

BAB II

TINJAUAN PUSTAKA

2.1 Metode Pengembangan Jaringan NDLC

Network Development Life Cycle (NDLC) berasal dari *Systems Development Life Cycle (SDLC)* yang merupakan teknik analisis terstruktur yang digunakan untuk merencanakan dan mengelola proses pengembangan sistem. Meskipun belum ada standar yang ditetapkan untuk NDLC yang secara universal disepakati. NDLC memiliki enam tahap yaitu analisis, desain, simulasi, implementasi, pemantauan dan manajemen.(Goldman, 2004)

1. Analisis

Tahap awal dilakukan analisa kebutuhan, analisa permasalahan yang muncul, analisa keinginan user, dan analisa topologi / jaringan yang sudah ada saat ini.

2. Design

Dari data-data yang didapatkan sebelumnya, tahap design ini akan membuat gambar desain topologi jaringan interkoneksi yang akan dibangun. Diharapkan dengan gambar ini akan memberikan gambaran seutuhnya dari kebutuhan yang ada. Desain bisa berupa desain struktur topologi, desain akses data, desain layout perkabelan, dan sebagainya yang akan memberikan gambaran jelas tentang proyek yang akan dibangun.

3. Simulasi

Beberapa penulis akan membuat dalam bentuk simulasi dengan bantuan tools khusus di bidang *network* seperti *Boson*, *Packet Tracert*, *Netsim*, dan sebagainya. Hal ini dimaksudkan untuk melihat kinerja awal dari jaringan yang akan dibangun dan sebagai bahan presentasi dan sharing dengan *teamwork* lainnya.

4. Implementasi

Pada tahapan ini akan memakan waktu lebih lama dari tahapan sebelumnya. Dalam implementasi pekerja jaringan akan menerapkan semua yang telah direncanakan dan didesain sebelumnya.

5. Pemantauan

Setelah tahap implementasi tahapan memonitoring merupakan tahapan yang penting, agar jaringan komputer dan komunikasi berjalan dengan keingin dan tujuan awal dari user pada tahap awal analisis, maka perlu melakukan tahapan kegiatan memonitoring.

6. Manajemen

Pada manajemen atau pengaturan salah satu menjadi perhatian khusus adalah *policy*, kebijakan perlu dibuat untuk membuat atau mengatur agar sistem yang dibangun dan berjalan dengan baik dapat berlangsung lama dan unsur reliabiliti terjaga.

2.2 Penelitian Terdahulu

Penelitian terdahulu ini menjadi satu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Pada penelitian sebelumnya yang dilakukan (Mukri dan Dasmen, 2019) dalam jurnal yang berjudul **“Prototipe Manajemen *Bandwidth* pada Jaringan Internet Hotel Harvani dengan Mikrotik RB 750r2”** Permasalahan dari penelitian ini yaitu pada hotel Harvani pembagian *bandwidth* kepada setiap user yang tidak merata, hal ini menyebabkan hanya user yang pertama kali terhubung jaringan *internet* yang memperoleh *bandwidth* yang besar serta merasakan kecepatan akses *internet* yang cukup tinggi. Disisi lain hal ini memberi dampak ketidaknyamanan terhadap *user* lainnya dalam mengakses jaringan *internet*, karena *user* yang lain memperoleh *bandwidth* yang jauh lebih kecil sehingga kecepatan akses *internet*-nya juga lambat.

Pada penelitian sebelumnya yang dilakukan (Asnawi, 2017) dalam jurnal yang berjudul **“Aplikasi Konfigurasi Mikrotik Sebagai Manajemen *Bandwidth* dan *Internet Gateway* Berbasis Web”** Permasalahan dari penelitian ini yaitu konfigurasi mikrotik dilakukan dengan melakukan console ke perangkat dengan

beberapa cara yaitu dengan *CLI (Command Line Interface)*, program *winbox*, dan *Web interface (Webbox)*. Namun walaupun sudah menggunakan *winbox* yang menggunakan *GUI* sebagai user interfacenya tetapi dalam prakteknya akan mengalami kesulitan dalam melakukan konfigurasi karena memerlukan beberapa langkah yang perlu diatur untuk dapat melakukan manajemen *bandwidth* dan *share* koneksi *internet*. Berdasarkan hal itu dibutuhkan sebuah aplikasi yang dapat mempermudah dalam mengkonfigurasi sebuah router mikrotik tanpa harus hafal dan paham dengan perintah - perintah *CLI* mikrotik tersebut yaitu menggunakan *API RouterOS* mikrotik. Sehingga diharap dapat menggunakannya untuk mengkonfigurasi *router* mikrotik sebagai manajemen *bandwidth client* dan *internet gateway* untuk keperluan *share* koneksi internet.

Pada penelitian sebelumnya yang dilakukan (Sanjaya dan Setiadi, 2019) dalam jurnal yang berjudul “**Network Development Life Cycle (NDLC) Dalam Perancangan Jaringan Komputer Pada Rumah Shalom Mahanaim**” Permasalahan dari penelitian ini adalah infrastruktur jaringan komputer di rumah shalom mahanaim masih memiliki beberapa kekurangan seperti tidak adanya pemblokiran website, autentikasi *login* dan pembagian *bandwidth* yang teratur. Untuk mengatasi permasalahan tersebut maka diperlukan perancangan jaringan tersebut dengan baik. Hasil dari penelitian ini berupa jaringan komputer rumah shalom mahanaim yang mengimplementasikan *Capartive Portal*, *Virtual Lan*, *Firewall Filtering*, *L7 protocol* dan *Bandwidth Management*.

Pada penelitian sebelumnya yang dilakukan (Jayanto, 2019) dalam jurnal yang berjudul “**Rancang Bangun Sistem Monitoring Jaringan menggunakan Mikrotik RouterOs**” Dalam penelitian ini, sebuah website menampilkan data berupa grafik memudahkan seorang *admin* untuk memantau sebuah jaringan, terdapat juga tampilan *log mikrotik*, *firewall*, *bandwidth* dan notifikasi melalui *email/pop-up* pada website menggunakan *API PHP mikrotik*.

Pada penelitian sebelumnya yang dilakukan (Afriansyah dan Ardhy, 2021) dalam jurnal yang berjudul “**Membangun Jaringan Wireless LAN dan Manajemen Bandwidth Menggunakan Metode Simple Queue pada Kantor Pekon Gunung Meraksa**” permasalahan dari penelitian ini yaitu Kantor Pekon

Gunung Meraksa belum menggunakan teknologi internet. Kebutuhan akan *wireless LAN* pada Kantor Pekon sangatlah penting sebab dalam proses pemindahan data harus dilakukan dengan waktu yang sangat cepat, akses internet sangat dibutuhkan dalam sistem administrasi antar instansi pemerintahan serta pemilihan metode *simple queue* pada *bandwidth* yang merupakan metode yang cukup sederhana dan mudah dipahami.

Sedangkan penelitian yang dilakukan penulis tidak jauh berbeda dengan penelitian sebelumnya yaitu memprioritaskan *bandwidth* pada aplikasi *zoom* menggunakan mikrotik dan *API* sehingga bisa digunakan untuk mengakses dan melakukan konfigurasi serta monitoring *traffic* pada mikrotik menggunakan *API website*. Fitur aplikasi ini meliputi *live traffic upload/download bandwidth* pada *zoom* dan penggunaan *browsing* dan konfigurasi yang diperlukan untuk memprioritaskan *bandwidth* pada *zoom*.

2.2.1 Referensi Jurnal Internasional

1. Jurnal “Mikrotik Hotspot Network Implementation Using Simple Queue As Bandwidth Management”

Jurnal internasional ini dibuat pada tahun 2020 oleh Saragih, dkk. Pada jurnal ini membahas mengenai suatu instansi yaitu “Departemen Perumahan Rakyat dan Permukiman” dimana jaringan internet yang digunakan dengan kecepatan akses 10Mbps belum dikelola secara *bandwidth*, sehingga penggunaan jaringan tersebut masih menjadi tarik ulur antar pengguna satu dengan lainnya. Yang mengakibatkan kecepatan internet tidak stabil karena tidak adanya *bandwidth* manajemen pada jaringan. Setelah dilakukan pengaturan, *max download 1 Mbps* dan *upload 64 Kbps*.

2. Jurnal “A study of network bandwidth management by using queue tree with per connection queue”

Jurnal internasional ini dibuat pada tahun 2019 oleh Smansub, dkk. Pada jurnal ini membahas tentang manajemen *bandwidth* jaringan dengan menggunakan *queue tree* dengan *per connection queue (PCQ)*. untuk menguji manajemen *bandwidth* jaringan dengan *download file testing* dari internet. Hasilnya menunjukkan kinerja fungsi *queue tree* dan fungsi *PCQ* untuk manajemen

bandwidth. Semua pengguna dalam jaringan telah diterima dan *bandwidth* jaringan pemanfaatan yang sama yaitu sekitar 5 Mbps.

3. Jurnal “*Analysis Of Quality Of Service Bandwidth Management On Computer Network Using Mikrotik Rb951Ui-2Hnd*”

Jurnal internasional ini dibuat pada tahun 2021 oleh Indrajaya, dkk. Pada jurnal ini membahas tentang Penggunaan jaringan komputer saat ini mengalami peningkatan yang cukup signifikan akibat pandemi Covid-19, namun demikian masih banyak permasalahan dengan jaringan komputer, seperti yang sering terjadi adalah lambatnya jaringan komputer dalam mentransfer data yang disebabkan oleh keterbatasan *bandwidth* atau batas maksimum. Untuk itu diperlukan suatu metode manajemen *bandwidth* jaringan komputer yang disebut dengan metode *QoS* (*Quality of Service*) seperti *throughput*, *packetloss*, *delay* dan *jitter*.

4. Jurnal “*Impact of queue tree and layer 7 filtering combination to bandwidth management in 802.3 Network*”

Jurnal internasional ini dibuat pada tahun 2021 oleh Ichsan, dkk. Pada jurnal ini membahas tentang manajemen *bandwidth* di jaringan area lokal LAN 802.3 yang harus dijaga untuk mendukung kegiatan dengan melakukan *firewall L7 protocol*. Manajemen *bandwidth* mampu mengelola *bandwidth* unggah dan unduh, menurunkan lalu lintas yang tidak penting dan membatasi lalu lintas media sosial., metode gabungan *queue tree* menjamin *bandwidth* rata-rata 2 *mbps* untuk lalu lintas unggah / unduh dengan rata-rata *CIR* 250 *kbps* per *klien*. Pengujian *ICMP* menghasilkan *delay* rata-rata 20 ms yang termasuk dalam kategori sangat baik. Rata-rata *jitter* 18 ms dan *packet loss* adalah nol.

5. Jurnal “*Queue Tree Implementation for Bandwidth Management in Modern Campus Network Architecture*”

Jurnal internasional ini dibuat pada tahun 2018 oleh Perwira dan Liantoni. Pada jurnal ini membahas tentang penerapan *queue* pada kampusnya dengan menggunakan manajemen *bandwidth*. Metode yang digunakan dalam penelitian ini adalah dengan observasi langsung, wawancara untuk identifikasi masalah dan

menetapkan persyaratan layanan Internet dan ketersediaan sumber daya perangkat keras dan perangkat lunak. Hasilnya, tingkat kepuasan pengguna implementasi *Queue Tree* mencapai 59,2%.

2.3 Pengertian *Bandwidth*

Bandwidth merupakan suatu ukuran rentang frekuensi maksimum yang dapat mengalirkan data dari suatu tempat ke tempat lain dalam suatu waktu tertentu. (Rofiq, 2013).

Bandwidth adalah suatu ukuran dari banyaknya informasi atau data (*bit*) yang dapat dikirim dari suatu tempat ke tempat lain dalam satu detik (Ardiansya, Primananda dan Ichsan, 2017).

Jadi didefinisikan *bandwidth* sebagai lebar cakupan frekuensi yang dipakai oleh sinyal dalam medium transmisi yang dikirim dari suatu tempat ke tempat yang lain.

2.4 Jaringan

2.4.1 Pengertian Jaringan

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung. Informasi dan data bergerak melalui kabel-kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data, mencetak pada printer yang sama dan bersama sama menggunakan *hardware/software* yang terhubung dengan jaringan. Tiap komputer, printer atau *periferal* yang terhubung dengan jaringan disebut *node*. Jaringan komputer mempunyai jenis dan membedakannya yaitu berdasarkan jangkauan, media transmisi, dan fungsinya dan penjelasannya (Arta dan Nugraha, 2020).

Jaringan komputer adalah dua komputer dapat dikatakan saling terkoneksi dalam sebuah jaringan jika keduanya mempunyai kemampuan untuk saling berkomunikasi dan bertukar informasi (Putri, Fatoni dan Solikin, 2017)

Jadi jaringan merupakan kumpulan dari beberapa titik yang saling terhubung dan saling berkomunikasi. Media penghubung antar titik ini merupakan media transmisi yang dapat mengirimkan data dari satu titik ke titik yang lain.

2.4.2 Jenis - Jenis Jaringan

Pada jaringan komputer, terdapat jenis jaringan yang berbeda. diantaranya :

1. LAN (*Local Area Network*)

Jaringan LAN dibentuk oleh sekumpulan komputer atau *device* lainnya yang saling terkoneksi, dan semuanya berada pada satu lokasi yang sama yang tidak begitu luas. Jaringan komputer untuk warnet, sekolah, rumah, atau laboratorium biasanya menggunakan jaringan LAN, karena hanya pada satu area saja. Umumnya jaringan satu LAN terdiri dari beberapa komputer dengan maksimal 100 komputer, yang saling terhubung menggunakan media kabel. Umumnya jenis kabel yang digunakan adalah UTP/STP dan terdiri dari satu atau beberapa *switch* dengan teknologi berbasis IEEE 802.3 Ethernet. Namun, dalam perkembangannya, jaringan LAN dapat dibangun menggunakan Wi-Fi (*wireless fidelity*) yang menggunakan teknologi 802.11b. Jaringan LAN yang menggunakan media Wi-Fi disebut juga jaringan WLAN (*Wireless Local Area Network*) (Suprpto, 2020: 8).

2. MAN (*Metropolitan Area Network*)

Jaringan MAN (*Metropolitan Area Network*) dibangun dengan menghubungkan jaringan-jaringan LAN, sehingga komunikasi Jaringan MAN dapat mencakup area kota. Tujuan dari jaringan MAN adalah menghubungkan jaringan komputer yang ada di suatu kota menjadi sebuah jaringan yang lebih besar. Jangkauan yang dimiliki oleh Jaringan MAN dapat mencapai 50 KM. Jaringan MAN terbentuk dari Jaringan LAN yang sangat banyak yang saling terhubung. Jaringan MAN sangat cocok diterapkan di kampus yang mempunyai lokasi yang berbeda-beda, atau perusahaan yang mempunyai kantor cabang yang berbeda letaknya. Jaringan MAN juga dapat dibentuk dengan menggunakan media *unguided/wireless*, dan biasa disebut dengan WMAN (Suprpto, 2020: 9).

3. WAN (*Wide Area Network*)

Jaringan WAN adalah jaringan yang mempunyai lingkup area yang sangat luas. Jaringan WAN dapat menghubungkan antar kota, bahkan negara dan benua. Jaringan WAN terdiri dari jaringan-jaringan berbeda yang saling terhubung,

sehingga untuk membangun jaringan WAN membutuhkan *intermediate device*, yaitu router. WAN juga membutuhkan *device* lain seperti: Modems, CSU/DSU, *Access Server*, *Communication Server*, *WAN Switch*, dan *Core Router*. Sedangkan protocol yang digunakan oleh jaringan WAN adalah: HDLC (*High Level Data Link Protocol*), PPP (*Point to Point Protocol*), X.25 *Protocol*, *Frame Relay*, ISDN (*Integrated Services Digital Network*) (Suprpto, 2020: 12)

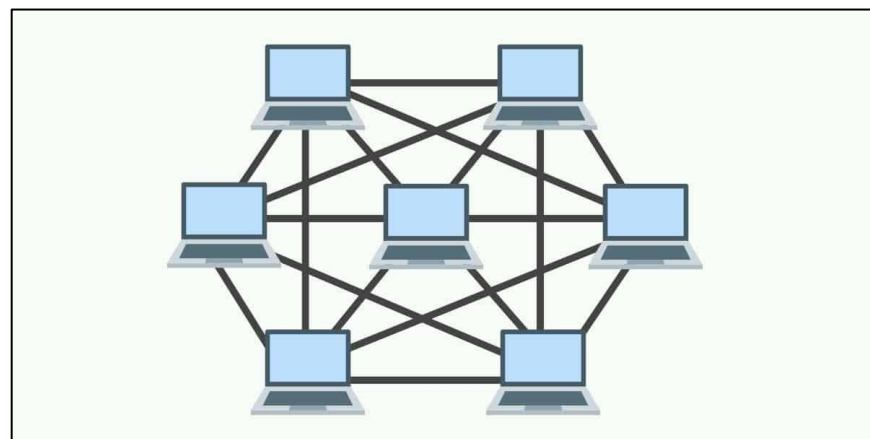
2.4.3 Topologi Jaringan

Topologi adalah rancangan yang dibuat untuk menghubungkan komputer satu dengan komputer lainnya, sehingga membentuk sebuah jaringan komputer.

2.4.4 Jenis - Jenis Topologi Jaringan

1. Topologi Star

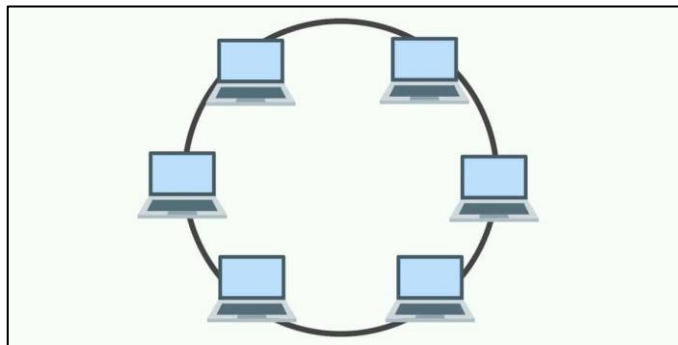
Topologi *star* atau disebut topologi bintang adalah topologi yang menghubungkan setiap *node* atau *workstation* dengan menggunakan kabel ke konsentrator jaringan, yaitu *hub* atau *switch*. Jumlah *node* pada setiap topologi *star*, tergantung dari jumlah *port* yang ada pada konsentrator. Topologi ini sangat mengandalkan kemampuan *hub* atau *switch* yang dapat menghubungkan dan menjaga lalu lintas komunikasi antar *node*. Biasanya topologi *star* diimplementasikan di jaringan sekolah, warnet, dan perkantoran (Suprpto, 2020: 20).



Gambar 2.1 Topologi Star

2. Topologi Mesh

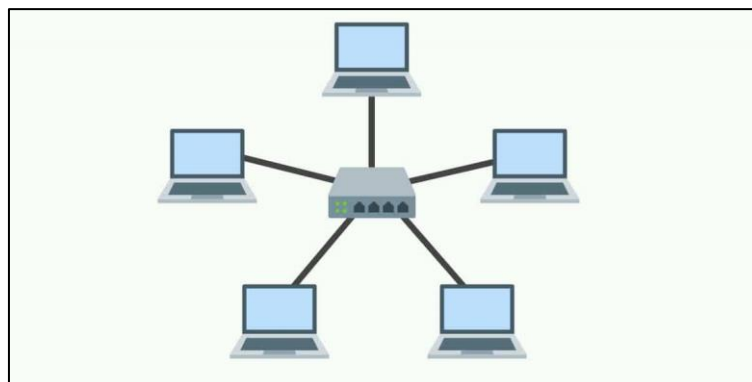
Topologi *mesh* dibangun dengan cara menghubungkan secara langsung antara *node* satu dengan *node* lainnya di dalam jaringan dengan menggunakan media kabel. Setiap *node* akan membutuhkan NIC sejumlah *node* yang terhubung dengannya. Topologi *mesh* umumnya digunakan untuk jaringan yang tidak terlalu besar (Suprpto, 2020: 23).



Gambar 2.2 Topologi Mesh

3. Topologi Ring

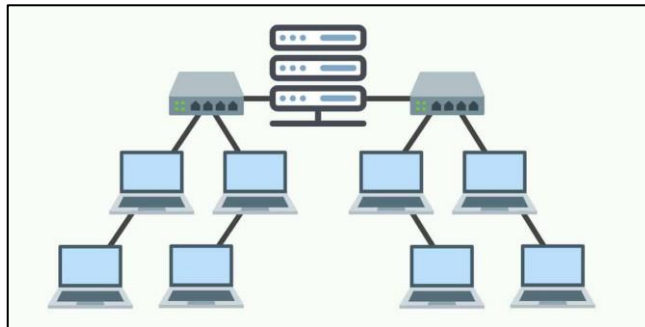
Topologi *ring* atau disebut dengan topologi cincin, adalah topologi yang mempunyai desain jaringan seperti cincin, yang berarti setiap *node* terhubung dengan dua titik kiri dan kanannya dan menyambung menjadi sebuah lingkaran. Pada topologi *ring*, setiap *node* yang terhubung di dalam jaringan juga berfungsi sebagai repeater atau penguat sinyal dari sinyal yang melaluinya. Setiap *node* dalam topologi *ring* membutuhkan 2 NIC (Suprpto, 2020: 26).



Gambar 2.3 Topologi Ring

4. Topologi Bus

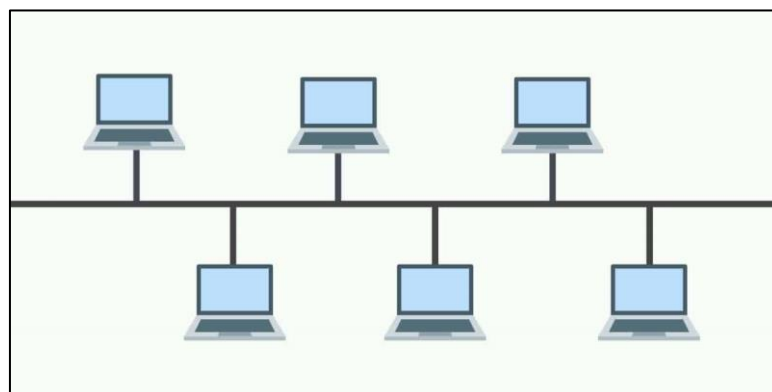
Topologi *bus* adalah topologi yang menghubungkan setiap *node* dengan menggunakan kabel *coaxial*. Kegunaan dari topologi *bus* adalah untuk menghubungkan antara satu jaringan dengan jaringan lainnya, sehingga dapat bertukar data atau informasi (Suprpto, 2020: 27).



Gambar 2.4 Topologi Bus

5. Topologi Tree

Topologi *tree* atau topologi pohon adalah topologi di jaringan komputer yang di bentuk dari kombinasi antara topologi *star* dan topologi *bus*. Topologi ini juga disebut sebagai topologi jaringan bertingkat (*hierarchy*). Topologi *tree* dapat digunakan untuk membangun jaringan pada skala besar (Suprpto, 2020: 29).



Gambar 2.5 Topologi Tree

2.5 Router

Router adalah *device* jaringan yang mempunyai fungsi utama yaitu menghubungkan dua atau lebih jaringan yang berbeda. Router termasuk

intermediate *device*, dan merupakan *device* jaringan yang cerdas. Router adalah *device* seperti *switch* yang meneruskan paket data berdasarkan pada *IP Address*. Router berada pada *layer Network*. Router normalnya menghubungkan LAN dan WAN secara bersama-sama dan mempunyai kemampuan untuk memperbaharui *routing table* secara dinamis, di mana router membuat keputusan tentang jalur dari paket data. Router memisahkan setiap *broadcast domain* pada *host* yang terhubung dengannya. *Routing* adalah proses penentuan jalur terbaik (*best path*) untuk mencapai tujuan tertentu. (Suprpto, 2020: 40).

Router adalah perangkat yang berfungsi untuk satu jaringan dengan jaringan lain yang memakai protokol komunikasi yang berbeda (Gustina dan Mutiara, 2017).

Jadi *router* adalah sebuah alat yang menghubungkan jaringan satu dengan yang lain untuk mengirimkan paket data.



Gambar 2.6 Router

2.6 Mikrotik

MikroTik merupakan sebuah perusahaan produsen penyedia perangkat jaringan komputer yang berkantor di Latvia. MikroTik *RouterOS™*, merupakan sistem operasi *Linux base* yang diperuntukkan sebagai *network router*. Didesain untuk memberikan kemudahan bagi penggunanya. Administrasinya bisa dilakukan melalui *Windows Application (Winbox)*. Selain itu instalasi dapat dilakukan pada Standard komputer PC (*Personal Computer*). PC yang akan dijadikan *router* mikrotik pun tidak memerlukan *resource* yang cukup besar untuk penggunaan standard, misalnya hanya sebagai *gateway*. Untuk keperluan beban yang besar (*network* yang kompleks, *routing* yang rumit) disarankan untuk mempertimbangkan pemilihan *resource* PC yang memadai (Handriyanto, 2015).

2.7 Access Point

Access Point dapat disebut dengan *Wireless Access Point (WAP)* adalah sebuah alat yang terdiri dari antena, *transiver* dan *Port RJ-45* yang berguna untuk membuat jaringan nirkabel atau WLAN. *Transceiver* pada *Access point* bertugas untuk memancarkan sinyal dan juga menerima sinyal dari *client* dan *server*. *Port RJ-45* di dalam *access point* berfungsi untuk mengoneksikan kabel yang menghubungkan antara *router* dengan *switch* atau *router* di atasnya (Suprpto, 2020: 41).

2.8 Switch

Switch adalah *multiport bridge* dan *buffer* dan di desain untuk meningkatkan efisiensi dan performansi. *Switch* berada pada *layer data link*. *Switch* dapat melakukan pengecekan kesalahan sebelum data diteruskan, sehingga menjadi sangat efisien. *Switch* membagi *collision domain* dari *host*, dan memiliki 1 *broadcast domain*. *Device* ini sudah mampu berkomunikasi secara *full-duplex*, sehingga mampu melakukan komunikasi dua arah dalam waktu yang sama tanpa adanya tabrakan data. *Switch* dalam konsep OSI berada di dalam *layer 1* yaitu *physical layer* (Suprpto, 2020: 39).



Gambar 2.7 *Switch*

2.9 Media Transmisi Jaringan

2.9.1 Jaringan Kabel (*Wired*)

Jaringan kabel adalah jaringan komputer yang menggunakan media transmisi kabel. Jaringan ini juga disebut dengan jaringan *guided*. Media *guided* atau dapat disebut dengan media tertuntun adalah media yang memiliki bentuk fisik, yaitu kabel. Jenis-jenis kabel yaitu: kabel *coaxial*, kabel *twisted pair*, dan

kabel *fiber optic*. Jenis kabel tersebut mempunyai karakteristik yang berbeda, baik bahan, bentuk fisik, teknologi, dan penggunaannya (Suprpto, 2020: 35)

2.9.2 Jaringan Nirkabel (Wireless)

Jaringan nirkabel atau jaringan wireless adalah jaringan komputer yang menggunakan media udara yang mengantarkan gelombang elektromagnetik untuk menghubungkan antar perangkat. Jenis-jenis jaringan nirkabel yaitu: gelombang radio (*radio frequency*), gelombang mikro (*microwave*), satelit, infra merah (*infrared*) (Suprpto, 2020: 35).

2.10 Zoom Meeting

Zoom merupakan aplikasi yang menggabungkan konferensi video, pertemuan *online*, obrolan, hingga kolaborasi seluler untuk menyediakan layanan konferensi jarak jauh. *Zoom* dapat menghadirkan kelas secara virtual sehingga pengajar dan peserta didik dapat bertatap muka secara virtual, melaksanakan proses pembelajaran secara dua arah dan memiliki efektifitas yang sama dengan pembelajaran tatap muka secara langsung (Kholis dan Syarif, 2020).

2.11 Application Programming Interface (API)

Sebuah dokumentasi *programming* yang terdiri dari fungsi, *interface*, *class*, struktur dan sebagainya supaya bisa membangun sebuah perangkat lunak yang disebut sebagai *Application Programming Interface* atau *API*. *API* merupakan sebagai kumpulan kode pemrograman yang menghubungkan antara perangkat lunak dengan perangkat lunak lainnya yang memungkinkan (Rahardja, Aini dan Santoso. 2018)

API memiliki fungsi untuk keperluan pengembangan aplikasi yang berperan dalam membawa pesan permintaan dari *user* untuk memberitahu yang harus dilakukan oleh *system*, setelah itu memberitahu respon yang sesuai dengan request yang dilakukan *user*.

2.12 Simple Queue

Pada sebuah jaringan yang mempunyai banyak *client*, diperlukan sebuah mekanisme pengaturan *bandwidth* dengan tujuan mencegah terjadinya monopoli

penggunaan *bandwidth* sehingga semua client bisa mendapatkan jatah *bandwidth* masing-masing. *QoS* (*Quality of services*) atau lebih dikenal dengan *Bandwidth* Manajemen, merupakan metode yang digunakan untuk memenuhi kebutuhan tersebut. Pada *RouterOS Mikrotik* penerapan *QoS* bisa dilakukan dengan fungsi *Queue*. Cara paling mudah untuk melakukan *queue* pada *RouterOS* adalah dengan menggunakan *Simple Queue*. Diantaranya bisa melakukan pengaturan *bandwidth* secara sederhana berdasarkan *IP Address client* dengan menentukan kecepatan *upload* dan *download* maksimum yang bisa dicapai oleh *client*. (Nugroho, Afandi dan Rahardjo, 2014).

Pada *Queue* terdapat fungsi *CIR* dan *MIR*. *CIR* (*Committed Information Rate*) yaitu menunjukkan *bandwidth* yang akan didapatkan saat jaringan paling sibuk sesuai dengan “*limit-at*”. Sedangkan *MIR* (*Maximal Information Rate*) yaitu kecepatan maksimum yang mungkin didapatkan saat jaringan tidak sibuk “*max-limit*”

2.13 *Quality of Service (QoS)*

QoS merupakan kemampuan suatu *network* menyediakan *service* lebih baik untuk user dalam membagi *bandwidth* sesuai kebutuhan data dan *voice* yang digunakan (Danur dan Febrizal, 2016). Parameter *QoS* antara lain :

1. **Throughput**

Throughput merupakan kecepatan transfer efektif yang diukur dalam bit per second. *Throughput* merupakan total jumlah bit paket selama transfer dibagi dengan durasi selang waktu transfer tersebut. Untuk menghitung nilai *throughput* (dalam *Bytes*) dapat menggunakan persamaan berikut (Danur dan Febrizal, 2016):

$$\textit{Throughput} = \frac{\textit{panjang paket diterima}}{\textit{waktu antar paket}}$$

2. **Packet Loss**

Packet loss merupakan penyebab utama pelemahan transfer *VoIP*. *Packet loss* terjadi karena pembuangan paket dalam jaringan di *gateway* sampai kedatangan

terakhir (*late loss*). Untuk menghitung nilai *packet loss* dapat menggunakan persamaan berikut (Danur dan Febrizal, 2016):

$$\text{Packet loss} = \frac{(\text{Total Packet} - \text{Packet Terkirim})}{\text{Packet total tercapture}} \times 100\%$$

3. *Delay*

Delay merupakan akumulasi dari berbagai waktu tunda dari ujung ke ujung pada jaringan internet. *Delay* menjadi acuan waktu transmisi paket dari pengirim hingga ke penerima (Danur dan Febrizal, 2016). Untuk menghitung nilai *delay*, dapat menggunakan persamaan berikut :

$$\text{Delay} = \frac{\text{waktu antar paket}}{\text{jumlah paket}}$$

2.14 *Wireshark*

Wireshark adalah salah satu *software* penyaring paket yang digunakan untuk menganalisa sebuah *trafik* pada jaringan. *Wireshark* dapat melihat trafik yang menuju sebuah alamat *interface*, tidak hanya itu bahkan *wireshark* dapat melihat semua *broadcast* dan juga *multicast*. Metode yang digunakan *wireshark* adalah *packet sniffing* yang mana *wireshark* memiliki kemampuan untuk *capture* semua paket yang dikirimkan dan diterima sebuah *interface* dalam jaringan kemudian di *decodekan* untuk dianalisa Informasi yang diperoleh dari *software wireshark* berupa sumber paket, tujuan paket, tipe paket, Panjang bit dan waktu *capture* paket (Danur dan Febrizal, 2016).

2.15 *Website*

Website dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara, dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia (Abdulloh, 2018). Halaman *website* dibuat menggunakan bahasa standar yaitu *HTML*. *Script HTML* ini akan diterjemahkan oleh *web browser* sehingga dapat ditampilkan dalam bentuk informasi yang dapat dibaca oleh semua orang. Secara umum, *website* terbagi menjadi tiga jenis, yaitu :

- *Website Statis*

Website statis yaitu jenis *website* yang isinya tidak diperbaharui secara berkala, sehingga isinya dari waktu ke waktu akan selalu tetap. *Website* jenis ini biasanya hanya digunakan untuk menampilkan profil dari pemilik *website* seperti profil perusahaan atau organisasi.

- *Website Dinamis*

Website dinamis yaitu jenis *website* yang isinya terus diperbaharui secara berkala oleh pengelola *web* atau pemilik *website*. *Website* jenis ini banyak dimiliki oleh perusahaan atau perorangan yang aktifitas bisnisnya memang berkaitan dengan internet. Contoh paling mudah dari *website* jenis ini yaitu *web* blog dan *website* berita.

- *Website Interaktif*

Website interaktif pada dasarnya termasuk dalam kategori *website* dinamis, dimana isi informasinya selalu diperbaharui dari waktu ke waktu. Hanya saja, isi informasi tidak hanya diubah oleh pengelola *website* tetapi lebih banyak dilakukan oleh pengguna *website* itu sendiri. Contoh *website* jenis ini yaitu *website* jejaring *social* seperti facebook dan twitter atau *website marketplace* seperti bukalapak, tokopedia, dan sebagainya (Abdulloh, Rohi, 2018:1).

2.16 Bootstrap

Bootstrap merupakan salah satu *framework CSS* yang paling populer dari sekian banyak *framework CSS* yang ada. *Bootstrap* memungkinkan desain sebuah *web* menjadi *responsive* sehingga dapat dilihat dari berbagai macam ukuran *device* dengan tampilan tetap menarik. Bootstrap juga membuat proses pengaturan desain menjadi lebih cepat karena tidak perlu lagi banyak menulis *CSS*, bahkan hampir tidak perlu kecuali jika memerlukan pengaturan desain yang berbeda dengan *style Bootstrap*. *Bootstrap* telah didukung oleh hampir semua *browser* baik pada *desktop* maupun *mobile* (Abdulloh, 2018:261).

Bootstrap adalah *framework open source* yang awalnya dibuat oleh Mark Otto dan Jacob Thornton untuk memudahkan dan mempercepat pengembangan *web* khusus untuk *front end*. *Framework* yang merupakan sekumpulan sintaks yang bisa

digunakan untuk membuat *website* secepat mungkin. Sehingga para *developer* tidak perlu menjalankan perintah dan fungsi dasar pada pembangunan *website*.

2.17 XAMPP

XAMPP menurut Madcoms (2016) , “XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *Perl*, *Filezilla* dan lain-lain.” Fungsi XAMPP sendiri adalah *server* yang berdiri sendiri (*localhost*).

XAMPP adalah sebuah *software open source* yang sistem penamaannya diambil dari gabungan kata *Apache*, *MySQL/MariaDB*, *PHP* dan *Perl*. Sedangkan untuk “X” yang terdapat diawalan kata berasal dari istilah *cross platform* sebagai simbol bahwa *software* ini bisa dijalankan diberbagai macam sistem operasi seperti *windows*, *linux*, *Mac OS*, dan juga *Solaris*.

2.18 HTML

HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar *web* yang dikelola penggunaanya oleh W3C (*Word Wide Web Consortium*) berupa *tag-tag* yang menyusun setiap elemen dari *website*. HTML berperan sebagai penyusun stuktur halaman *website* yang menempatkan setiap elemen *website* sesuai *layout* yang diinginkan.

Untuk mengetikkan *script HTML* dapat menggunakan *text editor* seperti Notepad sebagai paling bentuk sederhana atau *text editor* khusus yang dapat mengenali setiap unsur *script HTML* dan menampilkannya dengan warna yang berbeda sehingga mudah dibaca, seperti *Notepad++*, *Sublime Text*, dan masih banyak lagi aplikasi lain yang sejenis (Abdulloh, 2018:7).

HTML disimpan dengan ekstensi *.html, jika di dalamnya tidak mengandung skrip PHP. Jika di dalamnya mengandung skrip PHP, maka disimpan dengan ekstensi *.php. Dalam penulisan HTML ada beberapa *tag* yang wajib dituliskan dengan struktur yang sudah ditentukan.

2.19 *Javascript*

JavaScript merupakan bahasa pemrograman *web* yang pemrosesnya dilakukan di sisi *client*. Karena berjalan di sisi *client*, *JavaScript* dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan *PHP* yang bekerja di sisi *server*, untuk menjalankan *script JavaScript* tidak memerlukan *refresh* pada *browser*. *JavaScript* biasanya dijalankan ketika ada *event* tertentu yang terjadi pada *halam web*. Baik *event* yang dilakukan oleh *user*, maupun *event* yang terjadi karena adanya perubahan pada halaman *website* (Abdulloh, 2018:193).

2.20 *PHP*

Menurut Abdulloh (2018:3), “*PHP* singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses disisi *server*”. Sedangkan, menurut Madcoms (2016:2), “*PHP (Hypertext preprocessor)* adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam *HTML*”.

Jadi, dapat disimpulkan bahwa *PHP* adalah bahasa pemrograman *web* berbasis *server-side programming* (bahasa pemrograman yang diproses disisi *server*) yang memarsing kode *PHP* dari kode *web* dengan ekstensi *(.php)*, yang kemudian akan dikirim ke *browser web*.

2.21 *MySQL*

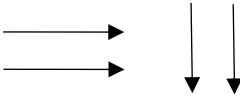
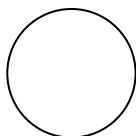
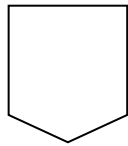

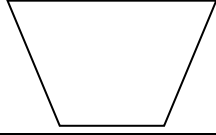
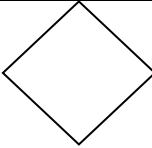
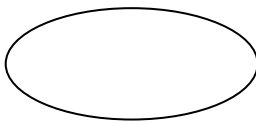
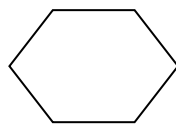
Menurut Shalahuddin, Muhammad dan Rosa A.S (2018:46), “*SQL (Structured Query Language)* adalah bahasa yang digunakan untuk mengelola data pada *RDBMS*. *SQL* awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”

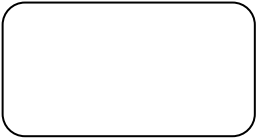
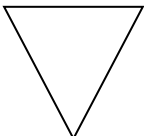


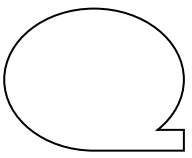


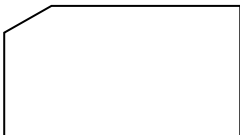
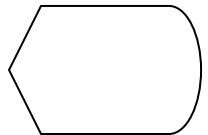
MySQL adalah sebuah *DBMS (database management system) open-source* yang cukup populer keberadaannya. *MySQL* bisa diakses atau dijalankan diberbagai macam sistem operasi, seperti *windows, linux, mac OS*, dll. *MySQL* menggunakan perintah *SQL (Structured Query Language)* yang banyak digunakan dalam pembuatan *website*.

2.22 Flowchart

Flowchart adalah sebuah bagan alir yang menggambarkan jalannya sebuah sistem. Gambaran ini dinyatakan dengan simbol, dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung (Arta dan Nugraha 2020). Simbol diagram *Flowchart* dapat dilihat pada Tabel 2.1 dibawah ini:

Tabel 2.1 Simbol Diagram *Flowchart*

NO	SIMBOL	KETERANGAN
1		Simbol arus/ <i>flow</i> (Simbol penghubung antar prosedur/proses)
2		Simbol <i>connector</i> (Simbol untuk keluar/masuk prosedur atau proses dalam lembar/halaman yang sama)
3		Simbol <i>off-line connector</i> (Simbol untuk keluar/masuk prosedur atau proses dalam lembar/halaman yang lain)
4		Simbol <i>process</i> (Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer)
5		Simbol <i>manual operation</i> (Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer)
6		Simbol <i>decision</i> (Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi)
7		Simbol <i>terminal</i> (Simbol untuk permulaan atau akhir dari suatu program)
8		Simbol <i>predefined process</i> (Simbol untuk mempersiapkan penyimpanan yang akan

		digunakan sebagai tempat pengolahan di dalam <i>storage</i>)
9		Simbol <i>keying operation</i> (Simbol operasi dengan menggunakan mesin yang mempunyai <i>keyboard</i>)
10		Simbol <i>off-line storage</i> (Simbol yang menunjukkan bahwa data di dalam simbol ini akan disimpan)
11		Simbol <i>manual input</i> (Simbol untuk pemasukan data secara <i>manual on-line keyboard</i>)
12		Simbol <i>input-output</i> (Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya)
13		Simbol <i>magnetic-tape unit</i> (Simbol yang menyatakan <i>input</i> berasal dari pita magnetik atau <i>output</i> disimpan ke pita magnetik)
14		Simbol <i>disk and on-line storage</i> (Simbol untuk menyatakan <i>input</i> berasal dari disk atau <i>output</i> disimpan ke disk)
15		Simbol <i>document</i> (Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas)
16		Simbol <i>punched card</i> (Simbol yang menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu)
17		Simbol <i>display</i> (Simbol yang menyatakan peralatan <i>output</i> yang digunakan yaitu layar, <i>plotter</i> , printer, dan sebagainya)