BAB I

PENDAHULUAN

A. Latar Belakang

Penggunaan internet telah berkembang pesat dalam beberapa dekade terakhir, dan banyak organisasi dan perusahaan bergantung pada konektivitas internet yang Handal dan Efisien untuk operasional sehari-hari mereka. Dalam lingkungan jaringan yang padat, sering kali diperlukan manajemen *Bandwidth* yang tepat guna untuk memastikan pengalokasian sumber daya yang adil dan efisien kepada pengguna, serta memastikan layanan internet yang stabil dan responsif.

Mikrotik sebagai salah satu produsen perangkat jaringan terkemuka, menawarkan solusi yang populer dan terjangkau untuk manajemen jaringan. Perangkat *Mikrotik*, seperti router dan *switch*, menyediakan berbagai fitur yang kuat untuk mengelola lalu lintas jaringan, termasuk kontrol *Bandwidth*, pengaturan kebijakan akses, dan fungsi *Gateway* internet. Dalam pengaturan jaringan skala kecil hingga menengah, tugas konfigurasi dan manajemen *Mikrotik* biasanya dilakukan melalui antarmuka berbasis teks CLI (*Command Line Interface*). Namun, bagi beberapa pengguna, terutama mereka yang tidak berpengalaman dalam konfigurasi perangkat jaringan melalui CLI, mungkin menghadapi hambatan dalam menggunakan perangkat *Mikrotik* Oleh karena itu, munculnya aplikasi manajemen *Bandwidth* dan *Gateway* internet *Mikrotik* berbasis web menjadi sangat berharga. Aplikasi semacam ini menyediakan antar muka grafis yang ramah pengguna, yang

memungkinkan administrator jaringan untuk mengkonfigurasi dan mengelola perangkat *Mikrotik* tanpa perlu mengetikkan perintah-perintah kompleks secara manual.

Aplikasi manajemen *Mikrotik* berbasis web menyediakan berbagai fitur penting, termasuk: Pengaturan *Bandwidth*: Administrator dapat dengan mudah mengalokasikan *Bandwidth* yang sesuai untuk setiap pengguna atau kelompok pengguna, memberikan prioritas tertentu untuk aplikasi atau layanan tertentu, dan mengatur pembatasan kecepatan untuk mencegah satu pengguna yang menguasai seluruh *Bandwidth*. Kontrol Akses Internet: Aplikasi ini memungkinkan pengaturan kebijakan akses internet yang dapat disesuaikan, seperti pembatasan waktu akses, pemblokiran situs web tertentu, atau mengizinkan akses hanya pada jam kerja tertentu. *Monitoring* dan Statistik: Administrator dapat memantau lalu lintas jaringan secara *real-time* dan melihat statistik yang relevan, seperti penggunaan *Bandwidth*, aktivitas koneksi, dan lalu lintas tertinggi. Pengelolaan *Hotspot*: Jika jaringan menyediakan layanan *hotspot*, aplikasi ini dapat mempermudah pengaturan dan manajemen pengguna *hotspot*, termasuk *autentikasi* dan perangkat lunak *voucher*.

Dengan adanya aplikasi manajemen *Bandwidth* dan *Gateway* internet *Mikrotik* berbasis web, para administrator jaringan dapat mengelola perangkat *Mikrotik* dengan cepat. Antarmuka yang ramah pengguna memungkinkan pengguna yang tidak berpengalaman dalam pengaturan jaringan untuk menggunakan perangkat *Mikrotik* secara efektif, sehingga meningkatkan performa jaringan, memberikan pengalaman internet yang lebih baik bagi pengguna jaringan.

Maka dari itu, penulis tertarik melakukan penelitian dengan judul **"APLIKASI MANAJEMEN *BANDWIDTH* DAN INTERNET *GATEWAY MIKROTIK* BERBASIS WEB"**.

B. Rumusan Masalah

Berdasarkan uraian latar belakang diatas, maka dapat dirumuskan suatu masalah sebagai berikut :

1. Bagaimana membuat aplikasi manajemen *Mikrotik* berbasis web?
2. Bagaimana melakukan pengaturan *Bandwidth* menggunakan aplikasi yang sudah dibuat?

C. Tujuan Penelitian

Adapun tujuan penelitian sebagai berikut :

1. Untuk membuat aplikasi manajemen *Mikrotik* berbasis web.
2. Untuk melakukan implementasi *Bandwidth* dan *Hotspot* menggunakan aplikasi yang sudah dibuat di UPT SMKN 2 Pinrang.

D. Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Rancangan aplikasi hanya dibuat berbasis web.
2. Aplikasi ini tidak menggunakan *database* dari *Mysql.*
3. *Trafik* jaringan hanya dapat dimunculkan ketika ada koneksi internet.
4. Hanya menjelaskan fitur-fitur aplikasi yang dibuat untuk mengelola *Mikrotik.*

E. Manfaat Penelitian

Manfaat yang ingin dicapai untuk penelitian ini yaitu memberikan solusi untuk menangani masalah pada jaringan yang dapat dilakukan dengan cara

mengelola jaringan tersebut merata sehingga tidak ada lagi komplain jika jaringan hanya lari ke satu titik saja.

F. Sistematika Penulisan

Secara garis besar sistematika penulisan yang dapat disajikan sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi hal – hal yang menjadi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini pembahasan tentang teori yang diperlukan pada saat melakukan penelitian tugas akhir ini, seperti teori jaringan, keamanan jaringan, definisi *Mikrotik,* dan untuk teori khusus seperti, aplikasi dan bahasa pemrograman yang dapat mengembangkan aplikasi ini.

BAB III METODE PENELITIAN

Bab ini membahas metode penelitian yang digunakan dalam merancang sistem.

BAB IV HASIL PENGUJIAN

Bab ini membahas tentang sistem yang berjalan, sistem yang diusulkan, perancangan sistem, pengujian.

BAB V PENUTUP

Pembahasan pada bab ini meliputi kesimpulan tentang hasil dan pengujian yang dilakukan, serta saran bagi penulis untuk memperbaiki kesalahan pada perencanaan yang dibuat.

BAB II

TINJAUAN PUSTAKA

A. Kajian Hasil Penelitian Terdahulu

Adapun penelitian terkait, sebagian merupakan hasil dari penelitian-penelitian sebelumnya untuk dijadikan acuan dalam pelaksanaan tugas akhir. Dari penelitian ini, peneliti menemukan beberapa penelitian pendukung untuk mengangkat permasalahan yang berkaitan dengan penelitian penulis. Oleh karena itu, penulis mempertimbangkan dan mempelajari beberapa hasil penelitian untuk perbandingan, termasuk tugas akhir/disertasi atau jurnal yang terdapat di internet.

1. Penelitian oleh (Sujalwo et al., 2020), yang berjudul "Manajemen Jaringan Komputer dengan Menggunakan *Mikrotik* Router". Penelitian ini mengenai manajemen jaringan komputer dengan menggunakan salah satu *router operating system* (OS) yang ada yaitu *Mikrotik Router*. Dengan tujuan untuk mengatur dan mengetahui alokasi *Bandwidth* untuk akses jaringan *local internet* pada *client* dalam sebuah jaringan komputer dan untuk melakukan pengelolaan pengguna pada *wireless access point.*
2. (Tasanah Assakur et al., 2020) dengan judul "Implementasi API *Mikrotik* untuk *Manajement Router* Berbasis Android (Studi Kasus: PT Sigma Adi Perkasa)" berdasarkan penelitian Aplikasi ini memiliki fungsi *update* data, sehingga administrator jaringan dapat melakukan penambahan konfigurasi, edit, simpan dan hapus secara berkala. Aplikasi yang dibuat telah berhasil dilakukan pengujian *Black Box* dengan fitur-fitur yang telah dibuat dapat

digunakan dengan baik. Dengan aplikasi *mobile*, konfigurasi dan *troubleshooting router Mikrotik* telah memberi pengaruh yang positif dan mempermudah pekerjaan seorang administrator jaringan.

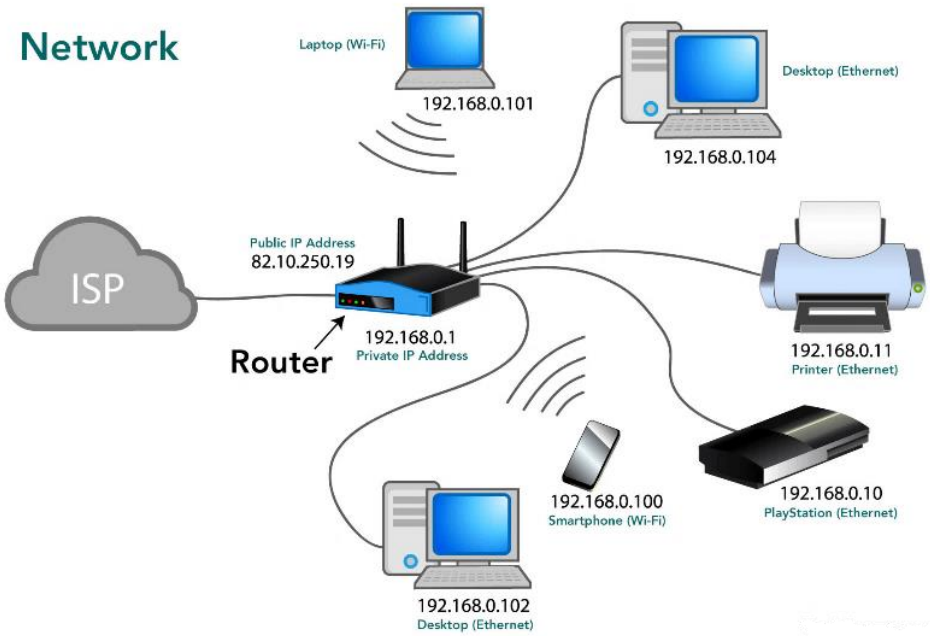
3. Penelitian ini dilakukan oleh (Anwar et al., 2017) Dengan judul "Rancang Bangun Sistem Manajemen *User Hotspot* Menggunakan *Mikrotik* Php Api Berbasis Web Di Pondok Pesantren Al-Luqmaniyyah" dari penelitian ini membangun Sistem Manajemen *User Hotspot* yang ada di Pondok Pesantren Al-Luqmaniyyah. Pondok pesantren tersebut menggunakan *Mikrotik* routerboard sebagai *routing* jaringan lokal dan sekaligus sebagai *Gateway* internet *hotspot*. Berdasarkan masalah tersebut, penelitian ini bertujuan untuk membuat aplikasi alternatif yang bisa mempermudah pengguna dalam memanajemen *User hotspot* di Pondok Pesantren Al-Luqmaniyyah.

Dari tiga jurnal di atas perbedaan dari judul peneliti yaitu pada jurnal pertama secara khusus tidak membahas tentang pengembangan aplikasi sedangkan judul peneliti merancang sebuah Aplikasi yang digunakan untuk melakukan manajemen *mikrotik.* pada jurnal kedua menekankan pengembangan aplikasi *mobile* untuk manajemen *mikrotik* sedangkan judul peneliti mengembangkan aplikasi manajemen *mikrotik* berbasis web dengan memanfaatkan *Application Programming Interfaces* (API) web *mikrotik* untuk merancang aplikasi.

B. Kajian Teori

1. Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer yang saling terhubung dan terkoneksi satu sama lain sehingga dapat bertukar dan saling bertukar data satu sama lain. Komputer yang bersifat berdiri sendiri mempunyai banyak keterbatasan. Adanya jaringan komputer akan membuat komputer dapat melakukan banyak hal dan dapat membantu efisiensi dan efektivitas dalam dunia kerja. Contohnya dua komputer yang masing-masing memiliki kartu jaringan, kemudian dihubungkan melalui kabel atau nirkabel sebagai sarana transmisi data dan dengan perangkat lunak sistem operasi jaringan yang akan membentuk jaringan komputer sederhana. Jika ingin membangun jaringan komputer dengan jangkauan yang lebih luas maka diperlukan lebih banyak perangkat seperti *hub, bridge, switch, router*, *Gateway* sebagai perangkat penghubung.



Gambar 2. 1 Jaringan Komputer (https://freevpnplanet.com/blog/11-types-of-networks-explained-vpn-lan-more/)

Jaringan komputer adalah studi yang membahas berbagai teori dan konsep yang mendefinisikan fungsi, organisasi, dan komunikasi antara perangkat dan perangkat dalam jaringan. Berikut beberapa teori yang berkaitan dengan jaringan komputer sebagai berikut:

1. Model Referensi OSI: *Open Systems Linked Reference Model* (OSI) adalah kerangka kerja yang diperkenalkan oleh *International Organization for Standardization* (ISO). Model ini membagi proses komunikasi jaringan menjadi tujuh lapisan yang berbeda, dari lapisan fisik hingga lapisan aplikasi. Setiap kelas memiliki fungsi dan tanggung jawabnya masing-masing, dan model ini memberikan panduan umum untuk memahami hubungan antar kelas.
2. Model referensi TCP/IP: Model referensi TCP/IP adalah model yang paling umum digunakan dalam implementasi jaringan saat ini. Model ini terdiri dari empat *layer* yaitu *network layer*, *internet layer, transport layer* dan *application layer*. Model ini memberikan panduan tentang bagaimana protokol jaringan seperti *Transport Control Protocol* (TCP) dan *Internet Protocol* (IP) bekerja sama untuk mengirim data melalui jaringan.
3. Protokol jaringan: Protokol jaringan adalah aturan dan prosedur yang dirancang untuk menentukan bagaimana perangkat di jaringan dapat berkomunikasi satu sama lain. Protokol seperti TCP, IP, UDP (*User Datagram Protocol*) dan ICMP (*Internet Control Message Protocol*) adalah contoh protokol jaringan yang umum digunakan. Protokol ini melibatkan

pemikiran tentang cara kerja, format paket data, algoritma perutean, pengecekan kesalahan, dan manajemen aliran data.

4. *Topologi* jaringan : *Topologi* jaringan mengacu pada cara fisik atau logis bahwa perangkat dalam jaringan terhubung satu sama lain. Dalam jaringan, ada beberapa *topologi* umum, termasuk *topologi bus, ring, star, mesh*, dan *tree*. Memahami *topologi* jaringan melibatkan pemahaman kekuatan, kelemahan, dan karakteristik masing-masing *topologi*, dan bagaimana perangkat jaringan berkomunikasi dalam konteks *topologi* tersebut.
5. Algoritma perutean: Algoritma *routing* adalah metode yang digunakan dalam jaringan komputer untuk menentukan jalur terbaik untuk paket data yang akan dikirim dari sumber ke tujuan. Beberapa algoritma perutean yang umum digunakan termasuk algoritma vektor jarak, algoritma status tautan, dan algoritma perutean terarah. Memahami algoritma *routing* melibatkan pemahaman kriteria pemilihan jalur, tabel *routing*, metrik kinerja jaringan, dan strategi pengiriman paket.
6. Keamanan dunia maya: *Cybersecurity* berurusan dengan teori dan praktik yang berfokus pada melindungi jaringan dari ancaman dan serangan. Ini termasuk aspek seperti *otentikasi, enkripsi*, *firewall*, IDS (sistem deteksi intrusi), IPS (sistem pencegahan intrusi), dan kebijakan keamanan. Memahami teori keamanan siber melibatkan pemahaman tentang kerentanan, serangan, teknik deteksi, dan penanggulangan yang tepat.
7. Manajemen jaringan: Manajemen jaringan berurusan dengan teori dan praktik yang berkaitan dengan administrasi jaringan, pemantauan, dan

pemeliharaan. Ini mencakup topik-topik seperti konfigurasi perangkat, manajemen kinerja, perencanaan pemeliharaan, dan pemecahan masalah. Memahami manajemen jaringan melibatkan pemahaman alat dan protokol yang digunakan dalam manajemen jaringan, serta prinsip-prinsip manajemen yang efektif.

Di atas memberikan gambaran tentang berbagai teori yang berkaitan dengan jaringan komputer. Namun, perlu diingat bahwa jaringan komputer adalah bidang yang selalu berubah dan ada banyak konsep dan teknologi baru yang dapat memperluas pemahaman penelitian teori jaringan komputer.

Dalam jaringan komputer terdapat beberapa jenis dan dapat dibedakan berdasarkan cakupan, arsitektur dan jenis teknologi yang digunakan yaitu sebagai berikut :

1. Berdasarkan Area Cakupan:
    a. LAN (*Local Area Network*): Jaringan lokal yang mencakup area terbatas seperti kantor, gedung, atau kampus. Digunakan untuk berbagi sumber daya dan menjadi fasiltas komunikasi antara perangkat yang berada dalam jangkauannya.
    b. WAN (*Wide Area Network*): Jaringan yang mencakup area yang luas, bahkan bisa mencakup wilayah yang sangat luas, seperti kota, negara, atau benua. Biasanya digunakan untuk menghubungkan LAN yang berjauhan atau lokasi yang terpisah.
    c. MAN (*Metropolitan Area Network*): Jaringan yang memiliki cakupan lebih besar daripada LAN tetapi lebih kecil daripada WAN, mencakup wilayah perkotaan atau metropolitan.

d. WLAN (*Wireless Local Area Network*): Jaringan lokal tanpa kabel yang menggunakan teknologi seperti *Wi-Fi* (*Wireless Fidelity*) untuk menghubungkan perangkat secara nirkabel di dalam suatu area tertentu.

2. Berdasarkan Arsitektur :

   a. *Client-Server Network*: Jaringan yang terdiri dari klien (*client*) dan *server*. *Server* menyediakan layanan, sumber daya, dan informasi kepada klien yang meminta.

   b. *Peer-to-Peer Network*: Jaringan dimana setiap perangkat memiliki peran yang setara dan saling berbagi sumber daya secara langsung tanpa server sentral. Biasanya digunakan dalam jaringan kecil dengan sedikit perangkat.

3. Berdasarkan Jenis Teknologi :

   a. *Ethernet Network*: Jaringan yang menggunakan protokol *Ethernet* sebagai metode pengiriman data melalui kabel.

   b. *Token Ring Network*: Jaringan dimana data dikirimkan dalam bentuk token dari satu perangkat ke perangkat lain secara berurutan.

   c. *Wireless Network*: Jaringan yang menggunakan teknologi nirkabel untuk menghubungkan perangkat, seperti *Wi-Fi* dan *Bluetooth*.

   d. *Internet*: Jaringan global terbesar yang menghubungkan jutaan jaringan dan perangkat di seluruh dunia, termasuk WAN dan LAN.

   e. *Intranet*: Jaringan internal suatu organisasi atau perusahaan yang menggunakan teknologi internet untuk berbagi informasi dan sumber daya.

f. *Extranet*: Jaringan yang memungkinkan organisasi untuk berbagi informasi dan sumber daya dengan mitra, pelanggan, atau pihak eksternal lainnya melalui *internet*.

2. *Mikrotik*

Mikrotik adalah perangkat jaringan komputer berupa *hardware* dan *software* yang dapat berperan sebagai *router*, sebagai *filter*, *switch* dan alat bantu lainnya. *Hardware Mikrotik* bisa berupa PC *router* (dipasang di PC) atau berupa *router card* (*built* langsung oleh perusahaan *Mikrotik*) (Amarudin, 2018).

Gambar 2. 2 *Mikrotik Router*

Mikrotik adalah sebuah perusahaan yang mengembangkan perangkat keras (*hardware*) dan perangkat lunak (*software*) untuk keperluan jaringan komputer. Perusahaan ini terkenal karena sistem operasi *router* mereka yang bernama *Mikrotik RouterOS*. Berikut ini adalah kajian teori yang terkait dengan *Mikrotik* :

1. RouterOS: *Mikrotik* RouterOS adalah sistem operasi berbasis Linux yang dikembangkan oleh *Mikrotik.* RouterOS adalah inti dari perangkat *Mikrotik* dan menyediakan berbagai fitur dan fungsionalitas yang penting dalam jaringan komputer, seperti *routing, switching, firewall*, VPN (*Virtual Private Network*), QoS (*Quality of Service*), dan banyak lagi. Pemahaman

teori tentang *RouterOS* melibatkan pemahaman tentang konfigurasi dasar, manajemen perangkat, antarmuka pengguna, dan fitur-fitur yang tersedia.

2. Konfigurasi Dasar: Untuk menggunakan perangkat *Mikrotik* dengan efektif, pemahaman tentang konfigurasi dasar sangat penting. Ini meliputi pengaturan alamat IP, *routing* statis dan dinamis, pengaturan antarmuka jaringan, pengaturan protokol jaringan seperti OSPF (*Open Shortest Path First*) dan BGP (*Border Gateway Protocol*), dan pengaturan dasar *firewall*. Pemahaman teori tentang konfigurasi dasar *Mikrotik* melibatkan pemahaman tentang sintaksis perintah dan aturan-aturan dasar konfigurasi.
3. Manajemen Jaringan: *Mikrotik* RouterOS menyediakan berbagai alat dan fitur untuk manajemen jaringan yang efektif. Ini termasuk manajemen pengguna dan hak akses, manajemen *Bandwidth*, manajemen log, manajemen perangkat keras dan perangkat lunak, dan pemantauan kinerja jaringan. Pemahaman teori tentang manajemen jaringan *Mikrotik* melibatkan pemahaman tentang konsep-konsep seperti *User* manajement, queueing, SNMP (*Simple Network Manajement Protocol*), dan monitoring tools yang disediakan oleh *Mikrotik.*
4. Keamanan: Keamanan jaringan adalah aspek yang sangat penting dalam penggunaan perangkat *Mikrotik. RouterOS* menyediakan fitur-fitur keamanan seperti *firewall*, VPN, *tunneling, autentikasi*, enkripsi, dan kontrol akses. Pemahaman teori tentang keamanan *Mikrotik* melibatkan pemahaman tentang pengaturan *firewall*, konfigurasi VPN (seperti PPTP, L2TP/IPsec*, Open*VPN), enkripsi, serta praktik keamanan yang umum

seperti penggunaan *password* yang kuat dan pemantauan keamanan jaringan.

5. Penyelesaian Masalah (*Troubleshooting*): Kemampuan untuk menyelesaikan masalah jaringan merupakan keterampilan yang penting dalam penggunaan *Mikrotik.* Pemahaman teori tentang *troubleshooting Mikrotik* melibatkan pemahaman tentang log sistem, analisis pesan *error*, pemecahan masalah koneksi jaringan, pengecekan konfigurasi, dan pemahaman tentang alat bantu yang disediakan oleh *Mikrotik* untuk membantu dalam proses *troubleshooting*.

3. ISP (*Internet Service Provider*)

Menurut Wikipedia ISP (*Internet Service Provider*) adalah perusahaan atau badan usaha yang menjual koneksi internet atau sejenisnya kepada pelanggan. ISP awalnya sangat identik dengan jaringan telepon, karena dulu ISP menjual koneksi atau *access* internet melalui jaringan telepon. Seperti salah satunya adalah *telkomnet instant* dari Telkom. Sekarang, dengan perkembangan teknologi ISP itu berkembang tidak hanya dengan menggunakan jaringan telepon tapi juga menggunakan teknologi seperti *fiber optic* dan *wireless* (Santoso, 2012).

Kajian teori tentang ISP (*Internet Service Provider*) melibatkan pemahaman tentang berbagai aspek yang terkait dengan penyedia layanan internet. Berikut ini adalah beberapa poin penting yang perlu dipahami:

1. Infrastruktur Jaringan: ISP memiliki infrastruktur jaringan yang kompleks untuk menghubungkan pengguna internet ke internet global. Pemahaman tentang infrastruktur jaringan melibatkan pengetahuan tentang perangkat

keras (*hardware*) seperti *router, switch*, dan server, serta pemahaman tentang *topologi* jaringan seperti jaringan area luas (*Wide Area Network*) dan jaringan area lokal (*Local Area Network*).

2. Koneksi Internet: ISP menyediakan koneksi internet kepada pelanggan mereka. Pemahaman tentang koneksi internet meliputi teknologi akses seperti DSL (Digital *Subscriber Line*), kabel, serat optik, satelit, dan nirkabel. Pemahaman teori juga mencakup pengetahuan tentang kecepatan koneksi, latensi (*ping*), dan kapasitas *Bandwidth* yang tersedia.
3. Layanan Internet: ISP menyediakan berbagai layanan internet kepada pelanggan mereka. Ini termasuk layanan seperti akses internet, telepon VoIP (*Voice over Internet Protocol*), televisi kabel atau IPTV (*Internet Protocol Television*), web *hosting*, penyimpanan *cloud*, dan banyak lagi. Pemahaman tentang layanan internet melibatkan pengetahuan tentang cara kerja dan konfigurasi layanan-layanan ini.
4. *Routing* dan *Peering*: ISP bertanggung jawab untuk mengirimkan data dari pelanggan mereka ke tujuan yang diinginkan di internet. Pemahaman tentang *routing* melibatkan pengetahuan tentang protokol *routing*, algoritma *routing*, dan pengelolaan tabel *routing* untuk mengarahkan lalu lintas data dengan efisien. Selain itu, pemahaman tentang *peering* (interkoneksi) antara ISP yang berbeda juga penting untuk memahami bagaimana data diarahkan di antara ISP.
5. Manajemen *Trafik*: ISP perlu mengelola dan memantau lalu lintas jaringan untuk memastikan kinerja yang baik dan pengalaman pengguna yang

memadai. Pemahaman tentang manajemen *trafik* melibatkan pengetahuan tentang teknik seperti *QoS (Quality of Siervice), traffic shaping, Bandwidth manajement*, dan *caching* untuk mengoptimalkan penggunaan *Bandwidth* dan memastikan kualitas layanan yang konsisten.

6. Keamanan Jaringan: ISP bertanggung jawab untuk menjaga keamanan jaringan mereka dan melindungi pelanggan mereka dari ancaman keamanan *online*. Pemahaman tentang keamanan jaringan melibatkan pengetahuan tentang *firewall*, IDS/IPS *(Intrusion Detection System/Intrusion Prevention System),* enkripsi, kebijakan keamanan, dan praktik keamanan lainnya untuk melindungi jaringan dan data pengguna.
7. Kebijakan dan Regulasi: ISP beroperasi dalam kerangka kebijakan dan regulasi yang ditetapkan oleh otoritas pemerintah atau badan pengatur. Pemahaman tentang kebijakan dan regulasi melibatkan pengetahuan tentang peraturan yang mengatur industri telekomunikasi, persyaratan izin, hak dan kewajiban pengguna, privasi data, dan aturan-aturan lain yang berlaku dalam konteks penyediaan layanan internet.

Pemahaman teori tentang ISP mencakup aspek-aspek yang telah disebutkan di atas, tetapi juga dapat berkaitan dengan perkembangan dan tren terbaru dalam industri, teknologi baru yang diperkenalkan, dan tantangan yang dihadapi oleh ISP dalam memenuhi kebutuhan pengguna internet.

4. *Bandwidth*

Bandwidth merupakan nilai hitung atau perhitungan konsumsi *transfer* data telekomunikasi yang dihitung dalam satuan bit per detik (bps) yang terjadi antara

komputer *server* dan komputer *client* dalam waktu tertentu dalam sebuah jaringan komputer. *Bandwidth* dialokasikan ke komputer yang ada di dalam jaringan dan akan mempengaruhi kecepatan transfer data pada jaringan tersebut, sehingga semakin besar *Bandwidth* jaringan maka semakin cepat pula kecepatan *transfer* data yang dapat dilakukan oleh *client* maupun *server* (Muhammad M & Hasan I, 2016).

Jika dilihat dari sudut pandang proses pengiriman data, *Bandwidth* dapat dibagi menjadi 2 kategori, yaitu *download* dan *upload*.

a. Unduh (*Download*)

   Download adalah proses menerima data (umumnya berbentuk berkas) dari sebuah sistem seperti web *server*, FTP *server*, *mail server* atau sistem serupa lainnya. *Download* juga merupakan kegiatan dimana seseorang dapat memperoleh *file-file* tertentu yang ada di internet. *File* yang dapat diunduh bermacam-macam, ada yang berupa *file* video, mp3, *document*, dan lain-lain.

b. Unggah (*Upload*)

   Upload adalah proses mengirim data (umumnya berbentuk berkas) dari komputer pribadi ke suatu sistem seperti web *server*, FTP *server* atau sistem serupa lainnya yang kemudian akan dipublikasikan di internet baik secara pribadi atau umum (dapat dinikmati oleh semua pengguna internet).

Bandwidth melibatkan pemahaman tentang konsep dan pengukuran *Bandwidth* dalam konteks jaringan komunikasi. Berikut ini adalah beberapa poin penting yang perlu dipahami:

1. Definisi *Bandwidth*: *Bandwidth* mengacu pada kapasitas maksimum dari saluran komunikasi yang dapat digunakan untuk mentransmisikan data dalam suatu periode waktu tertentu. *Bandwidth* diukur dalam bit per detik (bps) dan menunjukkan berapa banyak data yang dapat dikirimkan atau diterima dalam jaringan.
2. *Bandwidth Upstream* dan *Downstream*: *Bandwidth* dapat dibagi menjadi *Bandwidth upstream* dan *Bandwidth downstream*. *Bandwidth upstream* mengacu pada kecepatan data yang dikirimkan dari pengguna atau perangkat ke jaringan, sedangkan *Bandwidth downstream* mengacu pada kecepatan data yang diterima oleh pengguna atau perangkat dari jaringan.
3. *Bandwidth* Efektif: *Bandwidth* efektif adalah *Bandwidth* yang tersedia untuk pengguna atau perangkat untuk digunakan dalam komunikasi. Faktor-faktor seperti *overhead* protokol, penggunaan jaringan oleh pengguna lain, dan kondisi jaringan dapat mempengaruhi *Bandwidth* efektif yang dapat diakses oleh pengguna.
4. Kapasitas dan *Throughput*: Kapasitas *Bandwidth* merujuk pada jumlah total data yang dapat ditransmisikan dalam jaringan, sedangkan *throughput* merujuk pada jumlah data yang berhasil ditransmisikan dalam suatu periode waktu tertentu. *Throughput* dapat dipengaruhi oleh faktor-faktor seperti latensi, kehilangan paket, dan kesalahan transmisi.
5. *Bandwidth* Terukur dan Dijamin: ISP seringkali menyediakan layanan dengan *Bandwidth* terukur dan dijamin. *Bandwidth* terukur mengacu pada *Bandwidth* yang dapat diukur dan dipantau oleh pengguna, sementara

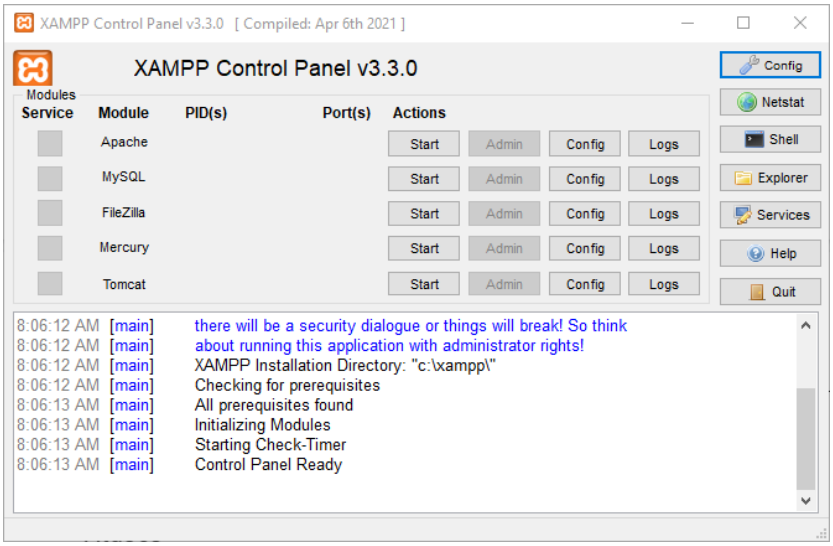
Bandwidth dijamin adalah jumlah minimum *Bandwidth* yang dijamin oleh ISP kepada pelanggan.

6. Pembagian *Bandwidth*: Dalam lingkungan jaringan yang dibagikan, *Bandwidth* harus dibagi secara adil antara pengguna atau perangkat yang menggunakan jaringan. Teknik seperti pembatasan *Bandwidth* (*Bandwidth* throttling) dan manajemen antrian (*queue manajement*) digunakan untuk mengatur alokasi *Bandwidth* yang sesuai.
7. Peningkatan *Bandwidth*: Dalam beberapa situasi, peningkatan *Bandwidth* dapat diperlukan untuk memenuhi kebutuhan komunikasi yang lebih tinggi. Ini dapat dicapai dengan meningkatkan kapasitas fisik jaringan, mengadopsi teknologi yang lebih cepat, atau menggunakan teknik kompresi dan *caching* untuk mengoptimalkan penggunaan *Bandwidth*.

Pemahaman teori tentang *Bandwidth* melibatkan juga pemahaman tentang teknologi akses internet seperti DSL, kabel, serat optik, dan nirkabel, serta konsep seperti *latency* (jeda waktu), *jitter* (ketidakstabilan waktu transmisi), dan *Quality of Service* (QoS) untuk mengatur prioritas dan kualitas layanan dalam jaringan. Perkembangan teknologi dan inovasi terus-menerus mempengaruhi konsep dan pemahaman tentang *Bandwidth*.

5. XAMPP

XAMPP adalah *software open source* berbasis web yang berisi berbagai program. Aplikasi ini mendukung berbagai sistem operasi Linux, Windows, MacOS, dan Solaris. Fungsi xampp adalah sebagai server lokal/*localhost*, didalamnya sudah mencakup program *Apache*, Mysql dan PHP.



Gambar 2. 3 *XAMPP Control Panel*, **Sumber**(https://www.dewaweb.com/)

XAMPP adalah paket perangkat lunak yang dikembangkan oleh *Apache Friends* yang digunakan untuk menginstal dan menjalankan server web lokal pada komputer pribadi. XAMPP merupakan singkatan dari X (sistem operasi apa pun), *Apache*, MySQL, PHP, dan Perl. Berikut ini adalah kajian teori yang terkait dengan XAMPP:

1. *Apache*: XAMPP menggunakan server web *Apache*, yang merupakan salah satu server web yang paling populer dan sering digunakan. *Apache* menyediakan lingkungan untuk meng-host dan melayani halaman web serta mendukung berbagai fitur seperti modul ekstensi, konfigurasi *virtual host*, manajemen akses, dan pengaturan keamanan.
2. *MySQL*: XAMPP juga mencakup *MySQL*, sistem manajemen basis data relasional (*RDBMS*) yang populer. *MySQL* digunakan untuk menyimpan dan mengelola data dalam aplikasi web. Pemahaman tentang *MySQL* melibatkan konsep seperti pengaturan basis data, tabel, *query SQL*, *indeks*, koneksi *database*, dan manajemen pengguna.

3. PHP: XAMPP mengintegrasikan bahasa pemrograman PHP (*Hypertext Preprocessor*), yang sering digunakan dalam pengembangan aplikasi web. PHP memungkinkan penulisan kode yang dinamis di dalam halaman web dan berinteraksi dengan basis data. Pemahaman teori tentang PHP melibatkan sintaksis, variabel, kontrol aliran, fungsi, koneksi ke basis data, dan pengelolaan sesi.
4. *Perl*: XAMPP juga mencakup bahasa pemrograman *Perl*, yang digunakan untuk pemrosesan teks dan tugas-tugas pemrograman umum. *Perl* memiliki banyak modul dan fungsionalitas yang kuat untuk manipulasi teks, pemrosesan *file*, dan tugas-tugas pemrograman lainnya. Pemahaman teori tentang *Perl* melibatkan *sintaksis, variabel, ekspresi reguler*, dan fungsi bawaan.
5. Konfigurasi XAMPP: XAMPP menyediakan antarmuka pengguna yang sederhana untuk mengonfigurasi dan mengelola server web lokal. Pemahaman teori tentang konfigurasi XAMPP melibatkan pemahaman tentang pengaturan *Apache*, konfigurasi *MySQL*, pengaturan ekstensi PHP, pengelolaan direktori, dan manajemen *log*.
6. Pengembangan *Web* Lokal: XAMPP digunakan secara luas untuk pengembangan web lokal, dimana pengembang dapat menguji dan mengembangkan aplikasi *web* sebelum mereka diunggah ke server produksi. Pemahaman tentang pengembangan web lokal dengan XAMPP melibatkan pemahaman tentang pengaturan *virtual host*, *debugging*,

penggunaan alat pengembangan seperti *Xdebug*, dan manajemen sumber daya web.

Pemahaman teori tentang *XAMPP* mencakup aspek-aspek yang telah disebutkan di atas, tetapi juga dapat berkaitan dengan konsep-konsep dan teknologi terkait seperti manajemen *server*, konfigurasi *firewall*, pengujian keamanan, dan integrasi dengan alat pengembangan lainnya. Studi lebih lanjut, dokumentasi resmi *XAMPP*.

6. PHP

PHP (*Hypertext Preprocessor)* PHP adalah singkatan dari "PHP: *Hypertext Preprocessor*", yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs *website* dan bisa digunakan bersamaan dengan HTML. PHP diciptakan oleh *Rasmus Lerdorf* pertama kali tahun 1994.Pada awalnya PHP adalah singkatan dari "*Personal Home Page Tools*".Selanjutnya diganti menjadi FI ("*Forms Interpreter*"). Sejak versi 3.0, nama bahasa ini diubah menjadi "PHP: *Hypertext Preprocessor*" dengan singkatannya "PHP", (Suarman et al., 2017).

PHP (*Hypertext Preprocessor*) melibatkan pemahaman tentang bahasa pemrograman tersebut dan bagaimana digunakan dalam pengembangan aplikasi web. Berikut ini adalah beberapa poin penting yang perlu dipahami:

1. Pengenalan PHP: PHP adalah bahasa pemrograman sisi server yang dirancang khusus untuk pengembangan aplikasi web. PHP digunakan untuk menghasilkan konten dinamis di dalam halaman web dan berinteraksi dengan basis data. Pemahaman teori tentang PHP melibatkan pemahaman

tentang sintaksis dasar, tipe data, variabel, operator, ekspresi, dan struktur kendali.

2. Pemrograman Web dengan PHP: PHP memungkinkan pengembang web untuk membangun aplikasi web yang interaktif dan dinamis. Pemahaman teori tentang pemrograman web dengan PHP melibatkan pemahaman tentang konsep-konsep seperti pemrosesan formulir, penggunaan variabel *superglobal*, penanganan *file*, manipulasi *string*, dan penggunaan fungsi bawaan PHP.
3. Integrasi dengan HTML dan CSS: PHP sering digunakan bersama HTML dan CSS untuk menghasilkan halaman web yang dinamis. Pemahaman teori tentang integrasi PHP dengan HTML dan CSS melibatkan pemahaman tentang cara menggunakan kode PHP didalam *tag* HTML, menampilkan data dinamis dalam elemen HTML, dan menggunakan CSS untuk memperindah tampilan halaman web.
4. Koneksi ke Basis Data: Salah satu kekuatan PHP adalah kemampuannya untuk berinteraksi dengan basis data. Pemahaman teori tentang koneksi ke basis data dengan PHP melibatkan pemahaman tentang PDO (PHP Data *Objects*), *mysqli extension*, dan fungsi-fungsi *database* PHP lainnya. Ini termasuk pembuatan koneksi, pengiriman *query* SQL, penanganan hasil query, dan pengamanan terhadap serangan *SQL injection*.
5. *Framework* PHP: Ada banyak *framework PHP* yang tersedia, seperti Laravel, CodeIgniter, dan Symfony. *Framework* PHP menyediakan struktur dan alat bantu untuk mempercepat pengembangan aplikasi web.

Pemahaman teori tentang *framework* PHP melibatkan pemahaman tentang konsep MVC (*Model-View-Controller*), *routing*, penggunaan ORM (*Object-Relational Mapping*), dan fitur-fitur khusus yang ditawarkan oleh masing-masing *framework*.

6. Keamanan: Keamanan adalah aspek penting dalam pengembangan aplikasi web dengan PHP. Pemahaman teori tentang keamanan PHP melibatkan pemahaman tentang serangan umum seperti SQL *injection*, XSS (*Cross-Site Scripting*), CSRF (*Cross-Site Request Forgery*), dan teknik perlindungan seperti validasi *input*, penggunaan *parameterized queries*, dan penghapusan data yang tidak aman.
7. Pengujian dan *Debugging*: Pemahaman teori tentang pengujian dan *debugging* PHP melibatkan pemahaman tentang teknik pengujian seperti pengujian unit dan integrasi, penggunaan alat pengujian seperti *PHPUnit*, serta teknik *debugging* menggunakan alat seperti *Xdebug* atau *logging*.

Pemahaman teori tentang PHP melibatkan juga pemahaman tentang pengembangan aplikasi web secara umum, konsep desain web, dan teknologi terkait seperti *JavaScript*, AJAX, dan XML.

7. API PHP *Class*

API PHP *Class* merupakan sebuah *Class scripting* PHP yang dirancang oleh pihak *Developer Mikrotik* untuk menghubungkan antara PHP dengan *Mikrotik Router*. Melalui *Script* PHP *Class* ini PHP dapat bekerja untuk mengakses dan mengeksekusi berbagai perintah *Router Mikrotik.* Salah satunya menambahkan, mengedit maupun menghapus *User hotspot*. PHP *Class* ini bersifat sebagai

penghubung antara PHP dengan *Mikrotik,* untuk melakukan explorasi *scripting* sepenuhnya tergantung pada logika programmer, (Suarman et al., 2017).

API *Class Mikrotik* adalah komponen dalam *RouterOS Mikrotik* yang memungkinkan pengembang untuk berinteraksi dengan perangkat *Mikrotik* melalui antarmuka pemrograman aplikasi (API). Kajian teori tentang API *Class Mikrotik* melibatkan pemahaman tentang cara menggunakan dan mengintegrasikan API *Mikrotik* dalam pengembangan aplikasi atau skrip. Berikut ini adalah beberapa poin penting yang perlu dipahami:

1. Fungsi API *Class Mikrotik* : *API Class Mikrotik* memungkinkan pengembang untuk melakukan operasi dan interaksi dengan perangkat *Mikrotik* melalui protokol API (seperti API RPC atau API *RESTful*). Penggunaan API *Class Mikrotik* dapat mencakup operasi seperti membaca dan menulis konfigurasi perangkat, melakukan tindakan administratif, mengambil informasi tentang perangkat, dan banyak lagi.
2. Protokol API: API *Class Mikrotik* mendukung beberapa protokol API yang dapat digunakan dalam pengembangan aplikasi, termasuk API RPC dan API *RESTful*. Pemahaman tentang protokol API ini melibatkan pemahaman tentang metode HTTP (*GET, POST, PUT, DELETE*) dan format data yang digunakan dalam permintaan dan tanggapan API.
3. Autentikasi dan Otorisasi: API *Class Mikrotik* menyediakan mekanisme autentikasi dan otorisasi untuk mengamankan akses ke perangkat *Mikrotik.* Pemahaman teori tentang autentikasi dan otorisasi melibatkan pemahaman

tentang cara menghasilkan token autentikasi, mengelola izin pengguna, dan mengamankan permintaan API.

4. Manipulasi Konfigurasi: API *Class Mikrotik* memungkinkan pengembang untuk membaca, menulis, dan memperbarui konfigurasi perangkat *Mikrotik.* Ini melibatkan pengaturan pengguna, konfigurasi jaringan, pengaturan *firewall*, konfigurasi *routing*, manajemen antarmuka, dan banyak lagi. Pemahaman teori tentang manipulasi konfigurasi melibatkan pemahaman tentang struktur konfigurasi *Mikrotik,* sintaksis konfigurasi, dan operasi yang tersedia melalui API.
5. Pengambilan Informasi: API *Class Mikrotik* memungkinkan pengembang untuk mengambil informasi dari perangkat *Mikrotik* seperti status jaringan, statistik kinerja, log, dan lainnya. Pemahaman teori tentang pengambilan informasi melibatkan pemahaman tentang jenis informasi yang tersedia, format data yang diterima, dan operasi API yang dapat digunakan untuk mengambil informasi tersebut.
6. *Error Handling*: Pemahaman teori tentang penanganan kesalahan (*error handling*) dalam pengembangan dengan API *Class Mikrotik* melibatkan pemahaman tentang kode status HTTP yang diterima dalam tanggapan API, penanganan kesalahan yang mungkin terjadi, dan cara menangani dan melaporkan kesalahan secara efektif.

Pemahaman teori tentang API *Class Mikrotik* mencakup aspek-aspek yang telah disebutkan di atas, tetapi juga dapat berkaitan dengan konsep-konsep dan

teknologi terkait seperti pemrograman aplikasi berbasis REST, penanganan permintaan asinkron, dan pengoptimalan kinerja API.

8. *Simple Queues*

Simple Queues adalah fitur yang disediakan oleh *Mikrotik* RouterOS, sebuah sistem operasi yang digunakan oleh perangkat *Mikrotik* untuk fungsi *routing* dan manajemen jaringan. Fitur *Simple Queues* ini memungkinkan pengguna untuk mengontrol dan membatasi lalu lintas data yang melewati router berdasarkan aturan-aturan tertentu.

Berikut adalah beberapa poin penting dalam kajian teori tentang *Simple Queues*:

1. Konsep *Simple Queues: Simple Queues* adalah mekanisme yang digunakan untuk membatasi lalu lintas data yang melewati antarmuka (*interface*) tertentu pada router. Pengguna dapat menentukan batasan kecepatan (*Bandwidth*) untuk lalu lintas *upload* (*upload traffic*) dan lalu lintas *download* (*download traffic*) untuk setiap aturan yang ditentukan.
2. Jenis-jenis *Simple Queues*: Ada dua jenis utama *Simple Queues* dalam *Mikrotik* RouterOS:
   a. *Simple Queue*: Memungkinkan pengguna untuk membatasi lalu lintas berdasarkan alamat IP dan/atau subnet tertentu.
   b. PCQ (*Per Connection Queues*): Memungkinkan pembagian *Bandwidth* yang lebih adil antara pengguna berdasarkan koneksi individu.
3. Prioritas dan Pembatasan *Bandwidth*: Pengguna dapat memberikan prioritas tertentu untuk aturan *Simple Queues*. Hal ini memungkinkan lalu lintas data

yang mendapatkan prioritas lebih tinggi untuk diutamakan dalam alokasi *Bandwidth*.

4. Pengaturan Antrian (*Queueing*): *Simple Queues* memungkinkan pengguna untuk mengatur antrian lalu lintas data berdasarkan berbagai faktor seperti batas *Bandwidth*, prioritas, dan hingga aturan *burst* untuk mengatasi situasi dimana lalu lintas melebihi batas *Bandwidth* yang ditentukan.
5. Penggunaan dalam Manajemen Jaringan: *Simple Queues* digunakan untuk mengontrol lalu lintas data dalam berbagai skenario, seperti memastikan pengguna internet tetap mendapatkan *Bandwidth* yang cukup, membatasi lalu lintas *download* atau *upload*, atau memberikan prioritas untuk aplikasi atau layanan khusus.
6. *Monitoring* dan Statistik: *Mikrotik* RouterOS menyediakan alat untuk memantau lalu lintas dan mengumpulkan statistik tentang penggunaan *Simple Queues*, yang memungkinkan admin jaringan untuk melacak dan menganalisis bagaimana lalu lintas data dikendalikan dan diprioritaskan.

Penting untuk memahami kajian teori tentang *Simple Queues* untuk mengimplementasikan pengaturan lalu lintas yang efektif dan efisien dalam jaringan *Mikrotik.* Dengan menggunakan *Simple Queues* dengan bijaksana, admin jaringan dapat meningkatkan kualitas layanan, mengoptimalkan kinerja jaringan, dan memastikan pengalaman pengguna yang lebih baik.

9. Hotspot

Hotspot merupakan istilah yang digunakan untuk menggambarkan area dimana terdapat akses nirkabel (*Wi-Fi*) yang dapat digunakan oleh pengguna untuk

terhubung ke internet. Kajian teori tentang *hotspot* melibatkan pemahaman tentang konsep, manfaat, dan implementasi *hotspot* dalam konteks teknologi komunikasi. Berikut adalah beberapa hal penting dalam teori tentang *hotspot*:

1. Konsep *Hotspot*: *Hotspot* adalah area yang menyediakan akses nirkabel ke jaringan komunikasi, biasanya menggunakan teknologi *Wi-Fi*. *Hotspot* dapat ditemukan di berbagai lokasi seperti kafe, hotel, bandara, pusat perbelanjaan, dan tempat umum lainnya. Pengguna dapat terhubung ke *hotspot* menggunakan perangkat elektronik seperti laptop, *smartphone*, atau tablet.
2. Infrastruktur *Hotspot*: *Hotspot* memerlukan infrastruktur yang terdiri dari akses poin (*access point*) *Wi-Fi* yang terhubung ke jaringan internet. Access *point* adalah perangkat keras yang mengirimkan sinyal *Wi-Fi* sehingga perangkat pengguna dapat terhubung ke jaringan. Infrastruktur *hotspot* juga melibatkan server otentikasi yang memvalidasi pengguna dan memberikan akses ke jaringan.
3. Manfaat *Hotspot*: *Hotspot* memberikan manfaat yang signifikan dalam menyediakan akses internet yang mudah di tempat-tempat umum. Beberapa manfaat *hotspot* antara lain:
    a. Kemudahan Akses: Pengguna dapat dengan mudah terhubung ke internet tanpa perlu kabel atau konfigurasi jaringan yang rumit.
    b. Mobilitas: *Hotspot* memungkinkan pengguna untuk tetap terhubung saat berpindah dari satu lokasi *hotspot* ke lokasi *hotspot* lainnya.

c. Produktivitas: *Hotspot* memungkinkan akses internet di tempat-tempat seperti kafe atau pusat bisnis, meningkatkan produktivitas pengguna yang membutuhkan akses internet saat bepergian.

d. Layanan Publik: *Hotspot* dapat menjadi layanan publik yang meningkatkan kepuasan pengunjung di tempat-tempat umum seperti bandara atau stasiun.

4. Keamanan *Hotspot*: Keamanan *hotspot* menjadi aspek penting dalam kajian teori. Pengguna *hotspot* perlu memperhatikan langkah-langkah keamanan seperti enkripsi data menggunakan protokol WPA/WPA2 (*WiFi Protected Access/WiFi Protected Access* 2), penggunaan kata sandi yang kuat, dan kesadaran akan resiko keamanan seperti serangan penyadapan data (*sniffing*) di *hotspot* umum.

5. Implementasi *Hotspot*: Implementasi *hotspot* melibatkan penyiapan infrastruktur *hotspot*, konfigurasi *access point*, pengaturan keamanan, dan manajemen pengguna. Administrasi *hotspot* dapat melibatkan pengelolaan pemantauan dan manajemen kapasitas agar pengguna *hotspot* dapat mengalami akses internet yang andal dan berkualitas.

Teori tentang *hotspot* terus berkembang seiring dengan perkembangan teknologi komunikasi nirkabel. Ini mencakup aspek-aspek seperti kecepatan, kehandalan, keamanan, manajemen jaringan, dan inovasi terkait *hotspot.*

10. Flowchart

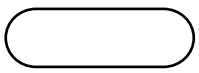
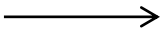
Flowchart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses

(instruksi) dengan proses lainnya dalam suatu program. (Hastuty, 2021) Perancangan dan desain flowchart memperkenalkan simbol-simbol flowchart tersebut melalui beberapa media yaitu dengan cara menggambarkannya di papan tulis, presentase dengan memanfaatkan proyektor dan referensi-referensi lainnya baik melalui buku-buku literatur, jurnal, sumber-sumber dari internet dan sebagainya. Penggunaan media belajar untuk memperkenalkan hal tersebut kurang memadai jika tidak ada pendukung lainnya, dimana teknologi informasi dan komunikasi mempunyai peranan dalam hal itu (Harahap, 2017).

Dalam perancangan *flowchart* sebenarnya tidak ada rumus atau patokan yang bersifat mutlak (pasti). Hal ini didasari oleh bagan alir (*flowchart*) adalah sebuah gambaran dari hasil pemikiran dalam menganalisis suatu permasalahan dalam komputer. Karena setiap analisa akan menghasilkan hasil yang bervariasi antara satu dan lainnya. Kendati begitu secara garis besar setiap perancangan *flowchart* selalu terdiri dari tiga bagian, yaitu *input*, proses dan *output*.

Berikut ini adalah beberapa simbol yang digunakan dalam menggambar suatu *flowchart* :

Tabel 2. 1 *Simbol flowchart*

Simbol	Nama	Fungsi
	*Terminator*	Permulaan / akhir program
	Garis alir	Arah aliran program
	Preparation	Proses *inisialisasi* /pemberian harga awal

	Proses	Proses perhitungan/proses pengolahan data
	Decision	Perbandingan pernyataan penyelesaian data yang memberikan pilihan untuk langkah selanjutnya.
	Input/Ouput data	Proses *input/ouput* data.
	On page	Penghubung bagian-bagian *flowchart* yang berada pada satu halaman

11. UML (Unified Modelling Language)

UML *(Unified Modelling Language)* menurut menurut Adi nugroho (2010:6), "*Unified Modelling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek". Pemodelan (modeling) sebenarnya digunakan untuk menyederhanakan masalah yang kompleks agar lebih mudah dipelajari dan dipahami.

Metodologi UML (*Unified Modelling Language*) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem atau perangkat lunak yang akan dikembangkan yaitu :

1. Sesuatu (*things*)

Ada empat *things* dalam *UML* (*Unified Modelling Language*):

a. *Structural things*, bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.

b. *Behavorial things,* bagian dinamis biasanya merupakan kata kerja dari model *UML* yang mencerminkan perilaku sepanjang waktu

c. *Grouping things,* bagian pengorganisasian dalam *UML*. Dalam

penggambaran model *UML* yang rumit diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta *subsitem-subsistem*.

d. *An notational things,* merupakan bagian yang memperjelas model *UML*. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap elemen dalam model *UML*.

2. Relasi (*relationship*)

Ada empat *relationship* (hubungan) dalam *Unified Modelling Language* (*UML*):

a. Ketergantungan (d*ependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemen *independent* akan mempengaruhi elemen yang bergantung padanya.

b. *Asosiasi* adalah apa dan bagaimana yang mengubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.

c. *Generalisasi* adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada diatasnya (objek induk). Arah dari objek induk ke objek anak dinamakan spesialisasi sedangkan arah sebaliknya dinamakan generalisasi.

d. Realisasi adalah operasi yang benar-benar dilakukan oleh suatu objek.

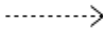
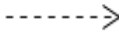
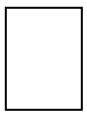
3. Diagram, *Unified Modelling Language* (*UML*) menyediakan Sembilan jenis diagram yang dapat dikelompokkan berdasarkan sifatnya (statis dan dinamis).

a. Diagram *use case*

Diagram *use case* menyajikan interaksi antara *use case* dan aktor, dimana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang di bangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai

Adapun simbol-simbol *Use Case* Diagram antara lain :

Tabel 2. 2 *Simbol Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menypesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2. 3 *Simbol Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri

Tabel 2. 4 *Simbol Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-

			informasi tentang aktifitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 2. 5 *Simbol StateChart Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*State*	Nilai atribut dan nilai *link* pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah *state* objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 6 *Simbol Actifity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

12. *Flowgraph*

Grafik alir (*flowgraph*) adalah sebuah notasi sederhana yang merepresentasikan aliran kontrol dari sebuah struktur program. Dalam sebuah grafik alir, anak panah disebut sebagai sisi (*edge*, E) merepresentasikan aliran kontrol. Lingkaran disebut sebagai simpul (*node*, N) merepresentasikan satu atau lebih aksi/pernyataan logis. Daerah yang dibatasi oleh sisi dan simpul disebut area (region, R). Simpul yang mengandung keputusan disebut sebagai (*predicate node,* P) yaitu simpul yang mengeluarkan lebih dari satu sisi(Rafli et al., 2024)

Flowgraph: Langkah pertama dalam teknik basis *path* adalah membuat *flowgraph* atau diagram alir dari program yang akan diuji. *Flowgraph* ini berisi simpul-simpul yang mewakili titik-titik keputusan (*decision points*) dan jalur-jalur yang menghubungkan simpul simpul tersebut. Menentukan Jumlah *Edges* (E) dan *Nodes* (N) Setelah *flowgraph* selesai dibuat, langkah berikutnya adalah menghitung jumlah *edges* (garis yang menghubungkan simpul) dan *nodes* (simpul) dalam *flowgraph* tersebut. *Edges* menggambarkan jalur atau perpindahan dari satu simpul ke simpul lainnya, sedangkan *nodes* adalah titik-titik keputusan atau pilihan dalam program (Khamaeni, 2023).

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian ini bersifat kuantitatif, karena aplikasi dipergunakan untuk mengelola aspek teknis manajemen *bandwidth* dan *internet gateway* dengan menggunakan platform *Mikrotik* berbasis *web*. Penekanan pada manajemen dan pengaturan teknis mengarahkan pada pengumpulan data yang dapat diukur secara kuantitatif, seperti kecepatan internet, utilitas *bandwidth*, dan parameter jaringan lainnya.

B. Waktu Penelitian

Waktu yang dipergunakan untuk pelaksanaan penelitian ini mulai dari Januari 2023 – Agustus 2023.

C. Metode Pengumpulan Data

Metode pengumpulan data merupakan tahapan dari proses penelitian dalam proses pengembangan sistem. Penulis membutuhkan data yang relevan agar penelitian dapat berjalan sesuai dengan rumusan masalah yang teridentifikasi. Metode pengumpulan data yang digunakan penulis adalah studi pustaka dan studi literatur.

1. Studi Pustaka pada tahap ini mengumpulkan materi yang berkaitan dengan judul topik, dengan membaca buku-buku dari perpustakaan dan mencari referensi artikel dan e-*book* di *internet*. Judul lengkap buku dan situs web dapat dilihat di daftar pustaka .

2. Studi Literatur pada tahap ini penulis membandingkan dan dijadikan sebagai acuan simulasi pembelajaran dan ujian. Oleh karena itu, perlu untuk mengamati penelitian serupa yang telah dilakukan. Informasi lebih lanjut dapat ditemukan di bab ii

D. Alat Dan Bahan Praktek

Adapun alat dan bahan yang kami gunakan terbagi menjadi 2 yaitu *Hardware* dan *Software* :

Tabel 3. 1 *Hardware*

No	Nama	Volume	Keterangan
1	Modem Indihome ZTE F609	1 Jalur	Kecepatan *Bandwidth* 50Mbps
2	*Mikrotik*	1 Unit	Router Board RB951-2ND
3	*Handphone*	1 Unit	Vivo Y50
4	Asus X441NA	1 Unit	Processor intel pentium, RAM 4GB, HDD 500GB, SSD 120GB.

Tabel 3. 2 *Software*

No	Nama	Keterangan
1	*Mikrotik* RouterOS™	Versi. 6.0
2	XAMPP	Versi. 3.3.0
3	Visual Code Studio	Versi 1.79.0
4	Windows 10	Professional 2019

E. Tahapan Penelitian

Tahapan penelitian yang akan dilakukan didalam penelitian ini diantaranya yaitu persiapan penelitian, pengumpulan data, analisis data, perancangan sistem, pengujian dan implementasi. Adapun uraian dari tahapan tersebut sebagai berikut:

1. Persiapan penelitian

   Tahapan ini peneliti melakukan persiapan penelitian yang dimaksud menyiapkan buku, artikel dan tentang topik penelitian serta aplikasi yang digunakan selama meneliti.

2. Pengumpulan data

   Tahap ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu mempelajari buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain. Yang bertujuan untuk mendapatkan landasan teori mengenai masalah yang akan diteliti dengan tepat dan sesuai dengan kerangka pikir ilmiah.

3. Analisis kebutuhan

   Pada tahapan analisis, peneliti melakukan identifikasi kebutuhan pengguna terkait manajemen *Bandwidth* dan pengaturan *Gateway* serta menentukan fitur-fiur yang harus ada dalam aplikasi, seperti pengaturan *Bandwidth* dan pemantauan kinerja jaringan.

4. Perancangan sistem

   Peneliti merancang sistem aplikasi termasuk antarmuka pengguna serta alur aplikasi, seperti bagaimana penguna akan mengatur *Bandwidth*.

5. Pengujian

   Setelah melakukan perancangan, peneliti kemudian melakukan pengujian dari hasil perancangan yang telah dibuat untuk memastikan semua bekerja sesuai harapan. Jika hasil perancangan terdapat kekurangan maka Kembali ke tahap analisis sistem.

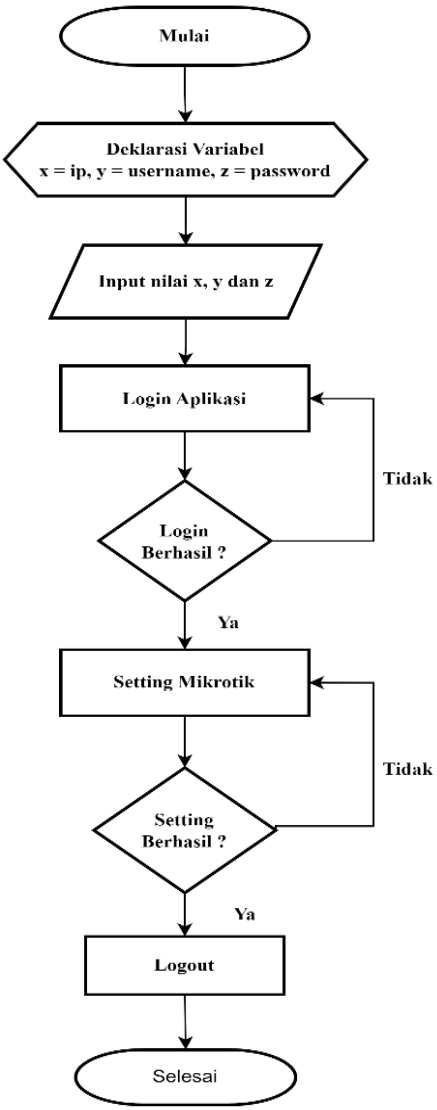
6. Implementasi

   Setelah tahap pengujian tidak terdapat kekurangan maka aplikasi siap untuk di gunakan.

F. Desain Sistem

Dalam merancang sebuah sistem jaringan dibutuhkan alur kerja agar perancangan langkah demi langkah terselesaikan

1. *Flowchart* Perancangan Aplikasi Jaringan

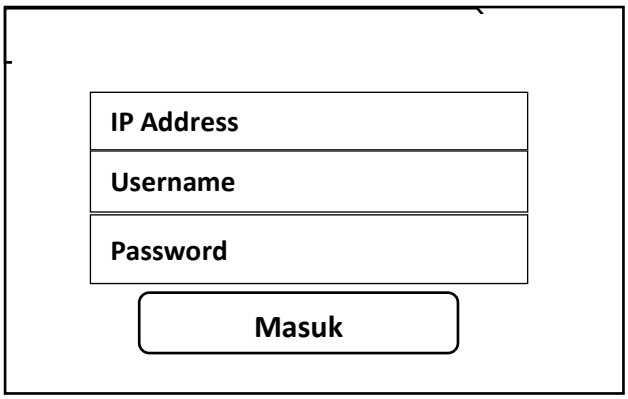


Gambar 3. 1 *Flowchart* Perancangan Sistem

2. *Perancangan AntarMuka (Mockup Interface)*

Perancangan antarmuka (*mockup*) sistem bertujuan untuk memberikan gambaran terkait sistem yang akan dibangun dan terdiri dari beberapa bagian *mockup* yaitu:

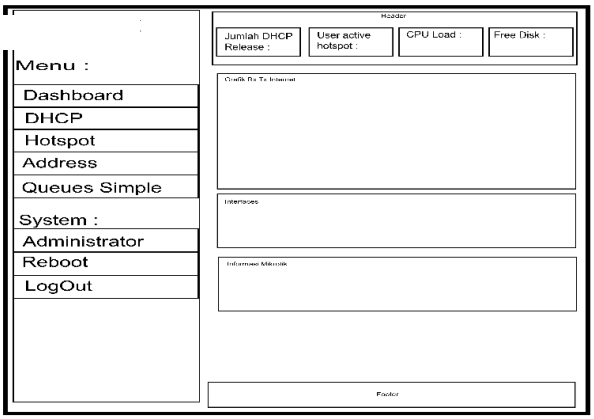
a. Halaman *Login*



Gambar 3. 2 Tampilan Halaman *Login*

Pada **Gambar 3. 2**, halaman *login* menampilkan kolom *ip Address*, *Username*, dan *Password, Admin* dapat mengakses ke dalam *router Mikrotik.*

b. Halaman Utama



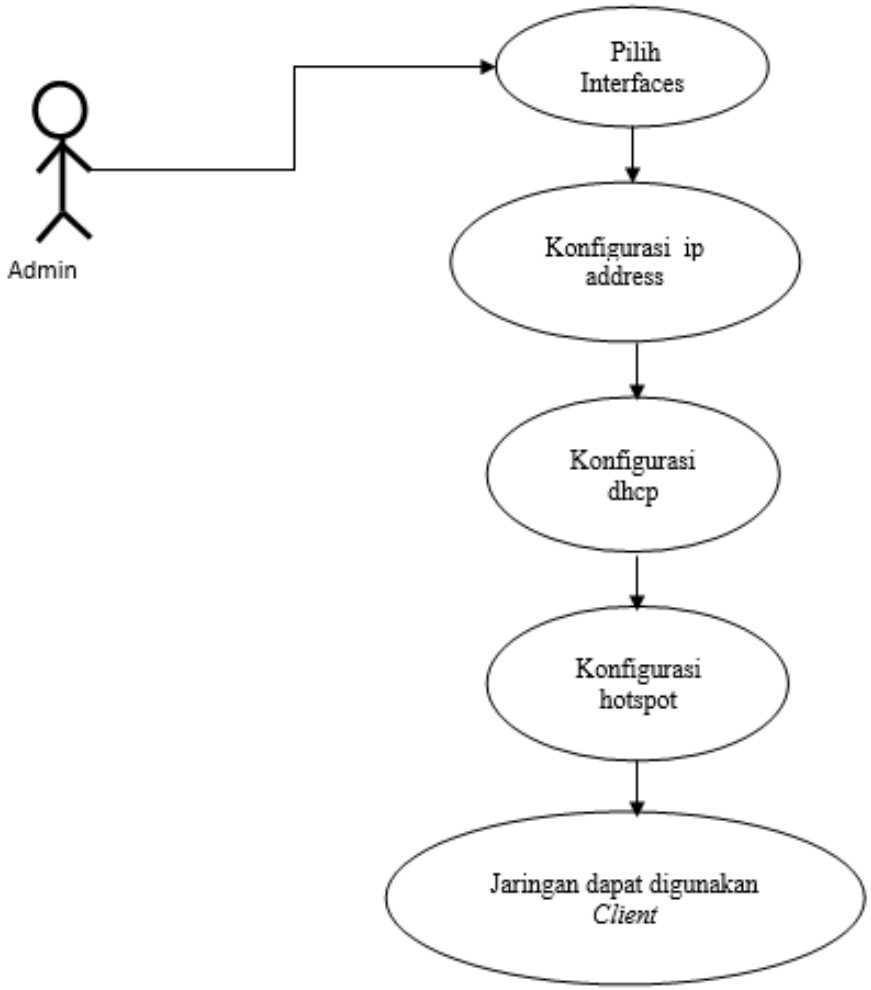
Gambar 3. 3 Tampilan Halaman Utama

Pada Gambar 3.3 terdapat tampilan halaman utama yang terdapat beberapa menu *Interface, Hotspot, Dhcp, Address, Simple Queues, Administrator, Reboot* dan *logout.*

BAB IV

HASIL DAN PEMBAHASAN

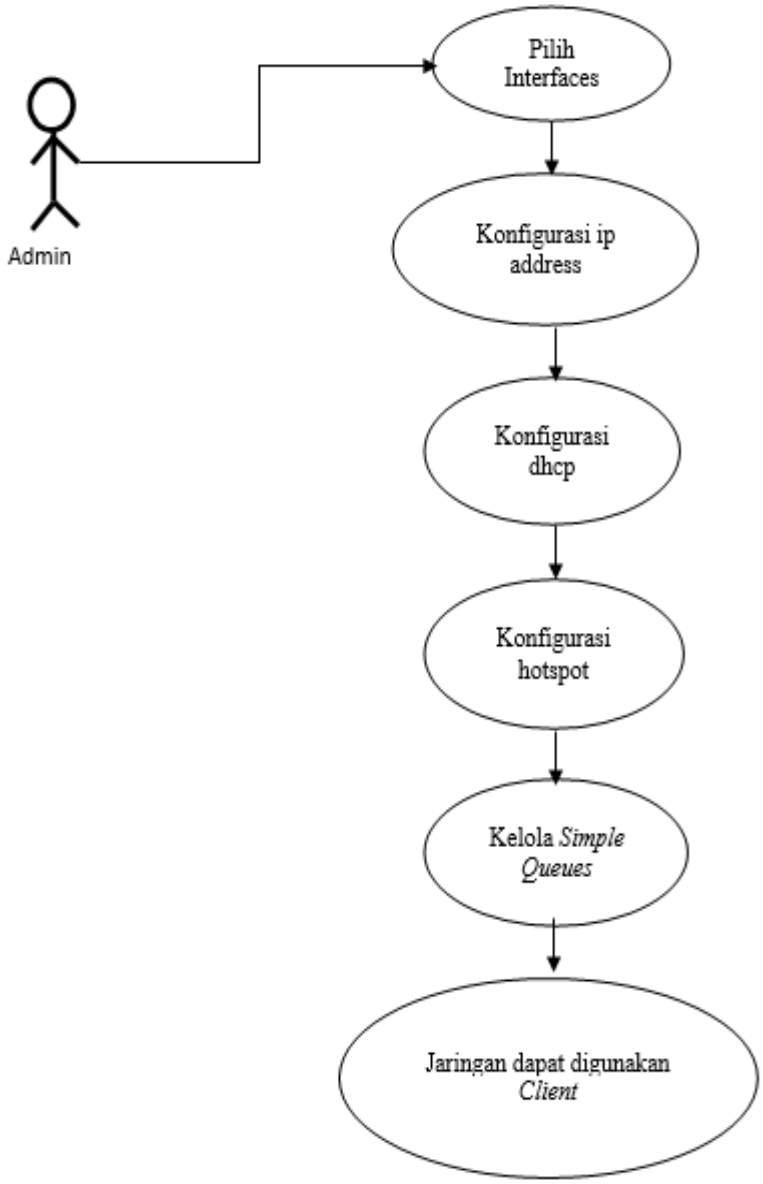
A. Sistem Yang Berjalan



Gambar 4. 1 Sistem Yang Berjalan

Sistem yang berjalan saat ini adalah admin yang dapat *login* menggunakan *ip Address* dari ip default *Mikrotik* kemudian memilih *interface* kemudian Konfigurasi *Address* lalu Konfigurasi *DHCP* (*Dinamic Host Configuration Protocol*) selanjutnya Konfigurasi *Hotspot* dan jaringan dapat digunakan *client*.

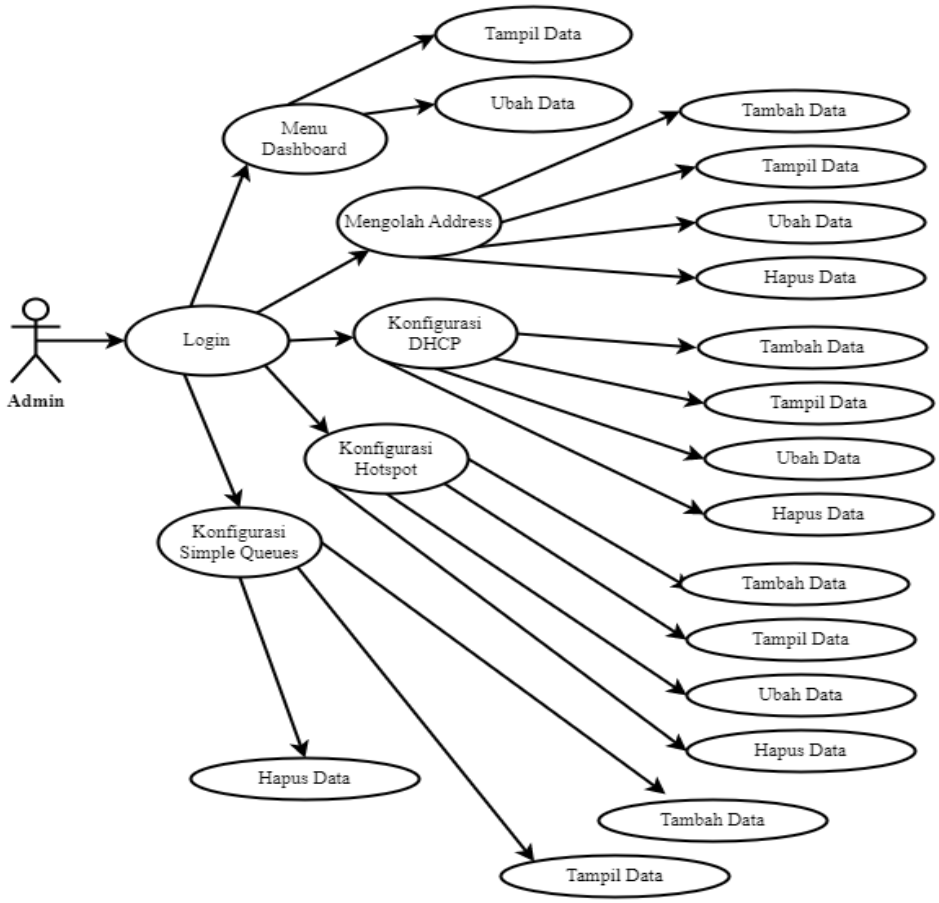
B. Sistem yang diusulkan



Gambar 4. 2 Sistem yang diusulkan

Sistem yang berjalan saat ini adalah admin yang dapat *login* menggunakan *ip Address* dari ip default *Mikrotik* kemudian memilih *interface* kemudian pilih *interfaces* kemudian Konfigurasi *ip Address* selanjutnya Konfigurasi *DHCP(Dynamic Host Configuration Protocol)* lalu Konfigurasi *hotspot* kemudian dapat mengelola data *Simple Queues* dan jaringan dapat digunakan *client*.

C. *Use Case* Aplikasi Manajemen dan Monitoring *Mikrotik*



Gambar 4. 3 *Use Case Admin*

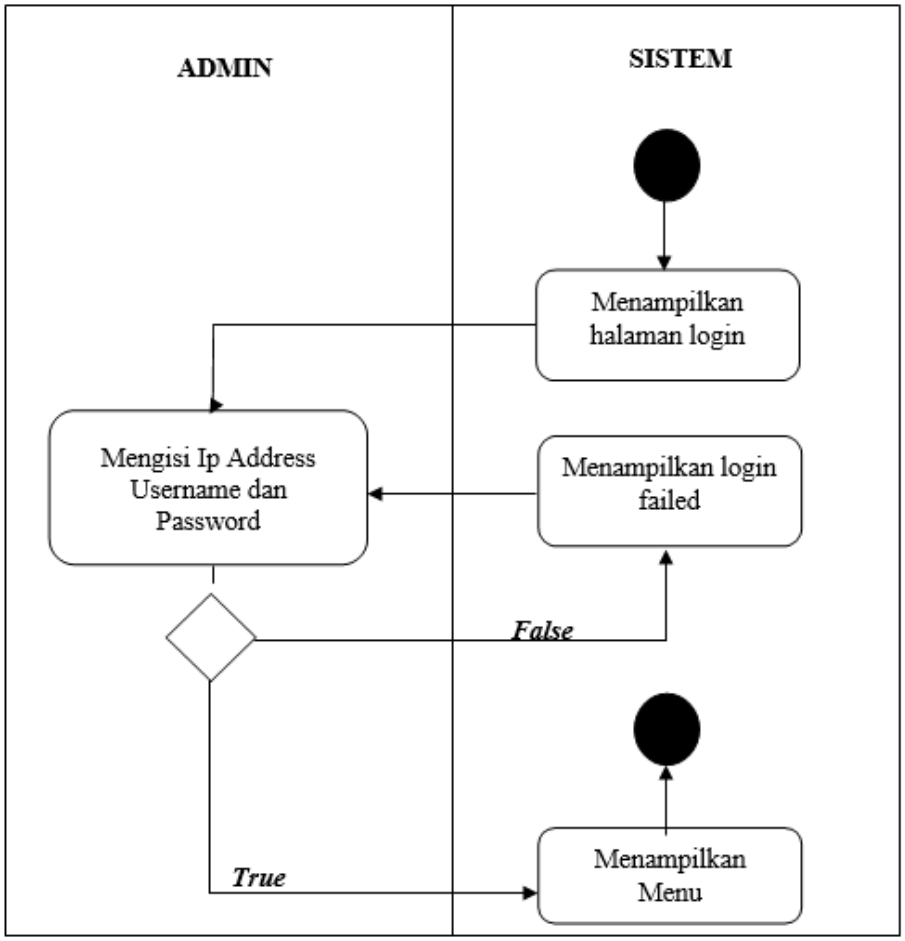
Pada Gambar 4.3 Merupakan *Use Case* untuk admin diatas menjelaskan apa yang dapat dilakukan pada aplikasi Manajemen dan *Monitoring Mikrotik.* Untuk menggunakan menu pada aplikasi yang ada admin harus melakukan *login*, setelah *login* berhasil admin dapat memulai memonitoring server melalui pilihan menu yang ada. Secara teknis admin dapat mengkonfigurasi *mikrotik* menggunakan fitur-fitur aplikasi.

D. Perancangan Sistem Aplikasi

1. *Activity* Diagram

Diagram aktivitas atau *activity* diagram menggambarkan alur kerja atau pengoperasian suatu sistem atau proses bisnis atau menu dalam perangkat lunak. Yang harus diperhatikan di sini adalah diagram aktivitas menggambarkan aktivitas sistem dan bukan apa yang dilakukan aktor, sehingga aktivitas yang dapat dilakukan oleh sistem. Di bawah ini adalah diagram detail dari setiap aktivitas yang dibuat oleh administrator :

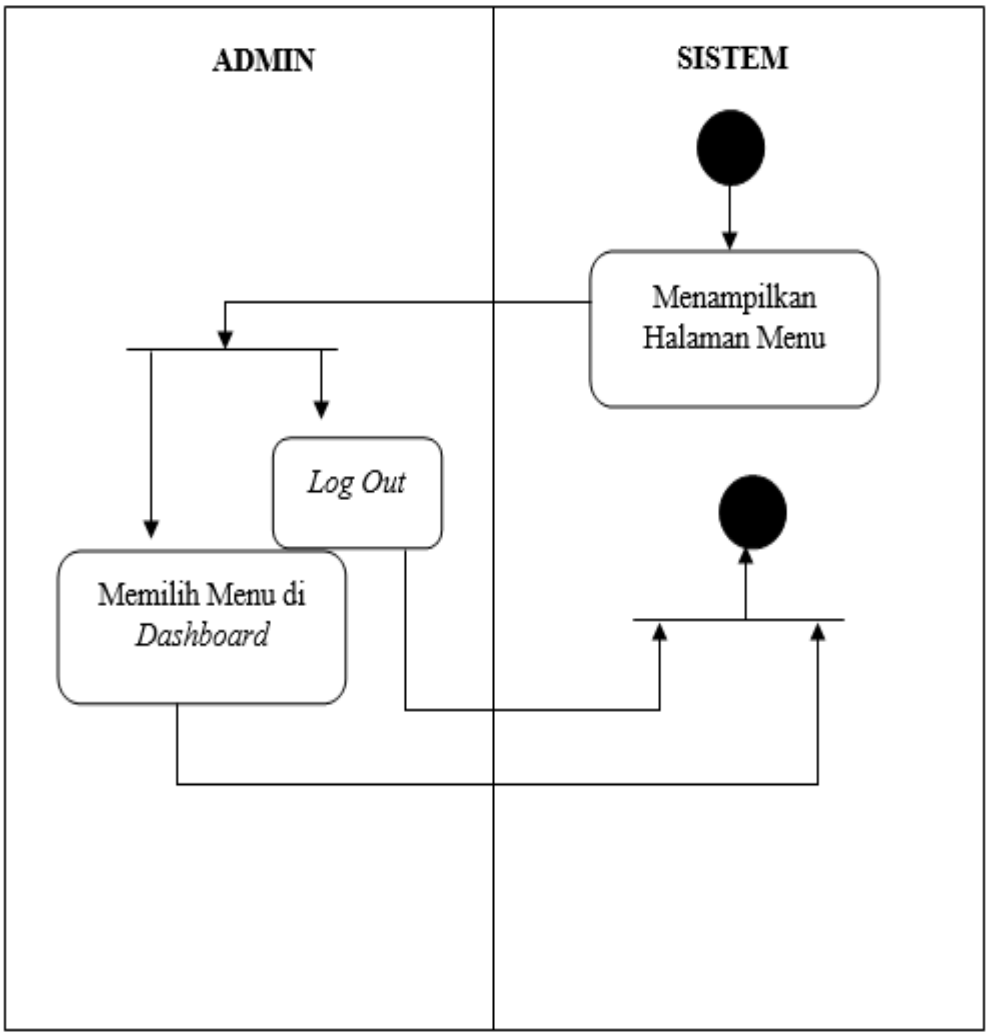
a. *Activity Login*



Gambar 4. 4 *Activity Login*

Pada **Gambar 4.4** diatas Menjelaskan aktivitas yang dilakukan admin saat *login Mikrotik* melalui aplikasi. Saat aplikasi digunakan maka akan muncul sebuah halaman *login* dimana admin harus memasukkan *ip Address*, *Username*, dan *password* untuk memasuki menu Manajemen dan *monitoring Mikrotik*

b. *Activity Dashboard*

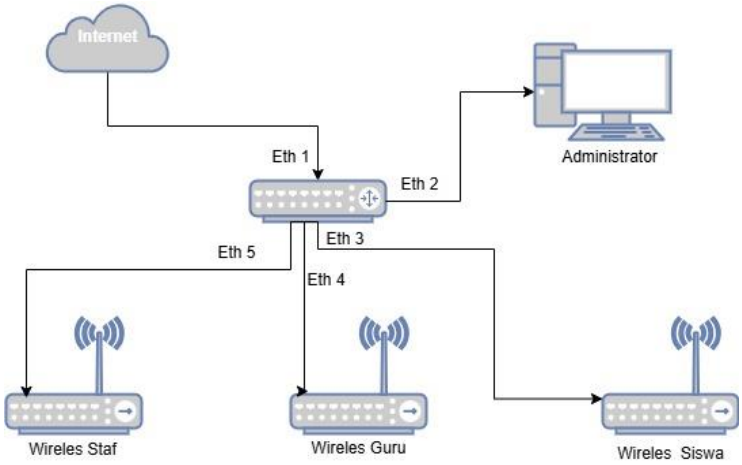


Gambar 4. 5 *Activity Dashboard*

Diatas adalah aktivitas *admin* untuk melihat *Dashboard* yang sedang digunakan maupun yang tidak digunakan.

E. Desain dan Implementasi

Pada tahap awal dilakukan konfigurasi *Mikrotik* menggunakan *router Mikrotik.* Konfigurasi yang dilakukan dengan menggunakan *Mikrotik* tipe RB951UI-2HND sehingga tidak melakukan proses penginstalan. Dalam sistem ini pengaksesan menggunakan aplikasi yang telah dibuat.



Gambar 4. 6 *Topologi* Jaringan SMK Negeri 2 Pinrang

Pada **Gambar 4. 6** merupakan *topologi* jaringan *Local Area Network* (LAN) yang akan di implementasikan di SMK Negeri 2 Pinrang. Jaringan ini dihubungkan ke beberapa jaringan yang terdapat kantor, ruangan tata usaha dan lab.

Tabel 4. 1 Pengguna dan *Bandwidth* SMK Negeri 2 Pinrang

PENGGUNA	***BANDWIDTH***
Siswa	10 Mbps
Guru	10 Mbps
Staf	30 Mbps

Di **Tabel 4.1** kebutuhan *Bandwidth* untuk masing-masing siswa disini diberikan sebesar 10 Mbps. Disebabkan kebutuhan *download*, *streaming*, dan *browsing* tidak sebesar guru dan tata usaha. Kebutuhan siswa disekolah hanya diperuntukkan untuk mencari materi-materi yang berhubungan dengan materi

Pelajaran yang tidak terdapat didalam buku panduan. Sedangkan kebutuhan tata usaha membutuhkan *Bandwidth* lebih besar 30 Mbps karena digunakan untuk mengakses data siswa seperti sistem informasi Data Pokok Pendidikan (Dapodik). Dan *Bandwidth* untuk guru sebesar 10 Mbps digunakan untuk mencari bahan ajar. Selain itu digunakan untuk mengakses video pembelajaran yang digunakan untuk mendukung pembelajaran.

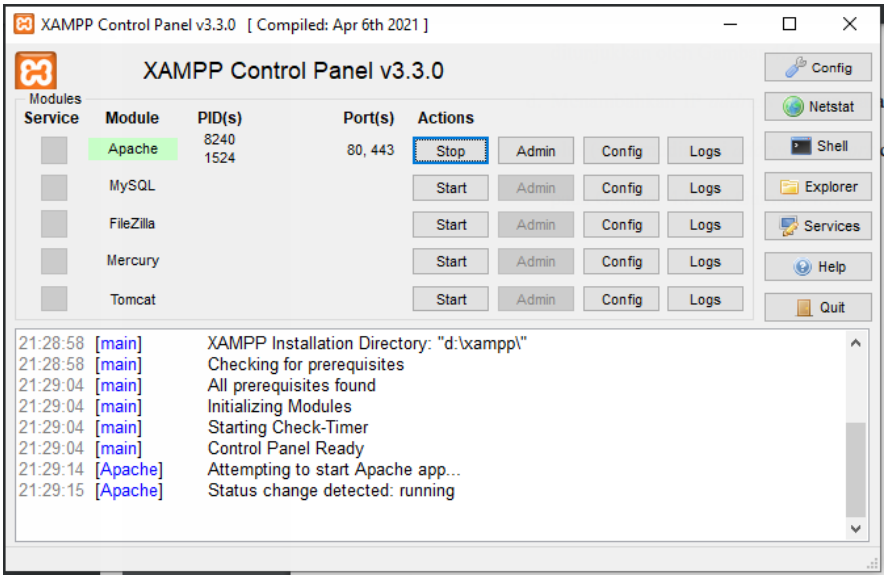
Tabel 4. 2 *Ip Address* di *Interfaces*

NAMA *INTERFACES*	***ADDRESS***
Eth 1 (*Mikrotik*)	*IP DHCP* (*Dynamic Host Configuration Protocol*)
Eth 2 (*Mikrotik*)	192.168.2.1/24
Bridge (Eth 3 dan *Wireless*) *Mikrotik*	172.16.0.1/22 *DHCP Pool* (172.16.1.1 – 172.16.3.254)
Wireless Siswa	172.16.0.2/22
Wireless Guru	172.16.0.3/22
Wireless Staf	172.16.0.4/22

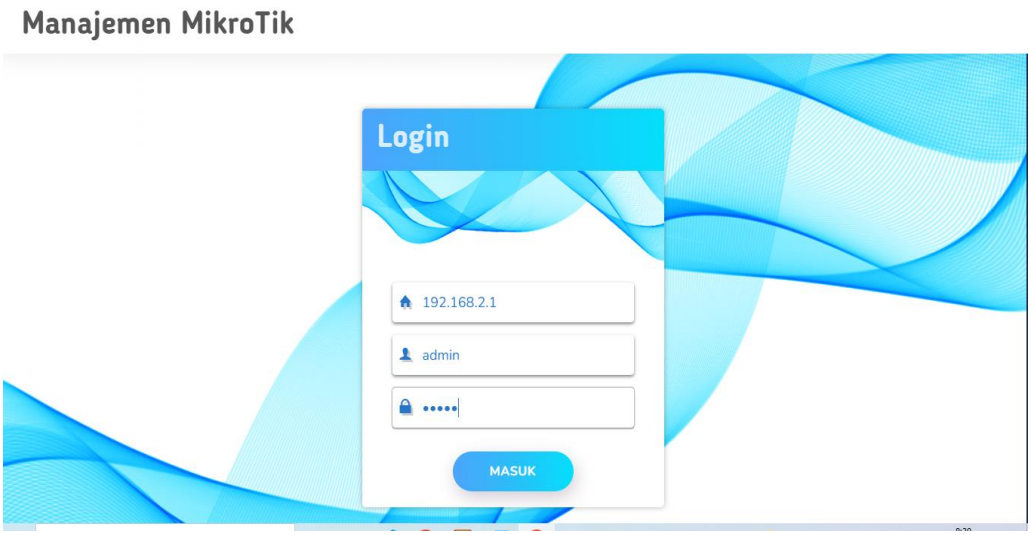
Berikut ini merupakan tampilan dan implementasi aplikasi untuk manajemen *Mikrotik* yang telah dibuat.

a. Desain Tampilan Aplikasi

Sebelum tahap konfigurasi *Mikrotik* terlebih dahulu hubungkan kabel utp dari *internet* atau ISP (*internet service provider*) ke *Mikrotik* di *interface* 1 (eth1) kemudian kabel utp ke-2 dari *interface* 2 (eth2) *Mikrotik* dihubungkan ke komputer/laptop yang digunakan sebagai *administrator* untuk mengkonfigurasi menggunakan aplikasi manajemen *Mikrotik* yang telah dibuat.

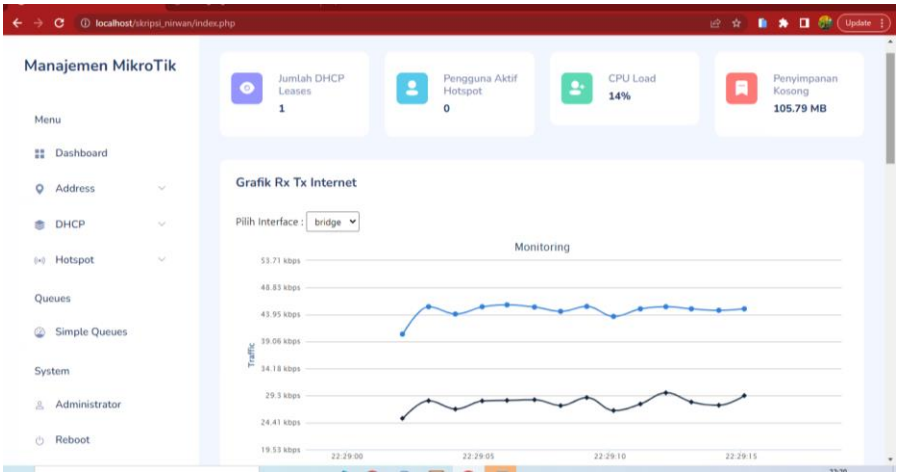


Gambar 4. 7 *Running Apache XAMPP*



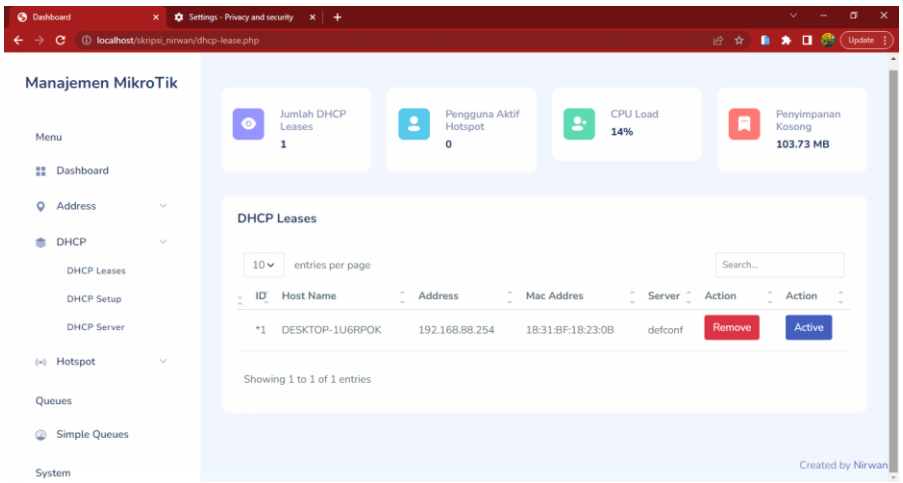
Gambar 4. 8 *Form Login*

Pada **Gambar 4.8** Kemudian membuka browser dengan mengetik http://localhost/skripsi_nirwan dan akan muncul halaman *login* untuk *setting Mikrotik.*



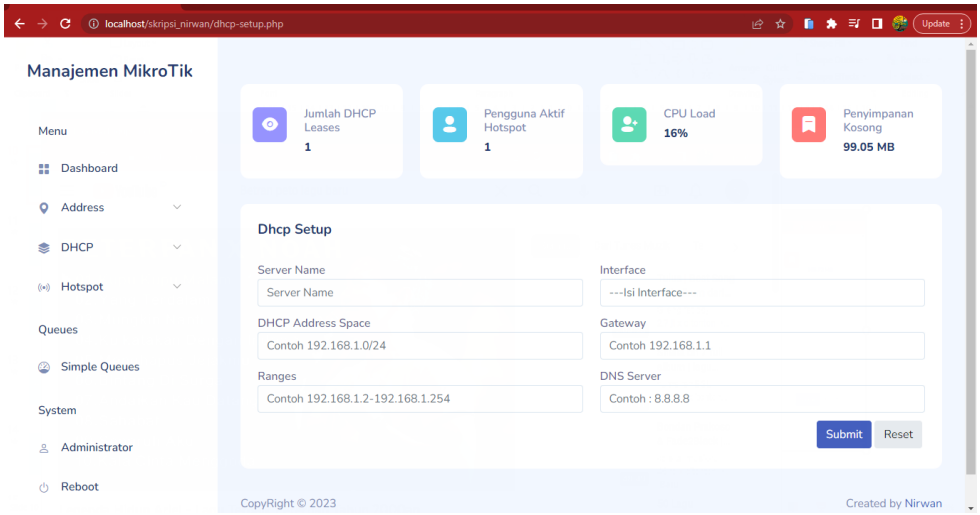
Gambar 4. 9 Tampilan *Dashboard*

Pada **Gambar 4.9** Merupakan tampilan *Dashboard* setelah berhasil *Login*, pada halaman ini terdapat *monitoring* Grafik internet pada *Mikrotik.*



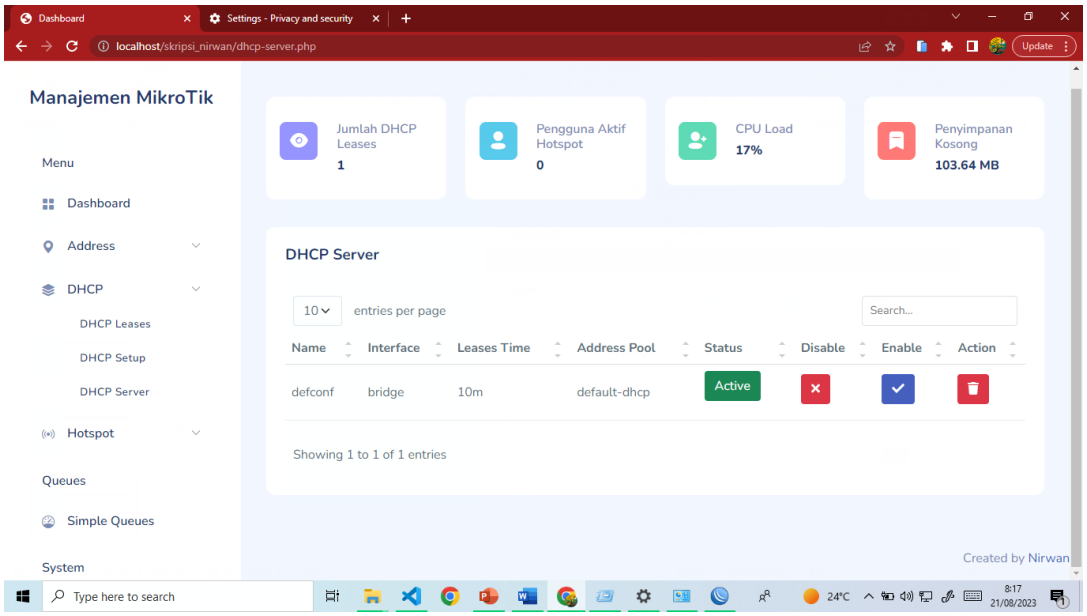
Gambar 4. 10 Tampilan *Menu DHCP Leases*

Pada **Gambar 4.10** Tampilan DHCP *Leases* yang bertujuan untuk melihat perangkat yang terhubung ke jaringan, pada DHCP *Leases* memiliki 2 (Dua) *Button* yaitu *Remove* dan *Active* yang bertujuan untuk mematikan dan mengaktifkan *Address* tersebut.



Gambar 4. 11 Tampilan DHCP Setup

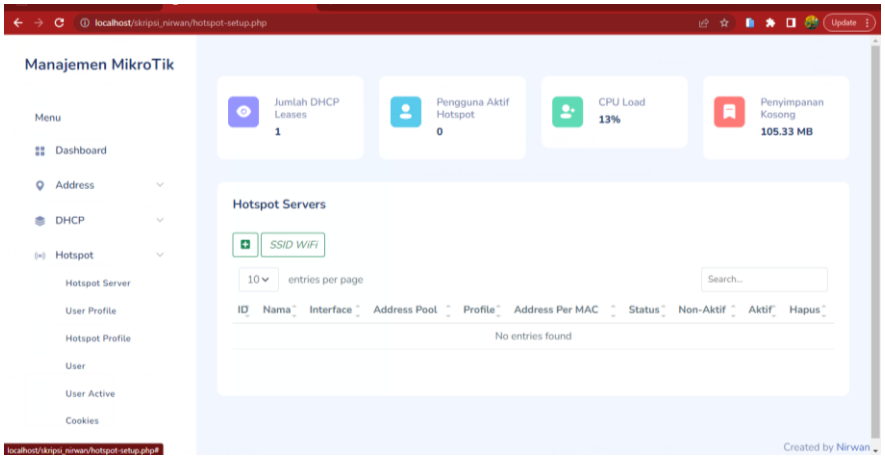
Pada **Gambar 4.11** terdapat DHCP Setup yang dimana menu ini bertujuan untuk penetapan alamat Ip Otomatis kepada perangkat yang terhubung ke jaringan. konfigurasi jaringan lainnya seperti *Gateway, Ranges, interface*, dan server *name*.



Gambar 4. 12 Tampilan *DHCP Server*

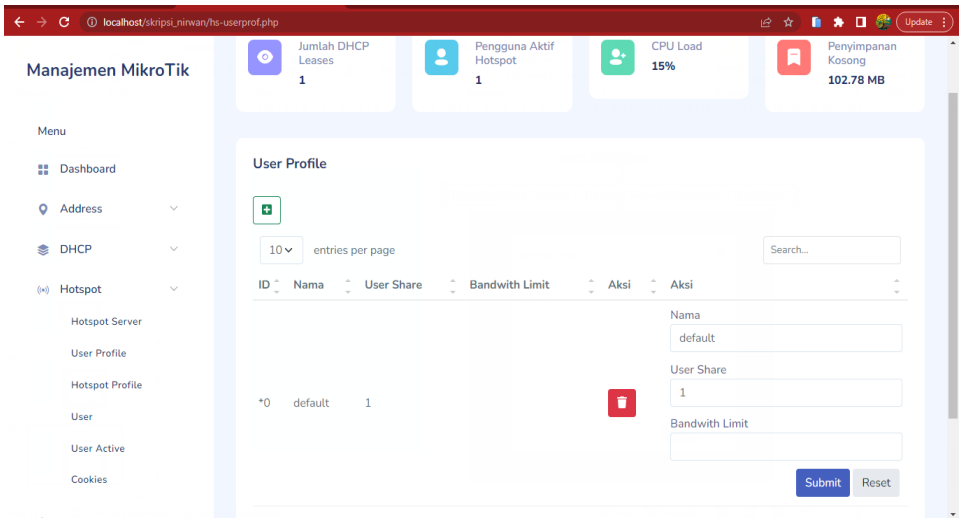
Pada **Gambar 4.12** menu DHCP Server dilengkapi dengan fitur pemantauan dan manajemen. dapat memantau aktivitas DHCP, melihat log, dan memeriksa status klien yang terhubung.

1. Tampilan Menu *Hotspot*



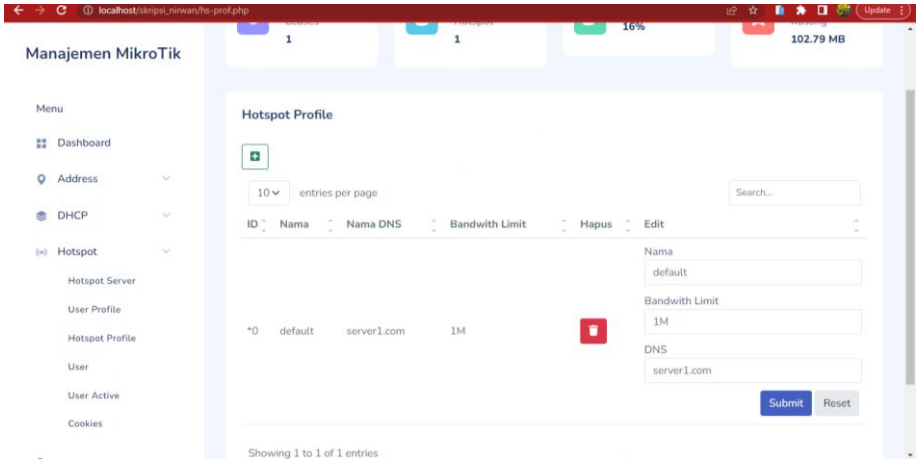
Gambar 4. 13 Tampilan *Hotspot Servers*

Pada **Gambar 4.13** terdapat menu *hotspot servers* yang bertujuan untuk membuat beberapa pengguna yang akan memiliki *ID, Name, Interface, Address Pool, Profile, Address Per MAC, Status* masing-masing pada menu ini terdapat tombol membuat *hotspot*, tombol *disable, enable*, Hapus dan Ubah Nama *WiFi*



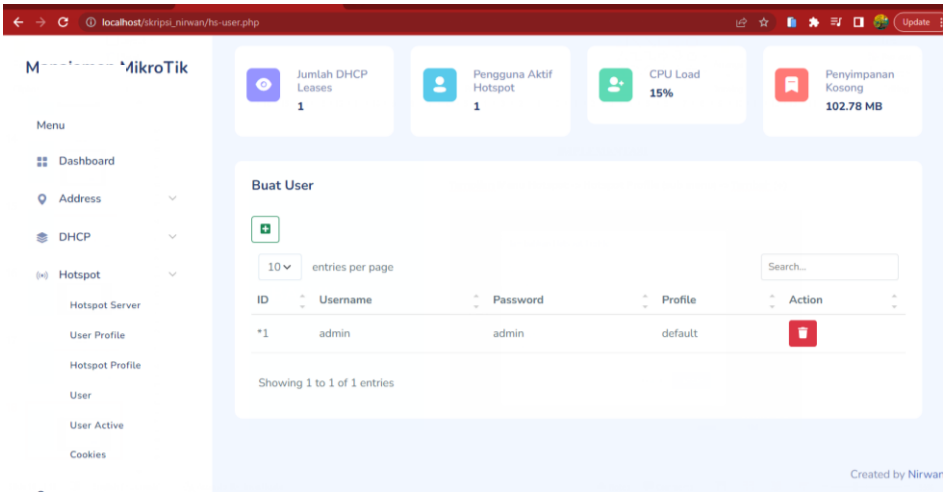
Gambar 4. 14 Tampilan *Hotspot User Profile*

Pada **Gambar 4.14** *Hotspot User Profile* Untuk menambahkan Profile *User* dan mengedit,dan menghapus profile *User*



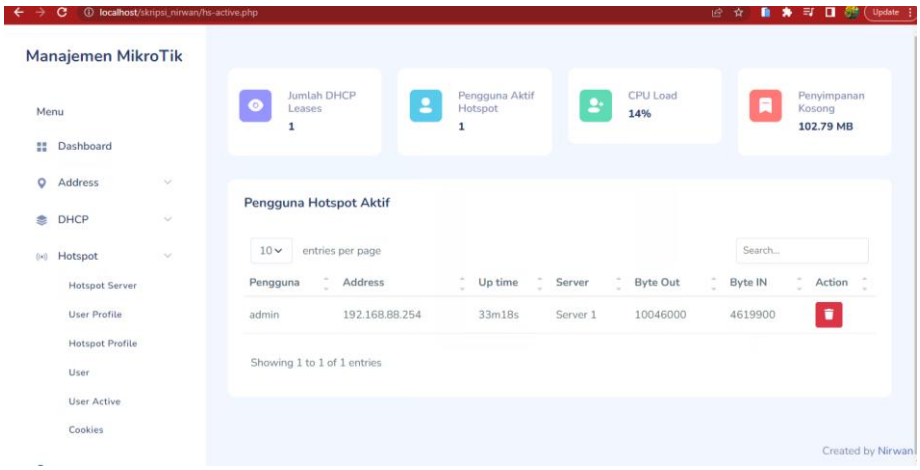
Gambar 4. 15 Tampilan *Hotspot Profile*

Pada **Gambar 4.15** *Hotspot Profile* memungkinkan Anda untuk mengatur dan mengelola sesi pengguna *hotspot*. Pada fitur ini dapat menambahkan *profile* tersendiri untuk pengguna dengan *ID, name, DNS (Domain Name System) name, Bandwidth limit* yang berbeda.



Gambar 4. 16 Tampilan *User*

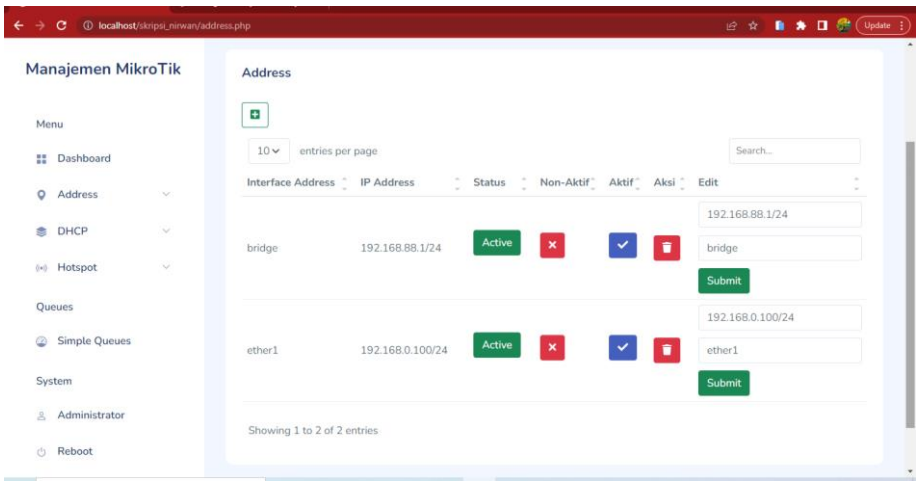
Pada **Gambar 4.16** tampilan *Menu User* Bertujuan untuk pembuatan *Username* dan *password* untuk penggunanya, pada menu ini terdapat tombol *Plus* untuk menambahkan pengguna baru dan tombol *Action* untuk menghapus *User* tersebut.



Gambar 4. 17 Tampilan *Hotspot User Active*

Pada **Gambar 4.17** terdapat menu *Hotspot User Active* dimana pada menu ini admin dapat mengetahui *User* yang sedang aktif dan dapat mengetahui *User, Address, Up Time, Sever, Byte Out, dan Byte IN* pada pengguna tersebut dan memiliki *Action* Untuk menghapus pengguna yang sedang aktif.

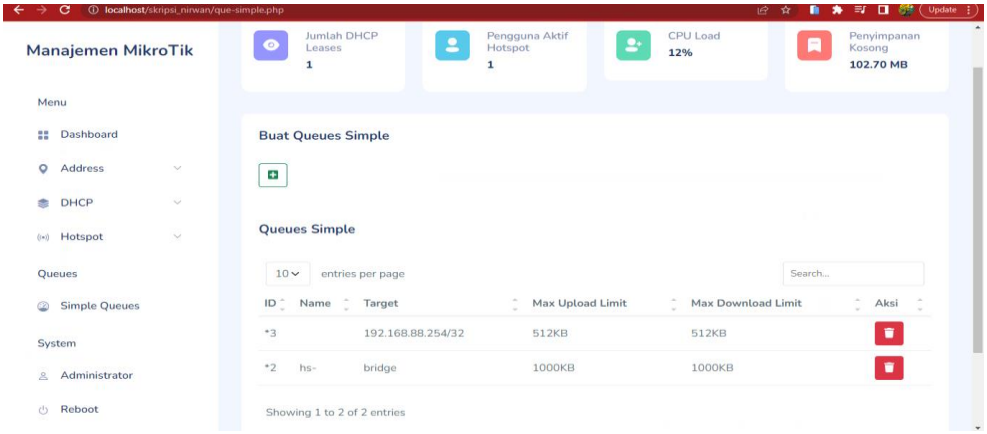
2. Tampilan Menu *Address*



Gambar 4. 18 Tampilan *Menu Address*

Pada **Gambar 4.18** terdapat tampilan menu *Address* yang dimana pada menu ini digunakan untuk mengelola alamat IP pada antarmuka jaringan *Mikrotik* dan dapat menambahkan alamat IP Statis pada antarmuka yang menghubungkannya ke Jaringan Lokal atau Internet.

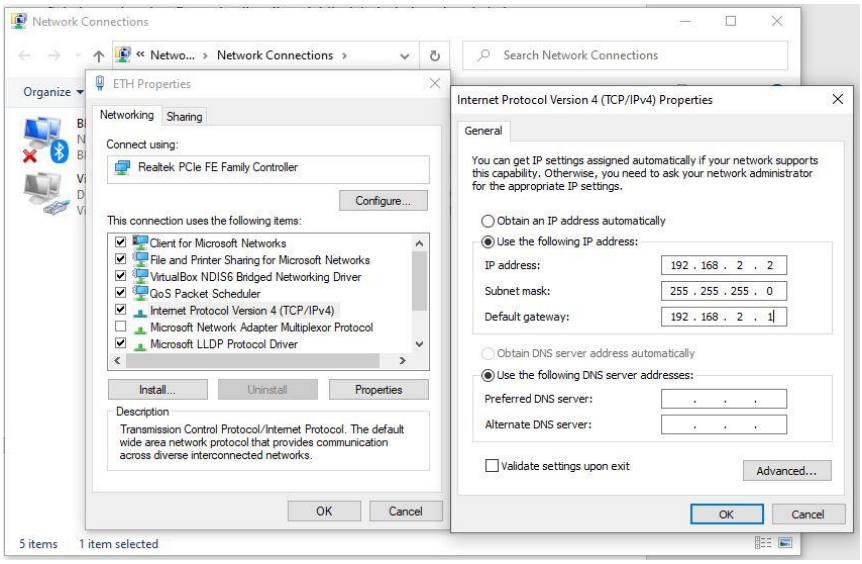
3. Tampilan *Queues Simple*



Gambar 4. 19 Tampilan *Queues Simple*

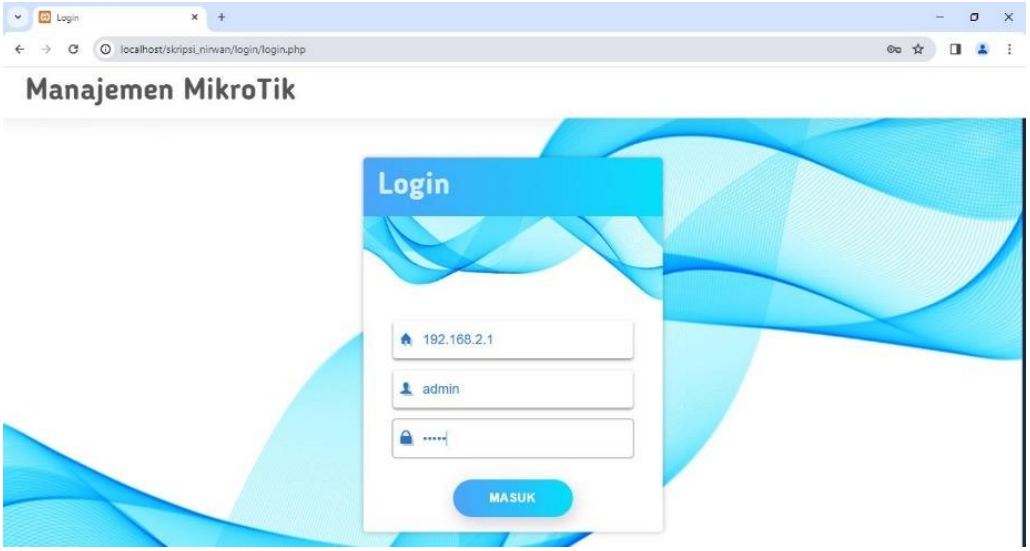
Pada Gambar 4.19 merupakan tampilan *Queues simple* yang menampilkan *client* yang terhubung pada *Mikrotik* dan berfungsi untuk membatasi *Bandwidth* berdasarkan Ip pengguna.

b. Implementasi Jaringan Internet



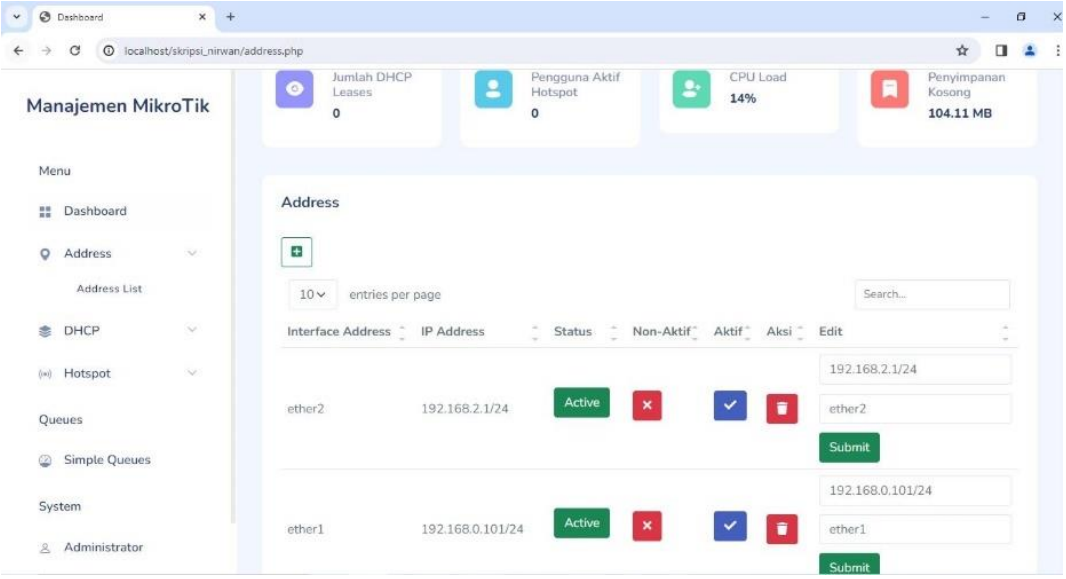
Gambar 4. 20 Konfigurasi *Ip Address* Pada Komputer

Pada **Gambar 4.20** diatas adalah implementasi setting jaringan komputer dengan memberikan ipv4 dimana *ip Address* nya 192.168.2.2, *subnet mask* 255.255.255.0 dan *default gateway* 192.168.2.1 setelah itu OK.



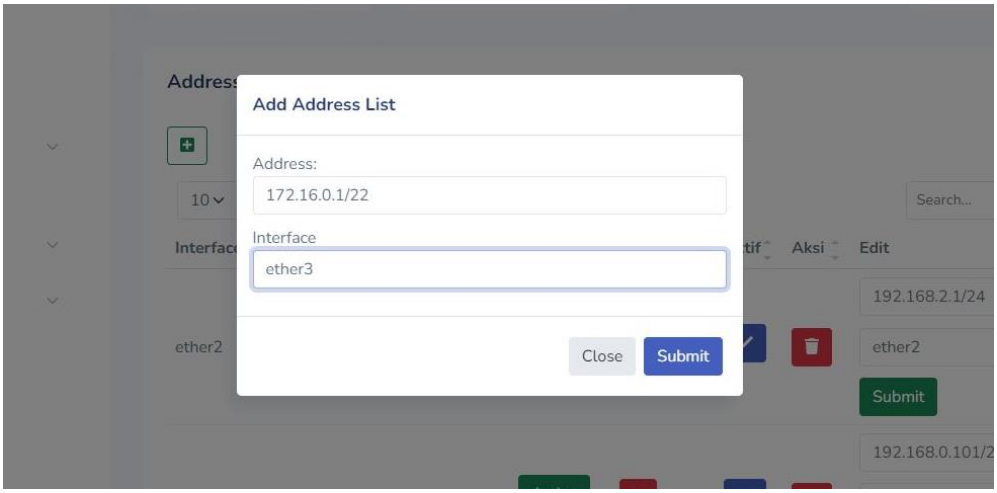
Gambar 4. 21 *Login* pada Aplikasi

Selanjutnya **Gambar 4.21** setelah *Setting Ip* pada komputer kemudian masuk di *browser*, masuk *localhost* dan akan muncul halaman *login*, setelah berhasil *login*.



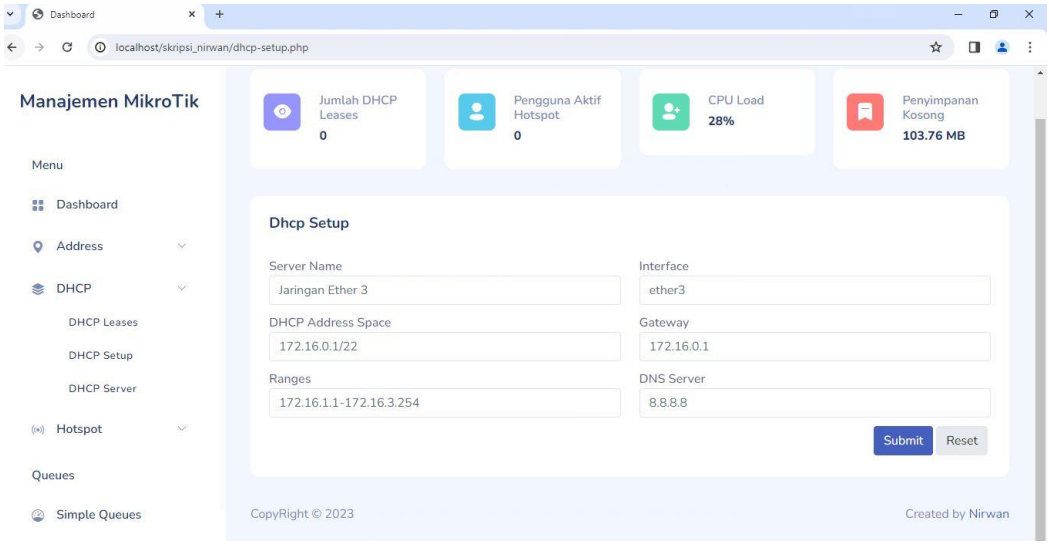
Gambar 4. 22 Penambahan Eth3 di Menu *Address list*

Pada **Gambar 4.22** diatas selanjutnya tambahkan eth3 di menu *Address list* kemudian klik tombol tambah.



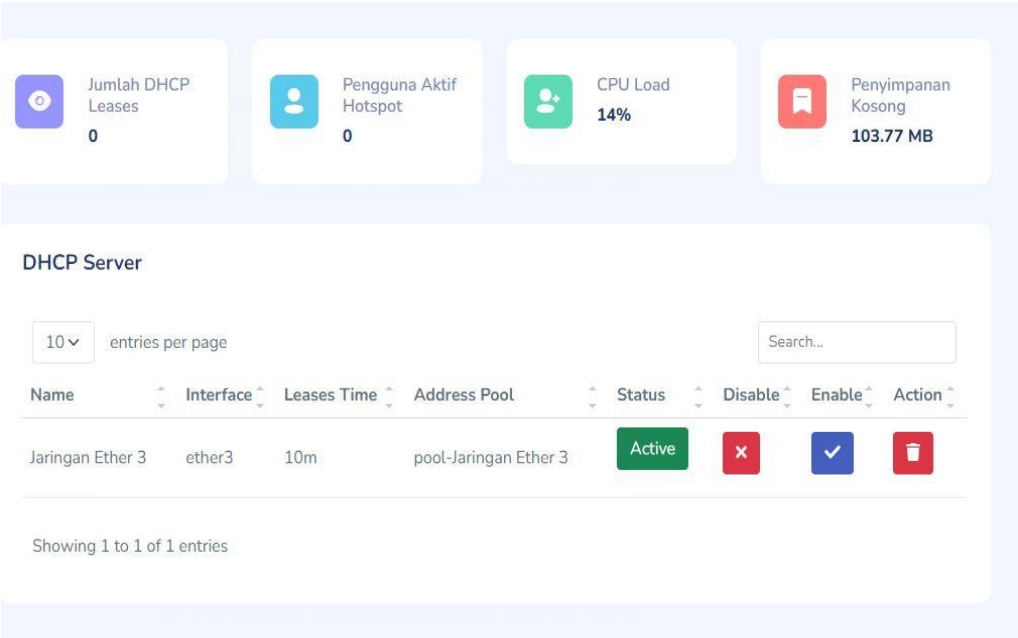
Gambar 4. 23 Memberikan *Ip Address* Pada Ether3

pada **Gambar 4.23** di atas adalah menambahkan *Ip Address* untuk *ether* 3 kemudian *submit*.



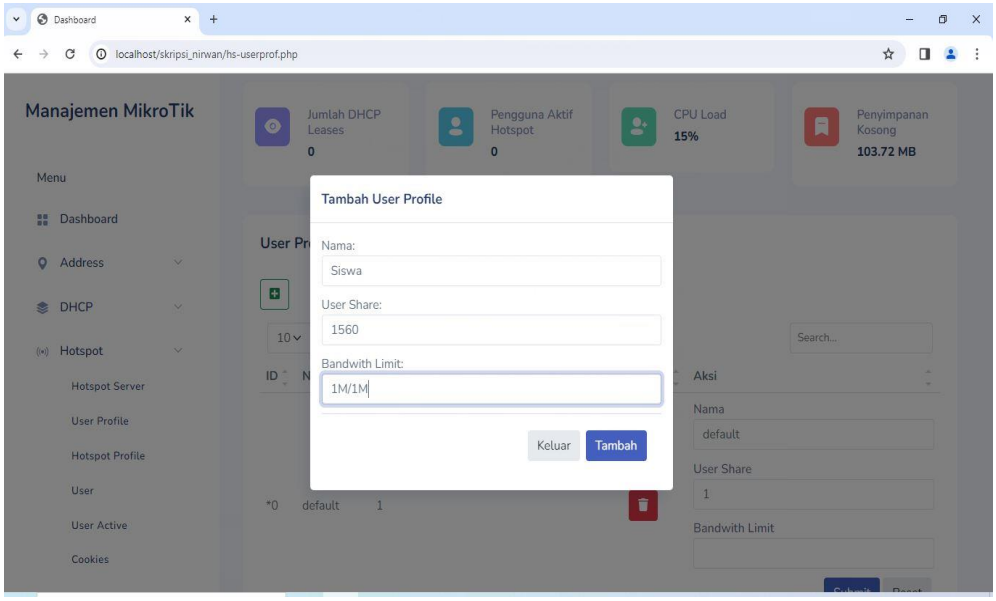
Gambar 4. 24 Buat *Dhcp* Untuk *Client* Mendapatkan *Ip* Secara Otomatis Dari Port Ether3

selanjutnya pada **Gambar 4.24** di atas untuk Buat dhcp di menu dhcp setup untuk membuat *client* mendapatkan *Ip* secara otomatis dari *port ethernet* 3 dari *Mikrotik.*



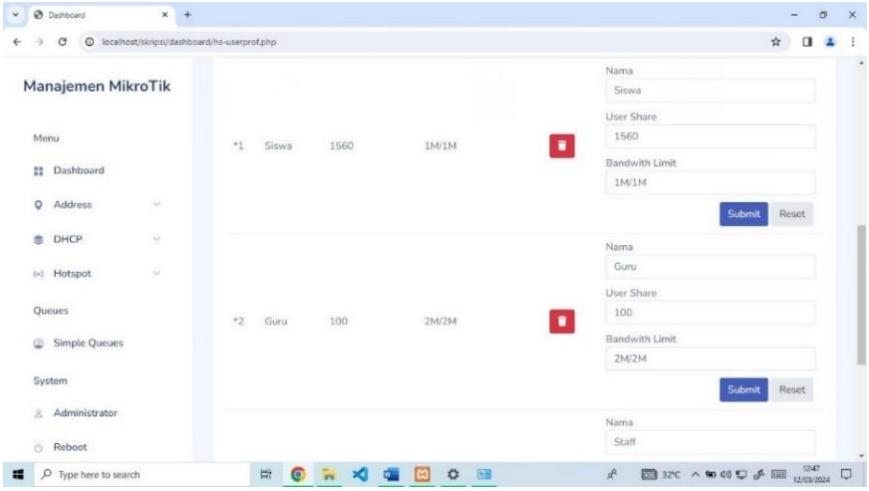
Gambar 4. 25 DHCP Server Untuk Ether3 Status Aktif

pada **Gambar 4.25** jaringan dhcp server untuk melihat satatus *ether* 3 aktif.

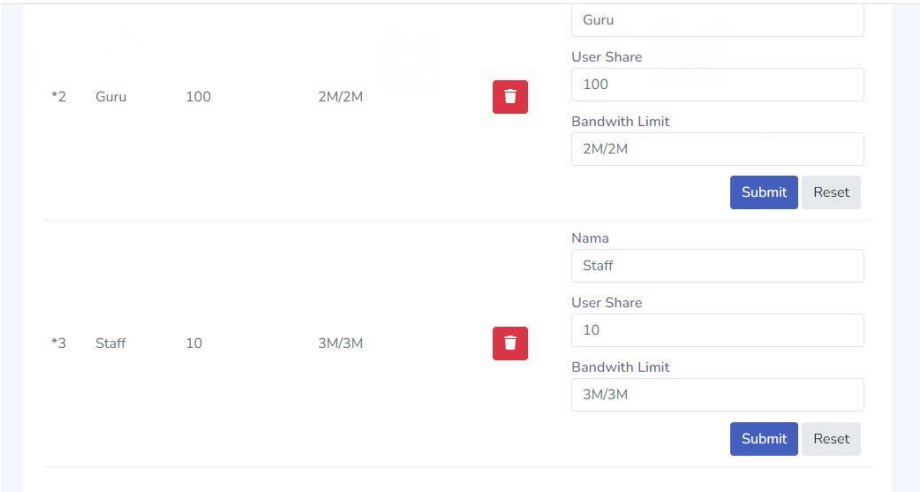


Gambar 4. 26 Tambah *User Profile* di *Hotspot* Siswa 10Mbps

Pada **Gambar 4.26** berikut untuk Tambah *User Profile* di *hotspot* siswa 10Mbps, guru 10Mbps dan staf 30Mbps.



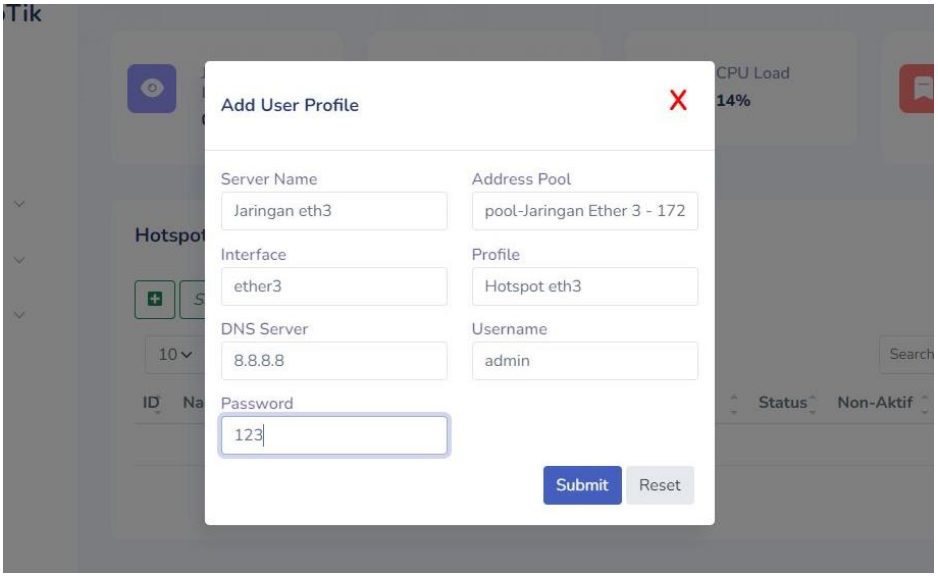
Gambar 4. 27 *User Profile* Di *Hotspot* Siswa 10Mbps.



Gambar 4. 28 *User Profile* Di *Hotspot*, Guru 10Mbps Dan Staf 30Mbps.

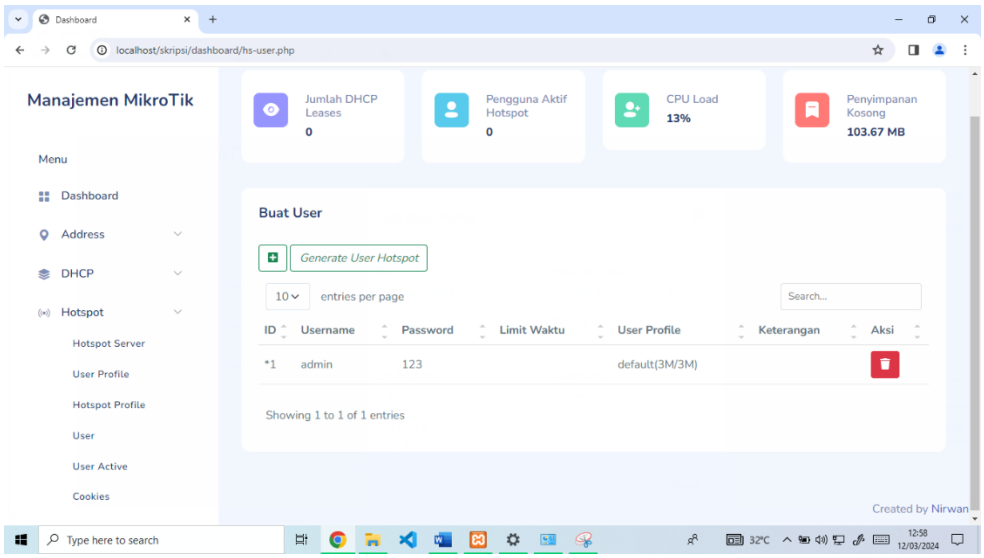
Gambar 4. 29 *Hotspot* Kosongkan *Bandwith* Limit Jaringan Agar Jaringan Dari *Bandwith* Terpakai Semua.

Pada **Gambar 4.29** Membuat *hotspot profil* di menu *Hotspot* kosongkan *bandwith limit* jaringan agar jaringan dari *bandwith* terpakai semua.



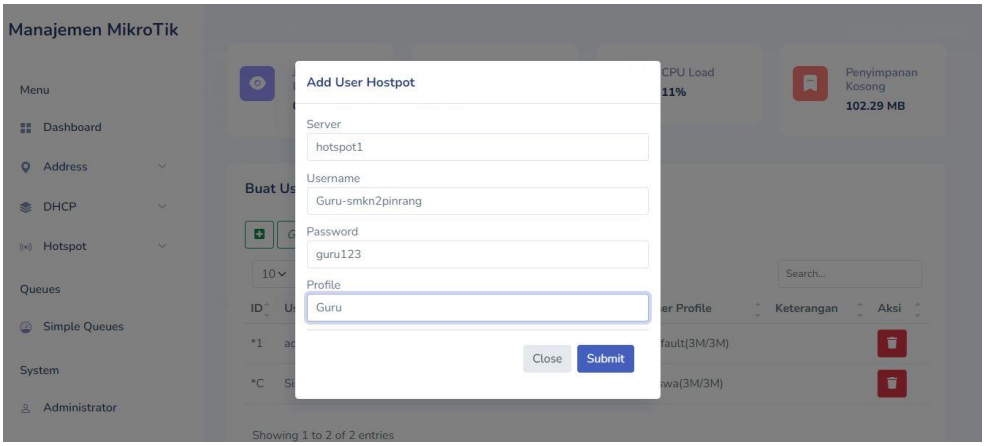
Gambar 4. 30 Buat *Hotspot* Di Menu *Hotspot* Server

Selanjutnya pada **Gambar 4.30** untuk membuat *hotspot* di menu *hotspot* server.

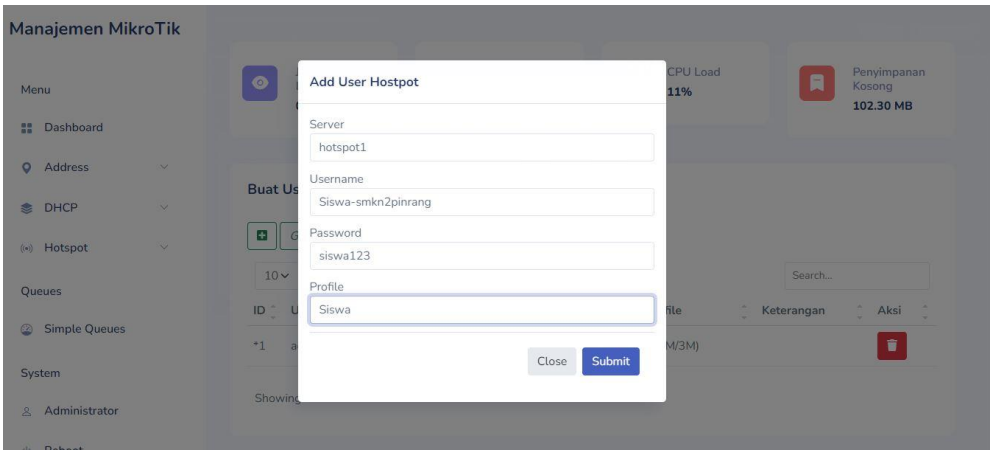


Gambar 4. 31 Membuat *User Login* Untuk *Client*

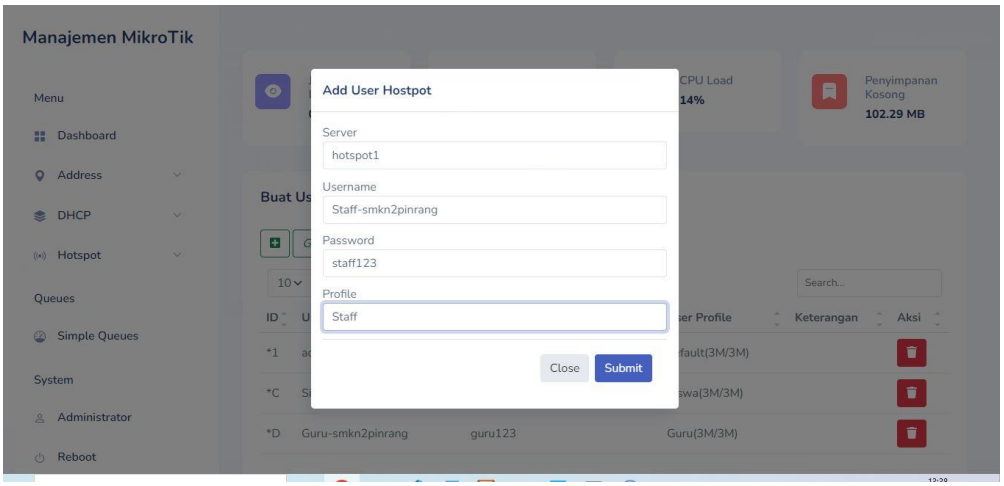
Gambar 4.31 membuat *User login* untuk *client* agar bisa terkoneksi dengan internet di *hotspot*, submenu *User*.



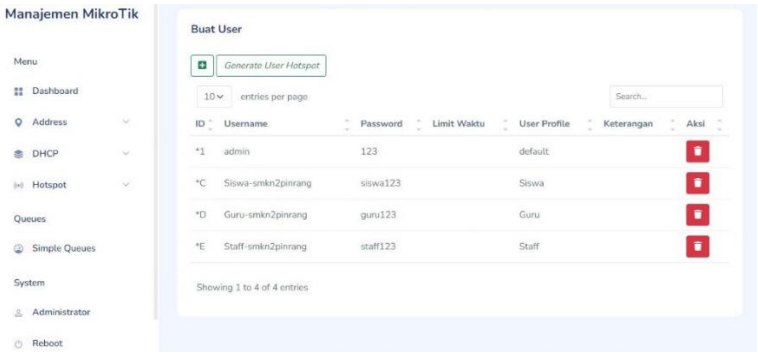
Gambar 4. 32 Menambahkan *User Hotspot* Untuk Guru



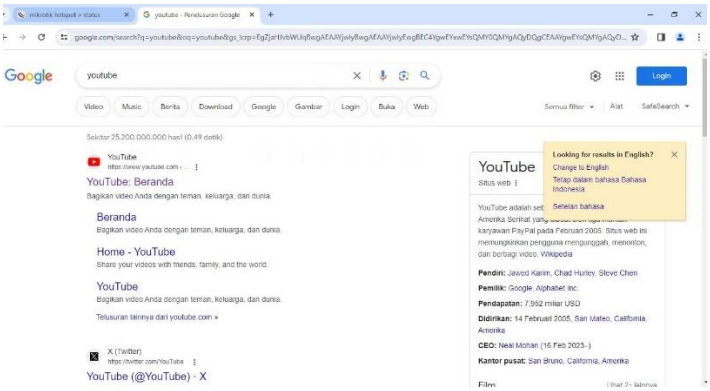
Gambar 4. 33 Menambahkan *User Hotspot* untuk Siswa



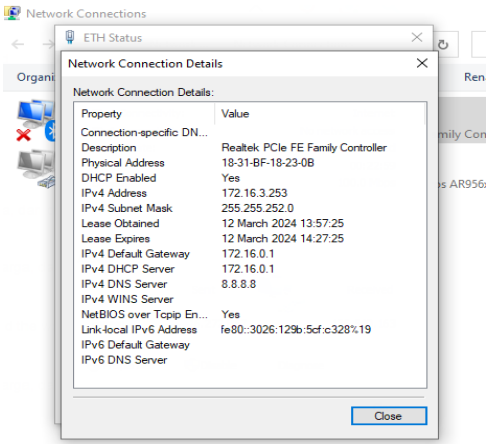
Gambar 4. 34 Menambahkan Akun *User Hotspot* untuk *Staff*



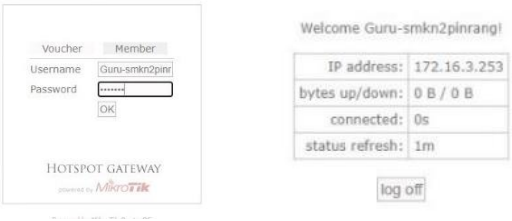
Gambar 4. 35 *User Hotspot* yang berhasil ditambahkan



Gambar 4. 36 Hasil Pengujian Jaringan Internet



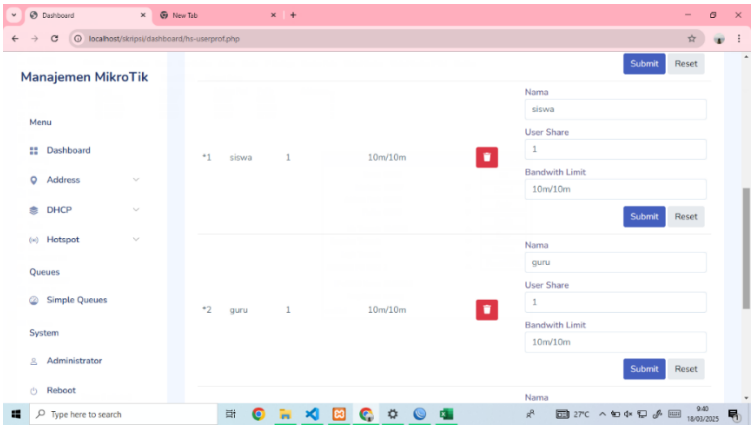
Gambar 4. 37 *Ip* DHCP Yang Di dapatkan Dari *Ethernet* 3



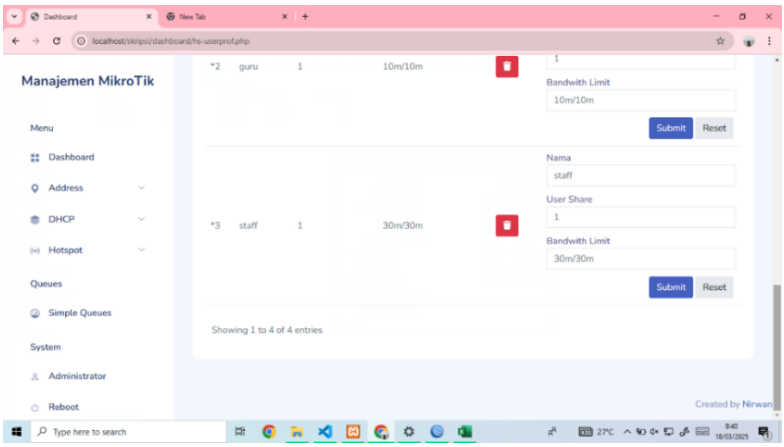
Gambar 4. 38 *Login* ke Jaringan Menggunakan *User* Guru

Gambar 4. 39 *Login* ke Jaringan Menggunakan *User* Siswa

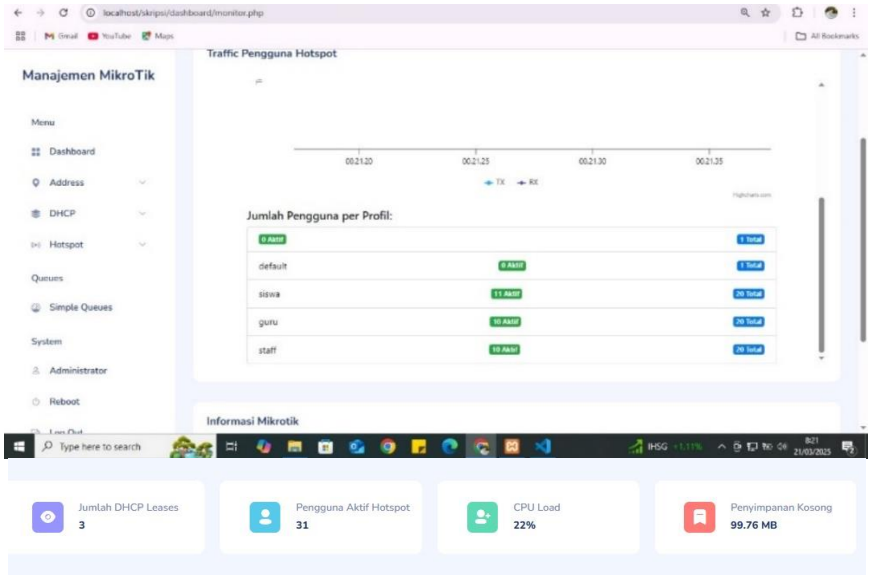
Gambar 4. 40 *Login* ke Jaringan Menggunakan *User Staff*



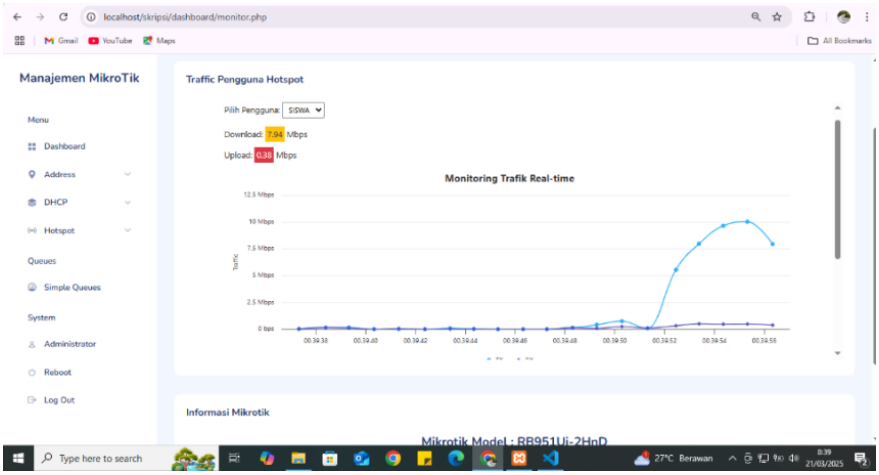
Gambar 4. 41 Pembagian jaringan *user profile* siswa 10 mbps dan guru 10 mbps



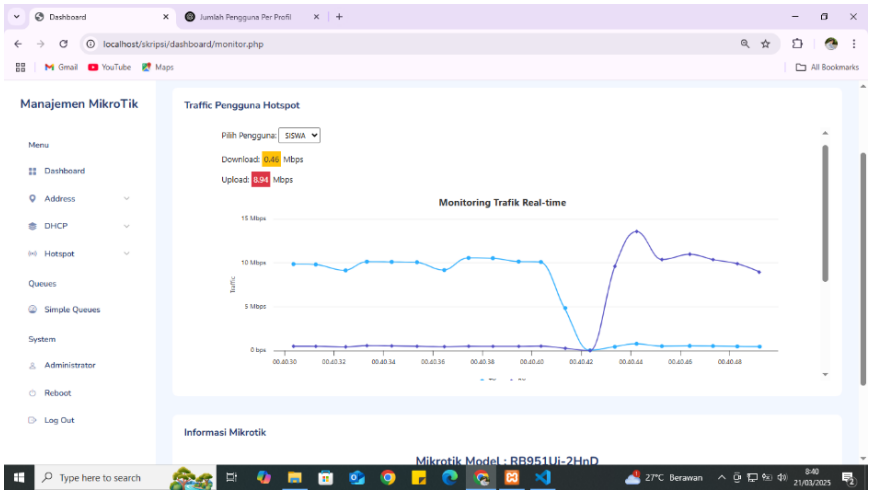
Gambar 4. 42 Pembagian jaringan *user profile* Staf 30 mbps



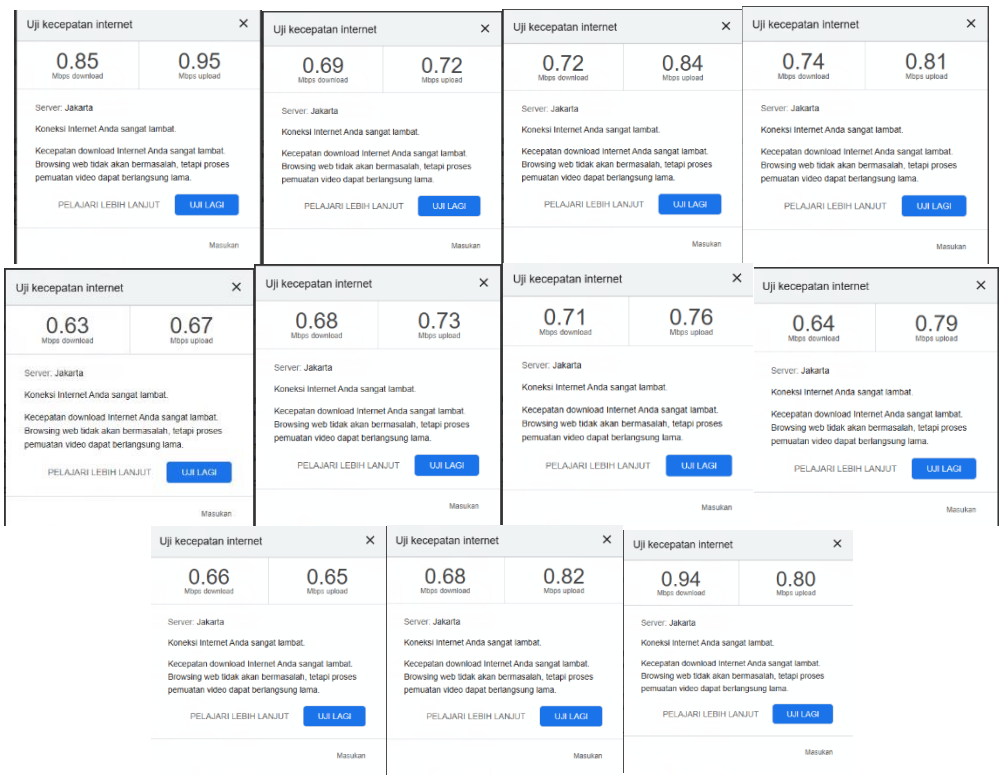
Gambar 4. 43 Jumlah Pengguna aktif Perprofile



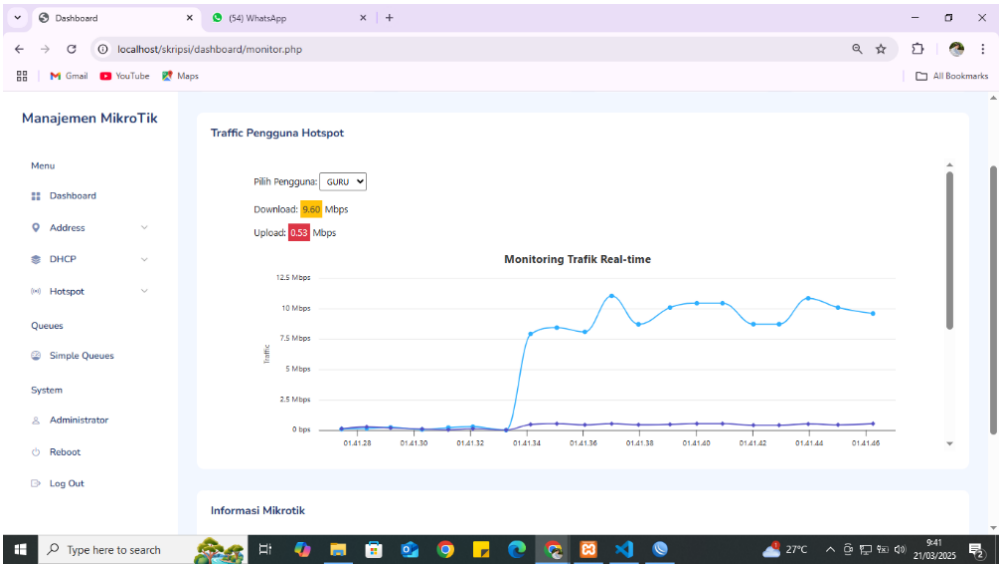
Gambar 4. 44 Hasil Pengujian Kecepatan *Download* Jaringan *User* Siswa



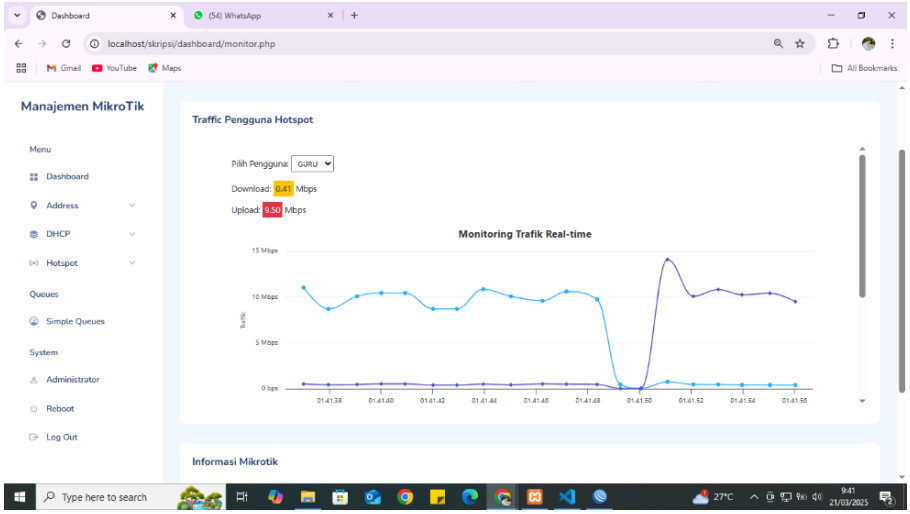
Gambar 4. 45 Hasil Pengujian Kecepatan *Upload* Jaringan pengguna Siswa



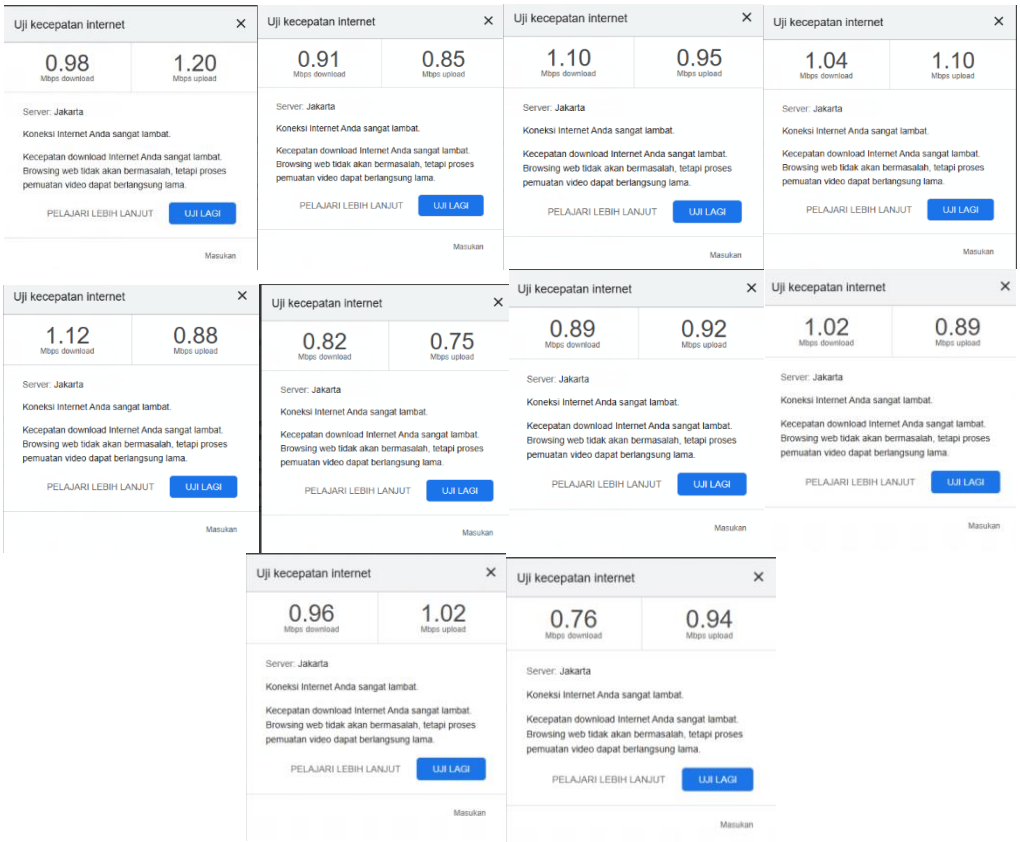
Gambar 4. 46 Hasil Uji Kecepatan jaringan pada *Client* 11 Siswa



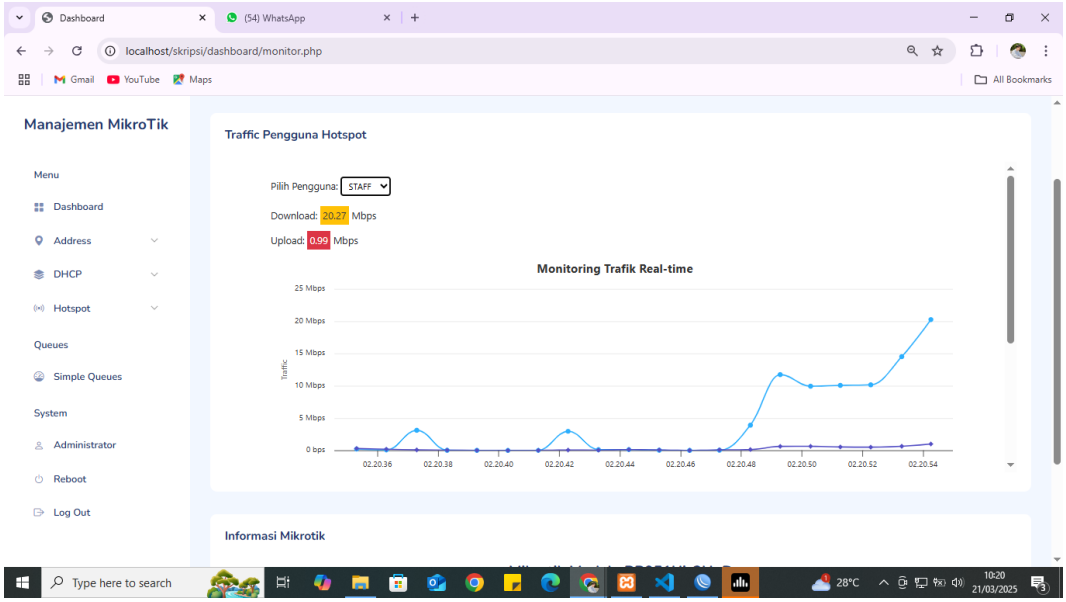
Gambar 4. 47 Hasil Pengujian Kecepatan *Download* Jaringan pengguna Guru



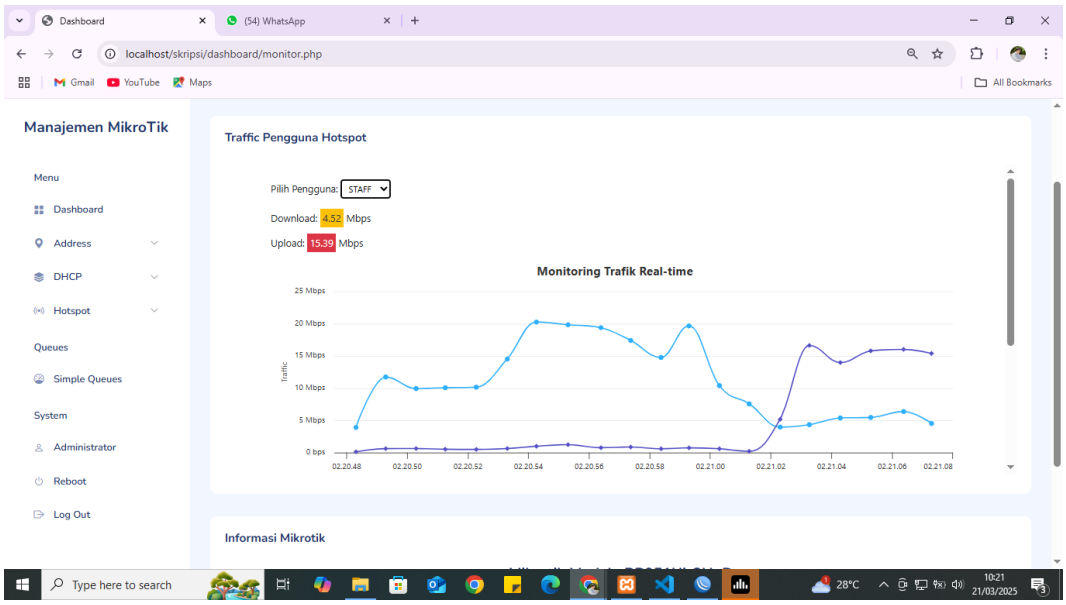
Gambar 4. 48 Hasil Pengujian Kecepatan *Upload* Jaringan pengguna Guru



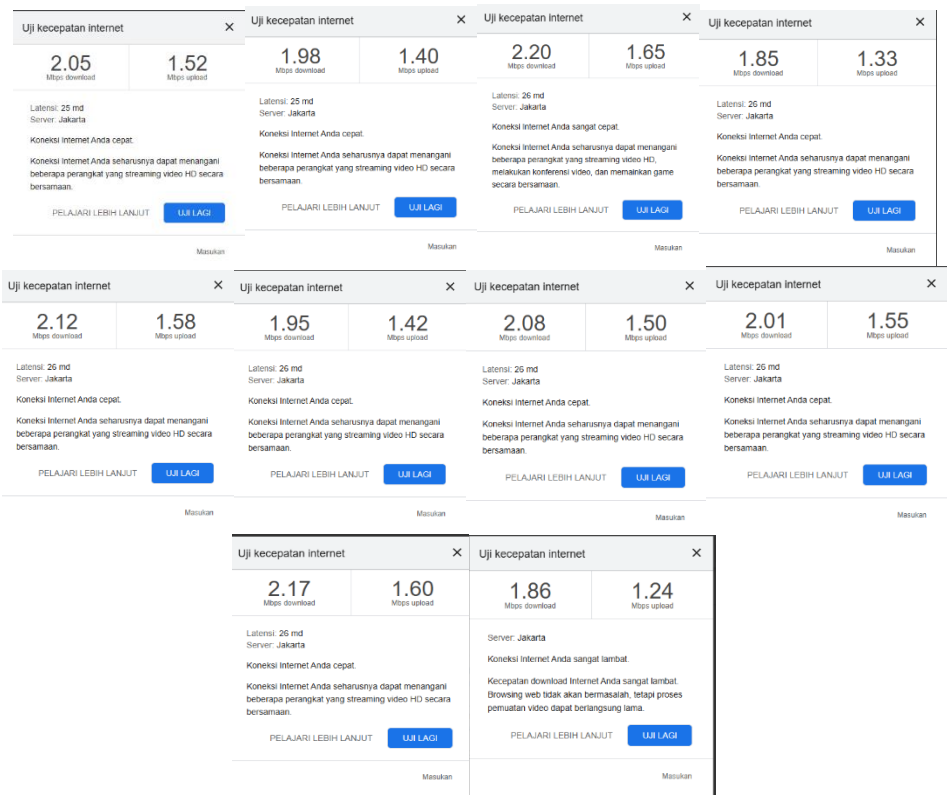
Gambar 4. 49 Hasil Uji Kecepatan jaringan pada *Client* 10 Guru



Gambar 4. 50 Hasil Pengujian Kecepatan *Download* Jaringan pengguna Staf



Gambar 4. 51 Hasil Pengujian Kecepatan *Upload* Jaringan pengguna Staf



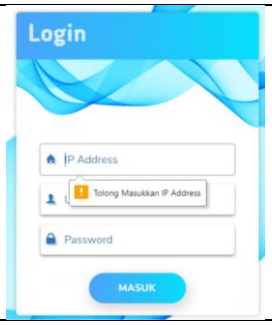
Gambar 4. 52 Hasil Uji Kecepatan jaringan pada *Client* 10 Staf

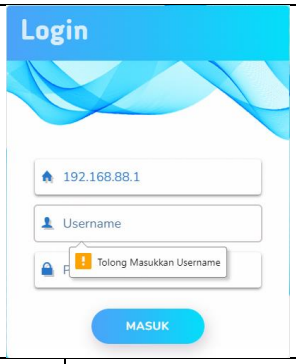
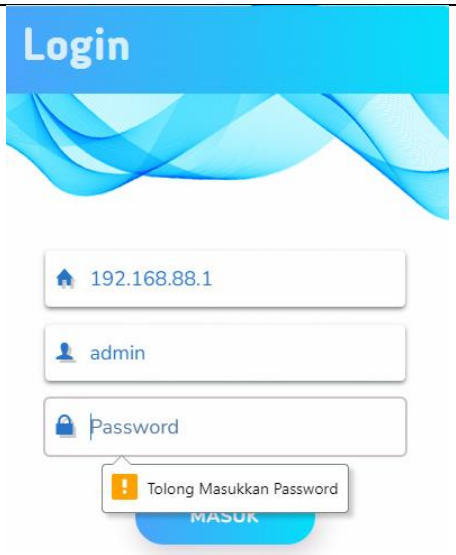
F. Pengujian Sistem

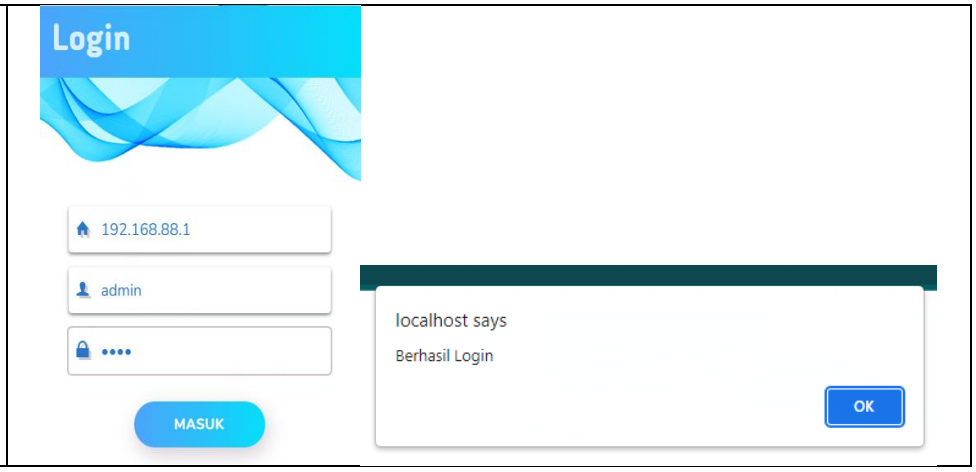
1. *Black Box*

Pengujian sistem dilakukan dengan pengujian *Black Box*.

Tabel 4. 1 *Black Box* Halaman *Login*

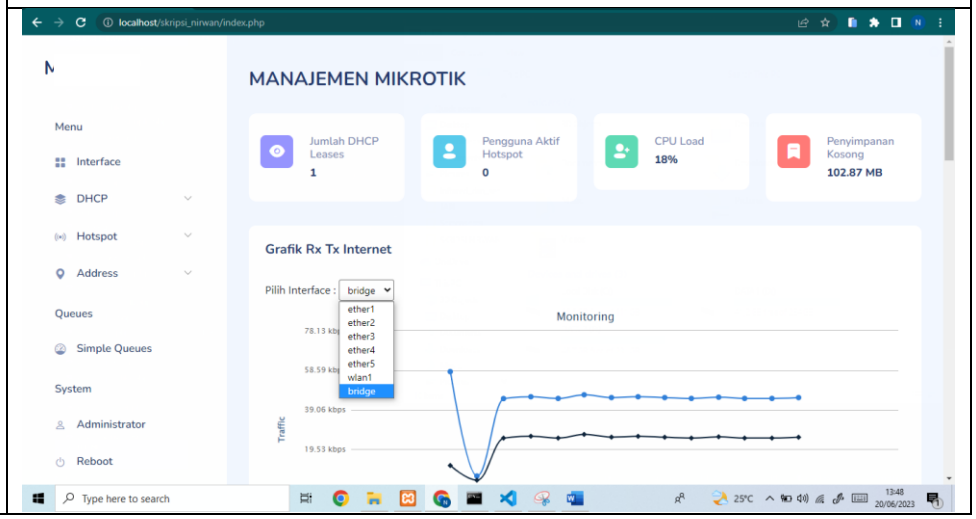
No.	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
1	Mengosongkan Semua isian data *Login* Admin langsung Klik Tombol masuk	Sistem akan menolak dan memberi pasan "tolong masukkan *Ip Address*"	Valid
	Screen Shot		
			

2	Hanya Mengisi *Ip Address* dan mengosongkan *Username* dan *Password* langsung Klik Tombol masuk	Sistem akan menolak dan memberi pasan "tolong masukkan *User*name"	Valid
	Screen Shot		
			
No.	**Skenario Pengujian**	**Hasil**	**Kesimpulan**
3	Hanya Mengisi *Username* dan mengosongkan *Ip Address* dan *Password* langsung Klik Tombol masuk	Sistem akan menolak dan memberi pasan "Tolong masukkan Password"	Valid
	Screen Shot		
			
No.	**Skenario Pengujian**	**Hasil**	**Kesimpulan**
4	Mengisi *Ip Address*, *User*name dan Password dan klik tombol masuk	Sistem Akan menerima dan mengirimkan pesan "Berhasil *Login*"	Valid
	Screen Shot		



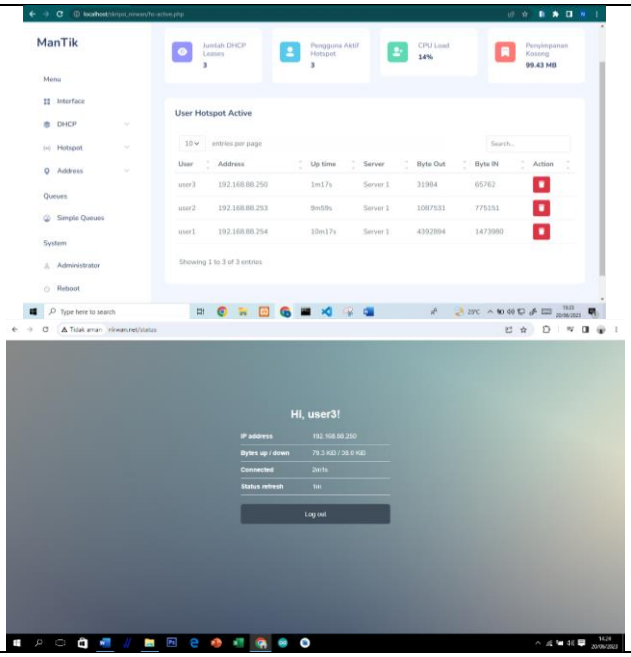
Tabel 4. 2 *Black Box Menu Interface*

No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Admin Menjalankan sistem pertama Kali	Halaman Utama tampil	valid
	Screen Shot		

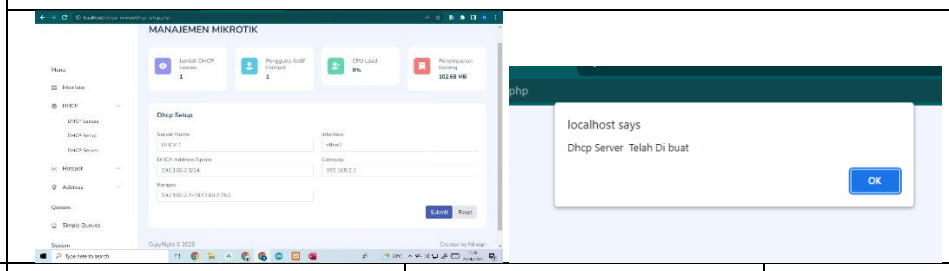
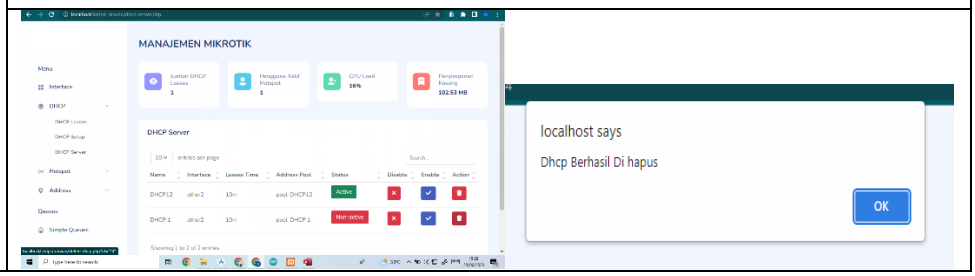


Tabel 4. 3 *Black Box* penambahan *User hotspot*

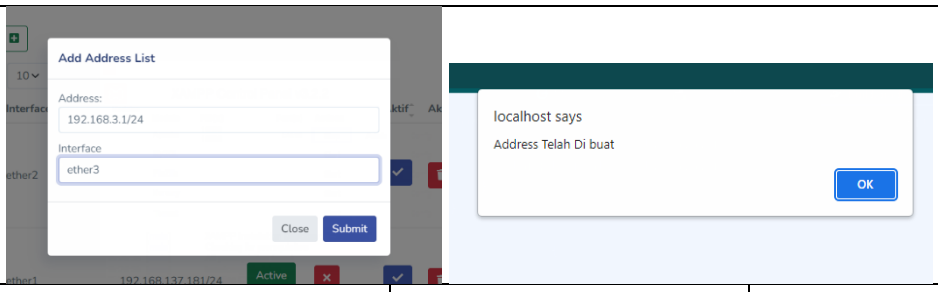
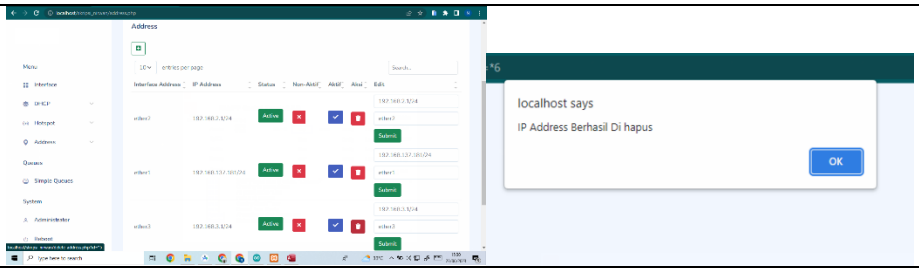
No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Penambahan pengguna *Hotspot*	Data pengguna *hotspot* tampil	Valid
	ScreenShot		



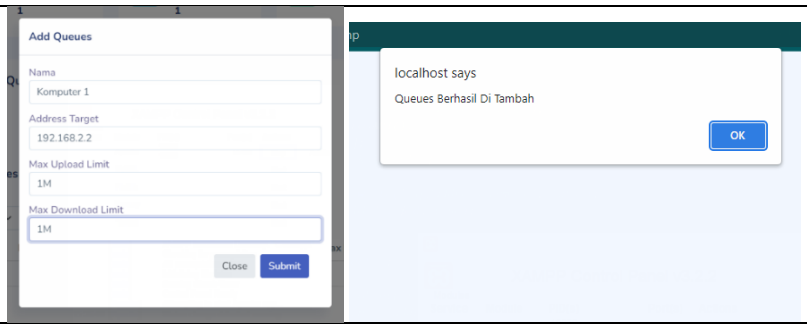
Tabel 4. 4 *Black Box* DHCP

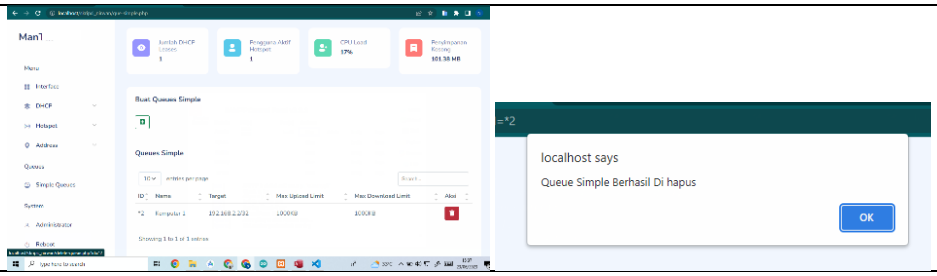
No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Penambahan DHCP *client*	Data tampil muncul pesan "Dhcp Server Telah Di Buat"	
	Screen Shot		
			
2	Menghapus data DHCP	Data terhapus dan muncul pesan "DHCP Berhasil Dihapus"	Valid
	Screen Shot		
			

Tabel 4. 5 *Black Box Address*

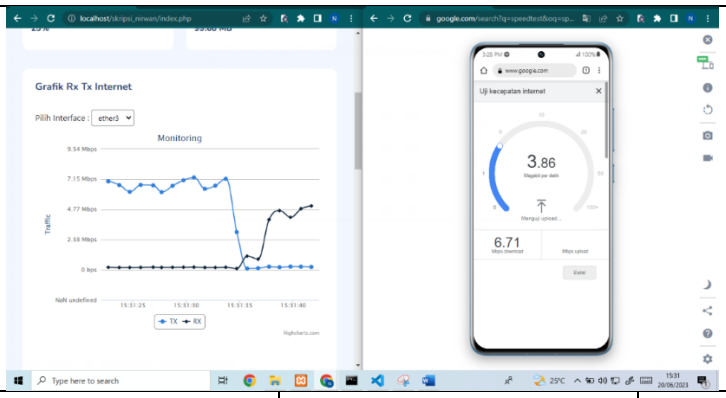
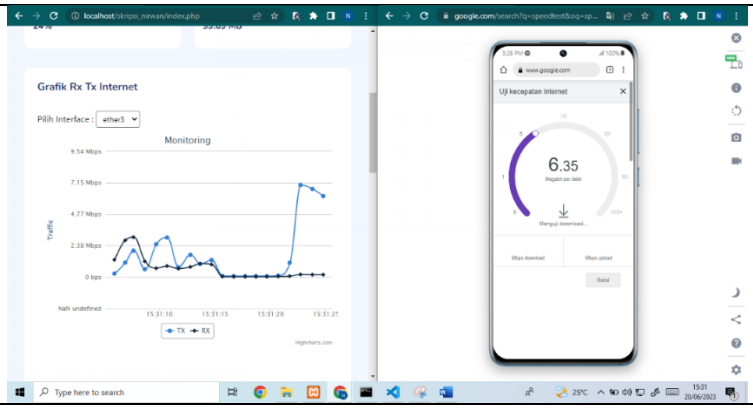
No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Penambahan data *Address*	Data tampil dan muncul pesan "*Address* telah dibuat"	Valid
	Screen Shot		
			
No.	**Skenario Pengujian**	**Hasil Yang diharapkan**	**Kesimpulan**
2	Menghapus data *Address*	Data terhapus dan muncul pesan "*Ip Address* berhasil dihapus"	Valid
	Screen Shot		
			

Tabel 4. 6 *Black Box Simple Queues*

No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Penambahan *Queues*	data tampil, muncul pesan "*Queues* berhasil ditambah"	Valid
	Screen Shot		
			

2	Menghapus data *Queues*	Data terhapus dan muncul pesan "*Queues Simple* berhasil dihapus"	valid
	Screen Shot		
			

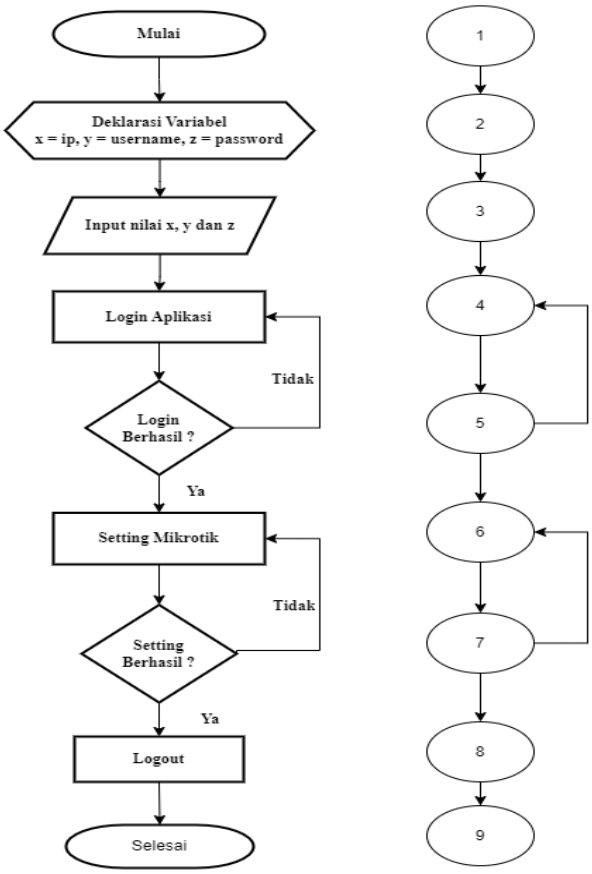
Tabel 4. 7 *Black Box* perbandingan *Bandwidth*

No.	Skenario Pengujian	Hasil Yang diharapkan	Kesimpulan
1	Perbandingan *Bandwidth Rx = Upload*	Perbandingan *Bandwidth* Rx *Upload* Sama	Valid
	Screen Shot		
			
2	Perbandingan *Bandwidth* Tx *Download*	Perbandingan *Bandwidth* Tx *Download* Sama	
	Screen Shot		
			

2. *White Box*

Pengujian aplikasi dilakukan dengan cara pengujian *White Box*:

a. *Flowchart* dan *flowgraph* secara keseluruhan



Gambar 4. 53 *Flowchart & Flowgraph*

Pada gambar 4.18 *Flowchar*t dan *flowgraph* untuk pengujian perangkat lunak, *Flowgraph* diatas dapat dilakukan perhitungan sebagai berikut:

1. Menghitung *Cyclomatic Complexity* V(G) dari *Edge* dan *Node*:

   Dengan Rumus : V (G) = E – N + 2

   N (*Node*) = 9

   E (*Edge*) = 8

P (*Predikat node*) = 2

Penyelesaian : V(G) = E – N + 2

= 8 – 9+ 2

= 1

Predikat Node (N) = P + 1

= 2 + 1

= 3

2. Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 3
3. *Independent Path* pada *Flowgraph* diatas adalah :

   Path 1 = 1-2-3-4-5-6-7-8-9

   Path 2 = 1-2-3-4-5-4-5-6-7-6-8-9

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan Hasil penelitian dan pengujian yang telah dilakukan penulis dalam pembuatan Aplikasi Manajemen *Bandwidth* Dan *Internet Gateway Mikrotik* Berbasis *Web* :

1. Pada penelitian ini menggunakan jaringan internet untuk melakukan pembagian jaringan menggunakan *Mikrotik* melalui aplikasi ini. Setelah penelitian ini memperoleh hasil untuk memanajemen *Mikrotik* mendapatkan hasil perbandingan yang sesuai pada aplikasi *speed test* sebagai perbandingan *Bandwidth*. aplikasi manajemen melakukan pengelolaan jaringan atau mengkonfigurasi jaringan *Mikrotik*.
2. Pada pengujian pengaturan menggunakan aplikasi yang telah dibuat, dapat pengaturan *Bandwidth* dapat dilakukan dengan akurat, dan terukur, sesuai dengan kebutuhan dan kebijakan jaringan yang telah ditetapkan. Sebelum penerapan manajemen *bandwidth*, kecepatan *download* adalah 43.11 Mbps dan *upload* 20.32 Mbps. kecepatan yang sudah ditentukan untuk guru 10Mbps, Siswa 10 Mbps dan Staf 30 Mbps. Setelah diterapkan, kecepatan untuk guru menjadi 9.60 Mbps *download* dan 9.50 Mbps *upload*, untuk Staf 20.27 Mbps *download* dan *upload* 15.39 Mbps, serta untuk siswa 7.94 Mbps *download* dan 8.94 Mbps *upload* dari pengujian 31 pengguna yang terhubung Ini menunjukkan pembagian *bandwidth* sesuai rancangan.

Secara keseluruhan, manajemen *bandwidth* berhasil mengatur penggunaan jaringan sesuai kapasitas yang telah ditentukan.

B. Saran

Penulis masih merasa banyak hal yang dapat dikembangkan. Penulis berharap, untuk peneliti selanjutnya dapat mengembangkan aplikasi ini atau membuat aplikasi yang berbeda dan mendapatkan hasil yang lebih akurat. Diharapkan, penulis selanjutnya bisa mengembangkan aplikasi sejenisnya yang dapat mengelola jaringan internet, manajemen dan *monitoring Mikrotik* dapat menggunakan metode yang berbeda untuk manajemen jaringan *Mikrotik.* Demikian saran yang dapat penulis berikan, semoga saran tersebut dapat dipertimbangkan sebagai bahan masukan untuk pembaca.

DAFTAR PUSTAKA

Amarudin. (2018). Analisis Dan Implementasi Keamanan Jaringan Pada Mikrotik Router Menggunakan Metode Port Knocking. *Seminar Nasional Sains Dan Teknologi 2018*, 1–7.

Anwar, K., Fairuzzabadi, M., & Sari, M. W. (2017). Rancang Bangun Sistem Manajemen User Hotspot Menggunakan Mikrotik Php Api Berbasis Web Di Pondok. *Dinamika Informasi*, *1*(1), 256–260.

Harahap, S. H. (2017). Pemanfaatan Aplikasi Penggambar Diagram Alir (Flowchart) sebagai Bahan Ajar untuk Mata Kuliah Sistem Akuntansi di Fakultas Ekonomi pada Perguruan Tinggi Swasta di Kota Medan. *Kitabah*, *I*, 14.

Hastuty, A. (2021). Dasar Pemrograman , Cetakan 2 (revisi), Makassar : CV Bangun Bumitama, Hal 18-23.

Khamaeni, M. G. AL. (2023). Implementasi White Box Testing Berbasis Path Pada Aplikasi Berbasis Web. *Jurnal Siliwangi*, *9*(1), 8–13.

Muhammad M, & Hasan I. (2016). Analisa Dan Pengembangan Jaringan Wireless Berbasis Mikrotik Router Os V.5.20 Di Sekolah Dasar Negeri 24 Palu. *Stmik-Binamulia.Ac.Id*, *2*(1), 10–19.

Rafli, H., Zen, R., Nuryasin, I., Informatika, P., Malang, U. M., Raya, J., No, T., 246, K., & Lowokwaru, K. M. (2024). JOISIE licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0) PENERAPAN WHITEBOX TESTING PADA PENGUJIAN SISTEM MENGGUNAKAN TEKNIK BASIS PATH. *Journal Of Information Systems And Informatics Engineering*, *8*(1), 101–111. https://doi.org/10.35145/joisie.v8i1.4229

Santoso, H. (2012). Strategi Memilih Internet Service Provider Terbaik untuk Perguruan Tinggi (Studi Kasus: STMIK ATMA LUHUR). *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, *2012*(Snati), 1.

Suarman, H., Triyono, J., & K, R. Y. R. (2017). *Jurnal JARKOM Vol . 3 No . 2 Desember 2017 MANAJEMEN Free RADIUS HOTSPOT MENGGUNAKAN MIKROTIK API DAN PHP (Studi Kasus Uwong Coffee) Jurnal JARKOM Vol . 3 No . 2 Desember 2017*. *3*(2), 35–42.

Sujalwo, Bana Handaga, & Heru Supriyono. (2020). Manajemen Jaringan Komputer dengan Menggunakan Mikrotik Router. *Jurnal Komunikasi Dan Teknologi Informasi*, *2*(2), 32–43.

Tasanah Assakur, Y. H., Fahrudin, M. S., & Ferdiansyah, F. (2020). Implementasi API Mikrotik untuk Management Router Berbasis Android (Studi Kasus: PT Sigma Adi Perkasa). *Jurnal Sains Dan Informatika*, *6*(1), 92–101. https://doi.org/10.34128/jsi.v6i1.217