

Perancangan Dashboard Monitoring Berbasis Android Pada Sistem Medis Troli Otonom Terintegrasi

Rafli Gymnastiar^{1*)}, Dian Budhi Santoso², Indri Purwita Sary³, Jordan Zefanya Purba⁴, Galih Setyaji⁵

^{1,2,3,4,5}Jurusan Teknik Elektro, Universitas Singaperbangsa Karawang, Kabupaten Karawang, Jawa Barat, Jalan H.S. Ronggowaluyo, Desa Puseurjaya, Kecamatan Telukjambe Timur.

*Penulis Korespondensi, e-mail: 2210631160016@student.unsika.ac.id

Received:13/05/2026

Revised:30/06/2026

Accepted:01/07/2026

ABSTRAK

Perkembangan teknologi digital mendorong pemanfaatan aplikasi monitoring berbasis Android untuk mendukung sistem kesehatan yang lebih efisien. Sistem Medis Troli Otonom Terintegrasi (SIMETRI) memerlukan dashboard yang mampu mendukung proses monitoring dan controlling secara mudah dan fleksibel. Penelitian ini bertujuan merancang dashboard berbasis Android pada SIMETRI menggunakan HTML, CSS, dan JavaScript yang dikembangkan melalui Visual Studio Code serta dikonversi menjadi aplikasi Android menggunakan AppsGeyser. Metode penelitian meliputi perancangan sistem, pengembangan dashboard, serta pengujian menggunakan Black Box Testing dan White Box Testing. Hasil penelitian menunjukkan bahwa aplikasi berhasil dikembangkan dengan fitur Login, Dashboard, Monitoring, dan Controlling yang dapat berjalan dengan baik pada perangkat Android. Dashboard dirancang untuk mendukung integrasi dengan Raspberry Pi dan Firebase Realtime Database sehingga memungkinkan pertukaran data monitoring dan controlling secara real-time. Hasil pengujian menunjukkan seluruh fungsi aplikasi berjalan sesuai dengan rancangan sehingga dashboard berpotensi mendukung operasional SIMETRI secara lebih efektif dan terintegrasi.

Kata Kunci: Dashboard, Pengamatan, SIMETRI, AppsGeyser

ABSTRACT

The development of digital technology has encouraged the use of Android-based monitoring applications to support more efficient healthcare systems. The Integrated Autonomous Medical Trolley System (SIMETRI) requires a dashboard capable of supporting both monitoring and controlling processes in an easy and flexible manner. This study aims to design an Android-based dashboard for SIMETRI using HTML, CSS, and JavaScript developed through Visual Studio Code and converted into an Android application using AppsGeyser. The research methods include system design, dashboard development, and testing using Black Box Testing and White Box Testing. The results show that the application was successfully developed with Login, Dashboard, Monitoring, and Controlling features that function properly on Android devices. The dashboard is designed to support integration with Raspberry Pi and Firebase Realtime Database, enabling real-time monitoring and controlling data exchange. The testing results indicate that all application functions operate according to the design, demonstrating the potential of the dashboard to support SIMETRI operations in a more effective and integrated manner.

Keywords: Dashboard, Monitoring, SIMETRI, AppsGeyser.

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi telah mendorong transformasi digital pada berbagai sektor, termasuk sektor kesehatan. Pemanfaatan teknologi digital dalam layanan kesehatan bertujuan untuk meningkatkan efisiensi operasional, mempercepat penyampaian informasi, serta mendukung pengambilan keputusan secara lebih efektif. Salah satu teknologi yang banyak dikembangkan

p-ISSN: 2356-0533; e-ISSN: 2355-9195



adalah sistem *Monitoring* berbasis *Internet of Things (IoT)* dan perangkat bergerak (*mobile device*) yang memungkinkan proses pemantauan dilakukan secara jarak jauh melalui jaringan internet [1].

Di lingkungan rumah sakit, kebutuhan akan sistem *monitoring* semakin meningkat seiring berkembangnya penggunaan perangkat otomatis dan sistem cerdas. Berbagai aktivitas operasional, seperti pemantauan kondisi pasien, distribusi obat, serta pengelolaan logistik medis, memerlukan penyajian informasi yang cepat dan mudah diakses oleh tenaga kesehatan. Oleh karena itu, diperlukan suatu media *monitoring* yang mampu menampilkan data operasional secara terintegrasi melalui antarmuka yang sederhana dan mudah digunakan [2].

Perangkat berbasis *Android* menjadi salah satu *platform* yang banyak digunakan dalam pengembangan sistem *Monitoring* karena memiliki tingkat penggunaan yang tinggi, fleksibilitas implementasi, serta kemudahan integrasi dengan teknologi berbasis *web* dan *IoT* [2]. Penelitian yang dilakukan oleh Priyulida dkk. menunjukkan bahwa aplikasi *monitoring* berbasis *Android* mampu menampilkan data hasil pemantauan secara real-time sehingga memudahkan pengguna dalam melakukan pengawasan terhadap kondisi sistem yang dipantau [3]. Selain itu, pengembangan aplikasi *monitoring* berbasis *Android* juga telah diterapkan pada berbagai bidang, seperti *monitoring* kesehatan, *monitoring* energi listrik, serta *monitoring* aktivitas pengguna dengan hasil yang menunjukkan peningkatan efektivitas proses pemantauan [4].

Sistem Medis Troli Otonom Terintegrasi (SIMETRI) merupakan suatu pengembangan troli medis otonom yang dirancang untuk membantu distribusi logistik medis di lingkungan rumah sakit. Agar operasional SIMETRI dapat dipantau dengan baik, diperlukan subsistem *monitoring* yang mampu menyajikan informasi kondisi sistem melalui perangkat bergerak. *Dashboard monitoring* menjadi komponen penting karena berfungsi sebagai media visualisasi informasi yang memungkinkan pengguna memperoleh gambaran kondisi sistem secara cepat dan mudah dipahami [5]. Dengan adanya *dashboard monitoring*, operator dapat melakukan pemantauan terhadap status operasional troli tanpa harus berinteraksi langsung dengan perangkat [6].

Sebagian besar pengembangan sistem *monitoring* berbasis *Android* masih berfokus pada pemantauan kondisi kesehatan pasien dan perangkat *IoT* umum [7]. Penelitian yang secara khusus membahas perancangan *dashboard monitoring* pada sistem troli medis otonom masih relatif terbatas. Oleh karena itu, penelitian ini bertujuan merancang *dashboard monitoring* berbasis *Android* pada Sistem Medis Troli Otonom SIMETRI. *Dashboard* dikembangkan menggunakan teknologi *web* berbasis *HTML*, *CSS*, dan *JavaScript* yang kemudian dikonversi menjadi aplikasi *Android* [8]. Hasil penelitian ini diharapkan dapat menjadi dasar pengembangan subsistem *monitoring* SIMETRI serta mendukung implementasi teknologi otomasi di lingkungan rumah sakit.

Berdasarkan penelitian-penelitian sebelumnya, sebagian besar *dashboard monitoring* berbasis *Android* dikembangkan untuk pemantauan perangkat *IoT* umum, kondisi kesehatan pasien, maupun sistem *monitoring* jaringan. Namun, penelitian yang secara khusus membahas perancangan dashboard monitoring pada sistem medis troli otonom masih sangat terbatas. Sistem SIMETRI memiliki karakteristik yang berbeda karena mengintegrasikan fungsi *navigasi otonom*, *monitoring* kondisi baterai, pemantauan posisi troli, serta komunikasi data secara terintegrasi dalam satu platform. Oleh karena itu, penelitian ini menawarkan kontribusi berupa perancangan *dashboard monitoring* yang dirancang khusus untuk mendukung operasional Sistem Medis Troli Otonom Terintegrasi (SIMETRI), sehingga informasi kondisi sistem dapat diakses secara lebih mudah melalui perangkat Android.

2. METODE PENELITIAN

Pengujian sistem dilakukan untuk memastikan seluruh fungsi pada *Dashboard Monitoring* berbasis *Android* berjalan sesuai dengan rancangan. Metode pengujian yang digunakan dalam penelitian ini adalah *Black Box Testing* dan *White Box Testing* [9]. *Black Box Testing* digunakan untuk menguji fungsi aplikasi dari sisi pengguna, sedangkan *White Box Testing* digunakan untuk menganalisis alur logika program pada fitur utama aplikasi [10].



2.1 *Black Box Testing*

Black Box Testing merupakan metode pengujian yang berfokus pada fungsi sistem tanpa memperhatikan struktur kode program yang digunakan. Pengujian dilakukan dengan memberikan input tertentu dan mengamati output yang dihasilkan oleh sistem. Pada penelitian ini, pengujian dilakukan pada fitur-fitur utama Dashboard Monitoring SIMETRI yang meliputi Login, navigasi menu, Dashboard Monitoring, dan instalasi aplikasi Android.

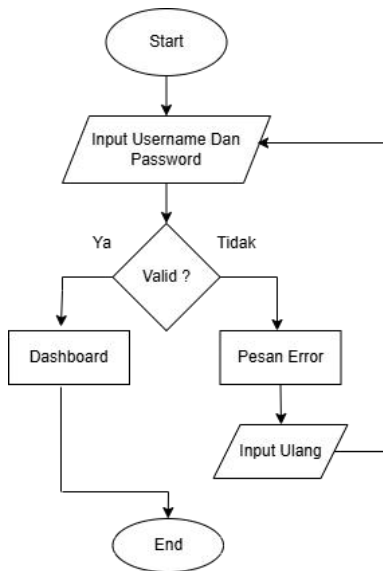
TABEL I : Pengujian Black Box

No	Fitur	Pengujian	Hasil	Status
1	<i>Login</i>	Username dan password benar	Menampilkan <i>Dashboard</i>	Berhasil
2	<i>Login</i>	Username atau password salah	Menampilkan pesan kesalahan	Berhasil
3	<i>Dashboard</i>	Membuka halaman <i>Dashboard</i>	<i>Dashboard</i> Tampil	Berhasil
4	<i>Monitoring</i>	Membuka halaman <i>Monitoring</i>	Data <i>Monitoring</i> Tampil	Berhasil
5	<i>Controlling</i>	Membuka halaman <i>Controlling</i>	Menu <i>Controlling</i> tampil	Berhasil
6	<i>Logout</i>	Menekan tombol <i>Logout</i>	Kembali ke halaman <i>Login</i>	Berhasil

2.2 *White Box Testing*

White Box Testing dilakukan untuk menganalisis struktur logika program yang digunakan pada fitur utama aplikasi. Pengujian ini berfokus pada alur proses yang terjadi di dalam program untuk memastikan setiap jalur eksekusi dapat berjalan dengan benar. Pada penelitian ini, contoh *White Box Testing* dilakukan pada fitur *Login* karena merupakan proses utama sebelum pengguna dapat mengakses *Dashboard Monitoring*.





Gambar 1. Flowchart Login

TABEL II : Keterangan Node	
Node	Keterangan
1	Start
2	Input Username dan Password
3	Validasi Data
4	Keputusan Valid/Tidak
5	Dashboard
6	Pesan Error
7	End

Jumlah *Node* yang terdapat pada *flowchart* adalah 7 *Node*, dan jumlah *Edge Flowchart* adalah 7. *Edge* merupakan setiap alur atau panah yang menghubungkan satu *Node* dengan *Node* lainnya.

2.2.1 Perhitungan *Cyclomatic Complexity*

Cyclomatic Complexity digunakan untuk menentukan jumlah jalur independen yang terdapat pada program.

Rumus yang digunakan adalah:

$V(G) = E - N + 2$

Keterangan:

$E = \text{Jumlah edge} = 7$

$N = \text{Jumlah Node} = 7$

Maka:

$V(G) = 7 - 7 + 2 = 2$



Berdasarkan hasil perhitungan diperoleh nilai Cyclomatic Complexity sebesar 2. Nilai tersebut menunjukkan bahwa terdapat 2 jalur independen yang harus diuji untuk memastikan seluruh logika program berjalan dengan baik. Jalur Independen (Independent Path) :

a. Jalur 1 (*Login Berhasil*)

1 → 2 → 3 → 4 → 7

Pada jalur ini pengguna memasukkan username dan password yang valid sehingga sistem mengarahkan pengguna menuju halaman *Dashboard* dan proses berakhir.

b. Jalur 2 (*Login Gagal dan Input Ulang*)

1 → 2 → 3 → 5 → 6 → 2 → 3 → 4 → 7

Pada jalur ini pengguna memasukkan data *Login* yang tidak valid sehingga sistem menampilkan pesan kesalahan. Pengguna kemudian melakukan input ulang hingga data yang dimasukkan valid dan berhasil masuk ke halaman *Dashboard*.

Hasil *White Box Testing* menunjukkan bahwa seluruh jalur independen pada fitur *Login* dapat dieksekusi dengan baik tanpa ditemukan kesalahan logika program. Dengan demikian, proses autentikasi pada *Dashboard Monitoring* SIMETRI telah berjalan sesuai dengan rancangan sistem dan siap digunakan sebagai bagian dari aplikasi *Monitoring* berbasis *Android*.

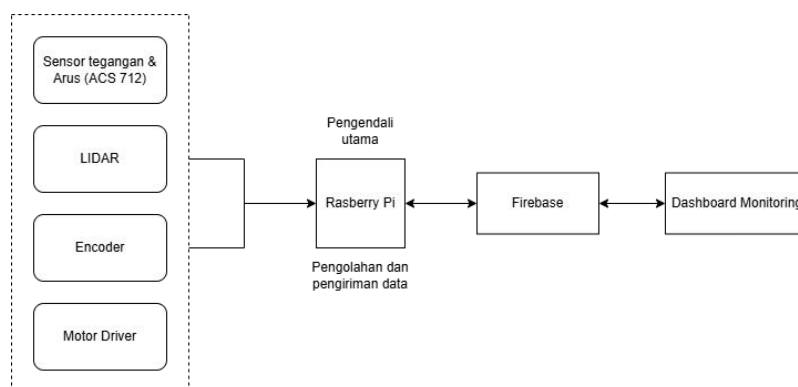
3. HASIL DAN PEMBAHASAN

3.1 Hasil Perancangan *Dashboard*

Meskipun pada penelitian ini data yang ditampilkan masih berupa data simulasi (dummy data), struktur dashboard telah dirancang untuk mendukung integrasi dengan subsistem monitoring dan controlling SIMETRI yang sebenarnya. Pada implementasi sistem, Raspberry Pi berperan sebagai pusat pengolahan data yang menerima informasi dari berbagai sensor yang terpasang pada SIMETRI. Data hasil pembacaan sensor kemudian dikirimkan ke Firebase Realtime Database melalui jaringan WiFi dan ditampilkan pada dashboard secara real-time. Parameter yang direncanakan untuk dimonitor meliputi tegangan baterai, arus baterai, daya listrik, kapasitas baterai, status operasional robot, serta informasi posisi dan koordinat troli selama proses navigasi. Selain berfungsi sebagai media monitoring, dashboard juga dirancang untuk mendukung fungsi controlling, yaitu memungkinkan operator memberikan perintah tujuan kepada SIMETRI melalui antarmuka aplikasi. Perintah yang dikirimkan pengguna akan diteruskan melalui Firebase dan diterima oleh Raspberry Pi untuk diproses sebagai instruksi navigasi robot menuju lokasi yang ditentukan.

Mekanisme pertukaran data dilakukan secara dua arah (two-way communication), yaitu pengiriman data kondisi sistem dari SIMETRI ke dashboard untuk keperluan monitoring, serta pengiriman perintah dari dashboard ke SIMETRI untuk keperluan controlling. Dengan mekanisme tersebut, dashboard tidak hanya berfungsi sebagai media visualisasi data, tetapi juga sebagai sarana pemantauan dan pengendalian operasional SIMETRI secara terintegrasi sehingga dapat mendukung mobilitas troli medis otonom secara lebih efektif dan efisien.





Gambar 2. Diagram Blok

Diagram blok pada Gambar di atas menunjukkan rancangan integrasi antara perangkat keras SIMETRI dengan dashboard monitoring berbasis Android. Sistem monitoring terdiri dari beberapa sensor dan perangkat utama yang saling terhubung untuk memperoleh, mengolah, mengirimkan, dan menampilkan data operasional robot secara real-time. Sensor tegangan dan arus ACS712 digunakan untuk memantau kondisi konsumsi daya pada sistem, seperti besar arus yang digunakan serta kondisi sumber daya listrik robot. Sensor LiDAR digunakan untuk mendukung proses navigasi dan pendeteksian lingkungan sekitar selama robot beroperasi. Encoder digunakan untuk memperoleh informasi pergerakan roda yang dapat dimanfaatkan untuk menghitung jarak tempuh maupun kecepatan robot. Selain itu, motor driver berfungsi sebagai antarmuka pengendali motor sekaligus menyediakan informasi status aktuator yang digunakan oleh sistem.

Seluruh data dari sensor dan perangkat tersebut dikirimkan ke Raspberry Pi yang berperan sebagai pengendali utama dan pusat pengolahan data. Raspberry Pi melakukan proses akuisisi, pengolahan, dan pengemasan data sebelum mengirimkannya ke Firebase Realtime Database melalui jaringan WiFi. Firebase berfungsi sebagai media penyimpanan dan pertukaran data secara real-time antara perangkat SIMETRI dan aplikasi monitoring. Data yang tersimpan pada Firebase kemudian diakses oleh dashboard monitoring sehingga operator dapat memantau kondisi sistem melalui perangkat Android. Informasi yang ditampilkan pada dashboard meliputi kondisi daya, status operasional robot, data navigasi, serta parameter lainnya yang diperlukan untuk mendukung proses monitoring SIMETRI secara terintegrasi.

3.1.1 Halaman *Login*



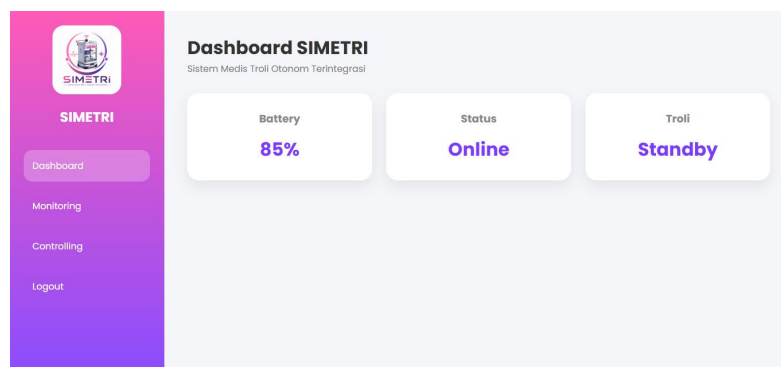
Gambar 3. Halaman *Login*

Halaman *Login* dirancang sebagai mekanisme autentikasi untuk membatasi akses terhadap sistem *monitoring* dan *controlling* SIMETRI. Penerapan autentikasi diperlukan untuk menjaga keamanan data operasional serta mencegah akses oleh pihak yang tidak berwenang. Desain antarmuka dibuat sederhana



dengan jumlah komponen yang minimal sehingga pengguna dapat melakukan proses login secara cepat melalui perangkat Android. Pendekatan ini dipilih untuk mendukung kemudahan penggunaan pada lingkungan operasional rumah sakit yang menuntut akses informasi secara efisien.

3.1.2 Halaman *Dashboard*



Gambar 4. Halaman *Dashboard*

Halaman *Dashboard* dirancang sebagai pusat informasi utama yang memberikan gambaran umum mengenai kondisi SIMETRI. Informasi ditampilkan menggunakan konsep *card-based interface* sehingga data dapat dibaca dengan cepat pada layar *smartphone*. Pemilihan desain ini bertujuan untuk meningkatkan keterbacaan informasi dan mengurangi kompleksitas tampilan. Selain itu, *dashboard* berfungsi sebagai pusat navigasi yang menghubungkan pengguna dengan fitur *monitoring* dan *controlling* sehingga operator dapat mengakses seluruh fungsi sistem dari satu halaman utama.

3.1.3 Halaman *Monitoring*



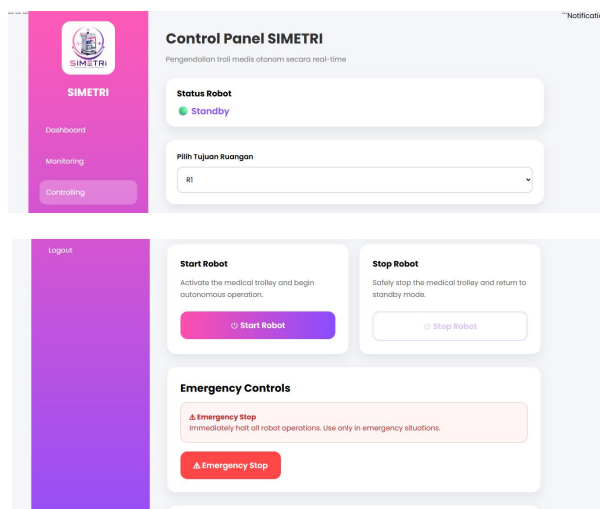
Gambar 5. Halaman *Monitoring*

Halaman *Monitoring* dirancang untuk menampilkan parameter operasional SIMETRI secara terpusat dan *real-time*. Informasi seperti kondisi baterai, konsumsi daya, status operasional robot, serta data posisi dan navigasi ditampilkan dalam format yang mudah dipahami oleh pengguna. Penyajian data secara terstruktur bertujuan untuk membantu operator dalam memantau kondisi sistem tanpa harus melakukan pemeriksaan langsung pada perangkat. Fitur *monitoring* ini memiliki peran penting dalam mendukung keandalan



operasional SIMETRI karena memungkinkan deteksi dini terhadap kondisi yang berpotensi mengganggu kinerja robot selama menjalankan tugas distribusi logistik medis.

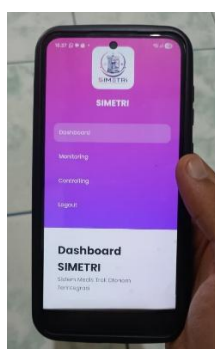
3.1.4 Halaman *Controlling*



Gambar 6. Halaman *Controlling*

Halaman *Controlling* dirancang untuk memungkinkan operator memberikan perintah operasional kepada SIMETRI melalui perangkat *Android*. Fitur ini mendukung konsep interaksi jarak jauh antara pengguna dan robot sehingga operator dapat menentukan tujuan atau memberikan instruksi tanpa harus berada di dekat perangkat. Dalam implementasi sistem, perintah yang diberikan melalui dashboard akan diteruskan ke *Raspberry Pi* melalui *Firebase Realtime Database* untuk diproses sebagai instruksi navigasi. Kehadiran fitur *controlling* menjadi komponen penting karena tidak hanya mendukung fungsi *monitoring*, tetapi juga memungkinkan pengendalian operasional SIMETRI secara terintegrasi dalam satu platform.

3.2 Hasil Implementasi Aplikasi *Android*



Gambar 7. Hasil Implementasi

Hasil implementasi menunjukkan bahwa dashboard yang dikembangkan dapat diakses melalui perangkat *Android* dalam bentuk aplikasi tanpa memerlukan proses pengembangan aplikasi native. Penggunaan AppsGeyser memungkinkan dashboard berbasis web dikemas menjadi aplikasi *Android* sehingga pengguna dapat mengakses sistem monitoring dan controlling melalui antarmuka yang lebih



praktis. Pendekatan ini dipilih karena mampu mempercepat proses pengembangan serta memudahkan distribusi aplikasi kepada pengguna. Implementasi dashboard pada perangkat Android memberikan fleksibilitas bagi operator dalam melakukan pemantauan dan pengendalian SIMETRI. Operator dapat mengakses informasi kondisi sistem maupun memberikan perintah operasional melalui perangkat bergerak yang mudah dibawa dan digunakan. Hal ini sesuai dengan kebutuhan operasional di lingkungan rumah sakit yang menuntut mobilitas tinggi serta akses informasi yang cepat dan mudah.

Selain itu, penggunaan teknologi web memungkinkan pengembangan dan pemeliharaan sistem dilakukan secara lebih efisien karena perubahan pada dashboard dapat dilakukan pada sisi server tanpa memerlukan proses instalasi ulang aplikasi pada setiap perangkat pengguna. Dengan demikian, implementasi dashboard berbasis Android ini dapat menjadi solusi yang praktis dan fleksibel untuk mendukung pengoperasian SIMETRI secara terintegrasi.

3.3 Pengujian Sistem

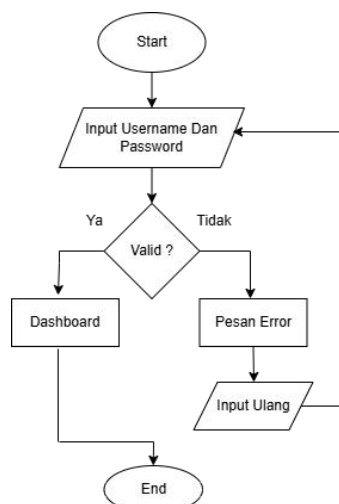
3.3.1 Black Box Testing

TABEL III : *Black Box Testing*

No	Fitur	Pengujian	Hasil	Status
1	<i>Login</i>	Username dan password benar	Menampilkan <i>Dashboard</i>	Berhasil
2	<i>Login</i>	Username atau password salah	Menampilkan pesan kesalahan	Berhasil
3	<i>Dashboard</i>	Membuka halaman <i>Dashboard</i>	<i>Dashboard</i> Tampil	Berhasil
4	<i>Monitoring</i>	Membuka halaman <i>Monitoring</i>	Data <i>Monitoring</i> Tampil	Berhasil
5	<i>Controlling</i>	Membuka halaman <i>Controlling</i>	Menu <i>Controlling</i> tampil	Berhasil
6	<i>Logout</i>	Menekan tombol <i>Logout</i>	Kembali ke halaman <i>Login</i>	Berhasil

Berdasarkan hasil pengujian *Black Box*, seluruh fungsi utama aplikasi dapat berjalan sesuai dengan rancangan sistem. Semua menu dapat diakses dengan baik dan tidak ditemukan kesalahan fungsi selama proses pengujian.

3.3.2 White Box Testing



p-ISSN: 2356-0533; e-ISSN: 2355-9195



Gambar 3.7 *White Box Testing*

Tabel 3.2 *White Box Testing*

<i>Node</i>	Keterangan
1	Start
2	Input Username dan Password
3	Validasi Data
4	Keputusan Valid/Tidak
5	<i>Dashboard</i>
6	Pesan Error
7	End

E = Jumlah edge = 7

N = Jumlah *Node* = 7

Maka:

$V(G) = 7 - 7 + 2 = 2$

Hasil perhitungan menunjukkan nilai Cyclomatic Complexity sebesar 2 yang menandakan terdapat dua jalur independen pada proses *Login*. Seluruh jalur independen berhasil dieksekusi tanpa ditemukan kesalahan logika program.

3.4 Pembahasan

Berdasarkan hasil implementasi dan pengujian, *Dashboard Monitoring* berbasis *Android* berhasil dikembangkan sesuai dengan kebutuhan penelitian. Aplikasi mampu menampilkan halaman *Login*, *Dashboard*, *Monitoring*, dan *Controlling* dengan baik pada perangkat *Android*. Penggunaan *HTML*, *CSS*, dan *JavaScript* memberikan kemudahan dalam proses pengembangan antarmuka, sedangkan *Visual Studio Code* mendukung proses pemrograman secara efektif. Konversi aplikasi menggunakan *AppsGeyser* memungkinkan *Dashboard* web dijalankan sebagai aplikasi *Android* tanpa perlu melakukan pengembangan native *Android*. Meskipun demikian, penelitian ini masih memiliki keterbatasan karena data *Monitoring* yang ditampilkan masih berupa data simulasi (dummy data) dan belum terhubung secara langsung dengan sensor maupun perangkat keras SIMETRI. Oleh karena itu, pengembangan selanjutnya dapat difokuskan pada integrasi *Dashboard* dengan sensor dan sistem komunikasi data secara real-time menggunakan teknologi Internet of Things (IoT).

Berdasarkan hasil pengujian yang telah dilakukan, dashboard monitoring yang dikembangkan tidak hanya mampu menjalankan seluruh fungsi aplikasi sesuai dengan rancangan, tetapi juga berpotensi meningkatkan efektivitas proses pemantauan operasional SIMETRI. Melalui dashboard ini, operator dapat memperoleh informasi kondisi sistem secara terpusat tanpa harus melakukan pemeriksaan langsung pada perangkat. Informasi seperti kondisi baterai, status operasional robot, serta posisi troli dapat ditampilkan dalam satu antarmuka sehingga mempermudah proses pengawasan selama sistem beroperasi.

Dari sisi kemudahan penggunaan (usability), dashboard dirancang dengan antarmuka yang sederhana dan responsif sehingga dapat diakses melalui perangkat *Android* yang umum digunakan oleh pengguna.



Penyajian informasi dalam bentuk menu yang terstruktur memungkinkan operator mengakses fitur monitoring maupun controlling secara lebih cepat dan intuitif. Dengan demikian, pengguna tidak memerlukan pengetahuan teknis yang mendalam untuk memahami kondisi sistem maupun memberikan perintah operasional kepada SIMETRI.

Dashboard ini berpotensi mendukung pengelolaan operasional troli medis otonom secara lebih efektif. Operator dapat melakukan pemantauan kondisi robot secara real-time sekaligus mengirimkan perintah tujuan tanpa harus berada di dekat perangkat. Kemampuan tersebut dapat membantu meningkatkan efisiensi distribusi logistik medis, mengurangi kebutuhan interaksi langsung dengan troli, serta mempercepat proses pengambilan keputusan ketika terjadi gangguan atau kondisi yang memerlukan tindakan tertentu. Oleh karena itu, dashboard monitoring dan controlling yang dikembangkan dapat menjadi salah satu komponen pendukung dalam implementasi sistem troli medis otonom yang lebih terintegrasi di lingkungan rumah sakit.

4. KESIMPULAN

Berdasarkan hasil penelitian, dashboard monitoring berbasis Android pada Sistem Medis Troli Otonom Terintegrasi (SIMETRI) berhasil dirancang dan diimplementasikan menggunakan HTML, CSS, dan JavaScript serta dikonversi menjadi aplikasi Android menggunakan AppsGeyser. Hasil pengujian Black Box dan White Box menunjukkan bahwa seluruh fungsi aplikasi berjalan sesuai dengan rancangan. Selain mendukung fungsi monitoring, dashboard juga dirancang untuk mendukung controlling SIMETRI melalui integrasi dengan Raspberry Pi dan Firebase Realtime Database. Pengembangan selanjutnya dapat difokuskan pada implementasi data real-time dari sensor dan perangkat keras SIMETRI untuk mendukung operasional sistem secara lebih optimal.

DAFTAR PUSTAKA

- [1] R. Mukhaiyar, "Monitoring Kecepatan Angin berbasis Mikrokontroler dan IoT," vol. 3, no. 2, pp. 437–447, 2022.
- [2] U. I. Batam, "Mengembangkan Situs Web Front-End dengan HTML , CSS , dan JavaScript," vol. 5, no. September, pp. 1121–1127, 2023.
- [3] J. Otomasi, U. Latifa, U. S. Maret, P. Teknik, T. Elektro, and U. S. Karawang, "Prototipe Sistem Peringatan Dini Bendungan Walahar Karawang Berbasis Internet of Things (IoT) dengan Antarmuka Web dan Aplikasi Mobile," vol. 14, no. 1, pp. 31–40, 2022.
- [4] I. K. Putri, "Sistem Kontrol Instalasi Rumah Berbasis IoT (Internet of Things)," vol. 4, no. 2, pp. 675–682, 2023.
- [5] T. Online, F. Harri, D. Kurniawan, and J. Sutopo, "Jurnal Politeknik Caltex Riau Rancang Bangun Aplikasi Monitoring Latihan Kebugaran Jasmani Berbasis Internet of Things," vol. 9, no. 2, pp. 173–181, 2023.
- [6] I. Y. Zaki *et al.*, "Rancang Bangun Sistem Monitoring Detak Jantung , Suhu Tubuh , dan Cairan Infus Berbasis Internet of Things Lilik Anifah , Farid Baskoro , Pradini Puspitaningayu," pp. 14–22, 2012.
- [7] A. Nurfebrian, A. Pindarwati, and R. Hidayat, "Perancangan Dan Implementasi Sistem Monitoring Jaringan Berbasis Web PT . Maxindo Mitra Solusi," vol. 1, no. 2, pp. 486–495, 2023.
- [8] R. Setiawan, D. Kurniadi, and Y. Supriatna, "Perancangan Sistem Informasi Monitoring dan Pelaporan Kinerja Aparatur Sipil Negara Berbasis Web dan Android," pp. 156–167, 2014.
- [9] A. C. Praniffa, A. Syahri, F. Sandes, U. Fariha, and Q. A. Giansyah, "Jurnal Testing dan Implementasi Sistem Informasi PARKIR BERBASIS WEB BLACK BOX AND WHITE BOX TESTING OF WEB-BASED PARKING," vol. 1, no. 1, pp. 1–16, 2023.
- [10] H. Raihan, I. Fikri, and A. Voutama, "P ENGUJIAN B LACK B OX P ADA A PLIKASI," vol. 17, no. 1, pp. 1–18, 2023.

