondisi kualitas air dengan lebih cepat tanpa melakukan pencatatan secara manual.
2. Metode Random Forest berhasil diterapkan untuk melakukan klasifikasi kualitas air tambak udang. Proses penerapan diawali dengan pembentukan kelas berdasarkan nilai DO virtual menjadi tiga kategori, yaitu tercemar ringan, tercemar sedang, dan tercemar berat. Selanjutnya, model dilatih menggunakan parameter suhu dan pH sebagai data masukan yang memiliki korelasi dengan DO, kemudian diimplementasikan pada *back-end* sehingga mampu melakukan klasifikasi kualitas air secara otomatis.
3. Berdasarkan hasil evaluasi, model klasifikasi menggunakan metode Random Forest memperoleh nilai akurasi sebesar 97,42% dengan nilai

macro average Precision sebesar 96,72%, *Recall* 96,41%, dan *F1-Score* 96,55%. Hasil implementasi menggunakan data baru menunjukkan bahwa model mampu melakukan klasifikasi kualitas air dengan baik, meskipun masih terdapat beberapa perbedaan antara hasil prediksi model dan kondisi aktual pada sebagian data, sehingga peningkatan variasi data pelatihan masih diperlukan untuk meningkatkan kemampuan generalisasi model pada penelitian selanjutnya.

5.2 Saran

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem peringatan dini risiko kualitas air tambak udang menggunakan metode Random Forest, masih terdapat beberapa hal yang dapat dikembangkan pada penelitian selanjutnya. Oleh karena itu, penulis memberikan beberapa saran sebagai berikut.

1. Disarankan untuk menggunakan sensor yang dirancang khusus untuk pengukuran kualitas air payau, sehingga pengukuran parameter yang berkaitan dengan kandungan TDS dapat dilakukan dengan lebih baik.
2. Penelitian selanjutnya diharapkan dapat menggunakan jumlah dan variasi *dataset* yang lebih banyak agar model Random Forest memiliki kemampuan yang lebih baik dalam mengenali berbagai kondisi kualitas air tambak.
3. Disarankan dapat menambahkan parameter kualitas air lainnya, seperti salinitas, amonia, maupun kekeruhan (*turbidity*), sehingga dapat digunakan sebagai fitur tambahan pada model Random Forest untuk meningkatkan kualitas hasil klasifikasi.

DAFTAR PUSTAKA

Akbar, M. F., & Irawan, D. (2023). *Sistem Kontrol Kualitas Air Tambak Udang Berbasis Fuzzy Logic Fuzzy Logic Based Shrimp Pond Water Quality Control System.* **Jurnal Riset Rekayasa Elektro**, Vol.5*,* No.1, pp.23–32.

Alfiansyah, M. W., Wedashwara, I. G. P. W., & Mardiansyah, A. Z. (2021). *Implementasi IoT Untuk EWS Menggunakan Metode DES Model Holt Pada Tambak Udang Vaname.* **J-COSINE**, Vol.5, No.1, pp.20–29.

Ananta, N., Rachmawani, D., & Iromo, H. (2025). *Analisis Usaha Tambak Tradisional di Kalimantan Utara (Studi Kasus Petambak di Kota Tarakan).* **Prosiding Seminar Nasional Teknologi Komputer Dan Sains**, Vol.3, No.1, pp.109–114.

Bisong, E. (2019). **Google Colaboratory BT - Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners.** Apress, Berkeley, CA.

Boa, H., Suwannathep, S., Gunawan, B. I., & Bunnag, B. (2023). *Studi of Traditional Shrimp Farming; A Case Studi of Pond Productivity of Several Region in Indonesia.* **Cipasung Techno Pesantren: Scientific Journal**, Vol.17, No.2, pp.1–9.

Caesar, N. R., Yanuhar, U., Isdianto, A., Wiratno, E. N., Abidin, Z., Rusydi, A. N., Zakiyah, U., Kartikaningsih, H., Aida, G. R., & Ilmi, L. F. (2025). *Monitoring Kualitas Air pada Tambak Udang Vaname (Litopenaeus vannamei) di Perairan Pantai Utara Jawa Timur.* **Samakia: Jurnal Ilmu Perikanan**, Vol.16, No.1, pp.23–29.

Cholil, M. N. (2025). *Implementasi Algoritma Random Forest pada Model Klasifikasi Sekuens DNA Manusia Dalam Penyakit Diabetes*. **Tugas Akhir**, Universitas Islam Negeri Maulana Malik Ibrahim.

Forhad, H. M., Uddin, R., Chakrovorty, R. S., Ruhul, A. M., Faruk, H. M., Kamruzzaman, S., Sharmin, N., Jamal, A. H. M. S. I. M., Haque, M. M.-U., &

Morshed, A. M. (2024). *IoT based real-time water quality monitoring system in water treatment plants (WTPs)*. **Heliyon**, Vol.10, No.23, pp.1–11.

Gunawan, A. I., Gunawan, M. W., Syah, D., Suasono, Z. S., & Hamida, S. N. (2025). *Optimalisasi Kualitas Air pada Tambak Udang Vannamei Menggunakan Modul IoT.* **GUYUB: Journal of Community Engagement**, Vol.6, No.1, pp.314–338.

Halim, A. M., Fauziah, A., & Aisyah, N. (2022). *Kesesuaian Kualitas Air Pada Tambak Udang Vannamei.* **Jurnal Penelitian Chanos Chanos**, Vol.20, No.2, pp.77–88.

Harianto, D., Bintang, H. S., Ardiyanto, A., & Widyawan, V. L. D. (2021). *Development and Evaluation of an ESP32-based Temperature and Humidity Control Unit for Textile Storage.* **International Journal of Engineering Continuity**, Vol.4, No.1, pp.1–19.

Ibrahim, F. R., Syifa, F. T., & Pujiharsono, H. (2023). *Penerapan Sensor Suhu DS18B20 dan Sensor pH sebagai Otomatisasi Pakan Ikan Berbasis IoT.* **Journal of Telecommunication, Electronics, and Control Engineering (JTECE)**, Vol.05, No.02, pp.63–72.

Kementerian Kelautan dan Perikanan. (2023). **Profil Pasar Udang**. Kementerian Kelautan dan Perikanan, Jakarta.

Mansyur, A., & Rangka, N. A. (2008). *Potensi dan Kendala Pengembangan Budidaya Udang Vanamei di Sulawesi Selatan.* **Media Akuakultur**, Vol.3, No.1, pp.11–14.

Martin, R. S., & Dr. Yohanes Dewanto, M. (2023). *Prototipe kunci pintu otomatis menggunakan sensor kamera berbasis raspberry.* **Jurnal Teknologi Industri**, Vol.12, No.1, pp.21–29.

Mohan, M., Ch, B. P., Prasad, G. S., Srinu, R., Konduri, A., Goud, P. M. K., & Kumar, M. K. (2022). *Application and Importance of Aeration System in Aquaculture Pond: A Review.* **Asian Journal of Microbiology, Biotechnology and Environmental Sciences**, Vol.24, No.2, pp.371–377.

Mude, A. P. S. K., Bhoyar, N., Chaware, S., Bisen, S., & Meshram, A. (2024). *Solid State Relay*. **International Journal of Innovative Research in Technology (IJIRT)**, Vol.10, No.11, pp.1116–1119.

Nadhif, M. (2016). *Pengaruh Pemberian Probiotik Pada Pakan Dalam Berbagai Konsentrasi Terhadap Pertumbuhan dan Moralitas Udang Vaname*. **Tugas Akhir**, Universitas Airlangga.

Ningsih, G. A. D. S. A., & Pramartha, C. (2024). *Klasifikasi Kualitas Air Layak Minum menggunakan Algoritma Random Forest Classifier dan GridsearchCV*. **Jurnal Elektronik Ilmu Komputer Udayana,** Vol.12, No.1, pp.217–226.

Ningsih, L., Jaman, J. H., Salam, N. I., & Haikal, M. (2024). *Perbandingan Kinerja Algoritma Klasifikasi Status Mutu Air.* **Indonesian Journal of Multidisciplinary on Social and Technology**, Vol.2, No.1, pp.72–76.

Nurshofa, K. K., Ramadan, A. F., Faiz, N. F., Manurung, P., & Karlina, D. L. (2024). *Penggunaan Kontaktor pada Sistem Pengaman Motor Induksi 3 Phasa.* **Teknik Mesin, Industri, Elektro Dan Informatika**, Vol.3, No.4, pp.280–289.

Paepae, T., Bokoro, P. N., & Kyamakya, K. (2021). *From Fully Physical to Virtual Sensing for Water Quality Assessment : A Comprehensive Review of the Relevant State-of-the-Art*. **Sensors**, Vol.21, No.21, pp.6971.

Patkaew, S., Direkbusarakom, S., Hirono, I., Wuthisuthimethavee, S., Powtongsook, S., & Pooljun, C. (2024). *Effect of supersaturated dissolved oxygen on growth- , survival- , and immune-related gene expression of Pacific white shrimp (Litopenaeus vannamei).* **Veterinary World**, Vol.17, No.1, pp.50–58.

Pemerintah Republik Indonesia. (2021). *Peraturan Pemerintah Republik Indonesia Nomor 22 Tahun 2021 tentang Penyelenggaraan Perlindungan dan Pengelolaan Lingkungan Hidup*. URL:https://peraturan.bpk.go.id/Details/161852/?utm

Piłat-rożek, M., & Łagód, G. (2025). *Improvement in classification capabilities of surface water samples based on analysis of multidimensional data from gas sensor array.* **Annals of Agricultural and Environmental Medicine**, Vol.32, No.2, pp.222–229.

Pratama, E. W., & Kiswantono, A. (2022). *Electrical Analysis using ESP-32 Module in Realtime.* **Journal of Electrical Engineering and Computer Sciences**, Vol.7, No.2, pp.1273–1284.

Putra, A. S., Febri, S. P., AS, A. P., & Azmi, F. (2024). *Parameter Fisika, Kimia, dan Biologi Air pada Tambak Udang Vannamei (Litopenaeus vannamei) Semi Intensif.* **Jurnal Ilmiah Samudra Akuatika**, Vol.8, No.2, pp.43–52.

Rachmi, A. N. (2020). *Implementasi Metode Random Forest dan XGBOOST pada Klasifikasi Customer Churn*. **Tugas Akhir**, Universitas Islam Indonesia.

Sa'idi, M. M. (2020). *Analisis Parameter Kualitas Air Minum (pH, ORP, TDS, DO, dan Kadar Garam) pada Produk Air Minum Dalam Kemasan.* **Tugas Akhir**, Universitas Islam Indonesia.

Safrilly, S. M., & Badarudin, R. (2024). *Sistem Monitoring Suhu secara Real-Time berbasis Arduino Uno untuk Pemantauan Lingkungan*. **ELECTROPS: Jurnal Ilmiah Teknik Elektro**, Vol.3, No.2, pp.1–7.

Sasnita, P., Jewansah, & Wadud, R. F. (2024). A*nalisis Aplikasi Desain Printed Circuit Board untuk Mengembangkan Kemampuan Computational Thinking Siswa.* **UPGRADE: Jurnal Pendidikan Teknologi Informasi**, Vol.2, No.1, pp.41–48.

Shilman, M. I., Purnamawati, Susanti, R., & Redha, A. R. (2025). *Performance Kualitas Air Tambak Terhadap Kondisi Pertumbuhan Udang Vaname (Litopenaeus vannamei)*. **Jurnal Akuatiklestari**, Vol.9, No.1, pp.125–135.

Sihite, O. A., Febri, S. P., Putriningtias, A., Haser, T. F., & Nazlia, S. (2023). *Pengaruh pemberian jenis batu aerasi yang berbeda terhadap kelimpahan oksigen terlarut*. **Jurnal Ilmiah Satya Minabahari**, Vol.08, No.02, pp.56–63.

Sinuhaji, N. (2023). *Sistem Monitoring Tingkat Kebisingan Dan Kendali Intensitas Cahaya Pengunjung Pada Perpustakaan Berbasis Internet Of Things.* **RESOLUSI : Rekayasa Teknik Informatika Dan Informasi**, Vol.4, No.1, pp.100–105.

Sitohang, E. P., Mamahit, D. J., & Tulung, N. S. (2018). *Rancang Bangun Catu Daya DC Menggunakan Mikrokontroler ATmega 8535*. **Jurnal Teknik Elektro Dan Komputer**, Vol.7, No.2, pp.135–142.

Sukendar, W., Diniarti, N., Laheng, S., Redha, A. R., Dwiyanti, S., Lestari, D. P., Mardika, H. P. K., & Affandi, R. I. (2025). **Manajemen Kualitas Air Budidaya Ikan Air Tawar**. PT Penerbit Qriset Indonesia, Banjarnegara.

Supriana, Mahmudi, M., Musa, M., & Kusriani. (2020). *Hubungan pH dengan Parameter Kualitas Air Pada Tambak Intensif Udang Vannamei (Litopenaeus vannamei)*. **Journal of Fisheries and Marine Research**, Vol.4, No.3, pp.369–374.

Telaumbanua, M. (2025). *Pengaruh Oksigen Terlarut (DO) dalam Budidaya Perairan.* **PERAUT: Jurnal Perikanan Dan Kelautan**, Vol.02, No.01, pp.200–206.

Wijaya, J., Syauqy, D., & Primananda, R. (2021). *Sistem Monitoring dan Rekomendasi Kualitas Air Budidaya Bibit Ikan Nila Menggunakan Parameter Kekeruhan, Suhu, dan pH dengan Algoritma Random Forest*. **Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer**, Vol.1, No.1, pp.1–8.

Yuliantari, R. V., Novianto, D., Hartono, M. A., & Widodo, T. R. (2021). *Pengukuran Kejenuhan Oksigen Terlarut pada Air menggunakan Dissolved Oxygen Sensor.* **Jurnal Fisika Flux**, Vol.18, No.2, pp.101–104.

Yunarty, Y., Kurniaji, A., Budiyati, B., Renitasari, D. P., & Resa, M. (2022). *Karakteristik Kualitas Air dan Performa Pertumbuhan Udang Vaname Pola Intensif.* **Pena Akuatika : Jurnal Ilmiah Perikanan Dan Kelautan**, Vol.21 No.1, pp.75–88.

"Halaman ini sengaja dikosongkan"

LAMPIRAN

Lampiran 1. Surat Izin Pengambilan Data Penelitian

PEMERINTAH PROVINSI JAWA TIMUR
DINAS KELAUTAN DAN PERIKANAN
Jalan Ahmad Yani Nomor 152-B, Gayungan, Surabaya, Jawa Timur, 60235
Telepon (031) 8281672, Laman dkp.jatimprov.go.id, Pos-el diskanla@jatimprov.go.id

Surabaya, 13 April 2026

Nomor : 800.1/ 10332 /120.1/2026
Sifat : Biasa
Lampiran : -
Hal : **Izin Data Penelitian**

Yth. Wakil Direktur Bidang Akademik
Politeknik Perkapalan Negeri Surabaya
di-
Surabaya

Memperhatikan surat nomor: 397/DST/PL19/PP.02.10/2026, tanggal 1 April 2026 perihal Permohonan Data, maka kami dapat mengizinkan mahasiswa tersebut:

Nama / NIM	: AJIIB DZAKI AL KHOIRI / 0922040003
Judul Tugas Akhir	: "Sistem Peringatan Dini Resiko Kualitas Air Tambak Udang Vaname berdasarkan *Dissolved Oxygen* menggunakan Random Forest"
Nama / NIM	: MUHAMMAD NAUFAL RASHIF HARISMAN / 0922040018
Judul Tugas Akhir	: "Implementasi Sensor Virtual *Dissolved Oxygen* berbasis regresi linear pada system control aerator tambak udang vannamei"
Program Studi	: D4 Teknik Otomasi
Jurusan	: Teknik Kelistrikan Kapal
Periode Kegiatan	: 15 April 2026 s/d 15 Juli 2026
Kebutuhan Data	: Kualitas Air Tambak Udang Vannamei
Lokasi Kegiatan	: Instalasi Budidaya Air Payau (IBAP) Banjarkemuning

Diharapkan yang bersangkutan dapat berkordinasi dengan staf yang terkait di lokasi kegiatan dimaksud dan wajib mentaati segala peraturan yang berlaku.
Demikian atas perhatian dan kerjasamanya disampaikan terima kasih.

a.n Kepala Dinas Kelautan dan Perikanan
Provinsi Jawa Timur
Sekretaris

PEMERINTAH PROVINSI JAWA TIMUR DINAS KELAUTAN DAN PERIKANAN

Ir. SONI RUSTANTO, M.T
Pembina Tingkat I (IV/b)
NIP 197002221994031008

Tembusan :
1. Kepala Dinas Kelautan dan Perikanan Provinsi Jawa Timur;
2. Kepala UPT BAPL Bangil;
3. Koordinator IBAP Banjarkemuning;
4. Yang bersangkutan;
5. Arsip

"Halaman ini sengaja dikosongkan"

Lampiran 2. Data Hasil Klasifikasi

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
30.125	6.07451	0.0	6.229	6.235	False	tercemar berat	7/17/2026 01:17	1526
30.125	6.08465	0.0	7.436	7.43502	False	tercemar berat	7/17/2026 01:18	1529
30.125	4.5933	0.0	7.775	7.95383	False	tercemar berat	7/17/2026 01:19	1541
30.0625	1.05949	0.0	3.406	3.65276	True	tercemar berat	7/17/2026 01:21	1552
30.125	1.05974	0.0	3.383	3.63053	True	tercemar berat	7/17/2026 01:22	1554
30.125	1.07809	0.0	3.345	3.6297	True	tercemar berat	7/17/2026 01:23	1566
30.1875	6.23691	0.0	7.529	7.43678	False	tercemar berat	7/17/2026 01:24	1575
28.375	6.76788	0.0	6.372	8.11669	False	tercemar ringan	7/17/2026 01:25	1587
28.375	6.93538	201.126	5.079	4.8411	False	tercemar ringan	7/17/2026 01:26	1599
28.4375	6.98984	219.674	4.933	4.89984	False	tercemar ringan	7/17/2026 01:27	1611
29.875	2.53027	1299.33	6.919	7.48291	False	tercemar berat	7/17/2026 01:28	1623
29.9375	1.15294	1277.88	6.586	7.02735	False	tercemar berat	7/17/2026 01:29	1635
28.5625	5.25163	649.76	1.353	1.42808	True	tercemar sedang	7/17/2026 01:30	1647
28.6875	6.31412	654.12	0.865	1.06664	True	tercemar sedang	7/17/2026 01:31	1659
28.6875	6.57041	551.892	9.378	9.85304	False	tercemar sedang	7/17/2026 01:32	1671
28.6875	6.58552	563.598	15.126	14.0627	False	tercemar sedang	7/17/2026 01:33	1683
28.5625	6.62774	402.233	7.545	9.30603	False	tercemar sedang	7/17/2026 01:34	1695
28.625	6.74916	1303.24	2.029	2.3207	True	tercemar sedang	7/17/2026 01:35	1707
28.6875	1.83137	1311.02	0.916	1.54735	True	tercemar sedang	7/17/2026 01:36	1719
29.5625	1.11995	1288.33	0.932	1.24827	True	tercemar berat	7/17/2026 01:37	1731
29.625	4.583	1279.39	1.143	1.26193	True	tercemar berat	7/17/2026 01:38	1743
29.6875	6.70647	1278.75	0.567	0.73113	True	tercemar berat	7/17/2026 01:39	1755
29.625	6.51876	1320.53	0.332	0.952972	True	tercemar berat	7/17/2026 01:40	1767
29.5625	6.30521	1280.02	0.519	0.626688	True	tercemar berat	7/17/2026 01:41	1779
28.875	6.23173	1296.36	0.548	0.69397	True	tercemar sedang	7/17/2026 01:42	1791

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.8125	6.17811	1298.7	0.548	0.772544	True	tercemar sedang	7/17/2026 01:43	1803
28.875	6.2203	1308.15	0.436	0.712572	True	tercemar sedang	7/17/2026 01:44	1815
28.9375	6.24656	1295.69	0.359	0.489009	True	tercemar berat	7/17/2026 01:45	1827
28.9375	6.34394	1298.21	0.471	0.73153	True	tercemar berat	7/17/2026 01:46	1839
28.875	5.47302	1293.84	1.036	1.34061	True	tercemar sedang	7/17/2026 01:47	1851
28.875	1.05787	670.116	1.396	1.82746	True	tercemar sedang	7/17/2026 01:48	1863
29.0	1.05479	601.766	1.159	1.35463	True	tercemar berat	7/17/2026 01:49	1875
28.9375	1.05827	582.295	1.036	1.45231	True	tercemar berat	7/17/2026 01:50	1887
29.0	2.85583	537.611	1.979	1.33163	True	tercemar berat	7/17/2026 01:51	1899
28.9375	6.55222	538.234	1.772	1.9242	True	tercemar berat	7/17/2026 01:52	1911
29.0	6.58692	532.121	1.184	1.49302	True	tercemar berat	7/17/2026 01:53	1923
28.4375	7.34819	262.788	1.43	1.39898	True	*ERROR*_AI	7/21/2026 00:06	1933
28.5625	7.35046	261.845	1.807	1.81685	True	ERROR_AI	7/21/2026 00:07	1943
28.5625	7.35394	266.124	1.558	1.64704	True	ERROR_AI	7/21/2026 00:08	1955
28.5	7.35977	255.555	1.55	1.55953	True	ERROR_AI	7/21/2026 00:09	1967
28.5625	7.36133	262.833	1.661	1.65533	True	ERROR_AI	7/21/2026 00:10	1979
28.5625	7.3645	249.32	1.618	1.44099	True	ERROR_AI	7/21/2026 00:11	1991
28.5625	7.36621	264.808	1.456	1.68017	True	ERROR_AI	7/21/2026 00:12	1997
28.625	7.36029	260.882	1.653	1.51959	True	tercemar sedang	7/21/2026 00:13	2007
28.625	7.36808	248.041	1.421	1.44137	True	tercemar sedang	7/21/2026 00:14	2019
28.5625	7.37393	252.949	1.43	1.74727	True	tercemar sedang	7/21/2026 00:15	2031
28.5625	7.40858	271.388	2.055	2.02902	True	tercemar sedang	7/21/2026 00:16	2043
28.5625	7.40186	267.111	2.004	1.85783	True	tercemar sedang	7/21/2026 00:17	2055
28.4375	7.39857	249.242	1.653	1.68478	True	tercemar ringan	7/21/2026 00:18	2062
28.5	7.40066	257.864	1.738	1.62654	True	tercemar sedang	7/21/2026 00:19	2071
28.4375	7.4037	253.21	2.141	1.74463	True	tercemar ringan	7/21/2026 00:20	2083

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.5	7.40328	271.047	1.764	1.8744	True	tercemar sedang	7/21/2026 00:21	2095
28.5625	7.40602	261.845	1.952	1.74671	True	tercemar sedang	7/21/2026 00:22	2107
28.5625	7.40101	243.707	1.798	1.99178	True	tercemar sedang	7/21/2026 00:23	2119
28.5625	7.40524	250.64	1.824	1.89879	True	tercemar sedang	7/21/2026 00:24	2126
28.5625	7.40358	259.869	1.764	1.70386	True	tercemar sedang	7/21/2026 00:25	2134
28.5625	7.40706	253.608	1.892	1.77966	True	tercemar sedang	7/21/2026 00:26	2146
28.5625	7.40945	249.98	1.987	1.82195	True	tercemar sedang	7/21/2026 00:27	2158
28.625	7.41072	252.654	1.798	2.00981	True	tercemar sedang	7/21/2026 00:28	2170
28.5625	7.4154	252.619	1.798	1.78813	True	tercemar sedang	7/21/2026 00:29	2182
28.5625	7.4271	247.67	1.961	2.07654	True	tercemar sedang	7/21/2026 00:30	2190
28.5625	7.41953	246.68	2.132	2.00815	True	tercemar sedang	7/21/2026 00:31	2198
28.625	7.42999	255.947	1.85	1.86505	True	tercemar sedang	7/21/2026 00:32	2210
28.625	7.42874	244.743	1.944	1.72773	True	tercemar sedang	7/21/2026 00:33	2222
28.6875	7.42677	267.146	1.832	1.72177	True	tercemar sedang	7/21/2026 00:34	2234
28.6875	7.43073	251.044	1.91	1.74584	True	tercemar sedang	7/21/2026 00:35	2246
28.6875	7.43743	268.459	1.875	1.66174	True	tercemar sedang	7/21/2026 00:36	2254
28.6875	7.43609	261.892	1.653	1.67806	True	tercemar sedang	7/21/2026 00:37	2261
28.6875	7.43569	256.306	1.995	1.82273	True	tercemar sedang	7/21/2026 00:38	2273
28.75	7.44186	259.291	2.021	2.07007	True	tercemar sedang	7/21/2026 00:39	2285
28.6875	7.45132	262.549	1.952	2.10412	True	tercemar sedang	7/21/2026 00:40	2297
28.6875	7.45366	262.549	1.875	1.72905	True	tercemar sedang	7/21/2026 00:41	2309
28.75	7.45859	252.066	1.884	2.06859	True	tercemar sedang	7/21/2026 00:42	2318
28.6875	7.45306	264.191	2.158	1.82273	True	tercemar sedang	7/21/2026 00:43	2324
28.6875	7.45271	256.635	2.089	1.88167	True	tercemar sedang	7/21/2026 00:44	2336
28.6875	7.45304	247.093	1.875	1.83793	True	tercemar sedang	7/21/2026 00:45	2348
28.6875	7.45331	249.398	1.901	1.89783	True	tercemar sedang	7/21/2026 00:46	2360

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.6875	7.39225	256.306	2.004	1.76501	True	tercemar sedang	7/21/2026 00:47	2372
28.75	7.02514	247.464	1.875	1.73062	True	tercemar sedang	7/21/2026 00:48	2381
28.75	7.07321	269.786	1.764	1.86695	True	tercemar sedang	7/21/2026 00:49	2387
28.75	7.07846	262.572	1.918	1.88289	True	tercemar sedang	7/21/2026 00:50	2399
28.75	7.04936	246.806	1.961	1.86572	True	tercemar sedang	7/21/2026 00:51	2411
28.75	7.00102	256.008	1.824	2.14134	True	tercemar sedang	7/21/2026 00:52	2423
28.75	6.98519	254.366	1.91	1.89424	True	tercemar sedang	7/21/2026 00:53	2435
28.75	6.98539	271.097	2.218	2.246	True	tercemar sedang	7/21/2026 00:54	2445
28.75	6.98482	267.819	2.244	2.33919	True	tercemar sedang	7/21/2026 00:55	2450
28.75	6.85261	250.423	2.072	2.13853	True	tercemar sedang	7/21/2026 00:56	2462
28.75	6.60989	272.081	1.858	1.88807	True	tercemar sedang	7/21/2026 00:57	2474
28.75	6.48296	251.738	2.338	2.29159	True	tercemar sedang	7/21/2026 00:58	2486
28.75	6.40993	271.425	2.689	2.40825	True	tercemar sedang	7/21/2026 00:59	2498
28.625	6.31006	261.54	3.777	3.89831	True	tercemar sedang	7/21/2026 01:00	2508
28.625	6.31932	275.671	3.4	3.42986	True	tercemar sedang	7/21/2026 01:01	2513
28.6875	6.43034	0.0	3.117	3.17714	True	tercemar sedang	7/21/2026 01:02	2524
28.6875	6.53492	262.878	2.389	2.72131	True	tercemar sedang	7/21/2026 01:03	2536
28.75	6.64735	260.603	2.081	2.12091	True	tercemar sedang	7/21/2026 01:04	2548
28.75	6.57293	269.458	2.064	2.03961	True	tercemar sedang	7/21/2026 01:05	2560
28.625	8.05779	260.553	2.132	2.18327	True	tercemar berat	7/21/2026 01:06	2568
28.6875	7.40524	258.607	2.286	2.2818	True	tercemar sedang	7/21/2026 01:07	2572
28.6875	6.61656	274.366	2.278	2.15683	True	tercemar sedang	7/21/2026 01:08	2584
28.75	6.63516	256.008	2.295	2.23935	True	tercemar sedang	7/21/2026 01:09	2596
28.6875	6.62438	267.474	2.406	2.46212	True	tercemar sedang	7/21/2026 01:10	2607
28.625	12.6545	246.966	2.629	18.578	False	tercemar berat	7/21/2026 01:11	2611
28.6875	7.62044	257.015	2.595	2.95406	False	tercemar sedang	7/21/2026 01:12	2615

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.6875	7.15637	259.839	2.449	1.52096	False	tercemar sedang	7/21/2026 01:13	2619
28.625	6.93344	255.445	2.483	0.910511	False	tercemar sedang	7/21/2026 01:14	2631
28.6875	6.76895	261.408	2.98	0.324542	False	tercemar sedang	7/21/2026 01:15	2643
28.625	6.64402	251.678	2.98	0.016734	False	tercemar sedang	7/21/2026 01:16	2655
28.6875	6.74249	252.306	2.552	0.242829	False	tercemar sedang	7/21/2026 01:17	2666
28.625	12.9286	251.666	2.86	2.58434	True	tercemar berat	7/21/2026 01:29	2670
28.625	9.74414	253.971	2.766	2.6516	True	tercemar berat	7/21/2026 01:30	2677
28.625	7.18926	278.956	2.757	2.7186	True	tercemar sedang	7/21/2026 01:31	2689
28.625	7.05173	268.772	2.843	2.79142	True	tercemar sedang	7/21/2026 01:32	2701
28.6875	7.08324	260.25	2.792	2.78185	True	tercemar sedang	7/21/2026 01:33	2713
28.625	6.9601	251.666	2.723	2.75107	True	tercemar sedang	7/21/2026 01:34	2725
28.625	6.95507	251.995	2.715	2.97313	True	tercemar sedang	7/21/2026 01:35	2737
28.625	7.04296	259.895	2.766	2.72329	True	tercemar sedang	7/21/2026 01:36	2749
28.625	7.07286	256.276	2.586	2.63605	True	tercemar sedang	7/21/2026 01:37	2761
28.625	13.2432	236.586	2.954	20.396	False	tercemar berat	7/21/2026 01:40	2763
28.625	10.5143	239.419	2.954	11.9687	False	tercemar berat	7/21/2026 01:41	2768
28.625	7.49869	255.445	2.86	2.65609	False	tercemar sedang	7/21/2026 01:42	2780
28.625	7.20688	247.594	2.8	1.75493	False	tercemar sedang	7/21/2026 01:43	2792
28.5625	7.12518	246.966	2.98	1.58065	False	tercemar sedang	7/21/2026 01:44	2804
28.5625	7.09262	246.337	2.963	1.48011	False	tercemar sedang	7/21/2026 01:45	2816
28.625	7.07092	246.337	2.877	1.33507	False	tercemar sedang	7/21/2026 01:46	2828
28.5625	6.99905	263.918	3.623	1.19115	False	tercemar sedang	7/21/2026 01:47	2840
28.5625	7.0459	262.663	3.64	1.33582	False	tercemar sedang	7/21/2026 01:48	2852
28.625	7.63963	275.838	4.522	3.09133	False	tercemar sedang	7/21/2026 01:49	2864
28.6875	7.09755	264.859	3.092	1.33931	False	tercemar sedang	7/21/2026 01:50	2876
28.6875	7.18696	255.759	2.706	1.61543	False	tercemar sedang	7/21/2026 01:51	2888

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.625	11.43	220.236	1.13	0.928121	True	tercemar berat	7/21/2026 01:52	2895
28.625	9.9078	262.526	2.997	2.3467	True	tercemar berat	7/21/2026 01:53	2899
28.6875	7.65441	273.382	2.757	2.66391	True	tercemar sedang	7/21/2026 01:54	2911
28.625	7.30794	268.772	2.757	2.74605	True	tercemar sedang	7/21/2026 01:55	2923
28.625	7.23737	277.97	2.415	2.63975	True	tercemar sedang	7/21/2026 01:56	2935
28.625	7.219	267.129	2.475	2.74094	True	tercemar sedang	7/21/2026 01:57	2947
28.625	7.19199	277.97	2.646	2.74324	True	tercemar sedang	7/21/2026 01:58	2959
28.625	7.13994	256.934	2.475	2.70171	True	tercemar sedang	7/21/2026 01:59	2971
28.625	7.14763	259.895	2.629	2.52074	True	tercemar sedang	7/21/2026 02:00	2983
28.625	7.16046	261.54	2.663	2.58	True	tercemar sedang	7/21/2026 02:01	2995
28.625	7.16854	275.014	2.646	2.46959	True	ERROR_AI	7/21/2026 02:02	3007
28.625	7.19234	267.457	2.441	2.5533	True	tercemar sedang	7/21/2026 02:03	3019
28.625	7.19003	259.566	2.886	2.56952	True	tercemar sedang	7/21/2026 02:04	3031
28.625	7.19802	258.579	2.963	2.68841	True	tercemar sedang	7/21/2026 02:05	3043
28.5625	7.19112	257.893	2.989	2.80734	True	tercemar sedang	7/21/2026 02:06	3055
28.625	7.17263	283.554	2.68	2.89694	True	tercemar sedang	7/21/2026 02:07	3067
28.625	7.18589	269.429	2.972	2.92875	True	tercemar sedang	7/21/2026 02:08	3079
28.5625	7.18537	277.308	2.877	2.95476	True	tercemar sedang	7/21/2026 02:09	3091
28.625	7.17502	264.17	2.775	2.68205	True	tercemar sedang	7/21/2026 02:10	3103
28.5625	7.16991	256.575	2.912	2.78231	True	tercemar sedang	7/21/2026 02:11	3115
28.625	7.19187	255.947	2.835	2.96095	True	tercemar sedang	7/21/2026 02:12	3127
28.5625	7.21395	266.453	2.903	3.02002	True	tercemar sedang	7/21/2026 02:13	3139
28.625	7.22503	261.54	3.126	2.89238	True	tercemar sedang	7/21/2026 02:14	3151
28.5625	7.24093	256.245	3.057	2.83079	True	tercemar sedang	7/21/2026 02:15	3163
28.625	7.25709	256.605	2.732	2.84767	True	tercemar sedang	7/21/2026 02:16	3175
28.5625	7.27098	257.893	2.963	2.79479	True	tercemar sedang	7/21/2026 02:17	3187

temp	ph	tds	do_real	do_virtual	aerator	kualitas_air	created_at	id
28.5625	7.3054	260.198	2.757	2.66616	True	tercemar sedang	7/21/2026 02:18	3199
28.5625	7.33052	263.162	2.757	2.71605	True	tercemar sedang	7/21/2026 02:19	3211
28.5	7.36081	256.545	2.929	3.01174	True	tercemar sedang	7/21/2026 02:20	3223
28.625	7.38262	256.605	2.98	2.85892	True	tercemar sedang	7/21/2026 02:21	3235
28.5625	7.40587	270.073	2.954	3.0714	True	tercemar sedang	7/21/2026 02:22	3247
28.625	7.41956	256.934	3.151	3.06217	True	tercemar sedang	7/21/2026 02:23	3259
28.5625	7.41291	262.833	2.895	2.8406	True	tercemar sedang	7/21/2026 02:24	3271
28.5	7.41592	273.352	2.946	3.06265	True	tercemar sedang	7/21/2026 02:25	3283
28.5625	7.43379	262.174	2.98	2.82269	True	tercemar sedang	7/21/2026 02:26	3295
28.5	7.41943	280.924	3.049	3.23333	True	tercemar sedang	7/21/2026 02:27	3307
28.5	7.38889	271.376	3.742	3.16512	True	tercemar sedang	7/21/2026 02:28	3319
28.5	7.25186	92.5684	3.323	3.22205	True	tercemar sedang	7/21/2026 02:29	3331
28.5	7.22642	252.584	4.222	4.47449	True	tercemar sedang	7/21/2026 02:30	3343
28.4375	7.31063	264.107	3.657	3.38011	True	tercemar ringan	7/21/2026 02:31	3355
28.5	7.29266	265.117	2.638	2.76901	True	tercemar sedang	7/21/2026 02:32	3367
28.4375	7.29032	261.467	2.843	2.84467	True	tercemar ringan	7/21/2026 02:33	3379
28.4375	7.28449	263.777	2.68	2.63232	True	tercemar ringan	7/21/2026 02:34	3391
28.375	7.28666	273.992	2.346	2.46151	True	tercemar ringan	7/21/2026 02:35	3403
28.4375	7.29124	267.736	2.432	2.28331	True	tercemar ringan	7/21/2026 02:36	3415
28.4375	7.29786	279.935	2.655	2.70096	True	tercemar ringan	7/21/2026 02:37	3427

"Halaman ini sengaja dikosongkan"

Lampiran 3. Biodata Mahasiswa

1. Nama : Ajiib Dzaki Al Khairi
2. NRP : 0922040003
3. Program Studi : D4 Teknik Otomasi
4. Agama : Islam
5. Status : Belum menikah
6. Alamat Asal : Demangan, Siman, Ponorogo
7. Nomor Telepon : 085732195658
8. Jenis Kelamin : Laki-laki
9. *Email* : ajibdzaki@student.ppns.ac.id
10. Tempat Tanggal Lahir : Jakarta, 10 Januari 2003
11. Nama Orang Tua/Wali : Moh. Imam Syafi'i
12. Alamat Orang Tua/Wali : Demangan, Siman, Ponorogo

PENDIDIKAN FORMAL			
Pendidikan	Tahun	Tempat Pendidikan	Jurusan
Diploma 4	2022 - 2026	Politeknik Perkapalan Negeri Surabaya	Teknik Otomasi
SMA	2018 - 2021	MAN 2 Ngawi	IPA
SMP	2015 – 2018	SMP Lukman Al Hakim	-
SD	2009 – 2015	SDN 1 Wonoketro	-

“Halaman ini sengaja dikosongkan”