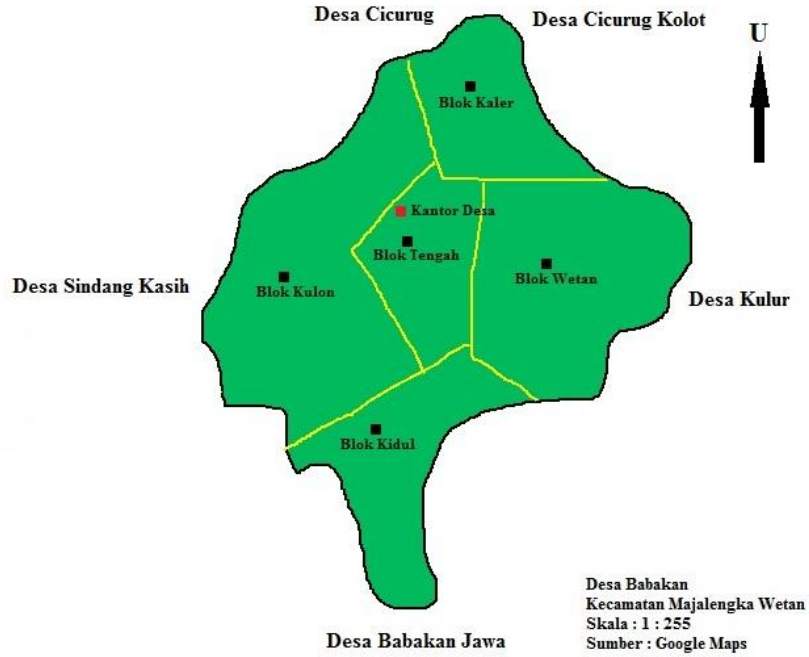
BAB II

TINJAUAN PUSTAKA

II.1. Profil Desa Babakan

Memberikan sebuah informasi mengenai keadaan wilayah, pemerintahan,keadaan penduduk, perekonomian penduduk sampai dengan lembaga yang berada di desa tersebut.

II.1.1. Gambaran Umum



Gambar II.1 Peta Lokasi Desa Babakan

Desa Babakan yang berada dilingkungan Kecamatan Majalengka Wetan, KabupatenMajalengka. Dalam kehidupan sehari-hari, di Desa Babakan ini ditemukan bahwa mayoritas penduduknya bekerja di sektor pertanian tetapi ada juga yang bekerja di sektor peternakan khususnya peternakan lele. Peternakan lele

yang ada di Desa Babakan sudah berorientasi ke bisnis oleh karena itu di desa ini sudah terdapat fasilitas yang mendukung kelancaran distribusi produk peternakan khususnya olahan lele. Desa Babakan mempunyai tempat penampungan daging lele atau KUD sehingga dapat memudahkan masyarakat sekitar untuk memasarakan hasil dari panen lele tersebut.Desa Babakan terletak di Kecamatan Majalengka Wetan, KabupatenMajalengka dengan luas desa sekitar 530 Ha.

Desa Babakanmemiliki batas-batas wilayah, adapun sebagai berikut :

Sebelah Utara : Desa Cicurug. Cicurug Kolot

Sebelah Selatan : Desa Babakan Jawa

Sebelah Barat : Desa Sindangkasih

Sebelah Timur : Desa Kulur

Desa Babakan berada sedikit ke dalam dari jalan utama, sehingga akses untuk ke kota memakan waktu yang tidak terlalu lama. Sarana transportasi yang umumnya digunakan oleh para penduduk adalah angkot, ojek dan kendaraan milik pribadi.

II.1.2. Keadaan Penduduk dan Perekonomian

Penduduk Desa Babakan berjumlah sekitar 6685 jiwa dengan tingkat pendidikan mayoritas yang tergolong masih rendah yaitu Sekolah Dasar (SD). Penduduknya sebagian besar berprofesi sebagai petani dan peternak. Penduduk kaum laki-lakinya sebagian besar bekerja menjadi petani atau peternak, sedangkan perempuannya sebagai ibu rumah tangga, selain itu ada juga yang bekerja sebagai petani, apabila mempunyai ternak maka yang mencarikan pakannya adalah suami yang dibantu oleh anaknya. Keahlian lain dari penduduk desa Babakan adalah

sebagai dekorator bunga hias, baik untuk acara pernikahan, penataan taman dan untuk pembuatan papan ucapan.

Sektor peternakan tidak menjadi sumber pendapatan utama bagi masyarakat Desa Babakan dalam menghidupi kebutuhan keluarga mereka. Hal tersebut dikarenakan sebagian besar dari mereka bermata pencaharian sebagai petani padi (mayoritas) dan juga peternakikan lele. Keberadaan sektor peternakan di Desa Babakan ini sebagian besar terdapat di Dusun rebo dan Dusun senen, dengan komoditi ternak di dusun rebo yaitu ternak lele dan di dusun senes ternak kambing Kultur dari Desa Babakan mayoritas penduduknya beragama Islam, hal ini dapat dilihat dari kegiatan keagamaan yang terstuktur dengan baik seperti tahlilan dan pengajian rutin setiap malam jumat.

II.1.3. Keadaan Pemerintahan dan Kelembagaan

II.1.3.1. Pemerintahan Desa

Desa Babakan memiliki perintahan untuk mengatur segala bentuk kegiatan yang terlibat dalam suatu desa, maka diperlukan bentuk kepengurusan seperti kepala desa, sekretaris dan jabatan-jabatan yang lain, adapun sebagai berikut :

Kepala Desa :H. Anwar Abdillah

Sekretaris Desa : Agus Setiawan

Kepala Urusan

Pemerintahan : Nur Ikhwan

Pembangunan : Mu’adi

Keuangan : Solehin

Kelembangaan : Iman

Kepala Dusun

BlokKulon : Nanang

Blok Wetan :Dadang

Blok Kaler : Deden

Blok Kidul : Satim

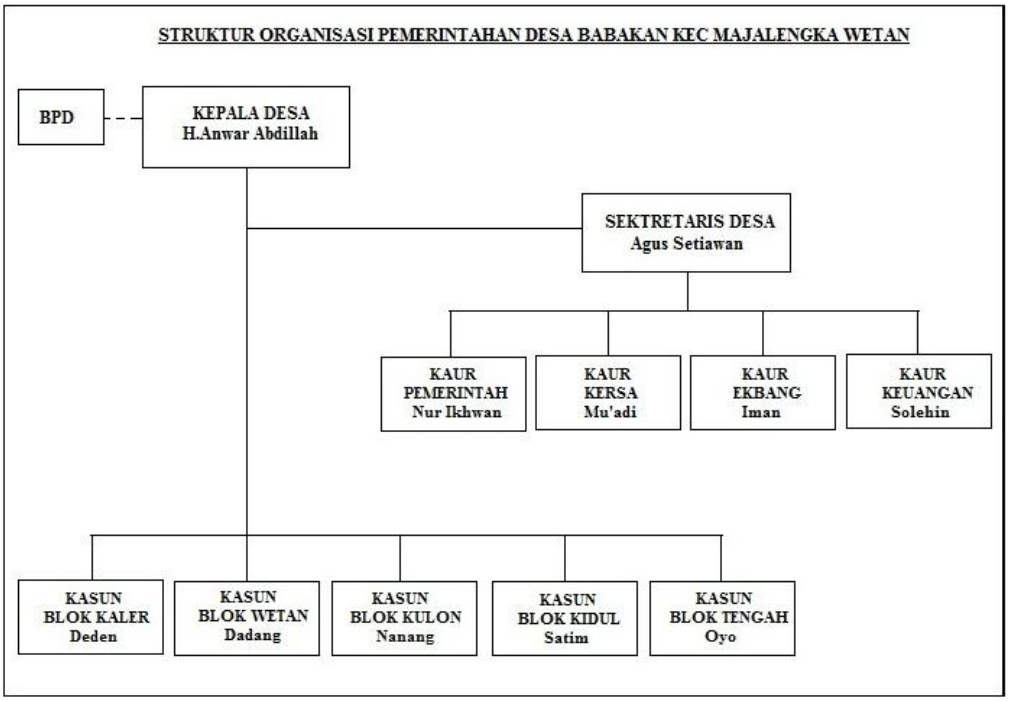
Blok Tengah : Oyo

II.1.3.2. Kelembagaan Desa

Disamping lembaga resmi seperti pemerintahan, Desa Babakan memiliki lembaga non pemerintahan, adapun sebagai berikut :

1. PKK (Program Kesejahteraan Keluarga)
2. Posyandu / Polindes
3. Karang Taruna
4. Badan Perwakilan Desa (BPD)
5. Kegiatan keagamaan
6. Sinoman

II.1.4. Struktur Organisasi Desa



Gambar II.2 Struktur Organisasi Desa Babakan

II.1.5. Deskripsi Tugas

Deskripsi tugas digunakan untuk mengetahui tugas, wewenang dan tanggung jawab dari masing-masing bagian. Adapun deskripsi tugas yang ada di Desa Babakan adalah sebagai berikut :

1. BPD

   Badan perwakilan pemerintah daerah yang memiliki tugas untuk mengawasi hal-hal yang terjadi dalam sebuah desa.

2. Kepala Desa

   Memiliki kewenangan dalam memutuskan suatu hal yang berkaitan dengan kemajuan maupun perubahan desa yang di pimpinnya.

   1. Memiliki wewenang dalam memerintahkan anggotanya dalam melaksanakan tugas yang harus diselesaikan.

2. Bertanggung jawab penuh atas segala sesuatu yang terjadi dalam sebuah desa.

3. Sekertaris Desa

4. Memiliki tugas sebagai asisten kepala desa, dimana semua laporan dari masing-masing anggota yang akan di laporkan kepada kepala desa melalui sekretaris.

5. Kapala Urusan Pemerintahan

   Memiliki tugas sebagai asisten kepala desa untuk bagian pemerintahan.

6. Kepala Urusan Pembangunan

   Memiliki tugas dalam hal pembangunan yang terjadi di desa.

7. Kepala Urusan Keuangan

   Memiliki tugas sebagai bendahara desa, bertanggung jawab atas keuangan desa.

8. Kepala Urusan Kelembangaan

   Memiliki tugas dalam mengawasi seluruh kelembagaan yang ada di desa.

9. Kepala Dusun

   Memiliki tugas untuk melaporkan dan pendataan warga masyarakat yang ada di lingkungan dusun yang di pimpinnya.

II.2. Teori Keilmuan/Landasan Teori

Teori keilmuan atau landasan teori dimanfaatkan sebagai pemandu agar fokus penelitian sesuai dengan fakta di lapangan.Selain itu landasan teori juga bermanfaat untuk memberikan gambaran umum tentang latar penelitian dan sebagai bahan pembahasan hasil penelitian.

II.2.1. Jaringan Komputer

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 perangkat yang terhubung dengan jaringan disebut ***node***.Sebuah jaringan komputer dapat memiliki dua, puluhan, ribuan atau bahkan jutaan *node*. Sebuah jaringan biasanya terdiri dari 2 atau lebih komputer yang saling berhubungan diantara satu dengan yang lain, dan saling berbagi sumber daya misalnya *CDROM*, Printer, pertukaran data atau emungkinkan untuk saling berkomunikasi secara elektronik. Komputer yang terhubung tersebut, dimungkinkan berhubungan dengan media kabel, saluran telepon, gelombang radio, satelit, atau sinar infra merah.

II.2.1.1. Manfaat Jaringan Komputer

Sharing resources, bertujua n agar seluruh program, peralatan lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan komputer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai. Dengan kata lain, seorang *computer user* yang letaknya jauh sekalipun dapat memanfaatkan data maupun informasi yang lainnya tanpa mengalami kesulitan.

Dengan jaringan komputer kita akan mendapatkan reliabilitas yang tinggi dengan memiliki sumber-sumber alternatif persediaan. Misal semua data data disimpan atau disalin ke dua, tiga atau lebih komputer yang terkoneksi ke suatu

jaringan.Sehingga bila salah satu komputer rusak, maka data masih dapat digunakan dengan menggunakan komputer lainnya.

Jaringan komputer dapat menghemat uang. Kita dapat membeli suatu barang di Negara lain tanpa harus mengeluarkan uang yang banyak. Hanya dengan menggunakan jaringan komputer kita dapat membeli barang tersebut.

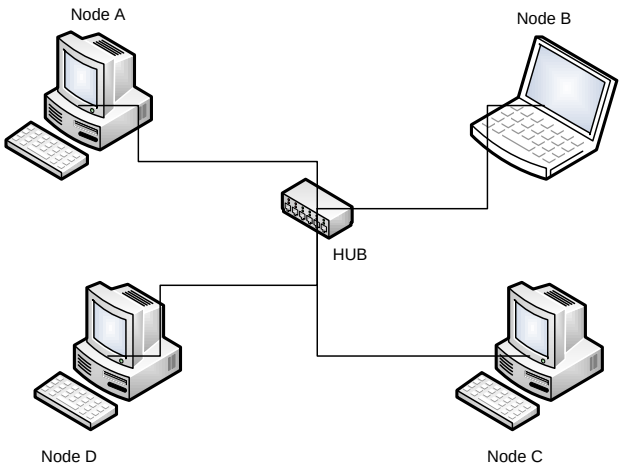
Sistem jaringan komputer memberikan perlindungan terhadap data.Jaminan keamanan tersebut diberikan melalui pengaturan hak akses para pemakai, serta teknik perlindungan terhadap *harddisk* (media penyimpanan data) sehingga data mendapatkan perlindungan yang efektif.

II.2.1.2. Jenis Jaringan Komputer

Jaringan komputer dapat dibedakan berdasarkan cakupan geografisnya. Ada empat kategori utama jaringan komputer yaitu :

1. LAN (*Local Area Network*)

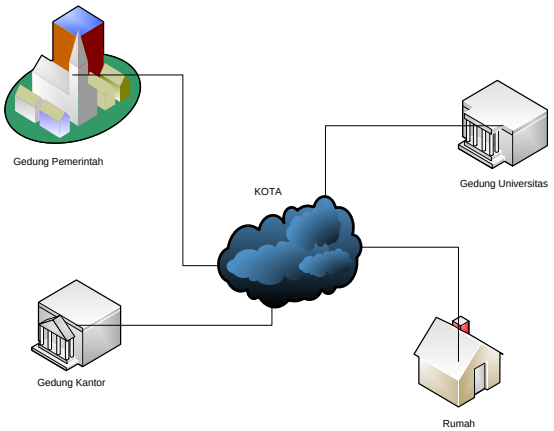
   LAN digunakan untuk menghubungkan komputer yang berada di dalam suatu area yang kecil, misalnya di dalam suatu gedung perkantoran atau kampus. Jarak antar komputer yang dihubungkannya bisa mencapai 5 sampai 10 km. Suatu LAN biasanya bekerja pada kecepatan mulai 10 Mbps sampai 100 Mbps. LAN menjadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya secara bersama-sama. Contoh dari sumber daya yang dapat digunakan itu misalnya suatu mainframe, *file server*, printer, dan sebagainya.



Gambar II.3 LAN

2. MAN (*Metropolitan Area Network*)

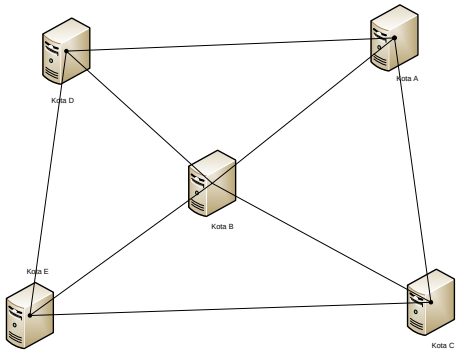
MAN merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN bisa mencapai 10 km sampai beberapa ratus km. Suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps. Pada gambar 2.4 anda dapat melihat suatu ilustrasi tentang MAN.



Gambar II.4 MAN

3. WAN (*Wide Area Network*)

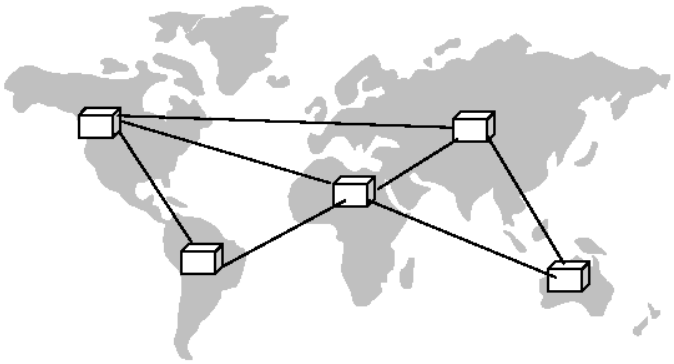
   WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas, seperti hubungan dari satu kota ke kota lain di dalam suatu negara. Cakupan WAN bisa meliputi 100 km sampai 1.000 km, dan kecepatan antar kota bisa bervariasi antara 1,5 Mbps sampai 2,4 Gbps. Dalam WAN, biaya untuk peralatan transmisi sangat tinggi, dan biasanya jaringan WAN dimiliki dan dioperasikan sebagai suatu jaringan publik. Para pelaku bisnis dapat menyewa sistem transmisi tersebut untuk menghubungkan kantor-kantor cabang yang dimilikinya. Gambar 2.5 mengilustrasikan suatu WAN.



Gambar II.5 MAN

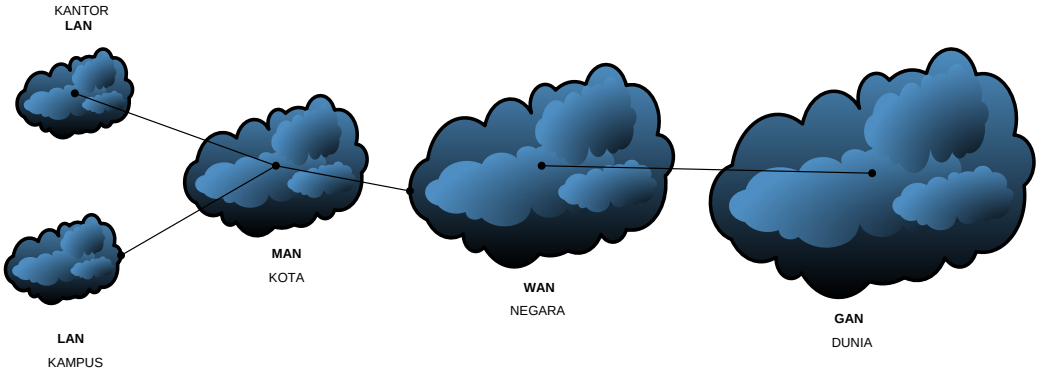
4. GAN (*Global Area Network*)

   GAN merupakan suatu jaringan yang menghubungkan negara-negara diseluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps sampai dengan100 Gbps dan cakupannya mencakapi ribuan kilometer. Contoh yang sangat baik dari GAN ini adalah *Internet*.Gambar 2.6 memperlihatkan contoh suatu GAN.



Gambar II.6 GAN

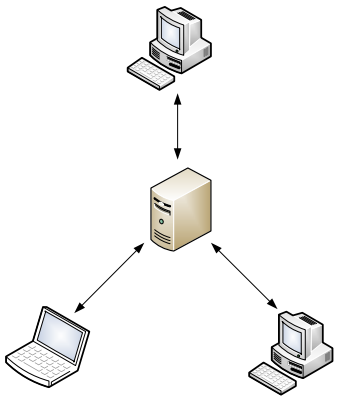
LAN, MAN, WAN dan GAN dapat berinteraksi satu sama lain. Gambar 2.7 memperlihatkan interaksi antara jaringan-jaringan tersebut.



Gambar II.7*Relationship* LAN, MAN, WAN, GAN

II.2.1.3. *Client* – Server

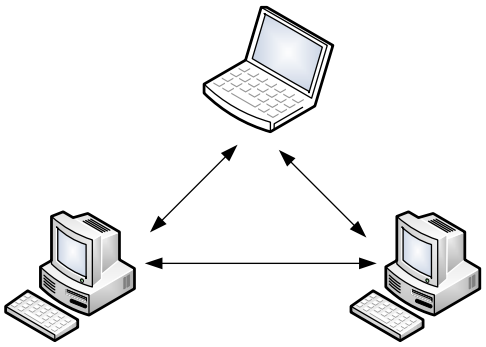
Yaitu jaringan komputer dengan komputer yang didedikasikan khusus sebagai *server*.Sebuah *service*/layanan bisa diberikan oleh sebuah komputer atau lebih.Contohnya adalah sebuah domain seperti *www.detik.com* yang dilayani oleh banyak komputer *webserver*.Atau bisa juga banyak *service*/layanan yang diberikan oleh satu komputer.Contohnya adalah *serverjtk.polban.ac.id* yang merupakan satu komputer dengan multi *service* yaitu *mail server, web server, fileserver, database server* dan lainnya.



Gambar II.8*Client-Server*

II.2.1.4. *Peer - To – Peer*

Yaitu jaringan komputer dimana setiap *host* dapat menjadi *server* dan juga menjadi *client* secara bersamaan. Contohnya dalam *file* sharing antar komputer di Jaringan *Windows Network Neighbourhood* ada 5 komputer (kita beri nama A,B,C,D dan E) yang memberi hak akses terhadap *file* yang dimilikinya. Pada satu saat A mengakses *fileshare* dari B bernama data_nilai.xls dan juga memberi akses *file* soal_uas.doc kepada C. Saat A mengakses *file* dari B maka A berfungsi sebagai *client* dan saat A memberi akses *file* kepada C maka A berfungsi sebagai *server*. Kedua fungsi itu dilakukan oleh A secara bersamaan maka jaringan seperti ini dinamakan *peer to peer.*



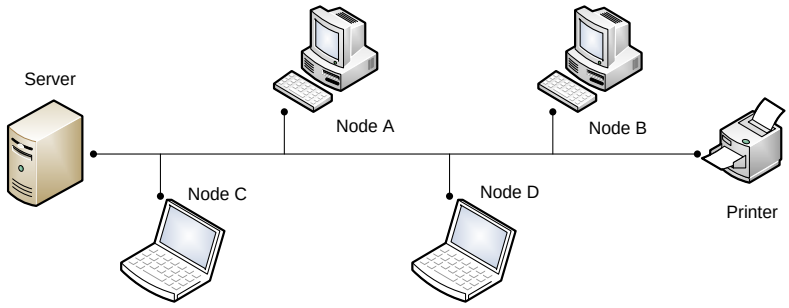
Gambar II.9*Peer-To-Peer*

II.2.1.5. Topologi Jaringan

Topologi jaringan komputer adalah bentuk perancangan baik secara fisik maupun secara logik yang digunakan untuk membangun sebuah jaringan komputer. Adapun hal yang menjelaskan hubungan geometris antara unsur-unsur dasar penyusun jaringan, yaitu *node*, *link*, dan *station*.

II.2.1.5.1 Topologi *Bus*

Topologi Bus adalah sebuah kabel tunggal atau kabel pusat dimana seluruh *workstation* dan *server* dihubungkan. Keunggulan topologi Bus adalah pengembangan jaringan atau penambahan *workstation* baru dapat dilakukan dengan mudah tanpa mengganggu *workstation* lain. Kelemahan dari topologi ini adalah bila terdapat gangguan di sepanjang kabel pusat maka keseluruhan jaringan akan mengalami gangguan.

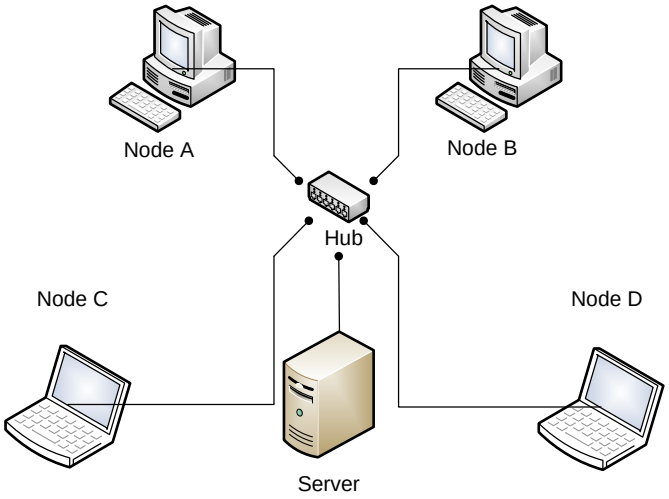


Gambar II.10 Topologi *Bus*

II.2.1.5.2 Topologi *Star*

Pada topologi Star, masing-masing workstation dihubungkan secara langsung ke server atau hub. Keunggulan dari topologi tipe Star ini adalah bahwa dengan adanya kabel tersendiri untuk setiap workstation ke server, maka bandwidth atau lebar jalur komunikasi dalam kabel akan semakin lebar sehingga akan meningkatkan unjuk kerja jaringan secara keseluruhan. Dan juga bila

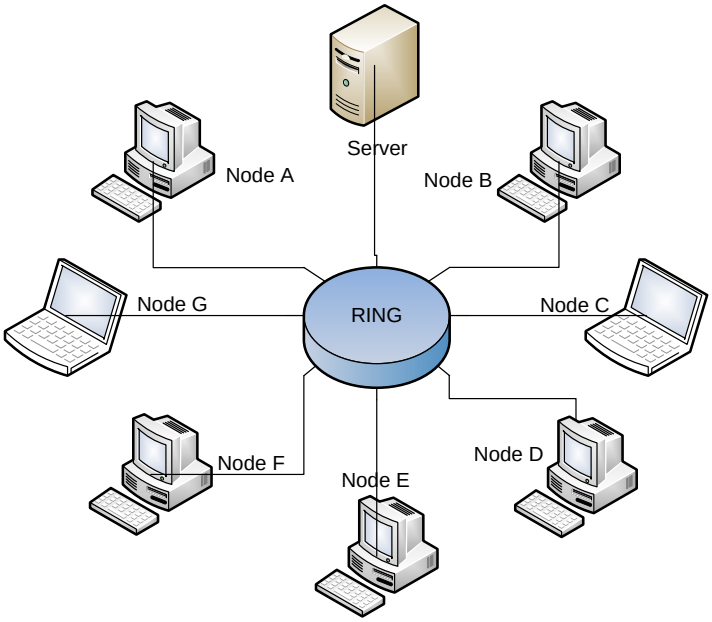
terdapat gangguan di suatu jalur kabel maka gangguan hanya akan terjadi dalam komunikasi antara workstation yang bersangkutan dengan server, jaringan secara keseluruhan tidak mengalami gangguan. Kelemahan dari topologi Star adalah kebutuhan kabel yang lebih besar dibandingkan dengan topologi lainnya.



Gambar II.11 Topologi *Star*

II.2.1.5.3 Topologi *Ring*

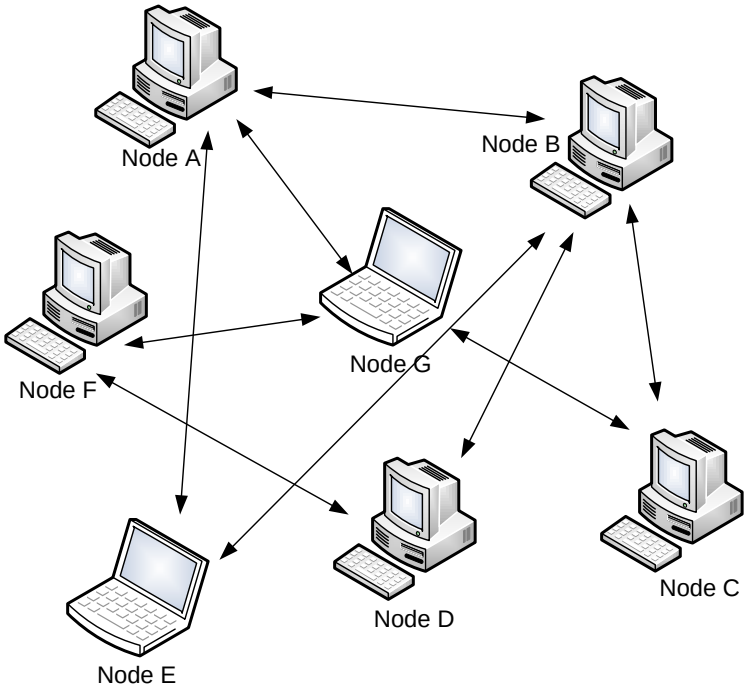
Di dalam topologi *Ring* semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin. Tiap *workstation* ataupun *server*akan menerima dan melewatkan informasi dari satu komputer ke komputer lain, bila alamat-alamat yang dimaksud sesuai maka informasi diterima dan bila tidak informasi akan dilewatkan. Kelemahan dari topologi ini adalah setiap *node* dalam jaringan akan selalu ikut serta mengelola informasi yang dilewatkan dalam jaringan, sehingga bila terdapat gangguan di suatu *node* maka seluruh jaringan akan terganggu. Keunggulan topologi *Ring* adalah tidak terjadinya *collision* atau tabrakan pengiriman data seperti pada topologi *Bus*, karena hanya satu *node* dapat mengirimkan data pada suatu saat.



Gambar II.12 Topologi *Ring*

II.2.1.5.4 Topologi *Mesh*

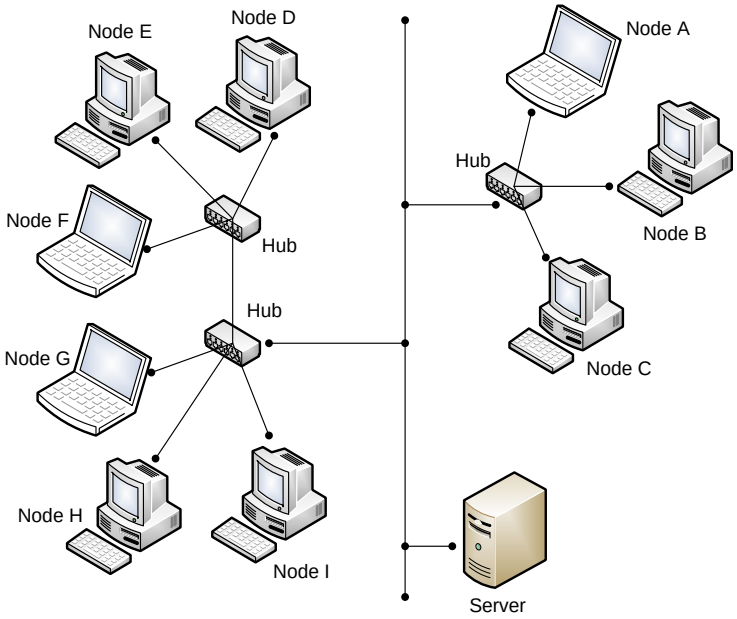
Pada topologi *mesh* tiap komputer terhubung langsung dengan komputer lain (pear to pear). Setiap komputer mempunyai jalur sendiri-sendiri dengan komputer lain, sehigga tidak akan terjadi collision domain. Kerugian dari penggunaan topologi ini adalah penggunaan ethernet dan kabel yang banyak sehingga dibutuhkan dana yang besar. Keuntungan dari penggunaan topologi ini adalah apabila ada salah satu jalur pada komputer putus, komputer masih dapat berhubungan dengan jalur yang lain.



Gambar II.13 Topologi *Mesh*

II.2.1.5.5 Topologi *Tree*

Topologi Jaringan Pohon (*Tree*) Topologi jaringan ini disebut juga sebagai topologi jaringan bertingkat.Topologi ini biasanya digunakan untuk interkoneksi antar sentral dengan *hirarki* yang berbeda.Untuk *hirarki* yang lebih rendah digambarkan pada lokasi yang rendah dan semakin keatas mempunyai hirarki semakin tinggi.Topologi jaringan jenis ini cocok digunakan pada sistem jaringan komputer. Kelebihan Seperti topologi star perangkat terhubung pada pusat pengendali / *Hub*. Tetapi *Hub* dibagi menjadi dua, central *Hub*, dan *secondaryHub* Topologi*tree* ini memiliki keunggulan lebih mampu menjangkau jarak yang lebih jauh dengan mengaktifkan fungsi *Repeater* yang dimiliki oleh *Hub*. Kelemahan topologi *tree*kabel yang digunakan menjadi lebih banyak sehingga diperlukan perencanaan yang matang dalam pengaturannya , termasuk di dalamnya adalah tata letak ruangan.



Gambar II.14 Topologi *Tree*

II.2.1.6. Jaringan *Wireless*

Wireless atau *wirelessnetwork* merupakan sekumpulan komputer yang saling terhubung antara satu dengan lainnya sehingga terbentuk sebuah jaringan komputer dengan menggunakan media udara/gelombang sebagai jalur lintas datanya. Pada dasarnya *wireless* dengan LAN merupakan sama-sama jaringan komputer yang saling terhubung antara satu dengan lainnya, yang membedakan antara keduanya adalah media jalur lintas data yang digunakan, jika LAN masih menggunakan kabel sebagai media lintas data, sedangkan *wireless* menggunakan media gelombang radio/udara. Penerapan dari aplikasi *wirelessnetwork* ini antara lain adalah jaringan nirkabel diperusahaan, atau mobile communication seperti handphone, dan HT.

II.2.1.6.1 *Wireless* LAN

WirelessLocal AreaNetwork pada dasarnya sama dengan jaringan Local Area *Network* yang biasa kita jumpai. Hanya saja, untuk menghubungkan antara *node* device antar *client* menggunakan media *wireless*, chanel frekuensi serta SSID yang unik untuk menunjukkan identitas dari *wireless* device. Tidak seperti pada LAN konvensional (kabel), pada *Wireless* LAN hanya terbagi ke dalam dua *mode* pemasangan (instalasi), yaitu *mode* add hock dan infrastruktur. Komunikasi Add Hock adalah sambungan komunikasi langsung antara masing-masing komputer/laptop dengan menggunakan media *wireless*. Penggunaan *mode* ini sama halnya dengan hubungan komunikasi *point* to multi *point* pada jaringan LAN konvensional.

Masing-masing PC atau Laptop yang akan dihubungkan dengan *mode* add hock ini harus mempunyai SSID sebagai identitas dari PC yang akan digunakan untuk komunikasi dengan yang lainnya.Pada komunikasi *Add-hock,* tidak memerlukan *accesspoint* untuk bisa saling berhubungan. Masing-masing host hanya harus memiliki transceiver serta receiver *wireless* untuk bisa berkomunikasi secara langsung. *Mode* yang kedua adalah infrastruktur, dimana jaringan ini diperlukan sebuah akses *point* untuk melayani komunikasi utama pada jaringan *wireless*.Keberadaan *accesspoint* dimaksudkan untuk mentransmisikan data pada PC untuk jangkauan tertentu pada suatu area/wilayah.Pada *mode* infrastruktur ini dapat diperluas lagi menjadi jaringan *Wireless* LAN yang lebih besar dan kompleks dengan menambahkan beberapa *AccessPoint* pada titik-titik tertentu untuk memperluas jangkauannya.

Wireless LAN diperlukan ketika sebuah LAN konvensional tidak lagi bisa dikembangkan karena alasan tertentu, misal, sulitnya pengembangan *mode*l LAN konvensional karena keterbatasan tempat, ruang dan hal lainnya.

II.2.1.6.2 *WirelessMesh*

Wirelessmesh network (WAN) adalah suatu jaringan komunikasi radio pada sebuah topologi *mesh*. Cakupan area dari *node* nya bekerja seperti suatu jaringan tunggal. Akses ke jaringan tunggal *mesh* ini tergantung pada *node* nya dan bekerja seirama satu dengan lainnya untuk membentuk satu jaringan radio. Sebuah jaringan *mesh* dapat diunggulkan dan memiliki kelebihan. Ketika satu *node* tidak dapat beroperasi, *node* tersebut masih bisa berkomunikasi dengan yang lainnya, langsung atau melewati satu *node* atau lebih.

II.2.1.6.2.1 Arsitektur *Mesh*

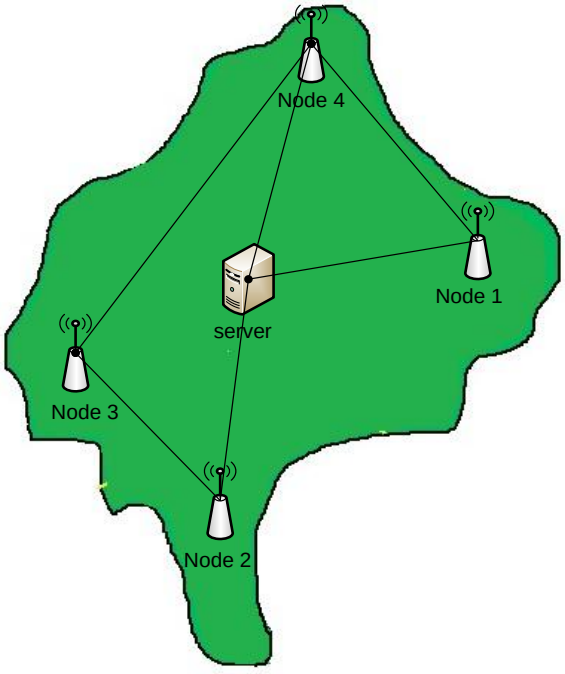
Arsitekur *wirelessmesh* adalah langkah pertama untuk menuju jaringan *bandwidth* yang besar yang melebihi kawasan yang terjangkau. Infrasukturnya jaringan ini tidak memakai kabel sebagai penghubung antar *node.*Arsitektur*mesh*menopang sinyal yang kuat dengan menahan jarak yang jauh menjadi sebuah jaringan seri dan lebih pendek. Perantara *node*nya tidak hanya mendorong sinyal, tetapi saling kerjasama membuat penetapan pada skema jaringannya, contohnya performansi *routing*.

Contoh tiga jenis jaringan *wirelessmesh*:

1. *Infrastructure wirelessmeshnetworks* : bentuk *routermesh* sebuah infrastruktur untuk *user*.

2. *Clientwirelessmeshnetworks* :*nodeclient* membentuk jaringan aktual untuk fungsionalitas konfigurasi performa *routing*.
3. *Hybrid wirelessmesh networks*: *clientmesh* dapat mengetahui performansi fungsional *mesh* dengan *clientmesh* lainnya sama seperti jaringan akses.

Jaringan *wirelessmesh* relatif stabil untuk sebuah jaringan kecuali untuk panambahan *node*baru terkadang terjadi kesalahan. Trafiknya, menjadi berubah secara tidak frekuensial terhadap jumlah *user*nya. Kenyataannya semua trafik yang terdapat di jaringan infrastruktur *mesh* diteruskan atau berasal dari *gateway* yang lain.

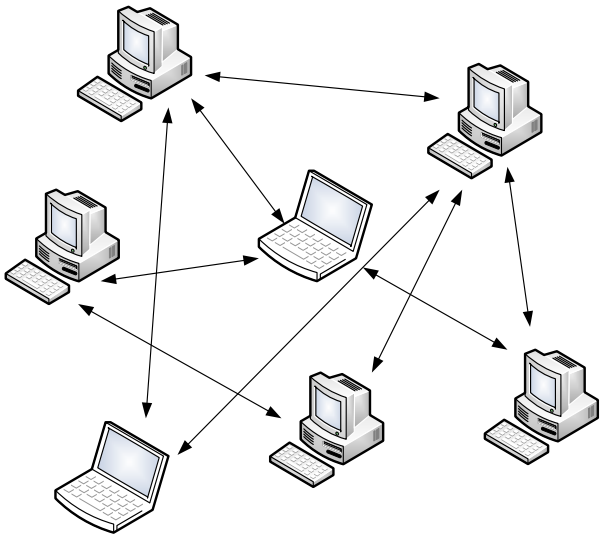


Gambar II.15*WirelessMesh*

II.2.1.6.2.2 Topologi *mesh*

Jaringan *mesh* adalah suatu alternatif untuk perutingan data, suara dan instruksi antar *node*. Jaringan *mesh* ini melanjutkan koneksi dan merekonfigurasi kanal yang rusak atau kanal yang di blok oleh *"hopping"* dari semua *node-node*sampai tujuan tercapai. Sehingga semua *node*nya terkoneksi. Jaringan

*mesh*berbeda dengan bagian komponen jaringan yang dapat di koneksikan kejaringan yang lain melalui *multiplehops* dan umumnya bukan bagian dari komponen *mobile*. Jaringan *mesh* dapat dilihat sebagai suatu jaringan *ad-hoc*. Jaringan *mobile ad hoc* (MANETs) dan jaringan *mesh* merupakan suatu jaringan tertutup, tetapi MANETs dapat menyelesaikan masalah dengan cara perpindahan *node*.



Gambar II.16 Topologi *Mesh*

II.2.1.7. Protokol TCP/IP

II.2.1.7.1 TCP (*Transmission Control Protocol*)

Transmission Control Protocol (TCP) melakukan transmisi data per segmen, artinya paket data dipecah dalam jumlah yang sesuai dengan besaran paket, kemudian dikirim satu per satu hingga selesai. Agar pengiriman data sampai dengan baik, maka pada setiap paket pengiriman TCP akan menyertakan nomor seri (*sequence number*). Komputer *client*yang menerima paket tersebut harus mengirim balik sebuah sinyal *ACKnowledge* dalam satu periode yang ditentukan. Bila pada waktunya komputer *client*belum juga memberikan *ACK*,

maka akan terjadi *time out* yang menandakan pengiriman paket gagal dan harus diulang kembali. *Mode*l protokol TCP disebut sebagai *connection oriented protocol*.

Port merupakan pintu masuk datagram dan paket data.*Port* data dibuat mulai dari 0 sampai 65.536.*Port* 0 sampai 1024 diesdiakan untuk layanan standar, seperti FTP, TELNET, Mail, *Web* dan lainnya. *Port* ini lebih dikenal dengan nama*well-know port.*

Tabel II.1 port TCP

NO Port	Keterangan
21	FTP
23	Telnet
25	SMTP
80	HTTP/*Web*
110	POP3
161	SNMP
162	SNMP Trop

II.2.1.7.2 IP (*Internet Protocol*)

Internet protokol menggunakan IP *address* sebagai identitas. Pengiriman data akan dibungkus dalam paket dengan label berupa IP *address* si pengirim dan IP address penerima. Apabila IP penerima melihat pengiriman paket tersebut dengan identitas IP *address* yang sesuai, maka datagram tersebut akan diambil dan disalurkan ke TCP melalui *port.* IP *address* dibagi menjadi dua bagian, yaitu :

1. *Network ID* (menyatakan identitas jaringan)

2. *HOST ID* (menyatakan identitas komputer)

Sedangkan untuk penulisan IP *address* terbagi atas empat oktet, yang masing-masing mempunyai nilai maksimal 255 atau lebih jelasnya dapat dilihat pada sub bab berikut dalam Tugas Akhir ini.

II.2.1.7.3 IPv4

Alamat IP versi 4 sering disebut dengan Alamat IPv4 adalah sebuah jenis pengalamatan jaringan yang digunakan di dalam protokol jaringan TCP/IP yang menggunakan protokol IP versi 4. Panjang totalnya adalah 32-bit, dan secara teoritis dapat mengalamati hingga 4 miliar *host* komputer atau lebih tepatnya 4.294.967.296 *host* di seluruh dunia, jumlah *host* tersebut didapatkan dari 256 (didapatkan dari 8 bit) dipangkat 4 karena terdapat 4 oktet, sehingga nilai maksimal dari alamt IP versi 4 tersebut adalah 255.255.255.255 dimana nilai dihitung dari nol sehingga nilai nilai *host* yang dapat ditampung adalah 256x256x256x256 = 4.294.967.296 *host*. sehingga bila *host* yang ada diseluruh dunia melebihi kuota tersebut maka dibuatlah IP versi 6 atau IPv6.

II.2.1.7.4 IP Address

Alamat IP (*Internet Protocol Address*) atau sering disingkat IP biasa digunakan untuk mengindentifikasi *interface* jaringan pada *host* dari suatu mesin. adalah deretan angka biner antar 32-bit sampai 128-bit yang dipakai sebagai alamat identifikasi untuk tiap komputer *host* dalam jaringan Internet. Panjang dari angka ini adalah 32-bit untuk IPv4 atau IP versi 4, dan 128-bit untuk IPv6 atau IP versi 6 yang menunjukkan alamat dari komputer tersebut pada jaringan Internet berbasis TCP/IP.

II.2.1.7.5 Pembagian Kelas IP

Alamat IP dibagi-bagi menjadi kelas-kelas yang masing-masing mempunyai kapasitas jumlah IP yang berbeda-beda.seperti terlihat pada tabel. Sebenarnya yang menjadi pembeda kelas IP versi 4 adalah pola biner yang terdapat dalam oktet pertama utamanya adalah bit-bit awal/*high-order bit*, tapi untuk lebih mudah mengingatnya, akan lebih cepat diingat dengan menggunakan representasi desimal.

Tabel II.2 pembagian kelas IP

Kelas Alamat IP	Oktet pertama (desimal)	Oktet pertama (biner)	Digunakan oleh
Kelas A	1–126	0xxx xxxx	Alamat *unicast* untuk jaringan skala besar.
Kelas B	128–191	10xx xxxx	Alamat *unicast* untuk jaringan skala menengah hingga skala besar.
Kelas C	192–223	110x xxxx	Alamat *unicast* untuk jaringan skala kecil.
Kelas D	224–239	1110 xxxx	Alamat *multicast* (bukan alamat *unicast*).
Kelas E	240–255	1111 xxxx	Direservasikan umumnya digunakan sebagai alamat percobaan (eksperimen) bukan alamat *unicast.*

Kelas A

Alamat-alamat kelas A diberikan untuk jaringan skala besar. Nomor urut bit tertinggi di dalam alamat IP kelas A selalu diset dengan nilai 0 (nol). Tujuh bit berikutnya untuk melengkapi oktet pertama akan membuat sebuah *network identifier*. 24 bit sisanya atau tiga oktet terakhir merepresentasikan *host identifier*. Ini mengizinkan kelas A memiliki hingga 126 jaringan, dan 16,777,214 *host* tiap jaringannya. Alamat dengan oktet awal 127 tidak diizinkan, karena digunakan untuk mekanisme Interprocess Communication (IPC) di dalam mesin yang bersangkutan.

Kelas B

Alamat-alamat kelas B dikhususkan untuk jaringan skala menengah hingga skala besar. Dua bit pertama di dalam oktet pertama alamat IP kelas B selalu diset ke bilangan biner10. 14 bit berikutnya untuk melengkapi dua oktet pertama, akan membuat sebuah *network identifier*. 16 bit sisanya dua oktet terakhir merepresentasikan *host identifier*. Kelas B dapat memiliki 16,384 *network*, dan 65,534 *host* untuk setiap *network*-nya.

Kelas C

Alamat IP kelas C digunakan untuk jaringan berskala kecil. Tiga bit pertama di dalam oktet pertama alamat kelas C selalu diset ke nilai biner 110. 21 bit selanjutnya untuk melengkapi tiga oktet pertama akan membentuk sebuah *network identifier*. 8 bit sisanya sebagai oktet terakhir akan merepresentasikan *host identifier*. Ini memungkinkan pembuatan total 2,097,152 buah *network*, dan 254 *host* untuk setiap *network*-nya.

Kelas D

Alamat IP kelas D disediakan hanya untuk alamat-alamat *IP multicast*, sehingga berbeda dengan tiga kelas di atas. Empat *bit* pertama di dalam IP kelas D selalu diset ke bilangan biner 1110. 28 *bit* sisanya digunakan sebagai alamat yang dapat digunakan untuk mengenali *host*.

Kelas E

Alamat IP kelas E disediakan sebagai alamat yang bersifat "eksperimental" atau percobaan dan dicadangkan untuk digunakan pada masa depan. Empat bit pertama selalu diset kepada bilangan biner1111. 28 bit sisanya digunakan sebagai alamat yang dapat digunakan untuk mengenali *host*.

II.3. Teknologi Pendukung

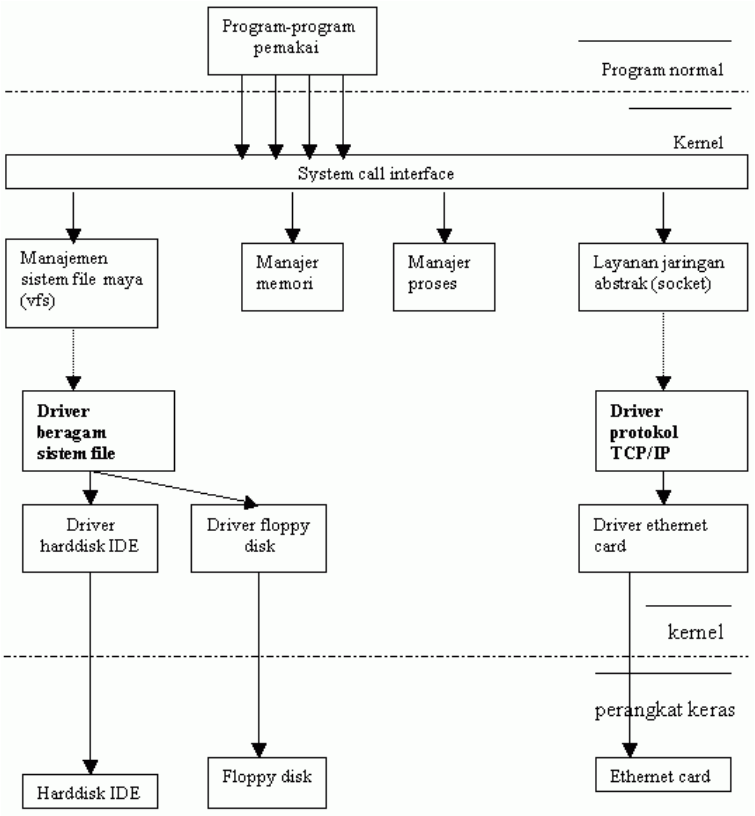
Dalam Tugas Akhir ini Teknologi pendukung adalah *hardware* maupun *software* yang akan digunakan dalam simulasi jaringan.

II.3.1. Linux

GNU/Linux adalah sistem operasi yang dibuat oleh Linus *BenedictTorvalds* dan disebarkan secara bebas di internet dimana orang lain bisa mengembangkan dan menggunakan untuk keperluannya sendiri. Namun, perlu dijelaskan bahwa GNU/Linux disini bisa bermakna ganda.Pertama, GNU/Linux berarti kernel linux. Pengertian kedua berarti sebuah sistem yang didalamnya sudah terdapat kernel, *shell* dan program pendukung lain yang siap di distribusikan dan dipakai. GNU/Linux dalam penelitian ini mengacu pada pengertian yang kedua.GNU/Linux adalah sistem operasi yang bebas dipakai,

didistribusikan dan dikembangkan kembali.Oleh karena itu, GNU/Linux mempunyai banyak varian yang lebih dikenal dengan istilah distro.

Sistem operasi GNU/Linux terdiri atas kernel linux (inti), program sistem, dan beberapa program aplikasi. Kernel linux merupakan inti dari sistem operasi. Program sistem dan semua program-program lainnya yang berjalan di atas kernel disebut*user mode*. Perbedaan antara program sistem dan program aplikasi adalah program sistem dibutuhkan agar suatu sistem operasi dapat berjalan sedangkan program aplikasi adalah program yang dibutuhkan untuk menjalankan suatu aplikasi tertentu.



Gambar II.17 Arsitektur SO Linux

II.3.2. Distro Linux

Distro adalah bundel dari kernel linux, beserta sistem dasar linux, program instalasi, *tools basic*, dan program-program lain yang bermanfaat sesuai dengan tujuan pembuatan distro. Menurut fungsi dan target penggunanya, distro dibagi menjadi dua, yaitu :

1. Distro Umum

   Distro umum adalah distro yang dipakai untuk keperluan umum.Distro jenis ini umumnya disebarkan dalam beberapa keping CD/DVD dengan paket *software* yang sangat lengkap. Distro ini bias dipakai untuk keperluan desktop, server ataupun untuk keperluan pengembangan *software*.

2. Distro Khusus

   Distro khusus adalah distro yang ditujukan untuk keperluan khusus seperti untuk keamanan jaringan, pemrograman atau untuk keperluan spesifik lainnya. Umumnya distro-distro ini berukuran kecil dan bias diinstall dimedia floppy *disk*, USB *flashdrive* atau didistribusikan dalam *mode Live* CD/DVD. Contoh distro jenis ini adalah damnsmall Linux, puppy Linux dan openSuSE.

3. Distro Ubuntu Linux

   Ubuntu adalah salah satu distribusi Linux yang berbasiskan pada Debian GNU/Linux.Proyek Ubuntu disponsori oleh Canonical Ltd (Perusahaan milik Mark Shuttleworth). Nama Ubuntu diambil dari nama sebuah konsep ideologi di Afrika Selatan. ”Ubuntu” berasal dari bahasa kuno Afrika,

yang berarti “Rasa perikemanusian terhadap sesame manusia”.Ubuntu juga bisa berarti “Aku adalah aku karena keberadaan kita semua”.Tujuan dari distribusi Linux Ubuntu adalah membawa semangat yang terkandung didalam Ubuntu ke dalam dunia perangkat lunak.

Ubuntu adalah sistem operasi lengkap berbasis GNU/Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional.Perihal kebebasan inilah yang membuat Ubuntu berbeda dari perangkat lunak berpemilik (*proprietary*).

II.3.3. *Ad-hoc Network*

Jaringan *Ad-hoc* merupakan jaringan *wireless* yang bersifat berdiri sendiri dalam membangun suatu jaringan Komputer. Jaringan berupa *Ad-hoc* karena setiap *node* memperbolehkan meneruskan data untuk *node* lainnya, penentuan *node-node* mana yang akan men*forward* data akan ditetapkan secara dinamis berdasarkan konektifitas jaringan. hal ini kontras dengan jaringan via kabel, *router* berperan dalam hal mencari jalan, dan hal ini kontras dalam mengatur infrastuktur jaringan nirkabel, dimana *node* spesial yang kita ketahui sebagai akses point bertugas mengatur komunikasi dengan *node-node* lainnya.

Jaringan *Ad-hoc* nirkabel yang terdesentralisasi membuat jaringan ini cocok digunakan untuk berbagai aplikasi tanpa *node* pusat, dan dapat meningkatkan skalabilitas jaringan *Ad-hoc* nirkabel dibandingkan *wirelessmanagednetwork*, walaupun secara teori maupun praktik batasan kapasitas untuk setiap jaringan telah diindentifikasi.

Konfigurasi yang minimal dan penyebaran yang cepat membuat jaringan *Ad-hoc* nirkabel cocok digunakan untuk situasi darurat contohnya bencana alam.Selain itu, *protocolrouting* dinamis dan adaptif juga membuat jaringan *Ad-hoc* nirkabel terbentuk secara cepat.*WirelessAd-hoc* network dapat diklasifikasikan sebagai berikut :

1. *mobile ad-hoc network* (MANETs)
2. *wirelessmesh network.*
3. *wireless sensor network.*

II.3.4. Nirkabel 802.11

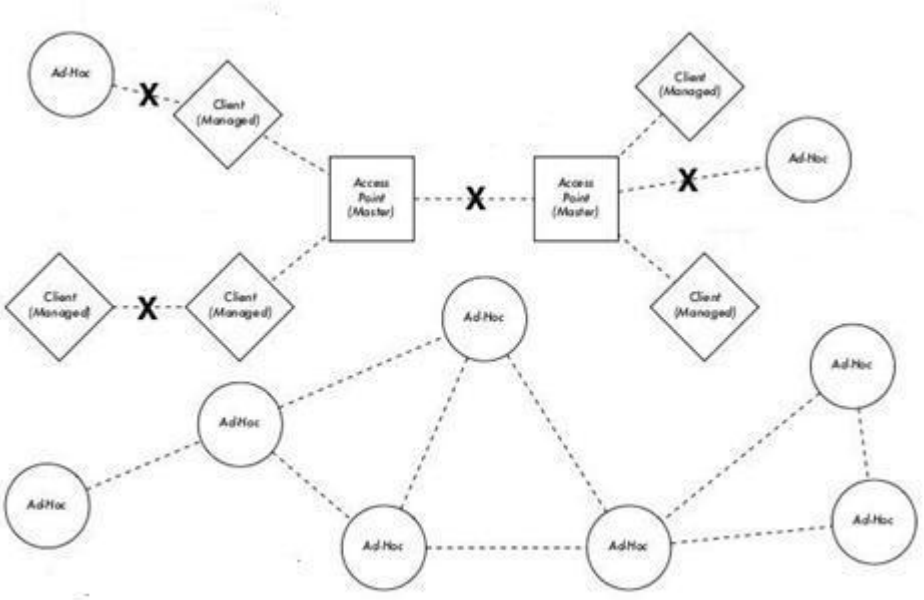
Nirkabel 802.11 adalah konektifitas fisik jaringan *wireless,* perangkat jaringan nirkabel harus beroperasi di frekuensi yang sama dari spektrum radio. Dalam hal ini berarti bahwa radio 802.11a akan berbicara dengan radio 802.11a di sekitar 5 GHz, dan 802.11b/g akan berbicara dengan radio 802.11b/g lainnya di sekitar 2,4 GHz. Tetapi 802.11a sebuah perangkat tidak dapat *interoperate* dengan perangkat 802.11b/g, karena mereka menggunakan bagian spektrum elektromagnetik yang berbeda.

Secara khusus, *cardwireless* harus setuju untuk menggunakan saluran yang sama. Jika sebuah *card* radio 802.11b diset ke saluran 2 sementara yang lain menggunakan saluran 802.11a, maka radio tersebut tidak dapat berkomunikasi satu sama lain. Ketika dua *cardwireless* yang dikonfigurasi untuk menggunakan protokol yang sama pada saluran radio yang sama, maka mereka siap untuk bernegosiasi konektivitas pada lapisan data link. Setiap perangkat 802.11a/b/g dapat beroperasi menggunakan salah satu dari empat kemungkinan *mode* :

Modus Master digunakan untuk memberikan layanan seperti jalur akses tradisional. *Card* nirkabel membuat jaringan dengan nama tertentu (***SSID***) dan kanal tertentu, dan menawarkan layanan untuk jaringan tersebut. Sementara dalam *master mode*, *card* nirkabel mengatur semua komunikasi yang berhubungan dengan jaringan (authenticating klien nirkabel, penanganan perebutan kanal, pengulangan paket, dll).*Cardwireless* pada *mode* master hanya dapat berkomunikasi dengan *card* yang terkait dengan itu di modus *managed*.

1. ***Modus Managed*** disebut sebagai ***modus klien***. *Card* nirkabel di modus Managed akan bergabung dengan jaringan yang diciptakan oleh master, dan secara otomatis akan menyesuaikan ke kanal yang digunakan master. Mereka kemudian mengirimkan data kepercayaan (credential) kepada master, dan jika data kepercayaan diterima, mereka dikatakan ***berasosiasi (associated)*** dengan master. *Card* dalam Modus *Managed* tidak berkomunikasi dengan satu sama lain secara langsung dan hanya akan berkomunikasi dengan master.
2. ***Modus ad-hoc***membuat jaringan multi*point*-to-multi*point* di mana tidak ada satu master *node* atau AP. Dalam modus *ad-hoc*, setiap *card* nirkabel berkomunikasi langsung dengan tetangga. *Node* harus dalam jangkauan satu sama lainnya untuk berkomunikasi, dan harus setuju pada nama jaringan (SSID) dan kanal yang digunakan.
3. ***Modus monitor*** digunakan oleh beberapa alat untuk dapat secara pasif mendengarkan trafik data yang lewat pada satu saluran radio tertentu. Pada *mode* monitor, *card* nirkabel tidak dapat transmit / mengirim data. Hal ini

berguna untuk menganalisis masalah pada sambungan nirkabel atau memerhatikan penggunaan spektrum di jaringan lokal. Modus monitor biasanya tidak digunakan untuk komunikasi.



Gambar II.18Klien, *and nodeAd-hoc.*

Ketika mengimplementasi sambungan *point*-to-*point* atau *point*-to-multi*point*, sebuah radio biasanya akan beroperasi dalam modus master, sedangkan yang lain beroperasi pada modus managed. Dalam jaringan multi*point*-to-multi*pointmesh*, semua radio beroperasi pada modus *ad-hoc* sehingga mereka dapat berkomunikasi satu sama lain secara langsung.

II.3.5. NMS(Network Management System)

NMS (*Network Management* Sistem*)* adalah kombinasi dari *hardware* dan *software* yang digunakan untuk memonitor dan mengatur jaringan. Setiap elemen jaringan di jaringan akan dikelola oleh sebuah elemen sistem manajemen. NMS merupakan suatu sistem yang bekerja untuk memonitor masalah *network* karena *overloadserver*, *server* yang rusak, koneksi *network*, atau *device* lain. NMS

bekerja sesuai kebutuhan yang diperlukan. Misalkan sistem akan memonitor HTTP, maka sistem itu akan mengirimkan HTTP request sesuai jadwal yang disetting. Contoh lain adalah jika sistem akan memonitor mail server, maka sistem akan mengirimkan request tertentu yang menggunakan protokol SMTP.

II.4. Aplikasi Pendukung

II.4.1. OpenNMS

OpenNMS adalah sebuah program untuk memonitor jaringan yang dikembangkan pada *software* yang bersifat *open source*. OpenNMS didukung oleh komunitas *open source* yang bergerak dibidang pelatihan atau servis komersial.

Tujuan nya adalah agar openNMS lebih terdistribusi, alat yang terukur untuk semua aspek dari *mode*l manajemen jaringan FCAPS, dan untuk membuat platform itu tersedia untuk kedua source dan aplikasi yang komersil.Semua kode telah terkoordinir dengan projek yang tersedia di bawah GNU (*General Public License*).

Berikut merupakan *service* yang diberikan oleh OpenNMS :

1. *Servis polling* – menyediakan servis yang tersedia dan latensi. Termasuk pengiriman paket dari servis yang tersedia dan latensi. Dan pelaporan pada servis.
2. *Data collection* – mengumpulkan, menyimpan dan melaporkan data yang telah terkumpul dari *node*melalui protokol-protokol yang mencakup SNMP, JMX, HTTP, dan NS*Client*.

3. *Treshodling* – evaluasi latensi data atau mengumpulkan performansi data yang *error*
4. *Event management* – menerima masukan baik internal maupun eksternal, termasuk SNMP

1. *Alarm and automation* – mengurangi masukan untuk pengurangan kunci dan skrip yang secara otomatis berada di pusat alarm
2. *Notification* – mengirim pesan berdasarkan perubahan yang terjadi *via email*, XMPP atau yang lainnya.

II.4.2. OLSR (*Opmitized Link State Routing*)

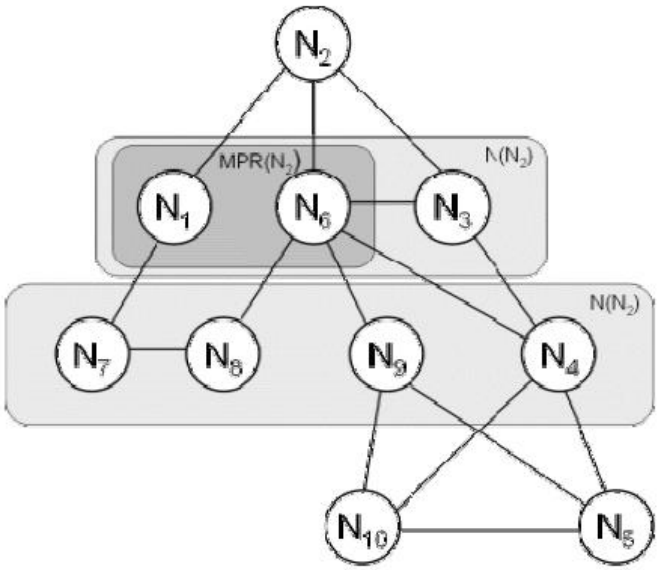
OLSR *(Opmitized Link State Routing)* merupakan *proactiverouting* dalam *network ad-hoc mobile*.Protocol ini memiliki stabilitas dan algoritma *Link State* dan memiliki keuntungan dengan *rute* yang tersedia dengan cepat ketika diperlukan. OLSR merupakan optmasi dari *protocol link state* klasik yang dikhususkan bagi pengguna *wireless network.*

OLSR memperkecil *overhead* yang terjadi pada pengiriman pesan pada *control traffic* dengan menggunakan beberapa *node* yang telah dipilih sebelumnya. Metode ini secara segnifikan mengurangi jumlah pengulangan transmisi yang diperlukan untuk mengirimkan pesan kepada semua *node* dalam jaringan.yang kedua, OLSR hanya memerluan sebagian *link state* untuk dikirimi pesan sehingga didapat rute path terpendek.

OLSR didesign untuk bekerja pada jaringan terdistribusi dan tidak bergantung pada *access point*.*Protocol* ini tidak memerlukan transmisi dari *control* pesan yang realible karena setiap *node* mengirimkan *protocol* pesan secara

periodik dan tidak terpengaruh dari *loss control* pesan yang terjadi. Dimana *loss* biasanya sering terjadi dalam *wirelessnetwork* karena *collision* maupun *problem* transmisi yang lain.

Selain itu, OLSR tidak memerlukan pengiriman secara *sequential*. Setiap *control* pesan mempunyai *sequencenumber* yang meningkat untuk setiap pesannya. Hal ini menyebabkan penerima dari *control* pesan dapat mengidentifikasikan informasi mana yang terbaru.



Gambar II.19*Routing* OLSR

Setiap *node*pada *network*, pada contoh gambar (2.1) adalah *node*N2 memilih beberapa *node*tetangga dalam *network*-nya.*Node-node*tersebut akan mengirim paket *node*N2. *Node*tetangga, yaitu N1 dan N6 disebut *Multipoint Relays*dari *node*N2. *Node*N2 memilih MPR-nya untuk meng-*cover* semua *node*yang berada dua *hop* jauhnya, yaitu N7, N8, N9, dan N4. Sebuah *node*yang bukan merupakan MPR dapat membaca paket yang dikirim dari N2 tapi tidak dapat meneruskannya.

II.4.2.1. Algoritma Prim

Algoritma Prim adalah sebuah algoritma dalam teori graf untuk mencari pohon rentang minimum untuk sebuah graf berbobot yang saling terhubung. Ini berarti bahwa sebuah himpunan bagian dari edge yang membentuk suatu pohon yang mengandung *node*, di mana bobot keseluruhan dari semua edge dalam pohon diminimalisasikan. Bila graf tersebut tidak terhubung, maka graf itu hanya memiliki satu pohon rentang minimum untuk satu dari komponen yang terhubung. Algoritma ini ditemukan pada 1930 oleh matematikawan Vojtěch Jarník dan kemudian secara terpisah oleh computer scientist Robert C. Prim pada 1957 dan ditemukan kembali oleh Dijkstra pada 1959. Karena itu algoritma ini sering dinamai algoritma DJPatau algoritma Jarnik.

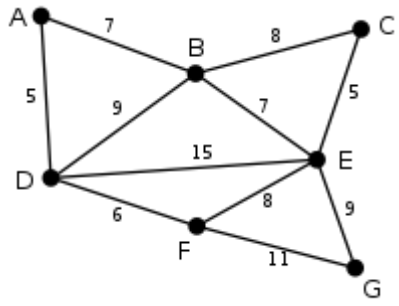
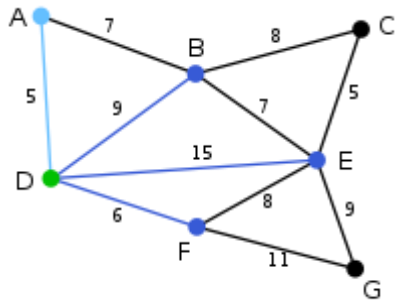
Langkah-langkahnya adalah sebagai berikut:

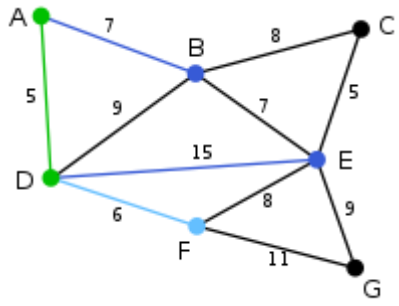
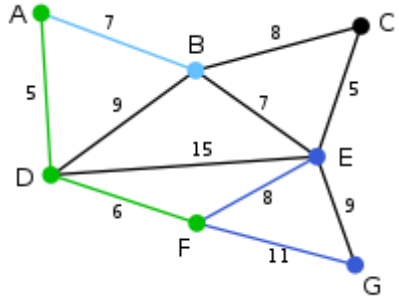
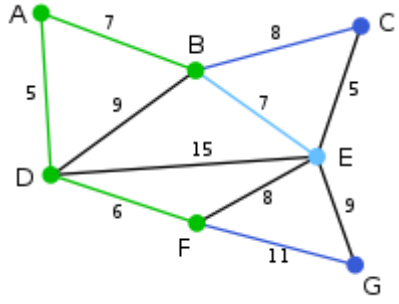
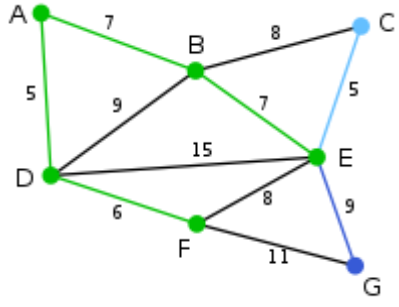
1. buat sebuah pohon yang terdiri dari satu *node*, dipilih secara acak dari graf
2. buat sebuah himpunan yang berisi semua cabang di graf
3. loop sampai semua cabang di dalam himpunan menghubungkan dua *node* di pohon
    a. hapus dari himpunan satu cabang dengan bobot terkecil yang menghubungkan satu *node* di pohon dengan satu *node* di luar pohon
    b. hubungkan cabang tersebut ke pohon

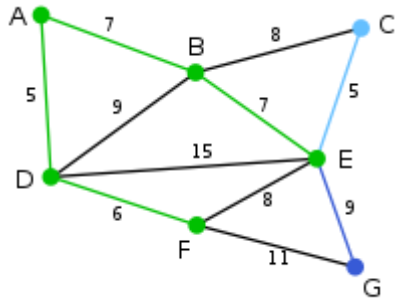
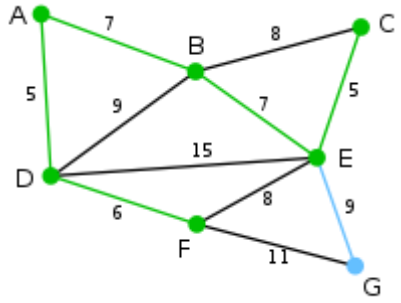
Dengan struktur data binary heap sederhana, algoritma Prim dapat ditunjukkan berjalan dalam waktu O(*E*log *V*), di mana E adalah jumlah cabang dan V adalah jumlah *node*. Dengan Fibonacci heap, hal ini dapat ditekan menjadi

O(E + Vlog V), yang jauh lebih cepat bila grafnya cukup padat sehingga E adalah Ω(Vlog V).

Tabel II.3 Contoh Penyelesaian dengan Algoritma Prim

	Ini adalah graf berbobot awal. Graf ini bukan pohon karena ada circuit. Nama yang lebih tepat untuk diagram ini adalah graf atau jaringan. Angka-angka dekat garis penghubung adalah bobotnya. Belum ada garis yang ditandai, dan *node* **D** dipilih secara sembarang sebagai titik awal.
	Ini adalah graf berbobot awal. Graf ini bukan pohon karena ada circuit. Nama yang lebih tepat untuk diagram ini adalah graf atau jaringan. Angka-angka dekat garis penghubung adalah bobotnya. Belum ada garis yang ditandai, dan *node* **D** dipilih secara sembarang sebagai titik awal.

	*Node* berikutnya yang dipilih adalah yang terdekat dari **D** *atau* **A**. **B** jauhnya 9 dari **D** dan 7 dari **A**, **E** jauhnya 15, dan **F** 6. 6 adalah yang terkecil, jadi kita tandai *node* **F** dan cabang **DF**.
	Algoritma ini berlanjut seperti di atas. *Node* **B**, yang jauhnya 7 dari **A**, ditandai. Di sini, cabang **DB** ditandai merah, karena baik *node* **B** dan *node* **D**telah ditandai hijau, sehingga DB tidak dapat digunakan.
	Dalam hal ini, kita dapat memilih antara **C**, **E**, dan **G**. **C** jauhnya 8 dari **B**, **E** 7 dari **B**, dan **G** 11 dari **F**. **E** yang terdekat, jadi kita tandai *node* **E** dan cabang **EB**. Dua cabang lain ditandai merah, karena kedua *node* yang terhubung telah digunakan.
	*Node* **G** adalah satu-satunya yang tersisa. Jauhnya 11 dari **F**, dan 9 dari **E**. **E** lebih dekat, jadi kita tandai cabang **EG**. Sekarang semua *node* telah terhubung, dan pohon rentang

	minimum ditunjukkan dengan warna hijau, bobotnya 39.
	Di sini, *node* yang tersedia adalah **C** dan **G**. **C** jauhnya 5 dari **E**, dan **G** 9 dari **E**. **C** dipilih, jadi ditandai bersama dengan cabang **EC**. Cabang **BC** juga ditandai merah.
	*Node* **G** adalah satu-satunya yang tersisa. Jauhnya 11 dari **F**, dan 9 dari **E**. **E** lebih dekat, jadi kita tandai cabang **EG**. Sekarang semua *node* telah terhubung, dan pohon rentang minimum ditunjukkan dengan warna hijau, bobotnya 39.

Misalkan *P* adalah sebuah graf terhubung berbobot. Pada setiap iterasi algoritma Prim, suatu cabang harus ditemukan yang menghubungkan sebuah node di graf bagian ke sebuah node di luar graf bagian. Karena *P* terhubung, maka selalu ada jalur ke setiap node. Keluaran *Y* dari algoritma Prim adalah sebuah pohon, karena semua cabang dan node yang ditambahkan pada *Y* terhubung. Misalkan Y_1 adalah pohon rentang minimum dari P. Bila $Y_1=Y$ maka *Y* adalah pohon rentang minimum. Kalau tidak, misalkan *e* cabang pertama yang ditambahkan dalam konstruksi *Y* yang tidak berada di Y_1, dan *V* himpunan semua

node yang terhubung oleh cabang-cabang yang ditambahkan sebelum *e*. Maka salah satu ujung dari *e* ada di dalam*V* dan ujung yang lain tidak. Karena Y_1 adalah pohon rentang dari *P*, ada jalur dalam Y_1 yang menghubungkan kedua ujung itu. Bila jalur ini ditelusuri, kita akan menemukan sebuah cabang *f* yang menghubungkan sebuah node di *V* ke satu node yang tidak di *V*. Pada iterasi ketika *e* ditambahkan ke *Y*, *f* dapat juga ditambahkan dan akan ditambahkan alih-alih *e* bila bobotnya lebih kecil daripada *e*. Karena *f* tidak ditambahkan, maka kesimpulannya

$$w(f) \geq w(e).$$

Misalkan Y_2 adalah graf yang diperoleh dengan menghapus *f* dan menambahkan *e* dari Y_1. Dapat ditunjukkan bahwa Y_2 terhubung, memiliki jumlah cabang yang sama dengan Y_1, dan bobotnya tidak lebih besar daripada Y_1, karena itu ia adalah pohon rentang minimum dari *P* dan ia mengandung *e* dan semua cabang-cabang yang ditambahkan sebelumnya selama konstruksi *V*. Ulangi langkah-langkah di atas dan kita akan mendapatkan sebuah pohon rentang minimum dari *P* yang identis dengan *Y*. Hal ini menunjukkan bahwa *Y* adalah pohon rentang minimum.

II.4.2.2. *Djikstra*

Algoritma Dijkstra diterapkan untuk mencari lintasan terpendek pada graf berarah.Namun, algoritma ini juga benar untuk graf tak berarah.Algoritma Dijkstra mencari lintasan terpendek dalam sejumlah langkah.Algoritma ini menggunakan prinsip greedy.Prinsip greedy pada algoritma dijkstra menyatakan

bahwa pada setiap langkah kita memilih sisi yang berbobot minimum dan memasukannya dalam himpunan solusi.

Misalkan sebuah graf berbobot dengan n buah simpul dinyatakan dengan matriks M=[mij], yang dalam hal ini:

mij = bobot sisi (i,j) *(pada graf tak berarah mij =mji)*

mii = 0

mij = ∞ , jika tidak ada sisi dari simpul I ke simpul j

Selain matriks M, juga menggunakan tabel S=[si], yang dalam hal ini:

si = 1, jika simpul i termasuk ke dalam lintasan terpendek

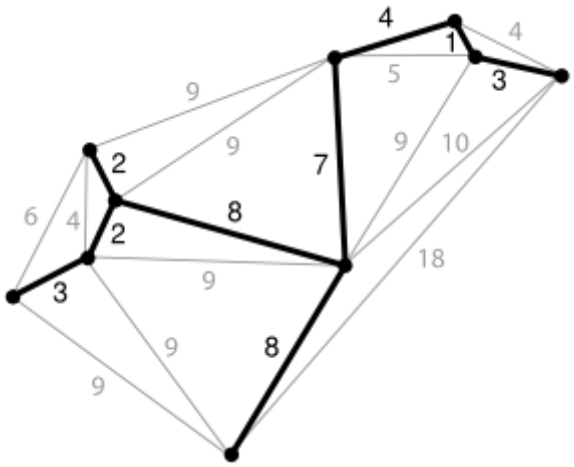
si = 0, jika simpul i tidak termasuk ke dalam lintasan terpendek

dan tabel D=[di], yang dalam hal ini

di = panjang lintasan dari simpul awal a ke simpul i.

II.4.2.3. *Minimun Spanning Tree*

Minimum spanning tree adalah suatu pohon yang dapat didefinisikan dengan sebuah graf. Grafberarah dan graf tidak berarah adalah subgraf yang setiap node/simpulnya terkoneksi satu sama lain. Sebuah graf, dapat memberikan pohon rentang yang berbeda. Pada setiap ruas/edge, kita dapat memberikan suatu bobot untuk menentukan suatu nilai. Setiap bobot tersebut akan dibandingkan dengan bobot yang lain yang mengarah pada simpul berikutnya, selanjutnya akan dipilih bobot yang terkecil. Hal ini akan terus dilakukan sampai menuju simpul tujuan. Ini yang disebut dengan minimum spanning tree.



Gambar II.20 *Minimun Spanning Tree*

Gambar II.2 adalah graf berbobot awal. Pada awalnya graf tersebut bukan pohon karena terdapat sirkuit. Setelah penjelasan sebelumnya dilakukan, maka akan terdapat suatu jalur dan itu disebut minimum spanning tree. Salah satu contohnya adalah perusahaan TV kabel yang memasang kabel ke lingkungan baru. Jika dibatasi untuk mengubur kabel hanya di sepanjang jalan tertentu, maka ada graf yang mewakili suatu jalur. Beberapa jalur tersebut mungkin akan lebih lama, karena memerlukan kabel yang akan dikubur lebih dalam, sehingga membutuhkan biaya yang lebih mahal. Dengan adanya minimum spanning tree, maka akan didapatkan total biaya yang lebih rendah.

Kemungkianan Keseragaman

Dalam suatu graf memungkinkan adanya kesamaan bobot pada setiap ruas untuk menuju simpul tujuan. Dengan ini, maka akan adanya perbedaan jalur, tetapi stiap jalur tersebut memiliki total bobot yang sama.

Keunikan

Jika setiap ruas memiliki bobot yang berbeda, maka akan ada satu minimum spanning tree yang unik. Hal ini dapat dibuktikan dengan induksi atau kontradiksi. Hal ini berlaku dalam situasi yang realistis, seperti contoh perusahaan TV kabel di atas, dimana tidak ada dua jalur yang memiliki biaya yang sama persis.

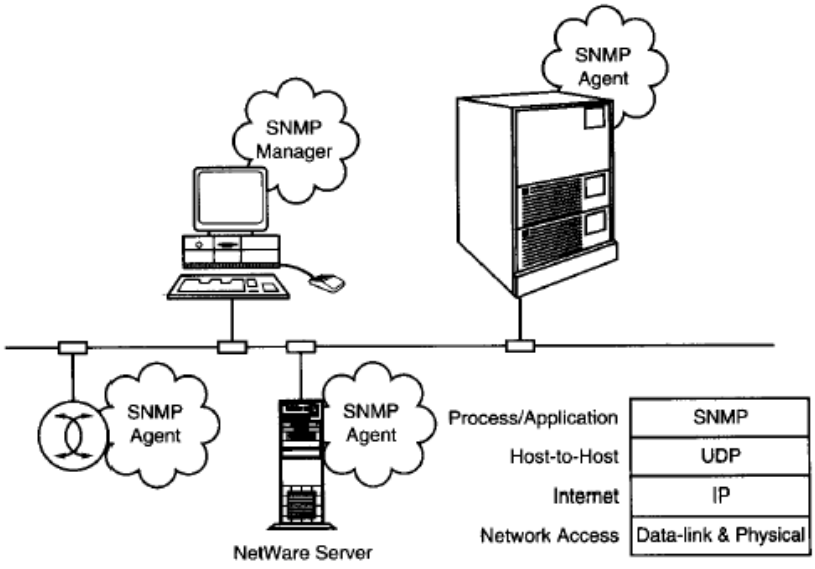
Minimum Spanning tree juga dapat disebut sebagai biaya graf minimum. Sebagai contoh, apabila kita akan menuju suatu tempat dengan banyak jalur yang ada, kita akan membutuhkan biaya yang mahal untuk mencapai suatu tempat tersebut dengan jalur yang tidak optimal. Dengan adanya minimum spanning tree ini, kita dapat mengetahui jalur mana yang harus dilewati sehingga hanya memerlukan biaya yang murah, dan waktu yang lebih cepat.

Sirkuit

Pada dasarnya dalam sebuah pohon, tidak diperkenankan adanya sirkuit. Untuk mendapatkan minimum spanning tree maka akan ditelusuri suatu jalur yang memiliki bobot terkecil dengan menghindari adanya sirkuit.

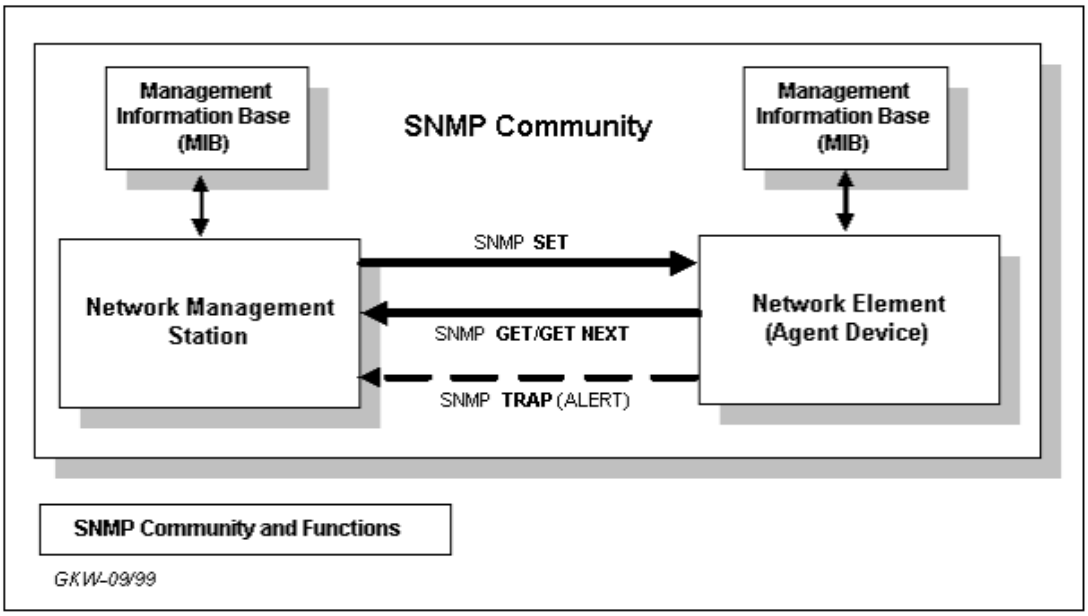
II.4.3. SNMP (*Simple Network Management Protocol*)

SNMP merupakan sistem yang dikembangkan untuk menyediakan peralatan manajemen jaringan yang mendasar dan yang mudah diimplementasikan untuk rangkaian protokol TCP/IP. Pada gambar dibawah ditunjukkan bahwa SNMP *manager* adalah suatu *host* khusus dalam jaringan yang dapat dipergunakan untuk mengirimkan pertanyaan yang berkisar untuk manajemen jaringan kepada *host/node*yang lain.



Gambar II.21 Arsitektur pemakai SNMP

*Node*yang dikelola bisa berupa *Bridge, Router atau Host*, yang masing masing menjalankan program yang disebut dengan SNMP *agent*.Program SNMP *agent* ini menangkap pertanyaan yang dikirim oleh SNMP *manager* dan mengirimkan kembali informasi yang diinginkan. SNMP menggunakan UDP sebagai protokol transport untuk mengirimkan pertanyaan dan menerima jawaban dari SNMP *agent*. SNMP merupakan bagian dari *Application layer* pada struktur protokol TCP/IP.*Simple Network Management Protocol* (SNMP) memudahkan seorang *administrator* dalam mengatur performansi jaringan, mencari dan menyelesaikan masalah dalam suatu jaringan.Dan membuat rencana perubahan dalam jaringan.Elemen *network management* terdiri dari NMS (*Network Management Station*) yang merupakan pusat pengendali untuk pengumpulan dan analisa data *network management.*



Gambar II.22 Interaksi NMS dengan Agen

II.4.4. *Imagemagick*

Imagemagick merupakan paket *software* yang digunakan untuk memanipulasi gambar dan menampilkan gambar serta mendukung lebih dari ratusan format gambar.*Imagemagick* biasanya digunakan untuk melakukan berbagai jenis operasi transformasi dan konversi pada format gambar.*Imagemagick* dirilis di bawah lisensi *free software*, sehingga bersifat bebas untuk digunakan.Pada tugas akhir ini, *Imagemagick* digunakan untuk menampilkan gambar peta jaringan yang merupakan hasil eksekusi *script* bahasa *perl*.*Imagemagick* dipilih karena selain bersifat *free*, juga mendukung *flatform Linux.*

II.4.5. *Graphviz*

Graphviz (*Graph Visualization*) merupakan *software*yang digunakan untuk menampilkan informasi struktural, seperti diagram dari suatu grafik ataupun peta jaringan.*Graphviz* merupakan *software*gratis untuk memvisualisasikan

gambar.*Graphviz* memiliki beberapa program *layout* grafik.Program *layout* grafik *Graphviz* mendeskripsikan suatu grafik berdasarkan bahasa teks yang sederhana, dan membuat grafik sederhana dengan format yang sudah umum digunakan untuk *website*, seperti misalnya JPEG dan SVG, PDF atau menampilkan gamar yang interaktif.Pada tugas akhir ini, *graphviz* juga digunakan sebagai pendukung untuk menampilkan gambar hasil visualisasi dan *Imagemagick* bersama-sama diperlukan untuk menampilkan *node-node*dan juga *link* yang berhubungan antar *node*pada jaringan *wirelessmesh*ini.