

BAB II

TEORI PENUNJANG

2.1 Konsep Dasar Sistem

2.1.1 Pengertian Sistem

Secara umum sistem adalah himpunan objek dengan sekumpulan proses yang saling berinteraksi dengan batasan yang jelas dan batasan ditentukan oleh perancang sistem berdasarkan gambaran detail dari pengguna (*user*).

Beberapa pengertian tentang sistem:

Menurut Jerry Fitz Gerald "Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

Prosedur adalah suatu urutan-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 yang terjadi.

Menurut Jogiyanto, "Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu".

Sistem adalah komponen dari berbagai objek yang berada di dunia ini yang dapat diamati dan memiliki ciri-ciri tingkah laku tertentu, fungsi tertentu serta tujuan tertentu (Wieringa, R.J. 1996).

Suatu sistem mempunyai maksud tertentu. Sebagian perancang sistem menyebutkan maksud dari suatu sistem adalah untuk mencapai suatu tujuan (*goal*) dan sebagian lagi menyebutkan untuk mencapai suatu sasaran (*objectives*).

2.1.2 Karakteristik Sistem

Suatu sistem selalu memiliki karakteristik tertentu, karakteristik tersebut yang membedakan suatu sistem dengan sistem lainnya dan membedakan suatu sistem dengan subsistemnya, karakteristik dari sistem diantaranya :

a. Komponen sistem (*subsistem*)

Komponen-komponen yang saling berhubungan satu dengan lainnya bersatu membentuk suatu sistem. Setiap sistem hampir bisa dipastikan selalu memiliki komponen-komponen atau subsistem-subsistem.

b. Batas sistem (*boundary*)

Batas sistem merupakan lingkup sistem yang menentukan apakah suatu komponen termasuk ke dalam sistem atau di luar sistem.

c. Lingkungan luar sistem (*environment*)

Merupakan komponen yang proses kegiatannya diluar batasan sistem.

d. Penghubung sistem (*interface*)

Penghubung sistem menyatukan semua proses pada masing-masing komponen sistem menjadi satu kesatuan proses sistem.

e. Masukan sistem (*input*)

Segala sesuatu yang masuk ke dalam sistem, baik yang bertujuan untuk proses sistem yang menghasilkan keluaran ataupun untuk operasi sistem semata.

f. Keluaran sistem (*output*)

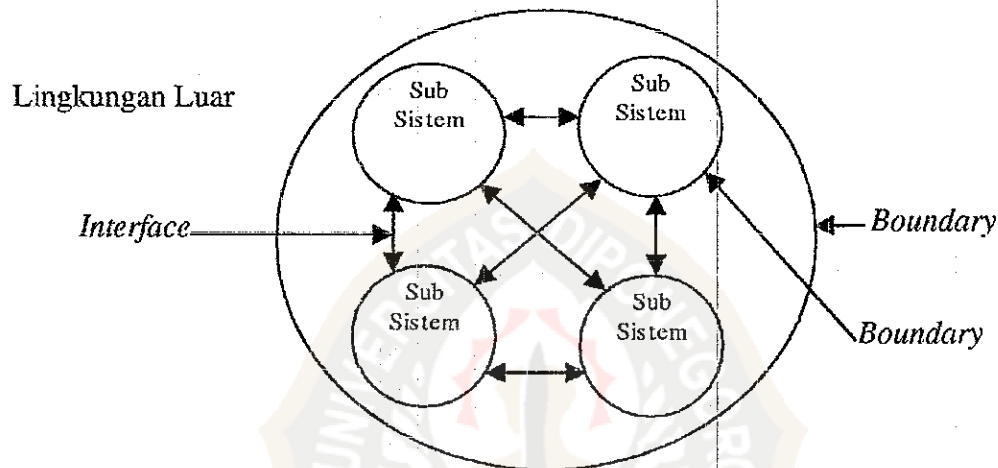
Sesuatu yang dihasilkan dari masukan sistem melalui suatu proses sistem.

g. Pengolah sistem

Pengolah sistem memproses masukan sistem menjadi keluaran sistem.

h. Sasaran sistem

Sistem tanpa tujuan atau sasaran adalah sistem yang tidak berguna dan gagal. Sistem akan menjadi suatu sistem yang berguna dan berhasil apabila sistem tersebut mencapai tujuan atau sasarannya.



Gambar 1. Ilustrasi karakteristik suatu sistem

2.2 Tinjauan Umum Pengembangan Sistem

Pengembangan sistem merupakan suatu istilah yang umum, di lapangan banyak dikenal dengan istilah lain seperti Perancangan Sistem, *System Development Life Cycle (SDLC)*, *software engineering* ataupun *Requirements Engineering*.

2.2.1 Pendekatan Pengembangan Sistem

Berdasarkan metodologi yang digunakan, pendekatan pengembangan sistem dibagi menjadi dua pendekatan yang saling berlawanan yaitu pendekatan klasik dan pendekatan terstruktur.

Pendekatan klasik dikenal juga dengan nama pendekatan tradisional atau pendekatan konvensional adalah pendekatan yang dalam pengembangan sistemnya tidak menggunakan alat dan teknik yang memadai, sedangkan sistem informasi yang ada sekarang ini semakin kompleks dan terus bertambah kompleks sehingga pendekatan klasik akan menimbulkan beberapa permasalahan, diantaranya :

- a. Perancangan perangkat lunak akan menjadi sulit, karena tanpa alat dan teknik yang memadai maka perancangan perangkat lunak tidak terarah dan sulit untuk dikerjakan oleh pemrogram, berbeda dengan pendekatan terstruktur yang memiliki alat-alat seperti diagram arus data, kamus data, bagan terstruktur, dan alat dokumentasi lainnya.
- b. Biaya perawatan atau pemeliharaan sistem akan menjadi mahal, hal ini karena dokumentasi tidak lengkap dan tidak terstruktur.
- c. Kemungkinan kesalahan sistem besar, karena pengembangan sistem dilakukan tanpa modul-modul terstruktur sehingga penyetoran modul tidak dapat dilakukan secara terpisah yang berakibat kepada kesulitan dalam pencarian kesalahan sistem.
- d. Keberhasilan sistem kurang terjamin, karena sistem dirancang tidak berorientasi kepada pemakai melainkan dengan asumsi bahwa analis sistem telah mengerti kebutuhan pemakai.
- e. Masalah dalam penerapan sistem, karena kurang melibatkan pemakai dalam perancangannya maka pemakai akan butuh waktu penyesuaian pada saat implementasi sistem.

Melalui pendekatan terstruktur maka permasalahan-permasalahan pada pendekatan klasik dapat dipecahkan dan hasil dari sistem akan mudah untuk dipelihara dan dikembangkan lebih lanjut, fleksibel, lebih memuaskan pemakainya, mempunyai dokumentasi yang baik, tepat pada target waktu penyelesaian proyek, sesuai dengan anggaran biaya pengembangan, dapat meningkatkan produktivitas dan kualitas yang lebih baik lagi.

2.2.2 Metodologi Pengembangan Sistem

Metodologi adalah kesatuan metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, aturan-aturan dan postulat-postulat yang digunakan oleh suatu ilmu pengetahuan, seni atau disiplin lainnya. Sedang metode adalah suatu cara, teknik yang sistematis untuk mengerjakan sesuatu.

Metodologi pengembangan sistem berarti urutan atau langkah-langkah atau tahapan-tahapan kerja yang dilaksanakan dalam mengembangkan sistem, maka dengan menggunakan metodologi dalam pengembangan suatu sistem akan dihasilkan pengembangan sistem yang terencana dan tepat waktu.

Menurut *A.Ziyas Aktas* metodologi dapat diklasifikasikan dalam tiga kelompok metodologi pengembangan sistem, yaitu

1. Functional decomposition methodologies

Metodologi ini menggunakan teknik pemecahan sistem ke dalam subsistem-subsistem yang lebih kecil sehingga lebih mudah untuk dipahami, dirancang, dan diterapkan. Beberapa metodologi kelompok ini adalah

- HIPO (*Hierarchy plus Input-Process-Output*).
- *Stepwise Refinement (SR)* atau *Iterative Stepwise Refinement (ISR)*.

2. *Data-oriented methodologies*

Metodologi dititik beratkan pada karakteristik data yang akan diproses, metodologi ini dikelompokkan ke dalam dua kelas yaitu:

a. *Data-flow oriented methodologies*

Metodologi ini membagi sistem ke dalam modul-modul berdasarkan tipe elemen data dan tingkah-laku logika modul di dalam sistem, sehingga semua gambaran sistem dan hubungan antar subsistem terdefinisi dalam modul-modul.

Beberapa metodologi kelas ini adalah:

- *Composite Design*.
- *Structured System Analysis and Design (SSAD)* atau lebih dikenal sebagai Analisis dan Disain Sistem Terstruktur.

b. *Data-structure oriented methodologies*.

Metodologi ini menekankan pada struktur input dan output yang ada di dalam sistem. Struktur ini digunakan sebagai dasar struktur dari sistemnya, dan hubungan fungsi antar modul dijelaskan dari struktur sistemnya. Beberapa metodologi yang termasuk dalam kelas ini adalah:

- *JSD (Jakson's Systems Development)*.
- *W/O (Wanier/Orr)*.

3. *Prescriptive methodologies*

Beberapa metodologi yang termasuk dalam kelompok ini adalah:

- *ISDOS (Information System Design and Optimization System)*.

Perangkat lunak ini dikembangkan di University of Michigan, berguna untuk otomatisasi pengembangan sistem.

- *SDM/70 (System Development Methodology/70)*

Dikembangkan dan dipasarkan oleh Atlantic Software, Inc merupakan salah satu perusahaan Amerika Serikat. Software ini berisi kumpulan metode, estimasi, dokumentasi dan petunjuk untuk pengembangan dan perawatan sistem.

- *SRES (Software Requirement Engineering System)*

Dikembangkan untuk SDS (*Software Development System*) angkatan udara Amerika Serikat (U.S Air Force). SRES memiliki konsep yang hampir sama dengan ISDOS.

Metodologi yang digunakan pada skripsi ini adalah metodologi analisis dan disain sistem terstruktur, merupakan metodologi kelompok *Data-oriented methodologies* untuk kelas *Data-flow oriented methodologies*, dimana metodologi ini menangani analisa sistem dan disain sistem dengan pendekatan terstruktur.

Tahapan utama dalam setiap metodologi pengembangan sistem adalah analisa sistem, disain sistem dan implementasi sistem, beberapa pengembang konsep sistem menambahkan tahapan-tahapan untuk mengenali sistem sebelum mulai melakukan analisa sistem dan memahaminya lebih dalam.

2.3 Tahapan Dalam Pengembangan Sistem

Sejak tahun 1970 telah banyak dirumuskan tahapan-tahapan dalam pengembangan sistem oleh para pengembang sistem informasi, metodologi standar yang digunakan oleh perusahaan-perusahaan di Indonesia saat ini adalah metodologi pengembangan sistem menurut *Raymond McLeod, Jr* dengan tahapan-tahapan yang dilakukan yaitu:

2.3.1 Perencanaan Sistem

Pada tahapan perencanaan, identifikasi terhadap kebutuhan sistem mulai dilakukan dengan mengamati alur proses yang terjadi pada sistem sebelumnya. Tahapan perencanaan menggambarkan lingkup proyek atau sistem yang akan ditangani. Adapun lingkup sistem yang ditangani dijabarkan dalam bentuk DFD konteks, atau disebut juga diagram konteks. DFD (*Data Flow Diagram*) sering diterjemahkan menjadi diagram arus data (DAD). DAD merupakan alat yang biasa dipakai untuk mendokumentasikan proses dalam sistem. DAD menekankan pada fungsi-fungsi di dalam sistem, cara menggunakan informasi yang tersimpan dan pemindahan informasi antar fungsi di dalam sistem. DAD konteks adalah DAD yang memperlihatkan sistem sebagai sebuah proses. Tujuannya adalah memberikan pandangan umum sistem. DAD konteks memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya. Ada pihak luar atau lingkungan yang memberi masukan dan ada pihak yang menerima keluaran sistem. Dalam hal ini pihak luar dapat berupa sistem lain, suatu perangkat keras, orang atau organisasi.

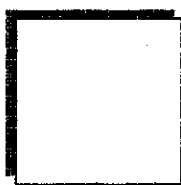
2.3.2 Analisa Sistem

Pada tahapan analisis sistem, analis sistem (orang yang bertanggung jawab terhadap pengembangan sistem secara menyeluruh) sering berdialog dengan pengguna untuk memperoleh informasi detail kebutuhan pengguna. Pengumpulan kebutuhan pengguna biasa dilakukan melalui wawancara atau observasi. Hasil yang didapatkan dipakai sebagai bahan untuk menyusun DAD pengembangan sistem. Untuk memperinci DAD, item-item yang terdapat pada

arus data dan penyimpanan data dijabarkan dalam bentuk kamus data. Kamus data adalah deskripsi formal mengenai seluruh elemen yang tercakup dalam DAD. Pada tahapan desain sistem, elemen-elemen dalam kamus data akan menjadi bahan untuk menyusun basis data.

Menurut Kendall dalam bukunya "*System Analysis and Design*", Kelebihan dari penggunaan diagram arus data (DAD) adalah kebebasan dari keterikatan untuk menentukan spesifikasi teknis sistem terlalu awal, kemudahan dalam pemahaman keterkaitan antara sistem dan subsistem, dan kemudahan dalam mengkomunikasikan pengetahuan tentang sistem yang dirancang kepada pengguna.

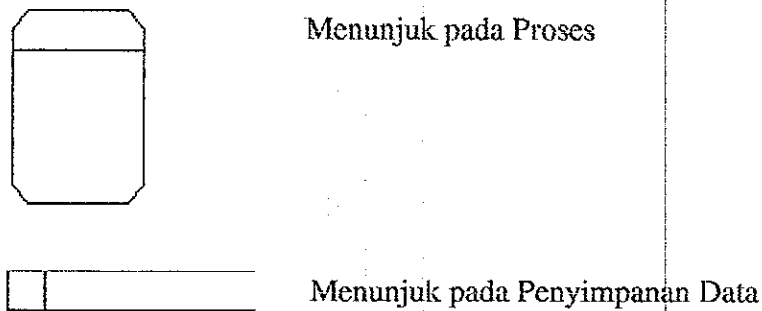
Terdapat empat simbol dasar yang digunakan untuk menggambarkan diagram arus data (DAD), yaitu segiempat rangkap, tanda panah, segiempat dengan ujung bulat, dan segiempat dengan sisi terbuka sebelah (sisi kanan), masing-masing simbol menunjuk pada setiap komponen dari sistem secara keseluruhan, sehingga dengan kombinasi penggambaran dari keempat simbol tersebut seluruh sistem dan subsistemnya dapat dipelajari dan dipahami dengan mudah. Berikut adalah simbol DAD berdasarkan konsep C.Gane dan T. Sarson (Kendall,1988).



Menunjuk pada Entitas



Menunjuk pada Arus Data



Gambar 2. Simbol yang digunakan dalam penggambaran DAD

2.3.3 Disain Sistem

Tahapan disain sistem dibagi menjadi dua tahapan, yaitu:

1. Disain basis data
2. Disain proses

Disain basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Penyusunan basis data berlandaskan pada kamus data yang telah dibuat pada tahapan analisis sistem. Teknik yang digunakan dalam disain basis data ada dua cara yaitu Teknik Normalisasi dan Teknik Entity Relationship (Harianto K, 1996).

Disain proses biasanya menghasilkan dokumentasi pengembangan sistem dalam bentuk spesifikasi program dan bagan struktur sistem. Spesifikasi program dipakai sebagai petunjuk bagi pemrogram agar dengan mudah dapat menuangkan proses ke dalam program. Bagan struktur sistem memperlihatkan seluruh program dalam sistem yang dirancang dan hirarki kontrol terhadap program tersebut. Sederhananya perancangan proses menghasilkan bagan struktur sistem yang mengacu kepada menu tampilan.

2.3.4 Implementasi Sistem

Tahapan implementasi sistem mencakup pengkodean program dan pengujian program. Setelah tahapan implementasi sukses maka akan sampai pada tahap penggunaan. Dalam hal ini aplikasi mulai dioperasikan oleh pengguna untuk melakukan berbagai transaksi.

2.4 Sistem Terintegrasi

Sistem informasi harus didisain terpadu diantara unit-unit di dalam sistem, harus dapat berhubungan dan berkomunikasi dengan baik antara satu komponen dengan komponen lainnya. Integrasi adalah suatu kebutuhan yang mendesak karena tujuan dari sistem informasi sendiri adalah melayani informasi multilevel, cross-functional, tepat waktu, akurat dan relevan kepada semua komponen sistem.

Basis data dan teknologi jaringan komputer yang digabungkan oleh teknologi perangkat lunak merupakan kunci untuk mencapai sistem yang terintegrasi.

2.4.1 Jaringan Komputer

Sebuah jaringan minimal terbentuk dari dua komputer yang terhubung sehingga masing-masing dapat membagi sumber dayanya (*share resources*). Adapun pada kenyataannya jaringan (*Networks*) lebih kompleks dari pada skenario koneksi dua komputer, semua jaringan berdasar pada konsep pembagian (*sharing*). Jika sebuah komputer terhubung dan berkomunikasi dengan komputer lainnya tentu sangat tergantung pada teknologi, untuk itu terdapat banyak tipe koneksi fisik dan perangkat lunak terkait.

Tipe jaringan (*Networks*) berdasarkan daerah jangkauan diantaranya:

- **Local Area Networks (LAN):** LAN adalah kumpulan komputer yang terhubung dengan jaringan pada satu daerah sempit (10 m – 1 km), seperti pada sebuah gedung perkantoran.
- **Metropolitan Area Networks (MAN):** MAN menggunakan teknologi WANs untuk menghubungkan (*interconnect*) LAN pada sebuah daerah geografik tertentu (1 km – 10 km), seperti sebuah kota.
- **Wide Area Networks (WAN):** WAN mencakup koneksi jaringan lintas dunia (10 km – 1000 km), untuk mengirim komunikasi antar jaringan.

Tipe jaringan berdasarkan teknologi komputer yang terhubung, yaitu:

a. **Jaringan *peer-to-peer***

Komputer pada sebuah jaringan *peer-to-peer* dapat berfungsi sebagai sebuah client maupun sebagai sebuah server, jaringan *peer-to-peer* tidak memiliki kontrol terpusat pada sumber daya yang terbagi (*shared resources*). Semua peralatan dapat berbagi sumber dayanya dengan semua komputer pada jaringan yang sama.

Beberapa keuntungan jaringan *peer-to-peer*, yaitu:

- Masing-masing komputer tidak tergantung pada sebuah server khusus.
- Pengguna dapat mengatur pembagian sumber dayanya sendiri.
- Biaya investasi dan operasional jaringan *peer-to-peer* tidak mahal.
- Anda tidak membutuhkan peralatan atau software lainnya selain sebuah sistem operasi, kartu jaringan, dan kabel.
- Cocok untuk jaringan dengan jumlah pengguna 10 atau kurang.

Beberapa kerugian jaringan *peer-to-peer*, yaitu:

- Pada waktu yang sama anda hanya dapat menerapkan atau merubah keamanan jaringan pada sebuah sumber daya saja (penerapan atau perubahan keamanan sumber daya yang ada pada jaringan dilakukan satu per satu).
- Kunci (*password*) yang harus diingat oleh pengguna bertambah banyak seiring dengan bertambahnya sumber daya yang di bagi (*shared resources*).
- Anda harus melakukan penyelamatan data (*backup*) pada setiap komputer untuk melindungi data yang terbagi (*shared data*).
- Pada saat seseorang meng-akses sumber daya yang terbagi, komputer dimana sumber daya tersebut berada mengalami penurunan unjuk kerja (*performance*).
- Tidak adanya pola organisasi yang terpusat untuk meng-alokasi atau mengatur hak akses kepada data.

b. Jaringan *Client/Server*

Client/Server biasa di deskripsikan sebagai "*network servers*". Sebagai definisi sederhana sebuah server adalah sebuah mesin (komputer) yang hanya berfungsi untuk melayani atau menjawab (dengan kata lain *me-respond*) permintaan client.

Beberapa keuntungan jaringan *Client/Server*, yaitu:

- Jaringan berbasis-server memberikan pemusatan *user accounts*, keamanan (*security*) dan pengaturan akses (*access control*) yang memudahkan administrator jaringan.

- Peralatan yang lebih berdayaguna berarti lebih efisien-nya proses akses ke sumber daya jaringan dengan baik.
- Pengguna (*users*) hanya perlu mengingat satu password untuk login jaringan, yang akan memberikan mereka hak akses ke seluruh sumber daya yang diijinkan untuk di akses.

Beberapa kerugian jaringan *Client/Server*, yaitu:

- Kegagalan sebuah server dapat mengakibatkan tidak berfungsinya jaringan; sehingga mengakibatkan hilangnya sumber daya jaringan.
- Seluruh jaringan membutuhkan staff yang berpengalaman untuk mengatur perangkat lunak server yang khusus yang kompleks, sehingga menambah biaya secara keseluruhan.

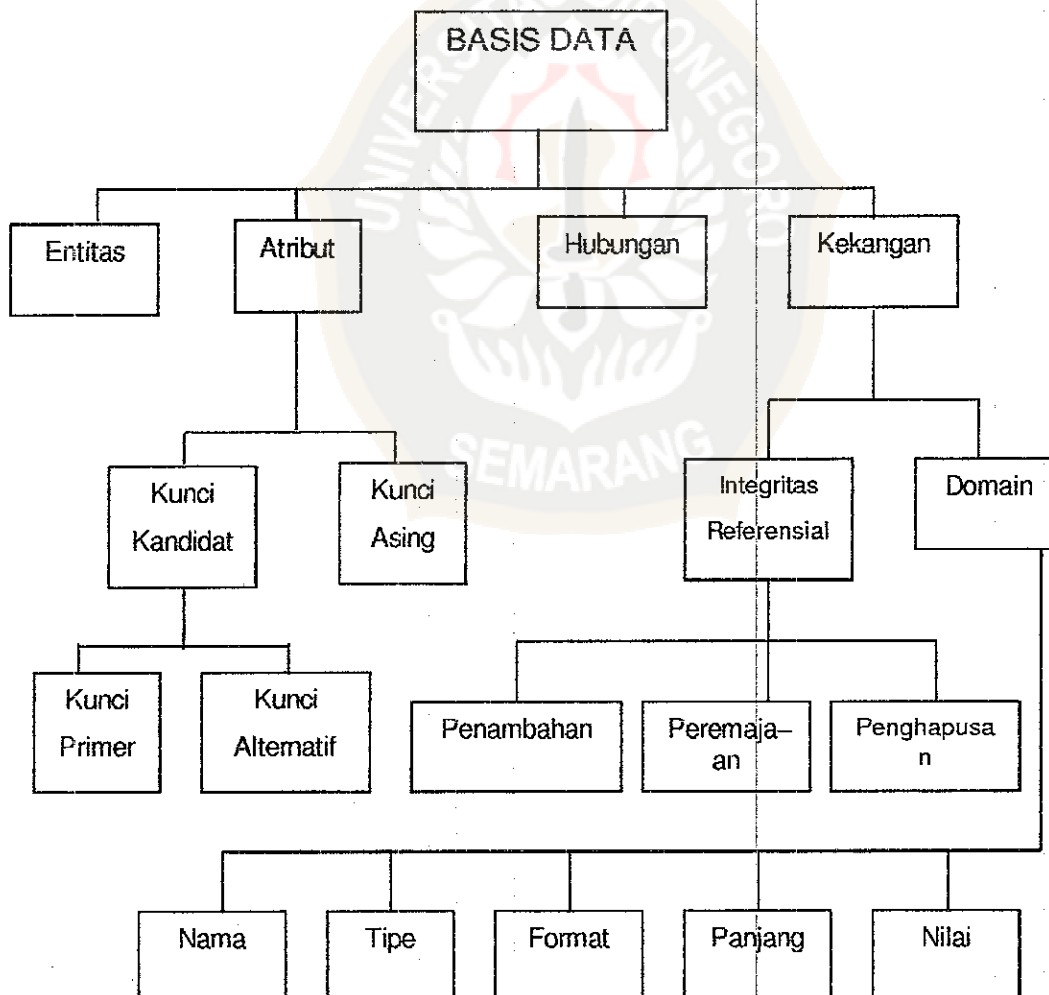
2.4.2 Basis Data

Menurut C.J. Date "Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara informasi dan menjamin informasi tersebut tersedia saat dibutuhkan" (Abdul Kadir, 2001).

Disain dari basis data harus menyimpan semua data yang ada dalam suatu simpanan yang tunggal untuk keperluan semua orang atau unit kerja yang mempunyai hak untuk mengaksesnya. Dengan teknologi jaringan komputer maka jumlah data yang besar yang berasal dari lokasi lokal atau lokasi jarak jauh dapat ditangkap, dimanipulasi dan ditransmisikan dengan cepat. Pengaksesan basis data yaitu memasukkan, mengubah, menghapus, memanipulasi dan memperoleh data biasanya ditangani oleh suatu *Database Management System* (DBMS).

DBMS adalah suatu perangkat lunak sistem yang memberi kemudahan kepada pengguna untuk mendefinisikan, menciptakan, dan memelihara basis data dengan pengaturan akses tertentu terhadap data tertentu tergantung kepada hak aksesnya (Mata-Toledo, R.A., Cushman, P.K., 2000).

Beberapa komponen basis data secara bertingkat adalah entitas, atribut, hubungan, kekangan, kunci-kunci entitas, integritas referensial, dan domain yang mendefinisikan karakteristik setiap data, untuk lebih jelasnya tingkatan komponen basis data dapat diilustrasikan sebagai berikut,



Gambar 3. Komponen basis data dan tingkatannya

Pengertian mengenai beberapa komponen di atas adalah sebagai berikut:

- **Entitas**

Entitas terkadang disebut tipe entitas atau kelas entitas. Entitas menyatakan objek atau kejadian. Karyawan merupakan contoh entitas pada suatu sistem perusahaan. Pada model relasional, entitas akan menjadi tabel.

- **Atribut**

Atribut adalah item data yang menjadi bagian dari suatu entitas. Istilah lain dari atribut adalah properti. Nomor Induk Karyawan ataupun Nama Karyawan adalah contoh atribut yang terdapat pada entitas Karyawan.

- **Hubungan**

Hubungan adalah asosiasi atau kaitan antara dua entitas.

- **Kekangan**

Kekangan digunakan untuk melindungi integritas data, misalnya melindungi kesalahan sewaktu pengisian data.

- **Domain**

Domain adalah himpunan nilai yang berlaku bagi suatu atribut. Kekangan domain mendefinisikan nama, tipe, format, panjang, dan nilai masing-masing item data. Misal tipe data CHAR yang menyatakan tipe alfanumerik atau karakter dan NUMERIC menyatakan tipe bilangan.

- **Integritas referensial**

Integritas referensial adalah aturan-aturan yang mengatur hubungan antara kunci primer dengan kunci lainnya milik tabel-tabel yang berada dalam suatu basis data relasional untuk menjaga konsistensi data.

Untuk bisa menangani teknologi jaringan komputer dan basis data maka dipilih perangkat lunak PHP dan MySQL yang merupakan *free-software* tetapi kemampuannya terus dikembangkan dan dites oleh semua pengguna di seluruh dunia sehingga setiap kekurangan program terus diperbaiki.

2.5 PHP

PHP singkatan berulang dari *PHP: Hypertext Preprocessor* merupakan bahasa berbentuk skrip yang ditempatkan pada *server* dan diproses di *server*. Hasilnya yang dikirimkan ke *client*, tempat pemakai menggunakan *browser*. PHP diturunkan dari bahasa *C* sehingga memiliki sintaks yang mirip dengan bahasa *C*, PHP dapat berkomunikasi penuh dengan database dan dapat disisipkan pada bahasa HTML (*Hyper Text Mark Up Language*) sehingga untuk antar muka (*interface*) program aplikasi dapat dibuat semudah (*user friendly*) dan seindah mungkin.

Penulisan skrip PHP diawali tanda "<?" dan diakhiri dengan tanda ">", misal:

```
<? printf("Tanggal Sekarang adalah: %s ", Date("d F Y")); ?>
```

PHP menangani Variabel dan Konstanta, pernyataan kontrol, array, fungsi dan operasi-operasi untuk pengolahan data, PHP juga merupakan pemrograman berorientasi objek. Berikut gambaran umum dari kemampuan skrip PHP.

2.5.1 Variabel dan Konstanta

Variabel mendefinisikan nilai yang tidak tetap dan berubah sesuai dengan kondisi yang diinginkan, variabel pada PHP merupakan pengenal yang diawali dengan tanda "\$", misal : `$Nama=" andri "` ;

Selain variabel tunggal, PHP juga menangani variabel pada variabel, sehingga variabel dapat ditentukan dan digunakan sesuai dengan kondisi yang diharapkan, pernyataan ini melibatkan dua tanda "\$",

misal : `$$Nama= "Namaku";`

maka pernyataan diatas dapat didefinisikan menjadi variabel \$andri yang isinya Namaku. Konstanta didefinisikan dengan penulisan :

```
define ("nama_konstanta","nilai konstanta");
```

2.5.2 Pernyataan Kontrol

2.5.2.1 Pernyataan *if*

Bentuk *if* pada PHP dibagi menjadi tiga yaitu *if* saja, *if-else* dan *if-elseif*, penulisannya yaitu :

- untuk *if*

```
if (ekspresi)
  pernyataan
```

- untuk *if-else*

```
if (ekspresi)
  { pernyataan_1 }
else { pernyataan_2 }
```

- untuk *if-elseif*

```
if (ekspresi)
  { pernyataan _1 }
elseif (kespresi)
  { pernyataan _2 }
elseif (ekspresi )
  { pernyataan _3 }
  M
else
  { pernyataan _n }
```

2.5.2.2 Pernyataan switch

Fungsi dari switch ini mirip dengan pernyataan if-elseif, untuk pengambilan keputusan dengan banyak pilihan kondisi, tetapi beda dalam penulisan skrip, yaitu

```
switch (ekspresi)
{
  case ekspresi_case_1 :
    pernyataan_1 ;
    break ;
  case ekspresi_case_2 :
    pernyataan_2 ;
    break ;
  ...
  default :
    pernyataan_default ;
}
```

2.5.2.3 Pernyataan while

Menangani pengulangan perintah sampai kondisi tertentu, pengulangan akan terus berjalan sampai ditemukan kondisi salah (*False*). Penulisannya adalah

```
while (kondisi)
  { pernyataan_pernyataan }
```

2.5.2.4 Pernyataan do-while

Pernyataan ini memiliki fungsi yang mirip dengan pernyataan while, perbedaannya pada while kondisi diperiksa di awal proses pengulangan sedangkan pada pernyataan do-while kondisi diperiksa pada akhir proses, sehingga pada pernyataan do-while minimal satu kondisi terpenuhi. Penulisan skripnya adalah

```
do
{
  pernyataan-pernyataan
} while (kondisi) ;
```

2.5.2.5 Pernyataan for

Perintah for menangani proses pengulangan dengan banyak pengulangan sudah diketahui jumlahnya, pada pernyataan for kondisi tidak didefinisikan karena nilai awal dan nilai akhir variabel penghitung yang didefinisikan pada pernyataan for merupakan jumlah pengulangan yang menyatakan kondisi.

Penulisannya adalah

```
For (nilai_awal, nilai_akhir, peningkatan/penurunan)
{
    pernyataan_pernyataan
}
```

Peningkatan dapat didefinisikan dengan memberi tanda "++" untuk peningkatan dan tanda "--" untuk penurunan pada akhir variabel, misal \$var++ mengindikasikan peningkatan nilai variabel \$var.

2.5.3 Array

Array merupakan sebuah tempat dimana sejumlah nilai dapat ditampung, berbeda dengan variabel yang hanya dapat menampung sebuah nilai. Array menyimpan data secara berurutan, urutan data ini ditunjukkan oleh suatu indeks. Array memiliki dimensi, baik dalam bentuk dimensi satu yang mewakili bentuk suatu vektor, dimensi dua yang mewakili bentuk dari matrik atau tabel, ataupun dimensi tiga yang mewakili bentuk ruang. Array yang memiliki bentuk dimensi lebih dari satu disebut array multidimensi. Contoh deklarasi array pada PHP,

- Untuk array dimensi satu

```
$kota [0] = "Semarang" ;
```

```
$kota [1] = "Purwakarta" ;
```

```
$kota [2] = "Bandung" ;
```


- Untuk array multidimensi

```
$propinsi = array(
    "Jabar"=> array(
        "Cirebon",
        "Ckampek",
        "Purwakarta",
        "Bandung"),
    "Jateng"=>array(
        "Tegal",
        "Kendal",
        "Semarang",
        "Solo")
    );
```

2.5.4 Fungsi

Suatu program biasanya menggunakan satu blok perintah yang digunakan berkali-kali, blok perintah ini agar tidak ditulis berkali-kali dapat digunakan subrutin, subrutin yaitu blok perintah yang ditandai dengan label nama, dan dapat dipanggil berkali-kali. Subrutin ada dua macam yaitu prosedur dan fungsi, perbedaannya fungsi mengembalikan nilai argumen atau parameter sedangkan prosedur tidak. PHP menangani perintah untuk fungsi, dengan deklarasi yaitu

```
Function nama_fungsi (argumen)
{ blok perintah
  return ($variabel_pada_blok_perintah) }
```

Sehingga sebenarnya prosedur pun dapat ditangani oleh PHP dengan deklarasi fungsi tanpa perintah return.

2.5.5 Operasi Matematika

PHP mendukung penggunaan fungsi-fungsi untuk operasi matematika, berikut tabel nama fungsi operasi matematika, deklarasi fungsi dan kegunaannya.

Tabel 1. Deklarasi fungsi matematika dan kegunaannya pada PHP

Nama Fungsi	Penulisan	Kegunaan
Sinus	sin (bilangan)	Memperoleh nilai sinus.
Cosinus	cos (bilangan)	Memperoleh nilai cosinus.
Tangen	tan (bilangan)	Memperoleh nilai tangen.
Arccosinus	acos (bilangan)	Memperoleh nilai arc cosinus dari suatu bilangan (satuan radian).
Arcsinus	asin (bilangan)	Memperoleh nilai arc sinus dari suatu bilangan (satuan radian).
Arctangen	atan (bilangan)	Memperoleh nilai arc tangen dari suatu bilangan (satuan radian).
Arctan2	atan2 (x,y)	Memperoleh nilai arc tangen berdasarkan dua bilangan, fungsi identik dengan atan(x/y).
Rad2deg	rad2deg (bilangan)	Konversi nilai satuan radian menjadi nilai satuan derajat.
Deg2rad	rad2deg (bilangan)	Konversi nilai satuan derajat menjadi nilai satuan radian.
Pi	pi ()	Memperoleh nilai π (3,14).
Absolut	abs (bilangan)	Memperoleh nilai absolut dari suatu bilangan.
Exponen	exp (bilangan)	Memperoleh nilai exponen bilangan (e^x).
Logaritma	log (bilangan)	Memperoleh nilai logaritma alami suatu bilangan ($^e\log x$).
Logaritma10	log10 (bilangan)	Memperoleh nilai logaritma berbasis 10 suatu bilangan ($^{10}\log x$).
Power	pow (x,y)	Menghitung pangkat x^y .
Square root	sqrt (bilangan)	Menghitung akar kuadrat suatu bilangan.

2.5.6 PHP Pemrograman Berorientasi Objek

Pemrograman berorientasi objek merupakan model evolusi pemrograman yang paling terakhir dan menjadi *trend* saat ini, lebih dikenal dengan istilah OOP (*Object Oriented Programming*). Objek dalam OOP dibentuk dari kelas, kelas dapat diilustrasikan sebagai cetakan untuk objek, jadi untuk satu kelas dapat dicetak banyak objek. Suatu kelas menyimpan *properti* (variabel) dan mempunyai *metode* (fungsi) untuk mengakses properti.

Secara garis besar ada tiga langkah dalam menciptakan suatu blok program yang menggunakan kelas dan objek, yaitu mendefinisikan kelas, menciptakan objek dan mengakses properti dengan metode. Untuk lebih mudah dalam pemahaman kelas dan objek maka contoh berikut dapat dijadikan sebagai analogi,

<?

//Medefinisikan kelas

class kelas_tanggal

{

 //Properti

var \$tanggal;

var \$bulan;

var \$tahun;

 //Metode

function set_tanggal (\$tg, \$bl, \$th)

 {

\$this->tanggal = \$tg;

\$this->bulan = \$bl;

\$this->tahun = \$th;

 }

}

```
//Menciptakan objek
$tgl_lahir = new kelas_tanggal ;

//Mengakses Properti dengan Metode
$tgl_lahir->set_tanggal(6, 4, 1980);

printf(
"Saya Lahir Tanggal: $tgl_lahir->tanggal/$tgl_lahir->bulan/$tgl_lahir->
tahun");

?>
```

2.6 MySQL

MySQL merupakan salah satu jenis program database server. Database server adalah sebuah program yang bertugas melayani permintaan *query* database dari *client* (Farrar U,2002). Beberapa produk database server yang ada saat ini adalah MySQL, Microsoft SQL Server, Oracle, SyBase dan PostgreSQL. MySQL menggunakan SQL (*Structured Query Language*) sebagai bahasa dasar untuk mengakses databasenya.

Berikut tabel deklarasi fungsi yang digunakan untuk mengakses database MySQL dan kegunaannya,

Tabel 2. Deklarasi fungsi akses basis data pada MySQL

| Nama Fungsi | Penulisan | Kegunaan |
|-----------------|---|--|
| MySQL Connect | mysql_connect(host, nama_pemakai,password) | Memulai hubungan ke database MySQL yang terdapat pada suatu host (server). |
| MySQL Close | mysql_close (pengenal_hubungan) | Mengakhiri Hubungan ke database MySQL. |
| MySQL Select DB | mysql_select_db (database, pengenal_hubungan) | Memilih database yang akan dipakai. |

| | | |
|---------------------|--|--|
| MySQL Query | mysql_query (permintaan, pengenal_hubungan) | Mengeksekusi permintaan operasi terhadap tabel. |
| MySQL Num Rows | mysql_num_rows (pengenal_hasil) | Memperoleh jumlah baris dari permintaan select. |
| MySQL Affected rows | mysql_affected_rows (pengenal_hubungan) | Memperoleh jumlah baris dari insert, update dan delete. |
| MySQL Num Fields | mysql_num_fields (pengenal_hasil) | Memperoleh jumlah kolom hasil permintaan. |
| MySQL Fetch Field | mysql_fetch_field (pengenal_hasil) | Memperoleh informasi suatu kolom. |
| MySQL Fetch Row | mysql_fetch_row (pengenal_hasil) | Menghasilkan array berisi seluruh kolom dari sebuah baris dan array disimpan dengan indeks angka. |
| MySQL Fetch Array | mysql_fetch_array (pengenal_hasil) | Menghasilkan array berisi seluruh kolom dari sebuah baris dan array disimpan dengan indeks angka dan nama kolom. |
| MySQL Data Seek | mysql_data_seek (pengenal_hasil, nomor_baris) | Memindah pointer pada suatu himpunan hasil agar menunjuk ke baris tertentu. |
| MySQL Field Seek | mysql_field_seek (pengenal_hasil, nomor_kolom) | Memindah pointer pada suatu himpunan hasil agar menunjuk ke kolom tertentu. |
| MySQL Create DB | mysql_create_db(database, pengenal_hubungan) | Membuat databse MySQL. |
| MySQL Drop DB | mysql_drop_db (database, pengenal_hubungan) | Menghapus database MySQL. |
| MySQL List Field | mysql_list_field (database, nama_tabel, pengenal_hubungan) | Memperoleh daftar nama kolom pada suatu tabel dalam database MySQL. |

Untuk menjelaskan pengenalan dalam penulisan fungsi MySQL yaitu pengenalan_hubungan, permintaan dan pengenalan_hasil maka dapat didefinisikan dari contoh berikut dengan contoh nama database "mipa" dan nama tabel "matematika",

```
<?
```

```
$host="localhost";
```

```
$pemakai = "andri";
```

```
$password = "bukadonk";
```

```
$id_mysql= mysql_connect($host,$pemakai,$password);
```

```
mysql_select_db("mipa",$id_mysql)
```

```
$pilih=select * from matematika ;
```

```
$hasil=mysql_query($pilih,$id_mysql);
```

```
?>
```

Variabel \$id_mysql merupakan pengenalan_hubungan karena merupakan variabel yang menyimpan perintah untuk berhubungan dengan database, variabel \$pilih merupakan permintaan karena menyimpan perintah permintaan untuk memilih semua data pada tabel "matematika" dan variabel \$hasil merupakan pengenalan_hasil karena menyimpan hasil data yang siap untuk ditampilkan. MySQL mengenal beberapa permintaan untuk pemeliharaan data antara lain permintaan *select*, *insert*, *update* dan *delete*.

2.7 Job Order

Job order (J/O) adalah prosedur penanganan gangguan dengan format laporan pada kertas yang diserahkan oleh TMC ke Programmer/Dispatcher (P/D) untuk didistribusikan ke AWG di lokasi-lokasi kerusakan perangkat telekomunikasi dan AWG harus melaporkan setiap kemajuan dari penugasan yang diinformasikan pada Job Order (J/O) ke TMC.

TMC kependekan dari *Technology Maintenance Center* adalah unit kerja PT. Telkom Divre IV pusat yang bertugas mengontrol keadaan perangkat telekomunikasi di seluruh daerah Jateng dan DIY. Setiap kerusakan dan gangguan perangkat telekomunikasi daerah operasi Jateng dan DIY hanya dapat diketahui oleh TMC melalui display server yang diawasi setiap saat.

Programmer/Dispatcher (P/D) adalah unit kerja PT Telkom Divre IV pusat yang bertugas mendistribusikan permintaan penanganan gangguan oleh TMC ke AWG di daerah-daerah di seluruh Jawa Tengah dan Daerah Istimewa Yogyakarta.

AWG kependekan dari *Area Work Group* adalah perwakilan PT Telkom pusat di daerah-daerah yang bertanggung jawab untuk menangani secara langsung ke lapangan setiap kerusakan perangkat telekomunikasi.