



Penerapan REST API pada Aplikasi Antarmuka Alat Pemantauan Tambak Udang

Fiqri Abdul Aziz¹, M. Asep Subandri², Armada³

^{1,2}Teknik Informatika, Politeknik Negeri Bengkalis

³Teknik Sipil, Politeknik Negeri Bengkalis

¹fiqri.aaziz@gmail.com, ²msubandri@polbeng.ac.id, ³armada@polbeng.ac.id

Abstract

Increasing shrimp pond productivity is a top priority in fisheries cultivation, especially with the challenge of maintaining water quality as a determining factor for success. The use of modern technology such as the Internet of Things (IoT) allows the integration of monitoring tools with software-based interface applications. This study aims to develop a shrimp pond monitoring tool interface application that uses REST API to connect IoT devices with a Flutter-based application system. The methods used include software development using prototyping with a system design approach based on evaluations received from users to obtain results that meet user needs. The data obtained is sent to Firebase via REST API and displayed in real-time on a Flutter-based application. The results of the study show that the application of REST API allows faster and reliable data transmission between IoT devices and interface applications. The REST API achieved an average response time between 70–87 ms with 0% packet loss during 10-minute testing using Apache JMeter. The monitoring sensors reached accuracy levels of 93.3–96.0% for temperature and 93.7–96.4% for pH measurements. The resulting application makes it easy for users to monitor pond conditions in real-time via mobile devices. This technology supports faster pond management by delivering accurate information. This research is expected to be the basis for further development of IoT-based pond monitoring systems, focusing on improving system features and reliability to support the sustainability of shrimp pond cultivation.

Keywords: Firebase, Flutter, REST API, *Internet of Things* (IoT), *Shrimp Ponds Monitoring*.

Abstrak

Peningkatan produktivitas tambak udang menjadi prioritas utama dalam budidaya perikanan, khususnya dengan tantangan menjaga kualitas air sebagai faktor penentu keberhasilan. Pemanfaatan teknologi modern seperti *Internet of Things* (IoT) memungkinkan integrasi alat monitoring dengan aplikasi antarmuka berbasis perangkat lunak. Penelitian ini bertujuan mengembangkan aplikasi antarmuka alat monitoring tambak udang yang menggunakan REST API untuk menghubungkan perangkat IoT dengan sistem aplikasi berbasis Flutter. Metode yang digunakan meliputi pengembangan perangkat lunak menggunakan prototyping dengan pendekatan merancang sistem berdasarkan evaluasi yang diterima dari pengguna untuk mendapatkan hasil yang sesuai dengan kebutuhan pengguna. Data yang diperoleh dikirimkan ke Firebase melalui REST API dan ditampilkan secara real-time pada aplikasi berbasis Flutter. Hasil penelitian menunjukkan bahwa penerapan REST API menghasilkan performa komunikasi yang cepat dan stabil, dengan rata-rata waktu respon antara 70–87 ms dan tingkat packet loss 0% selama pengujian berkelanjutan selama 10 menit menggunakan Apache JMeter. Akurasi sensor monitoring mencapai 93,3–96,0% untuk suhu dan 93,7–96,4% untuk pH. Aplikasi yang dihasilkan memberikan kemudahan kepada pengguna untuk memantau kondisi tambak secara real-time melalui perangkat mobile. Teknologi ini mendukung pengelolaan tambak yang lebih cepat dengan penyampaian informasi secara akurat. Penelitian ini diharapkan menjadi dasar untuk pengembangan lebih lanjut sistem monitoring tambak berbasis IoT, dengan fokus pada peningkatan fitur dan keandalan sistem guna mendukung keberlanjutan budidaya tambak udang.

Kata kunci: Firebase, Flutter, REST API, *Internet of Things* (IoT), Pemantauan Tambak Udang.

1. Pendahuluan

Sektor perikanan telah beradaptasi dengan potensi ekonomi dari budidaya tambak udang sebagai sebuah bisnis. Kementerian Kelautan dan Perikanan Republik Indonesia menyatakan bahwa komoditas udang mencapai nilai ekspor tertingginya sebesar USD 2,04 miliar pada tahun 2020. Pemeliharaan kondisi tambak yang optimal, khususnya suhu air, tetap menjadi

tantangan signifikan karena berdampak langsung pada kesehatan dan pertumbuhan udang [1]. Pemantauan secara manual masih umum dilakukan oleh pengelola tambak, namun metode ini memakan waktu dan rawan kesalahan. Untuk mengatasi hal ini, alat pemantau berbasis *Internet of Things* (IoT) telah dikembangkan, yang mampu mengukur parameter seperti suhu dan pH secara *real-time* serta mengirimkan data ke basis data



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

terpusat [2]. Meskipun memiliki keunggulan, masalah secara *real-time* dari jarak jauh melalui aplikasi mobile komunikasi antara perangkat IoT dan aplikasi mobile—karena kondisi jaringan yang tidak stabil atau volume data yang besar—dapat membatasi efektivitas sistem [3].

Untuk menyelesaikan masalah ini, pengembangan aplikasi lintas platform dengan REST API menyediakan integrasi data yang lebih mudah dan memainkan peran penting dalam memungkinkan transfer data yang lebih cepat [4]. Selain itu, aplikasi mobile dapat dengan mudah disesuaikan baik untuk perangkat Android maupun iOS menggunakan Flutter, sehingga menjamin antarmuka yang konsisten dan respon yang cepat [5].

REST API adalah alat layanan web yang memungkinkan dua sistem berkomunikasi dengan menyediakan cara untuk menangani permintaan HTTP (GET, POST, PUT, dan DELETE) secara terstruktur. REST API menyediakan cara untuk mentransfer data di seluruh aplikasi web seperti XML atau JSON [6].

Sistem ini dirancang dengan pengembangan lintas platform, ini sangat penting untuk aplikasi sehingga dapat berjalan di berbagai jenis sistem operasi atau perangkat termasuk Windows, Linux, iOS, dan Android. Dukungan untuk pengembangan lintas platform memastikan bahwa sistem dapat digunakan dan diakses oleh lebih banyak pengguna [7].

Flutter, yang dikembangkan oleh Google sebagai SDK pengembangan aplikasi *mobile* lintas platform, digunakan sebagai pembuat antarmuka pengguna dalam sistem. Para perancang Flutter menyatakan bahwa tujuan utamanya adalah menciptakan pengalaman dan desain yang konsisten di seluruh platform, yaitu Android dan iOS [8].

Firebase digunakan untuk backend dan operasi *real-time* dalam sistem ini. Realtime Database memproses perubahan data dalam format JSON, yang secara otomatis dan cepat menjaga sinkronisasi semua klien yang terhubung [9].

Penulis mengutip beberapa jurnal sebelumnya dalam membantu menyelesaikan permasalahan dalam studi ini. Penelitian pertama adalah implementasi *Internet of Things* (IoT) dalam sistem pencarian lahan parkir menggunakan mikrokontroler NodeMCU. Studi ini bertujuan membantu pengguna dalam menemukan tempat parkir yang tersedia dengan memberikan pembaruan status secara *real-time* melalui IoT. Dengan menggunakan sensor HC-SR04 dan REST API, sistem berhasil mengirimkan data status parkir ke server web setiap lima detik, yang kemudian dapat diakses pengguna melalui aplikasi mobile [10].

Penelitian kedua adalah implementasi pengendalian perangkat dari jarak jauh melalui internet. Sebuah studi tentang antarmuka kontrol untuk mikrokontroler ESP8266 menggunakan Flutter 3.0 membuktikan bahwa pengguna dapat mengendalikan perangkat seperti LED

secara *real-time* dari jarak jauh melalui aplikasi mobile lintas platform. Sistem ini menggunakan Flutter dan Firebase Realtime Database untuk memastikan konektivitas dan kendali yang lancar [11].

Penelitian ketiga adalah penerapan IoT dalam pemantauan dan pengendalian aliran serta ketinggian air di inkubator pemberian ikan nila. Studi ini mengintegrasikan Arduino, sensor, dan Firebase dengan aplikasi mobile berbasis Flutter, memungkinkan pengamatan secara *real-time* dan kontrol pompa secara otomatis. Penggunaan algoritma pohon keputusan meningkatkan responsivitas dan efisiensi sistem [12].

Penelitian keempat menguraikan praktik terbaik dalam perancangan RESTful API untuk meningkatkan skalabilitas, keamanan, dan performa layanan web. Penerapan prinsip seperti penggunaan HTTP yang benar, skema penamaan sumber daya, serta protokol keamanan seperti HTTPS dan OAuth menjadi kunci utama. Dengan mematuhi standar ini, API dapat lebih efisien, mudah diintegrasikan, dan mendukung pertumbuhan teknologi digital yang berkelanjutan [13].

Penelitian kelima membandingkan Android Native dan Flutter. Flutter lebih halus dalam tampilan, lebih cepat dikembangkan karena berbasis *widget*, dan dianggap lebih mudah dipelajari. Survei menunjukkan banyak pengembang memilih Flutter untuk efisiensi dan kemudahan pengembangan [14].

Penelitian ke enam membahas implementasi RESTful API pada framework Laravel serta penggunaan simulator IoT Wokwi untuk memonitor suhu dan kelembaban. Metode waterfall digunakan sebagai pendekatan pengembangan perangkat lunak yang terstruktur. Hasil penelitian menunjukkan bahwa RESTful API mampu menciptakan antarmuka program aplikasi yang efisien dan terintegrasi menggunakan teknologi web dan IoT [15].

Penelitian ke tujuh bertujuan untuk mengimplementasikan arsitektur client-server berbasis Web Socket guna melakukan pertukaran data antara perangkat IoT dan server. Hasil pengujian menunjukkan bahwa protokol Web Socket memiliki nilai latency paling rendah di antara protokol lainnya, yaitu dengan rata-rata antara 10,75 ms hingga 23,29 ms. Nilai tersebut menunjukkan bahwa Web Socket memiliki efisiensi tinggi dan dapat digunakan sebagai alternatif pengganti MQTT dalam komunikasi *real-time* pada perangkat IoT [16].

Penelitian ke delapan menghasilkan sistem I-WAKE berbasis mikrokontroler ESP32 yang mampu memantau suhu dan pH air tambak secara *real-time*, serta memberikan notifikasi otomatis apabila kondisi air tidak normal. Sistem ini juga dilengkapi dengan kontrol relay yang terhubung pada pompa air untuk menjaga kestabilan kualitas air tambak. Hasil pengujian menunjukkan sistem bekerja dengan baik dalam

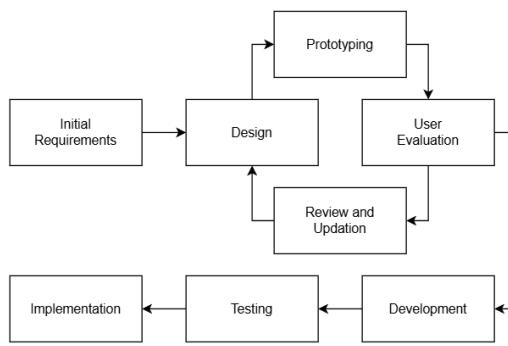
menjaga kualitas air ikan Koi dengan tingkat akurasi tinggi [17].

Penelitian ke sembilan membahas penerapan Firebase Realtime Database sebagai media penyimpanan dan sinkronisasi data antara mikrokontroler ESP32 dan antarmuka pengguna. Sistem dirancang menggunakan sensor ultrasonik HC-SR04 untuk mendeteksi sisa pakan, servo motor untuk mekanisme buka tutup pakan, dan integrasi Telegram Bot untuk notifikasi *real-time*. Hasil pengujian menunjukkan bahwa sistem mampu melakukan pemberian pakan otomatis sesuai jadwal, sinkronisasi data berjalan baik, serta seluruh fitur bekerja dengan tingkat respons yang tinggi dan akurasi deteksi mencapai lebih dari 95% [18].

Penelitian ke sepuluh mengembangkan aplikasi mobile berbasis Flutter dengan integrasi sensor oksimeter MAX30102 dan sensor suhu MLX90614 yang dihubungkan melalui mikrokontroler ESP32 menggunakan komunikasi Bluetooth Low Energy (BLE). Sistem ini memungkinkan pengguna memantau detak jantung, saturasi oksigen, dan suhu tubuh secara real-time dengan hasil akurasi pengukuran mencapai 95,55% untuk detak jantung, 99,01% untuk saturasi oksigen, dan 99,27% untuk suhu tubuh. Hasil pengujian menunjukkan bahwa aplikasi berfungsi dengan baik dalam menampilkan data kesehatan secara real-time dan dapat menjadi solusi monitoring kesehatan pribadi berbasis IoT [19].

2. Metode Penelitian

Sistem ini mengadopsi metode pengembangan *prototyping* di mana pengembang dan pengguna dapat melakukan iterasi melalui jalur berulang yang melibatkan komunikasi, perencanaan, desain, pembuatan prototipe, implementasi, dan umpan balik. Pada Gambar 1 dapat dilihat *prototyping* memungkinkan pengembang menyesuaikan sistem mereka sesuai kebutuhan berdasarkan pengalaman pengguna dan perubahan kebutuhan [20].



Gambar 1. Model Prototyping

2.1. Analisis Kebutuhan

Pada tahap ini, analisis spesifikasi awal dilakukan bersama pekerja lapangan yang bertugas untuk menjaga tambak, dengan merancang sebuah sistem atau

mengadakan pertemuan untuk mengumpulkan data kebutuhan. Proses ini dilakukan melalui wawancara. Hasil dari wawancara dibagi menjadi kebutuhan fungsional dan kebutuhan non-fungsional yang dapat dilihat pada Tabel 1 dan 2.

Tabel 1. Kebutuhan Fungsional

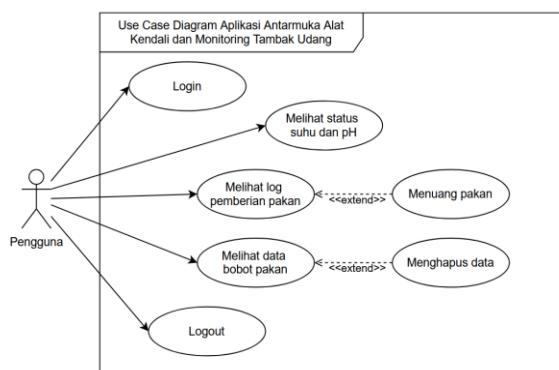
No.	Kebutuhan Pengguna
1.	Melihat status suhu air dari tambak.
2.	Melihat status pH air dari tambak.
3.	Memberi perintah pada alat untuk menuang pakan ke dalam anco.
4.	Memasukkan data bobot pakan pada aplikasi.

Tabel 2. Kebutuhan Non-Fungsional

No.	Kebutuhan Pengguna
1.	Sistem harus merespons setiap permintaan dalam waktu kurang dari 2 detik.
2.	Antarmuka pengguna harus intuitif dan mudah digunakan.
3.	Aplikasi harus dapat diakses dan berfungsi dengan baik pada berbagai perangkat seperti tablet, dan ponsel.

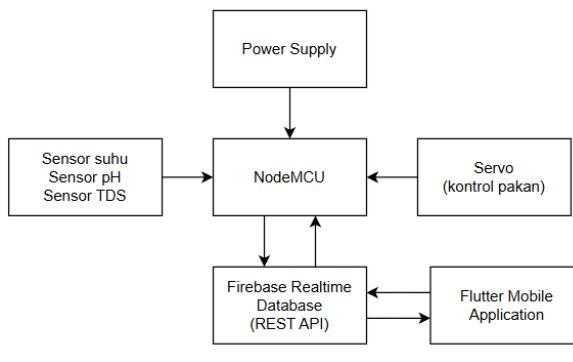
2.2. Merancang Desain

Tahap ini melibatkan pemodelan dengan menggunakan diagram-diagram UML. Pemodelan ini digunakan sebagai acuan untuk evaluasi dan perbaikan jika terdapat ketidaksesuaian. Sistem ini dijalankan oleh satu aktor, yaitu Pengguna yang merupakan seorang penjaga tambak. Pengguna yang melakukan kontrol langsung terhadap tambak udang dan memeriksa secara langsung kondisi tambak. Use case diagram dapat dilihat pada Gambar 2.



Gambar 2. Use Case Diagram

Sistem pemantauan tambak udang menggunakan NodeMCU ESP32 untuk mengintegrasikan berbagai sensor dan servo dengan *database* melalui REST API dan Firebase. Sistem ini memungkinkan pengguna untuk memantau kondisi tambak secara *real-time* melalui aplikasi, sekaligus memberikan kemampuan untuk mengontrol perangkat seperti servo. Data dari sensor, seperti suhu, pH air, dan *Total Dissolved Solids* (TDS), dikirimkan ke Firebase dalam format JSON stream untuk disimpan dan diakses oleh aplikasi pengguna. Alur sistem dapat dilihat pada Gambar 3.



Gambar 3. Alur Sistem

2.3. Evaluasi Prototipe

Setelah hasil rancangan awal atau prototipe dari aplikasi pemantauan tambak udang ini telah dibuat, perlu dilakukan evaluasi terhadap aplikasi untuk mendapatkan umpan balik dari pengguna tentang fitur dan tampilan dari prototipe aplikasi. Prototipe aplikasi digunakan oleh penjaga tambak untuk memberikan masukan sebagai umpan balik terhadap sistem. Hasil dari evaluasi yang diberikan pengguna dapat dilihat pada Tabel 3.

Tabel 3. Umpan Balik Pengguna

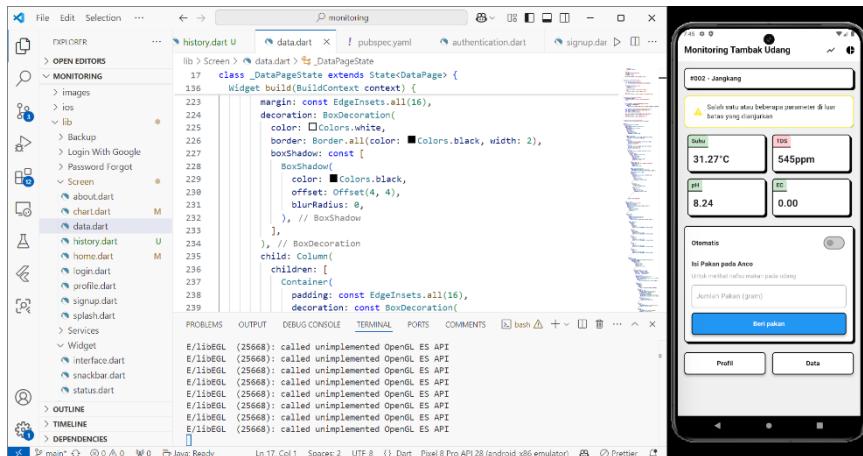
No.	Umpan balik pengguna terhadap prototipe
1.	Tambahkan opsi buat akun menggunakan google.
2.	Parameter tambak udang dimasukkan ke dalam halaman beranda.
3.	Menu kontrol pakan dimasukkan ke halaman beranda.

3. Hasil dan Pembahasan

Bagian ini menyajikan data hasil pengujian perangkat keras dan perangkat lunak, beserta analisis terhadap data tersebut berdasarkan uji coba yang dilakukan.

3.1. Pengkodean Sistem

Pengkodean sistem dibagi menjadi dua bagian yaitu perangkat lunak dan perangkat keras. Pengkodean perangkat lunak dilakukan pada Visual Studio Code berdasarkan desain *mockup* yang telah dibuat menggunakan Figma. Framework yang digunakan untuk membangun aplikasi *mobile* adalah Flutter dan bahasa pemrograman yang digunakan adalah Dart. Berikut merupakan tahapan pengkodean sistem pada aplikasi *mobile* pemantauan tambak udang, dapat dilihat pada Gambar 4 dan 5.



Gambar 4. Tahap Pengkodean Menggunakan Visual Studio Code

```

monitoring8 | Arduino 1.8.19
File Edit Sketch Tools Help
monitoring8
DallasTemperature dallasTemperature(soneWire);
Servo servo;

FirebaseDatabase db;
FirebaseAuth auth;
FirebaseConfig config;
unsigned long sendDataPrevMillis = 0;
bool signmpok = false;

void setup() {
  Serial.begin(115200);
  display.begin(SSD1064_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.setTextColor(WHITE);
  delay(1000);
}

void loop() {
  ...
}
  
```

Gambar 5. Tahap Pengkodean Menggunakan Arduino IDE

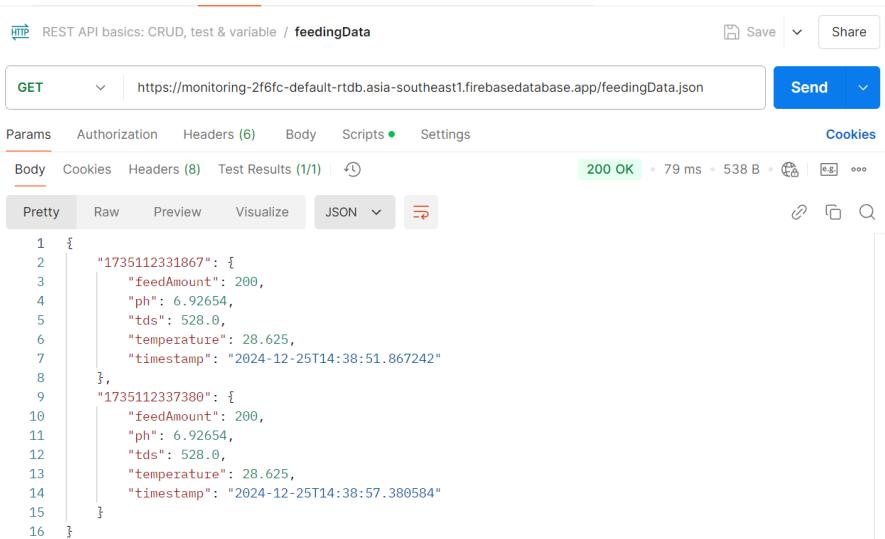
Pengkodean perangkat keras dilakukan pada perangkat lunak Arduino IDE berdasarkan rangkaian skematik yang telah dibuat.

3.2. Pengujian

Pada bagian pembahasan, dilakukan pengujian pada sistem ini untuk mengukur kapabilitas sistem dalam mengerjakan tugasnya sesuai dengan spesifikasi yang telah ditentukan. Dalam pengujian sistem ini terdapat beberapa pengujian yang dilakukan seperti, pengujian aplikasi *mobile*, pengujian pada alat yang digunakan dan pengujian pada REST API.

Pengujian Kinerja REST API dilakukan untuk mengevaluasi performa REST API dalam menerima, memproses, dan mengirimkan data. *Endpoint* diuji menggunakan POSTMAN dengan permintaan HTTP GET, POST, PUT, dan DELETE. Setiap respon server, termasuk kode status dan waktu respon, dicatat. Data yang diterima dibandingkan dengan data yang telah diunggah ke *database* untuk memastikan konsistensi.

Gambar 6 adalah pengujian *endpoint* GET feedingData menggunakan POSTMAN dengan waktu respon 79ms dan status 200 OK.

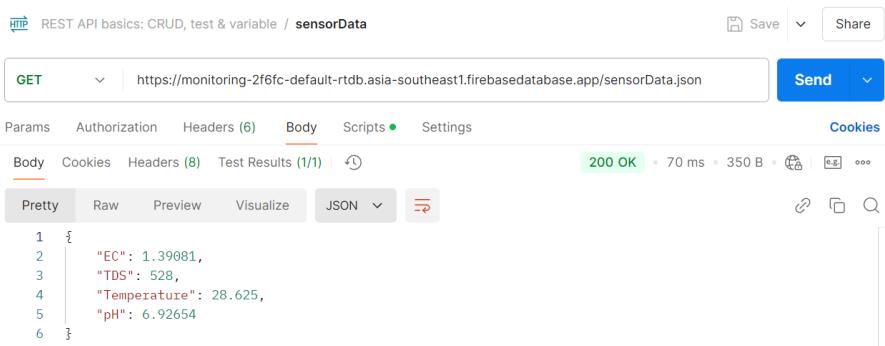


The screenshot shows a POSTMAN interface with the following details:

- Method: GET
- URL: https://monitoring-2f6fc-default.firebaseio.com/southeast1.firebaseio/app/feedingData.json
- Status: 200 OK
- Time: 79 ms
- Size: 538 B
- Headers: (6)
- Body (Pretty):

```
1 {
2   "1735112331867": {
3     "feedAmount": 200,
4     "ph": 6.92654,
5     "tds": 528.0,
6     "temperature": 28.625,
7     "timestamp": "2024-12-25T14:38:51.867242"
8   },
9   "1735112337380": {
10    "feedAmount": 200,
11    "ph": 6.92654,
12    "tds": 528.0,
13    "temperature": 28.625,
14    "timestamp": "2024-12-25T14:38:57.380584"
15  }
16 }
```

Gambar 6. Pengujian Endpoint GET feedingData



The screenshot shows a POSTMAN interface with the following details:

- Method: GET
- URL: https://monitoring-2f6fc-default.firebaseio.com/southeast1.firebaseio/app/sensorData.json
- Status: 200 OK
- Time: 70 ms
- Size: 350 B
- Headers: (6)
- Body (Pretty):

```
1 {
2   "EC": 1.39081,
3   "TDS": 528,
4   "Temperature": 28.625,
5   "pH": 6.92654
6 }
```

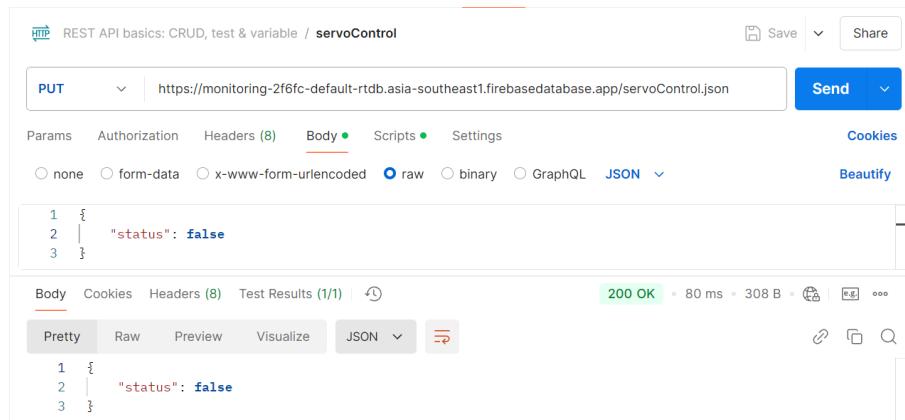
Gambar 7. Pengujian Endpoint POST sensorData

Gambar 7 adalah pengujian *endpoint* GET sensorData menggunakan POSTMAN dengan waktu respon 70ms dan status 200 OK.

Gambar 8 adalah pengujian *endpoint* PUT servoControl menggunakan POSTMAN dengan waktu respon 80ms dan status 200 OK.

Untuk mengevaluasi efektivitas REST API dalam sistem monitoring tambak udang, dilakukan perbandingan kinerja dengan protokol komunikasi lainnya yang umum digunakan dalam aplikasi IoT, yaitu MQTT dan metode Data Logging. Perbandingan kinerja protokol komunikasi dapat dilihat pada Tabel 4.

Hasil pengujian menunjukkan bahwa REST API memiliki waktu respon yang lebih lambat dibandingkan dengan MQTT, namun masih jauh lebih cepat dan stabil dibandingkan dengan metode Data Logging konvensional. MQTT menunjukkan performa terbaik dengan latency terendah berkisar antara 10,75-23,29 ms seperti pengujian yang dilakukan oleh Nugraha [16].



Gambar 8. Pengujian Endpoint PUT servoControl

Tabel 4. Perbandingan Kinerja Protokol Komunikasi

No.	Protokol	Rata-rata Waktu Respon (ms)	Packet Loss (%)
1.	REST API	70-87	0
2.	MQTT	10,75-23,29	0
3.	Data Logging	150-200	2-5

REST API menjadi pilihan yang untuk sistem monitoring tambak udang ini karena beberapa alasan di antaranya arsitektur REST API lebih mudah diintegrasikan dengan Firebase Realtime Database dan layanan cloud lainnya, REST API memiliki dukungan yang lebih luas untuk berbagai platform dan framework, termasuk Flutter yang digunakan dalam penelitian ini, Waktu respon 70-87 ms masih sangat memadai untuk monitoring parameter tambak yang tidak memerlukan update data dalam skala milidetik, dan REST API tidak memerlukan broker tambahan seperti MQTT, sehingga arsitektur sistem lebih sederhana. Dibandingkan dengan metode Data Logging yang masih banyak digunakan dalam monitoring tambak konvensional [2], REST API menunjukkan peningkatan performa yang signifikan dengan waktu respon 2-3 kali lebih cepat dan tingkat keandalan yang lebih tinggi dengan packet loss 0%. Hal ini membuktikan bahwa penerapan REST API memberikan nilai tambah yang substansial dalam modernisasi sistem monitoring tambak udang.

Pengujian akurasi alat pemantauan tambak udang ini bertujuan untuk mengukur keakuratan alat pemantauan tambak udang berbasis ESP32 dalam mendekripsi suhu dan pH. Sampel larutan dengan variasi suhu diuji dalam kondisi lingkungan dinamis yaitu kolam tambak udang. Pengukuran dilakukan menggunakan alat referensi sebagai banding. Hasilnya digunakan untuk menghitung selisih dan persentase akurasi perangkat dengan rumus $(\text{selisih}/\text{nilai aktual}) \times 100\%$. Data pengujian dapat dilihat pada Tabel 5.

Pengujian akurasi juga dilakukan pada sensor pH untuk mengetahui tingkat akurasi sensor pH. Sampel larutan dengan variasi pH diuji dalam kondisi lingkungan dinamis. Pengukuran dilakukan menggunakan alat

referensi sebagai banding. Hasilnya digunakan untuk menghitung selisih dan persentase akurasi perangkat dengan rumus $(\text{selisih}/\text{nilai aktual}) \times 100\%$. Data pengujian dapat dilihat pada Tabel 6.

Tabel 5. Pengujian Akurasi Sensor Suhu

No.	Hasil Sensor Suhu (°C)	Suhu Aktual (°C)	Selisih Suhu (°C)	Akurasi (%)
1.	24.83	25.58	0.75	97.07
2.	26.23	27.04	0.81	97.00
3.	25.56	26.45	0.89	96.64
4.	28.03	28.79	0.76	97.36
5.	23.87	25.14	1.27	94.95
6.	27.45	28.38	0.93	96.72
7.	26.78	27.82	1.04	96.26
8.	29.12	30.09	0.97	96.78
9.	25.34	26.21	0.87	96.68
10.	28.67	29.59	0.92	96.89

Tabel 6. Pengujian Akurasi Sensor pH

No.	Hasil Sensor pH	pH Aktual	Selisih pH	Akurasi (%)
1.	6.52	6.86	0.34	95.04
2.	6.64	6.92	0.28	95.95
3.	7.18	7.45	0.27	96.38
4.	7.82	8.05	0.23	97.14
5.	6.95	7.18	0.23	96.80
6.	8.91	9.18	0.27	97.06
7.	8.34	8.67	0.33	96.19
8.	9.02	9.26	0.24	97.41
9.	7.56	7.84	0.28	96.43
10.	8.65	8.82	0.17	98.07

Pengujian akurasi alat pemantauan dilakukan dalam kondisi dinamis yang merepresentasikan lingkungan tambak udang sebenarnya, di mana terdapat fluktuasi suhu akibat paparan sinar matahari, pergerakan air, dan aktivitas biologis. Kondisi dinamis dipilih untuk memastikan bahwa sistem dapat bekerja dengan andal dalam kondisi lapangan yang sesungguhnya. Hasil pengujian menunjukkan bahwa sensor suhu mencapai akurasi rata-rata 96,61% dengan selisih berkisar antara 0,75°C hingga 1,27°C dari nilai aktual. Selisih maksimum 1,27°C masih berada dalam toleransi yang

dapat diterima untuk monitoring tambak udang, kombinasi perangkat keras dan perangkat lunak. Data mengingat perubahan suhu dalam skala ini tidak berdampak signifikan terhadap kesehatan udang dalam jangka pendek. Sensor pH menunjukkan performa yang sangat baik dengan akurasi rata-rata 96,65% dan selisih berkisar antara 0,17 hingga 0,34 dari nilai pH aktual. Tingkat akurasi ini sangat memadai karena pH air tambak yang ideal untuk budidaya udang berkisar antara 7,5-8,5, dan fluktuasi kecil dalam pengukuran tidak akan memengaruhi keputusan pengelolaan tambak secara signifikan. Hasil ini menunjukkan bahwa alat monitoring mampu mengukur suhu dan pH dengan presisi yang cukup baik dalam berbagai kondisi pengujian.

Pengujian kompatibilitas dilakukan untuk memastikan bahwa sistem yang dikembangkan dapat berjalan dengan baik pada berbagai perangkat dan lingkungan yang berbeda. Tujuan dari pengujian ini adalah untuk mengidentifikasi kemungkinan masalah terkait tampilan, fungsionalitas, dan performa pada berbagai

Tabel 7. Hasil Pengujian Kompatibilitas

No.	Perangkat/ OS	Resolusi	Hasil
1.	Pixel 8/ Android	1080x1920px	Kompatibel
2.	Pixel 4/ Android	720x1280px	Kompatibel
3.	Pixel Tablet/ Android	2560x1600px	Kompatibel
4.	Pixel 8/ Android	1080x1920px	Kompatibel

Pengujian Fungsionalitas Aplikasi seperti terlihat pada Tabel 8, mencakup semua tampilan aplikasi, yaitu halaman masuk, halaman daftar, halaman beranda, halaman grafik, halaman data, halaman profil, halaman histori dan tombol untuk keluar dari akun.

Tabel 8. Pengujian Blackbox Aplikasi Pemantauan Tambak Udang

No.	Halaman	Data Uji	Skenario	Status
1.	Daftar	Sebuah email yang belum terdaftar	Daftar menggunakan email	Berhasil
2.	Masuk	Kredensial yang valid	Memasukkan kredensial yang valid	Berhasil
3.	Beranda	Akun yang telah dibuat sebelumnya	Pengguna masuk ke halaman beranda	Berhasil
4.	Grafik	Data yang baru masuk	Pengguna masuk ke halaman grafik	Berhasil
5.	Data	Data yang baru masuk	Memberi pakan melalui menu kontrol pakan	Berhasil
6.	Profil	Akun yang telah dibuat sebelumnya	Masuk dengan akun yang telah dibuat	Berhasil
7.	Histori dan Analisis	Data yang baru masuk	Pengguna menambah data baru	Berhasil
8.	Keluar	Tombol keluar pada halaman Profil	Pengguna menekan tombol keluar	Berhasil

Pengujian dilakukan menggunakan metode *blackbox* TDS, dan EC. Data ini penting untuk memastikan yang bertujuan untuk mengetahui semua fungsi berjalan dengan baik tanpa ada bug. Setiap fitur diuji menggunakan skenario yang mencerminkan kondisinya.

3.3 Penerapan pada Aplikasi Mobile

Setelah melewati proses pengkodean sistem dan berbagai macam pengujian, maka didapatkan hasil berupa aplikasi mobile pemantauan tambak udang seperti berikut.

Halaman beranda berfungsi sebagai pusat informasi utama yang menampilkan berbagai fitur penting untuk memantau dan mengontrol kondisi tambak secara *real-time*. Pada bagian atas halaman, terdapat informasi ID tambak yang dipantau, seperti contoh "#002 - Jangkang". Di bawahnya, terdapat indikator status parameter yang memberikan informasi apakah semua parameter berada dalam kondisi ideal, ditandai dengan ikon centang berwarna hijau dan teks "Semua parameter berada dalam rentang ideal". Jika salah satu berada di luar batas normal, maka ikon akan berubah warna menjadi kuning, dan jika semua parameter berada diluar batas normal maka ikon akan berubah warna menjadi merah. Selanjutnya, ditampilkan empat kotak parameter utama yang masing-masing menunjukkan data suhu, pH,

Selain fitur pemantauan, halaman ini juga menyediakan kontrol pemberian pakan otomatis. Pada bagian "Isi Pakan pada Anco", pengguna dapat memasukkan jumlah pakan dalam satuan gram, lalu menekan tombol "Beri pakan" untuk mengaktifkan alat pemberi pakan otomatis. Di bagian bawah halaman, terdapat dua tombol navigasi utama, yaitu "Profil" yang mengarah ke informasi akun pengguna, dan "Data" yang membawa pengguna ke halaman data historis. Aplikasi ini juga memiliki fitur tambahan seperti tampilan grafik dan analisis lebih lanjut terhadap data sensor. Seluruh fitur pada halaman beranda dirancang untuk memberikan kemudahan dalam memantau kondisi tambak serta melakukan kontrol operasional secara efisien dan terintegrasi. Halaman beranda dapat dilihat pada Gambar 9.

Pada halaman Data Pakan, pengguna dapat melihat riwayat pemberian pakan yang telah dilakukan melalui aplikasi. Setiap kali pengguna memasukkan jumlah pakan dan menekan tombol "Beri pakan" di halaman beranda, sistem secara otomatis merekam data tersebut bersama dengan parameter lingkungan saat itu, yaitu

suhu, TDS, dan pH. Data ini ditampilkan dalam bentuk tabel yang memuat informasi tanggal dan waktu, berat pakan yang diberikan, serta nilai parameter seperti suhu, TDS, dan pH. Dengan fitur ini, pengguna dapat memantau pola pemberian pakan serta kondisi lingkungan tambak saat pakan diberikan.



Gambar 9. Tampilan Halaman Beranda

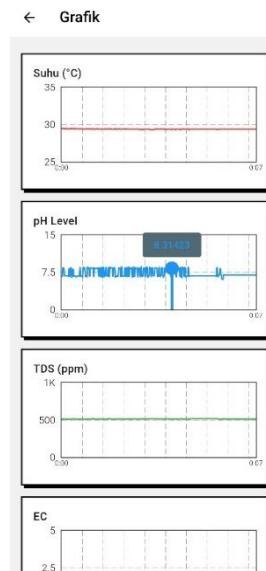
Selain menampilkan data, halaman ini juga menyediakan ikon untuk melakukan ekspor data ke format CSV yang ditandai dengan ikon panah hijau ke atas, serta ikon tempat sampah merah untuk menghapus data jika dianggap tidak lagi diperlukan. Fitur ekspor sangat berguna untuk keperluan dokumentasi, analisis, atau pelaporan, sedangkan fitur hapus memberikan fleksibilitas dalam manajemen data. Dengan adanya halaman ini, pengguna dapat dengan mudah meninjau, mengelola, dan menyimpan catatan aktivitas pemberian pakan dengan akurat dan terorganisir. Halaman data dapat dilihat pada Gambar 10.

Data Pakan				
Tanggal	Berat Pakan	Suhu	TDS	pH
07/12/2024 11:33	250g	31°C	0ppm	10.642
07/12/2024 11:33	300g	31°C	0ppm	10.642
07/12/2024 11:33	200g	31°C	0ppm	10.642
07/12/2024 11:33	150g	31°C	0ppm	10.642

Gambar 10. Tampilan Halaman Data

Halaman Grafik dapat diakses dari halaman beranda melalui tombol dengan ikon grafik yang terletak di pojok

kanan atas. Halaman ini menyajikan visualisasi data parameter kualitas air tambak udang secara *real-time* dalam bentuk grafik garis. Parameter yang ditampilkan meliputi suhu air, pH level, TDS, serta EC. Masing-masing parameter ditampilkan dalam grafik terpisah, yang memudahkan pengguna untuk memantau fluktuasi atau ketebalan nilai setiap parameter dari waktu ke waktu. Halaman grafik dapat dilihat pada Gambar 11.



Gambar 11. Tampilan Halaman Grafik

Pada halaman Histori & Analisis, pengguna dapat melihat rekapan data parameter kualitas air tambak udang berdasarkan periode waktu tertentu. Halaman ini dilengkapi dengan fitur pemilihan tanggal yang memungkinkan pengguna untuk menampilkan data historis pada hari tertentu. Setelah memilih tanggal, sistem akan menampilkan nilai rata-rata dari masing-masing parameter, yaitu suhu, pH, TDS, dan EC, dalam tiga kategori waktu: harian, mingguan, dan bulanan. Halaman histori dapat dilihat pada Gambar 12.

Histori & Analisis		
18 July 2025		
Rata-rata Suhu		
Harian	31.26	
Mingguan	31.26	
Bulanan	31.26	
Rata-rata pH		
Harian	8.23	
Mingguan	8.23	
Bulanan	8.23	
Rata-rata TDS		
Harian	544.56	
Mingguan	544.56	
Bulanan	544.56	
Rata-rata EC		
Harian	2.53	
Mingguan	2.53	
Bulanan	2.53	

Gambar 12. Tampilan Halaman Histori

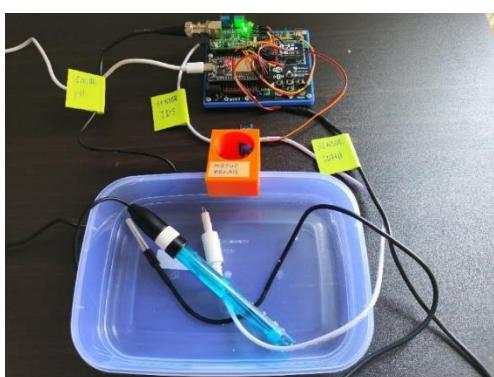
Halaman Profil memungkinkan pengguna untuk melihat informasi akun mereka secara lengkap, termasuk nama pengguna, alamat email, dan peran pengguna dalam sistem. Di bawah informasi akun, terdapat beberapa tombol yang berfungsi untuk membantu pengguna berinteraksi lebih lanjut dengan aplikasi, yaitu tombol “Bantuan & Dukungan” untuk mengirim pesan berupa email kepada pengembang jika terdapat kendala pada aplikasi, tombol “Kirim umpan balik” yang memungkinkan pengguna menyampaikan saran atau laporan kepada pengembang aplikasi, serta tombol “Tentang” untuk melihat informasi terkait aplikasi itu sendiri. Terakhir, terdapat tombol “Keluar”, yang berfungsi untuk mengakhiri sesi login dan mengamankan akun pengguna dari penggunaan tidak sah. Halaman profil dapat dilihat pada Gambar 13.



Gambar 13. Tampilan Halaman Profil

3.4. Penerapan Alat pada Tambak Udang

Berikut merupakan alat yang telah dirangkai berdasarkan rangkaian skematis dan melewati proses pengkodean. Alat yang telah dirangkai dapat dilihat pada Gambar 14.



Gambar 14. Alat yang telah dirangkai

Alat yang telah dirangkai kemudian diletakkan ke dalam wadah plastik agar terlindung dari air, sementara itu semua sensornya dikeluarkan agar bisa dimasukkan ke dalam air kolam. Wadah yang menampung alat selanjutnya diikat menggunakan tali agar tidak terbawa arus. Alat yang telah berhasil dipasang pada tambak udang dapat dilihat pada Gambar 15.



Gambar 15. Pemasangan Alat pada Tambak Udang

Selanjutnya dilakukan pengecekan untuk memastikan alat berfungsi sebagaimana mestinya dengan menggunakan alat ukur parameter tambak udang seperti pada Gambar 16.



Gambar 16. Perbandingan dengan Alat Ukur Parameter Tambak Udang

4. Kesimpulan

Penerapan jalur komunikasi REST API menunjukkan keberhasilan dalam menyediakan koneksi antara alat pemantauan tambak udang dengan aplikasi antarmuka. REST API mendukung transfer data yang stabil dan cepat dalam sistem pemantauan real-time, dengan rata-rata waktu respon berkisar antara 70–87 ms dan tingkat keberhasilan 100% tanpa packet loss selama pengujian menggunakan Apache JMeter. Sensor suhu menunjukkan akurasi rata-rata 93,3–96,0%, sedangkan sensor pH mencapai akurasi 93,7–96,4%.

Pengembangan antarmuka aplikasi menggunakan Flutter sebagai framework utama memberikan keunggulan dalam menciptakan antarmuka yang responsif, user-friendly, serta kompatibel dengan berbagai perangkat Android. Hasil ini menunjukkan bahwa sistem yang dikembangkan memiliki performa tinggi, stabilitas baik, serta potensi besar untuk diterapkan secara luas dalam pengelolaan tambak udang berbasis Internet of Things (IoT).

Daftar Rujukan

- [1] M. F. Anggoro, B. Yulianto, and S. Suryono, “Analisis Kadar TAN Terhadap Bobot Udang di Tambak Udang Mangrove Jembatan Api-Api, Kulonprogo,” *J. Mar. Res.*, vol. 13, no. 2, pp. 381–388, 2024, doi: 10.14710/jmr.v13i2.40094.
- [2] H. P. Ramadhan, C. Kartiko, and A. Prasetyadi, “Monitoring

- Kualitas Air Tambak Udang Menggunakan Metode Data Logging,” vol. 6, no. April, pp. 102–114, 2020.
- [3] A. N. Aprianto, A. Suganda Girsang, Y. Nugroho, and W. Kumala Putra D, “Analisis Kinerja Streaming RabbitMQ dan Nats untuk Komunikasi di Layanan Mikro,” <Https://Journal.Unipdu.Ac.Id/Index.Php/Teknologi/ArticleView/4498/1716>, vol. 14, no. 1, pp. 37–47, 2022, [Online]. Available: <https://journal.unipdu.ac.id/index.php/teknologi/article/view/4498/1716>
- [4] D. R. Pratama, B. Irmawati, and R. Robbani, “Pengembangan REST API SiAbang (Sistem Administrasi Pembangunan) Menggunakan Java,” *J. Begawe Teknol. Inf.*, vol. 4, no. 1, pp. 133–142, 2023, doi: 10.29303/jbegati.v4i1.978.
- [5] S. Ainah *et al.*, “Implementasi Framework Flutter Untuk Pengembangan Aplikasi Restoran Dengan Penerapan Api ChatGPT,” *J. Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 3802–3809, 2024.
- [6] S. Widyaningtyas and T. Wahyono, “IMPLEMENTASI REST API MENGGUNAKAN RETROFIT PADA APLIKASI MONITORING GROOMING BERBASIS ANDROID,” *IT-Explore J. Penerapan Teknol. Inf. dan Komun.*, vol. 3, no. 2, pp. 147–161, 2024.
- [7] M. F. Osama, I. Purnamasari, and R. Mayasari, “MULTIPLATFORM UNTUK MENDUKUNG LANGKAH EKONOMI DIGITAL (STUDI KASUS : PAWON IBU SNACKS KARAWANG),” vol. 7, no. 4, pp. 2877–2883, 2023.
- [8] Nelly Sofi and Riza Dharmawan, “Perancangan Aplikasi Bengkel Csm Berbasis Android Menggunakan Framework Flutter (Bahasa Dart),” *J. Tek. dan Sci.*, vol. 1, no. 2, pp. 53–64, 2022, doi: 10.56127/jts.v1i2.125.
- [9] R. A. Setyawan, “Penerapan Firebase Realtime Database Pada Aplikasi Catatan Harian Diabetes Melitus,” *J. Inform. Komputer, Bisnis dan Manaj.*, vol. 22, no. 1, pp. 1–9, 2024, doi: 10.61805/fahma.v22i1.102.
- [10] A. N. A. Subingat, N. Dening, A. Prafanto, and M. Taruk, “Implementasi Internet of Things Pada Sistem Pencarian Parkir Berbasis Mikrokontroler Node-MCU,” *J. Rekayasa Teknol. Inf.*, vol. 5, no. 2, p. 101, 2021, doi: 10.30872/jurti.v5i2.7070.
- [11] A. Sabril *et al.*, “Rancang bangun perangkat lunak antarmuka kendali mikrokontroler ESP826 dengan jaringan internet menggunakan flutter 3 . 0,” vol. 8698, pp. 27–34, 2023.
- [12] G. Alam, S. N. Rohman, and M. Y. Saputra, “Sistem Monitoring Dan Kontrol Inkubator Bibit Ikan Nila Dengan Implementasi Internet of Things Berbasis Mobile,” vol. 2, no. 3, pp. 629–637, 2024.
- [13] P. Gowda and A. N. Gowda, “Best Practices in REST API Design for Enhanced Scalability and Security,” *J. Artif. Intell. Mach. Learn. Data Sci.*, vol. 2, no. 1, pp. 827–830, 2024, doi: 10.51219/jaimld/priyanka-gowda/202.
- [14] H. Hussain, K. Khan, F. Farooqui, Q. Ali Arain, and I. Farah Siddiqui, “Comparative Study of Android Native and Flutter App Development,” *KSII 13th Int. Conf. Internet*, no. December, p. 2021, 2021, [Online]. Available: <https://www.researchgate.net/publication/361208165>
- [15] V. Y. P. Ardhana, M. T. Hidayat, M. Jannah, S. Sumiati, P. Rini, and N. Sari, “Implementasi RESTful API Pada Laravel dan Simulator IoT Wokwi Untuk Pengukuran Suhu dan Kelembaban Menggunakan Metode Waterfall,” *Arcitech J. Comput. Sci. Artif. Intell.*, vol. 3, no. 2, p. 93, 2023, doi: 10.29240/arctech.v3i2.9334.
- [16] K. A. Nugraha, “Efisiensi Pertukaran Data Client-Server menggunakan Web Socket pada Perangkat Berbasis Internet of Things,” *J. Edukasi dan Penelit. Inform.*, vol. 10, no. 1, p. 33, 2024, doi: 10.26418/jp.v10i1.73145.
- [17] S. Y. Damayanti, T. Andriyanto, and A. Ristiyawan, “Sistem Monitoring Kualitas Air Tambak Ikan Koi (*Cyprinus carpio*) Berbasis Teknologi of Things (IOT),” *Semin. Nas. Inov. Teknol.*, pp. 141–147, 2021.
- [18] I. N. W. R. S. Rizqi Samkayana, “Implementasi Firebase Realtime Database Pada Smart Cat Feeder Berbasis Mikrokontroler Esp32,” *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 3, 2025, doi: 10.23960/jitet.v13i3.7198.
- [19] N. Rahayu, “Pengembangan Aplikasi Mobile Untuk Pemantauan Kesehatan Mental Menggunakan Flutter,” *J. Komput.*, vol. 3, no. 1, pp. 25–30, 2024, doi: 10.70963/jk.v3i1.104.
- [20] N. A. Razaq and M. R. A. H, “Perancangan Sistem Kontrol Dan Monitoring Air Conditioner Ruangan Berbasis Internet Of Things,” vol. 10, 2024.