

BAB 2

LANDASAN TEORI

2.1 Teori Umum

2.1.1 Pengertian Jaringan Komputer

Jaringan komputer dapat diartikan sebagai dua atau lebih komputer beserta perangkat-perangkat lain yang dihubungkan agar dapat saling berkomunikasi dan bertukar informasi, sehingga membantu menciptakan efisiensi dan optimasi dalam kerja. (Norton, 1999, p5)

2.1.2 Jenis – Jenis Jaringan Komputer

2.1.2.1 LAN (*Local Area Network*)

Jaringan area local atau *Local Area Network* (LAN) merupakan inti dari jaringan komputer, dimana sejumlah komputer dapat berhubungan satu dengan yang lain serta dapat menggunakan secara bersama sumber daya (*resource*) seperti *printer* atau *hardisk* (Wijaya, 2003, p1).

LAN merupakan sistem komunikasi data yang mengizinkan sejumlah *device* saling berkomunikasi secara langsung satu dengan yang lainnya, dalam sebuah cakupan area geografis yang terbatas (Forouzan, 2003, p4)

Sistem LAN yang sering digunakan adalah sistem *Ethernet* yang dikembangkan oleh Perusahaan Xerox. Penggunaan *node intermediate* (seperti *repeater*, *bridge*, dan *switch*) memungkinkan LAN terkoneksi

membentuk jaringan yang lebih luas. LAN juga dapat terkoneksi ke LAN, WAN (*Wide Area Network*), atau MAN (*Metropolitan Area Network*) lain menggunakan *router*.

Akses ke jaringan gedung lain dalam kantor dikontrol melalui sebuah *router*. *Switch / router* lain mungkin digunakan untuk koneksi ke perlengkapan lain guna menguji konektivitas LAN. Hubungan *Ethernet* menghubungkan beberapa *workstation*.

2.1.2.2 MAN (*Metropolitan Area Network*)

Metropolitan Area Network (MAN) yaitu jaringan komputer yang saling terkoneksi dalam satu kawasan kota yang jaraknya bisa lebih dari 1 Km, ini merupakan pilihan untuk membangun jaringan kantor dalam suatu kota (Sarosa dan Anggoro, 2000).

Metropolitan Area Network (MAN) pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya memakai teknologi yang sama dengan LAN. MAN dapat mencakup kantor – kantor perusahaan yang berdekatan dan dapat dimanfaatkan untuk keperluan pribadi (*Swasta*) atau umum. MAN biasanya mampu menunjang data dan suara, dan bahkan dapat berhubungan dengan jaringan televisi kabel. MAN hanya memiliki satu atau dua buah kabel dan tidak mempunyai elemen *switching*, yang berfungsi untuk mengatur paket melalui beberapa *output* kabel. Adanya elemen *switching* membuat rancangan menjadi lebih sederhana.

Alasan utamanya memisahkan MAN sebagai kategori khusus adalah telah ditentukannya standart untuk MAN, dan Standart ini sekarang sedang diimplementasikan.

2.1.2.3 WAN (*Wide Area Network*)

Wide Area Network (WAN) mencakup daerah geografis yang luas, seringkali mencakup sebuah negara atau benua. WAN terdiri dari kumpulan mesin yang bertujuan untuk menjalankan program-program aplikasi.

Menurut <http://id.wikipedia.org>, WAN atau *Wide Area Network* merupakan jaringan komputer yang mencakup area yang besar. Sebagai contoh yaitu jaringan komputer antar wilayah, kota atau bahkan negara, atau dapat didefinisikan juga sebagai jaringan komputer yang membutuhkan *router* dan saluran komunikasi publik.

WAN digunakan untuk menghubungkan jaringan lokal yang satu dengan jaringan lokal yang lain, sehingga pengguna atau komputer di lokasi yang satu dapat berkomunikasi dengan pengguna dan komputer di lokasi yang lain.

Jaringan WAN menghubungkan beberapa WAN dari beberapa kota atau negara yang berbeda. WAN biasanya terhubung via satelit. WAN mempunyai daerah yang sangat luas dan menggunakan siklus komunikasi yang menghubungkan *node-node intermediate*. Kecepatan transmisinya beragam dari 2 Mbps, 34 Mbps, 45 Mbps, 15 Mbps, sampai 625 Mbps (atau kadang-kadang lebih). Faktor khusus yang

mempengaruhi desain dan *performance*-nya terletak pada siklus komunikasi, seperti jaringan telepon, satelit, atau komunikasi pembawa lain yang digunakannya.

Ciri-ciri dari WAN adalah adanya penekanan pada fasilitas transmisi sehingga komunikasi dapat berjalan efisien. Sangatlah penting untuk mengontrol jumlah lalu lintas data dan mencegah *delay* yang berlebihan. Karena topologi WAN lebih kompleks dari MAN maka algoritma rute aliran data juga menjadi perhatian.

WAN menjadi sangat luas, membentang ke seluruh dunia. Organisasi yang mendukung WAN dikenal sebagai NSP (*Network Service Provider*). NSP merupakan bagian inti dari *internet*.

WAN yang dikoneksikan melalui NSP bersama-sama terkoneksi membentuk suatu jaringan *internet* yang bersifat *global*. NSP yang ada bervariasi dalam kecepatan transmisi data, *transit delay*, dan konektivitas yang diberikan.

2.1.3 Arsitektur Jaringan

2.1.3.1 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 (misalkan A, B, C, D dan E) yang memberi hak akses terhadap *file* yang dimilikinya. Pada satu saat A mengakses *file share* 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*.

2.1.3.2 Client – Servers

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 *web server*. Atau bisa juga banyak *service* / layanan yang diberikan oleh satu komputer. Contohnya adalah *server* jtk.polban.ac.id yang merupakan satu komputer dengan *multi service* yaitu *mail server*, *web server*, *file server*, *database server* dan lainnya.

2.1.4 Topologi Jaringan Komputer

2.1.4.1 Physical Topology

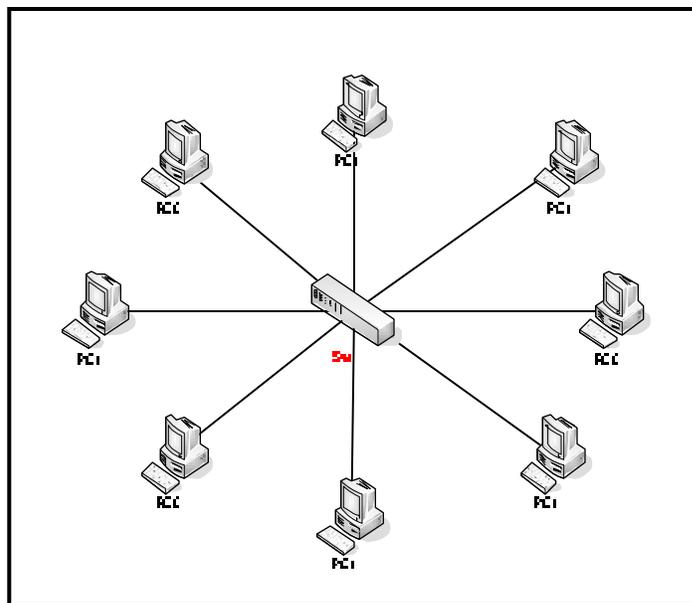
2.1.4.1.1 Star

Pada topologi *star*, semua komputer terhubung ke *hub* / *switch* pusat yang mengontrol komunikasi jaringan dan dapat berkomunikasi dengan *hub* / *switch* lain. Batas jarak komputer dengan *hub* / *switch* sekitar 100 meter. Setiap node pada jaringan akan berkomunikasi melalui node pusat atau konsentrator terlebih dahulu sebelum menuju *server*. Jaringan

star lebih fleksibel dan luas dibandingkan topologi lainnya. Keunggulan tipe star adalah jika salah satu node putus maka tidak akan mempengaruhi kinerja jaringan yang lain.

Topologi *logical* yang mendukung topologi Star ini adalah *Ethernet, Broadcast*.

Berikut ini adalah gambar dari topologi *Star* :



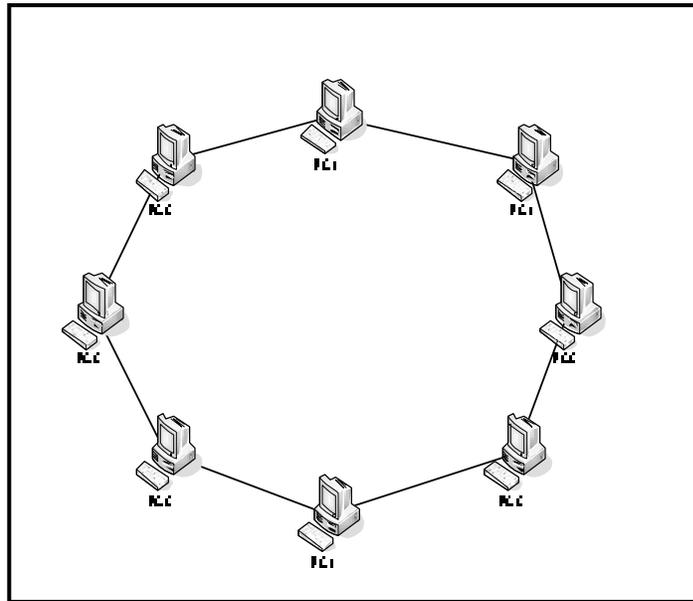
Gambar 2.1 Topologi Star

2.1.4.1.2 Ring

Pada topologi *ring* (cincin), jaringan membentuk *loop* tertutup sehingga mengesankan cincin tanpa ujung. Komputer terkoneksi satu dengan yang lain seperti sebuah cincin. Sinyal akan mengalir satu arah sehingga dapat menghindari terjadinya tabrakan paket data. Namun, apabila ada salah satu

komputer yang putus akan tetap mempengaruhi ke seluruh jaringan. Topologi *logical* yang mendukung topologi ring ini adalah *token-passing / token ring*.

Berikut ini adalah gambar dari topologi *Ring* :



Gambar 2.2 Topologi Ring

2.1.4.1.3 Bus

Bus topology banyak disebut dengan *Common Bus Multipoint Technology*. Adalah sebuah jaringan komputer yang menggunakan 1 kabel yang digunakan dari satu akhir dari jaringan komputer ke komputer lainnya.

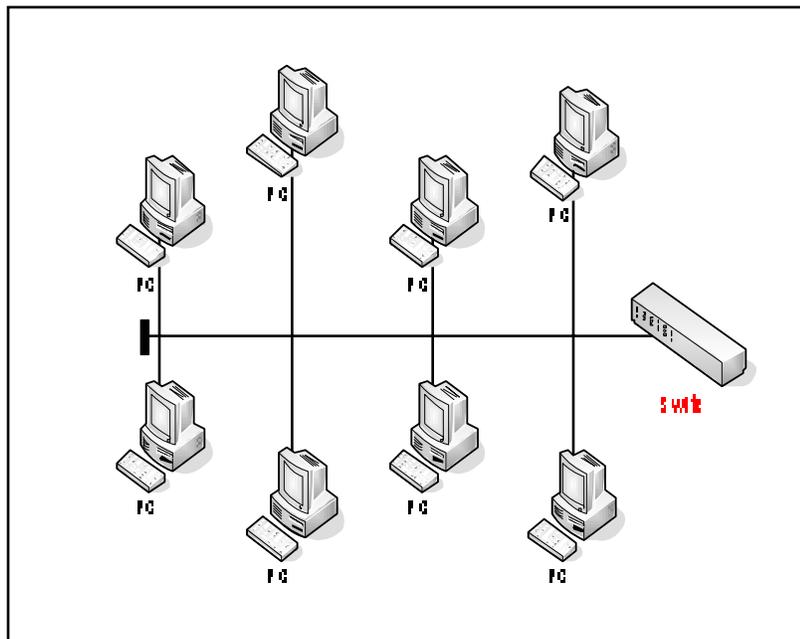
Pada topologi *bus*, kedua ujung jaringan harus diakhiri dengan sebuah *terminator*. *Barel connector* dapat digunakan untuk memperluas jaringan. Instalasi jaringan *bus* sangat sederhana, murah dan maksimal terdiri atas 5 – 7 komputer.

Kesulitan yang sering *dialami* adalah terjadinya tabrakan data karena mekanisme jaringan relatif sederhana dan jika salah satu *node* putus maka akan mengganggu kinerja dan trafik seluruh jaringan.

Biasanya topologi *Bus* menggunakan kabel *coaxial*. Seiring dengan perkembangan teknologi, topologi *bus* juga bisa menggunakan media *fiber optic* dan biasanya digunakan untuk *backbone* jaringan.

Topologi *logical* yang mendukung topologi *Bus* ini adalah *Ethernet, Broadcast*.

Berikut ini adalah gambar dari topologi *Bus* :



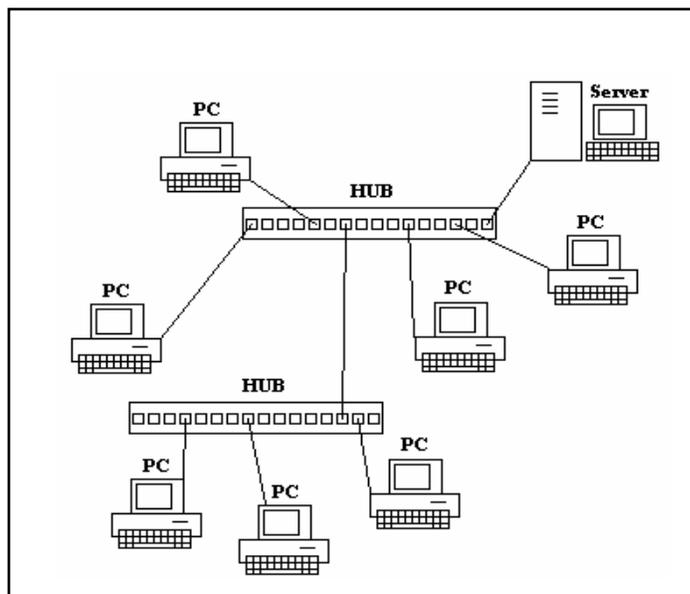
Gambar 2.3 Topologi Bus

2.1.4.1.4 *Extended Star Topology*

Extended Star Topology menggabungkan beberapa topologi *star* menjadi satu. *Hub / Switch* yang dipakai untuk menghubungkan beberapa komputer pada satu jaringan dengan menggunakan topologi *star*, akan dihubungkan lagi ke *hub / switch* utama.

Topologi *logical* yang mendukung topologi *Extended Star* ini adalah Ethernet, Broadcast.

Berikut ini adalah gambar dari *Extended Star* :



Gambar 2.4 Extended Star

2.1.4.2 *Logical Topology*

2.1.4.2.1 *Ethernet*

Teknologi *ethernet* adalah teknologi yang paling umum untuk teknologi jaringan LAN. *Ethernet* dirancang untuk

jaringan yang tidak membutuhkan kecepatan tinggi, berjarak dekat, jaringan komputer dalam ruangan. Cara kerjanya adalah *broadcast* (menyebarkan data ke seluruh komputer). Saat sebuah *host* mengirimkan data ke sebuah komputer maka seluruh komputer yang ada dalam jaringan tersebut akan menerima data tersebut. Tetapi hanya komputer yang dituju saja yang memproses data tersebut. Dengan keadaan seperti ini akan memungkinkan terjadinya tabrakan data saat dua buah *host* atau lebih mengirimkan data secara bersamaan. Hal ini dapat diatasi dengan sebuah mekanisme deteksi tabrakan dan pemulihan yang disebut dengan CSMA / CD (*Carrier Sense Multiple Access / Collision Detection*). Topologi fisik yang didukung *ethernet* adalah semua topologi kecuali *ring*.

2.1.4.2.2 Broadcast

Topologi ini mengartikan bahwa setiap *host* yang mengirim *packet* akan mengirimkan *packetnya* ke semua *host*.

2.1.4.2.3 Token – Pasing

Mengendalikan akses jaringan dengan mempass-kan sebuah *token* elektronik yang secara sekuensial akan melalui masing-masing anggota dari jaringan komputer. Ketika sebuah komputer mendapatkan *token* tersebut, berarti komputer tersebut diperbolehkan mengirim data. Jika komputer tersebut

tidak memiliki data yang akan dikirim, maka *token* akan dilewatkan ke komputer berikutnya.

2.1.4.2.4 FDDI

Teknologi ini adalah teknologi yang paling mahal dan canggih dibandingkan *token-passing* dan *ethernet*. FDDI dirancang untuk jaringan yang berkecepatan sangat tinggi. Biasanya FDDI ditaruh pada jaringan *backbone* untuk menangani arus data yang sibuk. Cara kerjanya hampir sama dengan *token ring* tetapi kelebihanannya FDDI memiliki jalur cadangan seandainya jalur utama putus. Topologi fisik yang didukung FDDI adalah topologi *dual ring*.

2.1.5 Perangkat Jaringan

2.1.5.1 NIC

Network Interface Card / NIC atau *LAN card* adalah sebuah komputer *Circuit Board* atau kartu yang terpasang pada sebuah komputer sehingga komputer tersebut mampu terkoneksi pada jaringan. NIC menyediakan antarmuka ke media. NIC bisa menggunakan *transceiver eksternal* atau melalui *transceiver internal* terintegrasi yang dipasang pada *Network Interface Card PCB*. NIC biasanya dilengkapi dengan *protocol control firmware* dan *ethernet controller* yang dibutuhkan untuk mendukung protokol *data-link Medium Access Control* (MAC) yang digunakan oleh *ethernet*.

Pada komputer *personal* dan *workstation* dalam suatu *Local Area Network* biasanya terpasang *Network Interface Card* yang didesain khusus untuk transmisi data pada *Local Area Network*. Teknologi pada *Local Area Network card* yang umumnya digunakan adalah *ethernet / token ring*.

2.1.5.2 Repeater

Repeater adalah perangkat yang beroperasi di *layer* fisik. *Repeater* menerima sinyal, dan sebelum sinyal menjadi lemah atau rusak, maka *repeater* akan membangkitkan pola-pola *bit*, kemudian *repeater* akan meneruskan sinyal yang telah diperbaiki. *Repeater* dapat meningkatkan panjang LAN secara fisik dan dapat berfungsi menghubungkan bagian-bagian dari LAN. *Repeater* juga akan meneruskan setiap *frame* yang dikirim, dan tidak memiliki kemampuan untuk menyaring setiap *frame*. Fungsi *repeater* adalah sebagai pembangkit ulang atau *regenerator* dan bukan penguat (*amplifier*).

2.1.5.3 Hub

Hub adalah alat yang berfungsi sebagai tempat untuk menerima data dan meneruskannya menuju tempat yang lainnya. *Hub* terdiri dari *port-port* RJ-45 *female* sehingga kabel - kabel *twisted pair* yang sudah terpasang konektor RJ-45 pada ujung - ujungnya dapat dengan mudah dihubungkan pada *Hub*.

Hub digunakan pada topologi *star* dan dapat diparalel hingga maksimum tiga buah *Hub*. *Port* yang dimiliki *Hub* bervariasi mulai dari *5 Port*, *10 Port*, *12 Port*, *16 Port*, *24 Port*, *32 Port* dan yang terbanyak *64 Port*. Untuk jaringan komputer yang menggunakan *Hub* yang diparalel lebih dari 3 *Hub*, maka diperlukan sebuah *Router* untuk menghubungkan *Hub* yang keempat dan seterusnya.

Semua *network connections* yang terhubung ke *Hub* secara langsung akan men-*share* sebuah *single collision domain*, dimana jika sebuah *client* mengirim data, maka semua *client* yang terhubung ke *Hub* tersebut akan menerima data yang dikirim oleh pengirim.

2.1.5.4 Bridge

Bridge adalah alat yang menghubungkan sebuah LAN dengan LAN yang lain dengan teknologi yang sama (misalkan pada dua LAN yang menggunakan teknologi ethernet yang sama). *Bridge* akan memisahkan data mana yang harus dikirim pada LAN-nya sendiri atau dengan LAN lain yang terhubung dengannya.

Bridge bekerja pada level *data-link layer* (*physical network*) di suatu jaringan. *Bridge* mengkopi *frame* data dari suatu jaringan ke jaringan yang lain sepanjang jalur komunikasi terjalin. Beberapa produk dari *bridge* dapat berfungsi juga sebagai *router* sehingga dikenal dengan istilah *brouter* (*bridge-router*).

2.1.5.5 Switch

Switch berada pada *layer* fisik dan *data link*. *Switch* adalah *bridge* yang memungkinkan kinerja lebih cepat. Perbedaan *bridge* dengan *switch* adalah pada *switch* terdapat banyak *port* yang spesifik untuk masing-masing *node*, sehingga tidak terjadi *collision* dalam jaringan.

2.1.5.6 Router

Pada jaringan yang besar, seperti *internet*, diperlukan adanya *router*. *Router* dapat berupa suatu alat (*dedicated*) atau berupa suatu aplikasi. Aplikasi *router* ter-*install* di sebuah komputer *personal* sehingga sering dikenal dengan istilah *PC-router*. *Router* berfungsi untuk memutuskan pada titik manakah paket data harus diteruskan. *Router* dapat tersambung pada dua jaringan atau lebih dan dapat memutuskan jaringan mana yang akan menerima paket data yang diteruskan oleh *router*.

Router umumnya terletak pada *gateway* pada suatu jaringan. *Router* memiliki apa yang dinamakan *routing table*, yaitu sebuah daftar dari rute - rute yang tersedia dan mampu memilih rute terbaik untuk sebuah paket data. Fungsi *routing* merupakan standar *Layer 3* (*Network Layer*) pada model OSI.

Pada jaringan yang menggunakan koneksi *internet* berkecepatan tinggi seperti DSL, *router* berfungsi ganda sebagai *firewall*. Produsen *router* yang terkenal antara lain adalah 3Com dan Cisco.

2.1.5.7 Access Point

Access Point adalah alat bantu pada jaringan *wireless* atau WLAN (*Wireless-LAN*). *Access Point* menerima dan memancarkan kembali data yang berupa gelombang. *Access Point* menghubungkan antara komputer yang satu dengan yang lain pada WLAN dan kadang berfungsi pula menjadi jembatan (*Bridge*) antara WLAN dengan jaringan yang menggunakan kabel. *Access Point* memiliki fungsi yang sama seperti *hub* bagi jaringan yang menggunakan kabel. WLAN berukuran kecil cukup menggunakan satu *Access Point* saja, namun WLAN yang besar membutuhkan beberapa *Access Point* sekaligus.

2.1.6 Jenis – Jenis Media Transmisi

2.1.6.1 Copper Media

2.1.6.1.1 Coaxial

Kabel *coaxial* adalah kabel yang terdiri dari 1 buah kabel yang menghantarkan sinyal dan dibungkus dengan lapisan kabel. Kabel yang berada di sisi luar berfungsi sebagai *ground*. Kabel *coaxial* dengan bantuan *repeater* dapat menghantarkan data hingga jarak yang jauh.

2.1.6.1.2 Twisted Pair

Twisted pair adalah tipe yang terdiri dari dua kabel tembaga atau lebih yang berukuran kecil. Kabel-kabel

tembaga tersebut masing-masing memiliki pelindung tersendiri dan saling membelit satu sama lain. Kabel dipilih untuk mengurangi *noise*, *crosstalk*, dan induksi *elektromagnetik*. Pada jaringan saat ini, digunakan kabel *twisted pair* yang berisi 8 buah kabel kecil yang masing-masing memiliki warna yang berbeda.

Kabel *twisted pair* umumnya dijual di pasaran dengan dua macam varian, yaitu UTP (*Unshielded Twisted Pair*) dan STP (*Shielded Twisted Pair*). Perbedaan antara dua jenis varian ini adalah kabel STP memiliki pelindung tambahan berupa aluminium foil sehingga *noise* dan induksi *elektromagnetik* dapat lebih diperkecil. Namun, harga kabel STP jauh lebih mahal dibandingkan dengan kabel UTP sehingga kabel UTP lebih umum ditemui pada jaringan.

UTP mempunyai panjang maksimum 100 meter dan mempunyai kecepatan 4 – 100 Mbps. Adapun STP juga mempunyai panjang maksimum 100 meter, kecepatan transmisi 16 – 155 Mbps dan interferensi elektriknya lebih rendah dibandingkan UTP.

Kabel ini memerlukan konektor spesial untuk *grounding*. Kabel ini tahan terhadap interferensi elektrik, tetapi kurang dapat mengatasi penyadapan.

Ada dua standar pengurutan kabel yang saat ini digunakan, yaitu standar EIA (*Electronics Industrial Asosiation*) dan

standar AT&T. Perbedaan antara kedua standar tersebut ada pada urutan warna kabel saja. Standar pengkabelan *twisted pair* yang sering digunakan adalah standar EIA.

Kabel *twisted pair* digunakan untuk menghubungkan antar-LAN *Card* maupun antara LAN *Card* dengan *Hub* atau *Switch*. Pada ujung kabel-kabel tersebut dipasang konektor RJ-45 sehingga dapat dengan mudah ditancapkan pada LAN *Card* ataupun *Hub*.

Ada dua macam cara pemasangan kabel *twisted pair*. Untuk kabel yang menghubungkan antar-LAN *Card* menggunakan metode *Crossover*, sedangkan untuk menghubungkan LAN *Card* dengan *Hub* menggunakan metode *Straight*.

Straight artinya pin 1 terhubung dengan pin 1, pin 2 dengan pin 2, dan seterusnya. Sedangkan pada *crossover*, pin 1 terhubung dengan pin 3, pin 2 terhubung dengan pin 6, dan pin yang lainnya seperti pada *straight*.

Standar 568 ANSI/EIA (*American National Standards Institute/Electronic Industries Association*) adalah salah satu standar yang menentukan kategori kabel (biasanya disebut “CAT”) pada sistem pengkabelan *twisted pair* (kabel, junction, konektor) sehubungan dengan kecepatan data yang bisa didukung secara efektif. Spesifikasinya menguraikan

material kabel, tipe konektor dan blok junction yang digunakan dalam pengelompokan kategori.

2.1.6.1.3 Fiber Optic

Kabel *fiber optik* menggunakan serat kaca atau plastik untuk mentransfer data. Kabel *fiber optik* terdiri dari sekumpulan serat kaca yang masing-masing memiliki kemampuan untuk mentransmisikan data dalam bentuk gelombang cahaya. Pada *fiber optic*, data ditransmisikan melalui cahaya bukan *elektron*. Biasanya ada dua *fiber*, satu *fiber* untuk masing-masing arah. Panjang kabel sekitar 2 kilometer, kecepatan transmisi 100 Mbps sampai 2 Gbps. Keistimewaan kabel ini adalah tidak adanya interferensi. Dua tipe kabel fiber optik adalah kabel model tunggal (digunakan dengan laser) dan kabel model ganda (digunakan dengan *driver Light Emitting Diode / LED*).

2.1.6.2 Radio

Gelombang radio merupakan salah satu alternatif media transmisi pada jaringan. Gelombang radio cukup digemari, bahkan di Indonesia sendiri sempat mengalami *booming* karena infrastruktur yang diperlukan tidak terlalu mahal dan cukup mudah dalam instalasinya. Jaringan yang menggunakan gelombang radio cukup memiliki

pemancar pada jarak- jarak tertentu saja tanpa harus mengeluarkan biaya yang banyak untuk membeli kabel.

2.1.6.2.1 Infrared

Infrared adalah gelombang yang panjangnya di bawah panjang gelombang cahaya. *Infrared* menggunakan LED (*Light Emitting Diode*) atau laser untuk membangkitkannya. Gelombang ini tidak mampu menembus benda sehingga jarak yang mampu ditempuh relatif pendek dan berjalan pada satu garis lurus (*point to point*). Frekuensi dari gelombang *infrared* berada pada 100 Ghz – 1000 Ghz. Dalam pemanfaatannya, jaringan *infrared* digunakan untuk menghubungkan komputer pada jarak pendek saja. Infrared diterapkan untuk pengiriman data pada jarak dekat antara perangkat *portabel* (laptop, PDA, dan ponsel) dengan komputer *desktop*. Koneksi *infrared* dilakukan melalui port *infrared*. *Infrared* sangat rentan terhadap gangguan dari gelombang lain yang lebih kuat seperti gelombang cahaya tampak.

2.1.6.2.2 Bluetooth

Bluetooth ditemukan pada tahun 1998 atas kerjasama dari perusahaan Ericsson, IBM, Intel, Nokia, dan Toshiba. *Bluetooth* digunakan pada *Personal Area Network* (PAN).

Kini, *bluetooth* diterapkan untuk pengiriman data suara dan digital pada jarak pendek antara perangkat *portabel* (laptop, PDA, dan ponsel) dengan komputer *desktop*. *Bluetooth* mendukung komunikasi *point to point* dan juga *multi point*. *Bluetooth* mampu mentransferkan data hingga jarak 100 meter dengan bantuan *power booster*. Jika ada gangguan dari gelombang lain maka sinyal *bluetooth* tidak akan terputus, melainkan hanya berpengaruh pada kecepatannya saja.

2.1.6.2.3 Wireless

Wireless Local Area Network atau WLAN beroperasi pada frekuensi 2,4 dan 5,8 Ghz serta menggunakan teknologi *spread spectrum* yang hingga kini masih terus dikembangkan. Dengan menggunakan gelombang ini, kecepatan transfer data berkisar 1Mbps hingga 10 Mbps. Standar WLAN 2,4 Ghz menggunakan standar IEEE 802.11, sedangkan WLAN 5,8 atau sering disebut HIPERLAN menjadi standar di negara-negara Eropa.

2.1.6.2.4 VSAT

Very Small Aperture Terminal atau VSAT adalah stasiun pemancar yang digunakan untuk komunikasi via satelit berupa data, suara, dan sinyal video. VSAT terdiri dari dua bagian, yaitu *tranceiver* yang berada di luar ruangan dan harus berada

di garis lurus (*Line of Sight*) dengan satelit yang dituju, serta perangkat yang berada di dalam sebagai perantara antara *transceiver* dengan perangkat yang menggunakan komunikasi VSAT seperti komputer atau mesin ATM. *Transceiver* menerima dan mengirimkan sinyal menuju satelit *transponder* di angkasa. Kemudian, satelit bertugas meneruskan data tersebut pada VSAT-VSAT lainnya sehingga jika diperhatikan maka jaringan yang terbentuk adalah jaringan bertopologi *star* dengan satelit sebagai *hub*. VSAT dapat mentransferkan data hingga 56 Kbps.

2.1.7 Referensi Model Jaringan

2.1.7.1 Model OSI

Teknologi *networking* adalah teknologi yang *global*. Berbagai macam *vendor hardware* dan *programmer* yang membuat *hardware* dan aplikasi untuk *networking*. Karena terdiri dari berbagai jenis variasi maka diperlukan suatu *standard* tertentu sehingga setiap *hardware* maupun *software* antar *vendor* dan *programmer* dapat saling berkomunikasi berdasarkan *standard* tersebut. Salah satu *standard networking* yang paling luas dipakai adalah OSI. Dikembangkan oleh *Internasional Organization for Standardization (ISO)* pada tahun 1984. Terbagi dalam 7 bagian :

1. *Layer 7 Application Layer*
2. *Layer 6 Presentation Layer*

3. *Layer 5 Session Layer*
4. *Layer 4 Transport Layer*
5. *Layer 3 Network Layer*
6. *Layer 2 Data Link Layer*
7. *Layer 1 Physical Layer*

Masing-masing *layer* mempunyai tugasnya masing-masing. Dari 7 *layer* tersebut dapat dikelompokkan menjadi dua bagian *upper layer* dan *lower layer*.

Upper layer berkaitan dengan *software*, semakin tinggi *layernya* maka semakin sering berinteraksi dengan *user*. *Lower layer* berkaitan dengan data transport, semakin rendah maka semakin berkaitan dengan perangkat keras.

Physical Layer menjelaskan tentang :

1. *Voltages levels*
2. *Timing of voltages changes*
3. *Physical data rates*
4. *Maximum transmission distances*
5. *Physical connector*

Data Link Layer menjelaskan tentang :

1. *Physical addressing*
2. *Network topology*
3. *Network architecture*
4. *Sequencing of frames*
5. *Flow control*

Network layer menjelaskan tentang :

1. *Logical addressing*
2. *Routing protocol*

Transport Layer menjelaskan tentang :

1. *Flow control*
2. *Virtual circuit management*
3. *Error checking dan recovery*

Session Layer menjelaskan tentang :

Establish, manages dan terminates sesi komunikasi antar *layer* yang di atasnya.

Presentation Layer menjelaskan tentang :

Menyediakan format data yang dapat dikenali oleh *Application Layer*.

Application Layer menjelaskan tentang :

Aplikasi-aplikasi *networking* yang langsung berinteraksi dengan *end user*.

Ada beberapa keuntungan yang diperoleh dengan membagi-bagi *network* menjadi bagian-bagian yang lebih kecil.

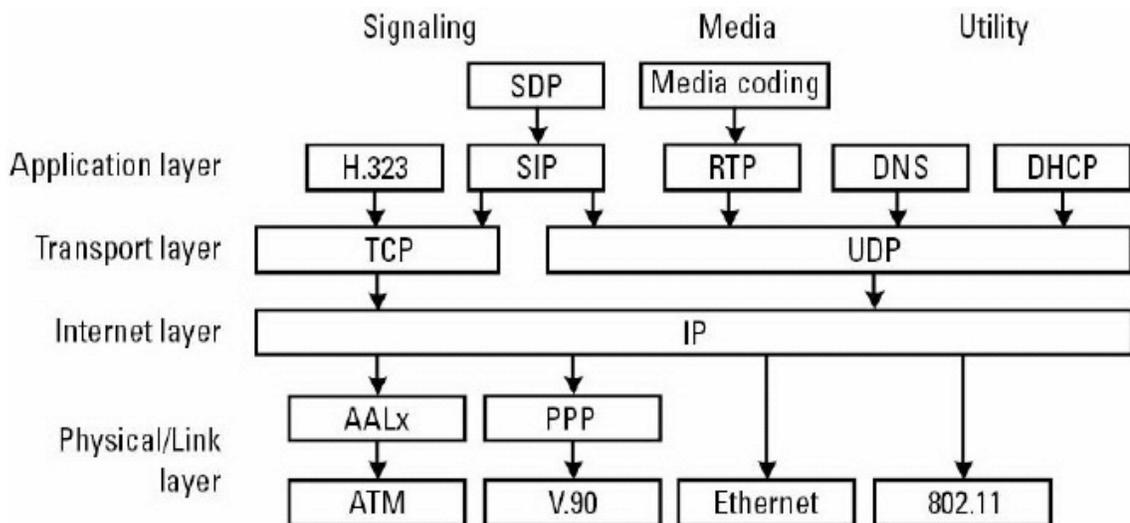
Keuntungan-keuntungan tersebut antara lain :

1. Menyederhanakan kompleksitas dalam *networking*
2. Membuat adanya standarisasi yang memungkinkan banyak *vendor* untuk mengembangkan produknya tetapi tetap bisa saling berkomunikasi satu dengan lainnya.
3. Memungkinkan pengembangan yang lebih cepat, karena perubahan pada suatu *layer* tidak mempengaruhi *layer* lainnya.

4. *Networking* menjadi lebih mudah dipelajari.

2.1.7.2 Model TCP / IP

Protocol ini pertama kali dipakai dikembangkan untuk sistem operasi UNIX. Lambat laun berkembang dan pada saat ini merupakan salah satu *protocol* yang paling banyak dipakai di *networking*. TCP / IP juga menjadi *protocol* utama untuk jaringan *internet*.



Gambar 2.5 TCP/IP Layer

TCP / IP *protocol* hanya terdiri atas 4 *layer*, yaitu :

2.1.7.2.1 *Application Layer*

Application layer terdiri atas *protocol - protocol* yang biasa digunakan disisi *client*, contoh dari *protocol - protocol* tersebut adalah :

1. HTTP : merupakan *protocol* yang menterjemahkan bahasa HTML supaya bisa ditampilkan berupa halaman *web* di *web browser*.
2. FTP : merupakan *protocol* yang digunakan untuk transfer data di *internet*. Sifat transfer datanya adalah *reliable* karena menggunakan TCP.
3. TFTP : merupakan *protocol* yang digunakan untuk transfer data, tetapi tidak *reliable*. Biasanya digunakan di LAN untuk menyimpan konfigurasi *router* maupun *switch* yang *manageable*.
4. DNS : Merupakan metode yang mengubah alamat IP menjadi sebuah alamat *web server*. DNS merupakan sebuah *server* yang *datasenya* tersebar di seluruh dunia. Untuk melihat *ip address* dari sebuah alamat *website* gunakan *command* "nslookup" di *command prompt*.
5. DHCP : Merupakan *protocol* yang mempermudah dalam pemberian *IP address* kepada komputer / *host*. Karena setiap komputer / *host* yang terjoin ke jaringan akan langsung mendapatkan *IP address* apabila terdapat DHCP *server*.
6. Telnet : Merupakan *protocol* yang digunakan untuk melakukan *remote login* terhadap sebuah PC atau *networking device*.

2.1.7.2.2 Transport layer

Pada *layer* ini terdiri atas 2 macam transport *protocol* yaitu TCP dan UDP.

1. TCP

- *Protocol* ini bertanggung jawab terhadap pengiriman data.
- *Protocol* ini bertugas untuk memeriksa apakah *packet* data yang kirim sampai ke tujuannya dalam keadaan yang benar.
- *Connection-oriented*
- *Reliable* dengan mengirimkan ACK setiap pengiriman data.
- Mengencapsulasi pesan kedalam *segment*
- *Reassembles messages* di tujuan
- Mengirimkan ulang paket - paket yang tidak sampai

2. UDP

- UDP merupakan *transport layer protocol* yang bekerja secara *unreliable*.
- *Connectionless*
- Paket UDP sering disebut sebagai datagram
- Proses pengecekan *error* dilakukan oleh *layer* di atasnya
- Tidak menggunakan ACK (*acknowledgements*)
- Tidak terdapat *flow control*

2.1.7.2.3 Internet Layer

Pada *internet layer*, terdapat pengalamatan *logical* yang disebut sebagai *IP address*. *IP address* merupakan alamat yang digunakan di *Internet* dan *TCP / IP protocol*. *IP address* terdiri atas 32 *bit biner* yang dituliskan dalam bentuk *decimal*.

IP address terbagi atas 5 kelas, yaitu :

Kelas A : 1- 126

Kelas B : 128-191

Kelas C : 192-223

Kelas D : 224-239

Kelas E : 240-255

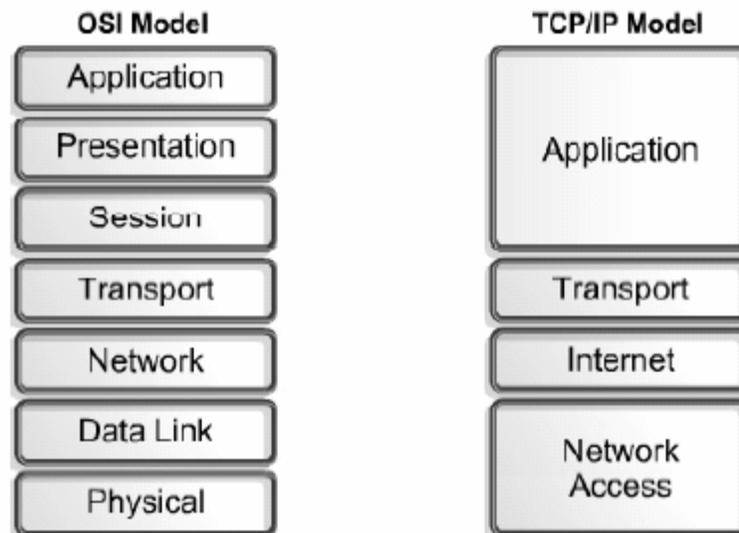
2.1.7.2.4 Network Access Layer

Tugas dari *network access layer* adalah memberikan pengalamatan *physical* yang dikenal sebagai *MAC address* dan merubah data menjadi sinyal listrik untuk kemudian dikirim melalui media.

Media yang digunakan sangatlah bervariasi. Bisa menggunakan kabel UTP, ScTP, STP maupun *fiber optic*. Disinilah fungsi dari *physical layer* untuk menentukan standarisasinya.

Pada *network access layer*, terdapat beberapa teknologi yang meliputi : *ethernet*, *fastethernet*, SLIP & PPP, FDDI, ATM, *Frame Relay*, ARP, *Proxy ARP* dan RARP.

2.1.7.3 Perbandingan Model OSI dan TCP / IP



Gambar 2.6 Perbandingan Model OSI Layer dan TCP/IP Layer

Persamaan antara model OSI dan model TCP / IP berdasarkan *Cisco Sistem* (2006) yaitu :

1. Keduanya memiliki *layer*.
2. Keduanya memiliki *application layer*, walupun memiliki fungsi yang berbeda.
3. Keduanya memiliki *transport layer* dan *network layer* yang sebanding
4. Keduanya harus diketahui oleh *network professional*
5. Keduanya mengasumsikan paket sebagai pemilih

Ini berarti setiap paket dapat mengambil jalan yang berbeda-beda untuk mencapai tujuan yang sama, ini bertolak belakang dengan *circuit-switched network* dimana semua paket mengambil jalan yang sama.

Perbedaan antara model OSI dan model TCP / IP berdasarkan Cisco Sistem (2006) yaitu :

1. TCP / IP menggabungkan *Presentation layer* dan *session layer* OSI ke dalam *application layer*.
2. TCP / IP menggabungkan data *link layer* dan *physical layer* OSI ke dalam *network access layer*.
3. TCP / IP lebih sederhana karena hanya memiliki 4 *layer*
4. Protokol TCP / IP merupakan standar untuk pengembangan *internet*, sehingga model TCP / IP mendapatkan kredibilitas hanya karena protokolnya. Sebaliknya, jaringan tidak biasa dibangun dengan *protocol* OSI, meskipun OSI digunakan sebagai panduan.

2.1.7.4 Pengalamatan TCP / IP (IP Address)

Pengalamatan TCP / IP (IPv4) terdiri dari 4 byte (32 bit) dengan dipisahkan oleh titik dengan masing-masing 8 bit. Setiap bit dalam oktet tersebut mempunyai bobot *biner* (128, 64, 32, 16, 8, 4, 2, 1). Nilai minimum oktet tersebut adalah 0 dan maksimum adalah 255.

Setiap alamat IP ini terdiri dari bagian *network* dan *host*. Bagian *network* adalah alamat yang menandakan alamat jaringan, sedangkan bagian *host* adalah alamat yang menandakan alamat *workstation* tersebut.

2.1.7.4.1 Kelas pada IP

Dalam pengalamatan IP *Address* ini, dikenal adanya kelas IP. Kelas IP tersebut dibedakan menjadi 5, yaitu A, B, C, D, E.

Alamat *network* pada kelas A adalah 1.0.0.0 – 126.0.0.0, di mana IP 127.0.0.0 digunakan untuk *looping back*. Pada IP kelas A ini, oktet pertama IP digunakan untuk alamat *network* dan tiga oktet di belakang untuk alamat *host*.

Alamat *network* kelas B adalah 128.1.0.0 – 191.254.0.0. Dua oktet pertama digunakan untuk alamat *network* dan dua oktet selanjutnya untuk alamat *host*.

Alamat *network* kelas C adalah 224.0.0.0 – 223.255.255.0. Tiga oktet pertama digunakan untuk alamat *network* dan oktet selanjutnya untuk alamat *host*.

Alamat *network* kelas D adalah 224.0.0.0 – 239.255.255.255.

Alamat *network* kelas E adalah 240.0.0.0 – 254.255.255.255.

Dari kelima kelas IP ini, IP yang digunakan untuk keperluan publik adalah IP kelas A, B, dan C. Sedangkan IP kelas D digunakan untuk grup *multicast*, di mana dalam jaringan dengan IP kelas D ini, semua alamat dipakai untuk

alamat jaringan. Kelas E dipakai untuk eksperimental dan keperluan mendatang.

2.1.7.4.2 IP Subnet Mask

Suatu alamat IP dapat dibagi menjadi beberapa sub-jaringan dengan cara meminjam bit dari bagian *host* untuk dijadikan bagian *network*. *Subnet mask* dari IP tersebut diubah menjadi satu, yang menandakan bahwa bit tersebut adalah bagian *network*.

2.1.7.4.3 Public IP Address dan Private IP Address

Public IP adalah alamat IP yang dapat dipakai untuk koneksi di *Internet*, di mana IP tersebut bersifat global, dan tidak mungkin ada dua buah *public IP* yang sama di *Internet*. Namun demikian, karena terbatasnya jumlah alamat IP yang dapat dialokasikan, maka dipakai alamat *private IP* untuk pemberian alamat IP.

Private IP adalah alamat IP yang hanya bersifat lokal untuk suatu jaringan. Karena antara suatu jaringan dengan jaringan lainnya tidak terhubung, maka pemberian alamat IP yang sama pada dua jaringan tidak akan menimbulkan masalah.

Untuk menghubungkan jaringan lokal tersebut ke jaringan *Internet*, diperlukan suatu *public IP*, di mana semua *private IP*

jaringan lokal dalam *Internet* akan diterjemahkan sebagai *public IP* tersebut. Prosedur tersebut yaitu NAT (*Network Address Translation*), di mana *private IP* diterjemahkan menjadi *public IP*.

2.1.8 Bandwidth dan Throughput

Bandwidth adalah kecepatan maksimum yang dapat digunakan untuk melakukan transmisi data antar komputer pada jaringan IP atau *Internet*. Dalam perancangan VoIP, *bandwidth* merupakan suatu yang harus diperhitungkan agar dapat memenuhi kebutuhan pelanggan yang dapat digunakan menjadi parameter untuk menghitung jumlah peralatan yang dibutuhkan dalam suatu jaringan. Perhitungan ini juga sangat diperlukan dalam efisiensi jaringan dan biaya serta acuan pemenuhan kebutuhan untuk pengembangan di masa mendatang.

Bandwidth adalah nilai kotor kapasitas maksimal sebuah jaringan. Sedangkan *Throughput* adalah nilai riil dari penggunaan jaringan yang bisa digunakan. *Throughput* adalah *bandwidth actual* yang diukur secara spesifik. Jadi nilai *bandwidth* selalu lebih besar dari pada nilai *Throughput*.

Throughput yang didapatkan kadang bisa sangat jauh dari harapan. Penyebabnya banyak. Diantaranya adalah:

1. Perangkat jaringan (misalnya, sudah terlalu tinggi loadnya, setting yang kurang tepat, dll)
2. Tipe data yang ditransfer (misalnya, umumnya *web* lebih cepat dari ftp)
3. Topologi jaringan

4. Jumlah pengguna
5. Spesifikasi komputer pengguna / *user /server*
6. *Interferensi* (misalnya listrik, cuaca, dll.)

2.1.9 Sistem Telepon

Sistem Telepon (Anonymous, 2006) merupakan sistem komunikasi yang mampu menyediakan komunikasi dua arah (*full-duplex*) antara dua ataupun lebih unit telepon. Sistem telepon terdiri atas unit telepon, yang mana terdiri dari unit penerima suara (*Receiver*) dan unit pengirim suara (*transceiver*). Unit telepon tersebut, bersama dengan unit telepon lainnya yang berdekatan, tersambung ke suatu stasiun lokal selanjutnya stasiun – stasiun lokal tersebut tersambung ke stasiun utama. Pada stasiun – stasiun tersebut terjadi mekanisme *switching* yang memungkinkan seseorang untuk memanggil (*dial*) pihak yang dituju, di mana akan terjadi pemilihan (*switching*) jalur yang akan dilalui sampai dengan tujuan.

2.1.9.1 Operasi Telepon (*Call Flow*)

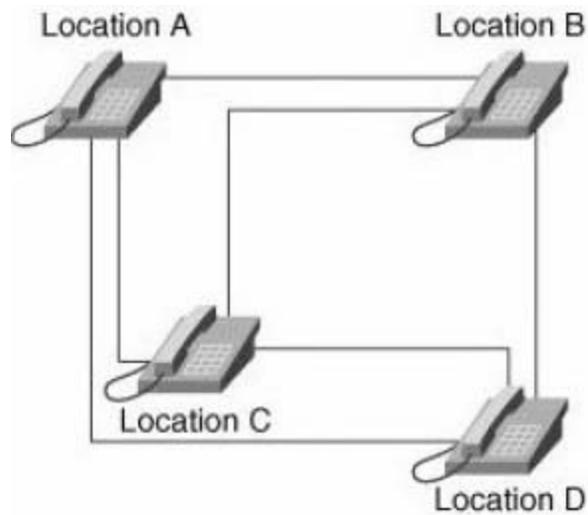
Operasi telepon dimulai ketika seseorang mengangkat gagang telepon, kemudian *central office* mendeteksi status *on-hook* (telepon tertutup) berubah ke status *off-hook* (telepon terangkat). Kemudian *central office* mengirimkan nada panggil (*dial tone*) ke telepon tersebut dan sirkuit pada jaringan digunakan untuk mengenali adanya nomor yang ditekan (baik *pulse* maupun DTMF), segera setelah *central office* mendapatkan nomor pertama, nada panggil dihentikan.

Setelah semua nomor ditekan, komponen dalam jaringan telepon membuat koneksi ke pihak yang dituju dan ringing *voltage* generator dihubungkan untuk membuat telepon yang dituju berdering (dengan asumsi tidak sibuk). Ketika pihak yang dituju mengangkat telepon, *central office* untuk telepon tersebut mendeteksinya dan memutuskan ringing *voltage* generator. Lalu sirkuit audio untuk kedua telepon yang berpartisipasi dihubungkan dan percakapan dapat dimulai. (Schweber, 1991)

2.1.9.2 PSTN (*Public Switched Telephone Network*)

Sistem komunikasi konvensional atau yang dikenal dengan *Public Switched Telephone Network* (PSTN) telah berkembang sejak ditemukannya transmisi suara melalui kawat pada tahun 1878 oleh Alexander Graham Bell, yang dikenal dengan *ring - down circuit*. *Ring - down circuit* berarti tidak ada pemanggilan (*dialing*) nomor, namun menggunakan sebuah kawat fisik untuk menghubungkan dua *devices*. Secara mendasar, seseorang mengangkat telepon dan orang lain berada di ujung lainnya (tidak ada *ringing*).

Sistem ini kemudian berkembang dari transmisi suara satu arah, dimana hanya satu *user* dapat berbicara, menjadi transmisi suara bidirectional (dua-arah), yang memungkinkan kedua *user* dapat berbicara. Untuk memindahkan suara sepanjang kawat diperlukan kabel fisik diantara tiap lokasi dimana *user* ingin melakukan panggilan.



Gambar 2.7 Jaringan Dasar 4-Telepon

Proses pemasangan kabel diantara perangkat yang memerlukan akses telepon sangat tidak efisien, memerlukan biaya yang besar dan sulit untuk diimplementasikan. Karena itu, diperkenalkanlah penggunaan *switch*, dimana tiap pengguna telepon hanya membutuhkan satu kabel yang terhubung secara terpusat ke kantor *switch*. Pada awalnya, seorang operator telepon berperan sebagai *switch*. Operator ini bertanya kepada pemanggil mengenai lokasi panggilan yang dituju kemudian secara manual menghubungkan kedua jalur suara.

Sistem telepon terus berkembang dan hingga saat ini, *switch* dengan operator manusia telah diganti dengan *switch* elektronik maupun *softswitch*.

2.1.9.3 Packet Telephony (Packet Switched Network)

Mekanisme *switching* pada sistem telepon dibagi dua, yaitu :

1. *Circuit Switched Network* : adalah jaringan yang mempunyai saluran terdedikasi yang *dialokasikan* selama beberapa saat selama transmisi data (percakapan), dimana jalur tersebut tidak bisa digunakan oleh *host* lain selama belum didealokasikan. Pada *circuit switched*, *host* yang akan mengirim data serta yang menerima data akan membuka jalur data yang hanya dipakai oleh pengirim dan penerima. Jalur data tersebut akan ditutup bila transmisi data tersebut telah selesai. Jaringan telepon konvensional (PSTN) menggunakan mekanisme *circuit switching*.

2. *Packet Switched Network* : adalah jaringan-jaringan yang dihubungkan oleh *router*, dimana setiap *host* yang terhubung dalam jaringan tersebut, secara teori, dapat mengirimkan paket data kepada *host* lain. Paket tersebut berisi alamat yang dituju, dan *router* meneruskan paket tersebut ke alamat yang dituju tersebut. Protokol *packet switched* ini membagi data menjadi paket – paket sebelum dikirim. Protokol ini menggunakan prinsip *multiplexing*, dimana paket – paket tersebut dapat melalui jalur – jalur yang berbeda bersama paket – paket yang berasal dari data lain untuk sampai di tujuan. Begitu sampai di tujuan, paket – paket tersebut akan dirangkai kembali menjadi data asli. Sebuah *host* dapat membuka banyak sesi sekaligus dan mengirimkan data – data dari sesi yang berlainan tersebut melalui satu jalur yang sama.

Keunggulan *packet switched* dibandingkan dengan *circuit switched* adalah lebih hemat *bandwidth* dan lebih efisien, karena jalur data tersebut dapat dipakai oleh banyak transmisi data, serta tidak perlu

membuka koneksi jalur jika tidak ada pengiriman atau penerimaan data. Sistem komunikasi *internet telephony* atau VoIP menggunakan mekanisme *packet switching*.

2.1.9.4 PBX (*Private Branch eXchange*)

PBX (*Private Branch eXchange*) atau biasa disebut *phone switch* adalah perangkat yang menghubungkan telepon – telepon dalam suatu jaringan lokal dengan jaringan telepon umum. Fungsi utama dari PBX adalah untuk mengatur panggilan yang datang ke *extention* atau cabang tertentu sesuai dengan yang dituju dalam jaringan lokal tersebut, dan untuk membagi saluran telepon diantara semua *extention*. *Extention* adalah sebuah nama atau nomor yang merepresentasikan *user* dari pbx ini. Saat ini, telah banyak fitur – fitur lain yang dimiliki pbx, antara lain seperti *automated greetings* untuk pemanggil, koneksi ke *voice mail*, *automatic call distribution* (ACD), *telekonfrensi*, dan lainnya.

Salah satu keuntungan utama dari PBX adalah mengurangi *local loops* yang diperlukan dari *central office switch* PSTN. Keuntungan lain dari memiliki PBX sendiri adalah *control* seperti *setup*. Misalnya jika ingin menambah *user* baru, mengubah fitur, atau memindahkan *user* ke lokasi baru, maka tidak perlu menghubungi carrier PSTN. Namun sistem PBX menambah *level* kompleksitas yang lain karena harus melakukan konfigurasi dan *maintain call routing* pada PBX.

2.2 Teori Khusus

2.2.1 VoIP

Voice over Internet Protocol (VoIP) adalah teknologi yang memungkinkan percakapan suara jarak jauh melalui media *internet*. Teknologi ini merubah data suara menjadi kode digital dan *dialirkan* melalui jaringan yang mengirimkan paket-paket data, dan bukan lewat sirkuit *analog* telepon biasa.

Pada jaringan pesawat telepon suara konvensional langsung terhubung dengan PABX (*Private Automated Branch eXchange*) atau jaringan telepon yang terhubung langsung dengan STO (Sentral telepon Otomatis) terdekat. Dalam STO ini ada daftar nomor-nomor telepon yang disusun secara bertingkat sesuai dengan daerah cakupannya. Jika dari pesawat telepon tersebut mau menghubungi rekan yang lain maka tuts pesawat telepon yang ditekan akan menginformasikan lokasi yang dituju melalui nada-nada DTMF (*Dual-tone multi-frequency*), kemudian jaringan akan secara otomatis menghubungkan kedua titik tersebut.

Bentuk yang paling sederhana dalam sistem VoIP adalah dua buah komputer terhubung dengan jaringan *internet*. Syarat-syarat dasar untuk melakukan koneksi VoIP adalah komputer yang terhubung ke *internet*, mempunyai kartu suara (*sound card*) yang dihubungkan dengan *speaker* dan mikrofon. Dengan dukungan perangkat lunak khusus (*software*), kedua pemakai komputer bisa saling terhubung dalam koneksi VoIP satu sama lain.

Hubungan tersebut di atas dapat berupa pertukaran *file*, suara atau gambar. Penekanan utama dalam sistem VoIP adalah hubungan keduanya dalam bentuk suara. Jika kedua lokasi terhubung dengan jarak yang cukup jauh (antar kota,

antar negara) maka bisa dilihat keuntungan dari segi biaya. Kedua pihak hanya cukup membayar biaya pulsa *internet* saja (pulsa lokal), yang biasanya akan lebih murah daripada biaya pulsa telepon sambungan langsung jarak jauh (SLJJ) atau internasional (SLI). Dengan menggunakan jaringan VoIP dibandingkan dengan menggunakan PSTN (*Public Switch Telephone Network*) atau pesawat telepon biasa, dengan VoIP dapat berbicara selamanya dengan setiap orang yang diinginkan dimana pun berada serta pada waktu yang bersamaan anda dapat saling bertukar data berupa gambar, grafik metode presentasi dan video conference.

2.2.1.1 Komponen –komponen VoIP

2.2.1.1.1 *User Agent*

User Agent merupakan komponen yang digunakan oleh pengguna untuk memulai dan menerima sesi komunikasi. Dalam VoIP, *user Agent* berupa komponen yang melakukan *dial* nomor telepon VoIP dan atau menerimanya. Komponen ini dapat berupa aplikasi perangkat lunak (atau biasa disebut dengan *softphone*) atau dapat pula dalam bentuk perangkat keras.

2.2.1.1.2 *Proxy*

Proxy atau bisa juga disebut dengan *server*, *gatekeeper*, *register* adalah komponen yang menerima registrasi *User Agent* dan bertugas mengatur penomoran dan *call routing*.

Perlu dicatat dan diingat bahwa *Proxy* yang dimaksud disini bukanlah *Proxy server Internet*, melainkan aplikasi *server* yang mengatur jaringan VoIP.

2.2.1.1.3 Protokol

Protokol adalah komponen berupa seperangkat aturan komunikasi antar *User Agent*, antar *Proxy* atau *User Agent* dengan *Proxy* dan sebaliknya. Yang paling tua, stabil, dan andal adalah protokol H.323. Ia merupakan koleksi dari beberapa protokol lain yang mengatur *session* dan media transfer. Namun, H.323 memiliki kekurangan yang cukup fatal yaitu tidak dapat dengan mudah menembus NAT atau *Network Address Translation*. Dengan demikian diperlukan *gatekeeper* yang harus dioperasikan di setiap node jaringan LAN yang menggunakan fasilitas NAT. *Gatekeeper* tersebut berfungsi sebagai jembatan antara pengguna di dalam jaringan dengan NAT tersebut dan dengan mereka yang berada di luar jaringan LAN.

Selanjutnya, adalah protokol SIP atau *Session Initiation Protocol*. Ia tugasnya hanya menciptakan, menghapus, dan memodifikasi *session* atau sesi komunikasi. Tidak seperti H.323, SIP bukanlah protokol koleksi. SIP hanya mengatur *session* saja, sedangkan data *voice* disalurkan melalui protokol lain yang bukan bagian dari SIP.

SIP dapat dengan mudah menembus NAT sehingga implementasinya dapat terpusat pada satu *server* saja. Bagi Anda yang berada dalam jaringan dengan NAT tidak perlu khawatir tidak dapat terhubung dengan *server*. Beda halnya jika pada jaringan Anda tersebut tidak hanya NAT saja yang terpasang tetapi juga Firewall yang terkonfigurasi menutup komunikasi protokol SIP.

Protokol lain yang mulai sering digunakan adalah IAX2 atau Inter *Asterisk* eXchange versi 2. Protokol ini sangat andal. Ia dapat menembus NAT dengan mudah dan hanya menggunakan satu port saja untuk membentuk *session* dan media transfer. Protokol ini juga mempunyai *feature-feature* yang dapat mengurangi konsumsi *bandwidth*, seperti IAX-*trunking* dan menjernihkan suara seperti jitter buffer. Namun, tidak seperti dua protokol lainnya, SIP dan H.323 yang *Open Standard*, IAX2 masih dalam *draft* menuju *Open Standard*. Oleh karena itu dukungan vendor perangkat keras yang menerapkannya dalam perangkat-perangkat mereka masih sangat minim dibandingkan dengan SIP, apalagi H.323. Kendati demikian, kode program untuk implementasi IAX2 merupakan *Open Source*, sehingga bisa dipastikan lambat laun akan banyak tersedia pula *softphone* dan perangkat keras berbasis protokol IAX2.

2.2.1.1.4 Codec

Codec adalah teknologi yang memaketkan data *voice* ke dalam format data lain dengan perhitungan matematis tertentu, sehingga menjadi lebih teratur dan mudah dipaketkan. Dengan menggunakan *Codec* tertentu *bandwidth* dapat dihemat. Namun risikonya, suara dapat menjadi kurang jernih atau berubah warna suaranya. Apabila mengejar kualitas suara yang baik, jernih, dan tidak berubah warna suaranya, dibutuhkan *codec* dengan perhitungan matematis yang minim. Konsekuensinya kebutuhan *bandwidth* meningkat.

Codec dengan *bandwidth* terboros adalah G.711, menghabiskan *bandwidth* sekitar 87 kbps. Sebaliknya, *codec* yang paling hemat dan umum digunakan adalah G.723.1, menghabiskan *bandwidth* sekitar 22 kbps. *Codec* lain yang umum digunakan karena suaranya yang lebih jernih dari pada G.723.1, tetapi *bandwidth*-nya jauh lebih kecil dibanding G.711 adalah G.729. *Codec* ini menghabiskan *bandwidth* sekitar 24 kbps. Adapun yang *codec* lain yang umum dan gratis adalah GSM dan iLBC yang menghabiskan *bandwidth* sekitar 29 – 31 kbps.

2.2.2 Asterisk

Asterisk adalah implementasi *software* dari PBX yang bersifat *freeware* / *open source*. *Asterisk* menyediakan banyak fitur seperti *voice mail*, *call*

conferencing, automatic call distribution, dan lainnya. Asterisk mendukung penggunaan teknologi VoIP (Voice over Internet Protocol) melalui beberapa protokol sebagai berikut :

1. H323 : Protokol yang dikembangkan oleh komunitas telepon. Dijadikan standar oleh ITU-T (*International Telecommunication Union Telecommunication Sector*).
2. SIP : *Session Initiation Protocol*, dikembangkan oleh komunitas internet dan merupakan standar yang digunakan oleh IETF (*Internet Engineering Task Force*). Semakin terkenal, mengalahkan protokol pendahulunya, H323.
3. IAX : *Inter Asterisk's eXchange*. Salah satu protokol proprietary untuk *Voice over Internet Protocol* antar server Asterisk.

Terdapat lebih dari 20 berkas konfigurasi yang terlibat dalam suatu *call center asterisk* namun dalam keadaan sesungguhnya lebih ditekankan pada 5 berkas konfigurasi yaitu 4 berkas pengatur *channel* telepon masuk dan 1 berkas pengatur jalanya (navigasi) *call center*, 5 berkas tersebut adalah :

1. Sip.conf : Berisi parameter – parameter yang mengatur akses melalui SIP (*Session Initiation Protocol*) terhadap server asterisk. Client juga harus di konfigurasi dalam berkas ini agar dapat menghubungi atau menerima telepon menggunakan server asterisk.
2. Zapata.conf : Berisi parameter – parameter yang berhubungan dengan TDM *channel* (perangkat keras). *Channel* harus didefinisikan pada berkas ini sebelum mereka dapat digunakan oleh asterisk.

3. *H323.conf* : berkas ini mengatur konfigurasi *client* yang berhubungan melalui protokol H323
4. *Iax.conf* : berkas ini mengatur konfigurasi *client* yang berhubungan melalui protokol iax
5. *Extention.conf* : berkas utama yang merupakan implementasi dari *dialplan*. Berkas ini mengatur navigasi jalannya *call center*. Berkas ini juga mengatur variable – variable global untuk *asterisk*.

2.2.3 SIP

2.2.3.1 Pengenalan SIP

SIP (*Session Initiation Protocol*) atau dikenal juga dengan IETF RFC 3261 di desain sebagai protokol multimedia yang dapat memanfaatkan kegunaan dari arsitektur aplikasi *internet* yang sudah ada. Sebagai sebuah protokol, SIP hanya mengatur bagaimana cara membangun dan menutup sebuah sesi komunikasi. SIP menggunakan protokol lainnya dari IETF untuk mengatur semua aspek dalam VoIP dan sesi komunikasi, seperti RTP untuk media transfer, SDP untuk menentukan cara berkomunikasi, URL untuk pengalamatan, *Domain Name Sistem* (DNS) untuk menemukan suatu alamat, dan *Telephony Routing over IP* (TRIP) untuk pengaturan jalur panggilan.

SIP (Raharja, 2004) adalah sebuah *signaling protocol* (*application layer control*) untuk menciptakan, mengatur dan menghentikan sesi komunikasi multimedia antara dua atau lebih peserta. Sesi komunikasi ini meliputi *internet multimedia conference*,

internet telephone calls dan distribusi multimedia. SIP bukan media transfer *protocol*, sehingga SIP tidak membawa *packet* suara atau video. SIP memanfaatkan RTP (*Real Time Protocol*) untuk media transfer.

2.2.3.2 Fungsi SIP

1. *Call initiation* : Membangun sebuah sesi komunikasi dan mengundang *user* lain untuk bergabung di dalam sesi komunikasi.
2. *Call modification* : Bila perlu, SIP dapat memodifikasi sesi komunikasi.
3. *Call termination* : Menutup sesi komunikasi
4. *Presence* : Mengetahui status *user* dan mengumumkan status *user* pada *user* lain, *online* atau *offline*, *away* atau *busy* (Raharja, 2004)

2.2.3.3 Komponen SIP

Pada SIP, terdapat beberapa komponen, yaitu SIP *server* yang meliputi *Proxy server*, *registrar server*, dan *redirect server*, serta SIP *user Agent*.

2.2.3.3.1 User Agent

User Agent merupakan komponen SIP yang memulai, menerima, dan menutup sesi komunikasi. *User Agent* ini dibagi menjadi dua jenis yaitu (Raharja, 2004) :

1. *User Agent Client* (UAC), yaitu *User Agent* yang memulai sesi komunikasi. UAC menginisialisasi SIP *request* dan menerima SIP *response*.
2. *User Agent Server* (UAS), yaitu *User Agent* yang menerima atau menanggapi sesi komunikasi. UAS menerima SIP *request* dan mengirim kembali SIP *response*.

Baik kedua jenis UA tersebut dapat menutup sesi komunikasi.

User Agent Client dapat berupa *software* pada komputer atau *softphone* (X-Lite, SJPhone, Windows Messenger) dan berupa *hardware* atau *hardphone* (IP Phone, USB Phone).

2.2.3.3.2 Proxy Server

Proxy server merupakan komponen penengah antara *user* dan *Agent*, bertindak sebagai *server* dan *client* yang menerima *request message* dari *user Agent* dan menyampaikan pada *user Agent* lainnya.

Request yang diterima dapat dilayani sendiri atau disampaikan (forward) pada *Proxy* lain atau *server* lain. *Proxy server* menterjemahkan dan/atau menulis ulang *request message* sebelum menyampaikan pada *user Agent* tujuan atau *Proxy* lain.

Proxy server juga bertugas menyimpan data hasil sesi komunikasi yang terjadi antara UAC dan UAS.

Proxy server merupakan pusat komunikasi yang dapat dicapai oleh *user Agent* secara langsung. Ketika *user Agent* mengirimkan pesan “INVITE”, maka *Proxy server* akan mencari alamat IP dari alamat yang di panggil oleh pesan tersebut ke *database* dan meneruskan pesan “INVITE” tersebut ke alamat yang di panggil berdasarkan alamat IP yang didapat (Johnston, 2004).

2.2.3.3.3 Redirect Server

Redirect server merupakan komponen yang menerima *request message* dari *user Agent*, memetakan alamat SIP *user Agent* atau *Proxy* tujuan kemudian menyampaikan hasil pemetaan kembali pada *user Agent* pengirim (UAC).

Tidak seperti *Proxy server*, *redirect server* tidak menyimpan data hasil sesi komunikasi antara UAC dan UAS setelah pemetaan disampaikan pada UAC. *Redirect server* juga tidak dapat memulai inisiasi *request message*.

Redirect server berfungsi sebagai perantara dan membelokkan panggilan, bilamana alamat yang di panggil oleh *user Agent* tidak terdapat pada *Proxy server* tersebut, melainkan pada *Proxy server* yang lain. Bila *redirect server* berhasil mendapatkan alamat yang

dipanggil pada *Proxy server* lain, maka alamat tersebut akan dikirimkan kembali kepada *user Agent* untuk selanjutnya memakai alamat itu untuk menghubungi pihak yang dipanggil. Bila alamat yang dituju tidak dapat dicari oleh registrar *server*, maka registrar mengirimkan paket “CANCEL” (Johnston, 2004)

2.2.3.3.4 Registrar Server

Registrar server merupakan komponen yang menerima *request message* Sip dan *port REGISTER*. Registrar menyimpan *database user* untuk otentikasi dan lokasi sebenarnya (berupa IP dan port) agar pengguna yang terdaftar dapat dihubungi oleh komponen SIP lainnya.

Proses kerja registrar *server* berfungsi menerima autentikasi dari *user Agent* dengan menerima paket “REGISTER” dan membalas dengan pesan “ok” bila berhasil dan “cancel” bila gagal.

User Agent ataupun *Proxy server* lain dapat melakukan registrasi pada registrar *server*. Ketika registrasi dilakukan oleh *Proxy server*, maka *Proxy server* tersebut dianggap sebagai *client* oleh registrar *server*. Dengan demikian, *Proxy server* itu dapat

melakukan panggilan ke SIP *server* lainnya bila terdapat panggilan yang *dialamatkan* ke sana (Johnston, 2004).

2.2.3.3.5 Media Gateway

Media *Gateway* adalah komponen SIP yang berfungsi untuk menjembatani protokol yang berbeda, dalam hal ini SIP dengan protokol lainnya seperti H.323, MGCP, maupun dengan telepon analog (PSTN). Umumnya Media Gateway dipakai untuk menghubungkan antara SIP dengan PSTN. Ada 2 tipe interface yang terdapat dalam media gateway:

1. FXO (*Foreign Exchange Office*) : interface yang menggantikan telepon analog untuk hubungan ke PSTN atau ke PBX.
2. FXS (*Foreign Exchange Office*) : interface yang menggantikan PSTN untuk hubungan ke peralatan seperti telepon analog, modem, *fax*, dan lain – lain.

Proses perubahan format protokol adalah: ketika ada panggilan yang di-*route* oleh SIP *Proxy server* untuk menuju ke saluran PSTN, maka data tersebut akan melewati media gateway. Pada media gateway tersebut, terjadi penterjemahan dari *signaling* SIP ke *signaling* lainnya dan sebaliknya. Hal serupa terjadi pula ketika terjadi panggilan menuju jaringan H.323

ataupun protokol lainnya. Media gateway akan mengubah *signaling* dari protokol SIP menjadi protokol tersebut dan sebaliknya (Johnston,2004).

2.2.3.4 Cara Kerja SIP

Setiap komponen SIP mempunyai alamat SIP, dengan format : [sip:]<username@host>. Contohnya : sip:john-doe@binus.ac.id. Cara kerja dari SIP adalah sebagai berikut : pemanggil (UAC) dan penerima (UAS) dikenali dari alamat SIP-nya. Ketika melakukan panggilan, pemanggil (UAC) pertama-tama menentukan lokasi *server* yang tepat dan mengirimkan *request message*. Operasi SIP yang paling biasa digunakan adalah INVITE. Namun panggilan ini tidak langsung mencapai penerima melainkan dapat membentuk rantai *Proxy server* yang saling melemparkan panggilan untuk mencapai si penerima.

Message pada SIP dapat dikirimkan dengan menggunakan TCP atau UDP. TCP menyediakan transportasi data yang terkontrol dan terjamin, tetapi lebih lambat dibandingkan dengan UDP yang tidak memperhatikan error. Pengiriman *message* dengan TCP baik untuk digunakan pada jaringan dengan TCP baik untuk digunakan pada jaringan dengan kecepatan tinggi seperti LAN, ADSL, VSAT, dan sebagainya. Namun pada kenyataannya, pengiriman *message* ini umumnya menggunakan UDP yang lebih cepat, dan penanganan error dilakukan pada *layer* atas.

Message pada SIP berbasis teks standard an menggunakan karakter ISO 10646 dengan encoding UTF-8. Setiap baris harus diakhiri dengan CRLF (Carriage Return-Line Feed). Hampir semua sintaks dari *message* ini dapat berupa *request message* (untuk melakukan panggilan atau meminta layanan) atau dapat berupa *response message* (merespon panggilan atau layanan).

Struktur *Request Message*

Format dari sebuah *request message* adalah :

Metoda	URL Pengirim	Versi SIP
--------	--------------	-----------

Metoda :

INVITE – mengundan *user Agent* lain untuk bergabung dalam sebuah sesi komunikasi.

ACK – untuk konfirmasi bahwa *user Agent* telah menerima pesan terakhir dari serangkaian pesan INVITE.

BYE – untuk menutup sesi.

CANCEL – membatalkan INVITE

OPTIONS – meminta informasi tentang kemampuan *server*.

REGISTER – registrasi di Registrar *server*.

INFO – digunakan untuk membawa pesan informasi lainnya, seperti informasi *inline* DTMF.

URI (*Uniform Resource Identifier*) pengirim merupakan alamat URL dari pengguna atau layanan berasal. Sedangkan versi SIP adalah versi yang digunakan pada saat melakukan layanan. Contoh sebuah *request message* :

INVITE sip:bob@Proxy.company.com SIP/2.0

Struktur Response Message

Format dari sebuah response *message* adalah :

Versi SIP	Kode Status	Informasi tambahan
-----------	-------------	--------------------

Kode status (dengan awalan tertentu) dan artinya:

1xx - *Informational Message*

2xx – *Successful Response*

3xx – *Redirection Response*

4xx – *Request Failure Response*

5xx – *Server Failure Response*

6xx – *Global Failure Response*

Contoh sebuah response *message* :

SIP/2.0 200 OK

2.2.3.5 Protokol Pendukung SIP

SIP tidak menyediakan fungsi yang diperlukan untuk mengirim single-media atau multimedia sepanjang *network*, atau *services* untuk

program komunikasi. Melainkan SIP merupakan komponen yang bekerjasama dengan protokol lain untuk mengirim data, mengatur media streaming, menghubungkan ke PSTN, dan *service* lainnya. Protokol – protokol pendukung SIP antara lain (Johnston, 2004):

1. *Session Description Protocol* (SDP)

Session Description Protocol (SDP) atau IETF RFC 2327 digunakan untuk mengirim deskripsi informasi yang penting ketika mengirim data multimedia sepanjang *network*. Selama inisiasi sesi, SDP menyediakan informasi mengenai jenis multimedia yang di-*request user Agent* dan informasi lain yang penting untuk mengirim data ini. SDP adalah protokol berbasis teks yang menyediakan informasi pada pesan yang dikirim dalam paket UDP.

2. *Real-Time Transport Protocol* (RTP)

Real-time Transport Protocol (RTP) atau IETF RFC 3550 merupakan protokol yang berfungsi untuk mengatur pengiriman data real-time seperti audio, video, simulasi data melalui UDP. RTP ini secara spesifik menyediakan cara untuk membawa bagian audio atau media lain dari komunikasi VoIP. Walaupun RTP berjalan pada UDP, namun RTP menyediakan *reliability* dari data yang dikirim antar *user Agent*.

2.2.4 Pemrograman Web

2.2.4.1 Halaman Web

Halaman *web* adalah dokumen yang berada di *internet*, yang dapat diakses dengan mengetik URL (*Uniform Resource Locator*).

Halaman *web* terbagi dua, yaitu statis dan dinamis. Perbedaannya adalah pada halaman statis, halaman *web* tersebut tidak akan berubah, semua *user* akan melihat halaman yang sama bagaimanapun cara mengaksesnya. Sedangkan halaman dinamis adalah halaman *web* yang dapat berubah sesuai permintaan *user*, perubahan tersebut dikarenakan pada halaman *web* tersebut telah deprogram untuk menghasilkan halaman sesuai input *user*. Ekstensi *file – file* tersebut umumnya adalah html, htm, asp, jsp, aspx, php, dan sebagainya.

2.2.4.2 Client dan Server Side Scripting

Pemrograman *web* (scripting) terbagi atas dua jenis, yaitu *client-side* scripting dan *server-side* scripting. *Client-side* scripting adalah script yang pengolahannya dilakukan pada komputer *client* setelah mendownloadnya dari *web server* dan menampilkannya pada *browser*. Sedangkan *server-side* scripting adalah pemrograman *web* dimana pengolahan script tersebut dilakukan pada *web server*, dan mengirimkan hasil olahan tersebut berupa halaman *web* kepada *user*. Contoh *client-side* scripting adalah vbscript dan javascript. Sedangkan *server-side* scripting contohnya adalah ASP, ASP.NET, PHP, dan JSP.

PHP : Hypertext preprocessor (PHP) adalah bahasa scripting untuk *web programming* yang bersifat *server-side* sepertinya JSP dan ASP. Sebagian besar perintahnya berasal dari C, Java, dan Perl dengan beberapa tambahan fungsi khusus PHP. PHP tidak perlu dikompilasi

oleh *user* tapi *server* yang akan melakukan tugas tersebut. Berbeda dengan ASP PHP merupakan *open source*, karenanya gratis dan bebas.

2.2.4.3 Web Server

Web server (Wikipedia, 2008) adalah komputer yang berfungsi sebagai penyedia halaman *web*. Setiap *web server* mempunyai alamat IP, dan mungkin memiliki suatu nama domain. Pada dasarnya, setiap komputer dapat difungsikan sebagai *web server* dengan instalasi aplikasi *web server*, dan memiliki koneksi *internet*.

Beberapa aplikasi *web server* tersebut antara lain adalah Microsoft IIS, Apache HTTP *server*, dan Apache Tomcat *server*.

2.2.4.4 Teori IMK

Dalam perancangan GUI (Graphical *User Interface*) suatu situs *web* ataupun portal, perlu diperhatikan beberapa hal yang dapat mempengaruhi *user* dalam menggunakan *web* tersebut. Perancangan GUI tersebut hendaknya memudahkan dalam mengaksesnya dan menarik bagi *user*. Berikut ini adalah prinsip – prinsip dalam perancangan GUI :

1. Kenali Perbedaan

Yaitu membedakan siapakah yang akan menggunakan rancangan tersebut, apakah pemula (*first time user*), menengah (*knowledgeable intermittent user*), ataupun mahir (*expert frequent user*). Kemudian rancang GUI tersebut sesuai kebutuhan masing – masing *user* tersebut.

Gunakan Delapan Aturan Emas Perancangan GUI

Adapun Delapan Aturan Emas tersebut adalah :

- a. Berusaha Konsisten
- b. Memungkinkan frequent *users* menggunakan shortcuts.
- c. Memberikan umpan balik yang informatif.
- d. Merancang *dialog* yang memberikan penutupan (keadaan akhir).
- e. Memberikan pencegahan kesalahan dan penanganan kesalahan yang sederhana.
- f. Memungkinkan pembalikan aksi yang mudah.
- g. Mendukung pusat kendali internal (*internal locus of control*).
- h. Mengurangi beban ingatan jangka pendek.

2. Cegah kesalahan

Yaitu berusaha mencegah kesalahan – kesalahan yang dapat timbul oleh *user*, yaitu dengan membetulkan pasangan yang bersesuaian seperti tanda '{' dan '}', melengkapi urutan aksi dengan makro dan sebagainya, dan membetulkan perintah dengan mengenali kekurangan perintah ataupun memberi alternative pilihan sebagai ganti mengetik.

2.2.5 Basis Data

Basis Data (Wikipedia, 2008) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan

memanggil kueri (query) basis data disebut *Database Management Sistem* (DBMS) atau sistem manajemen basis data.

Database server menggunakan DBMS sebagai basis datanya untuk melayani kebutuhan penyimpanan informasi. Beberapa DBMS yang sering digunakan antara lain MySQL, *SQL Server*, PostgreSQL, Oracle.

MySQL adalah structured query language *database server*. Tata bahasa pemrograman SQL di aplikasi MySQL menggunakan tata bahasa (syntax) SQL standar. MySQL digunakan karena memiliki kecepatan yang baik, reliabilitas dan kemudahan. MySQL mengimplementasikan konsep *client-server* yang terdiri dari daemon *mysqld* dan beragam jenis aplikasi *client* dan library.

2.2.6 *State Transition Diagram* (STD)

State Transition Diagram (STD) digunakan untuk menggambarkan sifat dinamis dari suatu objek. STD mengilustrasikan berbagai keadaan (state) yang dimiliki suatu objek, *event* yang menyebabkan perubahan state, serta aturan yang ada untuk transisi antar state pada suatu objek. Dengan kata lain STD menjelaskan state apa dari objek yang dapat melakukan transisi ke state lain.

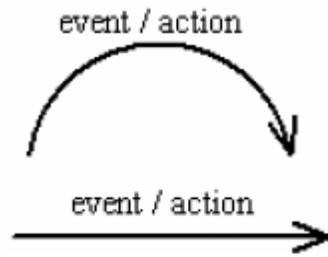
Symbol – symbol STD antara lain :

1. State, symbol :



Merepresentasikan keadaan pada suatu waktu.

2. *Event* / action, symbol :



Merepresentasikan hubungan antara keadaan (state) yang berbeda. Pada panah tersebut ditulis dengan *event* yang menyebabkan perubahan tersebut dan akibat yang dihasilkan.

2.2.7 Unified Modelling Language (UML)

Menurut Grady Booch(1998,p13), UML adalah bahasa standard untuk melukiskan *software blueprints*. UML digunakan untuk menggambarkan, menspesifikasikan, merancang, dan dokumentasi dari *artifacts of a software-intensive sistem*.

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, menggambarkan, dan membangun sistem perangkat lunak seperti halnya pada business modelling dan sistem lainnya [OMG01]. UML tidak berdasarkan pada bahasa pemrograman tertentu. Standar spesifikasi UML dijadikan standar defacto oleh OMG (Object Management Group) pada tahun 1997.UML yang berorientasikan object mempunyai beberapa notasi standar.

Spesifikasi ini menjadi populer dan standar karena sebelum adanya UML, telah ada berbagai macam spesifikasi yang berbeda. Hal ini menyulitkan komunikasi antar pengembang perangkat lunak. Untuk itu beberapa pengembang spesifikasi yang sangat berpengaruh berkumpul untuk membuat

standar baru. UML dirintis oleh Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (Object Modeling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering).

UML mendeskripsikan OOP (Object Oriented Programming) dengan beberapa diagram.

2.2.7.1 Use-case Diagram

Use-case modeling merupakan proses pemodelan fungsi-fungsi sistem dalam konteks peristiwa-peristiwa bisnis, siapa yang mengawalinya, dan bagaimana sistem itu merespons hal tersebut.

Menurut Grady Booch (1998, p97) diagram use-case adalah diagram yang menunjukkan serangkaian use case dan aktor dan hubungannya. Diagram use-case digunakan untuk mengilustrasikan gambaran statis use case sebuah sistem. Diagram use-case sangat penting dalam pengaturan dan pemodelan behavior sistem.

Diagram use-case menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Dengan kata lain, secara grafis menggambarkan siapa yang menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem.

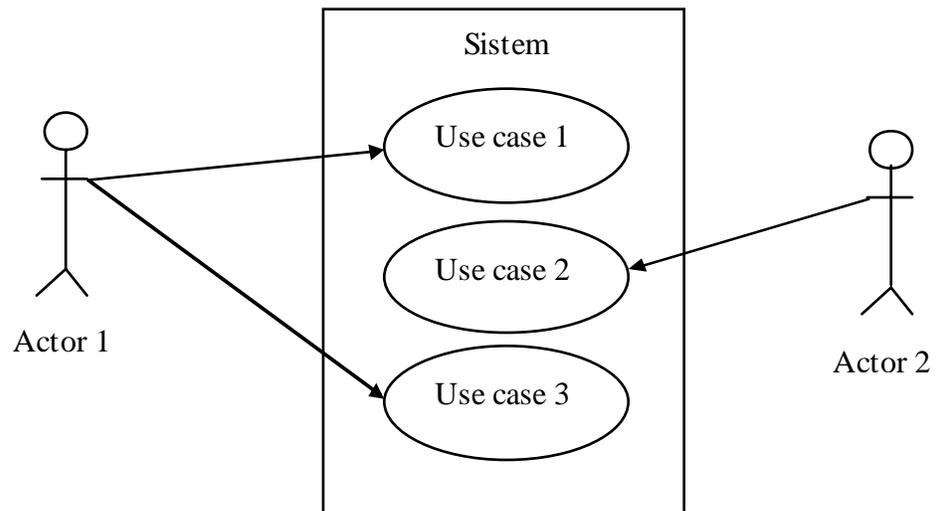
Salah satu tantangan bagi semua tim pengembangan sistem informasi, dan khususnya analis sistem, adalah kemampuan untuk memperoleh persyaratan sistem yang benar dan yang diperlukan para *stakeholder* dan menetapkannya dalam sebuah cara yang dapat

dipahami para *stakeholder* agar persyaratan-persyaratan itu dapat di validasi dan di verifikasi. Industri pengembangan perangkat lunak telah mempelajari bahwa untuk meraih sukses perencanaan, analisis, design, konstruksi, dan penyebaran sistem informasi, analisis sistem pertamanya harus memahami kebutuhan para *stakeholder* dan mengapa sistem harus dikembangkan- konsep yang disebut user-centered development/ pengembangan berpusatkan pengguna. Dengan fokus kepada pengguna sistem, analisis dapat berkonsentrasi untuk mengembangkan bagaimana sistem akan digunakan dan bukan bagaimana sistem dibangun.

Pemodelan use-case awalnya disusun oleh Dr. Ivar Jacobson pada tahun 1986 dan menjadi populer setelah beliau menerbitkan buku, *Object-Oriented Software Engineering*, pada tahun 1992. Dr Jacobson menggunakan pemodelan use-case sebagai kerangka kerja untuk metodologi *objectory*-nya dengan sukses digunakannya untuk mengembangkan sistem informasi berorientasi-objek. Penggunaan pemodelan use-case memfasilitasi dan mendorong keterlibatan pengguna, yang merupakan faktor sukses kritis untuk memastikan sukses proyek.

Ada dua alat utama yang digunakan saat menyajikan pemodelan use-case. Pertama adalah use-case diagram, yang secara grafis menggambarkan sistem sebagai sebuah kumpulan use-case, pelaku(pengguna), dan hubungan keduanya. Diagram ini mengkomunikasikan lingkup kejadian bisnis yang harus diproses oleh sistem. Detil setiap kejadian bisnis dan bagaimana pengguna dapat

berinteraksi dengan sistem digambarkan dalam artefak kedua yang dinamakan use-case narrative, yang merupakan deskripsi tekstual tentang kejadian bisnis dan bagaimana pengguna akan berinteraksi dengan sistem untuk menyelesaikan tugas.

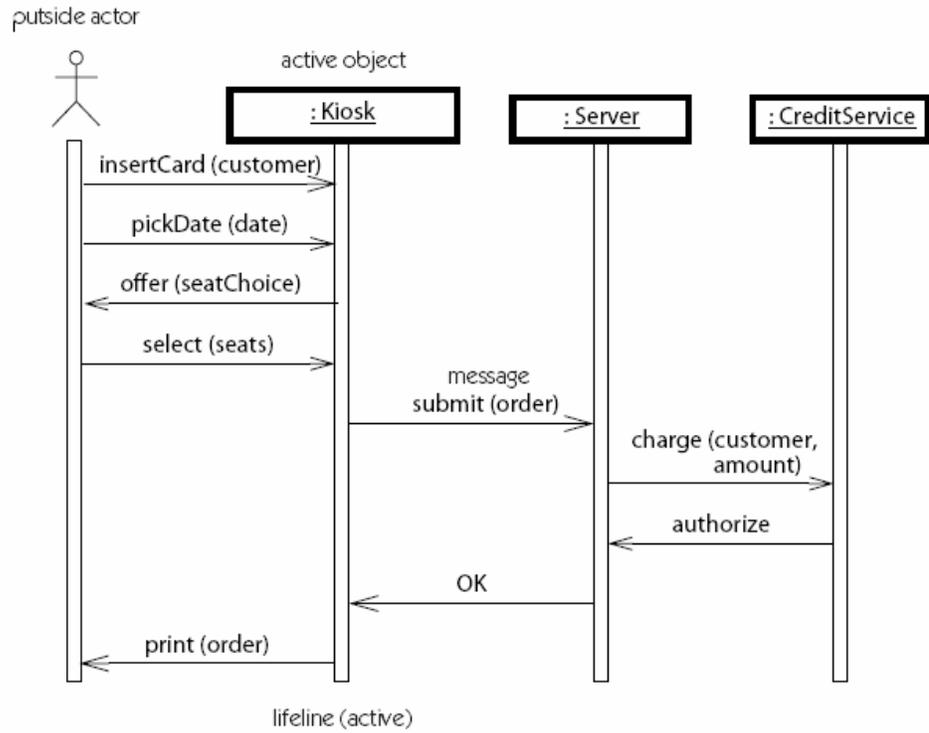


Gambar 2.8 Diagram Use Case

2.2.7.2 Sequence Diagram

Menurut Grady Booch (1998, p97) diagram sequence adalah diagram interaksi yang menekankan urutan waktu dalam pengiriman pesan. Sequence diagram menunjukkan interaksi objek dengan waktu yang direpresentasikan dalam grafik dua dimensi. Dimensi vertical menunjukkan waktu, digambarkan melintang kebawah. Dimensi Horizontal menunjukkan jenis peranan yang menggambarkan individu objek dalam diagram collaboration. Durasi aktivitas objek ditunjukkan

oleh lifeline yang berupa garis putus-putus. Message ditampilkan sebagai panah dari satu lifeline sebuah objek ke lifeline objek yang lainnya.



Gambar 2.9 Diagram Sequence