# SISTEM MONITORING JARINGAN BERBASIS WEB DENGAN FITUR NOTIFIKASI INSIDEN OTOMATIS DAN PENCATATAN LOG HISTORIS PADA PT. MITRA BERSAMA JAYA

**Wildi Baydowi, Hidayatullah Al Islami**
Teknik Informatika, Universitas Pamulang
Jl. Raya Puspitek Kota Tangerang Selatan Indonesia
*wildibaydowi@gmail.com*

**ABSTRAK**

Monitoring jaringan merupakan komponen penting dalam menjaga ketersediaan dan stabilitas layanan teknologi informasi di lingkungan perusahaan. Permasalahan yang sering dihadapi dalam pengelolaan jaringan adalah keterlambatan deteksi gangguan serta tidak tersedianya pencatatan insiden yang terstruktur, sehingga menghambat proses evaluasi dan penanganan gangguan jaringan. Penelitian ini bertujuan untuk mengimplementasikan sistem monitoring jaringan berbasis web yang dilengkapi dengan fitur notifikasi insiden otomatis dan pencatatan log historis. Metode pengembangan sistem yang digunakan adalah metode *Prototype*, yang memungkinkan pengembangan sistem dilakukan secara iteratif berdasarkan kebutuhan pengguna. Sistem yang dibangun mampu memantau status perangkat jaringan secara *real time*, mencatat insiden jaringan secara terstruktur, serta mengirimkan notifikasi otomatis melalui Bot Telegram ketika terjadi gangguan. Hasil implementasi menunjukkan bahwa sistem ini dapat meningkatkan kecepatan respons administrator terhadap insiden jaringan serta mendukung evaluasi kinerja jaringan berdasarkan data historis yang terdokumentasi.

***Kata kunci :*** *Monitoring Jaringan, Sistem Berbasis Web, Notifikasi Insiden, Log Historis, Telegram Bot*

## 1. PENDAHULUAN

Monitoring jaringan merupakan komponen penting dalam pengelolaan infrastruktur teknologi informasi, khususnya pada lingkungan perusahaan yang bergantung pada ketersediaan layanan jaringan secara berkelanjutan. Gangguan jaringan yang tidak terdeteksi secara cepat dapat berdampak pada terhambatnya proses operasional dan menurunnya kualitas layanan. Oleh karena itu, dibutuhkan sistem monitoring jaringan yang mampu memantau kondisi perangkat secara *real time* serta menyampaikan informasi gangguan secara tepat waktu kepada administrator jaringan [1].

Permasalahan yang sering dihadapi dalam pengelolaan jaringan adalah keterlambatan deteksi gangguan, keterbatasan mekanisme monitoring yang masih bersifat manual, serta belum tersedianya pencatatan insiden jaringan yang terstruktur. Selain itu, penyelesaian insiden umumnya ditandai secara otomatis tanpa adanya validasi dari administrator, sehingga berpotensi menyebabkan kurang akuratnya dokumentasi penanganan gangguan dan menyulitkan proses evaluasi kinerja jaringan dalam jangka panjang [2][3].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk mengimplementasikan sistem monitoring jaringan berbasis web yang dilengkapi dengan fitur notifikasi insiden otomatis dan pencatatan log historis. Sistem yang dikembangkan diharapkan mampu membantu administrator jaringan dalam memantau status perangkat secara *real time*, mencatat insiden jaringan secara terstruktur, serta meningkatkan efektivitas proses penanganan gangguan jaringan [4].

Sebagai rencana penyelesaian masalah, penelitian ini mengembangkan sistem monitoring jaringan berbasis web yang terintegrasi dengan mekanisme notifikasi otomatis melalui Bot Telegram. Sistem ini juga menerapkan pencatatan log historis serta validasi manual dalam penyelesaian insiden untuk memastikan setiap gangguan terdokumentasi secara akurat. Dengan pendekatan tersebut, sistem diharapkan dapat mendukung proses evaluasi dan pengelolaan jaringan secara lebih efektif di lingkungan PT. Mitra Bersama Jaya.

## 2. TINJAUAN PUSTAKA

### 2.1. Monitoring Jaringan

*Monitoring* jaringan merupakan proses pengamatan terhadap kondisi dan kinerja perangkat jaringan untuk memastikan ketersediaan layanan serta mendeteksi gangguan secara dini. Melalui *monitoring* jaringan, administrator dapat mengetahui status perangkat, mengidentifikasi penurunan kinerja, serta melakukan tindakan perbaikan secara lebih cepat dan terarah [5]. Penerapan *monitoring* jaringan juga berperan penting dalam menjaga keandalan dan stabilitas infrastruktur jaringan pada lingkungan perusahaan [6].

### 2.2. Sistem *Monitoring* Jaringan Berbasis *Web*

Sistem *monitoring* jaringan berbasis *web* memanfaatkan teknologi *web* untuk menampilkan informasi kondisi jaringan secara terpusat melalui *browser*. Pendekatan ini memberikan fleksibilitas bagi administrator jaringan dalam melakukan pemantauan dari berbagai lokasi tanpa ketergantungan pada sistem operasi tertentu [2]. Selain itu, sistem *monitoring* berbasis *web* mendukung visualisasi data *monitoring* secara *real time*, sehingga memudahkan proses

analisis kondisi jaringan dan pengambilan keputusan [6].

### 2.3. Notifikasi Insiden Otomatis

Notifikasi insiden otomatis merupakan mekanisme penyampaian informasi gangguan jaringan kepada administrator secara langsung ketika terjadi perubahan status perangkat. Integrasi aplikasi pesan instan seperti *Telegram* banyak digunakan sebagai media notifikasi karena mampu menyampaikan informasi secara cepat dan *real time* [7]. Beberapa penelitian menunjukkan bahwa penggunaan *Bot Telegram* dalam sistem *monitoring* jaringan dapat meningkatkan kecepatan respons administrator terhadap insiden jaringan serta mengurangi ketergantungan pada pemantauan manual [8]

### 2.4. Pencatatan *Log* Historis

Pencatatan *log* historis adalah proses penyimpanan data kejadian dan perubahan status perangkat jaringan dalam jangka waktu tertentu. *Log* historis berfungsi sebagai dasar evaluasi kinerja jaringan, analisis pola gangguan, serta audit insiden jaringan [4]. Dengan pencatatan *log* yang terstruktur, administrator dapat melakukan penelusuran kejadian secara lebih sistematis dan mendukung pengambilan keputusan berbasis data dalam pengelolaan jaringan [8], [9].

### 2.5. Metode *Prototype* dalam Pengembangan Sistem

Metode *Prototype* merupakan pendekatan pengembangan sistem yang dilakukan secara iteratif melalui pembuatan model awal, pengujian oleh pengguna, serta penyempurnaan sistem berdasarkan umpan balik yang diperoleh [7]. Metode ini dinilai sesuai untuk pengembangan sistem *monitoring* jaringan karena kebutuhan pengguna bersifat dinamis dan memerlukan penyesuaian selama proses implementasi. Pendekatan *prototyping* juga membantu memastikan bahwa sistem yang dikembangkan selaras dengan kebutuhan operasional pengguna.

### 2.6. Penelitian Terdahulu

Beberapa penelitian terdahulu telah membahas pengembangan sistem monitoring jaringan dengan berbagai pendekatan dan teknologi. Rahman dan Amnur [1] mengembangkan sistem monitoring server menggunakan Prometheus dan Grafana yang terintegrasi dengan notifikasi Telegram untuk membantu administrator dalam memantau kondisi server secara *real time*. Penelitian ini menekankan pentingnya visualisasi data dan notifikasi otomatis dalam meningkatkan respons terhadap gangguan jaringan.

Andriani dan Sa'di [2] mengimplementasikan sistem monitoring perangkat jaringan yang dilengkapi dengan notifikasi Bot Telegram. Hasil penelitian menunjukkan bahwa penggunaan notifikasi otomatis dapat mempercepat penyampaian informasi gangguan kepada administrator jaringan. Namun, penelitian ini belum membahas pencatatan insiden secara terstruktur dan validasi penyelesaian insiden.

Penelitian lain dilakukan oleh Saputra dan Syaripudin [3] yang merancang sistem monitoring jaringan berbasis web menggunakan Simple Network Management Protocol (SNMP). Sistem yang dikembangkan mampu menampilkan status perangkat jaringan secara terpusat dan membantu proses pemantauan jaringan. Akan tetapi, penelitian ini belum mengintegrasikan mekanisme pencatatan log historis dan notifikasi insiden berbasis aplikasi pesan instan.

Berdasarkan penelitian terdahulu, sistem monitoring jaringan berbasis web dengan notifikasi otomatis telah banyak dikembangkan. Namun, integrasi pemantauan status perangkat secara *real-time*, pencatatan log historis terstruktur, serta validasi manual dalam penyelesaian insiden masih terbatas. Oleh karena itu, penelitian ini dikembangkan untuk melengkapi kekurangan tersebut melalui penerapan sistem monitoring jaringan berbasis web yang terintegrasi dengan notifikasi insiden otomatis dan pencatatan log historis.

## 3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan pengembangan sistem untuk mengimplementasikan sistem *monitoring* jaringan berbasis *web* yang dilengkapi dengan notifikasi insiden otomatis dan pencatatan *log* historis. Metode penelitian disusun untuk memastikan sistem yang dikembangkan sesuai dengan kebutuhan operasional jaringan pada lingkungan PT. Mitra Bersama Jaya.

### 3.1. Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui observasi langsung dan wawancara dengan tim teknologi informasi di lingkungan PT. Mitra Bersama Jaya. Observasi dilakukan untuk memahami kondisi infrastruktur jaringan yang berjalan, mekanisme *monitoring* yang digunakan, serta alur penanganan insiden jaringan. Wawancara dilakukan untuk menggali kebutuhan sistem, kendala yang dihadapi administrator jaringan, serta ekspektasi terhadap sistem *monitoring* yang akan dikembangkan.

Pendekatan observasi dan wawancara digunakan untuk memperoleh gambaran kondisi operasional jaringan secara nyata, sehingga kebutuhan fungsional dan nonfungsional sistem dapat dirumuskan secara lebih akurat.

### 3.2. Metode Pengembangan Sistem

Pengembangan sistem *monitoring* jaringan dalam penelitian ini menggunakan metode *Prototype*. Metode *Prototype* merupakan pendekatan pengembangan sistem yang dilakukan secara iteratif melalui pembuatan model awal sistem, pengujian oleh pengguna, serta penyempurnaan berdasarkan umpan balik yang diperoleh [7], [10]. Metode ini dipilih karena mampu mengakomodasi kebutuhan pengguna

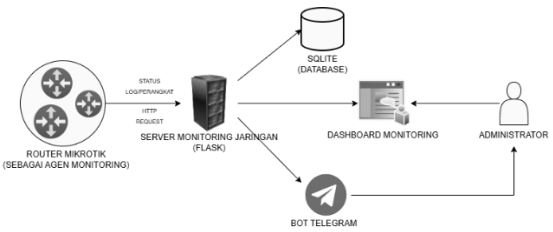
yang bersifat dinamis serta memungkinkan penyesuaian sistem selama proses implementasi.

### 3.3. Tahapan Pengembangan Sistem

Tahapan pengembangan sistem monitoring jaringan dengan metode *prototype* dalam penelitian ini dilakukan secara bertahap sesuai dengan pendekatan pengembangan sistem *iteratif* [7], sebagai berikut:

a. Analisis kebutuhan, tahap ini bertujuan untuk mengidentifikasi kebutuhan fungsional dan nonfungsional sistem. Kebutuhan fungsional meliputi pemantauan status perangkat jaringan secara *real time*, pengiriman notifikasi insiden otomatis, serta pencatatan *log* historis. Kebutuhan nonfungsional mencakup kemudahan akses sistem berbasis *web* dan keandalan sistem dalam menyimpan data historis.
b. Perancangan sistem, tahap perancangan meliputi perancangan arsitektur sistem, antarmuka pengguna (*dashboard*), struktur basis data, serta mekanisme integrasi dengan layanan eksternal seperti *Telegram Bot*. Hasil perancangan digunakan sebagai acuan dalam pengembangan *prototype* sistem.
c. Implementasi *Prototype*, pada tahap ini, *prototype* sistem dikembangkan menggunakan *framework web* dan diimplementasikan sesuai dengan rancangan yang telah dibuat. Implementasi mencakup modul *monitoring* status perangkat, modul pencatatan insiden, serta modul notifikasi otomatis.
d. Pengujian dan evaluasi, *prototype* yang telah dikembangkan diuji oleh administrator jaringan untuk mengevaluasi fungsionalitas sistem. Pengujian meliputi keakuratan pemantauan status perangkat, pencatatan insiden, serta pengiriman notifikasi. Masukan dari pengguna digunakan sebagai dasar evaluasi sistem.
e. Penyempurnaan sistem, tahap penyempurnaan dilakukan berdasarkan hasil evaluasi pengujian. Penyempurnaan mencakup perbaikan antarmuka, peningkatan stabilitas sistem, serta optimalisasi mekanisme *monitoring*, notifikasi, dan pencatatan *log* historis sebelum sistem digunakan secara operasional.

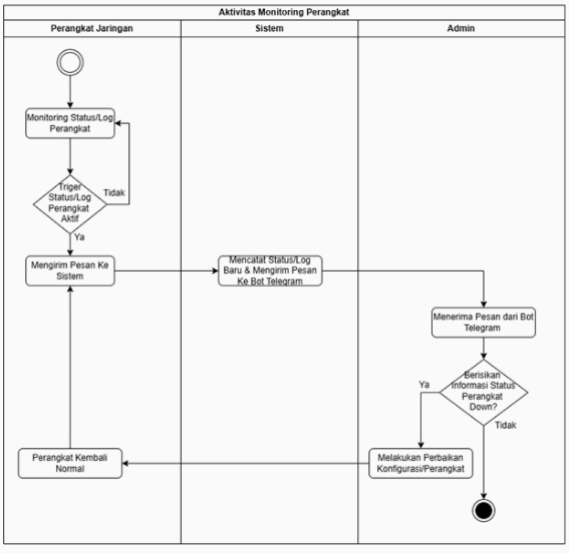
### 3.4. Diagram Arsitektur Sistem




Gambar 1. Diagram arsitektur sistem

Diagram arsitektur sistem menggambarkan hubungan dan alur pertukaran data antar komponen utama, meliputi perangkat jaringan, server monitoring berbasis *web*, basis data, serta layanan notifikasi. Diagram ini memberikan gambaran umum mekanisme kerja sistem monitoring jaringan secara terintegrasi.

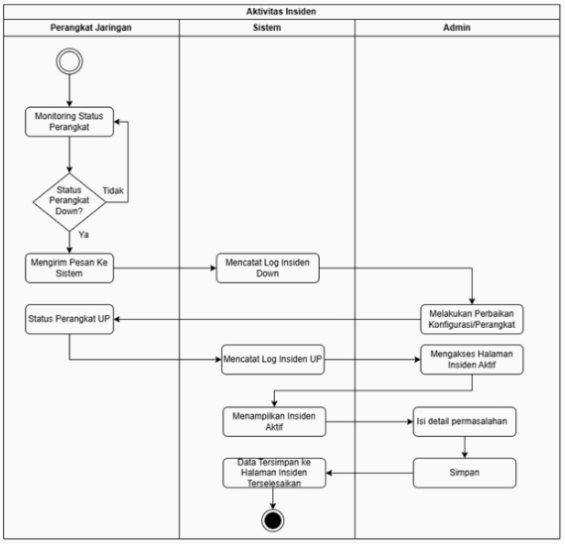
### 3.5. Activity Diagram Monitoring Status dan Log Perangkat




Gambar 2. Alur proses monitoring status dan log perangkat

*Activity diagram* ini menggambarkan alur proses pemantauan status perangkat jaringan secara *real time*, pencatatan log historis, serta penyampaian informasi kepada administrator. Diagram digunakan untuk memodelkan urutan aktivitas mulai dari penerimaan status perangkat hingga penyimpanan data log dalam sistem.

### 3.6. Activity Diagram Manajemen Insiden




Gambar 3. Alur proses manajemen insiden

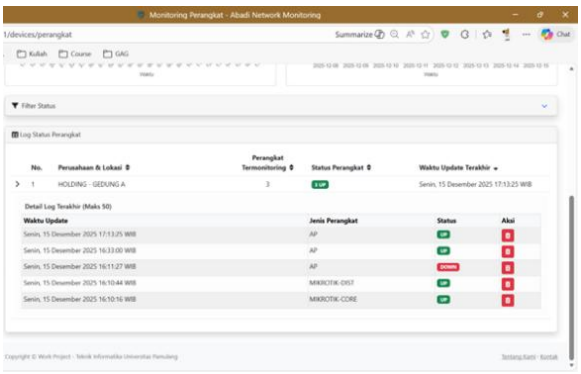
Activity diagram ini menggambarkan alur proses manajemen insiden jaringan yang terjadi akibat perubahan status perangkat menjadi tidak aktif (*down*). Insiden dicatat oleh sistem dan ditampilkan kepada

administrator untuk dilakukan penanganan. Penyelesaian insiden dilakukan setelah administrator memberikan keterangan penanganan dan memverifikasi status perangkat kembali aktif (*up*), sehingga dokumentasi insiden tersimpan secara terstruktur.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Hasil Implementasi Sistem

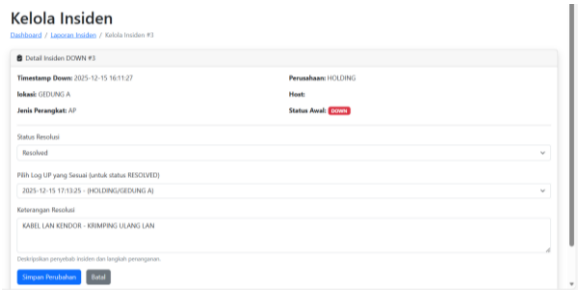
Hasil implementasi menunjukkan bahwa sistem monitoring jaringan berbasis *web* telah berjalan sesuai dengan perancangan yang ditetapkan. Sistem mampu menerima dan mengolah data status perangkat jaringan, menampilkan informasi kondisi perangkat secara *real time* melalui antarmuka *web*, serta mendukung pencatatan log historis dan pengelolaan insiden jaringan secara terstruktur.



Gambar 4. Tampilan monitoring status perangkat

Gambar ini menampilkan halaman monitoring status perangkat jaringan yang menyajikan informasi jumlah perangkat, status perangkat (*up/down*), serta waktu pembaruan terakhir. Tampilan ini memudahkan administrator jaringan dalam melakukan pemantauan kondisi jaringan secara terpusat dan mendeteksi gangguan secara cepat.

Selain pemantauan status perangkat, sistem juga menyediakan mekanisme pencatatan insiden jaringan secara otomatis. Ketika terjadi perubahan status perangkat menjadi *down*, sistem akan mencatat kejadian tersebut sebagai insiden aktif yang memerlukan penanganan lebih lanjut.




Gambar 5. Tampilan kelolal insiden aktif

Gambar tersebut menunjukkan halaman kelola insiden aktif yang digunakan administrator untuk menganalisis dan memberikan keterangan terkait insiden jaringan. Insiden tidak langsung dinyatakan selesai ketika perangkat kembali *up*, melainkan memerlukan validasi administrator, sehingga proses penanganan dan dokumentasi insiden dapat dilakukan secara lebih akurat.

### 4.2. Hasil Pengujian Fungsional Sistem

Pengujian fungsional dilakukan untuk memastikan seluruh fungsi utama sistem monitoring jaringan berjalan sesuai dengan kebutuhan yang telah ditetapkan. Pengujian dilakukan terhadap fitur pemantauan status perangkat, pencatatan log historis, pengelolaan insiden, serta pengiriman notifikasi otomatis berdasarkan skenario pengujian yang telah dirancang.

Tabel 1. Hasil pengujian fungsional sistem

| No | Skenario Pengujian | Hasil yang diharapkan | Hasil Pengujian |
|---|---|---|---|
| 1 | Monitoring status perangkat | Status perangkat tampil real time | Berhasil |
| 2 | Pencatatan log status | Log tersimpan di sistem | Berhasil |
| 3 | Pencatatan insiden DOWN | Insiden tercatat otomatis | Berhasil |
| 4 | Validasi penyelesaian insiden | Status berubah setelah validasi admin | Berhasil |
| 5 | Notifikasi Bot Telegram | Notifikasi terkirim saat perubahan status/log perangkat | Berhasil |

Berdasarkan hasil pengujian tersebut menunjukkan bahwa seluruh fungsi utama sistem dapat berjalan dengan baik dan sesuai dengan kebutuhan fungsional sistem monitoring jaringan.

### 4.3. Evaluasi Kinerja Sistem

Evaluasi kinerja sistem dilakukan untuk menilai efektivitas sistem monitoring jaringan dalam mendukung proses pemantauan dan penanganan insiden jaringan. Salah satu aspek yang dievaluasi adalah mekanisme penyampaian informasi gangguan melalui notifikasi otomatis yang diintegrasikan dengan Bot Telegram.

Notifikasi ini menampilkan perubahan status dan log perangkat jaringan yang dikirimkan melalui Bot Telegram, sehingga administrator dapat menerima informasi insiden secara cepat tanpa perlu melakukan pengecekan manual melalui sistem web. Hasil evaluasi menunjukkan bahwa integrasi Bot Telegram meningkatkan kecepatan penyampaian informasi insiden dan mendukung respons yang lebih cepat terhadap gangguan jaringan.


Gambar 6. Notifikasi bot *Telegram*

### 4.4. Pembahasan

Hasil implementasi, pengujian, dan evaluasi menunjukkan bahwa sistem monitoring jaringan berbasis *web* yang dikembangkan mampu memenuhi tujuan penelitian, yaitu menyediakan pemantauan status perangkat secara *real time*, pencatatan log historis terstruktur, serta notifikasi insiden otomatis. Penerapan validasi manual dalam penyelesaian insiden menjadi keunggulan sistem karena memastikan bahwa setiap insiden benar-benar telah ditangani oleh administrator sebelum dinyatakan selesai.

Integrasi notifikasi *Bot Telegram* meningkatkan responsivitas terhadap gangguan jaringan, sementara data log historis mendukung proses evaluasi dan analisis performa jaringan. Dengan demikian, sistem yang dikembangkan dinilai efektif dalam meningkatkan efisiensi dan kualitas pengelolaan jaringan.

## 5. KESIMPULAN DAN SARAN

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, dapat disimpulkan bahwa sistem monitoring jaringan berbasis web yang dikembangkan mampu memantau status perangkat jaringan secara *real time*, mencatat insiden jaringan secara terstruktur, serta mengirimkan notifikasi insiden otomatis melalui Bot Telegram. Penerapan mekanisme validasi manual dalam penyelesaian insiden memungkinkan dokumentasi penanganan gangguan tercatat secara lebih akurat dan mendukung evaluasi kinerja jaringan. Selain itu, pencatatan log historis dan visualisasi statistik gangguan yang tersedia dalam sistem membantu administrator dalam melakukan analisis kondisi jaringan secara berkelanjutan. Untuk pengembangan selanjutnya, sistem ini dapat ditingkatkan melalui pengembangan komunikasi dua arah pada Bot Telegram agar dapat mengambil dan menampilkan data langsung dari sistem berbasis web, serta penambahan dukungan *multi-vendor* agar sistem monitoring dapat digunakan pada berbagai jenis perangkat jaringan.

## DAFTAR PUSTAKA

[1] D. Rahman and H. Amnur, “Monitoring Server dengan Prometheus dan Grafana serta Notifikasi Telegram Jurnal Ilmiah Teknologi Sistem Informasi,” 2020.

[2] R. Andriani and A. Sa’di, “Notifikasi Bot Telegram pada Sistem Monitoring Perangkat Jaringan Implementation Of Telegram Bot Notification On Network Device Monitoring System,” 2023.

[3] M. A. Saputra and A. Syaripudin, “Implementasi Simple Network Management Protocol (SNMP) dalam Perancangan Sistem Monitoring Jaringan Berbasis Web (Studi Kasus: PT BFI Finance Indonesia Tbk, Cisauk),” 2024.

[4] Nugraha and A. Saffira, “Indonesian Journal Of Community Empowerment Implementasi Sistem Monitoring Jaringan Internal Menggunakan Uptime Kuma Yang Terintegrasi Dengan Bot Telegram Di PT Bentang Johar Awal,” 2025.

[5] M. Jaelani, R. N. Whidhiasih, and R. T. Handayanto, “Implementasi Sistem Monitoring Jaringan Mikrotik Dengan The Dude,” *Journal of Students‘ Research in Computer Science*, vol. 6, no. 1, pp. 83–98, May 2025.

[6] A. Nurfebrian, A. Pindarwati, and R. Hidayat, “Perancangan Dan Implementasi Sistem Monitoring Jaringan Berbasis Web PT. Maxindo Mitra Solusi,” *Teknik dan Multimedia*, vol. 1, no. 2, 2023.

[7] H. Situmorang and M. I. Zul, “Implementasi Metodologi Prototype dalam Pengembangan Sistem Manajemen Kehadiran Pegawai Perusahaan Berbasis Web,” *JTIM : Jurnal Teknologi Informasi dan Multimedia*, vol. 6, no. 3, pp. 260–270, Sep. 2024.

[8] T. A. Penerapan, M. Jaringan, D. Zabbix, T. Ariyadi, M. Fikri, and H. Yudiastuti, “Penerapan Monitoring Jaringan Dengan Zabbix Pada PT. PLN (Persero) UIP BAGIAN SUMBAGSEL,” Sep. 2024.

[9] M. Rivan, A. Arsandi, and A. Syaripudin, “Monitoring Jaringan berbasis Web Server Terintegrasi Zabbix dan Notifikasi Telegram Pada PT Time Excelindo,” vol. 3, no. 6, 2024.

[10] Bagus Ageng Alfahri, Alfani Azhar, Legiman Samosir, and Khalil Gibran, “Penerapan Metode Prototyping pada Pengembangan Aplikasi ‘We Listen We Don’t Judge’ untuk Analisis Tautan dan Pemberian Saran Otomatis Berbasis Web,” *Neptunus: Jurnal Ilmu Komputer Dan Teknologi Informasi*, vol. 3, no. 3, pp. 55–62, Jun. 2025