BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan (Mcload, 2004).**[1,p.1]**

Terdapat dua kelompok pendekatan di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya.

Pengertian sistem adalah yang lebih menekankan pada prosedurnya didefinisikan sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang tertentu. Sedangkan pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urut-urutan operasi di dalam sistem.

‘ Prosedur adalah suatu urut-urutan operasi *klerikal* (tulis-menulis) biasanya melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis terjadi.’ **[2, p. 1]**

2.1.1 Bentuk Dasar Sistem

Menurut Jogiyanto, sistem dapat didefinisikan kedalam 2 (dua) pendekatan yaitu Pendekatan yang menekankan pada prosedurnya dan pendekatan yang

menekankan pada elemennya. Pendekatan yang lebih menekankan pada prosedurnya sistem dapat didefinisikan sebagai berikut :

’Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang Saling berhubungan, berkumpul bersama-sama melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu’. **[2, p. 1]**

Pendekatan yang lebih menekankan pada elemennya sistem dapat didefinisikan sebagai berikut:

’Sistem adalah kumpulan dari elemen-elemen yang saling berinteraksi untuk mencapai tujuan tertentu.’ **[2, p. 2]**

Berikut beberapa elemen untuk membangun sebuah sistem yaitu :

a. Tujuan dari suatu sistem bisa hanya satu atau banyak., tujuan inilah yang menjadi motivator bagi sistem dan sekaligus mengarahkan sistem Tanpa tujuan sistem menjadi tak terarah dan terkendali. Tujuan antara sistem satu dengan sistem yang lain berbeda-beda.

b. Masukan (*input*) adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses.

c. Proses merupakan bagian yang melakukan perubahan atau *transfromasi* dari masukan menjadi keluaran yang berguna.

d. Keluaran (*output*) merupakan hasil dari pemrosesan., keluaran memiliki nilai lebih (*value added*) dari masukannya.

e. Mekanisme Pengendalian dan Umpan Balik merupakan mekanisme pengendalian (*control mechanism*) dapat diwujudkan dengan adanya umpan balik (*feed back*) yang menganalisa keluaran apakah telah memenuhi tujuan. Hal ini digunakan untuk mengendalikan baik masukan maupun proses agar menghasilkan keluaran yang sesuai tujuan.

2.1.2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung sistem, masukan sistem, keluaran sistem, pengolah sistem, sasaran sistem. Berikut ini yang merupakan karakteristik sistem adalah sebagai berikut :

1. Komponen Sistem (*Component*)

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerja sama membentuk satu kesatuan.

2. Batasan Sistem (*Boundary*)

   Merupakan daerah yang membatasi antara satu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sisten (*Environment*)

   Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengruhi operasi sistem.

4. Penghubung Sistem (*Interface*)

   Merupakan media penghubung antara satu subsistem dengan subsistem lainnya.

5. Masukan Sistem (*Input*)

   Masukan adalah energi yang dimasukan kedalam sistem berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

   Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolahan Sistem (*Objective*)

   Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem (*objective*)

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. **[3, p. 4]**

2.1.3 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainya, karena sistem memiliki sasaran yang berbeda untuk setiap kasus

yang terjadi yang ada didalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandangan yaitu :

1. Sistem Abstrak (*Abstract System*) dan Sistem Fisik (*Physical System*)

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, sedangkan sistem fisik merupakan sistem yang ada secara fisik.

2. Sistem Alamiah (*Natural System*) dan Sistem Buatan Manusia (*Human Made System*).

   Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin yang disebut dengan *human machine system*.

3. Sistem Tertentu (*Deterministic System*) dan Sistem Tak tentu (*Probabilistic System*).

   Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sedangkan sistem tak tentu merupakan sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung sistem probabilitas.

4. Sistem Tertutup (*Closed Sytsem*) dan Sistem Terbuka (*Open System*).

   Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya, bekerja secara otomatis tanpa adanya

turut campur tangan dari pihak luarnya. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau sub sistem lainya. **[1.p.4]**

2.2 Pengertian Informasi

Menurut (Mcload, 2004) **[1.p.8]** Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Berdasarkan pendapat diatas informasi dapat diartikan hasil dari sebuah data yang berari dan diolah sehingga menghasilkan manfaat dan menambah ilmu pengetahuan bagi yang menerimanya.

2.2.1 Konsep Dasar Informasi

Informasi adalah data yang telah diolah menjadi suatu bentuk yang lebih berarti dan berguna bagi si penerima untuk mengambil keputusan masa kini maupun yang akan datang. mendefinisikan informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti bagi penerimanya. **[3, p. 8]**

Alat pengolah informasi dapat meliputi elemen komputer, elemen bukan komputer atau kombinasinya. Kegunaan informasi adalah untuk mengurangi ketidakpastian didalam proses pengambilan keputusan tentang suatu keadaan. Informasi yang digunakan didalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut. **[3, p. 9]**

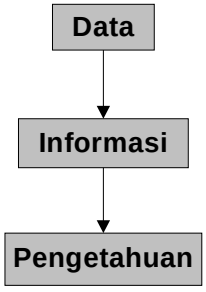
Sistem informasi tidak akan berjalan atau bekerja apabila tidak ada sesuatu yang menggerakan sistem informasi tersebut :

1. *Hardware* dan *Software* yang berfungsi sebagai mesin.
2. *User* (manusia) dan *procedures* (prosedur) yang merupakan manusia dan tatacara menggunakan mesin.
3. Data merupakan jembatan penghubung antara manusia dan mesin agar terjadi suatu proses pengolahan data.

2.2.2 Hirarki Informasi

Sumber informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum atau data item.

Informasi tanpa adanya data maka informasi tersebut tidak akan terbentuk. Begitu pentingnya peranan data dalam terjadinya suatu informasi yang berkualitas. Keakuratan data sangat mempengaruhi terhadap keluaran informasi yang akan terbentuk.



Gambar 2.1 Hirarki Informasi
(Sumber: Analisis dan Desain **[2,p.9]**)

2.2.3 Siklus Informasi

Menurut (Jogianto 1999) **[1.p.11]** siklus informasi dapat digambarkan sebagai berikut :

1. Pertama data dimasukan kedalam model yang umumnya memiliki urutan proses tertentu dan pasti, setelah diproses akan menghasilkan nilai yang bermanfaat bagi penerima sebagai dasar dalam pembuatan keputusan atau melakukan tindakan tertentu.
2. Kemudian dari keputusan atau tindakan tersebut dapat menghasilkan atau diperoleh kejadian-kejadian tertentu yang akan digunakan kembali sebgai data yang nantinya akan dimasukan kedalam (proses), dan akan begitu seterusnya.

2.2.4 Kualitas Informasi

Kualitas suatu informasi dapat dilihat dari dimensi-dimensi yang dimiliki informasi. Menurut (Jogianto, 1999) **[1.p.9]** kualitas dari informasi bergantung dari tiga hal yaitu :

a. *Relevan*, berarti informasi tersebut mempunyai manfaat untuk pemakainya dan *relevasnsi* informasi untuk setiap orang akan berbeda-beda.
b. Tepat waktu, berarti informasi tersebut dating pada penerima tidak boleh terlambat. Informasi yang sudah using tidak akan mempunyai

nilai lagi, karena informasi merupakan landasan didalam pengambilan keputusan.

c. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksud. Informasi harus akurat karena dari sumber informasi samapai kepenerima informasi kemungkinan banyak tejadi gangguan yang dapat merusak informasi.

2.2.5 Nilai Informasi

Menurut (Jogianto, 1999) **[1.p.9]** nilai dari informasi ditentukan dari dua hal yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan biaya mendapatkannya. Informasi tidak dapat ditaksir keuntungannya dengan nilai uang, tetapi dapat dinilai dengan nilai efektifnya.

2.3 Sistem Informasi

Sistem informasi adalah kombinasi terartur dari orang-orang , perangkat keras, perangkat lunak, jaringan komunikasi dan sumberdaya data yang mengumpulkan, mengubah dan menyebarkan informasi dalam sebuah organisasi. (O'Brian, 2005) **[1.p.17]**.

2.3.1 Pengertian Sistem Informasi

Sistem informasi dapat didefinisikan sebagai berikut: **[3, p. 13]**

1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi.
3. Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.3.2 Komponen Sistem Informasi

Komponen sistem informasi merupakan sebuah susunan yang terdiri dari beberapa komponen atau elemen. Berikut komponen pada system informasi :

a. Blok Masukan, Input memiliki data yang masuk kedalam system informasi, juga metode-metode untuk menangkap data yang dimasukkan.
b. Blok model, terdiri dari kombinasi prosedur logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan dibasis data.
c. Blok keluaran, produk dari system informasi adalah keluaran yang merupakan informasi berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai system.
d. Blok teknologi, digunakan untuk menerima input, menyimpan, mengakses data, menghasilkan dan mengirimkan keluaran dari system secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu : teknisi, perangkat lunak, dan perangkat keras.

e. Basis data, merupakan kumpulan data yang saling berhubungan satu dengan yang lainny, tersimpan di perangkat keras computer dan digunakan perangkat lunak untuk memanipulasi. **[1.p.20]**

2.4 Analisa & Perancangan Sistem

Analisis sistem sangat bergantung pada teori sistem umum sebagai sebuah landasan konseptual. Tujuannya dalah untuk memperbaiki berbagai fungsi didalam sistem yang sedang berjalan agar menjadi lebih efisien, mengubah sasaran sistem yang sedang berjalan, merancang atau mengganti output yang sedang digunakan, untuk mencapai tujuan yang sama dengan seperangkat input yang lain. Tahapan dalam menganalisis sistem adalah sebagai berikut:

1. Definisi masalah (mencakup definisi *input*, proses, *output* dari sistem yang sedang berjalan dan sistem yang akan dibangun atau dikembangkan).
2. Memahami sistem yang sedang berjalan tersebut dan membuat definisinya (mendefinisikan *input*, proses, *output*).
3. Alternatif apa saja yang tersedia untuk mencapai tujuan dengan memperhatikan modifikasi sistem tersebut.
4. Pilih salah satu alternatif yang telah dirumuskan pada tahap sebelumnya.
5. Implementasikan alternatif terpilih dari sekian alternatif yang telah ditawarkan tersebut.
6. Mengevaluasi dampak yang ditimbulkan akibat perubahan yang telah dilakukan terhadap sistem.

2.4.1 Perancangan Sistem

Perancangan Sistem adalah tahapan setelah analisis sistem dan siklus pengembangan sistem, pendefinisian kebutuhan-kebutuhan fungsional dan persiapan untuk merancang dengan implementasi, mengambarkan suatu sistem yang akan dibentuk. Pada tahapan ini akan dirancang (*software*) dengan memanfaatkan alat bantu:

1. *Use case* diagram

Use case menjelaskan urutan kegiatan yang dilakukan aktor dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan, namun *use case* hanya menjelaskan ”apa” yang dilakukan oleh aktor dan sistem bukan ”bagaimana” aktor dan sistem melakukan kegiatan tersebut.

1.1 *Use-case* Konkret adalah use case yang dibuat langsung karena keperluan aktor. Aktor dapat melihat dan berinisiatif terhadapnya.

Aktor adalah sesuatu (entitas) yang berhubungan dengan sistem dan berpartisipasi dalam *use case*. Aktor menggambarkan orang, sistem atau entitas eksternal yang secara khusus membangkitkan sistem dengan input atau masukan kejadian-kejadian, atau menerima sesuatu dari sistem.

Dalam *use case* diagram terdapat satu aktor pemulai atau *initiator* aktor yang membangkitkan rangsangan awal terhadap sistem, dan mungkin sejumlah

aktor lain yang berpartisipasi atau *participating* aktor. Akan sangat berguna untuk mengetahui siapa aktor pemulai tersebut.

1.2 *Use-case Abstrak* adalah *use case* yang tidak pernah berdiri sendiri. *Use cas*e abstrak senantiasa termasuk didalam (*include*), diperluas dari (*extend*) atau memperumum (*generalize*) *use case* lainnya.

Untuk menggambarkannya dalam use case model biasanya digunakan *association relationship* yang memiliki *stereotype include, extend* atau *generalization relationship*.

a. *Include*, pemanggilan *use case* oleh *use case* lain atau untuk menggambarkan suatu use case termasuk di dalam use case lain (diharuskan). Contohnya adalah pemanggilan sebuah fungsi program. Digambarkan dengan garis lurus berpanah dengan tulisan <<*include*>>.
b. *Extend,* digunakan ketika hendak menggambarkan variasi pada kondisi perilaku normal dan menggunakan lebih banyak kontrol form dan mendeklarasikan ekstension pada *use case* utama. Atau dengan kata lain adalah perluasan dari *use case* lain jika syarat atau kondisi terpenuhi. Digambarkan dengan garis berpanah dengan tulisan <<*extend*>>.
c. *Generalization/Inheritance*, dibuat ketika ada sebuah kejadian yang lain sendiri atau perlakuan khusus dan merupakan pola berhubungan *base-parent use case*. Digambarkan dengan garis berpanah tertutup dari *base use case* ke *parent use case*.

Relasi (*relationship*) digambarkan sebagai bentuk garis antara dua simbol dalam use case diagram. Relasi antara aktor dan use case disebut juga dengan asosiasi *(association)*. Asosiasi ini digunakan untuk menggambarkan bagaimana hubungan antara keduanya.

Relasi-relasi yang terjadi pada use case diagram bisa antara aktor dengan *use case* atau *use case* dengan *use case*.

2. Skenario *use case*

1) Menjelaskan detail langkah-langkah untuk masing – masing *use case*

2) Disini menggambarkan identifikasi kebutuhan dan skenario utama

3) Pada skenario utama akan terlihat aksi aktor dan reaksi sistem

4) Selain itu pada skenario utama menjelaskan kondisi awal dan kondisi akhir

3. *Activity* diagram

Diagram aktivitas menggambarkan proses bisnis dan urutan aktivitas-aktivitas yang mendukung penggambaran tindakan sistem baik yang bersifat kondisional maupun paralel. Tindakan kondisional dilukiskan dengan cabang (*branch*) dan penyatuan (*merge*).

Sebuah *branch* memiliki sebuah *transition* masuk atau yang disebut dengan *incoming transition* dan beberapa *transition* keluar atau yang disebut dengan *outgoing transition* dari *branch* yang berupa keputusan-keputusan. Hanya satu dari *outgoing transition* yang dapat diambil, maka keputusan-keputusan tersebut harus

bersifat *mutually exclusive*. [*else*] digunakan sebagai keterangan singkat yang menunjukkan bahwa transition "*else*" tersebut harus digunakan jika semua keputusan yang ada pada branch salah.

Sebuah *merge* memiliki banyak *input transition* dan sebuah *output*. *Merge* menandakan akhir dari suatu kondisi yang diawali dengan sebuah *branch.*

Selain *branch* dan *merge,* di dalam diagram aktivitas terdapat *fork* dan *join*. *Fork* memiliki satu *incoming transition* dan beberapa *outgoing transition.* Sedangkan pada *join, outgoing transition* diambil atau digunakan hanya ketika semua *state* pada *incoming transition* telah menyelesaikan aktivitasnya.

4. *Sequence* diagram

Sebuah diagram yang menggambarkan interaksi antar objek di dalam sebuah sistem. Interaksi tersebut berupa pesan yang digambarkan terhadap waktu. Sequence diagram terdiri dari dimensi *horizontal* (objek-objek) dan dimensi *vertical* (waktu).

a. Mirip dengan activity diagram
b. Menggambarkan alur kejadian sebuah aktivitas
c. Lebih detail dalam menggambarkan aliran data, termasuk data atau behaviour yang dikirimkan atau diterima
d. Namun kurang mampu menjelaskan detail dari sebuah algoritma (*loop, branching*)

5. *Collaboration* diagram

a. Diagram Interaksi

b. Fokus pada relasi-relasi yang terjadi antara objek yang satu dengan objek-objek yang lainnya

c. Interaksi antara objek dan aktor tidak ditampilkan berdasarkan urutan waktu (diagram *Sequence*)

6. *Class* diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (*atribut/properti*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram memberikan pandangan secara luas dari suatu sistem dengan menunjukan kelas-kelasnya dan hubungan mereka

Secara garis besar terdapat 3 jenis *class*. Ketiga jenis *class* tersebut dikelompokkan berdasarkan fungsi dan karakternya masing-masing, yaitu :

a. *Entity Class* Diagram

Merupakan paket utama dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk sistem dan menjadi landasan untuk menyusun basis data pada model data konseptual.

b. *Control Class* Diagram

Berisi kumpulan kelas yang menjadi kontrol program termasuk koneksi dengan basis data dan merupakan kelas perantara atau penghubung antara *entity class* dengan kelas antar muka pemakai *(interface)*.

c. *Boundary Class* Diagram

Berisi kumpulan kelas yang menjadi *interface* antara pemakai (user) dengan sistem, seperti tampilan *form* untuk pencetakan.

2.4.2 Desain Sistem

Desain sistem didefenisikan sebagai berikut: "Desain Sistem adalah penggambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi."

1. Tahap setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian dari kebutuhan-kebutuhan fungsional.
3. Persiapan untuk rancang bangun implentasi.
4. Menggambarkan bagaimana suatu sistem dibentuk.

Sistem dibentuk dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem. (Josh Burch dan Gery Grudnitski).

2.4.3 Tujuan Desain Sistem

Tahap desain sistem mempunyai dua maksud atau tujuan utama, yaitu sebagai berikut :

a. Untuk memenuhi kebutuhan kepada pemakai sistem.

b. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknik lainnya yang terlibat.

Untuk mencapai tujuan ini, analis sistem harus dapat mencapai sasaran-sasaran sebagai berikut :

1. Desain sistem harus berguna, mudah dipahami dan nantinya mudah digunakan.Ini berarti bahwa data harus mudah ditangkap, metode-metode harus mudah diterapkan dan informasi harus mudah dihasilkan serta mudah dipahami dan digunakan.

2. Desain sistem harus dapat mendukung tujuan utama perusahaan sesuai dengan yang didefinisikan pada tahap perencanaan sistem yang dilanjutkan pada tahap analisis sistem.

3. Desain sistem harus efisien dan efektif untuk dapat mendukung pengolahan transaksi, pelaporan manajemen dan mendukung keputusan yang akan dilakukan oleh manajemen, termasuk tugas-tugas yang lainnya yang tidak dilakukan oleh computer.

2.4.4 Perancangan Basis Data

Basis Data Merupakan sekumpulan data yang saling berhubungan yang disimpan secara bersamaan sedemikian rupa tanpa pengulangan *(redudansi)* yang tidak perlu untuk memenuhi berbagai keperluan dalam media penyimpanan elektronik. **[6, p. 10]**

Komponen-komponen utama sebuah sistem basis data :

1. Perangkat Keras (*Hardware*)
2. Sistem Operasi (*Operating Sistem*)
3. Basis Data (*Data Base*)
4. Sistem Pengolahan Basis Data
5. Pemakai (*User*)
6. Aplikasi (perangkat lunak) lain (bersifat opsional)

2.5 Jaringan Komputer

Untuk mendukung terlaksananya program aplikasi yang akan dibangun ini maka diperlukan beberapa unit komputer, dimana semua komputer tersebut saling berhubungan dan melakukan komunikasi data agar proses komunikasi data dapat berjalan lancar dengan baik melalui media atau perantara yang disebut dengan jaringan komputer.

2.5.1 Pengertian Jaringan Komputer

Jaringan komputer adalah hubungan dua buah simpul (umumnya berupa komputer) atau lebih yang tujuan utamanya adalah untuk melakukan pertukaran data. Dalam prakteknya, jaringan komputer memungkinkan untuk melakukan

berbagi (*sharing*) perangkat lunak, perangkat keras, dan bahkan berbagi (*sharing*) kekuatan pemrosesan. **[5, p. 150]**

2.5.2 Jenis-jenis Jaringan Komputer

Menurut Budi Irawan **[7, p. 19]**, jaringan komputer dapat dibedakan berdasarkan cakupan geografisnya. Ada empat kategori utama jaringan komputer, yaitu :

1. LAN (*Local Area Network*)

LAN adalah jaringan lokal yang dibuat pada area tertutup. LAN digunakan untuk menghubungkan komputer yang berada dalam suatu area yang kecil, jarak antara komputer yang dihubungkannya hanya bias mencapai 5 sampai 10 km. Suatu LAN biasanya bekerja pada kecepatan 10 sampai 100 Mbps. LAN menjadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya secara bersama-sama seperti penggunaan *printer* dan penggunaan media penyimpanan.

2. MAN (*Metropolitan Area Network*)

MAN merupakan suatu jaringan yang menggunakan metode sama dengan LAN namun cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN biasa mencapai 10 sampai beberapa ratus km. Suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps.

3. WAN (*Wide Area Network*)

WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas, seperti hubungan dari satu kota ke kota

lain dalam suatu negara. Cakupan WAN dapat meliputi 100 sampai 1.000 km, dan kecepatan antar kota bisa bervariasi antara 1,5 Mbps sampai 2,4 Gbps.

4. GAN (*Global Area Network*)

GAN merupakan jaringan yang menghubungkan negara-negara di seluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps sampai 100 Gbps dan cakupannya mencapai ribuan kilometer.

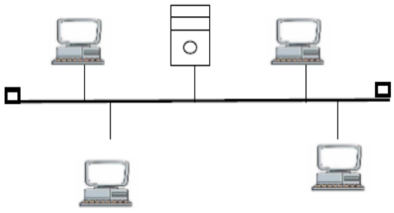
2.5.3 Topologi Jaringan Komputer

Topologi adalah suatu aturan bagaimana menghubungkan komputer satu sama lain secara fisik dan pola hubungan antara komponen-komponen yang berkomunikasi melalui media atau peralatan jaringan. Ada dua jenis topologi, yaitu *physical topology* (topologi fisik) dan *logical topology* (topologi logika). Topologi fisik berkaitan dengan bentuk jaringan, seperti bagaimna memilih perangkat dan melakukan pemasangan perangkat jaringan. Sedangkan topologi logika berkaitan dengan bagaimana data mengalir di dalam topologi fisik.

Topologi fisik jaringan yang digunakan di dalam jaringan lokal diantaranya Budi irawan **[7, p. 26]**:

a. Linear Bus (Garis lurus)

Topologi linear bus terdiri dari satu jalur kabel utama dimana masing-masing ujungnya diberikan sebuah terminator. Semua nodes pada jaringan terkoneksi pada sebuah kabel utama.



Gambar 2.2 Topologi *Linear Bus*
(Sumber: Jaringan Komputer (2005:26))

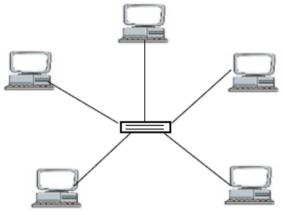
Kelebihan topologi linear bus :

1. Mudah di dalam mengkonfigurasi komputer atau perangkat lain ke dalam sebuah kabel utama.
2. Tidak terlalu banyak menggunakan kabel dibandingkan dengan topologi *star*/bintang.
3. Seluruh jaringan akan mati jika ada kerusakan pada kabel utama (*backbone*).
4. Membutuhkan *terminator* pada kedua sisi dari kabel utamanya
5. Sangat sulit mengidentifikasi kesalahan jika jaringan sedang jatuh atau rusak.
6. Sangat tidak disarankan dipakai sebagai salah satu solusi pada penggunaan jaringan digedung besar.

b. Star (Bintang)

Topologi model ini dirancang yang mana setiap *nodes* (*file server, workstation* dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator*. Data yang dikirim ke jaringan lokal akan melewati *concentrator*

sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengendalikan keskuruhan fungsi jaringan dan juga bertindak sebagai *repeater* (penguat aliran data).



Gambar 2.3 Topologi *Star*
(Sumber: Jaringan Komputer**)**

Kelebihan dari topologi *star* (bintang)

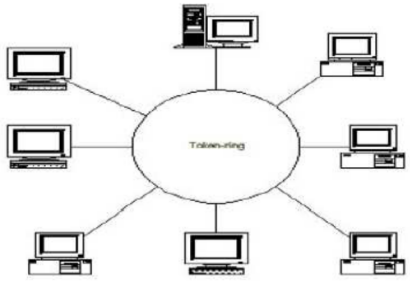
1. Mudah di dalam pemasangan dan pengkabelan.
2. Tidak mengakibatkan gangguan pada jaringan ketika akan memasang atau memindahkan perangkat jaringan lainnya.
3. Mudah untuk mendeteksi kesalahan dan memindahkan perangkat-perangkat lainnya.

Kekurangan topologi star (bintang) :

1. Membutuhkan lebih banyak kabel daripada topologi *linear bus*
2. Membutuhkan *concentrator* dan bilamana *concentrator* tersbut rusak maka semua node yang terkoneksi tidak dapat terdeteksi.
3. Lebih mahal daripada topologi linear bus karena biaya untuk pengadaan *concentrator.*

c. *Ring* (cincin)

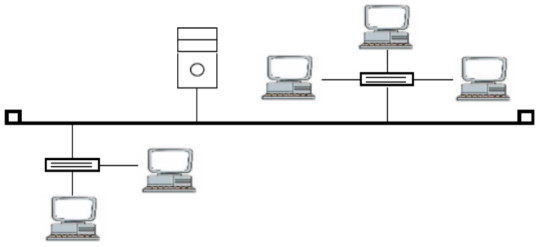
Topologi *ring* (cincin) mnggunakan tekhnik konfigurasi yang sama dengan topologi *star* tetapi pada topologi ini terlihat bahwa jalur media transmisi menyerupai suatu lingkaran trtutup menyerupai cincin (lingkaran) sehingga diberi nama topologi bintang dalam lingkaran atau *star – wired ring*



Gambar 2.4 *Topologi* Cincin
(Sumber: Jaringan Komputer**)**

d. *Tree* (Pohon)

Topologi ini merupakan perpaduan antara topologi *linear bus* dan *star* yang mana terdiri dari kelompok-kelompok dari *workstation* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *linear bus.* Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada dan memungkinkan untuk mengkonfigurasi jaringan sesuai dengan kebutuhan.



Gambar 2.5 *Topologi Tree*
(Sumber: Jaringan Komputer**)**

Kelebihan *topologi Tree* (pohon) :

1. Proses konfigurasi jaringan dilakukan dari titik ke titik pada masing-masing segmen.
2. Didukung oleh banyak perangkat keras dan perangkat lunak.

Kekurangan dari *topologi Tree (*pohon) :

1. Keseluruhan panjang kabel pada tiap-tiap segmen dibatasi pada tipe kabel yang digunakan.
2. Jika jaringan utama (*backbone*) rusak maka keseluruhan segmen ikut rusak juga.
3. Sangat relative sulit untuk di konfigurasi dan proses pengabelannya dibandingkan dengan topologi yang lain.

2.5.4 *Client Server*

Istilah *client, server* dan *client/server* dapat digunakan untuk merunjuk kepada konsep yang sangat umum atau hal spesifik dari perangkat keras atau perangkat lunak. Pada level yang sangat umum, sebuah *client* adalah setiap komponen dari sebuah sistem yang meminta layanan atau sumber daya *(resources)*

dari komponen sistem lainnya. Sedangkan sebuah *server* adalah setiap komponen sistem yang menyediakan layanan atau sumber daya ke komponen sistem lainnya.

Server adalah computer yang menyediakan fasilitas bagi komputer-komputer lain didalam jaringan dan *client* adalah komputer-komputer yang menerima atau menggunakan fasilitas yang disediakan oleh *server*. *Server* dijaringan tipe *client server* disebut dengan *Dedicated Server* karena murni berperan sebagai *server* yang menyediakan fasilitas kepada *workstation* dan *server* tersebut tidak dapat berperan sebagai *workstation*. Sistem *client-server* biasanya berjalan pada setidaknya dua sistem yang berbeda. Satu komputer bertindak sebagai *client* dan lainnya sebagai *server*, tetapi *client* dan *server* juga bisa berada pada satu sistem komputer. Biasanya sebuah *server* melayani beberapa komputer *client* walaupun mungkin juga hanya melayani satu *client*.

Fungsi *client/ server* biasanya dilakukan oleh file *server*, kecuali apabila dibutuhkan kinerja yang maksimal maka digunakanlah *server* yang khusus. *Client* biasanya berupa komputer dekstop yang terhubung dalam jaringan. Apabila pemakai ingin mengambil atau menyimpan informasi bagian aplikasi *client* akan mengeluarkan permintaan yang akan dikirim ke *server*, *server* kemudian menjalankan permintaan dan mengirimkan informasi kepada *client*. **[7, p. 30]**

2.5.4.1 Keunggulan

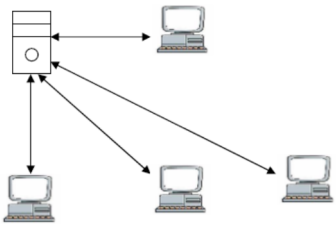
Adapun keunggulan yang dimiliki oleh *client server* adalah sebagai berikut :

a. Kecepatan akses lebih tinggi
b. Sistem keamanan dan administrasi jaringan lebih baik
c. Sistem *back-up* data lebih baik
d. Terpusat (sumber daya dan keamanan data dikontrol melalui *server*)
e. Skalabilitas
f. Fleksibel
g. Teknologi baru dengan mudah terintegrasi ke dalam sistem
h. Keseluruhan komponen (*client/network/server*) dapat bekerja bersama

2.5.4.2 Kelemahan

Adapun kelemahan yang dimiliki oleh *client server* adalah sebagai berikut :

a. Biaya operasional relatif lebih mahal.
b. Diperlukan adanya satu komputer khusus yang berkemampuan lebih untuk ditugaskan sebagai *server*.
c. Kelangsungan jaringan sangat tergantung pada *server*. Bila *server* mengalami gangguan maka secara keseluruhan jaringan akan terganggu.
d. Perbaikan (jaringan besar membutuhkan seorang *staff* untuk mengatur agar sistem berjalan secara efisien).



Gambar 2.6 Model Hubungan Client Server
(Sumber : Jaringan Komputer**)**

2.6 Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan adalah *Sublime Text* dengan menggunakan bahasa pemograman PHP dan mengunakan *MySql* untuk mengelola *database*, dan *Code Igniter Framework* sebagai pengelola fungsi dan class.

2.6.1 Sekilas Tentang PHP

PHP merupakan (akronim dari *PHP: Hypertext Preprocessor*), merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Secara khusus PHP dirancang untuk membentuk aplikasi *web* dinamis. Skrip PHP berkedudukan sebagai *tag* dalam bahasa HTML. HTML (*HyperText Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan *database*, file dan folder, sehingga membuat PHP bisa menampilkan konten yang dinamis dari sebuah website. PHP termasuk bahasa yang *cross-platform*, ini artinya PHP bisa berjalan pada sistem operasi yang berbeda-beda (*Windows, Linux,*

ataupun *Mac*). Program PHP ditulis dalam file *plain text* (teks biasa) dan mempunyai akhiran ".php".

2.6.2 Sekilas Tentang *MySql*

Database Mysql merupakan sistem manajemen basis data *SQL* yang sangat terkenal dan bersifat open source. *MySQL* dibangun, didistribusikan, dan didukung oleh *MySQL AB*. *MySQL AB* merupakan perusahaan komersial yang dibiayai oleh pengembang *MySQL*.sebenarnya software *MySQL* mempunyai dua macam lisensi. Lisensi pertama bersifat Open Source dengan menggunakan GNU dan lisensi kedua berupa lisensi komersial standar (*standard commercial licence*) yang dapat anda beli dari *MySQL AB*. *MySQL* dapat didefinisikan sebagai sebuah sistem manajemen database. Database merupakan struktur penyimpanan data.untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah database komputer, diperlukan sistem manajemen database seperti *MySQL Server*.

MySQL merupakan sistem manajemen database atau basis data terhubung. database terhubung menyimpan data pada tabel-tabel terpisah.hal tersebut akan menambah kecepatan dan fleksibilitasnya. Kata *SQL* pada *MySQL* merupakan singkatan dari "Structured Query Language". *SQL* merupakan bahasa standar yang digunakan untuk mengakses database dan ditetapkan oleh *ANSI/ISO SQL Standard*. *MySQL* merupakan software Open Source. Open source berarti semua orang diizinkan menggunakan dan memodifikasi software.

2.6.3 Sekilas Tentang *Code Igniter, Frame Work dan Sublime Text*

a. Framework

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang *programmer*, tanpa harus membuat fungsi atau *class* dari awal. Ada beberapa alasan mengapa menggunakan Framework:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi *web.*
2. *Relatif* memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah *framework* .
3. Umumnya *framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya *validasi, pagination, multiple database,* pengaturan *session, error handling,* dll
4. Lebih bebas dalam pengembangan jika dibandingkan CMS.

b. CodeIgniter

CodeIgniter merupakan salah satu dari sekian banyak *framework* PHP yang ada. *CodeIgniter* dikembangkan oleh Rick Ellis. Tujuan dari pembuatan framework CodeIgniter ini menurut user manualnya adalah untuk menghasilkan *framework* yang akan dapat digunakan untuk pengembangan proyek pembuatan *website* secara lebih cepat dibandingkan dengan pembuatan *website* dengan cara koding secara manual, dengan menyediakan banyak sekali pustaka yang dibutuhkan dalam pembuatan *website*, dengan antarmuka yang sederhana dan struktur logika untuk mengakses pustaka yang dibutuhkan. *CodeIgniter* membiarkan kita untuk

memfokuskan diri pada pembuatan *website* dengan meminimalkan pembuatan kode untuk berbagai tujuan pembuatan *website.*

c. Sublime Text

Sublime Text adalah *cross-platform* editor teks dan kode sumber, dengan antarmuka pemrograman aplikasi *Python* (API). *Sublime Text* adalah perangkat lunak *proprietary*. Teks Sublime tidak memiliki dialog pengaturan grafis dan sepenuhnya dikonfigurasi dengan mengedit file teks. *Sublime Text* mendukung banyak bahasa pemrograman dan juga *Sublime Text* bisa di jalankan di *Windows, Mac Os* dan juga *Linux.*