

# PENGEMBANGAN SISTEM VISUALISASI TOPOLOGI JARINGAN OLT BERBASIS WEBSITE DENGAN FITUR MONITORING TRAFFIC DAN DETEKSI PERANGKAT

Wangsa Reisyah Fatahillah<sup>1</sup>, Asri Wulandari<sup>2</sup>

<sup>1,2</sup>Program Studi Broadband Multimedia, Jurusan Teknik Elektro, Politeknik Negeri Jakarta; Jl. Prof. DR. G.A. Siwabessy, Kukusan, Beji, Depok City, West Java 16425; (021) 7270036

## Keywords:

FTTH; Congestion; Monitoring; OLT; Topology; Visualization.

## Correspondent Email:

asri.wulandari@elektro.pnj.ac.id

**Abstrak.** Peningkatan kebutuhan layanan internet berkecepatan tinggi mendorong pemanfaatan teknologi Fiber to the Home (FTTH) yang mengandalkan perangkat Optical Line Terminal (OLT) sebagai pusat distribusi jaringan. Namun, sistem monitoring OLT yang digunakan saat ini masih bersifat statis dan tidak menampilkan relasi antar perangkat secara visual, sehingga menyulitkan operator dalam melakukan analisis kondisi jaringan secara menyeluruh. Penelitian ini bertujuan untuk mengembangkan sistem visualisasi topologi jaringan OLT berbasis website yang dilengkapi dengan fitur monitoring trafik dan deteksi perangkat. Sistem dibangun menggunakan stack teknologi Node.js, Express.js, React.js, PostgreSQL, dan library vis-network, serta terintegrasi dengan API dari Network Management System (NMS). Pengujian dilakukan melalui metode fungsional dan evaluasi berbasis ISO 25010 pada tiga aspek utama, yaitu functional suitability, usability, dan performance efficiency. Hasil pengujian menunjukkan bahwa sistem berhasil menampilkan topologi secara otomatis, mendeteksi kondisi *congestion*, perangkat *offline*, dan perangkat tanpa *redundancy link*, serta memberikan efisiensi waktu hingga lebih dari 70% dibandingkan metode manual. Nilai pengujian menunjukkan functional suitability sebesar 100%, skor usability 78,125 (kategori “Good”), dan performa rata-rata 78,6% dengan nilai LCP sebesar 1,89 detik.



Copyright © [JITET](#) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

**Abstract.** The growing demand for high-speed internet services has driven the adoption of Fiber to the Home (FTTH) technology, which relies on Optical Line Terminal (OLT) devices as the core of network distribution. However, existing OLT monitoring systems are often static and do not visually display device relationships, making it difficult for network operators to analyze network conditions comprehensively. This study aims to develop a web-based OLT network topology visualization system equipped with traffic monitoring and device detection features. The system was developed using a technology stack consisting of Node.js, Express.js, React.js, PostgreSQL, and the vis-network library, and is integrated with APIs from the Network Management System (NMS). System testing was carried out using functional methods and ISO 25010-based evaluation, focusing on three key aspects: functional suitability, usability, and performance efficiency. The results show that the system successfully visualizes the network topology automatically, detects congestion conditions, offline devices, and devices lacking redundancy links, and improves topology creation time efficiency by more than 70% compared to manual methods. The evaluation results indicate a functional suitability score of 100%, a usability score of 78.125 (classified as “Good”), and an average performance score of 78.6% with an LCP value of 1.89 seconds.

## 1. PENDAHULUAN

Dengan semakin tingginya permintaan akan layanan internet berkecepatan tinggi, khususnya di wilayah perkotaan dengan tingkat penetrasi pelanggan yang besar, teknologi *Fiber to the Home* (FTTH) muncul sebagai solusi utama dalam penyediaan koneksi internet yang andal. Teknologi ini memungkinkan koneksi langsung antara *Optical Line Terminal* (OLT) yang berada di sisi penyedia layanan (ISP) dengan *Optical Network Unit* (ONU) di sisi pelanggan, sehingga dapat menjamin kestabilan dan efisiensi akses internet. Dalam infrastruktur jaringan FTTH, OLT memiliki peran vital sebagai pusat kendali yang mengatur distribusi data ke seluruh pelanggan [1]. Oleh sebab itu, pemantauan terhadap performa dan kondisi OLT menjadi hal yang sangat penting dalam menjaga mutu layanan [2].

Namun demikian, sistem *monitoring* jaringan yang tersedia saat ini umumnya masih bersifat statis dan kurang mampu menyajikan visualisasi topologi secara menyeluruh. Operator jaringan membutuhkan sebuah alat bantu yang dapat menampilkan informasi struktur topologi jaringan OLT secara interaktif, sekaligus menampilkan status perangkat serta *traffic* jaringan OLT secara *real time*.

Melihat kondisi tersebut, diperlukan sebuah sistem berbasis *website* yang mampu menyajikan topologi jaringan secara otomatis dan interaktif, serta dilengkapi dengan fitur pemantauan trafik dan deteksi perangkat yang mengalami gangguan. Selain itu, sistem ini juga diharapkan dapat memberikan notifikasi berupa *pop-up* informasi apabila trafik pada *uplink* OLT melebihi ambang batas utilisasi tertentu. OLT dikategorikan mengalami kondisi *congestion* apabila tingkat penggunaan *bandwidth* pada jalur *uplink* mencapai lebih dari 80%.

Penelitian ini juga dilakukan untuk menjawab keterbatasan dari studi sebelumnya. Ilham dan Rosyid [3] merancang sistem visualisasi topologi jaringan berbasis SDN menggunakan Node.js dan vis.js dengan protokol sFlow, namun sistem tersebut hanya mendukung perangkat SDN tertentu dan kurang fleksibel untuk jaringan secara umum. Sementara itu, Romadhondaru dan Basuki [4]

mengembangkan visualisasi berbasis Laravel dan d3.js dengan data *routing* BGP, tetapi belum dilengkapi fitur *monitoring* performa jaringan secara *real-time*. Penelitian oleh Perrone *et al.* [5] menunjukkan visualisasi interaktif jaringan menggunakan Pyvis dan Vis.js, namun tanpa integrasi sistem *monitoring*. Sedangkan Hizriadi *et al.* [6] membangun sistem *monitoring* jaringan berbasis GIS dan SNMP, namun tidak menampilkan hubungan antar perangkat dalam topologi jaringan.

Dengan merujuk pada berbagai keterbatasan tersebut, tujuan dari penelitian ini adalah menempatkan diri pada posisi yang lebih spesifik, yaitu membangun sistem visualisasi topologi jaringan OLT berbasis *website* yang tidak hanya menyajikan hubungan antar perangkat secara otomatis dan interaktif, tetapi juga dilengkapi fitur *traffic monitoring* dan deteksi perangkat yang mengalami gangguan.

## 2. TINJAUAN PUSTAKA

### 2.1 Konsep Jaringan FTTH

*Fiber to the Home* (FTTH) merupakan teknologi jaringan telekomunikasi yang memanfaatkan media serat optik untuk menghantarkan data secara langsung ke setiap rumah pelanggan. Mekanisme transmisi berbasis cahaya ini memungkinkan pengiriman data dengan kecepatan tinggi, kapasitas *bandwidth* yang besar, serta tingkat kehilangan sinyal (*redaman*) yang sangat rendah. FTTH umumnya digunakan untuk layanan internet *broadband*, televisi kabel, dan telepon rumah dengan kualitas layanan yang tinggi [7].

Dalam arsitektur jaringan *Fiber to the Home* (FTTH), jaringan akses aktif memegang peranan penting dalam pengelolaan dan distribusi data pelanggan. *Optical Line Terminal* (OLT) berfungsi sebagai titik sentral yang menghubungkan sambungan serat optik dari sisi pelanggan, serta menangani komunikasi data baik arah *downstream* maupun *upstream*. Data dari pelanggan yang diterima oleh OLT kemudian diteruskan ke *Broadband Network Gateway* (BNG), yang bertanggung jawab atas proses autentikasi, pemberian alamat IP, pengaturan kualitas layanan, dan pemantauan penggunaan data untuk keperluan

penagihan. Jalur komunikasi antara OLT dan BNG inilah yang membentuk *uplink* aktif menuju jaringan agregasi dan *backbone* milik ISP [8].

## 2.2 Visualisasi Topologi Jaringan

Visualisasi adalah suatu pendekatan untuk menyajikan informasi dalam bentuk gambar, diagram, atau animasi yang bertujuan untuk mempermudah pemahaman. Secara umum, visualisasi merupakan proses transformasi data ke dalam representasi visual atau tabel, sehingga pola, karakteristik, dan hubungan antar elemen dalam data dapat lebih mudah dianalisis maupun dikomunikasikan [9].

Sementara itu, topologi jaringan menggambarkan pola atau susunan perangkat yang saling terhubung dalam suatu sistem jaringan komputer. Topologi ini tidak hanya menjelaskan struktur fisik koneksi antar perangkat, tetapi juga alur komunikasi logis serta mekanisme pergerakan data di antara perangkat-perangkat dalam suatu jaringan [10].

## 2.3 Monitoring Traffic Jaringan

*Monitoring* dapat diartikan sebagai proses pengamatan terhadap suatu informasi yang dianggap penting untuk diketahui dan dianalisis [11]. Dalam konteks jaringan, *monitoring* berfungsi untuk mengevaluasi kinerja dan kelayakan jaringan, termasuk menilai apakah kapasitas yang tersedia sudah mencukupi atau perlu ditingkatkan. Informasi yang diperoleh dari proses ini juga dapat dimanfaatkan oleh *administrator* dalam melakukan perancangan ulang terhadap jaringan yang telah ada. Beberapa parameter penting yang biasanya dipantau mencakup beban trafik pada perangkat serta status operasional perangkat jaringan, seperti apakah perangkat dalam kondisi aktif (*up*) atau tidak aktif (*down*) [12].

## 2.4 Bandwidth

*Bandwidth* merupakan indikator kapasitas maksimum suatu jaringan dalam mentransmisikan data selama periode waktu tertentu. Umumnya, satuan yang digunakan untuk mengukur *bandwidth* adalah bit per detik (bps), beserta kelipatannya seperti Kbps, Mbps, hingga Gbps, tergantung pada volume trafik yang terjadi pada jaringan. Besarnya *bandwidth* menunjukkan seberapa besar kemampuan suatu jalur atau koneksi jaringan dalam

mentransmisikan data dalam jumlah besar secara efisien [10].

## 2.5 Konsep Uplink dan Downlink Jaringan

*Uplink* merupakan proses pengiriman data dari perangkat jaringan menuju perangkat lain yang berada pada level hierarki lebih tinggi, seperti *server* atau *router* utama. Proses ini umumnya digunakan untuk menghubungkan perangkat dalam jaringan lokal (LAN) ke jaringan yang lebih luas, seperti jaringan inti atau internet. Sebaliknya, *downlink* merujuk pada aliran data dari perangkat yang berada di level lebih tinggi ke perangkat dengan level yang lebih rendah, misalnya dari *router* ke switch, atau dari switch ke perangkat pengguna seperti komputer [13].

## 2.6 Redundansi Link

Mekanisme yang diterapkan dalam jaringan untuk memastikan ketersediaan konektivitas secara optimal, khususnya dalam menghadapi kegagalan pada jalur utama. Dengan menggunakan protokol redundansi, ketika satu jalur mengalami kegagalan, jalur lain dapat berfungsi sebagai cadangan untuk menjaga konektivitas untuk tetap aktif dan dalam proses perpindahan *link* atau jalur, tidak membuat koneksi terputus [14].

## 2.7 Network Congestion

*Network congestion* terjadi ketika jumlah lalu lintas data yang dikirim secara bersamaan oleh sejumlah *node* dalam jaringan melebihi kapasitas transmisi yang tersedia, sehingga menimbulkan berbagai dampak negatif, seperti meningkatnya kehilangan paket, penundaan antrean yang lebih lama, serta turunnya rasio keberhasilan pengiriman data ke *node* tujuan [15].

## 2.8 Network Management System (NMS)

NMS adalah kumpulan aplikasi yang digunakan untuk mengelola infrastruktur jaringan berskala besar. NMS berfungsi dalam identifikasi perangkat, pengaturan konfigurasi, pemantauan kondisi operasional, serta penanganan gangguan pada perangkat jaringan, baik kabel maupun nirkabel. Sistem ini juga menyediakan informasi performa jaringan secara menyeluruh untuk mendukung optimasi dan penyesuaian konfigurasi sesuai kebutuhan pada suatu jaringan [16].

## 2.9 Efisiensi Waktu

Penelitian ini menerapkan pendekatan *efisiensi waktu* dengan membandingkan durasi pengerjaan antara metode manual dan sistem otomatis. Efisiensi waktu menjadi indikator penting untuk menilai sejauh mana sistem dapat mempercepat proses kerja dibandingkan metode konvensional. Mengacu pada studi Romadhoni (2022) [17], yang menunjukkan peningkatan efisiensi waktu melalui otomatisasi proses pemberian pakan ayam, prinsip serupa diterapkan dalam penelitian ini. Waktu pembuatan dan pembaruan topologi jaringan secara manual (menggunakan Google Slides) dibandingkan dengan sistem visualisasi otomatis, lalu dihitung persentase efisiensinya untuk mengukur peningkatan kecepatan kerja.

Rumus untuk menghitung efisiensi waktu dapat dituliskan pada persamaan 1 sebagai berikut:

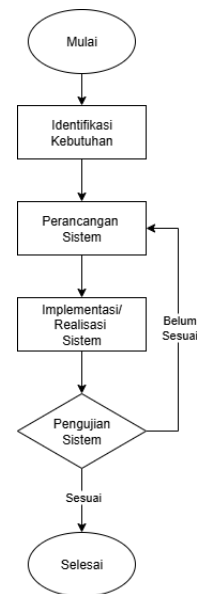
$$\text{Efisiensi Waktu (\%)} = \left( \frac{T_{\text{manual}} - T_{\text{sistem}}}{T_{\text{manual}}} \right) \times 100 \quad (1)$$

Keterangan pada persamaan:

- $T_{\text{manual}}$  = durasi pengerjaan dengan metode manual (menit).
- $T_{\text{sistem}}$  = durasi pengerjaan dengan sistem secara otomatis (menit).

## 3. METODE PENELITIAN

Penelitian ini menggunakan metode pengembangan sistem secara eksperimental dengan pendekatan studi kasus pada salah satu penyedia layanan internet berbasis FTTH. Proses pengembangan sistem visualisasi topologi dilakukan melalui beberapa tahapan, yaitu identifikasi kebutuhan sistem, perancangan arsitektur, implementasi, serta pengujian fungsional. Adapun tahapan penelitian ini dapat ditunjukkan pada Gambar 3.1 berikut ini.



Gambar 3. 1 Tahapan Penelitian

### 3.1 Identifikasi Kebutuhan

Identifikasi dilakukan berdasarkan permasalahan pada sistem *monitoring* perangkat OLT *congestion* yang telah berjalan di perusahaan, yaitu tampilan data *monitoring* bersifat statis dan tidak menampilkan relasi topologi jaringan antar perangkat. Hasil identifikasi menjadi dasar dalam perancangan sistem yang lebih informatif dan interaktif.

### 3.2 Perancangan Sistem

Sistem dirancang berbasis *website* dengan arsitektur *client-server*. Data *monitoring* diperoleh dari *Network Management System* (NMS) melalui API, dan data deteksi perangkat diperoleh dari data manual melalui *form input* yang diberikan oleh tim, berdasar kondisi aktual perangkat OLT.

Perancangan sistem yang akan dibuat meliputi 3 fungsi utama, yaitu:

- **Visualisasi Topologi Dinamis:** Merepresentasikan hierarki atau grafis koneksi jaringan berbasis pada koneksi fisik yang menghubungkan antara perangkat akses (OLT) dengan infrastruktur *core network*.
- **Monitoring Traffic:** Sistem dapat memantau utilisasi trafik *bandwidth* pada setiap *interface uplink port* yang aktif pada setiap OLT, menggunakan protokol *Simple Network Management Protocol* (SNMP). Sehingga, setiap *traffic* pada *interface uplink* di OLT

dapat terpantau dan diharapkan untuk memberikan peringatan untuk mencegah *network congestion* pada perangkat jaringan.

- Deteksi Perangkat Bermasalah: Sistem diharapkan dapat mengidentifikasi status setiap perangkat (*Up/Down*) dan memberikan peringatan jika status perangkat tersebut teridentifikasi mati (*down*). Hasil deteksi pada sistem diintegrasikan dengan visualisasi topologi untuk menandai perangkat yang bermasalah.

*Library* dan *framework* yang digunakan dalam pengembangan sistem berbasis *website* ini adalah Express.js dan Node.js sebagai *backend*, React.js untuk *frontend*, dan vis-network (Vis.js) untuk visualisasi topologi. Setelah itu, *database* untuk penyimpanan data yang digunakan adalah PostgreSQL.

### 3.3 Implementasi/Realisasi Sistem

Implementasi sistem dilakukan dengan menghubungkan sistem yang dikembangkan ke sumber data *monitoring* perangkat jaringan yang tersedia pada *Network Management System* (NMS) perusahaan melalui API. Data yang diambil berupa *hostname* perangkat, jenis perangkat, serta persentase utilisasi *traffic*. Data tersebut kemudian divisualisasikan dalam antarmuka berbasis *website* menggunakan *framework* React.js dan *library* vis-network.

Pada bagian deteksi perangkat, data ini diambil dari *database*, yang sebelumnya sudah berisi status kondisi setiap perangkat OLT yang diberikan secara manual oleh tim melalui *form input* pada *website*. Data status perangkat tersebut diberikan oleh tim, berdasarkan kondisi aktual perangkat di jaringan (*down/up*).

Setiap *node* pada visualisasi topologi mewakili satu perangkat jaringan dan ditampilkan secara otomatis sesuai struktur koneksi antar perangkat. Ketika pengguna mengklik *node* tertentu, akan muncul *pop-up/tooltip* yang berisi informasi lengkap seperti *hostname*, status koneksi, serta persentase utilisasi. Selain itu, sistem juga secara otomatis memberikan tanda berwarna merah pada *node* yang terdeteksi *down* dan memberikan peringatan apabila trafik *uplink* pada perangkat OLT telah melebihi 80%, yang menandakan kondisi *network congestion*.

### 3.4 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fitur utama berjalan sesuai dengan kebutuhan dan tujuan yang telah dirancang. Metode pengujian yang digunakan adalah pengujian fungsional, di mana setiap fitur diuji berdasarkan fungsionalitas yang diharapkan. Pengujian dilakukan menggunakan data perangkat aktual yang digunakan oleh penyedia layanan jaringan FTTH.

Setelah melakukan pengujian fungsional kepada 3 fungsi utama sistem. Pengujian selanjutnya adalah pengujian *website*, dilakukan untuk menilai kualitas perangkat lunak secara menyeluruh berdasarkan standar ISO/IEC 25010. Pengujian difokuskan pada tiga aspek utama, yaitu *Functional Suitability*, *Usability*, dan *Performance Efficiency*

## 4. HASIL DAN PEMBAHASAN

Tahap selanjutnya dalam penelitian ini adalah hasil dan pembahasan. Berikut ini adalah hasil realisasi atau implementasi pada sistem yang telah dibangun. Selanjutnya dilakukan pengujian dari masing-masing fungsi utama sistem yang meliputi visualisasi topologi jaringan, *monitoring traffic* dan deteksi perangkat.

Tahapan terakhir yaitu, dengan menguji sistem berdasarkan ISO 25010 untuk menilai kualitas dari sistem berbasis *website*. Pengujian difokuskan pada tiga aspek utama, yaitu *Functional Suitability*, *Usability*, dan *Performance Efficiency*.

Pengujian aspek *Functional Suitability* bertujuan untuk memastikan bahwa seluruh fungsi utama dalam sistem telah berjalan sesuai dengan kebutuhan dan spesifikasi yang telah dirancang.

Aspek *Usability* dievaluasi menggunakan metode *System Usability Scale* (SUS), yakni kuesioner standar yang terdiri dari 10 pernyataan untuk menilai tingkat kemudahan penggunaan suatu sistem. Kuesioner ini diisi oleh calon pengguna sistem, yaitu teknisi atau *engineer* jaringan yang nantinya akan menggunakan sistem dalam kegiatan operasional pekerjaan.

Aspek terakhir dalam pengujian *website* adalah *Performance Efficiency*, yang mengukur efisiensi sistem dalam hal kecepatan dan *response* saat digunakan. Pengujian ini difokuskan untuk mengetahui seberapa cepat

sistem memproses permintaan dari pengguna dan menampilkan halaman yang diminta. Setiap halaman tersebut diuji dari sisi waktu respons (*response time*) menggunakan fitur *Developer Tools* pada browser yaitu, Google Lighthouse.

#### 4.1 Hasil Visualisasi Topologi Jaringan

Pengujian terhadap sistem visualisasi topologi jaringan dilakukan berdasarkan beberapa skenario utama guna memastikan kesesuaian antara tampilan grafis dengan data perangkat yang dimasukkan melalui halaman *Devices Management*, serta menilai kemampuan sistem dalam merespons perubahan data dan menampilkan status perangkat secara visual. Pengamatan dilakukan pada tiga bagian utama berikut:

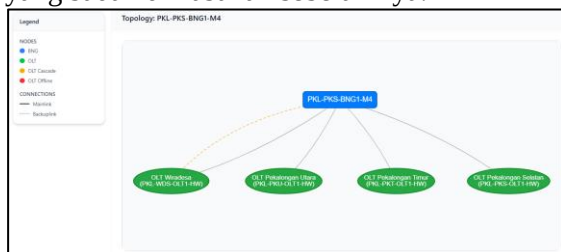
##### 4.1.1 Hasil Visualisasi Data Perangkat

Pengujian ini dilakukan dengan memasukkan data perangkat ke dalam sistem, termasuk informasi nama perangkat, *hostname*, relasi *uplink*, dan *port*. Pengujian ini dilakukan untuk topologi kota Pekalongan. Kota ini memiliki 4 perangkat OLT dengan 1 perangkat uplink BNG. Hasil *input* data perangkat akan disimpan di tabel *database* dan, terlihat pada Gambar 4.1 di bawah ini:

| NO | DEVICE                                    | LOKASI     | HOSTNAME        | UPLINK                             | BACKUPLINK | STATUS  | ACTIONS |
|----|-------------------------------------------|------------|-----------------|------------------------------------|------------|---------|---------|
| 1  | OLT Wiradesa<br>PRL-WDS-OLT1-HW           | Pekalongan | PRL-WDS-OLT1-HW | PRL-PKS-BNG1-M4<br>Prt: 2020/03/01 | YES        | Online  | + - ⚙️  |
| 2  | OLT Pekalongan Selatan<br>PRL-PKS-OLT1-HW | Pekalongan | PRL-PKS-OLT1-HW | PRL-PKS-BNG1-M4<br>Prt: 2020/03/01 | NO         | Offline | + - ⚙️  |
| 3  | OLT Pekalongan Utara<br>PRL-PKS-OLT1-HW   | Pekalongan | PRL-PKS-OLT1-HW | PRL-PKS-BNG1-M4<br>Prt: 2020/03/01 | NO         | Online  | + - ⚙️  |
| 4  | OLT Pekalongan Timur<br>PRL-PKT-OLT1-HW   | Pekalongan | PRL-PKT-OLT1-HW | PRL-PKS-BNG1-M4<br>Prt: 2020/03/01 | NO         | Online  | + - ⚙️  |

Gambar 4. 1 Daftar Perangkat Tersimpan

Setelah data tersimpan, pengguna mengakses halaman *Topology* dan memilih salah satu *uplink*. Berikut ini pada Gambar 4.2 adalah tampilan topologi dari kota Pekalongan yang sudah dimasukkan sebelumnya:



Gambar 4. 2 Hasil Visualisasi Topologi Jaringan

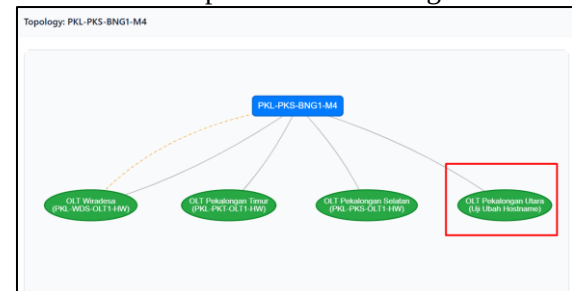
Gambar 4.2 menampilkan visualisasi topologi jaringan di kota Pekalongan dengan uplink utama PKL-PKS-BNG1-M4 yang terhubung ke empat OLT, yaitu Wiradesa,

Pekalongan Utara, Pekalongan Selatan, dan Pekalongan Timur.

Pada visualisasi topologi ini, *node* berwarna biru merepresentasikan *uplink* BNG, hijau untuk perangkat OLT aktif, kuning untuk OLT *cascade*, dan merah menunjukkan OLT yang dalam kondisi *offline*. Sementara itu, garis utuh menandakan *mainlink* dan garis putus-putus menunjukkan *backuplink* antar perangkat.

##### 4.1.2 Hasil Ubah Data pada Topologi

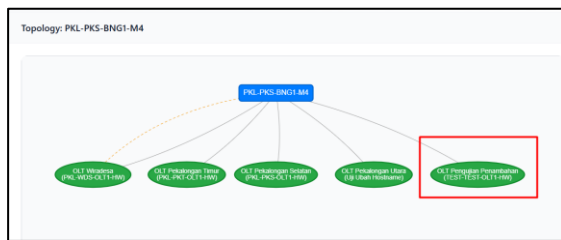
Pengujian ini bertujuan untuk melihat sistem merespon perubahan data pada perangkat. Perubahan yang diuji mencakup penambahan perangkat baru, pengubahan *hostname* atau relasi *uplink*, serta penghapusan perangkat. Berikut ini dilakukan percobaan mengubah nama *hostname* pada OLT Pekalongan Utara.



Gambar 4. 3 Hasil Perubahan Data Topologi

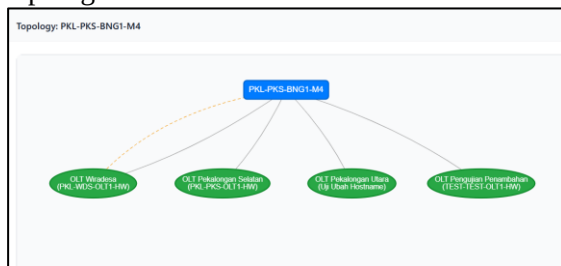
Saat ini dilakukan percobaan untuk mengubah *hostname* pada OLT Pekalongan Utara menjadi “Uji Ubah *Hostname*”. Kemudian pada Gambar 4.3, terlihat dengan membuka halaman *Topology* dan memilih *uplink* dari BNG pekalongan. Maka terlihat bahwa *hostname* dari OLT Pekalongan Utara telah berubah sesuai *update* data yang diberikan.

Selanjutnya, dilakukan dengan menambahkan perangkat baru, di halaman *Devices Management*, penambahan perangkat baru dengan nama “OLT Pengujian Penambahan” dengan *hostname* dan *port* sembarang, tetapi dengan *uplink* yang sama dengan perangkat OLT lainnya. Berikut ini pada Gambar 4.4 menampilkan topologi baru dengan perangkat yang bertambah.



Gambar 4. 4 Hasil Penambahan Perangkat di Topologi

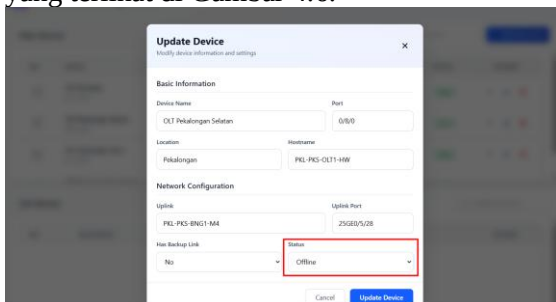
Percobaan terakhir dalam perubahan data di topologi ini adalah menghapus data salah satu *node* yang ada. Dilakukan percobaan untuk menghapus OLT Pekalongan Timur di *Devices Management*, dengan klik *button delete* dan konfirmasi penghapusan data. Gambar 4.5 di bawah ini menunjukkan hasil bahwa *node* OLT yang telah dihapus, sudah tidak tersedia lagi di topologi tersebut.



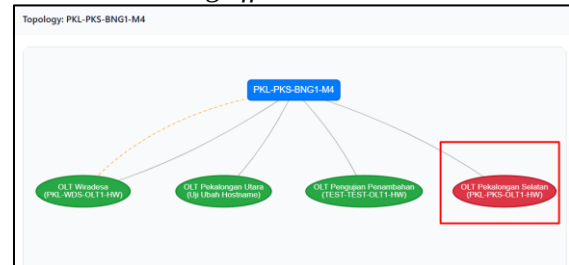
Gambar 4. 5 Hasil Hapus Data Perangkat di Topologi

#### 4.1.3 Hasil Visual Status Perangkat Melalui Perubahan Warna Node

Pada skenario ini, pengujian dilakukan dengan mengubah status perangkat (*online* atau *offline*) melalui halaman *Devices Management*, kemudian mengamati apakah sistem mengubah warna *node* sesuai status tersebut. Pada percobaan ini, dilakukan percobaan mengubah status menjadi *offline* OLT Pekalongan Selatan, yang terlihat di Gambar 4.6.

Gambar 4. 6 Mengubah Status Menjadi *Offline* pada Perangkat

Selanjutnya terlihat pada Gambar 4.7, bahwa *node* topologi sudah berubah warna menjadi merah. Warna merah pada *node* menunjukkan bahwa perangkat tersebut berstatus *down* atau sedang *offline*.



Gambar 4. 7 Hasil Ubah Status Perangkat di Topologi

## 4.2 Hasil Efisiensi Waktu Sistem

Pengujian efisiensi waktu dilakukan untuk membandingkan kecepatan pembuatan topologi jaringan menggunakan sistem otomatis dengan metode manual yang sebelumnya diterapkan oleh tim *Network Access*. Pada metode manual, topologi dibuat secara satu per satu menggunakan media seperti Google Slide, mulai dari menambahkan node, garis untuk menghubungkan perangkat, hingga menyusun label dan posisi secara visual.

Sebaliknya, melalui sistem visualisasi topologi OLT berbasis *website* yang dikembangkan, pengguna cukup memasukan data perangkat dan relasi *uplink* melalui halaman *Devices*. Setelah data disimpan, visualisasi topologi akan ditampilkan secara otomatis sesuai struktur jaringan yang dipilih, lengkap dengan posisi *node* dan koneksi antar perangkat.

Telah dilakukan pembuatan ulang topologi dari kedua kota yaitu Pekalongan dan Garut. Tabel 4.1 berikut menyajikan hasil pengukuran waktu yang dibutuhkan dalam proses pembuatan topologi menggunakan kedua metode tersebut.

Tabel 4. 1 Perbandingan Waktu Pengerjaan Metode Manual dan Sistem

| Kota       | Jumlah Perangkat | Durasi Manual (Menit) | Durasi Sistem (Menit) | Efisiensi Waktu (%) |
|------------|------------------|-----------------------|-----------------------|---------------------|
| Pekalongan | 5                | 17                    | 4,5                   | 73,5 %              |
| Garut      | 8                | 33                    | 9,5                   | 71,2 %              |



Keterangan:

- Durasi Manual: waktu pengerjaan menggunakan metode Google Slide.
- Durasi Sistem: waktu pengerjaan menggunakan sistem visualisasi otomatis.
- Efisiensi dihitung menggunakan persamaan:

$$Efficiency(\%) = \left( \frac{T_{manual} - T_{sistem}}{T_{manual}} \right) \times 100$$

Keterangan:

- $T_{manual}$  = durasi pengerjaan dengan metode manual
- $T_{sistem}$  = durasi pengerjaan dengan sistem otomatis

Efisiensi waktu kota Pekalongan:

Diketahui:  $T_{manual} = 17, T_{sistem} = 4,5$

$$Efficiency(\%) = \left( \frac{17-4,5}{17} \right) \times 100 = 73,5$$

Efisiensi waktu kota Garut:

Diketahui:  $T_{manual} = 33, T_{sistem} = 9,5$

$$Efficiency(\%) = \left( \frac{33-9,5}{33} \right) \times 100 = 71,2$$

Berdasarkan hasil pada Tabel 4.1, sistem visualisasi topologi jaringan yang dikembangkan terbukti mampu meningkatkan efisiensi waktu secara signifikan dibandingkan metode manual menggunakan Google Slide.

Pada pengujian di kota Pekalongan dengan 5 perangkat, metode manual memerlukan waktu 17 menit, sedangkan sistem otomatis hanya membutuhkan 4,5 menit, menghasilkan efisiensi sebesar 73,5%. Di kota Garut dengan 8 perangkat, waktu pengerjaan manual mencapai 33 menit, sementara sistem otomatis menyelesaikannya dalam 9,5 menit, dengan efisiensi 71,2%.

Hasil ini menunjukkan bahwa sistem mampu menghemat waktu lebih dari 70% terlepas dari jumlah perangkat, sekaligus meminimalkan potensi kesalahan yang umum terjadi pada proses manual.

### 4.3 Hasil Monitoring Traffic

Pengujian ini dilakukan untuk memastikan bahwa sistem mampu memantau kondisi *congestion* pada OLT secara tepat dan akurat. Data monitoring diperoleh melalui API eksternal yang menyediakan informasi persentase utilisasi *bandwidth* dari masing-masing perangkat OLT. Informasi tersebut kemudian ditampilkan secara informatif dalam

bentuk tabel pada halaman *Monitoring*, sehingga mudah dipahami oleh pengguna.

| ID | HOSTNAME         | TYPE | % UTILIZATION | STATUS  |
|----|------------------|------|---------------|---------|
| H1 | 1980-024-0271-H1 | OLT  | 100%          | Congest |
| H2 | 8180-081-0271-H2 | OLT  | 100%          | Congest |
| H3 | 8180-024-0271-H3 | OLT  | 100%          | Congest |
| H4 | 3PT-1980-0271-H4 | OLT  | 100%          | Congest |
| H5 | 1980-080-0271-H5 | OLT  | 100%          | Congest |
| H6 | 7980-024-0271-H6 | OLT  | 100%          | Congest |
| H7 | 1980-080-0271-H7 | OLT  | 100%          | Congest |
| H8 | 024-7980-0271-H8 | OLT  | 100%          | Congest |

Gambar 4. 8 Hasil Tampilan Data di Halaman *Monitoring*

Gambar 4.8 menunjukkan bahwa sistem berhasil menampilkan halaman *monitoring* yang memuat data OLT yang mengalami *congestion*, berdasarkan informasi yang diambil dari API. Data tersebut disajikan dalam bentuk tabel yang mencantumkan *hostname* dan persentase utilisasi *bandwidth* dari masing-masing perangkat. Seluruh perangkat yang ditampilkan berada dalam kondisi *congest*, yaitu ketika *traffic* pada uplink OLT mendekati atau melebihi ambang batas kapasitas sebesar 80%.

### 4.4 Hasil Deteksi Perangkat

Pengujian ini bertujuan untuk mengevaluasi kemampuan sistem dalam mendeteksi perangkat OLT yang mengalami gangguan, baik akibat *congestion*, tidak adanya jalur redundansi, maupun status *offline*. Selain itu, diuji pula respons visual sistem dalam menampilkan kondisi tersebut pada topologi, seperti perubahan tampilan *node* saat terjadi *network congestion* pada perangkat tersebut.

#### 4.4.1 Hasil Deteksi Perangkat OLT (Congestion, Need Redundancy, dan Status Offline/Down).

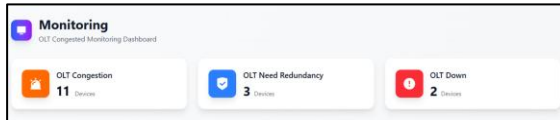
Pengujian dilakukan dengan mensimulasikan kondisi gangguan pada perangkat, yaitu dengan mengatur nilai *backuplink* menjadi NO dan status menjadi *offline*. Berdasarkan Gambar 4.9, terdapat 2 perangkat yang diset *offline* dan 3 perangkat tanpa jalur *redundancy*.

| ID | HOSTNAME                  | LINKAGE    | TYPE | % UTILIZATION | BACKUPLINK | STATUS  | ACTIONS |
|----|---------------------------|------------|------|---------------|------------|---------|---------|
| 1  | OLT Pekalongan Utara      | Pekalongan | OLT  | 100%          | NO         | Congest | +       |
| 2  | OLT Pekalongan Selatan    | Pekalongan | OLT  | 100%          | NO         | Congest | +       |
| 3  | OLT Cirebon               | Pekalongan | OLT  | 100%          | NO         | Offline | +       |
| 4  | OLT Pekalongan Selatan II | Pekalongan | OLT  | 100%          | NO         | Offline | +       |

Gambar 4. 9 Mengubah Kondisi *Backuplink* dan Status Perangkat



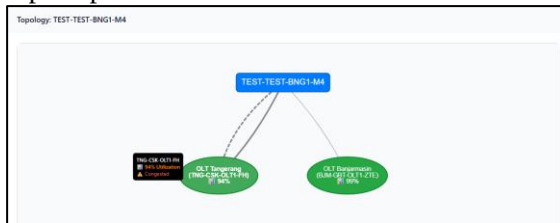
Selanjutnya, hasil ditampilkan pada halaman monitoring di Gambar 4.10, yang secara akurat menampilkan jumlah perangkat sesuai kondisi: 11 perangkat mengalami *congestion*, 3 perangkat tanpa *backup link*, dan 2 perangkat dalam kondisi *down*. Hal ini menunjukkan bahwa sistem berhasil mendeteksi dan mengklasifikasikan kondisi perangkat dengan benar.



Gambar 4. 10 Hasil Tampilan Deteksi Perangkat

#### 4.4.2 Hasil Tampilan Node Topologi Saat Terjadi Congestion

Pengujian ini bertujuan untuk menilai apakah sistem visualisasi topologi dapat secara otomatis menampilkan status *congested* ketika trafik uplink pada OLT melebihi ambang batas utilisasi (>80%). Dua perangkat OLT yang digunakan sebagai sampel adalah BJM-GBT-OLT1-ZTE (99%) dan TNG-CSK-OLT1-FH (94%), yang datanya dimasukkan melalui halaman *Devices Management* dan kemudian divisualisasikan pada halaman *Topology*, seperti pada Gambar 4.11 di bawah ini.



Gambar 4. 11 Hasil Topologi Mendeteksi Network Congestion

Hasil menunjukkan bahwa sistem berhasil menampilkan status *congested* dalam bentuk *tooltip* interaktif yang muncul saat pengguna mengarahkan kursor ke *node* perangkat. *Tooltip* tersebut menyajikan informasi berupa *hostname*, nilai utilisasi, dan indikator visual berupa ikon berwarna oranye sebagai tanda kondisi *congestion*.

Sistem terbukti mampu mendeteksi dan memvisualisasikan perangkat yang melewati ambang batas *bandwidth* secara konsisten, serta menampilkan informasi tersebut secara *real-time* dalam bentuk *pop-up* pada tampilan topologi. Dengan demikian, pengujian ini dinyatakan berhasil karena sistem memberikan

umpan balik visual yang informatif terhadap kondisi *network congestion*.

#### 4.5 Hasil Pengujian Sistem Website dengan Berdasarkan ISO 25010

Pengujian *website* difokuskan pada tiga aspek utama yang menentukan kualitas dan kinerja sistem, yaitu *functional suitability*, *usability*, dan *performance efficiency*. Ketiga aspek ini digunakan untuk mengevaluasi sistem secara menyeluruh, guna memastikan bahwa sistem tidak hanya berfungsi sesuai kebutuhan pengguna, tetapi juga efisien dan mudah diakses dalam penggunaannya.

##### 4.5.1 Functional Suitability

Pengujian aspek *functional suitability* dilakukan untuk memastikan seluruh fitur sistem berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Evaluasi mencakup fungsi inti seperti autentikasi pengguna, manajemen perangkat, visualisasi topologi, serta *monitoring* trafik dan status perangkat OLT.

Tabel 4. 2 Hasil Ketercapaian Aspek *Functional Suitability*

| Ketercapaian/Keberhasilan |       |
|---------------------------|-------|
| Ya                        | Tidak |
| 47                        | 0     |

Matriks *Feature Completeness website* berdasarkan aspek *functional suitability* dapat dihitung sesuai pada persamaan 2, dengan hasil sebagai berikut:

$$X = \frac{i}{p} = \frac{47}{47} = 1 \quad (2)$$

Keterangan:

- $i$  = jumlah fungsi yang berhasil diimplementasikan
- $p$  = jumlah fungsi yang dirancang

Setiap fitur diuji menggunakan metode penilaian *biner*, dengan nilai 1 jika fungsi berhasil dijalankan dan 0 jika tidak. Berdasarkan pengujian terhadap 47 fitur, seluruhnya berhasil berfungsi sebagaimana mestinya, sehingga sistem memperoleh nilai kelengkapan fungsional sebesar 100%. Hasil ini

menunjukkan bahwa sistem telah sepenuhnya memenuhi aspek *functional suitability* dan tergolong sangat layak digunakan dari sisi fungsionalitas.

#### 4.5.2 Performance Efficiency

Pengujian *performance efficiency* bertujuan untuk menilai seberapa cepat dan responsif sistem dalam merespons interaksi pengguna, khususnya pada halaman dengan aktivitas tinggi seperti visualisasi topologi dan *monitoring* trafik. Evaluasi dilakukan menggunakan alat Lighthouse pada Google Chrome DevTools, dengan fokus pada metrik *Performance Score* dan *Largest Contentful Paint* (LCP).

Hasil dari pengujian pada masing-masing halaman disajikan dalam Tabel 4.3. Pengujian ini dilakukan sebanyak 3 kali pada setiap halaman utama *website*.

Tabel 4. 3 Hasil Pengujian *Performance Efficiency*

| No        | Halaman Website    | Skor Performance (%) |    |    |       | Skor LCP (S) |     |     |       |
|-----------|--------------------|----------------------|----|----|-------|--------------|-----|-----|-------|
|           |                    | 1                    | 2  | 3  | Total | 1            | 2   | 3   | Total |
| 1         | Login              | 84                   | 82 | 76 | 80.6  | 1.7          | 1.6 | 1.9 | 1.73  |
| 2         | Dashboard          | 78                   | 75 | 78 | 77    | 2.0          | 2.0 | 1.9 | 1.96  |
| 3         | Users Management   | 81                   | 78 | 79 | 79.3  | 1.9          | 1.9 | 1.9 | 1.9   |
| 4         | Devices Management | 76                   | 79 | 79 | 78    | 1.9          | 1.8 | 1.9 | 1.86  |
| 5         | Network Topology   | 80                   | 79 | 79 | 79.3  | 1.9          | 1.9 | 1.9 | 1.9   |
| 6         | Monitoring         | 78                   | 79 | 76 | 77.6  | 2.0          | 1.9 | 2.1 | 2.0   |
| Rata-Rata |                    | 78.6                 |    |    |       | 1.89         |     |     |       |

Setiap halaman utama diuji sebanyak tiga kali, dan hasil pengukuran menunjukkan skor performa berkisar antara 75–84%, dengan rata-rata 78,6%. Berdasarkan standar penilaian Lighthouse, nilai ini masuk dalam kategori sedang, yang berarti performa sistem tergolong cukup baik.

Metrik LCP menunjukkan rata-rata waktu pemuatan sebesar 1,89 detik, yang masih berada dalam batas wajar dan termasuk dalam kategori *moderate*. Meskipun belum tergolong sangat cepat, waktu ini tetap dinilai layak karena tidak melewati ambang batas 2,4 detik yang diklasifikasikan sebagai *slow*.

#### 4.5.3 Usability

Pengujian pada aspek *usability* dilakukan untuk mengetahui sejauh mana *website* dapat digunakan oleh pengguna secara efektif, efisien, serta memberikan pengalaman

penggunaan yang memuaskan dalam konteks operasionalnya. Pengujian ini mengacu pada metode standar *System Usability Scale* (SUS).

Tabel 4. 4 Hasil Pengujian Aspek *Usability*

| Responden       | Poin Pertanyaan |   |   |   |   |   |   |   |   |    | Total  | Nilai SUS:  |
|-----------------|-----------------|---|---|---|---|---|---|---|---|----|--------|-------------|
|                 | 1               | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Skor   | Total x 2.5 |
| 1               | 4               | 3 | 4 | 3 | 4 | 3 | 3 | 3 | 4 | 4  | 23     | 57,5        |
| 2               | 5               | 2 | 5 | 2 | 5 | 1 | 4 | 2 | 5 | 3  | 34     | 85          |
| 3               | 4               | 1 | 5 | 2 | 4 | 1 | 4 | 1 | 5 | 3  | 34     | 85          |
| 4               | 5               | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1  | 40     | 100         |
| 5               | 1               | 1 | 5 | 4 | 5 | 1 | 5 | 1 | 5 | 2  | 32     | 80          |
| 6               | 4               | 2 | 5 | 4 | 4 | 5 | 3 | 1 | 4 | 4  | 24     | 60          |
| 7               | 5               | 1 | 5 | 2 | 4 | 2 | 5 | 1 | 5 | 2  | 36     | 90          |
| 8               | 4               | 2 | 4 | 2 | 3 | 3 | 4 | 2 | 4 | 3  | 27     | 67,5        |
| Jumlah SUS      |                 |   |   |   |   |   |   |   |   |    | 625    |             |
| Nilai rata-rata |                 |   |   |   |   |   |   |   |   |    | 78.125 |             |

Berdasarkan hasil pengisian kuesioner oleh 8 responden, diperoleh skor rata-rata SUS sebesar 78.125, yang menurut standar interpretasi berada pada kategori “Good”. Hal ini menunjukkan bahwa sistem dinilai memiliki tingkat kemudahan penggunaan yang baik, mudah dipahami, dan sesuai untuk mendukung operasional pengelolaan perangkat jaringan OLT.

## 5. KESIMPULAN

Kesimpulan berdasarkan pembahasan yang telah dilakukan adalah:

- Sistem visualisasi topologi jaringan berbasis *website* yang dikembangkan berhasil merepresentasikan struktur koneksi antar perangkat secara otomatis dan interaktif, serta dilengkapi fitur *monitoring* trafik dan deteksi perangkat OLT bermasalah, baik yang mengalami *congestion*, tidak memiliki *backup link*, maupun dalam kondisi *offline*.
- Sistem mampu meningkatkan efisiensi waktu pengerjaan topologi jaringan secara signifikan. Berdasarkan pengujian di dua kota, waktu pengerjaan berkurang lebih dari 70% dibandingkan metode manual, sehingga sistem dinilai efektif dan efisien dalam mendukung operasional teknis pekerjaan sehari-hari.
- Pengujian *monitoring* trafik menunjukkan bahwa sistem berhasil menampilkan status perangkat yang mengalami *congestion* secara *real-time* melalui data yang diambil dari API eksternal, serta menyajikan

informasi secara informatif dan responsif di halaman *monitoring*.

- d. Fitur deteksi perangkat berjalan dengan baik, ditandai dengan kemampuan sistem menampilkan status perangkat melalui perubahan warna node dan *tooltip* informatif, sehingga mempermudah identifikasi gangguan dalam topologi jaringan.
- e. Berdasarkan pengujian kualitas perangkat lunak menggunakan standar ISO 25010, sistem memperoleh nilai:
  - *Functional Suitability*: 100% ketercapaian fungsional (47 dari 47 fungsi berjalan dengan baik),
  - *Performance Efficiency*: Skor rata-rata 78,6% dan waktu muat rata-rata 1,89 detik, yang termasuk dalam kategori sedang,
  - *Usability*: Skor SUS sebesar 78,125 yang berada dalam kategori "Good", menandakan sistem mudah digunakan dan dipahami oleh pengguna.
- f. Secara keseluruhan, sistem dinilai cukup layak untuk diterapkan dalam lingkungan operasional penyedia layanan jaringan FTTH. Ke depannya, sistem ini masih memiliki ruang pengembangan, salah satunya dengan menambahkan fitur integrasi autentikasi langsung ke NMS. Dengan fitur tersebut, diharapkan sistem dapat secara otomatis memperoleh data status perangkat secara *real-time* tanpa memerlukan *input* manual dari *administrator*.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada PT Eka Mas Republik atas dukungan data, akses sistem, dan kesempatan yang telah diberikan dalam pelaksanaan penelitian ini. Ucapan terima kasih juga disampaikan kepada dosen pembimbing serta seluruh pihak yang telah membantu secara langsung maupun tidak langsung dalam proses penyusunan dan pengembangan sistem ini.

## DAFTAR PUSTAKA

- [1] P. Muliandhi, H. Faradiba, and B. A. Nugroho, "Analisa Konfigurasi Jaringan FTTH dengan Perangkat OLT Mini untuk Layanan Indihome di PT. Telkom Akses Witel Semarang," *Elektrika*, vol. 12, no. 1, pp. 7–14, 2020.
- [2] Z. Ramadan, Y. Yulindon, Y. Yustini, and A. A. Asril, "Installation and Activation of Fiber To The Home (FTTH) Network Using Gigabit Passive Optical Network (GPON) Technology and Quality of Service (QoS) Analysis," *JATAED: Journal of Appropriate Technology for Agriculture, Environment, and Development*, vol. 2, no. 1, pp. 17–24, Nov. 2024, doi: 10.62671/jataed.v2i1.62.
- [3] M. Ilham and N. R. Rosyid, Pengembangan Aplikasi Pemantauan Jaringan Berbasis Web pada Software-Defined Networking dengan Protokol sFlow," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 8, no. 6, pp. 1117–1126, Dec. 2021, doi: 10.25126/jtiik.202183367.
- [4] R. J. Romadhondaru and A. Basuki, "Visualisasi Topologi Jaringan berdasarkan Data Routing Border Gateway Protocol," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 9, pp. 4329–4338, Sep. 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [5] G. Perrone, J. Unpingco, and H. Lu, "Network visualizations with Pyvis and VisJS," Jun. 2020, [Online]. Available: <http://arxiv.org/abs/2006.04951>
- [6] A. Hizriadi, R. Shiddiq, I. Jaya, and S. Prayudani, "Network Device Monitoring System based on Geographic Information System dan Simple Network Management Protocol," *Journal of Informatics and Telecommunication Engineering*, vol. 3, no. 2, pp. 216–223, Jan. 2020, doi: 10.31289/jite.v3i2.3187.
- [7] C. Setyadi and D. W. Chandra, "Analysis of ODP point placement using algorithm K-means in RW. 01 Gendongan village (case study: PT. Indomedia)," *Journal of Soft Computing Exploration*, vol. 5, no. 3, pp. 263–270, Sep. 2024, doi: 10.52465/joscex.v5i3.473.
- [8] L. Peterson, C. Cascone, B. O'Connor, T. Vachuska, and B. Davie, *Software-Defined Networks: A Systems Approach*, 6th ed. 2022.
- [9] M. Radhi, D. Ryan Hamonangan Sitompul, S. Hamonangan Sinurat, and E. Indra, "Analisis Big Data dengan Metode Exploratory Data Analysis (EDA) dan Metode Visualisasi Menggunakan Jupyter Notebook," *Jurnal Sistem Informasi dan Ilmu Komputer Prima*, vol. 4, no. 2, 2021, doi: 10.34012/jurnalsisteminformasidanilmukomputer.v4i2.2475.
- [10] R. Octaviana and B. Soewito, "Perancangan Ulang Topologi Jaringan Dengan Kerangka

- Kerja Ppdioo,” *Teknologi: Jurnal Ilmiah Sistem Informasi*, vol. 13, no. 1, pp. 33–41, Jul. 2023, doi: 10.26594/teknologi.v13i1.3852.
- [11] N. Sari and D. Cahyani, “Perancangan Sistem Informasi Monitoring Sertifikat Menggunakan Extreme Programming,” *Jurnal Ilmiah Computer Science*, vol. 1, no. 1, pp. 1–6, Jul. 2022, doi: 10.58602/jics.v1i1.1.
- [12] R. Rahmat, R. Wiji Wahyuningrum, E. Haerullah, and S. Sodikin, “Analisis Monitoring Sistem Jaringan Komputer Menggunakan Aplikasi Spiceworks,” *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, vol. 9, no. 1, pp. 44–52, May 2022, doi: 10.30656/prosisko.v9i1.4671.
- [13] J. Jeffri, S. Hartanto, and S. P. Santosa, “Analisis Power Link Budget pada Jaringan FTTH di Kelurahan Jatirasa Bekasi,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, pp. 4028–4037, Oct. 2024, doi: 10.23960/jitet.v12i3S1.5247.
- [14] E. K. Silalahi, Y. C. Sitanggang, E. Suryaningsih, and D. Kiswanto, “Implementasi dan Analisis Protokol HSRP dan VRRP dalam Meningkatkan Redundansi Gateway pada Jaringan Virtual,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, pp. 1431–1444, Apr. 2025, doi: 10.23960/jitet.v13i2.6474.
- [15] M. Mahajan and Dr. S. Kaur, “An Intelligent Path Evaluation Algorithm for Congestion Control in Wireless Sensor Networks,” *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 6, pp. 3106–3114, Apr. 2021, doi: 10.17762/turcomat.v12i6.6953.
- [16] A. B. R. Pratama, G. Sukadarmika, and N. P. Sastra, “Rancang Bangun Evolved Network Management System Application (EVOMAC) pada Ekosistem Infrastruktur Jaringan Universitas Udayana,” *Majalah Ilmiah Teknologi Elektro*, vol. 21, no. 1, p. 89, Jul. 2022, doi: 10.24843/mite.2022.v21i01.p13.
- [17] A. Romadhoni, “Efisiensi Waktu Pemberian Pakan Ayam Pada Mesin Hopper,” *Journal of Science Nusantara*, vol. 2, no. 1, pp. 44–51, Aug. 2022, doi: 10.28926/jsnu.v2i1.292.