Implementasi *Network Monitoring System* Berbasis Web di Universitas Palangka Raya

Arman Juliansyah[1)], Septian Geges[2)], Novera Kristianti[3)]

[1),2),3)]Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya,
Kampus Tunjung Nyaho Jalan Yos Sudarso, Palangka Raya, Kalimantan Tengah, Indonesia
[1)]armjn1172@mhs.eng.upr.ac.id
[2)]septian.geges@it.upr.ac.id
[3)]noverakristianti@eng.upr.ac.id

Abstrak

Universitas Palangka Raya (UPR) merupakan institusi pendidikan tinggi di Kalimantan Tengah yang memiliki infrastruktur jaringan untuk menunjang berbagai kegiatan kampus. Namun, saat ini UPR belum memiliki sistem pemantauan jaringan terpusat yang mengakibatkan penanganan masalah jaringan dilakukan manual, sehingga dapat menggangu aktivitas di lingkungan kampus. Penelitian ini bertujuan merancang dan membangun sistem *monitoring* jaringan berbasis *web* di Universitas Palangka Raya. Sistem dikembangkan menggunakan metode *waterfall* melalui tahapan analisis, desain, implementasi (*Python*, *Javascript*, *HTML*, *CSS*, *Bootstrap*, *MySQL*), dan pengujian (*black-box*), serta memanfaatkan protokol standar (ICMP, SSH, API, SNMP) dan notifikasi Telegram. Hasil pengujian bahwa sistem berjalan baik dengan dapat memantau kondisi jaringan secara *real time*, mendapatkan *traffic* data, dan mengkonfigurasi dan verifikasi perangkat.

Kata kunci: Sistem *Monitoring* Jaringan, *Monitoring*, *Automation*, Universitas

Abstract

University of Palangka Raya (UPR) is a higher education institution in Central Kalimantan that has network infrastructure to support various campus activities. However, currently UPR does not have a centralized network monitoring system which results in manual handling of network problems, so that it can disrupt activities in the campus environment. This research aims to design and build a web-based network monitoring system at the University of Palangka Raya. The system is developed using the waterfall method through the stages of analysis, design, implementation (Python, Javascript, HTML, CSS, Bootstrap, MySQL), and testing (black-box), as well as utilizing standard protocols (ICMP, SSH, API, SNMP) and Telegram notifications. The test results show that the system is running well by being able to monitor network conditions in real time, obtain data traffic, and configure and verify devices.

Keywords: *Network Monitoring System, Monitoring, Automation, University.*

1. PENDAHULUAN

Perkembangan teknologi informasi telah meningkatkan kebutuhan infrastruktur jaringan terutama di lingkungan perguruan tinggi [1]. Universitas Palangka Raya (UPR) sebagai institusi pendidikan tinggi di Kalimantan Tengah memiliki infrastruktur jaringan yang mendukung kegiatan akademik, penelitian, administrasi, dan kegiatan lainnya di lingkungan kampus[2].

Pada saat ini, sistem *monitoring* jaringan di UPR saat ini belum terpusat dan masih bergantung pada laporan manual dari pengguna kepada administrator jaringan UPT TIK (Unit Pelayanan Terpadu Teknologi Informasi dan Komunikasi) Universitas Palangka Raya kemudian diteruskan ke penyedia layanan *internet* (*provider*). Sehingga Kondisi ini menyebabkan keterlambatan dalam penanganan masalah jaringan.

DOI:https://doi.org/10.47111/jointecoms.v6i2 *Received: 01-06-2026*
Accepted: 10-06-2026

Pada penelitian ini bertujuan untuk mengembangkan sistem *monitoring* jaringan berbasis *web* yang mampu menampilkan informasi kondisi perangkat jaringan secara *real-time*, mengelola data *traffic*, dan melakukan otomasi konfigurasi dan verifikasi perangkat Sistem yang dikembangkan memanfaatkan protokol ICMP, SSH, API, dan SNMP untuk komunikasi dan pengumpulan data, serta menggunakan mekanisme *background service* untuk proses pemantauan secara berkala yang terintegrasi dengan bot *Telegram* untuk penyampaian notifikasi. Sistem ini dapat memberikan solusi dalam mengelola dan memonitoring kondisi jaringan di Universitas Palangka Raya, terutama dalam proses kegiatan akademik berlangsung.

2. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian yang akan dilakukan memiliki keterkaitan dengan topik penelitian yang telah dilakukan sebelumnya.

Pada penelitian dengan judul *Network Automation* Pada Beberapa Perangkat Router Menggunakan Pemrograman Python, bertujuan untuk membuat sistem *dashboard* berbasis *website* yang dikelola oleh *Network Administrator* dengan menerapkan *Network Automation* sehingga dapat melakukan konfigurasi dan verifikasi pada beberapa jenis perangkat jaringan [3].

Pada penelitian dengan judul Perancangan dan Implementasi Sistem *Monitoring* Jaringan di MA Darut Taqwa Berbasis Web yang Mengintegrasikan dengan API MikroTik, bertujuan untuk membangun sistem *monitoring* jaringan dan pengelolaan data layanan *hostpot* secara *real time* berbasis *website* untuk memberikan solusi dalam pemantauan kondisi jaringan dan mengelola data *user* (siswa) dalam akses layanan *hotspot* di sekolah MA Darut Taqwa [4].

Pada penelitian dengan judul Sistem *Monitoring* Jaringan *Realtime* Berbasis *Internet Control Message Protocol*, bertujuan untuk membuat *sistem monitoring* jaringan berbasis *website* (SIMONIT) untuk melakukan pemantauan jaringan komputer secara *real time* dengan menggunakan protokol ICMP (*Internet Controk Message Protocol*) dengan proses yang berjalan dibelakang layar (*backdoor*) dan terintegrasi dengan *Telegram* sebagai notifikasi apabila ditemukkan permasalahan jaringan [5].

Pada penelitian dengan judul Rancang Bangun Sistem *Monitoring* Jaringan *Access Point* Menggunakan *Simple Network Management Protocol* (Snmp) Berbasis *Web*, bertujuan untuk membuat sistem *monitoring* berbasis *website* yang dapat melakukan *controlling* dan *monitoring* jaringan untuk perangkat AP (*Access Point*) di lingkungan Institut Teknologi Nasional Malang[2].

2.2 Monitoring Jaringan

Monitoring jaringan adalah proses *real-time* menggunakan perangkat lunak untuk memantau kondisi perangkat komputer dan jaringan. Proses ini melibatkan identifikasi perangkat, dan pengelolaan data dalam sistem untuk memecahkan masalah jaringan. Dengan menerapkan monitoring jaringan dalam sistem dapat mempercepat perbaikan masalah, memastikan ketersediaan layanan, dan membantu dalam pengelolaan jaringan oleh administrator jaringan[4].

2.3 Network Automation

Network Automation merupakan proses mengotomatisasi konfigurasi, mengelola, menguji, menyebarkan, dan mengoperasikan perangkat fisik dan *virtual* dalam jaringan [6]. Dengan proses otomatis dan berulang yang dilakukan secara otomatis, dapat mengurangi kesalahan manusia, dan menurunkan biaya operasional.

2.4 Jaringan Komputer

Jaringan komputer merupakan sebuah jaringan yang terdiri dari dua atau lebih komputer yang saling terhubung berkomunikasi data antar perangkat menggunakan media transmisi seperti sinyal, kabel, dan lainnya [7]. Jenis jaringan komputer yang sering digunakan berdasarkan jarak jangkauan yaitu LAN (*Local Area Network*) dan WAN (*Wide Area Network*) [8].

2.5 Protokol Jaringan

Protokol jaringan adalah proses aturan dalam komunikasi data antar perangkat dalam jaringan tanpa melihat perbedaan proses, struktur, atau desain internal perangkat [6]. Berikut beberapa protokol jaringan yang digunakan:

1. *Internet Control Message Protocol* (ICMP)

ICMP merupakan protokol jaringan yang dapat membantu memeriksa masalah konektivitas dan memastikan paket data ketika dikirim dari sumber (*source*) ke tujuan (*destination*). Implementasi yang digunakan baik itu pada sistem operasi atau dalam *programming* yaitu perintah "ping". ICMP akan mengirim pesan antar perangkat untuk memeriksa jangkauan dan potensi keterlambatan atau kehilangan paket. Apabila jaringan melambat, ICMP membantu mencari sumber masalah sehingga penyesuaian dapat dilakukan untuk meminimalkan gangguan.

2. *Application Programming Interface* (API)

API merupakan *interface* yang dapat menghubungkan antara kedua aplikasi atau lebih yang berbeda lainnya. Rest API merupakan cara atau aturan untuk membuat *web service* dengan menggunakan menggunakan metode HTTP atau HTTPS dalam mendapatkan atau mengirim informasi antar aplikasi atau perangkat.

3. *Simple Network Management Protocol* (SNMP)

SNMP merupakan protokol manajemen jaringan yang melakukan proses manajemen perangkat dalam pengiriman dan menerima data dengan terhubung dalam jaringan IP (*Internet Protocol*). SNMP terdiri dari empat komponen utama diantaranya sebagai berikut:

1) *SNMP Manager* yang mengirim perintah untuk mengambil data dari perangkat jaringan melalui *SNMP Agents*.
2) *Managed Device* yang mengaktifkan layanan SNMP pada perangkat.
3) *SNMP Agents* merupakan program pada perangkat yang mengumpulkan data kinerja dari MIB atas permintaan *SNMP Manager*.
4) MIB (*Management Information Base*), database hierarkis berisi objek terkelola dengan OID.

Pengunaan SNMP dapat dibantu dengan aplikasi MIB Browser. Aplikasi tersebut untuk mengambil dan mengatur data MIB (*Management Information Base*) dari perangkat jaringan SNMP. Alat ini menyediakan tampilan MIB, pengaturan variabel, tampilan pohon MIB, dan mendukung semua versi SNMP.

4. *Secure Shell* (SSH)

SSH (*Secure Shell*) merupakan metode untuk mengirim perintah dengan aman ke komputer melalui jaringan pada jaringan lokal atau *Internet*. Penggunaan metode ini menggunakan kriptografi untuk mengotentikasi dan mengenkripsi koneksi antar perangkat.

2.6 Bot Telegram

Aplikasi Telegram merupakan aplikasi perpesanan atau *message* yang masih sering digunakan berbagai pengguna sampai saat ini termasuk penggunaan *Bot Telegram*[9]. *Bot Telegram* sering digunakan untuk menjalankan tugas-tugas tertentu salah satunya yaitu pada sistem *monitoring* jaringan. Penggunaan pada sistem monitoring jaringan memberikan notifikasi *real-time* kepada administrator jaringan untuk memantau dan mengelola jaringan [10].

2.7 Metode Pengembangan Sistem

Metode Pengembangan sistem merupakan suatu proses pengembangan sistem yang formal dan mendefinisikan serangkaian aktivitas dalam rangka mengembangkan dan merawat sebagai keseluruhan sistem atau aplikasi. Metode pengembangan sistem sering digunakan yaitu pendekatan dari pendekatan SDLC (*System Development Life Cycle*) dengan model *Waterfall* (Royce,1970) menurut para ahli yaitu Pressman (2001). Metode *Waterfall* merupakan sebuah pendekatan yang sistematis dan sekuensial (berurutan) dengan tahapan-tahapan mulai dari analisis, desain, pengkodean dan pengujian [8].

2.8 Black Box Testing

Black Box Testing merupakan sebuah model dan metode pengujian yang dilakukan terhadap sebuah sistem, atau aplikasi untuk mengetahui fungsionalitas yang digunakan pada sistem atau aplikasi tersebut, tanpa perlu mengetahui mengenai sumber kode dan alur proses yang terjadi di dalamnya [8].

3. METODE PENELITIAN

Metode pengembangan sistem yang digunakan pada penlitian ini menggunakan metode *Waterfall* menurut Presman (2001). Berikut tahapan pengembangan website dalam penelitian ini:

3.1 Analisis

3.1.1 Pengumpulan Data

Adapun teknik yang digunakan dalam pengumpulan data terdapat beberapa cara sebagai berikut:

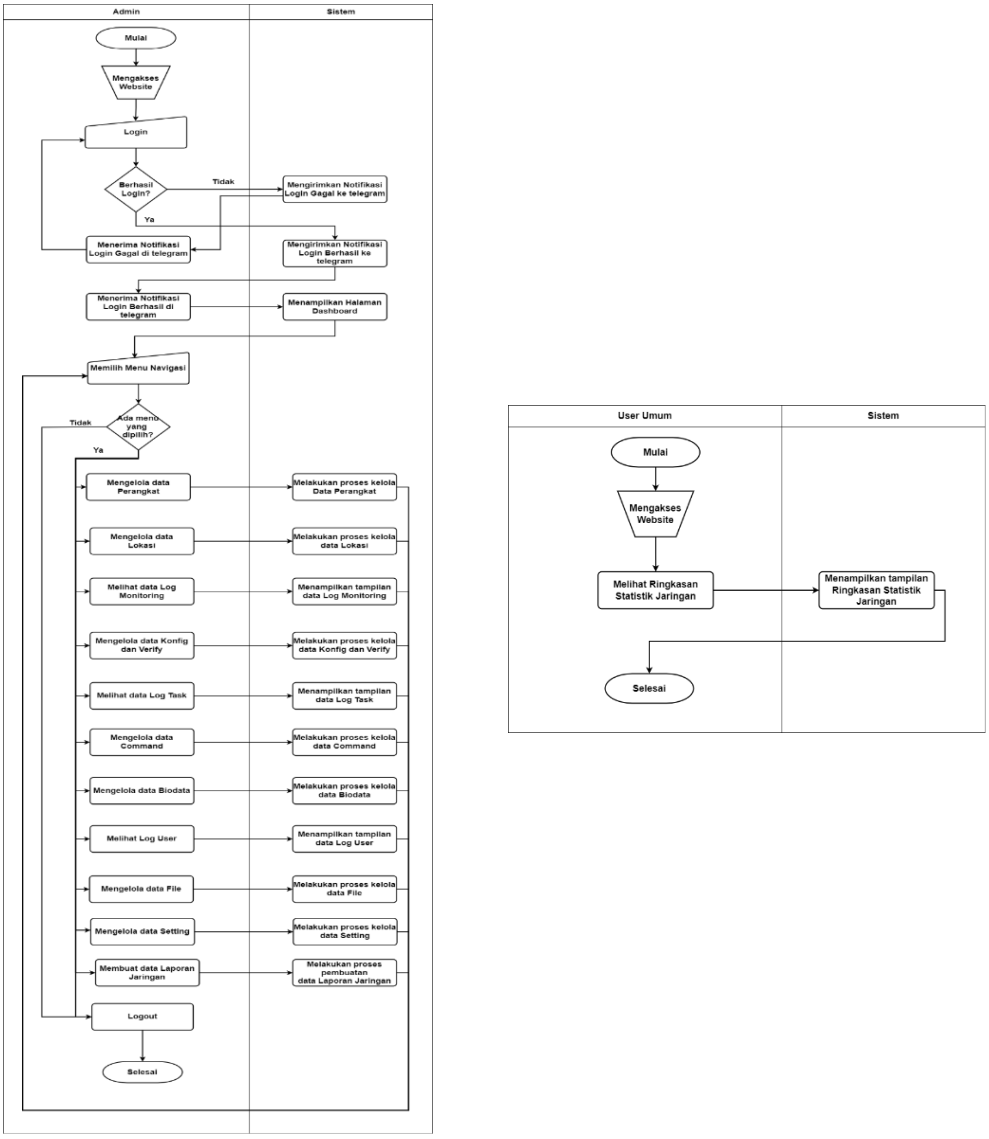
1. Observasi
   Metode ini dilakukan dengan pengamatan langsung ke lingkungan Universitas Palangka Raya yaitu UPT TIK (Unit Pelayanan Terpadu Teknologi Informasi dan Komunikasi), sehingga mendapatkan data *real* dari hasil penelitian yang dilakukan.
2. Wawancara
   Peneliti melakukan wawancara dengan administrator jaringan UPT TIK (Unit Pelayanan Terpadu Teknologi Informasi dan Komunikasi) Universitas Palangka Raya secara langsung. Hasil yang didapatkan bahwa belum tersedia aplikasi dalam melakukan pemantauan (*monitoring*) jaringan di lingkungan UPR. Sehingga pelaporan masalah jaringan masih dilakukan manual kepada administrator jaringan di UPT TIK dan penyedia layanan internet (*provider*), yang berpotensi menyebabkan keterlambatan.
3. Studi Pustaka
   Metode ini dilakukan pengumpulan data dengan cara mendalami materi melalui studi jurnal, artikel dan dokumen yang terkait dengan sistem *Monitoring* Jaringan.

3.1.2 Analisis Kebutuhan

Tahap analisis kebutuhan bertujuan untuk mengidentifikasi keperluan pengguna dalam pengembangan aplikasi. Proses ini meliputi pengumpulan spesifikasi data yang dibutuhkan mulai dari perangkat keras dan perangkat lunak untuk merancang dan membuat sistem *monitoring* jaringan.

3.1.3 Analisis Sistem

Pada tahap ini terdapat *flowchart* proses bisnis dari sistem baru yang dibuat untuk memperbaiki keterbatasan dari sistem lama. Fitur dan proses bisnis sistem baru memberikan pembaruan dari sistem yang lama dengan fitur dan proses bisnis yang dapat membantu dalam pemantauan kondisi jaringan. Proses bisnis sistem terdiri 2 bagian yang digunakan yaitu sistem utama sebagai mengelola data *monitoring* jaringan dan sistem User Umum sebagai halaman statistik monitoring jaringan yang dapat diakses secara *public*. *Flowchart* yang digunakan sebagai berikut:



Gambar 1. *Flowchart* Sistem Baru dari Sistem Utama dan User Umum

3.2 Desain

3.2.1 Topologi Jaringan

Desain topologi jaringan yang digunakan terdiri dari Router Core yang berperan sebagai sumber utama koneksi *Internet*, bertugas mengelola dan menyalurkan trafik jaringan ke seluruh sistem. *Web Server*, sebagai sistem *monitoring*, akan terhubung ke seluruh perangkat jaringan melalui jalur koneksi *Internet*. Selanjutnya, router distribusi berperan mengelola data jaringan *internet* dari Router Core dan mendistribusikannya ke jaringan pengguna melalui Switch distribusi. Pada Switch distribusi berperan mendistribusikan jalur *internet* yang dikelola Router distribusi ke jaringan pengguna.

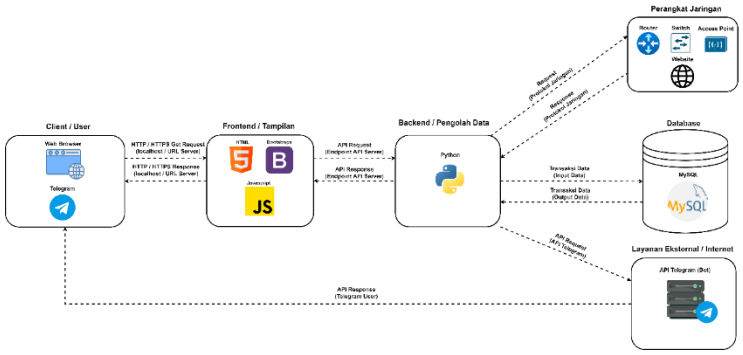
3.2.2 Arsitektur Aplikasi

Arsitektur aplikasi yang dibuat terdiri dari beberapa lapisan yang menjelaskan setiap alur kerja yang dilakukan. Berikut alur kerja pada arsitektur aplikasi:

1. Pengguna (User) berinteraksi dengan lapisan tampilan (*frontend*).
2. Lapisan tampilan (*frontend*) dengan teknologi (*JavaScript*, *HTML*, *CSS*, *Bootstrap*) mengirim permintaan ke lapisan pengolah data (backend).

3. Lapisan pengolah data (*backend*) dengan teknologi (*python*) memproses permintaan data secara manual atau melalui *background service*. Kemudian lapisan ini berinteraksi dengan basis data *MySQL*. untuk menyimpan data dari hasil permintaan terhadap perangkat jaringan atau hasil pemrosesan dan mengirimkan kembali ke pengguna melalui lapisan tampilan (*frontend*).
4. Lapisan pengolah data (*backend*) mengirimkan permintaan ke lapisan perangkat jaringan, dan lapisan perangkat jaringan merespons permintaan dari lapisan pengolah data (*backend*), berupa data yang diminta oleh lapisan *backend*.
5. Lapisan pengolah data (backend) terintegrasi dengan *API Telegram* yang mengirimkan data notifikasi melalui internet ke API Telegram (*Bot Telegram*) dan mengirimkan kembali berupan teks notifikasi ke aplikasi Telegram pengguna.

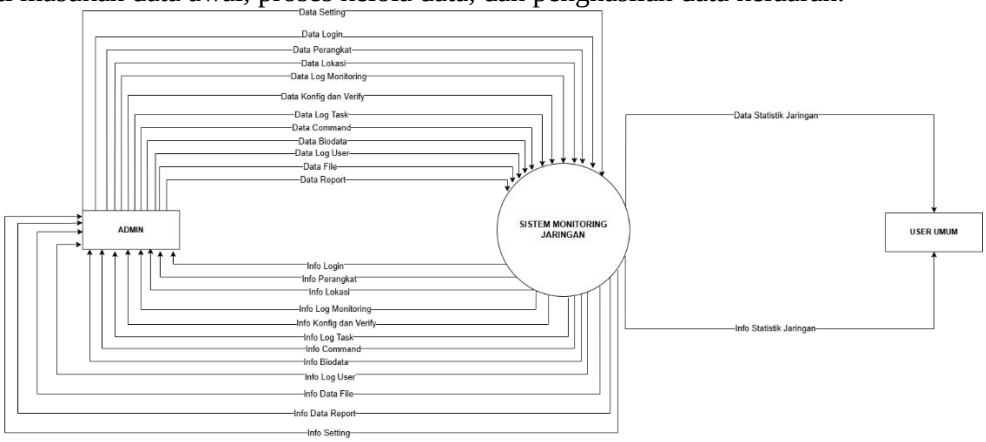
Arsitektur aplikasi ini dapat dilihat pada gambar 2.



Gambar 2. Arsitektur Aplikasi

3.2.3 Data Flow Diagram (DFD)

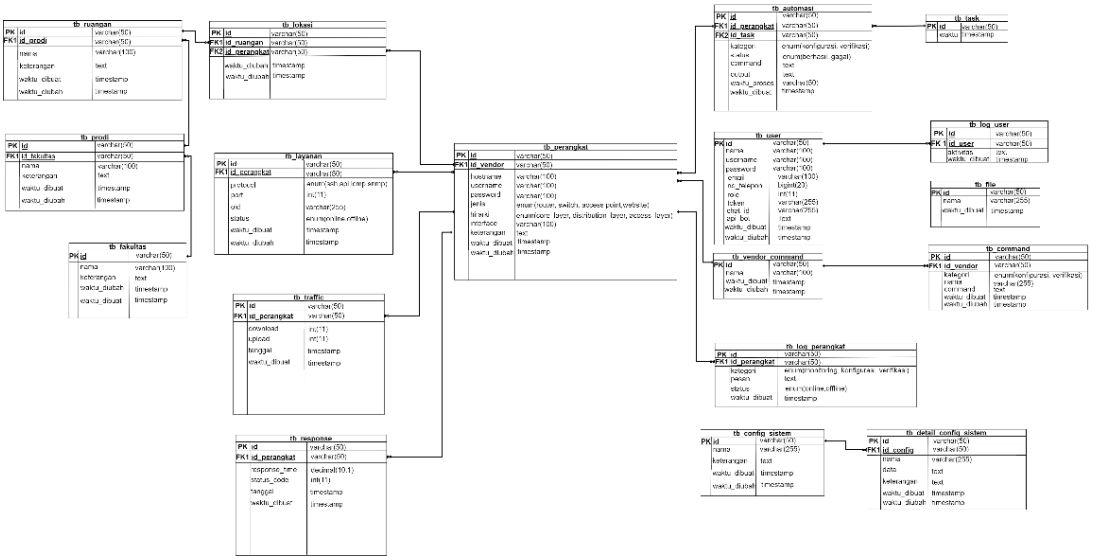
Data Flow Diagram (DFD) menggambarkan aliran data dalam suatu sistem. Penggambaran alur sistem ini mencakup interaksi antara sistem dengan data, serta interaksi antara sistem dengan entitas eksternal seperti pengguna (administrator dan pengguna umum). Proses yang digunakan meliputi masukan data awal, proses kelola data, dan penghasilan data keluaran.



Gambar 3. Diagram Konteks (DFD Level 0)

3.2.4 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) sistem *monitoring* jaringan ini merepresentasikan struktur basis data yang digunakan untuk menyimpan dan mengelola seluruh data yang dibutuhkan. ERD ini menggambarkan keterkaitan antar tabel, seperti tabel perangkat dengan tabel-tabel lainnya, yang diimplementasikan dalam database MySQL. Berikut adalah representasi tabel ERD yang diimplementasikan dalam database MySQL.



Gambar 4. *Entity Relationship Diagram* (ERD)

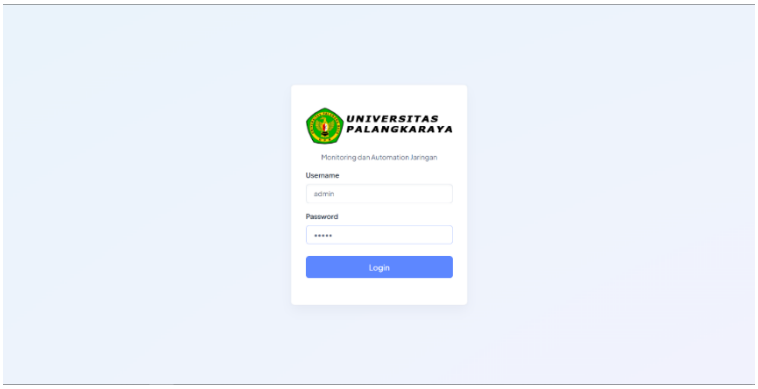
4. PEMBAHASAN

4.1 Implementasi Sistem

Implementasi sistem monitoring jaringan berbasis web untuk Universitas Palangka Raya dilakukan melalui serangkaian tahapan yang meliputi pengembangan antarmuka (UI) pengguna (*frontend*) menggunakan *HTML*, *CSS*, *Bootstrap* dan Javascript, pengelola data (*backend*) menggunakan *Python*, dan perancangan serta implementasi basis data menggunakan *MySQL*. Berikut ini adalah implementasi sistem monitoring jaringan yang digunakan:

1. Tampilan Halaman *Login*

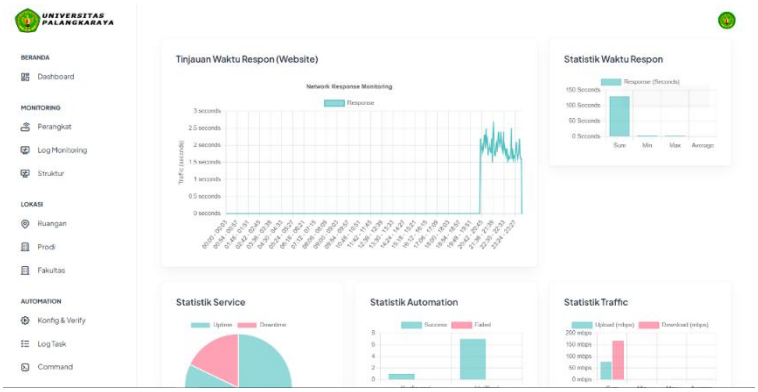
Halaman ini merupakan halaman awal saat pengguna (admin) mengakses *website*. Ketika ingin masuk halaman selanjutnya, maka penguna perlu memasukkan *username* dan *password* pada *form login*.



Gambar 5. Halaman *Login*

2. Tampilan Halaman *Dashboard*

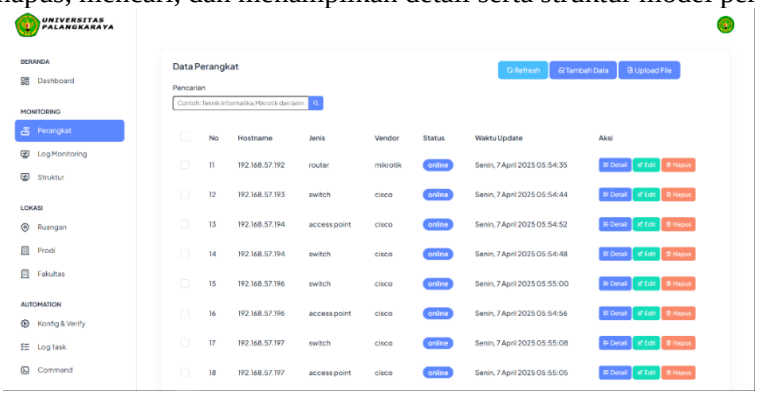
Ketika pengguna berhasil *login*, selanjutnya masuk ke halaman *dashboard*. Halaman ini menampilkan informasi berupa rangkuman kondisi jaringan yang terjadi saat ini diantaranya *traffic* perangkat, *response time website*, statistik penggunaan *bandwidth*, serta riwayat aktivitas perangkat dan pengguna.



Gambar 6. Halaman *Dashboard*

3. Halaman Kelola Perangkat

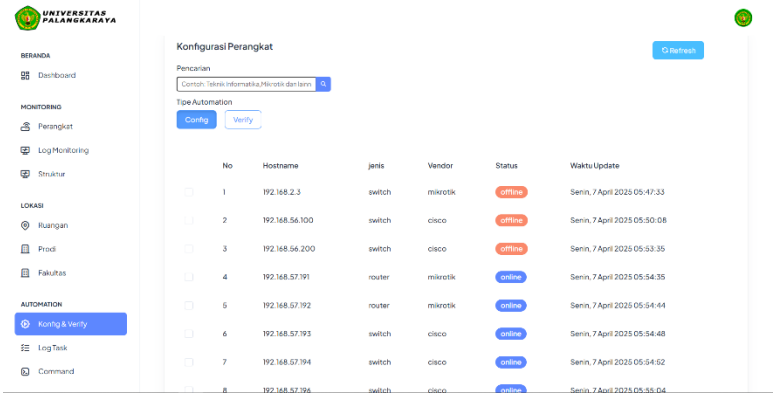
Pada halaman kelola perangkat, pengguna dapat mengelola data perangkat seperti menambah, mengubah, menghapus, mencari, dan menampilkan detail serta struktur model perangkat.



Gambar 7. Halaman Kelola Perangkat

4. Halaman Kelola Konfig dan Verify

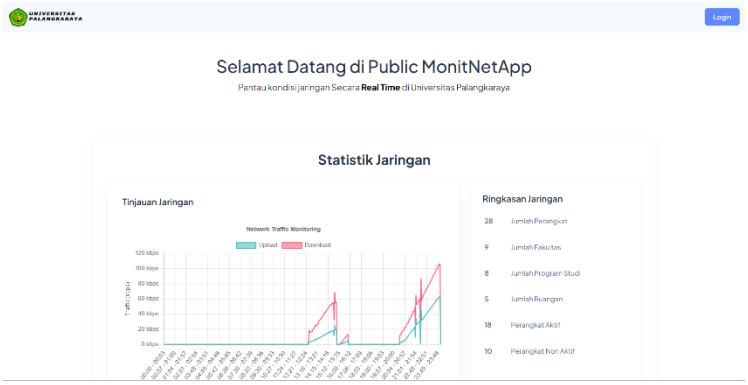
Pada halaman kelola konfig dan verify, pengguna dapat mengelola data *automation* yaitu konfigurasi dan verifikasi data perangkat serta menampilkan hasil *automation*.



Gambar 8. Halaman Konfig dan Verify

5. Halaman Statistik Jaringan

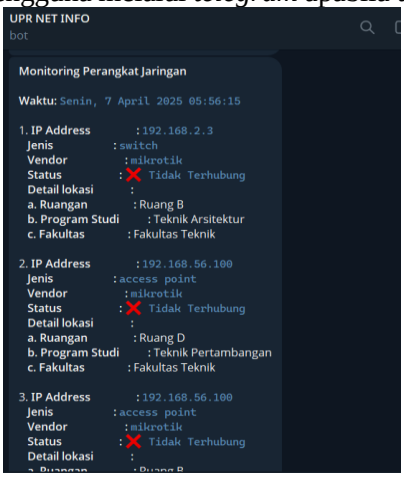
Pada halaman statistik jaringan, pengguna dapat melihat tampilan statistik jaringan umum secara *public* tanpa perlu melakukan *login* ke dalam sistem.



Gambar 9. Halaman Statistik Jaringan

6. Tampilan Notifikasi Sistem

Proses implementasi notifikasi *telegram* dilakukan integrasi terhadap sistem untuk memberikan notifikasi kepada pengguna melalui *telegram* apabila terjadi permasalahan jaringan.



Gambar 10. Notifikasi Sistem Periksa Kondisi Perangkat

4.2 Pengujian Sistem

Metode pengujian yang dilakukan pada penelitian ini menggunakan metode *blackbox testing*. Pengujian ini bertujuan untuk mengetahui kekurangan yang ada pada sistem. Berikut penjelasan setiap pengujian terhadap sistem.

Tabel 1. Pengujian

No	Pengujian	Langkah Pengujian	Hasil yang diharapkan	Hasil Pengujian
1.	Halaman *Login*	*User* menginputkan data *username* dan *password* login.	Menampilkan peringatan "berhasil" dan notifikasi pada *telegram* serta berpindah ke halaman *Dashboard*	OK
2.	Halaman *Dashboard*	*User* mengklik *button* pada setiap menu *sidebar*	Menampilkan tampilan sesuai fungsi atau menu yang dipilih	OK
3.	Halaman Kelola Perangkat	*User* mengklik *button* dan menginputkan data perangkat	Data perangkat dapat ditambah, diubah, dihapus, dicari dan menampilkan detail dan struktur model perangkat	OK

4.	Halaman Kelola Konfig dan Verify	*User* mengklik *button* dan menginputkan data perangkat berupa *command text* atau *file* pada kategori konfigurasi dan verifikasi	Data konfig dan verify dapat mengirimkan data berupa *text* dan *file* dengan kategori konfigurasi dan verifikasi	OK
5.	Halaman Statistik Jaringan	*User* memasukan alamat atau *url* halaman statistik jaringan	Data statistik jaringan dapat ditampilkan	OK

5. KESIMPULAN

Dari Hasil penelitian yang dilakukan terhadap sistem *monitoring* jaringan yang dibangun, bahwa sistem telah berjalan dengan baik sesuai dengan tujuan yang diinginkan yaitu dapat memantau kondisi perangkat jaringan secara *real time*, mendapatkan data *traffic* perangkat atau *response time website*, serta dapat mengkonfigurasi dan verifikasi perangkat. Oleh karena itu, sistem ini dapat menjadi bahan pertimbangan administrator jaringan UPT TIK Universitas Palangkaraya dalam mengelola jaringan.

6. DAFTAR PUSTAKA

[1] A. Aris, S. I. Alawiyah, and V. Melani, "Dasboard Monitoring Perangkat Jaringan Pada Server Universitas Raharja," *CICES*, vol. 8, no. 1, 2022, doi: 10.33050/cices.v8i1.2124.

[2] P. Kukuh Prayogi, M. Orisa, and F. Ariwibisono, "RANCANG BANGUN SISTEM MONITORING JARINGAN ACCESS POINT MENGGUNAKAN SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP) BERBASIS WEB," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 4, no. 1, 2020, doi: 10.36040/jati.v4i1.2327.

[3] S. Nugroho and B. Pujiarto, "Network Automation pada Beberapa Perangkat Router Menggunakan Pemrograman Python," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 1, 2022, doi: 10.25126/jtiik.2022913947.

[4] N. Sobah and M. F. Amrulloh, "Perancangan dan Implementasi Sistem Monitoring Jaringan di MA Darut Taqwa Berbasis Web yang Mengintegrasikan dengan API MikroTik," *BIOS J. Teknol. Inf. dan Rekayasa Komput.*, vol. 4, no. 2, 2023, doi: 10.37148/bios.v4i2.75.

[5] M. F. Qomarudin and A. Amrullah, "SISTEM MONITORING JARINGAN REALTIME BERBASIS INTERNET CONTROL MESSAGE PROTOCOL," *JINTECH J. Inf. Technol.*, vol. 3, no. 2, 2022, doi: 10.22373/jintech.v3i2.1935.

[6] L. G. Mauboy and T. Wellem, "Studi Perbandingan Library Untuk Implementasi Network Automation Menggunakan Paramiko Dan Netmiko Pada Router Mikrotik," *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 4, 2022, doi: 10.30865/jurikom.v9i4.4420.

[7] M. A. Wardana, A. Z. Nusri, and J. Juliandika, "Jaringan Virtual Private Network (Vpn) Berbasis Mikrotik Pada Kantor Kecamatan Marioriawa Kabupaten Soppeng," *J. Ilm. Sist. Inf. dan Tek. Inform.*, vol. 5, no. 2, 2022, doi: 10.57093/jisti.v5i2.135.

[8] I. P. A. E. Pratama and P. B. S. W. Putra, "Pengujian IaC Berbasis DevOps dan Ansible Menggunakan Metode Black Box Testing," *J. Fakt. Exacta*, vol. 15, no. 2, 2022.

[9] M. A. Husna and P. Rosyani, "Implementasi Sistem Monitoring Jaringan dan Server Menggunakan Zabbix yang Terintegrasi dengan Grafana dan Telegram," *JURIKOM (Jurnal Ris. Komputer)*, vol. 8, no. 6, 2021, doi: 10.30865/jurikom.v8i6.3631.

[10] M. R. Haikal, "ANALISIS QoS (QUALITY Of SERVICE) MENGGUNAKAN METODE QUEUE TREE SEBAGAI MANAGEMENT BANDWIDTH PADA JARINGAN INTERNET," 2023.