BAB 2

TINJAUAN PUSTAKA

2.1 Profil Kantor Jasa Penilai Publik (KJPP) GUNAWAN

Kantor Jasa Penilai Publik (KJPP) Gunawan dengan penanggung jawab penilai bersertifikat Ir. P. Gunawan Eka Sutanta yang telah berprofesi sebagai penilai sejak 1995. Berdasarkan Keputusan Menteri Keuangan No. 406/KMK.06/2004 dan Peratutan Menteri Keuangan No. 125/PMK.01/2008 maka didirikanlah Kantor Jasa Penilai Publik yang berbadan hukum perseorangan atau persekutuan perdata yaitu Kantor Jasa Penilai Publik (KJPP) Gunawan pada tanggal 1 Desember 2008.

2.1.1 Struktur Organisasi



Gambar 2.1 Struktur Organisasi Kantor Jasa Penilai Publik GUNAWAN

2.1.2 Deskripsi Pekerjaan

Deskripsi pekerjaan dari tiap-tiap bagian pada struktur organisasi di atas adalah sebagai berikut.

1. Pimpinan Kantor Jasa Penilai Publik (KJPP) GUNAWAN, yang bertanggung jawab atas segala kegiatan yang dilakukan berkaitan penialain di Kantor Jasa Penilai Publik (KJPP) GUNAWAN.
2. Market and Development, yang bertanggung jawab atas segala pemasaran dan pengembangan dalam hal penilian publik.
3. Accounting, bertanggung jawab atas segala kegiatan penyusunan anggaran per tahunnya untuk biaya-biaya kegiatan penilaian publik per tahunnya.
4. Keuangan, yang bertanggung jawab terhadap pengelolaan keuangan.
5. Asisten Penilai I, Asisten Penilai II yang bertanggung jawab terhadap urusan teknis yaitu penilain.
6. Riset and Agency, yang bertanggung jawab terhadap penelitian teknis dalam kegiatan penilaian.

2.1.3 Logo



Gambar 2.2 Logo Kantor Jasa Penilai Publik (KJPP) GUNAWAN

Arti Logo :

1. Lingkaran bulat yang mengelilingi hurup G yang berwarna biru yaitu melambangkan bahwa para penilai/surveyor *appraisal* di lingkungan Kantor Jasa Penilai Publik (KJPP) GUNAWAN di tuntut untuk menjadi penilai/surveyor yang menjunjung tinggi sikap profesionalisme dalam menjalankan perannya sebagai penilai/surveyor yang melayani semua klien dari seluruh Indonesia dan mentaati hukum-hukum KEPI (Kode Etik Penilai Indonesia).
2. Huruf G yang berwarna kuning yang artinya GUNAWAN yaitu melambangkan bahwa KJPP GUNAWAN ini mampu mencetak pribadi-pribadi penilai/surveyor yang berwawasan dan berintelektual tinggi di bidang penilaian publik untuk dapat bersaing dengan penilai/surveyor di seluruh Indonesia untuk menjadi yang terdepan sebagai penilai/surveyor Indonesia.

2.1.4 Badan Hukum

Berdasarkan Keputusan Menteri Keuangan No. 406/KMK.06/2004 dan Peratutan Menteri Keuangan No. 125/PMK.01/2008 maka didirikanlah Kantor Jasa Penilai Publik yang berbadan hukum perseorangan atau persekutuan perdata yaitu Kantor Jasa Penilai Publik (KJPP) Gunawan pada tanggal 1 Desember 2008.

2.1.5 Visi, dan Misi

Visi

Kualitas Tinggi dari Appraisals terpercaya dan Jasa Konsultasi Properti

Misi

- Tumbuh bersama-sama dengan semua Stakeholder kami
- Sumber daya manusia pemberdayaan
- Integritas dan peningkatan profesionalisme
- Pasar dan orientasi pelanggan

2.2 Landasan Teori

Landasan teori membahas berbagai konsep dasar dan teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang telah dilakukan sebelumnya.

2.2.1 Sistem Pendukung Pengambilan Keputusan (SPPK)

Pada dasarnya pengambilan keputusan adalah suatu pendekatan sistematis pada hakekat suatu masalah, pengumpulan fakta-fakta, penentuan yang matang dari alternatif yang dihadapi, dan pengambilan tindakan yang menurut perhitungan merupakan tindakan yang paling tepat.

2.2.2 Appraisal

Appraisal atau penilaian yang didefinisikan sebagai proses mengestimasi nilai. Penilaian meliputi deskripsi atas property yang dinilai, opini penilai tentang kondisi properti yang bertujuan untuk mendapatkan nilai finansial properti berdasar data pasar pada tahun yang relatif baru. Dalam proses penilaian diperlukan kepiwaian penilai dalam mempergunakan berbagai pendekatan yang tepat untuk menghasilkan nilai yang akurat berdasar berbagai data faktual di lapangan. Sekalipun *Appraisal* didasarkan pada data lapangan , penilaian tetap merupakan opini.

2.2.3 Standar Penilai Indonesia (SPI)

Standar penilai Indonesia adalah sebuah pedoman yang dasar pelaksanaan tugas penilaian secara professional yang sangat penting arinya bagi para penilai untuk menghasilkan kajian berupa analisa, pendapat, dan saran-saran dengan menyajikannya dalam bentuk laporan penilaian sehingga tidak terjadi salah tafsir bagi para pengguna jasa dan masyarakat pada umumnya. Dalam pelaksanaan tugas penilaian, penilai menerapkan nilai pasar yang di tetapkan oleh SPI sendiri, tetapi apabila melakukan penyimpangan, penilai yang bersangkutan menjelaskan pertimbangannya secara rinci dalam laporan penilaian.

2.2.4 Basis Data (*Database*)

Basis data merupakan kumpulan dari item data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu.[2]

2.2.4.1 Tujuan Basis Data

1. Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi,
2. Menentukan kualitas informasi : akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya,
3. Mengurangi duplikasi data (*data redudancy*),
4. Hubungan data dapat ditingkatkan (*data relatability*),
5. Mengurangi pemborosan tempat simpanan luar.

2.2.4.2 Tahap Perancangan Basis Data

Perencanaan *database* harus terintegrasi dengan strategi dari sistem informasi. Tiga faktor utama yang harus diperhatikan dalam menentukan strategi sistem informasi :[2]

1. Apa yang harus dikerjakan
2. Sumber daya yang diperlukan untuk melaksanakan misi
3. Dana yang dibutuhkan

2.2.5 Kualitas Sistem (*System Quality*)

Kualitas sistem biasanya berfokus pada karakteristik kinerja sistem. Menurut DeLone dan McLean dalam Livari (2005) kualitas sistem merupakan sistem ciri karakteristik kualitas yang diinginkan dari sistem informasi itu sendiri, dan kualitas informasi yang diinginkan informasi karakteristik produk. Kualitas sistem ini juga berarti kombinasi *hardware* dan *software* dalam sistem informasi (DeLone dan McLean, 1992) [7]

Kualitas sistem memerlukan indikator untuk dapat mengukur seberapa besar kualitas dari sistem. Indikator diperlukan karena kualitas sistem merupakan variabel laten yang tidak dapat diukur secara langsung. Indikator kualitas sistem diwujudkan dalam seperangkat pertanyaan kualitas sistem yang dapat diukur melalui beberapa indikator sebagai berikut.

1. ***Ease of use*** **(Kemudahan Penggunaan)**

Suatu sistem informasi dapat dikatakan berkualitas jika system tersebut dirancang untuk memenuhi kepuasan pengguna melalui kemudahan dalam menggunakan sistem informasi tersebut. Kemudahan penggunaan dalam konteks ini bukan saja kemudahan untuk mempelajari dan menggunakan suatu system, tetapi juga mengacu pada kemudahan dalam melakukan suatu pekerjaan, atau tugas dimana pemakaian suatu sistem akan semakin memudahkan seseorang dalam bekerja dibanding mengerjakan secara manual. Pengguna sistem informasi mempercayai bahwa sistem informasi yang lebih fleksibel, mudah dipahami dan mudah pengoperasiannya sebagai karakteristik kemudahan penggunaan.

2. ***Response Time*** **(Kecepatan Akses)**

Kecepatan akses merupakan salah satu indikator kualitas system informasi. Jika akses sistem informasi memiliki kecepatan yang optimal maka layak dikatakan bahwa sistem informasi yang diterapkan memiliki kualitas yang baik. Kecepatan akses akan meningkatkan kepuasan pengguna dalam menggunakan sistem informasi.

3. *Reliability* (Keandalan Sistem)

Sistem informasi yang berkualitas adalah sistem informasi yang dapat diandalkan. Jika sistem tersebut dapat diandalkan maka sistem informasitersebut layak digunakan. Keandalan sistem informasi dalam konteks ini adalah ketahanan sistem informasi dari kerusakan dan kesalahan. Keandalan sistem informasi ini juga dapat dilihat dari sistem informasi yang melayani kebutuhan pengguna tanpa adanya masalah yang dapat mengganggu kenyamanan pengguna dalam menggunakan sistem informasi.

4. *Flexibility* (fleksibilitas)

Fleksibilitas suatu sistem informasi menunjukkan bahwa system informasi yang diterapkan tersebut memiliki kualitas yang baik. Fleksibilitas yang dimaksud adalah kemampuan sistem informasi dalam melakukan perubahan-perubahan kaitannya dengan memenuhi kebutuhan pengguna. Pengguna akan merasa lebih puas menggunakan suatu sistem informasi jika sistem tersebut fleksibel dalam memenuhi kebutuhan pengguna.

5. *Security* (keamanan)

Suatu sistem informasi dapat dikatakan baik jika keamanan system tersebut dapat diandalkan. Keamanan sistem ini dapat dilihat melalui data pengguna yang aman disimpan oleh suatu sistem informasi. Data pengguna ini harus terjaga kerahasiaannya dengan cara data disimpan oleh sistem informasi sehingga pihak lain tidak dapat mengakses data pengguna secara bebas.

2.2.6 Kualitas Informasi (*Information Quality*)

Kualitas informasi berfokus pada informasi yang dihasilkan oleh system informasi. Kriteria yang dapat digunakan untuk menilai kualitas informasi antara lain adalah kelengkapan, keakuratan, ketepatan waktu, ketersediaan, relevansi, konsistensi, dan *data entry*. Menurut Shannon dan Weaver dalam DeLone dan McLean (2003), Kualitas informasi dalam suatu sistem informasi menunjukkan sebagai kesuksesan semantik. Level semantik ini merupakan kesuksesan informasi dalam menyampaikan maksud atau arti yang diharapkan. Kualitas informasi juga berarti menentukan kesuksesan desain dari suatu *website.* Berikut ini merupakan indikatorindikator dari kualitas informasi.

1. ***Completeness*** **(Kelengkapan)**

Suatu informasi yang dihasilkan oleh sistem informasi dapat dikatakan berkualitas jika informasi yang dihasilkan lengkap. Informasi yang lengkap ini sangat dibutuhkan oleh pengguna dalam pengambilan keputusan. Informasi yang lengkap ini mencakup seluruh informasi yang dibutuhkan oleh pengguna dalam menggunakan sistem informasi tersebut. Jika informasi yang tersedia dalam sistem informasi lengkap maka akan memuaskan pengguna. Pengguna mungkin akan menggunakan sistem informasi tersebut secara berkala setelah merasa puas terhadap sistem informasi tersebut.

2. ***Relevance*** **(Relevan)**

Kualitas informasi suatu sistem informasi dikatakan baik jika relevan terhadap kebutuhan pengguna atau dengan kata lain informasi tersebutmempunyai manfaat untuk penggunanya. Relevansi informasi untuk tiap-tiap pengguna satu dengan yang lainnya berbeda sesuai dengan kebutuhan.

3. ***Accurate*** **(Akurat)**

Menurut DeLone dan McLean (1992), informasi yang dihasilkan oleh sistem informasi harus akurat karena sangat berperan bagi pengambilan keputusan penggunanya. Informasi yang akurat berarti harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksud informasi yang disediakan oleh sistem informasi. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.

4. ***Timeliness*** **(Ketepatan Waktu)**

Informasi yang datang pada penerima tidak boleh terlambat, informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan didalam pengambilan keputusan. Jika pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi sebagai pengguna suatu sistem informasi tersebut. Oleh karena itu dapat dikatakan bahwa kualitas informasi yang dihasilkan sistem informasi baik jika informasi yang dihasilkan tepat waktu.

5. *Format* (Penyajian informasi)

Format sistem yang memudahkan pengguna untuk memahami informasi yang disediakan oleh sistem informasi mencerminkan kualitas informasi yang baik. Jika penyajian informasi disajikan dalam bentuk yang tepat maka informasi yang dihasilkan dianggap berkualitas sehingga memudahkan pengguna untuk memahami informasi yang dihasilkan oleh suatu sistem informasi.

2.2.7 Sistem Informasi

Sistem informasi adalah:[2]

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

2.2.7.1 Komponen Sistem Informasi

Terdapat 5 komponen pada Sistem Informasi yang diklasifikasikan sebagai berikut :[2]

1. *Hardware* dan *Software* yang berfungsi sebagai mesin.
2. *People* dan *Procedures* yang merupakan manusia dan tatacara menggunakan mesin.
3. Data merupakan jembatan penghubung antara manusia dan mesin agar terjadi suatu proses pengolahan data.

2.2.7.2 Kegiatan Sistem Informasi

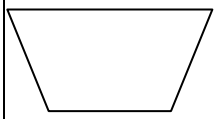
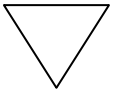
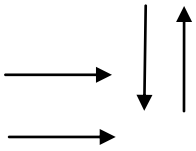
1. *Input* : Menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.
2. *Proses* : Menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
3. *Output* : Suatu kegiatan untuk menghasilkan laporan dari proses diatas tersebut.
4. Penyimpanan : Suatu kegiatan untuk memelihara dan menyimpan data.
5. *Control* : Suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

2.2.8 Flow Map (Bagan Alir)

Flowmap atau bagan alir adalah bagan yang menunjukan aliran di dalam program atau prosedur sistem secara logika. *Flowmap* ini berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *flowmap* ini harus dapat memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukan dengan jelas.
3. Harus ditunjukan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan didalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan didalam flow map harus didalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol simbol *flowmap* yang standar.

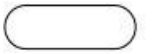
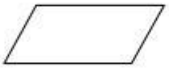
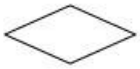
Tabel 2.1 Simbol Flowmap

No	Simbol	Keterangan
1		Dokumen Menunjukan dokumen sebagai masukan/keluaran baik secara manual/melalui komputer
2		Proses Manual Menunjukan proses yang dikerjakan secara manual
3		Penyimpanan Dokumen Digunakan untuk menyimpan data sebagai arsip secara manual
4		Aliran Data Menunjukan aliran data antarproses

Flowchart (Diagram Alur)

Flowchart adalah representasi grafik dari langkah-langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri atas sekumpulan simbol, dimana masing-masing simbol merepresentasikan suatu kegiatan tertentu. Flowchart diawali dengan penerimaan input, pemrosesan input, dan diakhiri dengan penampilan output.

Tabel 2.2 Simbol Flowchart

SIMBOL	NAMA	FUNGSI
	**TERMINATOR**	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	**PREPARATION**	Proses inisialisasi/pemberian harga awal
	**PROSES**	Proses perhitungan/proses pengolahan data
	**INPUT/OUTPUT DATA**	Proses input/output data, parameter, informasi
	**DECISION**	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	**ON PAGE CONNECTOR**	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	**OFF PAGE CONNECTOR**	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

2.2.9 Konsep Perancangan Basis Data

Perancangan basis data sangat diperlukan dalam membangun suatu sistem agar sistem yang akan dingangun tersebut sesuai dengan yang kita inginkan, dimana dapat melalui tahapan berikut.[3]

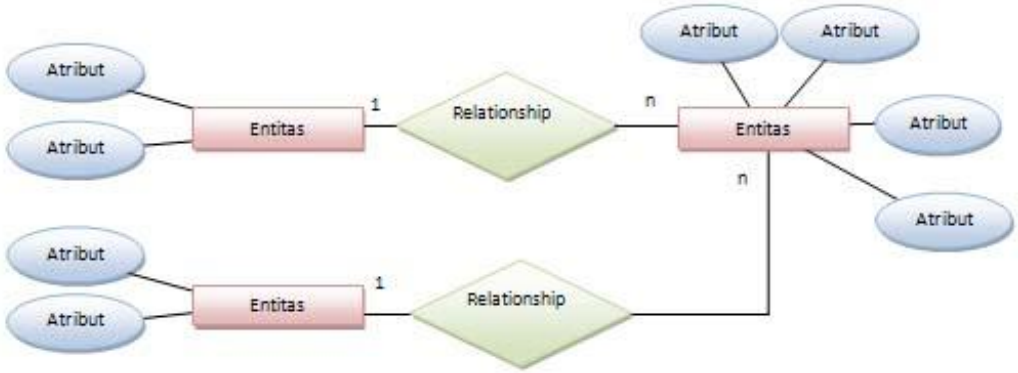
2.2.9.1 Entity Relationship Diagram (ERD)

Dalam rekayasa perangkat lunak, sebuah *Entity-Relationship Model* (ERM) merupakan abstrak dan konseptual representasi data. *Entity-Relationship* adalah salah satu metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual untuk jenis/model data semantik sistem. Dimana sistem seringkali memiliki basis data relasional, dan ketentuannya bersifat *top-*

down. Diagram untuk menggambarkan model *Entitiy-Relationship* ini disebut *Entitiy-Relationship diagram, ER diagram,* atau *ERD*.[3]

2.2.9.1.1 Notasi ERD

Ada sejumlah konvensi mengenai notasi ERD. Notasi klasik sering digunakan untuk model konseptual. Berbagai notasi lain juga digunakan untuk menggambarkan secara logis dan fisik dari suatu basis data, salah satunya adalah *IDEF1X*.[3]



Gambar 2.3 Model ERD

Notasi-notasi simbolik yang digunakan dalam *Entity Relationship Diagram* adalah sebagai berikut :[3]

1. *Entitas,* Adalah segala sesuatu yang dapat digambarkan oleh data. Entitas juga dapat diartikan sebagai individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain (Fathansyah, 1999). Ada dua macam entitas yaitu entitas kuat dan entitas lemah. Entitas kuat merupakan entitas yang tidak memiliki ketergantungan dengan entitas lainnya. Contohnya entitas anggota. Sedangkan entitas lemah merupakan entitas yang kemunculannya tergantung pada keberadaaan entitas lain dalam suatu relasi.
2. *Atribut,* Atribut merupakan pendeskripsian karakteristik dari entitas. Atribut digambarkan dalam bentuk lingkaran atau elips. Atribut yang menjadi kunci entitas atau key diberi garis bawah.

3. *Relasi atau Hubungan,* Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.
4. *Penghubung antara himpunan relasi* dengan himpunan entitas dan himpunan entitas dengan atribut dinyatakan dalam bentuk garis.

2.2.9.1.2 Derajat Relasi atau Kardinalitas

Menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Macam-macam kardinalitas adalah:[3]

1. Satu ke satu *(one to one),* Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.
2. Satu ke banyak *(one to many)*, Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.
3. Banyak ke banyak (*many to many),* Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

2.2.10 Konsep Perancangan Sistem Informasi

Desain sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang harus diselesaikan, tahap ini menyangkut mengkonfigurasikan dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar benar memuaskan rancang bangun yang telah ditetapkan pada akhir analisis sistem. Berikut ini akan dijelaskan rangkaian atau ruang lingkup sistem yang akan dirancang dengan memanfaatkan alat bantu sebagai berikut.[3]

2.2.10.1 Diagram Aliran Data / Data Flow Diagram (DFD)

DFD merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan DFD adalah memudahkkan pemakai atau user yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan.[3]

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Sistem dibatasi oleh *boundary* (dapat digambarkan dengan garis putus). Dalam diagram konteks hanya ada satu proses. Tidak boleh ada store dalam diagram konteks. Diagram konteks berisi gambaran umum (secara garis besar) sistem yang akan dibuat. Secara kalimat, dapat dikatakan bahwa diagram konteks ini berisi "siapa saja yang memberi data (dan data apa saja) ke sistem, serta kepada siapa saja informasi (dan informasi apa saja) yang harus dihasilkan sistem."[3]

2. Diagram Nol/*Zero* (*Overview Diagram*)

Diagram nol adalah diagram yang menggambarkan proses dari *data flow diagram*. Diagram nol memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan *eksternal entity*. Pada level ini sudah dimungkinkan adanya *data store* yang digunakan. Untuk proses yang tidak dirinci lagi pada level selanjutnya, symbol '*' dapat ditambahkan pada akhir nomor proses.[3]

3. Diagram Rinci (*Level Diagram*)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram *zero* atau diagram level diatasnya.[3]

2.2.10.1.1 Tujuan DFD

Tujuan DFD adalah :

1. Memberikan indikasi mengenai bagaimana data ditransformasi pada saat data bergerak melalui sistem.
2. Menggambarkan fungsi-fungsi (dan sub fungsi) yang mentransformasi aliran data.

2.2.10.1.2 Manfaat DFD

1. *Data Flow Diagram* (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi.
2. DFD ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem.Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem.
3. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

2.2.10.1.3 Simbol DFD

1. Terminator/Kesatuan luar (*External Entity*)

Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan membeikan input atau menerima *output* dari sistem (Jogiyanto, 1989).

Suatu kesatuan luar dapat disimbolkan dengan suatu notasi kotak.



Gambar 2.4 Notasi Terminator/Kesatuan Luar di DFD

Terminator dapat berupa orang, sekelompok orang, organisasi, departemen di dalam organisasi, atau perusahaan yang sama tetapi di luar kendali sistem yang sedang dibuat modelnya. Terminator dapat juga berupa departemen, divisi atau sistem di luar sistem yang berkomunikasi dengan sistem yang sedang dikembangkan.

2. Arus data (*Data flow*)

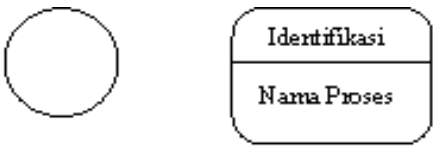
Arus data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus data yang dapat berupa masukkan untuk sistem atau hasil dari proses sistem.[3]

Arus Arus data dapat berbentuk sebagai berikut.

a. Formulir atau atau dokumen dokumen yang yang digunakan digunakan perusahaan perusahaan
b. Laporan tercetak tercetak yang yang dihasilkan dihasilkan sistem sistem
c. *Output* dilayar komputer
d. Masukan untuk komputer komputer
e. Komunikasi ucapan
f. Surat atau memo
g. Data yang dibaca atau atau direkam di *file*
h. Suatu isian yang yang dicatat pada buku agenda
i. Transmisi data dari suatu komputer ke komputer lain

3. Proses (*Proccess*)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin, atau komputer dan hasil suatu arus data yang masuk ke dalam proses untuk dilakukan arus data yang akan keluar dari proses. Suatu proses dapat ditunjukkan dengan simbol lingkaran atau dengan simbol empat persegi panjang tegak dengan sudut-sudutnya tumpul.[3]



Gambar 2.5 Notasi Proses di DFD

Ada beberapa hal yang perlu diperhatikan tentang proses :

a. Proses harus memiliki input dan output.
b. Proses dapat dihubungkan dengan komponen terminator, data store atau proses melalui alur data.
c. Sistem/bagian/divisi/departemen yang sedang dianalisis oleh profesional sistem digambarkan dengan komponen proses.

4. Simpanan Data (*Data store*)

Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa *file* atau *database* di sistem komputer, arsip atau catatan manual, kotak tempat data di meja seseorang, tabel acuan manual, agenda atau buku. Simpanan data di DFD dapat disimbolkan dengan sepasang garis horizontal paralel yang tertutup di salah satu ujungnya.[3]



Gambar 2.6 Simbol dari Simpanan Data di DFD

2.2.10.1.4 Syarat Memuat DFD

Syarat-syarat pembuatan DFD ini adalah :

1. Pemberian nama untuk tiap komponen DFD
2. Pemberian nomor pada komponen proses
3. Penggambaran DFD sesering mungkin agar enak dilihat
4. Penghindaran penggambaran DFD yang rumit
5. Pemastian DFD yang dibentuk itu konsiten secara logika

2.2.10.2 Kamus Data

Kamus data berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan secara detail dan mengorganisasi semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. [3]

Kamus data sering disebut juga dengan sistem data dictionary adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analisis sistem adapat mendefinisikan data yang mengalir di sistem dengan lengkap. Pada tahap analisis, kamus data digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir dari sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database*.

2.2.11 Database Management System (DBMS)

Database manajemen sistem (DBMS) merupakan pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak atau sistem yang khusus atau spesifik. Sistem ini yang akan menentukan bagaimana data akan diorganisasikan, disimpan, diubah dan diambil kembali. Disamping itu, sistem ini juga menerapkan mekanisme pengamanan data, pemakaian dan secara bersama, pemaksaan keakuratan atau konsistensi data dan sebagainya. Perangkat lunak yang termasuk DBMS misalnya *dBase II+, dBaseIV, FoxBase, Rbase, MS-Access, dan Borland Paradox atau Borand Interbase, MS-SQL Server, CA-Open Ingres, Oracle, Informix*, dan *Sybase.*

2.2.11.1 Fitur-fitur Database Management System (DBMS)

Pada dasarnya DBMS mampu menyediakan berbagai fitur-fitur untuk memudahkan dalam pemrograman, diantaranya adalah sebagai berikut :

1. *Independensi data – program*

   Karena basis data ditangani oleh DBMS, program dapat ditulis, sehingga tidak bergantung pada stuktur data dalam basis data. Dengan kata lain program tidak akan terpengaruh sekiranya bentuk fisik data diubah.

2. *Keamanan*

   Keamanan dimaksudkan untuk mencegah pengaksesan data oleh orang yang tidak berwenang.

3. *Integritas*

   Hal ini ditujukan untuk menjaga agar data selalu dalam keadaan yang valid dan konsisten.

4. *Konkurensi*

   Konkurensi memungkinkan data dapat diakses oleh banyak pemakai tanpa menimbulkan masalah.

5. Pemulihan (*recovery*)

   DBMS menyediakan mekanisme untuk mengembalikan basis data ke keadaan semula yang konsisten sekiranya terjadi gangguan perangkat keras atau kegagalan perangkat lunak.

6. Katalog sistem.
7. Perangkat produktivitas.

2.2.11.2 Keunggulan Database Management System (DBMS)

Selain memiliki fitur-fitur, DBMS juga memiliki keunggulan, di bawah ini adalah keunggulan dari DBMS :

1. Mengendalikan/mengurangi duplikasi data.
2. Menjaga konsistensi dan integritas data.
3. Memudahkan pemerolehan informasi yang lebih banyak dari data yang sama disebabkan data dari berbagai bagian dalam organisasi dikumpulkan menjadi satu.
4. Meningkatkan keamanan data dari orang yang tidak berwenang.
5. Memaksakan penerapan standar.
6. Dapat menghemat biaya karena data dapat dipakai oleh banyak departemen.
7. Menanggulangi konflik kebutuhan antarpemakai karena basis data di bawah kontrol administrator basis data.
8. Meningkatkan tingkat respon dan kemudahan akses bagi pemakai akhir.
9. Meningkatkan produktivitas pemrograman.
10. Meningkatkan pemeliharaan melalui independensi data.

11. Meningkatkan konkurensi (pemakai data oleh sejumlah data) tanpa menimbulkan masalah kehilangan informasi atau integritas.
12. Meningkatkan layanan *back up* dan *recovery*.

2.2.11.3 Kelemahan Database Management System (DBMS)

DBMS selain memiliki keunggulan, juga memiliki kelemahan. Berikut adalah kelemahan dari DBMS :

1. Kompleksitas yang tinggi membuat administrator dan pemakai akhir harus benar-benar memahami fungsi-fungsi dalam DBMS agar diperoleh manfaat yang optimal. Kegagalan memahami DBMS dapat mengakibatkan keputusan rancangan yang salah, yang akan memberikan dampak serius bagi organisasi.
2. Ukuran penyimpanan yang dibutuhkan oleh DBMS sangat besar dan memerlukan memori yang besar agar bisa bekerja secara efisien.
3. Rata-rata harga DBMS yang handal sangat mahal.
4. Terkadang DBMS meminta kebutuhan perangkat keras dengan spesifikasi tertentu, sehingga diperlukan biaya tambahan.
5. Biaya konversi sistem lama (yang mencakup biaya pelatihan staf dan biaya untuk jasa konversi) ke sistem baru yang memakai DBMS terkadang sangat mahal melebihi biaya untuk membeli DBMS.
6. Kinerja terkadang kalah dengan sistem yang berbasis berkas. Hal ini bisa dipahami, karena DBMS ditulis supaya dapat menangani hal-hal yang bersifat umum.
7. Dampak kegagalan menjadi lebih tinggi karena semua pemakai sangat bergantung pada ketersediaan DBMS. Akibatnya, kalau terjadi kegagalan dalam komponen lingkungan DBMS akan membuat operasi dalam organisasi tersendat atau bahkan terhenti.

Administrator basis data (DBA atau *database administrator*) adalah orang yang bertanggung jawab terhadap manajemen basis data. Secara lebih detail, DBA bertugas sebagai :

1. Mendefinisikan basis data.
2. Mendefinisikan pemeliharaan basis data secara rutin.
3. Menentukan keamanan basis data.

Setiap pemakai diberi hak akses terhadap basis data secara tersendiri. Tidak semua bisa menggunakan data yang bersifat sensitif. Penentuan hak akses disesuaikan dengan wewenang pemakai dalam organisasi.

2.2.11.11 Borland Delphi 7

Borland delphi merupakan suatu bahasa pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrograman yang terstruktur. Keunggulan lain delphi adalah dapat dipergunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis windows.[4]

1. Komponen Delphi

Component palette terdiri dari beberapa komponen yang dapat dipilih yang digunakan untuk menangani beberapa tugas pemrograman. Komponen-komponen yang terletak pada bagian *component palette* sudah ditata dalam beberapa tab yang masing-masing menunjukan maksud dan fungsi. Masing-masing tab ditampilkan dalam konfigurasi default yang semua juga tergantung pada versi delphi yang digunakan

2. Fitur Pada Delphi 7

Fitur baru dan perbaikan yang ada pada Borland Delphi 7 ini adalah :[5]

a. IDE (*Interface Development Environtment*)

Lingkungan pengembangan aplikasi (IDE) Borland Delphi 7 telah mengalami perubahan dari versi sebelumnya. Diantaranya, terdapat *Compiler Message*, perubahan pada *Component Pallete*, *Code Insight* dan *Debugger*.

b. Web

Borland Delphi 7 menyediakan Intraweb buatan AtoZed *Software*, yang dapat digunakan untuk membuat aplikasi *web server* dengan sarana standar. Borland Delphi 7 juga mendukung pada Apache 2. Borland juga menghilangkan Win-CGI sebagai target aplikasi web server dan web service. Fasilitas untuk *Web server* juga mengalami perbaikan – perbaikan.

c. COM

Sekarang ini dengan Delphi 7, dapat membuat CoClass *wrapper* bagi pengembangan – pengembangan .NET dengan cara menggunakan kotak dialog *Import Type Library*. Dengan adanya *resulting wrapper*, maka fitur interoperabilitas dari Microsoft's .NET Framework dapat digunakan.

d. *Database*

Pada Delphi 7 driver dbExpress telah diupdate bagi Informix SE, Oracle 9i, DB2 7.2, InterBase 6.5, dan MySQL 3.23.49. Driver baru bagi 30 MSSQL 2000 juga tersedia. Disamping itu, beberapa hal baru dan perubahan juga dilakukan pada komponen *database*. Borland juga telah membuang SQL Links. Borland merekomendasikan pemakaian dbExpress bagi database SQL Server yang diakses di Delphi.

e. *Component Library*

Jika ditelusuri komponen librari Delphi 7, maka akan ditemukan komponen baru, unit baru, komponen yang berubah, komponen yang hilang dan komponen yang mendukung bagi tema Windows XP.

f. *Runtime Library*

Beberapa perubahan di *Runtime Library* antara lain ialah perubahan pada unit Classes, Math, StdConv,StrUtils, SysUtils, VarCmplx, dan Variants.

g. *Compiler*

Kompiler Delphi dcc32 sekarang ini *support* terhadap tiga *warning* kompiler tambahan, yaitu *Unsafe_Type*, *Unsafe_Code*, dan *Unsafe_Cast*. Warnings tersebut defaultnya adalah *disabled*, tetapi dapat di-*enabled*. Fitur ini

sangat membantu kita ketika akan mem-port kode ke lingkungan eksekusi terkendali di *platform* Microsoft's .NET.

h. *Model Maker*

Sarana baru yang disebut *ModelMaker* dapat membantu memudahkan proses desain, konstruksi, dan pengelolaan *class* dan *interface*. *Model Maker* juga memiliki sarana untuk pembuatan diagram UML-*style*, yang dapat dipakai untuk membuat dan memodifikasi *source code project.*

2.12 MYSQL

MySQL adalah suatu sistem manajemen *database*. Suatu *database* adalah sebuah kumpulan data yang terstruktur, untuk menambahkan, mengakses, dan memproses data yang tersimpan pada suatu *database* komputer anda memerlukan sistem manajemen *database* seperti *MySQL*. Karena komputer sangat unggul dalam menangani sejumlah besar data, sistem manajemen *database* memainkan suatu peranan yang penting dalam komputasi, baik sebagai *utility stand-alone* maupun bagian dari aplikasi lainnya.

Suatu database relasional menyimpan data dalam tabel-tabel terpisah. Hal ini memungkinkan kecepatan dan fleksibilitas. Tabel-tabel yang dihubungkan dengan relasi yang ditentukan membuatnya bisa mengkombinasikan data dari beberapa tabel pada suatu permintaan. Bagian *SQL* dari kata *MYSQL* berasal dari "*Structured Query Language*" bahasa paling umum yang dipergunakan untuk mengakses *database*, [4].

Beberapa perintah dasar SQL yang sering dipergunakan pada MySQL adalah sebagai berikut :[6]

1. Create Database

Yaitu perintah yang digunakan untuk membuat *database* baru.

Sintaks : `Create database database_nama database`

2. Drop Database

Yaitu perintah yang digunakan untuk menghapus *database*.

Sintaks : `Drop Tabel Tabel_name`

3. Create Tabel

Yaitu perintah yang digunakan untuk membuat tabel baru.

Sintaks : `Create Tabel tabel_name (create_definition)`

4. Describe

Yaitu perintah yang digunakan untuk mendeskripsikan tabel atau logam

Sintaks : `Describe (Desc) tabel [colum].`

5. Alter Tabel

Yaitu perintah yang digunakan untuk memodifikasi tabel

Sintaks : `Alter [Ignor] Tabel table_name`

6. Drop Tabel

Yaitu perintah yang digunakan untuk menghapus tabel

Sintaks : `Drop Tabel tabel_name [tabel_name..]`

7. Delete

Yaitu perintah yang digunakan untuk menghapus record dri tabel

Sintaks : `Delete From tabel_name Where Where_definiition`

8. Select

Yaitu perintah yang digunakan untuk query ke database

Sintaks : `Select * from tabel_name.`

2.13 Pengertian *Client Server*

Semakin berkembangnya teknologi jaringan komputer, sekarang ini terdapat kecenderungan sebuah sistem yang menggunakan jaringan untuk saling berhubungan. Dalam jaringan tersebut, biasanya terdapat sebuah komputer yang disebut *server*, dan beberapa komputer yang disebut client, dalam pendistribusian suatu aplikasi.

Komponen *client* sering disebut juga sebagai *front-end*, sementara komponen *server* disebut sebagai *back-end*. Komponen client dari aplikasi dijalankan dalam sebuah *workstation* dan menerima masukan data dari pengguna. Komponen client tersebut akan menyiapkan data yang dimasukkan oleh pengguna

dengan menggunakan teknologi pemrosesan tertentu dan mengirimkannya kepada komponen *server* yang dijalankan di atas mesin *server*, umumnya dalam bentuk *request* terhadap beberapa layanan yang dimiliki oleh *server*. Komponen *server* akan menerima *request* dari *client*, dan langsung memprosesnya dan mengembalikan hasil pemrosesan tersebut kepada client. *Client* pun menerima informasi hasil pemrosesan data yang dilakukan *server* dan menampilkannya kepada pengguna, dengan menggunakan aplikasi yang berinteraksi dengan pengguna.

Atau dengan kata lain, *server* adalah komputer yang dapat memberikan *service* ke *server*, sedangkan *client* adalah komputer yang mengakses beberapa *service* yang ada di *client*. Ketika client membutuhkan suatu *service* yang ada di *server*, dia akan mengirim *request* kepada *server* lewat jaringan. Jika request tersebut dapat dilaksanakan, maka *server* akan mengirim balasan berupa *service*

1. Kelebihan model hubungan *Client Server* :
    a. Terpusat (Sumber daya dan keamanan data dikontrol melalui *server*).
    b. *Skalabilitas*.
    c. *Fleksibel*.
    d. Teknologi baru dengan mudah terintegrasi kedalam sistem.
    e. Keseluruhan komponen (*client/network/server*) dapat bekerja bersama.
2. Kekurangan model hubungan Client Server :
    a. Mahal.
    b. Membutuhkan investasi untuk *dedicated file server.*
    c. Perbaikan (Jaringan besar membutuhkan seorang staf untuk mengatur agar sistem berjalan secara *efisien*).
    d. Berketergantungan.

Ketika *server* jatuh, mengakibatkan keseluruhan operasi pada network akan jatuh pula.