

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Untuk menerapkan sistem ini, penulis mengacu pada berbagai penelitian sebelumnya yang relevan. Penelitian terkait adalah kumpulan sumber penelitian sebelumnya yang mendukung dan menjadi acuan penulis, serta menghindari duplikasi hasil. Selain itu, penelitian tersebut menginspirasi penemuan baru atau pembaruan dengan metode ilmiah. Berikut adalah beberapa penelitian yang dilakukan dalam 5 tahun terakhir yang digunakan sebagai referensi :

Tabel 2. 1 Penelitian Terkait

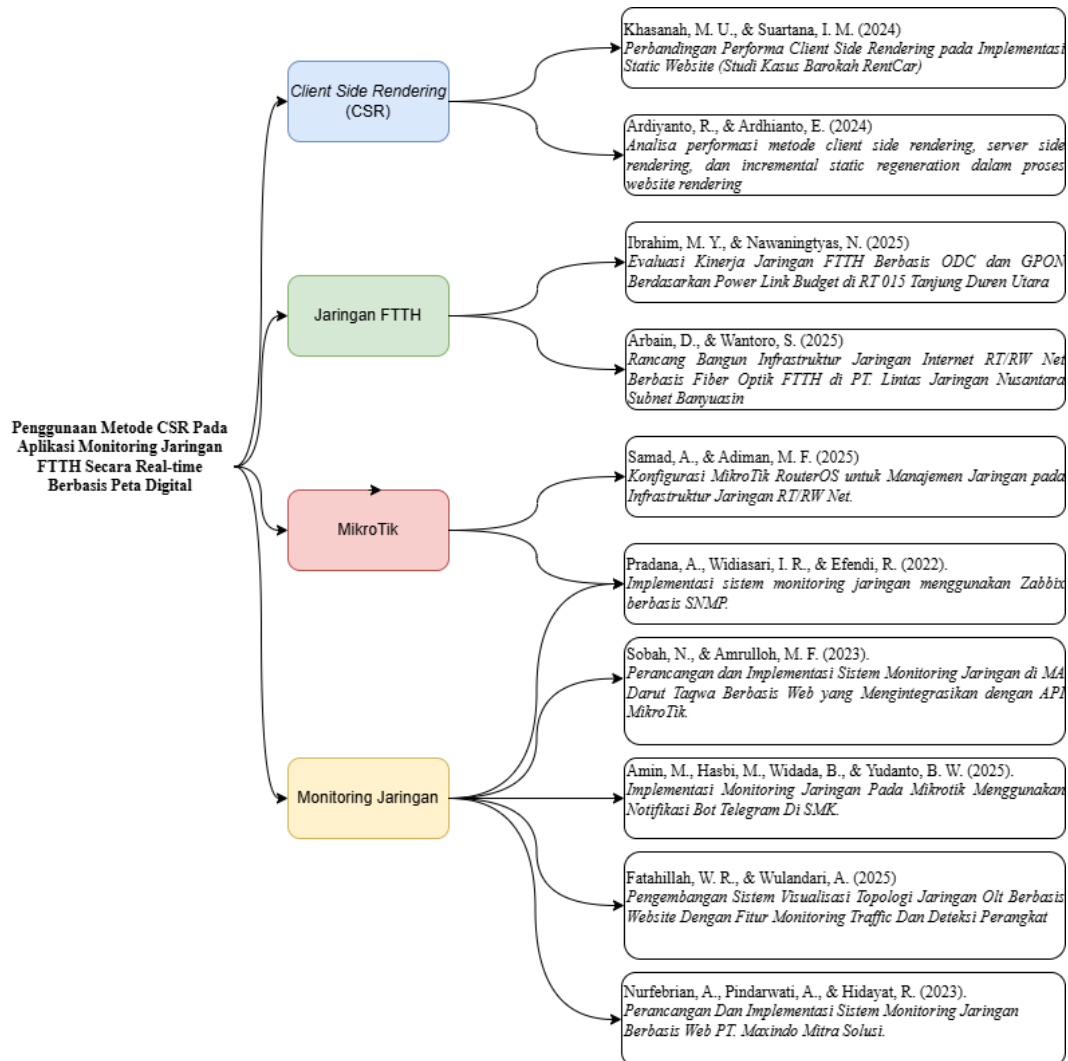
No	Nama Peneliti & Tahun	Keterangan Penelitian
1	Khasanah, M. U., & Suartana, I. M. (2024)	Judul : <i>Perbandingan Performa Client Side Rendering pada Implementasi Static Website (Studi Kasus Barokah RentCar)</i> Metode : Eksperimen. Hasil Penelitian : - Penelitian ini membandingkan performa metode <i>Client Side Rendering</i> (CSR) dan <i>Incremental Static Regeneration</i> (ISR) pada website statis. - Hasil penelitian menunjukkan bahwa ISR memiliki performa <i>loading</i> lebih baik dibandingkan CSR. - Penelitian ini menjadi dasar dalam mengevaluasi performa <i>rendering</i> website pada penelitian yang akan dilakukan.
2	Ardiyanto, R., & Ardianto, E. (2024)	Judul : <i>Analisa performasi metode client side rendering, server side rendering, dan incremental static regeneration dalam proses website rendering</i> Metode : Eksperimen. Hasil Penelitian : - Penelitian ini membandingkan metode CSR, SSR, dan ISR menggunakan <i>framework Next.js</i>. - Hasil penelitian menunjukkan bahwa metode <i>rendering</i> mempengaruhi

		performa website, terutama pada waktu <i>loading</i> dan ukuran halaman.
3	Ibrahim, M. Y., & Nawaningtyas, N. (2025)	Judul : <i>Evaluasi Kinerja Jaringan FTTH Berbasis ODC dan GPON Berdasarkan Power Link Budget di RT 015 Tanjung Duren Utara.</i> Metode : Kuantitatif, Observasi lapangan Hasil Penelitian : - Penelitian ini mengevaluasi kinerja jaringan FTTH berbasis ODC dan GPON/EPON sebagai jalur distribusi layanan ke pelanggan. - Komponen kabel, splitter, konektor yang berkualitas dapat meningkatkan performa distribusi power redaman. - Topologi FTTH yang menggunakan teori dari penelitian ini akan meningkatkan kualitas jaringan internet.
4	Arbain, D., & Wantoro, S. (2025)	Judul : <i>Rancang Bangun Infrastruktur Jaringan Internet RT/RW Net Berbasis Fiber Optik FTTH di PT. Lintas Jaringan Nusantara Subnet Banyuasin</i> Metode : Penelitian terapan (applied research) Hasil Penelitian : - Penelitian ini berfokus pada rencana pembangunan jaringan internet FTTH pada mitra PT. Lintas Jaringan Nusantara. - Topologi jaringan, konfigurasi, infrastruktur yang akan digunakan menjadi titik pembahasan dalam penelitian ini. - Implementasi MikroTik OS sebagai <i>server</i> jaringan dan konfigurasi PPPoE sebagai manajemen <i>user</i> pelanggan.
5	Samad, A., & Adiman, M. F. (2025)	Judul : <i>Konfigurasi MikroTik RouterOS untuk Manajemen Jaringan pada Infrastruktur Jaringan RT/RW Net.</i> Metode : Eksperimen Hasil Penelitian :

		- Penelitian ini berfokus pada MikroTik OS sebagai <i>server</i> manajemen jaringan. - Kesimpulan yang dapat diambil dari penelitian ini antara lain adalah : 1. Integrasi PPPoE multi-profil, <i>load balancing</i>, <i>firewall</i> terbukti efektif. 2. Meningkatkan efisiensi jaringan, konektivitas stabil, pemerataan <i>bandwidth</i>. 3. Sangat efisien untuk diterapkan pada ISP kelas kecil.
6	Pradana, A., Wideasari, I. R., & Efendi, R. (2022).	Judul : <i>Implementasi sistem monitoring jaringan menggunakan Zabbix berbasis SNMP.</i> Software : <i>Zabbix server v5.0</i> Hasil Penelitian : - Penelitian ini menjelaskan instalasi zabbix pada ubuntu <i>server</i> sehingga dapat membaca data dari MikroTik dengan menggunakan protokol <i>Simple Network Management Protocol</i> (SNMP) pada layer aplikasi (OSI layer 7) berbasis TCP/IP. - <i>Monitoring</i> berbasis <i>real-time</i> dapat berjalan maksimal, beberapa fitur yang dapat digunakan adalah : 1. Monitor <i>traffic</i> jaringan masuk dan keluar. 2. Mendeteksi problem jaringan secara <i>real-time</i>. 3. Menampilkan <i>log error</i> dan <i>trigger</i> secara detail. 4. Memberikan informasi performa <i>server</i> dan perangkat jaringan.
7	Sobah, N., & Amrulloh, M. F. (2023).	Judul : <i>Perancangan dan Implementasi Sistem Monitoring Jaringan di MA Darut Taqwa Berbasis Web yang Mengintegrasikan dengan API MikroTik.</i> Metode : Observasi Hasil Penelitian :

		- Penelitian ini berhasil mengembangkan sistem <i>monitoring</i> jaringan berbasis web menggunakan API MikroTik yang mampu menampilkan <i>monitoring</i> hotspot dan <i>user</i> PPPoE secara <i>real-time</i>.
8	Amin, M., Hasbi, M., Widada, B., & Yudanto, B. W. (2025).	Judul : <i>Implementasi Monitoring Jaringan Pada Mikrotik Menggunakan Notifikasi Bot Telegram Di SMK.</i> Metode : Analisa, Desain, Simulasi Prototyping, Implementasi. Software : Telegram Hasil Penelitian : - Hasil dari implementasi tersebut antara lain : 1. Sistem berhasil mendeteksi perubahan status perangkat secara otomatis (<i>online/offline</i>) 2. Notifikasi dikirim ke telegram secara <i>real-time</i> 3. Administrator jaringan tidak perlu melakukan pengecekan secara manual.
9	Fatahillah, W. R., & Wulandari, A. (2025)	Judul : <i>Pengembangan Sistem Visualisasi Topologi Jaringan Olt Berbasis Website Dengan Fitur Monitoring Traffic Dan Deteksi Perangkat.</i> Metode : Eksperimental Software : <i>Node.js, Express.js, React.js, SNMP (simple network management protocol), Google Lighthouse (chrome devtools), google slide</i> Hasil Penelitian : - Penelitian ini mengembangkan sistem visualisasi topologi jaringan berbasis website dengan fitur monitoring <i>traffic</i> dan deteksi perangkat secara <i>real-time</i>. - Hasil penelitian menunjukkan bahwa visualisasi jaringan dapat membantu efisiensi <i>monitoring</i> jaringan FTTH.
10	Nurfebrian, A., Pindarwati, A., & Hidayat, R. (2023).	Judul : <i>Perancangan Dan Implementasi Sistem Monitoring Jaringan Berbasis Web PT. Maxindo Mitra Solusi.</i>

		Metode : Studi kasus PT. Maxindo Mitra Solusi, Pengumpulan data, Wawancara Software : - <i>Framework Frontend : React.js</i> - Database : MySQL - OS : Windows / Linux - <i>Web Browser : Google Chrome</i> - Code Editor : Visual Studio Code Hasil Penelitian : - Penelitian ini berhasil merancang dan mengimplementasikan sistem <i>monitoring</i> jaringan berbasis <i>web</i> yang mampu memberikan informasi jaringan secara <i>real-time</i>, efisien dan mudah diakses. - Sistem ini memungkinkan administrator dan customer service untuk : 1. Mendeteksi gangguan secara cepat 2. Mengakses informasi status perangkat 3. Memantau <i>traffic</i> 4. Melihat <i>log</i> aktivitas tanpa membuka sistem terpisah.
--	--	--



Gambar 2. 1 *Mind Map* Penelitian

Penyusunan penelitian ini didasarkan pada analisis alur berpikir yang tertuang dalam *Mind Map* serta kajian terhadap berbagai penelitian terdahulu yang disajikan pada Tabel Penelitian Terkait. Berdasarkan kedua instrumen tersebut, penulis mengidentifikasi beberapa kesenjangan penelitian (*research gap*) yang menjadi dasar dilakukannya penelitian ini, sekaligus merumuskan kebaruan (*novelty*) yang ditawarkan.

1. Gap Penelitian

Berdasarkan hasil telaah terhadap penelitian-penelitian terdahulu, ditemukan beberapa kesenjangan penelitian yang mendasari pentingnya penelitian ini dilakukan, yaitu sebagai berikut :

a. Keterbatasan Integrasi Monitoring Jaringan Berbasis Peta Digital

Penelitian terdahulu mengenai monitoring jaringan umumnya berfokus pada pemantauan kondisi perangkat, penggunaan bandwidth, maupun status pengguna menggunakan aplikasi seperti *Zabbix*, *The Dude*, dan API

MikroTik. Meskipun sistem tersebut mampu menampilkan informasi jaringan secara *real-time*, penyajian data masih didominasi dalam bentuk tabel, grafik, atau daftar perangkat sehingga belum memberikan visualisasi geografis yang terintegrasi dengan lokasi pelanggan. Kondisi tersebut menyebabkan proses identifikasi lokasi gangguan dan distribusi jaringan masih kurang efektif, khususnya pada jaringan FTTH yang memiliki cakupan wilayah cukup luas.

- b. Pemanfaatan *Client Side Rendering* pada Sistem Monitoring Jaringan Masih Terbatas

Penelitian mengenai *Client Side Rendering* (CSR) sebagian besar membahas perbandingan performa metode rendering pada aplikasi web modern. Implementasi CSR sebagai pendekatan utama dalam sistem monitoring jaringan FTTH yang memerlukan pembaruan data secara *real-time* masih relatif sedikit ditemukan. Akibatnya, belum banyak penelitian yang mengevaluasi bagaimana CSR dapat menunjukkan performa aplikasi yang baik dan mendukung responsivitas antarmuka, efisiensi pembaruan data, dan pengalaman pengguna pada aplikasi monitoring jaringan.

- c. Masih terbatas ditemukan Integrasi Monitoring *Real-Time* dalam Satu Platform

Beberapa penelitian telah berhasil mengembangkan sistem monitoring berbasis *web* maupun aplikasi visualisasi jaringan. Namun, penelitian tersebut umumnya hanya berfokus pada satu aspek monitoring, seperti status pengguna PPPoE, *traffic bandwidth*, atau visualisasi topologi jaringan. Belum banyak penelitian yang mengintegrasikan seluruh informasi tersebut ke dalam satu aplikasi monitoring berbasis peta digital yang mampu menyajikan data secara *real-time* melalui pendekatan *Client Side Rendering*.

2. Novelty

Kebaruan yang ditawarkan dalam penelitian ini, yang membedakannya dengan penelitian-penelitian sebelumnya, terletak pada beberapa aspek berikut :

- a. Integrasi Monitoring Jaringan FTTH Berbasis Peta Digital

Penelitian ini mengembangkan aplikasi monitoring jaringan FTTH yang mengintegrasikan informasi pengguna PPPoE, penggunaan *bandwidth*, status perangkat, dan lokasi pelanggan ke dalam satu dashboard berbasis peta digital. Pendekatan ini memberikan kemudahan bagi administrator maupun teknisi dalam memantau kondisi jaringan secara lebih informatif dan mempercepat proses identifikasi gangguan berdasarkan lokasi geografis pelanggan.

- b. Penerapan *Client Side Rendering* untuk Monitoring Jaringan *Real-Time*

Berbeda dengan penelitian sebelumnya yang lebih banyak membahas CSR sebagai metode *rendering* pada aplikasi *web* umum, penelitian ini menerapkan *Client Side Rendering* sebagai arsitektur utama dalam sistem monitoring jaringan FTTH. Pendekatan tersebut memungkinkan proses

pembaruan data dilakukan secara dinamis melalui *WebSocket* tanpa memuat ulang halaman, sehingga menghasilkan antarmuka yang lebih responsif dan interaktif untuk kebutuhan monitoring *real-time*.

c. Evaluasi Sistem Melalui Pengujian Fungsional dan Performa

Penelitian ini tidak hanya berfokus pada pembangunan aplikasi monitoring jaringan, tetapi juga melakukan evaluasi terhadap kualitas sistem melalui dua pendekatan pengujian. Pengujian fungsional dilakukan menggunakan metode *Black Box Testing* untuk memastikan seluruh fitur berjalan sesuai kebutuhan pengguna, sedangkan pengujian performa dilakukan menggunakan *Google Lighthouse* dan *Chrome DevTools* untuk mengukur kinerja aplikasi *web* berbasis *Client Side Rendering*. Pendekatan ini diharapkan dapat memberikan gambaran yang lebih komprehensif mengenai efektivitas sistem yang dikembangkan.

B. Landasan Teori

1. Jaringan *Fiber to the Home* (FTTH)

Jaringan *Fiber to the Home* (FTTH) merupakan teknologi jaringan *broadband* yang menggunakan serat optik sebagai media transmisi utama hingga ke pelanggan akhir. Teknologi ini termasuk ke dalam jaringan FTTx yang menawarkan kapasitas *bandwidth* tinggi, latensi rendah, serta stabilitas koneksi yang lebih baik dibandingkan jaringan berbasis tembaga dan wireless[8].

Teknologi ini menawarkan keunggulan dibandingkan jaringan berbasis tembaga, dimana media tembaga umumnya memiliki keterbatasan kecepatan hingga sekitar 4 Mbps, sedangkan FTTH mampu menyediakan kecepatan hingga 100 Mbps atau lebih[9]. Sejalan dengan itu, perkembangan teknologi telekomunikasi saat ini berlangsung sangat pesat, terutama pada teknologi akses data berkecepatan tinggi yang beralih ke penggunaan media serat optik sebagai solusi utama dalam meningkatkan kualitas layanan komunikasi[10].

2. Arsitektur FTTH

Dalam arsitektur FTTH umumnya menggunakan teknologi GPON (*Gigabit Passive Optical Network*) atau EPON (*Ethernet Passive Optical Network*) yang terdiri dari komponen OLT (*Optical Line Terminal*). Struktur ini memungkinkan distribusi jaringan secara luas dengan efisiensi biaya dan performa yang optimal[11].

3. *Monitoring* Jaringan

Monitoring jaringan adalah proses pengawasan terhadap kondisi jaringan untuk memastikan kinerja, ketersediaan dan keamanan sistem tetap optimal. Sistem *monitoring* modern mampu menampilkan data secara *real-time* sehingga mempermudah proses *troubleshooting*[12].

Monitoring jaringan merupakan kegiatan yang dilakukan untuk mengelola suatu sistem jaringan di lokasi atau area tertentu dengan topologi tertentu. Sistem *monitoring* dipergunakan untuk mempermudah tim teknisi dalam melakukan pemantauan secara rutin kondisi jaringan di lapangan. Sistem *monitoring* biasanya memanfaatkan protokol seperti SNMP, ICMP atau API untuk mengumpulkan data dari perangkat jaringan[13], [14].

4. Metode *Rendering* Pada Aplikasi Web

Metode *rendering* merupakan mekanisme yang digunakan aplikasi web untuk menghasilkan dan menampilkan antarmuka pengguna (user interface) pada browser. Secara umum, terdapat dua pendekatan utama yang banyak digunakan dalam pengembangan aplikasi web modern, yaitu *Server Side Rendering* (SSR) dan *Client Side Rendering* (CSR). Perbedaan utama kedua metode tersebut terletak pada lokasi proses *rendering* dilakukan, yaitu di sisi server atau di sisi browser pengguna. Pemilihan metode *rendering* akan

memengaruhi performa aplikasi, waktu muat halaman, interaktivitas, pemanfaatan sumber daya server, serta pengalaman pengguna secara keseluruhan. Oleh karena itu, pemahaman terhadap karakteristik masing-masing metode diperlukan agar pendekatan yang dipilih sesuai dengan kebutuhan sistem yang dikembangkan[15].

Server Side Rendering (SSR) merupakan metode rendering halaman web yang dilakukan di sisi server sebelum halaman dikirimkan kepada browser pengguna. Ketika pengguna mengakses suatu halaman, server akan memproses permintaan, mengambil data yang diperlukan, kemudian menghasilkan dokumen HTML secara lengkap sebelum dikirimkan ke browser. Dengan pendekatan ini, pengguna dapat langsung melihat isi halaman tanpa harus menunggu proses eksekusi JavaScript secara penuh. SSR banyak diterapkan pada aplikasi yang berorientasi pada penyajian konten karena mampu meningkatkan kecepatan pemuatan awal (*initial page load*) dan mempermudah proses pengindeksan oleh mesin pencari (*Search Engine Optimization/SEO*). Meskipun demikian, proses rendering di sisi server menyebabkan beban komputasi server menjadi lebih besar, terutama ketika jumlah pengguna yang mengakses aplikasi meningkat secara bersamaan[16].

Client Side Rendering (CSR) merupakan metode rendering halaman website yang dilakukan di sisi *client* menggunakan *browser* pengguna. Dalam metode ini, *server* hanya mengirimkan data dasar dan file JavaScript, sedangkan proses rendering tampilan dilakukan oleh *browser*[4].

CSR memungkinkan aplikasi *web* menjadi lebih interaktif karena proses pembaruan data dapat dilakukan tanpa melakukan *reload* halaman secara keseluruhan. Pendekatan ini banyak digunakan pada aplikasi berbasis *real-time* seperti *Dashboard Monitoring* dan aplikasi berbasis peta digital [17]. Untuk memberikan gambaran yang lebih jelas mengenai karakteristik kedua metode rendering tersebut, diperlukan perbandingan terhadap beberapa aspek penting, seperti lokasi proses rendering, performa pemuatan awal, interaktivitas aplikasi, optimasi mesin pencari, beban server, serta kesesuaiannya terhadap jenis aplikasi yang dikembangkan. Perbandingan tersebut disajikan pada tabel berikut.

Tabel 2. 2 Perbandingan Metode SSR dan CSR

NO	Aspek	SSR	CSR
1	Lokasi Rendering	Server	Browser pengguna
2	Initial Loading	Lebih cepat	Sedikit lebih lambat
3	Interaktivitas	Terbatas	Sangat tinggi
4	SEO	Sangat baik	Kurang optimal tanpa optimasi tambahan
5	Beban Server	Lebih tinggi	Lebih rendah
6	Beban Browser	Rendah	Lebih tinggi
7	Pembaruan Data	Memerlukan request ulang	Tanpa reload halaman
8	Cocok Digunakan	Website berita, profil perusahaan, e-commerce	Dashboard, monitoring, aplikasi real-time, SPA

5. Keunggulan CSR dalam *Monitoring real-time*

Pada website *monitoring* jaringan real-time, *Client Side Rendering* (CSR) unggul karena halaman *dirender* langsung di browser menggunakan *JavaScript*, sehingga pembaruan tampilan dapat dilakukan secara dinamis tanpa memuat ulang halaman secara penuh[18]. Pendekatan ini cocok untuk *Dashboard Monitoring* karena aplikasi *live dashboard* membutuhkan respons cepat terhadap perubahan data dan interaksi pengguna yang sering, sementara *React* dikenal efektif untuk membangun antarmuka yang responsif dan interaktif.

Selain itu, CSR dapat membantu mengurangi beban render di server karena visualisasi dan pembaruan antarmuka diproses di sisi klien, sehingga server lebih fokus menyediakan data melalui API[19]. Dengan demikian, CSR mendukung pengalaman pemantauan jaringan yang lebih cepat, lebih interaktif, dan lebih nyaman digunakan pada kondisi data yang terus berubah.

6. Sistem *Real-Time*

Sistem *real-time* adalah sistem yang mampu memproses dan menampilkan data secara langsung atau dengan delay yang sangat kecil. Dalam konteks *monitoring* jaringan, *real-time* berarti perubahan kondisi jaringan dapat langsung terlihat oleh administrator[20].

Monitoring jaringan secara *real-time* merupakan kebutuhan penting dalam menjaga kinerja dan keandalan infrastruktur Teknologi Informasi (TI) pada perusahaan modern. Ketika terjadi gangguan pada sistem jaringan atau *server*, keterlambatan dalam mendeteksi dan menindaklanjuti masalah tersebut dapat berdampak pada penurunan produktivitas, bahkan gangguan layanan bisnis[21].

7. Visualisasi Peta Digital

Visualisasi peta digital merupakan teknik untuk menampilkan data geografis dalam bentuk peta interaktif[22]. Dalam sistem *monitoring* jaringan FTTH, peta digital digunakan untuk menampilkan lokasi pelanggan dan perangkat jaringan.

Peta *online* yang didesain pada sebuah aplikasi disajikan sebagai peta digital yang menampilkan titik koordinat. Peta bisa disajikan dalam berbagai cara yang berbeda, mulai dari peta konvensional yang tercetak hingga peta digital yang tampil di layar komputer[23].

8. Arsitektur Sistem *Monitoring* Jaringan

Arsitektur *monitoring* jaringan FTTH (*Fiber to the Home*) umumnya dibangun menggunakan teknologi GPON (*Gigabit Passive Optical Network*) yang terdiri dari perangkat OLT (*Optical Line Terminal*), ODC (*Optical Distribution Cabinet*), ODP (*Optical Distribution Point*), serta ONT/ONU pada sisi pelanggan. OLT berfungsi sebagai pusat pengendali jaringan dan penghubung utama antara *server* dengan pelanggan melalui media serat optik[24]. Pada sistem *monitoring*, data kondisi jaringan seperti status koneksi, trafik, redaman optik, dan kondisi perangkat pelanggan dikumpulkan dari perangkat OLT kemudian dikirim ke *server monitoring* untuk diproses dan ditampilkan secara *real-time* dalam bentuk dashboard atau visualisasi topologi jaringan[25]. Arsitektur *monitoring* ini memudahkan administrator dalam melakukan pengawasan jaringan secara terpusat, mempercepat deteksi gangguan, serta meningkatkan efisiensi pemeliharaan jaringan FTTH karena informasi jaringan dapat dipantau secara langsung melalui aplikasi berbasis *web*.

Sistem *monitoring* jaringan modern umumnya menggunakan arsitektur terintegrasi yang menggabungkan proses pengumpulan data, pemrosesan, serta visualisasi dalam satu kesatuan sistem. Data yang diperoleh dari perangkat jaringan dikumpulkan melalui berbagai metode, kemudian diproses untuk dianalisis, dan selanjutnya disajikan dalam bentuk visualisasi seperti grafik atau *dashboard* guna memudahkan pemantauan oleh administrator[26].