BAB II

LANDASAN TEORI

2.1. Pengertian Teknologi Informasi

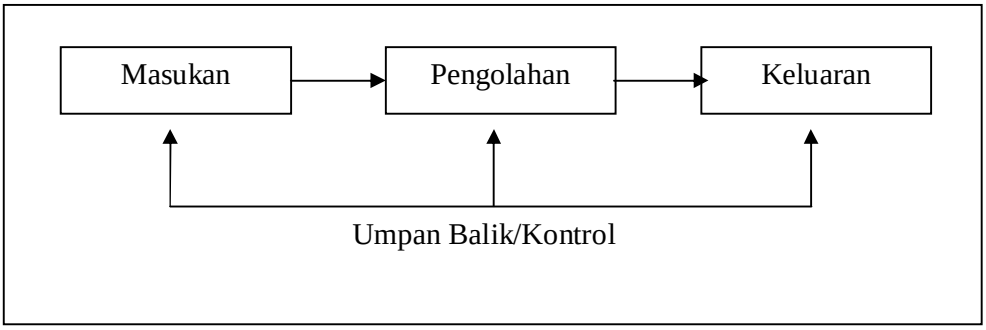
Teknologi informasi adalah serangkaian komputer dan alat pendukung lainnya untuk mengolah dan menyimpan informasi.

2.2. Konsep Dasar Sistem

2.2.1. Pengertian Sistem

Sistem menurut Webster's Unabridged adalah elemen – elemen yang saling berhubungan membentuk satu kesatuan atau organisasi.

2.2.2. Modul Sistem



Gambar 2.1 Modul Sistem

(Sumber : Manajemen Sistem Informasi, Drs. Zulkifli Amsyah, MLS. (2005))

Modul sistem terdiri dari empat elemen subsistem, yaitu :

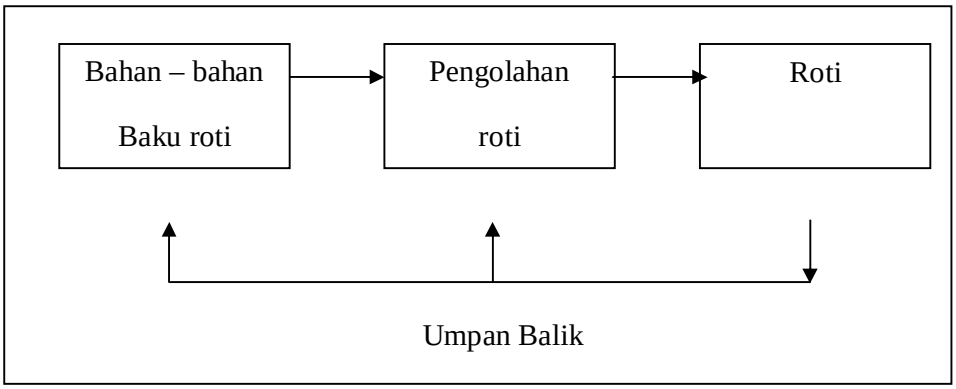
1. Masukan
2. Pengolahan
3. Keluaran
4. Umpan Balik/Kontrol.

2.2.3. Karakteristik Sistem

Karakteristik sistem untuk keperluan pekerjaan informasi, yaitu :

a. Sistem Fisik

Pada sistem fisik yang diolah adalah bentuk fisik, seperti pengolahan roti. Misalnya, masukan bahan mentahnya seperti tepung, gula, mentega dan lain – lain dicampur dan diolah dalam suatu tempat pengolahan (pabrik), maka dihasilkannya keluaran dalam bentuk roti.

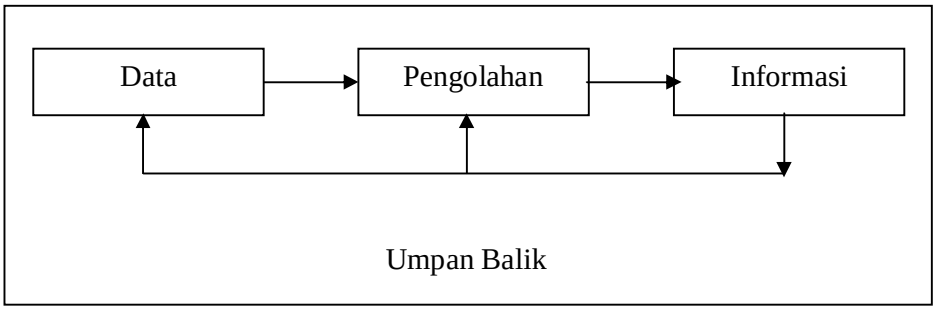


Gambar 2.2 Contoh Sistem Fisik

(Sumber : Manajemen Sistem Informasi, Drs. Zulkifli Amsyah, MLS. (2005))

b. Sistem Non – Fisik

Pada sistem non – fisik yang diolah adalah sesuatu yang abstrak, misalnya buah pikiran, pendapat, perhitungan, huruf – huruf, angka – angka, dan lain – lain yang kesemuanya bukan barang nyata. Contohnya adalah sistem informasi. Masukannya adalah data dalam bentuk angka – angka, huruf – huruf atau simbol – simbol. Pengolahannya adalah dengan mengklasifikasikan, menyortir, mengkalkulasikan atau menyimpulkan sehingga akhirnya dihasilkan suatu informasi.



Gambar 2.3 Contoh Sistem Non – Fisik

(Sumber : Manajemen Sistem Informasi, Drs. Zulkifli Amsyah, MLS. (2005))

2.3. Konsep Dasar Informasi

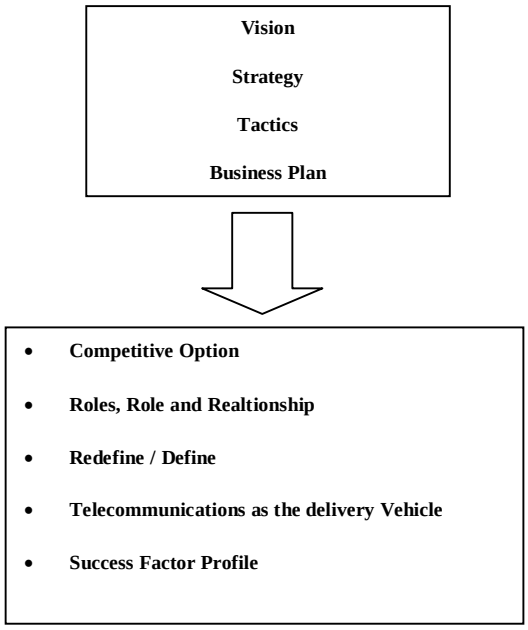
Menurut Gordon B. Davis Informasi adalah data yang sudah diproses menjadi bentuk yang berguna bagi pemakai dan mempunyai nilai pikir yang nyata bagi pembuat keputusan pada saat sedang berjalan atau untuk prospek masa depan.

Informasi harus memiliki nilai yang berkualitas, seperti akurat, cepat dan relevan. Jenis informasi dapat dikelompokan berdasarkan :

1. Isi Informasi
2. Bentuk Informasi
3. Keluaran Informasi

2.4. Konsep Sistem Informasi

Menurut Ubuh B. Hidayat (2003 : 33) bahwa *Information System* yang digunakan perusahaan akan mempengaruhi keberhasilan perusahaan, dimana visi, strategi, taktik dan *business plan* akan menentukan bagaimana peran *Information System* sebagaimana ditunjukkan gambar berikut:



Gambar 2.4 Peran Sistem Informasi dalam Bisnis

(Sumber : Jurnal Manajemen Teknologi)

2.5. Sistem Informasi Akademik

Sistem Informasi Akademik adalah suatu sistem yang dirancang untuk memenuhi keperluan dalam pengolahan data – data akademik dengan menggunakan teknologi komputer.

1. Pendaftaran Siswa Baru

   Pendaftaran siswa baru adalah proses pendaftaran siswa baru di sebuah sekolah dengan di lampiri beberapa persyaratan yang telah ditentukan oleh sekolah.

2. Pembagian Siswa Perkelas dan Wali Kelas

   Pembagian ini dilakukan untuk membagi siswa yang ada ke beberapa kelas agar sama rata dan peran wali kelas adalah sebagai khusus bimbingan dan konseling.

3. Jadwal Pelajaran

   Jadwal pelajaran bertujuan untuk menyesuaikan waktu dengan rencana pelaksana dalam menyampaikan materi untuk siswa.

4. Pengolahan Nilai

   Pengolahan nilai bertujuan untuk memperoleh informasi dalam kegiatan dan kemajuan belajar siswa di pelaksanaan kurikulum.

5. Daftar Ulang

   Daftar ulang bertujuan untuk mendaftarkan siswa setelah di bagikan raport dan naik kelas.

2.6. Java

Java merupakan bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini banyak mengadopsi sintaks yang terdapat pada C dan C++. Aplikasi - aplikasi yang berbasis java umumnya dikompilasi ke dalam *p-code (bytecode)* dan dapat dijalankan pada berbagai *Java Virtual Machine* (JVM).

Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa *platform* sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *website*.

2.6.1. Sejarah Java

Sejarah Java berawal pada tahun 1991 ketika perusahaan *Sun Microsystem* memulai *Green Project*, yaitu proyek penelitian untuk membuat bahasa yang akan digunakan pada chip – chip *embedded* untuk *device intelligent consumer electronic*. Bahasa tersebut haruslah bersifat *multi-platform*, tidak tergantung pada vendor yang memanufaktur chip tersebut.

Dalam penelitiannya, *Green Project* berhasil membuat *prototipe* semacam PDA (*Personal Data Assistance*) yang dapat berkomunikasi antara satu dengan yang lain dan diberi nama *Star 7*. Ide berawal untuk membuat sistem operasi bagi *Star 7* berbasis C dan C++. Setelah berjalan beberapa lama, James Gosling, salah

seorang anggota tim, merasa kurang puas dengan beberapa karakteristik dari kedua bahasa tersebut berusaha mengembangkan bahasa lain. Bahasa tersebut kemudian dinamakan Oak. Nama itu terinspirasi ketika ia melihat pohon di seberang kaca ruang kantornya. Kemudian Oak beralih nama menjadi Java.

2.6.2. Keunggulan Java

Java memiliki beberapa keunggulan bila dibandingkan dengan bahasa pemrograman lainnya. Keunggulan Java antara lain : Java bersifat sederhana dan relatif mudah, Java berorientasi pada objek, Java bersifat terdistribusi, Java bersifat *multiplatform*, Java bersifat *MultiThread*. (Hermawan, Benny, 2004 : 7)

Berikut adalah penjelasan dari keunggulan – keunggulan java yang telah disebutkan diatas.

1. Java bersifat sederhana dan relatif mudah

   Java dimodelkan sebagian dari bahasa C++, namun dengan memperbaiki beberapa karakteristik C++, seperti mengurangi kompleksitas beberapa fitur, penambahan fungsionalitas, serta penghilangan beberapa aspek pemicu ketidakstabilan sistem pada C++.

2. Java berorientasi pada objek

   Java adalah bahasa pemrograman yang berorientasi objek. Dalam memecahkan masalah, java membagi program menjadi objek–objek, kemudian memodelkan sifat dan tingkah laku masing-masing, setelah itu menentukan dan mengatur interaksi antara objek yang satu dengan yang lainnya.

3. Java bersifat terdistribusi

   Saat ini, sistem komputerisasi cenderung terdistribusi, mulai dari *workstation client, e-mail server, database server, web server, proxy server,* dan sebagainya.

4. Java bersifat *multiplatform*

   Java bersifat *multiplatform,* yakni dapat di- "terjemahkan" oleh Java Interpreter pada berbagai sistem operasi.

5. Java bersifat *MultiThread*

   Thread adalah proses yang dapat dikerjakan oleh program dalam suatu waktu. Java bersifat *MultiThread,* artinya dapat mengerjakan beberapa proses dalam waktu yang hampir bersamaan.

2.7. Java 2 Micro Edition (J2ME)

Sun memperkenalkan nama "Java 2" (J2) sebagai kelanjutan Java 1.2 pada Desember 1998. Konvensi nama baru ini diterapkan untuk semua edisi Java yaitu Standard Edition (J2SE), Enterprise Edition (J2EE), dan Micro Edition (J2ME).

Sun Microsystems telah mendefinisikan tiga *platform* java yang masing-masing diarahkan untuk tujuan tertentu dan untuk lingkungan komputasi yang berbeda-beda, yaitu:

a. *Standard Edition* (J2SE): Didesain untuk jalan pada komputer desktop dan komputer *workstations.*
b. *Enterprise Edition* (J2EE): Dengan *built-in* mendukung untuk *Servlets*, JSP, dan XML, edisi ini ditujukan untuk aplikasi berbasis *server.*

c. *Micro Edition* (J2ME): Didesain untuk piranti dengan memori terbatas, layar *display* terbatas dan *power* pemrosesan yang juga terbatas.

J2ME merupakan bentuk bahasa java yang dioptimalkan dan diarahkan untuk piranti kecil dan *mobile* seperti PDA dan *handphone*. J2ME dirancang oleh *Sun Microsystem*, pada mulanya dikembangkan di bawah *Java Community Process* sebagai JSR 68 (*Java Specification Request*), perbedaan kebutuhan dari J2ME telah menghasilkan JSR yang terpisah. Perusahaan *Sun Microsystem* menyediakan referensi untuk implementasinya. Pada tanggal 22 Desember 2006, *source code* dari J2ME telah bersifat *open source* dibawah lisensi GNU (*General Public Lisence*).

J2ME terbagi menjadi dua yaitu CDC dan CLDC. Pada CLDC (*Connected Limited Device Configuration*) umumnya untuk aplikasi java pada *handphone* seperti produk-produk Siemens, Nokia, Motorola dan lain-lain. Sedangkan CDC (*Connected Device Configuration*) umumnya digunakan pada perangkat dengan memori setidaknya 2 Megabyte.

2.7.1. Java Development Kit (JDK)

Java Development Kit (JDK) merupakan perangkat pengembangan dalam membangun sebuah aplikasi java. JDK memuat program – program dan *library* yang dibutuhkan untuk meng – *compile* dan me – *launch* program Java. Dua program utama yang ada di JDK adalah javac dan java.

2.7.2. Sun Java Wireless Toolkit

J2ME *Wireless Toolkit* ini digunakan untuk mengembangkan aplikasi Java yang telah dibuat. Dalam *Toolkit* ini terdapat API J2ME dan emulator beberapa perangkat *nirkabel*. *Toolkit* ini berfungsi untuk mengkompile dan mensimulasikan aplikasi pada emulator.

Keterangan direktori – direktori yang ada didalam *Sun Java Wireless Toolkit* adalah sebagai berikut.

Tabel 2.1 Keterangan Direktori *Sun Java Wireless Toolkit*

File/Direktori	Keterangan
appdb\	Direktori berisi file-file database, seperti file-file RMS file
apps\	Direktori yang berisi demo aplikasi dan aplikasi-aplikasi yang dibuat dengan KToolbar
bin\	Batch file dan file-file executable tool
docs\	Direktori yang berisi dokumtasi API
lib\midpapi.zip	Arsip kelas-kelas API CLDC dan MIDP. File-file ini digunakan pada proses kompilasi dan preverification
sessions\	Direktori default yang berisi profiling, monitoring memori dan session file pada monitoring network
wktlib\devices\	Direktori yang berisi file-file properti device-device yang digunakan pada emulator

2.8. NetBeans IDE

NetBeans IDE adalah IDE (*Integrted Development Kit*) *open source* yang ditulis sepenuhnya dengan bahasa Java menggunakan *platform* NetBeans. NetBeans IDE mendukung pengambangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis *Ant*, control versi, dan *refactoring*.

Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk

bahasa pemrograman java, *editing*, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan java dalam sekali *download*, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul – modul juga mengijinkan NetBeans untuk dikembangkan. Fitur – fitur baru, seperti dukungan bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan. Sebagai contoh, *Sun Studio*, *Sun Java Studio Enterprise*, dan *Sun Java Studio Creator* dari *Sun Microsistem* semuanya berbasis NetBeans IDE.

2.8.1. Sejarah NetBeans IDE

Pengembangan NetBeans diawali dari Xelifi, sebuah proyek mahasiswa tahun 1997 di bawah bimbingan Fakultas Matematika dan Fisika Universitas Charles, Praha. Sebuah perusahaan kemudian dibentuk untuk proyek tersebut dan menghasilkan versi komersial NetBeans IDE hingga kemudian dibeli oleh *Sun Microsistem* pada tahun 1999. *Sun* kemudian menjadikan NetBeans *open source* pada bulan Juni tahun 2000. Sejak itu komunitas NetBeans terus berkembang. NetBeans mengacu pada dua hal, yakni *platform* untuk pengembangan aplikasi desktop java, dan sebuah *Integrated Development Kit* (IDE) yang dibangun menggunakan *platform* NetBeans.

Platform NetBeans memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak moduler yang disebut ‘modul’. Sebuah modul adalah suatu arsip Java (*Java archive*) yang memuat kelas – kelas java untuk berinteraksi dengan NetBeans *Open* API dan *file* manifestasi yang mengidentifikasinya sebagai modul. Aplikasi yang dibangun dengan modul–modul dapat dikembangkan dengan menambahkan modul – modul baru. Karena modul dapat

dikembangkan secara independen, aplikasi berbasis *platform* NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerfull* .

2.9. MySQL

MySQL adalah sebuah *database server* yang dapat berperan sebagai *client* sehingga disebut database *client/server*, yang *open source* dengan kemampuan dapat berjalan baik di operating sistem dengan *platfrom windows* maupun *linux*, sedangkan *SQL* adalah bahasa perintah dalam (*Query*) program *MySQL* ada beberapa yang mengunakan program *database* standar *query* berupa *SQL* seperti *Oracle, PostgresSQL, MySQL front, MsQL,SQL Server 97* dll. Beberapa kelebihan MySQL diantaranya : *MySQL* sebagai *Database Management System* (DBMS).

2.10. Jaringan Komputer

Sebuah sistem yang terdiri dari beberapa komputer dan peralatan penunjang lainnya yang saling terhubung sehingga dapat saling berbagi dan bertukar informasi.

2.10.1. Jenis – jenis Jaringan Komputer

Jenis – jenis jaringan komputer, yaitu :

1. LAN (*Local Area Network*)

   Yaitu jaringan lokal yang dibuat pada area tertutup. Contohnya dalam satu gedung atau dalam satu ruangan.

2. MAN (*Metropolitan Area Network*)

   Menggunakan metode yang sama dengan LAN namun dalam cakupannya lebih luas. Contohnya beberapa kantor yang berbeda dalam kompleks yang sama, satu kota, bahkan satu provinsi.

3. WAN (*Wide Area Network*)

   Cakupannya lebih luas dari pada MAN. Cakupan WAN meliputi satu kawasan, satu Negara, satu pulau, bahkan satu benua.

4. GAN (*Global Area Network*)

   Cakupannya lebih luas dari WAN, yaitu meliputi Negara – Negara di dunia.

2.10.2. Topologi Jaringan Komputer

Topologi adalah suatu aturan bagaimana cara menghubungkan komputer satu sama lain secara fisik dan pola hubungan antara komponen – komponen yang berkomunikasi melalui media atau peralatan jaringan yang meliputi *server, workstation, switch* dan pengkabelannnya. Media seperti *Twisted pair, coaxial cable optical cable* dan *wireless* biasanya digunakan di setiap topologi. Topologi ditinjau dari segi keberadaannya dapat diklasifikasian menjadi 2, yaitu :

1. Topologi Logikal

   Topologi ini menjelaskan aliran *message* atau data dari satu *user* ke *user* lainnya dalam jaringan. Topologi logikal merupakan penentuan bagaimana media diakses oleh *host* untuk mengirim sebuah data. Topologi logikal merupakan *rules communication* yang dipakai *station* dalam

berkomunikasi dalam *network*. Topologi logikal hanya akan beroperasi pada topologi fisikal tertentu. Sebagai contoh, *ethernet* hanya akan beroperasi pada *Bus, Star* dan *Point to point* dari topologi fisikal. Tapi tidak akan bekerja pada *ring*. Ada 2 metode untuk mengakses media ini yakni *broadcast (undeterministic)* dan *token pasing (deterministic)*. Jenis-jenis topologi logikal, yaitu :

a) FDDI (*Fiber Distributed-Data Interface*)

Adalah standar komunikasi data menggunakan *fiber optic* pada LAN dengan panjang sampai 200 km. Protokol FDDI berbasis pada protokol *Token Ring*. FDDI terdiri dari dua *Token Ring* , yang satu *ring* -nya berfungsi sebagai *ring backup* jika seandainya ada *ring* dari dua *ring* tersebut yang putus atau mengalami kegagalan dalam bekerja. Sebuah *ring* FDDI memiliki kecepatan 100 Mbps.

b) Token Ring

Adalah sebuah cara akses jaringan berbasis teknologi *ring* yang pada awalnya dikembangkan dan diusulkan oleh Olaf Soderblum pada tahun 1969. Perusahaan IBM selanjutnya membeli hak cipta dari *Token Ring* dan memakai akses *Token Ring* dalam produk IBM pada tahun 1984. Elemen kunci dari desain *Token Ring* milik IBM ini adalah penggunaan konektor buatan IBM sendiri *(proprietary)*, dengan menggunakan kabel *twisted pair*, dan memasang hub aktif yang berada di dalam sebuah jaringan komputer.

c) Ethernet.

Ethernet bekerja berdasarkan *broadcast*. *Ethernet* merupakan jenis skenario perkabelan dan pemrosesan sinyal untuk data jaringan komputer yang dikembangkan oleh Robert Metcalfe dan David Boggs di Xerox Palo Alto Research Center (PARC) pada tahun 1972 .

2. Topologi Fisikal

Topologi ini menjelaskan hubungan perkabel dan lokasi *node* atau *workstation*. Topologi fisikal adalah suatu bentuk bagaimana media transmisi dihubungkan secara bersama-sama. Topologi ini menggambarkan hubungan fisik antara perangkat-perangkat yang ada. Jenis-jenis topologi fisikal, yaitu :

a) Topologi Bus

Topologi ini memiliki karakteristik sebagai berikut:

1. Merupakan satu kabel yang kedua ujungnya ditutup, dimana sepanjang kabel terdapat *node – node*.
2. Umum digunakan karena sederhana dalam instalasi.
3. *Signal* melewati kabel dalam dua arah dan mungkin terjadi *collision*.
4. *Problem* terbesar pada saat kabel putus. Jika salah satu segmen kabel putus, maka seluruh jaringan akan terhenti.

b) Topologi Ring

Topologi dimana satu *host* terkoneksi ke *host* berikutnya dan *host* terakhir terkoneksi kembali ke *host* pertama. Topologi ini mempunyai karakteristik sebagai berikut:

1. Lingkaran tertutup yang berisi *node – node*.
2. Sederhana dalam *layout*.
3. *Signal* mengalir dalam satu arah, sehingga dapat menghindarkan terjadinya *collision* (dua paket data bercampur), sehingga memungkinkan pergerakan data yang cepat dan *collision detection* yang lebih sederhana.
4. *Problem* sama dengan topologi *bus*.
5. Biasanya topologi *ring* tidak dibuat secara fisik melainkan direalisasikan dengan sebuah *consentrator* dan kelihatan seperti topologi *star*.

c) Topologi Star

Topologi yang menghubungkan semua *host* kesatu titik pusat. Topologi ini mempunyai karakteristik sebagai berikut:

1. Setiap *node* berkomunikasi langsung dengan *central node, traffic data* mengalir dari *node* ke *central node* dan kembali lagi.
2. Mudah dikembangkan, karena setiap *node* hanya memiliki kabel yang langsung terhubung ke *central node*.
3. Keunggulannya adalah jika satu kabel *node* terputus yang lainnya tidak terganggu.

4. Dapat digunakan kabel yang "*lower grade*" karena hanya menghandel satu *traffic node*, biasanya digunakan kabel UTP.

d) Topologi Extended Star

Merupakan perkembangan lanjutan dari topologi *star* dimana karakteristiknya tidak jauh berbeda dengan topologi *star* yaitu :

1. Setiap *node* berkomunikasi langsung dengan *sub node*, sedangkan *sub node* berkomunikasi dengan *central node. traffic* data mengalir dari *node* ke *sub node* lalu diteruskan ke *central node* dan kembali lagi.
2. Digunakan pada jaringan yang besar dan membutuhkan penghubung yang banyak atau melebihi dari kapasitas maksimal penghubung.
3. Keunggulan jika satu kabel *sub node* terputus maka *sub node* yang lainnya tidak terganggu, tetapi apabila *central node* terputus maka semua *node* disetiap *sub node* akan terputus.
4. Tidak dapat digunakan kabel yang "*lower grade*" karena hanya menghandel satu *traffic node*, karena untuk berkomunikasi antara satu *node* ke *node* lainnya membutuhkan beberapa kali *hops.*

e) Topologi hierarchical

Topologi ini biasa disebut sebagai topologi *tree*. Dibangun oleh seperti halnya topologi *extended star* yang dihubungkan melalui *sub node* dalam satu *central node*. Topologi ini dapat mensupport baik *baseband* maupun *broadband* signaling dan juga mensupport baik *contention* maupun *token bus access.*

f) Topologi Mesh

Topologi *mesh* dibangun dengan memasang *link* diantara *station-station*. Sebuah "*fully-connected mesh*" adalah sebuah jaringan dimana setiap terminal terhubung secara langsung ke semua terminal – terminal yang lain. Biasanya digunakan pada jaringan komputer kecil. Topologi ini secara teori memungkinkan, akan tetapi tidak praktis dan biasanya cukup tinggi untuk di-implementasikan. Topologi *mesh* memiliki tingkat *redundancy* yang tinggi. Sehingga jika terdapat satu *link* yang rusak maka suatu *station* dapat mencari *link* yang lainnya.