LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

Arsitektur Monitoring Jaringan Wifi Berbasis Agen Lokal Dengan Notifikasi Telegram Pada Kafe

Muhammad Yusup[1], Vany Septia Roza[2], Muhammad Rafi Sunarya[3], Wasis Haryono[4*]

[1]Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia
Email: [1]muhammad.yusupdw@gmail.com, [2]vanyseptiaroza781@gmail.com, [3]rafisunarya850@gmail.com, [4*]wasish@unpam.ac.id
(* : coressponding author)

Abstrak − Kafe membutuhkan jaringan WiFi yang stabil untuk melayani pelanggan dan menunjang transaksi kasir, namun pada usaha skala kecil gangguan jaringan umumnya baru diketahui setelah ada keluhan. Penelitian ini merancang dan membangun aplikasi monitoring jaringan WiFi berbasis web dengan notifikasi *real-time* di Cafe Formaggio menggunakan metode *Research and Development* dengan model *Prototype*. Kebaruan penelitian terletak pada arsitektur hibrida, yaitu agen ringan berbasis Termux pada telepon pintar Android di jaringan lokal yang memeriksa perangkat melalui ICMP setiap 60 detik dan mengirimkan hasilnya melalui API bertoken ke aplikasi Laravel pada *Virtual Private Server* (VPS), sehingga dasbor dapat diakses dari mana saja tanpa bergantung pada merek *router*, SNMP, atau server lokal. Notifikasi Telegram dikirim hanya ketika status perangkat berubah sehingga tidak terjadi pesan berulang. Pengujian *black box* menunjukkan 53 dari 53 skenario berjalan sesuai harapan, dan *User Acceptance Testing* (UAT) bersama Pemilik, Teknisi, dan Admin menilai aplikasi sangat layak digunakan. Verifikasi fungsi analisis *Quality of Service* (QoS) pada basis data uji menghasilkan indeks TIPHON 3,67 dengan kategori Memuaskan. Sistem ini menjadi alternatif monitoring jaringan yang murah dan mudah dioperasikan bagi usaha kecil.

Kata Kunci: Monitoring Jaringan; Agen Monitoring; Notifikasi Telegram; Quality of Service; Laravel

Abstract *− Cafes need stable WiFi to serve customers and support cashier transactions, yet in small businesses network disruptions are usually noticed only after complaints. This study designs and builds a web-based WiFi network monitoring application with real-time notification at Cafe Formaggio using the Research and Development method with the Prototype model. The novelty lies in a hybrid architecture: a lightweight Termux-based agent on an Android smartphone inside the local network checks devices via ICMP every 60 seconds and pushes the results through a token-protected API to a Laravel application on a Virtual Private Server (VPS), so the dashboard is accessible anywhere without depending on a router vendor, SNMP, or an on-premise server. Telegram notifications are sent only when a device changes status, preventing repeated alerts. Black box testing shows that 53 of 53 scenarios behave as expected, and User Acceptance Testing (UAT) with the owner, technician, and administrator judged the application highly feasible. Verification of the Quality of Service (QoS) analysis function on a test database yields a TIPHON index of 3.67 (Satisfactory). The system offers a low-cost, easy-to-operate network monitoring alternative for small businesses.*

Keywords: *Network Monitoring; Monitoring Agent; Telegram Notification; Quality of Service; Laravel*

1. PENDAHULUAN

Jaringan WiFi telah menjadi fasilitas utama pada usaha kuliner seperti kafe. Selain digunakan pelanggan, jaringan yang sama menopang kegiatan operasional seperti transaksi kasir dan pembayaran nontunai. Kualitas layanan jaringan dapat dinilai secara kuantitatif melalui parameter *Quality of Service* (QoS) seperti *delay*, *jitter*, dan *packet loss* yang umumnya dibandingkan dengan standar TIPHON (Hidayat et al., 2024; Saleh et al., 2022; Simargolang & Widarma, 2022).

Cafe Formaggio di Kota Tangerang menyediakan layanan WiFi bagi pelanggan dan pegawai. Berdasarkan wawancara dengan *Floor Manager* pada September 2026, gangguan seperti koneksi lambat, tidak stabil, atau terputus baru diketahui setelah ada laporan pelanggan atau pengecekan manual dengan perintah *ping*. Tidak ada pencatatan waktu, durasi, maupun frekuensi gangguan, sehingga kualitas jaringan tidak dapat dievaluasi berdasarkan data dan penanganan gangguan menjadi lambat.

Berbagai penelitian telah mengembangkan sistem monitoring jaringan. Pratama dan Sutomo (2026) membaca data *router* MikroTik melalui RouterOS API untuk menghitung QoS, sedangkan Yuazijah et al. (2024) memanfaatkan The Dude. Penelitian lain menggabungkan SNMP atau

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

platform monitoring dengan notifikasi Telegram, seperti SNMP dan LibreNMS (Fergina et al., 2024; Wahyuriyanto & Yulianton, 2024; Wijaya & Achlaq, 2023), LibreNMS multikanal (Fachrurrozi et al., 2023), Prometheus dan Grafana (Hakim et al., 2025), serta Zabbix dengan pemeriksaan ICMP yang menurunkan rata-rata waktu henti dari sekitar 30 menit menjadi sekitar 3 menit (Febriana et al., 2026). Dasbor kinerja WLAN berbasis QoS juga terbukti membantu keputusan pemeliharaan (Nuryani et al., 2024), dan sistem monitoring berbasis web telah diterapkan untuk pelaporan pekerjaan instalasi fiber optik (Rozi et al., 2026) serta pemantauan pemeliharaan aset (Setiawan et al., 2026).

Penelitian tersebut umumnya bergantung pada merek perangkat tertentu, perangkat yang mendukung SNMP, atau platform yang membutuhkan server di jaringan lokal. Kondisi ini kurang sesuai bagi kafe yang memakai perangkat jaringan konsumen, tidak memiliki server khusus, dan dikelola pegawai nonteknis. Selain itu, aplikasi di server *cloud* tidak dapat menjangkau alamat IP privat perangkat kafe yang berada di balik NAT. Oleh karena itu, penelitian ini menawarkan kebaruan berupa **arsitektur monitoring hibrida**: agen ringan berbasis Termux pada telepon pintar Android di jaringan lokal memeriksa perangkat melalui ICMP, lalu mengirimkan hasilnya melalui API bertoken ke aplikasi Laravel pada *Virtual Private Server* (VPS). Notifikasi Telegram dikirim berdasarkan transisi status sehingga teknisi hanya menerima pesan saat perangkat terputus dan pulih.

Penelitian ini bertujuan untuk: (1) merancang dan membangun aplikasi monitoring jaringan WiFi berbasis web dengan arsitektur agen lokal dan server *cloud*; (2) menerapkan notifikasi Telegram berbasis transisi status; (3) menguji fungsionalitas dan penerimaan sistem melalui pengujian *black box* dan *User Acceptance Testing* (UAT); serta (4) memverifikasi kemampuan aplikasi mengolah data monitoring menjadi penilaian QoS berdasarkan standar TIPHON.

2. METODE

2.1 Tahapan Penelitian

Penelitian menggunakan metode *Research and Development* (R&D) dengan model *Prototype* (Pressman & Maxim, 2015), yang banyak dipakai pada pengembangan sistem berbasis web, seperti aplikasi pemesanan ruang (Fadhillah et al., 2026), *Learning Management System* (Israf et al., 2026), dan sistem berbasis Laravel (Kadim et al., 2025; Nugroho et al., 2024). Model ini dipilih karena kebutuhan mitra berkembang selama penelitian. Penelitian berlangsung pada 15 September sampai 5 November 2026 dengan tahapan berikut:

a. Komunikasi: observasi lokasi dan wawancara dengan pengelola kafe untuk mengidentifikasi masalah dan kebutuhan.

b. Perencanaan cepat: menentukan ruang lingkup, peran pengguna, parameter QoS, dan teknologi.

c. Pemodelan cepat: menyusun *use case*, *activity*, *sequence*, dan *class diagram*, ERD, serta rancangan antarmuka.

d. Konstruksi prototipe: membangun aplikasi web Laravel, agen monitoring Termux, dan integrasi Telegram.

e. Penyerahan dan umpan balik: demonstrasi kepada mitra, penyempurnaan fitur (notifikasi Telegram, *reset uptime*, dan pengelolaan data gangguan), *deployment* ke VPS, serta pengujian.

2.2 Lokasi, Partisipan, dan Pengumpulan Data

Penelitian dilakukan di Cafe Formaggio, Kota Tangerang, yang memakai koneksi ISP 100 Mbps, *router* Huawei sebagai *gateway*, *switch*, dan *access point* TP-Link. Agen dijalankan pada telepon pintar Android 15 dengan aplikasi Termux. Partisipan dipilih secara *purposive sampling*, yaitu pihak yang terlibat langsung dalam penggunaan dan pengelolaan jaringan kafe, seperti pada Tabel 1. Data dikumpulkan melalui teknik berikut:

a. Observasi: mengamati jenis dan posisi perangkat jaringan, jangkauan sinyal di area *indoor*, kasir, dan pelanggan, serta kendala koneksi pada jam operasional.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

b. Wawancara: dilakukan dengan *Floor Manager* untuk menggali tujuan penggunaan jaringan, kendala yang sering terjadi, cara mengetahui gangguan, dan kebutuhan sistem monitoring.

c. Dokumentasi: foto perangkat dan lokasi, tangkapan layar aplikasi, serta data hasil monitoring yang tersimpan pada basis data.

d. Studi pustaka: jurnal dan dokumentasi terkait monitoring jaringan, QoS, notifikasi *real-time*, dan pengembangan aplikasi web.

Tabel 1. Partisipan Penelitian

Peran	Keterlibatan	Jumlah
Floor Manager	Narasumber wawancara dan evaluasi prototipe	1
Pemilik	UAT peran Pemilik (dasbor dan laporan)	1
Teknisi	UAT peran Teknisi (target, gangguan, notifikasi)	1
Pegawai/Admin	UAT peran Admin (dasbor, laporan, kelola pengguna)	1

2.3 Pengukuran Quality of Service

Agen menjalankan perintah *ping -c 3 -W 3* terhadap setiap perangkat, yaitu tiga paket ICMP dengan batas waktu tiga detik (Postel, 1981), lalu mengambil *round-trip time* (RTT) rata-rata sebagai *latency*. Setiap pemeriksaan disimpan sebagai satu sampel. Parameter dihitung dengan persamaan (1) sampai (4):

$$Latency = \frac{\sum D_i}{n} \quad (1)$$

$$Jitter = \frac{\sum |D_i - D_{i-1}|}{n - 1} \quad (2)$$

$$Packet\ Loss = \frac{N_{offline}}{N_{total}} \times 100\% \quad (3)$$

$$Availability = \frac{N_{online}}{N_{total}} \times 100\% \quad (4)$$

dengan *Di* adalah *latency* sampel ke-*i*, *n* jumlah sampel *online*, dan *N* jumlah pemeriksaan. Karena setiap pemeriksaan diperlakukan sebagai satu sampel, *packet loss* dihitung sebagai rasio pemeriksaan gagal terhadap seluruh pemeriksaan. Nilai setiap parameter dikategorikan menurut standar TIPHON (ETSI, 1999) pada Tabel 2, lalu indeksnya dirata-ratakan menjadi indeks QoS.

Tabel 2. Kategori QoS Menurut Standar TIPHON

Kategori	Delay	Jitter	Packet Loss	Indeks
Sangat Bagus	< 150 ms	0 ms	0 - 2%	4
Bagus	150 - 300 ms	0 - 75 ms	3 - 14%	3
Sedang	300 - 450 ms	75 - 125 ms	15 - 24%	2
Jelek	> 450 ms	125 - 225 ms	≥ 25%	1

2.4 Metode Pengujian

Pengujian fungsional menggunakan metode *black box* dengan skenario masukan valid dan tidak valid untuk setiap modul (Fadhillah et al., 2026; Zidan et al., 2022). Skenario dijalankan secara

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

otomatis menggunakan peramban *headless* terhadap basis data uji berisi tiga perangkat dan akun untuk setiap peran. Pengujian *white box* dengan teknik *basis path* diterapkan pada fungsi penentuan dan pengiriman notifikasi, dilengkapi pengujian logika notifikasi dengan rangkaian perubahan status. Penerimaan pengguna dievaluasi melalui UAT secara kualitatif: setiap responden menjalankan skenario tugas sesuai perannya (Putro et al., 2024), kemudian memberikan penilaian kelayakan aplikasi.

3. ANALISA DAN PEMBAHASAN

3.1 Analisis Sistem Berjalan dan Kebutuhan

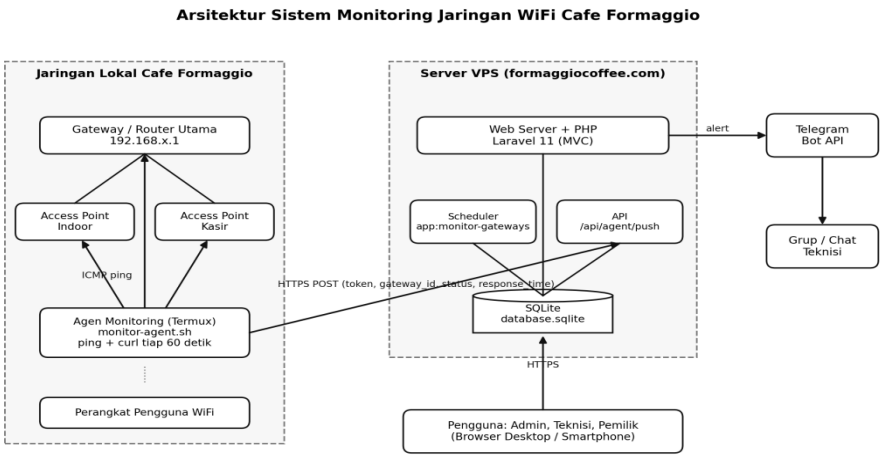
Hasil observasi dan wawancara menunjukkan bahwa pemantauan jaringan di Cafe Formaggio masih dilakukan secara manual. Kondisi ini menimbulkan empat kelemahan utama, yaitu gangguan terlambat diketahui karena bergantung pada laporan pengguna, tidak ada data histori koneksi, tidak ada ukuran kualitas jaringan yang objektif, dan tidak ada pemberitahuan otomatis kepada penanggung jawab jaringan.

Berdasarkan kelemahan tersebut, kebutuhan fungsional yang dirumuskan meliputi autentikasi pengguna, dasbor monitoring dengan pembaruan otomatis, histori koneksi, laporan per periode yang dapat dicetak, pengelolaan data gangguan, pengelolaan akun, pengelolaan perangkat target, cek koneksi manual, *reset uptime*, pengaturan dan pengiriman notifikasi Telegram, serta API penerima data agen. Kebutuhan nonfungsional meliputi akses melalui peramban komputer maupun telepon pintar, pembatasan hak akses per peran, serta perlindungan API dengan token dan pembatasan laju.

3.2 Arsitektur Sistem

Arsitektur sistem ditunjukkan pada Gambar 1. Sistem terdiri atas tiga bagian, yaitu agen monitoring di jaringan lokal kafe, aplikasi web pada VPS yang diakses melalui domain formaggiocoffee.com, dan Telegram Bot API. Agen memeriksa setiap perangkat terdaftar setiap 60 detik, lalu mengirimkan status (*online* atau *offline*) dan waktu respons melalui HTTP POST ke *endpoint* /api/agent/push.

Server memvalidasi token agen yang disimpan pada berkas konfigurasi lingkungan, memvalidasi isi data, membatasi laju 60 permintaan per menit, menyimpan hasil ke tabel *connection_logs*, lalu membandingkan status terbaru dengan status sebelumnya untuk menentukan perlu tidaknya notifikasi. Model *push* dipilih karena server *cloud* tidak dapat menjangkau alamat IP privat perangkat kafe yang berada di balik NAT. Dengan agen di jaringan lokal, pemeriksaan tetap dilakukan dari dalam jaringan, sedangkan dasbor dan notifikasi dapat diakses dari mana saja. Dasbor memperbarui data setiap 5 detik melalui permintaan asinkron ke server.

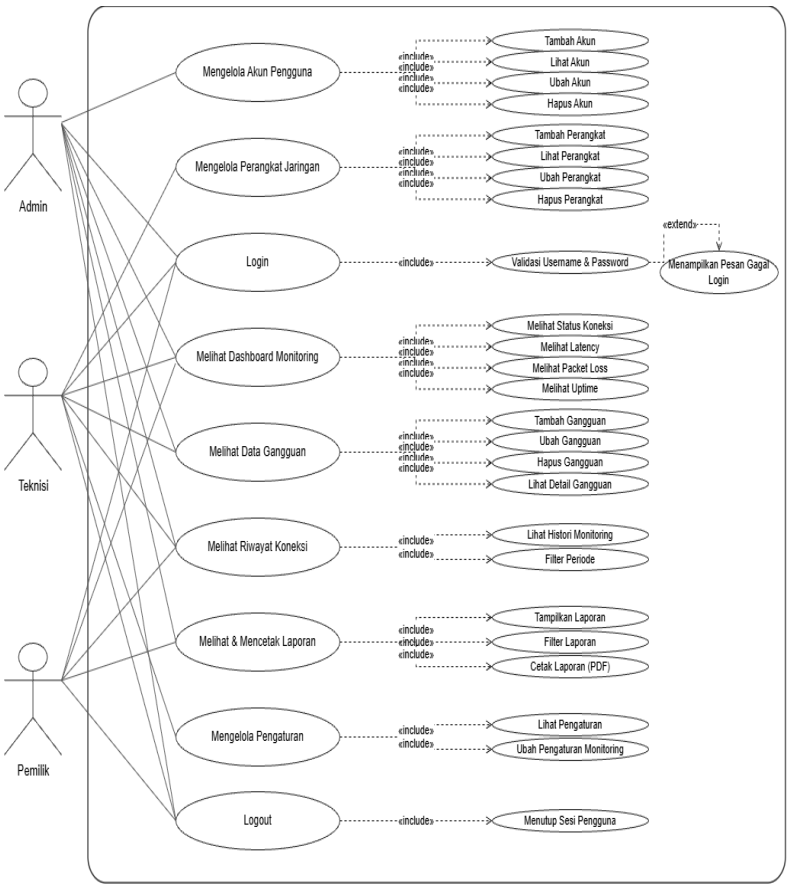


Gambar 1. Arsitektur Sistem Monitoring Hibrida Agen Lokal dan Server VPS

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

3.3 Use Case Diagram

Use case diagram pada Gambar 2 menggambarkan interaksi tiga aktor manusia dengan sistem. Seluruh aktor dapat *login*, *logout*, melihat dasbor monitoring, dan melihat histori koneksi. *Use case* laporan diperuntukkan bagi Admin dan Pemilik, pengelolaan akun bagi Admin, sedangkan pengelolaan target, data gangguan, *reset uptime*, dan pengaturan notifikasi diperuntukkan bagi Teknisi. Pembatasan ini diterapkan melalui *middleware* berbasis peran yang menolak akses tidak sah dengan HTTP 403.



Gambar 2. *Use Case Diagram* Aplikasi Monitoring Jaringan WiFi

Rincian hak akses setiap aktor, termasuk agen sebagai aktor sistem, ditunjukkan pada Tabel 3.

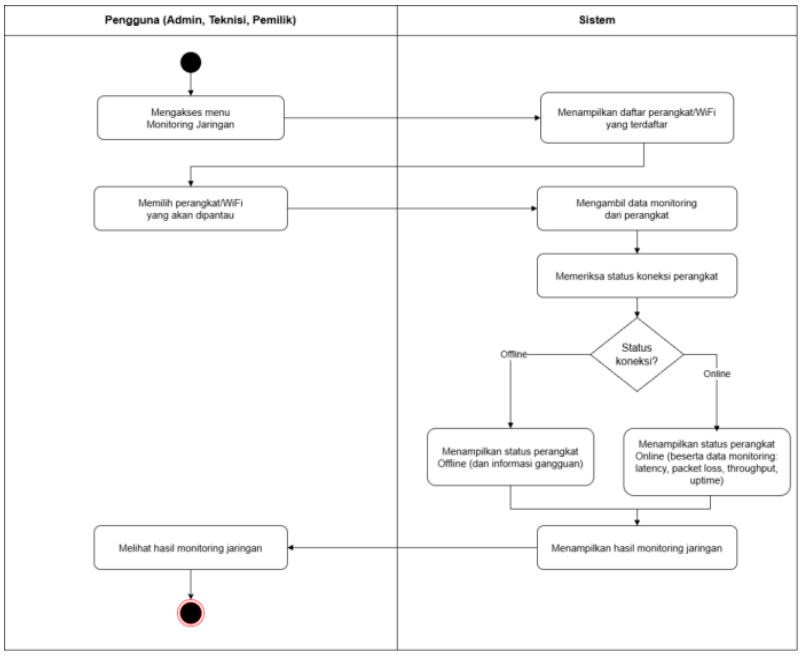
Tabel 3. Aktor dan Hak Akses Sistem

Aktor	Hak Akses Utama
Admin	Dasbor, histori, laporan, melihat data gangguan, kelola akun pengguna
Teknisi	Dasbor, histori, kelola *gateway*/target, cek koneksi, *reset uptime*, kelola data gangguan, pengaturan Telegram
Pemilik	Dasbor, histori, laporan
Agen (sistem)	Mengirim hasil pemeriksaan melalui API dengan token

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

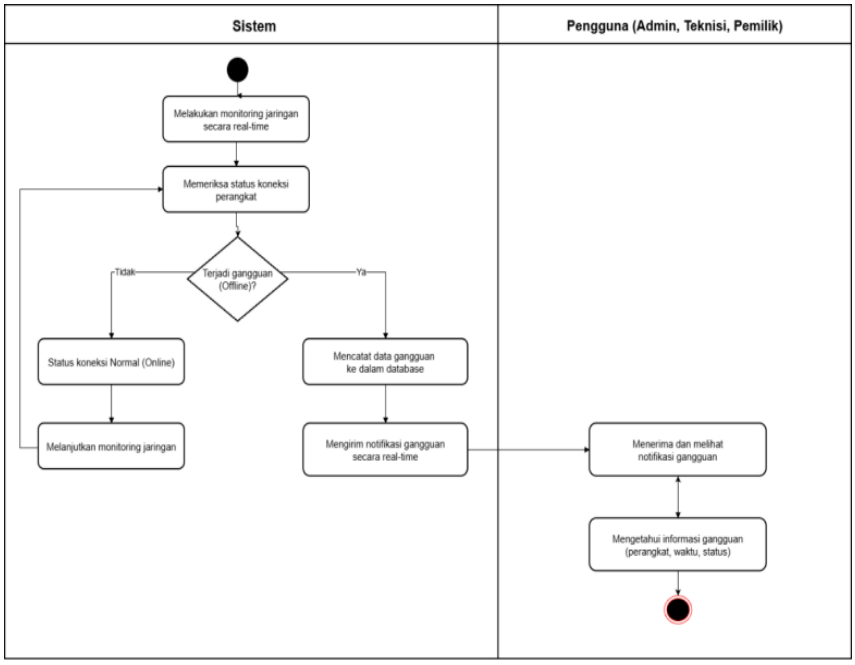
3.4 Activity Diagram

Gambar 3 menunjukkan aktivitas monitoring jaringan. Pengguna memilih perangkat yang akan dipantau, sistem mengambil data monitoring terbaru, memeriksa status koneksi, lalu menampilkan informasi gangguan apabila perangkat *offline* atau menampilkan *latency*, *packet loss*, dan *uptime* apabila *online*.



Gambar 3. *Activity Diagram* Monitoring Jaringan WiFi

Gambar 4 menunjukkan aktivitas notifikasi gangguan. Ketika gangguan terdeteksi, sistem mencatat data gangguan dan mengirim notifikasi kepada teknisi, sedangkan pada kondisi normal sistem melanjutkan pemantauan.

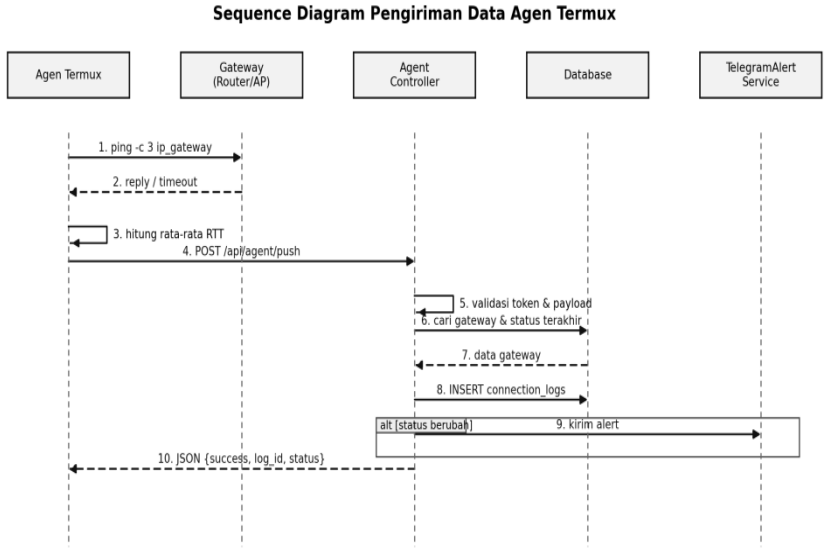


Gambar 4. *Activity Diagram* Notifikasi Gangguan Jaringan

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

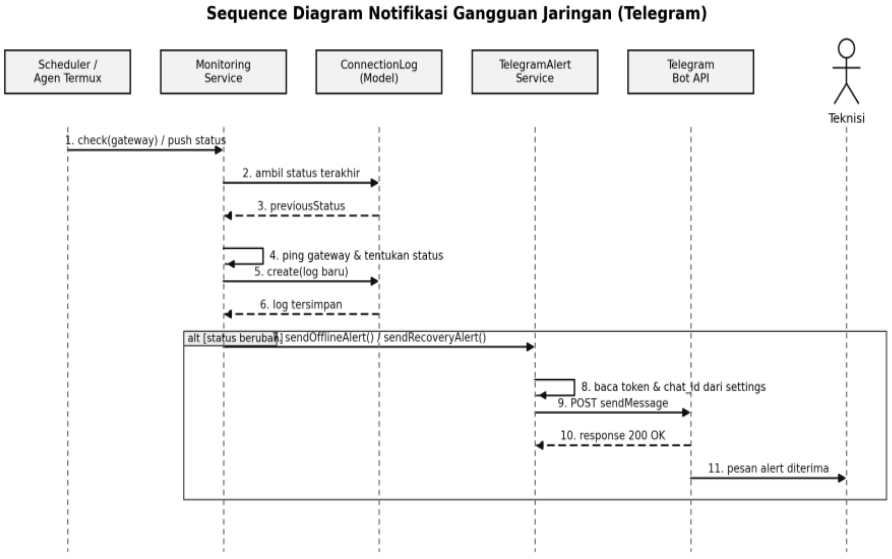
3.5 Sequence Diagram

Gambar 5 memperlihatkan urutan pengiriman data oleh agen. Agen mengirim tiga paket *ping* ke *gateway*, menghitung RTT rata-rata, lalu mengirim data ke AgentController. *Controller* memvalidasi token dan isi data, mencari data *gateway*, menyimpan log koneksi, dan memicu TelegramAlertService apabila status berubah.



Gambar 5. *Sequence diagram* pengiriman data agen Termux

Gambar 6 memperlihatkan alur notifikasi. Layanan membaca *bot token* dan *chat id* dari pengaturan, mengirim pesan ke Telegram Bot API dengan batas waktu 10 detik, dan menangkap kegagalan jaringan agar penyimpanan data monitoring tidak terhenti ketika Telegram tidak dapat dihubungi.



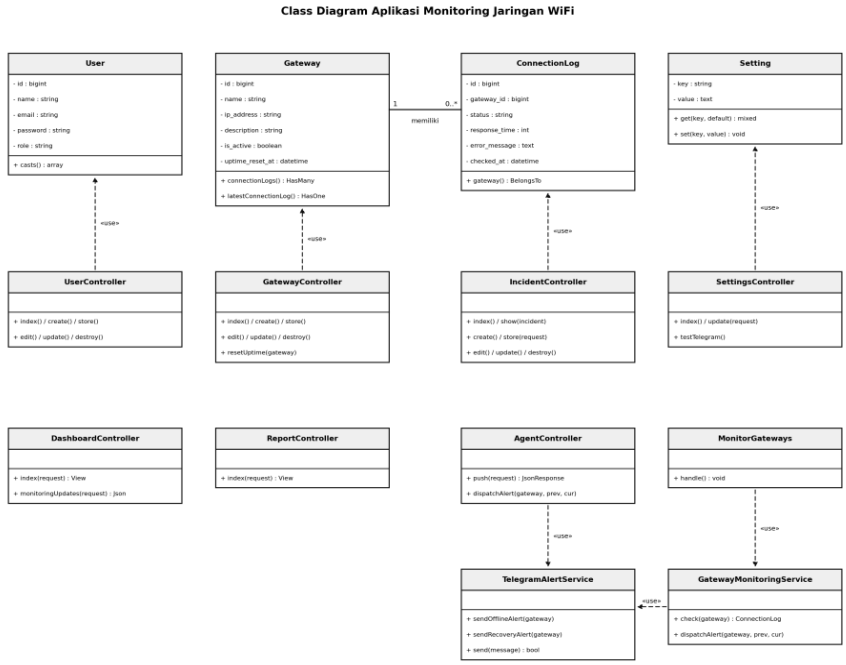
Gambar 6. *Sequence Diagram* Notifikasi Gangguan Jaringan Melalui Telegram

3.6 Class Diagram

Aplikasi dibangun dengan *framework* Laravel 11 yang berorientasi objek dengan pola *Model-View-Controller*. *Class diagram* pada Gambar 7 terdiri atas empat model (User, Gateway,

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

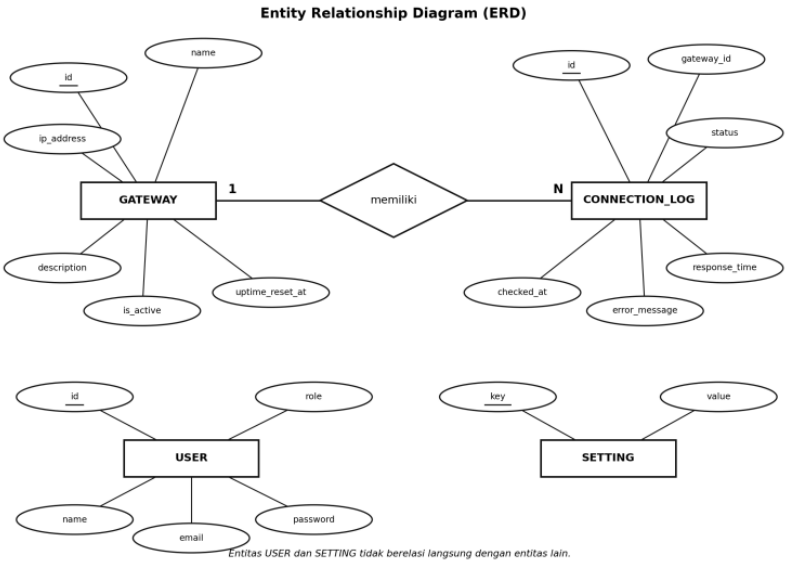
ConnectionLog, dan Setting), *controller* untuk setiap modul (Dashboard, Gateway, Incident, Report, User, Settings, dan Agent), serta dua kelas layanan, yaitu GatewayMonitoringService untuk pemeriksaan terjadwal dan cek manual, dan TelegramAlertService untuk pengiriman notifikasi.



Gambar 7. *Class Diagram* Aplikasi Monitoring Jaringan WiFi

3.7 *Entity Relationship Diagram*

Basis data menggunakan SQLite dan terdiri atas empat tabel. *Entity Relationship Diagram* pada Gambar 8 menunjukkan relasi satu ke banyak antara *gateway* dan *connection_log*, sedangkan tabel *users* dan *settings* berdiri sendiri. Ringkasan struktur tabel ditunjukkan pada Tabel 4. Kolom *uptime_reset_at* memungkinkan teknisi memulai ulang perhitungan *uptime* setelah perbaikan tanpa menghapus data histori.



Gambar 8. *Entity Relationship Diagram*

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

Tabel 4. Struktur Tabel Basis Data

Tabel	Atribut Utama
users	id, name, email, password, role
gateways	id, name, ip_address, description, is_active, uptime_reset_at
connection_logs	id, gateway_id (FK), status, response_time, error_message, checked_at
settings	key, value (*bot token*, *chat id*, status notifikasi)

3.8 Logika Notifikasi Berbasis Transisi Status

Agar teknisi tidak menerima pesan berulang pada setiap pemeriksaan, notifikasi hanya dikirim ketika status berubah, seperti ditunjukkan pada Tabel 5. Pendekatan ini berbeda dari notifikasi yang dikirim pada setiap siklus pemeriksaan, sehingga satu kejadian gangguan menghasilkan tepat satu pesan gangguan dan satu pesan pemulihan.

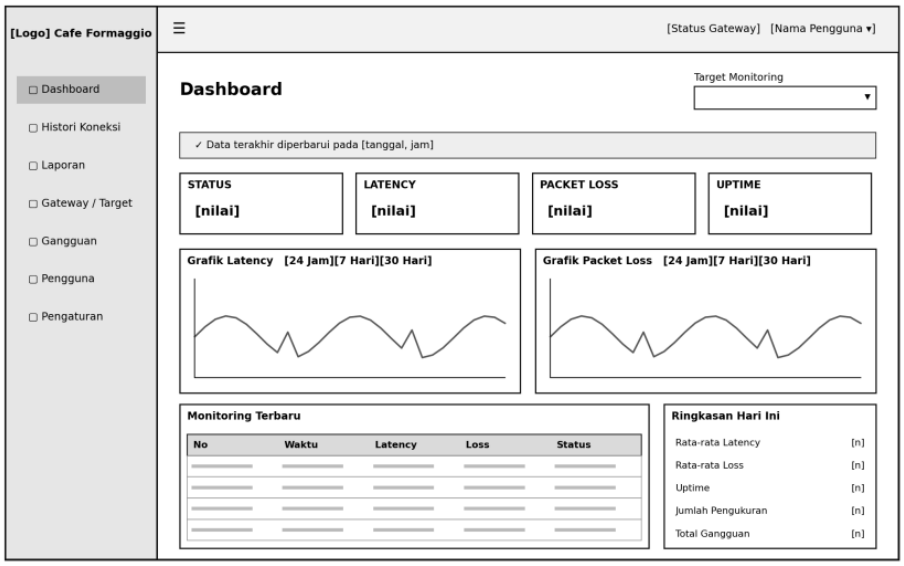
Tabel 5. Transisi Status dan Tindakan Notifikasi

Status Sebelumnya	Status Baru	Tindakan
(belum ada data)	*online*	Tidak mengirim notifikasi
(belum ada data)	*offline*	Kirim GATEWAY OFFLINE
online	*online*	Tidak mengirim notifikasi
online	*offline*	Kirim GATEWAY OFFLINE
offline	*offline*	Tidak mengirim (mencegah pesan berulang)
offline	*online*	Kirim GATEWAY ONLINE (pemulihan)

3.9 Perancangan Antarmuka

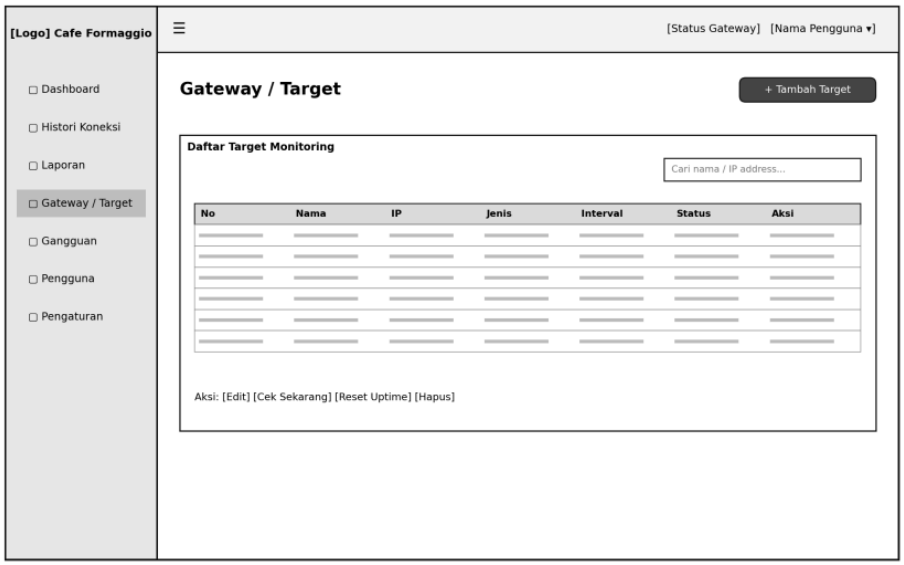
Rancangan antarmuka dibuat dalam bentuk *wireframe* sebelum implementasi dan menjadi acuan pembuatan prototipe. Gambar 9 menunjukkan rancangan halaman dasbor yang terdiri atas menu samping, kartu ringkasan status, *latency*, *packet loss*, dan *uptime*, grafik *latency* dan *packet loss* dengan pilihan periode 24 jam, 7 hari, dan 30 hari, tabel monitoring terbaru, serta ringkasan harian.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401



Gambar 9. Rancangan Antarmuka Halaman Dasbor

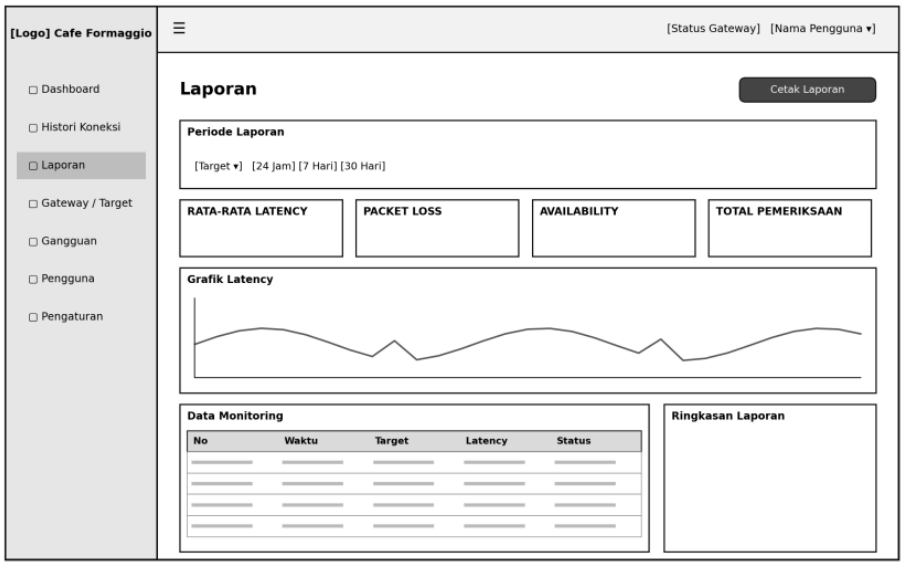
Gambar 10 menunjukkan rancangan halaman *Gateway*/Target bagi Teknisi, berisi tabel perangkat target beserta alamat IP dan status, tombol tambah target, serta tombol aksi untuk cek koneksi, *reset uptime*, ubah, dan hapus pada setiap baris.



Gambar 10. Rancangan Antarmuka Halaman *Gateway*/Target

Gambar 11 menunjukkan rancangan halaman laporan bagi Admin dan Pemilik, berisi pilihan periode dan perangkat, ringkasan statistik *latency*, *packet loss*, dan *uptime*, grafik, serta tabel data monitoring yang dapat dicetak.

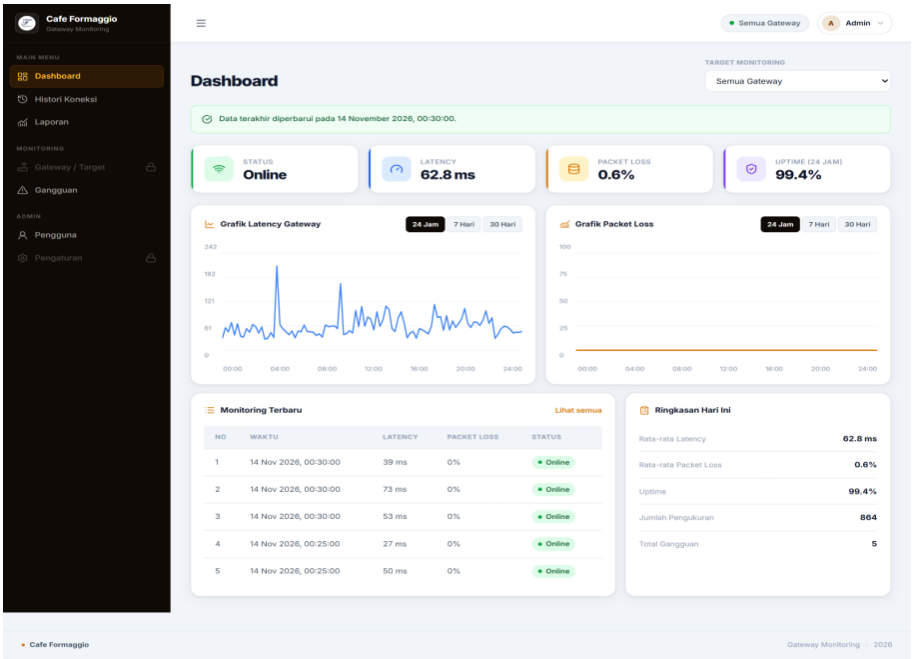
LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401



Gambar 11. Rancangan Antarmuka Halaman Laporan

3.10 Implementasi Sistem

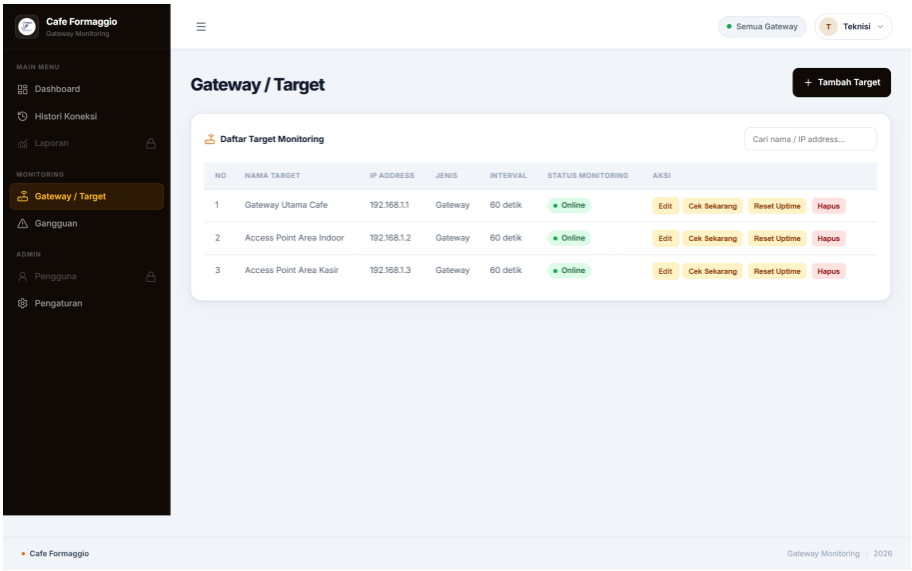
Prototipe diimplementasikan sesuai rancangan pada Gambar 9 sampai Gambar 11 dan dijalankan pada VPS berbasis Linux yang dapat diakses melalui domain formaggiocoffee.com. Gambar 12 menunjukkan halaman dasbor hasil implementasi rancangan Gambar 9, yang menampilkan status, rata-rata *latency*, *packet loss*, dan *uptime* per periode, grafik, serta tabel monitoring terbaru yang diperbarui otomatis. Untuk menjaga kinerja tampilan, grafik dibatasi maksimal 100 titik yang diambil merata dari data periode yang dipilih.



Gambar 12. Halaman Dasbor Monitoring

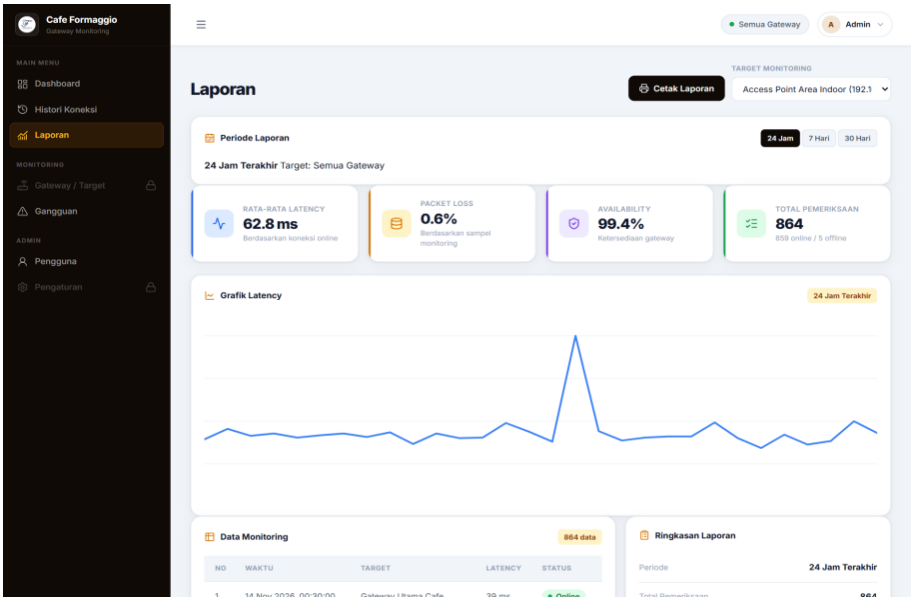
Gambar 13 menunjukkan halaman *Gateway*/Target hasil implementasi rancangan Gambar 10, yang digunakan Teknisi untuk menambah, mengubah, dan menghapus perangkat target, menjalankan cek koneksi seketika, serta melakukan *reset uptime*.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401



Gambar 13. Halaman *Gateway*/Target pada Peran Teknisi

Gambar 14 menunjukkan halaman laporan hasil implementasi rancangan Gambar 11, yang menyajikan statistik per periode dan per perangkat serta dapat dicetak oleh Admin dan Pemilik. Antarmuka juga dirancang responsif sehingga dapat diakses melalui telepon pintar. Setelah berjalan di lokasi, sistem mencatat *gateway* kafe tidak dapat dijangkau pada 5 November 2026 pukul 16.45.15 WIB dan kembali dapat dijangkau pukul 16.45.47 WIB, dan setiap perubahan tersebut menghasilkan satu pesan Telegram sesuai logika pada Tabel 5.



Gambar 14. Halaman Laporan Monitoring

3.11 Pengujian Black Box

Pengujian *black box* dilakukan terhadap 53 skenario pada 13 kelompok fungsi. Rekapitulasi hasil ditunjukkan pada Tabel 6, sedangkan contoh skenario pada fungsi utama dan aspek keamanan ditunjukkan pada Tabel 7.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

Tabel 6. Rekapitulasi Pengujian *Black Box*

Kelompok Fungsi	Skenario	Berhasil	Kelompok Fungsi	Skenario	Berhasil
Autentikasi	5	5	Gateway/Target	8	8
Dasbor	4	4	Pengaturan Telegram	2	2
Histori koneksi	2	2	Tampilan responsif	1	1
Data gangguan	1	1	API agen	4	4
Laporan	2	2	Keamanan	2	2
Kelola pengguna	6	6	Kelola data gangguan	11	11
Hak akses	5	5	**Total**	**53**	**53 (100%)**

Tabel 7. Contoh Skenario Pengujian *Black Box*

Kode	Skenario	Data Uji	Hasil yang Diharapkan	Status
BB-01	Mengakses dasbor tanpa *login*	URL /dashboard	Diarahkan ke halaman *login*	Berhasil
BB-02	*Login* dengan *password* salah	Email terdaftar, *password* salah	Muncul pesan kesalahan, tetap di halaman *login*	Berhasil
BB-04	*Login* akun Admin yang valid	Email dan *password* benar	Masuk ke halaman dasbor	Berhasil
BB-19	Admin mengakses halaman target	URL /targets	Akses ditolak (HTTP 403)	Berhasil
BB-23	Menambah target dengan IP tidak valid	IP 999.1.1.1	Data ditolak, muncul pesan validasi	Berhasil
BB-37	Agen mengirim data valid	Token benar, *gateway* 1, *online*, 37 ms	HTTP 200, data tersimpan	Berhasil
BB-38	Agen mengirim token salah	Token tidak valid	HTTP 401 *Unauthorized*	Berhasil
BB-39	Agen mengirim status tidak dikenal	status = rusak	HTTP 422 *Invalid payload*	Berhasil
BB-40	Agen mengirim *gateway* tidak terdaftar	gateway_id = 9999	HTTP 404 *Gateway not found*	Berhasil
BB-41	Pembatasan laju API agen	62 permintaan dalam 1 menit	Permintaan ke-61 dst. ditolak (HTTP 429)	Berhasil
BB-42	Pembatasan percobaan *login*	7 percobaan gagal dalam 1 menit	Percobaan ke-6 dst. ditolak (HTTP 429)	Berhasil

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

3.12 Pengujian White Box dan Logika Notifikasi

Pengujian *white box* dengan teknik *basis path* dilakukan pada fungsi *dispatchAlert()* dengan *cyclomatic complexity* V(G) = 4 dan fungsi *send()* dengan V(G) = 3. Permintaan ke Telegram disimulasikan sehingga kondisi berhasil, token salah, dan putus koneksi dapat diuji. Hasil pada Tabel 8 menunjukkan sembilan kasus uji berhasil dan seluruh jalur independen dieksekusi minimal satu kali.

Tabel 8. Hasil Pengujian *White Box* (*Basis Path*)

Kode	Fungsi	Data Uji	Hasil	Status
WB-01	dispatchAlert()	*previous* = *online*, *current* = *online*	Tidak ada pemanggilan	Berhasil
WB-02	dispatchAlert()	*previous* = *online*, *current* = *offline*	sendOfflineAlert	Berhasil
WB-03	dispatchAlert()	*previous* = *offline*, *current* = *online*	sendRecoveryAlert	Berhasil
WB-04	dispatchAlert()	*previous* = *null*, *current* = *online*	Tidak ada pemanggilan	Berhasil
WB-05	send()	Notifikasi dinonaktifkan	*false*, tanpa permintaan	Berhasil
WB-06	send()	Notifikasi aktif, API merespons 200	*true*	Berhasil
WB-07	send()	Notifikasi aktif, token salah (401)	*false*	Berhasil
WB-08	send()	Notifikasi aktif, koneksi gagal	*false*	Berhasil
WB-09	send()	*Chat id* kosong	*false*	Berhasil

Pengujian logika notifikasi dilakukan dengan mengirimkan rangkaian status ke API agen untuk satu *gateway* uji. Hasil pada Tabel 9 menunjukkan notifikasi hanya dikirim saat terjadi perubahan status, tidak dikirim berulang ketika perangkat tetap *offline*, dan tidak dikirim saat notifikasi dinonaktifkan.

Tabel 9. Pengujian Logika Notifikasi Telegram

Kode	Skenario	Harapan	Hasil	Status
NT-01	Status pertama kali *online*	Tidak ada notifikasi	0 pesan	Berhasil
NT-02	Status berubah *online* ke *offline*	1 notifikasi GATEWAY OFFLINE	1 pesan	Berhasil
NT-03	Status tetap *offline* pada pemeriksaan berikutnya	Tidak ada notifikasi berulang	0 pesan	Berhasil
NT-04	Status berubah *offline* ke *online*	1 notifikasi GATEWAY ONLINE	1 pesan	Berhasil
NT-05	Status berubah ke *offline* saat notifikasi nonaktif	Tidak ada notifikasi	0 pesan	Berhasil

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

3.13 User Acceptance Testing

UAT melibatkan tiga responden sesuai Tabel 1, yaitu Pemilik, Teknisi, dan Pegawai/Admin. Setiap responden menjalankan skenario sesuai perannya seperti pada Tabel 10, kemudian memberikan penilaian terhadap kelayakan aplikasi.

Tabel 10. Skenario *User Acceptance Testing* Per Peran

Peran	Skenario yang Dijalankan
Pemilik	Login, melihat dasbor status, *latency*, *packet loss*, dan *uptime*, melihat histori koneksi, menampilkan dan mencetak laporan per periode
Teknisi	Menambah dan mengubah perangkat target, menjalankan cek koneksi, melakukan *reset uptime*, mengelola data gangguan, mengatur dan menerima notifikasi Telegram
Pegawai/Admin	Melihat dasbor dan histori, menampilkan laporan, mengelola akun pengguna, memastikan pembatasan hak akses per peran

Ketiga responden menilai aplikasi **sangat layak** digunakan untuk memantau jaringan WiFi Cafe Formaggio. Penilaian diberikan secara kualitatif setelah responden menjalankan skenario sesuai perannya, sehingga belum dinyatakan dalam skor atau persentase.

3.14 Analisis Quality of Service

Fungsi analisis QoS diverifikasi menggunakan basis data uji berisi tiga perangkat dengan data pemeriksaan berinterval 5 menit selama 7 hari (7 sampai 14 November 2026), yaitu 2.014 sampai 2.015 pemeriksaan per perangkat. Parameter dihitung dengan persamaan (1) sampai (4) dan dinilai menurut Tabel 2. Hasilnya ditunjukkan pada Tabel 11.

Tabel 11. Hasil Verifikasi Analisis Qos Pada Basis Data Uji

Perangkat	Availability (%)	Delay (ms)	Jitter (ms)	Packet Loss (%)	Indeks
Gateway Utama	99,31	55,0	22,1	0,69	3,67
AP Area Indoor	98,76	72,0	24,7	1,24	3,67
AP Area Kasir	98,71	62,6	20,9	1,29	3,67

Ketiga perangkat uji memperoleh kategori *delay* Sangat Bagus (indeks 4), *jitter* Bagus (indeks 3), dan *packet loss* Sangat Bagus (indeks 4), sehingga indeks QoS bernilai 3,67 dengan kategori Memuaskan. Hasil ini membuktikan bahwa data yang dikumpulkan aplikasi dapat diolah menjadi penilaian QoS sesuai standar TIPHON. Karena berasal dari basis data uji, nilai tersebut menggambarkan kemampuan fungsi analisis aplikasi, bukan kualitas jaringan aktual Cafe Formaggio.

3.15 Pembahasan

Dibandingkan penelitian yang membaca data melalui API *router* MikroTik (Pratama & Sutomo, 2026), sistem ini tidak bergantung pada merek perangkat karena hanya membutuhkan respons ICMP dari perangkat target. Dibandingkan pendekatan berbasis SNMP dan platform monitoring seperti LibreNMS, Grafana, dan Zabbix (Fachrurrozi et al., 2023; Febriana et al., 2026; Hakim et al., 2025; Wahyuriyanto & Yulianton, 2024), sistem ini tidak memerlukan server di jaringan lokal karena peran pemeriksa dijalankan oleh telepon pintar, sementara aplikasi dan data berada di VPS. Sejalan dengan Rozi et al. (2026) dan Setiawan et al. (2026), digitalisasi pemantauan

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

berbasis web terbukti mempercepat pelaporan dan pengambilan tindakan dibandingkan pencatatan manual.

Seperti penelitian Wijaya dan Achlaq (2023), Hakim et al. (2025), dan Febriana et al. (2026), notifikasi Telegram terbukti mempercepat informasi gangguan kepada teknisi. Perbedaannya, penelitian ini menerapkan notifikasi berbasis transisi status yang telah diverifikasi melalui pengujian *basis path* dan pengujian logika notifikasi, sehingga satu kejadian gangguan hanya menghasilkan satu pesan gangguan dan satu pesan pemulihan.

Penelitian ini memiliki beberapa keterbatasan. Pertama, *throughput* belum diukur sehingga indeks QoS hanya mencakup *delay*, *jitter*, dan *packet loss*. Kedua, *jitter* dihitung dari selisih *latency* antarpemeriksaan berjarak 60 detik sehingga nilainya merupakan pendekatan. Ketiga, *packet loss* dihitung dari rasio pemeriksaan *offline* sehingga identik dengan komplemen *availability*. Keempat, analisis QoS masih diverifikasi menggunakan basis data uji. Kelima, agen berjalan pada satu telepon pintar sehingga status terakhir tidak lagi mencerminkan kondisi sebenarnya bila perangkat tersebut mati. Keenam, UAT dilakukan secara kualitatif dengan tiga responden.

Penelitian selanjutnya dapat menambahkan pengukuran *throughput*, pembacaan *packet loss* langsung dari keluaran *ping*, pengamatan QoS jangka panjang menggunakan data agen, mekanisme *heartbeat* untuk mendeteksi agen yang tidak lagi mengirim data, notifikasi berbasis ambang batas *latency*, serta UAT dengan kuesioner berskala dan responden yang lebih banyak.

4. KESIMPULAN

Penelitian ini menghasilkan aplikasi monitoring jaringan WiFi berbasis web dengan arsitektur hibrida untuk Cafe Formaggio. Agen berbasis Termux di jaringan lokal memeriksa perangkat setiap 60 detik dan mengirimkan hasilnya ke aplikasi Laravel pada VPS melalui API yang dilindungi token dan pembatasan laju, sehingga dasbor dapat diakses dari mana saja tanpa server lokal, perangkat SNMP, atau API merek *router* tertentu. Notifikasi Telegram berbasis transisi status hanya mengirim pesan saat perangkat terputus dan pulih, dan perilaku ini terverifikasi pada pengujian *basis path* maupun kejadian nyata di lokasi.

Pengujian *black box* menunjukkan 53 dari 53 skenario berjalan sesuai harapan, UAT bersama Pemilik, Teknisi, dan Admin menilai aplikasi sangat layak digunakan, dan verifikasi fungsi analisis QoS pada basis data uji menghasilkan indeks TIPHON 3,67 (Memuaskan). Sistem ini dapat menjadi alternatif monitoring jaringan yang murah dan mudah digunakan bagi usaha kecil.

REFERENCES

European Telecommunications Standard Institute (ETSI). (1999). *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); General aspects of Quality of Service (QoS)* (ETSI TR 101 329 V2.1.1). ETSI.

Fachrurrozi, N. R., Wirabudi, A. A., & Rozano, S. A. (2023). Design of network monitoring system based on LibreNMS using Line Notify, Telegram, and Email notification. *Sinergi, 27*(1), 111–122. https://doi.org/10.22441/sinergi.2023.1.013

Fadhillah, N., Raihan, M. D., & Haryono, W. (2026). Rancang bangun aplikasi prototype booking dan penjadwalan ruang rapat, laboratorium, dan ruang musik berbasis web. *Riau Jurnal Teknik Informatika, 5*(1), 27–36.

Febriana, Waluyo, R., & Mustofa, D. (2026). Network monitoring using Zabbix ICMP Ping and Telegram notifications using the Network Development Life Cycle model. *Journal of Technology and Informatics (JoTI), 8*(1), 62–72. https://doi.org/10.37802/joti.v8i1.1079

Fergina, A., Somantri, & Nuralam, F. (2024). Penggunaan metode NDLC dalam sistem monitoring jaringan di PT Proxi Jaringan Nusantara untuk cabang Pulau Jawa dan Bengkulu dengan LibreNMS dan integrasi bot Telegram. *Jurnal Sistem Informasi dan Telematika, 15*(2), 162–170. https://doi.org/10.36448/jsit.v15i2.3705

Hakim, N. F. A., Elvyanti, S., & Ardiansyah, Y. (2025). Design of a Telegram-integrated Grafana network monitoring system for enhanced fault detection and real-time notification. *Jurnal Ilmiah Teknologi dan Rekayasa, 30*(1), 95–107. https://doi.org/10.35760/tr.2025.v30i1.14058

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 4, No. 04 Agustus Tahun 2026
ISSN 2985-4172 (media online)
Hal 385-401

Hidayat, R. B. N., Oktaviana, L. D., & Kuncoro, A. P. (2024). Quality of Service analysis of WiCo network GraPARI Telkom Purwokerto using TIPHON standard. *Jurnal Teknik Informatika (JUTIF), 5*(4), 395–405. https://doi.org/10.52436/1.jutif.2024.5.4.2362

Israf, Z., Setiawan, N., Valentino, J., & Haryono, W. (2026). Implementasi sistem dan prototype learning management system (LMS) akademik berbasis web dengan menggunakan metode Schoology. *Jurnal Informatika dan Teknik Elektro Terapan, 14*(1). https://doi.org/10.23960/jitet.v14i1.8635

Kadim, A. A., Hadjaratie, L., Mokoginta, M., & Zakaria, A. (2025). Rancang bangun sistem informasi rekam medis berbasis web dengan framework Laravel. *Diffusion: Journal of Systems and Information Technology, 5*(1), 195–204. https://doi.org/10.37031/diffusion.v5i1.30066

Nugroho, I. F., Wahyono, A., & Setiawati, D. (2024). Rancang bangun sistem informasi pusat data berbasis website menggunakan metode prototype. *Jurnal Sistem Informasi dan Telematika, 15*(2), 224–232. https://doi.org/10.36448/jsit.v15i2.3788

Nuryani, R., Rahmatuloh, M., & Resdiana, W. (2024). Aplikasi dashboard kinerja Wireless Local Area Network (WLAN) menggunakan metode Quality of Service (QoS). *Jurnal Kecerdasan Buatan dan Teknologi Informasi, 3*(3), 146–155. https://doi.org/10.69916/jkbti.v3i3.153

Postel, J. (1981). *Internet Control Message Protocol* (RFC 792). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc792

Pratama, A. R., & Sutomo, B. (2026). Implementation of a web-based network performance monitoring system using the Quality of Service (QoS) method on MikroTik router infrastructure. *Djtechno: Jurnal Teknologi Informasi, 7*(2), 319–331. https://doi.org/10.46576/djtechno.v7i2.8797

Pressman, R. S., & Maxim, B. R. (2015). *Software engineering: A practitioner's approach* (8th ed.). McGraw-Hill Education.

Putro, A. T. A., Wibowo, A., & Sutikno. (2024). Evaluasi usability pada aplikasi sistem pencatatan pegawai menggunakan metode usability testing dan USE questionnaire. *Jurnal Masyarakat Informatika, 15*(2), 126–148. https://doi.org/10.14710/jmasif.15.2.67263

Rozi, F., Amelia, B. P., Anggraini, S., & Haryono, W. (2026). Sistem digitalisasi laporan dan monitoring pekerjaan instalasi fiber optik. *Nusantara of Engineering (NOE), 9*(01), 225–234. https://doi.org/10.29407/noe.v9i01.26177

Saleh, A. K., Tjahyaningtijas, H. P. A., Nurhayati, & Rakhmawati, L. (2022). Quality of Service (QoS) comparative analysis of wireless network. *Indonesian Journal of Electrical and Electronics Engineering (INAJEEE), 5*(2), 30–37. https://doi.org/10.26740/inajeee.v5n2.p30-37

Setiawan, R., Praja, Y. N., & Haryono, W. (2026). Web-based asset maintenance monitoring system for CV Tartila Group: Sistem pemantauan pemeliharaan aset berbasis web untuk Grup CV Tartila. *Nuansa Informatika, 20*(1), 42–49.

Simargolang, M. Y., & Widarma, A. (2022). Quality of Service (QoS) for network performance analysis Wireless Area Network (WLAN). *CESS (Journal of Computer Engineering, System and Science), 7*(1), 162–171. https://doi.org/10.24114/cess.v7i1.29758

Wahyuriyanto, A. G., & Yulianton, H. (2024). Implementasi sistem monitoring jaringan berbasis Simple Network Management Protocol dengan notifikasi Telegram. *Progresif: Jurnal Ilmiah Komputer, 20*(2), 632–647. https://doi.org/10.35889/progresif.v20i2.1939

Wijaya, W. P. C., & Achlaq, M. M. (2023). Implementasi API bot Telegram untuk sistem notifikasi LibreNMS pada perusahaan BLiP Integrator. *JATI (Jurnal Mahasiswa Teknik Informatika), 7*(6), 3152–3159. https://doi.org/10.36040/jati.v7i6.7991

Yuazijah, A., Solehudin, A., & Jaman, J. H. (2024). Sistem monitoring jaringan menggunakan metode Quality of Service (QoS) dengan perangkat lunak The Dude: Studi kasus PT. Atlas Lintas Indonesia. *JATI (Jurnal Mahasiswa Teknik Informatika), 8*(6), 12137–12142. https://doi.org/10.36040/jati.v8i6.11811

Zidan, M., Nur'aini, S., Wibowo, N. C. H., & Ulinuha, M. A. (2022). Black box testing pada aplikasi Single Sign On (SSO) di Diskominfostandi menggunakan teknik equivalence partitions. *Walisongo Journal of Information Technology, 4*(2), 127–137. https://doi.org/10.21580/wjit.2022.4.2.12135