Jurnal Riset Multidisiplin Edukasi
https://journal.hasbaedukasi.co.id/index.php/jurmie
Halaman : 596 - 611

IMPLEMENTASI ALGORITMA RULE-BASED PADA SISTEM MONITORING ROBOT PEMBERSIH LANTAI BERBASIS MOBILE

M. Putra Mahdi Pratama Lubis[1], Khairuddin Nasution[2], Mhd.Zulfansyuri Siambaton[3]
Universitas Islam Sumatera Utara, Medan, Indonesia [1,2,3]
Email: m.putramahdipratama@gmail.com[1], khairuddin_nst@uisu.ac.id[2], zulfansyuri@ft.uisu.ac.id[3]

Informasi

Volume : 3
Nomor : 11
Bulan : November
Tahun : 2026
E-ISSN : 3062-9624

Abstract

This study aims to implement the Rule-Based algorithm in a mobile-based floor cleaning robot monitoring system and develop an application allowing users to monitor the robot's condition via smartphone. The underlying problem is the limitation of users in knowing the robot's operational state without inspecting it directly. The robot's hardware architecture utilizes an Arduino Uno as the main microcontroller for sensor processing and motor control, and a separate ESP32-CAM module specifically dedicated to transmitting live camera video streaming. The research method follows Research and Development (R&D), comprising problem identification, literature review, requirements analysis, design, implementation, and testing. The mobile app was built using the Flutter framework and Dart language, utilizing Firebase Realtime Database for data synchronization. The Rule-Based algorithm operates on the app side using IF-THEN logic and threshold values to translate raw monitoring data into clear status updates, battery conditions, and dustbin capacities. Functionality was tested using Black Box Testing. The result is a mobile monitoring system that integrates data sourced from the Arduino Uno hardware along with video streaming from the ESP32-CAM.

Keyword: *Rule-Based, Monitoring, Floor Cleaning Robot, Arduino Uno, ESP32-CAM, Flutter, Firebase.*

Abstrak

Penelitian ini bertujuan untuk mengimplementasikan algoritma Rule-Based pada sistem monitoring robot pembersih lantai berbasis mobile serta mengembangkan aplikasi yang membantu pengguna memantau kondisi robot melalui smartphone. Permasalahan yang melatarbelakangi penelitian ini adalah keterbatasan pengguna dalam mengetahui kondisi operasional robot ketika sedang bekerja tanpa harus melakukan pemeriksaan langsung. Arsitektur perangkat keras robot memanfaatkan Arduino Uno sebagai mikrokontroler utama untuk pengolahan data sensor dan penggerak, serta modul ESP32-CAM yang bekerja secara terpisah khusus untuk mentransmisikan tayangan live camera stream. Metode penelitian yang digunakan adalah Research and Development (R&D), meliputi identifikasi masalah, studi literatur, analisis kebutuhan, perancangan, implementasi, dan pengujian. Aplikasi dikembangkan menggunakan framework Flutter dengan bahasa pemrograman Dart, serta Firebase Realtime Database sebagai media penyimpanan data monitoring. Algoritma Rule-Based diterapkan pada sisi aplikasi menggunakan logika IF-THEN dan nilai threshold untuk mengolah data sensor menjadi informasi status operasional, kondisi baterai, dan kapasitas penampung kotoran. Pengujian fungsionalitas menggunakan metode Black Box Testing. Hasil pengembangan menunjukkan bahwa aplikasi mobile monitoring mampu mengintegrasikan data dari Arduino Uno dan stream video dari ESP32-CAM secara real-time dan informatif.

Kata Kunci : *Rule-Based, Monitoring, Robot Pembersih Lantai, Arduino Uno, ESP32-CAM, Flutter, Firebase.*

A. PENDAHULUAN

Penggunaan mesin otomatis dalam kehidupan sehari-hari kini semakin diminati, salah satunya adalah pemanfaatan robot pembersih lantai otomatis untuk membantu meringankan pekerjaan membersihkan ruangan (Darusman dkk., 2025). Robot ini merupakan perangkat elektromekanis yang dikendalikan oleh mikrokontroler dan dirancang agar mampu bernavigasi serta menghindar dari halangan secara mandiri (Ihsan & Rahmawaty, 2023). Dalam struktur perangkat kerasnya, Arduino Uno bertindak sebagai pemroses utama (main controller) yang mengolah masukan data sensor serta mengendalikan sistem penggerak roda dan penyapu lantai (Yuliza & Kholifah, 2015). Di sisi lain, untuk menyediakan fasilitas pengawasan visual, modul ESP32-CAM dipasang secara mandiri khusus untuk mentransmisikan tayangan video langsung (live stream) melalui jaringan Wi-Fi (Arifin & Mulyana, 2022).

Kendati demikian, mobilitas robot sering kali menemui kendala di luar dugaan ketika ditinggal beroperasi. Pengguna kerap mengalami kesulitan karena tidak dapat memantau kondisi perangkat dari jarak jauh, seperti mengetahui persentase sisa baterai atau mendeteksi apabila tempat penampung sampah sudah penuh. Oleh karena itu, fokus penelitian ini bertumpu pada pembangunan perangkat lunak aplikasi mobile sebagai pusat monitoring jarak jauh (Fauzi & Hidayat, 2023). Aplikasi ini dibangun menggunakan kerangka kerja Flutter dan bahasa pemrograman Dart karena terbukti andal dalam menghasilkan antarmuka yang responsif serta mendukung kompilasi lintas platform (Rahayu dkk., 2024; Wijaya dkk., 2024).

Di dalam aplikasi pemantauan ini, diterapkan algoritma Rule-Based, yaitu teknik pemrograman logis yang mengevaluasi nilai parameter masukan berdasarkan batas ambang (threshold) menggunakan struktur bersyarat IF-THEN (Sandi dkk., 2024). Pendekatan ini dipilih karena efisien dan ringan untuk memproses data pemantauan secara real-time pada perangkat seluler tanpa membebani komputasi smartphone (Mustofa, 2017).

Berdasarkan pemaparan pada latar belakang, rumusan masalah dalam penelitian ini adalah bagaimana merancang dan membangun aplikasi pada sistem monitoring robot pembersih lantai berbasis mobile, bagaimana mengimplementasikan algoritma rule-based pada sistem monitoring robot pembersih lantai berbasis mobile, serta bagaimana hasil pengujian fungsionalitas terhadap sistem aplikasi yang telah dibangun.

Agar penelitian ini lebih terarah dan berfokus pada tujuan utama, ruang lingkup penelitian dibatasi pada pengembangan perangkat lunak sisi klien, yaitu pembangunan aplikasi mobile menggunakan bahasa pemrograman Dart dan kerangka kerja Flutter. Perangkat keras

robot pembersih lantai menggunakan Arduino Uno sebagai pengolah utama data sensor dan penggerak fisik, sedangkan modul ESP32-CAM berfungsi secara terpisah untuk mengirimkan umpan video live stream pasif tanpa pemrosesan image recognition di dalam aplikasi. Sistem aplikasi mengevaluasi dan merespons data sensor serta status operasional secara eksklusif menggunakan logika pemrograman bersyarat atau algoritma rule-based, tanpa menyertakan modul atau library kecerdasan buatan (AI) di dalam aplikasi. Media pertukaran data monitoring antara perangkat keras dan aplikasi mobile menggunakan Firebase Realtime Database. Selain itu, proses pembangunan dan pengujian program dilakukan di lingkungan lokal menggunakan perangkat lightweight standalone SDK tools guna menghemat ruang penyimpanan laptop, tanpa memerlukan instalasi penuh Android Studio.

Adapun tujuan penelitian ini adalah merancang dan membangun aplikasi mobile yang interaktif dan responsif sebagai antarmuka (user interface) pada sistem monitoring robot pembersih lantai, mengimplementasikan algoritma rule-based pada sistem monitoring robot pembersih lantai berbasis mobile guna memproses data sensor menjadi status operasional secara langsung (real-time), serta melakukan pengujian fungsionalitas terhadap sistem aplikasi yang telah dibangun menggunakan metode Black Box Testing untuk memastikan setiap fitur berjalan dengan baik sesuai dengan kebutuhan tanpa mengalami kendala atau error.

B. METODE PENELITIAN

Penelitian ini dilaksanakan di Laboratorium Robotika, Fakultas Teknik Informatika, Universitas Islam Sumatera Utara (UISU), selama kurang lebih tiga bulan pada tahun 2026. Penelitian menggunakan metode Research and Development (R&D) untuk mengembangkan aplikasi mobile monitoring robot pembersih lantai CLEANBOT. Produk yang dikembangkan berfokus pada perangkat lunak aplikasi, sedangkan robot dan perangkat kerasnya menjadi objek pemantauan serta sumber data. Tahapan penelitian meliputi identifikasi masalah, studi literatur, analisis kebutuhan sistem, perancangan, implementasi aplikasi, pengujian, analisis hasil, dan penarikan kesimpulan. Identifikasi masalah diarahkan pada kebutuhan pemantauan kondisi robot dari jarak jauh, seperti status perangkat, baterai, jarak sensor, sensor debu, dan kapasitas penampung debu. Studi literatur mencakup aplikasi mobile, Flutter, Dart, Firebase Realtime Database, algoritma Rule-Based, live stream, serta pengujian perangkat lunak.

Analisis kebutuhan mencakup perangkat keras, perangkat lunak, kebutuhan fungsional, dan nonfungsional. Perangkat yang digunakan antara lain laptop, smartphone Oppo A53, Arduino Uno, ESP32-CAM, sensor, kabel USB, dan jaringan Wi-Fi. Aplikasi dibangun

menggunakan Flutter dan Dart, dengan Firebase Realtime Database sebagai media penyimpanan serta pertukaran data. Fungsi aplikasi meliputi dashboard monitoring, tampilan status dan baterai, pengambilan data, pengolahan Rule-Based, penyajian peringatan, live stream kamera, serta riwayat pembersihan. Algoritma Rule-Based diterapkan di sisi aplikasi melalui aturan IF-THEN dan nilai threshold untuk menginterpretasikan data, bukan mengendalikan gerakan robot. Perancangan meliputi arsitektur sistem, struktur basis data, diagram UML, flowchart, dan antarmuka. Pengujian dilakukan menggunakan Black Box Testing dan Equivalence Partitioning, dengan membandingkan keluaran aktual dan yang diharapkan pada berbagai kondisi input. Hasil pengujian menjadi dasar evaluasi keberhasilan fitur, identifikasi kendala, penyusunan kesimpulan, serta saran pengembangan selanjutnya.

C. HASIL DAN PEMBAHASAN

Hasil Penelitian

Implementasi Sistem

Implementasi Perangkat Lunak

Aplikasi dirancang menggunakan Flutter sebagai framework pengembangan antarmuka dan Dart sebagai bahasa pemrograman. Firebase Realtime Database digunakan sesuai rancangan sebagai layanan penyimpanan dan pertukaran data monitoring. ESP32-CAM berfungsi sebagai sumber video yang ditampilkan melalui fitur live stream. Kesesuaian implementasi dengan kode program aktual perlu diperiksa sebelum naskah final diserahkan.

Secara umum, alur aplikasi dimulai ketika pengguna membuka aplikasi, aplikasi mengambil data monitoring, data yang tersedia ditampilkan pada dashboard dan dievaluasi menggunakan aturan Rule-Based. Pengguna juga dapat mengakses fitur kamera dan riwayat pembersihan sesuai fitur yang tersedia.

Implementasi Algoritma Rule-Based

Algoritma Rule-Based digunakan pada sisi aplikasi untuk menginterpretasikan data monitoring berdasarkan aturan IF-THEN. Kondisi IF diperiksa terlebih dahulu, kemudian keluaran THEN ditentukan apabila kondisi tersebut terpenuhi. Algoritma ini digunakan untuk menghasilkan informasi monitoring dan tidak digunakan untuk mengendalikan navigasi atau pergerakan robot.

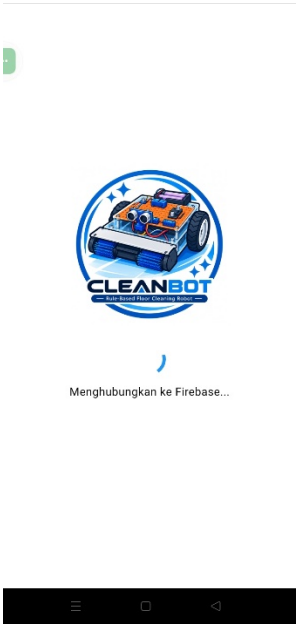
Contoh aturan konseptual untuk baterai: IF nilai baterai kurang dari atau sama dengan threshold baterai, THEN aplikasi menampilkan peringatan baterai lemah. Contoh tersebut merupakan ilustrasi aturan. Nilai threshold akhir harus dicocokkan dengan kode program dan

ketentuan sistem yang sebenarnya.

Implementasi Antarmuka Aplikasi

Tampilan Splash Screen

Splash screen merupakan tampilan awal yang muncul ketika aplikasi dijalankan. Tampilan ini menampilkan identitas aplikasi sebelum pengguna masuk ke halaman utama. Fungsi utamanya adalah sebagai tampilan pembuka, bukan untuk melakukan pengolahan data monitoring atau pengendalian robot.



Gambar 1 Tampilan Splash Screen Aplikasi

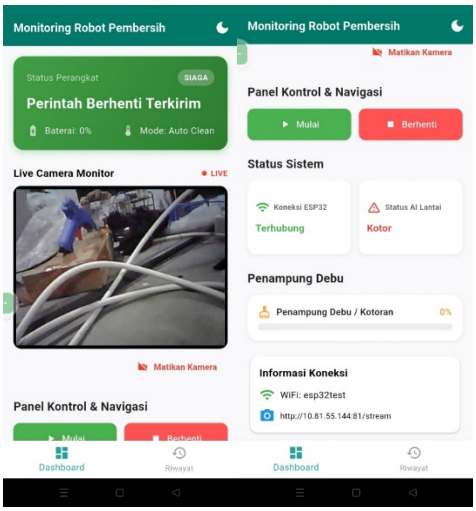
Tampilan Halaman Dashboard

Dashboard merupakan halaman utama untuk menyajikan informasi monitoring. Dokumentasi antarmuka menunjukkan status perangkat, persentase baterai, mode operasi, dan bagian Live Camera Monitor. Pada tangkapan layar yang tersedia,. Hal ini dicatat sebagai kondisi pada saat dokumentasi diambil dan belum cukup untuk menyimpulkan penyebab kegagalan.

Dashboard merupakan halaman utama yang digunakan untuk memantau kondisi robot pembersih lantai. Pada halaman ini terdapat beberapa bagian, yaitu: (1) Status Perangkat, menampilkan informasi kondisi robot berdasarkan data monitoring yang diterima aplikasi. (2) Indikator Baterai, menampilkan persentase kapasitas baterai robot agar pengguna dapat mengetahui kondisi daya. (3) Mode Operasi, menampilkan informasi mode pengoperasian robot, seperti mode pembersihan otomatis. (4) Live Camera Monitor, merupakan bagian yang digunakan untuk melihat kondisi area di sekitar robot melalui kamera ESP32-CAM. (5) Panel Kontrol, menyediakan tombol mulai dan berhenti sesuai rancangan antarmuka. (6) Indikator

Sensor dan Sistem, menampilkan informasi kondisi sensor debu, sensor ultrasonik, dan koneksi ESP32-CAM. (7) Indikator Kapasitas Penampung Debu, menampilkan persentase kapasitas wadah debu melalui progress bar. (8) Menu Navigasi Bawah, digunakan untuk berpindah ke bagian atau halaman aplikasi yang tersedia.

Dengan adanya bagian tersebut, pengguna dapat memperoleh informasi kondisi robot melalui satu halaman utama.

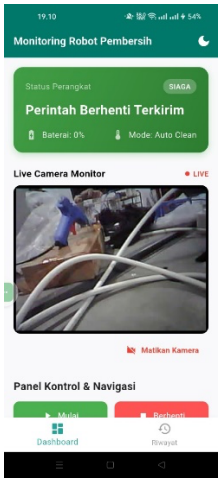


Gambar 2 Tampilan Dashboard

Tampilan Pemantauan Kamera (Live Camera Monitor)

Fitur Live Camera Monitor digunakan untuk menampilkan video secara langsung dari kamera ESP32-CAM yang terpasang pada robot. Tampilan ini berfungsi sebagai media pemantauan visual sehingga pengguna dapat melihat kondisi area yang sedang dipantau oleh robot.

Video ditampilkan menggunakan mekanisme real-time MJPEG stream. Dengan fitur ini, pengguna dapat melihat kondisi lingkungan melalui aplikasi tanpa harus memeriksa robot secara langsung.

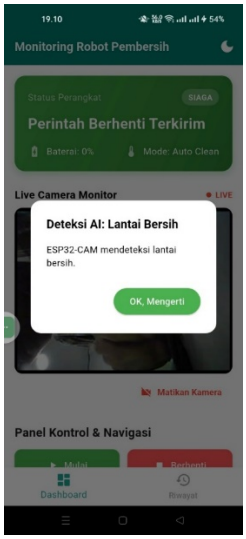


Gambar 3 Tampilan Live Camera Monitor ESP32-CAM Terhubung

Tampilan Sistem Notifikasi Pop-up Rule-Based

Sistem notifikasi pop-up merupakan fitur pada aplikasi CLEANBOT yang berfungsi memberikan informasi mengenai kondisi robot berdasarkan data monitoring dari Firebase Realtime Database. Data tersebut diperiksa menggunakan algoritma Rule-Based dengan aturan IF-THEN, kemudian aplikasi menampilkan notifikasi secara otomatis sesuai kondisi yang diterima.

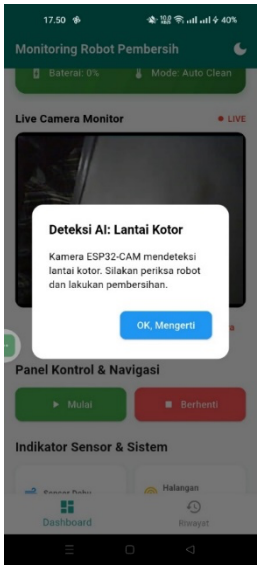
Notifikasi yang ditampilkan meliputi kondisi lantai bersih, lantai kotor, dan penampung sampah penuh. Adapun tampilan masing-masing notifikasi dijelaskan sebagai berikut.



Gambar 4 Tampilan Notifikasi Pop-up AI: Lantai Bersih

Pada Gambar 4 ditampilkan notifikasi pop-up ketika sistem menerima informasi bahwa kondisi lantai berada dalam status **BERSIH**. Ketika nilai kondisi lantai (floor_condition) yang diterima aplikasi menunjukkan status tersebut, aturan Rule-Based akan memeriksa kesesuaian data dengan aturan yang telah ditentukan. Setelah kondisi terpenuhi, aplikasi secara otomatis menampilkan dialog informasi berwarna hijau dengan pesan *"ESP32-CAM mendeteksi lantai bersih"*.

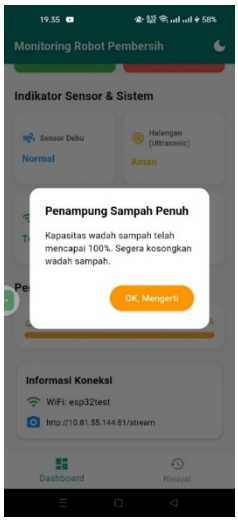
Warna hijau pada notifikasi digunakan sebagai penanda bahwa kondisi lantai berada dalam keadaan bersih. Pesan tersebut memberikan informasi kepada pengguna bahwa berdasarkan data yang diterima sistem, lantai telah teridentifikasi bersih. Dengan adanya notifikasi ini, pengguna dapat mengetahui kondisi lantai melalui aplikasi tanpa harus selalu mengamati tampilan kamera secara langsung.



Gambar 5 Tampilan Notifikasi Pop-up AI: Lantai Kotor

Pada Gambar 5 ditampilkan notifikasi pop-up ketika sistem menerima informasi bahwa kondisi lantai berada dalam status **KOTOR**. Kondisi ini terjadi ketika nilai floor_condition yang diterima aplikasi menunjukkan status kotor. Selanjutnya, algoritma Rule-Based memeriksa data tersebut berdasarkan aturan yang telah ditentukan. Apabila kondisi sesuai dengan aturan, aplikasi akan menampilkan dialog peringatan berwarna biru.

Notifikasi tersebut menampilkan pesan *"Kamera ESP32-CAM mendeteksi lantai kotor. Silakan periksa robot dan lakukan pembersihan."* Pesan ini bertujuan memberikan peringatan kepada pengguna bahwa kondisi lantai memerlukan perhatian dan tindakan pembersihan. Melalui informasi tersebut, pengguna dapat mengetahui bahwa lantai masih dalam kondisi kotor dan perlu dilakukan pemeriksaan atau pembersihan lebih lanjut.



Gambar 6 Tampilan Notifikasi Pop-up Penampung Sampah Penuh

Pada Gambar 6 ditampilkan notifikasi pop-up ketika kapasitas penampung sampah telah

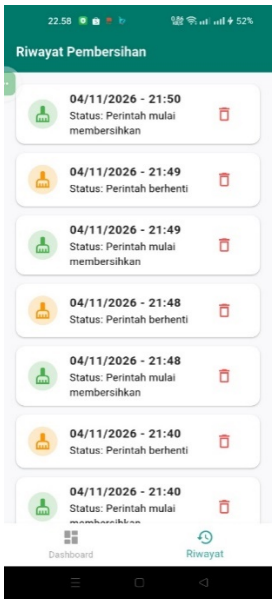
mencapai batas maksimum. Sistem menggunakan nilai sensor kapasitas debu (dust_level) sebagai parameter untuk mengetahui kondisi penampung sampah. Ketika nilai tersebut mencapai 100% atau 1.0, aturan Rule-Based akan mengenali bahwa penampung sampah telah penuh dan memicu munculnya dialog peringatan secara otomatis.

Notifikasi ditampilkan dengan warna oranye dan berisi pesan *"Kapasitas wadah sampah telah mencapai 100%. Segera kosongkan wadah sampah."* Warna oranye digunakan sebagai tanda peringatan agar pengguna memperhatikan kondisi penampung sampah. Melalui notifikasi ini, pengguna memperoleh informasi bahwa penampung sampah perlu segera dikosongkan agar proses penggunaan robot dapat dilanjutkan dengan kondisi penampung yang memadai.

Berdasarkan ketiga tampilan tersebut, fitur notifikasi pop-up pada aplikasi CLEANBOT membantu pengguna memperoleh informasi mengenai kondisi lantai dan kapasitas penampung sampah secara otomatis. Penerapan algoritma Rule-Based memungkinkan aplikasi memeriksa data berdasarkan aturan yang telah ditentukan, kemudian menampilkan notifikasi sesuai dengan kondisi yang diterima. Dengan demikian, pengguna dapat mengetahui kondisi yang perlu diperhatikan melalui mobile monitoring.

Tampilan Halaman Riwayat Pembersihan

Halaman riwayat pembersihan dirancang menampilkan catatan aktivitas yang tersedia. Dokumentasi tampilan menunjukkan waktu, durasi, status kegiatan, dan ikon penghapusan riwaya.



Gambar 7 Tampilan Halaman Riwayat Pembersihan

Halaman riwayat pembersihan digunakan untuk menampilkan catatan aktivitas robot

yang telah tersimpan. Pada halaman ini terdapat beberapa informasi, yaitu: (1) Waktu Aktivitas, menampilkan tanggal dan jam ketika aktivitas dilakukan. (2) Status Aktivitas, menampilkan keterangan aktivitas, seperti perintah mulai membersihkan dan perintah berhenti. (3) Durasi Aktivitas, menampilkan informasi lama kegiatan apabila data durasi tersedia. (4) Ikon Hapus, berupa ikon tempat sampah berwarna merah yang digunakan untuk menghapus catatan riwayat tertentu dari database.

Implementasi Algoritma Rule-Based

Rancangan Aturan Rule-Based

Tabel 1 Matriks Aturan Algoritma Rule-Based Sistem CLEANBOT

No.	Data Masukan	Kondisi IF (Rule)	Keluaran THEN (Action & UI)	Status Verifikasi
1	status	Nilai = "SIAGA" / "STANDBY"	Menampilkan Card Status Standby (Siaga)	Berhasil
2	status	Nilai = "MEMBERSIHKAN"	Menampilkan Card Status Membersihkan Ruangan...	Berhasil
3	floor_condition	String mengandung "BERSIH"	Menampilkan Indikator Hijau & Pop-up Dialog Deteksi AI: Lantai Bersih	Berhasil
4	floor_condition	String mengandung "KOTOR"	Menampilkan Indikator Merah & Pop-up Dialog "Deteksi AI: Lantai Kotor"	Berhasil
5	battery	Nilai <= 20%	Menampilkan Pop-up Dialog Peringatan Baterai Lemah"	Berhasil
6	dust_level	Nilai >= 1.0 (100%)	Menampilkan Pop-up Dialog Penampung Sampah Penuh"	Berhasil
7	esp32_connection	Stream IP Terhubung	Menampilkan Indikator Koneksi ESP32-CAM: Terhubung	Berhasil
8	log_event	Pengguna menekan Mulai/Berhenti	Menambahkan item baru pada Halaman Riwayat Pembersihan	Berhasil

Penerapan Rule-Based

Penerapan Rule-Based diharapkan membantu pengguna memahami data monitoring melalui status dan peringatan yang dihasilkan aplikasi. Untuk memastikan ketepatannya, pengujian perlu dilakukan pada kondisi normal, kondisi peringatan, nilai tepat pada threshold, serta kondisi data kosong atau tidak valid. Keberhasilan algoritma belum dapat disimpulkan sebelum hasil pengujian tersedia.

Pengujian Sistem

Rencana Pengujian Black Box Testing

Tabel 2 Hasil Pengujian Black Box Testing

No.	Fungsi	Skenario Pengujian	Keluaran yang Diharapkan	Hasil Aktual	Status
1	Splash Screen	Membuka aplikasi dari ikon launcher	Tampilan logo & nama aplikasi muncul sebelum ke Dashboard	Tampilan pembuka muncul dengan lancar	Berhasil
2	Dashboard UI	Navigasi ke Halaman Utama	Seluruh komponen status, tombol kontrol, dan indikator tampil	Dashboard tampil lengkap sesuai rancangan	Berhasil
3	Live Stream	Menekan tombol nyalakan kamera	Umpan video real-time dari ESP32-CAM dimuat pada layar	Stream video http://10.81.55.144:81/stream tampil (Gambar 4.3)	Berhasil
4	Rule-Based AI Kotor	Data Firebase floor_condition = "Kotor"	Teks indikator berubah merah & Pop-up Notifikasi timbul	Pop-up "Deteksi AI: Lantai Kotor" muncul otomatis (Gambar 4.5)	Berhasil
5	Rule-Based AI Bersih	Data Firebase floor_condition = "Bersih"	Teks indikator berubah hijau & Pop-up Notifikasi timbul	Pop-up "Deteksi AI: Lantai Bersih" muncul otomatis (Gambar 4.4)	Berhasil

6	Rule-Based Sampah Penuh	Data Firebase dust_level = 1.0	Pop-up Peringatan Wadah Sampah Penuh dipicu	Pop-up "Penampung Sampah Penuh" muncul (Gambar 4.6)	Berhasil
7	Riwayat Pembersihan	Menekan tab navigasi Riwayat	Menampilkan daftar log aktivitas beserta tombol hapus	Log aktivitas dengan timestamp dan status tampil runtut (Gambar 4.7)	Berhasil
8	Hapus Riwayat	Menekan ikon tempat sampah pada baris riwayat	Catatan riwayat terhapus dari daftar dan database	Catatan riwayat berhasil terhapus	Berhasil
9	Koneksi Firebase	Memutus koneksi internet saat aplikasi sedang menerima data monitoring	Aplikasi menampilkan informasi koneksi terputus dan tidak memperbarui data monitoring	Data monitoring tidak diperbarui dan informasi koneksi terputus tidak ditampilkan	Gagal
10	Live Stream	Membuka fitur Live Stream ketika ESP32-CAM tidak terhubung ke jaringan	Aplikasi menampilkan informasi bahwa kamera tidak dapat dihubungkan	Tampilan Live Stream tidak menampilkan informasi kesalahan koneksi	Gagal
11	Data monitoring	Mengirim kan data monitoring kosong atau tidak valid dari Firebase	Aplikasi menampilkan informasi yang sesuai dan tidak mengalami error	Data kosong menyebabkan indikator monitoring tidak menampilkan informasi dengan benar	Gagal

D. KESIMPULAN

Berdasarkan hasil penelitian mengenai Implementasi Algoritma Rule-Based pada Sistem Monitoring Robot Pembersih Lantai Berbasis Mobile, dapat disimpulkan bahwa algoritma Rule-Based berhasil dirancang untuk mengolah data dari perangkat sumber menjadi informasi

kondisi robot menggunakan aturan IF-THEN dan nilai threshold, seperti status baterai, kondisi lantai, serta kapasitas penampung sampah, yang ditampilkan dalam bentuk informasi dan notifikasi secara real-time pada aplikasi tanpa digunakan untuk mengendalikan pergerakan robot. Aplikasi mobile monitoring berhasil dikembangkan menggunakan framework Flutter dan bahasa pemrograman Dart dengan Firebase Realtime Database sebagai media penyimpanan dan pertukaran data. Aplikasi menyediakan fitur dashboard, indikator status perangkat, informasi baterai, notifikasi otomatis berdasarkan algoritma Rule-Based, pemantauan kamera secara langsung melalui ESP32-CAM, serta riwayat pembersihan yang dapat mempermudah pengguna dalam memantau kondisi robot melalui smartphone. Selain itu, pengujian fungsionalitas menggunakan metode Black Box Testing dilakukan untuk memeriksa kesesuaian fungsi aplikasi dengan kebutuhan yang telah ditentukan, meliputi fitur splash screen, dashboard, live stream, notifikasi Rule-Based, dan riwayat pembersihan. Hasil pengujian dicatat berdasarkan keluaran aktual dari setiap skenario untuk mengetahui kesesuaian fungsi aplikasi dengan hasil yang diharapkan.

E. DAFTAR PUSTAKA

Adinda, A. P. M., & Maiyana, E. (2025). Pengembangan aplikasi manajemen data buku menggunakan aplikasi Visual Studio Code dengan framework Laravel. METHODIKA: Jurnal Teknik Informatika dan Sistem Informasi, 11(2), 58–62. https://doi.org/10.46880/mtk.v11i2.4345

Arifin, Z., & Mulyana, A. (2022). Penerapan Internet of Things (IoT) pada sistem monitoring berbasis ESP32. Jurnal Jaringan Komputer dan Telekomunikasi, 4(2), 115–122.

Azis, A., & Sujono, S. (2026). Sistem monitoring lingkungan pintar berbasis mikrokontroler. Jurnal Teknologi dan Sistem Embedded, 8(1), 45–52.

Darusman, D., Abbas, A., & Firmanto, A. D. (2025). Robot pembersih lantai otomatis berbasis Arduino Uno. Informatics for Educators and Professionals: Journal of Informatics, 10(2), 39–49.

Faraby, S., Hidayat, R., & Santoso, B. (2017). Rancang bangun robot pembersih lantai otomatis menggunakan mikrokontroler. Jurnal Teknik Elektro dan Komputer, 6(3), 141–150.

Farhanah, N., & Almais, H. (2022). Analisis perangkat lunak aplikasi dalam pembangunan sistem informasi berbasis bergerak. Jurnal Rekayasa Perangkat Lunak Indonesia, 3(1), 12–20.

Firmansyah, A., Pratama, R., & Gunawan, D. (2022). Perancangan sistem monitoring perangkat

keras jarak jauh. Jurnal Sistem Cerdas, 5(2), 88–96.

Gunawan, I., & Rahmat, A. (2020). Implementasi basis data terstruktur untuk aplikasi berbasis mobile. Jurnal Sistem Cerdas, 3(1), 23–30.

Hakim, L., & Hakim, A. (2021). Penerapan algoritma rule-based dalam sistem pengolahan keputusan otomatis. Jurnal Ilmu Komputer dan Artificial Intelligence, 2(1), 10–18.

Hidayat, M., Kurniawan, R., & Setiawan, B. (2022). Pembangunan aplikasi mobile untuk layanan informasi warga. Jurnal Teknologi Informasi dan Ilmu Komputer, 9(4), 781–790.

Ihsan, M., & Rahmawaty, D. (2023). Analisis kinerja robot elektromekanis berbasis mikrokontroler tanpa intervensi manusia. Jurnal Rekayasa Elektro, 15(2), 102–109.

Itqan, M., Zulkarnain, & Siregar, M. (2018). Pengembangan aplikasi Android menggunakan metode research and development. Jurnal Media Infotama, 14(2), 55–63.

Kurniawan, D., Pratama, Y., & Santoso, H. (2021). Perbandingan kinerja basis data relasional dan NoSQL pada sistem skala menengah. Jurnal Core IT, 7(2), 65–72.

Lesmana, H., Nugroho, A., & Wibowo, S. (2021). Pengujian perangkat lunak menggunakan metode black box testing. Jurnal Pengujian Sistem Informasi, 4(1), 33–40.

Muhammad, F., & Wijanarko, T. (2025). Integrasi driver motor L298N dan sensor ultrasonik HC-SR04 pada sistem robot bergerak. Jurnal Elektronika dan Robotika, 11(1), 21–29.

Mustofa, A. (2017). Penerapan rule based reasoning pada sistem monitoring distribusi bahan bakar minyak secara online dan real-time. Jurnal Teknologi Informasi, 11(2), 99–108.

Object Management Group. (2017). OMG Unified Modeling Language (OMG UML), Version 2.5.1. https://www.omg.org/spec/UML/2.5.1/

Pramana, A., Putra, D., & Wijaya, K. (2020). Analisis penggunaan Firebase Realtime Database pada aplikasi mobile lintas platform. Jurnal Rekayasa Perangkat Lunak, 2(2), 80–88.

Putra, A., & Andriani, R. (2023). Pengembangan aplikasi mobile cross-platform menggunakan framework Flutter dan Dart. Jurnal Pengembangan TI, 7(3), 1230–1238.

Rahayu, S., Hidayat, T., & Wijaya, M. (2024). Pemanfaatan Flutter dan Dart untuk perangkat lunak skala lintas platform. Jurnal Ilmiah Teknologi Informasi, 12(1), 44–51.

Rahmawati, E., & Sujono, S. (2026). Sistem monitoring lingkungan berbasis IoT menggunakan ESP32. Jurnal Jaringan dan Sistem Komputer, 6(1), 15–24.

Santoso, B., & Irawan, C. (2020). Karakteristik dan penggunaan sensor elektronik dalam sistem otomasi. Jurnal Instrumentasi dan Sensor, 5(2), 110–118.

Saputro, D., & Wibowo, A. (2021). Implementasi Firebase Realtime Database pada aplikasi berbasis cloud. Jurnal Pengembangan IT, 6(3), 180–185.

Setiawan, R., Fauzi, A., & Almais, H. (2020). Konsep dasar perangkat lunak aplikasi dalam sistem informasi. Jurnal Sistem Informasi dan Teknologi, 8(2), 77–84.

Siswanti, S., & Suryono, S. (2016). Sistem pemantauan kadar gas amonia menggunakan pendekatan berbasis aturan (rule-based). Jurnal Fisika dan Aplikasinya, 12(3), 120–125.

Suharni, S., Susilowati, E., & Pakusadewa, F. (2023). Perancangan website Rumah Makan Ninik sebagai media promosi menggunakan Unified Modelling Language. Jurnal Rekayasa Informasi, 12(1), 1–12. https://ejournal.istn.ac.id/index.php/rekayasainformasi/article/view/1527

Sumarto, S. (2023). Pengembangan aplikasi point of sale berbasis mobile menggunakan framework Flutter dan pengujian black box testing. Jurnal Teknologi dan Sistem Informasi, 4(2), 90–98.

Susanto, H., & Mulyo, A. (2021). Penerapan metode research and development (R&D) dalam pengembangan produk teknologi informasi. Jurnal Teknologi dan Rekayasa, 9(1), 45–54.

Tamam, M. (2023). Manajemen pengolahan data sensor mentah pada perangkat keras mikrokontroler. Jurnal Teknik Komputer, 9(2), 112–119.

Widianto, E., Pratama, I., & Santoso, D. (2021). Peran mikrokontroler dalam membaca masukan sensor pada sistem elektronik modern. Jurnal Teknik Elektro, 13(1), 33–40.

Wayahdi, M. R., & Ruziq, F. (2023). Pemodelan sistem penerimaan anggota baru dengan Unified Modeling Language (UML) (Studi kasus: Programmer Association of Battuta). Jurnal Minfo Polgan, 12(1), 1514–1521. https://doi.org/10.33395/jmp.v12i1.12870

Wijaya, A., & Juliadi, R. (2021). Rancang bangun sistem pengendali robot pembersih lantai berbasis Android. Jurnal Algoritma dan Robotika, 3(2), 50–58.

Wijaya, H., Pratama, D., & Rahayu, S. (2024). Integrasi layanan Firebase dan framework Flutter dalam pengembangan aplikasi mobile terkini. Jurnal Ilmu Komputer dan Aplikasi, 15(1), 89–98.

Yoski, M., & Mukhaiyar, R. (2020). Prototipe robot pembersih lantai menggunakan sensor ultrasonik berbasis mikrokontroler. Jurnal UNP Elektronika, 9(1), 14–20.

Yuliza, E., & Kholifah, N. (2015). Rancang bangun robot pembersih lantai berbasis Arduino Uno dengan sensor ultrasonik. Jurnal Teknik Komputer AMIK BSI, 1(2), 150–158.

Yuniarto, D., Arifin, Z., & Mulyana, A. (2022). Implementasi ESP32 dalam sistem monitoring industri berbasis Internet of Things. Jurnal Otomasi dan Kendali, 7(2), 101–110.

Zidan, M., Hidayat, R., & Lesmana, H. (2022). Pengujian fungsionalitas aplikasi menggunakan black box testing dan teknik equivalence partitioning. Jurnal Sistem Informasi dan

Rekayasa Perangkat Lunak, 6(3), 200–209.