

Implementasi Firebase Pada Aplikasi Jemuran Pintar Berbasis IoT Dan Pengukuran Parameter QoS

1st Farhan Ramadhan Putra Hendratno
School of Electrical Engineering
Telkom University
Bandung, Indonesia
Farhanramadhanph@gmail.com

2nd Rita Purnamasari
School of Electrical Engineering
Telkom University
Bandung, Indonesia
ritapurnamasari@telkomuniversity.ac.id

3rd Fardan
School of Electrical Engineering
Telkom University
Bandung, Indonesia
fardanfn@telkomuniversity.ac.id

Cuaca yang tidak menentu di Indonesia seringkali menyulitkan aktivitas menjemur pakaian, terutama saat tidak ada penghuni di rumah. Untuk menjawab permasalahan tersebut, penelitian ini mengembangkan sistem jemuran pintar berbasis Internet of Things (IoT) yang mampu bekerja secara otomatis dan dapat dikendalikan melalui aplikasi mobile. Sistem ini menggunakan mikrokontroler ESP8266 yang terintegrasi dengan sensor cahaya (LDR) dan sensor hujan (raindrop), serta Firebase Realtime Database sebagai media komunikasi dan penyimpanan data secara real-time. Aplikasi mobile bernama Jepri dikembangkan menggunakan framework Flutter sebagai antarmuka pengguna. Pengujian dilakukan untuk mengevaluasi fungsi sistem serta kualitas layanan (Quality of Service/QoS) menggunakan parameter packet loss, delay, dan jitter. Hasil pengujian menunjukkan bahwa sistem berjalan secara akurat dan sinkron antara data sensor, Firebase, dan aplikasi, dengan nilai packet loss 0%, serta delay dan jitter yang rendah. Selain itu, sistem telah memenuhi aspek-aspek QoS berdasarkan standar ITU-T T.1010, seperti konektivitas, skalabilitas, dan kecepatan respon. Dengan demikian, sistem ini terbukti andal dan efisien sebagai solusi otomatis rumah tangga yang adaptif terhadap perubahan cuaca.

Kata kunci— Internet of Things, Jemuran Pintar, ESP8266, Flutter, ITU-T Y.2060

I. PENDAHULUAN

Indonesia sebagai negara tropis memiliki intensitas hujan tinggi dan sinar matahari yang melimpah, namun kondisi cuaca yang tidak menentu menyulitkan aktivitas menjemur pakaian, terutama jika tidak ada penghuni rumah di tempat tinggal. Ketika hujan turun secara tiba-tiba, pakaian yang sedang dijemur berisiko basah, sehingga menghambat proses pengeringan dan menambah beban pekerjaan.

Sebagai solusi, dibutuhkan sistem jemuran pintar yang mampu memantau dan mengontrol jemuran secara *real-time*. Teknologi *Internet of Things* (IoT) memberikan pendekatan untuk mengatasi masalah ini dengan memanfaatkan *light dependent resistor* (LDR) dan sensor *raindrop*. Dalam IoT, perangkat seperti sensor, kendaraan, peralatan rumah tangga, dan mesin industri saling terhubung dan dapat berkomunikasi satu sama lain secara otomatis untuk melakukan berbagai fungsi, seperti pengumpulan data, analisis, atau pengambilan keputusan secara mandiri [1]. Sensor LDR dan sensor

raindrop akan bekerja secara simultan untuk memantau kondisi lingkungan secara *realtime*. Kemudian Nilai nilai yang terukur oleh sensor akan dikirimkan ke *realtime database*, dimana *realtime database* yang digunakan adalah google Firebase. Firebase memungkinkan data sensor disimpan dan diakses secara cepat dan sinkron oleh aplikasi mobile. Melalui integrasi ini, aplikasi bernama jepri yang dirancang menggunakan *framework* Flutter, dapat menampilkan kondisi cuaca secara aktual yang memungkinkan pengguna untuk memberikan perintah manual dan memonitor jemuran secara efisien dan dari jarak jauh.

Untuk memastikan sistem jemuran pintar ini bekerja secara optimal, dibutuhkan pengukuran performa komunikasi data antara mikrokontroler dan aplikasi melalui Firebase menggunakan parameter *Quality of Service* (QoS),

II. KAJIAN TEORI

Pengembangan aplikasi Jepri memerlukan pemahaman mendalam guna memastikan sistem dapat berjalan secara lancar dalam mendukung otomatisasi jemuran pintar. Kajian teori ini difokuskan pada tiga komponen utama yang menjadi landasan dasar yaitu Firebase, *Quality of Service* (QoS), dan flutter. Firebase sebagai tempat penyimpanan data sensor, QoS untuk memastikan kualitas performa sistem dalam pengiriman dan penerimaan data secara *realtime*, dan Flutter sebagai *framework* yang digunakan dalam mengembangkan aplikasi Jepri.

A. Firebase

Firebase adalah platform pengembangan aplikasi. Didirikan pada tahun 2011, Firebase kemudian diakuisisi oleh Google pada tahun 2014. Kini, Firebase menjadi solusi utama untuk pengembangan web dan seluler. Firebase menawarkan berbagai produk untuk membangun, merilis, memantau, dan mengoptimalkan aplikasi. Beberapa layanan yang disediakan oleh Firebase meliputi Cloud Firestore, Realtime Database, Cloud Storage, Cloud Functions, Cloud Messaging, Autentikasi, Hosting, Crashlytics, dan Pengujian A/B. [2]

B. *Quality of Service* (QoS)

Quality of Service (QoS) ialah sebuah metode yang digunakan untuk mengetahui kualitas dari jaringan internet, serta usaha untuk dapat mendefinisikan sifat dan karakteristik dari layanan tersebut. Quality of Service (QoS) dalam sebuah jaringan dilihat dari tingkat kecepatan dan kehandalan disegala macam data dalam suatu komunikasi. [3]

C. Flutter

Flutter adalah suatu framework yang bersifat open source, dikembangkan oleh Google untuk membangun antar muka (user interface). Flutter dapat dikembangkan untuk perangkat yang memiliki sistem operasi Android dan iOS. [4]

D. *Internet of Things* (IoT)

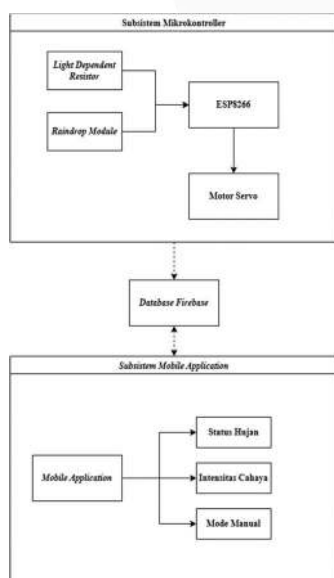
aplikasi IoT digunakan untuk mempermudah kehidupan manusia dalam melakukan monitoring terhadap suatu obyek atau benda secara otomatis. IoT pada selain bidang pertanian telah banyak diimplementasikan dalam berbagai hal. [5].

III. METODE

Metode penelitian ini bertujuan untuk menguji dan mengembangkan aplikasi sistem jemuran pintar yang terintegrasi dengan Firebase sebagai penyedia layanan komunikasi dan penyimpanan data secara *realtime*. Firebase memiliki peran penting dalam pengembangan aplikasi sistem jemuran pintar. Selain itu, untuk mengevaluasi performa komunikasi data dalam sistem, pengukuran *Quality of Service* (QoS) menjadi pilihan yang tepat. Parameter QoS seperti delay, jitter, dan packet loss digunakan sebagai indikator untuk menilai kualitas sistem dalam mengirimkan dan menyimpan data secara *realtime*. *Framework* Flutter digunakan dalam pengembangan aplikasi jemuran pintar dengan kemampuan dalam membangun antarmuka dengan layanan Firebase.

A. Arsitektur Sistem

Arsitektur sistem jemuran pintar terdiri tiga bagian utama, yaitu subsistem mikrokontroler, Firebase, dan subsistem aplikasi mobile.



GAMBAR 1
Arsitektur sistem

Sistem jemuran pintar ini terdiri dari tiga komponen utama, yaitu subsistem mikrokontroler, Firebase, dan Aplikasi mobile. Mikrokontroler berfungsi sebagai pengendali utama yang terhubung dengan dua sensor, yaitu LDR untuk mengukur tingkat pencahayaan dan sensor hujan untuk mengidentifikasi adanya hujan. Sensor mengirimkan data yang diproses oleh ESP8266 dan mengatur motor servo. Data tersebut kemudian dikirim ke Firebase, yang berfungsi sebagai tempat penyimpanan dan komunikasi data secara *realtime*. Firebase juga mendapatkan *input* dari pengguna melalui aplikasi jepri jika pengguna ingin memilih untuk mengontrol jemuran secara manual. Aplikasi mobile yang dibuat dengan Flutter menunjukkan keadaan cuaca dan memberikan pengaturan manual jemuran secara *realtime*. Alur komunikasi di antara komponen ini divisualisasikan pada gambar 3.1.

B. Implementasi Firebase

Firebase berfungsi sebagai media komunikasi dua arah antara ESP8266 dan aplikasi mobile. Firebase menerima data sensor yang dikirim melalui mikrokontroler dan menyimpannya secara *realtime* dengan Firebase realtime database, sehingga data tersebut dapat langsung diakses melalui aplikasi mobile. Pada Firebase ditentukan kondisi yang terintegrasi dengan mikrokontroler yang memungkinkan data diaplikasi sesuai dengan status cuaca pada sistem jemuran pintar, dan juga posisi jemuran.

C. Metode Pengujian *Quality of Service* (QoS)

Pengujian performa sistem dilakukan dengan metode pengukuran Quality of Service (QoS) untuk mengetahui seberapa andal sistem dalam mengirimkan dan menerima data secara real-time. Tiga parameter utama yang diuji adalah:

1. Delay

Mengukur rata-rata waktu yang dibutuhkan dari saat data dikirim oleh ESP8266 ke Firebase hingga data diterima dan ditampilkan di aplikasi. Perhitungan delay menggunakan persamaan (1), dan Tabel 3.1.

$$Ad = \frac{T_d}{P_r} \quad (1)$$

Ket:

Ad : Rata – rata delay (s)
 T_d : Total delay
 P_r : Paket yang diterima

TABEL 1
Standard ITU-T G.1010 untuk delay

Kategori	Delay (ms)	Indeks
Sangat Bagus	<150 ms	4
Bagus	150 ms s/d 300 ms	3
Sedang	300 ms s/d 450ms	2
Jelek	>450ms	1

2. Jitter

Mengukur ketidakteraturan waktu delay antar paket data, yang dapat mempengaruhi konsistensi sistem. Perhitungan jitter menggunakan persamaan (2) dan Tabel 3.2

$$J = \frac{T_{dv}}{P_r} \quad (2)$$

Ket:

J : Jitter (ms)

T_{dv} : Total variasi delay (ms)

TABEL 2 S
tandard ITU-T G.1010 untuk jitter

Kategori	Jitter (ms)	Indeks
Sangat Bagus	<150 ms	4
Bagus	0 ms s/d 75ms	3
Sedang	75 ms s/d 125ms	2
Jelek	125 ms s/d 225ms	1

3. Packet Loss

Mengukur berapa banyak data yang hilang selama proses transmisi dari perangkat ke Firebase atau sebaliknya. Perhitungan Packet loss menggunakan Persamaan (3) dan Tabel 3.3

$$Pl = \frac{(P_s - P_r)}{T_{ps}} \times 100\% \quad (3)$$

Ket:

Pl : Packet loss (%)

P_s : Paket data dikirim

P_r : Paket data diterima

T_{ps} : Total paket yang terkirim

TABEL 3
Standard ITU-T G.1010 untuk packet loss

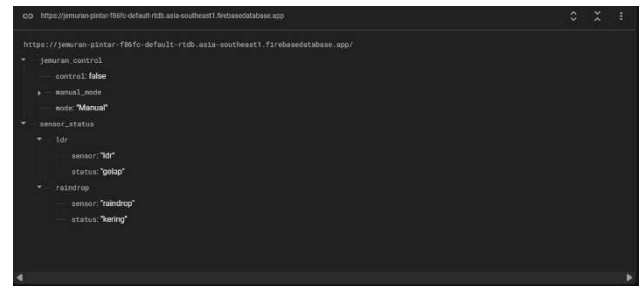
Kategori	Packet Loss (%)	Indeks
Sangat Bagus	0%	4
Bagus	1% s/d 3%	3
Sedang	3% s/d 15%	2
Jelek	>25%	1

IV. HASIL DAN PEMBAHASAN

Pembahasan terhadap performa dan fungsionalitas sistem berdasarkan pengujian. Fokus utama dalam pengujian adalah memastikan bahwa seluruh komponen dari sistem, mulai dari Firebase dan aplikasi bekerja secara terintegrasi dan lancar. Selain itu pengukuran kualitas performa sistem dengan parameter QoS seperti delay, jitter, dan packet loss, beserta aspek QoS. Hasil pengujian dianalisis untuk menilai keandalan sistem dalam merespons perubahan kondisi cuaca secara *realtime* serta efektivitas kendali jemuran melalui aplikasi.

A. Pengujian Firebase

Firebase digunakan dalam sistem jemuran pintar sebagai *platform* untuk menyimpan data sensor secara *realtime* dan sebagai media yang menghubungkan antara perangkat keras dan aplikasi mobile. Untuk memastikan proses pengiriman dan penerimaan data dari mikrokontroler ke Firebase, serta dari Firebase ke aplikasi berjalan dengan baik, dilakukan beberapa pengujian yang difokuskan pada pencatatan data sensor di Firebase dan kesesuaian data pada aplikasi jepri.



GAMBAR 2

Gambar 4.1 menunjukkan struktur data pada Firebase Realtime Database yang digunakan dalam sistem jemuran pintar. Bagian *sensor_status*, terdapat dua jenis data sensor, yaitu LDR dan *raindrop*. Sensor LDR memiliki status “gelap” yang menandakan bahwa intensitas cahaya pada alat terdeteksi rendah, sedangkan sensor *raindrop* memiliki status “kering” yang menunjukkan tidak ada curah hujan yang terdeteksi. Struktur data ini menunjukkan bahwa Firebase berhasil dalam merekam kondisi sensor secara *realtime* dan menyediakannya untuk dibaca oleh aplikasi.

Pengujian juga dilakukan dengan sensor hujan selama 2 hari pada waktu yang berbeda untuk memastikan status basah dan kering yang terdeteksi sensor dikirimkan dengan benar ke Firebase dan ditampilkan secara akurat pada aplikasi jepri. Tabel 4.1 menunjukkan hasil pengujian berdasarkan sinyal digital yang dihasilkan sensor:

TABEL 4
Hasil Pengujian Sinyal Digital Sensor Hujan

Tanggal	Waktu	Cuaca BMKG	Sinyal Digital	Status Sensor	Status di Aplikasi
09/07/2025	06.00 WIB	Hujan Ringan	High	Basah	Status Hujan: “Basah”
09/07/2025	10.00 WIB	Cerah Berawan	Low	Kering	Status Hujan: “Kering”
09/07/2025	13.00 WIB	Hujan Ringan	High	Basah	Status Hujan: “Basah”
09/07/2025	16.00 WIB	Berawan	Low	Kering	Status Hujan: “Kering”
09/07/2025	19.00 WIB	Cerah Berawan	Low	Kering	Status Hujan: “Kering”
09/07/2025	21.00 WIB	Cerah	Low	Kering	Status Hujan: “Kering”
09/07/2025	23.00 WIB	Cerah	Low	Kering	Status Hujan: “Kering”
10/07/2025	05.00 WIB	Cerah Berawan	Low	Kering	Status Hujan: “Kering”
10/07/2025	07.00 WIB	Cerah Berawan	Low	Kering	Status Hujan: “Kering”
10/07/2025	10.00 WIB	Cerah	Low	Kering	Status Hujan: “Kering”
10/07/2025	13.00 WIB	Hujan Petir	High	Basah	Status Hujan: “Basah”

10/07/2025	16.00 WIB	Hujan Ringan	High	Basah	Status Hujan: "Basah"
10/07/2025	19.00 WIB	Cerah	Low	Kering	Status Hujan: "Kering"
10/07/2025	22.00 WIB	Cerah	Low	Kering	Status Hujan: "Kering"

Dari hasil pengujian, terlihat bahwa setiap perubahan sinyal digital dari sensor berhasil direkam oleh Firebase. Tidak ditemukan perbedaan antara status yang terdeteksi oleh sensor dengan status yang ditampilkan oleh aplikasi, yang membuktikan bahwa Firebase mampu melakukan sinkronisasi data secara *realtime*.

Pengujian selanjutnya dilakukan pada sensor cahaya (LDR) untuk mengamati hubungan antara intensitas cahaya, nilai ADC yang terbaca oleh ESP8266, dan status jemuran pada aplikasi. Data yang dikirim ke Firebase mencakup nilai ADC serta status jemuran (Masuk/Keluar), yang kemudian ditampilkan pada aplikasi. Tabel 4.2 menunjukkan hasil pengujian sensor LDR.

TABEL 5
Hasil pengujian Sensor LDR

Hari Ke-	Jam	Kondisi Cuaca BMKG	Nilai ADC (bit)	Nilai Light Meter (lux)	Status Jemuran di Aplikasi
1	06.00 WIB	Cerah	191	6009	Keluar
	09.00 WIB	Cerah	34	33754	Keluar
	12.00 WIB	Cerah	25	45906	Keluar
	15.00 WIB	Cerah Berawan	122	9409	Keluar
	18.00 WIB	Hujan	882	23	Masuk
2	06.00 WIB	Cerah	242	4744	Keluar
	09.00 WIB	Cerah	31	37032	Keluar
	12.00 WIB	Cerah	27	42519	Keluar
	15.00 WIB	Berawan	172	6676	Keluar
	18.00 WIB	Hujan	850	30	Masuk

Hasil pengujian menunjukkan bahwa perubahan nilai cahaya yang terdeteksi oleh sensor berhasil dikirim dan disimpan serta dikirimkan ke aplikasi dengan baik oleh Firebase. Ketika LDR threshold >242, maka status jemuran secara otomatis berubah menjadi "Masuk" dan sebaliknya. Ketika LDR threshold <242, jemuran berada dalam status "Keluar". Hal ini menunjukkan Firebase menjalankan fungsi secara optimal sebagai jembatan komunikasi antara perangkat dan aplikasi.

B. Pengujian QoS

Pengujian Quality of Service (QoS) dilakukan untuk mengukur performa komunikasi data dalam sistem jemuran pintar menggunakan aplikasi Wireshark. Parameter utama yang diuji meliputi packet loss, delay, dan jitter, yang mengacu pada standar ITU-T G 1010, standar ITU-T Y.2060, sebagai acuan kualitas layanan dalam implementasi IoT. Sistem IoT yang baik harus mampu memenuhi berbagai aspek seperti konektivitas, skalabilitas, respons cepat

terhadap data penting, dan seterusnya. Tabel 4.4 dan 4.5 menunjukkan hasil pengujian dengan standar ITU-T G.1010 dengan sensor yang berbeda.

TABEL 6
Pengujian QoS pada sensor hujan

Packet Loss	Delay	Jitter
= 0% [Sangat Bagus]	= 69,4132 ms [Sangat Bagus]	= 67,5039 ms [Sangat Bagus]

TABEL 7
Pengujian QoS pada sensor LDR

Packet Loss	Delay	Jitter
= 0% [Sangat Bagus]	= 93,0875 ms [Sangat Bagus]	= 93,0864 ms [Sangat Bagus]

Pada pengujian sensor hujan, diperoleh nilai *packet loss* sebesar 0%, *delay* sebesar 69,4132 ms, dan *jitter* sebesar 67,5039 ms, ini menunjukkan bahwa sistem berjalan secara stabil dan cepat. Sedangkan pada pengujian sensor LDR, menunjukkan nilai *packet loss* sebesar 0%, *delay* sebesar 93,0875 ms, dan *jitter* sebesar 93,0864 ms, seperti ditampilkan pada Tabel 4.3. Walaupun *delay* sedikit lebih tinggi dibanding sensor hujan, nilainya masih berada pada kategori sangat baik. Adapun Tabel 4.5 berupa pengujian standar ITU-T Y.2060 terhadap penggunaan IoT

TABEL 8
Standar ITU-T Y.2060 penggunaan IoT

Aspek QoS	Penjelasan	Sistem Jemuran Pintar
Konektivitas	Perangkat harus bisa terhubung ke jaringan.	ESP8266 tersambung ke WiFi untuk kirim data ke Firebase.
Perangkat Berbeda Bisa Terhubung	Berbagai jenis perangkat harus bisa saling bekerja sama.	Sensor LDR, hujan, dan motor servo bekerja bersama dalam satu sistem.
Skalabilitas	Sistem harus bisa bekerja meskipun jumlah perangkat bertambah.	Beberapa unit jemuran otomatis dapat ditambahkan ke dalam sistem tanpa mengganggu kinerja atau kestabilan sistem secara keseluruhan.
Data Harus Cepat Direspon	Data penting seperti hujan harus segera ditindaklanjuti.	Data hujan dikirim dan jemuran langsung menutup otomatis.
Mudah Dipasang	Perangkat bisa langsung digunakan saat dipasang.	Sensor otomatis dikenali saat sistem menyala.
Bisa Dipantau dan Diatur	Sistem bisa dikendalikan dari jarak jauh.	Aplikasi digunakan untuk memantau dan mengatur posisi jemuran.
Keamanan	Data dan akses harus terlindungi.	Data pengguna disimpan secara aman dan terlindungi melalui Firebase Realtime Database.
Penyimpanan Data	Sistem butuh tempat menyimpan dan proses data.	Firestore menyimpan data sensor cahaya dan hujan.
Pantau Kinerja	Sistem bisa mengukur apakah koneksi dan respons berjalan baik.	Waktu kirim dan respon servo bisa diukur untuk melihat delay.

V. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian Firebase dan QoS, sistem jemuran pintar yang terintegrasi Firebase berhasil memenuhi fungsinya sebagai solusi *realtime* dalam mengontrol jemuran berdasarkan kondisi cuaca. Integrasi antara *hardware*, Firebase Realtime Database, serta aplikasi jepri berbasis Flutter berjalan dengan baik, ditunjukkan oleh akurasi data yang dikirim dan ditampilkan pada aplikasi. Firebase berfungsi optimal sebagai media komunikasi data antara perangkat dan aplikasi. Hasil pengujian QoS menggunakan Wireshark menunjukkan performa yang baik dengan semua parameter “Sangat Bagus” pada kedua sensor. Dengan demikian, sistem ini terbukti handal, responsive, dan layak diterapkan sebagai sistem otomatisasi rumah tangga yang efisien dan adaptif terhadap perubahan cuaca.[6] [7] [8] [9] [10] [11]

REFERENSI

- [1] N. R. Laboy, P. A. Vahlevi, T. Sutabri, M. Rizki, U. Bina, and D. Palembang, “Analisis Penerapan Internet of Things (Iot) Dalam Smart Home System,” *J. Ilmu Tek.*, vol. 1, no. 2, pp. 283–285, 2024, [Online]. Available: <https://doi.org/10.62017/tektunik>
- [2] A. Zewdie Ayezabu, “Supabase vs Firebase: Evaluation of performance and development of Progressive Web Apps,” *Metropolia*, no. May, pp. 1–58, 2022.
- [3] R. F. Bari, A. Solehudin, and N. Heryana, “Analisis Quality of Service (QoS) Jaringan Internet Berbasis Wireless LocalArea Network pada Layanan Indihome,” *J. Ilm. Wahana Pendidik.*, vol. 8, no. 10, pp. 320–335, 2022, [Online]. Available: <https://doi.org/10.5281/zenodo.6820184>
- [4] P. Rachmat Setiawan, Syefriani, and R. Wandri, “Inovasi Teknologi Melalui Pembelajaran Flutter Menyongsong Era Aplikasi Mobile,” *J. Sains dan Teknol. dalam Pengabd. Masy.*, vol. 2, no. 1, pp. 1–9, 2025, [Online]. Available: <https://jurnal-rjb.com/index.php/jstpm/article/view/29>
- [5] I. D. Irawati, D. N. Ramadan, S. Hadiyoso, R. Purnamasari, G. Budiman, N. R. Fachrurrozi, Yudiansyah, S. N. Y. Yutia, S. H. Y. Tyas, and S. Z. Fajriyah, “Workshop Implementasi Sistem Monitoring dan Kendali Kualitas Air pada Media Tanam Aquaponik Menggunakan Gawai Pintar,” *Jurnal Abdi Insani*, vol. 10, no. 1, pp. 44–53, Mar. 2023. [Online]. Available: <https://doi.org/10.29303/abdiinsani.v10i1.805>.
- [6] A. T. Gaikwad, “Firebase - Overview and usage,” *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 3(12), no. August, pp. 1178–1183, 2022.
- [7] N. F. Za’bah and A. Mohamad, “Mobile Application Monitoring System Using Internet of Things (IOT) and Firebase,” *J. Inf. Syst. Digit. Technol.*, vol. 3, no. 1, pp. 16–30, 2021, doi: 10.31436/jisdt.v3i1.168.
- [8] A. Saputri and A. M. Hirzan, “Aplikasi Manajemen Inventori Berbasis Mobile Menggunakan Flutter Dan Firebase Realtime Database,” *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, pp. 1586–1592, 2024, doi: 10.23960/jitet.v12i3.4324.
- [9] G. R. Payara and R. Tanone, “Penerapan Firebase Realtime Database Pada Prototype Aplikasi Pemesanan Makanan Berbasis Android,” *J. Tek. Inform. dan Sist. Inf.*, vol. 4, no. 4, pp. 397–406, 2018, [Online]. Available: <https://journal.maranatha.edu/index.php/jutisi/article/view/1476>
- [10] D. Enda, M. A. Subandri, and Supria, “Analisis Qos (Quality of Service) Sistem Monitoring Pintar Mitigasi Penularan Covid-19 Berbasis Iot,” *J. Inform. Polinema*, vol. 8, no. 1, pp. 39–46, 2021, doi: 10.33795/jip.v8i1.705.
- [11] R. Andrianto and M. H. Munandar, “Aplikasi E-Commerce Penjualan Pakaian Berbasis Android Menggunakan Firebase Realtime Database,” *J. Comput. Sci. Inf. Technol.*, vol. 3, no. 1, pp. 20–29, 2022, [Online]. Available: <https://jurnal.ulb.ac.id/index.php/JCoInT/article/view/2478>